



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ  
ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

***ΑΞΙΟΠΟΙΗΣΗ ΠΛΑΤΦΟΡΜΑΣ ΔΙΑΧΕΙΡΙΣΗΣ  
ΠΕΡΙΕΧΟΜΕΝΟΥ ΑΝΟΙΧΤΟΥ ΚΩΔΙΚΑ ΓΙΑ ΤΟ  
ΣΥΣΤΗΜΑ ΑΠΟΔΕΛΤΙΩΣΗΣ PERSONAL.***

Μπουρέλος Παναγιώτης  
ΑΜ 3446

*Υπεύθυνος Καθηγητής:*

**Χρήστος Μπούρας,  
Καθηγητής**

**ΠΑΤΡΑ,  
Φεβρουάριος 2012**

---

*Σε αυτούς που με στηρίζουν*

# ΠΡΟΛΟΓΟΣ

Χωρίς καμία αμφιβολία, η δημιουργία και αξιοποίηση του διαδικτύου έχει επιφέρει δραματικές αλλαγές σε δύο τεχνολογικούς τομείς που έχουν αποδειχθεί κυρίαρχοι στη σύγχρονη εποχή: την πληροφορική και τις τηλεπικοινωνίες. Το διαδίκτυο αποτελεί το καλύτερο μέσο για την εύρεση αλλά και τη διάδοση πληροφορίας. Μπορεί να χαρακτηριστεί σαν μια αχανή βιβλιοθήκη όπου δεν υπάρχουν κανόνες για τον τρόπο οργάνωσης του περιεχομένου της. Ο τεράστιος όγκος πληροφορίας που είναι διαθέσιμος στο Internet αποτελεί μια άριστη πηγή πληροφόρησης, η οποία όμως χωρίς τα κατάλληλα εργαλεία θα ήταν πολύ δύσκολο να αξιοποιηθεί επαρκώς από τους χρήστες.

Οι μηχανές αναζήτησης, οι οποίες δημιουργούν ευρετήρια μεγάλων τμημάτων του διαδικτύου, αποτελούν κυρίαρχο εργαλείο στην αναζήτηση πληροφορίας από το διαδίκτυο. Ωστόσο η ολοένα και αυξανόμενη πληροφορία στο διαδίκτυο δεν συντελεί στην εξόρυξη της κατάλληλης πληροφορίας, με αποτέλεσμα να απαιτείται ένα είδος φιλτραρίσματος ώστε να παρέχονται ποιοτικά αποτελέσματα στο χρήστη.

Η πληροφόρηση μέσω του διαδικτύου φαίνεται να κερδίζει όλο και περισσότερο έδαφος με κύριο σύμμαχο τις νεότερες ηλικίες που έχουν στενότερη επαφή με τις καινούργιες τεχνολογίες. Οι ειδησεογραφικές πύλες θα πρέπει να είναι προσεκτικές ως προς την πληροφορία που διαδίδουν καθώς το περιεχόμενό τους αναγιγνώσκεται από διάφορες ηλικίες και η προσωποποίηση του με βάση τον εκάστοτε χρήστη κρίνεται επιτακτική. Ο χρόνος αναζήτησης που θα πρέπει να σπαταλάται από το χρήστη για την ενημέρωσή του από ειδησεογραφικά δελτία θα πρέπει να είναι ελάχιστος και σε αυτό τον τομέα αρκετές τεχνικές έχουν δώσει σημαντικά αποτελέσματα. Το σύστημα *reRSSonal* αποτελεί το σημαντικότερο ίσως εργαλείο για την επίτευξη της εξατομίκευσης τέτοιου είδους διαδικτυακής πληροφορίας το οποίο αξιοποιώντας κατάλληλους μηχανισμούς αποδίδει το καλύτερο δυνατό αποτέλεσμα.

Στην παρούσα διπλωματική εργασία θα μελετήσουμε την αρχιτεκτονική και τον τρόπο λειτουργίας του συστήματος αυτού, αλλά κύριο μέλημά μας είναι η αξιοποίηση κάποιας πλατφόρμας ανοιχτού κώδικα για τη μεταφορά του συστήματος *reRSSonal* σε αυτή, δίνοντας ιδιαίτερη έμφαση στα πλεονεκτήματα που προκύπτουν από τη χρήση τέτοιου είδους λογισμικού και από την εξέλιξη που μπορεί να επιφέρει στο υπάρχον σύστημα.

Με αυτόν τον πρόλογο θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου Χρήστο Μπούρα (Καθηγητής του τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών) για την επίβλεψη της συγκεκριμένης διπλωματικής εργασίας. Ακόμα θα ήθελα να ευχαριστήσω τον αγαπητό συνάδελφο και μεταπτυχιακό φοιτητή Γεώργιο Αδάμ για την βοήθεια και την καθοδήγηση στην εκπόνηση αυτής της διπλωματικής εργασίας.

Πάτρα, Φεβρουάριος 2012,  
Μπουρέλος Παναγιώτης

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΠΡΟΛΟΓΟΣ.....</b>	<b>3</b>
<b>ΠΕΡΙΕΧΟΜΕΝΑ.....</b>	<b>4</b>
<b>ΛΙΣΤΑ ΠΙΝΑΚΩΝ.....</b>	<b>8</b>
<b>ΛΙΣΤΑ ΕΙΚΟΝΩΝ.....</b>	<b>10</b>
<b>ΓΛΩΣΣΑΡΙΟ.....</b>	<b>13</b>
<b>ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ.....</b>	<b>15</b>
<b>1 ΕΙΣΑΓΩΓΗ.....</b>	<b>16</b>
1.1 Προσδιορισμός του Προβλήματος.....	16
1.2 Δομή της Εργασίας.....	17
<b>2 ΘΕΜΑΤΑ ΠΟΥ ΘΑ ΜΑΣ ΑΠΑΣΧΟΛΗΣΟΥΝ.....</b>	<b>18</b>
2.1 Η πληροφορία στο διαδίκτυο και η γλώσσα HTML.....	18
2.1.1 Η Δομή μιας HTML Σελίδας.....	19
2.2 Η τεχνολογία XML.....	20
2.3 Τι είναι το RSS.....	20
2.4 Οι ιστοσελίδες από την αρχή της εμφάνισής τους.....	21
2.5 Τι είναι τα CSS.....	22
2.6 Δυναμικές ιστοσελίδες με χρήση PHP/MYSQL.....	23
<b>3 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΚΑΙ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ.....</b>	<b>26</b>
3.1 Στόχοι του Συστήματος.....	26
3.2 Γενική αρχιτεκτονική.....	27
3.3 Υποσυστήματα.....	27
3.3.1 ΣΥΛΛΟΓΗ ΚΑΙ ΕΞΑΓΩΓΗ ΧΡΗΣΙΜΗΣ ΠΛΗΡΟΦΟΡΙΑΣ.....	27
3.3.2 ΠΡΟΕΠΕΞΕΡΓΑΣΙΑ ΚΕΙΜΕΝΟΥ.....	28
3.3.3 ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ.....	29
3.3.4 ΕΞΑΓΩΓΗ ΠΕΡΙΛΗΨΗΣ.....	30
3.3.5 ΠΑΡΟΥΣΙΑΣΗ ΤΕΛΙΚΗΣ ΠΛΗΡΟΦΟΡΙΑΣ ΚΑΙ ΠΡΟΣΩΠΟΠΟΙΗΣΗ.....	31
3.4 Αρχιτεκτονική του υποσυστήματος συλλογής πληροφορίας.....	32
3.5 Αρχιτεκτονική του υποσυστήματος εξαγωγής περιεχομένου....	35
<b>4 ΤΟ WEB ΚΑΙ ΤΑ ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ.....</b>	<b>38</b>
4.1 Τι είναι τα συστήματα διαχείρισης περιεχομένου.....	38
4.1.1 ENTERPRISE CMS.....	38
4.1.2 COMPONENT CMS.....	38
4.2 Web CMS.....	39
4.2.1 ΚΛΕΙΣΤΟΥ ΚΩΔΙΚΑ CMS.....	40
4.2.2 ΑΝΟΙΧΤΟΥ ΚΩΔΙΚΑ CMS.....	40
4.3 Δημοτικότητα των CMS.....	42

<b>4.4 Αρχιτεκτονική των CMS.....</b>	<b>43</b>
4.4.1 ΤΟ ΥΠΟΣΥΣΤΗΜΑ ΣΥΛΛΟΓΗΣ.....	44
4.4.2 ΤΟ ΥΠΟΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ.....	45
4.4.3 ΤΟ ΥΠΟΣΥΣΤΗΜΑ ΔΗΜΟΣΙΕΥΣΗΣ.....	46
<b>5 ΤΟ JOOMLA.....</b>	<b>48</b>
<b>5.1 Η ιστορία του Joomla.....</b>	<b>48</b>
<b>5.2 Χαρακτηριστικά του Joomla.....</b>	<b>49</b>
<b>5.3 Αρχιτεκτονική του Joomla.....</b>	<b>49</b>
<b>5.4 Δομή του Joomla.....</b>	<b>50</b>
5.4.1 ΤΟ ΔΗΜΟΣΙΟ ΤΜΗΜΑ ΚΑΙ Η ΠΕΡΙΟΧΉ ΔΙΑΧΕΙΡΙΣΗΣ (FRONT-END ΚΑΙ BACK-END).....	51
5.4.2 ΤΟ ΠΕΡΙΕΧΌΜΕΝΟ.....	51
5.4.3 ΟΙ ΕΠΕΚΤΑΣΕΙΣ.....	51
5.4.4 ΤΑ ΣΥΣΤΑΤΙΚΑ.....	51
5.4.5 ΤΑ ΠΡΌΤΥΠΑ.....	51
5.4.6 ΤΑ ΠΡΌΣΘΕΤΑ (PLUG-INS).....	52
<b>5.5 Εργαλεία που απαιτούνται για το Joomla.....</b>	<b>52</b>
<b>5.6 Η μορφή του perSSonal πριν ενσωματωθεί στο Joomla.....</b>	<b>52</b>
<b>5.7 Βελτιώσεις στο σύστημα perSSonal κατά τη μεταφορά του στο Joomla.....</b>	<b>57</b>
5.7.1 ΒΕΛΤΙΩΣΕΙΣ ΣΤΗ ΔΙΑΔΙΚΑΣΪΑ ΕΓΓΡΑΦΉΣ ΤΟΥ PERSSONAL ΣΤΟ JOOMLA..	57
5.7.2 ΒΕΛΤΙΩΣΕΙΣ ΣΤΟ ΤΡΌΠΟ ΕΜΦΆΝΙΣΗΣ ΤΩΝ ΣΤΟΙΧΕΪΩΝ ΤΟΥ PERSSONAL ΣΤΟ JOOMLA.....	57
5.7.3 ΝΈΕΣ ΔΥΝΑΤΌΤΗΤΕΣ ΠΟΥ ΠΡΟΣΤΈΘΗΚΑΝ ΣΤΟ PERSSONAL ΚΑΤΆ ΤΗ ΜΕΤΑΦΟΡΆ ΤΟΥ ΣΤΟ JOOMLA.....	58
5.7.4 ΔΥΝΑΤΌΤΗΤΑ ΠΡΟΒΟΛΉΣ ΌΛΩΝ ΤΩΝ ΠΡΌΣΘΕΤΩΝ ΤΑΥΤΌΧΡΟΝΑ.....	58
5.7.5 ΠΡΟΣ ΤΑ ΠΪΣΩ ΣΥΜΒΑΤΌΤΗΤΑ ΤΟΥ ΝΈΟΥ PERSSONAL ΣΤΟ JOOMLA ΜΕ ΤΟ ΠΑΛΑΙΟ PERSSONAL.....	58
<b>6 ΤΑ ΣΥΣΤΑΤΙΚΑ ΚΑΙ ΠΡΟΣΘΕΤΑ ΣΤΟΙΧΕΙΑ ΠΟΥ ΔΗΜΙΟΥΡΓΉΘΗΚΑΝ ΣΤΟ JOOMLA ΌΣΤΕ ΝΑ ΥΠΟΣΤΗΡΙΞΕΙ ΤΟ PERSSONAL.....</b>	<b>59</b>
<b>6.1 Το MVC μοντέλο δημιουργίας του Joomla.....</b>	<b>59</b>
6.1.1 ΤΟ ΜΟΝΤΈΛΟ.....	59
6.1.2 Η ΕΜΦΆΝΙΣΗ.....	60
6.1.3 Ο ΕΛΕΓΚΤΉΣ.....	60
<b>6.2 Το συστατικό στοιχείο com_users του Joomla.....</b>	<b>60</b>
<b>6.3 Το component com_perSSonal του Joomla για να υποστηρίξει τις λειτουργίες του perSSonal.....</b>	<b>62</b>
<b>6.4 Το πρόσθετο στοιχείο mod_perSSonalcategories .....</b>	<b>62</b>
<b>6.5 Το πρόσθετο στοιχείο mod_perSSonallatestnews.....</b>	<b>63</b>
<b>6.6 Το πρόσθετο στοιχείο mod_perSSonalpopulararticles.....</b>	<b>63</b>
<b>6.7 Το πρόσθετο στοιχείο mod_perSSonalreaderspik.....</b>	<b>63</b>
<b>6.8 Το πρόσθετο στοιχείο mod_perSSonalsimilararticles.....</b>	<b>64</b>
<b>6.9 Το πρόσθετο στοιχείο mod_perSSonaltagging .....</b>	<b>64</b>

---

<b>6.10 Εγκατάσταση και ρύθμιση όλων των modules και components του peRSSonal στο Joomla.....</b>	<b>64</b>
<b>6.11 Η τελική μορφή του peRSSonal στο Joomla.....</b>	<b>68</b>
6.11.1 ΕΓΓΡΑΦΗ ΧΡΗΣΤΩΝ ΣΤΟ PERSSONAL ΤΟΥ JOOMLA.....	68
6.11.2 ΤΟ ΝΕΟ PERSSONAL ΣΤΟ JOOMLA.....	71
6.11.3 ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΤΟΥ ΔΙΑΧΕΙΡΙΣΤΗ ΤΟΥ PERSSONAL ΣΤΟ JOOMLA.....	73
<b>6.12 Συμπεράσματα.....</b>	<b>74</b>
<b>6.13 Μελλοντική δουλειά.....</b>	<b>75</b>
<b>APPENDIX.....</b>	<b>76</b>
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</b>	<b>163</b>



# ΛΙΣΤΑ ΠΙΝΑΚΩΝ

<b>ΠΙΝΑΚΑΣ 1. ΤΟ DEFAULT.PHP ΑΡΧΕΙΟΥ ΤΟΥ FRONT-END COM_USERS ΓΙΑ ΤΟ REGISTRATION ΤΟΥ ΧΡΗΣΤΗ ΩΣΤΕ ΝΑ ΥΠΟΣΤΗΡΙΞΕΙ ΤΟ PERSONAL</b> .....	<b>79</b>
<b>ΠΙΝΑΚΑΣ 2. ΤΟ ΑΡΧΕΙΟ GETRSSDATA.PHP.....</b>	<b>81</b>
<b>ΠΙΝΑΚΑΣ 3. ΤΟ ΑΡΧΕΙΟ GETRSSDATA.JS.....</b>	<b>82</b>
<b>ΠΙΝΑΚΑΣ 4. ΤΟ ΑΡΧΕΙΟ REGISTRATION.PHP ΤΟΥ ΜΟΝΤΕΛΟΥ ΤΟΥ COM_USERS.....</b>	<b>95</b>
<b>ΠΙΝΑΚΑΣ 5. ΤΟ ΑΡΧΕΙΟ PROFILE.PHP ΤΟΥ ΜΟΝΤΕΛΟΥ ΤΟΥ COM_USERS ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΟΥ ΠΡΟΦΙΛ ΤΩΝ ΧΡΗΣΤΩΝ.....</b>	<b>100</b>
<b>ΠΙΝΑΚΑΣ 6. ΤΟ ΑΡΧΕΙΟ USER.PHP ΠΟΥ ΥΠΟΣΤΗΡΙΖΕΙ ΤΗ ΔΙΑΓΡΑΦΗ, ΕΝΕΡΓΟΠΟΙΗΣΗ ΚΑΙ ΜΙΛΟΚΑΡΙΣΜΑ ΤΩΝ ΧΡΗΣΤΩΝ ΑΠΟ ΤΗ ΒΑΣΗ ΤΟΥ PERSONAL ΚΑΙ ΤΟΥ JOOMLA.....</b>	<b>110</b>
<b>ΠΙΝΑΚΑΣ 7. ΤΟ XML ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ ΤΟΥ COMPONENT COM_USERS .....</b>	<b>113</b>
<b>ΠΙΝΑΚΑΣ 8. ΤΟ ΑΡΧΕΙΟ PERSONAL.PHP ΤΟΥ COMPONENT COM_PERSONAL ΣΤΟ FRONT-END.....</b>	<b>125</b>
<b>ΠΙΝΑΚΑΣ 9. ΤΟ ΑΡΧΕΙΟ PERSONAL.HTML.PHP ΣΤΟ COMPONENT COM_PERSONAL ΤΟΥ FRONT-END.....</b>	<b>126</b>
<b>ΠΙΝΑΚΑΣ 10. ΤΟ ΑΡΧΕΙΟ REJECT.PHP.....</b>	<b>127</b>
<b>ΠΙΝΑΚΑΣ 11. ΤΟ ΑΡΧΕΙΟ SEQUENCE.JS.....</b>	<b>128</b>
<b>ΠΙΝΑΚΑΣ 12. ΤΟ ΑΡΧΕΙΟ GET_IMAGES.PHP.....</b>	<b>129</b>
<b>ΠΙΝΑΚΑΣ 13. ΤΟ ΑΡΧΕΙΟ IMAGES.JS.....</b>	<b>130</b>
<b>ΠΙΝΑΚΑΣ 14. ΤΟ ΑΡΧΕΙΟ PERSONAL.PHP ΤΟΥ BACK-END.....</b>	<b>134</b>
<b>ΠΙΝΑΚΑΣ 15. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ ΤΟΥ COMPONENTN COM_PERSONAL .....</b>	<b>135</b>
<b>ΠΙΝΑΚΑΣ 16. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALCATEGORIES.PHP.....</b>	<b>135</b>
<b>ΠΙΝΑΚΑΣ 17. ΤΟ ΑΡΧΕΙΟ HELPER.PHP.....</b>	<b>136</b>
<b>ΠΙΝΑΚΑΣ 18. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP.....</b>	<b>137</b>
<b>ΠΙΝΑΚΑΣ 19. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ MOD_PERSONALCATEGORIES.XML.....</b>	<b>138</b>
<b>ΠΙΝΑΚΑΣ 20. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALLATESTNEWS.PHP.....</b>	<b>138</b>
<b>ΠΙΝΑΚΑΣ 21. ΤΟ ΑΡΧΕΙΟ HELPER.PHP.....</b>	<b>140</b>
<b>ΠΙΝΑΚΑΣ 22. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP.....</b>	<b>141</b>
<b>ΠΙΝΑΚΑΣ 23. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ MOD_PERSONALLATESTNEWS.XML.....</b>	<b>141</b>

<b>ΠΙΝΑΚΑΣ 24. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALPOPULARARTICLES.PHP</b> .....	<b>142</b>
<b>ΠΙΝΑΚΑΣ 25. ΤΟ ΑΡΧΕΙΟ HELPER.PHP</b> .....	<b>143</b>
<b>ΠΙΝΑΚΑΣ 26. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP</b> .....	<b>144</b>
<b>ΠΙΝΑΚΑΣ 27. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ</b> <b>MOD_PERSONALPOPULARARTICLES.XML</b> .....	<b>145</b>
<b>ΠΙΝΑΚΑΣ 28. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALREADERSPICK.PHP</b> ....	<b>145</b>
<b>ΠΙΝΑΚΑΣ 29. ΤΟ ΑΡΧΕΙΟ HELPER.PHP</b> .....	<b>146</b>
<b>ΠΙΝΑΚΑΣ 30. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP</b> .....	<b>147</b>
<b>ΠΙΝΑΚΑΣ 31. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ</b> <b>MOD_PERSONALREADERSPICK.XML</b> .....	<b>148</b>
<b>ΠΙΝΑΚΑΣ 32. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALSIMILARARTICLES.PHP</b> .....	<b>148</b>
<b>ΠΙΝΑΚΑΣ 33. ΤΟ ΑΡΧΕΙΟ HELPER.PHP</b> .....	<b>156</b>
<b>ΠΙΝΑΚΑΣ 34. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP</b> .....	<b>156</b>
<b>ΠΙΝΑΚΑΣ 35. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ</b> <b>MOD_PERSONALSIMILARARTICLES.XML</b> .....	<b>157</b>
<b>ΠΙΝΑΚΑΣ 36. ΤΟ ΑΡΧΕΙΟ MOD_PERSONALTAGGING.PHP</b> .....	<b>157</b>
<b>ΠΙΝΑΚΑΣ 37. ΤΟ ΑΡΧΕΙΟ HELPER.PHP</b> .....	<b>159</b>
<b>ΠΙΝΑΚΑΣ 38. ΤΟ ΑΡΧΕΙΟ DEFAULT.PHP</b> .....	<b>160</b>
<b>ΠΙΝΑΚΑΣ 39. ΤΟ ΒΟΗΘΗΤΙΚΟ ΑΡΧΕΙΟ ALTER.JS</b> .....	<b>161</b>
<b>ΠΙΝΑΚΑΣ 40. ΤΟ ΒΟΗΘΗΤΙΚΟ ΑΡΧΕΙΟ CHANGE_KW.PHP</b> .....	<b>162</b>
<b>ΠΙΝΑΚΑΣ 41. ΤΟ ΑΡΧΕΙΟ ΕΓΚΑΤΑΣΤΑΣΗΣ</b> <b>MOD_PERSONALTAGGING.XML</b> .....	<b>163</b>

# ΛΙΣΤΑ ΕΙΚΟΝΩΝ

<i>ΕΙΚΟΝΑ 1. ΠΕΡΙΛΗΨΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΣΕ RSS FEEDS.....</i>	<i>21</i>
<i>ΕΙΚΟΝΑ 2. ΕΞΑΓΩΓΗ RSS ΑΠΟ API .....</i>	<i>21</i>
<i>ΕΙΚΟΝΑ 3. ΈΝΑ ΑΠΛΟ ΠΑΡΑΔΕΙΓΜΑ HTML ΚΩΔΙΚΑ.....</i>	<i>22</i>
<i>ΕΙΚΟΝΑ 4. ΠΑΡΑΔΕΙΓΜΑ ΑΠΟ CSS.....</i>	<i>23</i>
<i>ΕΙΚΟΝΑ 5. ΓΕΝΙΚΗ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ PERSSONAL.....</i>	<i>27</i>
<i>ΕΙΚΟΝΑ 6. ΠΡΟΕΠΕΞΕΡΓΑΣΙΑ ΚΕΙΜΕΝΟΥ ΚΑΙ ΕΞΑΓΩΓΗ Λ'ΕΞΕΩΝ- ΚΛΕΙΔΙΩΝ.....</i>	<i>28</i>
<i>ΕΙΚΟΝΑ 7. ΔΙΑΔΙΚΑΣΙΑ ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗΣ ΤΟΥ ΚΕΙΜΕΝΟΥ.....</i>	<i>29</i>
<i>ΕΙΚΟΝΑ 8. ΔΙΑΔΙΚΑΣΙΑ ΕΞΑΓΩΓΗΣ ΠΕΡΙΛΗΨΗΣ.....</i>	<i>30</i>
<i>ΕΙΚΟΝΑ 9. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΠΑΡΟΥΣΙΑΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ.....</i>	<i>32</i>
<i>ΕΙΚΟΝΑ 10. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ADVARSS.....</i>	<i>33</i>
<i>ΕΙΚΟΝΑ 11. ΡΟΗ ΠΛΗΡΟΦΟΡΙΑΣ ΣΤΟ ADVARSS.....</i>	<i>34</i>
<i>ΕΙΚΟΝΑ 12. Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΗΣ ΕΞΑΓΩΓΗΣ ΚΕΙΜΕΝΟΥ.....</i>	<i>35</i>
<i>ΕΙΚΟΝΑ 13. DOM Δ'ΕΝΤΡΟ ΙΣΤΟΣΕΛΪΔΑΣ.....</i>	<i>36</i>
<i>ΕΙΚΟΝΑ 14. ΧΑΡΑΚΤΗΡΙΣΜΌΣ ΚΑΙ ΕΠΙΛΟΓΗ ΚΌΜΒΩΝ.....</i>	<i>37</i>
<i>ΕΙΚΟΝΑ 15. ΤΑ ΠΙΟ ΔΙΑΔΕΔΟΜΈΝΑ CMS.....</i>	<i>41</i>
<i>ΕΙΚΟΝΑ 16. ΔΗΜΟΤΙΚΌΤΗΤΑ ΤΩΝ CMS.....</i>	<i>42</i>
<i>ΕΙΚΟΝΑ 17. ΑΝΑΚΑΛΥΨΗ ΕΪΔΟΥΣ CMS ΔΙΚΤΥΑΚΌΥ ΤΌΠΟΥ.....</i>	<i>43</i>
<i>ΕΙΚΟΝΑ 18. ΠΑΡΑΔΕΙΓΜΑ ΑΠΟΤΕΛΈΣΜΑΤΟΣ ΑΝΑΖΗΤΗΣΗΣ ΕΪΔΟΥΣ CMS.....</i>	<i>43</i>
<i>ΕΙΚΟΝΑ 19. ΤΑ ΚΥΡΙΑ ΣΤΟΙΧΕΪΑ ΕΝΌΣ CMS.....</i>	<i>44</i>
<i>ΕΙΚΟΝΑ 20. ΤΟ ΥΠΟΣΎΣΤΗΜΑ ΣΥΛΛΟΓΗΣ.....</i>	<i>44</i>
<i>ΕΙΚΟΝΑ 21. ΤΟ ΥΠΟΣΎΣΤΗΜΑ ΔΗΜΟΣΪΕΥΣΗΣ ΧΡΗΣΙΜΟΠΟΙΕΪ ΠΡΌΤΥΠΑ ΚΑΙ ΥΠΗΡΕΣΪΕΣ ΔΗΜΟΣΪΕΥΣΗΣ ΓΙΑ ΝΑ ΠΑΡΑΓΕΪ ΙΣΤΟΣΕΛΪΔΕΣ ΚΑΘΌΣ ΚΑΙ ΆΛΛΑ ΕΪΔΗ ΔΗΜΟΣΪΕΎΣΕΩΝ.....</i>	<i>46</i>
<i>ΕΙΚΟΝΑ 22. ΤΟ ΣΎΣΤΗΜΑ ΔΗΜΟΣΪΕΥΣΗΣ ΔΙΑΔΙΚΤΎΟΥ ΜΠΟΡΕΪ ΝΑ ΠΑΡΑΓΕΪ ΜΙΑ ΠΛΗΡΌΣ ΔΎΝΑΜΙΚΗ ΙΣΤΟΣΕΛΪΔΑ.....</i>	<i>47</i>
<i>ΕΙΚΟΝΑ 23. Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ JOOMLA.....</i>	<i>50</i>
<i>ΕΙΚΟΝΑ 24. Η ΑΡΧΙΚΗ ΕΙΚΟΝΑ ΤΟΥ PERSSONAL.....</i>	<i>53</i>
<i>ΕΙΚΟΝΑ 25. ΕΓΓΡΑΦΗ ΣΤΟ PERSSONAL - ΒΉΜΑ 10 ΕΪΣΑΓΩΓΗ ΠΡΟΣΩΠΙΚΌΝ ΣΤΟΙΧΕΪΌΝ.....</i>	<i>53</i>
<i>ΕΙΚΟΝΑ 26. ΕΓΓΡΑΦΗ ΣΤΟ PERSSONAL - ΒΉΜΑ 20 ΕΪΣΑΓΩΓΗ ΕΠΙΘΎΜΗΤΌΝ ΚΑΤΗΓΟΡΌΝ ΓΙΑ ΠΡΟΒΟΛΗ RSS.....</i>	<i>54</i>

<b>ΕΙΚΟΝΑ 27. ΕΓΓΡΑΦΗ ΣΤΟ PERSSONAL - ΒΗΜΑ 30 ΕΙΣΑΓΩΓΗ RSS ΕΙΔΗΣΕΩΝ ΚΑΙ ΛΕΞΕΩΝ ΚΛΕΙΔΙΩΝ.....</b>	<b>54</b>
<b>ΕΙΚΟΝΑ 28. ΕΓΓΡΑΦΗ ΣΤΟ PERSSONAL - ΟΛΟΚΛΗΡΩΣΗ ΤΗΣ ΕΓΓΡΑΦΗΣ.....</b>	<b>55</b>
<b>ΕΙΚΟΝΑ 29. ΕΙΣΟΔΟΣ ΣΤΟ PERSSONAL.....</b>	<b>55</b>
<b>ΕΙΚΟΝΑ 30. Η ΑΡΧΙΚΗ ΕΙΚΟΝΑ ΤΟΥ PERSSONAL.....</b>	<b>55</b>
<b>ΕΙΚΟΝΑ 31. ΠΡΟΒΟΛΗ ΕΝΟΣ ΑΡΘΡΟΥ.....</b>	<b>56</b>
<b>ΕΙΚΟΝΑ 32. ΠΡΟΒΟΛΗ ΤΩΝ ΠΡΟΣΩΠΙΚΩΝ ΑΡΘΡΟΥ ΤΟΥ ΧΡΗΣΤΗ... </b>	<b>56</b>
<b>ΕΙΚΟΝΑ 33. Ο ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ MVC.....</b>	<b>59</b>
<b>ΕΙΚΟΝΑ 34. Ο ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ MVC ΣΤΟ JOOMLA.....</b>	<b>60</b>
<b>ΕΙΚΟΝΑ 35. ΤΑ ΑΡΧΕΙΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΕΙ ΤΟ DEFAULT.PHP ΑΡΧΕΙΟ ΤΟΥ REGISTRATION ΤΟΥ COM_ USERS.....</b>	<b>61</b>
<b>ΕΙΚΟΝΑ 36. ΕΠΙΛΟΓΗ ΤΟΥ ΔΙΑΧΕΙΡΙΣΤΗ ΕΠΕΚΤΑΣΕΩΝ ΤΟΥ JOOMLA .....</b>	<b>65</b>
<b>ΕΙΚΟΝΑ 37. Ο ΔΙΑΧΕΙΡΙΣΤΗΣ ΕΠΕΚΤΑΣΕΩΝ ΤΟΥ JOOMLA.....</b>	<b>65</b>
<b>ΕΙΚΟΝΑ 38. ΕΠΙΛΟΓΗ ΟΠΟΙΑΣΔΗΠΟΤΕ ΕΠΕΚΤΑΣΗΣ ΣΕ ZIP ΓΙΑ ΕΓΚΑΤΑΣΤΑΣΗ ΣΤΟ JOOMLA.....</b>	<b>65</b>
<b>ΕΙΚΟΝΑ 39. Ο ΔΙΑΧΕΙΡΙΣΤΗΣ ΥΠΟΜΟΝΑΔΩΝ ΤΟΥ JOOMLA.....</b>	<b>66</b>
<b>ΕΙΚΟΝΑ 40. ΔΗΜΟΣΤΕΥΣΗ ΥΠΟΜΟΝΑΔΩΝ ΤΟΥ JOOMLA.....</b>	<b>66</b>
<b>ΕΙΚΟΝΑ 41. ΡΥΘΜΙΣΕΙΣ ΥΠΟΜΟΝΑΔΩΝ ΓΙΑ ΤΗ ΘΕΣΗ ΤΟΥΣ ΣΤΟ ΤΕΜΠΛΑΤΕ, ΤΟ ΕΠΙΠΕΔΟ ΠΡΟΣΒΑΣΗΣ ΚΑΙ ΣΕΛΙΔΕΣ ΠΟΥ ΑΝΑΤΙΘΕΝΤΑΙ ΣΤΟ JOOMLA.....</b>	<b>67</b>
<b>ΕΙΚΟΝΑ 42. ΡΥΘΜΙΣΕΙΣ ΣΕΙΡΑΣ ΕΜΦΑΝΙΣΗΣ ΤΩΝ MODULES ΣΤΟ JOOMLA.....</b>	<b>67</b>
<b>ΕΙΚΟΝΑ 43. ΕΓΓΡΑΦΗ ΤΟΥ ΧΡΗΣΤΗ ΣΤΟ PERSSONAL ΤΟΥ JOOMLA</b>	<b>68</b>
<b>ΕΙΚΟΝΑ 44. ΦΟΡΤΩΣΗ ΤΩΝ ΚΑΤΗΓΟΡΙΩΝ ΤΟΥ PERSSONAL ΜΕΣΩ AJAX.....</b>	<b>69</b>
<b>ΕΙΚΟΝΑ 45. ΑΜΕΣΗ ΦΟΡΤΩΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΤΩΝ ΚΑΤΗΓΟΡΙΩΝ ΤΟΥ PERSSONAL ΠΟΥ ΜΑΣ ΕΝΔΙΑΦΕΡΟΥΝ.....</b>	<b>69</b>
<b>ΕΙΚΟΝΑ 46. ΕΠΙΛΟΓΗ ΤΩΝ RSS ΚΑΙ ΛΕΞΕΩΝ ΚΛΕΙΔΙΩΝ ΠΟΥ ΜΑΣ ΕΝΔΙΑΦΕΡΟΥΝ ΣΤΟ PERSSONAL.....</b>	<b>70</b>
<b>ΕΙΚΟΝΑ 47. ΕΙΣΑΓΩΓΗ ΠΡΟΣΩΠΙΚΩΝ ΣΤΟΙΧΕΙΩΝ ΓΙΑ ΤΗΝ ΕΓΓΡΑΦΗ ΣΤΟ PERSSONAL.....</b>	<b>70</b>
<b>ΕΙΚΟΝΑ 48. ΤΟ ΝΕΟ PERSSONAL ΣΤΟ JOOMLA.....</b>	<b>71</b>
<b>ΕΙΚΟΝΑ 49. ΠΡΟΒΟΛΗ ΑΡΘΡΟΥ ΣΤΟ ΝΕΟ PERSSONAL ΤΟΥ JOOMLA .....</b>	<b>72</b>
<b>ΕΙΚΟΝΑ 50. ΠΡΟΒΟΛΗ ΔΙΕΠΑΦΗΣ ΔΙΑΧΕΙΡΙΣΤΗ ΣΤΟ COMPONENT COMPERSSONAL ΣΤΟ JOOMLA.....</b>	<b>73</b>
<b>ΕΙΚΟΝΑ 51. ΠΡΟΒΟΛΗ ΧΡΗΣΤΩΝ ΠΟΥ ΕΙΝΑΙ ΕΓΓΕΓΡΑΜΜΕΝΟΙ ΣΤΟ ΠΑΛΑΙΟ PERSSONAL ΚΑΙ ΠΡΕΠΕΙ ΝΑ ΜΕΤΑΦΕΡΘΟΥΝ ΣΤΟ JOOMLA.....</b>	<b>74</b>



# ΓΛΩΣΣΑΡΙΟ

Back End	Διεπαφή για την διαχείριση του δικτυακού τόπου του Joomla.
Blog	Ιστοσελίδα που συνήθως υπάρχει μόνο ένας συγγραφέας.
Browser	Φυλλομετρητής.
Client	Πρόγραμμα πελάτη που ζητά εξυπηρέτηση.
Compiler	Μεταγλωττιστής.
Component	Συστατικό στοιχείο του ανοιχτού λογισμικού Joomla.
Drupal	Λογισμικό ανοιχτού κώδικα για τη δημιουργία δικτυακών τόπων.
Front-End	Διεπαφή για τη προβολή του δικτυακού τόπου του Joomla.
HTML Validator	Πρόγραμμα με το οποίο μπορεί να εξεταστεί η ορθότητα ενός HTML κώδικα
Javascript	Κώδικας και γλώσσα προγραμματισμού που εκτελείται στο περιβάλλον του φυλλομετρητή
Java applet	Πρόγραμμα γραμμένο σε Java και σχεδιασμένο για να εκτελείται στον φυλλομετρητή
Joomla	Λογισμικό ανοιχτού κώδικα για τη δημιουργία δικτυακών τόπων.
Keyword	Λέξη κλειδί
Link	Σύνδεσμος (αναφέρεται σε ιστοσελίδα)
Log	Αρχείο που λειτουργεί ως ημερολόγιο, κρατώντας στοιχεία για μηνύματα που ανταλλάσσονται και γεγονότα που συμβαίνουν
Module	Πρόσθετο στοιχείο για την επέκταση του ανοιχτού λογισμικού Joomla
News portal	Πύλη ενημερωτικού περιεχομένου
Open source software	Λογισμικό ανοιχτού κώδικα
Portal	Δικτυακή πύλη
Proxy server	Διαμεσολαβητής
RSS pool	Ένα οργανωμένο και ταξινομημένο σύνολο από RSS feeds
Search engine	Μηχανή αναζήτησης

---

Server	Αυτός που αναλαμβάνει να εξυπηρετήσει τους πελάτες
Subscriber	Μέλος και συνδρομητής κάποιας υπηρεσίας
Template	Πρότυπο που αναφέρεται κυρίως σε δικτυακό τόπο.
Text Categorization	Κατηγοριοποίηση κειμένου
Timer	Μετρητής που αυξάνεται σε κάθε κύκλο εκτέλεσης
Trigger	Σκανδαλιστής (διαδικασία που πυροδοτείται με κάποιο γεγονός)
User profile	Προφίλ χρήστη
Website	Ιστοσελίδα
Web clipping	Τεχνική εντοπισμού χρήσιμης πληροφορίας
Wordpress	Λογισμικό ανοιχτού κώδικα για τη δημιουργία δικτυακών τόπων.
XML Analysis	Ανάλυση κώδικα XML
XML	Γλώσσα σήμανσης που περιέχει σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση δεδομένων.

## ΣΥΝΤΟΜΟΓΡΑΦΊΕΣ

DBMS	DataBase Management System
HTML	HyperText Mark-up Language
HTTP	HyperText Transfer Protocol
IR	Information Retrieval
GPL	General Public License
ODBMS	Object DataBase Management System
RSS	Rich Site Summary
SSL	Secure Socket Layer
URL	Uniform Resource Locator
WWW	World Wide Web
XML	Extensible Markup Language
ΒΔ	Βάση Δεδομένων

## **1 ΕΙΣΑΓΩΓΗ**

Ο όγκος της πληροφορίας που αποθηκεύεται καθημερινά στο διαδίκτυο από τις ειδησεογραφικές πύλες ενημέρωσης αυξάνεται ραγδαία. Η ανάπτυξη ολοένα και περισσότερων ειδησεογραφικών πύλων ενημέρωσης δημιουργεί σύγχυση στους χρήστες του διαδικτύου, καθώς η διαδικτυακή πληροφορία και η ανάκτηση της επιθυμητής αποτελεί πολύτιμο αγαθό. Η δημιουργία ενός συστήματος που θα εξάγει προσωποποιημένα αποτελέσματα σύμφωνα με τον τρόπο οργάνωσης της πληροφορίας από το εκάστοτε χρήστη αποτελεί κυρίαρχο μέλημά μας, καθώς αναπτύσσονται ολοένα και περισσότερα εργαλεία που επιφέρουν σημαντικές αλλαγές σε αυτόν τον τομέα.

Στόχος της παρούσας διπλωματικής εργασίας είναι η μεταφορά του συστήματος *reRSSonal* στην πλατφόρμα ανοιχτού κώδικα Joomla και η μελέτη της εξατομίκευσης των ειδησεογραφικών πηγών του διαδικτύου. Μέσα από την χρήση του Joomla ανοίγονται νέοι δρόμοι, καθώς παρέχεται η δυνατότητα εισαγωγής υπομονάδων (*modules*) που θα συμβάλλουν σημαντικά στην περαιτέρω ανάπτυξη του υπάρχοντος συστήματος *reRSSonal* και θα βελτιώσουν την προσωποποίηση των ειδησεογραφικών πληροφοριών. Το σύστημα *reRSSonal* θα ανοίξει τις πύλες του σε ένα ανοιχτό λογισμικό με υπηρεσίες που προσφέρονται από χιλιάδες προγραμματιστές του κόσμου που συνεισφέρουν καθημερινά πληθώρα νέων υπομονάδων (*modules*) και που μπορούν να δώσουν ολοκαίνουργιες δυνατότητες στο υπάρχον σύστημα.

Στην εργασία αυτή γίνεται μια εκτενής αναφορά στον τρόπο λειτουργίας του *reRSSonal*, ώστε να τονιστούν όλες οι παράμετροι που επηρέασαν στην κατασκευή και τη λειτουργία των συστημάτων ανάκτησης πληροφορίας από πηγές συνεχούς ενημέρωσης, καθώς και οι δυσκολίες για τη μεταφορά του στην πλατφόρμα του Joomla.

Το σύστημα αποδελτίωσης *reRSSonal* που αποσκοπεί στην παροχή προσωποποιημένου περιεχομένου στον χρήστη αποτελεί ένα καινοτόμο σύστημα το οποίο αναπτύχθηκε στα πλαίσια της μιας ερευνητικής διατριβής [29] και οδήγησε σε αρκετές δημοσιεύσεις. Θα αναλύσουμε περαιτέρω το σύστημα *reRSSonal* στο κεφάλαιο 3.

### **1.1 Προσδιορισμός του Προβλήματος**

Η νέα μορφή πληροφόρησης στη καθημερινή μας ζωή παρέχεται από το λεγόμενο «ηλεκτρονικό» Τύπο, με κυριότερους εκπροσώπους να είναι οι πύλες ενημέρωσης (*news portals*) και τα ιστολόγια (*blogs*). Η ενημέρωση του κόσμου από το διαδίκτυο ολοένα και αυξάνεται με αποτέλεσμα να χρειάζονται νέες τεχνολογίες και υπηρεσίες που διαρκώς θα ανανεώνονται και θα εμπλουτίζονται με νέα στοιχεία.

Το ζήτημα έχει απασχολήσει την ερευνητική κοινότητα και έχουν αναπτυχθεί αρκετές προσεγγίσεις. Οι περισσότερες είναι γενικευμένες και σχετίζονται με την ευρύτερη έννοια της ανάκτησης πληροφορίας στον Ιστό [31].

Η ανάκτηση της πληροφορίας από τα διάφορα ειδησεογραφικά ιστολόγια έχει ομαδοποιηθεί με τη χρήση της τεχνολογίας RSS και παρουσιάζεται συμπυκνόμενη στο χρήστη. Όμως οι η διαρκής ροή ειδήσεων από διάφορες πηγές ενημέρωσης δεν εξυπηρετεί το χρήστη καθώς καθιστά αδύνατο την παρακολούθηση τους και κρίνεται επιβεβλημένη η προσωποποίηση του αποτελέσματος.

Το παραπάνω πρόβλημα επιλύεται σημαντικά από το σύστημα *reRSSonal* για αυτό και δεν θα ελεγκταθούμε σε αυτό. Το πρόβλημα στην παρούσα διπλωματική έγκειται στο κατά πόσο μπορεί να ελεγκταθεί το σύστημα *reRSSonal* σε μια πλατφόρμα ανοιχτού κώδικα όπως είναι το Joomla και ποια είναι τα πλεονεκτήματα από της χρήση τέτοιου είδους λογισμικών για την προσωποποίηση διαδικτυακής πληροφορίας.

## **1.2 Δομή της Εργασίας**

Στο 1<sup>ο</sup> κεφάλαιο προσδιορίζουμε ποιό είναι το αντικείμενο της μελέτης μας και πως θα το αντιμετωπίσουμε. Ποιοί είναι οι προβληματισμοί μας για την μεταφορά του υπάρχοντος συστήματος *reRSSonal* στο Joomla και πως θα το επιλύσουμε.

Το 2ο κεφάλαιο αφιερώνεται στην εξεικώση του αναγνώστη με κάποιες έννοιες όπως είναι η HTML, PHP, MYSQL, XML, RSS και που χρησιμοποιούνται μέσα από κάποια απλά παραδείγματα. Οι έννοιες αυτές θα μας χρησιμεύσουν στην μετέπειτα ανάγνωση της εργασίας.

Το 3ο κεφάλαιο αφορά το υπάρχον σύστημα *reRSSonal*. Το κεφάλαιο αυτό έχει αναφερθεί ήδη σε προηγούμενες διπλωματικές εργασίες όπου υλοποιήθηκε το *reRSSonal*. Το κεφάλαιο αυτό βοηθά τον αναγνώστη στην κατανόηση του υπάρχοντος συστήματος όπως έχει υλοποιηθεί, διότι αυτό θα μεταφερθεί στο Joomla.

Το 4ο κεφάλαιο αφορά τα συστήματα διαχείρισης περιεχομένου. Γίνεται εκτενής ανάλυση στα συστήματα που υπάρχουν διαθέσιμα στην αγορά, που χρησιμεύουν και πως χρησιμοποιείται το καθένα. Αναλύονται τα θετικά και αρνητικά χαρακτηριστικά τους και γίνεται μια σύγκριση αυτών.

Το 5ο κεφάλαιο αφιερώνεται στο Joomla, όπου γίνεται ανάλυση της αρχιτεκτονικής του, του τρόπου λειτουργίας του, των στοιχείων που διαθέτει καθώς και των επεκτάσεων και των πρόσθετων στοιχείων του. Παρουσιάζονται τα πλεονεκτήματά του και οι λόγοι για τους οποίους είναι από τα πιο δημοφιλή συστήματα διαχείρισης περιεχομένου.

Στο 6ο κεφάλαιο παρουσιάζουμε την υλοποίηση του *reRSSonal* στο Joomla μέσα από τα ολοκαίνουργια συστατικά και πρόσθετα στοιχεία που υλοποιήσαμε με βάση τον υπάρχοντα κώδικα *php* που χρησιμοποιούσε το *reRSSonal* κανοντας χρήση του Joomla framework και των επιπλέον χαρακτηριστικών που χρησιμοποιεί για την παραγωγή του τελικού αποτελέσματος που εμφανίζεται επίσης στο κεφάλαιο αυτό. Τέλος αναφέρουμε τα συμπεράσματα που αποκομίσαμε από την διπλωματική εργασία καθώς και τις μελλοντικές προσθήκες που μπορούν να υλοποιηθούν στο νέο μας σύστημα.

## **2 ΘΈΜΑΤΑ ΠΟΥ ΘΑ ΜΑΣ ΑΠΑΣΧΟΛΉΣΟΥΝ**

Στο κεφάλαιο αυτό παρουσιάζονται οι μηχανισμοί που κυριαρχούν στο διαδίκτυο και που συμβάλουν στην μετάδοση της πληροφορίας. Θα αναφερθούμε σε έννοιες του διαδικτύου οι οποίες θα μας χρησιμεύσουν αργότερα στην περιγραφή του συστήματος *reSSonal* και της αρχιτεκτονικής αυτού. Η ανάλυση αυτή θα βοηθήσει τον αναγνώστη να δημιουργήσει το θεωρητικό υπόβαθρο προκειμένου να κατανοήσει τους τρόπους λειτουργίας και μετάδοσης της διαδικτυακής πληροφορίας καθώς και των εργαλείων που συμβάλλουν σε αυτό.

Η αρχιτεκτονική του διαδικτύου είναι αρκετά πολύπλοκη και κάνει χρήση πολλών και διαφορετικών πρωτοκόλλων. Το κυρίαρχο διαδικτυακό πρωτόκολλο που χρησιμοποιείται είναι το TCP/IP στο οποίο βασίζεται όλη η μεταφορά δεδομένων. Στη συνέχεια θα ασχοληθούμε με τα κανάλια επικοινωνίας μέσω των οποίων η χλώδης πληροφορία του διαδικτύου που μεταφέρεται μέσω του πρωτοκόλλου HTTP συμπύσσεται και προβάλλεται ομοιόμορφα στο χρήστη. Τα κανάλια επικοινωνίας είναι ευρέως γνωστά στα περισσότερα ειδησεογραφικά ιστολόγια και αποτελούν πλέον αναπόσπαστο κομμάτι της καθημερινής διαδικτυακής ενημέρωσης.

Μια παράγραφος ακόμα αφιερώνεται στους *web crawlers* που αποτελούν ισχυρό εργαλείο αναζήτησης της πληροφορίας από τις μηχανές αναζήτησης. Θα παρουσιάσουμε επιγραμματικά την αρχιτεκτονική των *crawler* και τις κύριες εφαρμογές τους, καθώς γίνεται χρήση τους από το σύστημα *reSSonal* κατά την εξόρυξη της πληροφορίας από τα κανάλια επικοινωνίας.

Τέλος θα δώσουμε μια μικρή αναφορά στα λογισμικά ανοιχτού κώδικα, πως αναπτύσσονται και την τεράστια χρησιμότητά τους στην ανάπτυξη του διαδικτύου. Δεν θα επεκταθούμε ιδιαίτερα σε αυτό, διότι θα γίνει εκτενέστερη αναφορά στο κεφάλαιο 4.

### **2.1 Η πληροφορία στο διαδίκτυο και η γλώσσα HTML**

Κάθε σελίδα που εμφανίζεται στο Internet είναι ένα αρχείο γραμμένο με τη γλώσσα HTML (HyperText Markup Language, Γλώσσα Χαρακτηρισμού Υπερ-Κειμένου), που περιλαμβάνει το κείμενο της σελίδας, τη δομή της και τους συνδέσμους προς άλλα έγγραφα, εικόνες ή άλλα μέσα.

Ο φυλλομετρητής (*Web browser*) παίρνει τις πληροφορίες από τον *Web server*, τις μορφοποιεί και τις εμφανίζει κατάλληλα για το σύστημά μας. Διαφορετικά προγράμματα φυλλομετρητή μπορεί να μορφοποιούν και να εμφανίζουν το ίδιο αρχείο με διαφορετικό τρόπο, ανάλογα με τις δυνατότητες του συστήματος στο οποίο τρέχουν και τις επιλογές διαμόρφωσης του προγράμματος του φυλλομετρητή.

Μια *Web* σελίδα ή ιστοσελίδα (*Web page*) είναι ένα μεμονωμένο στοιχείο μιας παρουσίασης για το *Web* και περιέχεται σ' ένα αρχείο στον δίσκο, το οποίο ανακτάται από έναν *Web server* και μορφοποιείται μέσω ενός φυλλομετρητή.

Η αρχική σελίδα (home page) είναι η πρώτη ή κορυφαία σελίδα μιας παρουσίασης για το Web, είναι δηλαδή το σημείο εισόδου ή εκκίνησης για τις υπόλοιπες σελίδες της παρουσίασης και η πρώτη σελίδα που θα συναντήσουν οι αναγνώστες της παρουσίασής μας. Η αρχική σελίδα περιέχει συνήθως μια σύνοψη του περιεχομένου της παρουσίασης με τη μορφή ενός πίνακα περιεχομένων ή μιας ομάδας εικονιδίων.

Η HTML είναι το ακρωνύμιο των λέξεων HyperText Markup Language, δηλ. Γλώσσα Χαρακτηρισμού Υπερ-Κειμένου και βασίζεται στη γλώσσα SGML, Standard Generalized Markup Language, που είναι ένα πολύ μεγαλύτερο σύστημα επεξεργασίας εγγράφων. Η HTML ορίζει ένα σύνολο κοινών στυλ για τις Web σελίδες, όπως τίτλοι (titles), επικεφαλίδες (headings), παράγραφοι (paragraphs), λίστες (lists) και πίνακες (tables). Ορίζει επίσης στυλ χαρακτήρων, όπως η έντονη γραφή (boldface) και οι ενότητες κώδικα.

Κάθε στοιχείο έχει ένα όνομα και περιέχεται μέσα στα σύμβολα <>, που αποκαλούνται tags (ετικέτες). Όταν γράφουμε μια Web σελίδα με την HTML, στην ουσία δίνουμε τίτλους στα διάφορα στοιχεία της σελίδας μ' αυτά τα tags.

Οι φυλλομετρητές, μαζί με τη δυνατότητά τους να ανακτούν σελίδες από το Web, λειτουργούν επίσης και σαν μορφοποιητές για την HTML. Όταν διαβάζουμε μια σελίδα γραμμένη με την HTML σ' έναν φυλλομετρητή, ο φυλλομετρητής διαβάζει (διερμηνεύει) τα tags της HTML και μορφοποιεί το κείμενο και τις εικόνες στην οθόνη.

Διαφορετικοί φυλλομετρητές, οι οποίοι τρέχουν σε διαφορετικούς υπολογιστές, μπορεί να αντιστοιχίζουν διαφορετικά στυλ σε κάθε στοιχείο μιας σελίδας. Αυτό σημαίνει ότι οι σελίδες που δημιουργούμε με την HTML μπορεί να δείχνουν εντελώς διαφορετικές από σύστημα σε σύστημα και από φυλλομετρητή σε φυλλομετρητή. Δηλαδή, οι πραγματικές πληροφορίες και οι σύνδεσμοι που περιέχουν οι σελίδες μας θα είναι πάντα εκεί, αλλά η εμφάνιση των σελίδων στην οθόνη θα είναι διαφορετική.

## □□□□□ **Η Δομή μιας HTML Σελίδας**

Οι σελίδες που γράφουμε με την HTML είναι απλά αρχεία κειμένου σε μορφή ASCII, που σημαίνει ότι δεν περιέχουν πληροφορίες για κάποιο λειτουργικό σύστημα ή πρόγραμμα, αλλά μπορούν να διαβαστούν από οποιονδήποτε συντάκτη υποστηρίζει απλό κείμενο.

Τα αρχεία της HTML περιέχουν τα ακόλουθα :

- Το κείμενο της σελίδας.
- Τα tags της HTML, τα οποία υποδεικνύουν τα στοιχεία, τη δομή και τη μορφοποίηση των σελίδων, καθώς επίσης και τους συνδέσμους υπερ-κειμένου προς άλλες σελίδες ή προς αρχεία άλλων μορφών (πολυμέσα).

Τα περισσότερα tags της HTML έχουν την εξής μορφή :

□ <Όνομα Tag> επηρεαζόμενο κείμενο </Όνομα Tag>

Τα tags της HTML έχουν γενικά ένα tag αρχής και ένα tag τέλους ή ένα tag ανοίγματος και ένα tag κλεισίματος, τα οποία περικλείουν το κείμενο που επηρεάζουν. Το tag αρχής ενεργοποιεί μια λειτουργία ή ένα χαρακτηριστικό, όπως είναι για παράδειγμα η έντονη γραφή, ενώ το tag τέλους την απενεργοποιεί. Τα tags τέλους έχουν το ίδιο όνομα με τα tags αρχής, αλλά με πρόθεμα τον χαρακτήρα /.

Δεν αποτελούν ζευγάρι όλα τα tags της HTML, καθώς ορισμένα είναι "μονομελή", ενώ άλλα περιέχουν επιπλέον πληροφορίες και κείμενο μέσα στα σύμβολα <>. Ακόμη, όλα τα tags της HTML δεν κάνουν διάκριση μεταξύ κεφαλαίων και πεζών γραμμάτων, δηλ. μπορούμε να τα γράφουμε είτε με κεφαλαίους είτε με πεζούς χαρακτήρες είτε με οποιονδήποτε συνδυασμό τους. Συνήθως τα γράφουμε κεφαλαία για να μπορούμε να τα ξεχωρίζουμε ευκολότερα από το κείμενο της σελίδας.

Τα αρχεία της HTML πρέπει να τα γράφουμε σε συντάκτες κειμένων, δηλ. προγράμματα που μπορούν να αποθηκεύσουν αρχεία κειμένου σε μορφή ASCII. Τέτοια προγράμματα είναι τα Notepad, Wordpad και Write των Windows, το Edit του DOS και τα vi, emacs και pico του UNIX.

Το όνομα του αρχείου θα πρέπει να έχει επέκταση .html ή .htm σε συστήματα DOS ή Windows που επιτρέπουν επεκτάσεις μόνο τριών χαρακτήρων. Η HTML χρησιμοποιεί τρία tags για την περιγραφή της συνολικής δομής μιας σελίδας, τα οποία παρέχουν ορισμένες απλές πληροφορίες κεφαλίδας. Αυτά τα tags προσδιορίζουν τη σελίδα μας στους φυλλομετρητές και παρέχουν επίσης απλές πληροφορίες για τη σελίδα, όπως τον τίτλο ή τον συγγραφέα της, πριν από τη φόρτωση ολόκληρης της σελίδας.

ας.

## **2.2 Η τεχνολογία XML**

Η XML δεν είναι μια σημειακή γλώσσα όπως η HTML, αλλά χρησιμοποιείται για την περιγραφή μιας σημειακής γλώσσας. Άρα ουσιαστικά δεν πρόκειται για μια γλώσσα αυτή καθαυτή αλλά για μια μετα-γλώσσα. Η χρήση της επιφέρει σημαντικά αποτελέσματα διότι το μόνο που απαιτείται είναι ο καθορισμός της γλώσσας που προσδιορίζει ότι επιθυμείτε από τον εκάστοτε προγραμματιστή. Ο προσδιορισμός μπορεί να αφορά την περιγραφή ηλεκτρονικών κυκλωμάτων, ανταλλαγή δεδομένων, μηχανές αυτοκινήτων και οτιδήποτε μπορεί να περιγραφεί από μια σημειακή γλώσσα.

Ο χρήστης φτιάχνει τα αρχεία XML που περιέχουν κείμενα με την γλώσσα που περιγράφει το αρχείο αυτό και ύστερα με κατάλληλα προγράμματα γίνεται η επεξεργασία τους και εκτελείται η μετατροπή τους σε κάποια άλλη μορφή πιο ευανάγνωστη από αυτή της XML μορφής.

Η XML χρησιμοποιείται για τον προσδιορισμό μιας σημειακής γλώσσας. Οι σημειακές γλώσσες περιέχουν στοιχεία μαρκαρισμένα σε επιθέματα (tags). Κάθε επίθεμα χαρακτηρίζει ένα συγκεκριμένο στοιχείο του κειμένου. Μπορούμε να έχουμε και συνδυασμό επιθεμάτων για να χαρακτηρίσουμε ένα στοιχείο. Επίσης κάποια στοιχεία μπορεί να έχουν κάποιες ιδιοτιμές ή αλλιώς ιδιοχαρακτηριστικά τα οποία μπορούν να εμφανίζονται μια μόνο φορά ανά στοιχείο.

Το βασικότερο πλεονέκτημα της XML είναι ότι παρέχει τη δυνατότητα σε οποιαδήποτε γλώσσα προγραμματισμού να δημιουργεί και να επεξεργάζεται δεδομένα στην μορφή αυτή. Στην ουσία το XML είναι ένα έγγραφο κειμένου καλά δομημένο που περιέχει μοναδικά tags για να χαρακτηρίζει τα δεδομένα του.

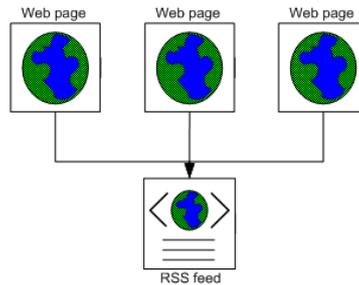
Ο παγκόσμιος ιστός, όπως είναι φυσικό, κάνει χρήση κατά κόρον της τεχνολογίας XML με διάφορους τρόπους. Οι πιο γνωστές εφαρμογές είναι τα web services , αλλά και τα RSS που θα αναλύσουμε στη συνέχεια.

## **2.3 Τι είναι το RSS**

Η τεχνολογία RSS (Really Simple Syndication) αποτελεί τον πιο απλό τρόπο αποστολής πληροφοριών από μια ιστοσελίδα, χωρίς να χρειάζεται ο χρήστης να επαναλαμβάνει τις επισκέψεις του στην ιστοσελίδα και να αναζητά πληροφορίες από το δικτυακό τόπο. Το μόνο που απαιτείται είναι να δηλώσει στο πρόγραμμα RSS (ή στην online υπηρεσία RSS) τι ακριβώς επιθυμεί να λαμβάνει και από ποιο δικτυακό τόπο και άμεσα αποστέλλονται κάθε φορά οι νεότερες πληροφορίες που έχει επιλέξει στον υπολογιστή του.

Το RSS είναι μια δομή ανταλλαγής περιεχομένου από δικτυακούς τόπους βασισμένο στη γλώσσα XML. Κάθε κανάλι RSS διαθέτει μια λίστα στοιχείων που περιέχουν έναν τίτλο και το σύνδεσμο προς το αντίστοιχο σύνδεσμο της ιστοσελίδας με τις περεταίρω πληροφορίες. Η χρήση των RSS καναλιών αλλάζει τα δεδομένα

στην καθημερινή ειδησιογραφία. Το κανάλι RSS επιτρέπει στους χρήστες να έχουν πρόσβαση σε μια περίληψη των περιεχομένων του ιστότοπου.



**Εικόνα 1. Περίληψη περιεχομένου σε RSS feeds**

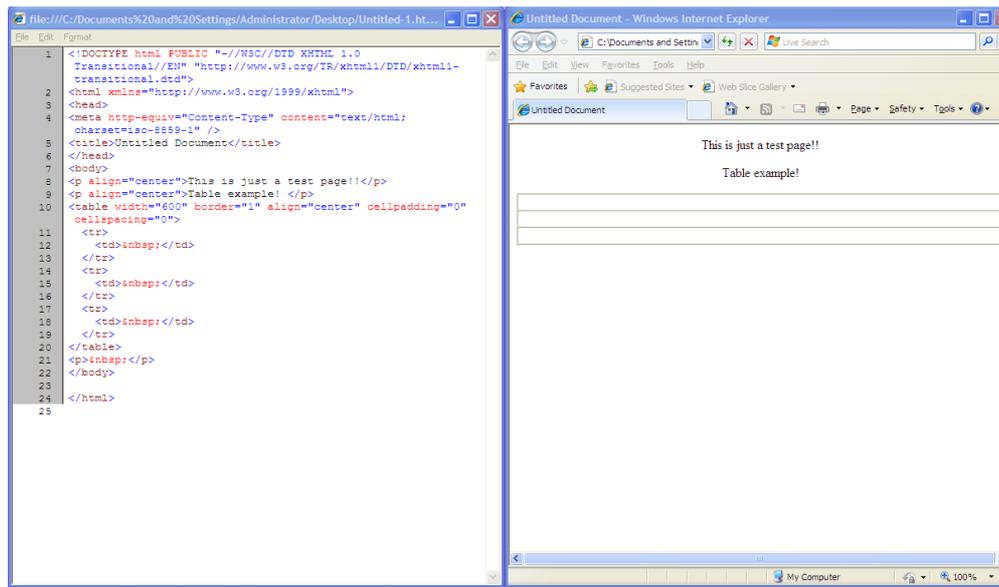
Τα RSS κανάλια επικοινωνίας επιτρέπουν τη διαχείριση του περιεχομένου απομακρυσμένα με τη χρήση ειδικών API που επιτρέπουν την εξαγωγή του περιεχομένου δίχως ο χρήστης να απαιτείται να έχει ενεργό κάποιο browser, ενώ πλέον υπάρχουν πολλά widgets που έχουν παρόμοια χρήση σε λειτουργικά συστήματα όπως είναι τα Windows, το Linux ubuntu καθώς και σε διάφορες φορητές συσκευές. Τα πλεονεκτήματα των χρηστών που χρησιμοποιούν τέτοιου είδους API από ειδησεογραφικά site είναι πάρα πολλά καθώς παρέχουν τη δυνατότητα ενημέρωσης μόνο με τίτλους και μόνο σε κατηγορίες ενδιαφέρουσες για τον εκάστοτε χρήστη.



**Εικόνα 2. Εξαγωγή RSS από API**

## **2.4 Οι ιστοσελίδες από την αρχή της εμφάνισης τους**

Ο σχεδιασμός των ιστοσελίδων αρχικά είχε στατικό περιεχόμενο με χρήση της γλώσσας προγραμματισμού HTML. Ο κατασκευαστής ιστοσελίδων χρησιμοποιώντας την HTML έκανε την αναπαράσταση της ιστοσελίδας και κάθε φορά που χρειαζόταν να γίνει κάποια αλλαγή θα έπρεπε να την κάνει στην ίδια την ιστοσελίδα φορτώνοντας κάθε φορά τη νέα έκδοση στο web server. Ένα απλό παράδειγμα απλής στατικής ιστοσελίδας φαίνεται στην παρακάτω εικόνα.



Εικόνα 3. Ένα απλό παράδειγμα HTML κώδικα

## 2.5 Τι είναι τα CSS

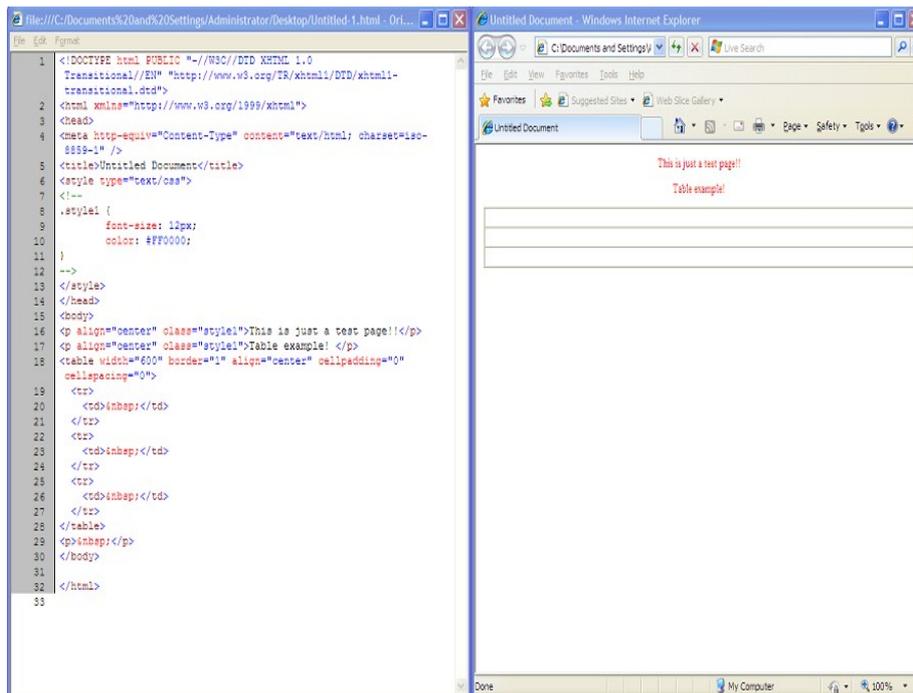
Καθώς οι ιστοσελίδες του διαδικτύου άρχισαν να αναπτύσσονται σε όγκο και εμπλουτίστηκαν με νέα στοιχεία όπως είναι βίντεο, ήχος, streaming και πολλές άλλες λειτουργίες, η ομαδοποίηση του περιεχομένου κατέστη επιτακτική. Ομαδοποίηση ως προς τον τρόπο εμφάνισης κοινών στοιχείων μιας ιστοσελίδας που εξυπηρετούσε στη χρήση λιγότερου κώδικα στις ιστοσελίδες και βοηθούσε στην επαναχρησιμοποίηση κώδικα.

Τα CSS (Common Style Sheet) ή αλλιώς διαδοχικά στυλ φύλλων εισήχθησαν στην πορεία ανάπτυξης της HTML και της XHTML. Τα CSS δεν είναι τίποτα άλλο παρά αρχεία τύπου .css τα οποία χρησιμοποιούνται για τον έλεγχο της εμφάνισης ενός εγγράφου και τα οποία γράφονται σε μια γλώσσα σήμανσης. Τα CSS είναι στην ουσία μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στιλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.

Η σύνταξη των CSS είναι απλή και αφορά τα ονόματα διάφορων στιλιστικών ιδιοτήτων μιας ιστοσελίδας. Το κάθε CSS αποτελείται από ένα σύνολο κανόνων. Κάθε κανόνας αποτελείται από μια ή περισσότερες επιλογές και ένα σύνολο δηλώσεων. Υπάρχουν και κάποιοι επιπλέον κανόνες δηλώσεων, γραφής και σύνταξης των CSS αλλά δεν θα ελεγκταθούμε περισσότερο σε αυτό το κομμάτι.

Η εισαγωγή των CSS σε μια σελίδα γίνεται είτε απευθείας είτε με έμμεση αναφορά από ένα αρχείο css. Για παράδειγμα εάν θέλουμε κάποιο κείμενο να είναι χρώματος κόκκινου και μεγέθους χαρακτήρων 12 μπορούμε να ορίσουμε ένα στυλ (π.χ. style1) το οποίο το εισάγουμε απευθείας στον HTML κώδικά μας ως εξής:

Στο προηγούμενο παράδειγμα HTML που δώσαμε ο κώδικας της HTML με την εισαγωγή του CSS και το αποτέλεσμα του όπως θα φαινόταν στον web browser μας φαίνεται στην παρακάτω εικόνα. Παρατηρήστε ότι οι δύο πρώτες γραμμές που αφορούν το κείμενο παραγράφου έχουν αποκτήσει τη στιλιστική ιδιότητα style1 που ορίσαμε στο CSS.



**Εικόνα 4. Παράδειγμα από CSS**

Με αυτόν τον τρόπο δεν χρειάζεται δηλαδή να δηλώσουμε ξανά ή ακόμα και να γράφουμε ξανά τα χαρακτηριστικά που θέλουμε να αποκτήσει ένα HTML στοιχείο. Αυτό μας αποδίδει σημαντικά αποτελέσματα σε προγραμματιστικό χρόνο διότι ελαχιστοποιούμε επαναλαμβανόμενες διαδικασίες σε οτιδήποτε στυλιστικό στον HTML κώδικα. Το σημαντικότερο όμως χαρακτηριστικό έγκειται στο ότι δεν απαιτείται να μετατρέψουμε κάθε στοιχείο ξεχωριστά στον HTML κώδικα. Έτσι το μόνο που χρειάζεται είναι να ομαδοποιήσουμε τα στοιχεία του HTML κώδικα με τα αντίστοιχα χαρακτηριστικά που έχουμε δημιουργήσει στα CSS και τότε οποιαδήποτε αλλαγή στο CSS εφαρμόζεται άμεσα σε όλα τα στοιχεία του HTML κώδικα.

Αυτό είναι το κυρίαρχο χαρακτηριστικό των CSS που μας εξυπηρετεί σημαντικά στο σχεδιασμό ιστοσελίδων και που χρησιμοποιείται μέχρι και σήμερα και συμβάλει στην ανάπτυξη των πρότυπων (templates) ιστοσελίδων που εισάγονται πλέον και στα συστήματα διαχείρισης περιεχομένου που θα αναλύσουμε και παρακάτω.

## **2.6 Δυναμικές ιστοσελίδες με χρήση PHP/MYSQL**

Οι ιστοσελίδες του διαδικτύου στην αρχή της εμφάνισής τους και για αρκετά χρόνια ήσαν στατικές. Τα περιεχόμενά τους δεν ανανεώνονταν συχνά και δεν υπήρχε τόση επισκεψιμότητα ώστε να γίνεται εμφανής η ανανέωσή τους μέσα από τη αλλαγή του HTML κώδικα. Σήμερα ωστόσο οι ιστοσελίδες ανανεώνονται αρκετά και στην περίπτωση των ειδησεογραφικών πυλών η ανανέωση είναι ακατάπαυστη.

Οι βάσεις δεδομένων είχαν ήδη κάνει την εμφάνισή της στον κόσμο των υπολογιστών για την συμπυκνόμενη και δομημένη αποθήκευση των δεδομένων των χρηστών. Η χρήση τους είναι επιτακτική καθώς παρέχεται η δυνατότητα ανανέωσης των δεδομένων των χρηστών δίχως την ανάγκη δημιουργίας τυποποιημένων αρχείων από το χρήστη. Οι δυνατότητες τους αυξάνονται ώστε να διαχειρίζονται ολόένα και περισσότερα δεδομένα, ενώ η ενσωμάτωση μέχρι και δυαδικών δεδομένων (binary

data) από αρχεία εικόνων, βίντεο, ήχου, αρχείων pdf και αμέτρητων άλλων ειδών αρχείων τις καθιστά ιδιαίτερα δημοφιλής στο χώρο των υπολογιστών και η φήμη τους δεν άργησε να περάσει και στο περιεχόμενο των ιστοσελίδων. Είναι πάρα πολλές περιπτώσεις βάσεων δεδομένων κάθε μια με τα δικά τους χαρακτηριστικά και δυνατότητες. Θα εξετάσουμε τη MYSQL η οποία είναι και η πιο διαδεδομένη βάση δεδομένων που χρησιμοποιείται και στο web.

Η MYSQL είναι μια σχεσιακή βάση δεδομένων με το ανάλογο σύστημα διαχείρισης περιεχομένου το οποίο φορτώνεται σε ένα διακομιστή και παρέχει τη δυνατότητα πολλαπλών συνδέσεων χρηστών. Η ομάδα ανάπτυξης της MYSQL έφτιαξε τον κώδικα ώστε να είναι διαθέσιμος κάτω από τους όρους της GNU General Public Licence. Αυτό σημαίνει ότι είναι διαθέσιμος χωρίς κανένα περιορισμό να αντιγραφεί, να αναδιανεμηθεί τροποποιημένος ή όχι χωρίς κανένα περιορισμό. Η φιλοσοφία του ανοιχτού κώδικα έγκειται στο ότι μπορεί καθένας να συνδράμει στην ανάπτυξη του καθώς και στην αποσφαλμάτωσή του διότι παρέχονται οι αλγόριθμοι που χρησιμοποιούνται, η γλώσσα ανάπτυξης και το περιβάλλον χρήσης.

Το σύνολο των εντολών που θα χρησιμοποιούμε από δω και πέρα για να λέμε στην MySQL τι να κάνει, αποτελεί μέρος ενός standard που αποκαλείται Δομημένη Γλώσσα Ερωτημάτων (Structured Query Language) ή SQL. Οι εντολές της SQL αποκαλούνται επίσης και ερωτήματα (queries). Η SQL αποτελεί την standard γλώσσα για αλληλεπίδραση με τις περισσότερες βάσεις δεδομένων, έτσι ακόμα κι αν αλλάξουμε στο μέλλον από την MySQL σε μια βάση δεδομένων όπως την Microsoft SQL Server, θα διαπιστώσουμε ότι οι περισσότερες από τις εντολές είναι ολόιδιες. Δεν πρέπει να συγχέουμε την SQL με την MySQL. Η MySQL είναι το λογισμικό του διακομιστή βάσεων δεδομένων (database server software) που χρησιμοποιούμε, ενώ η SQL είναι η γλώσσα που χρησιμοποιούμε για να αλληλεπιδράσουμε με την βάση δεδομένων.

Η σύνδεση της MYSQL με την HTML δεν είναι δυνατή άμεσα καθώς το HTML περιεχόμενο αφορά μια σελίδα HTML ή HTM που είναι αποθηκευμένη στο web server και δεν είναι δυνατό να μετατραπεί. Αυτό το οποίο όμως θέλουμε είναι μια ανάμειξη της HTML με μια άλλη γλώσσα που θα μας επιτρέπει τη σύνδεση με μια βάση δεδομένων. Μια από τις γλώσσες αναπτύχθηκε και εξελίσσεται ταχύτατα είναι η PHP.

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερους από 50.000 ιστότοπους που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των

επισκεπτών σε μορφή κώδικα HTML. Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. \*.php, \*.php4, \*.phtml κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα PHP. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με τα λειτουργικά συστήματα GNU/Linux και Microsoft Windows, υποστηρίζει εξ ορισμού την εκτέλεση κώδικα PHP.

Ο PHP κώδικας ενσωματώνεται στις ιστοσελίδες με τη χρήση της εξής αρχικής συμβολοσειράς «<?php» και τερματίζει με τη χρήση της συμβολοσειράς «?>». Οτιδήποτε περιλαμβάνεται μεταξύ αυτών των συμβολοσειρών μεταφράζεται από την PHP σαν script της γλώσσας της και παράγει το επιθυμητό αποτέλεσμα. Εάν για παράδειγμα θέλουμε να δούμε όλες τις πληροφορίες της PHP που είναι σεταρισμένες στο web server μας το μόνο που χρειάζεται να φτιάξουμε είναι ένα αρχείο index.php με τον εξής πολύ απλό κώδικα ο οποίος μας εμφανίζει τις ρυθμίσεις που έχουν οριστεί στον web server μας για τη χρήση της PHP σκιπτογλώσσας.

## **3 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΚΑΙ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ**

Σε αυτό το κεφάλαιο παρουσιάζονται εκτενώς τα χαρακτηριστικά του συστήματος. Αρχικά γίνεται μια αναφορά στους στόχους που τέθηκαν κατά τη σχεδίαση και κατόπιν αναλύονται όλα τα υποσυστήματα του *reRSSonal*. Κατόπιν, δίνεται η σχεδιαστική προσέγγιση για την εφαρμογή των μηχανισμών συλλογής και εξαγωγής χρήσιμης πληροφορίας στο *reRSSonal*, ως επιπρόσθετα στοιχεία του.

### **3.1 Στόχοι του Συστήματος**

Ο σχεδιασμός του συστήματος έχει γίνει με τέτοιο τρόπο, ώστε να καλύπτει τις ανάγκες συλλογής και προεπεξεργασίας πληροφορίας. Η ροή πληροφορίας στο διαδίκτυο είναι χαοτική και αυτό το γεγονός δυσκολεύει το χρήστη να εντοπίσει γρήγορα την πληροφορία που θέλει. Στόχος του συστήματος είναι να δημιουργηθεί ένα πρωταρχικό επίπεδο που θα συλλέγει και θα φιλτράρει την αδόμητη πληροφορία που κυκλοφορεί στο διαδίκτυο, ώστε να φτάνει στο χρήστη χρήσιμη και σχετική με τα ενδιαφέροντα του πληροφορία. Για να επιτευχθεί αυτός ο στόχος θα πρέπει να γίνει χρήση των καναλιών επικοινωνίας και αποδοτική προεπεξεργασία των δεδομένων.

Η ανάπτυξη του μηχανισμού έγινε με απώτερο σκοπό την ενσωμάτωση του στο *reRSSonal* έτσι ώστε να αποτελέσει ένα εργαλείο για τη λήψη, επεξεργασία και παρουσίαση νέων από διάφορες ειδησεογραφικές ιστοσελίδες. Όπως έχει ήδη αναφερθεί, ο μηχανισμός είναι προορισμένος να συλλέγει δεδομένα από ιστοσελίδες ειδησεογραφικών πρακτορείων. Αυτό σημαίνει ότι για να είναι αποδοτικός θα πρέπει να ληφθούν υπ' όψιν τα ιδιαίτερα χαρακτηριστικά αυτών των δικτυακών τόπων. Το σύστημα θα πρέπει να προσεγγίζει τη διαδικασία της συλλογής πληροφορίας ως ένα πρόβλημα εύρεσης ενός ρυθμού παρακολούθησης των σελίδων, έτσι ώστε οι ειδήσεις που λαμβάνονται να θεωρούνται σχετικά επίκαιρες για το χρήστη. Δηλαδή, θα πρέπει να μπορεί να προσαρμόζεται σε κάθε πηγή ενημέρωσης ξεχωριστά, ανάλογα με τη σημαντικότητα και το πλήθος των ειδήσεων που δημοσιεύονται.

Άλλος ένας στόχος του μηχανισμού είναι η παράδοση χρήσιμης πληροφορίας στο χρήστη. Οι ιστοσελίδες, μαζί με το χρήσιμο περιεχόμενο, περιέχουν ένα σύνολο από στοιχεία που δεν ενδιαφέρουν πραγματικά το χρήστη. Αυτό συχνά δεν αποτελεί πρόβλημα όταν η περιήγηση γίνεται μέσω φυλλομετρητή, αλλά η προβολή των δεδομένων σε φορητές συσκευές χειρός δεν είναι ικανοποιητική. Παράλληλα, η ύπαρξη ανεπιθύμητων δεδομένων σε μια ιστοσελίδα λειτουργεί αρνητικά στους μηχανισμούς ανάκτησης πληροφορίας. Στοιχεύοντας στην εξαγωγή χρήσιμης πληροφορίας, θα πρέπει να σχεδιαστεί με τέτοιο τρόπο το σύστημα, ώστε να λειτουργεί αποτελεσματικά με οποιαδήποτε είσοδο.

## 3.2 Γενική αρχιτεκτονική

Η αρχιτεκτονική του συστήματος peRSSonal, στο οποίο προστέθηκαν οι μηχανισμοί συλλογής πληροφορίας και εξαγωγής χρήσιμου περιεχομένου που αναπτύχθηκαν, αποτελείται από διαφορετικά υποσυστήματα. Αυτή η προσέγγιση ακολουθήθηκε προκειμένου να είναι εύκολη η ανεξάρτητη σχεδίαση, η κατασκευή και η συντήρηση τους. Ωστόσο, η ροή της πληροφορίας όπως είναι αναμενόμενο είναι σειριακή. Αυτό σημαίνει πως παρά το γεγονός πως κάθε μηχανισμός έχει δική του γενική είσοδο και έξοδο για την πληροφορία και συνεπώς μπορεί ανά πάσα στιγμή να εκτελεστεί και να παράγει αποτελέσματα, οι μηχανισμοί εντούτοις θα πρέπει να εκτελεστούν με συγκεκριμένη σειρά προκειμένου να έχουμε άμεσα το επιθυμητό αποτέλεσμα. Το παρακάτω σχήμα μας δίνει τη γενική αρχιτεκτονική του συστήματος.

### Εικόνα 5. Γενική αρχιτεκτονική του peRSSonal

Τα διαφορετικά συστήματα που συνθέτουν το τελικό σύστημα είναι χωρισμένα σε πολλά διαφορετικά επίπεδα ανάλογα με το ρόλο που επιτελούν στη λειτουργία του συστήματος. Με αυτό τον τρόπο το σύστημα γίνεται πολυεπίπεδο, γεγονός που συντελεί στην κλιμακωτή ανάλυση και μελλοντική επέκταση της λειτουργικότητας του μηχανισμού.

## 3.3 Υποσυστήματα

Σε αυτή την ενότητα παρουσιάζονται τα υποσυστήματα του peRSSonal. Δίνεται έμφαση στο ρόλο που επιτελεί κάθε ξεχωριστός μηχανισμός και παράλληλα αναλύεται το σύστημα στα διαφορετικά επίπεδα υλοποίησης.

### □□□□ Συλλογή και εξαγωγή χρήσιμης πληροφορίας

Το πρώτο υποσύστημα του peRSSonal είναι εκείνο που τροφοδοτεί το μηχανισμό με δεδομένα. Στη συγκεκριμένη περίπτωση, πρόκειται για δεδομένα που μεταδίδονται από τις ιστοσελίδες ενημέρωσης, όπως τα portals και τα blogs. Αμέσως μόλις συλλεχθεί η πληροφορία, ακολουθεί ένα επίπεδο ανάλυσης που αναλαμβάνει να απομακρύνει δεδομένα τα οποία δεν θεωρούνται χρήσιμα. Η αρχιτεκτονική αυτού του επιπέδου θα αναλυθεί εκτενώς παρακάτω.



## Προεπεξεργασία κειμένου

Το υποσύστημα που εκτελεί την προεπεξεργασία του κειμένου έχει γενικό στόχο την απομάκρυνση ανεπιθύμητων συμβολοσειρών από το σώμα του κειμένου και την εξαγωγή λέξεις κλειδιά. Αυτή η διαδικασία βασίζεται σε συγκεκριμένη τεχνική η οποία αφορά την προεπεξεργασία κάθε είδους κειμένου προκειμένου να εξαχθούν τα keywords. Η πληροφορία που δίνεται ως είσοδος στο μηχανισμό μπορεί να προέρχεται είτε από ένα αρχείο, είτε από μια βάση δεδομένων.

Στο μηχανισμό εξαγωγής λέξεων κλειδιών υπάρχουν συγκεκριμένες μεταβλητές που ρυθμίζουν την επεξεργασία που θα πραγματοποιηθεί. Αναλυτικότερα, αυτές οι παράμετροι αφορούν:

- Το ελάχιστο μήκος λέξης (ορίζοντας ότι οι λέξεις που είναι μικρότερες σε μήκος θα αφαιρεθούν),
- Την πολιτική αποθήκευσης των αριθμητικών δεδομένων (ακέραιοι, αριθμοί κινητής υποδιαστολής κτλ),
- Τον καθορισμό ενός συνόλου από λέξεις οι οποίες θα πρέπει να αφαιρεθούν, όπως οι τετριμμένες ή οι πολύ συχνά εμφανιζόμενες λέξεις που δεν προσδίδουν περισσότερη πληροφορία στο κείμενο,
- Τον αλγόριθμο stemming που θα χρησιμοποιηθεί,
- Τον καθορισμό της βαρύτητας που θα δοθεί στις λέξεις οι οποίες έχουν αναγνωριστεί ως ουσιαστικά

Γίνεται προφανές ότι η προεπεξεργασία του κειμένου είναι μια ακολουθιακή διαδικασία. Όπως φαίνεται και στο παρακάτω σχήμα, μπορεί να αναλυθεί σε δύο επίπεδα, το επίπεδο της εφαρμογής και το επίπεδο της βάσης δεδομένων.

### Εικόνα 6. Προεπεξεργασία κειμένου και εξαγωγή λέξεων-κλειδιών

Το πρώτο στάδιο της διαδικασίας περιλαμβάνει την αναγνώριση της γλώσσας του κειμένου και γίνεται αξιοποιώντας την πληροφορία του RSS feed που προήλθε το κείμενο. Στο επόμενο βήμα γίνεται ορθογραφικός έλεγχος και χωρισμός των προτάσεων και λέξεων, αφαιρώντας παράλληλα τα σημεία στίξης. Ακολουθεί η διαδικασία αντιστοίχισης των λέξεων σε κάποιο μέρος του λόγου, αξιοποιώντας κάποιον Part-of-speech tagger. Είναι φανερό ότι αρκετές από τις λέξεις του κειμένου δεν προσφέρουν σημαντική πληροφορία στο κείμενο και είναι προτιμότερο να απομακρυνθούν για λόγους καλύτερης αναπαράστασης του κειμένου. Σε αυτές τις λέξεις ανήκουν οι υπερβολικά συχνά εμφανιζόμενες λέξεις, όπως τα άρθρα κτλ. Μετά την εύρεση των ουσιαστικών, πραγματοποιείται η διαδικασία αποκατάληξης (stemming), προκειμένου να ομαδοποιηθούν λέξεις που είναι νοηματικά «κοντά». Είναι προφανές ότι η αναγνώριση ουσιαστικών, η δημιουργία του συνόλου των

λέξεων που πρέπει να αφαιρεθούν και η αποκατάληξη, είναι διαδικασίες που εξαρτώνται από τη γλώσσα στην οποία είναι γραμμένο το κείμενο .

Η διασύνδεση του μηχανισμού προεπεξεργασίας είναι δυνατόν να διασυνδεθεί εύκολα με τα υπόλοιπα συστήματα, μιας και η έξοδος του μπορεί να αποθηκευτεί είτε σε μια βάση δεδομένων, είτε σε ένα XML αρχείο. Τα στοιχεία που περιλαμβάνονται είναι τα εξής:

- Οι τελικές λέξεις κλειδιά
- Η θέση της πρότασης στην οποία ανήκει κάθε λέξη κλειδί (ως θέση χρησιμοποιείται η σχετική θέση της πρότασης)
- Την απόλυτη συχνότητα εμφάνισης της κάθε λέξης κλειδί μέσα στο κείμενο, δηλαδή πόσες φορές εμφανίζεται η λέξη
- Την σχετική συχνότητα εμφάνισης της κάθε λέξης κλειδί μέσα στο κείμενο
- Το αποτέλεσμα της διαδικασίας αναγνώρισης του μέρους του λόγου της λέξης

## □□□□ □ Κατηγοριοποίηση

Η κατηγοριοποίηση του κειμένου αποτελεί μια σημαντική διαδικασία του πυρήνα του `peRSSonal` . Ο ρόλος αυτού του επιπέδου είναι να κατηγοριοποιεί τα άρθρα που έχουν ανακτηθεί, σύμφωνα με πληροφορία που υπάρχει στη βάση δεδομένων. Τα δεδομένα που λαμβάνονται ως είσοδος, είναι οι λέξεις κλειδιά και οι συχνότητες τους στο άρθρο, όπως αυτά εξάγονται από το προηγούμενο στάδιο προεπεξεργασίας. Στο παρακάτω σχήμα φαίνεται η αρχιτεκτονική αυτού του υποσυστήματος.

### **Εικόνα 7. Διαδικασία κατηγοριοποίησης του κειμένου**

Η λειτουργία του μηχανισμού κατηγοριοποίησης κειμένου στηρίζεται σε υπάρχουσες κατηγορίες και μια σειρά από αλγορίθμους αντιστοίχισης του άρθρου με μια από τις κατηγορίες. Για να εκτελεστεί αποτελεσματικά αυτή η διαδικασία θα πρέπει να υπάρχει η κατάλληλη πληροφορία στη βάση δεδομένων για κάθε κατηγορία. Όπως αναφέρθηκε και προηγουμένως, τα άρθρα που φτάνουν ως είσοδος σε αυτό το στάδιο, αναπαριστώνται από τις λέξεις κλειδιά τους και τις αντίστοιχες συχνότητες εμφάνισης. Η κατηγοριοποίηση θα χρειαστεί να βρει την καταλληλότερη κατηγορία, βάσει αυτών των λέξεων κλειδιών. Γίνεται προφανές ότι η αναγκαία πληροφορία που πρέπει να υπάρχει στη βάση δεδομένων είναι ένα σύνολο από αντιπροσωπευτικές λέξεις κλειδιά για κάθε κατηγορία, οι οποίες προκύπτουν ύστερα από την ανάλυση μερικών άρθρων εκπαίδευσης (training set), γνωρίζοντας εξ' αρχής τις κατηγορίες που ανήκουν.

Για την αντιστοίχιση ενός νέου άρθρου σε κάποια κατηγορία, αξιοποιείται μια μετρική σύγκρισης μεταξύ των keywords του άρθρου και της κάθε κατηγορίας. Συγκεκριμένα, χρησιμοποιείται η μετρική ομοιότητας συνημίτονου που φαίνεται στην παρακάτω εξίσωση:

$$\text{similarity}(A, B) = \frac{A \bullet B}{\|A\| \|B\|} \quad (1)$$

Όπου A είναι το διάνυσμα αναπαράστασης του άρθρου και B το διάνυσμα μιας κατηγορίας. Το αποτέλεσμα της σύγκρισης μπορεί να κυμαίνεται από την τιμή 0 μέχρι 1 και εκφράζει το πόσο «κοντά» είναι ένα άρθρο σε μια κατηγορία. Η αποτελεσματικότητα αυτής της μεθόδου στηρίζεται στην ποιότητα του training set, αφού είναι προφανές ότι αν για μια κατηγορία έχουν εισαχθεί μη σχετικές λέξεις κλειδιά, τότε η μετρική ομοιότητας δε θα λειτουργήσει ικανοποιητικά.

## □ □ □ □ □ **Εξαγωγή περίληψης**

Η εξαγωγή περίληψης μπορεί να θεωρηθεί ως μια δύσκολη διαδικασία με περιορισμένη αποτελεσματικότητα, ακόμα και όταν γίνεται μη αυτοματοποιημένα. Οι πιο διαδεδομένοι τρόποι εξαγωγής που ακολουθούνται είναι δύο. Ο πρώτος αναλύει νοηματικά το κείμενο και συνθέτει δικές του προτάσεις χρησιμοποιώντας μεθόδους NLP. Ο δεύτερος, που χρησιμοποιείται και στο `reSSonal`, διαλέγει ένα σύνολο από αντιπροσωπευτικές προτάσεις από το άρθρο ως περίληψη. Η εξαγωγή μπορεί να προηγηθεί της κατηγοριοποίησης, αφού δεν υπάρχουν αλληλοεξαρτήσεις μεταξύ τους.

Το υποσύστημα εξαγωγής περίληψης δέχεται και αυτό ως είσοδο την αναπαράσταση του άρθρου με τις λέξεις κλειδιά του, συνοδευόμενα από τις συχνότητες εμφάνισης. Ακόμα, αξιοποιούνται πληροφορίες όπως η θέση των λέξεων μέσα στο άρθρο και ο χαρακτηρισμός κάθε λέξης ως κάποιο μέρος του λόγου, μιας και ενδιαφερόμαστε κυρίως για τα ουσιαστικά. Η εξαγωγή μπορεί να γίνει πιο αποδοτική όταν δοθεί ο τίτλος του άρθρου και το επιθυμητό πλήθος προτάσεων που θα συνθέτουν την περίληψη. Η αρχιτεκτονική του μηχανισμού, αποτυπώνεται στο παρακάτω σχήμα:

### **Εικόνα 8. Διαδικασία εξαγωγής περίληψης**

Αξίζει να σημειωθεί ότι το πλήθος των προτάσεων μπορεί να καθοριστεί είτε ως ποσοστό επί τοις εκατό των προτάσεων του αρχικού κειμένου είτε ως συνολικό πλήθος χαρακτήρων. Στην περίπτωση που το αρχικό κείμενο είχε 20 προτάσεις και επιλέξουμε να κρατάμε το 30% των προτάσεων, στην περίληψη θα κρατηθούν οι 6 σημαντικότερες προτάσεις του κειμένου. Αντίθετα, εάν επιθυμούμε η περίληψη του κειμένου να έχει μήκος περίπου ένα συγκεκριμένο πλήθος χαρακτήρων, θα επιλεγθούν τόσες προτάσεις από τις σημαντικότερες ώστε και να καλύπτεται το μήκος που τέθηκε και να μην ξεπερνιέται κατά πολύ αυτό. Δηλαδή, επιλέγουμε μόνο τόσες προτάσεις ώστε η περίληψη να ξεπερνά οριακά το επιθυμητό μήκος.

Η επιλογή των πιο σημαντικών προτάσεων, που τελικά θα αποτελέσουν και την περίληψη, γίνεται με έναν αλγόριθμο αξιολόγησης. Η σημαντικότητα μιας πρότασης εκφράζει την ικανότητα της να αναπαραστήσει το νόημα του κειμένου και βασίζεται στις λέξεις κλειδιά που περιέχονται. Ο αλγόριθμος αξιολόγησης θέτει τα εξής ερωτήματα:

- υπάρχει το keyword και στον τίτλο του κείμενου;
- υπάρχει πληροφορία για την κατηγορία που ανήκει το κείμενο;
- το keyword είναι ουσιαστικό;
- υπάρχει πληροφορία για τις προτιμήσεις του χρήστη σε κατηγορία ή keywords;

Η έξοδος αποτελείται από τις προτάσεις του κειμένου, ταξινομημένες σε φθίνουσα σειρά, βάσει του αποτελέσματος της αξιολόγησης. Είναι προφανές ότι όσο καλύτερα ζυγιστούν οι παράμετροι της αξιολόγησης, τόσο πιο αξιόπιστη θα είναι η απόφαση για το αν μια πρόταση θεωρείται σημαντική και πρέπει να συμπεριληφθεί στην περίληψη.

Το τελευταίο βήμα του μηχανισμού είναι η κατάταξη των προτάσεων, δηλαδή η επιλογή της σειράς εμφάνισης μέσα στην περίληψη. Μετά το πέρας της αξιολόγησης των προτάσεων, κάποια πρόταση που εμφανίζεται αργότερα από κάποια άλλη, μπορεί να έχει βρεθεί ότι είναι πιο σημαντική. Αν η περίληψη συνταχθεί με τις προτάσεις ταξινομημένες σε φθίνουσα σειρά βάσει της αξιολόγησης, τότε είναι πολύ πιθανό να μη βγαίνει νόημα. Επομένως, προτιμάται να επιλέγονται οι πιο σημαντικές προτάσεις αλλά τελικά να ταξινομούνται με βάση τη θέση τους στο πραγματικό κείμενο.

## □□□□ Παρουσίαση τελικής πληροφορίας και προσωποποίηση

Η παρουσίαση της πληροφορίας αποτελεί το τελικό στάδιο του γενικού μηχανισμού. Η λειτουργία αυτού του υποσυστήματος προϋποθέτει ότι έχουν εκτελεστεί επιτυχώς οι διαδικασίες προεπεξεργασίας, κατηγοριοποίησης και εξαγωγής περίληψης των άρθρων που έχουν συλλεχθεί. Πρόκειται για ένα πολύ σημαντικό στάδιο του συστήματος καθώς αποτελεί το περιβάλλον διεπαφής με τους χρήστες. Στόχος είναι ο χρήστης να μην αντιλαμβάνεται όλες τις διεργασίες που λαμβάνουν χώρα και να απολαμβάνει ποιοτικά και γρήγορα αποτελέσματα βάση των προσωπικών του επιλογών. Η αρχιτεκτονική του υποσυστήματος, φαίνεται στο σχήμα που ακολουθεί.

### **Εικόνα 9. Αρχιτεκτονική του συστήματος παρουσίασης πληροφορίας**

Εκτός από την παρουσίαση της τελικής πληροφορίας, σε αυτό το στάδιο πραγματοποιούνται και οι διαδικασίες προσωποποίησης. Για την προσωποποίηση στο χρήστη (user profile) μπορούν να χρησιμοποιηθούν δύο μέθοδοι:

- Ο χρήστης να δώσει κάποια πληροφορία στο σύστημα και το σύστημα να ξεκινήσει παρουσιάζοντας εξ αρχής προσωποποιημένα αποτελέσματα και να συγκλίνει γρήγορα στις πληροφοριακές ανάγκες του χρήστη.
- Ο χρήστης να μη δώσει καθόλου πληροφορία στο σύστημα, το σύστημα να ξεκινήσει παρουσιάζοντας γενικές πληροφορίες και να αργήσει να συγκλίνει στις προσωπικές επιλογές του χρήστη.

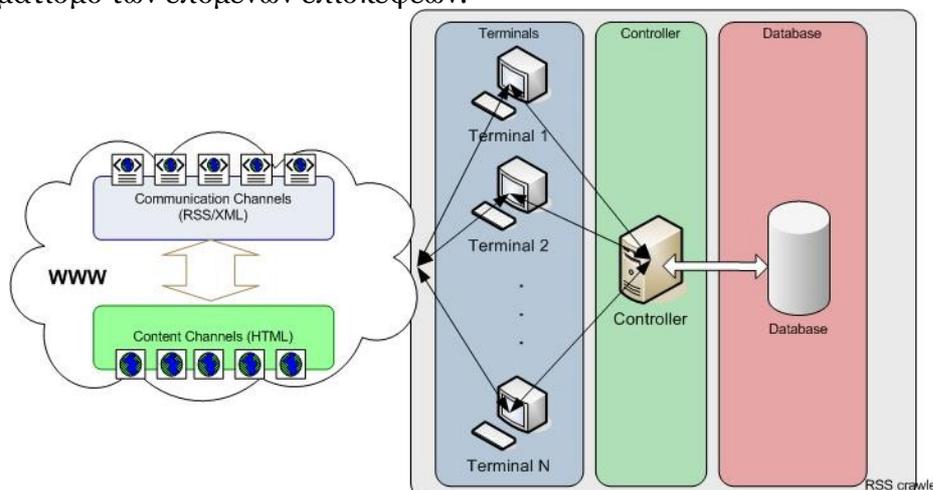
Και με τις δύο μεθόδους, το σύστημα τελικά προσαρμόζεται στα ενδιαφέροντα του χρήστη, διατηρώντας τη δυνατότητα σταδιακής αναπροσαρμογής σε νέες πληροφοριακές ανάγκες.

Η λειτουργία του μηχανισμού βασίζεται στην συνεχή ανάδραση και δεν είναι στατικός ο τρόπος παρουσίασης των δεδομένων στον τελικό χρήστη. Με την είσοδο του εγγεγραμμένου χρήστη στο σύστημα, εμφανίζεται κάποια πληροφορία η οποία είναι προσωποποιημένη ανάλογα με τις επιλογές που αυτός έχει κάνει. Αρχικά, εμφανίζονται άρθρα από τις κατηγορίες που έχει επιλέξει ενώ άρθρα από κατηγορίες που δεν έχει επιλέξει δε συμπεριλαμβάνονται καθόλου στα πρώτα αποτελέσματα, αφού δε θεωρούνται ότι ενδιαφέρουν το χρήστη. Κάθε ενέργεια που πραγματοποιεί στο δικτυακό τόπο, καταγράφεται με στόχο να επιτευχθεί καλύτερη ανάδραση, αφού έτσι θα μπορεί να διαμορφώνεται καλύτερα το προφίλ του χρήστη.

### **3.4 Αρχιτεκτονική του υποσυστήματος συλλογής πληροφορίας**

Ο μηχανισμός συλλογής πληροφορίας adnaRSS αποτελεί το πρώτο επίπεδο του reRSSonal. Πρόκειται ουσιαστικά για έναν προσαρμοσμένο crawler, που λαμβάνει ως είσοδο διευθύνσεις καναλιών επικοινωνίας (RSS feeds). Στόχος του είναι η

διατήρηση ενός επίκαιρου αντιγράφου της πληροφορίας που παρέχει κάποιο RSS, με στόχο την ανάκτηση περαιτέρω δεδομένων από το διαδίκτυο όταν προκύψει κάποια ενημέρωση. Ο μηχανισμός είναι σχεδιασμένος για να επεξεργάζεται κανάλια επικοινωνίας ειδησεογραφικών πηγών. Αυτό σημαίνει ότι η πληροφορία που ανακτάται είναι τελικά μια συλλογή από ειδήσεις και άρθρα. Η διαφορά με τους συνηθισμένους crawlers εντοπίζεται στο γεγονός ότι ο advaRSS δεν κατεβάζει απλά αλλά και αναλύει ιστότοπους προκειμένου να εντοπίσει αλλαγές και να βελτιώσει τον προγραμματισμό των επόμενων επισκέψεων.

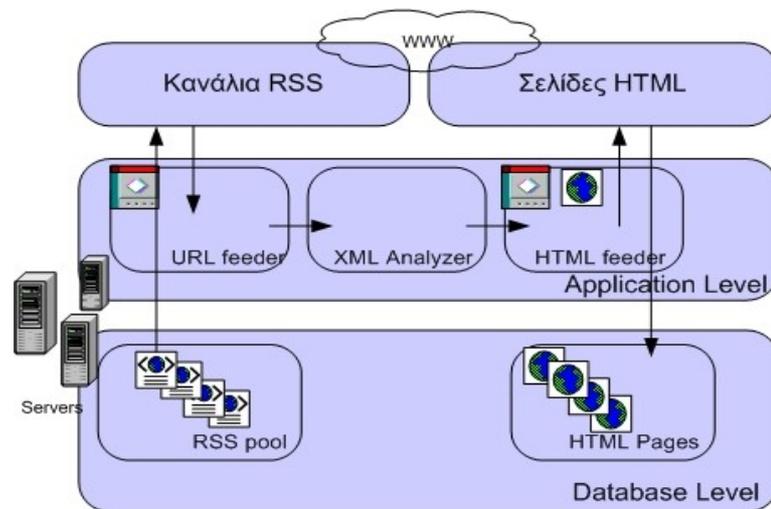


**Εικόνα 10. Αρχιτεκτονική του advaRSS**

Στην καρδιά του μηχανισμού βρίσκεται ένας κεντρικός συντονιστής (controller). Αυτός έχει όλη την εποπτεία της διαδικασίας που πραγματοποιείται και έχει κύριο ρόλο την ανάθεση εργασιών στα διαφορετικά τερματικά του συστήματος. Τα τερματικά είναι προγραμματισμένα να εκτελούν δυο ειδών ανακτήσεων, ανάκτηση του RSS και ανάκτηση του HTML κώδικα μιας σελίδας. Η πληροφορία αρχικά αποθηκεύεται στην τοπική μνήμη του τερματικού και μετέπειτα μεταφέρεται στην κεντρική βάση δεδομένων.

Ένα στοιχείο που διαχωρίζει τον advaRSS από τους υπόλοιπους crawlers είναι το γεγονός ότι τα αρχικά URLs προέρχονται αποκλειστικά από τα κανάλια επικοινωνίας που έχουν επιλεγεί να παρακολουθούνται και εν συνεχεία αυτά χρησιμοποιούνται για να καθοδηγήσουν την υπόλοιπη εκτέλεση. Στην ουσία πρόκειται για έναν mixed crawler καθότι από τη μία διαθέτει αρχικό feed URL το οποίο είναι αποκλειστικά και μόνον feeds ενώ η πληροφορία που εξάγει από αυτά, επανέρχεται σαν είσοδος στο σύστημα το οποίο κατεβάζει τις HTML σελίδες που χρειάζονται. Ταυτόχρονα, μπορούμε να τον χαρακτηρίσουμε και ως καταναμημένο crawler, αφού οι διεργασίες οργάνωσης και ανάκτησης πραγματοποιούνται από διαφορετικά υπολογιστικά συστήματα.

Αναλύοντας τη ροή πληροφορίας του μηχανισμού, βλέπουμε ότι σε πρώτη φάση δέχεται σαν είσοδο κανάλια επικοινωνίας από το RSS pool. Ύστερα, με την ανάλυση του XML κώδικα των feeds προκύπτουν link προς σελίδες οι οποίες περιέχουν άρθρα που δεν έχει προσπελάσει ακόμα το σύστημα. Στη συνέχεια το σύστημα λαμβάνει τον HTML κώδικα των σελίδων και αναλαμβάνει να τον αποθηκεύσει στη βάση δεδομένων. Η ροή της πληροφορίας στο υποσύστημα που περιγράφεται, μπορεί να αποτυπωθεί από την παρακάτω εικόνα.



**Εικόνα 11. Ροή πληροφορίας στο advaRSS**

Η διαδικασία της διατήρησης επίκαιρων αντιγράφων, δεν είναι πάντα απλή, καθώς ο μηχανισμός έχει να αντιμετωπίσει μία σειρά από σημαντικά ζητήματα στην προσπάθεια ανάκτησης άρθρων από σελίδες του διαδικτύου. Τα βασικά προβλήματα είναι:

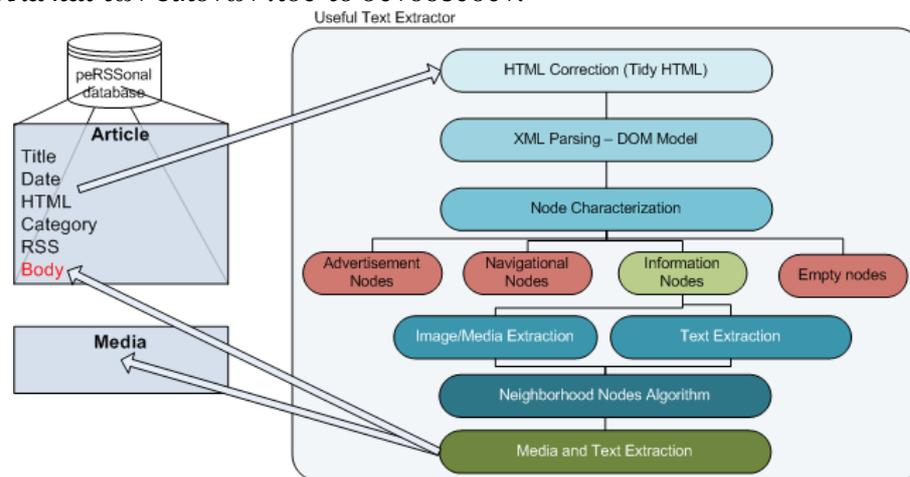
- Οι ειδησεογραφικές σελίδες του διαδικτύου, είναι δυνατόν να δημοσιεύουν άρθρα οποιαδήποτε στιγμή. Για να είναι αποδοτικός ο μηχανισμός, θα πρέπει να μένει πάντα ενημερωμένος με τα τελευταία άρθρα.
- Η είσοδος μπορεί να είναι πολύ μεγάλη με αποτέλεσμα να πρέπει να υπάρχει κάποια προτεραιότητα στα RSS feeds που λαμβάνει ο μηχανισμός.
- Θα πρέπει να υπάρχει κάποια φιλικότητα προς τους ιστότοπους που προσπελαύνονται, με την έννοια ότι δε θα πρέπει ο μηχανισμός να επισκέπτεται υπερβολικά συχνά συγκεκριμένες σελίδες.
- Ο ρυθμός δημοσίευσης άρθρων διαφέρει ανάλογα με την εκάστοτε ιστοσελίδα. Δηλαδή, διαφορετικές πύλες ενημέρωσης του διαδικτύου φαίνεται να έχουν διαφορετικές ώρες κατά τις οποίες αναρτούν τον κυριότερο όγκο των ειδήσεων τους.

Συνοψίζοντας, με βάση τα παραπάνω, οδηγούμαστε στο γεγονός πως θα πρέπει να εφαρμόζεται μία συγκεκριμένη πολιτική αναφορικά με τον σχεδιασμό των επισκέψεων στα RSS feeds, με στόχο να αυξήσουμε την αποδοτικότητα του συστήματός μας και να μειώσουμε την επιβάρυνση στους πόρους του δικτύου, των τερματικών και του απομακρυσμένου server. Ταυτόχρονα, όλα αυτά συνεισφέρουν μειωμένο χρόνο ανάκτησης άρθρου από τη στιγμή που αυτό προέκυψε.

Το υποσύστημα συλλογής ειδησεογραφικών σελίδων, εξάγει και άλλη πληροφορία που είναι πολύ χρήσιμη και για τα υπόλοιπα επίπεδα. Εκτός από τον HTML κώδικα μίας σελίδας, ο μηχανισμός είναι σε θέση να κάνει εξαγωγή του τίτλου ενός άρθρου ξεχωριστά από το κυρίως κείμενο, της ημερομηνίας ανάρτησης του άρθρου και τη γλώσσα στην οποία είναι γραμμένο το άρθρο. Ακόμα, παρέχει κάποια μεταδεδομένα που υπάρχουν ήδη αποθηκευμένα στο σύστημα όπως για παράδειγμα η κατηγορία στην οποία ανήκει το άρθρο (πολιτική, οικονομία, κ.α.). Αυτά τα δεδομένα σχετίζονται άμεσα με το κανάλι επικοινωνίας που προσπελαύνεται και κληρονομούνται από τα άρθρα του.

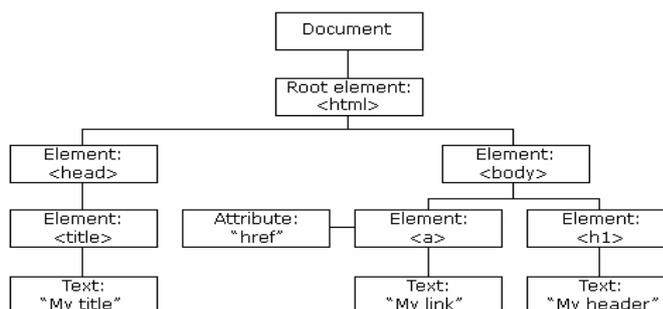
### 3.5 Αρχιτεκτονική του υποσυστήματος εξαγωγής περιεχομένου

Το CUTER αποτελεί το μηχανισμό εξαγωγής χρήσιμου κειμένου που χρησιμοποιείται στο σύστημα peRSSonal. Σαν χρήσιμο κείμενο εννοούμε αποκλειστικά και μόνο το κείμενο ενός άρθρου μαζί με ότι εικόνες το συνοδεύουν. Αρχικά το σύστημα είχε δημιουργηθεί για να εξάγει κείμενο, ωστόσο στην πορεία εμπλουτίστηκε για να μπορεί να εξάγει και όλες τις εικόνες που ακολουθούν το κείμενο και στη συνέχεια να μπορεί να επεκταθεί ώστε να βρίσκει κάθε πολυμεσικό περιεχόμενο που σχετίζεται με το ίδιο το άρθρο. Το CUTER είναι ένα εργαλείο που μπορεί να λειτουργεί αυτόνομα, και ανεξάρτητα από το όλο σύστημα, θεωρώντας δεδομένο ότι υπάρχει κάποια πληροφορία στη βάση δεδομένων. Όπως προαναφέρθηκε, το σύστημα που τροφοδοτεί το μηχανισμό εξαγωγής κειμένου, είναι συνήθως ένας crawler, με έξοδο τον HTML κώδικα των άρθρων. Στην πορεία πραγματοποιούνται μία σειρά από διαδικασίες και συστήματα που είναι ενσωματωμένα στο μηχανισμό προκειμένου να γίνει εξαγωγή του κειμένου του άρθρου αλλά και των εικόνων που το συνοδεύουν.



Εικόνα 12. Η αρχιτεκτονική της εξαγωγής κειμένου

Όπως φαίνεται στην εικόνα 2, η πρώτη μονάδα του μηχανισμού λαμβάνει τον HTML κώδικα, που αναμένεται να έχει συντακτικές ασυνέπειες και εφαρμόζει τεχνικές για την διόρθωση και τη μετατροπή του σε XML. Οι συντακτικές ασυνέπειες στις υλοποιήσεις ιστοσελίδων είναι ένα πάρα πολύ συχνό φαινόμενο, εξαιτίας της ίδιας της φύσης του HTML κώδικα ο οποίος μπορεί να εμφανίσει σωστά μια ιστοσελίδα ακόμα και αν δεν είναι συντακτικά ολόσωστος. Αυτό το γεγονός διευκολύνει μεν τους σχεδιαστές σελίδων του διαδικτύου, αλλά δυσχαιρένει το έργο των μηχανισμών ανάκτησης πληροφορίας από αυτές. Η μετατροπή σε XML γίνεται για να είναι εφικτή η κατασκευή του DOM δέντρου, στο οποίο θα παριστάνεται κάθε ετικέτα του HTML κώδικα ως ένας κόμβος. Για την παραπάνω διαδικασία χρησιμοποιήθηκε το Tidy Open Source Library Project . Πρόκειται για τη βιβλιοθήκη του Tidy HTML validator που είναι κατασκευασμένος για να εντοπίζει και να διορθώνει σφάλματα που μπορεί να υπάρχουν στις σελίδες. Αυτό το εργαλείο χρησιμοποιείται περισσότερο για να υποβοηθήσει τους κατασκευαστές δικτυακών τόπων στην εύρεση σφαλμάτων κατά τη διάρκεια της ανάπτυξης της ιστοσελίδας, αλλά στην περίπτωσή μας η χρήση του γίνεται αυτόματα από το μηχανισμό Το τελικό DOM δέντρο θα είναι παρόμοιο με αυτό που εμφανίζεται στην παρακάτω σχηματική αναπαράσταση.



**Εικόνα 13. DOM δέντρο ιστοσελίδας**

Μετά την κατασκευή του DOM δέντρου, ακολουθεί η λειτουργία ανάλυσης του, με στόχο την εύρεση των κόμβων που περιέχουν τη χρήσιμη πληροφορία. Κάθε κόμβος είναι ένα τμήμα της σελίδας που τις περισσότερες φορές οπτικοποιείται ξεχωριστά από τα υπόλοιπα τμήματα στο φυλλομετρητή. Στην πραγματικότητα τα δεδομένα που μας ενδιαφέρουν (κείμενο και πολυμέσα) αποθηκεύονται μόνο στα φύλλα του DOM δέντρου. Στη συνέχεια και ακολουθώντας έναν απλό αλγόριθμο πραγματοποιούμε χαρακτηρισμό όλων των κόμβων του δέντρου. Σύμφωνα με το μοντέλο μας οι κόμβοι μπορεί να είναι:

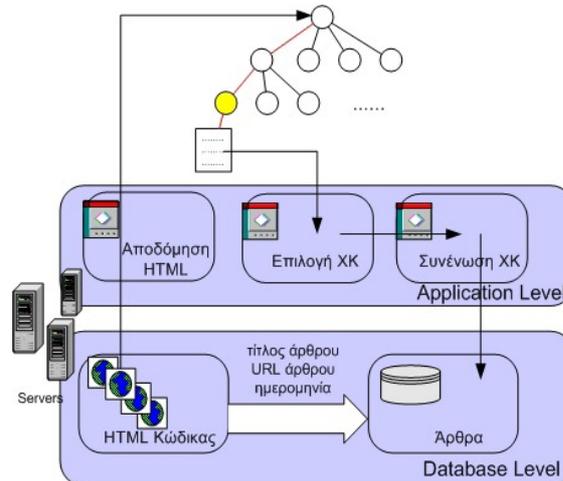
1. κόμβοι διαφήμισης
2. πλοήγησης
3. πληροφορίας
4. κενοί

Η πρώτη κατηγορία κόμβων είναι οι κόμβοι διαφήμισης, δηλαδή τμήματα της σελίδας που περιέχουν κάποια διαφημιστική πληροφορία. Όπως είναι προφανές, αυτή η πληροφορία είναι τις περισσότερες φορές άσχετη με το νοηματικό περιεχόμενο του άρθρου και τελικά ανεπιθύμητη από τον αναγνώστη. Η ανίχνευση αυτών των κόμβων είναι μια δύσκολη διαδικασία, αν αναλογιστούμε ότι πλέον οι διαφημίσεις ενσωματώνονται σε οποιαδήποτε θέση της σελίδας, ακόμα και ανάμεσα στο σώμα του άρθρου.

Στους κόμβους πλοήγησης συμπεριλαμβάνονται εκείνοι που έχουν στόχο τη διευκόλυνση της περιήγησης του χρήστη στη σελίδα. Αυτά τα τμήματα των ιστοσελίδων τείνουν να αυξάνονται με το χρόνο, αφού η πληροφορία στους δικτυακούς τόπους ολοένα και διογκώνεται. Έτσι δημιουργείται άμεσα η ανάγκη για ευκολότερη και ταχύτερη πρόσβαση στην επιθυμητή πληροφορία, αφού ο χρήστης δε διαθέτει χρόνο για περιηγηθεί σε μια χαοτική ιστοσελίδα. Οι κόμβοι πλοήγησης περιέχουν, στην πλειοψηφία τους, συνδέσμους προς το ίδιο το site ή και συνδέσμους προς άλλους δικτυακούς τόπους. Αυτό το χαρακτηριστικό τους, ότι δηλαδή περιέχουν μεγάλο αριθμό από συνδέσμους, βοηθά στην αναγνώριση τους από το σύστημα.

Οι κόμβοι που θα χρησιμοποιηθούν για την εξαγωγή κειμένου θα είναι αυτοί που εντάσσονται στην κατηγορία «κόμβος πληροφορίας» (3). Αυτοί αποτελούν το πιο σημαντικό σύνολο των κόμβων του μηχανισμού καθότι από αυτούς εξάγεται το χρήσιμο κείμενο. Πρόκειται για κόμβους με φύλλα τα οποία έχουν κείμενο (μορφοποιημένο ή μη) ή ακόμα και πολυμέσα. Είναι προφανές ότι και οι κόμβοι που περιέχουν το σώμα του άρθρου θα εντάσσονται σε αυτή την κατηγορία κόμβων, αφού περιέχουν την επιθυμητή πληροφορία. Ο εύρεση των κόμβων πληροφορίας υποβοηθά το μηχανισμό, αφού με τη χρήση των αλγορίθμων που ακολουθούν είναι δυνατόν να εξεταστούν τα χαρακτηριστικά των συγκεκριμένων κόμβων και να επιλεγούν εκείνα που τελικά θα συνθέτουν το περιεχόμενο που θα εξαχθεί. Παράλληλα, σε αυτά συνεπικουρεί και το γεγονός ότι έχουμε γνώση του σημείου έναρξης του άρθρου καθότι από τον crawler και το RSS feed έχουμε ήδη την πληροφορία για τον ακριβή τίτλο του άρθρου.

Τέλος, οι άδαιοι κόμβοι είναι επίσης ένα κομμάτι το οποίο παρατηρείται στις μοντέρνες σελίδες και αξίζουν αναφοράς διότι οι άδαιοι κόμβοι αν και φαίνονται παράλογο να υπάρχουν, συμβάλλουν καθοριστικά στην εμφάνιση της σελίδας. Ένας λόγος που συντελεί στη δημιουργία άδειων κόμβων είναι και η διατήρηση της συντακτικής συνέπειας της σελίδας. Για παράδειγμα η υλοποίηση ενός πίνακα, όπου κάθε κελί είναι και ένας κόμβος στο DOM δέντρο, είναι πολύ πιθανό να δημιουργεί κενούς κόμβους σε εκείνα τα κελιά στα οποία δεν υπάρχουν δεδομένα. Η χρησιμότητα του χαρακτηρισμού των κόμβων που αναφέρθηκαν, αποτυπώνεται στο παρακάτω σχήμα.



**Εικόνα 14. Χαρακτηρισμός και επιλογή κόμβων**

Αυτή ήταν η περιγραφή της αρχιτεκτονικής και του τρόπου λειτουργίας του συστήματος peRSSonal. Η εφαρμογή και οι κατάλληλες μετατροπές του συστήματος αυτού για την εφαρμογή του στο Joomla θα αναλυθούν στο κεφάλαιο 5 μαζί με την ανάλυση του Joomla.

## **4 ΤΟ WEB ΚΑΙ ΤΑ ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ**

Σε αυτό το κεφάλαιο παρουσιάζονται εκτενώς τα χαρακτηριστικά των συστημάτων διαχείρισης περιεχομένου, πως δημιουργήθηκαν και τι οδήγησε σε αυτά. Όλα τα εργαλεία που περιγράφηκαν στο 2<sup>ο</sup> κεφάλαιο αποτελούν κυρίαρχα στοιχεία των CMS. Χάρη σε αυτά τους δίνεται δυναμικότητα ως προς τον τρόπο διαχείρισης του περιεχομένου τους και τρομερή ευελιξία. Στις επόμενες ενότητες αναλύουμε περεταίρω τα χαρακτηριστικά τους, τις κατηγορίες τους και τα σημαντικότερα πλεονεκτήματά τους, ώστε ο αναγνώστης να αντιληφθεί γιατί η δημιουργία δικτυακού τύπου μέσω ενός CMS αποτελεί πλέον μονόδρομο για το διαδίκτυο.

### **4.1 Τι είναι τα συστήματα διαχείρισης περιεχομένου**

Τα CMS (Content Management Systems) είναι συστήματα διαχείρισης περιεχομένου που υποστηρίζουν τη δημιουργία, τη διαχείριση, τη διάθεση, την έκδοση και την ανακάλυψη εταιρικών πληροφοριών. Υπάρχουν 3 βασικές κατηγορίες για τα CMS και αυτές είναι:

- Enterprise CMS
- Web SMS
- Component CMS

#### **□□□□ Enterprise CMS**

Το enterprise CMS αναφέρεται σε τεχνολογίες, στρατηγικές, μεθόδους και εργαλεία που χρησιμοποιούνται για τη συλλογή, διαχείριση, αποθήκευση, διατήρηση και παράδοση περιεχομένων και των εγγράφων τους που σχετίζονται με έναν οργανισμό και τις υπηρεσίες του. Άρα αφορούν τη χρήση τους περισσότερο από οργανισμούς.

#### **□□□□ Component CMS**

Το component CMS διαχειρίζεται τα περιεχόμενα σε ένα σπυρωτό επίπεδο περιεχομένων παρά σε επίπεδο εγγράφου. Κάθε περιεχόμενο παρουσιάζει ένα μεμονωμένο θέμα, έννοια ή ένα απόκτημα. Τα συστατικά συγκεντρώνονται σε πολλαπλά περιεχόμενα και μπορούν να αναπαρασταθούν ψηφιακά ή παραδοσιακά έγγραφα. Κάθε συστατικό έχει το δικό του κύκλο ζωής και εντοπίζεται μεμονωμένα. Τα CCMS χρησιμοποιούνται κατά κόρον από πολλαπλά κανάλια όσον αφορά την επεξεργασία περιεχομένων στο επίπεδο του πελάτη.

## 4.2 Web CMS

Τα συστήματα αυτά αφορούν εφαρμογές που χρησιμοποιούνται για να δημιουργήσουν, επεξεργαστούν, διαχειριστούν και δημοσιεύσουν ιστοσελίδες στο διαδίκτυο. Οι δικτυακοί τόποι μπορεί να είναι:

- Εταιρικοί
- Προσωπικοί
- Εκπαιδευτικοί
- Ενημερωτικοί
- Ηλεκτρονικά καταστήματα
- Ψυχαγωγικοί
- Κοινωνικής δικτύωσης

Σχεδόν όλα τα είδη ιστοσελίδων μπορούν εν δυνάμει να καλυφθούν από τη χρήση των web CMS, περιλαμβάνοντας κείμενα, εικόνες, ήχο, βίντεο, ψηφιακά έγγραφα, animation και οτιδήποτε μπορεί να εμφανίζεται στο διαδίκτυο.

Για να είναι εύχρηστο και λειτουργικό ένα Web CMS θα πρέπει να υποστηρίζει:

- Εύκολη διαχείριση του περιεχομένου τους με τη χρήση μόνο ενός web browser.
- Δυνατότητα διαχωρισμού των χρηστών σε επίπεδα και απόδοσης διαφορετικών ρόλων σε αυτούς (διαχειριστής, συντάκτης κειμένου, απλός χρήστης, επισκέπτης κτλ).
- Δημοσίευση περιεχομένου ύστερα από έγκριση του διαχειριστή.
- Κατηγοριοποίηση του περιεχομένου για την ευκολότερη διαχείρισή του.
- Διαχωρισμός του περιεχομένου και εμφάνισης . Για παράδειγμα η γραμματοσειρά θα πρέπει να μπορεί να εφαρμοστεί σε όλες τις σελίδες με μία και μόνο αλλαγή.

Στη συνέχεια αναφέρουμε τα χαρακτηριστικά ενός CMS και τις δυνατότητες που παρέχει στο χρήστη.

- Δίνει τη δυνατότητα στο χρήστη να επικεντρωθεί στο περιεχόμενο της ιστοσελίδας του και όχι στην τεχνολογία και στον προγραμματισμό που απαιτείται για την εμφάνισή της.
- Παρέχεται η δυνατότητα λοιπόν συντήρησης του δικτυακού τόπου ακόμα και από άτομα χωρίς το κατάλληλο τεχνικό υπόβαθρο και την εξειδικευμένη τεχνογνωσία που εμπεριέχεται στα CMS.
- Η ενημέρωση του δικτυακού τόπου γίνεται άμεσα και από πολλούς χρήστες. Ο κάθε χρήστης ωστόσο έχει τη δυνατότητα να παρακολουθεί ταυτόχρονα τα δεδομένα που εισάγουν άλλοι χρήστες.
- Παρέχει τη δυνατότητα δημιουργίας μενού, υπομενού και εφαρμόζει μορφοποιήσεις σε όλες τις σελίδες.
- Δίνει τη δυνατότητα φόρτωσης αρχείων, φωτογραφιών και template τα οποία αλλάζουν εντελώς την μορφή του δικτυακού μας τόπου. Η λειτουργία αυτή είναι πολύ σημαντική καθώς μπορούν οι χρήστες να εφαρμόσουν ολοκαίνουργια παρουσία στο δικτυακό τους τόπο από ένα νέο template.
- Η επεξεργασία του κειμένου γίνεται με τη χρήση κειμενογράφων οι οποίοι χαρίζουν στο κείμενό μας μορφή αντάξια ενός κειμένου Word

με διαφορετικά στυλ, γραμματοσειρές, μέγεθος, χρώμα και πολλές από τις λειτουργίες που παρέχουν.

- Υπάρχει μικρότερος φόρτος στον εξυπηρετητή (server) καθώς χρησιμοποιούνται λιγότερες στατικές σελίδες αφού όλη η ανάπτυξη των σελίδων δυναμικά.
- Παρέχει μεγάλη ευελιξία, μεγαλύτερη ομοιομορφία στο ιστότοπό μας και αυξημένη ικανότητα ανάπτυξης του δικτυακού μας τόπου.
- Διατηρούνται πολύ καλύτερα backup και μπορεί εύκολα να γίνει ανάκτηση των δεδομένων μέσω restore.

Τα CMS διαχωρίζονται σε 3 κατηγορίες:

- Κλειστού κώδικα CMS.
- Ανοιχτού κώδικα CMS.
- CMS παραμετροποιήσιμα ανοιχτού κώδικα που αποτελούν μίξη των παραπάνω.

## □□□□□ **Κλειστού κώδικα CMS**

Τα CMS της κατηγορίας αυτής υποστηρίζονται από εταιρείες και όχι από εθελοντές προγραμματιστές. Αυτό σημαίνει ότι έχουν καλύτερη τεκμηρίωση και τεχνική υποστήριξη, καθώς και προσδιορισμένες υπηρεσίες αλλά έχουν ένα βασικό μειονέκτημα που είναι το κόστος. Τα πιο γνωστά CMS κλειστού κώδικα είναι:

- **IBM Workspace Content Management.** Η IBM είναι από τις πιο γνωστές και αξιόπιστες εταιρείες λογισμικού και υλικού. Παρέχει λογισμικά και λύσεις σε πολλών ειδών επιχειρήσεις και διαθέτει μεγάλη τεχνογνωσία και εμπειρία στο χώρο των υπολογιστών. Η IBM δημιούργησε μια πλατφόρμα το IBM Webphere που αφορά τελικούς χρήστες που θέλουν να δημιουργήσουν λογισμικό ή να αναβαθμίσουν υπηρεσίες διαφορετικές μεταξύ τους. Μια επέκταση του websphere είναι το Workspace Content Management. Το προϊόν αυτό παρέχει μια γκάμα λειτουργιών που εξυπηρετούν τελικούς χρήστες που χρειάζονται λογισμικό διαχείρισης περιεχομένου.
- **Jalios JCMS.** Η γαλλική εταιρεία Jalios που εξειδικεύεται σε web-based λύσεις προϊόντων ανέπτυξε το JCMS που περιλαμβάνει αρκετά γνωστά χαρακτηριστικά των CMS όπως διαχείριση περιεχομένου, εγγράφων, διασύνδεση, πύλες και πολλά άλλα. Αποτελεί μια αξιοπρεπή λύση για CMS επί πληρωμή από μια εταιρεία εξειδικευμένη σε αυτό.
- **Vignette Content Management.** Η Vignette είναι μια εταιρεία που έχει ως κύριο στόχο τη μείωση του κόστους των επιχειρήσεων, την αύξηση της παραγωγικότητας και στη βελτίωση της εμπειρίας του χρήστη. Σήμερα πολλές εταιρείες ευελπιστούν να δώσουν καινοτόμες λύσεις σε τέτοιους τομείς. Η λύσεις που παρέχονται από την εταιρεία αυτή στον τομέα της διαχείρισης περιεχομένου web συμπεριλαμβάνουν portal integration, higher level content management που αποδίδει σημαντικά πλεονεκτήματα.

## □□□□□ **Ανοιχτού κώδικα CMS**

Τα CMS της κατηγορίας αυτής επιτρέπουν τη πρόσβαση και την αλλαγή του πηγαίου κώδικα από όλους. Αυτό είναι και το σημαντικότερο πλεονέκτημά τους διότι

μειώνεται δραματικά το κόστος τους. Απαιτούνται βέβαια αρκετές τεχνικές γνώσεις για κάτι τέτοιο και λιγότερες για τη χρησιμοποίηση του λογισμικού. Θα πρέπει λοιπόν όποιος αναλάβει να στήσει ένα τέτοιο σύστημα να διαθέτει έστω κάποιες βασικές γνώσεις web και υπολογιστών προκειμένου να εγκαταστήσει κάτι τέτοιο. Παρέχονται βέβαια αρκετά tutorials, documents και βιβλία στο Internet που μπορούν να βοηθήσουν τον οποιοδήποτε να αξιοποιήσει ένα τέτοιου είδους λογισμικό αλλά θα πρέπει να διαθέτει ένα βασικό επίπεδο γνώσεων υπολογιστών.

Στην κατηγορία αυτή η ανάπτυξη τέτοιου είδους λογισμικού υλοποιείται από πληθώρα εθελοντών προγραμματιστών οι οποίοι εκτός από την ανάπτυξη των βασικών στοιχείων του συστήματος μπορεί να συμμετέχουν στην δημιουργία πρόσθετων (plug-ins) που επεκτείνουν το υπάρχον σύστημα και παρέχουν λειτουργίες που δεν παρέχονται στο βασικό πακέτο. Ας δούμε τα πιο δημοφιλή λογισμικά CMS ανοιχτού κώδικα που υπάρχουν διαθέσιμα.

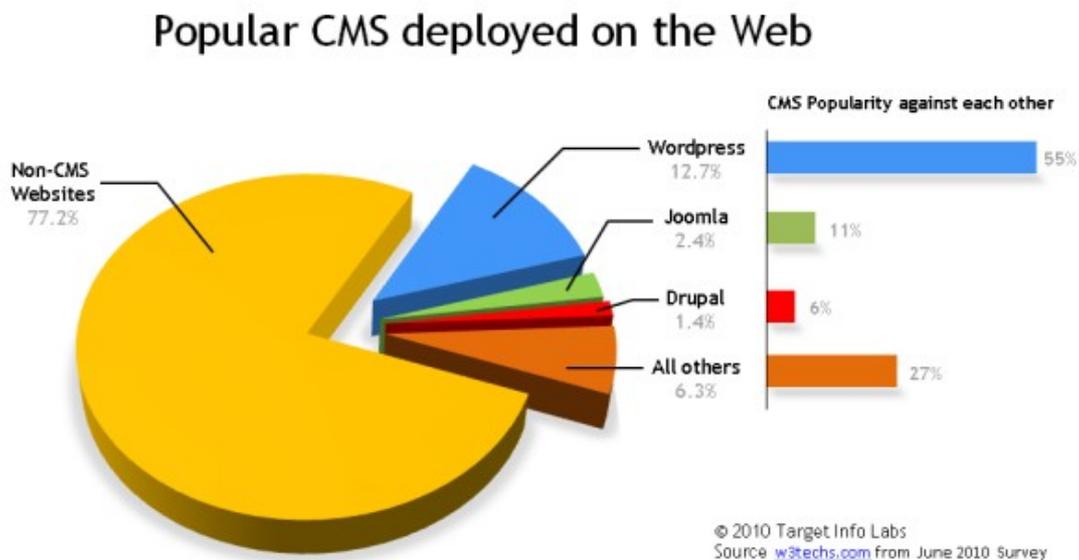
- **Joomla.** Το πιο γνωστό σύστημα διαχείρισης περιεχομένου με τα περισσότερα πλεονεκτήματα, εξαιρετικά ευέλικτο και αποτελεσματικό! Η εφαρμογή αυτή μπορεί να αξιοποιηθεί είτε για προσωπική είτε για εταιρική χρήση με απίστευτες δυνατότητες επέκτασης. Στην παρούσα διπλωματική θα γίνει εκτενής ανάλυση στο σύστημα αυτό, διότι αποτελεί αντικείμενο της μελέτης μας.
- **Drupal.** Το Drupal αποτελεί τον πιο γνωστό ανταγωνιστή του Joomla διότι εκτός από CMS παρέχει στο χρήστη ένα μεγάλο πλήθος από υπηρεσίες και συναλλαγές. Το Drupal όπως και το Joomla είναι γραμμένο σε PHP με τη χρήση της MySQL. Στην παρούσα διπλωματική θα γίνει και μια σύγκριση μεταξύ των δυο αυτών CMS για να αποφανθούμε πιο αξίζει περισσότερο σε κάθε περίπτωση.
- **Xoops.** Το Xoops αποτελεί έναν ισχυρό αντίπαλο όλων των παραπάνω CMS διότι πλέον αποτελεί αναπόσπαστο κομμάτι της ομάδας ανάπτυξης sourceforge ανοιχτού κώδικα. Εάν και ακόμα είναι πολύ νωρίς για να αποφανθεί κανείς την εξέλιξη ενός τέτοιου CMS σίγουρα δεν μπορούμε να είμαστε αρνητικοί στην χρήση του. Το Xoops ακόμα δεν έχει γνωρίσει την επιτυχία που θέλει διότι έχουν ήδη διαδοθεί περισσότερο το Joomla και Drupal. Δεν έχουν βγει ακόμα πολλά extensions, modules, templates και διάφορα άλλα τα οποία θα επεκτείνουν το σύστημα και θα του δώσουν κάποια σημεία υπεροχής από τους ανταγωνιστές του. Προς το παρόν διαθέτει απλά τα ίδια χαρακτηριστικά χωρίς όμως μεγάλη επέκταση γι' αυτό και δεν θα αποτελέσει σύστημα στο οποίο θα στηρίζουμε και θα κάνουμε περαιτέρω ανάλυση.



Εικόνα 15. Τα πιο διαδεδομένα CMS

### 4.3 Δημοτικότητα των CMS

Παρόλο που υπάρχουν αμέτρητα συστήματα διαχείρισης περιεχομένου διαθέσιμα, ύστερα από διεξοδική έρευνα που έγινε τον Ιούνιο του 2010 παρατηρήθηκε ότι ο παγκόσμιος ιστός απαρτίζεται στην ουσία από 3 βασικά CMS: το WordPress, το Joomla και το Drupal. Τα συστήματα αυτά αν και υπάρχουν αρκετό καιρό συγκεντρώνουν μόνο το 16% σχεδόν όλων των διαθέσιμων δικτυακών τόπων.



Εικόνα 16. Δημοτικότητα των CMS

Αυτό αποδεικνύει ότι υπάρχει ακόμα ένα μεγάλο κενό στην αγορά των δικτυακών τόπων όπου κάθε εταιρεία και λογισμικό ανοιχτού κώδικα προσπαθεί να εισχωρήσει και να απορροφήσει όσο το δυνατόν μεγαλύτερο κομμάτι πελατών και υποστηρικτών αντίστοιχα.

Για λόγους διερεύνησης έχει δημιουργηθεί μια μηχανή αναζήτησης που δίνει τη δυνατότητα να ανακαλύψουμε τι CMS διαθέτει ένας δικτυακός τόπος. Αυτό γίνεται μέσα από το δικτυακό τόπο <http://cms.shoutex.com> όπως φαίνεται στην παρακάτω εικόνα.



## Εικόνα 17. Ανακάλυψη είδους CMS δικτυακού τόπου

Μέσα από μια απλή αναζήτηση μπορούμε να διερευνήσουμε τι είδος CMS χρησιμοποιεί για παράδειγμα το <http://www.shoutex.com> όπως φαίνεται στην παρακάτω εικόνα.



### CMS Analysis for [shoutex.com](http://www.shoutex.com)

Identified CMS:  **WORDPRESS**

CMS popularity: 64%

Domain check date: Mar 3, 2011

Rendered in 0.0158 seconds.

[<< Back to check more domains](#)

---

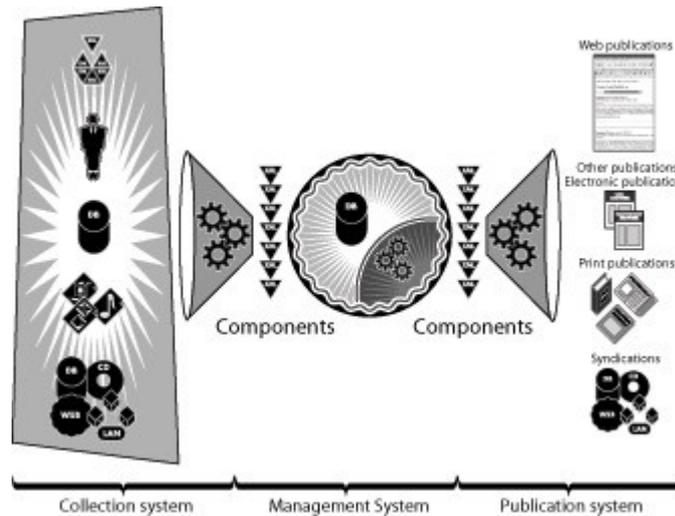
an experiment by [shoutex inc.](http://www.shoutex.com)

[Terms and conditions](#): This is a free service for informational purposes only.

## Εικόνα 18. Παράδειγμα αποτελέσματος αναζήτησης είδους CMS

### 4.4 Αρχιτεκτονική των CMS

Κάθε CMS αποτελείται από πολλά δομικά στοιχεία και η αρχιτεκτονική του κάθε συστήματος είναι άρρηκτα συνδεδεμένη με την εκάστοτε τεχνολογία που χρησιμοποιείται για την ανάπτυξή του. Από όλα τα δομικά στοιχεία θα αναφερθούμε σε αυτά που έχουν κυρίαρχη σημασία στον τρόπο λειτουργίας ενός CMS και αποτελούν αναπόσπαστο κομμάτι κάθε CMS που αναπτύσσεται. Τα 3 κυριότερα στοιχεία ενός CMS που θα μελετήσουμε στην συνέχεια είναι το υποσύστημα συλλογής, διαχείρισης και δημοσίευσης.

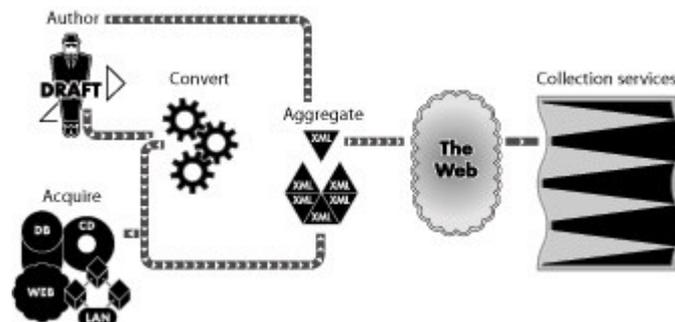


Εικόνα 19. Τα κύρια στοιχεία ενός CMS

□□□□□

## Το υποσύστημα συλλογής

Το υποσύστημα αυτό ευθύνεται για όλες τις διεργασίες που πρέπει να γίνουν προκειμένου η ακατέργαστη πληροφορία να είναι καλά οργανωμένη και δημοσιεύσιμη.



Εικόνα 20. Το υποσύστημα συλλογής

Το υποσύστημα αυτό περιλαμβάνει τα ακόλουθα στοιχεία:

- **Συγγραφή (Authorising).** Δημιουργεί το περιεχόμενο αποκλειστικά από την αρχή. Παρέχονται οι κατάλληλοι ρόλοι και τα κατάλληλα εργαλεία για την συγγραφή άρθρων. Ορίζεται ο ρόλος του συγγραφέα του οποίου ο ρόλος είναι να δημιουργεί το περιεχόμενο. Εάν τώρα υπάρχει και κάποιος συγγραφέας που δεν είναι εγκεκριμένος ή δημιουργεί περιεχόμενο για οποιοδήποτε σκοπό εκτός του περιεχομένου του CMS, τότε το περιεχόμενο αυτό θεωρείται ανέκδοτο υλικό.
- **Απόκτηση (Acquisition).** Συλλέγει το περιεχόμενο από κάποια υπάρχουσα πηγή. Η διαδικασία μπορεί να αυτοματοποιηθεί μερικώς ή εξολοκλήρου.

- **Μετατροπή (Conversion).** Η πληροφορία σε αυτό το στάδιο δεν βρίσκεται ακριβώς στην μορφή ή δομή που απαιτεί το σύστημα. Θα πρέπει να τροποποιηθεί ώστε να ταιριάζει στα πρότυπα του περιεχομένου του συστήματος. Θα πρέπει δηλαδή να μετατραπεί σε ροή δεδομένων, παρά σε αναλυμένη πληροφορία, έτσι ώστε να μην εισάγει αντικείμενα και χρώματα αλλά ένα ολοκληρωμένο σχέδιο. Η μετατροπή αποτελείται από τα ακόλουθα λογικά βήματα:
  - **Διαλογή.** Αφαίρεση και απομόνωση της άχρηστης περιβάλλουσας πληροφορίας όπως οι επικεφαλίδες, ανεπιθύμητο περιεχόμενο.
  - **Διαμόρφωση περιεχομένου.** Μετατροπή της δυαδικής πληροφορίας σε ένα πρότυπο που υποστηρίζει το CMS, και διαχωρισμός της προσφερόμενης διαμόρφωσης από τη δομή του.
  - **Δόμηση του περιεχόμενου.** Δημιουργία αποκλειστικής δομής της πληροφορίας ή κατάλληλη μετατροπή της.
- **Συσώρευση (Aggregation).** Τροποποιεί το περιεχόμενο, το διαιρεί σε συστατικά στοιχεία και το προσαυξάνει με τα απαραίτητα μεταδεδομένα.
- **Υπηρεσίες συλλογής (Collection services).** Είναι ένα σύνολο από υπηρεσίες οι οποίες έχουν ως κύριο στόχο να παρέχουν βοήθεια στον να αποθηκεύεται το περιεχόμενο στις αποθήκες δεδομένων. Οι υπηρεσίες αυτές διαχωρίζονται στις ακόλουθες ενέργειες:
  - Εναπόθεση των συστατικών απευθείας στην αποθήκη του CMS.
  - Φόρτωση προηγούμενων δημιουργούμενων συστατικών στην αποθήκη του CMS κατά σειρά ή χωρίς.



## Το υποσύστημα διαχείρισης

Το υποσύστημα διαχείρισης είναι υπεύθυνο για τη μακροχρόνια αποθήκευση των συστατικών περιεχομένου καθώς και για κάθε είδους αρχείο που χρησιμοποιείται. Περιέχει την αποθήκη περιεχομένου, το workflow καθώς και δυνατότητες διαχείρισης. Στο υψηλότερο επίπεδο, παρέχει τη δυνατότητα να γνωρίζουμε τι έχει συλλεχθεί και ποια είναι η διάθεση του. Για παράδειγμα το υποσύστημα διαχείρισης θα πρέπει να είναι ικανό να μας ενημερώνει για τα ακόλουθα:

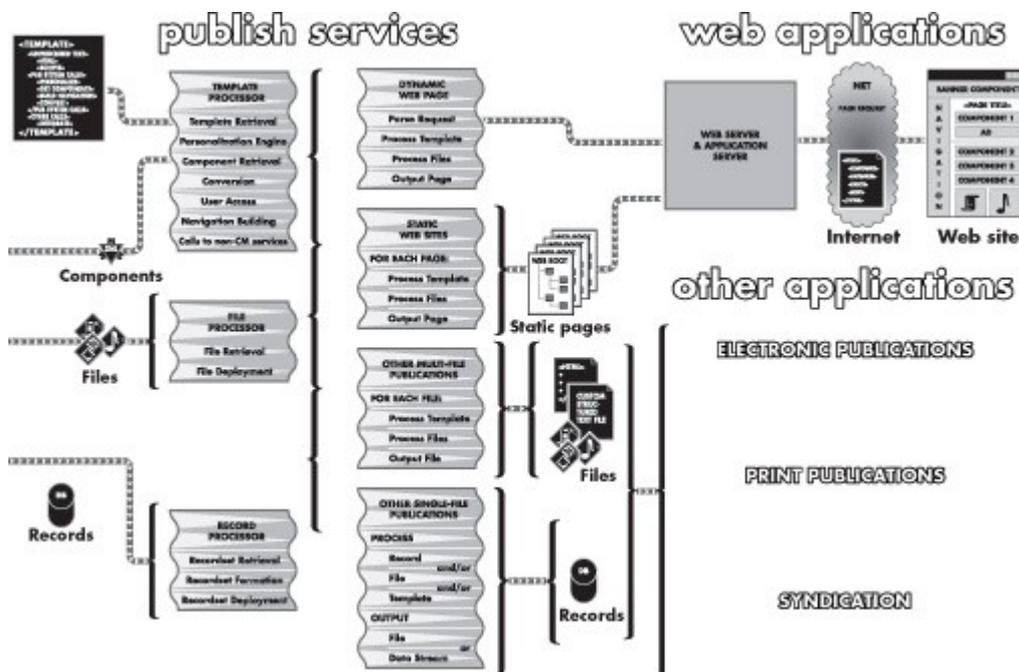
- Λεπτομέρειες του περιεχομένου, συμπεριλαμβάνοντας το είδος των συστατικών που υπάρχουν διαθέσιμα και σε ποιο στάδιο του κύκλου ζωής τους βρίσκονται.
- Πόσο καλά αρχικοποιημένο είναι το υλικό μας και εάν μπορεί να δημιουργηθεί κάποιου είδους συμφόρηση.
- Πως χρησιμοποιούνται τα συστατικά σε δημοσιεύσεις και ποιο περιεχόμενο έχει μείνει αχρησιμοποίητο ή έτοιμο προς αφαίρεση.
- Ποιος έχει πρόσβαση και σε ποιο περιεχόμενο και ποιος είναι αυτός που συνεισφέρει περισσότερο.

Συμπερασματικά θα λέγαμε ότι το υποσύστημα διαχείρισης θα πρέπει να παρέχει απαντήσεις σε οτιδήποτε ερωτήματα έχουμε σχετικά με το περιεχόμενο, τις δημοσιεύσεις ή από το υποσύστημα συλλογής. Για να επιτυγχάνεται αυτό το υποσύστημα διαχείρισης περιλαμβάνει τα ακόλουθα:

- Αποθήκες δεδομένων
- **Διαχείριση.** Ένα σύστημα για αρχικοποίηση και καθορισμό του CMS.
- **Ροής εργασίας.** Καθορίζει το σύνολο των βημάτων που απαιτούνται ώστε να γίνει σωστή δουλειά στο περιεχόμενο για να μπορεί να δημοσιευτεί.
- **Συνδέσεις.** Ένα σύνολο από συνδέσεις υλικού και λογισμικού με άλλα συστήματα μέσα στον οργανισμό μεταξύ δικτύων, εξυπηρετητών και αποθηκών δεδομένων.

## □□□□□ Το υποσύστημα δημοσίευσης

Το υποσύστημα δημοσίευσης είναι υπεύθυνο για την εξαγωγή περιεχομένου από τον αποθηκευτικό χώρο των δεδομένων και τη δημιουργία δημοσιεύσεων.



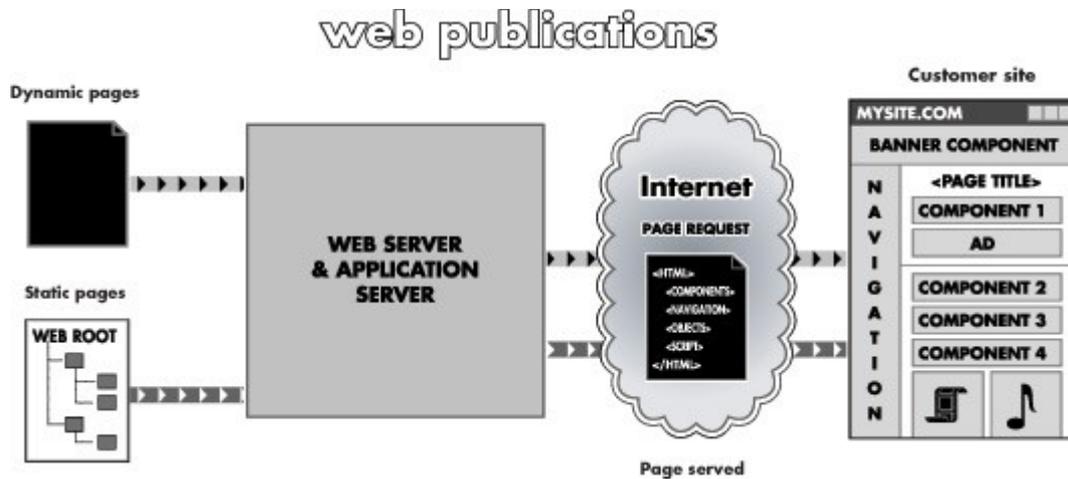
Εικόνα 21. Το υποσύστημα δημοσίευσης χρησιμοποιεί πρότυπα και υπηρεσίες δημοσίευσης για να παράγει ιστοσελίδες καθώς και άλλα είδη δημοσιεύσεων.

Το υποσύστημα δημοσίευσης περιλαμβάνει:

- **Τα πρότυπα δημοσίευσης.** Προγράμματα που φτιάχνουν τις δημοσιεύσεις αυτόματα.
- **Υπηρεσίες δημοσίευσης.** Ένα σύνολο από εργαλεία και μεθόδους για έλεγχο των δημοσιεύσεων και πως αυτές δημοσιεύονται.
- **Συνδέσεις.** Εργαλεία και μέθοδοι που χρησιμοποιούνται για να συμπεριλάβουν δεδομένα από άλλα (μη-CMS) συστήματα σε τελειωμένες δημοσιεύσεις.
- **Δημοσιεύσεις διαδικτύου.** Η πιο κοινή μορφή των περισσότερων συστημάτων διαχείρισης περιεχομένου.

- Άλλες δημοσιεύσεις. Άλλες δημοσιεύσεις μη διαδικτυακές, συμπεριλαμβάνοντας ηλεκτρονικές και εκτυπώσιμες μορφές.

Ο τρόπος με τον οποίο οι δημοσιεύσεις διαδικτύου υλοποιούνται σε ένα CMS εμφανίζονται στην παρακάτω εικόνα.



**Εικόνα 22.** Το σύστημα δημοσίευσης διαδικτύου μπορεί να παράγει μια πλήρως δυναμική ιστοσελίδα.

Οι δημοσιεύσεις διαδικτύου είναι συνήθως δυναμικές και το CMS παράγει αυτές τις ιστοσελίδες μια σελίδα τη φορά σε ανταπόκριση των κλικ των χρηστών. Στην περίπτωση αυτή, τα κλικ των χρηστών περνούν μια αίτηση σελίδας στον web server που πυροδοτεί τις υπηρεσίες δημοσίευσης του διαδικτύου να κάνουν τα ακόλουθα:

- Φόρτωση του πρότυπου της σελίδας.
- Πέρασμα οποιονδήποτε παραμέτρων που έρχονται μαζί με το αίτημα του χρήστη.
- Εκτέλεση του κώδικα του προτύπου για τη δημιουργία της ολοκληρωμένης σελίδας.
- Πέρασμα της ολοκληρωμένης σελίδας πίσω στο web server για τη προβολή στο φυλλομετρητή του χρήστη.

Αυτή είναι αρχιτεκτονική που χρησιμοποιούν τα περισσότερα συστήματα CMS και ο τρόπος λειτουργίας τους είναι βρίσκεται σε πλήρη αρμονία με αυτές τις αρχές. Στη συνέχεια θα εξετάσουμε το Joomla και να δώσουμε την ανάλυση της αρχιτεκτονικής αυτού του πασίγνωστου CMS, ενώ θα δούμε τι μετατροπές χρειάζεται να γίνουν ώστε να εφαρμόσουμε το peRSSonal στο Joomla.

## **5 Το Joomla**

Σε αυτό το κεφάλαιο θα μελετήσουμε το Joomla. Θα εξετάσουμε τη δομή του, την αρχιτεκτονική του και τα κύρια χαρακτηριστικά του που αποτέλεσαν εφαλτήριο ώστε να γίνει κυρίαρχο σύστημα διαχείρισης περιεχομένου του διαδικτύου. Τέλος για να ολοκληρώσουμε την έρευνά μας θα συγκρίνουμε το Joomla και με τα άλλα CMS ανοιχτού κώδικα ώστε να αποφανθούμε τους λόγους για τους οποίους στηριχθήκαμε στη χρήση του για το σύστημα perSSonal.

### **5.1 Η ιστορία του Joomla**

Το Joomla αποτελεί συνέχεια του συστήματος MAMBO. Το MAMBO ήταν ένα σύστημα διαχείρισης περιεχομένου που αποτελούσε ονομασία της εταιρείας Micro International Pvt Ltd. Η εταιρεία ήταν μη κερδοσκοπικού χαρακτήρα με μοναδικό σκοπό τη χρηματοδότηση του MAMBO.

Στις 17 Αύγουστου του 2005 η ομάδα του MAMBO διασπάστηκε με αποτέλεσμα τη δημιουργία μιας νέας ομάδας ανάπτυξης που αποτελείτο από προγραμματιστές, σχεδιαστές ιστοσελίδων έτοιμοι να συνεισφέρουν σε λογισμικό ανοιχτού κώδικα μέσα από στην ιστοσελίδα OpenSourceMatters. Μέσα από διάφορα προβλήματα που ακολούθησαν η ομάδα αποφάσισε να ακολουθήσει μια πλήρως ελεύθερη διανομή ενός συστήματος διαχείρισης που οι συντελεστές θα συνεισφέρουν όπως ορίζεται από τους ορισμούς «ανοιχτού κώδικα», δίχως να παρεμβάλλεται καμία εταιρεία και δίχως να ξεφεύγει από αυτή τη λογική. Η ομάδα έπειτα από συσκέψεις για το νέο project που δημιουργήθηκε αποφασίζει.

Την 1<sup>η</sup> Σεπτεμβρίου 2005 το νέο όνομα, Joomla, είναι πλέον γεγονός. Το όνομά του προέρχεται από την αραβική λέξη Jumla, όπου έπειτα από αγγλική ορθογραφία έγινε Joomla. Η αραβική ερμηνεία της λέξης σημαίνει «όλοι μαζί» ή «ως σύνολο» ερμηνεία η οποία αποτελεί και ακρογωνιαίο λίθο του ανοιχτού λογισμικού.

Η πρώτη έκδοση του Joomla (1.0.0) κυκλοφόρησε στις 16 Σεπτεμβρίου 2005. Ουσιαστικά αποτελούσε εμπορική απελευθέρωση του Mambo 4.5.2.3. Έπειτα από ανακάλυψη συνδυασμού σφαλμάτων και μέτρων διορθώσεων σε επίπεδο ασφαλείας κέρδισε το βραβείο Packt Publishing Open Source Management System το 2006 και το 2007.

Έτσι το Joomla κατέκτησε μια μεγάλη κοινότητα χρηστών και αναπτύσσεται δυναμικά ολοένα και περισσότερο, χαρίζοντας μεγάλη ευελιξία στον τρόπο διαχείρισης περιεχομένου στο διαδίκτυο. Στην επόμενη παράγραφο θα αναλύσουμε τα χαρακτηριστικά στα οποία βασίστηκε η ανάπτυξη του Joomla.

## 5.2 Χαρακτηριστικά του Joomla

Το Joomla στηρίζεται ως επί το πλείστον σε δυναμικό περιεχόμενο γι'αυτό και κάνει χρήση server side τεχνολογιών και βάσεων δεδομένων. Αφού το ίδιο αποτελεί λογισμικό ανοιχτού κώδικα και είναι ελεύθερο δεν θα μπορούσε να βασίζεται σε λογισμικά επί πληρωμή. Όλα τα συστατικά του στοιχεία βασίζονται σε ανοιχτό λογισμικό ελεύθερης διανομής που εξελίσσονται και τα ίδια από άλλες ομάδες που συμβάλλουν σε τεχνολογίες ανοιχτού κώδικα.

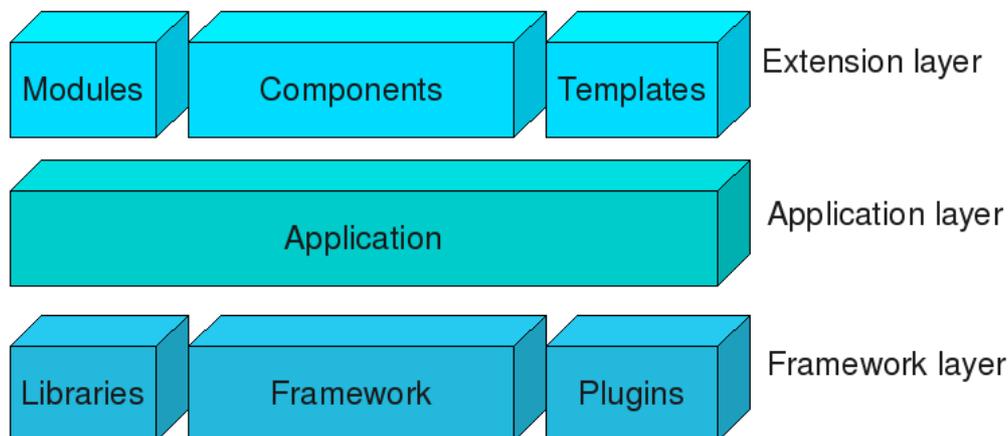
Συνοπτικά λοιπόν μπορούμε να αναφέρουμε ότι το Joomla απαρτίζεται από τα εξής χαρακτηριστικά:

- Χρησιμοποιεί τις καλύτερες τεχνολογίες ανοιχτού λογισμικού όπως είναι:
  - MySQL ως βάση δεδομένων
  - PHP ως server side script για το προγραμματισμό του συστήματος.
  - XML για τη διαχείριση περεταιίρω εργαλείων.
  - CSS2 που αποτελεί την τελευταία εξέλιξη των CSS που συζητήσαμε στην παράγραφο 2.5.
  - RSS δυνατότητα.
- Πλήρης διαχείριση της βάσης και του περιεχομένου του συστήματος από οποιοδήποτε web browser.
- Τμήματα από Νέα Προϊόντα ή Υπηρεσίες πλήρως επεξεργάσιμα, διαχωρίσιμα και εύχρηστα.
- Πλήρως παραμετροποιήσιμο περιεχόμενο και περιβάλλον, συμπεριλαμβανομένων και των θέσεων του αριστερού, κεντρικού και δεξιού μενού.
- Παρέχει τη δυνατότητα δημιουργίας πολλών επιπέδων χρηστών.
- Δυνατότητα υποστήριξης FORUM/VOTING.
- Είναι πανεύκολο στη χρήση ακόμα και από αρχάριους χρήστες.
- Υπάρχει ειδικός μηχανισμός για το αντίστοιχο ranking από μηχανές αναζήτησης χρησιμοποιώντας μετα-δεδομένα.
- Διατίθενται με πλήρη υποστήριξη πολλών γλωσσών συμπεριλαμβανομένου και της ελληνικής.
- Δεν είναι εξαρτώμενο από το λειτουργικό σύστημα, αρκεί αυτό να έχει εγκατεστημένα τα επιμέρους συστατικά του. Μπορεί λοιπόν να λειτουργήσει πλήρως σε Windows, Linux, Unix συστήματα, Mac, Solaris AIX.
- Η σύνταξη του περιεχομένου μπορεί να γίνει ταυτόχρονα και να είναι εμφανή μόνο στα αντίστοιχα επίπεδα των χρηστών που είναι συνδεδεμένοι.
- Δυνατότητα διαφορετικού τρόπου εμφάνισης του δικτυακού τύπου μέσα από πλήθος προτύπων που έχει ενσωματωμένα και που εύκολα μπορούν να εισαχθούν σε αυτό.
- Διαχείριση του τρόπου εμφάνισης του κειμένου από διαχειριστές εγγράφων που διαθέτει καθώς και δυνατότητα ορθογραφίας όπου αυτό είναι εφικτό.

## 5.3 Αρχιτεκτονική του Joomla

Το Joomla αποτελείται από πολλά και διαφορετικά συστατικά μέρη που επιτελούν συγκεκριμένες λειτουργίες το καθένα. Κάθε συστατικό του Joomla εξαιτίας της διαφορετικής του λειτουργίας αποτελεί από μόνο του ένα αυτόνομο τμήμα.

Εάν κάνουμε μια προεπισκόπηση του συστήματος θα δούμε ότι απαρτίζεται από 3 βασικά επίπεδα που αποτελούν τους κύριους μοχλούς και φαίνονται στην παρακάτω εικόνα. Είναι το κεντρικό, κατασκευαστικό επίπεδο, το επίπεδο των εφαρμογών και το επίπεδο των επεκτάσεων.



**Εικόνα 23. Η Αρχιτεκτονική του Joomla**

Το πάνω επίπεδο αφορά το επίπεδο των επεκτάσεων. Στο επίπεδο αυτό βρίσκονται τα συστατικά στοιχεία του Joomla, τα πρότυπα και οι ενότητες.

Το μεσαίο επίπεδο είναι το επίπεδο των εφαρμογών που αποτελείται από 3 βασικές εφαρμογές:

- Εγκατάσταση. Η εφαρμογή αυτή είναι υπεύθυνη για την εγκατάσταση του Joomla σε ένα web server και τη διαγραφή της όταν ολοκληρωθεί.
- Διαχείριση. Η εφαρμογή αυτή είναι υπεύθυνη για τον διαχειριστή που δουλεύει από πίσω και διαχειρίζεται το δικτυακό τόπο και το περιεχόμενό του.
- Ο δικτυακός τόπος. Η εφαρμογή αυτή είναι υπεύθυνη για την παρουσίαση του δικτυακού τόπου.
- Απομακρυσμένη πρόσβαση. Η εφαρμογή αυτή υποστηρίζει τη διαχείριση του δικτυακού τόπου του Joomla απομακρυσμένα.

. Το κατώτερο επίπεδο περιέχει τα εξής:

- Τις βιβλιοθήκες που απαιτούνται από το πλαίσιο του Joomla ή που εγκαθίστανται από εξωτερικούς προγραμματιστές.
- Το πλαίσιο λειτουργίας του Joomla
- Τα πρόσθετα που επεκτείνουν τη λειτουργία που είναι διαθέσιμη στο πλαίσιο λειτουργίας του Joomla.

Στη συνέχεια θα αναλύσουμε τη δομή του Joomla ώστε να γίνουν πιο κατανοητά τα περαιτέρω στοιχεία που χρησιμοποιεί το Joomla.

## **5.4 Δομή του Joomla**

Η δομή του Joomla αφορά κυρίως τον τρόπο με τον οποίο δομούνται τα συστατικά στοιχεία του πριν αυτά εμφανιστούν στο χρήστη. Ας δούμε ένα δομικό στοιχείο αναλυτικά.

## □□□□□ Το δημόσιο τμήμα και η περιοχή διαχείρισης (Front-end και Back-end).

Το δημόσιο τμήμα είναι στην ουσία το τμήμα που βλέπουν οι χρήστες του διαδικτύου όταν επισκέπτονται το δικτυακό μας τόπο είτε είναι απλοί επισκέπτες είτε συνδεδεμένοι χρήστες. Στο τμήμα αυτό εμφανίζονται ότι έχουμε δώσει εξουσιοδότηση να εμφανίζεται στο δικτυακό μας τόπο.

Η περιοχή της διαχείρισης είναι η περιοχή που γίνεται όλη η προετοιμασία του δικτυακού μας τόπου. Περιέχει το επίπεδο διαχείρισης του δικτυακού τόπου το οποίο χρησιμοποιούν οι διαχειριστές του δικτυακού μας τόπου για τη διαμόρφωση, συντήρηση, καθαρισμό, παραγωγή στατιστικών, καθώς και τη δημιουργία νέου περιεχομένου.

Στο κεφάλαιο των ρυθμίσεων του Joomla θα δούμε και τις δύο αυτές περιοχές και πως χρησιμοποιούνται οι κάθε μια.

### □□□□□ Το περιεχόμενο

Περιεχόμενο για το Joomla μπορεί να αποτελέσει οποιαδήποτε μορφή πληροφορίας με όποια μορφή και αν αυτή είναι αποθηκευμένη. Μπορεί να είναι από ένα απλό κείμενο, μέχρι φωτογραφίες, σύνδεσμοι προς άλλες σελίδες, ηχητικά ντοκουμέντα, μουσική, βίντεο και οτιδήποτε μπορεί να αποτελέσει ψηφιακή πληροφορία.

Για να οργανωθεί όσο το δυνατόν καλύτερα όλη αυτή η πληροφορία που μπορεί να προστεθεί το Joomla παρέχει τη δυνατότητα της διάταξης της πληροφορίας σε Ενότητες και κατηγορίες μέσα στις οποίες προστίθενται τα άρθρα των χρηστών. Η ιεραρχία είναι απλή και έτσι κάθε ενότητα, που αποτελεί τον κύριο κορμό της πληροφορίας μας, μπορεί να αποτελείται από μια ή περισσότερες κατηγορίες και κάθε κατηγορία μπορεί να αποτελείται από ένα ή περισσότερα άρθρα.

### □□□□□ Οι επεκτάσεις

Ως επέκταση (Extension) στο Joomla ορίζεται ως οτιδήποτε δεν αποτελεί κατασκευαστικό κομμάτι του και προστίθενται από το χρήστη. Έτσι, όλα τα συστατικά μέρη του Joomla που παράγονται από το χρήστη όπως είναι οι ενότητες, τα πρότυπα (templates) που δίνουν τη μορφή του δικτυακού τόπου καθώς και κάποια πρόσθετα που επεκτείνουν τον τρόπο λειτουργίας του Joomla αποτελούν τις επεκτάσεις του συστήματος. Οι επεκτάσεις είναι αυτές που δίνουν ολοκαίνουργιες λειτουργίες στο Joomla και αλλάζουν ολοκληρωτικά το δικτυακό τόπο.

### □□□□□ Τα συστατικά

Τα συστατικά (components) του Joomla έχουν σστη δυνατότητα να εκτελούν το δικό τους κομμάτι κώδικα και αποτελούν από μόνες τους μια αυτόνομη εφαρμογή. Αυτές αποτελούν τα components του Joomla και είναι τυπικές εφαρμογές που κάνουν ολοένα και περισσότερο την εμφάνισή τους τον τελευταίο καιρό στο Joomla. Τέτοιες είναι τα online καταστήματα, τα photo album, τα forum, καθώς και απομακρυσμένη μάθηση μέσω e-learning.

### □□□□□ Τα πρότυπα

Κάποιες άλλες μορφές επέκτασης του Joomla είναι τα πρότυπα (templates). Τα πρότυπα αφορούν την οπτική απεικόνιση του δικτυακού μας τόπου. Καθορίζουν τα

χρώματα, τις γραμματοσειρές, εικόνες φόντου, τις αποστάσεις και το διαχωρισμό της σελίδας. Ένα πρότυπο στο Joomla αποτελείται από τουλάχιστον ένα αρχείο HTML για τη δομή της σελίδας και ένα αρχείο CSS για το σχεδιασμό και της απεικόνιση των συστατικών της στοιχείων.

## □□□□ Τα πρόσθετα (Plug-ins)

Τα πρόσθετα αποτελούν και αυτά ξεχωριστά κομμάτια κώδικα που επιτελούν κάποιες λειτουργίες που βοηθούν το σύστημα να κάνει επιπλέον πράγματα που επιθυμεί ο χρήστης. Για παράδειγμα μπορεί plugin να χρησιμοποιηθεί στο εσωτερικό περιεχόμενο του κειμένου για να φορτώσει το περιεχόμενο ενός ενθέματος στο κείμενο ή ακόμα για να φορτώσει κάποια πρόσθετο από μια εφαρμογή.

## 5.5 Εργαλεία που απαιτούνται για το Joomla

Ας ξεκινήσουμε τώρα το ενδιαφέρον κομμάτι του Joomla και ας δούμε πως μπορούμε να χρησιμοποιήσουμε αυτό το πολύ δυνατό εργαλείο ανοιχτού κώδικα για δικτυακούς τόπους. Πριν δούμε όμως πως λειτουργεί το Joomla ας δούμε τι συστατικά στοιχεία απαιτούνται για να εγκαταστήσουμε το Joomla.

Όπως έχουμε αναφέρει παραπάνω το Joomla θα πρέπει να εγκατασταθεί σε οποιοδήποτε web server που διαθέτει server side τεχνολογία PHP, καθώς και μια βάση MYSQL. Εκτός από αυτό θα πρέπει να διαθέτει και δυνατότητα FTP συνδέσεων καθώς και αποστολής και λήψης e-mail μέσω SMTP πρωτοκόλλου. Αυτές είναι και οι ελάχιστες απαιτήσεις που έχουν πλέον οι περισσότεροι web server.

Για να γίνει η ζωή μας πιο εύκολη και να μην χρειάζεται να εγκαταστήσουμε ένα ένα στοιχείο από αυτά και ύστερα να πρέπει να εκτελέσουμε ένα σύνολο από ρυθμίσεις για να δημιουργήσουμε το web server μας, υπάρχουν ήδη ολοκληρωμένα πακέτα που περιλαμβάνουν όλα αυτά και δημιουργούν τις ρυθμίσεις αυτόματα. Τέτοια λογισμικά είναι το XAMPP και το WAMP.

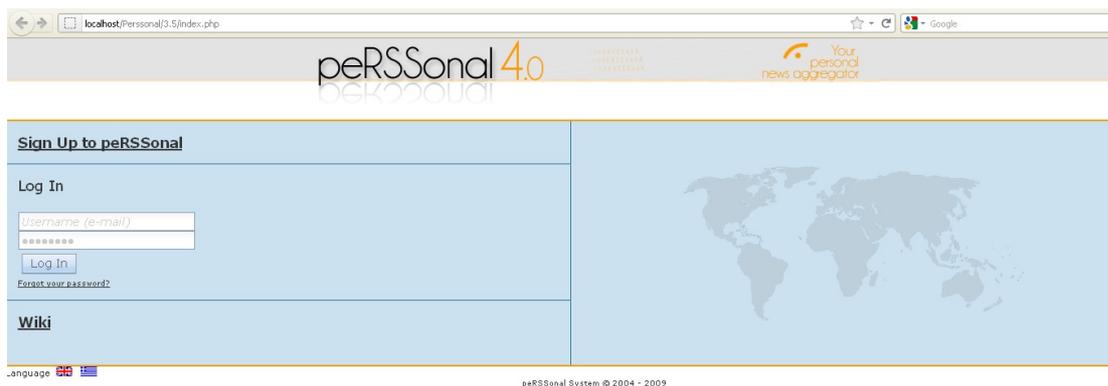
Εάν έχουμε windows μηχανήμα θα προτιμήσουμε το WAMP του οποίου η εγκατάσταση φαίνεται στις παρακάτω εικόνες. Το WAMP περιλαμβάνει τα εξής λογισμικά ανοιχτού κώδικα:

- Apache Web Server
- MySQL Database
- PHP Server side scripting language

## 5.6 Η μορφή του peRSSonal πριν ενσωματωθεί στο Joomla

Στην ενότητα αυτή θα περιγράψουμε τωρινή μορφή του peRSSonal και τις αλλαγές που επιδέχθηκε κατά τη μεταφορά του στο Joomla. Θα μελετήσουμε τον τρόπο λειτουργίας του, την εγγραφή στο σύστημα και την προβολή των απαραίτητων στοιχείων στο χρήστη.

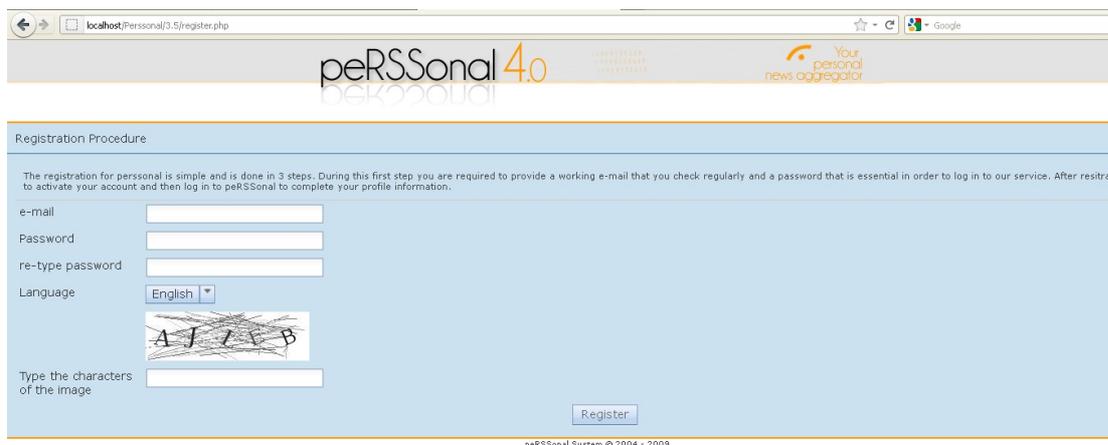
Η αρχική εικόνα που συναντά κανείς όταν επισκέπτεται το peRSSonal είναι η παρακάτω:



**Εικόνα 24. Η αρχική εικόνα του peRSSonal**

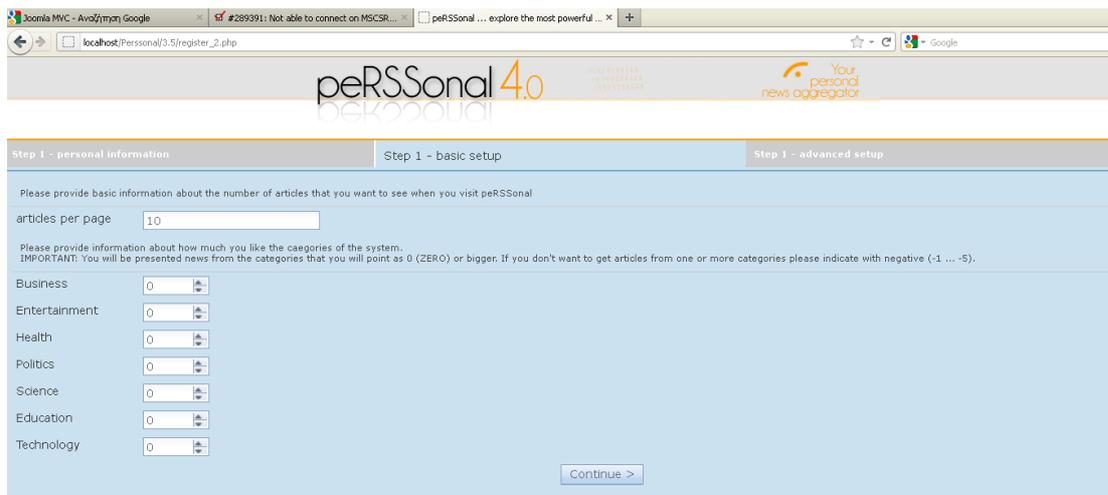
Το σύστημα στην αρχική σελίδα επιτρέπει την εγγραφή σε κάποιο χρήστη ή την είσοδό του εάν έχει ήδη πραγματοποιήσει μια εγγραφή. Κατά την περιήγηση του στο δικτυακό τόπο ο χρήστης έχει τη δυνατότητα να επιλέξει τη γλώσσα περιήγησης μεταξύ της Αγγλικής και της Ελληνικής γλώσσας.

Ας δούμε την επιλογή της εγγραφής στο σύστημα που υλοποιείται μέσα από 3 απλά βήματα. Στο 1<sup>ο</sup> βήμα ο χρήστης πρέπει να δώσει τα προσωπικά του στοιχεία όπως φαίνεται από την παρακάτω εικόνα.



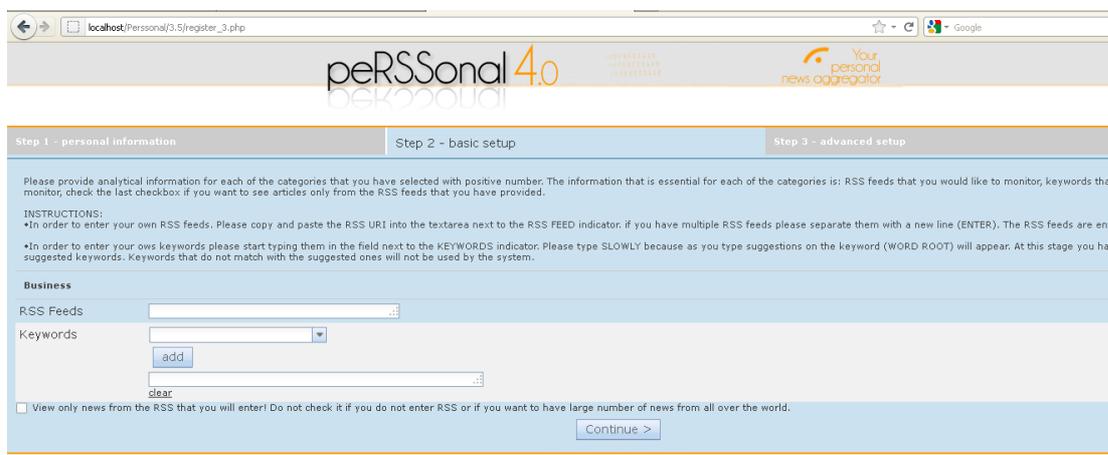
**Εικόνα 25. Εγγραφή στο peRSSonal - Βήμα 1<sup>ο</sup> Εισαγωγή προσωπικών στοιχείων**

Αφότου ο χρήστης δώσει το e-mail του, τον κωδικό πρόσβασης που επιθυμεί, και τη γλώσσα που επιθυμεί να χρησιμοποιεί το peRSSonal επιλέγει την εγγραφή στο σύστημα και ακολουθεί το 2<sup>ο</sup> βήμα όπως φαίνεται στην παρακάτω εικόνα.



**Εικόνα 26. Εγγραφή στο peRSSonal - Βήμα 2<sup>ο</sup> Εισαγωγή επιθυμητών κατηγοριών για προβολή RSS.**

Στο βήμα αυτό ο χρήστης ήδη έχει γραφτεί στη βάση του peRSSonal σύμφωνα με τα στοιχεία που έδωσε στο προηγούμενο βήμα και στο βήμα αυτό απλώς συμπληρώνει τις κατηγορίες που ενδιαφέρεται να βλέπει ειδήσεις από RSS. Στο επόμενο βήμα ο χρήστης παρέχει πληροφορίες σχετικά με τις κατηγορίες που τον ενδιαφέρουν για RSS όπως φαίνεται στην παρακάτω εικόνα.



**Εικόνα 27. Εγγραφή στο peRSSonal - Βήμα 3<sup>ο</sup> Εισαγωγή RSS ειδήσεων και λέξεων κλειδιά.**

Στο βήμα αυτό ο χρήστης έχει τη δυνατότητα να εισάγει δικά του RSS και λέξεις κλειδιά για την κατηγορία που τον ενδιαφέρει και να ολοκληρώσει τα βήματα της εγγραφής στο peRSSonal. Τέλος εμφανίζεται ένα μήνυμα για την ολοκλήρωση της εγγραφής όπως φαίνεται από την παρακάτω εικόνα.



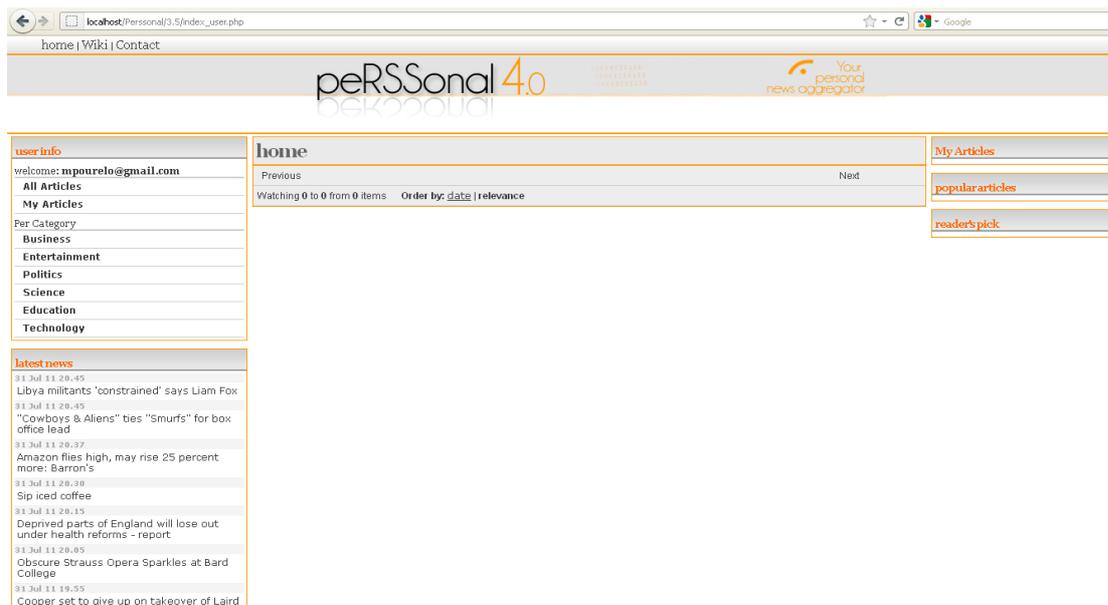
**Εικόνα 28. Εγγραφή στο peRSSonal - Ολοκλήρωση της εγγραφής.**

Αφού ολοκληρωθεί η εγγραφή το σύστημα στέλνει ένα e-mail επιβεβαίωσης στο e-mail του χρήστη που ενεργοποιεί το λογαριασμό του. Αφού ενεργοποιηθεί ο λογαριασμός ύστερα από την αρχική σελίδα μπορεί να κάνει εισαγωγή στο σύστημα όπως φαίνεται από την παρακάτω εικόνα.



**Εικόνα 29. Είσοδος στο peRSSonal.**

Μετά την εισαγωγή του χρήστη στο σύστημα εμφανίζεται η αρχική εικόνα του peRSSonal όπως φαίνεται στην παρακάτω εικόνα.



**Εικόνα 30. Η αρχική εικόνα του peRSSonal.**

Μέσα από το σύστημα ο χρήστης μπορεί να επιλέξει να δει τα άρθρα ανά κατηγορία, όλα τα άρθρα ή τα προσωπικά του άρθρα μέσα από ένα πρόσθετο που ονομάζεται userInfo. Ακριβώς κάτω από αυτό εμφανίζεται ένα άλλο πρόσθετο που εμφανίζει τα άρθρα που μπορεί να διαβάσει ο χρήστης. Στα δεξιά εμφανίζονται κάποια επιπλέον πρόσθετα όπου ο χρήστης μπορεί να δει κάποια δημοφιλή άρθρα (popular news), τα άρθρα του (my articles) και τις επιλογές του αναγνώστη (reader's pick).

Κάθε φορά που ο χρήστης κάνει μια επιλογή κάποια από τα modules επηρεάζονται και εμφανίζουν διαφορετικά στοιχεία και άλλα παραμένουν τα ίδια. Στο κέντρο το συστήματος εμφανίζεται πάντα η αίτηση του χρήστη από κάποιο module. Για παράδειγμα εάν επιλέξουμε να δούμε κάποιο άρθρο από τα τελευταία νέα η σελίδα που παραπέμπει εμφανίζεται στην παρακάτω εικόνα.

The screenshot shows the peRSSonal 4.0 interface. On the left is a 'user info' sidebar with categories like Business, Entertainment, Politics, Science, Education, and Technology. The main content area displays an article titled '"Cowboys & Aliens" ties "Smurfs" for box office lead' with a summary and a photo of the movie cast on a red carpet. On the right is a 'Similar Articles' sidebar with a 'reader's pick' section.

Εικόνα 31. Προβολή ενός άρθρου

Παρατηρούμε ότι η μορφή του template δεν αλλάζει και το μόνο που προστέθηκε είναι ένα νέο module που αφορά το tagging των λέξεων. Ο χρήστης μέσα από αυτό το module μπορεί να επιλέξει ποιες από τις λέξεις του άρθρου είναι σημαντικές. Συνεχίζοντας την περιήγηση στο peRSSonal θα δούμε και την επιλογή του χρήστη να δει τα άρθρα του όπως φαίνεται και στην παρακάτω εικόνα.

The screenshot shows the 'my articles' section of the peRSSonal 4.0 interface. The main area contains a table with columns for 'Previous', 'Next', 'Entertainment', 'Politics', and 'Watching 0 to 10 from 2 items'. The sidebar on the left includes 'latest news' with a list of recent articles and a search bar.

Εικόνα 32. Προβολή των προσωπικών άρθρου του χρήστη

Ο χρήστης εδώ έχει τη δυνατότητα άμεσης προβολής και διαχείρισης όλων των προσωπικών του άρθρων και προβολή ανά κατηγορία. Έως τώρα έχουμε καλύψει τα modules και τις επιλογές του peRSSonal και ο τρόπος με τον οποίο θα μεταφερθεί στο Joomla αρχίζει και γίνεται ολοένα και πιο ξεκάθαρος. Στη συνέχεια θα περιγράψουμε τις τροποποιήσεις που πρέπει να γίνουν στο Joomla ώστε να υποστηρίξει το peRSSonal και στο κεφάλαιο 6 θα δώσουμε μια αναλυτική περιγραφή καθενός module και component που φτιάχτηκε για αυτό.

## **5.7 Βελτιώσεις στο σύστημα peRSSonal κατά τη μεταφορά του στο Joomla**

Εδώ θα αναφέρουμε τις αλλαγές που θα πρέπει να γίνουν στο Joomla για να υποστηρίξει το σύστημα του peRSSonal, καθώς και τις βελτιώσεις που έγιναν κατά τη μεταφορά. Η ανάλυση θα γίνει τμηματικά όπως παρουσιάστηκε ακριβώς παραπάνω το peRSSonal.

### **□□□□ Βελτιώσεις στη διαδικασία εγγραφής του peRSSonal στο Joomla.**

Η διαδικασία εγγραφής στο peRSSonal αρχικά θα πρέπει να παραμείνει τουλάχιστον με τα ίδια στοιχεία που απαιτούνται από το χρήστη για την εγγραφή του στο σύστημα. Το Joomla όμως κρατά και κάποια επιπλέον στοιχεία του χρήστη όπως είναι το όνομά του και ένα username για την είσοδο του στο σύστημα. Αυτό είναι ένα επιπλέον χαρακτηριστικό το οποίο είναι εξυπηρετικό σε περίπτωση που επιθυμούμε αργότερα να βάλουμε και κάποιο forum στο σύστημα ώστε οι χρήστες να συνεργάζονται μεταξύ τους.

Η διαδικασία εγγραφής του peRSSonal δεν επιδέχεται πισωγυρίσματα του χρήστη σε προηγούμενα βήματα. Από την στιγμή που ο χρήστης προχωρά σε επόμενο βήμα δεν είναι δυνατή η μετάβαση σε κάποιο προηγούμενο βήμα. Αυτό είναι κάτι που έχει τροποποιηθεί στο Joomla καθώς ο χρήστης έχει τη δυνατότητα πλέον να μεταβεί σε όποιο από τα 3 βήματα της εγγραφής θέλει άμεσα μέσα από ένα tab panel.

Τέλος ο τρόπος ενεργοποίησης δεν έχει αλλάξει και ο χρήστης μέσα από το email επιβεβαίωσης μπορεί να ενεργοποιήσει το λογαριασμό του.

### **□□□□ Βελτιώσεις στο τρόπο εμφάνισης των στοιχείων του peRSSonal στο Joomla**

Το peRSSonal στηρίζεται αποκλειστικά στη χρήση της Javascript μέσω του Dojo toolkit. Όλα τα στοιχεία του στα modules που έχει φορτώνονται μέσω του Dojo toolkit. Αυτό έχει το θετικό ότι δεν χρειάζεται να φορτωθεί όλη η σελίδα από την αρχή αλλά φορτώνονται τα δεδομένα που αλλάζουν κατάλληλα στα modules.

Το μειονέκτημα σε αυτό είναι ότι αρκετές φορές καθυστερεί να φορτώσει και ίσως εάν προκύψει συγχρονισμός στο φόρτωμα των modules να μην φορτωθούν και καθόλου. Στο Joomla αυτό έχει εκλείψει και γίνεται μερική χρήση της Javascript και της τεχνολογίας AJAX (στην οποία στηρίζεται το Dojo toolkit) για την εμφάνιση και χρησιμοποίηση όλων στοιχείων πραγματικά χρειάζονται. Η αλλαγή αυτή επιφέρει σημαντικές αλλαγές στην φόρτωση άρθρων, αλλά και στην φόρτωση των σελίδων γενικότερα.

## □□□□□ Νέες δυνατότητες που προστέθηκαν στο peRSSonal κατά τη μεταφορά του στο Joomla.

Κάτι το οποίο πραγματικά έλλειπε από το peRSSonal ήταν η διαχείριση του λογαριασμού του χρήστη. Από τη στιγμή που ο χρήστης γράφεται στο σύστημα του peRSSonal δεν υπάρχει η δυνατότητα να διαχειριστεί το λογαριασμό του σε οτιδήποτε. Αυτό το χαρακτηριστικό υπάρχει πλέον στο Joomla όπου ο χρήστης μπορεί να διαχειριστεί τα στοιχεία του προφίλ του, τον κωδικό του, τη γλώσσα που θα χρησιμοποιεί στο peRSSonal.

Παρέχεται λοιπόν πλήρης έλεγχος των στοιχείων του χρήστη και υπάρχει η δυνατότητα μέχρι και αλλαγής του προσωπικού e-mail του χρήστη σε περίπτωση απώλειας. Όλα αυτά τα χαρακτηριστικά υπήρχαν μόνο μέσα από το διαχειριστή του peRSSonal και μόνο ύστερα από επικοινωνία. Τώρα πλέον η δυνατότητα αυτή παρέχεται και στο χρήστη.

## □□□□□ Δυνατότητα προβολής όλων των πρόσθετων ταυτόχρονα.

Στο peRSSonal υπήρχε ένα πρότυπο (template) το οποίο παρέπεμπε σε διαφορετικές σελίδες κάθε φορά. Για παράδειγμα από την στιγμή που ο χρήστης επέλεγε να δει ένα άρθρο τότε άνοιγε μια διαφορετική σελίδα που έδινε τη δυνατότητα να δει το άρθρο, να επιλέξει το αντίστοιχο tagging των λέξεων για το άρθρο. Από τη στιγμή που επέλεγε το άρθρο για να επιλέξει να δει κάποιο άλλο άρθρο έπρεπε να μεταβεί στην προηγούμενη σελίδα με το αντίστοιχο back κουμπί του browser.

Πλέον όλα τα modules είναι ενσωματωμένα στο template και ενεργοποιούνται κατάλληλα ώστε ο χρήστης να μπορεί να τα έχει όλα άμεσα στην οθόνη του. Όταν για παράδειγμα επιλέγει ο χρήστης να δει κάποιο άρθρο τότε το αντίστοιχο module του tagging εμφανίζει τις λέξεις κλειδιά, αλλά το module που εμφανίζει τα τελευταία νέα υπάρχει ήδη ανοιχτό επίσης και ο χρήστης μπορεί εύκολα να επιλέξει κάποιο άλλο άρθρο για να διαβάσει.

Η τροποποίηση αυτή επιφέρει σημαντικές αλλαγές και η εμφάνισή του νέου peRSSonal στο Joomla μπορεί να εμφανίζεται πολύ όμορφα εάν φτιαχτεί ένα κατάλληλο template όπου να μπορούν τα modules να μοιραστούν δεξιά και αριστερά όπως στο αρχικό template που έχει το peRSSonal και δεν χρειάζεται ο χρήστης να κάνει back και forward σε καμία σελίδα στο peRSSonal, αλλά όλη η πληροφορία υπάρχει μπροστά του από όλα τα modules.

## □□□□□ Προς τα πίσω συμβατότητα του νέου peRSSonal στο Joomla με το παλιό peRSSonal.

Ένα σημαντικό χαρακτηριστικό του peRSSonal όπως μεταφέρθηκε στο Joomla είναι ότι δεν χάνεται η συμβατότητα με το παλιό σύστημα του peRSSonal. Για την ακρίβεια όλη η βάση του τρέχοντος συστήματος του peRSSonal χρησιμοποιείται αποκλειστικά και από το νέο μας σύστημα του peRSSonal που δουλεύει μέσα από το Joomla. Όλες οι αλλαγές αποθηκεύονται στους ίδιους πίνακες και άρα ο τρόπος με τον οποίο δουλεύει το παλιό σύστημα δεν έχει αλλάξει καθόλου.

Έτσι λοιπόν όταν ο χρήστης γραφτεί στο peRSSonal μέσα από το Joomla άμεσα αποκτά πρόσβαση και στο παλιό peRSSonal και μπορεί να το χρησιμοποιεί ταυτόχρονα με παλιό. Το μόνο που χρειάζεται να γίνει είναι να φτιαχτεί μια κεντρική σελίδα που να παραπέμπει το χρήστη σε ποιο σύστημα θέλει να εισέλθει. Άρα ο χρήστης δεν χάνει τη δυνατότητα να συνεχίσει να χρησιμοποιεί το παλιό peRSSonal αλλά οι αλλαγές φαίνονται άμεσα και στα δύο συστήματα. Έτσι δεν χάνεται καμία πληροφορία μεταξύ των δύο συστημάτων.

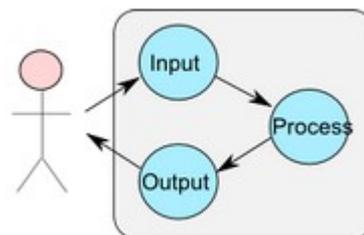
Στο επόμενο κεφάλαιο θα αναλύσουμε τα στοιχεία που άλλαξαν στο Joomla και τα στοιχεία που προστέθηκαν ώστε να υποστηρίξει το peRSSonal.

## 6 ΤΑ ΣΥΣΤΑΤΙΚΑ ΚΑΙ ΠΡΟΣΘΕΤΑ ΣΤΟΙΧΕΙΑ ΠΟΥ ΔΗΜΙΟΥΡΓΗΘΗΚΑΝ ΣΤΟ JOOMLA ΏΣΤΕ ΝΑ ΥΠΟΣΤΗΡΙΞΕΙ ΤΟ PERSONAL.

Σε αυτό το κεφάλαιο θα εξετάσουμε όλες τις απαραίτητες τροποποιήσεις που έπρεπε να γίνουν ώστε το Joomla να μπορεί να υποστηρίξει το peRSSonal. Θα αναλύσουμε τα δομικά στοιχεία του Joomla που τροποποιήθηκαν και τα νέα συστατικά στοιχεία που προστέθηκαν ώστε το Joomla να μπορεί να υποστηρίξει το peRSSonal. Σε κάθε συστατικό που προστέθηκε παρατίθεται και ο αντίστοιχος κώδικας του Joomla.

### 6.1 Το MVC μοντέλο δημιουργίας του Joomla

Το Joomla από την έκδοση 1.5 και έπειτα εισήγαγε ένα νέο τρόπο λειτουργίας των συστατικών και πρόσθετων στοιχείων του που ονομάζεται MVC (Model-View-Controller). Το MVC είναι ένα νέο μονοπάτι δημιουργίας που χρησιμοποιείται για να οργανώνει των κώδικα με τέτοιο τρόπο ώστε η επιχειρηματική λογική και η αναπαράσταση των δεδομένων να είναι δύο εντελώς ξεχωριστά κομμάτια. Η προϋπόθεση πίσω από αυτή την προσέγγιση έγκειται στο ότι η επιχειρηματική λογική ομαδοποιείται σε μια περιοχή, ενώ η διεπαφή και η διαδραστικότητα του χρήστη που περικλείει τα δεδομένα μπορεί να αναθεωρηθεί και να προσαρμοστεί δίχως να χρειάζεται να γίνει επαναπρογραμματισμός της επιχειρηματικής λογικής. Η βάση για αυτό το μοντέλο δημιουργίας αποτελείται από 3 βασικούς ρόλους.



Εικόνα 33. Ο τρόπος λειτουργίας του MVC

□□□□□

**Το Μοντέλο**

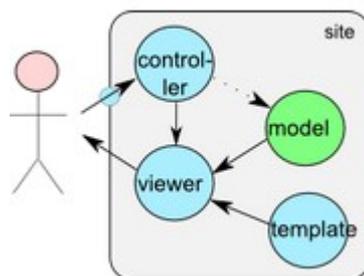
Το μοντέλο ουσιαστικά είναι το στοιχείο που παρέχει ενθυλάκωση στα δεδομένα. Συχνά παρέχει τις ρουτίνες για την παροχή και διαχείριση των απαιτούμενων δεδομένων με πολύ σημαντικό τρόπο. Πληροφορίες που αφορούν εισαγωγή, διαγραφή, τροποποίηση των στοιχείων του component. Επιπλέον περιέχει συναρτήσεις που παρέχουν δεδομένα από τη βάση δεδομένων του χρήστη και συνεπώς του Joomla.

## □□□□ **Η εμφάνιση**

Η εμφάνιση είναι το στοιχείο εκείνο που είναι υπεύθυνο για την απεικόνιση των δεδομένων στο χρήστη με τέτοιο τρόπο ώστε να μπορεί εύκολα να αλληλεπιδράσει κατάλληλα. Η εμφάνιση τραβά δεδομένα από το μοντέλο μέσω των συναρτήσεων και των μεθόδων και τροφοδοτεί τα δεδομένα αυτά στο πρότυπο εμφάνισης (template) του Joomla. Σε καμία περίπτωση η εμφάνιση δεν τροποποιεί τα δεδομένα του χρήστη.

## □□□□ **Ο ελεγκτής**

Ο ελεγκτής είναι υπεύθυνος για τις αλληλεπιδράσεις του χρήστη στη σελίδα μας. Ο ελεγκτής ανταποκρίνεται κατάλληλα στα αιτήματα των χρηστών στέλνοντας την κατάλληλη πληροφορία στο μοντέλο ώστε να παρουσιαστούν τα κατάλληλα στοιχεία στο χρήστη μετέπειτα μέσω της εμφάνισης. Σε καμία περίπτωση επίσης δεν εμφανίζει τα στοιχεία στο χρήστη.



**Εικόνα 34. Ο τρόπος λειτουργίας του MVC στο Joomla**

Για τη δημιουργία ενός component ή module χρησιμοποιώντας το MVC υπάρχει άπειρο υλικό στο Internet όπου ο χρήστης μπορεί να ξεκινήσει από το μηδέν και να φτιάξει τα δικά του συστατικά στοιχεία για το Joomla. Στη συνέχεια θα αναλύσουμε τα components που φτιάχτηκαν για το perSSonal στο Joomla.

## **6.2 Το συστατικό στοιχείο *com\_users* του Joomla**

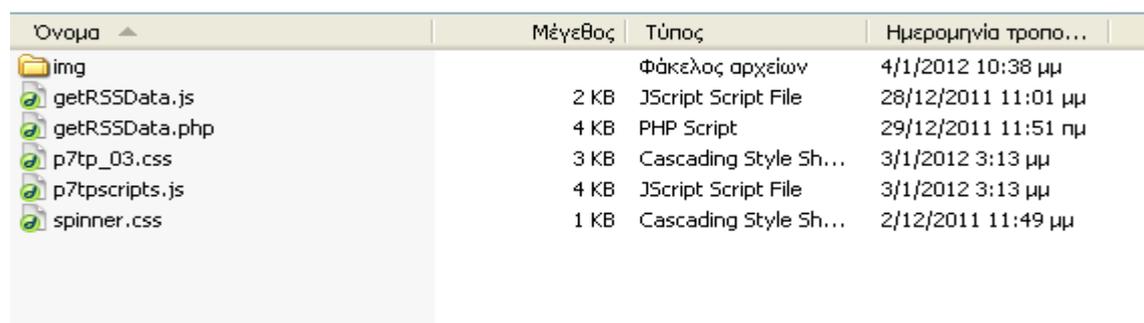
Το Joomla παρέχει ένα συστατικό στοιχείο και ένα πρόσθετο στοιχείο για την εγγραφή και διαχείριση των χρηστών. Τα δύο αυτά στοιχεία συνεργάζονται μεταξύ τους ώστε να επιτυγχάνεται η εγγραφή των χρηστών. Το πρόσθετο στοιχείο είναι υπεύθυνο για την είσοδο των χρηστών στο σύστημα είναι το mod\_login (συνήθως όλα τα πρόσθετα στοιχεία αρχίζουν με το πρόθεμα mod\_) και το αντίστοιχο component το οποίο αντιδρά στις κινήσεις του module αυτού στην περίπτωση εγγραφής, αποστολής κωδικού και διαχείρισης του προφίλ

του χρήστη είναι το component com\_users (συνήθως όλα τα συστατικά στοιχεία του Joomla αρχίζουν με το πρόθεμα com\_).

Για να μπορέσει το Joomla να υποστηρίξει το perSSonal κατά την εγγραφή του χρήστη με τα έξτρα στοιχεία που χρειάζονται έπρεπε να αλλάχθει το συστατικό στοιχείο com\_users. Ο λόγος για τον οποίο δεν φτιάχτηκε ένα νέο είναι διότι το συστατικό στοιχείο αυτό συντηρείται από την ομάδα του Joomla οπότε εάν βρεθεί οποιοδήποτε λάθος τότε ο administrator του perSSonal το μόνο που πρέπει να κάνει είναι να βρει τις αλλαγές που έγιναν στο component αυτό και να τις περάσει στο τροποποιημένο component που έχει φτιαχτεί πλέον για το Joomla.

Σύμφωνα με το MVC μοντέλο έπρεπε να πειράξουμε στο com\_users τα στοιχεία που εμφανίζει στο χρήστη στην περίπτωση εγγραφής στο σύστημα. Έτσι στην περίπτωση του front-end τα στοιχεία που εμφανίζονται στο χρήστη μέσα από το αρχείο default.php έπρεπε να τροποποιηθούν ώστε να συμπεριλάβουν τα νέα στοιχεία του perSSonal. Η τροποποίηση του αρχείου αυτού μεφανίζεται ολοκληρωτικά στον κώδικα του νέου default.php αρχείου εμφανίζεται στον πίνακα 1 του παραρτήματος.

Το αρχείο αυτό πλέον έχει αρκετά δυναμικά στοιχεία Javascript και Cascade Style Sheets ώστε να υποστηρίξει την νέα εμφάνιση εγγραφής του χρήστη του perSSonal στο Joomla. Στην παρακάτω εικόνα εμφανίζονται όλα αυτά τα αρχεία που χρησιμοποιεί το default.php αρχείο



Όνομα	Μέγεθος	Τύπος	Ημερομηνία τροπο...
img		Φάκελος αρχείων	4/1/2012 10:38 μμ
getRSSData.js	2 KB	JScript Script File	28/12/2011 11:01 μμ
getRSSData.php	4 KB	PHP Script	29/12/2011 11:51 μμ
p7tp_03.css	3 KB	Cascading Style Sh...	3/1/2012 3:13 μμ
p7tpscripts.js	4 KB	JScript Script File	3/1/2012 3:13 μμ
spinner.css	1 KB	Cascading Style Sh...	2/12/2011 11:49 μμ

**Εικόνα 35. Τα αρχεία που χρησιμοποιεί το default.php αρχείο του registration του com\_users**

Για την δυναμική εμφάνιση των RSS κατηγοριών που επιλέγει ο χρήστης πλέον χρησιμοποιούνται τα αρχεία getRSSData.php και getRSSData.js που μέσω AJAX εμφανίζουν στο χρήστη την πληροφορία δίχως να χρειάζεται να πάει στο επόμενο βήμα όπως στο παλαιό perSSonal. Ο κώδικας των 2 αρχείων εμφανίζεται στους παρακάτω πίνακες 2-3 του παραρτήματος.

Τα υπόλοιπα αρχεία χρησιμοποιούνται για την εμφάνιση και χρησιμοποίηση ενός tab panel που περιέχει τα 3 βήματα της εγγραφής του χρήστη στο σύστημα. Το tab panel αυτό δίνει τη δυνατότητα δυναμικής μετάβασης σε οποιοδήποτε βήμα της εγγραφής του χρήστη στο perSSonal μέσα από το Joomla.

Επιπλέον ο τρόπος με τον οποίο το μοντέλο του com\_users αντιδρά κατά την εγγραφή ενός νέου χρήστη έπρεπε να τροποποιηθεί ώστε να συμπεριλάβει και τα νέα στοιχεία εγγραφής στο σύστημα εξαιτίας του perSSonal. Έτσι στον αντίστοιχο folder του μοντέλου του perSSonal έπρεπε να τροποποιηθεί το αρχείο registration.php όπως φαίνεται στο παράρτημα στον πίνακα 4.

Επιπλέον έπρεπε να αλλάχθει το αρχείο του μοντέλου που διαχειρίζεται το προφίλ του χρήστη σε περίπτωση μεταβολής των στοιχείων ώστε να διαχειρίζεται τις αλλαγές και το παλαιό perSSonal σύστημα. Έτσι το αντίστοιχο αρχείο profile.php μετατράπηκε όπως εμφανίζεται στον πίνακα 5.

Ως προς το back-end κομμάτι του com\_users το μόνο που χρειάζεται να γίνει είναι η τροποποίηση ώστε εάν ο διαχειριστής του Joomla αποφασίσει να διαγράψει ένα χρήστη αυτός να διαγραφεί και από το peRSSonal εάν υπάρχει. Επίσης εάν μπλοκάρει κάποιο χρήστη θα πρέπει να μπλοκάρεται και από το peRSSonal Έτσι λοιπόν τροποποιήθηκαν κάποιοι μέθοδοι του αρχείου user.php όπως φαίνεται στον πίνακα 6.

Τέλος για να γίνει εγκατάσταση το τροποποιημένο component com\_users φτιάχτηκε και ένα installation.xml αρχείο όπως φαίνεται στον πίνακα 7.

### **6.3 Το component com\_perssonal του Joomla για να υποστηρίξει τις λειτουργίες του peRSSonal**

Το Joomla θα πρέπει να παρέχει ένα component το οποίο θα αντιδρά σε οτιδήποτε στέλνεται ως αίτηση από χρήστες του peRSSonal που χρησιμοποιούν τα αντίστοιχα modules. Για παράδειγμα ένα module μπορεί να επιλέξει την προβολή ενός άρθρου, την εμφάνιση του ανά κατηγορία ή την εμφάνιση των άρθρων των χρηστών. Για το λόγο αυτό φτιάχτηκε ένα νέο component το οποίο ονομάστηκε com\_perssonal και ανταποκρίνεται σε όλες τις αιτήσεις που επιθυμούν τα επιμέρους modules του peRSSonal.

Ως προς το front-end υπάρχει ένας controller που αναλαμβάνει να παίξει και το ρόλο της εμφάνισης των περιεχομένων και περιλαμβάνεται στο αρχείο perssonal.php και περιέχει τον κώδικα που περιλαμβάνεται στον πίνακα 8.

Το αρχείο αυτό κάνει χρήση και 2 κλάσεων για την χρονομέτρηση και εμφάνιση ημερομηνιών που περιλαμβάνονται στο αρχείο perssonal.html.php και ο κώδικάς τους περιλαμβάνεται στον πίνακα 9. Η χρονομέτρηση βοηθά στο να γνωρίζει ο χρήστης πόσο χρόνο διήρησε η προβολή ενός άρθρου. Οι μέθοδοι find\_time και find\_date χρησιμοποιούνται από το peRSSonal για την εμφάνιση ημέρας και ώρας στα άρθρα και δεν έχει τροποποιηθεί κάτι σε αυτές.

Επιπλέον χρησιμοποιούνται και κάποια βοηθητικά αρχεία από το component com\_perssonal προκειμένου να κάνουν reject άρθρα που ανήκουν σε κάποιο χρήστη και εμφανίζονται στην επιλογή 'τα άρθρα μου' καθώς και την επιλογή που εμφανίζουν τις εικόνες ενός άρθρου. Οι κώδικες για τα βοηθητικά αρχεία αυτά αντιστοιχούν στους πίνακες 10-13.

Τώρα ως προς το back-end κομμάτι του component com\_perssonal τα αρχεία perssonal.php και perssonal.html.php αναλαμβάνουν να εμφανίσουν στο χρήστη τη γλώσσα επιλογής και τους χρήστες που είναι εγγεγραμμένοι στο peRSSonal και όχι στο Joomla σε περίπτωση που ο διαχειριστής επιθυμεί να κάνει μεταφορά των χρηστών επιλέγοντας είτε ένα ένα είτε ολονών.

Έτσι λοιπόν ο κώδικας για το αντίστοιχο αρχείο perssonal.php για το back-end interface του διαχειριστή που αναλαμβάνει να κάνει αυτή τη δουλειά περιλαμβάνεται στον πίνακα 14. Το αντίστοιχο αρχείο που κάνει την εγκατάσταση του component φαίνεται στον παρακάτω 15.

Στις επόμενες ενότητες θα αναλύσουμε τα επιμέρους πρόσθετα στοιχεία του peRSSonal που χρειάστηκε να αναπτύξουμε ώστε να υποστηρίξει το Joomla το peRSSonal.

### **6.4 Το πρόσθετο στοιχείο mod\_perssonalcategories**

Το πρόσθετο αυτό στοιχείο είναι υπεύθυνο για την προβολή όλων των κατηγοριών που έχει εκδηλώσει ο χρήστης ενδιαφέρον κατά την εγγραφή, των άρθρων του, και όλων των άρθρων που έχει επιλέξει. Για τη δημιουργία του module αυτού όπως και κάθε module

απαιτούνται 3 βασικά αρχεία που αποτελούν τα δομικά στοιχεία του MVC μοντέλου. Το ένα είναι ο Controller μας (συνήθως είναι ένα php αρχείο που έχει την ίδια ονομασία με το module), το άλλο αρχείο είναι το model που μας παρέχει την πληροφορία που θέλουμε να έχει αυτό το μοντέλο (συνήθως είναι ένα php αρχείο που ονομάζεται helper.php) και τέλος υπάρχει και ο viewer μας για την απεικόνιση των δεδομένων που θέλουμε στο χρήστη (συνήθως είναι ένα php αρχείο που ονομάζεται default.php). Σε κάθε module, όπως και σε κάθε component, υπάρχει ένα xml αρχείο το οποίο παρέχει τις κατάλληλες πληροφορίες στο Joomla για το πώς θα γίνει η εγκατάσταση στο Joomla.

Η λογική αυτή ακολουθείται σε όλα τα modules που έχουμε υλοποιήσει και είναι απόλυτα συνυφασμένη με τη λειτουργία του MVC μοντέλου απεικόνισης. Στο module αυτό επιλόγονται οι κατηγορίες από αρχείο helper.php σε μια μεταβλητή πίνακα \$list η οποία χρησιμοποιείται αργότερα από το viewer για να εμφανίσει τις κατηγορίες του χρήστη. Κάθε αίτηση του module εξυπηρετείται από το component com\_personal που περιγράψαμε παραπάνω. Οι κώδικες για το πρόσθετο αυτό στοιχείο εμφανίζονται στους πίνακες 16-19.

## **6.5 Το πρόσθετο στοιχείο *mod\_personallatestnews***

Στο module αυτό επιλόγονται τα τελευταία νέα σύμφωνα τον τρόπο που επιλέγονταν από το παλιό σύστημα personal. Αυτά εμφανίζονται στο πρότυπο εμφάνισης του Joomla σε μια λίστα με την μορφή ημερομηνία-ώρα και από κάτω ο τίτλος. Τα αρχεία που συμμετέχουν στο module αυτό έχουν είναι παρόμοια με το προηγούμενο module και επιτελούν παρόμοιες λειτουργίες.

Το αρχείο helper.php αποθηκεύει τα τελευταία άρθρα σε μια μεταβλητή τύπου πίνακα \$list η οποία χρησιμοποιείται αργότερα από το viewer για να εμφανίσει τα νέα αυτά στο χρήστη. Κάθε αίτηση του module για την προβολή ενός άρθρου σε περίπτωση που επιλεγεί από το χρήστη εξυπηρετείται και αυτή από το component com\_personal που περιγράψαμε παραπάνω. Οι κώδικες των αρχείων που υλοποιούν το module αυτό φαίνονται στους πίνακες 20-23.

## **6.6 Το πρόσθετο στοιχείο *mod\_personalpopulararticles***

Στο module αυτό επιλόγονται τα πιο δημοφιλή νέα και εμφανίζονται με παρόμοιο τρόπο στο χρήστη μέσα από το πρότυπο του Joomla. Έτσι το αρχείο helper.php αποθηκεύει σε μια μεταβλητή τύπου πίνακα \$list τα πιο δημοφιλή νέα και ύστερα η μεταβλητή αυτή χρησιμοποιείται αργότερα από το viewer για να εμφανίσει τα δεδομένα αυτά στο χρήστη. Κάθε αίτηση του module εξυπηρετείται και αυτή από το component com\_personal που περιγράψαμε παραπάνω. Έτσι εάν ο χρήστης επιθυμεί να δει κάποιο άρθρο από τα δημοφιλή που εμφανίζονται τότε το συστατικό στοιχείο com\_personal αναλαμβάνει να το εμφανίσει στο χρήστη. Οι κώδικες που υλοποιούν το συστατικό στοιχείο αυτό φαίνονται στους πίνακες 24-27.

## **6.7 Το πρόσθετο στοιχείο *mod\_personalreaderspicks***

Στο module αυτό το αρχείο helper.php αποθηκεύει σε μια μεταβλητή πίνακα \$list τις επιλογές ανάγνωσης του χρήστη και αργότερα η μεταβλητή αυτή χρησιμοποιείται από το viewer για να εμφανίσει τα δεδομένα αυτά στο χρήστη. Κάθε αίτηση του module εξυπηρετείται και εδώ από το component com\_perSSonal που περιγράψαμε παραπάνω. Οι κώδικες που υλοποιούν το συστατικό στοιχείο αυτό φαίνονται στους πίνακες 28-31.

### **6.8 Το πρόσθετο στοιχείο *mod\_perSSonalsimilararticles***

Στο module αυτό το αρχείο helper.php αποθηκεύει σε μια μεταβλητή πίνακα \$list τα άρθρα εκείνα που είναι παρόμοια με αυτά που βλέπει ο χρήστης και αργότερα η μεταβλητή αυτή χρησιμοποιείται από το viewer για να εμφανίσει τα δεδομένα αυτά στο χρήστη. Κάθε αίτηση και αυτού του module εξυπηρετείται από το component com\_perSSonal που περιγράψαμε παραπάνω. Οι κώδικες που υλοποιούν το συστατικό στοιχείο mod\_perSSonalsimilararticles φαίνονται στους πίνακες 32-36.

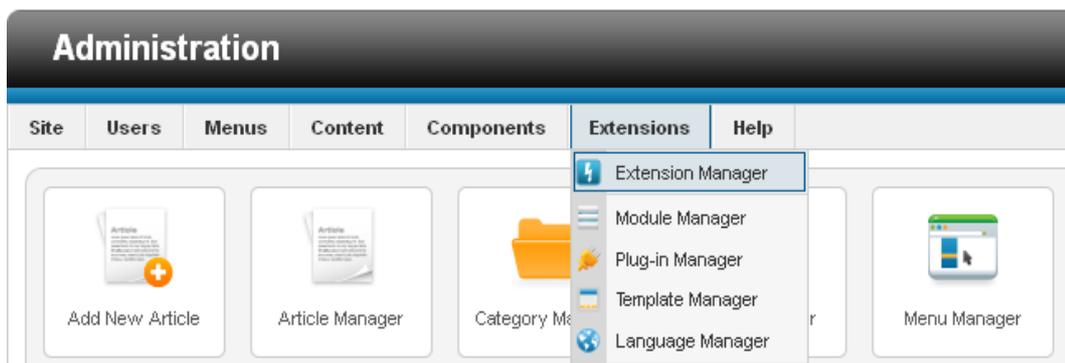
### **6.9 Το πρόσθετο στοιχείο *mod\_perSSonaltagging***

Το module αυτό ενεργοποιείται τη στιγμή που επιλέγεται κάποιο άρθρο για προβολή όπου εμφανίζονται οι λέξεις κλειδιά σύμφωνα με τις οποίες ο χρήστης έχει τη δυνατότητα να επιλέξει εάν αποτελούν λέξεις κλειδιά της αρεσκείας του ή όχι. Η εμφάνιση των λέξεων γίνεται σε μια λίστα όπου στα δεξιά κάθε λέξης εμφανίζονται δύο κουμπιά + και - που βοηθούν το χρήστη να επιλέξει εάν η λέξη αυτή έχει θετική ή αρνητική σημασία για αυτόν.

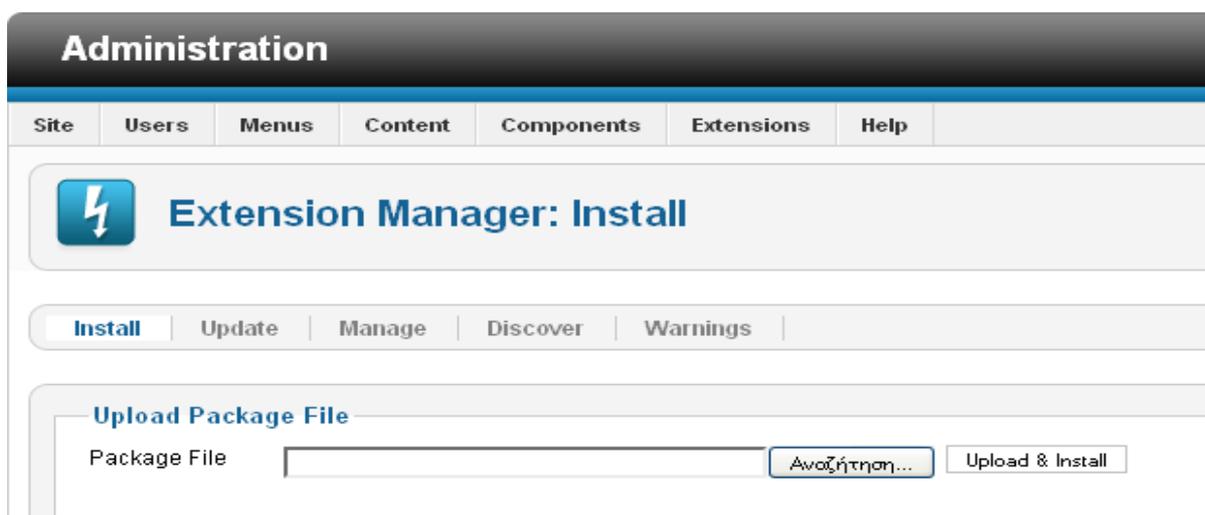
Στο module αυτό το αρχείο helper.php αποθηκεύει σε μια μεταβλητή τύπου πίνακα \$list τις λέξεις κλειδιά σύμφωνα με το άρθρο που επέλεξε ο χρήστης για προβολή και αργότερα η μεταβλητή αυτή χρησιμοποιείται από το viewer για να εμφανίσει όλες τις λέξεις στο χρήστη. Το module αυτό χρησιμοποιεί τα βοηθητικά αρχεία alter.js και change\_kw.php ώστε να μετατρέψει το βάρος κάθε λέξης ανάλογα με τις προτιμήσεις του χρήστη. Οι κώδικες των αρχείων αυτών εμφανίζονται στους πίνακες 37-41.

### **6.10 Εγκατάσταση και ρύθμιση όλων των *modules και components του peRSSonal στο Joomla***

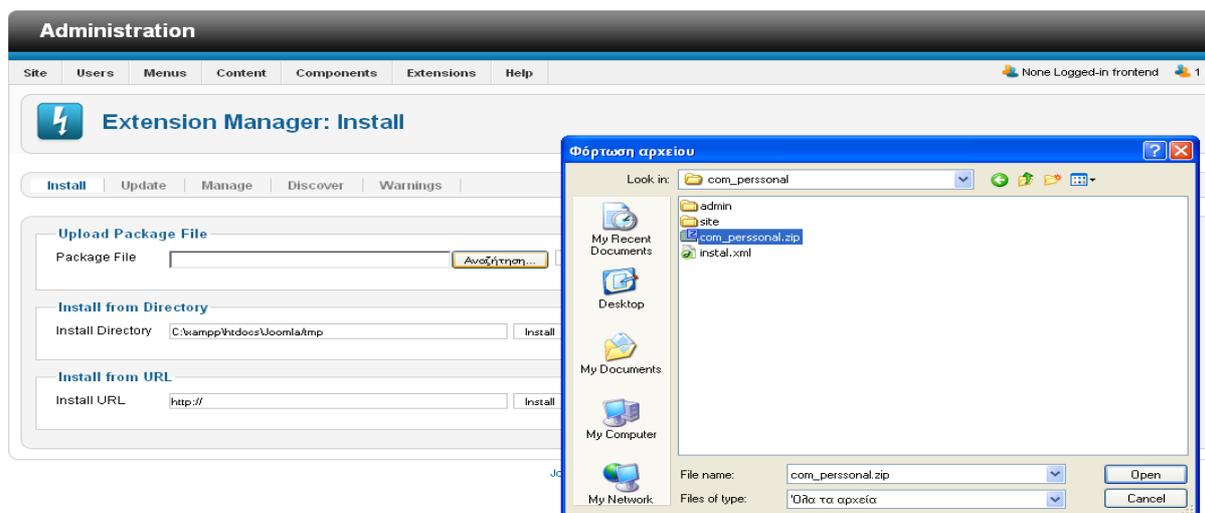
Για να δουλέψουν όλα τα components και modules που έχουμε φτιάξει ως τώρα θα πρέπει να τα εγκαταστήσουμε στο Joomla. Αφού εισέλθουμε στο περιβάλλον του διαχειριστή επιλέγουμε από το μενού Extensions->Extension Manager όπως φαίνεται στις παρακάτω εικόνες.



Εικόνα 36. Επιλογή του διαχειριστή επεκτάσεων του Joomla



Εικόνα 37. Ο διαχειριστής επεκτάσεων του Joomla



Εικόνα 38. Επιλογή οποιασδήποτε επέκτασης σε zip για εγκατάσταση στο Joomla

Από τη στιγμή που εγκαταστήσουμε όλα τα components και modules που έχουμε φτιάξει θα πρέπει να ρυθμίσουμε τον τρόπο που θα εμφανίζονται τα modules στο πρότυπο εμφάνισης του Joomla. Επιλέγουμε λοιπόν Extensions->Module Manager για να ανοίξουμε το διαχειριστή των υπομονάδων του Joomla όπως φαίνεται από την παρακάτω εικόνα.

<input type="checkbox"/>	Title	Status	Position	Ordering	Type	Pages	Acc
<input type="checkbox"/>	peRSSonal Similar Articles	None	:: None ::	0	peRSSonal Similar Articles	None	Pu
<input type="checkbox"/>	peRSSonal Latest News	None	:: None ::	0	peRSSonal Latest News	None	Pu
<input type="checkbox"/>	peRSSonal Readers Pick	None	:: None ::	0	peRSSonal Readers Pick	None	Pu
<input type="checkbox"/>	peRSSonal Tagging	None	:: None ::	0	peRSSonal Tagging	None	Pu
<input type="checkbox"/>	peRSSonal Categories	None	:: None ::	0	peRSSonal Categories	None	Pu
<input type="checkbox"/>	peRSSonal Popular Articles	None	:: None ::	0	peRSSonal Popular Articles	None	Pu
<input type="checkbox"/>	Breadcrumbs	Published	position-2	1	Breadcrumbs	All	Pu
<input type="checkbox"/>	Main Menu	Published	position-7	1	Menu	All	Pu
<input type="checkbox"/>	Login Form	Published	position-7	7	Login	All	Pu

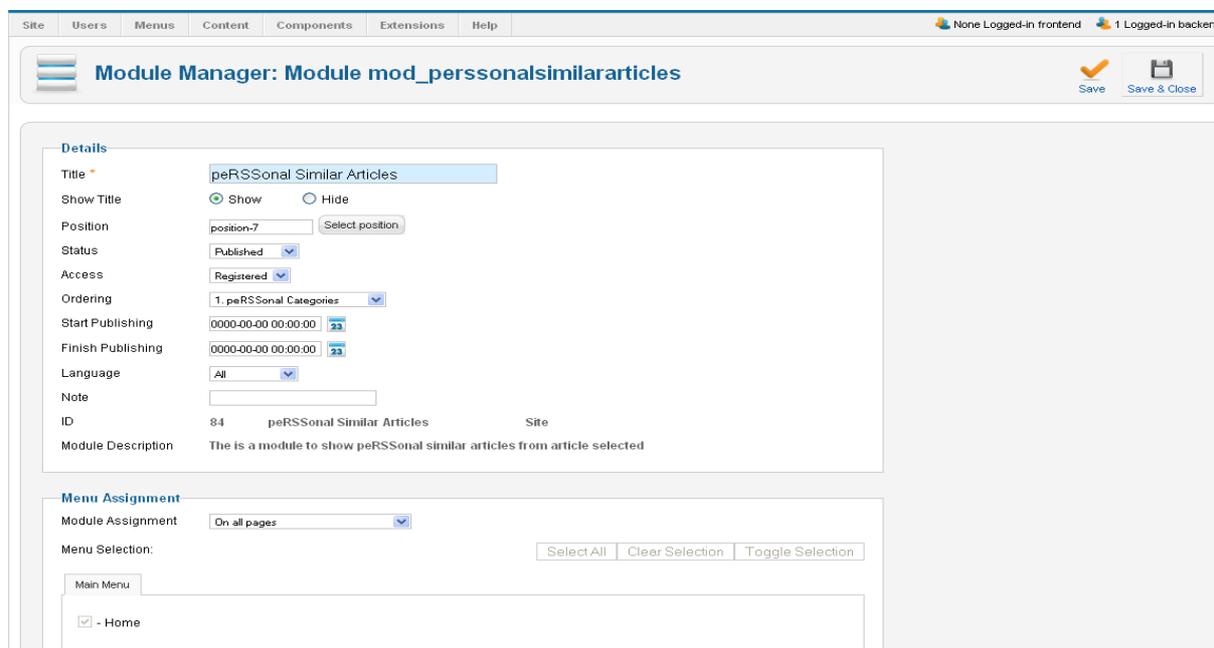
**Εικόνα 39. Ο διαχειριστής υπομονάδων του Joomla**

Παρατηρούμε ότι όλα τα modules που εγκαταστήσαμε είναι απενεργοποιημένα. Αρχικά τα επιλέγουμε και πατάμε publish (δημοσίευση) ώστε πλέον να είναι ενεργά, δηλαδή δημοσιευμένα όπως στην παρακάτω εικόνα.

<input type="checkbox"/>	Title	Status	Position	Ordering	Type	Pages
<input type="checkbox"/>	peRSSonal Similar Articles	Published	:: None ::	0	peRSSonal Similar Articles	None
<input type="checkbox"/>	peRSSonal Latest News	Published	:: None ::	0	peRSSonal Latest News	None
<input type="checkbox"/>	peRSSonal Readers Pick	Published	:: None ::	0	peRSSonal Readers Pick	None
<input type="checkbox"/>	peRSSonal Tagging	Published	:: None ::	0	peRSSonal Tagging	None
<input type="checkbox"/>	peRSSonal Categories	Published	:: None ::	0	peRSSonal Categories	None
<input type="checkbox"/>	peRSSonal Popular Articles	Published	:: None ::	0	peRSSonal Popular	None

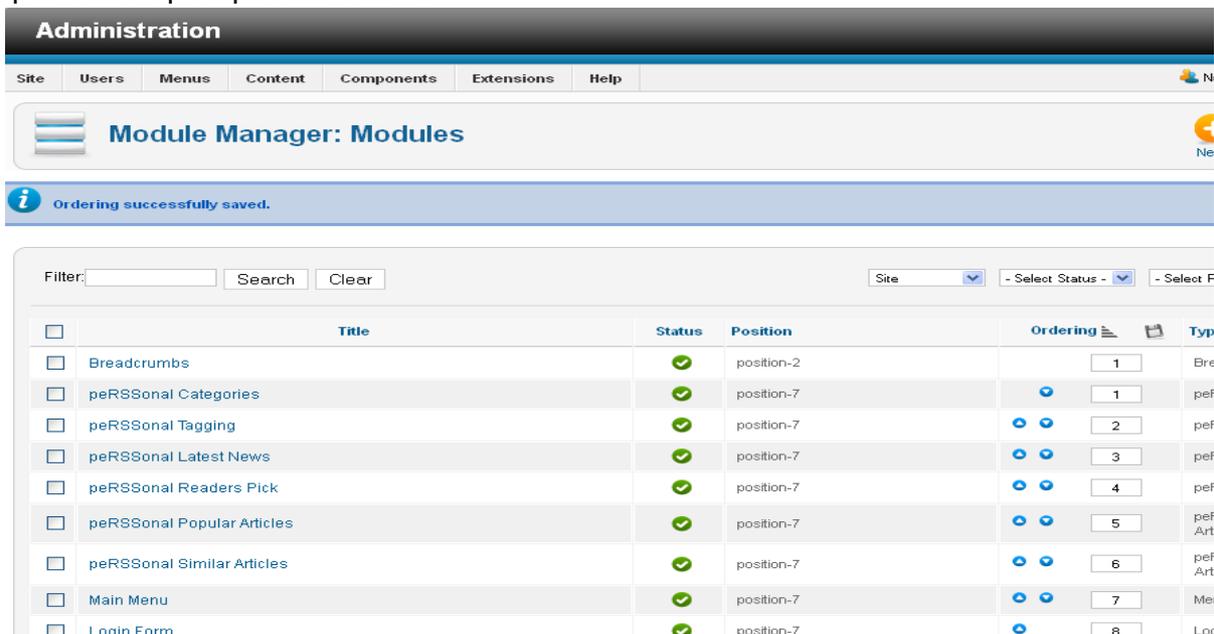
**Εικόνα 40. Δημοσίευση υπομονάδων του Joomla**

Ύστερα θα πρέπει να επιλέξουμε ένα ένα τα modules και πατώντας το edit να τα τροποποιήσουμε ώστε να καθορίσουμε τη θέση που θα εμφανίζονται στο template, για ποιους χρήστες και σε ποιες σελίδες θα εμφανίζονται όπως φαίνεται στην παρακάτω εικόνα.



**Εικόνα 41. Ρυθμίσεις υπομονάδων για τη θέση τους στο template, το επίπεδο πρόσβασης και σελίδες που ανατίθενται στο Joomla**

Αφού επιλέξουμε και αυτές τις ρυθμίσεις το σύστημά μας είναι έτοιμο προς χρήση! Το μόνο που χρειάζεται να ορίσουμε στο Joomla είναι η σειρά εμφάνισής τους (ordering) όπως φαίνεται στην παρακάτω εικόνα



**Εικόνα 42. Ρυθμίσεις σειράς εμφάνισης των modules στο Joomla**

## 6.11 Η τελική μορφή του peRSSonal στο Joomla

Έχουμε πλέον ολοκληρώσει οτιδήποτε χρειάζεται για να δουλέψει το peRSSonal. Ας δούμε τώρα ένα ένα τα βήματα και την τελική μορφή του στο Joomla.

### □□□□□□ Εγγραφή χρηστών στο peRSSonal του Joomla

Όταν ένας χρήστης επιλέγει πλέον να γραφτεί στο peRSSonal δεν συναντά τη γνωστή φόρμα του Joomla, αλλά μια πολύ πιο εξειδικευμένη και ευπαρουσίαστη φόρμα που του επιτρέπει να γραφτεί πλέον στο σύστημα του peRSSonal όπως φαίνεται στην παρακάτω εικόνα.

You are here: Home

#### Main Menu

» Home

#### Login Form

User Name

Password

Remember Me

> Log in

[Forgot your password?](#)

[Forgot your username?](#)

[Create an account](#)

peRSSonal Registration

Basic Setup | Advanced Setup | Personal Information

Select language to use for personal English

Next

Please provide basic information about the number of articles that you want to see when you visit peRSSonal

articles per page: 10

Please provide information about how much you like the categories of the system.  
IMPORTANT: You will be presented news from the categories that you will point as 0 (ZERO) or bigger. If you don't want to get articles from one or more categories please indicate with negative (-1 ... -5).

Business	0	↔
Entertainment	0	↔
Health	0	↔
Politics	0	↔
Science	0	↔
Education	0	↔
Technology	0	↔

Εικόνα 43. Εγγραφή του χρήστη στο peRSSonal του Joomla

Παρόλο που το Joomla δεν παρέχει άμεση μετάφραση η επιλογή γλώσσας για το peRSSonal έχει προστεθεί σαν δυνατότητα και πλέον ο χρήστης επιλέγει τη γλώσσα εγκατάστασης που θέλει από την αντίστοιχη σημαία που βρίσκεται στο πάνω μέρος.

Τα βήματα της εγκατάστασης έχουν τροποποιηθεί ώστε ο χρήστης πρώτα να εισάγει όλες τις απαραίτητες πληροφορίες που επιθυμεί για το peRSSonal και στο τέλος να εισάγει τις προσωπικές του πληροφορίες. Στο 1<sup>ο</sup> tab επιλέγει τις βασικές ρυθμίσεις του peRSSonal σχετικά με το ποιες κατηγορίες τον ενδιαφέρουν, πόσα νέα θέλει να διαβάζει και τη γλώσσα εμφάνισης του peRSSonal.

Στο 2<sup>ο</sup> βήμα μεταβαίνει είτε από το κουμπί Next είτε από την επιλογή του 2<sup>ου</sup> tab. Με την επιλογή του 2<sup>ου</sup> tab άμεσα εμφανίζονται οι πληροφορίες για τις κατηγορίες που έχει επιλέξει όπως φαίνεται στην παρακάτω εικόνα.

You are here: Home

## Main Menu

Home

## Login Form

User Name

Password

Remember Me

[Forgot your password?](#)

[Forgot your username?](#)

[Create an account](#)

peRSSonal Registration 

Basic Setup | **Advanced Setup** | Personal Information

[Previous](#) [Next](#)

Please provide analytical information for each of the categories that you have selected with positive number. The information that is essential for each of the categories is: RSS feeds that you would like to monitor, keywords that you would like to monitor, check the last checkbox if you want to see articles only from the RSS feeds that you have provided.

INSTRUCTIONS:

- In order to enter your own RSS feeds. Please copy and paste the RSS URI into the textarea next to the RSS FEED indicator. if you have multiple RSS feeds please separate them with a new line (ENTER). The RSS feeds are entered by category.
- In order to enter your own keywords please start typing them in the field next to the KEYWORDS indicator. Please type SLOWLY because as you type suggestions on the keyword (WORD ROOT) will appear. At this stage you have to select one of the suggested keywords. Keywords that do not match with the suggested ones will not be used by the system.

**peRSSonal loading**  
■ ■ ■ ■ ■

Loading

View only news from the RSS that you will enter! Do not check it if you do not enter RSS or if you want to have large number of news from all over the world.

Εικόνα 44. Φόρτωση των κατηγοριών του peRSSonal μέσω AJAX

## Main Menu

Home

## Login Form

User Name

Password

Remember Me

[Forgot your password?](#)

[Forgot your username?](#)

[Create an account](#)

peRSSonal Registration 

Basic Setup | **Advanced Setup** | Personal Information

[Previous](#) [Next](#)

Please provide analytical information for each of the categories that you have selected with positive number. The information that is essential for each of the categories is: RSS feeds that you would like to monitor, keywords that you would like to monitor, check the last checkbox if you want to see articles only from the RSS feeds that you have provided.

INSTRUCTIONS:

- In order to enter your own RSS feeds. Please copy and paste the RSS URI into the textarea next to the RSS FEED indicator. if you have multiple RSS feeds please separate them with a new line (ENTER). The RSS feeds are entered by category.
- In order to enter your own keywords please start typing them in the field next to the KEYWORDS indicator. Please type SLOWLY because as you type suggestions on the keyword (WORD ROOT) will appear. At this stage you have to select one of the suggested keywords. Keywords that do not match with the suggested ones will not be used by the system.

Business	
RSS Feeds	<input type="text"/>
Keywords	οικονομ-42625 (24.83) <input type="button" value="Add"/>
	<input type="text"/>
	<input type="button" value="Clear"/>

View only news from the RSS that you will enter! Do not check it if you do not enter RSS or if you want to have large number of news from all over the world.

Εικόνα 45. Άμεση φόρτωση των δεδομένων των κατηγοριών του peRSSonal που μας ενδιαφέρουν

Επιλέγουμε τις κατηγορίες των RSS που μας ενδιαφέρουν και τις λέξεις κλειδιά με τον ίδιο ακριβώς τρόπο που γίνονταν και στο παλιό peRSSonal όπως φαίνεται και στην παρακάτω εικόνα.

Main Menu

- Home

Login Form

User Name

Password

Remember Me

[Log in](#)

[Forgot your password?](#)  
[Forgot your username?](#)  
[Create an account](#)

peRSSonal Registration

Basic Setup | Advanced Setup | Personal Information

[Previous](#) [Next](#)

Please provide analytical information for each of the categories that you have selected with positive number. The information that is essential for each of the categories is: RSS feeds that you would like to monitor, keywords that you would like to monitor, check the last checkbox if you want to see articles only from the RSS feeds that you have provided.

INSTRUCTIONS:

- In order to enter your own RSS feeds. Please copy and paste the RSS URI into the textarea next to the RSS FEED indicator. If you have multiple RSS feeds please separate them with a new line (ENTER). The RSS feeds are entered by category.
- In order to enter your own keywords please start typing them in the field next to the KEYWORDS indicator. Please type SLOWLY because as you type suggestions on the keyword (WORD ROOT) will appear. At this stage you have to select one of the suggested keywords. Keywords that do not match with the suggested ones will not be used by the system.

**Business**

RSS Feeds

Keywords  [Add](#)

[Clear](#)

View only news from the RSS that you will enter! Do not check it if you do not enter RSS or if you want to have large number of news from all over the world.

**Εικόνα 46. Επιλογή των RSS και λέξεων κλειδιών που μας ενδιαφέρουν στο peRSSonal**

Τέλος στο 3<sup>ο</sup> βήμα δίνουμε τις προσωπικές μας πληροφορίες για να ολοκληρώσουμε το βήμα της εγγραφής στο peRSSonal όπως φαίνεται παρακάτω.

You are here: Home

Main Menu

- Home

Login Form

User Name

Password

Remember Me

[Log in](#)

[Forgot your password?](#)  
[Forgot your username?](#)  
[Create an account](#)

peRSSonal Registration

Basic Setup | Advanced Setup | Personal Information

[Previous](#)

Name: \*

Username: \*

Password: \*

Confirm password: \*

E-mail: \*

Confirm e-mail: \*

\* Required field

[Register](#) or [Cancel](#)

**Εικόνα 47. Εισαγωγή προσωπικών στοιχείων για την εγγραφή στο peRSSonal**

Ύστερα το σύστημα μας στέλνει ένα email για την ενεργοποίηση του λογαριασμού μας. Αφού ενεργοποιήσουμε το λογαριασμό μας πλέον έχουμε πρόσβαση στο peRSSonal και άμεσα αποκτάμε και πρόσβαση στο παλιό σύστημα του peRSSonal. Στην επόμενη ενότητα θα εξετάσουμε πως εμφανίζεται το peRSSonal στο Joomla πλέον.



## Το νέο peRSSonal στο Joomla

Όταν ο χρήστης εισέρχεται στο Joomla μέσω login εισέρχεται στην αρχική σελίδα του peRSSonal που περιλαμβάνει όλα τα modules όπως φαίνεται στην παρακάτω εικόνα.

You are here: Home

user info
Welcome: <b>mpourelo@gmail.com</b>
<a href="#">All Articles</a>
<a href="#">My Articles</a>
Per Category
<a href="#">Business</a>
<a href="#">Entertainment</a>
<a href="#">Politics</a>
<a href="#">Science</a>
<a href="#">Education</a>
<a href="#">Technology</a>

Home

tagging
---------

latest news
31 Jul 11 20:45 - 2011-07-31 20:45:07 sec Libya militants 'constrained' says Liam Fox
31 Jul 11 20:45 - 2011-07-31 20:45:03 sec "Cowboys & Aliens" ties "Smurfs" for box office lead
31 Jul 11 20:37 - 2011-07-31 20:37:27 sec Amazon flies high, may rise 25 percent more: Barron's
31 Jul 11 20:30 - 2011-07-31 20:30:57 sec Sip iced coffee
31 Jul 11 20:15 - 2011-07-31 20:15:27 sec Deprived parts of England will lose out under health reforms - report
31 Jul 11 20:05 - 2011-07-31 20:05:47 sec Obscure Strauss Opera Sparkles at Bard College
31 Jul 11 19:55 - 2011-07-31 19:55:21 sec Cooper set to give up on takeover of Laird
31 Jul 11 19:55 - 2011-07-31 19:55:18 sec Conn. Home Owned by Katharine Hepburn up for Sale
31 Jul 11 19:55 - 2011-07-31 19:55:15 sec Telecty in talks to buy Data Electronics
31 Jul 11 19:34 - 2011-07-31 19:34:09 sec 'Cowboys & Aliens,' 'Smurfs' Tie for No. 1 Spot

reader pick
-------------

Similar Articles
------------------

Popular articles
------------------

Login Form
Hi mpourelo@gmail.com,
<a href="#">Log out</a>

Main Menu
<a href="#">Home</a>
<a href="#">Edit profile</a>

Εικόνα 48. Το νέο peRSSonal στο Joomla

Εάν ο χρήστης επιλέξει ένα άρθρο για εμφάνιση τότε εμφανίζεται όπως στην παρακάτω εικόνα.

You are here: Home

user info
Welcome: <a href="mailto:mpourel@gmail.com">mpourel@gmail.com</a>
<a href="#">All Articles</a>
<a href="#">My Articles</a>
Per Category
<a href="#">Business</a>
<a href="#">Entertainment</a>
<a href="#">Politics</a>
<a href="#">Science</a>
<a href="#">Education</a>
<a href="#">Technology</a>

tagging
health <input type="checkbox"/> <input type="checkbox"/>
will <input type="checkbox"/> <input type="checkbox"/>
the <input type="checkbox"/> <input type="checkbox"/>
hs <input type="checkbox"/> <input type="checkbox"/>
part <input type="checkbox"/> <input type="checkbox"/>
england <input type="checkbox"/> <input type="checkbox"/>
inequ <input type="checkbox"/> <input type="checkbox"/>
manchest <input type="checkbox"/> <input type="checkbox"/>
public <input type="checkbox"/> <input type="checkbox"/>
would <input type="checkbox"/> <input type="checkbox"/>
tackl <input type="checkbox"/> <input type="checkbox"/>
area <input type="checkbox"/> <input type="checkbox"/>
labour <input type="checkbox"/> <input type="checkbox"/>

latest news
<a href="#">31 Jul 11 20:45 - 2011-07-31 20:45:07 sec</a> Libya militants 'constrained' says Liam Fox
<a href="#">31 Jul 11 20:45 - 2011-07-31 20:45:03 sec</a> "Cowboys & Aliens" ties "Smurfs" for box office lead
<a href="#">31 Jul 11 20:37 - 2011-07-31 20:37:27 sec</a> Amazon flies high, may rise 25 percent more: Barron's
<a href="#">31 Jul 11 20:30 - 2011-07-31 20:30:57 sec</a> Sip iced coffee
<a href="#">31 Jul 11 20:15 - 2011-07-31 20:15:27 sec</a> Deprived parts of England will lose out under health reforms - report
<a href="#">31 Jul 11 20:05 - 2011-07-31 20:05:47 sec</a> Obscure Strauss Opera Sparkles at Bard College
<a href="#">31 Jul 11 19:55 - 2011-07-31 19:55:21 sec</a> Cooper set to give up on takeover of Laird
<a href="#">31 Jul 11 19:55 - 2011-07-31 19:55:18 sec</a> Conn. Home Owned by Katharine Hepburn up for Sale
<a href="#">31 Jul 11 19:55 - 2011-07-31 19:55:15 sec</a> Telecity in talks to buy Data Electronics
<a href="#">31 Jul 11 19:34 - 2011-07-31 19:34:09 sec</a> 'Cowboys & Aliens,' 'Smurfs' Tie for No. 1 Spot

reader pick
-------------

Similar Articles
------------------

Popular articles
------------------

Login Form  
Hi [mpourel@gmail.com](mailto:mpourel@gmail.com),  
[Log out](#)

Main Menu  
[Home](#)  
[Edit profile](#)

## Deprived parts of England will lose out under health reforms - report

[Summary](#) | [Pure Text](#) | [Images](#)



Health secretary Andrew Lansley has dismissed observations that money was being taken away from parts of England. Deprived parts of England will lose out to wealthier regions under radical health spending reforms, according to a report by public health experts. Deprived parts of England will lose out to wealthier regions under radical health spending reforms, according to a report by public health experts. Changes to NHS funding formulas will see areas with unhealthy populations - such as those with low life expectancy or where levels of disease

are above average - given less funding to tackle entrenched health inequalities. Instead the government argues that local councils will be given cash and freedom to tackle issues of public health. The result of such a radical proposal, says an investigation by Public Health Manchester for the parliamentary health select committee, is that the NHS primary care trusts (PCTs) in poorer regions would lose cash. The result of such a radical proposal, says an investigation by Public Health Manchester for the parliamentary health select committee, is that the NHS primary care trusts (PCTs) in poorer regions would lose cash. NHS Manchester would see a cut of £42m from its budget to reduce health inequalities. Health trusts in the London borough of Tower Hamlets would lose almost £19m. By contrast, parts of the wealthy south-east gain considerably. Surrey PCT will gain £61m and Hampshire PCT will get another £52m. The report said the decision to reduce the NHS cash earmarked to ensure that people lived longer in poor areas was a political decision. "The reduction of the health inequalities weighting is a ministerial judgment rather than an evidence-based recommendation ... In fact, the decision seems to contradict evidence from the recent DH [Department of Health]-commissioned research on the subject., states a memorandum from Public Health Manchester, Manchester city council and NHS Manchester to MPs on the health select committee. "This change could be interpreted as a reduction in the priority of tackling health inequalities and could be seen as contradicting the aspirations described in the recent white papers, particularly in view of currently worsening health inequalities.". The evidence, unearthed by Labour MP Debbie Abrahams, who has a background in public health, undermines the coalition's claims that its reforms would target the least well-off. Labour said the "shocking figures reveal that [ministers] plans for the NHS will make inequality worse, not better". John Healey, the party's shadow health secretary, said: "[The government is] reducing funding to tackle poor health in the least healthy parts of the country, and shifting it to better-off, healthier areas. Less well-off areas like Manchester, Liverpool and County Durham will be among the biggest losers - with cash transferred instead to Hertfordshire, Hampshire and Surrey. "The Tory plans will hit services that help people stop smoking, promote healthy eating and exercise, and raise awareness about the risks of sexually transmitted diseases.

Page loaded in 304.64 milliseconds

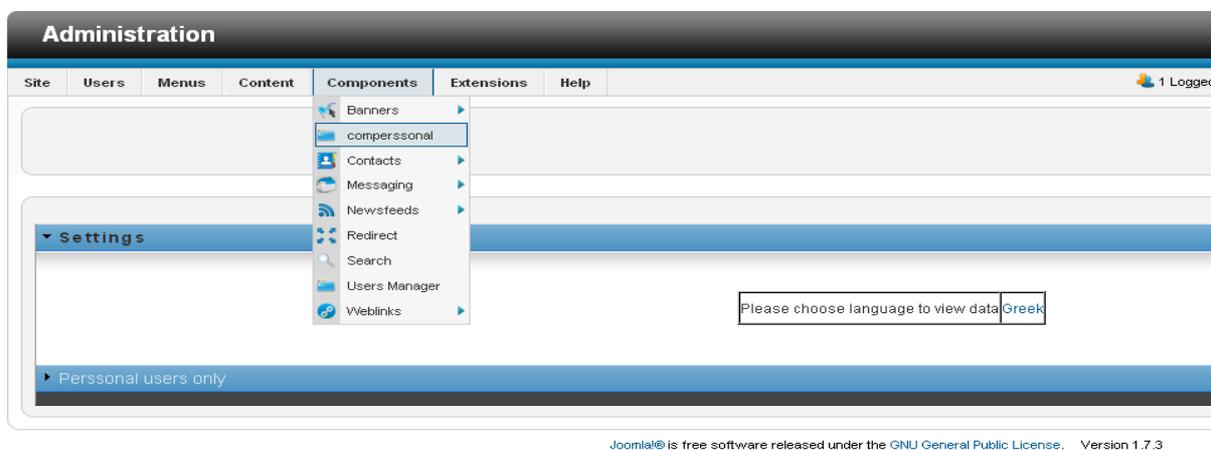
Original article: <http://www.guardian.co.uk/society/2011/jul/31/depr...>

Εικόνα 49. Προβολή άρθρου στο νέο perSSonal του Joomla

Μέσα από τα modules ο χρήστης μπορεί άμεσα να κάνει το αντίστοιχο tagging των λέξεων, ενώ μπορεί να δει συναφή άρθρα με αυτά που έχει επιλέξει, να επιλέξει ένα νέο άρθρο για προβολή και να εμφανίσει τα προσωπικά του δεδομένα. Ας δούμε τέλος και τη διεπαφή για το περιβάλλον του διαχειριστή ως προς το component του perssonal.

## □□□□□ Το περιβάλλον του διαχειριστή του peRSSonal στο Joomla

Όταν ο χρήστης εισέρχεται στο περιβάλλον του διαχειριστή που είναι το back-end interface του Joomla παρέχεται η δυνατότητα να μεταφέρει του χρήστες που είναι ήδη εγγεγραμμένοι στο peRSSonal στο νέο περιβάλλον του Joomla. Από τα components επιλέγουμε Components->comperssional για να μεταβούμε στη διεπαφή του διαχειριστή του component μας όπως φαίνεται και στην παρακάτω εικόνα.



**Εικόνα 50. Προβολή διεπαφής διαχειριστή στο component comperssional στο Joomla**

Μέσα από τη διεπαφή αυτή μπορούμε να αλλάξουμε τη γλώσσα σε Ελληνικά πολύ εύκολα και να επιλέξουμε το tab με τους χρήστες που είναι εγγεγραμμένοι στο peRSSonal μόνο όπως φαίνεται στην παρακάτω εικόνα.



## **Εικόνα 51. Προβολή χρηστών που είναι εγγεγραμμένοι στο παλαιό peRSSonal και πρέπει να μεταφερθούν στο Joomla**

Το σύστημα παρέχει τη δυνατότητα στο διαχειριστή να μεταφέρει ένα ένα ή όλους μαζί τους χρήστες που είναι εγγεγραμμένοι στο peRSSonal στο Joomla. Έτσι ολοκληρώνεται η χρήση όλων των components και modules που φτιάχτηκαν στο Joomla ώστε να υποστηρίξει το peRSSonal.

### **6.12 Συμπεράσματα**

Μέσα από αυτή τη διπλωματική ανακαλύψαμε τον κόσμο του ανοιχτού λογισμικού Joomla που παρέχει απίστευτες δυνατότητες για την κατασκευή δικτυακών τόπων οποιουδήποτε ενδιαφέροντος. Κύριος σκοπός μας όμως ήταν να μεταφέρουμε το υπάρχον σύστημα του peRSSonal στο Joomla, κάτι το οποίο έπρεπε να γίνει με τέτοιο τρόπο ώστε να μην περιορίζει το υπάρχον σύστημά μας, να το βελτιώνει και να προσθέτει όσο το δυνατόν περισσότερες λειτουργίες.

Πραγματικά λοιπόν ανανεώσαμε πλήρως την εγγραφή των χρηστών στο νέο μας σύστημα ώστε να υπάρχει πλήρης έλεγχος κάθε βήματος από το χρήστη. Προσθέσαμε την άμεση μεταγλώττιση της εγγραφής σε οποιοδήποτε βήμα σε μια από τις δύο επιλεγόμενες γλώσσες, αγγλική ή ελληνική. Τέλος όλες οι ρυθμίσεις κατά τη διαδικασία δεν απαιτούν κανένα προηγούμενο βήμα και μπορούν να λειτουργήσουν αυτόνομα.

Ως προς την εμφάνιση του peRSSonal στο Joomla κύριο χαρακτηριστικό βελτίωσης που προστέθηκε είναι η εμφάνιση όλων των modules στην ίδια σελίδα, πράγμα το οποίο δεν γινόταν στο παλαιό σύστημα καθώς παρέπεμπε κάθε φορά σε διαφορετική σελίδα με διαφοροποιημένα modules να εμφανίζονται στο χρήστη κάθε φορά. Το μεγαλύτερο πλεονέκτημα είναι ότι κατά τη διάρκεια προβολής ενός άρθρου μπορεί ο χρήστης άμεσα να μεταβεί σε ένα επόμενο.

Ως προς την ταχύτητα εμφάνισης το σύστημα έχει επιδεχθεί σημαντικές βελτιώσεις καθώς η ελαχιστοποίηση χρήσης της τεχνολογίας AJAX βελτιστοποιεί το χρόνο απόκρισης του συστήματος και συνεπώς της προβολής των άρθρων.

Επιπλέον ο χρήστης έχει τη δυνατότητα να αλλάξει τα προσωπικά του στοιχεία, κωδικούς, όνομα, e-mail και γλώσσα χρήσης του peRSSonal από την αλλαγή του προφίλ του, ενώ οποιαδήποτε αλλαγή στο μενού αυτό επιφέρει τις κατάλληλες αλλαγές και στο παλαιό σύστημα του peRSSonal καθώς οι πίνακες βρίσκονται στην ίδια βάση με αυτή του Joomla. Οι αλλαγές ακόμα και στη γλώσσα εμφάνισης των μενού του peRSSonal και των άρθρων που επιλέγονται εμφανίζονται και αυτά αρμονικά στο χρήστη του Joomla.

Ως προς τη διαχείριση του peRSSonal η δυνατότητα άμεσης μεταφοράς χρηστών από το παλαιό σύστημα του peRSSonal στο νέο αποτελεί τον τελευταίο κρίκο στην αλυσίδα των βελτιώσεων που επιδέχθηκε το σύστημά μας ώστε να φτάσει στη σημερινή του μορφή.

Μην ξεχνάμε τέλος ότι το σύστημά μας μεταφέρθηκε σε μια νέα κοινότητα με διαρκώς αυξανόμενα στοιχεία. Νέα συστατικά στοιχεία και πρόσθετα τα οποία μπορούν να απογειώσουν τις δυνατότητες του ήδη υπάρχοντος συστήματος peRSSonal. Φόρουμ, εμφανίσεις φωτογραφιών και άλμπουμ, πρότυπα, μενού, μπάρες κύλισης με τελευταία νέα και χιλιάδες άλλα προϊόντα που μπορούν να προσαρμοστούν και να επεκτείνουν το υπάρχον σύστημά μας.

## **6.13 Μελλοντική δουλειά**

Οποιοδήποτε σύστημα μπορούμε να αποφανθούμε ότι επιδέχεται νέα χαρακτηριστικά και προσθήκες που βελτιστοποιούν το τρόπο λειτουργίας του. Το νέο peRSSonal που φτιάχτηκε για το Joomla δεν θα μπορούσε να αποφύγει από αυτή τη νεωτεριστική άποψη και θα ήταν μεγάλο λάθος. Θα πρέπει να παρέχει επιπρόσθετα χαρακτηριστικά είτε ως προς το front-end είτε ως προς το back-end περιβάλλον χρήσης.

Ως προς το front-end σίγουρα μπορούμε να δημιουργήσουμε κάποια template που μπορούν να εμφανίζουν καλύτερα τα modules και να είναι σύμφωνα με τις προδιαγραφές που επιθυμούμε για το peRSSonal. Πρότυπα (templates) υπάρχουν αρκετά και εφόσον πρόκειται για κοινότητα ανοιχτού λογισμικού μπορεί να γίνει χρήση και τροποποίηση κάποιων ήδη υπάρχοντων. Έτσι η εμφάνιση του peRSSonal μπορεί να αλλάξει ριζικά με χρήση νέων πρότυπων εμφάνισης.

Τα RSS από μόνα τους αποτελούν μια τεχνολογία με εξαιρετικές δυνατότητες και το peRSSonal σίγουρα αποτελεί εφελκυστήριο για την χρήση τους. Οι δυνατότητες της τεχνολογίας αυτής εξελίσσεται και σίγουρα θα υπάρξουν νέες προσθήκες που θα πρέπει να εφαρμοστούν και στο peRSSonal ώστε να υποστηρίζει πλήρως την τεχνολογία των RSS.

Τέλος η διεπαφή του διαχειριστή μπορεί να δεχθεί μεγάλες αλλαγές ώστε να υποστηρίζει και άλλες λειτουργίες που έχουν να κάνουν με τους χρήστες και τους πίνακες που χρησιμοποιεί η βάση του peRSSonal. Το περιβάλλον του διαχειριστή άλλωστε είναι κατάλληλο για αλλαγές καθώς θέλουμε μεγαλύτερο έλεγχο για τον υπερχρήστη του συστήματός μας.

## APPENDIX

```

<?php
/**
 * @version          $Id: default.php 21543 2011-06-15 22:48:00Z chdemko $
 * @package          Joomla.Site
 * @subpackage       com_users
 * @copyright        Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license          GNU General Public License version 2 or later; see LICENSE.txt
 * @since           1.6
 */

defined('_JEXEC') or die;
define('DS', DIRECTORY_SEPARATOR);

JHtml::_('behavior.keepalive');
JHtml::_('behavior.tooltip');
JHtml::_('behavior.formvalidation');
$document = JFactory::getDocument();
// PERSSONAL documets used and javascript function created
$document-
>addStyleSheet(JURI::root()).'components/com_users/views/registration/tmpl/p7tp/p7tp_03.css');
$document->addStyleSheet(JURI::root()).'components/com_users/views/registration/tmpl/p7tp/spinner.css');
$document->addScript(JURI::root()).'components/com_users/views/registration/tmpl/p7tp/p7tpscripts.js');
$document->addScript(JURI::root()).'components/com_users/views/registration/tmpl/p7tp/getRSSData.js');
$javascript = '
function init() {
    P7_initTP(6,1);
}
function RotateSpinner(spinnerId, up) {
    if(up){
        if(parseInt(document.getElementById(spinnerId).value) < 5)
document.getElementById(spinnerId).value = parseInt(document.getElementById(spinnerId).value) + 1;
    }
    else{
        if(parseInt(document.getElementById(spinnerId).value) > -5)
document.getElementById(spinnerId).value = parseInt(document.getElementById(spinnerId).value) - 1;
    }
}

window.onload = init;';
$document->addScriptDeclaration( $javascript );
$categories = UsersViewRegistration::getData($params);
$language = intval(JRequest::getVar('lang','1'));

if($language == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');

$langHTTP_GR = JURI::current()."?view=registration&lang=16";
$langHTTP_EN = JURI::current()."?view=registration&lang=1";

// Save all variables if they were posted
$arpp_value = intval(JRequest::getVar('arpp','10','post'));
$jform_name_value = JRequest::getVar('jform_name','','post');
$jform_username_value = JRequest::getVar('jform_username','','post');
$jform_email1_value = JRequest::getVar('jform_email1','','post');
$jform_email2_value = JRequest::getVar('jform_email2','','post');

?>

```

```

<div class="registration"<?php echo $this->pageclass_sfx?>">
<?php if ($this->params->get('show_page_heading')) : ?>
    <h1><?php echo $this->escape($this->params->get('page_heading')); ?></h1>
<?php endif; ?>
    <form id="member-registration" action="<?php echo JRoute::_('index.php?
option=com_users&task=registration.register'); ?>" method="post" class="form-validate">
        <input type="hidden" id="getrssdata" name="getrssdata" value="<?php echo
JURI::root()."components/com_users/views/registration/tmpl/p7tp/"; ?>" />
        <input type="hidden" id="getrsslang" name="getrsslang" value="<?php echo $language; ?>" />
        <div id="p7TP1" class="p7TPpanel">
            <div class="p7TPheader">
                <h3><?php echo _PERSSONALREGISTRATION; ?>&nbsp;<?php if($language == 1) { ?><a href="<?php
echo $langHTTP_GR; ?>">" width="20" height="12"
border="0" /></a>&nbsp;<?php } else { ?>&nbsp;<a href="<?php echo $langHTTP_EN; ?>">"
width="20" height="12" border="0" /></a><?php } ?></h3>
            </div>
            <div class="p7TPwrapper">
                <div class="p7TP_tabs">
                    <div id="p7tpb1_1" class="down"><a class="down" href="javascript:;><?php echo
_PERSONALREGISTRATIONBASICSETUP; ?></a></div>
                    <div id="p7tpb1_2"><a href="javascript:;><?php echo
_PERSONALREGISTRATIONADVANCEDSETUP; ?></a></div>
                    <div id="p7tpb1_3"><a href="javascript:;><?php echo _PERSONALREGISTRATIONINFOSETUP; ?
></a></div>
                    <br class="p7TPclear" />
                </div>
                <div class="p7TPcontent">
                    <div id="p7tpc1_1">
                        <table align="right" width="50" height="20" bgcolor="#66CCFF"><tr><td
align="center"><a href="javascript:;" onclick="LoadData('<?php echo
JURI::root()."components/com_users/views/registration/tmpl/p7tp/"; ?>',<?php echo $language; ?
>);P7_TPtrig('p7tpc1_2');"><?php echo _PERSONALREGISTRATIONNEXT; ?></a></td></tr></table>
                        <br>
                        <table>
                            <tr>
                                <td><?php echo _PERSONALREGISTRATIONLANGUAGE; ?
></td>
                                <td>
                                    <select name="lang" id="lang">
                                        <option value="1"><?php echo
_PERSONALREGISTRATIONENGLISH; ?></option>
                                        <option value="16"><?php echo
_PERSONALREGISTRATIONGREEK; ?></option>
                                    </select>
                                </td>
                            </tr>
                        </table>
                        <h4><?php echo _PERSONALREGISTRATIONSTEP1INFO; ?></h4>
                        <p></p>
                        <table>
                            <tr>
                                <td><?php echo
_PERSONALREGISTRATIONARTICLESPERPAGE; ?></td>
                                <td><input type="text" name="arpp" id="arpp" value="<?php
echo $arpp_value; ?>" class="required" size="30"/></td>
                            </tr>
                        </table>
                        <h4><?php echo
_PERSONALREGISTRATIONPREFERENCEPERCATEGORY; ?></h4>
                        <table>
                            <?php
                            for ($i=0;$i< sizeof($categories["categories"]); $i++) {

```

```
?>
<tr>
    <td><?php echo $categories["categories"][$i]["cattitle"]; ?></td>
    <td>
        <input type="hidden" id="category<?php echo $i; ?>"
name="category<?php echo $i; ?>" value="pref_<?php echo $categories["categories"][$i]["catid"]; ?>" />
        <input type="text" name="pref_<?php echo $categories["categories"][$i]["catid"]; ?>"
id="pref_<?php echo $categories["categories"][$i]["catid"]; ?>" value="o" />
        <input type="image" name="upimageField<?php echo $i; ?>"
src="<?php echo JURI::root().'/components/com_users/views/registration/tmpl/p7tp/img' ?>
/uparrow.png" onclick="RotateSpinner('pref_<?php echo $categories["categories"][$i]["catid"]; ?>', true)" />
        <input type="image" name="downimageField<?php echo $i; ?>"
src="<?php echo JURI::root().'/components/com_users/views/registration/tmpl/p7tp/img' ?>
/downarrow.png" onclick="RotateSpinner('pref_<?php echo $categories["categories"][$i]["catid"]; ?>', false)"
/ >
    </td>
</tr>
</table>
<?php
    }
?>
</div>
<div id="p7tpc1_2">
    <table align="right" width="50" height="20" bgcolor="#66CCFF"><tr
align="center"><td>&nbsp;&nbsp;&nbsp;<a href="javascript:;" onclick="P7_TPTrig('p7tpc1_1');"><?php echo
_PERSSONALREGISTRATIONPREVIOUS; ?></a>&nbsp;&nbsp;&nbsp;</td><td align="center">&nbsp;&nbsp;&nbsp;<a
href="javascript:;" onclick="P7_TPTrig('p7tpc1_3');"><?php echo _PERSSONALREGISTRATIONNEXT; ?
></a>&nbsp;&nbsp;&nbsp;</td></tr></table>
    <br>
    <?php echo _PERSSONALREGISTRATIONRSSINFO; ?>
    <div id="txtHint">
        <input type="hidden" id="posted_items" name="posted_items"
value="o" />
    </div>
    <p><input name="onlymine" type="checkbox" id="onlymine"
value="on" /> <label for="onlymine"><?php echo _PERSSONALREGISTRATIONMYRSSONLY; ?
></label></p>
</div>
<div id="p7tpc1_3">
    <table align="right" width="50" height="20" bgcolor="#66CCFF"><tr
align="center"><td><a href="javascript:;" onclick="LoadData('<?php echo
JURI::root()."/components/com_users/views/registration/tmpl/p7tp/"; ?>', <?php echo $language; ?
>);P7_TPTrig('p7tpc1_2');"><?php echo _PERSSONALREGISTRATIONPREVIOUS; ?
></a></td></tr></table>
    <br>
    <fieldset>
        <dt><label id="jform_name-lbl" for="jform_name" class="hasTip
required" title="Name::Enter your full name"><?php echo _PERSSONALREGISTRATIONNAME; ?><span
class="star">&#160;*</span></label></dt>
        <dd><input type="text" name="jform[name]" id="jform_name" value="<?
php echo $jform_name_value; ?>" class="required" size="30"/></dd>
        <dt><label id="jform_username-lbl" for="jform_username" class="hasTip
required" title="Username::Enter your desired user name"><?php echo
_PERSSONALREGISTRATIONUSERNAME; ?><span class="star">&#160;*</span></label></dt>
        <dd><input type="text" name="jform[username]" id="jform_username"
value="<?php echo $jform_username_value; ?>" class="validate-username required" size="30"/></dd>
        <dt><label id="jform_password1-lbl" for="jform_password1"
class="hasTip required" title="Password::Enter your desired password - Enter a minimum of 4 characters"><?
php echo _PERSSONALREGISTRATIONPASSWORD; ?><span class="star">&#160;*</span></label></dt>
        <dd><input type="password" name="jform[password]"
id="jform_password1" value="" autocomplete="off" class="validate-password required" size="30"/></dd>
    </fieldset>
</div>
```

```

        <dt><label      id="jform_password2-lbl"      for="jform_password2"
class="hasTip  required"  title="Confirm Password::Confirm your password"><?php  echo
_PERSSONALREGISTRATIONPASSWORDRETYPE; ?><span class="star">&#160;*</span></label></dt>
        <dd><input      type="password"      name="jform[password2]"
id="jform_password2" value="" autocomplete="off" class="validate-password required" size="30"/></dd>

        <dt><label  id="jform_email1-lbl"  for="jform_email1"  class="hasTip
required" title="Email Address::Enter your email address"><?php echo _PERSSONALREGISTRATIONEMAIL;
?><span class="star">&#160;*</span></label></dt>
        <dd><input  type="text"  name="jform[email1]"  class="validate-email
required" id="jform_email1" value="<?php echo $jform_email1_value; ?>" size="30"/></dd>

        <dt><label  id="jform_email2-lbl"  for="jform_email2"  class="hasTip
required"  title="Confirm  email  Address::Confirm  your  email  address"><?php  echo
_PERSSONALREGISTRATIONEMAILRETYPE; ?><span class="star">&#160;*</span></label></dt>
        <dd><input  type="text"  name="jform[email2]"  class="validate-email
required" id="jform_email2" value="<?php echo $jform_email2_value; ?>" size="30"/></dd>

        <dt><span      class="spacer"><span      class="before"></span><span
class="text"><label  id="jform_spacer-lbl"  class=""><strong  class="red">*</strong>  <?php  echo
_PERSSONALREGISTRATIONREQUIREDFIELDS;  ?></label></span><span
class="after"></span></span></dt>
        <dd> </dd>
    </fieldset>
    <button      type="submit"      class="validate"><?php      echo
_PERSSONALREGISTRATIONREGISTERBUTTON; ?></button>
    <?php echo JText::_("COM_USERS_OR");?>
    <a      href="<?php      echo      JRoute::_("");?>"      title="<?php      echo
_PERSSONALREGISTRATIONCANCELBUTTON;      ?>"><?php      echo
_PERSSONALREGISTRATIONCANCELBUTTON; ?></a>
    <div>
        <input type="hidden" name="option" value="com_users" />
        <input type="hidden" name="task" value="registration.register" />
        <input type="hidden" id="total_categories" name="total_categories"
value="<?php echo sizeof($categories["categories"]); ?>" />
        <?php echo JHtml::_('form.token');?>
    </div>
</div>
</div>
</div>
</div>
</div>
</form>
</div>

```

**Πίνακας 1. Το default.php αρχείου του front-end com\_users για το registration του χρήστη ώστε να υποστηρίξει το perSSonal**

```

<?php
// Load the Joomla framework
define( '_JEXEC', 1 );
define( 'DS', DIRECTORY_SEPARATOR );
define( 'JPATH_BASE', realpath(dirname(__FILE__).DS.'..'.DS.'..'.DS.'..'.DS.'..'.DS.'..'.DS.'..') );

require_once ( JPATH_BASE .DS.'includes'.DS.'defines.php' );
require_once ( JPATH_BASE .DS.'includes'.DS.'framework.php' );

$mainframe =& JFactory::getApplication('site');
$mainframe->initialise();

```

```

$session =& JFactory::getSession();
$user = $session->get( 'user' );

$language = intval($_GET['lang']);
if($language == 1) require_once(dirname(__FILE__).DS.'..'.DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'..'.DS.'lang'.DS.'el-GR.php');
?>
<?php
/**
 * @version          $Id: default.php 21543 2011-06-15 22:48:00Z chdemko $
 * @package          Joomla.Site
 * @subpackage       com_users
 * @copyright         Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license          GNU General Public License version 2 or later; see LICENSE.txt
 * @since            1.6
 */

$query = "SELECT cat_id, cat_title FROM category WHERE cat_sub=0 AND sports=0 AND trash=0";
$db = & JFactory::getDBO();
$db->setQuery($query);
$rows = $db->loadObjectList();

// Get list items
if ($rows!=null)
{
    $i = 0;
    $output = "<table>";

    foreach ($rows as $row)
    {
        $cat_id = $row->cat_id;
        $cat_title = stripslashes($row->cat_title);

        $cat_name = "pref_".$row->cat_id;
        $u_answer = intval(JRequest::getVar($cat_name,'0'));
        if($u_answer > 0) {
            $output .= "<tr>";
            $output .= "<td><strong>".$row->cat_title."</strong></td><td></td>";
            $output .= "</tr>";
            $output .= "<tr>";
            $output .= "<td>RSS Feeds</td><td><textarea name=\"myrss_". $row->cat_id . "\" id=\"myrss_". $row->cat_id . "\"></textarea><input type=\"hidden\" id=\"rsscategory_". $row->cat_id . "\" name=\"rsscategory_". $row->cat_id . "\" value=\"\" . $row->cat_id . "\" /></td>";
            $output .= "</tr>";
            $output .= "<tr>";
            $output .= "<td>_PERSONALREGISTRATIONKEYWORDS .</td><td>";
            $output .= " = " . " <td>"

            $myquery = "SELECT a.kw_id, a.kw_name, b.rel_frequency FROM
extraction_kw a LEFT JOIN keywords_category_training b ON a.kw_id=b.kw_id WHERE
b.cat_id=$row->cat_id ORDER BY b.rel_frequency DESC";

            $mydb =& JFactory::getDBO();
            $mydb->setQuery($myquery);
            $myrows = $mydb->loadObjectList();
            $i++;

            // Get list
            if ($myrows!=null)
            {
                $output .= "<select name=\"keywords_". $row->cat_id . "\"
id=\"keywords_". $row->cat_id . "\">";
                foreach ($myrows as $myrow){

```

```

        $kw_id = $myrow->kw_id;
        $kw_name = $myrow->kw_name;
        $rel_fr = $myrow->rel_frequency;
        if($kw_name == "unwatch") continue;
        $output .= "<option value=\"\" . $kw_name .\"-\" . $kw_id.
        \"$\">\" . $kw_name .\"-\" . $kw_id.\" (\" . substr($rel_fr*1000,0,5) .\")</option>\";
    }
    $output .= "</select>\";
}
\"</td></tr>\";

    $output .= \"<tr><td></td><td><table width=\"50\" height=\"20\"
bgcolor=\"\#66CCFF\"><tr><td align='center'><a href=\"javascript;\"
onclick=\"document.getElementById('kw_\" . $row->cat_id .\"').value
+document.getElementById('keywords\" . $row->cat_id .\"').options[document.getElementById('keywords\"
.$row->cat_id .\"').selectedIndex].value + ' \">\"
._PERSONALREGISTRATIONADDKEYWORD .\"</a></td></tr></table></td></tr>\";
    $output .= \"<tr><td></td><td><textarea name=\"kw_\" . $row->cat_id . \"\"
id=\"kw_\" . $row->cat_id . \"\"></textarea></td></tr>\";
    $output .= \"<tr><td></td><td>
<a href=\"javascript;\" onclick=\"document.getElementById('kw_\" . $row-
>cat_id .\"').value=;\">\" ._PERSONALREGISTRATIONCLEARKEYWORDS .\"</a></td></tr>\";

    }
}

    $output .= \"<input type=\"hidden\" id=\"total_rsscategories\" name=\"total_rsscategories\"
value=\"\" . $i . \"\" />\";
    $output .= \"</table>\";
}

$output .= \"<input type=\"hidden\" id=\"posted_items\" name=\"posted_items\" value=\"1\" />\";

echo $output;
?>

```

## Πίνακας 2. Το αρχείο getRSSData.php

```

// JavaScript Document
function LoadData(scriptpath,language)
{
    if (window.XMLHttpRequest)
    { // code for IE7+, Firefox, Chrome, Opera, Safari
        xmlhttp=new XMLHttpRequest();
    }
    else
    { // code for IE6, IE5
        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }

    xmlhttp.onreadystatechange=function()
    {
        if (xmlhttp.readyState==4 && xmlhttp.status==200)
        {
            document.getElementById("txtHint").innerHTML=xmlhttp.responseText;
        }
        else {
            var spanElement = document.getElementById("txtHint");
            spanElement.innerHTML = "";
            try {
                spanElement.innerHTML = '<p align=\"center\"><img src=\"' + scriptpath

```

```

+ 'img/loading.gif"></p>';
        spanElement.innerHTML += '<p align=\"center\">Loading</p>';
    }
    catch (e) {
        // IE fails unless we wrap the string in another element.
        var wrappingElement = document.createElement('div');
        wrappingElement.innerHTML = '<p align=\"center\"><img src=\"' +
scriptpath + 'img/loading.gif\"></p>';
        wrappingElement.innerHTML += '<p align=\"center\">Loading</p>';
        spanElement.appendChild(wrappingElement);
    }

}

}

var params = 'getRSSData.php?lang=' + language;

for(var i=0;i<parseInt(document.getElementById("total_categories").value);i++){
    var myval = document.getElementById('category' + i).value;
    params += '&' + myval + '=' +
encodeURIComponent(document.getElementById(myval).value);

}

xmlhttp.open("GET",scriptpath + params,true);
xmlhttp.send();
}

```

**Πίνακας 3. Το αρχείο getRSSData.js**

```

<?php
/**
 * @version      $Id: registration.php 22355 2011-11-07 05:11:58Z github_bot $
 * @package      Joomla.Site
 * @subpackage   com_users
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

defined('_JEXEC') or die;

jimport('joomla.application.component.modelform');
jimport('joomla.event.dispatcher');
jimport('joomla.plugin.helper');

/**
 * Registration model class for Users.
 */
* @package      Joomla.Site
* @subpackage   com_users
* @since        1.6
*/
class UsersModelRegistration extends JModelForm
{
    /**
     * @var      object  The user registration data.
     * @since    1.6
     */
    protected $data;
    protected $mypass;

    /**

```

```

* Method to activate a user account.
*
* @param      string          The activation token.
* @return     mixed           False on failure, user object on success.
* @since      1.6
*/
public function activate($token)
{
    $config = JFactory::getConfig();
    $userParams = JComponentHelper::getParams('com_users');
    $db = $this->getDbo();

    // Get the user id based on the token.
    $db->setQuery(
        'SELECT '.$db->nameQuote('id').' FROM '.$db->nameQuote('#__users') .
        ' WHERE '.$db->nameQuote('activation').' = '.$db->Quote($token) .
        ' AND '.$db->nameQuote('block').' = 1' .
        ' AND '.$db->nameQuote('lastvisitDate').' = '.$db->Quote($db->getNullDate())
    );
    $userId = (int) $db->loadResult();

    // Check for a valid user id.
    if (!$userId) {
        $this->setError(JText::_('COM_USERS_ACTIVATION_TOKEN_NOT_FOUND'));
        return false;
    }

    // Load the users plugin group.
    JPluginHelper::importPlugin('user');

    // Activate the user.
    $user = JFactory::getUser($userId);

    // Admin activation is on and user is verifying their email
    if (($userParams->get('useractivation') == 2) && !$user->getParam('activate', 0))
    {
        $uri = JURI::getInstance();
        jimport('joomla.user.helper');

        // Compile the admin notification mail values.
        $data = $user->getProperties();
        $data['activation'] = JUtility::getHash(JUserHelper::genRandomPassword());
        $user->set('activation', $data['activation']);
        $data['siteurl'] = JUri::base();
        $base = $uri->toString(array('scheme', 'user', 'pass', 'host', 'port'));
        $data['activate'] = $base.JRoute::_('index.php?option=com_users&task=registration.activate&token='.$data['activation'], false);
        $data['fromname'] = $config->get('fromname');
        $data['mailfrom'] = $config->get('mailfrom');
        $data['sitename'] = $config->get('sitename');
        $user->setParam('activate', 1);
        $emailSubject = JText::sprintf(
            'COM_USERS_EMAIL_ACTIVATE_WITH_ADMIN_ACTIVATION_SUBJECT',
            $data['name'],
            $data['sitename']
        );

        $emailBody = JText::sprintf(
            'COM_USERS_EMAIL_ACTIVATE_WITH_ADMIN_ACTIVATION_BODY',
            $data['sitename'],
            $data['name'],
            $data['email'],
            $data['username'],
            $data['siteurl'].'index.php?option=com_users&task=registration.activate&token='.

```

```

$data['activation']
    );

    // get all admin users
    $query = 'SELECT name, email, sendEmail' .
            ' FROM #__users' .
            ' WHERE sendEmail=1';

    $db->setQuery( $query );
    $rows = $db->loadObjectList();

    // Send mail to all superadministrators id
    foreach( $rows as $row )
    {
        $return = JUtility::sendMail($data['mailfrom'], $data['fromname'], $row->email,
$emailSubject, $emailBody);

        // Check for an error.
        if ($return !== true) {
            $this-
>setError(JText::_('COM_USERS_REGISTRATION_ACTIVATION_NOTIFY_SEND_MAIL_FAILED'));
            return false;
        }
    }
}

//Admin activation is on and admin is activating the account
elseif (($userParams->get('useractivation') == 2) && $user->getParam('activate', 0))
{
    $user->set('activation', '');
    $user->set('block', '0');

    $uri = JURI::getInstance();
    jimport('joomla.user.helper');

    // Compile the user activated notification mail values.
    $data = $user->getProperties();
    $user->setParam('activate', 0);
    $data['fromname'] = $config->get('fromname');
    $data['mailfrom'] = $config->get('mailfrom');
    $data['sitename'] = $config->get('sitename');
    $data['siteurl'] = JUri::base();
    $emailSubject = JText::sprintf(
        'COM_USERS_EMAIL_ACTIVATED_BY_ADMIN_ACTIVATION_SUBJECT',
        $data['name'],
        $data['sitename']
    );

    $emailBody = JText::sprintf(
        'COM_USERS_EMAIL_ACTIVATED_BY_ADMIN_ACTIVATION_BODY',
        $data['name'],
        $data['siteurl'],
        $data['username']
    );

    $return = JUtility::sendMail($data['mailfrom'], $data['fromname'], $data['email'],
$emailSubject, $emailBody);

    // Check for an error.
    if ($return !== true) {
        $this-
>setError(JText::_('COM_USERS_REGISTRATION_ACTIVATION_NOTIFY_SEND_MAIL_FAILED'));
        return false;
    }
}

```

```

    }
    else
    {
        $user->set('activation', '');
        $user->set('block', 'o');
    }

    // Store the user object.
    if (!$user->save()) {
        $this-
>setError(JText::sprintf('COM_USERS_REGISTRATION_ACTIVATION_SAVE_FAILED', $user->getError()));
        return false;
    }
    else{
        // Activate user also for old PERSSONAL system
        $db =& JFactory::getDBO();
        $query = "UPDATE user_website SET active=0 WHERE email='".$user->email.'"";
        $db->setQuery($query);
        $db->query();
    }

    return $user;
}

/**
 * Method to get the registration form data.
 *
 * The base form data is loaded and then an event is fired
 * for users plugins to extend the data.
 *
 * @return mixed Data object on success, false on failure.
 * @since 1.6
 */
public function getData()
{
    $this->data = new stdClass();
    $app = JFactory::getApplication();
    $params = JComponentHelper::getParams('com_users');

    // Override the base user data with any data in the session.
    $temp = (array)$app->getUserState('com_users.registration.data', array());
    foreach ($temp as $k => $v) {
        $this->data->{$k} = $v;
    }

    // Get the groups the user should be added to after registration.
    $this->data->groups = isset($this->data->groups) ? array_unique($this->data->groups) :
array();

    // Get the default new user group, Registered if not specified.
    $system = $params->get('new_usertype', 2);

    $this->data->groups[] = $system;

    // Save original password for PERSSONAL system
    $this->mypass = $this->data->password1;

    // Unset the passwords.
    unset($this->data->password1);
    unset($this->data->password2);

    // Get the dispatcher and load the users plugins.
    $dispatcher = JDispatcher::getInstance();

```

```

        JPluginHelper::importPlugin('user');

        // Trigger the data preparation event.
        $results = $dispatcher->trigger('onContentPrepareData', array('com_users.registration', $this-
>data));

        // Check for errors encountered while preparing the data.
        if (count($results) && in_array(false, $results, true)) {
            $this->setError($dispatcher->getError());
            $this->data = false;
        }

        return $this->data;
    }

    /**
     * Method to get the registration form.
     *
     * The base form is loaded from XML and then an event is fired
     * for users plugins to extend the form with extra fields.
     *
     * @param array $data An optional array of data for the form to interrogate.
     * @param boolean $loadData True if the form is to load its own data (default case), false if not.
     * @return JForm A JForm object on success, false on failure
     * @since 1.6
     */
    public function getForm($data = array(), $loadData = true)
    {
        // Get the form.
        $form = $this->loadForm('com_users.registration', 'registration', array('control' => 'jform', 'load_data'
=> $loadData));
        if (empty($form)){
            return false;
        }

        return $form;
    }

    /**
     * Method to get the data that should be injected in the form.
     *
     * @return mixed The data for the form.
     * @since 1.6
     */
    protected function loadFormData()
    {
        return $this->getData();
    }

    /**
     * Override preprocessForm to load the user plugin group instead of content.
     *
     * @param object A form object.
     * @param mixed The data expected for the form.
     * @throws Exception if there is an error in the form event.
     * @since 1.6
     */
    protected function preprocessForm(JForm $form, $data, $group = 'user')
    {
        $userParams = JComponentHelper::getParams('com_users');

        //Add the choice for site language at registration time
        if ($userParams->get('site_language') == 1 && $userParams->get('frontend_userparams') == 1)
        {

```

```

        $form->loadFile('sitelang',false);
    }
    parent::preprocessForm($form, $data, $group);
}

/**
 * Method to auto-populate the model state.
 *
 * Note. Calling getState in this method will result in recursion.
 *
 * @since      1.6
 */
protected function populateState()
{
    // Get the application object.
    $app    = JFactory::getApplication();
    $params = $app->getParams('com_users');

    // Load the parameters.
    $this->setState('params', $params);
}

/**
 * Method to save the form data.
 *
 * @param      array          The form data.
 * @return     mixed          The user id on success, false on failure.
 * @since      1.6
 */
public function register($temp)
{
    $config = JFactory::getConfig();
    $params = JComponentHelper::getParams('com_users');

    // Initialise the table with JUser.
    $user = new JUser;
    $data = (array)$this->getData();

    // Merge in the registration data.
    foreach ($temp as $k => $v) {
        $data[$k] = $v;
    }

    // Prepare the data for the user object.
    $data['email']          = $data['email1'];
    $data['password'] = $data['password1'];
    $useractivation = $params->get('useractivation');

    // Save PERSSONAL password if not saved before
    if($this->mypass != $data['password']) $this->mypass = $data['password'];

    // Check if the user needs to activate their account.
    if (($useractivation == 1) || ($useractivation == 2)) {
        jimport('joomla.user.helper');
        $data['activation'] = JUtility::getHash(JUserHelper::genRandomPassword());
        $data['block'] = 1;
    }

    // Bind the data.
    if (!$user->bind($data)) {
        $this->setError(JText::sprintf('COM_USERS_REGISTRATION_BIND_FAILED', $user-
>getError()));
        return false;
    }
}

```

```

// Load the users plugin group.
JPluginHelper::importPlugin('user');

// Store the data.
if (!$user->save()) {
    $this->setError(JText::sprintf('COM_USERS_REGISTRATION_SAVE_FAILED', $user-
>getError()));
    return false;
}
else{
    // PERSSONAL SYSTEM DATA
    $perssonal = array();
    $perssonal['arpp'] = JRequest::getVar('arpp', '-1', 'post');
    $password_given = JRequest::getVar('jform_password1', '-1', 'post');
    $perssonal['total_categories'] = JRequest::getVar('total_categories', '-1', 'post');
    if($perssonal['total_categories'] != "-1"){
        for($i=0;$i<$perssonal['total_categories'];$i++){
            $pref = JRequest::getVar('category' . $i, '-1', 'post');
            $perssonal[$pref] = JRequest::getVar($pref, '-1', 'post');
        }
    }

    $perssonal['posted_items'] = JRequest::getVar('posted_items', '0', 'post');
    if($perssonal['posted_items']){
        $perssonal['total_rsscategories'] = JRequest::getVar('total_rsscategories', '-1', 'post');

        for($i=0;$i<$perssonal['total_rsscategories'];$i++){
            $rss_catid = JRequest::getVar('rsscategory' . $i, '-1', 'post');
            $perssonal['myrss_' . $rss_catid] = JRequest::getVar('myrss_' . $rss_catid, '-
1', 'post');
            $perssonal['kw_' . $rss_catid] = JRequest::getVar('kw_' . $rss_catid, '-1',
'post');
        }
    }

    if($perssonal['posted_items']){
        $user_email = $data['email'];

        if($user_email != "-1"){
            $db =& JFactory::getDBO();
            $nowdate = date("Y-m-d H:i:s");
            $language = JRequest::getVar('lang', '1', 'post'); // 1 is for English, 16 is for
Greek

            // Check the onlymine variable
            $onlyrss = JRequest::getVar('onlymine', '-1', 'post');
            if($onlyrss == "on") {
                $onlyrss = 1;
            }
            else {
                $onlyrss = 0;
            }

            $db =& JFactory::getDBO();
            $mypassword_md5 = MD5($this->mypass);
            $query = "INSERT INTO user_website VALUES ('", $user_email, "', '",
$mypassword_md5, "', '1024', '768', '0', $nowdate, '0', $perssonal['arpp'], '0', $onlyrss, '0', $language, '0', '2',
'.MD5($data['email']).')";

            $db->setQuery($query);
            $result = $db->query();

            //Debug of form output faults
            //$this->setError("Original password:" . $this->mypass . " Password read: " .
$password_given . ", query: " . $query);

```

```

//return false;

$user_website_id = $db->insertid();

$db =& JFactory::getDBO();
$query = "INSERT INTO test VALUES (,".$user_website_id.")";
$db->setQuery($query);
$result = $db->query();

// Database query
$all_rss = array();
$all_id_rss = array();
$count_rss = 0;

$db =& JFactory::getDBO();
$query = "SELECT id, rss FROM rss";
$db =& JFactory::getDBO();
$db->setQuery($query);
$rows = $db->loadObjectList();

// Get list items
if ($rows!=null)
{
    foreach ($rows as $row)
    {
        $all_rss[$count_rss] = $row->rss;
        $all_id_rss[$count_rss++] = $row->id;
    }
}

$db =& JFactory::getDBO();
$query = "SELECT cat_id, cat_title FROM category WHERE sports=0 and
trash=0";

$db->setQuery($query);
$rows = $db->loadObjectList();

// Get list items
if ($rows!=null)
{
    foreach ($rows as $row)
    {
        $cid = $row->cat_id;
        $cname = $row->cat_title;
        /* Edw vriskw poso prepei na prosthesw se ena kw. */
        /* $query_num_kw = "SELECT kw_id FROM
keywords_category_training WHERE cat_id=$cid";

$result_num_kw = mysql_query($query_num_kw);
$category_has = mysql_num_rows($result_num_kw); */
$weight = 1;
$proximo = 1;
$sweight = 1;
$limit=1;
$cat_name = "pref_".$cid;
$myrss_name = "myrss_".$cid;
$mykw_name = "kw_".$cid;

$pref_query = "INSERT INTO user_website_category
VALUES (". $user_website_id .",".$cid .",".$perssonal[$cat_name].")";
$prefdb =& JFactory::getDBO();
$prefdb->setQuery($pref_query);
$prefdb->query();

if($perssonal[$cat_name] < 0) {
    $weight = abs($perssonal[$cat_name]);

```

```

    }
    else {
        $weight = 1.3 * abs($perssonal[$cat_name]);
    }
    $sweight = 5 * $perssonal[$cat_name] *

$perssonal[$cat_name];

    if($weight == 0) {
        $weight = 1;
    }

    if($perssonal[$cat_name] < 0) {
        $weight = abs($perssonal[$cat_name]);
        $limit = intval(5 * abs($weight));
    }
    else if($perssonal[$cat_name] == 0) {
        $weight = 1;
        $limit = 8;
    }
    else {
        $weight = 1.3 * abs($perssonal[$cat_name]);
        $limit = intval(15 * abs($weight));
    }

    // $limit = intval(15 * abs($weight));
    // if($limit == 0) {
    //     $limit = 0;
    // }

    $proximo_finder = $perssonal[$cat_name];
    if($proximo_finder >= 0) {
        $proximo = 1;
    }
    else if($proximo_finder < 0) {
        $proximo = -1;
    }

    // Read kw if it exists for current category
    if(array_key_exists($mykw_name,$perssonal)    &&

array_key_exists($myrss_name,$perssonal)){

        $kw_array = $perssonal[$mykw_name];
        $kw_array = explode(" ", $kw_array);

        $keywords = array();
        for($i=0; $i < sizeof($kw_array); ++$i) {
            $temp = explode("-", $kw_array[$i]);
            $keywords[$i] = $temp[1];
        }

        $rss_array = "";
        $myrss_fetch = =

        $myrss_fetch = trim($myrss_fetch);
        $myrsss = explode("\n", $myrss_fetch);

        for($k=0; $k<sizeof($myrsss); $k++) {
            if($myrsss[$k] == "" || $myrsss[$k] ==

            null || $myrsss[$k] == "")

                continue;
        }
    }
}

```

```

if(!in_array($myrsss[$k], $all_rss)) {
    $rss = $myrsss[$k];
    $rss = trim($rss);
    $nameof = "UDR -" .

$user_website_id;

    $query_rss = "INSERT INTO
rss VALUES ('" . $nameof . "','" . $rss . "','" . $nowdate . "','" . '1' . "','" . $cid . "','" . 'o','1','2','2','o','" . "','" . 'o')";
    $db->setQuery($query_rss);
    $result = $db->query();
    if($result){
        $rss_id = $db-
>insertid();
        $query = "INSERT
INTO rss_user VALUES ('" . $rss_id . "','" . $user_website_id . "')";
        $db-
>setQuery($query);
        $result = $db-
>query();
    }
    else {
        $rss_id =
        if($rss_id != " && $rss_id !=
        $query = "INSERT
        $db-
        $db->query();
    }
}

}

if(sizeof($myrsss) > 0) {
    $to_us = 'poulop@cti.gr;
    $subject_us = '[peRSSonal] - RSS added
    $message_us = 'A user has added RSS
    $headers_us = 'From:
        'Reply-To: perssonal@cti.gr' .
        'X-Mailer: PHP/' .
    mail($to_us, $subject_us, $message_us,
}

if($limit == 0)
    $limit=10;

for($z=0; $z<sizeof($keywords); $z++) {
    $k_id = $keywords[$z];
    if($k_id == "" || $k_id == null)
        continue;

```

```

FROM keywords_category_training WHERE kw_id=".$k_id;

$query = "SELECT kw_id, rel_frequency
$db =& JFactory::getDBO();
$db->setQuery($query);
$kwrows = $db->loadObjectList();

if ($kwrows!=null)
{
    foreach ($kwrows as $haskw)
    {
        $thekw = $haskw-
        $kwfr = $haskw-
        $kweight = $proximo

>kw_id;
>rel_frequency;
* $sweight * $kwfr;

$query_already_in =
"SELECT user_id, kw_id, rel_frequency FROM user_website_keyword WHERE user_id=".$user_website_id ." AND
kw_id=".$thekw;
$db =&
JFactory::getDBO();
$db-
>setQuery($query_already_in);
$result_already_in =
$db->loadObjectList();
$already_in = $db-

$user_id =0;
$kw_id =0;
$rel_frequency =0;

foreach($result_already_in as $already_in_rows) {
    $user_id =
$already_in_rows->user_id;
    $kw_id =
$already_in_rows->kw_id;
    $rel_frequency = $already_in_rows->rel_frequencny;
}
if($already_in != 0) {

$rel_frequency = $rel_frequency + $kweight;
$query_upd
= "UPDATE user_website_keyword SET rel_frequency=".$rel_frequency .", user_defined=1 WHERE user_id=" .
$user_website_id ." AND kw_id=".$thekw;
$db =&
JFactory::getDBO();
$db-
>setQuery($query_upd);

//echo
$db-
$query_upd."<br>";
}
else {
$query_ins =
>query();
"INSERT INTO user_website_keyword VALUES (".$user_website_id .",".$thekw .",".$kweight .",".1)";
$db =&

```

```

JFactory::getDBO();
>setQuery($query_ins);
$query_upd."<br>";
>query();
}
}
} // End of for update tables for all keywords
} // End of if check that the current category rss and
keywords were sended from the user
} // End of for query for each category
}
else{
$this->setError(JText::sprintf('You have not set any e-mail', $user-
>getError()));
return false;
}
else{
$this->setError(JText::sprintf('You have not selected any advanced setup data! Please
choose advance setup data and then register', $user->getError()));
return false;
}
// END OF PERSSONAL DATA
} // End of data saved successfully to database

// Compile the notification mail values.
$data = $user->getProperties();
$data['fromname'] = $config->get('fromname');
$data['mailfrom'] = $config->get('mailfrom');
$data['sitename'] = $config->get('sitename');
$data['siteurl'] = JUri::base();

// Handle account activation/confirmation emails.
if ($useractivation == 2)
{
// Set the link to confirm the user email.
$url = JUri::getInstance();
$base = $url->toString(array('scheme', 'user', 'pass', 'host', 'port'));
$data['activate'] = $base.JRoute::_('index.php?
option=com_users&task=registration.activate&token='.$data['activation'], false);

$emailSubject = JText::sprintf(
'COM_USERS_EMAIL_ACCOUNT_DETAILS',
$data['name'],
$data['sitename']
);

$emailBody = JText::sprintf(
'COM_USERS_EMAIL_REGISTERED_WITH_ADMIN_ACTIVATION_BODY',
$data['name'],
$data['sitename'],
$data['siteurl'].'index.php?option=com_users&task=registration.activate&token='.

```

```

$data['activation'],
                $data['siteurl'],
                $data['username'],
                $data['password_clear']
            );
        }
        elseif ($useractivation == 1)
        {
            // Set the link to activate the user account.
            $uri = JURI::getInstance();
            $base = $uri->toString(array('scheme', 'user', 'pass', 'host', 'port'));
            $data['activate'] = $base.JRoute::_('index.php?
option=com_users&task=registration.activate&token='.$data['activation'], false);

            $emailSubject = JText::sprintf(
                'COM_USERS_EMAIL_ACCOUNT_DETAILS',
                $data['name'],
                $data['sitename']
            );

            $emailBody = JText::sprintf(
                'COM_USERS_EMAIL_REGISTERED_WITH_ACTIVATION_BODY',
                $data['name'],
                $data['sitename'],
                $data['siteurl'].'index.php?option=com_users&task=registration.activate&token='.
$data['activation'],
                $data['siteurl'],
                $data['username'],
                $data['password_clear']
            );
        } else {

            $emailSubject = JText::sprintf(
                'COM_USERS_EMAIL_ACCOUNT_DETAILS',
                $data['name'],
                $data['sitename']
            );

            $emailBody = JText::sprintf(
                'COM_USERS_EMAIL_REGISTERED_BODY',
                $data['name'],
                $data['sitename'],
                $data['siteurl']
            );
        }

        // Send the registration email.
        $return = JUtility::sendMail($data['mailfrom'], $data['fromname'], $data['email'], $emailSubject,
$emailBody);

        // Check for an error.
        if ($return !== true) {
            $this->setError(JText::_('COM_USERS_REGISTRATION_SEND_MAIL_FAILED'));

            // Send a system message to administrators receiving system mails
            $db = JFactory::getDBO();
            $q = "SELECT id
                FROM #__users
                WHERE block = 0
                AND sendEmail = 1";
            $db->setQuery($q);
            $sendEmail = $db->loadResultArray();
            if (count($sendEmail) > 0) {
                $jdate = new JDate();

```

```

// Build the query to add the messages
>nameQuote('user_id_from'). $q = "INSERT INTO ".$db->nameQuote('#__messages')." (".$db-
"; ".$db->nameQuote('user_id_to').", ".$db->nameQuote('date_time').
"; ".$db->nameQuote('subject').", ".$db->nameQuote('message')." ) VALUES ";
$mESSAGES = array();
foreach ($sendEmail as $userid) {
    $messages[] = "(".$userid.", ".$userid.", ".$db->toSQLDate($jdate).",
"".JText::_('COM_USERS_MAIL_SEND_FAILURE_SUBJECT')."",
"".JText::sprintf('COM_USERS_MAIL_SEND_FAILURE_BODY', $return, $data['username'])."");
}
$q .= implode(',', $messages);
$db->setQuery($q);
$db->query();
}
return false;
}

if ($useractivation == 1)
    return "useractivate";
elseif ($useractivation == 2)
    return "adminactivate";
else
    return $user->id;
}
}

```

**Πίνακας 4. Το αρχείο registration.php του μοντέλου του com\_users**

```

<?php
/**
 * @version      $Id: profile.php 21798 2011-07-10 05:52:40Z infograf768 $
 * @package      Joomla.Site
 * @subpackage   com_users
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

defined('_JEXEC') or die;

jimport('joomla.application.component.modelform');
jimport('joomla.event.dispatcher');
jimport('joomla.plugin.helper');

/**
 * Profile model class for Users.
 */
* @package      Joomla.Site
* @subpackage   com_users
* @since        1.6
*/
class UsersModelProfile extends JModelForm
{
    /**
     * @var      object    The user profile data.
     * @since    1.6
     */
    protected $data;
    protected $mypass;
    protected $myemail;
}

```

```

/**
 * Method to check in a user.
 *
 * @param integer The id of the row to check out.
 * @return boolean True on success, false on failure.
 * @since 1.6
 */
public function checkin($userId = null)
{
    // Get the user id.
    $userId = (!empty($userId)) ? $userId : (int)$this->getState('user.id');

    if ($userId) {
        // Initialise the table with JUser.
        $table = JTable::getInstance('User');

        // Attempt to check the row in.
        if (!$table->checkin($userId)) {
            $this->setError($table->getError());
            return false;
        }
    }

    return true;
}

/**
 * Method to check out a user for editing.
 *
 * @param integer The id of the row to check out.
 * @return boolean True on success, false on failure.
 * @since 1.6
 */
public function checkout($userId = null)
{
    // Get the user id.
    $userId = (!empty($userId)) ? $userId : (int)$this->getState('user.id');

    if ($userId) {
        // Initialise the table with JUser.
        $table = JTable::getInstance('User');

        // Get the current user object.
        $user = JFactory::getUser();

        // Attempt to check the row out.
        if (!$table->checkout($user->get('id'), $userId)) {
            $this->setError($table->getError());
            return false;
        }
    }

    return true;
}

/**
 * Method to get the profile form data.
 *
 * The base form data is loaded and then an event is fired
 * for users plugins to extend the data.
 *
 * @return mixed Data object on success, false on failure.
 * @since 1.6
 */

```

```

public function getData()
{
    if ($this->data === null) {

        $userId = $this->getState('user.id');

        // Initialise the table with JUser.
        $this->data = new JUser($userId);

        // Set the base user data.
        $this->data->email1 = $this->data->get('email');
        $this->data->email2 = $this->data->get('email');

        // Override the base user data with any data in the session.
        $temp = (array)JFactory::getApplication()-
>getUserState('com_users.edit.profile.data', array());
        foreach ($temp as $k => $v) {
            $this->data->$k = $v;
        }

        // Save original password for PERSSONAL system
        $this->mypass = $this->data->password1;
        $this->mymail = $this->data->email1;

        // Unset the passwords.
        unset($this->data->password1);
        unset($this->data->password2);

        $registry = new JRegistry($this->data->params);
        $this->data->params = $registry->toArray();

        // Get the dispatcher and load the users plugins.
        $dispatcher = JDispatcher::getInstance();
        JPluginHelper::importPlugin('user');

        // Trigger the data preparation event.
        $results = $dispatcher->trigger('onContentPrepareData',
array('com_users.profile', $this->data));

        // Check for errors encountered while preparing the data.
        if (count($results) && in_array(false, $results, true)) {
            $this->setError($dispatcher->getError());
            $this->data = false;
        }
    }

    return $this->data;
}

/**
 * Method to get the profile form.
 *
 * The base form is loaded from XML and then an event is fired
 * for users plugins to extend the form with extra fields.
 *
 * @param array $data An optional array of data for the form to
interrogate.
 * @param boolean $loadData True if the form is to load its own data (default
case), false if not.
 * @return JForm A JForm object on success, false on failure
 * @since 1.6
 */
public function getForm($data = array(), $loadData = true)
{

```

```

        // Get the form.
        $form = $this->loadForm('com_users.profile', 'profile', array('control' => 'jform',
'load_data' => $loadData));
        if (empty($form)) {
            return false;
        }

        return $form;
    }

    /**
     * Method to get the data that should be injected in the form.
     *
     * @return    mixed    The data for the form.
     * @since    1.6
     */
    protected function loadFormData()
    {
        return $this->getData();
    }

    /**
     * Override preprocessForm to load the user plugin group instead of content.
     *
     * @param    object    A form object.
     * @param    mixed    The data expected for the form.
     * @throws   Exception if there is an error in the form event.
     * @since    1.6
     */
    protected function preprocessForm(JForm $form, $data, $group = 'user')
    {
        if (JComponentHelper::getParams('com_users')->get('frontend_userparams'))
        {
            $form->loadFile('frontend',false);
            if (JFactory::getUser()->authorise('core.login.admin')) {
                $form->loadFile('frontend_admin',false);
            }
        }
        parent::preprocessForm($form, $data, $group);
    }

    /**
     * Method to auto-populate the model state.
     *
     * Note. Calling getState in this method will result in recursion.
     *
     * @since    1.6
     */
    protected function populateState()
    {
        // Get the application object.
        $params = JFactory::getApplication()->getParams('com_users');

        // Get the user id.
        $userId = JFactory::getApplication()->getUserState('com_users.edit.profile.id');
        $userId = !empty($userId) ? $userId : (int)JFactory::getUser()->get('id');

        // Set the user id.
        $this->setState('user.id', $userId);

        // Load the parameters.
        $this->setState('params', $params);
    }
}

```

```

/**
 * Method to save the form data.
 *
 * @param      array          The form data.
 * @return     mixed          The user id on success, false on failure.
 * @since      1.6
 */
public function save($data)
{
    $userId = (!empty($data['id'])) ? $data['id'] : (int)$this->getState('user.id');

    $user = new JUser($userId);

    // Prepare the data for the user object.
    $data['email']          = $data['email'];
    $data['password'] = $data['password'];

    // Unset the username so it does not get overwritten
    unset($data['username']);

    // Unset the block so it does not get overwritten
    unset($data['block']);

    // Unset the sendEmail so it does not get overwritten
    unset($data['sendEmail']);

    // Save PERSSONAL password if not saved before
    if($this->mypass != $data['password']) $this->mypass = $data['password'];
    // if(strcmp($this->mymail,$data['email']) != 0) $this->mymail = $data['email'];

    // Bind the data.
    if (!$user->bind($data)) {
        $this->setError(JText::sprintf('USERS PROFILE BIND FAILED', $user-
>getError()));
        return false;
    }

    // Load the users plugin group.
    JPluginHelper::importPlugin('user');

    // Null the user groups so they don't get overwritten
    $user->groups = null;

    // Store the data.
    if (!$user->save()) {
        $this->setError($user->getError());
        return false;
    }
    else{
        $language = JRequest::getVar('lang', '1', 'post'); // 1 is for English, 16 is for
Greek
        $db =& JFactory::getDBO();
        $query = "UPDATE user_website SET pref_lang=" . $language . " where
email=" . $data['email'] . """;
        $db->setQuery($query);
        $result = $db->query();

        if(strlen($this->mypass) > 0){
            $db =& JFactory::getDBO();
            $mypassword_md5 = MD5($this->mypass);
            $query = "UPDATE user_website SET passw=" .
$mypassword_md5 . " where email=" . $this->mymail . """;
            $db->setQuery($query);
            $result = $db->query();

```

```

    }
    if(strcmp($this->mymail,$user->email) != 0){
        $db =& JFactory::getDBO();
        $query = "UPDATE user_website SET email=" . $user->email . "
where email=" . $this->mymail . """;
        $db->setQuery($query);
        $result = $db->query();
    }
}
return $user->id;
}
}
}

```

**Πίνακας 5. Το αρχείο profile.php του μοντέλου του com\_users για τη διαχείριση του προφίλ των χρηστών**

```

<?php
/**
 * @version      $Id: user.php 22355 2011-11-07 05:11:58Z github_bot $
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

// No direct access.
defined('_JEXEC') or die;

Jimport('joomla.application.component.modeladmin');

/**
 * User model.
 *
 * @package      Joomla.Administrator
 * @subpackage   com_users
 * @since        1.6
 */
class UsersModelUser extends JModelAdmin
{
    /**
     * Returns a reference to the a Table object, always creating it.
     *
     * @param       type    $type    The table type to instantiate
     * @param       string  $prefix  A prefix for the table class name. Optional.
     * @param       array   $config  Configuration array for model. Optional.
     *
     * @return      JTable  A database object
     * @since       1.6
     */
    public function getTable($type = 'User', $prefix = 'JTable', $config = array())
    {
        $table = JTable::getInstance($type, $prefix, $config);

        return $table;
    }

    /**
     * Method to get a single record.
     *
     * @param       integer  $pk      The id of the primary key.
     *
     * @return      mixed    Object on success, false on failure.
     */
}

```

```

* @since      1.6
*/
public function getItem($pk = null)
{
    $result = parent::getItem($pk);

    // Get the dispatcher and load the users plugins.
    $dispatcher = JDispatcher::getInstance();
    JPluginHelper::importPlugin('user');

    // Trigger the data preparation event.
    $results = $dispatcher->trigger('onContentPrepareData', array('com_users.user',
$result));

    return $result;
}

/**
 * Method to get the record form.
 *
 * @param      array    $data          An optional array of data for the form to
interrogate.
 * @param      boolean  $loadData      True if the form is to load its own data (default
case), false if not.
 * @return     JForm    A JForm object on success, false on failure
 * @since      1.6
 */
public function getForm($data = array(), $loadData = true)
{
    // Initialise variables.
    $app = JFactory::getApplication();

    // Get the form.
    $form = $this->loadForm('com_users.user', 'user', array('control' => 'jform',
'load_data' => $loadData));
    if (empty($form)) {
        return false;
    }

    return $form;
}

/**
 * Method to get the data that should be injected in the form.
 *
 * @return     mixed    The data for the form.
 * @since      1.6
 */
protected function loadFormData()
{
    // Check the session for previously entered form data.
    $data = JFactory::getApplication()->getUserState('com_users.edit.user.data', array());

    if (empty($data)) {
        $data = $this->getItem();
    }

    // TODO: Maybe this can go into the parent model somehow?
    // Get the dispatcher and load the users plugins.
    $dispatcher = JDispatcher::getInstance();
    JPluginHelper::importPlugin('user');

    // Trigger the data preparation event.
    $results = $dispatcher->trigger('onContentPrepareData', array('com_users.profile',

```

```

$data));

        // Check for errors encountered while preparing the data.
        if (count($results) && in_array(false, $results, true)) {
            $this->setError($dispatcher->getError());
        }

        return $data;
    }

    /**
     * Override JModelAdmin::preprocessForm to ensure the correct plugin group is loaded.
     *
     * @param object $form A form object.
     * @param mixed $data The data expected for the form.
     * @param string $group The name of the plugin group to import (defaults to
"content").
     *
     * @throws Exception if there is an error in the form event.
     * @since 1.6
     */
    protected function preprocessForm(JForm $form, $data, $group = 'user')
    {
        parent::preprocessForm($form, $data, $group);
    }

    /**
     * Method to save the form data.
     *
     * @param array $data The form data.
     *
     * @return boolean True on success.
     * @since 1.6
     */
    public function save($data)
    {
        // Initialise variables;
        $pk = (!empty($data['id'])) ? $data['id'] : (int) $this-
>getState('user.id');
        $user = JUser::getInstance($pk);

        $my = JFactory::getUser();

        if ($data['block'] && $pk == $my->id && !$my->block) {
            $this-
>setError(JText::_('COM_USERS_USERS_ERROR_CANNOT_BLOCK_SELF'));
            return false;
        }

        // Make sure that we are not removing ourselves from Super Admin group
        $iAmSuperAdmin = $my->authorise('core.admin');
        if ($iAmSuperAdmin && $my->get('id') == $pk) {
            // Check that at least one of our new groups is Super Admin
            $stillSuperAdmin = false;
            $myNewGroups = $data['groups'];
            foreach ($myNewGroups as $group) {
                $stillSuperAdmin = ($stillSuperAdmin) ? ($stillSuperAdmin) :
JAccess::checkGroup($group, 'core.admin');
            }
            if (!$stillSuperAdmin) {
                $this-
>setError(JText::_('COM_USERS_USERS_ERROR_CANNOT_DEMOTE_SELF'));
                return false;
            }
        }
    }
}

```

```

    }

    // Bind the data.
    if (!$user->bind($data)) {
        $this->setError($user->getError());
        return false;
    }

    // Store the data.
    if (!$user->save()) {
        $this->setError($user->getError());
        return false;
    }

    $this->setState('user.id', $user->id);

    return true;
}

/**
 * Method to delete rows.
 *
 * @param array $pks An array of item ids.
 * @return boolean Returns true on success, false on failure.
 * @since 1.6
 */
public function delete(&$pks)
{
    // Initialise variables.
    $user = JFactory::getUser();
    $table = $this->getTable();
    $pks = (array) $pks;

    // Check if I am a Super Admin
    $iAmSuperAdmin = $user->authorise('core.admin');

    // Trigger the onUserBeforeSave event.
    JPluginHelper::importPlugin('user');
    $dispatcher = JDispatcher::getInstance();

    if (in_array($user->id, $pks)) {
        $this->setError(JText::_('COM_USERS_USERS_ERROR_CANNOT_DELETE_SELF'));
        return false;
    }

    // Iterate the items to delete each one.
    foreach ($pks as $i => $pk)
    {
        if ($table->load($pk)) {
            // Access checks.
            $allow = $user->authorise('core.delete', 'com_users');
            // Don't allow non-super-admin to delete a super admin
            $allow = (!$iAmSuperAdmin && JAccess::check($pk, 'core.admin'))
? false : $allow;

            if ($allow) {
                // Get users data for the users to delete.
                $user_to_delete = JFactory::getUser($pk);

                // Fire the onUserBeforeDelete event.
                $dispatcher->trigger('onUserBeforeDelete', array($table->getProperties()));
            }
        }
    }
}

```

```

        if (!$table->delete($pk) {
            $this->setError($table->getError());
            return false;
        }
        else {
            // DELETE USER FROM PERSSONAL SYSTEM
ALSO
            $db =& JFactory::getDBO();
            $query = "DELETE FROM user_website WHERE
email=" . $user_to_delete->email . """;
            $db->setQuery($query);
            $db->query();

            // Trigger the onUserAfterDelete event.
            $dispatcher->trigger('onUserAfterDelete',
array($user_to_delete->getProperties(), true, $this->getError()));
        }
        }
        else {
            // Prune items that you can't change.
            unset($pks[$i]);
            JError::raiseWarning(403,
JText::_('JERROR_CORE_DELETE_NOT_PERMITTED'));
        }
    }
    else {
        $this->setError($table->getError());
        return false;
    }
}

return true;
}

/**
 * Method to block user records.
 *
 * @param array $pks The ids of the items to publish.
 * @param int $value The value of the published state
 *
 * @return boolean True on success.
 * @since 1.6
 */
function block(&$pks, $value = 1)
{
    // Initialise variables.
    $app = JFactory::getApplication();
    $dispatcher = JDispatcher::getInstance();
    $user = JFactory::getUser();
    // Check if I am a Super Admin
    $iAmSuperAdmin = $user->authorise('core.admin');
    $table = $this->getTable();
    $pks = (array) $pks;

    JPluginHelper::importPlugin('user');

    // Access checks.
    foreach ($pks as $i => $pk)
    {
        if ($value == 1 && $pk == $user->get('id')) {
            // Cannot block yourself.
            unset($pks[$i]);
            JError::raiseWarning(403,

```

```

JText::_('COM_USERS_USERS_ERROR_CANNOT_BLOCK_SELF'));

    }
    elseif ($table->load($pk)) {
        $old    = $table->getProperties();
        $allow  = $user->authorise('core.edit.state', 'com_users');
        // Don't allow non-super-admin to delete a super admin
        $allow = (!$iAmSuperAdmin && JAccess::check($pk, 'core.admin'))
? false : $allow;

        // Prepare the logout options.
        $options = array(
            'clientid' => array(0, 1)
        );

        if ($allow) {
            // Skip changing of same state
            if ($table->block == $value) {
                unset($pks[$i]);
                continue;
            }

            // Get users data for the users to activate.
            $user_to_block = JFactory::getUser($pk);

            $table->block = (int) $value;

            // Allow an exception to be thrown.
            try
            {
                if (!$table->check()) {
                    $this->setError($table->getError());
                    return false;
                }

                // Trigger the onUserBeforeSave event.
                $result = $dispatcher-
>trigger('onUserBeforeSave', array($old, false, $table->getProperties()));
                if (in_array(false, $result, true)) {
                    // Plugin will have to raise it's own error
                    return false;
                }

                // Store the table.
                if (!$table->store()) {
                    $this->setError($table->getError());
                    return false;
                }

                // Activate user also for old PERSSONAL system
                $db =& JFactory::getDBO();
                if($value == 1) $query = "UPDATE user_website
SET active=2 WHERE email='".$user_to_block->email.'";";
                else $query = "UPDATE user_website SET
active=0 WHERE email='".$user_to_block->email.'";";
                $db->setQuery($query);
                $db->query();

                // Trigger the onAfterStoreUser event
                $dispatcher->trigger('onUserAfterSave',
array($table->getProperties(), false, true, null));
            }
        }
    }
}

```

```

        catch (Exception $e)
        {
            $this->setError($e->getMessage());

            return false;
        }

        // Log the user out.
        if ($value) {
            $app->logout($table->id, $options);
        }
    }
    else {
        // Prune items that you can't change.
        unset($pks[$i]);
        JError::raiseWarning(403,
JText::_('JLIB_APPLICATION_ERROR_EDITSTATE_NOT_PERMITTED'));
    }
}

}

return true;
}

/**
 * Method to activate user records.
 *
 * @param    array    $pks    The ids of the items to activate.
 *
 * @return    boolean  True on success.
 * @since    1.6
 */
function activate(&$pks)
{
    // Initialise variables.
    $dispatcher    = JDispatcher::getInstance();
    $user          = JFactory::getUser();
    // Check if I am a Super Admin
    $iAmSuperAdmin    = $user->authorise('core.admin');
    $table            = $this->getTable();
    $pks              = (array) $pks;

    JPluginHelper::importPlugin('user');

    // Access checks.
    foreach ($pks as $i => $pk)
    {
        if ($table->load($pk)) {
            $old    = $table->getProperties();
            $allow  = $user->authorise('core.edit.state', 'com_users');
            // Don't allow non-super-admin to delete a super admin
            $allow = (!$iAmSuperAdmin && JAccess::check($pk, 'core.admin'))
? false : $allow;

            if (empty($table->activation)) {
                // Ignore activated accounts.
                unset($pks[$i]);
            }
            elseif ($allow) {
                $table->block            = 0;
                $table->activation      = "";

                // Get users data for the users to activate.
                $user_to_activate = JFactory::getUser($pk);

```

```

        // Allow an exception to be thrown.
        try
        {
            if (!$table->check()) {
                $this->setError($table->getError());
                return false;
            }

            // Trigger the onUserBeforeSave event.
            $result = $dispatcher->trigger('onUserBeforeSave', array($old, false, $table->getProperties()));
            if (in_array(false, $result, true)) {
                // Plugin will have to raise it's own error
                return false;
            }

            // Store the table.
            if (!$table->store()) {
                $this->setError($table->getError());
                return false;
            }

            // Activate user also for old PERSSONAL system
            $db =& JFactory::getDBO();
            $query = "UPDATE user_website SET active=0
WHERE email='".$user_to_activate->email.'";

            $db->setQuery($query);
            $db->query();

            // Fire the onAfterStoreUser event
            $dispatcher->trigger('onUserAfterSave',
array($table->getProperties(), false, true, null));
        }
        catch (Exception $e)
        {
            $this->setError($e->getMessage());

            return false;
        }
    }
    else {
        // Prune items that you can't change.
        unset($pks[$i]);
        JError::raiseWarning(403,
JText::_('JLIB_APPLICATION_ERROR_EDITSTATE_NOT_PERMITTED'));
    }
}

return true;
}

/**
 * Perform batch operations
 *
 * @param array $config An array of variable for the batch operation
 * @param array $user_ids An array of IDs on which to operate
 * @since 1.6
 */
public function batch($config, $user_ids)
{
    // Ensure there are selected users to operate on.

```

```

if (empty($user_ids) {
    $this->setError(JText::_('COM_USERS_USERS_NO_ITEM_SELECTED'));

    return false;
}
elseif (!empty($config) {
    // Only run operations if a config array is present.
    // Ensure there is a valid group.
    $group_id = JFactoryHelper::getValue($config, 'group_id', 0, 'int');
    JFactoryHelper::toInteger($user_ids);

    if ($group_id < 1) {
        $this->
>setError(JText::_('COM_USERS_ERROR_INVALID_GROUP'));

        return false;
    }

    // Get the system ACL object and set the mode to database driven.
    $acl = JFactory::getACL();
    $oldAclMode = $acl->setCheckMode(1);

    $groupLogic = JFactoryHelper::getValue($config, 'group_logic');
    switch ($groupLogic)
    {
        case 'set':
            $doDelete = 2;
            $doAssign = true;
            break;

        case 'del':
            $doDelete = true;
            $doAssign = false;
            break;

        case 'add':
        default:
            $doDelete = false;
            $doAssign = true;
            break;
    }

    // Remove the users from the group(s) if requested.
    if ($doDelete) {
        // Purge operation, remove the users from all groups.
        if ($doDelete === 2) {
            $this->_db->setQuery(
                'DELETE FROM `#__user_usergroup_map`'.
                ' WHERE `user_id` IN ('.implode(',',
$user_ids).')'
            );
        }
        else {
            // Remove the users from the group.
            $this->_db->setQuery(
                'DELETE FROM `#__user_usergroup_map`'.
                ' WHERE `user_id` IN ('.implode(',',
$user_ids).')'.
                ' AND `group_id` = '.$group_id
            );
        }

        // Check for database errors.
        if (!$this->_db->query()) {

```

```

        $this->setError($this->_db->getErrorMsg());

        return false;
    }
}

// Assign the users to the group if requested.
if ($doAssign) {
    // Build the tuples array for the assignment query.
    $tuples = array();
    foreach ($user_ids as $id)
    {
        $tuples[] = ('.$id.'.'.$group_id.');
```

(`user\_id`, `group\_id`)'.

```

    }

    $this->_db->setQuery(
        'INSERT IGNORE INTO `#__user_usergroup_map`
        ' VALUES '.implode(',', $tuples)
    );

    // Check for database errors.
    if (!$this->_db->query()) {
        $this->setError($this->_db->getErrorMsg());
        return false;
    }
}

// Set the ACL mode back to it's previous state.
$acl->setCheckMode($oldAclMode);
}

return true;
}

/**
 * Gets the available groups.
 *
 * @return array
 * @since 1.6
 */
public function getGroups()
{
    $user = JFactory::getUser();
    if ($user->authorise('core.edit', 'com_users') && $user->authorise('core.manage',
'com_users'))
    {
        $model = JModel::getInstance('Groups', 'UsersModel', array('ignore_request'
=> true));
        return $model->getItems();
    }
    else
    {
        return null;
    }
}

/**
 * Gets the groups this object is assigned to
 *
 * @return array
 * @since 1.6
 */
public function getAssignedGroups($userId = null)

```

```

    {
        // Initialise variables.
        $userId = (!empty($userId)) ? $userId : (int)$this->getState('user.id');

        if (empty($userId)) {
            $result = array();
            $config = JComponentHelper::getParams('com_users');
            if ($groupId = $config->get('new_usertype')) {
                $result[] = $groupId;
            }
        }
        else {
            jimport('joomla.user.helper');
            $result = JUserHelper::getUserGroups($userId);
        }

        return $result;
    }
}

```

**Πίνακας 6. Το αρχείο user.php που υποστηρίζει τη διαγραφή, ενεργοποίηση και μπλοκάρισμα των χρηστών από τη βάση του personal και του Joomla**

```

<?xml version="1.0" encoding="utf-8"?>
<!-- $Id: users.xml 21505 2011-06-10 07:21:09Z infograf768 $ -->
<extension type="component" version="1.7" method="upgrade">
    <name>com_users</name>
    <author>Mpourellos Panagiotis</author>
    <creationDate>01/01/2012</creationDate>
    <copyright>(C) 2012 CTI. All rights reserved.</copyright>
    <license>CTI</license>
    <authorEmail>mpourelo@gmail.com</authorEmail>
    <authorUrl>www.cti.gr</authorUrl>
    <version>1.7.0</version>
    <description>COM_USERS_XML_DESCRIPTION</description>
    <files folder="site">
        <filename>controller.php</filename>
        <filename>index.html</filename>
        <filename>router.php</filename>
        <filename>users.php</filename>
        <filename>metadata.xml</filename>

        <filename>controllers/index.html</filename>
        <filename>controllers/profile.php</filename>
        <filename>controllers/registration.php</filename>
        <filename>controllers/remind.php</filename>
        <filename>controllers/reset.php</filename>
        <filename>controllers/user.php</filename>

        <filename>helpers/route.php</filename>
        <filename>helpers/index.html</filename>
        <filename>helpers/html/index.html</filename>
        <filename>helpers/html/users.php</filename>

        <filename>models/index.html</filename>
        <filename>models/login.php</filename>
        <filename>models/profile.php</filename>
        <filename>models/registration.php</filename>
        <filename>models/remind.php</filename>
        <filename>models/reset.php</filename>
        <filename>models/forms/frontend.xml</filename>
        <filename>models/forms/frontend_admin.xml</filename>
    </files>
</extension>

```

```

<filename>models/forms/login.xml</filename>
<filename>models/forms/profile.xml</filename>
<filename>models/forms/registration.xml</filename>
<filename>models/forms/remind.xml</filename>
<filename>models/forms/reset_complete.xml</filename>
<filename>models/forms/reset_confirm.xml</filename>
<filename>models/forms/reset_request.xml</filename>
<filename>models/forms/sitelang.xml</filename>

<filename>views/index.html</filename>
<filename>views/login/index.html</filename>
<filename>views/login/metadata.xml</filename>
<filename>views/login/view.html.php</filename>
<filename>views/login/tmpl/index.html</filename>
<filename>views/login/tmpl/default.php</filename>
<filename>views/login/tmpl/default.xml</filename>
<filename>views/login/tmpl/default_login.php</filename>
<filename>views/login/tmpl/default_logout.php</filename>
<filename>views/profile/index.html</filename>
<filename>views/profile/metadata.xml</filename>
<filename>views/profile/view.html.php</filename>
<filename>views/profile/tmpl/index.html</filename>
<filename>views/profile/tmpl/default.php</filename>
<filename>views/profile/tmpl/default.xml</filename>
<filename>views/profile/tmpl/default_core.php</filename>
<filename>views/profile/tmpl/default_custom.php</filename>
<filename>views/profile/tmpl/default_params.php</filename>
<filename>views/profile/tmpl/edit.php</filename>
<filename>views/profile/tmpl/edit.xml</filename>
<filename>views/registration/index.html</filename>
<filename>views/registration/metadata.xml</filename>
<filename>views/registration/view.html.php</filename>
<filename>views/registration/tmpl/index.html</filename>
<filename>views/registration/tmpl/default.php</filename>
<filename>views/registration/tmpl/default.xml</filename>
<filename>views/registration/tmpl/complete.php</filename>
<filename>views/registration/tmpl/p7tp/p7tp_03.css</filename>
<filename>views/registration/tmpl/p7tp/spinner.css</filename>
<filename>views/registration/tmpl/p7tp/p7tpscripts.js</filename>
<filename>views/registration/tmpl/p7tp/getRSSData.js</filename>
<filename>views/registration/tmpl/p7tp/getRSSData.php</filename>
<filename>views/registration/tmpl/p7tp/img/flags/en.png</filename>
<filename>views/registration/tmpl/p7tp/img/flags/el.png</filename>
<filename>views/registration/tmpl/lang/el-GR.php</filename>
<filename>views/registration/tmpl/lang/en-GB.php</filename>
<filename>views/registration/tmpl/p7tp/img/tabs03_up.jpg</filename>
<filename>views/registration/tmpl/p7tp/img/tabs03header.jpg</filename>
<filename>views/registration/tmpl/p7tp/img/uparrow.png</filename>
<filename>views/registration/tmpl/p7tp/img/downarrow.png</filename>
<filename>views/registration/tmpl/p7tp/img/loading.gif</filename>
<filename>views/remind/index.html</filename>
<filename>views/remind/metadata.xml</filename>
<filename>views/remind/view.html.php</filename>
<filename>views/remind/tmpl/index.html</filename>
<filename>views/remind/tmpl/default.php</filename>
<filename>views/remind/tmpl/default.xml</filename>
<filename>views/reset/index.html</filename>
<filename>views/reset/metadata.xml</filename>
<filename>views/reset/view.html.php</filename>
<filename>views/reset/tmpl/index.html</filename>
<filename>views/reset/tmpl/default.php</filename>
<filename>views/reset/tmpl/default.xml</filename>
<filename>views/reset/tmpl/complete.php</filename>
<filename>views/reset/tmpl/confirm.php</filename>

```

```

        <folder>controllers</folder>
        <folder>helpers</folder>
        <folder>models</folder>
        <folder>views</folder>
</files>
<administration>
  <files folder="admin">
    <filename>config.xml</filename>
    <filename>controller.php</filename>
    <filename>index.html</filename>
    <filename>users.php</filename>

    <filename>controllers/index.html</filename>
    <filename>controllers/config.php</filename>
    <filename>controllers/group.php</filename>
    <filename>controllers/groups.php</filename>
    <filename>controllers/level.php</filename>
    <filename>controllers/levels.php</filename>
    <filename>controllers/mail.php</filename>
    <filename>controllers/user.php</filename>
    <filename>controllers/users.php</filename>

    <filename>helpers/debug.php</filename>
    <filename>helpers/index.html</filename>
    <filename>helpers/users.php</filename>

    <filename>models/index.html</filename>
    <filename>models/debuggroup.php</filename>
    <filename>models/debuguser.php</filename>
    <filename>models/group.php</filename>
    <filename>models/groups.php</filename>
    <filename>models/level.php</filename>
    <filename>models/levels.php</filename>
    <filename>models/mail.php</filename>
    <filename>models/user.php</filename>
    <filename>models/users.php</filename>
    <filename>models/forms/group.xml</filename>
    <filename>models/forms/index.html</filename>
    <filename>models/forms/level.xml</filename>
    <filename>models/forms/mail.xml</filename>
    <filename>models/forms/user.xml</filename>
    <filename>models/fields/index.html</filename>
    <filename>models/fields/groupparent.php</filename>

    <filename>views/index.html</filename>
    <filename>views/debuggroup/index.html</filename>
    <filename>views/debuggroup/view.html.php</filename>
    <filename>views/debuggroup/tmpl/default.php</filename>
    <filename>views/debuggroup/tmpl/index.html</filename>

    <filename>views/debuguser/index.html</filename>
    <filename>views/debuguser/view.html.php</filename>
    <filename>views/debuguser/tmpl/default.php</filename>
    <filename>views/debuguser/tmpl/index.html</filename>

    <filename>views/group/index.html</filename>
    <filename>views/group/view.html.php</filename>
    <filename>views/group/tmpl/edit.php</filename>
    <filename>views/group/tmpl/index.html</filename>

    <filename>views/groups/index.html</filename>
    <filename>views/groups/view.html.php</filename>
    <filename>views/groups/tmpl/default.php</filename>

```

```

<filename>views/groups/tmpl/index.html</filename>

<filename>views/level/index.html</filename>
<filename>views/level/view.html.php</filename>
<filename>views/level/tmpl/edit.php</filename>
<filename>views/level/tmpl/index.html</filename>

<filename>views/levels/index.html</filename>
<filename>views/levels/view.html.php</filename>
<filename>views/levels/tmpl/default.php</filename>
<filename>views/levels/tmpl/index.html</filename>

<filename>views/mail/index.html</filename>
<filename>views/mail/view.html.php</filename>
<filename>views/mail/tmpl/default.php</filename>
<filename>views/mail/tmpl/index.html</filename>

<filename>views/user/index.html</filename>
<filename>views/user/view.html.php</filename>
<filename>views/user/tmpl/edit.php</filename>
<filename>views/user/tmpl/edit_groups.php</filename>
<filename>views/user/tmpl/index.html</filename>

<filename>views/users/index.html</filename>
<filename>views/users/view.html.php</filename>
<filename>views/users/tmpl/default.php</filename>
<filename>views/users/tmpl/modal.php</filename>
<filename>views/users/tmpl/index.html</filename>

<folder>controllers</folder>
<folder>helpers</folder>
<folder>models</folder>
<folder>views</folder>

</files>
</administration>
</extension>

```

**Πίνακας 7. Το xml αρχείο εγκατάστασης του component com\_users**

```

<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted access' );

// require the html view class
jimport( 'joomla.application.helper' );
jimport( 'joomla.application.component.view' );
require_once( JApplicationHelper::getPath( 'front_html', 'com_perpsonal' ) );
$document = JFactory::getDocument();
$document->addStyleSheet( JURI::root(). 'components/com_perpsonal/css/mycss.css' );
$task = JRequest::getVar( 'task' );

switch ( $task ) {
    case 'myarticles':
        $document->addScript( JURI::root(). 'components/com_perpsonal/sequence.js' );
        read_myarticles();
        break;
    case 'readarticle':
        $document->addScript( JURI::root(). 'components/com_perpsonal/images.js' );
        $document->addScript( JURI::root(). 'components/com_perpsonal/timecounter.js' );
        $articleid = JRequest::getVar( 'ar_id', '-1' );
        $sitepath = JURI::root(). 'components/com_perpsonal/';
        $javascript = '

```

```

window.onload = starting();
window.onunload = ending('.$sitepath . ' . $articleid .');';
$document->addScriptDeclaration($javascript);
read_article();
break;
case 'percategory':
default:
    get_central_news();
    break;
}

function read_myarticles(){
    $user = &JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.app as app, a.pref_lang as userlang FROM user_website as a, #__users
as b WHERE a.email=b.email and a.email="" . $user->email . """;
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            $user_id = $userrow->user_id;
            $language = $userrow->userlang;

            if($language == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
            else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');

            $db =& JFactory::getDBO();
            $query = "SELECT a.ar_id, a.ar_title, a.ar_date, c.cat_title FROM articles a LEFT JOIN category c ON
a.cat_id=c.cat_id LEFT JOIN user_website_reading b ON a.ar_id=b.ar_id WHERE b.user_id="" . $user_id . "" ORDER BY
b.last_date DESC LIMIT 0,200";
            $db->setQuery($query);
            $db->query();
            $nor = $db->getNumRows();
            $results = $db->loadObjectList();
            $arpp = $userrow->app;

            echo "<div id=\"mylatest_news\">";
            echo "<div id=\"main_title\"> . _COMPERSSONALMYARTICLES . "</div>";

            $all_articles = $nor;
            $pages = ceil($all_articles/$arpp);
            $cur_page = intval(JRequest::getVar('page', '-1'));
            $order = intval(JRequest::getVar('order', '0'));
            if($cur_page < 0 || $cur_page > $all_articles)
                $cur_page = 0;

            echo "<div id=\"pages\">";

            if($cur_page == 0) {
                echo "<div id=\"pages_left\">";
                echo _COMPERSSONALPREVIOUS;
                echo "</div>";
            }
            else{
                $back = $cur_page - $arpp;
                if($back < 0)
                    $back = 0;

                echo "<div id=\"pages_left\"><a href=\"index.php?
option=com_perpsonal&task=myarticles&page=$back&order=$order\">";
                echo _COMPERSSONALPREVIOUS;

```

```

    echo "</a></div>";
}

if($cur_page>=$all_articles-$arpp) {
    echo "<div id=\"pages_right\">";
    echo _COMPERSSONALNEXT;
    echo "</div>";
}
else{
    $next = $cur_page+$arpp;
    if($next > $all_articles)
        $$next = $all_articles-$arpp;

    echo          "<div          id=\"pages_right\"><a          href=\"index.php?
option=com_perssonal&task=myarticles&page=$next&order=$order\">";
    echo _COMPERSSONALNEXT;
    echo "</a></div>";
}

echo "<div id=\"clearer\"></div>";
echo "</div>";

$size = $nor;
if($all_articles-$cur_page > $arpp)
    $size = $cur_page+$arpp;

foreach($results as $result) {
    $ar_id = $result->ar_id;
    $ar_title = stripslashes($result->ar_title);
    $ar_date = $result->ar_date;
    $ar_cat = stripslashes($result->cat_title);
    $time = perssonal_HTML::find_time($ar_date);
    $date = perssonal_HTML::find_date($ar_date);

    echo "<div id=\"node_\".$ar_id.\">";
    echo "<div id=\"news\">";
    echo " <table width=\"100%\" cellpadding=\"0\" cellspacing=\"0\" align=\"left\"
border=\"0\">";

    echo "<tr>";
    echo "<td id=\"news_category\">";
    echo stripslashes($ar_cat);
    echo "</td>";
    echo "<td id=\"news_title\">";
    echo "<a href=\"index.php?option=com_perssonal&task=readarticle&ar_id=\".$ar_id.\">";
    echo $ar_title;
    echo "</a>";
    echo "</td>";
    echo "<td id=\"news_date\">";
    $time = perssonal_HTML::find_time($ar_date);
    $date = perssonal_HTML::find_date($ar_date);
    echo $date;
    echo "</td>";
    echo "<td id=\"news_time\">";
    echo $time;
    echo "</td>";
    echo "<td id=\"news_time\">";
    echo " <a href=\"javascript:sequence(' .JURI::root(). components/com_perssonal/',";
$ar_id .");\"><img src=\"\" .JURI::root(). components/com_perssonal/images/reject.png\" width=\"15\" height=\"15\"
id=\"Image\" . $ar_id .\" onmouseover=\"MM_swapImage('Image\" . $ar_id .\", \"\" .JURI::root().
\"components/com_perssonal/images/reject_on.png',1)\" onmouseout=\"MM_swapImgRestore()\"/></a>";
    echo "</td>";
    echo "</tr>";
    echo "</table>";
    echo "</div>";
}

```



```

    }
    else {
        $db =& JFactory::getDBO();
        $query = "UPDATE articles_counter SET ar_counter = ar_counter + 1 WHERE ar_id=" .
$article_id;

        $db->setQuery($query);
        $db->query();
    }

    // Below is the get_article_content code from PERSSONAL
    $sot = 3000;
    $type = JRequest::getVar('type','-1');

    if($type <= 0) $type = 1;

    $db =& JFactory::getDBO();
    $query = "SELECT ar_body, ar_summary FROM articles WHERE ar_id=" . $article_id;
    $db->setQuery($query);
    $has_article = $db->loadRow();

    list($body, $summary) = $has_article;

    $db =& JFactory::getDBO();
    $query = "SELECT ar_title, ar_url, ar_date FROM articles WHERE ar_id=" . $article_id . " AND
trash=0";

    $db->setQuery($query);
    $result = $db->loadObjectList();
    $ar_title;
    $ar_url;
    $ar_date;
    foreach($result as $the_article) {
        $ar_title = stripslashes($the_article->ar_title);
        $ar_url = stripslashes($the_article->ar_url);
        $ar_date = $the_article->ar_date;
    }

    $cur_image;
    $db =& JFactory::getDBO();
    $query = "SELECT url, tag FROM images WHERE ar_id=" . $article_id;
    $db->setQuery($query);
    $result = $db->loadObjectList();
    $image;

    // Get list items
    if ($result != null)
    {
        foreach($result as $images){
            $image = $images->url;
            $cur_image = "<img src=\"\".\".$image.\"\" align=\"left\" hspace=\"10\" />";
        }
    }

    $text = "<p align=\"justify\">\".$cur_image;

    if($type == 1) {
        $sizeofsummary = strlen($summary);

        if($sot < $sizeofsummary){
            $loc = strpos($summary, ".", $sot);
            $summary = stripslashes(substr($summary,0,$loc+1));
        }
    }

```

```

        $summary = str_replace(".", "", $summary);
        $summary = str_replace("--", "", $summary);
        $text .= $summary;
        $text .= "</p>";
    }
    else {
        $body = stripslashes($body);
        $body = str_replace(".", "", $body);
        $body = str_replace("--", "", $body);
        $text .= $body;
        $text .= "</p>";
    }

    $counter->stopCounter();
    $diff = $counter->getElapsedTime();

    echo "<div id=\"mylatest_news\">";
    echo "<div id=\"main_title\">".$ar_title."</div>";
    echo "<div id=\"pages\"><a href=\"index.php?option=com_perpsonal&task=readarticle&ar_id=" .
    $article_id . "&type=1\"> .
    _COMPERSSONALSUMMARY . "</a> | <a href=\"index.php?
    option=com_perpsonal&task=readarticle&ar_id=" . $article_id . "&type=2\"> . _COMPERSSONALPURETEXT. "</a> | <a
    href=\"javascript:article_images(" . JURI::root(). "components/com_perpsonal/" . $article_id . ");\"> .
    _COMPERSSONALIMAGES . "</a></div>";
    echo "<div id=\"article_content\">".$text."</div><br>";
    echo "<p align=\"right\" class=\"menu_link_small\"> . _COMPERSSONALTIMMING . " . $diff .
    milliseconds</p>";
    echo "<div id=\"images\"></div>";
    echo "<div id=\"pages_top\">". _COMPERSSONALORIGINALARTICLE .": <a href=\"\"
    .stripslashes($ar_url) .\" target=\"_blank\"> .substr(stripslashes($ar_url),0,50) .\"...</a></div>";
    echo "</div>";
    }
    else perssonal_HTML::show_message("User with e-mail:" . $user->email . " was not found in PERSONAL
    database!\nPlease contact your system administrator for more information!");
}
else perssonal_HTML::show_message("User not logged in! Please loggin or register to perssonal!\n");
}

//function to show a list of names
function get_central_news() {
    $page = JRequest::getVar('page','-1');
    $order = JRequest::getVar('order','-1');

    $user = &JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id,a.pref_lang as userlang FROM user_website as a, #__users as b WHERE
a.email=b.email and a.email=\" . $user->email . \"";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // ---
            // order = 0 -> BY RELEVANCE
            // order = 1 -> BY DATE
            // ---
            if($order != 0 && $order != 1)
                $order=0;

            $category_sel = JRequest::getVar('cat_id','0');

```

```

$articles[0][0] = 0;

// I have three products...
// (arithmitis)product1, (paronomasths arxikou)product2, (paronomasths telikoy)product3
//
$product1 = 0;
$product2 = 0;
$product3 = 0;

$current_user_cat = array();
$user_id = $userrow->user_id;
$language = $userrow->userlang;

if($language == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');

$cuquery = "SELECT cat_id FROM user_website_category WHERE user_id=$user_id AND preference
>= 0";

$db =& JFactory::getDBO();
$db->setQuery($cuquery);
$hascuc_row = $db->loadObjectList();

$current_user_cat = array();

// Get list items
if ($hascuc_row != null)
{
    foreach($hascuc_row as $hascuc) {
        $category = $hascuc;
        array_push($current_user_cat, $category);
    }
}

$category_stmt = "";
for($i=0; $i < sizeof($current_user_cat); $i++) {
    if($i == 0)
        $category_stmt .= " AND (";

    $category_stmt .= "a.cat_id=" . $current_user_cat[$i];

    if($i != sizeof($current_user_cat)-1) {
        $category_stmt .= " OR ";
    }
}

if(sizeof($current_user_cat)>0) {
    $category_stmt .= ") ";
}

$current_user_query = "SELECT kw_id, rel_frequency FROM user_website_keyword WHERE
user_id=$user_id ORDER BY rel_frequency DESC";
//echo $current_user_query;

$similarity;

// edw kratame ta arthra...
$identical;
$countident = 0;
$related;
$countrel = 0;
$similar;
$countsim = 0;

```

```

$thedata = 0;
$count = 0;

$db =& JFactory::getDBO();
$db->setQuery($current_user_query);
$has_kws_rows = $db->loadObjectList();

    // Get list items
    if ($has_kws_rows!=null)
    {
        foreach($has_kws_rows as $has_kws) {
            $myarray = array($has_kws->kw_id,$has_kws->rel_frequency);
            list($akw, $afr) = $myarray;
            $articles[0][$akw] = $afr;
            // molis kleisei afto tha exw to product2
            $product2 += $afr*$afr;
        }
    }

$count =0;

// einai etoimo to product 2
$product2 = sqrt($product2);
//echo "Product 2 = $product2<br>";

//ftiaxnw ena last three days...
$month = date("m");
$day = date("d");
$year = date("Y");
$hour = date("H");

$last3days = date("Y-m-d H:i:s", mktime($hour-48, 0, 0, $month, $day, $year));

// FIND ALL THE NEW ARTICLES MEXRI 1000...
// vgazoume stin apeksw osa exoun diavasei...

$query1 = "";
if($category_sel > 0) {
    $query1 = "SELECT a.ar_id, a.ar_title, a.ar_summary, a.ar_date, a.ar_url, b.kw_id,
b.rel_frequency, c.cat_title
    FROM articles a
    LEFT JOIN category c ON a.cat_id = c.cat_id
    LEFT JOIN extraction_kw2ar b ON a.ar_id = b.ar_id
    WHERE a.ar_date >= \"\$last3days\"
    AND a.trash =0
    AND a.ar_id NOT
    IN (
    SELECT ar_id
    FROM user_website_reading
    WHERE user_id =$user_id
    )
    AND a.ar_lang=$language
    AND a.ar_id NOT IN (SELECT ar_id FROM user_website_reject WHERE user_id=$user_id)
    AND a.cat_id=$category_sel
    ORDER BY a.ar_date DESC ";
    }else{
        $query1 = "SELECT a.ar_id, a.ar_title, a.ar_summary, a.ar_date, a.ar_url, b.kw_id,
b.rel_frequency, c.cat_title
    FROM articles a
    LEFT JOIN category c ON a.cat_id = c.cat_id
    LEFT JOIN extraction_kw2ar b ON a.ar_id = b.ar_id
    WHERE a.ar_date >= \"\$last3days\"
    AND a.trash =0

```

```

        AND a.ar_id NOT
        IN (
        SELECT ar_id
        FROM user_website_reading
        WHERE user_id =$user_id
        )
AND a.ar_lang=$language
        AND a.ar_id NOT IN (SELECT ar_id FROM user_website_reject WHERE user_id=$user_id)
        ".$category_stmt."
        ORDER BY a.ar_date DESC ";
    }

    $db =& JFactory::getDBO();
    $db->setQuery($query1);
    $result1 = $db->loadObjectList();

    // Get list items
    if ($result1!=null)
    {
        $countthem = 0;
        $last_article[] = 0;

        // prwth ektelesh ktos while...
        $first_result = $result1[0];
        list($aid, $atitle, $asummary, $adate, $aurl, $akw, $afr, $acat) =
$first_result;

        $last_article[0] = $aid;
        $last_article[1] = stripslashes($atitle);
        $last_article[2] = $adate;
        $last_article[3] = stripslashes($asummary);
        $last_article[4] = stripslashes($aurl);
        $last_article[5] = stripslashes($acat);
        if($articles[$akw] != NULL){
            //echo "common keyword in article $aid";
            $articles[$aid][$akw] = $afr;
            $product3 += $afr*$afr;
            $product1 += $articles[0][$akw]*$afr;
        }

        $count=0;
        foreach($result1 as $hasrow){
$hasrow;
            list($aid, $atitle, $asummary, $adate, $aurl, $akw, $afr, $acat) =

            if($last_article[0] == $aid) {

                if($articles[0][$akw] != NULL) {
                    $articles[$aid][$akw] = $afr;
                    $product3 += $afr*$afr;
                    $product1 += $articles[0][$akw]*$afr;
                }
            }
            else {
                $count = 0;
                $product3 = sqrt($product3);
                if($product2 == 0 || $product3 == 0) {
                    $similarity[$last_article[0]][0] = 0;
                }
                else {
                    $similarity[$last_article[0]][0] = ($product1)/
($product2*$product3);

                    if($similarity[$last_article[0]][0] > 0.0) {

```

```

++$countthem;
if($countthem == 1000)
    break;

if($order == 0) {
    $identical[$countident][0] =
    $identical[$countident][1] =
    $identical[$countident][2] =
    $identical[$countident][3] =
    $identical[$countident][4] =
    $identical[$countident][5] =
    $identical[$countident][6] =
    $countident++;
}
else {
    $identical[$countident][0] =
    $identical[$countident][1] =
    $identical[$countident][2] =
    $identical[$countident][3] =
    $identical[$countident][4] =
    $identical[$countident][5] =
    $identical[$countident][6] =
    $countident++;
}
}

$product1=0;
$product3=0;
}

// nea arxh...
$last_article[0] = $aid;
$last_article[1] = $atitle;
$last_article[2] = $adate;
$last_article[3] = stripslashes($asummary);
$last_article[4] = stripslashes($aurl);
$last_article[5] = stripslashes($acat);
if($articles[0][$akw] != NULL) {
    $articles[$aid][$akw] = $afr;
    $product3 += $afr*$afr;
    $product1 += $articles[0][$akw]*$afr;
}
}

}

if($identical != NULL)

```

```

        rsort($identical);
        if($similar != NULL)
            rsort($similar);

        $all_articles = sizeof($identical);
        $pages = ceil($all_articles/$arpp);
        $cur_page = $page;
        if($cur_page<0 || $cur_page>$all_articles)
            $cur_page = 0;
    }

    $all_articles = sizeof($identical);
    $pages = ceil($all_articles/$arpp);
    $cur_page = $page;
    if($cur_page<0 || $cur_page>$all_articles)
        $cur_page = 0;

    if($category_sel>0) {
        echo "<div id=\"main_title\">". $identical[0][6] . "</div>";
    }
    echo "<div id=\"pages\">";

    if($cur_page==0) {
        echo "<div id=\"pages_left\">";
        echo _COMPERSSONALPREVIOUS;
        echo "</div>";
    }
    else {
        $back = $cur_page-$arpp;
        if($back <0)
            $back = 0;
        echo "<div id=\"pages_left\"><a href=\"index.php?
option=com_perssonal&task=percategory&page=$back&order=$order&cat_id=$category_sel\">";
        echo _COMPERSSONALPREVIOUS;
        echo "</a></div>";
    }
    if($cur_page>=$all_articles-$arpp) {
        echo "<div id=\"pages_right\">";
        echo _COMPERSSONALNEXT;
        echo "</div>";
    }
    else {
        $next = $cur_page+$arpp;
        if($next > $all_articles)
            $$next = $all_articles-$arpp;
        echo "<div id=\"pages_right\"><a href=\"index.php?
option=com_perssonal&task=percategory&page=$next&order=$order&cat_id=$category_sel\">";
        echo _COMPERSSONALNEXT;
        echo "</a></div>";
    }
    echo "<div id=\"clearer\"></div>";
    echo "</div>";

    $size = sizeof($identical);
    if($all_articles-$cur_page > $arpp)
        $size = $cur_page+$arpp;

    for($i=$cur_page; $i < $size; ++$i) {
        echo "<div id=\"node_\" . $identical[$i][4] . \">";
        echo "<div id=\"news\">";
    echo "<table width=\"100%\" cellpadding=\"0\" cellspacing=\"0\" align=\"left\" border=\"0\">";
        echo "<tr>";
        echo "<td id=\"news_category\">";

```



?&gt;

**Πίνακας 8. Το αρχείο personal.php του component com\_personal στο front-end**

```

<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted access' );

class TimeCounter
{
    var $startTime;
    var $endTime;

    function TimeCounter()
    {
        $this->startTime=0;
        $this->endTime=0;
    }
    function getTimeStamp()
    {
        $timeofday = gettimeofday();
        //RETRIEVE SECONDS AND MICROSECONDS (ONE MILLIONTH OF A SECOND)
        //CONVERT MICROSECONDS TO SECONDS AND ADD TO RETRIEVED SECONDS
        //MULTIPLY BY 1000 TO GET MILLISECONDS
        return 1000*($timeofday['sec'] + ($timeofday['usec'] / 1000000));
    }
    function startCounter()
    {
        $this->startTime=$this->getTimeStamp();
    }
    function stopCounter()
    {
        $this->endTime=$this->getTimeStamp();
    }
    function getElapsedTime()
    {
        //RETURN DIFFERECE IN MILLISECONDS
        return number_format(($this->endTime)-($this->startTime), 2);
    }
}

class personal_HTML {
    function show_message($message){
        echo $message;
    }

    function find_time($insert_date) {
        $dateexp = explode(" ",$insert_date);
        $time = $dateexp[1];
        $time_min = explode(":", $time);
        return $time_min[0].".". $time_min[1];
    }

    function find_date($insert_date) {
        $dateexp = explode(" ",$insert_date);
        $date = $dateexp[0];
        $date_an = explode("-", $date);
        $year = date("y",mktime(0,0,0,$date_an[1],$date_an[2], $date_an[0]));
        $month = date("M",mktime(0,0,0,$date_an[1],$date_an[2], $date_an[0]));
        $day = date("j",mktime(0,0,0,$date_an[2],$date_an[2], $date_an[0]));
        return $day.".". $month." ".$year;
    }
}

```

```
}
?>
```

**Πίνακας 9. Το αρχείο perssonal.html.php στο component com\_perpsonal του front-end**

```
<?php
// Load the Joomla framework
define( '_JEXEC', 1 );
define( 'DS', DIRECTORY_SEPARATOR );
define( 'JPATH_BASE', realpath(dirname(__FILE__).DS.'..'.DS.'..') );

require_once ( JPATH_BASE .DS.'includes'.DS.'defines.php' );
require_once ( JPATH_BASE .DS.'includes'.DS.'framework.php' );

$mainframe =& JFactory::getApplication('site');
$mainframe->initialise();

$session =& JFactory::getSession();
$user = $session->get( 'user' );
?>
<?php
// USE OF DATABASE
$user = &JFactory::getUser();

if(!$user->guest){
    $userquery = "SELECT a.id as user_id, a.app as app FROM user_website as a, #__users as b
WHERE a.email=b.email and a.email="" . $user->email .""";
    $userdb =& JFactory::getDBO();
    $userdb->setQuery($userquery);
    $userrow = $userdb->loadObject();

    if($userrow != null){
        $user_id = $userrow->user_id;
        $ar_id = intval($_GET['ar_id']);

        $db =& JFactory::getDBO();
        $query = "INSERT INTO user_website_reject VALUES ('$user_id','$ar_id')";
        $db->setQuery($query);
        $db->query();

        $weight = -0.3;
        $db =& JFactory::getDBO();
        $query = "SELECT a.kw_id, a.rel_frequency FROM extraction_kw2ar a WHERE
a.ar_id=$ar_id ORDER BY a.rel_frequency DESC LIMIT 0,5";
        $db->setQuery($query);
        $rows = $db->loadObjectList();

        foreach($rows as $haskw){
            list($thekw, $kwfr) = $haskw;
            //echo "Prosimo : $prosimo ... Varos : $weight ... Frequency : $kwfr <br>";
            $kweight = $kwfr*$weight;
            //echo $kweight." <br>";
            $db =& JFactory::getDBO();
            $query_already_in = "SELECT kw_id, rel_frequency FROM
user_website_keyword WHERE user_id=$user_id AND kw_id=$thekw";
            $db->setQuery($query_already_in);
            $db->query();
            $already_in = $db->getNumRows();
```

```

list($kw_id, $rel_frequency) = $db->loadObject();

if($already_in != 0){
    $rel_frequency = $rel_frequency + ($kweight);
    $db =&JFactory::getDBO();
    $query_upd = "UPDATE user_website_keyword SET rel_frequency
= $rel_frequency WHERE user_id=$user AND kw_id=$thekw";
    $db->setQuery($query_upd);
    $db->query();
}
else {
    $rel_frequency = $rel_frequency + ($kweight);
    $db =&JFactory::getDBO();
    $query_ins = "INSERT INTO user_website_keyword VALUES
('$user','$thekw','$kweight')";
    $db->setQuery($query_ins);
    $db->query();
}

} // kleinei to while
}
?>

```

**Πίνακας 10. Το αρχείο reject.php**

```

// JavaScript Document

var xmlhttp_img;
var site_path;
var article_id;

function sequence(scriptpath,id)
{
xmlhttp_img=GetXmlHttpRequestObject_img()
if (xmlhttp_img==null)
{
    alert ("Your browser does not support AJAX!");
    return;
}
site_path = scriptpath;
article_id = id;
var url=scriptpath;
url=url+"reject.php?ar_id="+id;
xmlhttp_img.open("GET",url,true);
xmlhttp_img.onreadystatechange=stateChanged_img;
xmlhttp_img.send(null);
}

function stateChanged_img()
{
    if (xmlhttp_img.readyState==4){
        var spanElement = document.getElementById("node_" + article_id);
        if(spanElement != null) spanElement.parentNode.removeChild(spanElement);
    }
}

function GetXmlHttpRequestObject_img()
{

```

```

var xmlHttp_img=null;
try
{
// Firefox, Opera 8.0+, Safari
xmlHttp_img=new XMLHttpRequest();
}
catch (e)
{
// Internet Explorer
try
{
xmlHttp_img=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
xmlHttp_img=new ActiveXObject("Microsoft.XMLHTTP");
}
}
return xmlHttp_img;
}

function MM_preloadImages() { //v3.0
var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0; i<a.length; i++)
if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image; d.MM_p[j++].src=a[i];}}
}

function MM_swapImgRestore() { //v3.0
var i,x,a=document.MM_sr; for(i=0;a&&i<a.length&&(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
}

function MM_findObj(n, d) { //v4.01
var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
if(!x && d.getElementById) x=d.getElementById(n); return x;
}

function MM_swapImage() { //v3.0
var i,j=0,x,a=MM_swapImage.arguments; document.MM_sr=new Array; for(i=0;i<(a.length-2);i+=3)
if ((x=MM_findObj(a[i]))!=null){document.MM_sr[j++]=x; if(!x.oSrc) x.oSrc=x.src; x.src=a[i+2];}
}

```

Πίνακας 11. Το αρχείο sequence.js

```

<?php
// Load the Joomla framework
define( '_JEXEC', 1 );
define( 'DS', DIRECTORY_SEPARATOR );
define( 'JPATH_BASE', realpath(dirname(__FILE__).DS.'..'.DS.'..') );

require_once ( JPATH_BASE .DS.'includes'.DS.'defines.php' );
require_once ( JPATH_BASE .DS.'includes'.DS.'framework.php' );

$mainframe =& JFactory::getApplication('site');
$mainframe->initialise();

$session =& JFactory::getSession();
$user = $session->get( 'user' );
?>

```

```

<?php
// USE OF DATABASE
$article_id = JRequest::getVar('ar_id','o');
$db =&JFactory::getDBO();
$query = "SELECT url, tag FROM images WHERE ar_id=$article_id LIMIT 1,10";
$db->setQuery($query);
$rows = $db->loadObjectList();

    if($rows != null){
        foreach($rows as $images){
            $image_url = $images->url;
            $image_tag = $images->tag;
            echo "<div><img src=\"\$image_url\" /></div>";
        }
    }
    else{
        echo "";
    }
?>

```

**Πίνακας 12. Το αρχείο get\_images.php**

```

// JavaScript Document

var xmlhttp_img;
var site_path;

function article_images(scriptpath,one)
{
xmlhttp_img=GetXmlHttpRequestObject_img()
if (xmlhttp_img==null)
{
    alert ("Your browser does not support AJAX!");
    return;
}
site_path = scriptpath;
var url=scriptpath;
url=url+"get_images.php?ar_id="+one;
xmlhttp_img.open("GET",url,true);
xmlhttp_img.onreadystatechange=stateChanged_img;
xmlhttp_img.send(null);
}

function stateChanged_img()
{
    if (xmlhttp_img.readyState==4)
    {
        var spanElement = document.getElementById("images");
        spanElement.innerHTML = "";
        try {
            spanElement.innerHTML = xmlhttp_img.responseText;
        }
        catch (e) {
            // IE fails unless we wrap the string in another element.
            var wrappingElement = document.createElement('div');
            wrappingElement.innerHTML = xmlhttp_img.responseText;
            spanElement.appendChild(wrappingElement);
        }
    }
    else {
        var spanElement = document.getElementById("images");

```

```

        spanElement.innerHTML = "";
        try {
            spanElement.innerHTML = '<p align="center"><img src="" + site_path
+ 'images/loading.gif"></p>';
            spanElement.innerHTML += '<p align="center">Loading</p>';
        }
        catch (e) {
            // IE fails unless we wrap the string in another element.
            var wrappingElement = document.createElement('div');
            wrappingElement.innerHTML = '<p align="center"><img src="" +
site_path + 'images/loading.gif"></p>';
            wrappingElement.innerHTML += '<p align="center">Loading</p>';
            spanElement.appendChild(wrappingElement);
        }
    }
}

function GetXmlHttpRequest_img()
{
    var xmlhttp_img=null;
    try
    {
        // Firefox, Opera 8.0+, Safari
        xmlhttp_img=new XMLHttpRequest();
    }
    catch (e)
    {
        // Internet Explorer
        try
        {
            xmlhttp_img=new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch (e)
        {
            xmlhttp_img=new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
    return xmlhttp_img;
}

```

**Πίνακας 13. Το αρχείο images.js**

```

<?php
// no direct access
defined( '_JEXEC' ) or die( 'Restricted access' );
define( 'DS', DIRECTORY_SEPARATOR );

jimport( 'joomla.application.helper' );
jimport('joomla.application.component.view');

$document = JFactory::getDocument();
$document-
>addStyleSheet(JURI::root()).'administrator/components/com_personal/p7ap/p7ap_04.css');
$document->addScript(JURI::root()).'administrator/components/com_personal/p7ap/p7APscripts.js');

$task = JRequest::getVar('task');
$languagetouse = intval(JRequest::getVar('lang','1'));

if($languagetouse == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');

```

```

switch ($task){
    case 'registerToJoomla':
        register_user();
        show_all_user_info();
        break;
    case 'My':
    default:
        show_all_user_info();
        break;
}

function register_user(){

$userid = JRequest::getVar('userid', '-1');

if($userid != "-1"){
    if(intval($userid) > 0){
        $db =& JFactory::getDBO();
        $query = "SELECT * FROM user_website WHERE id=" .intval($userid);
        $db->setQuery($query);
        $result = $db->loadObject();

        if($result != null){

            $db =& JFactory::getDBO();
            $query = "SELECT * FROM #__users WHERE email=" . $result->email ."";
            $db->setQuery($query);
            $resultemail = $db->loadObject();

            if($resultemail == null){
                jimport('joomla.user.helper');
                $salt = JUserHelper::genRandomPassword(32);
                $crypted = JUserHelper::getCryptedPassword($result->email, $salt);
                $password = $crypted.'!'.$salt;

                $db =& JFactory::getDBO();
                $query = "INSERT INTO #__users VALUES (NULL , " . $result->email .", " . $result->email .", " . $result->email .", " . $password .", " , 'o', 'o', now(), now(), " . " )";
                $db->setQuery($query);
                $db->query();

                $id_inserted = (int)$db->insertid();
                $group_id = 2; // 2 is for registered users

                $db =& JFactory::getDBO();
                $query = "INSERT INTO #__user_usergroup_map VALUES (" . $id_inserted .", " . $group_id . " )";
                $db->setQuery($query);
                $db->query();

                echo " <h4>"
                . _ADMINCOMPERSSONALTABMOVETOJOOMLAUSERREGISTERED . "</h4>";
            }
            else
                echo " <h4>"
                . _ADMINCOMPERSSONALTABMOVETOJOOMLAUSERALREADY . "</h4>";
        }
    }

    if($userid == "all"){
        $db =& JFactory::getDBO();
        $query = "SELECT * FROM user_website WHERE email not in (SELECT email from
#_users)";

```

```

$db->setQuery($query);
$result = $db->loadObjectList();

if($result != null){
    jimport('joomla.user.helper');

    foreach($result as $joomla_user){
        $db =& JFactory::getDBO();
        $query = "SELECT * FROM #__users WHERE email=" .
$joomla_user->email ."";

        $db->setQuery($query);
        $resultemail = $db->loadObject();

        if($resultemail == null){
            $salt = JUserHelper::genRandomPassword(32);
            $crypted=
JUserHelper::getCryptedPassword($joomla_user->email, $salt);
            $password = $crypted.'.'.$salt;

            $db =& JFactory::getDBO();
            $query = "INSERT INTO #__users VALUES (NULL , " .
$joomla_user->email .", " . $joomla_user->email .", " . $password .", " , 'o',
'o', now(), now(), " , " );";

            $db->setQuery($query);
            $db->query();

            $id_inserted = (int)$db->insertid();
            $group_id = 2; // 2 is for registered users

            $db =& JFactory::getDBO();
            $query = "INSERT INTO #__user_usergroup_map
VALUES ( " . $id_inserted . " , " . $group_id . " )";
            $db->setQuery($query);
            $db->query();
        }
    }
}

echo "<h4>"
._ADMINCOMPERSSONALTABMOVETOJOOMLAUSERREGISTEREDALL . "</h4>";
}
}
}

function show_all_user_info(){
$language = intval(JRequest::getVar('lang','1'));

// Define both to current
$langHTTP_GR = JURI::current() ."?option=com_perpsonal&lang=16";
$langHTTP_EN = JURI::current() ."?option=com_perpsonal&lang=1";

if($language == 1) $registerUserLink = JURI::current() ."?
option=com_perpsonal&lang=1&task=registerToJoomla&userid=";
else $registerUserLink = JURI::current() ."?
option=com_perpsonal&lang=16&task=registerToJoomla&userid=";

echo "<div id=\"p7ABW1\" class=\"p7AB\">";
echo "<div class=\"p7ABtrig\">";
echo "<h3><a href=\"javascript:\"" id=\"p7ABt1_1\">" .

```

```

_ADMINCOMPERSSONALSETTINGSTAB ."/a></h3>";
echo "</div>";
echo "<div id=\"p7ABw1_1\">";
  echo "<div id=\"p7ABc1_1\" class=\"p7ABcontent\">";
    echo "<br>";
    echo " <table border=\"1\" cellpadding=\"0\" cellspacing=\"0\"
bordercolor=\"#000000\" align=\"center\">";
    echo "<tr>";
    echo "          <td          height=\"30\">"
_ADMINCOMPERSSONALSETTINGSTABLANGUAGE ."/td>";

    if($language == 1) echo "<td height=\"30\"><a href=\"\" . $langHTTP_GR .
\">Greek</a></td>";
    else echo " <td height=\"30\"><a href=\"\" . $langHTTP_EN .
\">Αγγλικά</a></td>";

    echo "</tr>";
    echo "</table><br>";

  echo "</div>";
echo "</div>";
$db =& JFactory::getDBO();
$query = "SELECT * FROM user_website WHERE email not in (SELECT email from #__users)";
$db->setQuery($query);
$result = $db->loadObjectList();

if($result != null){
  echo "<div class=\"p7ABtrig\">";
    echo " <h3><a href=\"javascript:;\" id=\"p7ABt1_2\">"
_ADMINCOMPERSSONALTAB ."/a></h3>";
  echo "</div>";
  echo "<div id=\"p7ABw1_2\">";
    echo "<div id=\"p7ABc1_2\" class=\"p7ABcontent\">";
    echo "<h4> . _ADMINCOMPERSSONALTABHEADER ."/h4><br>";
    echo "<p> . _ADMINCOMPERSSONALTABHELP ."/p><br>";
    echo " <table border=\"1\" cellpadding=\"0\" cellspacing=\"0\"
bordercolor=\"#000000\" align=\"center\">";
    echo "<tr>";
    echo " <td height=\"30\" bgcolor=\"#66CCFF\">"
_ADMINCOMPERSSONALTABEMAIL ."/td>";
    echo " <td height=\"30\" bgcolor=\"#66CCFF\">"
_ADMINCOMPERSSONALTABREGISTERDATE ."/td>";
    echo " <td height=\"30\" bgcolor=\"#66CCFF\"></td>";
    echo "</tr>";
    foreach($result as $joomla_user){
      echo "<tr>";
      echo " <td height=\"30\"> . $joomla_user->email ."/td>";
      echo " <td height=\"30\"> . $joomla_user-
>rec_date ."/td>";
      $finalUserLink = $registerUserLink . $joomla_user-
>id ."#pab1_2";
      echo " <td height=\"30\" bgcolor=\"#66CCFF\"><a href=\"\"
. $finalUserLink . "\"> . _ADMINCOMPERSSONALTABMOVETOJOOMLA ."/a></td>";
      echo "</tr>";
    }
    echo "</table><br>";

    echo " <table border=\"1\" cellpadding=\"0\" cellspacing=\"0\"
bordercolor=\"#000000\" align=\"center\">";
    echo "<tr>";
    $finalUserLink = $registerUserLink . "all" ."#pab1_2";
    echo " <td height=\"30\" bgcolor=\"#66CCFF\"><a href=\"\" .
$finalUserLink . "\"> . _ADMINCOMPERSSONALTABMOVEALLTOJOOMLA ."/a></td>";
    echo "</tr>";
    echo "</table><br>";

```

```

        echo "</div>";
        echo "</div>";
    }

    echo "<script type=\"text/javascript\">";
    echo "P7_opAB(1,1,1,1,0);";
    echo "</script>";
    echo "</div>";
}
?>

```

**Πίνακας 14. Το αρχείο perssonal.php του back-end**

```

<?xml version="1.0" encoding="utf-8"?>
<extension type="component" version="1.7" method="upgrade">
    <name>Perssonal</name>
    <author>Mpourelou Panagiotis</author>
    <creationDate>01/01/2012</creationDate>
    <copyright>(C) 2012 CTI. All rights reserved.</copyright>
    <license>CTI</license>
    <authorEmail>mpourelo@gmail.com</authorEmail>
    <authorUrl>www.cti.gr</authorUrl>
    <version>1.7.0</version>
    <description>Perssonal is the best RSS feed component for Joomla</description>
    <files folder="site">
        <filename>images.js</filename>
        <filename>get_images.php</filename>
        <filename>reject.php</filename>
        <filename>sequence.js</filename>
        <filename>perssonal.php</filename>
        <filename>perssonal.html.php</filename>
        <filename>css/mycss.css</filename>
        <filename>images/loading.gif</filename>
        <filename>images/reject.png</filename>
        <filename>images/reject_on.png</filename>
        <filename>lang/el-GR.php</filename>
        <filename>lang/en-GB.php</filename>
        <folder>css</folder>
        <folder>images</folder>
    </files>
    <administration>
        <files folder="admin">
            <filename>perssonal.php</filename>
            <filename>perssonal.html.php</filename>
            <filename>lang/el-GR.php</filename>
            <filename>lang/en-GB.php</filename>
            <filename>p7ap/p7ap_04.css</filename>
            <filename>p7ap/p7APscripts.js</filename>
            <filename>p7ap/images/p7ap_bluestonebut.jpg</filename>
            <filename>p7ap/images/p7ap_east_black.gif</filename>
            <filename>p7ap/images/p7ap_east_light_gray.gif</filename>
            <filename>p7ap/images/p7ap_east_med_gray.gif</filename>
            <filename>p7ap/images/p7ap_east_red.gif</filename>
            <filename>p7ap/images/p7ap_east_white.gif</filename>
            <filename>p7ap/images/p7ap_light_gray_bevel.jpg</filename>
            <filename>p7ap/images/p7ap_south_black.gif</filename>
            <filename>p7ap/images/p7ap_south_light_gray.gif</filename>
            <filename>p7ap/images/p7ap_south_med_gray.gif</filename>
            <filename>p7ap/images/p7ap_south_red.gif</filename>
            <filename>p7ap/images/p7ap_south_white.gif</filename>
        </folder>lang</folder>
    </administration>
</extension>

```

```

        <folder>p7ap</folder>
        </files>
    </administration>
</extension>

```

**Πίνακας 15. Το αρχείο εγκατάστασης του componentn com\_personal**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_personalcategories
 * @copyright        Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license          Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

// Include the syndicate functions only once
require_once dirname(__FILE__).'/helper.php';

// Get the user data
$list = mod_PerssonalCategories::getData($params);
$categorieslang = mod_PerssonalCategories::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_personalcategories', $params->get('layout', 'default'));
?>

```

**Πίνακας 16. Το αρχείο mod\_personalcategories.php**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_PerssonalCategories
 * @copyright        Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license          Computer Technology Institute - University of Patras
 */

// Class is used by module personalcategories to view all categories the user is interested
// Function getData is used to retrieve all categories from PERSSONAL user logged in.
defined('_JEXEC') or die;

class mod_PerssonalCategories
{
    function getLang()
    {
        // Get user information
        $user =& JFactory::getUser();

        if(!$user->guest){
            $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email=" . $user->email ."";
            $userdb =& JFactory::getDBO();
            $userdb->setQuery($userquery);
            $userrow = $userdb->loadObject();

            if($userrow != null){

```

```

        return intval($userrow->preflang);
    }

}

return 1;
}

function getData( &$params )
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id,a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='". $user->email ."'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // Database query
            $list = array();
            $query = "SELECT a.cat_id as category_id, b.cat_title as category_title FROM
user_website_category a LEFT JOIN category b ON a.cat_id=b.cat_id WHERE a.user_id='". $userrow-
>user_id ." AND a.preference>=0";
            $db =& JFactory::getDBO();
            $db->setQuery($query);
            $rows = $db->loadObjectList();

            // Get list items
            if ($rows!=null)
            {
                $i=0;
                foreach ($rows as $row)
                {
                    $list["categories"][$i]["catid"]=$row->category_id;
                    $list["categories"][$i]["cattitle"]=$row->category_title;
                    $i++;
                }
                return $list;
            }
            else return null;
        }
        else return null;
    }
    else return null;
}
}
?>

```

Πίνακας 17. Το αρχείο helper.php

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_personalcategories
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras

```



```

<authorUrl>http://www.cti.gr</authorUrl>
<version>4.0</version>
<description>The is a module to show peRSSonal user info</description>
<languages>
</languages>
<files>
<filename module="mod_perssonalcategories">mod_perssonalcategories.php</filename>
<filename>mod_perssonalcategories.xml</filename>
<filename>helper.php</filename>
<filename>index.html</filename>
<filename>index.html</filename>
<filename>tmpl/index.html</filename>
<filename>tmpl/default.php</filename>
<filename>tmpl/lang/el-GR.php</filename>
<filename>tmpl/lang/en-GB.php</filename>
<filename>tmpl/images/menu.jpg</filename>
<folder>tmpl</folder>
</files>
</extension>

```

**Πίνακας 19. Το αρχείο εγκατάστασης mod\_perssonalcategories.xml**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalLatestNews
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

// Include the syndicate functions only once
require_once dirname(__FILE__).'/helper.php';

// Get the user data
$list = modPerssonalLatestNews::getData($params);
$latestnewslang = modPerssonalLatestNews::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_perssonallatestnews', $params->get('layout', 'default'));
?>

```

**Πίνακας 20. Το αρχείο mod\_perssonallatestnews.php**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalLatestNews
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// Class is used by module mod_PerssonalLatestNews to view all latest news of PERSSONAL user logged
in.
defined('_JEXEC') or die;

class modPerssonalLatestNews
{

```

```

function getLang()
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='". $user->email ."'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            return intval($userrow->preflang);
        }
    }

    return 1;
}

function getData( &$params )
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as userlang FROM user_website
as a, #__users as b WHERE a.email=b.email and a.email='". $user->email ."'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // Database query
            $list = array();
            $language = $userrow->userlang;
            $query = "SELECT ar_id, ar_title, ar_date FROM articles WHERE trash=0 AND
ar_lang='". $language . "' AND cat_id NOT IN (SELECT cat_id FROM category WHERE sports=1) AND
ar_summary is not null ORDER BY ar_date DESC LIMIT 0,10";
            $db =& JFactory::getDBO();
            $db->setQuery($query);
            $rows = $db->loadObjectList();

            // Get list items
            if ($rows!=null)
            {
                $i=0;
                foreach ($rows as $row)
                {
                    $list["latestnews"][$i]["ar_id"]=$row->ar_id;
                    $list["latestnews"][$i]["ar_title"]=$row->ar_title;
                    $list["latestnews"][$i]["ar_date"]=$row->ar_date;
                    $i++;
                }
                return $list;
            }
            else return null;
        }
        else return null;
    }
    else return null;
}

```

```
}
}
?>
```

### Πίνακας 21. Το αρχείο helper.php

```
<?php
/**
 * @package      Joomla.Site
 * @subpackage   mod_PerssonalLatestNews
 * @copyright    Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license      Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;
$user =& JFactory::getUser();
$document = JFactory::getDocument();
$document->addStyleSheet(JURI::root().'/modules/mod_perssonallatestnews/tmpl/css/mystyle.css');

function find_time($insert_date) {
    $dateexp = explode(" ", $insert_date);
    $time = $dateexp[1];
    $time_min = explode(":", $time);
    return $time_min[0].".".$time_min[1];
}

function find_date($insert_date) {
    $dateexp = explode(" ", $insert_date);
    $date = $dateexp[0];
    $date_an = explode("-", $date);
    $year = date("y", mktime(0,0,0,$date_an[1],$date_an[2], $date_an[0]));
    $month = date("M", mktime(0,0,0,$date_an[1],$date_an[2], $date_an[0]));
    $day = date("j", mktime(0,0,0,$date_an[1],$date_an[2], $date_an[0]));
    return $day." ".$month." ".$year;
}

if($latestnewslang == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');
?>

<table width="100%" border="2" cellspacing="0" cellpadding="0">
<tr><td height="30" background="<?php echo
JURI::root().'/modules/mod_perssonallatestnews/tmpl/images/menu.jpg'; ?>"><?php echo
_MODPERSSONALLATESTNEWSNAME; ?></td></tr>
<tr><td>
<hr size=3 width=100% align="left" color="grey">
<?php if(!$user->guest){ ?>
<ul>
    <?php
        if($list != null){
            for ($i=0;$i< sizeof($list["latestnews"]); $i++) {
                $ar_date = $list["latestnews"][$i]["ar_date"];
                $ar_time = $list["latestnews"][$i]["ar_time"];
                $ar_id = $list["latestnews"][$i]["ar_id"];
                $ar_title = $list["latestnews"][$i]["ar_title"];
                $time = find_time($ar_date);
                $date = find_date($ar_date);

                ?>

                <div class="items_present"><div class="items_date"><?php echo $date." ".$time; ?
```



```

$list = mod_PerssonalPopularArticles::getData($params);
$populararticleslang = mod_PerssonalPopularArticles::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_PerssonalPopularArticles', $params->get('layout',
'default'));
?>

```

#### Πίνακας 24. Το αρχείο mod\_perssonalpopulararticles.php

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalPopularArticles
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// Class is used by module perssonalcategories to view all categories the user is interested
// Function getData is used to retrieve all categories PERSSONAL user is logged in to Joomla
defined('_JEXEC') or die;

class mod_PerssonalPopularArticles
{
    function getLang()
    {
        // Get user information
        $user =& JFactory::getUser();

        if(!$user->guest){
            $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='" . $user->email ."'";
            $userdb =& JFactory::getDBO();
            $userdb->setQuery($userquery);
            $userrow = $userdb->loadObject();

            if($userrow != null){
                return intval($userrow->preflang);
            }
        }

        return 1;
    }

    function getData( &$params )
    {
        // Get user information
        $user =& JFactory::getUser();

        if(!$user->guest){
            $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='" . $user->email ."'";
            $userdb =& JFactory::getDBO();
            $userdb->setQuery($userquery);
            $userrow = $userdb->loadObject();

            if($userrow != null){
                // Database query
                $list = array();
            }
        }
    }
}

```

```

$language = $userrow->preflang;
$task = JRequest::getVar('task');

if($task == 'myarticles'){
    $month = date("m");
    $day = date("d");
    $year = date("Y");
    $last3days = date("Y-m-d H:i:s", mktime(0, 0, 0, $month, $day-5,
$year));

    $query = "SELECT a.ar_id, a.ar_title, a.ar_date, b.ar_counter
FROM articles a LEFT JOIN articles_counter b ON a.ar_id=b.ar_id WHERE ar_date>=\"$last3days\"
AND a.ar_lang=$language AND a.trash=0 AND a.cat_id IN (SELECT cat_id FROM category WHERE
sports=0 AND trash=0) ORDER BY b.ar_counter DESC, ar_date DESC LIMIT 0,5";
    $db =& JFactory::getDBO();
    $db->setQuery($query);
    $rows = $db->loadObjectList();

    // Get list items
    if ($rows!=null)
    {
        $i=0;
        foreach ($rows as $row)
        {
            $list["popularnews"][$i]["ar_id"]=$row->ar_id;
            $list["popularnews"][$i]["ar_title"]=$row->ar_title;
            $list["popularnews"][$i]["ar_date"]=$row->ar_date;
            $i++;
        }
        return $list;
    }
    else return null;
}
else return null;
}
else return null;
}
else return null;
}
}
?>

```

### Πίνακας 25. Το αρχείο helper.php

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalPopularArticles
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;
$user =& JFactory::getUser();
$document = JFactory::getDocument();
$document-
>addStyleSheet(JURI::root().modules/mod_perssonalpopulararticles/tmpl/css/mystyle.css');

if($populararticleslang == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');

```

```

else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');
?>

<table width="100%" border="2" cellspacing="0" cellpadding="0">
<tr><td
            height="30"
            background="<?php
JURI::root().'modules/mod_perSSonalpopulararticles/tmpl/images/menu.jpg'; ?>"><?php
echo
echo
            _MODPERSSONALPOPULARARTICLESNAME; ?></td></tr>
<tr><td>
<hr size=3 width=100% align="left" color="grey">
<?php if(!$user->guest){ ?>
<ul>
    <?php
        if($list != null){
            for ($i=0;$i< sizeof($list["popularnews"]); $i++) {
                $ar_date = $list["popularnews"][$i]["ar_date"];
                $ar_id = $list["popularnews"][$i]["ar_id"];
                $ar_title = $list["popularnews"][$i]["ar_title"];
                $time = find_time($ar_date);
                $date = find_date($ar_date);

                ?>

                <div class="items_present"><div class="items_date"><?
php echo $date." ". $time; ?></div><div class="items_item"><a href="index.php?
option=com_perSSonal&task=readarticle&ar_id=<?php echo $ar_id; ?>"><?php echo
stripslashes($ar_title); ?></a></div><div class="items_item_clear"></div></div>

                <?php
                    }
                }
            else echo "&nbsp;";
        }
    ?>
</ul>
<?php
}
?>
</td></tr></table>

```

**Πίνακας 26. Το αρχείο default.php**

```

<?xml version="1.0" encoding="UTF-8"?>
<extension type="module" version="1.7" client="site" method="upgrade">
<name>peRSSonal Popular Articles</name>
<author>Mpourelou Panagiotis</author>
<creationDate>31/01/2012</creationDate>
<copyright>Copyright (C) 2012. All rights reserved.</copyright>
<license>http://www.cti.gr</license>
<authorEmail>mpourelo@gmail.com</authorEmail>
<authorUrl>http://www.cti.gr</authorUrl>
<version>4.0</version>
<description>The is a module to show peRSSonal popular articles when user chooses to see his own
articles</description>
<files>
<filename module="mod_perSSonalpopulararticles">mod_perSSonalpopulararticles.php</filename>
<filename>mod_perSSonalpopulararticles.xml</filename>
<filename>helper.php</filename>
<filename>index.html</filename>
<filename>tmpl/default.php</filename>
<filename>tmpl/index.html</filename>
<filename>tmpl/css/mystyle.css</filename>
<filename>tmpl/lang/el-GR.php</filename>

```

```

<filename>tmpl/lang/en-GB.php</filename>
<filename>tmpl/images/menu.jpg</filename>
<folder>tmpl</folder>
</files>
</extension>

```

**Πίνακας 27. Το αρχείο εγκατάστασης mod\_perssonalpopulararticles.xml**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_PerssonalReadersPick
 * @copyright         Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license           Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

// Include the syndicate functions only once
require_once dirname(__FILE__).'/helper.php';

// Get the user data
$list = mod_PerssonalReadersPick::getData($params);
$readerspicklang = mod_PerssonalReadersPick::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_PerssonalReadersPick', $params->get('layout', 'default'));
?>

```

**Πίνακας 28. Το αρχείο mod\_perssonalreaderspick.php**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_PerssonalReadersPick
 * @copyright         Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license           Computer Technology Institute - University of Patras
 */

defined('_JEXEC') or die;

class mod_PerssonalReadersPick
{
    function getLang()
    {
        // Get user information
        $user =& JFactory::getUser();

        if(!$user->guest){
            $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='".$user->email."'";
            $userdb =& JFactory::getDBO();
            $userdb->setQuery($userquery);
            $userrow = $userdb->loadObject();

            if($userrow != null){
                return intval($userrow->preflang);
            }
        }
    }
}

```

```

    }

    }

    return 1;
}

function getData( &$params )
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='". $user->email ."'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // Database query
            $list = array();
            $language = $userrow->preflang;

            $lastdate = date("Y-m-d H:i:s", mktime(date("H"), date("i"), date("s"),
date("m"), date("d")-5, date("Y")));
            $query_read = "SELECT sum(b.total_time) as TOTAL, a.ar_id, a.ar_title,
a.ar_date FROM user_website_reading b LEFT JOIN articles a on a.ar_id=b.ar_id WHERE
a.ar_date>=\"$lastdate\" AND a.ar_lang=$language GROUP BY a.ar_id ORDER BY sum(b.total_time)
DESC LIMIT 0,5";

            $db =& JFactory::getDBO();
            $db->setQuery($query_read);
            $rows = $db->loadObjectList();

            // Get list items
            if ($rows!=null)
            {
                $i=0;
                foreach ($rows as $row)
                {
                    $list["readerpick"][$i]["ar_id"]=$row->ar_id;
                    $list["readerpick"][$i]["ar_title"]=$row->ar_title;
                    $list["readerpick"][$i]["ar_date"]=$row->ar_date;
                    $list["readerpick"][$i]["ar_time"]=$row->TOTAL/1000;

                    $i++;
                }
            }
            else return null;
        }
        else return null;
    }
    else return null;
}
}
?>

```

**Πίνακας 29. Το αρχείο helper.php**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalReadersPick
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;
$user =& JFactory::getUser();
$document = JFactory::getDocument();
$document->addStyleSheet(JURI::root().'modules/mod_perssonalreaderspicks/tmpl/css/mystyle.css');

if($readerspickslang == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');
?>

<table width="100%" border="2" cellspacing="0" cellpadding="0">
<tr><td height="30" background="<?php
JURI::root().'modules/mod_perssonalreaderspicks/tmpl/images/menu.jpg'; ?>"><?php echo
_MODPERSONALREADERSPICKNAME; ?></td></tr>
<tr><td>
<hr size=3 width=100% align="left" color="grey">
<?php if(!$user->guest){ ?>
<ul>
<?php
    if($list != null){
        for ($i=0;$i< sizeof($list["readerpick"]); $i++) {
            $ar_date = $list["readerpick"][$i]["ar_date"];
            $ar_id = $list["readerpick"][$i]["ar_id"];
            $ar_title = $list["readerpick"][$i]["ar_title"];
            $time = find_time($ar_date);
            $date = find_date($ar_date);

            ?>

            <div class="items_present"><div class="items_date"><?
php echo $date." ".$time; ?></div><div class="items_item"><a href="index.php?
option=com_perssonal&task=readarticle&ar_id=<?php echo $ar_id; ?>"><?php echo
stripslashes($ar_title); ?></a></div><div class="items_item_clear"></div></div>

            <?php
                }
            }
        else echo "&nbsp;";
    }
?>
</ul>
<?php
}
?>
</td></tr></table>

```

**Πίνακας 30. Το αρχείο default.php**

```

<?xml version="1.0" encoding="UTF-8"?>
<extension type="module" version="1.7" client="site" method="upgrade">
<name>peRSSonal Readers Pick</name>
<author>Mpourellos Panagiotis</author>
<creationDate>31/01/2012</creationDate>
<copyright>Copyright (C) 2012. All rights reserved.</copyright>

```

```

<license>http://www.cti.gr</license>
<authorEmail>mpourel@gmail.com</authorEmail>
<authorUrl>http://www.cti.gr</authorUrl>
<version>4.0</version>
<description>The is a module to show perSSonal readers pick on his own articles</description>
<files>
<filename module="mod_perSSonalreaderspicks">mod_perSSonalreaderspicks.php</filename>
<filename>mod_perSSonalreaderspicks.xml</filename>
<filename>helper.php</filename>
<filename>index.html</filename>
<filename>tmpl/default.php</filename>
<filename>tmpl/index.html</filename>
<filename>tmpl/css/mystyle.css</filename>
<filename>tmpl/lang/el-GR.php</filename>
<filename>tmpl/lang/en-GB.php</filename>
<filename>tmpl/images/menu.jpg</filename>
<folder>tmpl</folder>
</files>
</extension>

```

**Πίνακας 31. Το αρχείο εγκατάστασης mod\_perSSonalreaderspicks.xml**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerSSonalSimilarArticles
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

// Include the syndicate functions only once
require_once dirname(__FILE__).'/helper.php';

// Get the user data
$list = mod_PerSSonalSimilarArticles::getData($params);
$similararticleslang = mod_PerSSonalSimilarArticles::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_PerSSonalSimilarArticles', $params->get('layout', 'default'));
?>

```

**Πίνακας 32. Το αρχείο mod\_perSSonalsimilararticles.php**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerSSonalSimilarArticles
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

defined('_JEXEC') or die;

class mod_PerSSonalSimilarArticles
{

```

```

function getLang()
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='" . $user->email . "'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            return intval($userrow->preflang);
        }
    }

    return 1;
}

function getData( &$params )
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='" . $user->email . "'";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // Database query
            $list = array();
            $language = $userrow->preflang;
            $task = JRequest::getVar('task');

            if($task == 'readarticle'){
                $ar_id = JRequest::getVar('ar_id', '-1');
                $type = "html";
                $limit = 10;
                // echo $type;

                // to be changed...
                // apo tin wra tou arhtrou +-8
                // kai to +8 kai to -8...

                // edw prepei na diavasw ola ta stoixeia toy prwttoy arhtrou...
                // tha diavasw ola ta keywords kai tha ta apothikefsw se enan
disdiastato pinaka

                // a[article id][keyword id] = frequency
                $articles[0][0] = 0;

                // I have three products...
                // (arithmitis)product1, (paronomasths arxikou)product2,
(paronomasths telikoy)product3

                //
                $product1 = 0;
                $product2 = 0;
                $product3 = 0;

                //echo $current_article_query;

```

```

// hmeromhnies apo palia ekdosh...
/*$current_date_result = mysql_query($current_date_query);
$thedata = mysql_fetch_row($current_date_result);
list($article_date) = $thedata;
$article_date = explode(" ", $article_date);
$article_thedata = explode("-", $article_date[0]);
$article_timing = explode(":", $article_date[1]);*/
$similarity;

// edw kratame ta arthra...
$idetical;
$countident = 0;
$related;
$countrel = 0;
$similar;
$countsim = 0;

$thedata = 0;
$count = 0;

$currentarticledb =& JFactory::getDBO();
$current_article_query = "SELECT a.ar_id, a.ar_title, a.ar_date,
b.kw_id, b.abs_frequency FROM articles a LEFT JOIN extraction_kw2ar b ON a.ar_id=b.ar_id
WHERE a.ar_id=$ar_id ORDER BY b.abs_frequency DESC";
$currentarticledb->setQuery($current_article_query);
$currentarticledb->query();
$total_first_article = $currentarticledb->getNumRows();
$rows = $currentarticledb->loadObjectList();

// edw apothikevoume ston pinaka ta stoixeia gia to prwto arthro...
$keep = floor(0.2*$total_first_article);
if($keep<10) {
    $keep = 10;
}
if($keep>20) {
    $keep = 20;
}
//$keep = 10;
//echo $keep;
foreach($rows as $has_article){
    if($count == $keep) {
        break;
    }
    $count++;

    // list($aid, $atitle, $adate, $akw, $afr) = $has_article;
    $articles[$has_article->aid][$has_article->akw] =
$has_article->afr;

    // molis kleisei afto tha exw to product2
    $product2 += $has_article->afr*$has_article->afr;
    $thedata = $has_article->adate;
}

$count = 0;

$article_date = explode(" ", $thedata);
$article_thedata = explode("-", $article_date[0]);
$article_timing = explode(":", $article_date[1]);
$product2 = sqrt($product2);

//echo "<br>";
//echo $product2;
//echo "<br>";

```

```

/*
    HMEROMHNIES APO PALIA EKDOSH
*/
    $lastdate = date("Y-m-d H:i:s", mktime($article_timing[0],
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2]-3, $article_thedate[0]));
    $firstdate = date("Y-m-d H:i:s", mktime($article_timing[0],
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2]+3,
$article_thedate[0]));

    $identical_last = date("Y-m-d H:i:s", mktime($article_timing[0]-8,
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2], $article_thedate[0]));
    $identical_first = date("Y-m-d H:i:s",
mktime($article_timing[0]+8, $article_timing[1], $article_timing[2], $article_thedate[1],
$article_thedate[2], $article_thedate[0]));

    $similar_last1 = date("Y-m-d H:i:s", mktime($article_timing[0]-8,
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2], $article_thedate[0]));
    $similar_last2 = date("Y-m-d H:i:s", mktime($article_timing[0]-24,
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2], $article_thedate[0]));
    $similar_first1 = date("Y-m-d H:i:s", mktime($article_timing[0]+8,
$article_timing[1], $article_timing[2], $article_thedate[1], $article_thedate[2], $article_thedate[0]));
    $similar_first2 = date("Y-m-d H:i:s",
mktime($article_timing[0]+24, $article_timing[1], $article_timing[2], $article_thedate[1],
$article_thedate[2], $article_thedate[0]));

    /*echo " >= ".$identical_last."<br>";
echo " <= ".$identical_first."<br>";
echo " < ".$similar_last1." ++ ";
echo " >= ".$similar_last2."<br>";
echo " > ".$similar_first1." ++ ";
echo " <= ".$similar_first2."<br>"; */
    //echo $lastdate;
    //echo $firstdate;
    //$lastdate2 = date("Y-m-d H:i:s", mktime(date("H"), date("i"),
date("s"), date("m"), date("d")-3, date("Y")));
    //$thewhere = similarity_check2($ar_id);
    // FIND ALL ABOUT ARTICLES...

    $aboutdb =& JFactory::getDBO();
    $query1 = "SELECT a.ar_id, a.ar_title, a.ar_date, b.kw_id,
b.abs_frequency FROM articles a LEFT JOIN extraction_kw2ar b ON a.ar_id=b.ar_id WHERE
a.ar_date>=\"$lastdate\" AND a.ar_date<=\"$firstdate\" AND a.ar_id<>$ar_id ORDER BY a.ar_date
DESC, b.abs_frequency DESC";
    $aboutdb->setQuery($query1);
    $rows = $aboutdb->loadObjectList();

    $countthem = 0;
    $last_article[] = 0;

    return $countthem;

    // prwth ektelesh ekτος while...
    $first_result = $rows[0];

    list($aid, $atitle, $adate, $akw, $afr) = $first_result;
    $last_article[0] = $aid;
    $last_article[1] = $atitle;
    $last_article[2] = $adate;
    if($articles[$ar_id][$akw] != NULL) {
        //echo "common keyword in article $aid";
        $articles[$aid][$akw] = $afr;
        $product3 += $afr*$afr;
        $product1 += $articles[$ar_id][$akw]*$afr;
    }

```

```

    }
    $count=0;
    for($i=1;$i < sizeof($rows);$i++) {
        list($aid, $atitle, $adate, $akw, $afr) = $rows[$i];

        if($last_article[0] == $aid){
            /* if($count == $keep) {
                continue;
            }
            $count++; */
            if($articles[$ar_id][$akw] != NULL) {
                //echo "$count and $keep - common
keyword $akw in article $aid<br><br>";
                $articles[$aid][$akw] = $afr;
                $product3 += $afr*$afr;
                $product1 += $articles[$ar_id]
[$akw]*$afr;
            }
        }
        else{
            $count = 0;
            // vriskoume tin prohgomenh sixxetisi...
            $product3 = sqrt($product3);
            if($product2 == 0 || $product3 == 0){
                $similarity[$last_article[0]][0] = 0;
            }
            else{
                $similarity[$last_article[0]][0] =
($product1)/($product2*$product3);
            }
            if($similarity[$last_article[0]][0] >
0.75){
                ++$countthem;
                // prepei na ginei check h
hmeromhnia...
                //if( ($last_article[2] >=
$identical_last && $last_article[2] <= $identical_first && $similarity[$last_article[0]][0] > 0.85) ||
$similarity[$last_article[0]][0] > 0.95) {
                    $identical[$countident][0] = number_format($similarity[$last_article[0]][0]*100, 1, ',', '.');
                    $identical[$countident][1] = $last_article[1];
                    $identical[$countident][2] = $last_article[2];
                    $identical[$countident][3] = $last_article[0];
                    $countident++;
                }
                /*}
                else if( (($last_article[2] >=
$similar_last2 && $last_article[2] < $similar_last1) || ($last_article[2] > $similar_first1 &&
$last_article[2] <= $similar_first2)) && $similarity[$last_article[0]][0] > 0.80) {
                    $similar[$countsim]
[0] = number_format($similarity[$last_article[0]][0]*100, 1, ',', '.');
                    $similar[$countsim]
[1] = $last_article[1];
                    $similar[$countsim]
[2] = $last_article[2];
                    $similar[$countsim]
[3] = $last_article[0];
                    $countsim++;
                }
            }
            else {
                $related[$countrel][0]

```

```

= number_format($similarity[$last_article[0]][0]*100, 1, ',', '.');
= $last_article[1];
= $last_article[2];
= $last_article[0];

$related[$countrel][1]
$related[$countrel][2]
$related[$countrel][3]

$countrel++;
}*/
}
$product1=0;
$product3=0;
}

// nea arxh...
$last_article[0] = $aid;
$last_article[1] = $atitle;
$last_article[2] = $adate;
if($articles[$ar_id][$akw] != NULL) {
    //echo "common keyword in article
$aid";
    $articles[$aid][$akw] = $afrr;
    $product3 += $afrr*$afrr;
    $product1 += $articles[$ar_id]
[$akw]*$afrr;
}
}
} // End of for

if($identical != NULL)
    rsort($identical);

if($similar != NULL)
    rsort($similar);

if($related != NULL)
    rsort($related);

//$the_xml = "";
$the_xml = "<?xml version='1.0' encoding='UTF-8'?'>";
$the_xml .= "<items>";

if($countident == 0){
    $the_xml .= "";
}
else{
    $the_xml .= "<identical>";
    for($i=0; $i < sizeof($identical); ++$i) {
        if($i == $limit)
            break;
        $the_xml .= "<item>";
        $the_xml .= "<id>";
        $the_xml .= $identical[$i][3];
        $the_xml .= "</id>";
        $the_xml .= "<title>";
        $the_xml .= strip_tags($identical[$i][1]);
        $the_xml .= "</title>";
        $the_xml .= "<date>";
        $the_xml .= $identical[$i][2];
        $the_xml .= "</date>";
        $the_xml .= "<formatted_date>";
        $the_xml .= find_date($identical[$i][2]);
    }
}

```

```

        $the_xml .= "</formatted_date>";
        $the_xml .= "<relevance>";
        $the_xml .= $identical[$i][0];
        $the_xml .= "</relevance>";
        $the_xml .= "<url>";
        $the_xml      .=      "index.php?
option=com_perpersonal&task=readarticle&ar_id=".$identical[$i][3];
        $the_xml .= "</url>";
        $the_xml .= "</item>";

    } // End of for

    $the_xml .= "</identical>";
}

if($countsim == 0){
    $the_xml .= "";
}
else{
    $the_xml .= "<similar>";
    for($i=0; $i < sizeof($similar); ++$i) {
        if($i == $limit)
            break;
        $the_xml .= "<item>";
        $the_xml .= "<id>";
        $the_xml .= $similar[$i][3];
        $the_xml .= "</id>";
        $the_xml .= "<title>";
        $the_xml .= $similar[$i][1];
        $the_xml .= "</title>";
        $the_xml .= "<date>";
        $the_xml .= $similar[$i][2];
        $the_xml .= "</date>";
        $the_xml .= "<formatted_date>";
        $the_xml .= find_date($similar[$i][2]);
        $the_xml .= "</formatted_date>";

        $the_xml .= "<relevance>";
        $the_xml .= $similar[$i][0];
        $the_xml .= "</relevance>";
        $the_xml .= "<url>";
        $the_xml      .=      "index.php?
option=com_perpersonal&task=readarticle&ar_id=".$similar[$i][3];
        $the_xml .= "</url>";
        $the_xml .= "</item>";

    } // End of for

    $the_xml .= "</similar>";
}

if($countrel == 0){
    $the_xml .= "";
}
else {
    $the_xml .= "<related>";
    for($i=0; $i < sizeof($related); ++$i) {
        if($i == $limit)
            break;
        $the_xml .= "<item>";
        $the_xml .= "<id>";
        $the_xml .= $related[$i][3];
        $the_xml .= "</id>";
        $the_xml .= "<title>";

```

```

        $the_xml .= $related[$i][1];
        $the_xml .= "</title>";
        $the_xml .= "<date>";
        $the_xml .= $related[$i][2];
        $the_xml .= "</date>";
        $the_xml .= "<formatted_date>";
        $the_xml .= find_date($related[$i][2]);
        $the_xml .= "</formatted_date>";
        $the_xml .= "<relevance>";
        $the_xml .= $related[$i][0];
        $the_xml .= "</relevance>";
        $the_xml .= "<url>";
        $the_xml .= "index.php?
option=com_personal&task=readarticle&ar_id=".$related[$i][3];
        $the_xml .= "</url>";

        $the_xml .= "</item>";

    } // End of for

    $the_xml .= "</related>";
}

$the_xml = ltrim($the_xml);

/* if ($type == "html") {
    $xsl_filename = "identical.xsl";

    $doc = new
    $xsl = new

    $doc-
    $xsl-
    $doc-
    echo $xsl-

}*/
$the_xml.="</items>\n";
$the_xml = str_replace("&", "&amp;", $the_xml);

if($type == "html"){
    //echo $the_xml;

    $xsl_filename =
    JURI::root()."components/com_personal/identical.xsl";
    //$xml_filename = "projects.xml";

    $doc = new DOMDocument();
    $xsl = new XSLTProcessor();

    $doc->load($xsl_filename);
    $xsl->importStyleSheet($doc);

    $doc->loadXML($the_xml);
    return $xsl->transformToXML($doc);
}
else {
    return $the_xml;
}

```

```

    }
    }
    else return null;
}
else return null;
}
else return null;
}
}
?>

```

**Πίνακας 33. Το αρχείο helper.php**

```

<?php
/**
 * @package Joomla.Site
 * @subpackage mod_PerssonalSimilarArticles
 * @copyright Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

if($similararticleslang == 1) require_once(dirname(__FILE__).DS.'lang'.DS.'en-GB.php');
else require_once(dirname(__FILE__).DS.'lang'.DS.'el-GR.php');
?>
<table width="100%" border="2" cellspacing="0" cellpadding="0">
<tr><td height="30" background="<?php
JURI::root().'modules/mod_perssonalsimilararticles/tmpl/images/menu.jpg'; ?>"><?php
echo
_MODPERSSONALSIMILARARTICLESNAME; ?></td></tr>
<tr><td>
<hr size=3 width=100% align="left" color="grey">
<?php if(!$user->guest){ ?>
<ul>
<?php
if($list != null){
echo $list;
}
else echo "&nbsp;";
?>
</ul>
<?php
}
?>
</td></tr></table>

```

**Πίνακας 34. Το αρχείο default.php**

```

<?xml version="1.0" encoding="UTF-8"?>
<extension type="module" version="1.7" client="site" method="upgrade">
<name>peRSSonal Similar Articles</name>
<author>Mpourelas Panagiotis</author>
<creationDate>31/01/2012</creationDate>
<copyright>Copyright (C) 2012. All rights reserved.</copyright>

```

```

<license>http://www.cti.gr</license>
<authorEmail>mpourelo@gmail.com</authorEmail>
<authorUrl>http://www.cti.gr</authorUrl>
<version>4.0</version>
<description>The is a module to show perSSonal similar articles from article selected</description>
<files>
<filename module="mod_perSSonalsimilararticles">mod_perSSonalsimilararticles.php</filename>
<filename>mod_perSSonalsimilararticles.xml</filename>
<filename>helper.php</filename>
<filename>index.html</filename>
<filename>identical.xsl</filename>
<filename>tmpl/default.php</filename>
<filename>tmpl/index.html</filename>
<filename>tmpl/lang/el-GR.php</filename>
<filename>tmpl/lang/en-GB.php</filename>
<filename>tmpl/images/menu.jpg</filename>
<folder>tmpl</folder>
</files>
</extension>

```

**Πίνακας 35. Το αρχείο εγκατάστασης mod\_perSSonalsimilararticles.xml**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_PerssonalTagging
 * @copyright        Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license          Computer Technology Institute - University of Patras
 */

// no direct access
defined('_JEXEC') or die;

// Include the syndicate functions only once
require_once dirname(__FILE__).'/helper.php';

// Get the user data
$list = mod_PerssonalTagging::getData($params);
$tagginglang = mod_PerssonalTagging::getLang();

// Get the layout
require JModuleHelper::getLayoutPath('mod_PerssonalTagging', $params->get('layout', 'default'));
?>

```

**Πίνακας 36. Το αρχείο mod\_perSSonaltagging.php**

```

<?php
/**
 * @package          Joomla.Site
 * @subpackage       mod_PerssonalTagging
 * @copyright        Copyright (C) 2012 CTI, Inc. All rights reserved.
 * @license          Computer Technology Institute - University of Patras
 */

defined('_JEXEC') or die;

class mod_PerssonalTagging
{
    function getLang()
    {

```

```

// Get user information
$user =& JFactory::getUser();

if(!$user->guest){
    $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='".$user->email.'"";
    $userdb =& JFactory::getDBO();
    $userdb->setQuery($userquery);
    $userrow = $userdb->loadObject();

    if($userrow != null){
        return intval($userrow->preflang);
    }
}

return 1;
}

function getData( &$params )
{
    // Get user information
    $user =& JFactory::getUser();

    if(!$user->guest){
        $userquery = "SELECT a.id as user_id, a.pref_lang as preflang FROM user_website as
a, #__users as b WHERE a.email=b.email and a.email='".$user->email.'"";
        $userdb =& JFactory::getDBO();
        $userdb->setQuery($userquery);
        $userrow = $userdb->loadObject();

        if($userrow != null){
            // Database query
            $list = array();
            $language = $userrow->preflang;
            $task = JRequest::getVar('task');

            if($task == 'readarticle'){
                $ar_id = JRequest::getVar('ar_id','-1');
                $query = "SELECT a.kw_id, a.kw_name FROM extraction_kw AS a
LEFT JOIN extraction_kw2ar AS b ON a.kw_id = b.kw_id WHERE b.ar_id=$ar_id ORDER BY
b.rel_frequency DESC LIMIT 0 , 13";
                $db =& JFactory::getDBO();
                $db->setQuery($query);
                $rows = $db->loadObjectList();

                // Get list items
                if ($rows!=null)
                {
                    $i=0;
                    foreach ($rows as $row)
                    {
                        $list["tagging"][$i]["kw_id"]=$row->kw_id;
                        $list["tagging"][$i]["kw_name"]=$row->kw_name;
                        $i++;
                    }
                    return $list;
                }
                else return null;
            }
            else return null;
        }
    }
    else return null;
}

```



```

    ?>
</ul>
<?php
}
else echo "&nbsp;";
?>
</td></tr></table>

```

**Πίνακας 38. Το αρχείο default.php**

```

// JavaScript Document

var xmlhttp_img;
var keyword_id;

function alter(scriptpath,action,kw_id,ar_id)
{
xmlhttp_img=GetXmlHttpRequestObject_img()
if (xmlhttp_img==null)
{
alert ("Your browser does not support AJAX!");
return;
}
keyword_id = kw_id;
var url=scriptpath+"change_kw.php?action="+action+"&kw_id="+kw_id+"&ar_id="+ar_id;
xmlhttp_img.open("GET",url,true);
xmlhttp_img.onreadystatechange=stateChanged_img;
xmlhttp_img.send(null);
}

function stateChanged_img()
{
    if (xmlhttp_img.readyState==4){
        var spanElement = document.getElementById("node_" + keyword_id);
        if(spanElement != null) spanElement.parentNode.removeChild(spanElement);
    }
}

function GetXmlHttpRequestObject_img()
{
var xmlhttp_img=null;
try
{
// Firefox, Opera 8.0+, Safari
xmlhttp_img=new XMLHttpRequest();
}
catch (e)
{
// Internet Explorer
try
{
xmlhttp_img=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
xmlhttp_img=new ActiveXObject("Microsoft.XMLHTTP");
}
}
return xmlhttp_img;
}

```

```

function MM_preloadImages() { //v3.0
  var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
  var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0; i<a.length; i++)
  if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image; d.MM_p[j++].src=a[i];}}
}

function MM_swapImgRestore() { //v3.0
  var i,x,a=document.MM_sr; for(i=0;a&&i<a.length&&(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
}

function MM_findObj(n, d) { //v4.01
  var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
  d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
  if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
  for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
  if(!x && d.getElementById) x=d.getElementById(n); return x;
}

function MM_swapImage() { //v3.0
  var i,j=0,x,a=MM_swapImage.arguments; document.MM_sr=new Array; for(i=0;i<(a.length-2);i+=3)
  if ((x=MM_findObj(a[i]))!=null){document.MM_sr[j++]=x; if(!x.oSrc) x.oSrc=x.src; x.src=a[i+2];}
}

```

**Πίνακας 39. Το βοηθητικό αρχείο alter.js**

```

<?php
// Load the Joomla framework
define( '_JEXEC', 1 );
define( 'DS', DIRECTORY_SEPARATOR );
define( 'JPATH_BASE', realpath(dirname(__FILE__).DS..'..DS..'));

require_once ( JPATH_BASE .DS.'includes'.DS.'defines.php' );
require_once ( JPATH_BASE .DS.'includes'.DS.'framework.php' );

$mainframe =& JFactory::getApplication('site');
$mainframe->initialise();

$session =& JFactory::getSession();
$user = $session->get( 'user' );
?>
<?php
// USE OF DATABASE
$user = &JFactory::getUser();

if(!$user->guest){
    $userquery = "SELECT a.id as user_id, a.app as app FROM user_website as a, #__users as b
WHERE a.email=b.email and a.email="" . $user->email .""";
    $userdb =& JFactory::getDBO();
    $userdb->setQuery($userquery);
    $userrow = $userdb->loadObject();

    if($userrow != null){
        $user_id = $userrow->user_id;
        $ar_id=$_GET['ar_id'];
        $kw_id = $_GET['kw_id'];
        $action = $_GET['action'];

        $db =&JFactory::getDBO();
        $query_kw_cat = "SELECT a.`rel_frequency` FROM `keywords_category_training`
a LEFT JOIN articles b ON a.cat_id=b.cat_id WHERE b.ar_id=$ar_id AND kw_id=$kw_id";
        $db->setQuery($query_kw_cat);

```

```

$rows = $db->loadObjectList();

foreach($rows as $haskw) {
    $thekw = $kw_id;
    $kwfr = $haskw->rel_frequency;

    $proximo;
    $weight;

    if($action == 1) {
        $proximo = -1;
        $weight = 2;
    }
    else if($action == 2) {
        $proximo = 1;
        $weight = 2.5;
    }
    $kweight = $proximo * $weight * $kwfr;

    $db =&JFactory::getDBO();
    $query_already_in = "SELECT user_id, kw_id, rel_frequency FROM
user_website_keyword WHERE user_id=$user_id AND kw_id=$thekw LIMIT 0,1";
    $db->setQuery($query_already_in);
    $db->query();
    $already_in = $db->getNumRows();
    $result_already_in = $db->loadObjectList();

    $user =0;
    $kw_id =0;
    $rel_frequency =0;
    foreach($result_already_in as $already_in_rows) {
        $user = $already_in_rows->user_id;
        $kw_id = $already_in_rows->kw_id;
        $rel_frequency = $already_in_rows->rel_frequency;
    }
    if($already_in != 0) {
        $rel_frequency = $rel_frequency + $kweight;

        $db =&JFactory::getDBO();
        $query_upd = "UPDATE user_website_keyword SET rel_frequency
= $rel_frequency WHERE user_id=$user AND kw_id=$thekw";
        $db->setQuery($query_upd);
        $db->query();
    }
    else {
        $db =&JFactory::getDBO();
        $query_ins = "INSERT INTO user_website_keyword VALUES
('$user','$thekw','$kweight','0)";
        $db->setQuery($query_upd);
        $db->query();
    }

    } // kleinei to while

    } // End of found user in Perssonal database
} // End of user is logged in
?>

```

**Πίνακας 40. Το βοηθητικό αρχείο change\_kw.php**

```

<?xml version="1.0" encoding="UTF-8"?>
<extension type="module" version="1.7" client="site" method="upgrade">
  <name>peRSSonal Tagging</name>
  <author>Mpourellos Panagiotis</author>
  <creationDate>31/01/2012</creationDate>
  <copyright>Copyright (C) 2012. All rights reserved.</copyright>
  <license>http://www.cti.gr</license>
  <authorEmail>mpourelo@gmail.com</authorEmail>
  <authorUrl>http://www.cti.gr</authorUrl>
  <version>4.0</version>
  <description>The is a module to show peRSSonal tagging of words for articles</description>
  <files>
    <filename module="mod_perssonaltagging">mod_perssonaltagging.php</filename>
    <filename>mod_perssonaltagging.xml</filename>
    <filename>helper.php</filename>
    <filename>index.html</filename>
    <filename>tmpl/default.php</filename>
    <filename>tmpl/index.html</filename>
    <filename>tmpl/images/minus.png</filename>
    <filename>tmpl/images/minus_on.png</filename>
    <filename>tmpl/images/plus.png</filename>
    <filename>tmpl/images/plus_on.png</filename>
    <filename>tmpl/images/menu.jpg</filename>
    <filename>tmpl/css/mystyle.css</filename>
    <filename>tmpl/lang/el-GR.php</filename>
    <filename>tmpl/lang/en-GB.php</filename>
    <folder>tmpl</folder>
  </files>
</extension>

```

**Πίνακας 41. Το αρχείο εγκατάστασης mod\_perssonaltagging.xml**

## ΒΙΒΛΙΟΓΡΑΦΙΑ

### ΒΙΒΛΙΑ / ΔΗΜΟΣΙΕΥΣΕΙΣ

- [1] C. Bouras, V. Pouloupoulos, V. Tsogkas, 25 - 28 June 2007, Efficient Summarization Based On Categorized Keywords. The 2007 International Conference on Data Mining (DMIN07), Las Vegas, Nevada, USA. pp. 285-291.
- [2] G. Adam, C. Bouras, V. Pouloupoulos, 2 - 4 September 2009, Monitoring RSS feeds. International Conference on Knowledge Management and Knowledge Technologies (I-KNOW 09), Gratz, Austria.
- [3] G. Adam, C. Bouras, V. Pouloupoulos, 26 - 29 May 2009, CUTER: An Efficient Useful Text Extraction Mechanism. The 2009 IEEE International Symposium on Mining and Web(WAM09), Bradford, UK.
- [4] G. Adam, C. Bouras, V. Pouloupoulos, 20-23 April, 2010, Image extraction from online text streams: A straightforward template independent approach without

training. The 24th International Conference on Advanced Information Networking and Applications (AINA), Perth, Australia.

[5] G. Adam, C. Bouras, V. Pouloupoulos, 24 - 28 May 2009, Utilizing RSS feeds for crawling the Web. The Fourth International Conference on Internet and Web Applications and Services - ICIW 2009, pp. 211 – 216.

[6] Pinkerton, B., 1994, Finding what people want: Experiences with the WebCrawler. In Proceedings of the First World Wide Web Conference, Geneva, Switzerland.

[7] Brin, S. and Page, L. 1998. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30(1-7):107–117.

[8] Edwards, J., McCurley, K. S., and Tomlin, J. A., 2001, "An adaptive model for optimizing performance of an incremental web crawler". In Proceedings of the Tenth Conference on World Wide Web: 106-113.

[9] Marc Najork and Janet L. Wiener. May 2001, Breadth-first crawling yields high-quality pages. In Proceedings of the Tenth Conference on World Wide Web, pages 114–118, Hong Kong, Elsevier Science.

[10] Zeinalipour-Yazti, D. and Dikaiakos, M. D., 2002, Design and implementation of a distributed crawler and filtering processor. In Proceedings of the Fifth Next Generation Information Technologies and Systems (NGITS), volume 2382 of Lecture Notes in Computer Science, pages 58–74, Caesarea, Israel. Springer.

[11] Boldi, P., Codenotti, B., Santini, M., and Vigna, S., 2004, UbiCrawler: a scalable fully distributed Web crawler. Software, Practice and Experience, 34(8):711–726.

[12] K.R. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, M.Y. Kan, B. Schiffman and S. Teufel, 2001, Columbia Multidocument Summarization: Approach and Evaluation, Document Understanding Conf.

[13] Suhit Gupta, Gail E. Kaiser, David Neistadt, Peter Grimm, 2003, DOM-based content extraction of HTML documents, WWW, pp. 207-214.

[14] Liu L., Pu C., and Tang W., 2000, WebCQ: Detecting and delivering information changes on the web, International Conference on Information and Knowledge Management.

[15] Chang, C-H, Lui, S-L., 2001, IEPAD: Information Extraction based on pattern discovery, WWW-10.

[16] Aidan Finn, Nicholas Kushmerick and Barry Smyth., 2001, Fact or fiction: Content classification for digital libraries, Joint DELOS-NSF Workshop on Personalisation and Recommender Systems in Digital Libraries (Dublin).

[17] A. F. R. Rahman, H. Alam and R. Hartono, 2001, Content Extraction from HTML Documents, 1st Int. Workshop on Web Document Analysis (WDA).

[18] S. Gupta, G. E. Kaiser, P. Grimm, M. F. Chiang, and J. Starren, 2005, "Automating content extraction of html documents," World Wide Web, vol. 8, no. 2, pp. 179–224.

- [19] R. Burget, 2007, "Layout Based Information Extraction from HTML Documents," Ninth International Conference on Document Analysis and Recognition, September 23-26, p.624-628.
- [20] J. Wang, X. He, C. Wang, J. Pei, J. Bu, C. Chen, Z. Guan, and W. V. Zhang, 2009, "Can we learn a template-independent wrapper for news article extraction from a single training site?," 15th SIGKDD, pp. 1345-1354.
- [21] K. Deschacht and M. Moens, 2007, "Text analysis for automatic image annotation,". 45th Annual Meeting of the Association of Computational Linguistics, pp. 1000 – 1007.
- [22] H. Zhigang, W. Xiang-Jun, L. Qingshan and L. Hanqing, November 06-11, 2005, "Semantic knowledge extraction and annotation for web images," 13th annual ACM international conference on Multimedia, Hilton, Singapore.
- [23] Y. Feng and M. Lapata, 2008, "Automatic Image Annotation Using Auxiliary Text Information", ACL HLT Columbus, Ohio, USA.
- [24] C. Bouras, V. Pouloupoulos, V. Tsogkas, 2008, Evaluating PeRSSonal: A Medium for Personalized Dynamically Created News Feeds, IADIS International Conference WWW/Internet Freiburg, Germany, 13 - 15 October 2008.
- [25] C. Bouras, V. Pouloupoulos, V. Tsogkas, 2008, PeRSSonal's core functionality evaluation: Enhancing text labeling through personalized summaries. Data and Knowledge Engineering Journal, Elsevier Science, Vol. 64, Issue 1, pp. 330 – 345.
- [26] C. Bouras, V. Pouloupoulos, V. Tsogkas, 3 - 8 July 2007, Personalizing text summarization based on sentence weighting, IADIS European First International Conference Data Mining (ECDM 2007), Lisbon, Portugal, C. Bouras, V. Pouloupoulos, V. Tsogkas, pp. 3 – 10.
- [27] Cho J, Garcia-Molina H., September 2000, The evolution of the Web and implications for an incremental crawler. Proceedings of the 26th International Conference on Very Large Databases, Morgan Kaufmann, pp. 200 – 209.
- [28] Ka Cheung Sia, Junghoo Cho, Hyun-Kyu Cho, July 2007, "Efficient Monitoring Algorithm for Fast News Alerts," IEEE Transactions on Knowledge and Data Engineering, volume 19, no. 7, pp. 950-961, doi:10.1109/TKDE.2007.1041.
- [29] Content Management Bible, Bob Boiko, 2002 Adv. Wiley Publishing, Inc.
- [30] Οδηγός του Joomla! 1.5, Δεύτερη έκδοση, Barrie M. North, 2009 Εκδόσεις Παπασωτηρίου

### ΔΙΠΛΩΜΑΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

- [31] Μηχανισμού και τεχνικές διαχείρισης επεξεργασίας, ανάλυσης, κατηγοριοποίησης, εξαγωγής, περίληψης και προσωποποίησης συχνά ανανεώσιμων δεδομένων του παγκόσμιου ιστού για παρουσίαση σε σταθερές και κινητές συσκευές, Βασίλειος Πουλόπουλος, Διδακτορική διατριβή 2010, Επιβλέπων Κ. Μπούρας
- [32] Προσωποποιημένη προβολή περιεχομένου του διαδικτύου σε Desktop εφαρμογή με τεχνικές ανάκτησης δεδομένων, προεπεξεργασίας κειμένου, αυτόματης

κατηγοριοποίησης και εξαγωγής περίληψης, Τσόγκας Βασίλειος, Μεταπτυχιακή εργασία 2008, Επιβλέπων Κ. Μπούρας.

[33] Μηχανισμός ανάκτησης ειδησεογραφίας από τον παγκόσμιο ιστό χρησιμοποιώντας σύγχρονα κανάλια επικοινωνίας και εξαγωγή χρήσιμης πληροφορίας από τις ανακτημένες σελίδες, Αδάμ Γεώργιος, Διπλωματική εργασία 2010, Επιβλέπων Κ. Μπούρας.

### ΔΙΚΤΥΑΚΟΙ ΤΟΠΟΙ

- [34] The free encyclopedia, Wikipedia. <http://en.wikipedia.org>
- [35] Ελληνική Ελεύθερη εγκυκλοπαίδεια, <http://el.wikipedia.org>
- [36] Μαθήματα για κατασκευή ιστοσελίδων, <http://www.w3schools.com>
- [37] Shoutex CMS Analysis, <http://cms.shoutex.com>
- [38] Μηχανή αναζήτησης Yahoo. <http://www.yahoo.com>
- [39] Μηχανή αναζήτησης Google. <http://www.google.com>
- [40] HTTrack web crawler. <http://www.httrack.com/>
- [41] Άδεια χρήσης GNU General Public License. <http://www.gnu.org/licenses/gpl-3.0.txt>
- [42] Aspseek web crawler. <http://www.aspseek.org/>
- [43] DataparkSearch Engine - open source search engine. <http://www.dataparksearch.org/>
- [44] Heritrix crawler. <http://crawler.archive.org/>
- [45] The Methabot Project. <http://bithack.se/projects/methabot/>
- [46] ht://Dig – λογισμικό μηχανών αναζήτησης. <http://www.htdig.org/>
- [47] ICDL crawling. <http://www.omfica.org/>
- [48] Nutch web crawler. <http://nutch.apache.org/>
- [49] Πολυχρηστικός crawler Pavuk. <http://pavuk.sourceforge.net/>
- [50] GNU Wget. <http://www.gnu.org/software/wget/>
- [51] HTML Tidy Open Source Project, <http://tidy.sourceforge.net/>
- [52] MySQL, Βάση Δεδομένων ανοιχτού κώδικα. <http://www.mysql.com/>
- [53] PostgreSQL, Βάση Δεδομένων ανοιχτού κώδικα. <http://www.postgresql.org/>
- [54] MSSQL, Σχεσιακή Βάση Δεδομένων <http://www.microsoft.com/sql/default.mspx/>

- [55] Sybase SQL Server, <http://www.sybase.com/>
- [56] Η γλώσσα C++. <http://en.wikipedia.org/wiki/C++>
- [57] Το χρονικό της Java. <http://ils.unc.edu/blaze/java/javahist.html>
- [58] libCurl, Βιβλιοθήκη λήψης δεδομένων από το διαδίκτυο. <http://curl.haxx.se/>
- [59] xerces-c, Βιβλιοθήκη ανάλυσης XML. <http://xerces.apache.org/xerces-c/>
- [60] boost-regex, Βιβλιοθήκη υποστήριξης κανονικών εκφράσεων. <http://www.boost.org/>
- [61] libGD, Λογισμικό γραφικών. <http://www.libgd.org>