

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

Διπλωματική Εργασία

---

**Ανάπτυξη εφαρμογής στο παραθυρικό περιβάλλον Sugar που εκμεταλλεύεται τη mesh δικτύωση του OLPC.**

---



Συγγραφέας:  
Ουρανία Ροδά  
Α.Μ.:3208

Υπεύθυνος καθηγητής:  
Χρήστος Μπούρας  
Επιβλέπων:  
Βασίλης Πουλόπουλος

23 Μαρτίου 2010

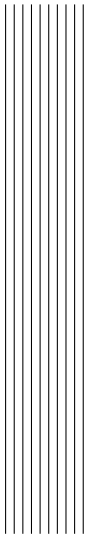
*Θα ήθελα να ευχαριστήσω πολύ για την πολύτιμη βοήθειά  
τους, τους γονείς μου Αντώνη και Λένα, τον αδερφό μου  
Κώστα, τους φίλους μου Φωτεινή Φ., Ηλιάνα Κ., Δημήτρη  
Β., Δημήτρη Τ., Ιωάννη Μ. και τέλος ιδιαίτερες ευχαριστίες  
στον Δημήτρη Κ.*



# Περιεχόμενα

<b>1</b>	<b>Εισαγωγή</b>	<b>9</b>
1.1	Πρόλογος	9
1.1.1	Τι είναι εκπαίδευση	9
1.1.2	Τι είναι παιχνίδι	10
1.1.3	Το παιχνίδι ως εκπαιδευτικό εργαλείο	10
1.2	Προσδιορισμός του προβλήματος	11
1.3	Δομή της εργασίας	11
<b>2</b>	<b>Ιστορικά στοιχεία</b>	<b>13</b>
2.1	Το μάθημα της πληροφορικής	13
2.1.1	Το μάθημα της πληροφορικής στο Γυμνάσιο	13
2.1.2	Το μάθημα της Πληροφορικής στο Λύκειο	15
2.2	Οι υπολογιστές στα σχολεία	16
2.2.1	Οι υπολογιστές στα σχολεία ανά τον κόσμο	17
2.2.2	Οι υπολογιστές στα σχολεία στην Ελλάδα	17
2.3	Πως φτάσαμε στο XO	17
2.3.1	Οι πρώτοι υπολογιστές	17
2.3.2	Το laptop	18
2.3.3	Το XO-1	18
2.3.4	Το XO-2	18
<b>3</b>	<b>Γενικές Πληροφορίες για το OLPC</b>	<b>21</b>
3.1	Τι είναι το OLPC	21
3.2	Πως ξεκίνησε το OLPC	21
3.3	Ιστορία του OLPC	22
3.4	Τα πρώτα στάδια του OLPC	22
3.5	Κόστος του OLPC	23
3.6	Γιατί το OLPC χρησιμοποιεί προγράμματα ανοιχτού λογισμικού	23

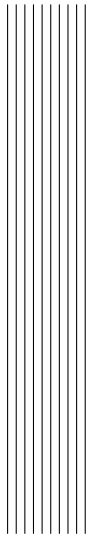
3.7	Windows στο OLPC . . . . .	25
<b>4</b>	<b>Ειδικότερα για το OLPC</b>	<b>27</b>
4.1	Hardware στο OLPC . . . . .	27
4.1.1	CL1 Σύστημα αρχιτεκτονικής . . . . .	30
4.1.2	Καινοτομίες στο OLPC . . . . .	47
4.2	Software στο OLPC . . . . .	47
4.2.1	Τι είναι το περιβάλλον Sugar . . . . .	56
4.2.2	Εφαρμογές που έχει το OLPC . . . . .	59
4.3	Τι είναι η mesh δικτύωση . . . . .	61
<b>5</b>	<b>Ανάπτυξη εφαρμογής για το OLPC</b>	<b>65</b>
5.1	Γλώσσες προγραμματισμού για ανάπτυξη εφαρμογής στο OLPC . . . . .	65
5.2	Η γλώσσα προγραμματισμού Python . . . . .	65
5.3	Πως αναπτύσσεται μια εφαρμογή στο περιβάλλον Sugar που εκμεταλλεύεται τη mesh δικτύωση . . . . .	68
5.4	Η εφαρμογή μας : Λεξο-μαχίες (Word-fight) . . . . .	73
5.4.1	Λόγοι επιλογής της εφαρμογής . . . . .	73
5.4.2	Σχεδιαστικές αποφάσεις . . . . .	74
<b>6</b>	<b>Τεχνική ανάλυση της εφαρμογής</b>	<b>75</b>
6.1	Γενικά χαρακτηριστικά της εφαρμογής . . . . .	75
6.1.1	Η διεπαφή του μαθητή . . . . .	75
6.1.2	Η διεπαφή του εκπαιδευτικού . . . . .	79
6.1.3	Η βασική διεπαφή . . . . .	81
6.2	Πειραματικές Διαδικασίες . . . . .	86
<b>7</b>	<b>Επίλογος</b>	<b>95</b>
7.1	Αποτίμηση . . . . .	95
7.2	Μελλοντική Εργασία . . . . .	95
7.3	Επίλογος . . . . .	95
<b>Α΄</b>	<b>Παράρτημα Α</b>	<b>97</b>
<b>Β΄</b>	<b>Παράρτημα Β</b>	<b>99</b>
Β΄.1	Ο κώδικας της διεπαφής του μαθητή . . . . .	99
Β΄.2	Ο κώδικας της διεπαφής της δασκάλας . . . . .	105
Β΄.3	Ο κώδικας της κεντρικής διεπαφής . . . . .	109
	<b>Βιβλιογραφία</b>	<b>117</b>



# Κατάλογος Σχημάτων

4.1 Το CL1 σύστημα Αρχιτεκτονικής . . . . .	30
4.2 Ο φορτιστής της μπαταρίας του OLPC . . . . .	33
4.3 Η μπαταρία του OLPC . . . . .	33
4.4 Η οθόνη στην κανονική της λειτουργία . . . . .	38
4.5 Η οθόνη στην ασπρόμαυρη λειτουργία . . . . .	39
4.6 Η mesh δικτύωση . . . . .	42
4.7 Οι κεραίες του OLPC . . . . .	43
4.8 Το πληκτρολόγιο του OLPC . . . . .	44
4.9 Το αρχικό Interface του Sugar . . . . .	56
4.10 Sugar home view . . . . .	57
4.11 Neighborhood . . . . .	58
4.12 Sugar home view from pre-0.82 releases . . . . .	58
4.13 Sugar Pippy activity . . . . .	59
4.14 Activity Chart . . . . .	60
4.15 Activities στο OLPC . . . . .	60
6.1 Η διεπαφή του μαθητή . . . . .	75
6.2 Το παράθυρο του μενού στη διεπαφή του μαθητή . . . . .	77
6.3 Η διεπαφή του εκπαιδευτικού . . . . .	79
6.4 Το παράθυρο του μενού στη διεπαφή του εκπαιδευτικού . . . . .	80
6.5 Η διεπαφή του παιχνιδιού . . . . .	81
6.6 Το παράθυρο που αναδύεται όταν τελειώνουν οι λέξεις . . . . .	85
6.7 Εισαγωγή των λέξεων από τον εκπαιδευτικό . . . . .	86
6.8 Η κεντρική διεπαφή μετά από την εισαγωγή των λέξεων . . . . .	87
6.9 Εισαγωγή ονόματος και πρότασης από μαθητή 1 . . . . .	87
6.10 Η κεντρική διεπαφή μετά την εισαγωγή του πρώτου μαθητή . . . . .	88
6.11 Εισαγωγή ονόματος και πρότασης από μαθητή 2 . . . . .	88
6.12 Η κεντρική διεπαφή μετά την εισαγωγή του δεύτερου μαθητή . . . . .	89

6.13	Εισαγωγή ονόματος και πρότασης από μαθητή 3 . . . . .	89
6.14	Η κεντρική διεπαφή μετά την εισαγωγή του τρίτου μαθητή . . . . .	90
6.15	Εισαγωγή ονόματος και πρότασης από μαθητή 4 . . . . .	90
6.16	Η κεντρική διεπαφή μετά την εισαγωγή του τέταρτου μαθητή . . . . .	91
6.17	Η κεντρική διεπαφή όταν ίδιοι παίκτες εισάγουν 2 προτάσεις . . . . .	93
A.1	Login Sugar Interface . . . . .	98



## Συντομογραφίες

ECDL	European Computer Driving License
CLASS	Computer Literacy and Studies in School
OLPC	One Laptop Per Child
ENIAC	Electronic Numerical Integrator and Computer
HDL	Hundred Dollar Laptop
CTO	Chief Technical Officer
WSIS	World Summit on the Information Society
G1G1	Give 1 Get 1
GM	General Manager
CPU	Central Processing Unit
I/O	Input Output
LED	Light Emitting Diode
LPC	Low Pin Count
RTC	Real Time Clock
DMA	Direct Memory Access
SD	Secure Digital
EC	Embedded Controller
GPU	Graphics Processing Unit
IC's	Integrated Circuits
DDR	Double Data Rate
SPD	Serial Presence Detect
DC	Direct Current
NiHM	Nickel Metal Hydride

LiFePo	Lithium Iron Phosphate battery
PS/2	Personal System/2
ACR	Accumulated Current Register
SCI	System Control Interrupt
ACPI	Advanced Configuration and Power Interface
LCD	Liquid Crystal Display
TFT	Transistor-Transistor Logic
DCON	Display Controller
SMBus	System Management Bus
SNR	Signal-to-Noise Ratio
ECC	Error Correcting Code Memory
SLC	Signal Layer Cell
MLC	Multiple Layer Cell
MMC	Memory Card
RF	Radio Frequency
F	Fibers
RS	Recommended Standard
JTAG	Joint Test Action Group
UL	Underwriters Laboratories
CSA	Canadian Standards Association
CE	Conformance European
IEC	International Electrotechnical Commission
RoHs	Restriction of Hazardous Substances
EPEAT	Electronic Product Environmental Assessment Tool
RSS	Really Simple Syndication
ALAS	Advanced Language Acquisition Software
GPL	General Public License
MANET	Mobile Ad-hoc Networking
Voip	Voice over internet protocol
AP	Access Point
CSMA	Carrier Sense Multiple Access
IEEE	Institute of Electrical and Electronics Engineers
OSI	Open Source Initiative
PSF	Python Software Foundation
STReP	Specified Targeted Research Project
FP6	Sixth Framework Programme
ICE	Internet Communications Engine
FOSS	Free and Open Source Software





# 1 Εισαγωγή

## 1.1 Πρόλογος

Ο ηλεκτρονικός υπολογιστής είναι μια μηχανή κατασκευασμένη κυρίως από ηλεκτρονικά κυκλώματα και δευτερευόντως από ηλεκτρικά και μηχανικά συστήματα, και έχει ως σκοπό να επεξεργάζεται πληροφορίες. Ο ηλεκτρονικός υπολογιστής είναι ένα αυτοματοποιημένο, ηλεκτρονικό, ψηφιακό επαναπρογραμματιζόμενο σύστημα γενικής χρήσης το οποίο μπορεί να επεξεργάζεται δεδομένα βάσει ενός συνόλου προκαθορισμένων οδηγιών, των εντολών, που συνολικά ονομάζονται πρόγραμμα.

Στη σημερινή εποχή, την εποχή της πληροφορικής η γνώση χειρισμού ηλεκτρονικών υπολογιστών είναι υψίστης σημασίας, καθώς οι υπολογιστές πλέον αποτελούν αναπόσπαστο κομμάτι της καθημερινότητας της σύγχρονης κοινωνίας. Γι' αυτό τον λόγο έχει ξεκινήσει εδώ και χρόνια η εισαγωγή των υπολογιστών και της πληροφορικής στη μαθησιακή διαδικασία. Στο πλαίσιο αυτό και σε συνδυασμό με την ιδιαίτερη αγάπη στα παιδιά αποφασίσαμε να ασχοληθούμε με την εργασία αυτή. Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τον κ. Χρήστο Μπούρα Αναπληρωτή Καθηγητή του τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών για την ευκαιρία που μου έδωσε καθώς και τον επιβλέποντα Βασίλη Πουλόπουλο για την πολύτιμη βοήθειά του στην εκπόνηση της εργασίας.

### 1.1.1 Τι είναι εκπαίδευση

Η εκπαίδευση είναι η απόκτηση καινούριας γνώσης, συμπεριφοράς, δεξιοτεχνίας, καινούριων αρχών και προτιμήσεων ή αντίληψης και μπορεί να περιλαμβάνει διάφορα είδη πληροφοριών. Ικανότητα για εκπαίδευση έχουν οι άνθρωποι τα ζώα και κάποιες μηχανές. Η εκπαίδευση του ανθρώπου μπορεί να πραγματοποιηθεί είτε ως μόρφωση είτε ως προσωπική εξέλιξη, μπορεί να είναι στοχευμένη και μπορεί να βοηθηθεί από κίνητρα. Η μελέτη για το πως προκύπτει η εκπαίδευση είναι αντικείμενο της νευροψυχολογίας, της εκπαιδευτικής ψυχολογίας, της θεωρίας μάθησης και της παιδαγωγικής. Η εκπαίδευση

μπορεί να προκύψει σαν αποτέλεσμα της συνήθειας ή κλασικών καταστάσεων γεγονός που έχει παρατηρηθεί σε πολλά είδη ζώων ή μπορεί να προκύψει από πιο σύνθετες δραστηριότητες όπως το παιχνίδι, σύμφωνα με παρατηρήσεις που έχουν γίνει, σε σχετικά ευφυή ζώα και στους ανθρώπους. Η εκπαίδευση μπορεί να προκύψει συνειδητά ή ασυνείδητα.

### 1.1.2 Τι είναι παιχνίδι

Το παιχνίδι αναφέρεται σε ένα φάσμα από δραστηριότητες στις οποίες κάποιος συμμετέχει εθελοντικά και εμπιέχει το κίνητρο της ενθάρρυνσης και σχετίζεται συνήθως με την ευχαρίστηση και την απόλαυση. Το παιχνίδι αποτελείται από διασκεδαστικές, προσποιητές ή πλασματικές και διαπροσωπικές αλληλεπιδράσεις. Το παιχνίδι το αντιλαμβάνονται οι άνθρωποι αλλά και τα ζώα, είναι μια τελετουργία που είναι φανερό κατά την εξέλιξη ιδιαίτερα κατά τη διαδικασία της εκμάθησης και της κοινωνικοποίησης. Το παιχνίδι συχνά περιλαμβάνει βοηθήματα, εργαλεία ή ζώα στα πλαίσια της εκπαίδευσης και της ψυχαγωγίας. Κάποια παιχνίδια έχουν σαφώς προσδιορισμένους στόχους και είναι δομημένα με κανόνες ενώ κάποια άλλα δεν έχουν ούτε στόχους ούτε κανόνες και ονομάζονται αδόμετα στην λογοτεχνία.

Έχουν γίνει συντονισμένες προσπάθειες για να αναγνωριστούν οι αξίες του παιχνιδιού αλλά υπάρχουν κάποιες αμφιταλαντεύσεις. Για παράδειγμα, το παιχνίδι έχει επικρατήσει σαν μια επιπόλαιη και ασήμαντη δραστηριότητα, ωστόσο, όταν παρακολουθήσει κάποιος ένα παιδί να παίζει μπορεί να διαπιστώσει την σοβαρότητα και τη σαγηνευτική αφοσίωση με την οποία συμμετέχει. Ένα άλλο κριτήριο του παιχνιδιού είναι ο χαλαρωτικός ρυθμός και η ελευθερία ενάντια στον εξαναγκασμό.

Το παιχνίδι μπορεί επίσης, να έχει τη μορφή πρόβας για γεγονότα της πραγματικής ζωής ή μπορεί να αποτελεί μια πρόφαση για να ανακαλύψει κάποιος τις αντιδράσεις των άλλων συμμετέχοντα και αλληλεπιδρώντα σε ένα παιχνίδι.

### 1.1.3 Το παιχνίδι ως εκπαιδευτικό εργαλείο

Το παιχνίδι προσεγγίζεται από πολλούς θεωρητικούς ως η πρώτη μορφή εκπαίδευσης. Τα παιδιά παίζουν, πειραματίζονται με τον κόσμο, μαθαίνουν τους κανόνες και μαθαίνουν να αλληλεπιδρούν. Ο Vygotsky, ένας πρωτοποριακός ψυχολόγος που μελέτησε το παιχνίδι ως ψυχολογικό φαινόμενο, υποστηρίζει ότι το παιχνίδι είναι ουσιωδέστατο για την εξέλιξη των παιδιών αφού μέσω του παιχνιδιού αντιλαμβάνονται το περιβάλλον.

Σε όλον τον κόσμο, οι εκπαιδευτικοί ενδιαφέρονται όλο και πιο πολύ για τον δυναμικό ρόλο των παιχνιδιών υπολογιστή και βίντεο όσον αφορά την υποστήριξη της μάθησης των νεαρών ατόμων. Πρόσφατες μελέτες δείχνουν ότι, όταν παίζουν παιχνίδια, τα παιδιά επιδίδονται σταθερά σε δραστηριότητες μάθησης οι οποίες είναι πολύ πιο περίπλοκες και προκλητικές από το μεγαλύτερο μέρος των κανονικών σχολικών εργασιών τους. Ακόμα και παιχνίδια χωρίς κανέναν εκπαιδευτικό σκοπό απαιτούν από τους παίκτες να μάθουν πολλά πράγματα. Τα παιχνίδια είναι περίπλοκα, απαιτούν προσαρμοστικότητα και αναγκάζουν τους παίκτες να αναπτύξουν ένα ευρύ φάσμα ικανοτήτων. Οι ερευνητές έχουν καταγράψει πολυάριθμα οφέλη από το παιχνίδι, μεταξύ των οποίων η αύξηση της κοινωνικό-αισθηματικής, γνωστικής και φυσικής ανάπτυξης του παιδιού, καθώς και το ακόνισμα των, λεπτών και μη, κινητικών ικανοτήτων, του συντονισμού χεριού-ματιού, της δημιουργικότητας, της συγκέντρωσης και των μηχανισμών επίλυσης προβλημάτων.

## 1.2 Προσδιορισμός του προβλήματος

Σε αυτή την εργασία θα αναπτύξουμε ένα εκπαιδευτικό λογισμικό για το OLPC που εκμεταλλεύεται τη mesh δικτύωση. Η εφαρμογή θα αναπτυχθεί σε γλώσσα PyGTK, μια γλώσσα Python που υποστηρίζει γραφικό περιβάλλον. Σκοπός μας είναι να συνδυάσουμε την εκπαίδευση με το παιχνίδι ώστε μέσω της διασκέδασης τα παιδιά να αποκομίζουν πολύτιμη γνώση αλλά και να εξετάζεται από τη δασκάλα το τι έχουν μάθει μέχρι τώρα. Θα προσπαθήσουμε η εφαρμογή μας να κρατάει το ενδιαφέρον των παιδιών, να εξυπηρετεί τις εκπαιδευτικές ανάγκες του δασκάλου, να έχει ένα ευχάριστο γραφικό περιβάλλον, να είναι λειτουργικό, να είναι εφικτό να λειτουργεί αποδοτικά στο XO δεδομένου του λογισμικού και του υλικού του XO, να εκμεταλλεύεται αποδοτικά τη mesh δικτύωση του XO.

## 1.3 Δομή της εργασίας

Στο κεφάλαιο 2 της εργασίας παραθέτουμε κάποια ιστορικά στοιχεία γενικά για τη χρήση των υπολογιστών στα σχολεία και στην εκπαιδευτική διαδικασία καθώς και την πορεία μέχρι να φτάσουμε στο OLPC. Στο κεφάλαιο 3 δίνουμε κάποιες πληροφορίες για το OLPC, για το πως ξεκίνησε την πορεία του, για το κόστος του και για το λόγο που χρησιμοποιεί προγράμματα ανοιχτού λογισμικού. Στο κεφάλαιο 4 δίνουμε κάποιες πιο ειδικές πληροφορίες για το OLPC, για τις καινοτομίες του, για τη mesh δικτύωση που χρησιμοποιεί, για το περιβάλλον Sugar και για τις εφαρμογές του. Στο κεφάλαιο 5 αναφερόμαστε στις γλώσσες προγραμματισμού που υποστηρίζει το OLPC, στη γλώσσα Python, στον τρόπο που αναπτύσσουμε μια εφαρμογή που εκμεταλλεύεται τη mesh δικτύωση, δίνουμε την πλήρη περιγραφή της εφαρμογής που θα αναπτύξουμε και αναφέρουμε τους λόγους για τους οποίους επιλέξαμε αυτή την εφαρμογή. Στο κεφάλαιο 6 κάνουμε μια τεχνική ανάλυση της εφαρμογής και στο κεφάλαιο 7 καταγράφουμε τις πειραματικές διαδικασίες, την αποτίμηση, αξιολόγηση του λογισμικού τη μελλοντική εργασία που θα μπορούσε να γίνει. Τέλος, έχουμε τα Παραρτήματα που περιλαμβάνουν οδηγίες για το πως μπορεί κάποιος να εγκαταστήσει το XO σε οποιοδήποτε σύστημα, τον κώδικα της εφαρμογής και τη βιβλιογραφία.





## 2 Ιστορικά στοιχεία

### 2.1 Το μάθημα της πληροφορικής

Η Πληροφορική είναι η επιστήμη που μελετά την κωδικοποίηση, τη διαχείριση και τη μετάδοση συμβολικών αναπαραστάσεων γνώσεων και πληροφοριών, καθώς επίσης και την ανάπτυξη και εξέλιξη διατάξεων, συσκευών, υπηρεσιών και συστημάτων συλλογής, αποθήκευσης, επεξεργασίας, εξόρυξης και ανταλλαγής των ανωτέρω συμβολικών αναπαραστάσεων.

Η Πληροφορική συνδέεται στενά με την επιστήμη υπολογιστών διότι η υλοποίηση των μεθόδων της βασίστηκε από την πρώτη στιγμή στα συστήματα υπολογιστών. Επιστημονικά όμως, έχει έναν ευρύτερο σκοπό που δεν περιορίζεται σε συγκεκριμένες τεχνολογικές επιλογές. Για παράδειγμα, ο αλγόριθμος της δυαδικής αναζήτησης μπορεί να εφαρμοστεί και σε τηλεφωνικό κατάλογο χειρωνακτικά, ενώ ένα πρωτόκολλο επικοινωνίας μπορεί να εφαρμοστεί ακόμη και σε σήματα καπνού. Η πληροφορική επομένως, μπορεί να μελετηθεί είτε ανεξάρτητα από τις τεχνολογικές της συνιστώσες, δηλαδή τη μηχανική υπολογιστών και τις τηλεπικοινωνίες, είτε ως ένα ενιαίο με αυτές επιστημονικό πεδίο.

Λόγω των τεχνολογικών εξελίξεων και της ραγδαίας εξάπλωσης της χρήσης των υπολογιστών σε πολλούς τομείς της καθημερινότητας, το μάθημα της Πληροφορικής εισήχθηκε αρχικά στα Πανεπιστήμια και στη συνέχεια στα σχολεία σε μια προσπάθεια εκσυγχρονισμού της μαθησιακής διαδικασίας. Δεν γινόταν άλλωστε διαφορετικά λόγω του ότι πλέον δεν νοείται εκπαίδευση χωρίς χρήση, έστω και σε μικρό βαθμό, του υπολογιστή.

#### 2.1.1 Το μάθημα της πληροφορικής στο Γυμνάσιο

Το μάθημα της Πληροφορικής διδάσκεται πλέον από την πρώτη Γυμνασίου. Σύμφωνα με το σχεδιασμό και το έργο του Παιδαγωγικού Ινστιτούτου, τελειώνοντας οι μαθητές το Γυμνάσιο, πρέπει να είναι σε θέση:

- Να εξηγούν βασικές έννοιες και όρους της Πληροφορικής (δεδομένα, πληροφορίες,

κωδικοποίηση, επεξεργασία δεδομένων, αρχείο, αποθήκευση, πρόγραμμα, λογισμικό, λογισμικό συστήματος κ.λ.π.).

- Να περιγράφουν τη λειτουργία των κυριότερων μονάδων του υπολογιστή.
- Να εξηγούν τις βασικές έννοιες και τη βασική ορολογία της σύγχρονης δικτυακής τεχνολογίας και της τεχνολογίας των πολυμέσων.
- Να χρησιμοποιούν με ευχέρεια ένα υπολογιστικό σύστημα σε γραφικό περιβάλλον επικοινωνίας.
- Να χρησιμοποιούν βασικά προγράμματα εφαρμογών για γραπτή έκφραση και επικοινωνία, σχεδίαση, ζωγραφική, μοντελοποίηση, αναζήτηση - συλλογή - επεξεργασία - παρουσίαση και μετάδοση πληροφοριών κ.λ.π.
- Να χρησιμοποιούν το παγκόσμιο διαδίκτυο και να αξιοποιούν τις υπηρεσίες που προσφέρει.
- Να επιλύουν απλά προγράμματα σε προγραμματιστικό περιβάλλον.
- Να συζητούν και να ενημερώνονται για τις τεχνολογικές εξελίξεις και να αναγνωρίζουν τις επιπτώσεις τους στους διάφορους τομείς της ανθρώπινης δραστηριότητας.

Πιο συγκεκριμένα, οι διδακτικές ενότητες Α' τάξης είναι :

- Γνωρίζω τον Υπολογιστή (Δεδομένα, Πληροφορίες και Υπολογιστές - Πώς φτάσαμε στους σημερινούς υπολογιστές - Το υλικό και το λογισμικό - Προστασία υλικού, λογισμικού και δεδομένων, εργονομία).
- Επικοινωνώ με τον Υπολογιστή (Γραφικά περιβάλλοντα επικοινωνίας).
- Διερευνώ - Δημιουργώ - Ανακαλύπτω (Εργαλεία, Τεχνικές - Συνθετικές εργασίες με λογισμικό εφαρμογών γενικής χρήσης, λογισμικό δικτύων και εκπαιδευτικό λογισμικό).

Οι διδακτικές ενότητες της Β' τάξης είναι :

- Γνωρίζω τον Υπολογιστή (Τεχνολογία υπολογιστών - Αναπαράσταση των πληροφοριών στον υπολογιστή - Αποθήκευση των πληροφοριών στον υπολογιστή - Πολυμέσα).
- Επικοινωνώ με τον Υπολογιστή (Διαχείριση αρχείων και φακέλων).
- Διερευνώ - Δημιουργώ - Ανακαλύπτω (Εργαλεία, Τεχνικές - Συνθετικές εργασίες με λογισμικό εφαρμογών γενικής χρήσης, λογισμικό δικτύων και εκπαιδευτικό λογισμικό).

Οι διδακτικές ενότητες της Γ' τάξης είναι :

- Διερευνώ - Δημιουργώ - Ανακαλύπτω (Εργαλεία, Τεχνικές - Συνθετικές εργασίες με λογισμικό εφαρμογών γενικής χρήσης, λογισμικό δικτύων, λογισμικό ανάπτυξης πολυμέσων, εκπαιδευτικό λογισμικό και προγραμματιστικά εργαλεία).

- Ελέγχω - προγραμματίζω τον Υπολογιστή (Η έννοια του αλγόριθμου - Ο κύκλος ανάπτυξης ενός προγράμματος - Το περιβάλλον μιας γλώσσας προγραμματισμού - Βασικές δομές μιας συμβολικής γλώσσας).
- Ο υπολογιστής στη ζωή μας (Γενική επισκόπηση των εφαρμογών της πληροφορικής - Όλα αλλάζουν... - Το μέλλον...).

### 2.1.2 Το μάθημα της Πληροφορικής στο Λύκειο

Στο Λύκειο διδάσκεται ένα αμιγές μάθημα γενικών γνώσεων Πληροφορικής στο οποίο προοδευτικά εντάσσεται και η χρήση των υπολογιστικών και δικτυακών τεχνολογιών ως μέσο στήριξης της μαθησιακής διαδικασίας σε όλα τα γνωστικά αντικείμενα του προγράμματος σπουδών. Στο πλαίσιο του μαθήματος δίνεται ιδιαίτερη έμφαση στις γνωστικές και τις κοινωνικές διαστάσεις της χρήσης της Πληροφορικής στην εκπαιδευτική διαδικασία. Με τον τρόπο αυτό συνδυάζονται παιδαγωγικά πλεονεκτήματα και χρήση των νέων τεχνολογιών, οι οποίες δεν αποτελούν μόνον ένα γνωστικό αντικείμενο, που είναι απαραίτητο σήμερα για τον τεχνολογικό αλφαριθμητισμό των μαθητών, αλλά και ένα εποπτικό «πολυμέσο» και γνωστικό εργαλείο διδασκαλίας για όλα τα μαθήματα, με πολλαπλές πηγές πληροφόρησης και επικοινωνίας με τον κόσμο της γνώσης.

Τα μαθήματα επιλογής, Εφαρμογές Πληροφορικής Α' Λυκείου και Εφαρμογές Υπολογιστών Β' και Γ' Λυκείου εντάσσονται στο ωρολόγιο πρόγραμμα, του Ενιαίου Λυκείου και έχουν γενικό σκοπό:

- την επέκταση της γενικής πληροφορικής παιδείας των μαθητών με έμφαση στην ανάπτυξη ικανοτήτων και δεξιοτήτων στη χρήση και αξιοποίηση των τεχνολογιών ως εργαλείων μάθησης και σκέψης.
- την ενημέρωση των μαθητών για τις εφαρμογές της Πληροφορικής και ειδικότερα για τις δυνατότητες που προσφέρει και τις προοπτικές που δημιουργεί στην επαγγελματική κατεύθυνση που επέλεξαν.
- την ευαισθητοποίηση, τον προβληματισμό και την ανάπτυξη κριτικής ικανότητας εκ μέρους των μαθητών, στα κοινωνικά, ηθικά, πολιτισμικά, κ.ά. ζητήματα που τίθενται με τις νέες τεχνολογίες στις ανθρώπινες δραστηριότητες.
- την απόκτηση του διπλώματος πιστοποίησης χρήσης Η/Υ (ECDL).

Για τους παραπάνω σκοπούς χρησιμοποιούνται λογισμικά εφαρμογών αλλά και ανάπτυξης εφαρμογών μέσα από τα οποία οι μαθητές πραγματοποιούν ασκήσεις και ολοκληρωμένες συνθετικές εργασίες. Σε προγραμματιστικό επίπεδο οι μαθητές εργάζονται με λογισμικό ψευδογλώσσας ενώ από λογισμικό ευρείας χρήσης χρησιμοποιούν εργαλεία επεξεργασίας κειμένου, βάσεων δεδομένων, δημιουργίας παρουσιάσεων, δημιουργίας φύλλων εργασίας και δημιουργίας ιστοσελίδων.

Ο γενικός σκοπός του μαθήματος Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον Γ' Λυκείου είναι να αναπτύξουν οι μαθητές αναλυτική και συνθετική σκέψη, να αποκτήσουν ικανότητες μεθοδολογικού χαρακτήρα και να μπορούν να επιλύουν προβλήματα σε προγραμματιστικό περιβάλλον. Η προσέγγιση των εννοιών και η καλλιέργεια δεξιοτήτων που απαιτούνται για την υλοποίηση του γενικού σκοπού γίνεται με την εξής

διαδικασία: οι μαθητές κατανοούν το πρόβλημα, το αναλύουν, προσεγγίζουν με αυστηρότητα την έννοια του αλγορίθμου και περιγράφουν την αλγοριθμική διαδικασία επίλυσής του.

Στα δημοτικά σχολεία της Ελλάδας προς το παρών δε διδάσκεται το μάθημα της πληροφορικής, ωστόσο είναι σύνηθες το φαινόμενο τα παιδιά να κατέχουν γνώσεις υπολογιστών από το σπίτι.

## **2.2 Οι υπολογιστές στα σχολεία**

Το 1998 ο Χρήστος Τόμπρας έγραφε στο ΒΗΜΑ «η εξοικείωση με τους υπολογιστές είναι πια απαραίτητη. Δεν θα θέλαμε βέβαια τα παιδιά μας να είναι οι ψηφιακοί αναλφάβητοι του αύριο. Υπολογιστές στα σχολεία, λοιπόν όσο πιο γρήγορα τόσο πιο καλά. Ολόκληρος ο δυτικός κόσμος σπεύδει να αναδιαμορφώσει τα εκπαιδευτικά προγράμματα όλων των βαθμίδων ώστε να συμπεριλάβει σε αυτά τους υπολογιστές, είτε ως αντικείμενο είτε ως εργαλείο διδασκαλίας. Και όμως, στις ΗΠΑ, όπου σπάνια πια συναντά κάποιος σχολείο χωρίς υπολογιστή και σύνδεση με το διαδίκτυο, έχουν αρχίσει να εκφράζονται οι πρώτες αντιρρήσεις» Σήμερα πλέον, η εκπαιδευτική διαδικασία στα σχολεία περιλαμβάνει από την πρώτη τάξη του Γυμνασίου το μάθημα της πληροφορικής όπως αναφέραμε και προηγουμένως.

Στην Ελλάδα καθυστέρησε αρκετά η εισαγωγή των υπολογιστών και των μαθημάτων της Πληροφορικής σε ολοκληρωμένη μορφή, καθώς κάποια σχολεία δε διέθεταν τον απαραίτητο εξοπλισμό ενώ κάποια άλλα δε διέθεταν καθηγητή πληροφορικής. Ωστόσο μέχρι σήμερα τα πράγματα έχουν αλλάξει αρκετά. Πολύ χαρακτηριστικό παράδειγμα είναι το 1ο Δημοτικό Σχολείο Μενεμένης. Στις 04/03/09 ηλεκτρονικοί υπολογιστές, προτζέκτορες και διαδραστικοί πίνακες αντικατέστησαν τον παραδοσιακό μαυροπίνακα, όπου δάσκαλοι και μαθητές φέρνουν σε πέρας το πρόγραμμα σπουδών με τη βοήθεια της νέας τεχνολογίας και του υποστηρικτικού λογισμικού, αφήνοντας πίσω τη συμβατική διδασκαλία.

Κάθε μία από τις εννέα αίθουσες του εξαθέσιου δημοτικού Μενεμένης διαθέτει, από την έναρξη της σχολικής χρονιάς, έναν ηλεκτρονικό υπολογιστή, συνδεδεμένο με δίκτυο γρήγορης πρόσβασης στο διαδίκτυο, έναν προτζέκτορα και έναν πίνακα μαρκαδόρου. Στην επιφάνειά του οι δάσκαλοι προβάλλουν το υλικό των μαθημάτων, εμπλουτισμένο κάθε φορά με ασκήσεις, παραδείγματα, εικόνες και άλλες πληροφορίες, που έχουν προετοιμάσει και συλλέξει από το Διαδίκτυο και το ειδικό υποστηρικτικό λογισμικό του Υπουργείου Παιδείας.

«Οι μαθητές έχουν μπροστά τους τα βιβλία, αλλά ταυτόχρονα παρακολουθούν την οπτική παρουσίαση του μαθήματος στην τάξη, όπως προτείνει κάθε φορά ο δάσκαλός τους», δήλωσε ο διευθυντής του σχολείου, Λευτέρης Λεωνίδης, ο οποίος είχε και την πρωτοβουλία εφαρμογής του νέου εξοπλισμού, σε συνεργασία με το Σύλλογο Διδασκόντων, τη Σχολική Επιτροπή και τον Σύλλογο Γονέων και Κηδεμόνων. Στον εξοπλισμό αυτό προστέθηκε πρόσφατα και ο διαδραστικός πίνακας, που με τη βοήθεια ενός σαρωτή-στιλό υπερήχων ενεργοποιεί την οθόνη του προτζέκτορα, δίνοντας τη δυνατότητα στους ίδιους τους μαθητές να αυτοσχεδιάζουν και να δημιουργούν πάνω σε αυτήν.

«Ο διαδραστικός πίνακας μετατρέπει την οθόνη προβολής σε ηλεκτρονικό πίνακα, όπως συμβαίνει με την οθόνη του υπολογιστή, με τη διαφορά ότι η χρήση του γίνεται ενώπιον όλης της τάξης», συμπλήρωσε ο κ. Λεωνίδης.



Οι μαθητές μπορούν να δημιουργήσουν κείμενα, να λύσουν ασκήσεις, να ζωγραφίσουν ή να συμπληρώσουν πληροφορίες στον διαδραστικό πίνακα, ενώ έχουν τη δυνατότητα να αποθηκεύσουν το κάθε μάθημα και να ανατρέξουν σε αυτό το αρχείο εάν χρειαστεί. Στη χρήση του νέου εξοπλισμού έχουν εκπαιδευτεί όλοι οι δάσκαλοι του σχολείου.

### **2.2.1 Οι υπολογιστές στα σχολεία ανά τον κόσμο**

Σε πολλές χώρες της Ευρώπης και κυρίως στις ΗΠΑ, οι υπολογιστές έχουν μπει σε κάθε σχολική τάξη και σχεδόν σε κάθε σπίτι. Η βιομηχανία εκπαιδευτικού λογισμικού αυξάνεται με ρυθμούς της τάξης του 20% ετησίως. Το πρόγραμμα Computer Literacy and Studies in Schools (CLASS) ξεκίνησε σε 250 σχολεία το 1984-85 στις ΗΠΑ. Επεκτάθηκε σε 500 ακόμα σχολεία το 1985-1986 και ακόμα 500 σχολεία το 1986-87. Σήμερα σε όλες τις αναπτυγμένες χώρες η πληροφορική αποτελεί μέρος του εκπαιδευτικού προγράμματος. Ωστόσο, στις υπό ανάπτυξη χώρες ακόμα γίνονται προσπάθειες τα σχολεία να εξελιχθούν. Σε αυτό το πλαίσιο κινείται και όλη η πρωτοβουλία OPLC.

### **2.2.2 Οι υπολογιστές στα σχολεία στην Ελλάδα**

Στην Ελλάδα τα τελευταία χρόνια έχουν γίνει προσπάθειες εκσυγχρονισμού της εκπαιδευτικής διαδικασίας και πλέον τα περισσότερα σχολεία διαθέτουν υπολογιστικά κέντρα. Ωστόσο, σε κάποια σχολεία παρατηρούνται βασικές ελλείψεις ή οι υπολογιστές που χρησιμοποιούνται είναι παλιάς τεχνολογίας συγκριτικά με την τεχνολογία του σήμερα. Το μάθημα της πληροφορικής αντιμετωπίζεται κάποιες φορές επιπόλαια λόγω του ότι είναι μάθημα επιλογής και οι μαθητές δεν εξετάζονται σε αυτό. Τέλος, σε κάποια σχολεία επικρατεί η αντίληψη ότι η γνώση χειρισμού του υπολογιστή περιορίζεται μόνο σε περιήγηση στο διαδίκτυο.

## **2.3 Πως φτάσαμε στο ΧΟ**

### **2.3.1 Οι πρώτοι υπολογιστές**

Ο πρώτος υπολογιστής γενικού σκοπού ήταν ο ENIAC (Electronic Numerical Integrator And Computer) και η κατασκευή του ολοκληρώθηκε το 1946. Ήταν Turing-πλήρης, ψηφιακός υπολογιστής με δυνατότητα επαναπρογραμματισμού για την επίλυση πλήρους φάσματος υπολογιστικών προβλημάτων. Ο ENIAC έπινε 63  $m^2$  και έκαιγε 150kW.

Έπειτα κατασκευάστηκαν υπολογιστές αναλόγων διαστάσεων και τεχνολογίας και στα τέλη του 1970 μέχρι και τα μέσα του 1980 περάσαμε στους προσωπικούς υπολογιστές. Οι δυνατότητες του προσωπικού υπολογιστή έχουν αλλάξει άρδην από την στιγμή που κατασκευάστηκαν. Στις αρχές του 1970 ακαδημαϊκοί και ερευνητικά κέντρα είχαν τη δυνατότητα για προσωπική χρήση υπολογιστικού συστήματος σε μορφή αλληλεπίδρασης για μεγάλη διάρκεια, παρότι αυτά τα συστήματα είχαν μεγάλο κόστος για να μπορέσει κάποιος να τα αγοράσει. Η εισαγωγή των μικροεπεξεργαστών, ένα ολοκληρωμένο κύκλωμα που περιέχει όλα τα κυκλώματα που προηγουμένως απαιτούσαν πολύ χώρο οδήγησε στην διάδοση των προσωπικών υπολογιστών γύρω στο 1975. Οι πρώτοι προσωπικοί υπολογιστές απευθύνονταν σε περιορισμένο κοινό. Τελικά, οι προσωπικοί υπολογιστές απέκτησαν

χαρακτηριστικά που τους καθιστούσαν κατάλληλους για χρήση από το ευρύ κοινό, όπως, γραφικά, ήχο παιχνίδια και λειτουργικά συστήματα φιλικά προς τον χρήστη.

### **2.3.2 Το laptop**

Το laptop είναι ένας προσωπικός φορητός υπολογιστής, τόσο μικρός ώστε να μπορεί να σταθεί στην αγκαλιά(=lap). Περιέχει όλα τα τυπικά μέρη ενός επιτραπέζιου υπολογιστή (οθόνη, πληκτρολόγιο, touchpad, ηχεία και μπαταρία). Στην αρχή τα laptops θεωρήθηκαν κατάλληλα για ειδικές εφαρμογές όπως π.χ. για τον στρατό. Σήμερα, στον επιχειρηματικό κόσμο υπάρχουν περισσότεροι φορητοί υπολογιστές απ' ότι επιτραπέζιοι και οι φορητοί γίνονται όλο και πιο απαραίτητοι για τους μαθητές αλλά και για γενική χρήση. Το 2008 πωλήθηκαν περισσότεροι φορητοί υπολογιστές απ' ότι επιτραπέζιοι στις Η.Π.Α.

Όταν το 1970 η χρήση του προσωπικού υπολογιστή φαινόταν πραγματοποιήσιμη, ακολούθησε η ιδέα για κατασκευή φορητού προσωπικού υπολογιστή. Ο Gavilan SC που κυκλοφόρησε το 1983 ήταν ο πρώτος φορητός υπολογιστής που χαρακτηρίστηκε ως laptop. Από το 1983 και μετά πολλές καινούριες τεχνικές ανακαλύφθηκαν και ενσωματώθηκαν στην τεχνολογία των φορητών ηλεκτρονικών υπολογιστών.

### **2.3.3 Το XO-1**

Το OLPC XO-1 (παλιότερα γνωστό σαν Hundred Dollar Laptop ή HDL, φορητός υπολογιστής των 100 δολαρίων) είναι ένας μικρός και φθηνός φορητός υπολογιστής που σχεδιάστηκε στο εργαστήριο Media Lab του MIT. Πρόκειται για έναν υπολογιστή ασύρματης ευρυζωνικής δικτύωσης, ο οποίος στο αρχικό του σχέδιο περιελάμβανε γεννήτρια (με μανιέλα) για να έχει αυτονομία. Μέσω ενός touchpad μπορεί ο χρήστης του, να γράφει πάνω σε αυτό με μολύβι ή και να ζωγραφίζει. Αποτελεί ένα μικρό ηλεκτρονικό βιβλίο και τετράδιο μαζί.

### **2.3.4 Το XO-2**

Στις 20 Μαΐου 2008 στο Global Country Workshop σε συνέδριο που έγινε, ο ιδρυτής της οργάνωσης OLPC, ο Nicholas Negreonte, ανακοίνωσε τη δεύτερη γενιά των OLPC XO laptops, που θα ονομάζονται XO-2. Ο Negreonte δεν μοιράστηκε πολλές λεπτομέρειες για το υλικό του XO-2, αλλά το νέο σύστημα θα έχει δύο οθόνες αφής. Το XO-2 θα είναι πολύ μικρότερο απ' ότι το αρχικό (περίπου μισό σύμφωνα με την έκδοση για τον τύπο) και θα έχει τη δυνατότητα να διπλώνει σαν e-book. Ο Negreonte είπε χαρακτηριστικά ότι ο φορητός υπολογιστής της επόμενης γενιάς θα έπρεπε να ήταν βιβλίο.

Το XO-2 θα έχει τη δυνατότητα για εναλλαγή της παρουσίασης για εσωτερικό και εξωτερικό χώρο, πρωτοπορία που ανακάλυψε η Mary Lou Jepsen που ήταν CTO του OLPC. Ο σχεδιασμός παρέχει δεξιά και αριστερή σελίδα σε κάθετη διάταξη, ένα περιστρεφόμενο laptop σε οριζόντια διάταξη, και μια επίπεδη συνεχόμενη επιφάνεια με 2 οθόνες για χρήση σε μορφή πλακέτας. «Τα μικρότερα παιδιά θα μπορούν να χρησιμοποιούν το απλό πληκτρολόγιο και τα μεγαλύτερα παιδιά θα μπορούν να εναλλάσσουν μεταξύ του πληκτρολογίου προσαρμοσμένου στις εφαρμογές καθώς και στο πληκτρολόγιο για πολλαπλές γλώσσες»

Επίσης το XO-2 θα μειώσει την κατανάλωση του στο 1 watt. Σύμφωνα με τον Negroni, το XO-2 έχει προγραμματιστεί να κυκλοφορήσει το 2010.





## **3** Γενικές Πληροφορίες για το OLPC

### **3.1 Τι είναι το OLPC**

Το OLPC είναι μια μικρή μηχανή με σημαντική αποστολή. Το XO είναι ένα ισχυρό μαθησιακό εργαλείο σχεδιασμένο και κατασκευασμένο ειδικά για παιδιά στις αναπτυσσόμενες χώρες, που ζουν στις πιο απομακρυσμένες περιοχές. Έχει περίπου το μέγεθος ενός μικρού βιβλίου. Έχει ενσωματωμένο ασύρματο δίκτυο και μια μοναδική οθόνη την οποία μπορούν να δουν τα παιδιά που πηγαίνουν σε υπαίθρια σχολεία ακόμα κι αν ο ήλιος χτυπάει ακριβώς πάνω σε αυτή. Είναι πολύ ανθεκτικό, λειτουργικό, ενεργειακά αποδοτικό και είναι ευχάριστο.

### **3.2 Πως ξεκίνησε το OLPC**

Το αρχικό πρωτότυπο του OLPC αποκαλύφθηκε από τον ιδρυτή του έργου, τον Nicholas Negroponte και στη συνέχεια από τον τότε Γενικό Γραμματέα των Ηνωμένων Εθνών Kofi Annan στις 16 Νοεμβρίου του 2001 στο World Summit on the Information Society (WSIS) στο Tunis της Τυνησίας. Ο Negroponte υπολόγισε ότι μόνο η οθόνη χρειαζόταν 3 ακόμα μήνες για την ανάπτυξη και εξέλιξή της. Το πλήρες λειτουργικό πρωτότυπο του έργου παρουσιάστηκε στο Country Task Force meeting στις 23 Μαΐου του 2006. Η έκδοση που θα παράγονταν θα είχε μεγαλύτερη οθόνη αλλά με ίδιο το μέγεθος του κουτιού.

Περίπου 400 boards ανάπτυξης διανεμήθηκαν στα μέσα του 2005, 8.752 λειτουργικά πρωτότυπα (Beta 78) παραδόθηκαν στα τέλη του 2006, 2400 Beta-2 διανεμήθηκαν στο τέλος του 2007 και η πλήρης παραγωγή ξεκίνησε στις 6 Νοεμβρίου του 2007. Η Quanta Computer, η εταιρία παραγωγής είπε τον Φεβρουάριο του 2007 ότι είχε παραγγελίες για ένα εκατομμύριο κομμάτια. Η Quanta υποστήριξε ότι μπορούσε να στείλει 5 με 10 εκατομμύρια κομμάτια εκείνη τη χρονιά γιατί επτά χώρες είχαν δεσμευτεί να αγοράσουν το XO-1 για τους μαθητές τους. Η Quanta έχει σκοπό να κυκλοφορήσει το XO-1 και στην ελεύθερη αγορά. Το OLPC project αρχικά δήλωσε ότι μια εμπορική έκδοση του XO δεν ήταν μέσα στα πλάνα της. Ωστόσο, το 2007 ίδρυσε το site [laptopgiving.org](http://laptopgiving.org) για δωρεές και

για “Give 1 Get 1” (πάρε 1 δώσε 1) προσφορές που ίσχυαν από τις 12 Νοεμβρίου του 2007 μέχρι και τις 31 Δεκεμβρίου του 2007. Για κάθε laptop που αγοράζόταν έναντι του ποσού των \$399, θα προσφερόταν ένα XO σε ένα παιδί σε μια αναπτυσσόμενη χώρα. Το OLPC έχει ξαναρχίσει το G1G1 πρόγραμμα μέσω του amazon.com. Στις 20 Μαΐου του 2008 η OLPC ανακοίνωσε την επόμενη γενιά του XO, το OLPC XO-2. Στα τέλη του 2008 το NYC Department of education ξεκίνησε ένα έργο για την αγορά μεγάλου αριθμού XO για χρήση από τους μαθητές της Νέας Υόρκης. Σύμφωνα με την Computer Aid International, σε μια δοκιμασία που έγινε τον Απρίλιο του 2009 το OLPC αξιολογήθηκε ως το χαμηλότερης κατανάλωσης σύστημα ανάμεσα σε άλλες πληροφοριακές και επικοινωνιακές τεχνολογίες που χρησιμοποιούνται στην εκπαίδευση.

### 3.3 Ιστορία του OLPC

Παρ’ όλο που η υλοποίηση του OLPC είναι σχετικά πρόσφατη, είναι μια ιδέα που προϋπήρχε εδώ και αρκετό καιρό. Οι διδάσκαλοι ήλπιζαν για την εισαγωγή της τεχνολογίας στα παιδιά σε όλο τον κόσμο και με την καθοδήγηση του Nicholas Negreponte, αυτό το όνειρο γίνεται γρήγορα πραγματικότητα.

Το 1982 ο Nicholas Negreponte και ο Seymour Papert έφεραν υπολογιστές Apple II σε σχολείο στο Dakar της Σενεγάλης. Αργότερα, ο Negreponte ίδρυσε το MIT Media Lab μαζί με τον Jerome Wisner. Στο MIT Media Lab, ο Papert δημιούργησε «Το σχολείο του μέλλοντος» που έφερε υπολογιστές στα σχολεία της Μασαχουσέτης όπου τα παιδιά έμαθαν να προγραμματίζουν σε Logo. Το 1988 που η LEGO/Logo εμπορευματοποιήθηκε, δόθηκε στα παιδιά η ευκαιρία να προγραμματίσουν self-made ρομπότ.

Το 2002, ο Negreponte πήγε σε ένα μικρό χωριό της Καμπότζης και έδωσε σε 20 παιδιά laptops με σύνδεση στο διαδίκτυο. Σημείωσε πως τα παιδιά και οι οικογένειές τους σύντομα ανακάλυψαν πολλαπλές χρήσεις για το μηχάνημα και πόσο εύκολα έμαθαν να πλοηγούνται στο διαδίκτυο. Στο μεταξύ ο Papert έπεισε τον κυβερνήτη Angus King του Maine ότι η ένα προς ένα είναι η μόνη αναλογία που έχει νόημα για την παράταξη των υπολογιστών στους μαθητές. Ως αποτέλεσμα, 42.000 laptops διανεμήθηκαν σε επίπεδο πολιτειών σε μαθητές της έβδομης τάξης.

### 3.4 Τα πρώτα στάδια του OLPC

Η πρώτη beta έκδοση του laptop, B1-Test κυκλοφόρησε στο τέλος του 2006. Τρεις ακόμα beta εκδόσεις αναδείχτηκαν στη διάρκεια του χρόνου μέχρι την τελική παραγωγή στο τέλος του 2007. Οι πολλές σύντομες επαναλήψεις βοήθησαν πολύ το έργο. Δόθηκε η δυνατότητα να γίνουν δοκιμές με χρήστες και να διαπιστώσουν τι δούλευε και τι όχι. Αξίζει να σημειωθεί ότι το B1-Test είχε 128mb περισσότερη ram και ξόδευε περισσότερη ενέργεια απ’ ότι η τελική έκδοση λόγω της διαρροής μνήμης στο λογισμικό και στη μη διαθεσιμότητα του hardware.

### 3.5 Κόστος του OLPC

Στις αρχές του 2005, ορμώμενος από την επιτυχία του με τη διανομή των laptops σε παιδιά αναπτυσσόμενων χωρών, ο Negreonte είχε την αρχική ιδέα για την κατασκευή ενός laptop με κόστος \$100. Παρουσίασε την ιδέα του στο World Economic Forum, όπου και έτυχε θερμής αποδοχής. Ο John Markoff της εφημερίδας New York Times αποκάλεσε τον Negreonte τον «Johnny Appleseed της ψηφιακής εποχής». Ο Johnny Appleseed, γεννημένος ως John Chapman (Σεπτέμβριος 26, 1774 - Φεβρουάριος 18, 1845), ήταν ένας πρωτοπόρος Αμερικανός δάσκαλος που παρουσίασε τις μηλιές σε μεγάλα μέρη του Ohio, της Indiana, και του Illinois. Έγινε θρύλος στην Αμερική όσο βρισκόταν στη ζωή κυρίως λόγω των ευγενικών και γενναιόδωρων τρόπων του, τις εξαιρετικές του ηγετικές ικανότητες σε συζητήσεις και τη συμβολική σημασία των μήλων.

Η ιδέα του Negreonte κέρδισε πολύτιμο έδαφος αφού υποστηρίχθηκε από μεγάλες εταιρίες όπως η AMD, η News Corp., η Google, η Red Hat, η Quanta Computers και η Nortel. Επιπλέον, μέχρι τα τέλη του 2005, ο Negreonte είχε ήδη έρθει σε επαφή με περισσότερες από 50 χώρες και είχε κερδίσει πολλούς υποστηρικτές. Ένας από αυτούς ήταν ο πρόεδρος Olusegun Obasanjo της Νιγηρίας που δεσμεύτηκε να αγοράσει ένα εκατομμύριο κομμάτια.

### 3.6 Γιατί το OLPC χρησιμοποιεί προγράμματα ανοιχτού λογισμικού

Το laptop θα φέρει τα παιδιά σε επαφή με την τεχνολογία ως μέσω ελευθερίας και ενίσχυσης των δυνατοτήτων. Η επιτυχία του έργου όσον αφορά την παγκοσμιότητα είναι εφικτή μόνο αν ενστερνιστεί το ανοιχτό λογισμικό παρέχοντας στους χρήστες του laptop αλλά και στους σχεδιαστές εμβριθή ελευθερία.

Όσο τα παιδιά μεγαλώνουν και αναζητούν καινούριες ιδέες, το λογισμικό και τα εργαλεία θα πρέπει να είναι ικανά να μεγαλώνουν με αυτά και να τους παρέχουν μια διέξοδο στην τεχνολογία. Για να επιτευχθούν όλα αυτά και άλλοι πρακτικοί στόχοι αλλά και για να αντεπεξέλθει το έργο στις αρχές πάνω στις οποίες έχει βασιστεί η επιτυχία του, υπάρχει η πεποίθηση ότι η λογισμική πλατφόρμα του έργου του OLPC:

- Πρέπει να περιλαμβάνει πηγαίο κώδικα και να επιτρέπει τις τροποποιήσεις έτσι ώστε οι σχεδιαστές, οι κυβερνήσεις που είναι οι πελάτες αλλά και τα παιδιά που είναι οι χρήστες του laptop να μπορούν να εμβαθύνουν με τη δυνατότητα να αλλάξουν το λογισμικό ώστε να ικανοποιεί ασύλληπτες ανάγκες και ασύλληπτα ποικίλες ανάγκες. Το λογισμικό επίσης πρέπει να παρέχει στον χρήστη μια πλατφόρμα ανάπτυξης self-hosting.
- Πρέπει να επιτρέπει τη διανομή διαφοροποιημένων αντιγράφων του λογισμικού υπό την ίδια άδεια έτσι ώστε οι ελευθερίες πάνω στις οποίες βασίζονται οι σχεδιαστές για την επιτυχία τους να παραμείνουν διαθέσιμες στους χρήστες και σχεδιαστές του λογισμικού την επόμενη γενιάς. Οι χρήστες και οι πελάτες πρέπει να μπορούν να εντοπίσουν λογισμικό στη γλώσσα τους, να διορθώσουν το λογισμικό για να αφαιρέσουν λάθη και να τροποποιήσουν το λογισμικό ώστε να ανταποκρίνεται στις ανάγκες τους.

- Πρέπει να επιτρέπει επαναδιανομή χωρίς άδεια, είτε ατομικά είτε ως κομμάτι μιας συγκεντρωτικής διανομής, γιατί δεν είναι δυνατόν να γνωρίζουμε και δε θα έπρεπε να ελέγχεται ο τρόπος που τα εργαλεία που αναπτύσσονται θα επαναπροσδιοριστούν στο μέλλον. Τα παιδιά ξεπερνούν σε ανάπτυξη την πλατφόρμα, οπότε πρέπει να είναι ευέλικτη να μεγαλώσει με αυτά.
- Δε πρέπει να απαιτεί πληρωμή πνευματικών δικαιωμάτων ή οποιασδήποτε άλλης πληρωμής για επαναδιανομή ή τροποποίηση για προφανείς λόγους οικονομίας και παραδειγματισμού στα πλαίσια του έργου του OLPC.
- Δεν πρέπει να κάνει διακρίσεις μεταξύ προσώπων, ομάδων ή πεδίων προσπάθειας. Η δύναμη του λογισμικού είναι η ικανότητα του να μεγαλώνει και να αλλάζει παράλληλα με τα παιδιά.
- Δεν πρέπει να βάζει περιορισμούς σε άλλα λογισμικά που μπορεί να διανέμονται παράλληλα. Οι άδειες λογισμικού δεν πρέπει να περιορίζονται ούτε από ιδιοκτησία ούτε από "copyright" λογισμικό λόγω του ότι διανέμεται στην πλατφόρμα. Ένας ολόκληρος κόσμος από εξαιρετικό λογισμικό θα χρησιμοποιηθεί για να επιτύχει αυτό το έργο - και ανοιχτό και κλειστό.
- Πρέπει να δίνουμε τη δυνατότητα να μπορούν να επιλέξουν από όλα.
- Πρέπει να επιτρέψουμε αυτά τα δικαιώματα να διαδίδονται μαζί με το λογισμικό. Αυτό σημαίνει ότι δεν πρέπει να παρέχουμε μια άδεια συγκεκριμένα για το έργο του laptop των \$100 ή για την οργάνωση ή για τους πελάτες. Ενώ είμαστε οι σχεδιαστές αυτής της πλατφόρμας σήμερα, οι χρήστες είναι οι σχεδιαστές του αύριο και μέσω αυτών η πλατφόρμα θα έχει επιτυχία, θα τροποποιηθεί και θα πορευτεί στο χρόνο. Πρέπει να έχουν τα ίδια δικαιώματα με τους σχεδιαστές.
- Δεν πρέπει να παρακωλύεται από πλατφόρμες λογισμικού που περιορίζουν τις τροποποιήσεις ή χρησιμοποιούνται όπως περιγράφηκε παραπάνω.
- Δεν πρέπει να παρακωλύεται από πατέντες λογισμικού που απαγορεύουν τις τροποποιήσεις ή τη χρήση με τους τρόπους που περιγράφονται παραπάνω. Όλες οι πατέντες που χρησιμοποιούνται από το λογισμικό πρέπει να μπορούν να κυκλοφορήσουν υπό κάποια άδεια και να επιτρέψουν στους χρήστες μας να χρησιμοποιήσουν ή να πουλήσουν παράγωγες εκδοχές που εφαρμόζουν την υπό συζήτηση πατέντα.
- Πρέπει να υποστηρίζει και να προωθεί ανοιχτή, πατενταρισμένη και απαρακώλυτη ανταλλαγή δεδομένων και αρχείων.
- Πρέπει να μπορεί να αναπτύσσεται χρησιμοποιώντας απαρακώλυτα εργαλεία (π.χ. μεταφραστές)

Το λογισμικό του XO προς το παρόν δεν ικανοποιεί όλους αυτούς τους στόχους. Για παράδειγμα, δεν μπορεί με κανέναν τρόπο να είναι self-hosted. Δεν έχει ποτέ κανείς χρησιμοποιήσει το XO, για να επανακατασκευάσει λογισμικό για ένα XO, παρόλο που είναι θεωρητικά εφικτό. Επίσης, το λογισμικό φτάνει με κλειδώματα που είναι πολύ δύσκολο να προσπεράσει κάποιος σε περίπτωση που η οργάνωση OLPC διαλυόταν (ή άλλαζε



τη γνώμη της για την ελευθερία του λογισμικού). Αυτά τα σημεία που είναι κλειδωμένα εμποδίζουν τους χρήστες να εγκαταστήσουν οποιοδήποτε λογισμικό επιθυμούν στο υλικό τους.

### 3.7 Windows στο OLPC

Το XO έχει δυνατότητα για dual-boot. Μπορεί να φορτώσει τα Windows XP ή οποιαδήποτε έκδοση Linux συμπεριλαμβάνοντας το ίδιο το σύστημα του OLPC που αναδεικνύεται από το παραθυρικό περιβάλλον Sugar στα Fedora Linux. Από τον Ιούλιο του 2009, καμία μαζική παραγωγή του OLPC δεν τρέχει Windows. Η Microsoft έχει αναπτύξει μια έκδοση των Windows XP που τρέχει στο OLPC laptop. Σύμφωνα με ένα άρθρο της ITWorld, ο Bill Gates είπε ότι η Microsoft σχεδιάζει να προσφέρει ένα πακέτο λογισμικού αξίας \$3 για το XO που θα αποτελείται από:

- Windows XP
- Microsoft Office Home and Student 2007
- Microsoft Math 3.0
- Learning Essentials 2.0 for Microsoft Office
- Windows Live Mail

Αυτή η προσφορά θα εφαρμοστεί μόνο στις αναπτυσσόμενες χώρες. Δεν δόθηκαν λεπτομέρειες περί της διαθεσιμότητας, τις συνθήκες της άδειας χρήσης ή της γλώσσας υποστήριξης που περιλαμβάνεται ή είναι διαθέσιμη γι' αυτό το λογισμικό. Το εύρος των γλωσσών που υποστηρίζονται φαίνεται να είναι πολύ περιορισμένο. Σε ένα άρθρο της DesktopLinux ο Eric Brown έγραψε: «Η έκδοση των Windows XP που θα εγκατασταθεί στο XO υποστηρίζει όλα τα χαρακτηριστικά του υπολογιστή, συμπεριλαμβανομένων της δικτύωσης, τα ηχεία, το μικρόφωνο, την κάμερα. Επιπλέον, επιτρέπει στην οθόνη να περιστρέφεται σε μορφή e-book και να αλλάζει σε μορφή μονόχρωμης οθόνη με δυνατότητα ανάγνωσης στον ήλιο, σύμφωνα με τη Microsoft». Ο James Utzschneider, ένας GM του marketing και της επικοινωνίας της Microsoft's Unlimited Potential group, έγραψε σε ένα blog ότι «η έκδοση των Windows για το XO είναι μια προσαρμοσμένη έτσι ώστε να μην αποκόπτει χαρακτηριστικά ή λειτουργίες με σκοπό να λειτουργεί στην περιορισμένη μνήμη και τον περιορισμένο αποθηκευτικό χώρο που προσφέρει το XO.» Ωστόσο να σημειώσουμε ότι τα Windows δεν υποστηρίζουν το περιβάλλον Sugar για ευνόητους λόγους.





## 4 Ειδικότερα για το OLPC

### 4.1 Hardware στο OLPC

Το XO-1 laptop είναι το βασικό μέλημα της οργάνωσης One Laptop Per Child. Ύστερα από τρία χρόνια ανάπτυξης, πλέον παράγεται μαζικά με εκατοντάδες χιλιάδες κομμάτια να κατασκευάζονται και χιλιάδες ακόμα να διατίθενται για δοκιμή στα σχολεία σε όλον τον κόσμο. Το Hardware του XO-1 είναι το εξής:

#### 1. Επεξεργαστής και πυρήνας του συστήματος:

- AMD LX700 CPU (433 MHz) με ολοκληρωμένη μονάδα επεξεργασίας γραφικών.
- AMD CS5536 ομοειδές ολοκληρωμένο κύκλωμα για περιφερειακές I/O (είσοδος/έξοδος).
- 256 MByte DDR SDRAM ολοκληρωμένο κύκλωμα συστήματος μνήμης που τρέχει στα 333MHz.
- Εμφωλευμένος ελεγκτής για παρακολούθηση του συστήματος.
- ISA Συμβατότητα: Υποστηρίζει και MMX και 3DNow! X86 instruction-set extensions.

#### 2. Αποθηκευτικοί χώροι:

- 1 GB NAND Flash μνήμη στη μητρική.
- 1 MB σειριακή Flash μνήμη που παρέχεται ξεχωριστά για firmware.
- Επεκτάσιμη μνήμη μέσω single SD/MMC module socket.

#### 3. Ήχος:

- AC' 97 ηχητικό υποσύστημα.
- Εσωτερικά ηχεία και ενισχυτής.
- Εσωτερικό μονοφωνικό μικρόφωνο.
- Υποδοχή για εξωτερικά ακουστικά.
- Υποδοχή για εξωτερικά μονοφωνικά, μικρόφωνο, αισθητήρες ή διακόπτες.

**4. Οθόνη:**

- 7.5in (19 cm) έγχρωμη/ασπρόμαυρη TFT LCD, 1200x900 (200dpi).
- Διαστάσεις: 152.4 mm x 114.3 mm 6 in x 4.5 in.
- Δυνατότητα ανάγνωσης κάτω από τον ήλιο.
- Ελεγκτής οθόνης που υποστηρίζει τη χρήση της οθόνης ακόμη κι όταν η CPU δεν λειτουργεί.
- Solid state (LED) backlight.

**5. Κάμερα:**

- Ενσωματωμένη έγχρωμη κάμερα.
- 640 x 480 ανάλυση.
- Full frame rate (30fps).

**6. Ασύρματο δίκτυο:**

- Ενσωματωμένη IEEE 802.11/b/g (2.4 GHz) διεπαφή ασύρματης δικτύωσης.
- Υποστηρίζει Mesh δικτύωση (variant of 802.11s).
- Δυνατότητα λειτουργίας της δικτύωσης ακόμα κι όταν η CPU είναι ανενεργή.
- CL1 Hardware Design Specification.
- Rev. 1.04, 9/29/08.
- Διπλές κεραιές ενσωματωμένες στο laptop.

**7. Θύρες επέκτασης:**

- Τρεις εξωτερικές θύρες USB 2.0.

**8. Συσκευές εισόδου:**

- Αδιάβροχο και αδιάβλητο στη σκόνη πληκτρολόγιο με 80+ πλαστικά κουμπιά, με 1mm πάτημα.
- Ικανό touchpad που χρησιμοποιείται ως συσκευή δείκτη.

**9. Κουμπιά:**

- Κουμπί εκκίνησης δίπλα στην οθόνη.
- Κουμπί περιστροφής οθόνης στην άκρη της οθόνης.

- Δύο σειρές κουμπιών ελέγχου τεσσάρων διαστάσεων.
- Μαγνητικός αισθητήρας που εντοπίζει το κλείσιμο του laptop.
- Μαγνητικός αισθητήρας που εντοπίζει τη χρήση σε μορφή eBook.

#### 10. Δείκτες:

- Power on/off LED.
- LED κατάστασης μπαταρίας.
- Δύο LEDs για την κατάσταση του ασύρματου.
- LED που ελέγχεται από το υλικό για το μικρόφωνο αν είναι σε χρήση.
- LED που ελέγχεται από το υλικό αν η κάμερα είναι σε χρήση.

#### 11. Μετασχηματιστές τροφοδοσίας:

- DC power input, από 11V σε 18V, περιορισμένο εσωτερικά για άντληση 15W.
- Ενσωματωμένο φορτιστή για Ni-MH/ LiFePO4 μπαταρίες.
- LED Backlight σύστημα ελέγχου υψηλής αποδοτικότητας control circuit.
- mm power input connector (1.65 mm center pin).

#### 12. Γενική ασφάλεια:

- IEC 60950-1 qualified.
- EN 60950-1 qualified.
- CSA/UL 60950-1 qualified.
- ASTM F 963 - Standard Consumer Spec on Toy Safety qualified.

#### 13. Μπαταρία:

- serial Ni-MH AF τύπος μπαταρίας με ενσωματωμένο gas gauge IC.
- ή 4 serial LiFePO4 AF τύπος μπαταρίας με ενσωματωμένο gas gauge IC.
- Ασφάλεια: UL 1642 qualified

#### 14. AC adapter:

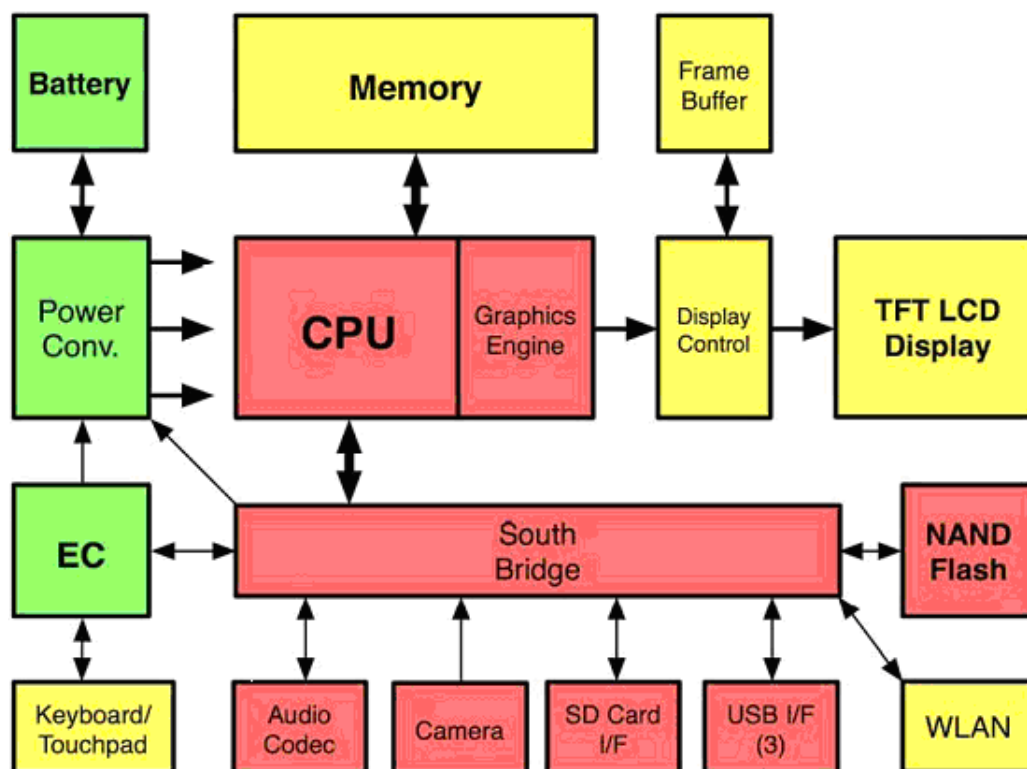
- CL1 αρχιτεκτονική.
- Rev. 1.04, 9/29/08.
- 17W external AC adapter.
- 90v(-10%) 240v(+25%), 35-60Hz, AC input.
- 12V DC output.
- Safety: IEC 60950-1, EN 60950-1 CSA/UL 60950-1 qualified.
- Extra Transient and Burst Immunity: IEC 61000-4-4 passed.
- Extra Surge Immunity: IEC 61000-4-5 passed.
- Βάρος: <1.5KG.
- Μέγεθος: 245mm (9.65") 230mm (9") 30.5mm (1.2").

### 4.1.1 CL1 Σύστημα αρχιτεκτονικής

Η αρχιτεκτονική CL1 διαφέρει από τον σχεδιασμό της αρχιτεκτονικής των συμβατικών φορητών υπολογιστών, στο ότι μεγάλο μέρος του υπολογιστή (συμπεριλαμβανομένου του CU) μπορούν να μην τροφοδοτούνται καθόλου χωρίς να επηρεάζεται η λειτουργία των υπόλοιπων λειτουργιών. Η CL1 σχεδιάστηκε με σκοπό να τρέχει Linux συστήματα ανοιχτού λογισμικού, παρ' όλο που μπορεί να υποστηρίξει και άλλα συστήματα.

#### Διάγραμμα συστήματος block

Στην παρακάτω εικόνα παρατίθεται ένα σχεδιάγραμμα της CL1 αρχιτεκτονικής. Σε αντίθεση με τους περισσότερους φορητούς υπολογιστές όλα τα συστήματα εκτός από το πληκτρολόγιο και το touchpad βρίσκονται πίσω από την LCD οθόνη. Αυτό ονομάζεται Core Module. Τα κυκλώματα του πληκτρολογίου, ο κωδικοποιητής του πληκτρολογίου, το εξάρτημα του touchpad και η μπαταρία συνδυάζονται μεταξύ τους σε μια ομάδα που ονομάζεται Input Module και συνδέονται στον πυρήνα μέσω δύο μικρών καλωδίων στο εσωτερικό του υπολογιστή.



Σχήμα 4.1: Το CL1 σύστημα Αρχιτεκτονικής

#### CPU & Companion Chip

Το σύστημα επεξεργασίας του πυρήνα που χρησιμοποιείται στη CL1 αρχιτεκτονική έχει έναν ενσωματωμένο x86 επεξεργαστή που συνδέεται με μια μονάδα διαχείρισης μνή-

μης που είναι συμβατό με x87 μονάδα κινητής υποδιαστολής, μια μονάδα επεξεργασίας γραφικών. Ο ενσωματωμένος χειριστής μνήμης παρέχει ένα ενιαίο μοντέλο μνήμης στον επεξεργαστή, τον επεξεργαστή γραφικών και τον ενσωματωμένο ελεγκτή της οθόνης. Ένα ολοκληρωμένο “South Bridge” ενσωματώνει πολλές περιφερειακές λειτουργίες συστήματος διαχείρισης. Αυτά περιλαμβάνουν μια διεπαφή Audio Codec (AC '97) στο σύστημα ήχου, low pin count (LPC), μονάδα διαχείρισης της ενέργειας και legacy PC system components όπως real time clock (RTC), memory transfer engines (DMA), και programmable interrupt controllers. Η συχνότητα του ρολογιού στον πυρήνα της CPU μπορεί να επιλεγεί με χρήση αλτών υλικού και μπορεί να ρυθμιστεί από το λογισμικό κατά της διάρκεια της φόρτωσης του firmware. Το ρολόι που χρησιμοποιείται από τον κύριο επεξεργαστή και τη μνήμη διαμοιράζεται φασματικά για να μειωθεί η ηλεκτρομαγνητική παρεμβολή. Ένα παράδειγμα συστήματος επεξεργασία πυρήνα που χρησιμοποιείται στη CL1 είναι ένας AMD Geode LX700 (που τρέχει στα 433MHz), κι ένας AMD CS5536 Southbridge.

### **Διακόπτες**

Αφιερωμένοι στο σύστημα διακόπτες παρέχονται για το ρολόι, το πληκτρολόγιο το AC '97, την SD κάρτα, την κάμερα, το RTC, τον ελεγκτή του συστήματος, τη θύρα USB (που χρησιμοποιείται και για δικτύωση), τον έλεγχο την οθόνης, το touchpad, τον επεξεργαστή των μαθηματικών και το DMA.

### **Firmware**

Η διεπαφή του Low Pin Count (LPC), που παρέχεται από την Southbridge και χρησιμοποιείται για να επικοινωνήσει με το ενσωματωμένο σύστημα ελέγχου (EC), χρησιμοποιείται επίσης και από τον κύριο επεξεργαστή για το φορτώσει το firmware του υπολογιστή. Η EC's LPC διεπαφή με τη σειρά της διαβάζει το firmware από τη σειριακή διεπαφή μνήμης Flash.

### **Μονάδα επεξεργασίας γραφικών**

Μια μονάδα επεξεργασίας γραφικών δύο διαστάσεων (GPU) παρέχεται από το CL1 για να επιταχύνει τα κοινά γραφικά και τις λειτουργίες βίντεο. Αυτή η μονάδα μοιράζεται την κύρια μνήμη με τον κύριο επεξεργαστή, χρησιμοποιώντας αρχιτεκτονική ενοποιημένης μνήμης. Παρέχει ξεχωριστό κανάλι επεξεργασίας για βίντεο και γραφικά, και συνδυάζονται με το υλικό πριν εμφανιστούν στην οθόνη. Για κάθε αγωγό παρέχονται ξεχωριστοί μεταλλάκτες χρώματος-χώρου και διακλιμάκωσης.

### **Μνήμη**

Ο CL1 επεξεργαστής υποστηρίζει ενιαία 64 bit εύρους DDR1 SDRAM διεπαφή και δε χρησιμοποιεί εξωτερικό buffering. Επιπλέον, τέσσερις 16-bit εύρους DDR SDRAM ICs φορτώνονται κατ' ευθείαν στη μητρική και υποστηρίζουν 256 MBytes κύριας μνήμης που χρησιμοποιεί 32M x 16 (512 Mbit) στοιχεία. Η ταχύτητα της μνήμης επιλέγεται στη διάρκεια φόρτωσης από το firmware, και με τη χρήση των παρόντων ολοκληρωμένων κυκλωμάτων μπορεί να είναι είτε DDR-266 είτε DDR-333. Αφού η μνήμη φορτώνεται

στη μητρική και δεν είναι επεκτάσιμη, δεν υπάρχει SPD διεπαφή για τον εντοπισμό της ταχύτητας ή του μεγέθους την μνήμης.

### **Power Conversion**

Το υποσύστημα τροφοδοσίας που έχει σχεδιαστεί για το CL1 είναι πολύπλοκο σε σχέση με έναν συμβατικό φορητό υπολογιστή. Με σκοπό την ελαχιστοποίηση την κατανάλωσης ενέργειας, η ενέργεια που παρέχεται σε έναν αριθμό υποσυστημάτων μπορεί να αλλάζει από on σε off ανάλογα με τις ανάγκες που υπάρχουν. Τα στοιχεία που είναι με πράσινο στην εικόνα 4.1 είναι αυτά που τροφοδοτούνται συνέχεια (καταναλώνοντας microamps αν ο υπολογιστής δε χρησιμοποιείται). Τα υποσυστήματα με κίτρινο χρώμα (η οθόνη, το ασύρματο δίκτυο και η κύρια μνήμη) μπορούν προαιρετικά να τροφοδοτούνται συνέχεια. Τα υπόλοιπα στοιχεία, με χρώμα κόκκινο τροφοδοτούνται μόνο όταν χρειάζεται να γίνουν υπολογισμοί ή για να αντιδράσουν σε μια αίτηση του χρήστη. Τα υποσυστήματα με κίτρινο χρώμα μπορούν να τροφοδοτηθούν ανεξάρτητα το ένα από το άλλο. Είναι δυνατό να λειτουργεί η οθόνη χωρίς να λειτουργεί το ασύρματο δίκτυο ή να λειτουργεί το ασύρματο δίκτυο χωρίς να λειτουργεί τίποτε άλλο στον υπολογιστή. Πολλές από τις περιφερειακές συσκευές με κόκκινο χρώμα, όπως η κάμερα, το σύστημα ήχου και οι διεπαφές των USB θυρών μπορούν ξεχωριστά να μην τροφοδοτούνται για να εξοικονομούν ενέργεια όταν δεν χρησιμοποιούνται.

### **Φορτιστής μπαταρίας**

Το CL1 παρέχει ένα DC power βύσμα. Μια DC πηγή ενέργειας μεταξύ 11 και 18 Volts είναι κατάλληλη για την τροφοδοσία του υπολογιστή. Το βύσμα μπορεί με ασφάλεια να αντέξει την είσοδο μεταξύ 32 και 40 Volts. Αν υπερβούν οι παράμετροι τροφοδοσίας δε θα προκληθεί μόνιμη βλάβη στον υπολογιστή, αλλά θα καταστραφεί μια τηκτασφάλεια και θα απαιτείται επισκευή. Ένας ενσωματωμένος φορτιστής μπαταρίας παρέχεται, που μετατρέπει την ενέργεια που εισέρχεται στον υπολογιστή στα κατάλληλα volts για φόρτιση και ελέγχει την παρούσα κατάσταση φόρτισης. Ο φορτιστής υποστηρίζει συνεχόμενη τάση και αλγόριθμους συνεχούς φόρτισης υπό τον έλεγχο του ενσωματωμένου ελεγκτή που τρέχει υπό το firmware. Ο φορτιστής είναι σχεδιασμένο να περιορίζει στο υλικό την ενέργεια που αντλεί από την εξωτερική είσοδο. Η συνολική τάση περιορίζεται στα 15 W. Αν τα στοιχεία του υπολογιστή αντλούν λιγότερο από αυτό το ποσό, η υπόλοιπη ενέργεια χρησιμοποιείται για να φορτίσει την μπαταρία. Το μέγιστο ποσό ρεύματος μπορεί να καθορισθεί από το λογισμικό να είναι είτε 0.3A είτε 2A. Τα υποστηριζόμενα volts φόρτισης είναι είτε 7.6V για for NiM μπαταρία είτε 6.4V για LiFePo μπαταρία.

### **Μπαταρία**

Το CL1 είναι σχεδιασμένο για να υποστηρίζει διάφορες τεχνολογίες μπαταριών. Προς το παρόν, οι υπολογιστές είναι διαθέσιμοι με μπαταρίες που απαρτίζονται είτε από πέντε NiMH κελιά σε σειρές είτε δύο/τέσσερα LiFePO κελιά σε σειρές. Κάθε μπαταρία περιέχει ένα ενσωματωμένο αέριο gauge IC. Η μπαταρία μπορεί να επαναφορτιστεί 2000 φορές (στο 50% της δυνατότητάς της, αφού η επαναλαμβανόμενη φόρτιση και αποφόρτιση μειώνει σταδιακά την ποσότητα της ενέργειας που μπορεί να αποθηκεύσει η μπαταρία).





Σχήμα 4.2: Ο φορτιστής της μπαταρίας του OLPC

Ωστόσο, αυτές οι μπαταρίες διαρκούν περίπου δεκατέσσερις φορές περισσότερο από μια συμβατική μπαταρία φορητού υπολογιστή που τυπικά καθορίζεται επαναφορτίσιμη 500 φορές (στο 50% της δυνατότητάς της). Η επικοινωνία μεταξύ του υπολογιστή και του ολοκληρωμένου gas gauge γίνεται μεταξύ ενός μόνο καλωδίου με χρήση των προδιαγραφών του Semiconductor One-Wire πρωτοκόλλου. Αυτή η διεπαφή του ενός μόνο καλωδίου επιτρέπει στον ενσωματωμένο ελεγκτή του υπολογιστή να διαβάζει πληροφορίες για την τεχνολογία της μπαταρίας, την κατάσταση και το ιστορικό της. Επιτρέπει επίσης, στον υπολογιστή να τροφοδοτήσει το ολοκληρωμένο gas gauge σε μπαταρίες που έχουν αποφορτιστεί τελείως. Ένας διασυνδέτης τριών pins χρησιμοποιείται μεταξύ της μπαταρίας και του CL1.



Σχήμα 4.3: Η μπαταρία του OLPC

### **Backlight control**

Το CL1 LCD backlight παρέχεται από άσπρα LEDs. Πολλαπλά LEDs συνδέονται σε σειρά πάνω σε τρεις αλυσίδες για να φέρουν σε ισορροπία την φωτεινότητα. Αυτές οι αλυσίδες τροφοδοτούνται παράλληλα. Τα LEDs και από τις τρεις αλυσίδες πρέπει να πολυπλέκονται, για να ελαχιστοποιείται ο αντίκτυπος όταν καίγεται μια αλυσίδα. Εώς 60 mA (στα 17.6V) παρέχεται για να οδηγήσει το backlight. Δύο σήματα χρησιμοποιούνται για να ελέγξουν την λειτουργία του backlight. Ένα ενεργοποιεί/απενεργοποιεί το backlight, κι ένα δεύτερο είναι pulse-width modulated στα 200Hz για να παράγει μια τάση που παρέχει μια ποικιλία για τη φωτεινότητα του backlight.

### **Power Timing Sequence**

Όποτε μια μπαταρία εισέρχεται στο CL1, ή παρέχεται ρεύμα στην είσοδο DC το ενσωματωμένο σύστημα ελέγχου τροφοδοτείται και λειτουργεί.

### **Power Off to On**

Αν πατηθεί το κουμπί για να ανοίξει ο υπολογιστής, το σύστημα ελέγχου παρέχει ενέργεια στη λογική μονάδα διαχείρισης στη South bridge, και ενημερώνει ότι πατήθηκε το κουμπί. Η South bridge θα επιτρέψει τη τροφοδοσία της κύριας μνήμης, του επεξεργαστή και των περισσότερων στοιχείων του συστήματος. Η μονάδα ελέγχου του συστήματος παρατηρεί αυτές τις ενέργειες και ενεργοποιεί τον ελεγκτή της οθόνης σε απάντηση. Ο ελεγκτής της οθόνης ενεργοποιεί την LCD οθόνη και το backlight.

### **On to Suspend**

Αν το λειτουργικό σύστημα καθορίσει ότι πρέπει να ανασταλεί η λειτουργία της κεντρικής μονάδας επεξεργασίας για εξοικονόμηση ενέργειας ακολουθεί τα απαραίτητα βήματα για να απενεργοποιήσει τις λειτουργίες των υπό-επεξεργαστών που μπορεί να έχουν πρόσβαση στην κύρια μνήμη, ενημερώνει τη διεπαφή της mesh δικτύωσης ότι πρέπει να απομονωθεί από τη USB δίοδο, τοποθετεί την κύρια μνήμη σε λειτουργία χαμηλής κατανάλωσης και αυτο-ανανέωσης, και τότε ενημερώνει τη λογική μονάδα διαχείρισης South bridge. Αυτή η λογική μονάδα θα απενεργοποιήσει την τροφοδοσία από τον επεξεργαστή και τα περισσότερα στοιχεία του συστήματος. Μέρη του South bridge, η κύρια μνήμη, η οθόνη, ο ελεγκτής της οθόνης, το backlight, το πληκτρολόγιο και η διεπαφή mesh δικτύωσης μπορούν να παραμείνουν ενεργοποιημένα προαιρετικά.

### **Suspend to On**

Όταν εντοπιστεί οποιοδήποτε πάτημα πλήκτρου, είτε από το πληκτρολόγιο είτε από τα κουμπιά γύρω από την οθόνη (εκκίνησης, περιστροφής και τα κουμπιά των παιχνιδιών), ο ελεγκτής του συστήματος ενημερώνει τον ελεγκτή της οθόνης και τη μονάδα διαχείρισης ενέργειας στη Southbridge. Ο ελεγκτής της οθόνης ενεργοποιεί αμέσως την οθόνη (αν είναι κενή λόγω της αδράνειας) η μονάδα διαχείρισης της ενέργειας τροφοδοτεί τον επεξεργαστή και τα περισσότερα υπόλοιπα στοιχεία.

## On to Off

Όταν το λειτουργικό σύστημα καθορίσει ότι ο υπολογιστής πρέπει να σβήσει (συνήθως ως αντίδραση στο πάτημα του κουμπιού τροφοδοσίας), στέλνει μια εντολή στον EC. Ο EC απενεργοποιεί την τροφοδοσία σε όλα τα υποσυστήματα, συμπεριλαμβανομένου του πληκτρολογίου, του ελεγκτή της οθόνης και της διεπαφής της mesh δικτύωσης. Αν ο υπολογιστής είναι συνδεδεμένος με τον φορτιστή ο EC παραμένει ενεργός για να διαχειριστεί την φόρτιση της μπαταρίας. Αν λειτουργεί από την μπαταρία ο EC μπαίνει σε μια κατάσταση αδράνειας και περιμένει την πίεση του πλήκτρου εκκίνησης.

## Embedded Controller

Ο ενσωματωμένος ελεγκτής (EC=Embedded Controller) είναι ένας μικροεπεξεργαστής που προσφέρει έναν αριθμό από σημαντικά συστήματα παρακολούθησης και διαχείρισης των λειτουργιών του συστήματος. Αυτό περιλαμβάνει μεγάλο μέρος του συστήματος διαχείρισης τροφοδοσίας, το gamepad, τα κουμπιά περιστροφής/εκκίνησης, μια διεπαφή PS/2 για το touchpad και το πληκτρολόγιο και την διαχείριση της μπαταρίας. Τροφοδοτείται όποτε υπάρχει μπαταρία με επαρκή φόρτιση ή όταν ο υπολογιστής είναι συνδεδεμένος με την εξωτερική τροφοδοσία, παρ' όλο που μπαίνει σε αδρανή κατάσταση όταν είναι δυνατό. Ο EC επικοινωνεί με τον κύριο επεξεργαστή μέσω διεπαφής OLPC. Αυτό επίσης χρησιμοποιείται για να παρέχει firmware στον κύριο επεξεργαστή από την εξωτερική Flash μνήμη του EC.

## Διεπαφή προγραμματισμού

Οι συναρτήσεις που υποστηρίζονται από το EC περιγράφονται καλύτερα από την προγραμματιστική διεπαφή που προσφέρεται για χρήση από το CL1 firmware και το λειτουργικό σύστημα. Αυτή η διεπαφή καθορίζει το υλικό του υπολογιστή ότι το EC πρέπει να παρακολουθεί και να ελέγχει. Το firmware και το λειτουργικό σύστημα που τρέχουν στο CL1 του υπολογιστή μπορούν να διαβάσουν την ακόλουθη κατάσταση του υλικού μέσω του EC, χρησιμοποιώντας θύρες εισόδου/εξόδου (στον x86 χώρο εισόδου/εξόδου διευθύνσεων) 0x6C και 0x68:

- Τάση μπαταρίας.
- Ρεύμα Μπαταρίας.
- ACR Μπαταρίας.
- Θερμοκρασία Μπαταρίας.
- Περιβάλλουσα Θερμοκρασία.
- Κατάσταση Μπαταρίας (αν είναι συνδεδεμένη, αν είναι πλήρως φορτισμένη, αν τελειώνει, αν έχει καταστραφεί, αν είναι στην πρίζα).
- Κατάσταση φόρτισης Μπαταρίας.

- Battery Gas Gauge Serial Number.
- Laptop motherboard ID.
- System Control Interrupt (SCI) Source.
  - Κουμπί παιχνιδιού.
  - Αλλαγή κατάστασης μπαταρίας.
    - ★ DC Power input provided or removed.
    - ★ Μπαταρία συνδεδεμένη ή μη.
    - ★ Χαμηλή μπαταρία.
    - ★ Μπαταρία πλήρως φορτισμένη.
    - ★ Μπαταρία κατεστραμμένη.
  - Battery SOC change.
  - Battery subsystem error.
  - EBook mode change.
  - Lid status change.
- System Control Interrupt Mask.
- Display Panel Button Status (eight directional game keys and display rotate).
- Battery subsystem error.
  - Pack info fail (LiFePO4 & NiMH).
  - Over voltage checking fail (LiFePO4).
  - Over temperature (58C) (LiFePO4).
  - Gauge stop or sensor break (LiFePO4 & NiMH).
  - Sensor out of control (NiMH).
  - Battery ID fail & temperature > 52C.
  - Accumulated Charge Register fail (NiMH).
- Laptop power status (Display Controller, Wireless mesh networking interface).

Το Firmware και το λειτουργικό σύστημα που τρέχουν στο CL1 μπορούν να ελέγξουν τις παρακάτω λειτουργίες υλικού μέσω του ενσωματωμένου ελεγκτή χρησιμοποιώντας τις θύρες 0x6C και 0x68:

- Set System Control Interrupt (SCI) Mask.
- Initialize NiMH and LiFePO4 Battery.
- Enable/Disable Wireless mesh networking interface (WLAN) power.
- Wake up WLAN.

- Reset WLAN.
- Enable/Disable Display Controller power.

Επιπλέον κατάσταση συστήματος και έλεγχος είναι διαθέσιμα μέσω ενός standard ACPI πρωτοκόλλου (που χρησιμοποιεί I/O θύρες 0x66 και 0x62). Ο EC υποστηρίζει τις ακόλουθες εντολές:

- Query System Control Interrupt Event Queue.
- Power Off.
- Keyboard Stop.
- Keyboard Pause.
- Keyboard Resume.

Το πληκτρολόγιο και το touchpad control/status και τα δεδομένα είναι διαθέσιμα μέσω του standard PS/2 πρωτοκόλλου (που χρησιμοποιεί τις I/O θύρες 0x64 και 0x60).

### **Embedded Controller Firmware**

Το firmware για τον ενσωματωμένο ελεγκτή αποθηκεύεται σε μια ξεχωριστή σειριακή Flash μνήμη μεγέθους 1 Mbyte. Από αυτά, τα 64 KB μνήμης δεσμεύονται για το firmware του EC. Αυτή η σειριακή μνήμη χρησιμοποιείται επίσης για να αποθηκεύσει πληροφορίες κατασκευής (έως 4Kbytes) και το firmware του κύριου επεξεργαστή (έως 832 Kbytes). Η διεπαφή του LPC στον ενσωματωμένο ελεγκτή πολυπλέκει με διαφάνεια την πρόσβαση στην σειριακή μνήμη μεταξύ του EC και του κύριου επεξεργαστή.

### **Write Protect Function**

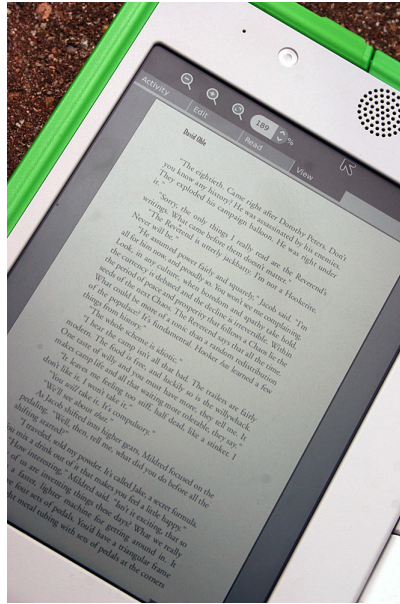
Με σκοπό να αποτρέψουν κακόβουλο λογισμικό να επαναγράψει κρίσιμο firmware και πληροφορίες κατασκευής που βρίσκονται αποθηκευμένες στην σειριακή Flash μνήμη του ενσωματωμένου ελεγκτή, το CL1 παρέχει έναν μηχανισμό υλικού ασφάλειας εγγραφής. Αυτός ο μηχανισμός απενεργοποιείται μόνο όταν γίνεται πλήρες reset στο σύστημα και ενεργοποιείται από τον EC. Όταν ενεργοποιηθεί, ο μηχανισμός ασφάλειας εγγραφής δεν μπορεί να απενεργοποιηθεί παρά μόνο από επόμενο reset συστήματος. Μετά από reset συστήματος, το firmware από την σειριακή Flash μνήμη θα φορτωθεί και θα εκτελεστεί από τον κύριο επεξεργαστή. Αν το firmware καθορίσει ότι υπάρχει διαθέσιμη αξιόπιστη ενημέρωση για το firmware, μπορεί να κάνει την ενημέρωση εκείνη τη στιγμή (και στη συνέχεια θα γίνει επανεκκίνηση και του EC και του κύριου επεξεργαστή). Αν δεν υπάρχει καμία διαθέσιμη ενημέρωση, ο κύριος επεξεργαστής λέει στον EC να ασφαλίσει την εγγραφή στη σειριακή Flash, πριν ξεκινήσει η εκτέλεση του λειτουργικού συστήματος. Αυτό αποτρέπει κακόβουλο λογισμικό ή τροποποιήσεις στο λειτουργικό σύστημα να αλλάξουν το βασικό firmware και τη διαδικασία φόρτωσης.

**Οθόνη**

Η TFT LCD οθόνη έχει έναν μοναδικό σχεδιασμό, που έχει αναπτυχθεί συγκεκριμένα για το CL1, και παράγεται από την CHI LIN Technology, Ltd. Είναι μια οθόνη με δύο λειτουργίες μεγέθους 19cm, που παρέχει μια έγχρωμη λειτουργία όταν χρησιμοποιείται σε διαυγή κατάσταση και μια μονόχρωμη λειτουργία πολύ χαμηλής κατανάλωσης ενέργειας όταν χρησιμοποιείται σε κατοπτρικό περιβάλλον. Μπορεί να χρησιμοποιηθεί σε έντονο ηλιακό φως. Στην κατοπτρική (μονόχρωμη) λειτουργία, υποστηρίζει εμφάνιση 1200 x 900 pixels (200 dots per inch), καθένα ικανό για 64 επίπεδα του γκρι. Στην διαυγή (έγχρωμη) λειτουργία, υποστηρίζει 693 x 520 τριάδες χρωμάτων, παρ' όλο που αντιληπτικά η ανάλυση είναι μεγαλύτερη και εξαρτάται από το επίπεδο φωτός στο περιβάλλον, σε ένα φωτεινό δωμάτιο οι δοκιμές δείχνουν ότι ανάλυση 1024 x 768 επιτυγχάνεται. Κάθε στοιχείο από τις τριάδες έχει 64 επίπεδα φωτεινότητας. Το LCD backlight για την διαυγή λειτουργία παρέχεται από άσπρα LEDs. Πολλαπλά LEDs συνδέονται σε μια σειρά από τρεις αλυσίδες για να εξισορροπείται η φωτεινότητα. Αυτές οι αλυσίδες τροφοδοτούνται παράλληλα και περιπλέκονται για να ελαχιστοποιήσουν τον αντίκτυπο σε περίπτωση που καεί μια αλυσίδα. Το ακριβές φάσμα που χρησιμοποιείται στα LEDs του backlight δεν διευκρινίζεται αλλά τα LEDs που χρησιμοποιούνται σε κάθε ένα backlight συνδυάζονται ώστε να συντελέσουν ένα ομοιόμορφο φάσμα στο backlight. Οι διαστάσεις της LCD οθόνης είναι 17.0cm x 14.0cm x 0.62cm (πάχος). Η ενεργή περιοχή εμφάνισης είναι 15.2cm by 11.4cm, και η bezel περιοχή είναι 15.5cm με 11.7cm. Ζυγίζει μόλις 172 gr. Ο σχεδιασμός του CL1 ελαχιστοποιεί το μήκος της καλωδίωσης μεταξύ του ελεγκτή της οθόνης και την οθόνης κι ένα dual-edge TTL πρωτόκολλο σηματοδότησης χρησιμοποιείται για τη διεπαφή. Ένα ευέλικτο καλώδιο με 55 pins παρέχει σύνδεση δεδομένων με ένα ξεχωριστό 4-pin καλώδιο πρίζας που συνδέει το ενιαίο LED backlight με τη μητρική του υπολογιστή.



Σχήμα 4.4: Η οθόνη στην κανονική της λειτουργία



Σχήμα 4.5: Η οθόνη στην ασπρόμαυρη λειτουργία

## Display Controller

Οι ελεγκτές οθόνης στους συμβατικούς ενιαίους επεξεργαστές και φορητούς υπολογιστές δεν πληρούν τις προϋποθέσεις χαμηλής κατανάλωσης του CL1. Όσοπου να κατασκευαστεί ένας ελεγκτής που να πληρεί αυτές τις προϋποθέσεις, χρησιμοποιείται ένας ξεχωριστός ελεγκτής οθόνης (DCON) που είναι ένα ενιαίο κύκλωμα για συγκεκριμένες εφαρμογές που έχει κατασκευαστεί μόνο για το CL1. Το (DCON) παρέχει τα παρακάτω χαρακτηριστικά:

- Αυτόνομη ανανέωση οθόνης, ανεξάρτητη από τη CPU ή την κατάσταση του GPU.
- Η εναλλαγή μεταξύ έγχρωμης και μονόχρωμης οθόνης δίνει τη δυνατότητα για συμβατικό πάνελ χρωμάτων των 24-bit.
- Η μονόχρωμη λειτουργία παρέχει pixel-addressable αυτόματο χρώμα (gray scale conversion mode).
- Προαιρετικό anti-aliasing βελτιώνει την οθόνη όταν έχει κείμενο και βρίσκεται στην έγχρωμη κατάσταση.
- Παρέχει dual-edge TTL output για διεπαφή με τη CL1 LCD οθόνη.

Το DCON έχει σχεδιαστεί για να δουλεύει με συμβατικό ελεγκτή οθόνης. Παρέχει μια διεπαφή εισόδου που εξομοιώνει μια TTL-συμβατή LCD οθόνη, που επιτρέπει την άμεση σύνδεση με έναν συμβατό ελεγκτή οθόνης. Στην ομαλή λειτουργία, όταν ο συμβατικός ελεγκτής οθόνης τροφοδοτείται και λειτουργεί το DCON απλά περνάει κατ' ευθείαν το βίντεο, κάνοντας τις απαραίτητες μετατροπές αλλά δεν κάνει κανένα buffering στο βίντεο. Πριν θέσει το λειτουργικό σύστημα τον υπολογιστή σε κατάσταση παύσης, ενημερώνει το DCON και τους DCON buffers για το καρέ του βίντεο που προβάλλεται στην μνήμη του.

Το DCON θα συνεχίσει να προβάλλει αυτό το καρέ μέχρι να ειδοποιηθεί από το λειτουργικό σύστημα ότι έχει φύγει από την κατάσταση παύσης και ξεκινά ένα έγκυρο βίντεο ζανά. Η διαμόρφωση του DCON γίνεται από μια σύνδεση SMBus που παρέχεται μέσω της South Bridge.

## **Ήχος**

Το CL1 παρέχει και εσωτερική και εξωτερική υποδοχή ήχου. Εσωτερικά, υπάρχουν δύο στέρεο ηχεία δεξιά και αριστερά από την οθόνη καθώς και μικρόφωνο μονού καναλιού που βρίσκεται στην μια μεριά της οθόνης. Υπάρχουν υποδοχές για ακουστικά, εξωτερικά ηχεία ή ένα εξωτερικό μικρόφωνο. Το υποσύστημα του ήχου έχει κατασκευαστεί με τα Audio Codec '97 standard. Ένα AC' 97 συμβατό codec παρέχει μετατροπή μεταξύ αναλογικών και ψηφιακών σημάτων καθώς και ικανότητες μίξης. Το codec υποστηρίζει δειγματοληψία εισόδου και εξόδου σε συχνότητες έως 96 KHz, χρησιμοποιώντας αναλογικούς/ψηφιακούς μετατροπείς με δυναμικό εύρος 80 dB και ψηφιακούς/αναλογικούς μετατροπείς δυναμικού εύρους 90 dB. Ένα AC '97 codec περιλαμβάνει αριθμό από κανάλια εισόδου και εξόδου που δεν υποστηρίζονται από το CL1. Αυτά απενεργοποιούνται από το firmware για να γίνεται εξοικονόμηση ενέργειας. Ολόκληρο το codec θα σταματήσει να τροφοδοτείται όταν κανένας ήχος δεν εισέρχεται ή εξέρχεται του συστήματος.

### **Εσωτερικά ηχεία**

Υπάρχουν δύο εσωτερικά ηχεία στον CL1 υπολογιστή, κάθε ένα από τα οποία οδηγείται από διαφορετικό κανάλι. Αυτά οδηγούνται από έναν αποκλειστικό ενισχυτή ήχου, με δυνατότητα να παρέχει 1.4 W συνεχόμενα σε κάθε ηχείο, με ολικό αρμονικό θόρυβο της τάξης του 1%. Η συχνότητα απόκρισης των εσωτερικών ηχείων έχει περίπου εύρος από 480 Hz έως 40 KHz.

### **Εσωτερικό μικρόφωνο**

Ένα εσωτερικό μικρόφωνο περιλαμβάνεται στο CL1, και βρίσκεται αριστερά της οθόνης. Αυτό παρέχει ήχο εισόδου μονού καναλιού. Το μικρόφωνο είναι ένας ηλεκτρικός πυκνωτής με ελάχιστο snr των 56 dB, και ελάχιστη ευαισθησία των -46 dB (0 dB στο 1Pa, συχνότητα του 1 KHz). Ένα ενδεικτικό φως υπάρχει κοντά στο μικρόφωνο αριστερά της οθόνης που δείχνει αν το μικρόφωνο χρησιμοποιείται (τροφοδοτείται). Αυτό το φως ελέγχεται από το υλικό όποτε το εσωτερικό μικρόφωνο μπορεί να χρησιμοποιηθεί και δεν μπορεί να απενεργοποιηθεί από το λογισμικό του υπολογιστή.

### **Εξωτερικά ακουστικά ή ηχεία**

Ο υπολογιστής διαθέτει μια στερεοφωνική υποδοχή για σύνδεση ακουστικών ή εξωτερικού ενισχυτή και ηχείων. Όταν τοποθετείται ένα βύσμα σε αυτήν την υποδοχή, τα εσωτερικά ηχεία αποσυνδέονται αυτόματα και ο ενισχυτής τους δεν τροφοδοτείται. Η κατάσταση αυτής της υποδοχής καθορίζεται από το λογισμικό του υπολογιστή. Τα ακουστικά οδηγούνται από τον εσωτερικό ενισχυτή του AC' 97 codec, με δυνατότητα συνεχόμενης παροχής 30mw σε αντίσταση 32 ohm.



### Εξωτερική υποδοχή μικροφώνου

Μια μονωμένη υποδοχή παρέχεται από τον φορητό υπολογιστή για σύνδεση εξωτερικού μικροφώνου, διακόπτη ή άλλου αισθητήρα. Αυτή η υποδοχή συνδέεται με την είσοδο του AC' 97 codec, παρέχοντας προγραμματιστικό όφελος προενισχυτή, με κέρδος 10, 20, ή 30 dB. Η αντίσταση εισόδου είναι της τάξης των 10 Kohm. Ένας DC blocking capacitor μπορεί να εισαχθεί στο μονοπάτι του σήματος υπό τον έλεγχο του λογισμικού. Αυτό χρησιμοποιείται όταν η είσοδος είναι ένα σήμα ήχου. Όταν χρησιμοποιείται, η συχνότητα απόκρισης της εισόδου είναι επίπεδη στα 10 Hz. Μια εγγενής τάση (2 V, με 3 Kohm αντίσταση πηγής) μπορεί επίσης να οδηγηθεί σε αυτή την υποδοχή εξωτερικού μικροφώνου, υπό τον έλεγχο του λογισμικού. Αυτό μπορεί να χρησιμοποιηθεί για να τροφοδοτήσει έναν εξωτερικό ηλεκτρικό πυκνωτή μικροφώνου, ή χρησιμοποιείται για να εντοπίζει αν ο εξωτερικός διακόπτης είναι ανοιχτός ή κλειστός. Λόγω της υψηλής εμπέδισης την εγγενούς πηγής, αυτή η είσοδος μπορεί με ασφάλεια να τοποθετηθεί στη γείωση. Η εξωτερική τάση που εφαρμόζεται σε αυτή την είσοδο είναι μεταξύ των 0 και 3.5 V. Υπέρβαση αυτής της τάσης μπορεί να προκαλέσει περίσσεια φορτίου να ρεύσει μέσα στην δίοδο προστασίας. Όταν χρησιμοποιείται η εξωτερική υποδοχή μικροφώνου, το εσωτερικό μικρόφωνο αποσυνδέεται, και το ενδεικτικό φως υποδεικνύει ότι το μικρόφωνο είναι απενεργοποιημένο. Η κατάσταση της υποδοχής μπορεί να εντοπιστεί από το λογισμικό.

### Σύστημα Αποθήκευσης

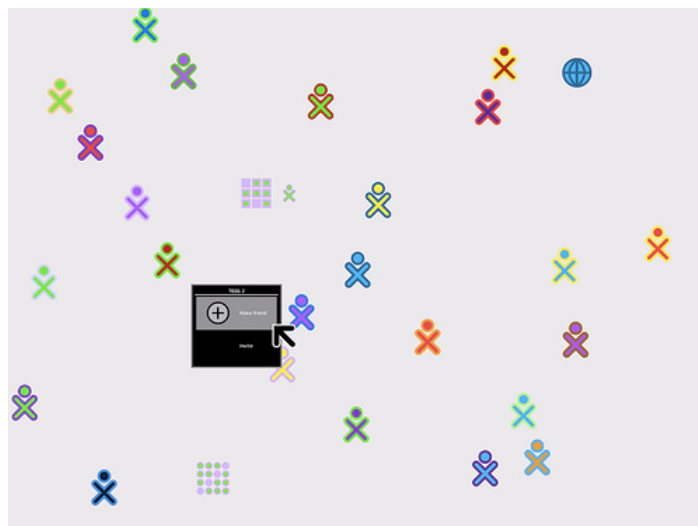
Το CL1 δεν έχει περιστρεφόμενη συσκευή αποθήκευσης μέσω. Βασίζεται σε μια NAND Flash μνήμη για μη-μεταβαλλόμενη αποθήκευση. Επιπλέον έχει 1 GByte SLC NAND Flash που χρησιμοποιεί μόνο ένα ενιαίο κύκλωμα. Η διεπαφή μεταξύ του επεξεργαστή και της NAND είναι ζωτικής σημασίας για την απόδοση του συστήματος. Έχει μεγάλη ταχύτητα και παρέχει ECC λειτουργίες στο υλικό. Η πρόσβαση στα δεδομένα γίνεται με ταχύτητα πάνω από 30 MBytes/sec. Η ανθεκτικότητα της συσκευής NAND είναι 100,000 erase κύκλοι με διατήρηση δεδομένων για τουλάχιστον δέκα χρόνια.

Στο υλικό δεν προσφέρεται wear-levelling αφού προσφέρεται από το λειτουργικό σύστημα. Το single layer κέλυφος τύπου (SLC) NAND Flash χρησιμοποιείται από τη CL1 για να επεκτείνει τη διάρκεια ζωής του υπολογιστή. Παρέχει έναν σημαντικά μεγαλύτερο αριθμό erase cycles απ' ό τι το denser multiple layer cell (MLC) NAND Flash devices. Η CL1 επίσης έχει single MMC/SD θέση κάρτας για να επεκτείνεται ο αποθηκευτικός χώρος του laptop.

### Διεπαφή mesh ασύρματης δικτύωσης

Η διεπαφή ασύρματου δικτύου του CL1 λειτουργεί όπως η συμβατική WiFi (802.11b/g) διεπαφή, αλλά παρέχει και πληθώρα άλλων μοναδικών χαρακτηριστικών κρίσιμης σημασίας για τους στόχους του XO. Σε αντίθεση με τα παραδοσιακά ασύρματα δίκτυα όπου οι υπολογιστές επικοινωνούν μόνο με τα σημεία πρόσβασης, το CL1 υποστηρίζει ασύρματη mesh δικτύωση όπου οι υπολογιστές μπορεί να αναμεταδίδουν πακέτα για άλλους υπολογιστές, επεκτείνοντας την πρόσβαση σε οποιοδήποτε σημείο πρόσβασης του δικτύου. Με σκοπό την ενθάρρυνση της mesh δικτύωσης, η διεπαφή του CL1 δικτύου έχει σχεδιαστεί για να έχει λειτουργία χαμηλής κατανάλωσης ως mesh κόμβος, ακόμα κι όταν ο υπολογιστής είναι κλειστός. Το πρωτόκολλο mesh δικτύωσης είναι προς το παρόν μια παραλλαγή

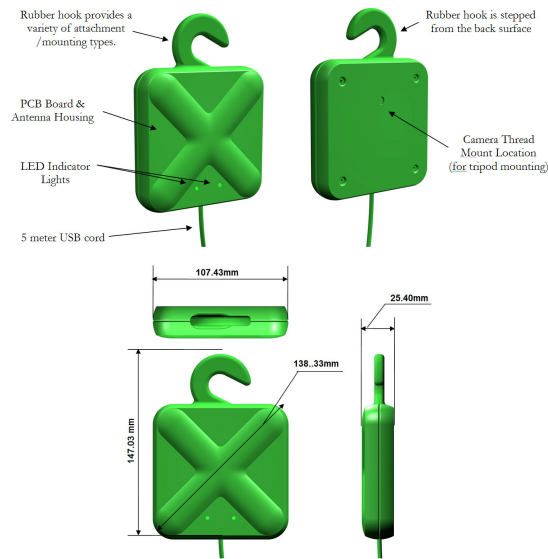
του επερχόμενου προτύπου 802.11s mesh δικτύωσης. Το CL1 θα υποστηρίξει 802.11s, όταν επικυρωθεί, μέσω μιας αναβάθμισης του firmware. Το παρών πρωτόκολλο και το 802.11s μπορούν να υποστηρίζονται ταυτόχρονα και τα δύο. Σε αντίθεση με τις διεπαφές των παραδοσιακών φορητών υπολογιστών, η CL1 διεπαφή ασύρματης δικτύωσης περιλαμβάνει έναν επεξεργαστή με την ικανότητα να προωθεί πακέτα και να διατηρεί τους πίνακες δρομολόγησης, ανεξάρτητα από κάθε άλλο πόρο των υπολογιστών. Η διεπαφή της δικτύωσης όταν ο επεξεργαστής εισέρχεται ή εξέρχεται από την λειτουργία παύσης και έχει τη δυνατότητα να ειδοποιεί τον ενσωματωμένο ελεγκτή ότι μια δικτυακή ενέργεια χρειάζεται προσοχή (όπως για παράδειγμα ένα πακέτο που προορίζεται για τον υπολογιστή) έχει συμβεί. Ενώ είναι πιθανό, το firmware προσωρινά περιορίζει τη λειτουργία του δικτύου όταν ο υπολογιστής είναι κλειστός ή σε κατάσταση παύσης. Για λειτουργία σε 802.11b, η RF έξοδος είναι 17 dBm (+/- 1.5 dBm), με ευαισθησία λήπτη -87 dBm. Στην 802.11g λειτουργία η RF τροφοδοσία εξόδου είναι 13 dBm (+/- 1.5 dBm), με ευαισθησία λήπτη -72 dBm.



Σχήμα 4.6: Η mesh δικτύωση

### **Σχεδιασμός κεραίας**

Οι κεραίες για την ασύρματη δικτύωση έχουν σχεδιαστεί για να παρέχουν όσο τον δυνατόν καλύτερη αποδοτικότητα. Υπάρχουν δύο περιστρεφόμενες κεραίες, που βρίσκονται στις πάνω γωνίες της οθόνης του υπολογιστή τοποθετημένες η μια αριστερά και η άλλη δεξιά και συνήθως τις αποκαλούν αυτιά λαγού. Η δυνατότητα για στροφή των κεραίων προς τα πάνω, πάνω από το σώμα του υπολογιστή παρέχει βέλτιστη απόδοση. Όταν τα αυτιά λαγού είναι κάτω, καλύπτουν τις θύρες εισόδου/εξόδου USB και ήχου και έχουν μηχανισμό μαντάλωσης.



Σχήμα 4.7: Οι κεραίες του OLPC

## Camera Module

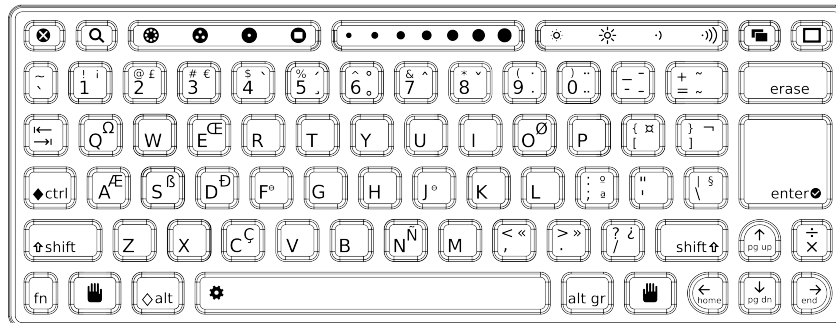
Το CL1 παρέχει μια κάμερα με σταθερούς φακούς που βρίσκεται στην δεξιά πλευρά της οθόνης. Η κάμερα έχει ανάλυση 640 x 480, και μπορεί να λειτουργήσει έως τριάντα καρέ το δευτερόλεπτο. Παρέχει ρυθμό σήματος/θορύβου τουλάχιστον 46 dB και έχει αισθητήρα διάταξης μεγέθους 2.36mm x 1.76mm. Ο αριθμός F του οπτικού συστήματος είναι 2.8 (+/- 5%), το μήκος αποδοτικής εστίασης είναι 2.45mm και το οπτικό πεδίο είναι 60 βαθμών. Οι παράμετροι της κάμερα (όπως αυτόματος χειρισμός, αυτόματος χειρισμός έκθεσης, αυτόματη εξισορρόπηση λευκού, διόρθωση φακών, χειροποίητη εξισορρόπηση χρωμάτων ή ρύθμιση του επιπέδου του μαύρου) μπορούν να προσαρμοστούν υπό το έλεγχο του λογισμικού. Ένα ενδεικτικό φως στη δεξιά πλευρά της οθόνης υποδεικνύει αν η κάμερα χρησιμοποιείται (τροφοδοτείται). Αυτό το φως ελέγχεται άμεσα από το υλικό όποτε είναι πιθανό να είναι η κάμερα σε χρήση. Δεν μπορεί να απενεργοποιηθεί από το λογισμικό.

## User Input Devices

Επιπρόσθετα με το μικρόφωνο και την κάμερα που περιγράφονται, οι συσκευές εισόδου που παρέχονται από το CL1 είναι πληκτρολόγιο, touchpad, ένα κατευθυντήριο gamepad, κουμπί περιστροφής οθόνης, τέσσερα κουμπιά για παιχνίδια και κουμπί τροφοδοσίας στη δεξιά πλευρά του υπολογιστή.

## Πληκτρολόγιο

Το CL1 έχει πληκτρολόγιο μεμβράνης με 72 κουμπιά κατασκευασμένα από πλαστικό με επιφάνεια πίεσης μεγέθους 1mm. Είναι αδιάβροχο και εμποδίζει και την είσοδο σκόνης. Το πληκτρολόγιο και το touchpad μοιράζονται μια σύνδεση καλωδίου με τη λειτουργία πυρήνα, που χρησιμοποιεί το PS/2 πρωτόκολλο σηματοδοσίας.



Σχήμα 4.8: Το πληκτρολόγιο του OLPC

### **Touchpad**

Το CL1 είναι εξοπλισμένο με touchpad που χρησιμοποιείται ως συσκευή ένδειξης και χρειάζεται γυμνό δάχτυλο για να λειτουργήσει. Το touchpad έχει ανάλυση και ενεργή περιοχή 6.0 cm x 4.4 cm. Βρίσκεται κάτω από το πλαστικό, μπροστά από το πληκτρολόγιο και με αυτό τον τρόπο μεγιστοποιείται η ανθεκτικότητα στην υγρασία, τη βρωμιά και την ηλεκτροστατική εκφόρτωση.

### **Core Module Buttons**

Το CL1 περιλαμβάνει δύο σειρές από κατευθυντήρια κουμπιά εισόδου, μια σε κάθε πλευρά της οθόνης. Στην αριστερή πλευρά, τα κουμπιά συνδυάζονται σε ένα μεγάλο κατευθυντήριο κουμπί. Στη δεξιά πλευρά της οθόνης, τα κουμπιά έχουν χαρακτηριστικά εικονίδια (τετράγωνο, κύκλος, τικ και χ) κι επίσης έχουν κατευθυντήρια ομαδοποίηση. Οι δύο σειρές αυτές προσανατολίζονται στο να παρέχουν κίνηση πάνω, κάτω και δεξιά σε σχέση με την οθόνη. Επιπλέον, υπάρχει ένα single user κουμπί εισόδου στην αριστερή πλευρά της οθόνης (που αντικατοπτρίζει το κουμπί τροφοδοσίας). Αυτό το κουμπί είναι αφιερωμένο στον προσανατολισμό περιστροφής της οθόνης (ώστε να υποστηρίζεται καλύτερα η λειτουργία ηλεκτρονικού βιβλίου). Ο ενσωματωμένος ελεγκτής παρέχει μια διεπαφή γι' αυτά τα κουμπιά, εισάγοντας τα στη ροή χαρακτήρων που λαμβάνει από το πληκτρολόγιο.

### **Lid and eBook Switches**

Υπάρχουν δύο μαγνητικοί αισθητήρες στη λειτουργία πυρήνα που καθορίζουν την γειννίαση και τον προσανατολισμό του σε σχέση με το Input Module. Ο ένας από αυτούς χρησιμοποιείται για να εντοπίζει αν ο υπολογιστής είναι κλειστός, με την οθόνη να μην έχει εικόνα και το πληκτρολόγιο να μην είναι προσβάσιμο. Το λειτουργικό σύστημα θα το χρησιμοποιήσει είτε για να παύσει είτε για να κλείσει τον υπολογιστή. Ο άλλος, χρησιμοποιείται για να εντοπίζει αν ο υπολογιστής είναι στην λειτουργία ηλεκτρονικού βιβλίου, όπου η οθόνη περιστρέφεται πριν κλείσει ο υπολογιστής. Σε αυτή τη λειτουργία το πληκτρολόγιο δεν χρησιμοποιείται αλλά η οθόνη χρησιμοποιείται. Αυτοί οι αισθητήρες

συνδέονται με τον ενσωματωμένο ελεγκτή, και οι αλλαγές στην κατάσταση τους επικοινωνούνται στο λειτουργικό σύστημα μέσω διακοπών ελέγχου συστήματος.

### **Θύρες εισόδου/εξόδου**

Εκτός από τις υποδοχές για τα ακουστικά (έξοδος ήχου) και το μικρόφωνο (είσοδος ήχου/αισθητήρα), το CL1 έχει θύρες USB για να δημιουργεί διεπαφές με εξωτερικές συσκευές.

### **USB**

Το CL1 υποστηρίζει τρεις USB 2.0 θύρες, δύο στη μια άκρη της οθόνης και μια στην άλλη. Όλες αυτές οι θύρες είναι τύπου A συνδαιτυμόνες για τη σύνδεση περιφερειακών συσκευών αποθήκευσης. Σε αντίθεση με έναν κανονικό φορητό υπολογιστή, που παρέχει έως και 2.5 Watts σε κάθε USB θύρα, το CL1 παρέχει ολική ενέργεια κατανάλωσης 5 W και για τις τρεις θύρες USB. Αυτό το ποσό ενέργειας μπορεί να καταναλωθεί ομοιόμορφα ανάμεσα στις θύρες ή να απορροφηθεί μόνο από τη μια, αλλά δεν δίνονται περισσότερα από 5 W.

### **Debugging Support**

Η μητρική του CL1 στη λειτουργία πυρήνα περιλαμβάνει έναν αριθμό συνδαιτυμόνων για να βοηθήσουν το firmware και το debugging του λειτουργικού συστήματος. Αυτοί, δεν βρίσκονται στις μηχανές παραγωγής, αλλά μπορούν να προστεθούν εύκολα στην ανάπτυξη του λογισμικού. Δεν είναι προσβάσιμοι εκτός κι αν ο υπολογιστής είναι μερικώς αποσυναρμολογημένος. Αυτοί οι συνδαιτυμόνες είναι:

- Ένα RS-232 πρωτόκολλο σειριακής σύνδεσης (3.3V) στον κύριο επεξεργαστή.
- Ένας main processor JTAG connector.
- Ένα RS-232 πρωτόκολλο σειριακής σύνδεσης στον ενσωματωμένο ελεγκτή.
- Ένας JTAG connector για το WLAN module.

### **Δοκιμές**

Η CL1 δοκιμάστηκε επιτυχώς σε μεγάλο εύρος περιβαλλοντικών, μηχανικών και ηλεκτρολογικών δοκιμών.

### **Ασφάλεια**

Η CL1 έχει UL και CE πιστοποίηση ασφάλειας. Ο υπολογιστής ανταποκρίνεται στις προδιαγραφές του IEC 60950-1, EN 60950-1, και CSA/UL 60950-1 (Safety of Information Technology Equipment). Επιπλέον συμμορφώνεται με UL 1310 και UL 498. Με σκοπό την εγγύηση της ασφάλειας των παιδιών που χρησιμοποιούν τον υπολογιστή, συμμορφώνεται με ASTM F 963 (Standard Consumer Safety Specification on Toy Safety, 2003 edition). Ο εξωτερικός φορτιστής υπακούει σε IEC, EN, και CSA/UL 60950-1. Η αποσπώμενη μπαταρία υπακούει σε IEC, EN, CSA/UL 60950-1 και UL 2054 (Household and Commercial Batteries).

### Περιβαλλοντικές συνθήκες

Η CL1 έχει σχεδιαστεί να λειτουργεί σε εκτεταμένο εύρος περιβαλλοντικών συνθηκών:

- Εύρος θερμοκρασίας: 0 έως 50°C
- Υψόμετρο Λειτουργίας: έως 5000 m
- Εύρος θερμοκρασίας μη-λειτουργίας: -20 έως 60°C

### Διάρκεια

Η CL1 αρχιτεκτονική σχεδιάστηκε για να έχει μεγάλη διάρκεια.

### Συσκευές εισόδου

Τα κουμπιά λειτουργίας πυρήνα (τροφοδοσία, περιστροφή οθόνης) ελέγχονται στους 500,000 κύκλους, το πληκτρολόγιο ελέγχεται στους 5,000,000 κύκλους, όλοι οι διασυνδεδεμένοι εισόδων εξόδων (τροφοδοσία, USB, ακουστικά και μικρόφωνο) ελέγχονται στους 5,000 κύκλους.

### Δοκιμή Πτώσης

Η CL1 δοκιμάστηκε επιτυχώς σε ελεύθερη πτώση από ύψος 150 cm σε χαλί που κάλυπτε ατσάλινο δίσκο και δοκιμάστηκε επιτυχώς σε ελεύθερη πτώση από ύψος 80 cm σε ατσάλινο δίσκο.

### Περιβαλλοντικά

Το XO είναι το πιο αποδοτικό ενεργειακά και φιλικό με το περιβάλλον laptop που κατασκευάστηκε ποτέ, σύμφωνα με ανεξάρτητες αξιολογήσεις και δεδομένα. Το XO καταναλώνει τη λιγότερη δυνατή ενέργεια, ελαχιστοποιεί τα τοξικά υλικά, είναι αξιοσημείωτα σκληροτράχηλο, έχει μεγάλη διάρκεια ζωής λειτουργεί με ανανεώσιμες πηγές ενέργειας, και μπορεί να ανακυκλωθεί και το ίδιο. Το XO έχει κερδίσει τα πιο υψηλόβαθμα περιβαλλοντικά πιστοποιητικά:

Είναι πλήρως συμμορφωμένο με τα πρότυπα του άτεγκτου RoHS της Ευρωπαϊκής Ένωσης, πληρεί τις προϋποθέσεις για Energy Star 4.0 κατηγορία (την πιο αυστηρή κατηγορία) και τέλος έχει λάβει από την εταιρία αξιολόγησης των προσωπικών και των φορητών υπολογιστών των Ηνωμένων Πολιτειών EPEAT χρυσή αξιολόγηση, ένα από τα οχτώ laptops που το έχουν καταφέρει ποτέ αυτό. Σύμφωνα με την εταιρία ENERGY STAR®, ένας μέσος επιτραπέζιος υπολογιστής στην αδρανή λειτουργία καταναλώνει 70 watt, ένας μέσος φορητός υπολογιστής στην αδρανή λειτουργία καταναλώνει 20 watts ενώ το XO στην αδρανή λειτουργία καταναλώνει μόλις 1 watt. Μεταξύ άλλων περιβαλλοντικά φιλικών καινοτομιών και χαρακτηριστικών το XO: είναι το πιο σκληροτράχηλο, θα αντέξει περισσότερο και ως αποτέλεσμα θα παραμείνει έξω από τις χωματερές. Έχει σχεδιαστεί για διάρκεια ζωής πέντε χρόνων ακόμα και σε ακραία περιβάλλοντα όπως το ύπαιθρο, η ζούγκλα και η έρημος. Ο μέσος φορητός υπολογιστής έχει διάρκεια ζωής 2 χρόνια όταν χρησιμοποιείται σε γραφείο και πολύ λιγότερο όταν χρησιμοποιείται στο ύπαιθρο ή σε έρημο. Ο

διπλασιασμός της διάρκειας ζωής του υπολογιστή μειώνει στο μισό το περιβαλλοντικό του αντίκτυπο.

Το XO είναι μισό σε μέγεθος και βάρος σε σχέση με το μέσο laptop. Το λιγότερο υλικό διαιρεί στη μέση τον αντίκτυπο στο περιβάλλον. Το XO είναι ο πρώτος φορητός υπολογιστής που κατασκευάστηκε με υλικά που έχουν ανανεώσιμη ενέργεια: μια μανιέλα χειρός, ένα μικρό ηλιακό έλασμα, ένα πετάλ ποδός, ή ένα σχοινί που θυμίζει τον τρόπο λειτουργίας των μηχανημάτων κοπής γκαζόν μπορεί να επαναφορτίσει τον υπολογιστή.

Το XO χρησιμοποιεί μια μπαταρία και καινούρια χημεία που ονομάζεται LiFeP που διαρκεί τέσσερις φορές περισσότερα απ' ό,τι οι συνηθισμένες μπαταρίες φορητών υπολογιστών, και είναι σαφώς πιο ασφαλής από την κυρίαρχη τεχνολογία Lithium Ion που χρησιμοποιείται αυτή τη στιγμή.

### 4.1.2 Καινοτομίες στο OLPC

Οι καινοτομίες στο OLPC είναι πολλές και αρκετές από αυτές περιγράφηκαν στην ανάλυση του Hardware. Ωστόσο, αξίζει να αναφέρουμε κάποιες λεπτομέρειες του XO.

Το DCON ολοκληρωμένο κύκλωμα μας επιτρέπει να καταναλώνουμε το ανανεωμένο πολύ χαμηλής ενέργειας επίπεδο πλαίσιο και ως αποτέλεσμα να κλείνουμε τελειώς τον επεξεργαστή. Δεδομένου ότι το επίπεδο πλαίσιο μπορεί να χρησιμοποιηθεί σε κλίμακες του γκρι με λειτουργία στα 0.1 watt, μπορούμε εύκολα να διαπιστώσουμε ότι η κατανάλωση ενέργειας μπορεί να ελεγχθεί εύκολα από το πλαίσιο τροφοδοσίας.

Μπορούμε επίσης να αφήσουμε την ασύρματη Marvell λειτουργία να λειτουργεί ανεξάρτητα, προωθώντας πακέτα στη mesh δικτύωση ενώ όλα τα υπόλοιπα στοιχεία του υπολογιστή είναι κλειστά.

## 4.2 Software στο OLPC

Το εξ ορισμού λειτουργικό σύστημα του XO φορητού υπολογιστή περιλαμβάνει εφαρμογές φιλικές προς τα παιδιά, τη διεπαφή Sugar και πολλά άλλα λογισμικά μέρη που έχουν αναπτυχθεί πάνω σε έναν τροποποιημένο πυρήνα του λειτουργικού συστήματος Fedora Linux. Σχεδόν όλο το λογισμικό είναι ανοιχτό και οι χρήστες μπορούν να το ερευνήσουν και να το τροποποιήσουν. Εναλλακτικά, εκδόσεις των Debian, Fedora, Ubuntu και άλλες Linux διανομές έχουν αναπτυχθεί και τρέχουν ομαλά στο XO.

### Live CDs και emulation

Αν κάποιος δε διαθέτει XO φορητό υπολογιστή, μπορεί να εξομοιώσει την τρέχουσα έκδοση περιβάλλοντος του XO (Fedora 9, Sugar, εφαρμογές πυρήνα) στον υπολογιστή του τρέχοντας είτε Live CD είτε μια OS images σε εξομοιωτή.

- Λεπτομέρειες λειτουργικού συστήματος
  - Ο πυρήνας του Linux βασίζεται στην πρόσφατη έκδοση Fedora 2.6.x.x Τα συγκεκριμένα bits του OLPC βασίζονται στο olpc-2.6 GIT δέντρο.
- Προγραμματιστικά περιβάλλοντα

- Το μεγαλύτερο μέρος της διεπαφής Sugar είναι γραμμένη σε Python (Έκδοση 2.5) όπως και πολλές εφαρμογές. Συγκεκριμένα υπάρχει μια εφαρμογή στον υπολογιστή, η Pippy, που βοηθά τον χρήστη να εξερευνήσει την Python.
- Άλλα προγραμματιστικά περιβάλλοντα είναι:
  - Forth σε Open Firmware, προσβάσιμο από το 'ok' prompt αν διαθέτουμε Developer Key.
  - JavaScript™.
  - Csound, μουσική προγραμματιστική γλώσσα.
  - Etoys, μια εφαρμογή του Squeak που χρησιμοποιεί Smalltalk, μια αντικειμενοστραφή γλώσσα προγραμματισμού.
  - Turtle Art, προγραμματιστικό περιβάλλον γραφικών.
  - Gnash, ένας δωρεάν και ανοιχτού λογισμικού browser plug-in για το Adobe Flash™. αρχείο που υποστηρίζει δείκτες και raster γραφικά, μια scripting γλώσσα προγραμματισμού που ονομάζεται ActionScript™ και streaming ήχο και βίντεο.
  - Adobe's Flash Player, Java™ εικονική μηχανή, και παίκτες από άλλα περιορισμένα formats μπορούν να προστεθούν μέσω των Yum ή Rpm αλλά δεν συμπεριλαμβάνονται στην αρχική διανομή.

### **Βιβλιοθήκες**

Ακολουθεί μια ενδεικτική λίστα των βιβλιοθηκών που χρησιμοποιεί το λογισμικό του XO. Αν κάποιος ενδιαφέρεται για την πλήρη λίστα των βιβλιοθηκών μπορεί να συμβουλευτεί το build log μιας εικόνας ή να τρέξει τις εντολές rpm στο XO.

- Mozilla Gecko/XULRunner.
- GUI toolkit (GTK+), που χρησιμοποιείται και από το Gnome.
- Κάποιες ακόμα Βιβλιοθήκες από το Gnome.
- Matchbox window manager.
- Pango text layout.
- Gnome accessibility toolkit (ATK).
- Python GTK+ bindings.
- Cairo 2D-graphics support.
- X Window System X.org Foundation.
- Font rendering (Freetype).
- Telepathy real-time communications framework.



- Avahi local service discovery.
- Multimedia framework: GStreamer.
- Bitfrost security system.
- Mesh network following IEEE 802.11s draft.
- Pygame.
- Gettext, the GNU internationalization library.
- Επιπλέον βιβλιοθήκες για να υποστηρίξουν τη λειτουργία των παραπάνω βιβλιοθηκών.

Οι χρήστες μπορούν να εγκαταστήσουν βιβλιοθήκες της επιλογής τους με τις εντολές yum ή rpm.

### Περιβάλλον Χρήστη

Το περιβάλλον χρήστη είναι Sugar γραφικό περιβάλλον, γραμμένο σε Python, στο πάνω μέρος του X Window System και του Matchbox Window Manager. Έχει σχεδιαστεί ειδικά για συνεργασία χρηστών μέσω των εφαρμογών που διαμοιράζονται στο δίκτυο (μέθοδος κλήσεων και σημάτων). Έχει "Zooming" διεπαφή σε σφαίρες συνδεσιμότητας δικτύου (τοπικές, συνεργατικές και γειτονικές) . Τέλος διαθέτει Jurnal διεπαφή για να αποθηκεύει γεγονότα, εφαρμογές, αντικείμενα (αρχεία) και 800 MiB flash αποθηκευτικό χώρο που είναι διαθέσιμος για άλλες λειτουργίες του λογισμικού.

Εναλλακτικά περιβάλλοντα χρήστη:

- Window-Maker-Interface: Μια διεπαφή που βρίσκεται γύρω από τα Linux για πολλά χρόνια. Τρέχει στο πάνω μέρος του X Windows συστήματος και είναι γραμμένο σε C++. Το WindowMaker είναι περισσότερο σύνηθες ως διεπαφή σε φορητό υπολογιστή. Αυτή η διεπαφή είναι ακόμα σε πρώιμο στάδιο στο OLPC και χρειάζεται πολύ τεκμηρίωση και δοκιμές πριν χρησιμοποιηθεί σε μια τάξη σχολείου. Το μόνο μειονέκτημα του WindowMaker στο OLPC είναι ότι υπάρχει μια δυσλειτουργία στο scrolling και τον ρυθμό ανανέωσης της οθόνης.
- OpenBox: Ένας μικρός συμπαγής διαχειριστής παραθύρου γραμμένος σε γλώσσα C++. Αν θέλουμε να αλλάξουμε το Sugar με ένα πιο συμβατικό περιβάλλον αυτό πρέπει να χρησιμοποιήσουμε. Είναι γρήγορο, αποδοτικό και προβάλλει το παράθυρο ωραία. Λειτουργεί πολύ καλά με abiwrod, firefox και gnome-terminal.
- Xfce: Μια ελαφριά αλλά δυνατή επιφάνεια εργασίας που λειτουργεί καλά στο XO. Έχει σκοπό να είναι γρήγορο και ελαφρύ, ενώ παράλληλα είναι αισθητικά ελκυστικό και εύκολο στη χρήση. Το Xfce χρησιμοποιείται λόγω της δυνατότητάς του να τρέχει σε μοντέρνα επιφάνεια εργασίας σε σχετικά μέτριο hardware και γι' αυτό είναι μια θαυμάσια επιλογή για το OLPC. Βασίζεται στο GTK+ 2 toolkit (αφού είναι GNOME). Χρησιμοποιεί το Xfwm window manager. Η διαμόρφωσή του γίνεται πλήρως μέσω του ποντικιού, και τα αρχεία διαμόρφωσης είναι κρυμμένα από τον περιστασιακό χρήστη.

### Εργαλεία εξερεύνησης

- Browse, ένας web browser που έχει αναπτυχθεί με βάση το Firefox engine.
- Read, ένα απλό εικονοσκόπιο κειμένου που μπορούμε να προσπελάσουμε μέσω του Browser και βασίζεται στο Evince (περιλαμβάνει και τη δυνατότητα για προβολή αρχείων PDF).
- News Reader, ένα RSS (“really simple syndication”) subscription reader (Penguin-TV).
- Αναπαραγωγή πολυμέσων χρησιμοποιώντας gstreamer (η πλατφόρμα Real Networks Helix έχει μεταφερθεί στο φορητό υπολογιστή και είναι διαθέσιμη για κατέβασμα, αλλά δεν αποτελεί μέρος της βασικής διανομής).
- OpenDocument Viewer, για να διαβάξει έγγραφα σε μορφή Opendocument, μία μορφή υψηλής-συμπίεσης που είναι ένα ανοιχτό διεθνές πρότυπο (ISO 26300).
- Ο Opera web browser και οι Real Networks Helix<sup>TM</sup> πλατφόρμες έχουν μεταφερθεί στο φορητό υπολογιστή και είναι διαθέσιμες για κατέβασμα, αλλά δεν είναι μέρος της βασικής διανομής.
- Measure, ένα εργαλείο για την εξερεύνηση του φυσικού κόσμου με τη μέτρηση συνεχούς και εναλλασσόμενης τάσης, παρατηρώντας τα σε παλμογραφική διασύνδεση, να είναι σε θέση να παρακολουθεί κυματομορφές στο πεδίο της συχνότητας (αναλυτής φάσματος), καταγραφή δεδομένων σε ένα συγκεκριμένο χρονικό διάστημα, και σχεδιασμό γραφικής παράστασης των δεδομένων.
- Distance, γνωστό και ως Acoustic Tape Measure, μετρά την απόσταση μεταξύ δύο XO φορητών υπολογιστών.

### Εργαλεία έκφρασης

- TamTam, ένα εργαλείο μουσικής σύνθεσης.
- Etoys.
- Ένας επεξεργαστής κειμένου που βασίζεται στο Abiword project.
- Record εργαλείο βιντεοσκόπησης, ηχογράφησης και φωτογράφισης.
- Draw, ένα πρόγραμμα ζωγραφικής.
- Journal.
- MikMik, ένα wiki με WYSIWYG editing, που χρησιμοποιεί Crossmark (υπό κατασκευή).
- VIM και nano επεξεργαστές κειμένου.

### Εργαλεία επικοινωνίας

- Chat και serverless instant messenger.
- Video Chat (υπό κατασκευή).
- VoIP client (υπό κατασκευή).
- Email μέσω του web-based Gmail service.
- Native email client (υπό κατασκευή).

### Άλλα εργαλεία

- Κομπιουτεράκι.
- Spreadsheet (υπό κατασκευή).
- Terminal, ένα κέλυφος για τις εντολές που χρησιμοποιεί Bash.
- Develop, ένας επεξεργαστής εφαρμογών (υπό κατασκευή).
- Εργαλεία Remote display και Remote Desktop, για την απομακρυσμένη πρόσβαση στο XO laptop.

### Online εφαρμογές

- Google Docs λειτουργούν άψογα από τον browser του XO. Επιπλέον είναι διαμοιραζόμενες εφαρμογές.
- Apple Web Apps, παρ' όλο που έχουν σχεδιαστεί για τα iPhone, δουλεύουν καλά και στο XO. Είναι προσβάσιμα μέσω του Browser. Σε πολλές περιπτώσεις πρέπει να κάνεις zoom στη σελίδα για να προσαρμοστεί στην οθόνη.
- ALEKS.
- Drupal εξηγεί πως να εγκαταστήσουμε ένα Drupal community content management σύστημα στο XO μας.
- THE ALAS PROJECT (Advanced Language Acquisition Software) είναι ένα δωρεάν και ανοιχτού λογισμικού online πρόγραμμα για την εκμάθηση της Αγγλικής γλώσσας σε ισπανόφωνους και την εκμάθηση ισπανικών σε αγγλόφωνους με αδελφά σχολεία στις Ηνωμένες Πολιτείες και τη Λατινική Αμερική. Το πρόγραμμα έχει αναπτυχθεί για να τρέχει στον XO φορητό υπολογιστή.

### Παιχνίδια

Μεγάλη ποικιλία από παιχνίδια που περιλαμβάνουν παραλλαγές του παιχνιδιού μνήμης, παιχνίδια στρατηγικής κ.α.

### Διαμοιραζόμενες εφαρμογές

Όλες οι εφαρμογές διαμοιράζονται κοινά δεδομένα αποθήκευσης μέσω του Journal. Πολλές εφαρμογές συμπεριλαμβανομένων της ανάγνωσης, της γραφής την εγγραφής και του browsing, επιτρέπουν την συνεργασία μαθητή με μαθητή και δασκάλου με μαθητή σε διάφορα επίπεδα μέσω του δικτύου. Σε αυτό το πλαίσιο κινείται και η εφαρμογή που αναπτύξαμε.

### Τρέχοντας άλλες εφαρμογές Linux

Μπορείτε να εγκαταστήσετε και να τρέξετε άλλες εφαρμογές Linux χρησιμοποιώντας yum ή rpm. Για παράδειγμα το Skype.

### Mesh δικτύωση

Τα χαρακτηριστικά της διεπαφής mesh δικτύωσης περιλαμβάνουν:

- Την ικανότητα ένας υπολογιστής να δρα ως κόμβος ακόμη κι όταν η CPU δεν είναι σε λειτουργία.
- Υποστήριξη ασυμμετρικών μονοπατιών και συνδέσεων.
- Βελτιωμένες εκδόσεις - η mesh δικτύωση είναι διαθέσιμη στο XO αλλά νέες εκδόσεις θα βελτιώσουν της λειτουργικότητα και την συμβατότητα με τα πρότυπα.
- Παράλληλη λειτουργία και ως κόμβος mesh και ως κόμβος υποδομή.
- Συμμόρφωση με τα πρότυπα : ακολουθεί το 802.11s προσχέδιο όπου είναι δυνατό.

### School server

Ο School server του XO, ή XS, είναι ένα λογισμικό που παρέχει επιπρόσθετο αποθηκευτικό χώρο και πρόσβαση το διαδίκτυο :

- XS Server Software.
- XS Installing Software.

### Πρόσθετες εργασίες Λογισμικού

Ακολουθεί μια ενδεικτική λίστα με τις εφαρμογές που έχουν αναπτυχθεί για το XO :

- Measure oscilloscope.
- OLPCities, ένας εικονικός κόσμος προγραμματιστικού περιβάλλοντος.
- FACIL, ένας επεξεργαστής ιστοσελίδων που έχει αναπτυχθεί για να χρησιμοποιείται από παιδιά.
- Musical Editor, ένα εργαλείο για μουσικές συνθέσεις.
- Paint, ένας διαμοιραζόμενος χώρος γραφικών.

- Tux Paint, ένα πρόγραμμα ζωγραφικής εξαιρετικά εύκολο στη χρήση.
- Block Party, ένα παιχνίδι που μοιάζει με το Tetris και εκμεταλεύεται τη mesh δικτύωση.
- Develop, ένας επεξεργαστής εφαρμογών.

### **Builds, images και schedules**

Το OLPC συνήθως μεταγλωττίζει και συναρμολογεί πιο εξελιγμένες εκδόσεις του λογισμικού που αναφέραμε. Αυτή η διαδικασία δημιουργεί images με τις οποίες οι χρήστες του XO μπορούν να αναβαθμίσουν το λογισμικό τους ή οι χρήστες του XO μέσω εξομίσωσης να αναβαθμίσουν το image που χρησιμοποιούν. Αυτές οι εφαρμογές οδηγούν και στις επίσημες διανομές του λογισμικού του.

Το laptop χρησιμοποιεί γραφικό περιβάλλον Sugar, γραμμένο σε Python. Αυτή η διεπαφή δε βασίζεται στη μορφή της τυπικής επιφάνειας εργασίας αλλά προσφέρει μια εικονική οπτική αντίληψη των προγραμμάτων και των εγγράφων και έναν χάρτη των γειτονικών χρηστών που είναι συνδεδεμένοι. Το ενεργό πρόγραμμα καλύπτει όλη την επιφάνεια εργασίας. Οι περισσότερες εφαρμογές πυρήνα του Sugar χρησιμοποιούν εικόνες, προσπερνώντας τα θέματα τοπικοποίησης. Το Sugar επίσης χαρακτηρίζεται από το γεγονός ότι δεν έχει φακέλους στη διεπαφή του.

Ο Steve Jobs προσέφερε το λογισμικό Mac OS X χωρίς χρέωση για χρήση στον υπολογιστή, αλλά σύμφωνα με τον Seymour Papert, έναν επίτιμο καθηγητή του MIT που είναι και ένας από τους αρχικούς ιδρυτές, οι σχεδιαστές θέλουν ένα λειτουργικό σύστημα που να μπορούν να τροποποιήσουν: «Αρνηθήκαμε διότι δεν είναι ανοιχτού κώδικα». Γι αυτόν τον λόγο επιλέχθηκαν τα Linux. Ωστόσο, μετά από συμφωνία με την Microsoft ο υπολογιστής θα έχει και Windows XP καθώς και εναλλακτικό λογισμικό ανοιχτού κώδικα.

### **Activity, home, friends and neighborhood software levels**

Ο Jim Gettys, ο υπεύθυνος για το σύστημα λογισμικού του OLPC, κάλεσε τους προγραμματιστές για επανεκπαίδευση, λέγοντας ότι πολλές εφαρμογές χρησιμοποιούν πολύ μνήμη ή ακόμα έχουν διαρροή μνήμης. «Υπάρχει η εσφαλμένη εντύπωση ότι από τους προγραμματιστές ότι η χρήση της μνήμης είναι καλή, αλλά με το παρόν υλικό είναι πολλές φορές πιο γρήγορο να επαναυπολογιστεί μια μεταβλητή παρά να έχει μια αναφορά στη μνήμη και να την πάρουμε από εκεί. Μια πλήρης αστοχία της κρυφής μνήμης μπορεί να στοιχίσει εκατοντάδες κύκλους ρολογιού και εκατοντάδες φορές περισσότερη κατανάλωση ενέργειας απ' ό,τι μια εντολή που βρίσκεται στο πρώτο επίπεδο της κρυφής μνήμης».

Στις 4 Αυγούστου του 2006, το Ίδρυμα Wikimedia ανακοίνωσε ότι στατικά αντίγραφα επιλεγμένων άρθρων της Wikipedia θα περιλαμβάνονται στους υπολογιστές. Ο Jimmy Wales, επικεφαλής του Ιδρύματος Wikimedia είπε ότι: «Η αποστολή του OLPC πάει χέρι χέρι με τον στόχο μας που είναι η διανομή εγκυκλοπαιδικής γνώσης δωρεάν για όλους τους ανθρώπους στον κόσμο». Δεν έχουν όλοι οι άνθρωποι πρόσβαση σε broadband συνδέσεις. Ο Negreponte είχε υποδείξει νωρίτερα ότι θα ήθελε να δει τη Wikipedia στα laptop. Ο Wales πιστεύει ότι η Wikipedia είναι ένα από τα πιο θετικά στοιχεία του υπολογιστή.

Ο Don Hopkins ανακοίνωσε ότι αναπτύσει ένα δωρεάν και ανοιχτού κώδικα λογισμικό του παιχνιδιού SimCity για το OLPC με τις ευλογίες του Will Wright και της Electronic Arts, κι έκανε επίδειξη του SimCity να τρέχει στο OLPC στο συνέδριο Game Developer's τον Μάρτιο του 2007. Τα σχέδια για το δωρεάν και ανοιχτού κώδικα SimCity επιβεβαιώθηκαν στο ίδιο συνέδριο από τον SJ Klein, διευθυντή περιεχομένου του OLPC, που επίσης ζήτησε από τους προγραμματιστές του παιχνιδιού να δημιουργήσουν εργαλεία με τα οποία τα παιδιά θα μπορούσαν να δημιουργήσουν τα δικά τους αντικείμενα.

Η ασφάλεια αρχιτεκτονικής του laptop παρουσιάστηκε δημόσια τον Φεβρουάριο του 2007. Δεν χρειάζονται κωδικοί για την συνηθισμένη χρήση του υπολογιστή. Στα προγράμματα ανατίθενται συγκεκριμένα πακέτα δικαιωμάτων κατά την εγκατάσταση τα οποία καθορίζουν την πρόσβασή τους στους πόρους. Οι χρήστες μπορούν στη συνέχεια να προσθέσουν δικαιώματα. Προαιρετικά, οι υπολογιστές μπορούν να διαμορφωθούν έτσι ώστε να ζητάνε άδεια από έναν κεντρικό server και να σταματήσουν να λειτουργούν όταν λήξει αυτή η άδεια. Σκοπός της παραπάνω λειτουργίας είναι να δρα ως αντικλεπτικός μηχανισμός. Οι εκδόσεις του λογισμικού πριν το 8.20 είχαν δεχτεί κριτική για την κακή ασύρματη δικτύωση και άλλα ζητήματα μικρής σημασίας.

### **Ομάδες εργασίας λογισμικού για το OLPC**

Το OLPC έχει δημιουργήσει ομάδες εργασίας λογισμικού σε γενικό πλαίσιο για να μπορέσει να καταστήσει δυνατή την επιτυχή ανάπτυξη και χρήση της πρωτοβουλίας OLPC. Οι περισσότερες από αυτές τις ομάδες εργασίας έχουν ομολόγους σε όλο τον κόσμο που ποικίλουν ανάλογα με τον τρόπο προετοιμασίας και διαχείρισης που επιλέγουν για το πρόγραμμα με σκοπό να εκμεταλλευτούν τις δυνάμεις, τις ικανότητες, τους πόρους, την κουλτούρα και τις στρατηγικές κατευθύνσεις κάθε τόπου. Οι κρίσιμοι παράγοντες επιτυχούς ανάπτυξης λογισμικού είναι:

- Παραδείγματα χρήσης του OLPC υψηλής εκπαιδευτικής ποιότητας.
- Συνθήκες για την ανάπτυξη των ανθρώπων και των ιδεών τους.
- Μηχανισμούς για την εξάπλωση των καλών ιδεών και εργαλείων.

Αφού το OLPC είναι μια πολυεθνική προσπάθεια, κάθε έθνος δε χρειάζεται να ανακαλύπτει πράγματα από της αρχή. Ο ένας μαθαίνει από τον άλλον. Η τεχνολογία η ίδια διευκολύνει τη διάδοση ιδεών και παραδειγμάτων. Ένας υπολογιστής είναι τόσο χρήσιμος όσο χρήσιμο είναι το λειτουργικό που τρέχει σε αυτόν. Αυτός είναι ένας από τους λόγους που το XO χρησιμοποιεί προγράμματα ανοιχτού λογισμικού, ώστε να μπορεί κάθε λαός να προσαρμόσει τη λειτουργία του στην ιδιοσυγκρασία του.

Η αποστολή των ομάδων ανάπτυξης λογισμικού είναι το ποιες αρχές πρέπει να ακολουθηθούν και ποιοι μηχανισμοί πρέπει να τεθούν σε εφαρμογή ώστε οι μαθητές και οι οικογένειές τους να λάβουν υψηλή εκπαιδευτική ποιότητα και αξιόλογο λογισμικό. Οι ομάδες εργασίας μπορούν να έχουν μια στρατηγική άποψη και να υιοθετήσουν λογισμικό υψηλής ποιότητας που έχει αναπτυχθεί κάπου αλλού, να το προσαρμόσουν τοπικά στο επίπεδο που είναι απαραίτητο (γλώσσα, τοπικό περιεχόμενο, τοπική κουλτούρα), και να αναπτύξουν νέο λογισμικό που θα συμπληρώσει τα κενά που υπάρχουν. Με αυτόν τον τρόπο μπορεί να μειωθεί το ολικό κόστος.

## Επέκταση

Εάν η χώρα δεν πρόκειται να παρέχει αμέσως laptop για κάθε παιδί, πρέπει να ληφθούν άμεσα αποφάσεις για το πώς θα κατανεμηθούν οι υπολογιστές ώστε να επιτευχθεί ο στόχος της πρωτοβουλίας OLPC. Η επαφή του κάθε παιδιού με έναν υπολογιστή πρόκειται να επιφέρει ουσιαστική βελτίωση στην εκπαίδευση. Επομένως, συστήνουμε ανεπιφύλακτα την προώθηση των υπολογιστών έναν προς έναν σε μια ολόκληρη περιοχή και όχι το μείρασμα τους σε χαμηλή πυκνότητα σε όλη τη χώρα. Η επιλογή κατά σειρά των περιοχών για τη λήψη των laptop δεν είναι μια εύκολη απόφαση. Η εμπειρία έχει δείξει ότι τα μέγιστα οφέλη θα επιτευχθούν όταν στις αρχικές περιοχές υπάρχουν ισχυρά υποδείγματα χρήσης. Αυτό είναι δυνατό μέσω της επιλογής των περιοχών με υψηλή πιθανότητα ορθής εκπαιδευτικής πρακτικής μέσω των υπολογιστών καθώς επίσης και με την κάλυψη κρίσιμων κοινωνικών αναγκών. Μια τακτική που έχει λειτουργήσει καλά είναι να προσκληθούν οι τοπικές κοινότητες για να προτείνουν σε τι θα συμβάλλουν προκειμένου να επιλεγθούν. Αυτό δεν υπονοεί τα κεφάλαια που θα προσφερθούν, παρότι είναι συχνή η υποστήριξη από τους γονείς. Η προσφορά είναι να δοθεί ο χρόνος και η υποστήριξη δασκάλων για να μάθουν πώς να χρησιμοποιούν σωστά τους υπολογιστές ή ακόμη και η κατάρτιση σχεδίων για την καινοτομία στην εκπαίδευση που στηρίζεται στους υπολογιστές.

## Ψηφιακό Περιεχόμενο

Είναι πολύ σημαντικό να ξεκινήσουμε να αναπτύσσουμε νέο περιεχόμενο που εκμεταλλεύεται την ευρεία παρουσία των υπολογιστών. Αντί να επαναχρησιμοποιούμε ήδη υπάρχουσες εφαρμογές, μπορούμε να στραφούμε σε άλλες προοπτικές και πιθανότητες. Έχει διαπιστωθεί ότι οι περιορισμοί που παρουσιάζονται σε εκπαιδευτικά περιβάλλοντα που ήταν απείθαρχα, βελτιώθηκαν όχι μόνο με χρήση νέων μεθοδολογικών προσεγγίσεων, αλλά και μέσω νέων περιεχομένων που εισάγει η τεχνολογία.

## Συνεργασία

Το OLPC δίνει τη δυνατότητα για νέα πιο κατάλληλα μαθησιακά περιβάλλοντα. Νέες κοινωνίες μαθητών μπορούν να συσταθούν γιατί πλέον κάθε παιδί έχει υπολογιστή και οι υπολογιστές μπορούν να συνδεθούν μεταξύ τους. Οι τοίχοι της σχολικής τάξης πλέον καταργούνται και οι μαθητές μπορούν να πάρουν πληροφορίες, να δουλέψουν με άλλους, να ακολουθήσουν τα ενδιαφέροντά τους και τα θέλω τους και να δουλέψουν με τρόπο που πριν θα ήταν αδύνατο.

## Στρατηγική Έρευνα

Το OLPC είναι καινούριο. Δεν υπάρχει μεγάλη εμπειρία σε σχέση με το πώς να χρησιμοποιείται ο υπολογιστής αποδοτικά. Δεν υπάρχει σχεδόν καθόλου εμπειρία στις αναπτυσσόμενες χώρες. Επομένως, είναι απαραίτητο μια ομάδα από διανοούμενους να μελετήσουν, να αναλογιστούν, να συζητήσουν και να προτείνουν βασισμένοι στο τι δουλεύει, τι χρειάζεται βελτίωση και τι νέες ευκαιρίες υπάρχουν. Επίσης, χρήσιμο θα ήταν η ομάδα αυτή να περιλαμβάνει όχι μόνο άτομα που σχετίζονται με την εκπαίδευση. Καλό θα ήταν να περιλαμβάνει διανοούμενους, καλλιτέχνες, δημάρχους για να υπάρχει ποικι-

λομορφία στην εμπειρία και στην αυθεντία.

### 4.2.1 Τι είναι το περιβάλλον Sugar

Το Sugar είναι ο πυρήνας του OLPC Human Interface. Ο σκοπός του είναι να κάνει τον υπολογιστή ευχάριστο, εύκολο στη χρήση, μια κοινωνική εμπειρία που προωθεί τον διαμοιρασμό και την εκπαίδευση. Το αρχικό Interface του είναι το εξής:



Σχήμα 4.9: Το αρχικό Interface του Sugar

Το Sugar είναι ένα περιβάλλον επιφάνειας εργασίας που αρχικά αναπτύχθηκε για το project υπολογιστής/εκπαίδευση της οργάνωσης One laptop per child και από τον Μάιο του 2008 αναπτύσσεται υπό την αιγίδα της εταιρίας Sugar Labs. Το περιβάλλον Sugar χρησιμοποιείται στο OLPC XO-1 laptop και είναι επίσης διαθέσιμο σαν επιλογή συνεδρίας στα λειτουργικά συστήματα Ubuntu και Fedora. Σε αντίθεση με τα συνηθισμένα περιβάλλοντα εργασίας, δε χρησιμοποιεί την μεταφορά της επιφάνειας εργασίας και εστιάζει σε μία λειτουργία τη φορά. Οι κύριοι συντελεστές αυτού του έργου είναι οι Christopher Blizzard και Marco Pesenti Gritti, Eben Eliason, Tomeu Vizoso, Simon Schampijer, Dan Williams, Walter Bender, Christian Schmidt, Lisa Strausfeld, και Takaaki Okada. Η κοινότητα του ανοιχτού λογισμικού έχει συμβάλει σημαντικά για την ανάπτυξη του Sugar. Κυκλοφορεί υπό GNU GPL άδεια κυκλοφορίας, το Sugar είναι ανοιχτού λογισμικού.

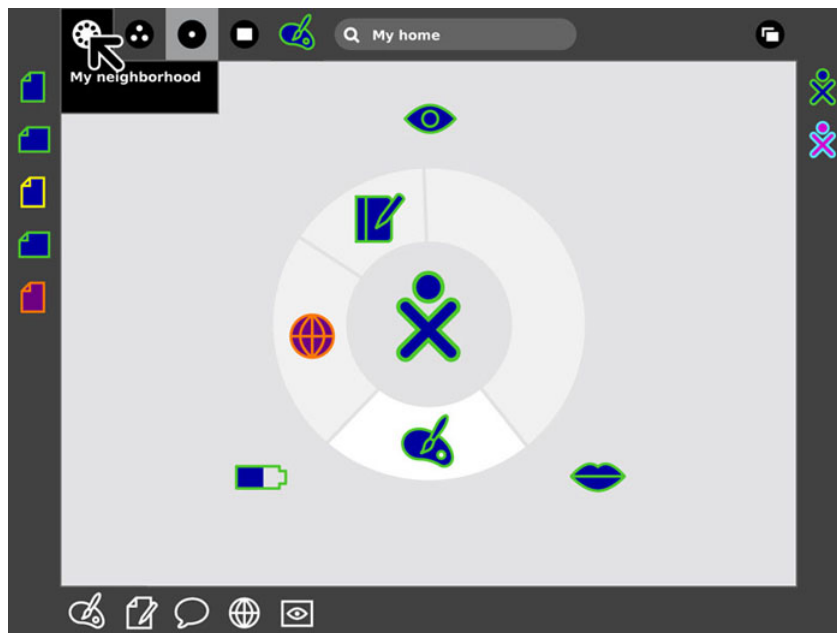
Το Sugar είναι γραμμένο σε γλώσσα προγραμματισμού Python, ενώ τα περισσότερα περιβάλλοντα είναι γραμμένα σε μεταφρασμένες γλώσσες όπως η C. Το Sugar επίσης αναφέρεται και ως OLPC Python περιβάλλον. Συνίσταται από τη γλώσσα Python, GTK GUI και Gecko HTML μηχανή.

Η έκδοση Sugar 0.84 κυκλοφόρησε στις 04/03/2009. Είναι διαθέσιμο ως USB-bootable διανομή Linux ("Sugar on a Stick") και ως λογισμικό κομμάτι που αποτελεί

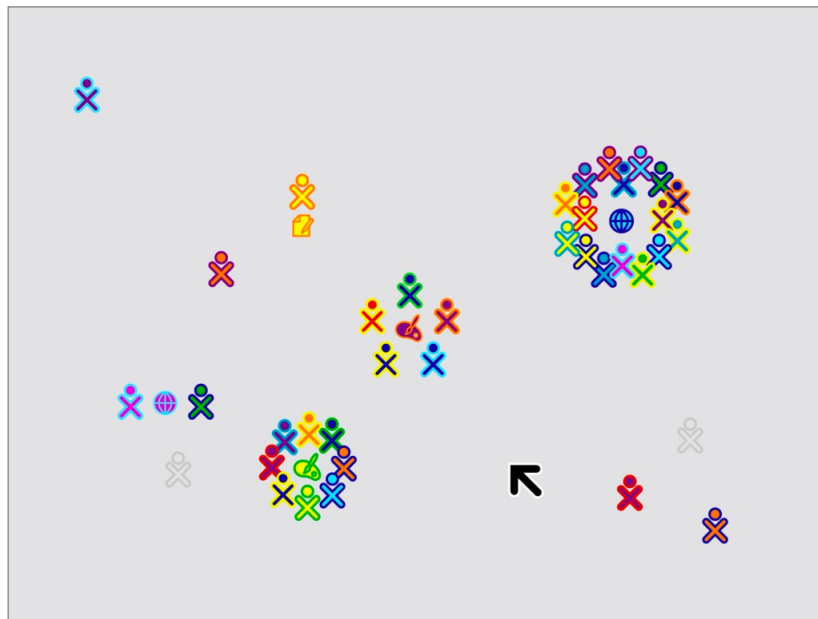


ένα επιπλέον περιβάλλον που μπορεί κάποιος να εγκαταστήσει στις περισσότερες Linux διανομές. Εγκαθίσταται με χρήση του Fedora Live USB Creator, που πλέον δίνει τη δυνατότητα να κατεβάσει κάποιος την έκδοση strawberry του Sugar σε USB.

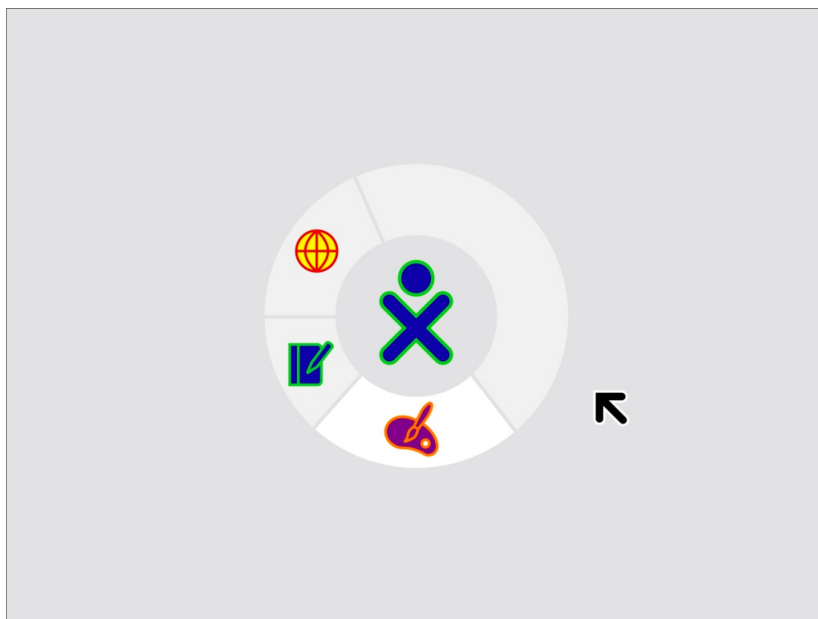
Η έκδοση 0.82.1 του Sugar περιλαμβάνεται στο λογισμικό σύστημα του OLPC για τα laptop XO-1. Το Sugar ακόμα αναπτύσσεται. Τον Μάιο του 2006, οι δημιουργοί του το περιέγραψαν ως ένα πρωταρχικό εργαλείο έκφρασης, ενώ, υπάρχουν σχέδια να συμπεριλαμβάνει χαρακτηριστικά πολυμέσων και κοινωνικής δικτύωσης. Στις αρχές του 2007, το Sugar μπορούσε να εγκατασταθεί (με κάποια δυσκολία) σε μια ποικιλία από λειτουργικά συστήματα, συμπεριλαμβανομένων Linux διανομών, Microsoft Windows και Mac OS X. Οι οδηγίες είναι διαθέσιμες στη σελίδα wiki για το έργο OLPC. Στα μέσα του 2008, το Sugar είναι διαθέσιμο σε Debian, Ubuntu, και Fedora διανομές του Linux, π.χ. στα Ubuntu 8.04 (Hardy Heron), το Sugar μπορεί να εγκατασταθεί από τις επίσημες universe repositories των Ubuntu. Το Fedora Live CD με το Sugar περιβάλλον επιφάνειας εργασίας μπορούμε να το κατεβάσουμε από το site <http://sdz.fedorapeople.org/olpc/> Ακολουθούν μερικές φωτογραφίες του Sugar:



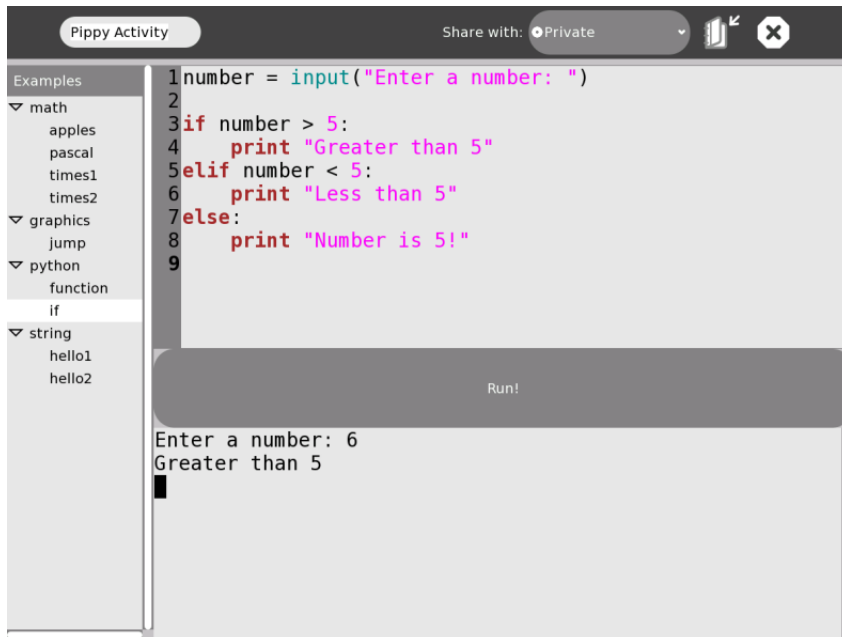
Σχήμα 4.10: Sugar home view



Σχήμα 4.11: Neighborhood



Σχήμα 4.12: Sugar home view from pre-0.82 releases



The screenshot shows the Sugar Pippy activity window. At the top, it says "Pippy Activity" and "Share with: Private". On the left, there is a sidebar with "Examples" and a tree view containing categories like "math", "graphics", "python", and "string". The main area displays Python code for an if-elif-else statement. Below the code is a "Run!" button. The output shows the user entering "6" and the program printing "Greater than 5".

```
1 number = input("Enter a number: ")
2
3 if number > 5:
4     print "Greater than 5"
5 elif number < 5:
6     print "Less than 5"
7 else:
8     print "Number is 5!"
9
```

Run!

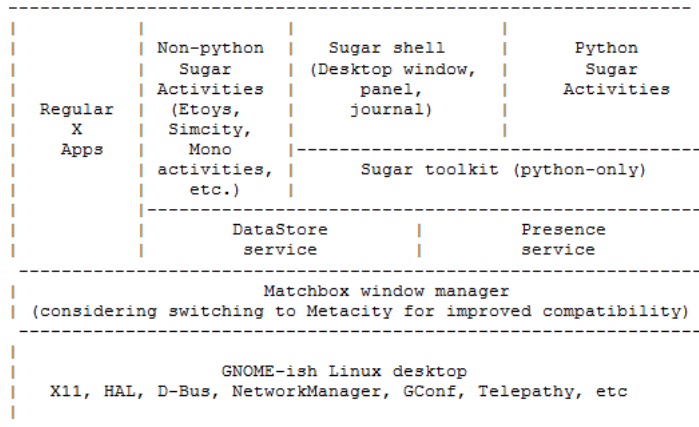
Enter a number: 6  
Greater than 5

Σχήμα 4.13: Sugar Pippy activity





























Η Sugar Labs είναι μια μη κερδοσκοπική οργάνωση με σκοπό να παράγει, να διανείμει και να υποστηρίξει την χρήση την πλατφόρμας εκμάθησης του Sugar. Η Sugar Labs υποστηρίζει την κοινότητα των εκπαιδευτικών και αυτών που αναπτύσσουν με σκοπό να εξελίξουν την πλατφόρμα ή δημιουργούν Sugar Activities. Η Sugar Labs είναι μέλος της Software Freedom Conservancy, μια οργάνωση που αποτελείται από τα projects ανοιχτού λογισμικού.

## 4.2.2 Εφαρμογές που έχει το OLPC

Υπάρχουν τρεις τύποι εφαρμογών που διανέμονται με το βασικό σύστημα: εργαλεία για εξερεύνηση, έκφραση και συνεργασία. Οι εφαρμογές που έχουν αναπτυχθεί ή θα αναπτυχθούν ονομάζονται δραστηριότητες (activities).



Σχήμα 4.14: Activity Chart

	<b>Help:</b> Documentation and Help for the XO		<b>Measure:</b> Oscilloscope and Data Logging
	<b>Browse:</b> Web browser based on xulrunner		<b>Distance:</b> Measure distance between two laptops
	<b>Read:</b> Book/PDF reader		<b>Memorize:</b> A game about finding matching pairs
	<b>Write:</b> Word processor		<b>Implode:</b> Logic game
	<b>Paint (Oficina):</b> Simple paint activity		<b>Maze:</b> Maze game
	<b>Record:</b> Still, video, and audio capture		<b>Moon:</b> Moon phase viewer, includes Lunar phase information, eclipse data, and various display options.
	<b>TamTamJam:</b> Music composition and synthesis. Tamtam has four separate activities : TamTam Jam, Edit, Synthlab, and Mini.		<b>Ruler:</b> Graphical cm/mm ruler and grids to take measurements of lengths and angles of objects the size of XO laptop screen
	<b>TamTamEdit:</b> Music composition and synthesis. See also TamTam Synthlab and Mini.		<b>Scratch:</b> An easy-to-learn, multimedia programming language. Snap together colorful command blocks to create interactive animations, stories, and games, then share your creations on the Scratch website.
	<b>TamTamSynthLab:</b> Music composition and synthesis. See also TamTam Mini.		<b>Speak:</b> An animated face that speaks whatever you type
	<b>TamTamMini:</b> A mini TamTamJam.		<b>Wiki Browse:</b> Offline English Wikipedia snapshot
	<b>Chat:</b> Text chat		<b>Terminal:</b> An activity version of the Sugar terminal
	<b>Piggy:</b> Python Programming language/environment		<b>Log:</b> An activity version of the Sugar logging tool
	<b>Etoys:</b> Learning / programming / authoring environment		<b>Analyze:</b> An activity version of the Sugar analyze tool
	<b>Turtle Art:</b> Pseudo-Logo graphical programming language		
	<b>Calculate:</b> Basic calculator		

Σχήμα 4.15: Activities στο OLPC

### 4.3 Τι είναι η mesh δικτύωση

Η ασύρματη mesh δικτύωση είναι μια mesh δικτύωση που χρησιμοποιεί ασύρματο σύστημα δικτύου, όπως ασύρματο LAN. Το Internet είναι κυρίως βασισμένο στα καλώδια και έχει συνεργατική υποδομή επικοινωνίας όπως και τα ταχυδρομικά πρακτορεία όπου τα μηνύματα παραδίδονται αμοιβαία και διαχωρίζονται ανάλογα με τους τομείς που απευθύνονται, χωρίς χρέωση. Η mesh δικτύωση είναι μια συνεργατική υποδομή επικοινωνίας ανάμεσα σε μεγάλο αριθμό ατομικών ασύρματων πομπών που όλοι έχουν δυνατότητα δρομολόγησης.

Η mesh δικτύωση είναι ένας τρόπος για δρομολόγηση δεδομένων, φωνής και οδηγιών μεταξύ κόμβων. Επιτρέπει τις συνεχείς συνδέσεις και την επαναρρύθμιση των σπασμένων ή μπλοκαρισμένων μονοπατιών πηδώντας από κόμβο σε κόμβο μέχρι να φτάσει στον προορισμό του. Ένα mesh δίκτυο όπου όλοι οι κόμβοι είναι συνδεδεμένοι μεταξύ τους είναι ένα πλήρως συνδεδεμένο δίκτυο. Τα mesh δίκτυα διαφέρουν από τα άλλα δίκτυα στο ότι όλα μέλη από τα οποία συνίσταται συνδέονται μεταξύ τους μέσω πολλαπλών hops και γενικά δεν είναι φορητά. Τα mesh δίκτυα συναντώνται ως τύπος ad hoc δικτύου. Τα φορητά ad hoc δίκτυα (MANET) και τα mesh δίκτυα συσχετίζονται αρκετά αλλά τα MANET αντιμετωπίζουν και τα προβλήματα φορητότητας των κόμβων.

Τα mesh δίκτυα έχουν τη δυνατότητα να επιδιορθώνονται από μόνα τους. Το δίκτυο μπορεί να λειτουργήσει ακόμα κι όταν ένας κόμβος χαλάσει ή χαλάσει η σύνδεση. Ως αποτέλεσμα, το δίκτυο είναι πολύ αξιόπιστο. Αυτή η ιδέα μπορεί να εφαρμοστεί σε ασύρματα και ενσύρματα δίκτυα και αλληλεπιδραστικό λογισμικό.

- Ιστορία

Η ασύρματη mesh δικτύωση έχει τρεις γενιές τεχνολογιών και τοπολογιών. Κάθε μια από αυτές, συμπεριλαμβάνει βελτιώσεις που επιτρέπουν μεγαλύτερο εύρος, μεγαλύτερη αξιοπιστία και αποδοτικό κόστος. Κάποιες αρχικές εκδόσεις το προϊόντων mesh δικτύωσης είχαν χαμηλή απόδοση σε περιβάλλοντα πολλών κόμβων (που συμπεριλάμβαναν πολλές συνδέσεις από κόμβο σε κόμβο). Ωστόσο, οι πιο πρόσφατες γενιές έχουν παρουσιάσει μεγάλη βελτίωση.

- Ασύρματη mesh δικτύωση

Η ασύρματη mesh δικτύωση είναι ένα δίκτυο επικοινωνίας που αποτελείται από ασύρματους κόμβους που οργανώνονται σε mesh τοπολογία. Ένα ασύρματο mesh δίκτυο συχνά αποτελείται από mesh clients, mesh routers και gateways. Οι mesh clients είναι συνήθως φορητοί υπολογιστές, κινητά τηλέφωνα και άλλες ασύρματες συσκευές ενώ οι mesh δρομολογητές προωθούν την κίνηση από και προς τις πύλες που μπορεί αλλά δε χρειάζεται να είναι συνδεδεμένες στον Internet.

Η περιοχή κάλυψης του ασύρματου κόμβου που λειτουργεί σαν ένα δίκτυο ονομάζεται κάποιες φορές mesh σύννεφο. Η πρόσβαση στο mesh σύννεφο εξαρτάται από τους ασύρματους κόμβους που εργάζονται αρμονικά ο ένας με τον άλλον για να δημιουργήσουν ένα ασύρματο δίκτυο. Ένα ασύρματο mesh δίκτυο είναι αξιόπιστο και προσφέρει ποικιλία. Όταν ένας κόμβος δεν μπορεί πλέον να λειτουργήσει, οι υπόλοιποι κόμβοι μπορούν να επικοινωνήσουν μεταξύ τους είτε απ' ευθείας είτε μέσω

κάποιου άλλου κόμβου χρησιμοποιώντας έναν ή περισσότερους ενδιάμεσους κόμβους. Η ασύρματη mesh δικτύωση μπορεί να ενσωματωθεί με διάφορες τεχνολογίες δικτύωσης όπως οι 802.11, 802.16, cellular τεχνολογίες ή συνδυασμό τεχνολογιών διαφορετικών τύπων.

Μπορούμε να θεωρήσουμε την ασύρματη mesh δικτύωση σαν μια ασύρματη ad-hoc δικτύωση ειδικού τύπου. Συχνά θεωρείται ότι όλοι οι κόμβοι στο ασύρματο δίκτυο δεν είναι κινητοί αλλά αυτό δε χρειάζεται να ισχύει. Οι mesh δρομολογητές μπορούν να είναι κινητοί. Συνήθως οι mesh δρομολογητές δεν περιορίζονται σε πόρους σε σύγκριση με άλλους κόμβους στο δίκτυο και ως αποτέλεσμα μπορούν να χρησιμοποιηθούν για να πραγματοποιούν λειτουργίες με πιο έντονη απαίτηση σε πόρους. Με αυτόν τον τρόπο, το ασύρματο mesh δίκτυο διαφέρει από το ad-hoc αφού συνήθως όλοι αυτοί οι κόμβοι έχουν περιορισμένους πόρους.

- Αρχιτεκτονική της mesh ασύρματης δικτύωσης

Η αρχιτεκτονική της ασύρματης mesh δικτύωσης είναι ένα βήμα μπροστά προσφέροντας ένα δίκτυο με μεγάλο εύρος ζώνης για τη συγκεκριμένη περιοχή κάλυψης. Η υποδομή της αρχιτεκτονικής είναι ουσιαστικά ένας δρομολογητής δικτύου χωρίς τα καλώδια ανάμεσα στους κόμβους. Είναι κατασκευασμένη από πρωτοπόρες ασύρματες συσκευές που δε χρειάζονται καλωδίωση σε ενσύρματη πύλη όπως συμβαίνει στα παραδοσιακά WLAN σημεία πρόσβασης. Η mesh αρχιτεκτονική διατηρεί την ισχύ του σήματος διαιρώντας τις μεγάλες αποστάσεις σε μια σειρά μικρότερων διαδρομών. Οι ενδιάμεσοι κόμβοι όχι μόνο ενισχύουν το σήμα αλλά συνεργάζονται ώστε να πάρουν τη σωστή απόφαση προώθησης βασισμένοι στη γνώση του δικτύου. Μια τέτοια αρχιτεκτονική μπορεί με προσεκτικό σχεδιασμό να προσφέρει υψηλό εύρος ζώνης, φασματική αποδοτικότητα, και οικονομικό πλεονέκτημα για την περιοχή κάλυψης. Παραδείγματα των τριών τύπων ασύρματης mesh δικτύωσης:

- Υποδομή του ασύρματου mesh δικτύου: Οι mesh δρομολογητές σχηματίζουν μια υποδομή για τους clients.
- Client ασύρματη mesh δικτύωση: οι κόμβοι clients αποτελούν το δίκτυο και κάνουν τη δρομολόγηση και τις λειτουργίες διαμόρφωσης.
- Υβριδικό ασύρματο mesh δίκτυο: Οι mesh clients μπορούν να εκτελέσουν τις mesh λειτουργίες με άλλους mesh clients όπως επίσης και να έχουν πρόσβαση στο δίκτυο.

Η ασύρματη mesh δικτύωση έχει σχετικά σταθερή τοπολογία εκτός από τις περιστασιακές αποτυχίες των κόμβων ή τις προσθήκες νέων κόμβων. Η κίνηση που συγκεντρώνεται από τον μεγάλο αριθμό των τελευταίων χρηστών αλλάζει σποραδικά. Πρακτικά όλη η κίνηση σε ένα mesh δίκτυο υποδομής είτε προωθείται από και προς μια πύλη ενώ είναι σε ad-hoc δίκτυο είτε η κίνηση ρέει μεταξύ αυθαίρετα ζεύγη κόμβων.

- Διαχείριση

Αυτός ο τύπος υποδομής μπορεί να αποκεντρωθεί (δηλαδή να μην έχει κεντρικό διακομιστή) ή να έχει κέντρο (να έχει κεντρικό διακομιστή). Και οι δύο τρόποι είναι σχετικά φθηνοί, πολύ αξιόπιστοι και ευπροσάρμοστοι, καθώς κάθε κόμβος χρειάζεται μόνο να εκπέμπει ως τον επόμενο κόμβο. Οι κόμβοι δρουν ως δρομολογητές για να εκπέμπουν δεδομένα από κοντινούς κόμβους σε σταθμούς που είναι πολύ μακριά για να τους φτάσουν με μια μετάβαση, με αποτέλεσμα ένα δίκτυο που μπορεί να γεφυρώνει τις μεγάλες αποστάσεις. Η τοπολογία του mesh δικτύου είναι πιο αξιόπιστη καθώς κάθε κόμβος συνδέεται με πολλούς άλλους κόμβους. Αν κάποιος κόμβος σταματήσει να λειτουργεί λόγω προβλήματος στο υλικό ή για οποιονδήποτε άλλο λόγο οι γείτονές του μπορούν να βρουν άλλον δρόμο χρησιμοποιώντας τα πρωτόκολλα δρομολόγησης.

- Εφαρμογές

Τα Mesh δίκτυα μπορεί να περιέχουν σταθερές ή φορητές συσκευές. Οι λύσεις είναι τόσο πολύπλοκες όσο ποικίλες είναι οι ανάγκες επικοινωνίας. Για παράδειγμα, χρησιμοποιείται σε δύσκολα περιβάλλοντα όπως επείγουσες καταστάσεις, σήραγγες, παρακολουθήσεις σε πεδία μάχης, εφαρμογές φορητού βίντεο υψηλής ταχύτητας σε δημόσιες μεταφορές ή αγώνες δρόμου τηλεμετρίας πραγματικού χρόνου. Μια σημαντική εφαρμογή για ασύρματο mesh δίκτυο είναι το VoIP. Χρησιμοποιώντας ένα σχέδιο ποιότητας υπηρεσίας, το ασύρματο mesh μπορεί να υποστηρίξει τη δρομολόγηση μέσω αυτού τοπικών τηλεφωνικών συνδιαλέξεων. Μερικές πρόσφατες εφαρμογές:

- Οι δυνάμεις του στρατού των Ηνωμένων Πολιτειών χρησιμοποιούν την ασύρματη mesh δικτύωση για να συνδέσουν τους υπολογιστές τους, κυρίως σκληροτράχηλα laptops, σε πεδία στρατιωτικών επιχειρήσεων. Δίνει τη δυνατότητα στα στρατεύματα να γνωρίζουν την τοποθεσία και την κατάσταση κάθε στρατιώτη ή πεζοναύτη και να συντονίζουν τις δραστηριότητές τους χωρίς να χρειάζονται καθοδήγηση από το κέντρο επιχειρήσεων.
- Ηλεκτρονικοί μετρητές πλέον εγκαθίστανται σε κατοικίες και μεταφέρουν τα ευρήματά τους ο ένας στον άλλον και τελικά στο κεντρικό γραφείο για να υπολογίζονται οι λογαριασμοί χωρίς την ανάγκη για ανθρώπους να διαβάζουν τους μετρητές ή την ανάγκη να συνδέουν τους μετρητές με καλώδια.
- Τα laptops στο πρόγραμμα one laptop per child χρησιμοποιούν ασύρματη mesh δικτύωση για να δίνεται η δυνατότητα στα παιδιά να ανταλλάσσουν αρχεία και να έχουν πρόσβαση στο Internet παρ' όλη την έλλειψη καλωδίων ή κινητών τηλεφώνων ή οποιασδήποτε άλλης φυσικής σύνδεσης στην περιοχή.

- Λειτουργία

Η αρχή της λειτουργίας έχει ομοιότητες με τον τρόπο που ταξιδεύουν τα πακέτα ανά τον κόσμο με την ενσύρματη σύνδεση. Τα δεδομένα μεταβιβάζονται από τη μια συσκευή στην άλλη μέχρι να φτάσουν στον προορισμό τους. Οι αλγόριθμοι δυναμικής δρομολόγησης έθεταν σε εφαρμογή αυτά σε κάθε συσκευή για να ολοκληρωθεί

η διαδικασία. Για να θέσουν σε λειτουργία πρωτόκολλα δυναμικής δρομολόγησης, κάθε συσκευή πρέπει να μπορεί να επικοινωνήσει πληροφορίες δρομολόγησης στο δίκτυο. Κάθε συσκευή τότε καθορίζει τι να κάνει με τα δεδομένα που έλαβε, είτε να τα μεταβιβάσει στην επόμενη συσκευή είτε να τα κρατήσει, αναλόγως το πρωτόκολλο. Ο αλγόριθμος δρομολόγησης που χρησιμοποιείται θα πρέπει να προσπαθεί πάντα να εξασφαλίσει ότι τα δεδομένα ακολουθούν το κατάλληλο (πιο γρήγορο) μονοπάτι για να φτάσουν στον προορισμό τους.

- **Mesh πολλαπλής μετάδοσης**

Η mesh πολλαπλής μετάδοσης αναφέρεται σε ένα μοναδικό ζεύγος αφοσιωμένων σταθμών ο ένας στον άλλον στο τέλος κάθε link. Αυτό σημαίνει ότι υπάρχει μια μοναδική συχνότητα που χρησιμοποιείται για κάθε ασύρματη μεταφορά και ως αποτέλεσμα μια αποκλειστική CSMA περιοχή σύγκρουσης. Αυτός είναι ένα πραγματικός mesh σύνδεσμος με τον οποίο μπορεί να επιτευχθεί βέλτιστη απόδοση χωρίς υποβάθμιση του εύρους ζώνης στη mesh δικτύωση και χωρίς να προστίθεται καθυστέρηση. Ως αποτέλεσμα, οι εφαρμογές ήχου και βίντεο δουλεύουν όπως ακριβώς θα δούλευαν σε ενσύρματο Ethernet δίκτυο. Στα πραγματικά 802.11 δίκτυα δεν υπάρχει η ιδέα για τη mesh, υπάρχουν μόνο σημεία πρόσβασης (Access Points) και σταθμοί. Επομένως, ένας κόμβος mesh δικτύωσης πολλαπλής μετάδοσης θα δεσμεύσει έναν από τους σταθμούς να δράσει σαν σταθμός, και να συνδέσει έναν γειτονικό κόμβο σε έναν AP σταθμό. Μονά ή διπλά mesh δίκτυα χρησιμοποιούν μέσα προτεραιότητας για να επαναλάβουν το σήμα, που σημαίνει ότι περισσότεροι από δύο κόμβοι είναι στην ίδια περιοχή και συχνότητα σύγκρουσης. Αυτό μπορεί να προκαλέσει υποβάθμιση του εύρους ζώνης και μεγάλη καθυστέρηση.

Το Open802.11s είναι μια συνεργασία για μια ανοιχτού κώδικα υλοποίηση του πρωτοκόλλου IEEE 802.11s για την mesh δικτύωση που περιλαμβάνει και το OLPC. Το λογισμικό που θα προκύψει θα τρέχει σε λειτουργικό σύστημα Linux σε κλασικό υλικό ενός PC. Στόχος είναι η δημιουργία μιας 802.11s υλοποίησης πάνω από την mac80211 στοιβά.





## 5 Ανάπτυξη εφαρμογής για το OLPC

### 5.1 Γλώσσες προγραμματισμού για ανάπτυξη εφαρμογής στο OLPC

Το laptop έχει Python 2.5 καθώς και το Python κέλυφος διεπαφής Pippy. Έχει επίσης, Javascript στο xulrunner, Squeak και Smalltalk στο Etoys, και Csound στο TamTam. Το Turtle Art επιτρέπει υλικά επιβεβλημένο προγραμματισμό με την ένωση μπλόκ γραφικών στοιχείων, αλλά προς το παρόν δεν υπάρχει ανάλογη γλώσσα με τη LOGO.

- Ο εκκινητής του laptop έχει ανοιχτό Firmware που είναι μια διάλεκτος FORTH
- Άλλες γλώσσες προγραμματισμού γενικού σκοπού είναι διαθέσιμες ως ανοιχτό λογισμικό για Linux όπως οι LOGO, Smalltalk/gst, LISP/Scheme/Guile, Perl, Ruby, bash, awk, TCL, C, C++, Java, APL/A+/J, FORTH, FORTRAN, Ada, Haskell, Pascal, Erlang, Kaya, Lua, Oz, Ocaml, Pike, Scala, Slate, και SPL.
- Το Debian Jr. Programming (junior-programming) είναι ένα πακέτο γλωσσών που απευθύνονται σε παιδιά και περιλαμβάνονται οι Logo και Littlewizard.
- Έχει γίνει έρευνα στη χρήση των APL, LISP και LOGO, τουλάχιστον σε αίθουσες του δημοτικού.

### 5.2 Η γλώσσα προγραμματισμού Python

Η Python είναι μια δυναμική αντικειμενοστραφής γλώσσα που μπορεί να χρησιμοποιηθεί σε πολλά είδη ανάπτυξης λογισμικού. Προσφέρει τη δυνατότητα για ενσωμάτωση με άλλες γλώσσες και εργαλεία, έχει εκτεταμένες πρότυπες βιβλιοθήκες και μπορεί κάποιος να τη μάθει σε λίγες μέρες. Πολλοί προγραμματιστές Python αναφέρουν ότι έχει

απτή παραγωγικότητα και ενθαρρύνει την ανάπτυξη κώδικα υψηλής ποιότητας και συντηρησιμότητας. Η Python τρέχει σε Windows, Linux/Unix, Mac OS X, OS/2, Amiga, Palm Handhelds, και Nokia κινητά τηλέφωνα. Η Python επίσης έχει ενταχθεί σε Java και .NET εξομοιωτές.

Η Python διανέμεται υπό OSI-approved άδεια κυκλοφορίας ανοιχτού λογισμικού οπότε είναι δωρεάν ακόμα και για διαφημιστικά προϊόντα. Το Python Software Foundation (PSF) έχει στην κατοχή του και προστατεύει τα πνευματικά δικαιώματα της Python που κατοχυρώνονται από το PyCon conference, και χορηγεί κονδύλια για κάποια projects της κοινότητας Python.

Η Python πρωτοδημιουργήθηκε από τον Guido van Rossum έναν Ολλανδό προγραμματιστή το 1990 και πήρε το όνομά της από το Brit-com Monty Python's Flying Circus μια κωμωδία κινουμένων σχεδίων του BBC από την κωμική ομάδα Monty Python. Από τότε έχει αναπτυχθεί μια μεγάλη ομάδα εθελοντών. Η πιο πρόσφατη έκδοση της είναι η 2.5 και κυκλοφόρησε τον Σεπτέμβριο του 2006. Υπάρχει επίσης μια Python εφαρμογή γραμμένη σε Python που ονομάζεται έργο PyPy. Το έργο ξεκίνησε το 2003 με σκοπό να διευκολύνει τους προγραμματιστές να αλλάζουν τη συμπεριφορά του μεταγλωττιστή Python ανάλογα με τις επιθυμίες τους. Το PyPy υποστηρίζεται από την Ευρωπαϊκή Ένωση ως Specified Targeted Research Project (STReP), μέρος του FP6 funding program.

Η Python 2.0 κυκλοφόρησε στις 16 Οκτωβρίου 2000, με πολλά σημαντικά χαρακτηριστικά συμπεριλαμβανομένου garbage collector και υποστήριξη Unicode. Ωστόσο, η πιο σημαντική αλλαγή ήταν η διαδικασία ανάπτυξης με στροφή στην διάφανη και υποστηριζόμενη από κοινότητα συγγραφή κώδικα. Η έκδοση Python 3.0 κυκλοφόρησε στις 3 Δεκέμβρη του 2008 ύστερα από μεγάλο χρονικό διάστημα δοκιμών. Μερικά πολύ χαρακτηριστικά στοιχεία της Python είναι τα εξής:

- Καθαρή και ευανάγνωστη σύνταξη.
- Δυνατές ενδοσκοπικές δυνατότητες.
- Ενορατική αντικειμενοστρέφεια.
- Φυσική έκφραση διαδικαστικού κώδικα.
- Πλήρη modularity, που υποστηρίζει ιεραρχία πακέτων.
- Αντιμετώπιση λαθών βασισμένη στις εξαιρέσεις.
- Πολύ υψηλής ποιότητας τύποι δυναμικών δεδομένων.
- Εκτεταμένες στάνταρ βιβλιοθήκες και third party modules for virtually every task.
- Επεκτάσεις και modules εύκολα γραμμένες σε C, C++ (ή Java για Jython, ή .NET γλώσσες για IronPython).
- Δυνατότητα ενσωμάτωσης σε εφαρμογές ως scripting διεπαφή.

## Η Python είναι δυνατή και γρήγορη

Οι θαυμαστές της Python χρησιμοποιούν την έκφραση «συμπεριλαμβάνονται οι μπαταρίες» για να περιγράψουν τις πρότυπες βιβλιοθήκες, που καλύπτουν τα πάντα από ασύγχρονη επεξεργασία ως αρχεία zip. Η ίδια η γλώσσα είναι ένα ευέλικτος παράγοντας που μπορεί να αντιμετωπίσει ουσιαστικά προβλήματα οποιουδήποτε τομέα. Μπορεί κάποιος να φτιάξει την ιστοσελίδα του με τρεις γραμμές κώδικα.

Η Python σου επιτρέπει να γράψεις τον κώδικα που χρειάζεσαι γρήγορα και χάρις σε byte μεταφραστή υψηλής ποιότητας και τις βιβλιοθήκες, ο κώδικας τρέχει αρκετά γρήγορα για όλες τις εφαρμογές.

## Η Python τα πάει καλά με τους άλλους

Η Python μπορεί να ενσωματωθεί σε COM, .NET, και CORBA αντικείμενα. Για βιβλιοθήκες Java μπορούμε να χρησιμοποιήσουμε Jython, μια εφαρμογή της Python για τις Java εικονικές μηχανές. Για .NET, μπορούμε να χρησιμοποιήσουμε IronPython, μια καινούρια εφαρμογή της Microsoft για την Python. Η Python επιπλέον υποστηρίζεται από την Internet Communications Engine (ICE) και πολλές άλλες τεχνολογίες ενσωμάτωσης.

Αν εντοπίσετε κάτι που η Python δεν μπορεί να το κάνει, ή αν χρειάζεστε τα πλεονεκτήματα της απόδοσης του κώδικα χαμηλού επιπέδου, μπορείτε να υλοποιήσετε επεκτάσεις γραμμένες σε C ή C++ ή να εντάξετε υπάρχοντες κώδικες με SWIG ή Boost-Python. Οι Wrapped modules κάνουν το πρόγραμμα σας να φαίνεται ακριβώς σαν αυθεντικός Python κώδικας. Μπορείτε επίσης να επιλέξετε τον αντίθετο τρόπο και να εμφωλεύσετε την Python στην εφαρμογή σας, δίνοντας στους χρήστες τη δυνατότητα να έρθουν σε επαφή με μια γλώσσα που θα απολαύσουν να χρησιμοποιούν.

## Η Python τρέχει παντού

Όπως ήδη αναφέραμε η Python είναι διαθέσιμη για όλα τα κύρια λειτουργικά συστήματα. Αυτό που είναι όμως χαρακτηριστικό είναι ότι σε όλα τα συστήματα τρέχει χωρίς χρειάζεται να γίνει η παραμικρή αλλαγή στον κώδικα. Σε περίπτωση που το σύστημα που χρησιμοποιείται δεν βρίσκεται στη λίστα αυτών που υποστηρίζουν Python, αν διαθέτουν μεταγλωττιστή για τη C μπορείτε να την ενσωματώσετε στο σύστημα σας ή μπορείτε να προσπαθήσετε να την μεταγλωττίσετε εσείς.

## Η Python είναι φιλική και μαθαίνεται εύκολα

Το Python newsgroup είναι γνωστό ως ένα από τα πλέον φιλικά. Η κοινότητα των ενθουσιωδών προγραμματιστών και χρηστών έχουν μια σελίδα wiki, που φιλοξενεί διενθεις και τοπικές συνεδρίες, αναπτύσσεται ταχύτατα και συμβάλει σε online βιβλιοθήκες κώδικα. Η Python έχει πλήρη βιβλιογραφία, και ενσωματωμένη στη γλώσσα αλλά και σε μορφή ιστοσελίδας. Τα εγχειρίδια που υπάρχουν στο διαδίκτυο έχουν στόχο και τον έμπειρο προγραμματιστή αλλά και τον νέο. Όλα είναι σχεδιασμένα ώστε να γίνει κάποιος παραγωγικός σε μικρό χρονικό διάστημα. Η διαθεσιμότητα των βιβλίων που έχουν αξιολογηθεί ως πρώτης κατηγορίας ολοκληρώνουν το πακέτο εκμάθησης της Python.

### 5.3 Πως αναπτύσσεται μια εφαρμογή στο περιβάλλον Sugar που εκμεταλλεύεται τη mesh δικτύωση

Το πρώτο βήμα για τη ανάπτυξη μιας εφαρμογής είναι να σκεφτούμε μια καλή ιδέα. Το OLPC δεν είναι απλά ένας ευφάνταστος υπολογιστής που χρησιμοποιούν τα παιδιά αλλά είναι κι ένας τρόπος να ξανασκεφτούμε πως μπορεί να χρησιμοποιηθεί ο υπολογιστής σαν εκπαιδευτικό εργαλείο. Με δεδομένο αυτό υπάρχουν πολλές σχεδιαστικές παράμετροι που πρέπει κάποιος να λάβει υπ' όψιν για να αναπτύξει μια εφαρμογή. Συγκεκριμένα πρέπει η εφαρμογή να περιλαμβάνει :

- Ανακαλυψιμότητα: Η εφαρμογή πρέπει να προωθεί την εκπαίδευση. Τα μεγάλα εγχειρίδια οδηγιών είναι βαρετά και δύσκολο να μεταφραστούν. Όσο πιο ενορατική είναι η διεπαφή τόσο το καλύτερο. Αυτό σημαίνει ότι πρέπει να χρησιμοποιηθεί ένα αλληλεπιδραστικό μοντέλο που ενθαρρύνει τον πειραματισμό, χρησιμοποιώντας παρεμφατικά κουμπιά αντί για κείμενο, κ.α.
- Επεκτασιμότητα: Η εφαρμογή πρέπει να μπορεί να τροποποιηθεί από τους δασκάλους και τα παιδιά για να επεκτείνουν τη χρήση του. Για παράδειγμα, το κουίζ δίνει τη δυνατότητα στον δάσκαλο να αλλάζει τις ερωτήσεις, έτσι ώστε να μπορεί να χρησιμοποιηθεί για οποιεσδήποτε ερωτήσεις και όχι μόνο αυτές που έχει σκεφτεί ο δημιουργός της εφαρμογής.
- Συνεργασία: Πρέπει η εφαρμογή να μπορεί να χρησιμοποιηθεί από περισσότερους από έναν μαθητές την ίδια στιγμή. Οι καλές εφαρμογές πρέπει να έχουν και single και multi-player εφόσον βέβαια είναι εφικτό για το είδος της εφαρμογής. Για παράδειγμα, ο κειμενογράφος επιτρέπει να γράφουν στο ίδιο αρχείο πολλοί χρήστες.

Για να κατασκευάσουμε μια εφαρμογή για το OLPC πρέπει επίσης να λάβουμε υπ' όψιν τις αρχές που υπάρχουν στον πυρήνα του XO της διεπαφής του Sugar. Πρέπει για μια πιο λεπτομερή ενημέρωση να μελετηθούν οι οδηγίες χρήσης του OLPC human interface και οι οδηγίες για τις εκπαιδευτικές εφαρμογές.

Επιπλέον, είναι πολύ σημαντικό για να αναπτύξουμε μια εφαρμογή να μελετήσουμε και τις εφαρμογές που έχουν αναπτυχθεί στο παρελθόν. Αξίζει να αναφέρουμε ότι αν κάποιος ενδιαφέρεται να πάρει μέρος στη διαδικασία, μπορεί να ενταχθεί στις υπάρχουσες εθελοντικές ομάδες ανάπτυξης εφαρμογών, οι οποίες περιλαμβάνουν καλλιτέχνες, εκπαιδευτικούς και προγραμματιστές. Για να πάρει μέρος κάποιος στην ανάπτυξη μιας εφαρμογής δε χρειάζεται να ξέρει τα πάντα περί των εργασιών της.

#### Σχεδιασμός

Πριν ξεκινήσει κάποιος να αναπτύσσει μια εφαρμογή πρέπει να λάβει υπ' όψιν του κάποια πολύ σημαντικά σχεδιαστικά στοιχεία του υλικού του OLPC.

#### Ανάπτυξη

Όταν κάποιος ξεκινάει να αναπτύσσει την εφαρμογή, είναι πολλά τα πράγματα που πρέπει να κάνει. Αρχικά, αν κάποιος δεν έχει προγραμματιστική εμπειρία πρέπει να

συμβουλευτεί εγχειρίδια για αρχάριους προγραμματιστές. Ενώ δουλεύει με τα εγχειρίδια (ή αναπτύσσει κάποιος τον κώδικα της εφαρμογής αν έχει προγραμματιστική εμπειρία), πρέπει να αρχίσει να συμμετέχει στην κοινότητα του OLPC. Πρέπει να γραφτεί σε μια λίστα ηλεκτρονικού ταχυδρομείου. Σε αυτές τις λίστες συζητούνται οι πιο πρόσφατες εξελίξεις του OLPC. Για αυτούς που αναπτύσσουν μια εφαρμογή, είναι επίσης χρήσιμο να γραφτούν στις λίστες του Sugar και των παιχνιδιών ώστε να μπορούν να επιταχύνουν την διαδικασία. Για αυτούς που ενδιαφέρονται για πολλές λεπτομέρειες όσον αφορά την πλατφόρμα του υλικού μπορούν να γραφτούν στη λίστα devel που περιέχει τις τελευταίες πληροφορίες.

### Προγραμματισμός της εφαρμογής

Υπάρχουν δύο βασικοί τρόποι να γράψει κάποιος εφαρμογή για το XO, και οι δύο περιλαμβάνουν Python:

- pyGTK - είναι η γλώσσα προγραμματισμού που έχει γραφτεί το Sugar και είναι χρήσιμη για εφαρμογές που περιλαμβάνουν πολλά κουμπιά και widgets.
- pygame - είναι ένα πλαίσιο για κατασκευή παιχνιδιών που δεν περιλαμβάνουν πολλά κουμπιά και widgets αλλά αντιθέτως λειτουργούν μετακινώντας εικόνες γύρω γύρω στην οθόνη.

Η pyGTK και η pygame είναι πρότυπα της Python που δουλεύουν σε πολλές πλατφόρμες. Αυτό μας επιτρέπει να ξεκινήσουμε να αναπτύσσουμε μια εφαρμογή χωρίς να έχουμε ένα XO στην κατοχή μας και χωρίς να χρειάζεται να προσπαθήσουμε να εξομοιώσουμε το Sugar. Ο καλύτερος τρόπος για να ξεκινήσουμε είναι να αποφασίσουμε με ποιο πρότυπο της Python θα αναπτύξουμε την εφαρμογή μας και να το μελετήσουμε. Ένα καλό σημείο εκκίνησης για την εφαρμογή μας είναι να κατεβάσουμε ένα παράδειγμα που λειτουργεί και να το αλλάξουμε ώστε να ταιριάζει με τον σχεδιασμό της εφαρμογής μας. Σε αυτό το σημείο είμαστε έτοιμοι για να ανεβάσουμε τον κώδικά μας στο OLPC.

### Όταν έχετε τουλάχιστον δέκα γραμμές κώδικα

Σε αυτό το σημείο είμαστε έτοιμοι να ανεβάσουμε τον κώδικά μας σε ένα αρχείο ώστε οι άλλοι σχεδιαστές να μπορούν να τον δουν. Επίσης, πρέπει να δημιουργήσουμε εισιτήρια ίχνους για να μπορούμε να εντοπίσουμε bugs και να αναδείξουμε αιτήσεις. Ένα από τα βασικά συστήματα επικοινωνίας του OLPC, είναι μέσω του [dev.laptop.org](http://dev.laptop.org) όπου οι άλλοι μπορούν να δουν την κατάσταση της εφαρμογής μας, καθώς προετοιμάζεται η έκδοσή της, οι δοκιμαστές μπορούν να αναφέρουν λάθη και οι μεταφραστές μπορούν να υποβάλλουν μεταφράσεις της εφαρμογής σας. Χρειάζεται να κάνετε τα εξής για να στήσετε αυτό το σύστημα :

- Να ζητήσετε project hosting: πρέπει να κάνετε μια αίτηση και να τη στείλετε στη λίστα devel. Αν έχετε μια ωραία σελίδα στο wiki που υπογραμμίζει το τι θέλετε να κάνετε και περιέχει και μέρος του κώδικα είστε έτοιμοι για την εφαρμογή.
- Να κάνετε αίτηση για ένα git tree, καθώς και ένα trac component για την εφαρμογή σας.

- Να προσθέσετε μια άδεια GPL στα αρχεία της εργασίας σας γιατί το repository του OLPC είναι δημόσιο, οπότε πρέπει να είστε προσεκτικοί ώστε να μην συμπεριλάβετε ιδιωτικό κώδικα. Επίσης, να σιγουρευτείτε ότι χρησιμοποιείτε άδεια ανοιχτού λογισμικού στον κώδικά σας ώστε κάποιος να μπορεί να τον τροποποιήσει ή να τον επεκτείνει με ευκολία. Οι οδηγίες του GPL περί της χρήσης του περιλαμβάνουν απλά βήματα που πρέπει να ακολουθήσετε (να συμπεριλάβετε ένα αντίγραφο της άδειας, να έχετε μια κατοχύρωση της πνευματικής ιδιοκτησίας σε κάθε αρχείο κ.α.).
- Να εισάγετε την εργασία σας σε ένα git tree.

### **Κάνοντας της εφαρμογή να δουλέψει στο XO**

Είναι καλό να δοκιμάσετε την εφαρμογή σας σε εξομοίωση του XO αργά ή γρήγορα γιατί υπάρχουν κάποια θέματα που μπορεί να προκύψουν. Αρχικά, η οθόνη του XO είναι μικρή σε μέγεθος αλλά έχει μεγάλη ανάλυση. Επιπλέον, η ταχύτητα μπορεί να είναι θέμα διότι ο υπολογιστής που αναπτύσσετε την εφαρμογή σας είναι πολύ πιο γρήγορος από το XO. Παρ' όλα αυτά για να τρέξετε την εφαρμογή σας πρέπει να ακολουθήσετε πολλά βήματα.

- Δημιουργείστε ένα περιτύλιγμα για την εφαρμογή σας.
- Φτιάξτε την εφαρμογή σας σε πακέτο.
- Βάλτε την εφαρμογή σας σε ένα XO ή σε εξομοιωτή.
- Tweak, tweak, tweak.

### **Κάνοντας την εφαρμογή κατάλληλη για Sugar**

Αναλόγως αν έχετε χρησιμοποιήσει pyGTK ή pygame αυτή η διαδικασία διαφέρει ελαφρώς. Και για τα δύο πρότυπα, πρέπει να ξεκινήσετε ακολουθώντας τα βήματα που δίνονται στο Sugar Activity Tutorial. Αυτό το εγχειρίδιο περιέχει τη βασική δομή του καταλόγου που χρειάζεστε για την εφαρμογή σας, καθώς και οδηγίες για το πως να φτιάξετε ένα icon για την εφαρμογή σας. Ωστόσο, δεν μπορείτε να ολοκληρώσετε τις οδηγίες του εγχειριδίου αν δεν έχετε Sugar εγκατεστημένο στον υπολογιστή σας.

### **Φτιάχνοντας την εφαρμογή σε πακέτο**

Μπορούμε να φτιάξουμε την εφαρμογή μας σε πακέτο ανεξάρτητα αν έχουμε χρησιμοποιήσει pyGTK ή pygame. Προς το παρόν δε θα το φτιάξουμε χρησιμοποιώντας το Sugar bundle builder γιατί μας ενδιαφέρει αν δουλεύει η εφαρμογή. Θα το διορθώσουμε αργότερα όταν θα θέλουμε να δώσουμε την εφαρμογή σε δοκιμαστές για να δοκιμάσουν την εφαρμογή. Βεβαιωθείτε ότι όλα τα αρχεία της εφαρμογής σας περιλαμβάνονται στο MANIFEST και βάλτε τον παρακάτω κώδικα στο αρχείο setup.py (αντικαθιστώντας το YourActivityName με το όνομα της εφαρμογής σας)

```

1 \#!/usr/bin/env python
2 try:
3     from sugar.activity import bundlebuilder
4     bundlebuilder.start("YourActivityName")
5 except ImportError:
6     import os
7     os.chdir('.')
8     os.system('cat YourActivityName.activity/MANIFEST|zip YourActivityName.xo -@')
9     os.system('mv YourActivityName.xo ./YourActivityName.activity')
10    os.chdir('YourActivityName.activity')

```

Listing 5.1: setup.py

Με αυτόν τον τρόπο μπορείτε να τρέξετε το:

```

1 ./setup.py

```

Listing 5.2: setup

Χωρίς να έχετε εγκατεστημένο το Sugar, στο τέλος θα βρείτε ένα zip αρχείο .xo που είναι έτοιμο για να βάλετε στον εξομοιωτή του XO.

### Βάζοντας την εφαρμογή σε εξομοιωτή του XO

Πρέπει να περάσετε το .xo αρχείο μέσα στον εξομοιωτή για να το δοκιμάσετε. Στο τερματικό του προγραμματιστή πηγαίνετε στον κατάλογο /home/olpc. Φτιάξτε ένα κατάλογο με το όνομα Activities μέσα στον /home/olpc, αν δεν υπάρχει ήδη. Ξεπακετάρετε το .xo αρχείο μέσα στον κατάλογο Activities (χρησιμοποιώντας την εντολή unzip). Πρέπει να δημιουργηθεί ένας κατάλογος /home/olpc/Activities/YourActivityName.activity. Επανεκκινήστε τον X-server (Ctrl-Alt-Backspace). Όταν φορτώσει το Sugar θα πρέπει να δείτε ένα εικονίδιο με τη νέα σας εφαρμογή στη μπάρα. Κάντε κλικ και δοκιμάστε την. Εναλλακτικά μπορείτε να βάλετε την εφαρμογή σας στον κατάλογο /usr/share/activities, αντί για τον /home/olpc/Activities.

### Αντιμετώπιση προβλημάτων

Αν δεν εμφανιστεί το εικονίδιο της εφαρμογής σας, μπορεί να υπάρχουν πολλές εφαρμογές εγκατεστημένες. Μετονομάστε κάποιες από αυτές σε name.activity.bak, και επανεκκινήστε. Αν κρασάρει, πηγαίνετε στην κονσόλα και ελέγξτε τα logs που σχετίζονται με την εφαρμογή σας.

### Tweak, tweak, tweak

Δοκιμάστε:

```

1     import sugar
2     dev = 0
3 except ImportError:
4     dev = 1
5
6 if dev:
7     SCREENRECT = Rect(0,0,400,300)
8     scale\_x = 1./3.

```

```
9     scale\_y = 1./3.
10    FONT\_SIZE = 36
11 else:
12    SCREENRECT = Rect(0,0,1200,825)
13    scale\_x = 1.
14    scale\_y = 1.
15    FONT\_SIZE = 36
```

Listing 5.3: tweak.py

### **110n και i18n**

Το 110n σημαίνει τοπικοποίηση (ημερομηνία, αριθμός, και currency formats), ενώ i18n σημαίνει παγκοσμιοποίηση (μετάφραση).

### **Φτιάχνοντας την εφαρμογή σε πακέτο με χρήση του Sugar Bundle Builder**

Αν θέλουμε να διανέμουμε την εφαρμογή μας ώστε να την δοκιμάσει κάποιος πρέπει να μπούμε μέσα στον φάκελο που περιέχει τα αρχεία της εφαρμογής μας και να πληκτρολογήσουμε:

```
1 python setup.py dist
```

Listing 5.4: python setup

Και στη συνέχεια όταν κάποιος θέλει να εγκαταστήσει την εφαρμογή πρέπει να πληκτρολογήσει:

```
1 sugar-install-bundle ExampleActivity-1.xo
```

Listing 5.5: Bundle Builder

Όπου το ExampleActivity το αντικαθιστούμε με τον όνομα που έχουμε δώσει στην εφαρμογή μας.

### **Δοκιμή και κριτική**

Αφού έχει εμφανιστεί το εικονίδιο της εφαρμογής σας στην επιφάνειας εργασίας του Sugar μπορείτε να δοκιμάσετε τη λειτουργικότητά του

### **Βάζοντας την εφαρμογή σας στον πυρήνα**

Έχουν γίνει συζητήσεις στη λίστα devel για τα κριτήρια που πρέπει να πληρεί μια εφαρμογή για να συμπεριληφθεί στον πυρήνα του XO. Κάποια προτεινόμενα κριτήρια είναι τα εξής:

- Επιστημονικό αντίκτυπο και σε ποιο βαθμό η εφαρμογή συμβάλει θετικά στην μάθηση που είναι και το πιο βασικό κριτήριο φυσικά·
- Είναι διασκεδαστικό και απολαυστικό·



- Ποιότητα - είναι η εφαρμογή αρκετά εύρωστη στη λειτουργία της ώστε να μην διακινδυνεύει η ακεραιότητα ή η υποστηρικτικότητα του συστήματος. Είναι η ποιότητα της εφαρμογής ανάξια των προσδοκιών μας. Μπορεί η κοινότητα να συμμετέχει στη διαδικασία δοκιμής, πιστοποίησης και συντήρησης της εφαρμογής.
- Συσχέτιση με Sugar - σε ποια έκταση έχει ενταχθεί η εφαρμογή στο περιβάλλον Sugar, συμπεριλαμβανομένων του Sugar interface, Journal, ασφάλεια, παγκοσμιοποίηση κ.α. Απαιτεί η εφαρμογή εμφώλευση επιπλέον βιβλιοθηκών και πόρων. (Αυτό έχει αντίκτυπο στην ευρωστία - θετική και αρνητική - υποστήριξη, τον μετεωρισμό και την ολική χρησιμότητα, την αισθητική και την αντίληψη ποιότητας την μηχανής).
- FOSS - είναι η εφαρμογή αποκλειστικά ανοιχτού λογισμικού και δωρεάν.
- Επεκτασιμότητα - είναι η εφαρμογή τέτοια ώστε να μπορεί η κοινότητα να την επεκτείνει. Εκτείνεται σε πολλαπλές ανάγκες. (έχει ή έχει την προοπτική για μια θετική κοινότητα υποστήριξης).
- Μοναδικότητα - προσθέτει η εφαρμογή κάτι μοναδικό στον πυρήνα.
- Προσδοκίες - ανταποκρίνεται η εφαρμογή στις προσδοκίες των παιδιών, δασκάλων, γονέων, G1G1, κοινό κ.α.
- Ανακαλυψιμότητα - είναι η εφαρμογή πυρήνα ανακαλύψιμη. (αυτό δε σημαίνει ότι δε θα είναι δύσκολο να εκμεταλλευτείς τη δύναμη της εφαρμογής, αλλά θα πρέπει να έχει ένα μικρό εμπόδιο εισόδου).

## 5.4 Η εφαρμογή μας : Λεξο-μαχίες (Word-fight)

Η εφαρμογή που υλοποιήσαμε είναι ένα εκπαιδευτικό παιχνίδι με λέξεις. Ο εκπαιδευτικός θα εισάγει λέξεις ανάλογα με το μάθημα πάνω στο οποίο θέλει να εφαρμόσει το παιχνίδι. Οι μαθητές ανοίγουν την εφαρμογή και βλέπουν τις λέξεις που έχει εισάγει ο εκπαιδευτικός. Παράλληλα, ανοίγει ένα δεύτερο παράθυρο όπου εισάγουν το όνομα τους και την πρόταση που έχουν σκεφτεί που περιλαμβάνει μία από τις λέξεις. Τέλος, πατάνε το κουμπί για να σταλούν τα δεδομένα στο βασικό πρόγραμμα και εμφανίζεται το όνομα που έχουν εισάγει και η πρόταση ώστε να μπορούν να τα δουν όλοι οι συμμετέχοντες.

Όταν έχουν παίξει όλοι οι μαθητές ή έχουν χρησιμοποιηθεί όλες οι λέξεις το παιχνίδι τελειώνει και τότε όλοι οι συμμετέχοντες μπορούν να συζητήσουν για τις προτάσεις και τη συνολική εικόνα της πορείας και των αποτελεσμάτων του παιχνιδιού.

Η υλοποίηση της εφαρμογής δεν απαιτήσε πολλές γραμμές κώδικα γιατί η Python προσφέρει πολύ χρήσιμα εργαλεία στον προγραμματιστή. Εκμεταλλευτήκαμε τη λειτουργία των κλάσεων και των συναρτήσεων.

### 5.4.1 Λόγοι επιλογής της εφαρμογής

- Θέλαμε μια εφαρμογή να μπορεί να εφαρμοστεί σε όλα τα μαθήματα, ακόμα και στα μαθηματικά, όπως π.χ. για τα θεωρήματα.

- Θέλαμε να είναι απλοί οι κανόνες, εύκολοι και για την δασκάλα και για τους μαθητές.
- Να έχει χρονικό περιθώριο για να θέτει τα παιδιά σε εγρήγορση.
- Αλληλεπίδραση μεταξύ μαθητών (ο ένας βλέπει την απάντηση του άλλου και μαθαίνει από την πρόταση που εισήγαγε).
- Δεν εισάγει τον ανταγωνισμό - δεν υπάρχει νικητής και χαμένος.
- Δίνει τροφή για σκέψη αφού πρέπει ο μαθητής να συντάξει μια πρόταση με μια λέξη κλειδί που να έχει σχέση με το μάθημα που έχουν κάνει.
- Δίνει στον μαθητή τη δυνατότητα να σκεφτεί ελεύθερα για να γράψει μια πρόταση χωρίς να του θέτει όρια στη σκέψη.

### **5.4.2 Σχεδιαστικές αποφάσεις**

Αρχικά, αποφασίσαμε τι εργαλεία θα χρησιμοποιήσουμε για την ανάπτυξη του προγράμματός μας. Εγκαταστήσαμε λοιπόν το λειτουργικό σύστημα Ubuntu 8.04 και χρησιμοποιήσαμε τον text editor Dr Python. Ο Dr Python είναι ένας text editor εύκολα προσαρμόσιμος στις ανάγκες του εκάστοτε προγραμματιστή. Έχει γραφτεί σε Python χρησιμοποιώντας το wxPython toolkit.

Στη συνέχεια, ξεκινήσαμε την εκμάθηση της Python φτιάχνοντας απλά προγράμματα με σκοπό την εξοικείωση με τις βασικές εντολές της γλώσσας. Έπειτα, ξεκινήσαμε τη δημιουργία παραθύρων με την PyGTK κι όταν αποκτήσαμε αρκετή εμπειρία με τη γλώσσα ξεκινήσαμε την ανάπτυξη του λογισμικού.

Όλα τα προγράμματα που αναπτύχθηκαν έχουν από μια κλάση και μια σειρά βοηθητικών συναρτήσεων για να πετύχουμε τη λειτουργικότητα που επιθυμούσαμε. Για την επικοινωνία μεταξύ των διεπαφών εκμεταλλευτήκαμε τη λειτουργία των sockets. Επίσης, χρησιμοποιήσαμε νήματα για δύο εξίσου σημαντικούς λόγους, πρώτον επειδή η εφαρμογή πρέπει να μπορεί να λειτουργεί με πολλούς συνδεδεμένους clients ταυτόχρονα και δεύτερον γιατί χρειαζόμασταν ένα νήμα για να κρατάει τα γραφικά ώστε παράλληλα να βλέπει ο χρήστης τα γραφικά και να γίνονται και οι απαραίτητες λειτουργίες της εφαρμογής. Τα νήματα στην Python είναι πολύ εύχρηστα, ωστόσο, υπήρξε δυσκολία στον συνδυασμό των threads με τα sockets και τα γραφικά. Επιπλέον, στην πορεία της ανάπτυξης της εφαρμογής έγιναν κάποιες τροποποιήσεις στις αρχικές προδιαγραφές ανάλογα με τα εργαλεία και τις δυνατότητες της Python.

Συμπερασματικά, μπορούμε να πούμε ότι η ενασχόληση με την Python ήταν ιδιαίτερα ευχάριστη, τα εργαλεία που υπάρχουν για ανάπτυξη κώδικα είναι εύχρηστα και είναι μια γλώσσα που μπορεί κάποιος να μάθει σε σχετικά μικρό χρονικό διάστημα. Προσφέρει πολλές δυνατότητες στον προγραμματιστή και ένα πολύ βασικό πλεονέκτημα είναι ότι υπάρχουν πάρα πολλές εφαρμογές για το OLPC αλλά και γενικότερα κώδικες σε Python που μπορούν να βοηθήσουν πάρα πολύ έναν προγραμματιστή. Τέλος, η κοινότητα του OLPC είναι πολύ οργανωμένη και προσφέρει βοήθεια σε όσους επιθυμούν να αναπτύξουν κάποια εφαρμογή ή γενικότερα να συμβάλλουν με οποιοδήποτε τρόπο στην ανάπτυξη και βελτίωση του XO.

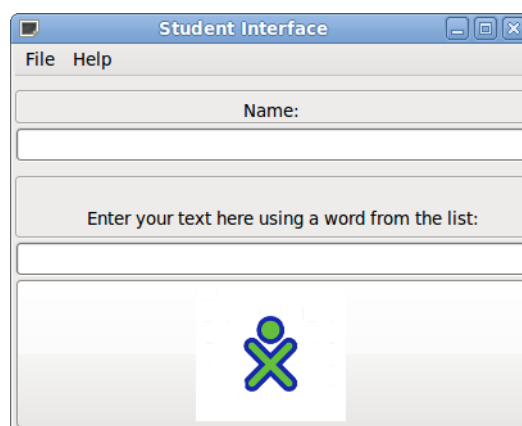
## 6 Τεχνική ανάλυση της εφαρμογής

### 6.1 Γενικά χαρακτηριστικά της εφαρμογής

Η εφαρμογή μας έχει τρεις διεπαφές. Η πρώτη που παρουσιάζουμε είναι η διεπαφή του μαθητή, αυτή δηλαδή στην οποία πάνω δουλεύει ο μαθητής. Η δεύτερη διεπαφή είναι αυτή που είναι στη διάθεση του εκπαιδευτικού, ή του ενίοτε προσώπου που θα επιθυμήσει να εκκινήσει ένα παιχνίδι εισάγοντας κάποιες λέξεις. Τέλος, η τρίτη διεπαφή είναι η κεντρική που λειτουργεί και ως server.

#### 6.1.1 Η διεπαφή του μαθητή

Η διεπαφή του μαθητή στην οποία υπάρχουν δύο πεδία και ένα κουμπί. Στο πρώτο πεδίο ο μαθητής εισάγει το όνομά του και στο δεύτερο πεδίο εισάγει την πρόταση την οποία θέλει να φτιάξει. Όταν ολοκληρώσει την πρότασή του πατάει το κουμπί και στέλνει στη βασική διεπαφή την πρόταση του και κλείνει το παράθυρο.



Σχήμα 6.1: Η διεπαφή του μαθητή

Στην αρχή του προγράμματος εισάγουμε τις βιβλιοθήκες και τα `modules` που είναι απαραίτητα για το πρόγραμμά μας. Οι λειτουργίες τους περιγράφονται αναλυτικά στο πρόγραμμα της βασικής διεπαφής. Στη συνέχεια ορίζουμε τις παραμέτρους για να γίνει η σύνδεση του `client` με τον `server`. Η παράμετρος `host` έχει όρισμα είτε κάποιον `host` είτε κάποια διεύθυνση IP. Επειδή εμείς δοκιμάσαμε το πρόγραμμα μόνο τοπικά επιλέξαμε να χρησιμοποιήσουμε το όρισμα `localhost`. Αν ένα πρόγραμμα τρέχει στο διαδίκτυο πρέπει να χρησιμοποιήσουμε τη διεύθυνση IP ή το `domain name`. Η παράμετρος `port` μπορεί να είναι ένας αριθμός ή ένα συμβολικό όνομα. Η παράμετρος `buff` ορίζει τη μέγιστη ποσότητα δεδομένων που μπορεί να μεταδοθεί κάθε φορά από το `socket`. Μπορεί να είναι οποιαδήποτε δύναμη του 2, ωστόσο πρέπει να λάβουμε υπόψιν τις απαιτήσεις της εφαρμογής μας. Η παράμετρος `addr` έχει δύο ορίσματα, το πρώτο είναι το `host` και το δεύτερο είναι το `port`. Η παράμετρος `addr` χρησιμοποιείται για την αποστολή των δεδομένων στον `server`. Τέλος, αφού έχουμε ορίσει τις παραμέτρους για να γίνει η σύνδεση με τον `server` ανοίγουμε το `socket` και ενημερώνουμε τον `client` ότι το `socket` έχει ανοίξει:

```

1 UDPSock = socket(AF_INET, SOCK_DGRAM) }
2 def_msg = " You are now connected ";
3 print "\n", def_msg

```

Listing 6.1: Initializing a socket

Προγραμματιστικά, η διεπαφή υλοποιείται με 1 κλάση και 5 συναρτήσεις 2 από τις οποίες βρίσκονται εκτός της κλάσης. Η κλάση είναι η `Student_Interface` που κληρονομεί από την `gtk.Window` και οι συναρτήσεις είναι οι εξής:

- `def timer`
- `def show`
- `def enter_callback`
- `def about_word_fight`
- `def __init__`

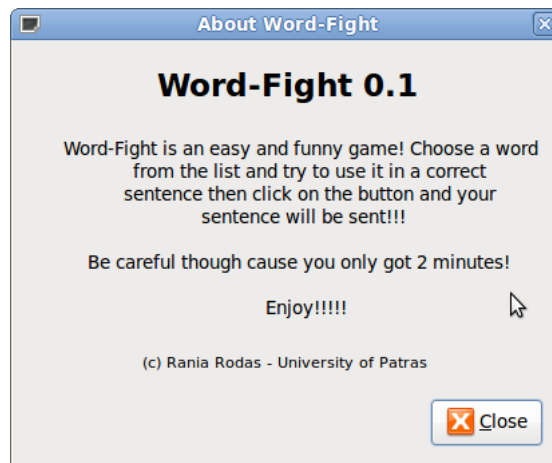
Αναλυτικότερα :

Η συνάρτηση **`def timer`** είναι μια συνάρτηση που κλείνει το παράθυρο του μαθητή και καλείται μετά από 2 λεπτά που το παράθυρο έχει ανοίξει και βρίσκεται έξω από την κλάση. Σκοπός αυτής της συνάρτησης είναι να κλείνει το παράθυρο σε περίπτωση που κάποιος μαθητής δε βρίσκεται στον υπολογιστή του. Το παράθυρο κλείνει με την εντολή `gtk.main_quit` και κλείνει και το `socket` με την εντολή `UDPSock.close()`. Επίσης τυπώνεται το μήνυμα `2 mins elapsed` για να ξέρει ο μαθητής σε περίπτωση που επιστρέψει στον υπολογιστή του, τι έχει συμβεί.

Η συνάρτηση **`def show`** είναι μια συνάρτηση που καλείται όταν ξεκινήσουν να τρέχουν τα γραφικά και μετά από 2 λεπτά καλεί τη συνάρτηση `timer` ώστε να κλείσει το παράθυρο και να κλείσει και το `socket`. Αυτό είναι εφικτό χρησιμοποιώντας την `glib.timeout_add(120000, timer` δηλαδή καλείται η συνάρτηση `timer` όταν έχουν περάσει `120.000ms = 2minutes`

Η συνάρτηση **def enter\_callback** είναι η συνάρτηση που στέλνει τα δεδομένα στον server. Αρχικά διαβάζει από τα entries τα δεδομένα που έχει εισάγει ο μαθητής, δηλαδή το όνομα και την πρότασή του. Στη συνέχεια, στέλνει στον server το string student ώστε να ξέρει ο server ότι λαμβάνει δεδομένα από τον student. Δίνει στο όνομα που έχει εισαχθεί τη μεταβλητή data και στην πρόταση που έχει εισαχθεί τη μεταβλητή data1. Στη συνέχεια, μέσω του socket στέλνει τα δεδομένα στον server και κλείνει το socket καθώς και το παράθυρο. Σε περίπτωση που κάποιο από τα πεδία δεν έχει συμπληρωθεί εκτυπώνει αντίστοιχα no name entered και no sentence entered. Τέλος, κλείνει το παράθυρο και το socket.

Η συνάρτηση **def about\_word\_fight** καλείται όταν πατηθεί στο menu το κουμπί about. Εμφανίζεται ένα παράθυρο που έχει ένα κουμπί close και περιέχει μια σύντομη περιγραφή του παιχνιδιού, πληροφορίες για το παιχνίδι και το όνομα του προγραμματιστή και το Πανεπιστήμιο. Όταν πατηθεί το κουμπί close το παράθυρο καταστρέφεται.



Σχήμα 6.2: Το παράθυρο του μενού στη διεπαφή του μαθητή

Η συνάρτηση **def \_\_init\_\_** είναι η συνάρτηση που δημιουργεί το γραφικό περιβάλλον και ορίζουμε μέσω της `super(Student_Interface, self).__init__()` να καλεί τον γονέα της που είναι η κλάση `Student_Interface` ώστε να μπορούν να καλούνται οι συναρτήσεις που ορίστηκαν και να λειτουργεί ομαλά το πρόγραμμα. Αρχικά, επειδή η συνάρτηση κληρονομεί τα χαρακτηριστικά της κλάσης δημιουργείται το παράθυρο και καθορίζουμε τα χαρακτηριστικά του, προσθέτουμε τον τίτλο `Student_Interface`, καθορίζουμε το μέγεθος του παραθύρου και το τοποθετούμε στο κέντρο. Στη συνέχεια, δημιουργούμε το menu ορίζουμε τα αρχεία και τα υποαρχεία του, τις συναρτήσεις που πρέπει να κληθούν για να πραγματοποιηθούν οι λειτουργίες του και τα κουμπιά που ενεργοποιούν τις συναρτήσεις. Στην PyGTK υπάρχουν διάφοροι τρόποι για να δημιουργήσουμε ένα μενού σε ένα παράθυρο. Επιλέξαμε να δημιουργήσουμε το μενού χειροκίνητα, χωρίς να χρησιμοποιήσουμε κάποια βιβλιοθήκη. Δημιουργούμε την μπάρα του μενού με την εντολή:

```
1 mb = gtk.MenuBar()
```

Listing 6.2: Create menubar

Δημιουργούμε το αντικείμενο του μενού και το υπομενού του με τις εντολές:

```
1 filemenu = gtk.Menu()
2 filem = gtk.MenuItem("File")
3 filem.set_submenu(filemenu)
4 exit = gtk.MenuItem("Exit")
```

Listing 6.3: Create menuitem

Συνδέουμε το αντικείμενο του μενού με τη λειτουργία του, που στη συγκεκριμένη περίπτωση είναι να καταστρέψουμε το παράθυρο. Αυτό μπορεί να γίνει απλά με την εντολή που περιλαμβάνεται στο connect όπως φαίνεται παρακάτω. Στην άλλη περίπτωση που έχει το μενού, το connect καλεί τη συνάρτηση που έχουμε ορίσει. Επιπλέον, προσθέτουμε στο υπομενού το αντικείμενο και στο μενού το υπομενού.

```
1 exit.connect("activate", gtk.main_quit)
2 filemenu.append(exit)
3 mb.append(filem)
```

Listing 6.4: Trigger of submenu item

Έπειτα, ορίζουμε τα δύο entries που χρειαζόμαστε, το ένα για το όνομα του μαθητή και το δεύτερο για την πρόταση που θα εισάγει ο μαθητής και τα συνδέουμε με τη συνάρτηση enter\_callback ώστε να μπορούν μετά να σταλούν τα δεδομένα στον server. Δημιουργούμε τα frames και τα labels που χρειαζόμαστε και προσθέτουμε όλα τα widgets στο vbox για να εμφανιστούν στο παράθυρο. Τα frames και τα labels ορίζονται ως εξής:

```
1 frame = gtk.Frame("")
2 label = gtk.Label("Name:")
3 frame.add(label)
4 vbox.pack_start(frame, False, False, 0)
5 frame.show()
6 label.show()
```

Listing 6.5: frames and labels in PyGTK

Στη συνέχεια, προσθέτουμε ένα κουμπί χρησιμοποιώντας τις βασικές εντολές της PyGTK που ορίζουν το κουμπί με την βασική του λειτουργία στο οποίο προσθέσαμε μια εικόνα ως διεπαφή και το οποίο καλεί τη συνάρτηση enter\_callback για να σταλούν τα δεδομένα. Για να χρησιμοποιήσουμε την εικόνα στο πρόγραμμά μας πρέπει να δηλώσουμε ότι χρησιμοποιούμε εικόνα και πρέπει και να δηλώσουμε και το όνομά της:

```
1 image = gtk.Image()
2 image.set_from_file("human.png")
3 image.show()
4 button = gtk.Button()
5 button.add(image)
6 button.show()
7 hbox.pack_start(button)
```

```
8 button.connect("clicked", self.enter_callback, entry, entry1)
```

Listing 6.6: Create a button with an image

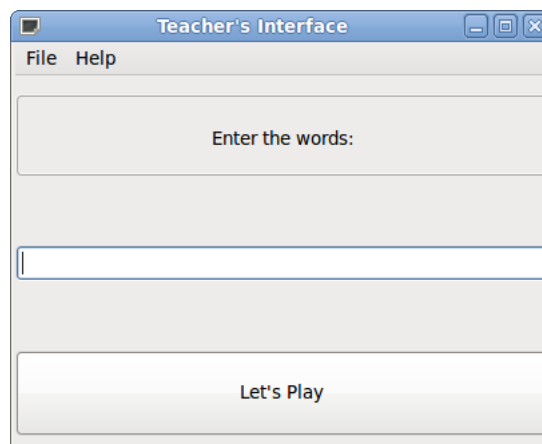
Τέλος καλούμε την κλάση και το `gtk.main` ώστε να τρέχει το πρόγραμμα :

```
1 Student_Interface()
2 gtk.main()
```

Listing 6.7: Running the program

### 6.1.2 Η διεπαφή του εκπαιδευτικού

Στη διεπαφή για τον εκπαιδευτικό εισάγονται οι λέξεις οι οποίες θα συμμετέχουν στο παιχνίδι . Έχει ένα πεδίο εισαγωγής των λέξεων κι ένα κουμπί που στέλνει τα δεδομένα στη βασική διεπαφή και κλείνει το παράθυρο. Στην αρχή του προγράμματος όπως και στο πρόγραμμα της διεπαφής του μαθητή εισάγονται οι βιβλιοθήκες και τα `modules` που είναι απαραίτητα, των οποίων η χρησιμότητα και λειτουργία περιγράφονται παρακάτω. Επιπλέον, ορίζουμε κι εδώ τις παραμέτρους που είναι απαραίτητες για να γίνει η σύνδεση με τον `server` όταν ο εκπαιδευτικός θέλει να στείλει δεδομένα. Δίνουμε τα ίδια ορίσματα για να έχουμε επικοινωνία τοπικά με τον `server` και δίνουμε ίδιο μέγεθος `buffer`. Επίσης, ανοίγουμε κι εδώ το `socket` και εκτυπώνουμε ένα μήνυμα που ενημερώνει τον εκπαιδευτικό ότι το `socket` άνοιξε.



Σχήμα 6.3: Η διεπαφή του εκπαιδευτικού

Προγραμματιστικά, η διεπαφή της δασκάλας υλοποιείται με την κλάση `Teacher_I` η οποία κληρονομεί την `gtk.Window` και 4 συναρτήσεις τις εξής :

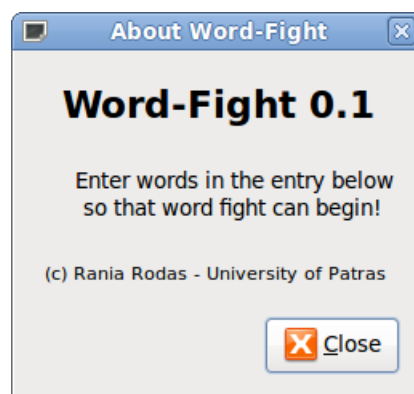
- `def enter_callback`
- `def about_word_fight`
- `def __init__`

- def main

Αναλυτικότερα :

Η συνάρτηση **def enter\_callback** παίρνει τα δεδομένα που έχει γράψει ο εκπαιδευτικός στο entry και στη συνέχεια αν δεν έχουν εισαχθεί δεδομένα τυπώνεται η φράση “ no words entered” αν λάβει δεδομένα τα στέλνει στον server μαζί με τη λέξη teacher για να ξέρει ο server ότι λαμβάνει δεδομένα από τον teacher. Στα δεδομένα που έχουν εισαχθεί αντιστοιχεί τη μεταβλητή data. Το ίδιο όνομα δώσαμε και στη μεταβλητή ονόματος του μαθητή αλλά λόγω του ότι στον server μπαίνουν σε διαφορετικό if αν λάβουν δεδομένα από τη δασκάλα και σε διαφορετικό αν λάβουν δεδομένα από έναν μαθητή, δεν υπάρχει πρόβλημα να χρησιμοποιήσουμε το ίδιο όνομα μεταβλητής. Στη συνέχεια, κλείνει το socket και το παράθυρο.

Η συνάρτηση **def about\_word\_fight** καλείται όταν πατηθεί το κουμπί about από το help του μενού. Περιέχει πληροφορίες για το παιχνίδι και δίνει οδηγίες στον εκπαιδευτικό για το τι πρέπει να κάνει και έχει το όνομα του προγραμματιστή και το Πανεπιστήμιο. Όταν πατηθεί το κουμπί close το παράθυρο καταστρέφεται.



Σχήμα 6.4: Το παράθυρο του μενού στη διεπαφή του εκπαιδευτικού

Η συνάρτηση **def \_\_init\_\_** είναι η συνάρτηση που δημιουργεί τη διεπαφή του εκπαιδευτικού. Όπως και στη διεπαφή του μαθητή ορίζουμε μέσω της super (Teacher\_Interface, self).\_\_init\_\_() να καλείται ο γονέας της συνάρτησης που είναι η κλάση Teacher\_Interface (gtk.Window). Αρχικά, δημιουργούμε το κεντρικό παράθυρο του βάζουμε τον τίτλο Teacher\_Interface, καθορίζουμε το μέγεθος του παραθύρου και το τοποθετούμε στο κέντρο. Στη συνέχεια, δημιουργούμε ένα vbox, δημιουργούμε την μπάρα του μενού και τα αντικείμενα που έχει το μενού το File - Quit και τα Help - About και ορίζουμε τις λειτουργίες τους σε περίπτωση που πατηθούν. Το Quit κλείνει το παράθυρο και το About καλεί τη συνάρτηση about\_word\_fight η οποία ανοίγει το παράθυρο που περιγράψαμε. Βάζουμε το μενού μέσα στο vbox, δημιουργούμε τα labels, frames και entry που χρειαζόμαστε. Ένα frame με label "Enter the words" ένα entry για να εισάγει ο εκπαιδευτικός τις λέξεις και ένα κουμπί για να κληθεί η συνάρτηση enter\_callback και να σταλούν τα δεδομένα. Τέλος προσθέτουμε όλα τα widgets στο vbox και συνδέουμε το κεντρικό παράθυρο με το destroy event με την εξής εντολή :



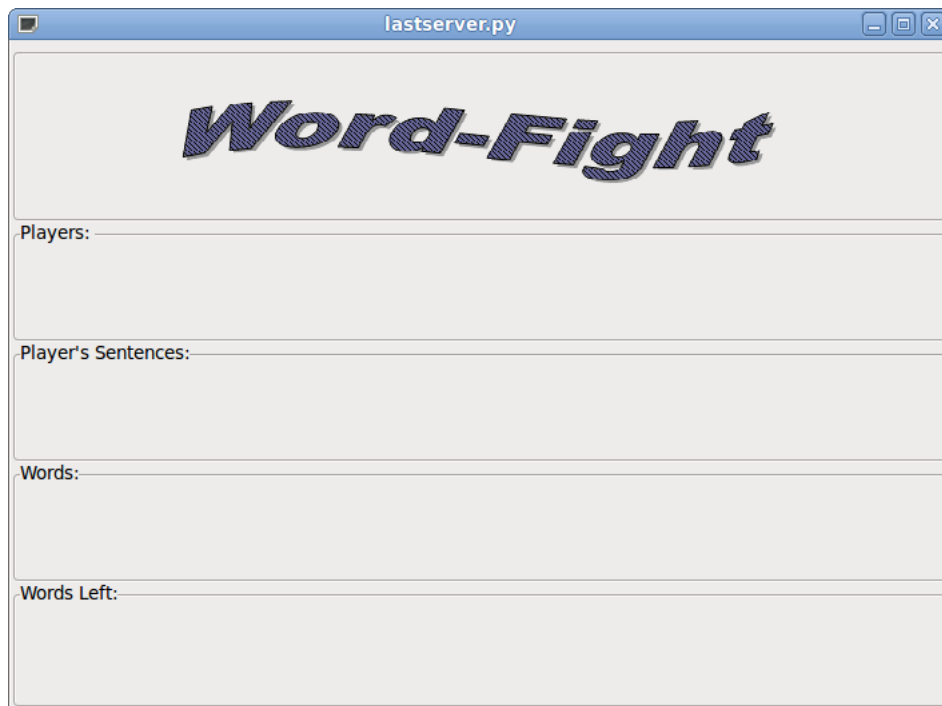
```
1 self.connect("destroy", gtk.main_quit)
```

Listing 6.8: Connect gtk with the destroy event

Σε αυτήν την περίπτωση επιλέξαμε ένα διαφορετικό τρόπο να τρέχει το πρόγραμμα απ' ότι στη διεπαφή του μαθητή χρησιμοποιώντας μια συνάρτηση κι έναν βρόγχο. Η συνάρτηση **def main** καλεί τα γραφικά και τέλος τρέχει ένας βρόγχος που καλεί την κλάση και τη **main**.

### 6.1.3 Η βασική διεπαφή

Η βασική διεπαφή του παιχνιδιού είναι αυτή στην οποία εμφανίζονται οι λέξεις που συμμετέχουν στο παιχνίδι, οι μαθητές που έχουν εισάγει μια πρόταση και η πρότασή τους.



Σχήμα 6.5: Η διεπαφή του παιχνιδιού

Σε αυτό το πρόγραμμα προσθέσαμε άδεια χρήσης GNU. Η GNU έχει σκοπό να εγγυηθεί την ελευθερία να χρησιμοποιούνται από κοινού με άλλους και να τροποποιούνται προγράμματα που διατίθενται ελεύθερα – δηλαδή να εγγυηθεί ότι το πρόγραμμα είναι ελεύθερο για όλους τους χρήστες. Αυτή η Γενική Άδεια Δημόσιας Χρήσης ισχύει για τα περισσότερα προγράμματα του Ιδρύματος Ελεύθερου Λογισμικού (Free Software Foundation), καθώς και για κάθε άλλο πρόγραμμα οι δημιουργοί του οποίου συμμορφώνονται με την άδεια αυτή. Επειδή ο κώδικας αποτελείται από 3 επιμέρους προγράμματα προσθέσαμε το ανάλογο κείμενο της GNU που λέει ότι αυτό το πρόγραμμα είναι μέρος του προγράμματος της εφαρμογής Word-Fight.

Αφού προσθέσαμε την άδεια χρήσης σε μορφή σχολίων, εισάγουμε τα modules που είναι απαραίτητα για το πρόγραμμά μας. Το πρώτο module που εισάγουμε είναι το `threading`. Το σύστημα νημάτων της Python βασίζεται στα υπάρχοντα νήματα του λειτουργικού συστήματος και προσθέτει έναν δικό της διαχειριστή νημάτων που τοποθετείτε ιεραρχικά πάνω από από το σύστημα νημάτων του λειτουργικού συστήματος. Η Python έχει δύο `threading` modules το `thread` και το `threading`. Εμείς χρησιμοποιήσαμε το `threading` γιατί καλύπτει ένα ευρύτερο φάσμα της λειτουργικότητας των νημάτων. Το `import sys` είναι το module που δίνει την δυνατότητα στον μεταφραστή να υποστηρίξει system calls που δεν υποστηρίζει η Python. Τα `import gtk` και `from gtk import gdk` είναι για τη χρήση της PyGTK δηλαδή για τα γραφικά. Το `from socket import *` εισάγει όλα τα modules που έχουν να κάνουν με sockets και δημιουργεί αναφορές στο παρών namespace για όλα τα αντικείμενα που καθορίζονται από αυτό το module. Τα `modules from time import *` και `import time` μας επιτρέπουν να χρησιμοποιήσουμε στο πρόγραμμά μας ό,τι έχει να κάνει με χρόνο. Το module `gobject` μας επιτρέπει να χρησιμοποιήσουμε μια βασική κλάση των gtk γραφικών. Το module `import subprocess` μας επιτρέπει να δημιουργήσουμε καινούριες διεργασίες και να συνδεθούμε με τα δεδομένα εισόδου και εξόδου τους και να πάρουμε τον κώδικα που προκύπτει από αυτές. Η βιβλιοθήκη `rango` είναι μια δωρεάν και ανοιχτού κώδικα υπολογιστική βιβλιοθήκη που προσφέρει διεθνοποιημένα κείμενα σε υψηλή ποιότητα. Η βιβλιοθήκη `glib` μας επιτρέπει να χρησιμοποιήσουμε συναρτήσεις για τα γραφικά. Στη συνέχεια θέτουμε σε λειτουργία τη μηχανή των νημάτων ως εξής:

```
1 gtk.gdk.threads_init()
```

Listing 6.9: Initialize thread engine

Επίσης, καθορίζουμε κι εδώ τις παραμέτρους της σύνδεσης ώστε να μπορεί να γίνει η επικοινωνία μέσω των sockets και αρχικοποιούμε τις λίστες και το `string` που θα χρειαστούμε για το πρόγραμμά μας. Έπειτα, δημιουργούμε το `socket` για να γίνει η σύνδεση αλλά στη συγκεκριμένη διεπαφή πρέπει να κάνουμε και `bind to address`:

```
1 UDPSock = socket(AF_INET, SOCK_DGRAM)
2 UDPSock.bind(addr)
```

Listing 6.10: Creating a socket and bind to address

Προγραμματιστικά, η βασική διεπαφή που λειτουργεί και ως server έχει μια κλάση και 6 συναρτήσεις τις εξής:

- `def run`
- `def stop`
- `def main_quit`
- `def game_over`
- `def remove_element`
- `def show`

- `def timer_show`

Η κλάση `Game_Interface` έχει ως όρισμα το `threading.thread` πράγμα που σημαίνει ότι αυτή η κλάση είναι υποκλάση της κλάσης `threading.thread`. Το `threading module` περιέχει μια κλάση `Thread`, της οποίας κάθε στιγμιότυπο αντιπροσωπεύει ένα νήμα. Κάθε τυπική εφαρμογή δημιουργεί υποκλάσεις αυτής της κλάσης για δύο βασικούς λόγους. Ο πρώτος είναι ότι πιθανότατα υπάρχουν κάποιες συγκεκριμένες μεταβλητές ή μέθοδοι στην εφαρμογή που κατασκευάζουμε και ο δεύτερος είναι ότι η κλάση `Thread` έχει ως μέλος τη μέθοδο `run` που έχει προορισμό να αγνοηθεί. Πριν τη μέθοδο `run` σταματάμε το νήμα αν υπάρχει κάποιο να τρέχει:

```
1 stopthread = threading.Event()
```

Listing 6.11: Stopping the thread

Αναλυτικότερα:

Η συνάρτηση **`def run`** αρχικά ανοίγει το νήμα και τρέχει ένα `while` ώστε να λαμβάνει δεδομένα από τους `clients`. Στη συνέχεια υπάρχουν 3 `if`. Στο πρώτο `if` μπαίνει σε περίπτωση που δεν λάβει δεδομένα από τους `clients`. Στο δεύτερο μπαίνει αν λάβει δεδομένα από τον εκπαιδευτικό και στο τρίτο μπαίνει αν λάβει δεδομένα από κάποιον μαθητή. Αν λάβει δεδομένα από τον `teacher` βάζει σε δύο λίστες τις λέξεις που έχει δώσει ο εκπαιδευτικός. Η λίστα `words` είναι αυτή που θα μείνει άθικτη ώστε να βλέπουν όλοι οι παίκτες τις αρχικές λέξεις που έχουν δοθεί από τον εκπαιδευτικό και η λίστα `words_left` είναι αυτή που θα τροποποιείται κάθε φορά που κάποιος μαθητής θα εισάγει μια πρόταση. Θέσαμε τις λίστες αυτές να είναι `global` μεταβλητές για να είναι ορατές σε όλο το πρόγραμμα και να μπορούν να τις βλέπουν τα γραφικά ώστε να παίρνουν τα ορίσματα τα `labels` και να ανανεώνονται. Αν λάβει δεδομένα από τον `student` αποθηκεύει το όνομά του στη λίστα `student_list` και κάθε επόμενο όνομα που λαμβάνει το κάνει `append` στη λίστα αυτή. Επίσης, αποθηκεύει την πρότασή του στη λίστα `sentence_list` και κάθε επόμενη πρόταση την κάνει `append` στη λίστα αυτή. Σε περίπτωση που κάποιος παίκτης συμμετέχει δεύτερη φορά στο παιχνίδι το όνομά του δεν γίνεται `append` στη λίστα με τα ονόματα αλλά φαίνεται μόνο δίπλα στην πρόταση. Χρησιμοποιήσαμε μια βοηθητική μεταβλητή, την `found`. Στην αρχή η τιμή της είναι 0. Ελέγχουμε αν το όνομα που προστέθηκε στη λίστα `student_list` υπάρχει ήδη. Αν δεν υπάρχει το προσθέτουμε στη λίστα και συνεχίζεται η εκτέλεση του προγράμματος. Αν υπάρχει ήδη στη λίστα η μεταβλητή `found` παίρνει την τιμή 1 και το όνομα δεν προστίθεται στη λίστα `student_list`, αλλά μόνο η πρότασή του προστίθεται στη λίστα `sentence_list` στην οποία βέβαια φαίνεται και το όνομά του για να γνωρίζουν οι παίκτες ποιος έχει εισάγει την κάθε πρόταση.

```
1 found = 0
2 lastuser = data.split()[0]
3 for i in student_list:
4     if lastuser == i:
5         found = 1
6 if found == 0:
7     student_list.append(data)
```

Listing 6.12: Check if name already exists in `student_list`

Στη συνέχεια, καλείτε η συνάρτηση `remove_element` ενώ αν αδειάσει η λίστα `word_left` καλείται η συνάρτηση `game_over`. Τέλος, κλείνει το `mutex` και κάνει `sleep` για 100 ms:

```
1 gtk.gdk.threads_leave()
2 time.sleep(0.1)
```

Listing 6.13: Leave thread and sleep

Η συνάρτηση **`def stop`** σταματάει τον βρόγχο του `thread`:

```
1 def stop(self):
2     self.stopthread.set()
```

Listing 6.14: Stop the thread

Η συνάρτηση **`def main_quit`** σταματάει το `thread` και τον βρόγχο των γραφικών:

```
1 def main_quit(obj):
2     global fs
3     fs.stop()
4     gtk.main_quit()
```

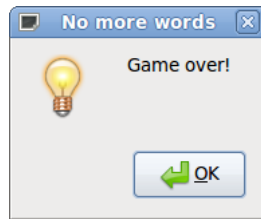
Listing 6.15: Stop all processes

Η συνάρτηση **`def game_over`** καλείται όταν η λίστα με τις λέξεις που δεν έχουν χρησιμοποιηθεί είναι άδεια και δημιουργεί ένα μικρό παράθυρο που ενημερώνει τους παίκτες ότι το παιχνίδι έχει τελειώσει. Έχει ένα κουμπί που γράφει `ok` κι όταν πατηθεί κλείνει το παράθυρο ώστε να μπορούν στη συνέχεια οι μαθητές με τον εκπαιδευτικό να έχουν μπροστά τους τη βασική διεπαφή και να συζητήσουν τα αποτελέσματα.

```
1 def game_over():
2     dialog = gtk.MessageDialog(
3     parent          = None,
4     flags           = gtk.DIALOG_DESTROY_WITH_PARENT,
5     type           = gtk.MESSAGE_INFO,
6     buttons        = gtk.BUTTONS_OK,
7     message_format = "Game over!")
8     dialog.set_title('No more words')
9     dialog.show()
10    dialog.connect('response', lambda dialog, response: dialog.destroy())
```

Listing 6.16: Game over dialog

Η συνάρτηση **`def remove_element`** παίρνει ως είσοδο δύο λίστες και αφαιρεί από τη μια λίστα όσα `string` είναι ίδια. Συγκεκριμένα, παίρνει ως είσοδο τις λέξεις που έχει εισάγει η δασκάλα και την πρόταση που έχει εισάγει κάποιος μαθητής και σβήνει από την λίστα των λέξεων αυτή που ο μαθητής έχει χρησιμοποιήσει. Ο βασικός βρόγχος της είναι ο εξής:



Σχήμα 6.6: Το παράθυρο που αναδύεται όταν τελειώνουν οι λέξεις

```

1 for i in s_list:
2     for j in w_list:
3         if i == j:
4             w_list.remove(j)
5             print i
6 print "new word list: "
7 return w_list

```

Listing 6.17: For loop of remove element

Η συνάρτηση **def show** καλείτε όταν αρχίσουν να τρέχουν τα γραφικά και κάθε 1ms καλεί τη συνάρτηση `timer_show`.

```

1 def show(widget, data=None):
2     glib.timeout_add(1000, timer_show)

```

Listing 6.18: Renew labels

Η συνάρτηση **timer\_show** καλείται από την `def show` κάθε 1ms και ανανεώνει τα πεδία των labels του server όταν κάποιος παίκτης έχει εισάγει δεδομένα και καλεί τον εαυτό της κάθε 1ms ώστε να ανανεώνονται τα πεδία συνεχώς. Επειδή τα labels παίρνουν ορίσματα της μορφής string ενώ εμείς δουλεύουμε με λίστες, χρησιμοποιούμε την παρακάτω εντολή για να μετατρέψουμε τη λίστα σε string:

```

1 b = '\n'.join(student_list)

```

Listing 6.19: Turn the list into a string

Στη συνέχεια ορίζονται τα γραφικά. Δημιουργούμε το κυρίως παράθυρο, ορίζουμε το μέγεθός του, το συνδέουμε με την συνάρτηση `show`. Δημιουργούμε ένα `vbox` και στη συνέχεια δημιουργούμε 5 frames και 4 labels. Το πρώτο περιέχει μια εικόνα που φτιάξαμε με το όνομα `Word-Fight`, γι αυτό και δεν χρειαστήκαμε label, το δεύτερο `Players` και περιέχει τα ονόματα των παικτών που έχουν συμμετάσχει στο παιχνίδι, το τρίτο έχει όνομα `Player's Sentences` και περιέχει τις προτάσεις που έχουν γράψει οι μαθητές, το τέταρτο έχει όνομα `Words` και περιέχει τις λέξεις που έχει εισάγει η δασκάλα και το πέμπτο έχει όνομα `Words Left` και περιλαμβάνει τις λέξεις που δεν έχουν χρησιμοποιηθεί ακόμα. Βάζουμε όλα τα frames μέσα στο `vbox`, βάζουμε το `vbox` μέσα στο παράθυρο και συνδέουμε το παράθυρο με το `delete event`. Τέλος, ξεκινάμε τη λειτουργία των νημάτων

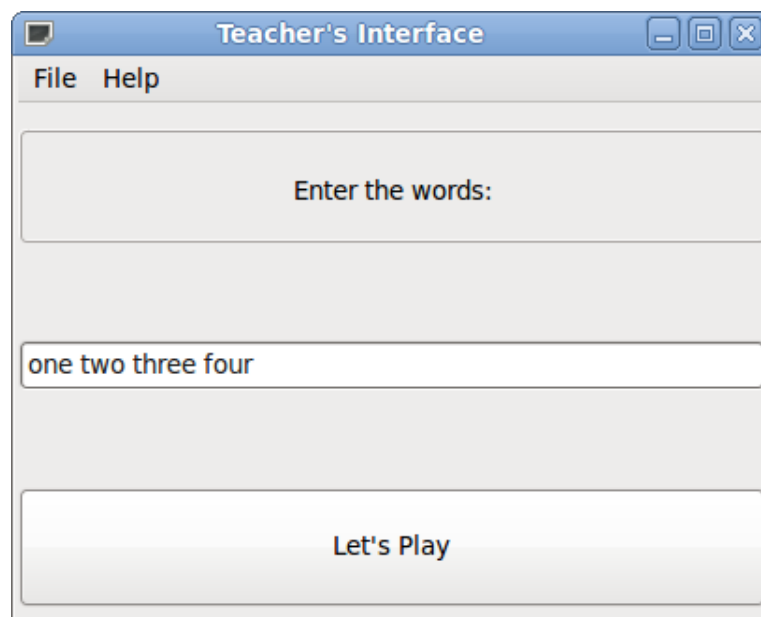
και των γραφικών:

```
1 fs = Game_Interface()  
2 fs.start()  
3 gtk.main()
```

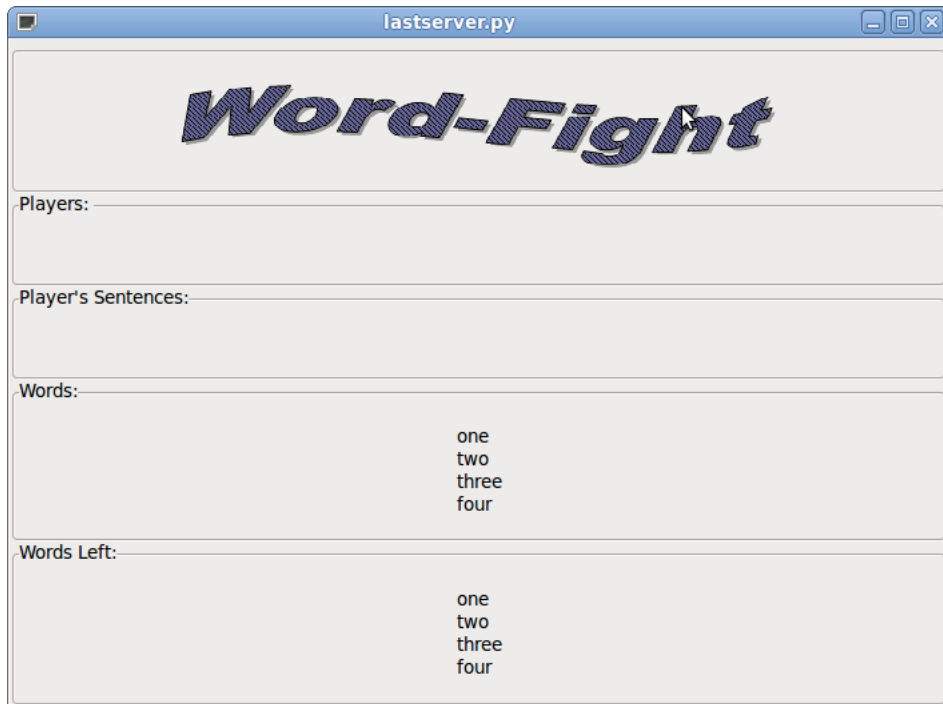
Listing 6.20: Initiate threads and gtk

## 6.2 Πειραματικές Διαδικασίες

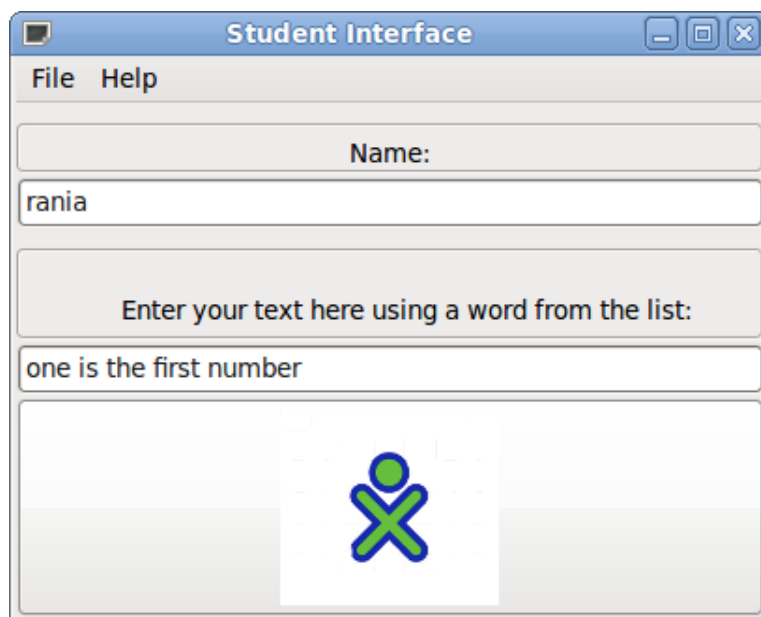
Δοκιμάσαμε την εφαρμογή μας για να διαπιστώσουμε αν δουλεύει με πολλούς μαθητές και πολλές λέξεις. Δουλέψαμε σε τοπικό επίπεδο και παρουσιάζουμε παρακάτω τις εικόνες με τα αποτελέσματα:



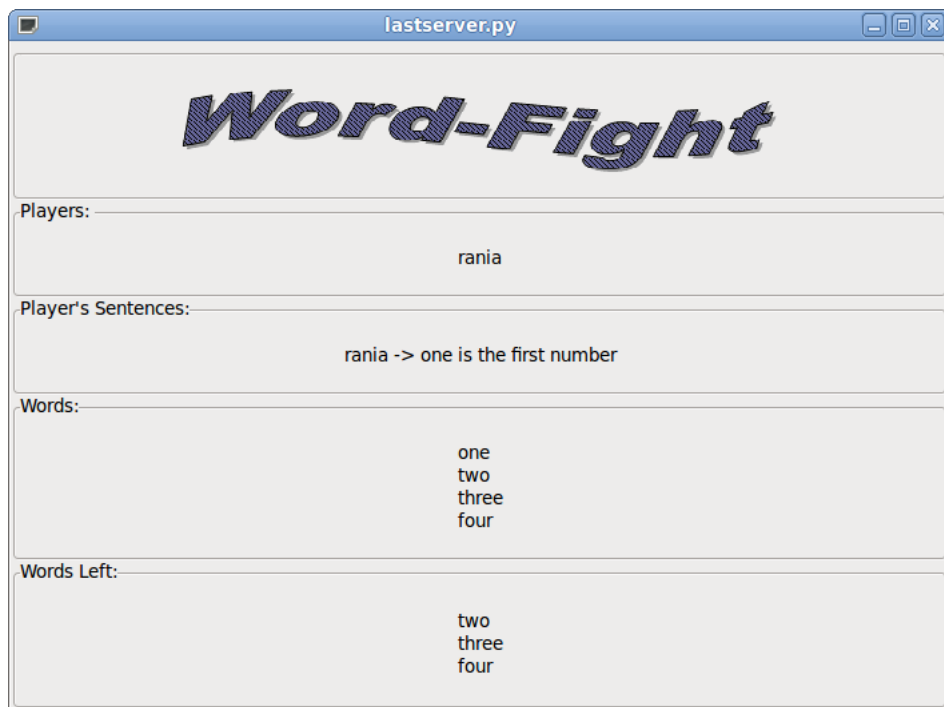
Σχήμα 6.7: Εισαγωγή των λέξεων από τον εκπαιδευτικό



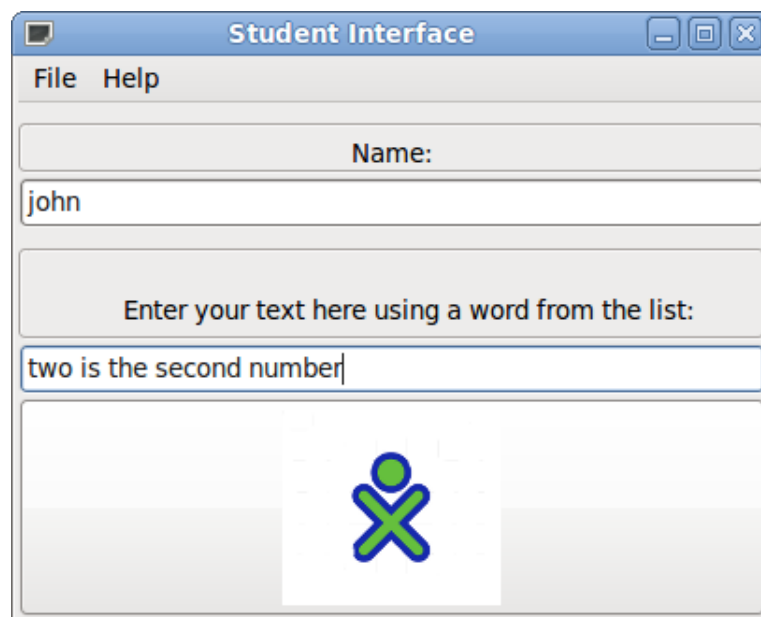
Σχήμα 6.8: Η κεντρική διεπαφή μετά από την εισαγωγή των λέξεων



Σχήμα 6.9: Εισαγωγή ονόματος και πρότασης από μαθητή 1

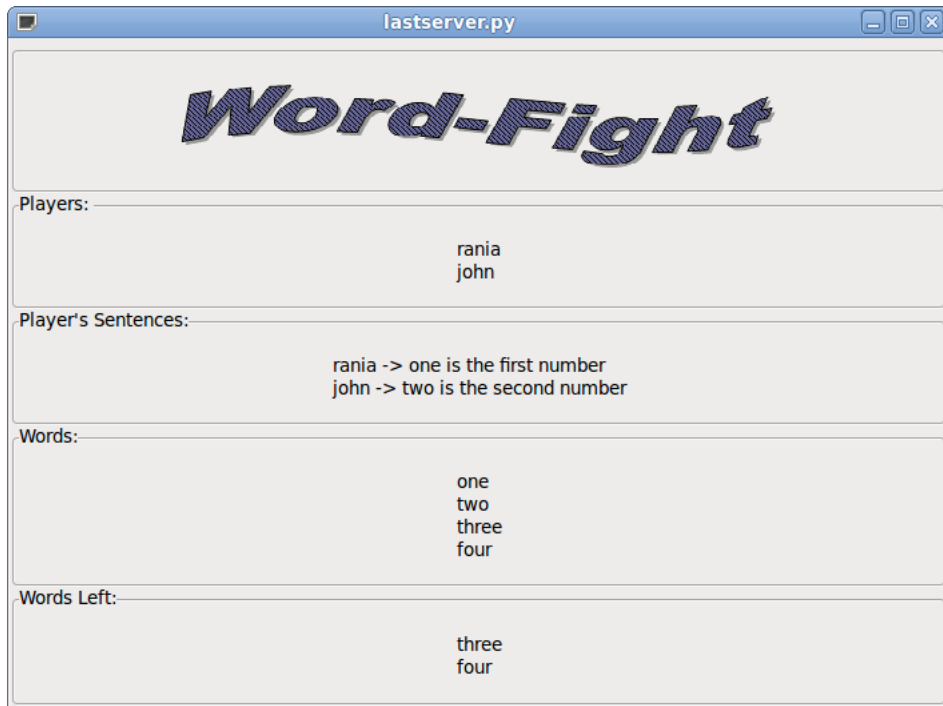


Σχήμα 6.10: Η κεντρική διεπαφή μετά την εισαγωγή του πρώτου μαθητή

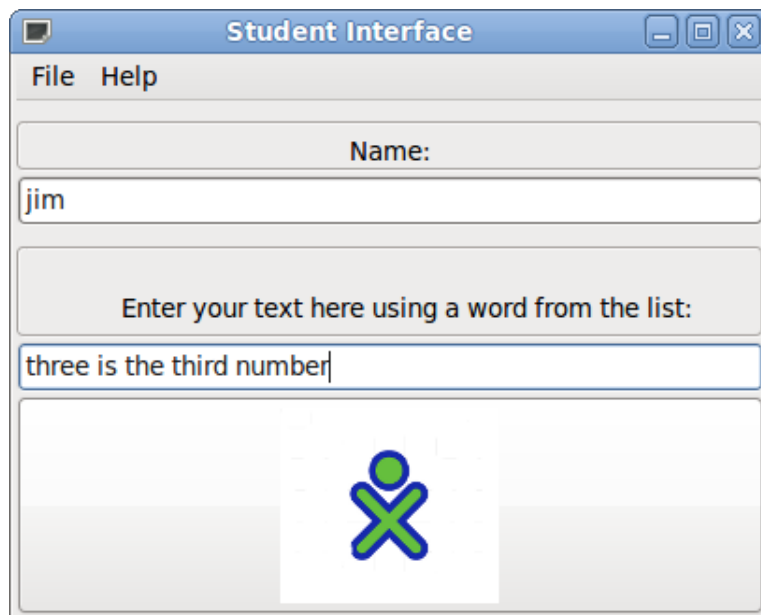


Σχήμα 6.11: Εισαγωγή ονόματος και πρότασης από μαθητή 2

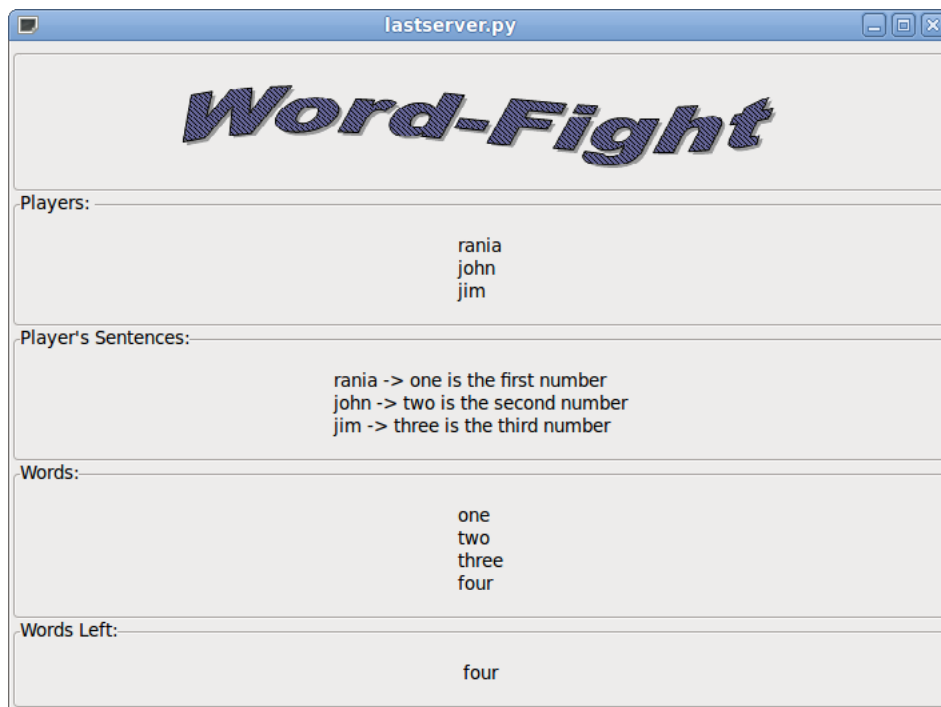




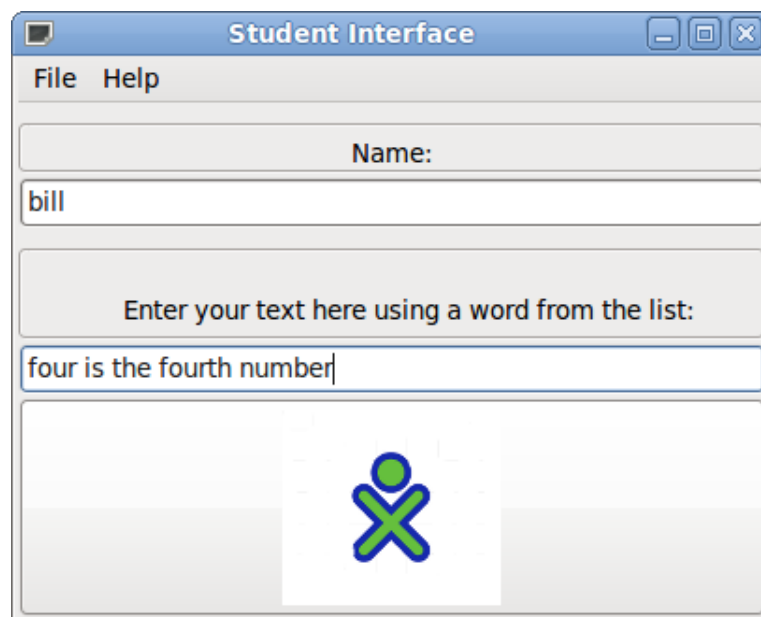
Σχήμα 6.12: Η κεντρική διεπαφή μετά την εισαγωγή του δεύτερου μαθητή



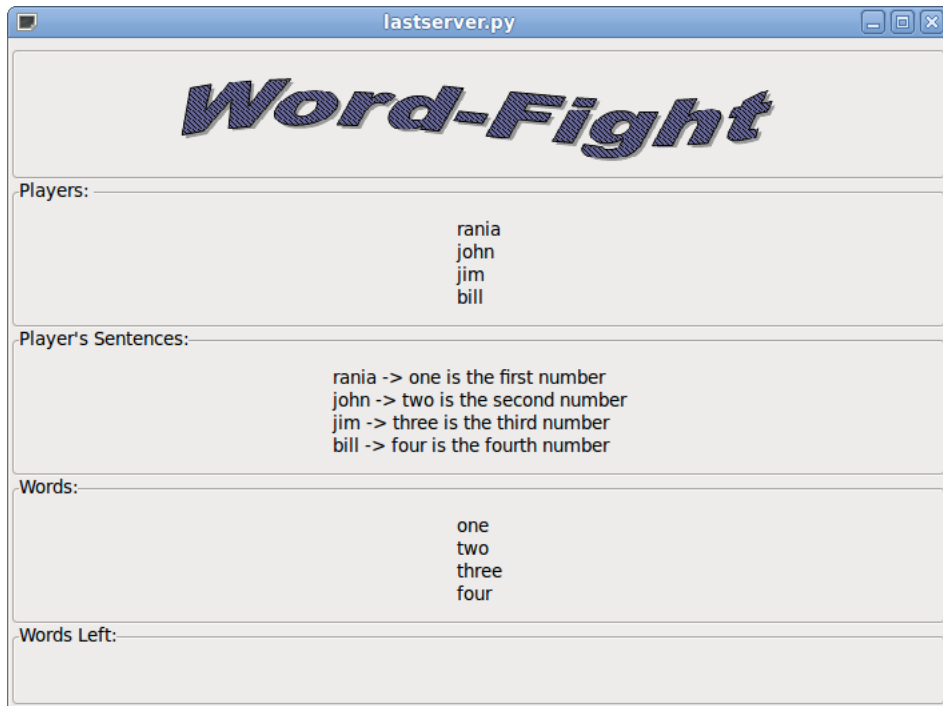
Σχήμα 6.13: Εισαγωγή ονόματος και πρότασης από μαθητή 3



Σχήμα 6.14: Η κεντρική διεπαφή μετά την εισαγωγή του τρίτου μαθητή



Σχήμα 6.15: Εισαγωγή ονόματος και πρότασης από μαθητή 4



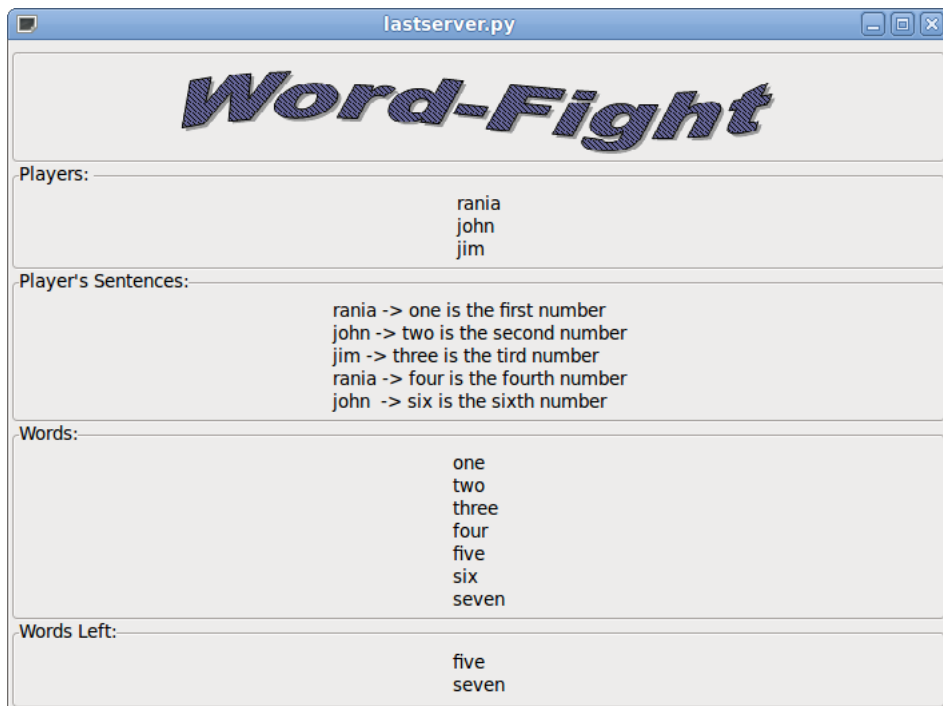
Σχήμα 6.16: Η κεντρική διεπαφή μετά την εισαγωγή του τέταρτου μαθητή

Κατά τη διάρκεια της εκτέλεσης του προγράμματος εμφανίζονται τα παρακάτω:

```
1 /usr/bin/python -u "/home/raninaki/Desktop/lastlast/lastserver.py"
2
3
4 Teacher connected and sent: ['one', 'two', 'three', 'four']
5
6 Student ' rania ' connected.
7 Current students: ['rania']
8 rania sent: one is the first number
9
10
11 ['one', 'two', 'three', 'four']
12 ['two', 'three', 'four']
13
14
15 Student ' john ' connected.
16 Current students: ['rania', 'john']
17 john sent: two is the second number
18
19
20 ['one', 'two', 'three', 'four']
21 ['three', 'four']
22
23
24 Student ' jim ' connected.
25 Current students: ['rania', 'john', 'jim']
26 jim sent: three is the third number
27
28
29 ['one', 'two', 'three', 'four']
30 ['four']
31
32
33 Student ' bill ' connected.
34 Current students: ['rania', 'john', 'jim', 'bill']
35 bill sent: four is the fourth number
36
37
38 ['one', 'two', 'three', 'four']
39 []
40
```

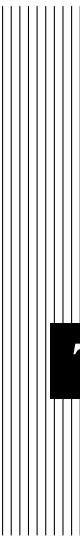
Listing 6.21: Python prompt

Σε περίπτωση που οι λέξεις είναι περισσότερες απ' ό τι οι συμμετέχοντες και οι παίκτες εισάγουν κάποια πρόταση περισσότερες από μια φορές το όνομά τους δε θα προστίθεται πάλι στη λίστα με τα ονόματα.



Σχήμα 6.17: Η κεντρική διεπαφή όταν ίδιοι παίκτες εισάγουν 2 προτάσεις





## 7 Επίλογος

### 7.1 Αποτίμηση

Συμπερασματικά, μπορούμε να πούμε ότι προσπαθήσαμε να φτιάξουμε μια εφαρμογή ευχάριστη και χρήσιμη για τα παιδιά. Η Python είναι μια γλώσσα προγραμματισμού που παρουσιάζει μεγάλο ενδιαφέρον και προσφέρει πάρα πολλές δυνατότητες στον προγραμματιστή. Μπορέσαμε να κινηθούμε με ευκολία στα γραφικά μέσω της PyGTK δοκιμάζοντας πολλούς διαφορετικούς τρόπους προσέγγισης της διεπαφής. Αξιοσημείωτο είναι ότι καταφέραμε σε λίγες γραμμές κώδικα να αναπτύξουμε ένα εκπαιδευτικό παιχνίδι το οποίο για να αναπτυχθεί σε κάποια άλλη γλώσσα θα χρειαζόνταν πάρα πολλές γραμμές κώδικα.

### 7.2 Μελλοντική Εργασία

Ως μελλοντική εργασία θα μπορούσαμε να αναπτύξουμε την εφαρμογή αυτή με τις απαραίτητες τροποποιήσεις ώστε να μπορεί να χρησιμοποιείται και σε *single-mode*.

Επιπλέον θα μπορούσαμε να βελτιώσουμε τη μορφή των διεπαφών ώστε να είναι βελτιωμένες αισθητικά. Θα μπορούσαμε να προσθέσουμε τη δυνατότητα για χρήση σε περισσότερα μαθήματα. Αν π.χ. η δασκάλα έδινε κάποιες πράξεις και ο μαθητής έγραφε την πράξη που επιλέγει να κάνει και την απάντηση που θέλει να δώσει στο πεδίο εισαγωγής κειμένου.

### 7.3 Επίλογος

Ολοκληρώνοντας αυτή την εργασία διαπιστώνουμε ότι είναι πολύ σημαντικό τα παιδιά να μάθουν από μικρή ηλικία να χειρίζονται τους υπολογιστές. Σε αυτό το πλαίσιο κινείται και η πρωτοβουλία OLPC που πρωταρχικό σκοπό έχει να δώσει σε όλα τα παιδιά του κόσμου την ευκαιρία να γνωρίσουν νέες τεχνολογίες και να έχουν ίσες ευκαιρίες. Είναι πολύ

σημαντικό λοιπόν να υποστηρίζουν οι πανεπιστημιακές κοινότητες τέτοιες πρωτοβουλίες με όποιον τρόπο μπορούν.

Σε αυτό το σημείο αξίζει να αναφέρουμε ότι όσο απαραίτητη κι αν είναι η εξοικείωση των παιδιών με τους υπολογιστές και την επιστήμη της Πληροφορικής, δεν πρέπει να ξεχνάμε πως πάνω απ' όλα είναι παιδιά και είναι υψίστης σημασίας να διατηρούμε στην ζωή τους το μέτρο. Παρ' όλο που επικρατεί η άποψη ότι ο υπολογιστής είναι πολύ χρήσιμο εργαλείο για την κοινωνικοποίηση των παιδιών, τίποτα δεν μπορεί να αντικαταστήσει την ανθρώπινη επαφή. Χρέος λοιπόν όλων μας είναι να διασφαλίσουμε την ορθολογική χρήση των υπολογιστών από τα παιδιά.





## Α' Παράρτημα Α

### **Εγκατάσταση του XO σε οποιοδήποτε σύστημα**

Το γεγονός ότι το XO λειτουργεί εξ' ολοκλήρου με λογισμικό και προγράμματα ανοιχτού λογισμικού μας δίνει τη δυνατότητα να μπορούμε να εγκαταστήσουμε έναν εξομοιωτή σε οποιοδήποτε σύστημα. Αυτό μπορεί να γίνει εύκολα με 2 απλά βήματα.

- Βήμα 1

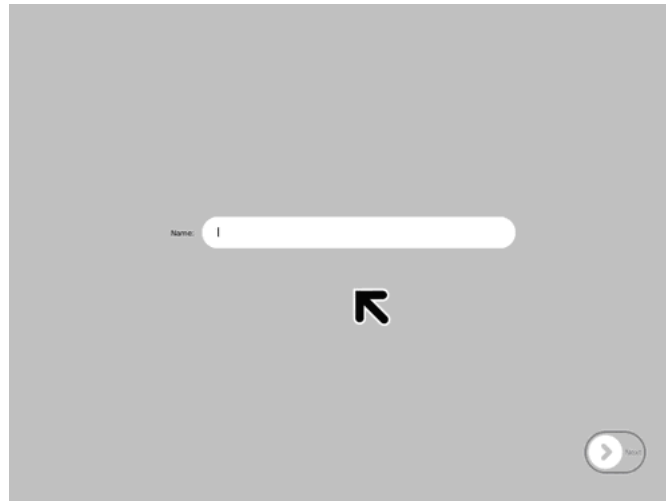
Πρέπει να κατεβάσουμε και να εγκαταστήσουμε έναν εξομοιωτή λειτουργικών συστημάτων. Χαρακτηριστικά αναφέρουμε τον εξομοιωτή QEMU, VMWare Player ή και το VirtualBox που μπορούμε να τους κατεβάσουμε δωρεάν από το διαδίκτυο, από τις ιστοσελίδες:

- <http://www.nongnu.org/qemu/download.html>
- <http://downloads.vmware.com/d/>
- <http://www.virtualbox.org/wiki/Downloads> αντίστοιχα.

Χρησιμοποιώντας το VMware Player: Κατεβάζουμε την έτοιμη εικόνα του vmware player για το OLPC, ανοίγουμε το πρόγραμμα και κατόπιν ανοίγουμε την εικόνα. Καλώς ήλθατε στο περιβάλλον εργασίας του OLPC.

- Βήμα 2

Εφόσον έχουμε προμηθευτεί με κάποιον εξομοιωτή πρέπει να κατεβάσουμε το κατάλληλο image του XO που είναι συμβατό με τον εξομοιωτή που κατεβάσαμε. Τα images του XO μπορούμε να τα βρούμε στην επίσημη σελίδα του OLPC καθώς και στη σελίδα wiki του OLPC. Πρέπει να διαθέτουμε χώρο τουλάχιστον 1Gb και να ορίσουμε τη RAM στα 256 Mb. Όταν φορτώσουμε το image στον εξομοιωτή μας θα εμφανιστεί η αρχική οθόνη του Sugar και το XO image θα είναι έτοιμο για χρήση.



Σχήμα Α.1: Login Sugar Interface



## Β' Παράρτημα Β

### Β'.1 Ο κώδικας της διεπαφής του μαθητή

Listing B.1: student.py

```
1 #!/usr/bin/env python
2 # import the libraries
3 # necessary for the program
4
5 from socket import *
6 import pygtk
7 pygtk.require('2.0')
8 import gtk
9 import time
10 import threading
11 import glib
12 from time import *
13
14 # set connection parameters
15 host = "localhost"
16 port = 21567
17 buf = 8192
18 addr = (host,port)
19
20 # open socket
21 UDPSock = socket(AF_INET,SOCK_DGRAM)
22
23 # inform the student that the socket opened
24 def_msg = "You are now connected";
25 print "\n",def_msg
26
27 # function timer closes the socket
28 # and destroys the gtk.window
29 def timer():
30
```

```
31     print "You run out of time"
32     gtk.main_quit()
33     UDPSock.close()
34     return False
35
36 # create the class Student_Interface
37 class Student_Interface(gtk.Window):
38
39     # function show calls function
40     # timer when 2 minutes have elapsed
41     def show(self, widget, data=None):
42
43         glib.timeout_add(120000, timer)
44
45     # function enter_callback gets the
46     # text input from the student and
47     # stores it in the variable entry_text the
48     # student's name and in the variable entry1_text
49     # the student's sentence
50     def enter_callback(self, widget, entry, entry1):
51
52         entry_text = entry.get_text()
53         entry1_text = entry1.get_text()
54
55         print "Student's sentence: %s\n" % entry1_text
56
57         # send the data, with the student's acknowledgment "s"
58         # for the server to know the data is being sent from
59         # a student
60         s = 'student'
61         data = entry_text
62         data1 = entry1_text
63
64         if not data:
65             print 'no name entered'
66
67         if not data1:
68             print 'no sentence entered'
69
70         else:
71             if(UDPSock.sendto(s,addr)):
72                 print "student sending..."
73
74             if(UDPSock.sendto(data,addr)):
75                 print "name sent: '",data,"'"
76
77             if(UDPSock.sendto(data1,addr)):
78                 print ('{0} wrote: \t {1}'.format(data,data1))
79
80         # close socket
81         UDPSock.close()
82
83         # close window after sending data
84         gtk.main_quit()
85
```

```
86 # function about_word_fight pops up
87 # a window that gives some info
88 # about the game. Also includes
89 # the instructions
90 def about_word_fight(self, About):
91
92     about = gtk.AboutDialog()
93     about.set_program_name("Word-Fight")
94     about.set_version("0.1")
95     about.set_copyright("(c) Rania Rodas - University of Patras ")
96     about.set_comments("""
97     Word-Fight is an easy and funny game! Choose a word
98     from the list and try to use it in a correct
99     sentence then click on the button and your
100     sentence will be sent!!!
101
102     Be careful though cause you only got 2 minutes!
103
104     Enjoy!!!!
105     """)
106     about.run()
107     about.destroy()
108
109 # function creating the student's interface
110 def __init__(self):
111
112     super(Student_Interface, self).__init__()
113
114     # name the main window
115     self.set_title("Student Interface")
116
117     # set the size of the main window
118     self.set_size_request(400, 300)
119
120     # place the window in the center
121     self.set_position(gtk.WIN_POS_CENTER)
122
123     # connect the window with function show
124     self.connect("show", self.show)
125
126     # create the menubar
127     mb = gtk.MenuBar()
128
129     # create the menu
130     filemenu = gtk.Menu()
131
132     # name the menu item
133     filem = gtk.MenuItem("File")
134
135     # set the submenu
136     filem.set_submenu(filemenu)
137
138     # create a submenu item
139     exit = gtk.MenuItem("Exit")
140
```

```
141     # set the trigger of submenu item
142     # exit that is to close the gtk
143     exit.connect("activate", gtk.main_quit)
144
145     # add submenu item exit to menu item file
146     filemenu.append(exit)
147     filemenu1 = gtk.Menu()
148
149     # name the second menu item
150     filem1 = gtk.MenuItem("Help")
151
152     # create a submenu for the second menu item
153     filem1.set_submenu(filemenu1)
154
155     # create a submenu item
156     about = gtk.MenuItem("About")
157
158     # set the trigger of submenu item about
159     # that is to call function about_word_fight
160     about.connect("activate", self.about_word_fight)
161
162     # add submenu item about to menu item help
163     filemenu1.append(about)
164
165     # add both menu items
166     # to the menu
167     mb.append(filem)
168     mb.append(filem1)
169
170     # create a vertical box
171     vbox = gtk.VBox(False, 2)
172
173     # add the menubar to the vertical box
174     vbox.pack_start(mb, False, False, 0)
175
176     # create 2 gtk entries
177     entry = gtk.Entry()
178     entry1 = gtk.Entry()
179
180     # set max length of the
181     # first entry to be 50 characters
182     # considered enough for a student
183     # to write his name
184     entry.set_max_length(50)
185
186     # set entry to connect with function enter_callback
187     entry.connect("activate", self.enter_callback, entry, entry1)
188
189     # set max length of the second entry
190     # to be 200 characters considered enough
191     # for a student to write a sentence
192     entry1.set_max_length(200)
193
194     # set the second entry to connect with
195     # function enter_callback
```

```
196     entry1.connect("activate", self.enter_callback, entry, entry1)
197     entry.show()
198     entry1.show()
199
200     vbox.show()
201
202     # create a frame
203     frame = gtk.Frame("")
204
205     # create a label
206     label = gtk.Label("Name:")
207
208     # add label in the vbox
209     frame.add(label)
210
211     # add the frame in the vbox
212     vbox.pack_start(frame, False, False, 0)
213     frame.show()
214     label.show()
215
216     # add entry in the vbox
217     vbox.pack_start(entry, False, False, 0)
218
219     # create a second frame
220     frame1 = gtk.Frame("")
221
222     # create a second label
223     labell = gtk.Label("""
224     Enter your text here using a word from the list:
225     """)
226     #adding the label in the frame
227     frame1.add(labell)
228     frame1.show()
229     labell.show()
230
231     # add frame1 in the vbox
232     vbox.pack_start(frame1, True, True, 0)
233
234     # add entry1 in the vbox
235     vbox.pack_start(entry1, False, False, 0)
236
237     # create a horizontal box
238     hbox = gtk.HBox(False, 3)
239
240     # add the horizontal box in the vbox
241     vbox.add(hbox)
242     hbox.show()
243
244     # set an image for the button
245     image = gtk.Image()
246     image.set_from_file("human.png")
247     image.show()
248
249     # create a button to contain the image widget
250     button = gtk.Button()
```

## 104 Παράρτημα Β

---

```
251         button.add(image)
252         button.show()
253         hbox.pack_start(button)
254
255         # connect the button with the function enter_callback
256         button.connect("clicked", self.enter_callback, entry, entry1)
257
258         # add the vbox to the main window
259         self.add(vbox)
260
261         # connect the main window with the destroy event
262         self.connect("destroy", gtk.main_quit)
263         self.show_all()
264
265 #start the GTK+ event
266 Student_Interface()
267 gtk.main()
```

---



## B.2 Ο κώδικας της διεπαφής της δασκάλας

Listing B.2: teacher.py

```
1  #!/usr/bin/env python
2  # import the libraries
3  # necessary for the programme
4  from socket import *
5  import pygtk
6  pygtk.require('2.0')
7  import gtk
8
9  #set the connection parameters
10 host = "localhost"
11 port = 21567
12 buf = 8192
13 addr = (host,port)
14
15 #open socket
16 UDPSock = socket(AF_INET,SOCK_DGRAM)
17
18 #inform the teacher that the socket opened
19 def_msg = " You are now connected ";
20 print "\n",def_msg
21
22 #create the class Teacher_Interface
23 class Teacher_Interface(gtk.Window):
24
25     #function enter_callback gets the
26     #text input from the teacher
27     def enter_callback(self, widget, entry):
28         entry_text = entry.get_text()
29         print "Entry contents: %s\n" % entry_text
30         words=entry_text
31         data = entry_text
32         #send the data through the socket
33         if not data:
34             print 'no words entered'
35         else:
36             if(UDPSock.sendto('teacher',addr)):
37                 print 'send t'
38             if(UDPSock.sendto(data,addr)):
39                 print "message sent: '",data,'" "
40         #close socket
41         UDPSock.close()
42
43         #close main window
44         gtk.main_quit()
45
46     #function about_word_fight pops up
47     #a window that gives some info
48     #about the game and gives
49     #the instructions
50     def about_word_fight(self, About):
51         about = gtk.AboutDialog()
```

```
52         about.set_program_name("Word-Fight")
53         about.set_version("0.1")
54         about.set_copyright("(c) Rania Rodas - University of Patras ")
55         about.set_comments("""
56         Enter words in the entry below
57         so that word fight can begin!
58         """)
59         about.run()
60         about.destroy()
61
62
63     def __init__(self):
64         super(Teacher_Interface, self).__init__()
65
66         # create a new window
67         self.set_title("Teacher's Interface")
68         self.set_size_request(400, 300)
69         self.set_position(gtk.WIN_POS_CENTER)
70
71         #create a vbox and add it to the window
72         vbox = gtk.VBox(False, 0)
73         self.add(vbox)
74         vbox.show()
75
76         #set a title to the window
77         self.set_title("Simple menu")
78         #set window size
79         self.set_size_request(400, 300)
80         #set window position
81         self.set_position(gtk.WIN_POS_CENTER)
82
83         #create a menubar
84         mb = gtk.MenuBar()
85
86         #create the menu
87         filemenu = gtk.Menu()
88         #name the menu item
89         filem = gtk.MenuItem("File")
90         #set the submenu
91         filem.set_submenu(filemenu)
92
93         #create a submenu item
94         exit = gtk.MenuItem("Exit")
95         #set the trigger of submenu item
96         #exit that is to close the gtk
97         exit.connect("activate", gtk.main_quit)
98         #add submenu item exit to menu item file
99         filemenu.append(exit)
100
101         filemenu1 = gtk.Menu()
102         #name the second menu item
103         filem1 = gtk.MenuItem("Help")
104         #create a submenu for the second menu item
105         filem1.set_submenu(filemenu1)
106         #create a submenu item
```

```
107     about = gtk.MenuItem("About")
108     #set the trigger of submenu item about
109     #that is to call function about_word_fight
110     about.connect("activate", self.about_word_fight)
111     #add submenu item about to menu item help
112     filemenu1.append(about)
113
114     #add both menu items
115     #to the menu
116     mb.append(filem)
117     mb.append(filem1)
118
119     #add the menubar in the vbox
120     vbox.pack_start(mb, False, False, 0)
121
122     #create a frame
123     frame = gtk.Frame("")
124     frame.set_shadow_type(gtk.SHADOW_ETCHED_OUT)
125     #create a label
126     label = gtk.Label("Enter the words:")
127     #add the label in the frame
128     frame.add(label)
129     #add the frame in the vbox
130     vbox.pack_start(frame, True, True, 5)
131     frame.show()
132     label.show()
133
134     #creat a text entry
135     entry = gtk.Entry(max=0)
136     #set entry to connect with function enter_callback
137     entry.connect("activate", self.enter_callback, entry)
138     #select the region of the entry according to the text
139     entry.select_region(0, len(entry.get_text()))
140
141     #add the entry in a vbox
142     vbox.pack_start(entry, True, True, 30)
143     entry.show()
144
145     #creat a button and put it in a vbox
146     button=gtk.Button("Let's Play")
147     button.connect("clicked", self.enter_callback, entry)
148     vbox.pack_start(button, True, True, 5)
149     button.show()
150
151     #connect the main window to the destroy event
152     self.connect("destroy", gtk.main_quit)
153     self.show_all()
154
155 #function def starts th gtk
156 def main():
157     gtk.main()
158     return 0
159
160 #if loop stars the class and
161 #call the main function
```

## 108 Παράρτημα Β

---

```
162 if __name__ == "__main__":  
163     Teacher_Interface()  
164     main()
```

---

## B.3 Ο κώδικας της κεντρικής διεπαφής

Listing B.3: game-server.py

```
1 # ----- #
2 # #
3 # This file is part of Word-Fight. #
4 # #
5 # Word-Fight is free software: you can redistribute it and/or modify #
6 # it under the terms of the GNU General Public License as published #
7 # by the Free Software Foundation, either version 3 of the License, or #
8 # (at your option) any later version. #
9 # #
10 # Word-Fight is distributed in the hope that it will be useful, #
11 # but WITHOUT ANY WARRANTY; without even the implied warranty of #
12 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the #
13 # GNU General Public License for more details. #
14 # #
15 # You should have received a copy of the GNU General Public #License #
16 # along with Word-Fight. If not, see <http://www.gnu.org/licenses/>. #
17 # #
18 # ----- #
19
20 # import the libraries
21 # necessary for the program
22 import threading
23 import sys
24 import gtk
25 from socket import *
26 from gtk import gdk
27 from time import *
28 import GObject
29 import time
30 import subprocess
31 import os
32 import pango
33 import glib
34
35 # initializing the gtk's thread engine
36 gtk.gdk.threads_init()
37
38 # set the socket parameters
39 host = "localhost"
40 port = 21567
41 buf = 8192
42 addr = (host, port)
43
44 # initialize the lists we need
45
46 # student_list is a list containing
47 # the names of the students
48 student_list = []
49
50 # words_left is a list containing the words
51 # that haven't been used yet
```

```
52 words_left = []
53
54 # sentence_list is a variable containing
55 # the sentences that have been entered
56 sentence_list = []
57
58 # words is a list containing the words that
59 # have been entered by the teacher initially
60 words = []
61
62 # create socket and bind to address
63 UDPSock = socket(AF_INET,SOCK_DGRAM)
64 UDPSock.bind(addr)
65
66 # create threading class game_interface
67 class Game_Interface(threading.Thread):
68
69     # thread event, stops the thread if it is set.
70     stopthread = threading.Event()
71
72     # function run
73     def run(self):
74
75         while not self.stopthread.isSet():
76
77             # acquiring the gtk global mutex
78             gtk.gdk.threads_enter()
79
80             # setting a random value for the fraction
81             while 1:
82
83                 data,addr = UDPSock.recvfrom(buf)
84
85                 if not data:
86                     print "Client has exited!"
87                     break
88
89                 else:
90                     # if data is sent from the teacher
91                     # the words entered are appended
92                     # in the list words
93                     if data == 'teacher':
94
95                         data,addr = UDPSock.recvfrom(buf)
96                         print "\nTeacher connected and sent: ", data.split()
97                             # we set variables words_left and words
98                             # to be global so that they are visible
99                             # to the rest of the program
100                     global words_left
101                     words_left = data.split()
102                     global words
103                     words =data.split()
104
105                     # if data is sent from a student
106                     # the name sent is appended
```

```
107     # in the list student_list
108     # and the sentence is appended
109     # in the sentence_list
110     if data == 'student':
111
112         data,addr = UDPSock.recvfrom(buf)
113         data1,addr = UDPSock.recvfrom(buf)
114         print "\nStudent '", format(data.split()[0]), "' connected."
115
116         # we set a variable named found to be 0
117         # which represents if the current student
118         # already exists in the student list
119         found = 0
120         lastuser = data.split()[0]
121
122         # we check the student list for the
123         # current student and if he already exists
124         # we set the variable found to be 1
125         for i in student_list:
126             if lastuser == i:
127                 found = 1
128
129         # if he is not found we add him to the student list
130         # otherwise we continue
131         if found == 0:
132             student_list.append(data)
133
134         print "Current students: " , student_list
135
136         # we store the name and sentence of the student
137         # to the existing list of participations
138         sentence_list.append(data + " -> " + data1)
139
140         print data.split()[0] + " sent: " + data1 + "\n"
141
142         # function remove element is triggered
143         remove_element(data1.split(),words_left)
144
145         # we print both lists to take a look
146         # at the initial word list and the words
147         # that have not been used yet
148         print "                "
149         print words
150         print words_left
151         print "                "
152
153         if not words_left:
154
155             # call the function game_over
156             # that pops up a window telling
157             # that the game is over
158             game_over()
159
160     # releasing the gtk global mutex
161     gtk.gdk.threads_leave()
```

## 112 Παράρτημα Β

---

```
162
163         # delaying 100ms until the next iteration
164         time.sleep(0.1)
165
166     # function stop, stops the tread
167     def stop(self):
168
169         self.stopthread.set()
170
171 # main_quit function, it stops the thread and the gtk's main loop
172 def main_quit(obj):
173
174     # importing the fs object from the global scope
175     global fs
176
177     # stopping the thread and the gtk's main loop
178     fs.stop()
179
180     # close the gtk window
181     gtk.main_quit()
182
183 # function game over create a dialog window
184 # with an ok button informing that the game
185 # is over
186 def game_over():
187
188     dialog = gtk.MessageDialog(
189     parent          = None,
190     flags           = gtk.DIALOG_DESTROY_WITH_PARENT,
191     type            = gtk.MESSAGE_INFO,
192     buttons         = gtk.BUTTONS_OK,
193     message_format = "Game over!")
194     dialog.set_title('No more words')
195     dialog.show()
196     dialog.connect('response', lambda dialog, response: dialog.destroy())
197
198 # function to remove the element that is
199 # common in both student and sentence lists
200 def remove_element(s_list, w_list):
201
202     for i in s_list:
203         for j in w_list:
204             if i == j:
205                 w_list.remove(j)
206     return w_list
207
208 # function show calls function
209 # timer_show every 1ms
210 def show(widget, data=None):
211
212     glib.timeout_add(1000, timer_show)
213
214 # function timer_show renews the
215 # text in the labels according to the
216 # new entries and calls itself every 1msec
```



```
217 def timer_show():
218
219     # convert lists into strings so that they
220     # can be entered as arguments into labels
221     a = '\n'.join(student_list)
222     label1.set_text(a)
223
224     b = '\n'.join(sentence_list)
225     label2.set_text(b)
226
227     c = '\n'.join(words)
228     label3.set_text(c)
229
230     d = '\n'.join(words_left)
231     label4.set_text(d)
232
233     glib.timeout_add(1000,timer_show)
234
235 # create a gtk window
236 window = gtk.Window()
237
238 # set size of the gtk window
239 window.set_size_request(700,500)
240
241 # connect the window with function show
242 window.connect("show", show)
243
244 # create a vertical box
245 vbox = gtk.VBox(False, 0)
246
247 # create a frame
248 frame = gtk.Frame('')
249
250 # set an image for the title logo
251 image1 = gtk.Image()
252 image1.set_from_file("background.png")
253 image1.show()
254
255 # add label in the frame
256 frame.add(image1)
257
258 # create a second frame
259 frame1 = gtk.Frame("Players: ")
260
261 # create a second label
262 label1 = gtk.Label("")
263
264 # add label in the frame
265 frame1.add(label1)
266
267 frame2 = gtk.Frame("Player's Sentences:")
268 label2 = gtk.Label()
269 frame2.add(label2)
270
271 frame3 = gtk.Frame("Words:")
```

## 114 Παράρτημα Β

---

```
272 label3 = gtk.Label("")
273 frame3.add(label3)
274
275 frame4 = gtk.Frame("Words Left:")
276 label4 = gtk.Label()
277 frame4.add(label4)
278
279 # add frames in the vertical box
280 vbox.pack_start(frame, True, True, 0)
281 vbox.pack_start(frame1, True, True, 0)
282 vbox.pack_start(frame2, True, True, 0)
283 vbox.pack_start(frame3, True, True, 0)
284 vbox.pack_start(frame4, True, True, 0)
285
286 # add the vertical box in the window
287 window.add(vbox)
288 window.show_all()
289
290 # connecting the destroy event to the main_quit function
291 window.connect('destroy', main_quit)
292
293 # creating and starting the thread
294 fs = Game_Interface()
295 fs.start()
296
297 # start the GTK+ event
298 gtk.main()
```

---



## Βιβλιογραφία

- [1] Summerfield, Mark: *Programming in Python 3: A Complete Introduction to the Python Language*. Pearson education, Inc, 2009.
- [2] Martelli, Alex and David Ascher: *Python Cookbook*. O'Reilly, 1986.
- [3] Preiss, Bruno R.: *Data Structures and Algorithms with Object-Oriented Design Patterns in Python*, 2003.
- [4] Stevens, W. Richard: *UNIX Network Programming: Networking APIs: Sockets and XTI*. Prentice Hall PTR, 1998. Volume 1.
- [5] Walton, Sean: *Linux Socket Programming*. Sams, 2001. 1st Edition.
- [6] Downey, Allen B.: *Think Python - How to Think Like a Computer Scientist*. Green Tea Press, 2002.
- [7] Matloff, Norman and Francis Hsu: *Tutorial on threads programming with python*, 2005.
- [8] Finlay, John: *Pygtk 2.0 tutorial*, April 13, 2005.
- [9] <http://wiki.laptop.org/go/Sugar>.
- [10] <http://wiki.laptop.org/go/Category:Sugar>.
- [11] [http://wiki.laptop.org/go/Sugar\\_Activity\\_Tutorial](http://wiki.laptop.org/go/Sugar_Activity_Tutorial).
- [12] [http://en.wikipedia.org/wiki/Wireless\\_mesh\\_network](http://en.wikipedia.org/wiki/Wireless_mesh_network).
- [13] [http://wiki.laptop.org/go/Mesh\\_Network\\_Details](http://wiki.laptop.org/go/Mesh_Network_Details).
- [14] <http://wiki.laptop.org/go/Chat>.

- [15] [http://wiki.laptop.org/go/Joke\\_Machine](http://wiki.laptop.org/go/Joke_Machine).
- [16] [http://wiki.laptop.org/go/Creating\\_an\\_Activity](http://wiki.laptop.org/go/Creating_an_Activity).
- [17] <http://www.linuxjournal.com/article/3946>.
- [18] <http://www.python.org/>.
- [19] <http://wiki.python.org/moin/BeginnersGuide>.
- [20] <http://python.fyxm.net/doc/Newbies.html>.
- [21] <http://en.wikipedia.org/wiki/Learning>.
- [22] <http://en.wikipedia.org/wiki/Vygotsky>.
- [23] [http://en.wikipedia.org/wiki/Play\\_%28activity%29](http://en.wikipedia.org/wiki/Play_%28activity%29).
- [24] <http://www.tovima.gr/default.asp?pid=2&ct=75&artid=102879&dt=13/09/1998>.
- [25] <http://www.geocities.com/notesgym/>.
- [26] <http://www.doukas.gr/tp/tp110.htm>.
- [27] [http://en.wikipedia.org/wiki/OLPC\\_XO-1](http://en.wikipedia.org/wiki/OLPC_XO-1).
- [28] <http://www-cs-faculty.stanford.edu/~eroberts/cs201/projects/one-laptop-per-child/history/olpc-history.html>.
- [29] [http://wiki.laptop.org/images/7/71/CL1A\\_Hdwe\\_Design\\_Spec.pdf](http://wiki.laptop.org/images/7/71/CL1A_Hdwe_Design_Spec.pdf).
- [30] [http://wiki.laptop.org/go/Power\\_Management](http://wiki.laptop.org/go/Power_Management).
- [31] [http://en.wikipedia.org/wiki/Mesh\\_networking](http://en.wikipedia.org/wiki/Mesh_networking).
- [32] [http://wiki.laptop.org/go/Mesh\\_Network\\_Details](http://wiki.laptop.org/go/Mesh_Network_Details).
- [33] [http://wiki.laptop.org/go/Software\\_components](http://wiki.laptop.org/go/Software_components).
- [34] [http://wiki.laptop.org/go/OLPC\\_Python\\_Environment](http://wiki.laptop.org/go/OLPC_Python_Environment).
- [35] [http://wiki.laptop.org/go/Programming\\_languages](http://wiki.laptop.org/go/Programming_languages).
- [36] <http://www.pegi.info/gr/index/id/228/>.
- [37] <http://en.wikipedia.org/wiki/ENIAC>.
- [38] [http://en.wikipedia.org/wiki/Personal\\_computer](http://en.wikipedia.org/wiki/Personal_computer).
- [39] <http://en.wikipedia.org/wiki/Laptop>.
- [40] [http://wiki.laptop.org/go/BTest-1\\_Release\\_Notes](http://wiki.laptop.org/go/BTest-1_Release_Notes).

- 
- [41] [http://wiki.laptop.org/go/BTest-2\\_Release\\_Notes](http://wiki.laptop.org/go/BTest-2_Release_Notes).
- [42] [http://wiki.laptop.org/go/BTest-3\\_Release\\_Notes](http://wiki.laptop.org/go/BTest-3_Release_Notes).
- [43] [http://wiki.laptop.org/go/BTest-4\\_Release\\_Notes](http://wiki.laptop.org/go/BTest-4_Release_Notes).
- [44] [http://wiki.laptop.org/go/OLPC\\_on\\_open\\_source\\_software](http://wiki.laptop.org/go/OLPC_on_open_source_software).
- [45] [http://en.wikipedia.org/wiki/Johnny\\_Appleseed](http://en.wikipedia.org/wiki/Johnny_Appleseed).
- [46] <http://el.wikipedia.org/wiki/%CE%A0%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CE%BA%CE%AE>.
- [47] [http://en.wikipedia.org/wiki/Sugar\\_%28desktop\\_environment%29](http://en.wikipedia.org/wiki/Sugar_%28desktop_environment%29).
- [48] <http://blog.laptopmag.com/first-look-olpc-xo-generation-20>.
- [49] [http://en.wikipedia.org/wiki/Python\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/Python_%28programming_language%29).
- [50] [http://python.about.com/od/gettingstarted/ss/whatispython\\_2.htm](http://python.about.com/od/gettingstarted/ss/whatispython_2.htm).
- [51] [http://en.wikipedia.org/wiki/Wireless\\_mesh\\_network](http://en.wikipedia.org/wiki/Wireless_mesh_network).
- [52] <http://wowio.files.wordpress.com/2008/04/xo-sunlight-big.jpg>.
- [53] [http://news.cnet.com/i/bto/20071228/OLPC\\_unboxing\\_03.jpg](http://news.cnet.com/i/bto/20071228/OLPC_unboxing_03.jpg).
- [54] <http://wiki.laptop.org/images/2/22/Battery-2.jpg>.
- [55] <http://www.itmweb.com/bimages/olpc.jpg>.
- [56] [http://farm3.static.flickr.com/2193/1959133423\\_e6e3f1bd09.jpg](http://farm3.static.flickr.com/2193/1959133423_e6e3f1bd09.jpg).
- [57] [http://www.express.gr/news/ellada/139435oz\\_20090304139435.php3](http://www.express.gr/news/ellada/139435oz_20090304139435.php3).
- [58] [http://en.wikipedia.org/wiki/OLPC\\_XO-1](http://en.wikipedia.org/wiki/OLPC_XO-1).
- [59] <http://www.gnu.org/copyleft/gpl.html>.
- [60] <http://www.thinkpython.com>.