

Ατομική Διπλωματική Εργασία

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟΥ  
ΝΕΦΟΥΣ ΓΙΑ ΔΙΚΤΥΑ ΚΙΝΗΤΩΝ ΣΥΣΚΕΥΩΝ**

Αντώνης Μαυρής

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ**



**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Μάιος 2011**

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Σχεδιασμός και Υλοποίηση Προγραμματιστικού Νέφους για Δίκτυα Κινητών  
Συσκευών**

**Αντώνης Μαυρής**

Επιβλέπων Καθηγητής

Δημήτρης Ζεϊναλιπούρ

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων  
απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου  
Κύπρου

Μάιος 2011

## Ευχαριστίες

Θα ήθελα αρχικά να ευχαριστήσω τον κύριο Δημήτρη Ζεϊναλιπούρ, για την ευκαιρία που μου έδωσε να δουλέψω στην συγκεκριμένη διπλωματική εργασία και για την πλήρη κατανόηση και υποστήριξη που είχα από αυτόν, κατά τη εκπόνησή της. Πρόκειται για ένα σπουδαίο καθηγητή και ερευνητή με πολλές γνώσεις σε όλους του τομείς της επιστήμης της Πληροφορικής, αλλά πάνω απ' όλα για ένα εξαιρετο άνθρωπο.

Θα ήθελα επίσης να ευχαριστήσω το Τμήμα της Πληροφορικής του Πανεπιστημίου Κύπρου για την αγορά και διάθεση όλων των συσκευών που ήταν απαραίτητα για την ολοκλήρωση της παρούσας διπλωματικής εργασίας.

Τέλος οφείλω να πω ένα μεγάλο ευχαριστώ στους φίλους μου εντός και εκτός Πανεπιστημίου οι οποίοι μου συμπαραστάθηκαν όλο αυτό τον καιρό αλλά και ένα τεράστιο ευχαριστώ στην οικογένειά μου και ιδιαίτερα στους γονείς μου Πρόδρομο και Ειρήνη για την ηθική και οικονομική υποστήριξη που μου πρόσφεραν όχι μόνο κατά τη διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας αλλά και καθ' όλη τη διάρκεια των σπουδών μου.

## Περίληψη

Η ανάπτυξη, η διαχείριση και ο έλεγχος εφαρμογών σε έξυπνες συσκευές (*Smart phones*) αποτελούν χρονοβόρες, δαπανηρές και δύσκολες διαδικασίες, αφού ο χρήστης χρειάζεται να έχει τις φυσικές συσκευές παρών ενώ επίσης απαιτούν χειρονακτική διαχείριση ξεχωριστά.

Στόχος της παρούσας διπλωματικής εργασίας είναι η προτυποποίηση ενός συστήματος το οποίο να μπορεί να χρησιμοποιηθεί για την ανάπτυξη και αξιολόγηση αλγορίθμων, πρωτοκόλλων, συστημάτων και εφαρμογών που αφορούν έξυπνες συσκευές (συγκεκριμένα **Android[1,2] based** κινητές συσκευές). Συγκεκριμένα, για την επίλυση των πιο πάνω δυσκολιών, σχεδιάστηκε και αναπτύχθηκε το “*SmartNet*”, ένα προγραμματιστικό νέφος το οποίο παρέχει δυνατότητα διαχείρισης, ανάπτυξης και έλεγχου εφαρμογών μέσω διαδικτύου, εύκολα, χωρίς έξοδα και χρονοβόρες διαδικασίες.

Το *SmartNet* αποτελείται από έναν εξυπηρετητή (*Apache2*) πάνω στον οποίο βρίσκονται ενωμένες μέσω *USB Android* κινητές συσκευές (*HTC Desire[3]/ HTC Hero[4]* με λειτουργικό σύστημα *Android OS 2.1* ), ο οποίος επιτρέπει την διαχείριση των εν λόγω κινητών συσκευών μέσω μιας Διαδικτυακής διαπροσωπείας. Έτσι παρέχεται η δυνατότητα για ανάπτυξη και έλεγχο εφαρμογών σε έξυπνες συσκευές σε πραγματικές συνθήκες, σε άτομα που ενδιαφέρονται χωρίς να έχουν τις πραγματικές συσκευές. Επίσης παρέχεται η δυνατότητα διαχείρισης πολλαπλών συσκευών αυτοματοποιώντας έτσι χρονοβόρες διαδικασίες για ανάπτυξη και έλεγχο.

Αρχικά έγινε εξοικείωση με γλώσσες προγραμματισμού όπως *HTML[5]*, *PHP[6]*, *Android Debug Bridge (ADB) [7]* οι οποίες χρησιμοποιήθηκαν για την δημιουργία και την πρόταση της αρχικής αρχιτεκτονικής και στη συνέχεια έγινε επισκόπηση της υφιστάμενης τεχνολογίας που υπάρχει αυτή την στιγμή αξιολογώντας παρόμοιες πλατφόρμες που έχουν ήδη υλοποιηθεί. Συγκεκριμένα μελετήθηκαν ως προς την αρχιτεκτονική, την υλοποίηση και την λειτουργικότητά τους συστήματα όπως το *MoteLab [8]* και το *Wireless Sensor*

*Network Testbed (WISEBED) [9]. Τα δύο αυτά συστήματα έχουν να κάνουν με Ασύρματα Δίκτυα Αισθητήρων και θα αναπτυχθούν πιο κάτω. Επίσης μελετήθηκε το **Remote Device Access [10]** από την *Nokia* και το **Amazon Elastic Compute Cloud (Amazon EC2) [11]** από την *Amazon*.*

Στη συνέχεια αποφασίσαμε και σχεδιάσαμε η αρχιτεκτονική του *SmartNet* έχοντας πάντα υπόψη τα συστήματα που μελετήθηκαν περνώντας έτσι στην υλοποίηση του συστήματος.

Τέλος θέσαμε το σύστημα υπό πειραματική λειτουργία, με σκοπό να δούμε την ορθότητα της λειτουργίας του αλλά και των ευκολιών που παρέχει. Συγκεκριμένα μετρήθηκε ο χρόνος που χρειάζεται για την εκτέλεση μιας σειρά ενεργειών σε πολλαπλές συσκευές τόσο χειρονακτικά όσο και με την χρήση του *SmartNet* βλέποντας έτσι τον χρόνο τον οποίο εξοικονομούμε στην περίπτωση χρήσης του. Σύμφωνα με τα πειράματα που εκτελέσαμε μπορούμε να ολοκληρώσουμε ένα απλό σενάριο δεκατέσσερις (14) φορές πιο γρήγορα χρησιμοποιώντας το *SmartNet*.

## Περιεχόμενα

Ευχαριστίες.....	i
Περίληψη.....	ii
<b>Κεφάλαιο 1 Εισαγωγή.....</b>	<b>1</b>
1.1 Το Βασικό Υπόβαθρο.....	1
1.2 Υποκίνηση Εργασίας.....	6
1.3 Περίγραμμα εργασίας.....	7
<b>Κεφάλαιο 2 Σχετική Δουλεία και Τεχνολογικό Υπόβαθρο.....</b>	<b>9</b>
2.1 MoteLab .....	10
2.2 WISEBED.....	12
2.3 Remote Device Access.....	14
2.4 Amazon Elastic Compute Cloud.....	17
2.5 Τεχνολογικό Υπόβαθρο.....	18
2.5.1 Android Debug Bridge (ADB).....	18
2.5.2 JavaScript / jQuery .....	20
2.5.3 PHP / MySQL.....	22
2.5.4 Java.....	23
<b>Κεφάλαιο 3 Η Αρχιτεκτονική του SmartNet.....</b>	<b>24</b>
3.1 Επισκόπηση Αρχιτεκτονικής SmartNet .....	25
3.1.1 Υλικό που χρησιμοποιήθηκε.....	26
3.2 Συστατικά συστήματος .....	29
3.2.1 MySQL Βάση Δεδομένων (Backend).....	29
3.2.2 SmartNet Scheduler.....	31
3.2.3 Android Debug Bridge (ADB).....	33
3.2.4 Monkeyrunner .....	33
3.2.5 Web Server .....	33
<b>Κεφάλαιο 4 Το Σύστημα Διαπροσωπείας του SmartNet.....</b>	<b>35</b>
4.1 Εγγραφή στο SmartNet .....	36
4.2 Εισαγωγή στο SmartNet.....	37

4.3	<i>Κράτηση Συσκευών.....</i>	38
4.4	<i>Χρονοπρογραμματισμός Κράτησης Συσκευών.....</i>	40
4.5	<i>Διαχείριση Συσκευών.....</i>	41
4.6	<i>Οδηγίες.....</i>	47
4.7	<i>Διαχειριστής Αρχείων.....</i>	47
4.8	<i>Γενικά Χαρακτηριστικά.....</i>	49
4.9	<i>Παραγωγή Κλειδιών Εγγραφής ( Admin ).....</i>	50
<b>Κεφάλαιο 5 Πειραματική Λειτουργία της Υποδομής SmartNet .....</b>		<b>51</b>
5.1	<i>Πειραματική Διαδικασία.....</i>	52
5.2	<i>SmartTrace.....</i>	53
5.3	<i>Πειραματικά Αποτελέσματα.....</i>	54
<b>Κεφάλαιο 6 Συμπέρασμα και Μελλοντική Δουλειά.....</b>		<b>55</b>
6.1	<i>Συμπεράσματα.....</i>	55
6.2	<i>Μελλοντική Δουλειά.....</i>	55
<b>Παράρτημα Α Οδηγός Εγκατάστασης SmartNet .....</b>		<b>1</b>
<b>Παράρτημα Β Πηγαίος Κώδικας.....</b>		<b>1</b>
<b>Παράρτημα Γ Ψηφιακός Δίσκος.....</b>		<b>1</b>

# Κεφάλαιο 1

## Εισαγωγή

---

1.1 Το Βασικό Υπόβαθρο	1
1.2 Υποκίνηση Εργασίας	6
1.3 Περίγραμμα εργασίας	7

---

### 1.1 Το Βασικό Υπόβαθρο

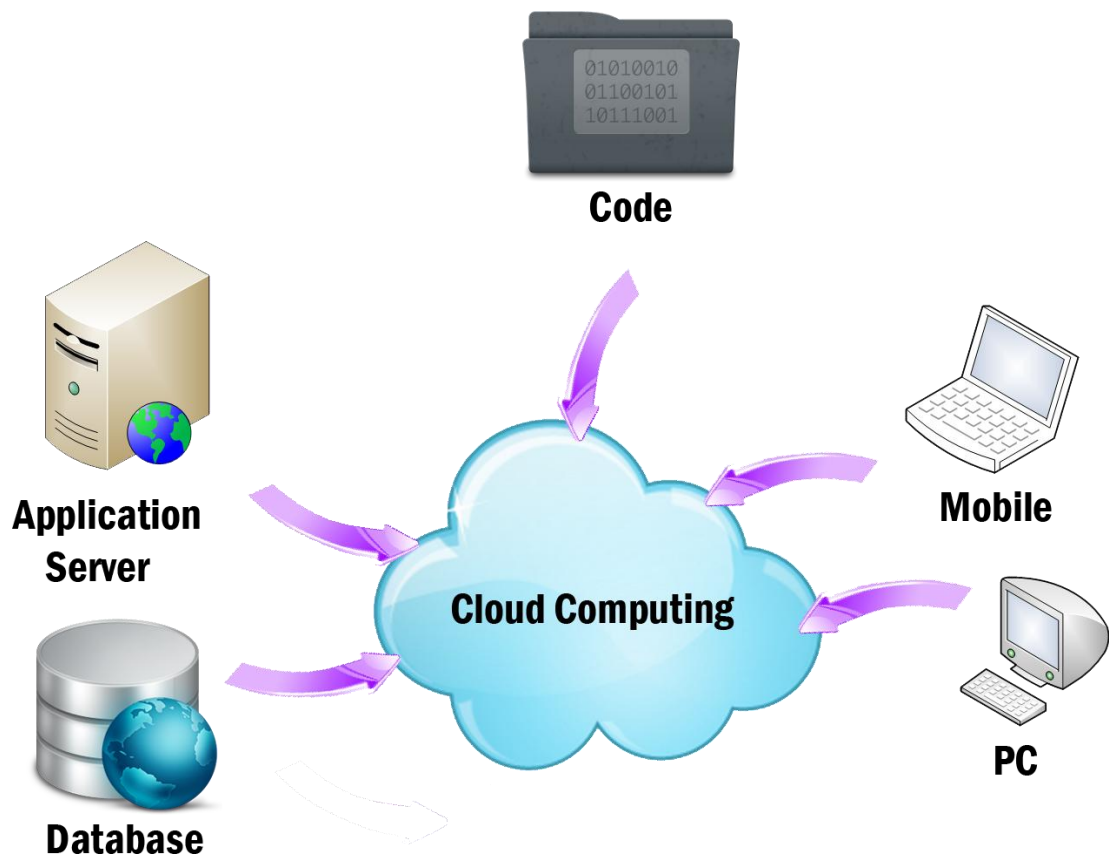
Το πρόβλημα της ανάπτυξης, διαχείρισης και ελέγχου εφαρμογών σε έξυπνες συσκευές και συγκεκριμένα συσκευές που έχουν ως λειτουργικό σύστημα το *Android* από την **Google** έρχεται να επιλύσει η δημιουργία του προγραμματιστικού νέφους *SmartNet*. Πριν όμως προχωρήσουμε σε θέματα που αφορούν τον σχεδιασμό και την υλοποίηση του, καλό θα ήταν να δώσουμε ένα υπόβαθρο γύρω από το τι είναι ένα προγραμματιστικό νέφος (*programming cloud*) καθώς επίσης και μια επεξήγηση για όρους και έννοιες που θα πρέπει να έχει λάβει κάποιος γνώση.

Σύμφωνα με το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (*NIST – National Institute of Standards and Technology*)[12] το **υπολογιστικό νέφος** (*cloud computing*), ορίζεται σαν ένα μοντέλο το οποίο επιτρέπει εύκολη πρόσβαση μέσω δικτύου σε κοινόχρηστους υπολογιστικούς πόρους (π.χ. δίκτυα, εξυπηρετητές, αποθηκευτικές μονάδες, εφαρμογές και άλλες υπηρεσίες) κατά παραγγελία, οι οποίοι μπορούν να κρατηθούν και να απελευθερωθούν εύκολα με ελάχιστη προσπάθεια για διαχείριση. Έτσι το μοντέλο αυτό παρέχει υπολογιστική ισχύ, λογισμικό, πρόσβαση δεδομένων και υπηρεσίες αποθήκευσης



δεδομένων στον τελικό χρήστη ενώ αυτός δεν χρειάζεται να γνωρίζει την φυσική τοποθεσία των υπηρεσιών ή την υποδομή του συστήματος που χρησιμοποιεί.

Μια υπηρεσία η οποία προσφέρεται από ένα υπολογιστικό νέφος έχει τρία χαρακτηριστικά τα οποία την διαφοροποιούν από τις κλασικές υπηρεσίες. Παρέχεται κατά παραγγελία, είναι ελαστική (ο χρήστης δηλαδή μπορεί να καθορίσει το “ποσό” της υπηρεσίας που χρειάζεται ανά πάσα στιγμή). Τέλος η υπηρεσία ελέγχεται πλήρως από τον πάροχο, δηλαδή ο χρήστης το μόνο που χρειάζεται για να την χρησιμοποιήσει είναι ένα υπολογιστή και σύνδεση στο διαδίκτυο.



Σχήμα 1.1 : Το μοντέλο του υπολογιστικού νέφους

Το *SmartNet* είναι ένα προγραμματιστικό νέφος το οποίο προσφέρει υπηρεσία η οποία έχει τα τρία χαρακτηριστικά που προαναφέρθηκαν. Ο χρήστης κατά παραγγελία μπορεί να

κρατήσει πόρους (κινητές συσκευές) ενώ είναι ελαστική αφού ανά πάσα στιγμή μπορεί να επιλέξει τον αριθμό και τον τύπο των συσκευών που θα κρατήσει. Επίσης το μόνο που χρειάζεται είναι ένας υπολογιστής και σύνδεση στο Διαδίκτυο για να χρησιμοποιήσει την συγκεκριμένη υπηρεσία. Το *SmartNet* χρησιμοποιείται για την ανάπτυξη και έλεγχο εφαρμογών και πρωτοκόλλων μέσω υπηρεσιών που παρέχει το Cloud Computing για προγραμματισμό πολλαπλών συσκευών, γι αυτό και αποτελεί αυτό που ονομάζουμε “*Programming Cloud*”.

Οι κινητές συσκευές στις οποίες αναφερόμαστε είναι αυτές που ανήκουν στην κατηγορία των “έξυπνων τηλεφώνων” (*Smartphones*)[13]. Smartphone θεωρείται οποιοδήποτε κινητό τηλέφωνο το οποίο προσφέρει περισσότερη υπολογιστική ικανότητα αλλά και συνδεσιμότητα (π.χ. πρόσβαση στο Διαδίκτυο) από ένα συμβατικό κινητό. Σε αυτά μπορούμε να ξεχωρίσουμε τρία κύρια γνωρίσματα. Πρώτον αποτελεί το **Λειτουργικό Σύστημα** (*Operating System*), το οποίο επιτρέπει σε ένα Smartphone να τρέχει διάφορες εφαρμογές. Το δεύτερο γνώρισμά τους είναι το **Λογισμικό**. Λόγω του Λειτουργικού Συστήματος που διαθέτουν έχουν την δυνατότητα να τρέχουν προγράμματα τα οποία στα συμβατικά κινητά δεν είναι δυνατό να τρέξουν (π.χ. *Adobe Reader* για ανάγνωση αρχείων τύπου *PDF*). Τέλος έχουν ως γνώρισμα την πρόσβαση στο Διαδίκτυο είτε μέσω των 3G δικτύων δεδομένων είτε μέσω *Wi-Fi*.

Παράδειγμα τέτοιων συσκευών αποτελούν δύο μοντέλα της **HTC** [14] τα οποία χρησιμοποιήθηκαν για την ανάπτυξη του *SmartNet*. Τα μοντέλα αυτά είναι το **HTC Desire** και το **HTC Hero** όπως φαίνονται στο *Σχήμα 1.2*. Τεχνικά χαρακτηριστικά των εν λόγω μοντέλων παρουσιάζονται στον *Πίνακα 1.1*



**Σχήμα 1.2 : HTC Desire (αριστερά) και HTC Hero (δεξιά)**

Μοντέλο	HTC Desire	HTC Hero
<b>2G Δίκτυο</b>	GSM 850 / 900 / 1800 / 1900	GSM 850 / 900 / 1800 / 1900
<b>3G Δίκτυο</b>	HSDPA 900 / 2100	HSDPA 900 / 2100
<b>Εσωτερική Μνήμη</b>	576 MB RAM; 512 MB ROM	288 MB RAM, 512 MB ROM
<b>Επέκταση Μνήμης</b>	microSD μέχρι 32GB (4GB)	microSD μέχρι 32GB (4GB)
<b>GPRS</b>	Class 10 (4+1/3+2 slots), 32 - 48 kbps	Class 10 (4+1/3+2 slots), 32 - 48 kbps
<b>3G</b>	HSDPA, 7.2 Mbps; HSUPA, 2 Mbps	
<b>WLAN</b>	Wi-Fi 802.11 b/g, Wi-Fi hotspot (Android 2.2)	Wi-Fi 802.11 b/g
<b>USB</b>	microUSB v2.0	miniUSB v2.0
<b>Λειτουργικό Σύστημα</b>	Android OS, v2.1 (Eclair), upgradable to v2.2	Android OS, v1.5 (Cupcake)
<b>Επεξεργαστής</b>	1 GHz Scorpion processor, Adreno 200 GPU, Qualcomm QSD8250 Snapdragon chipset	528 MHz ARM 11 processor, Adreno 130 GPU, Qualcomm MSM7200A chipset

**Πίνακας 1.1: Κόρια τεχνικά χαρακτηριστικά HTC Desire και HTC Hero**

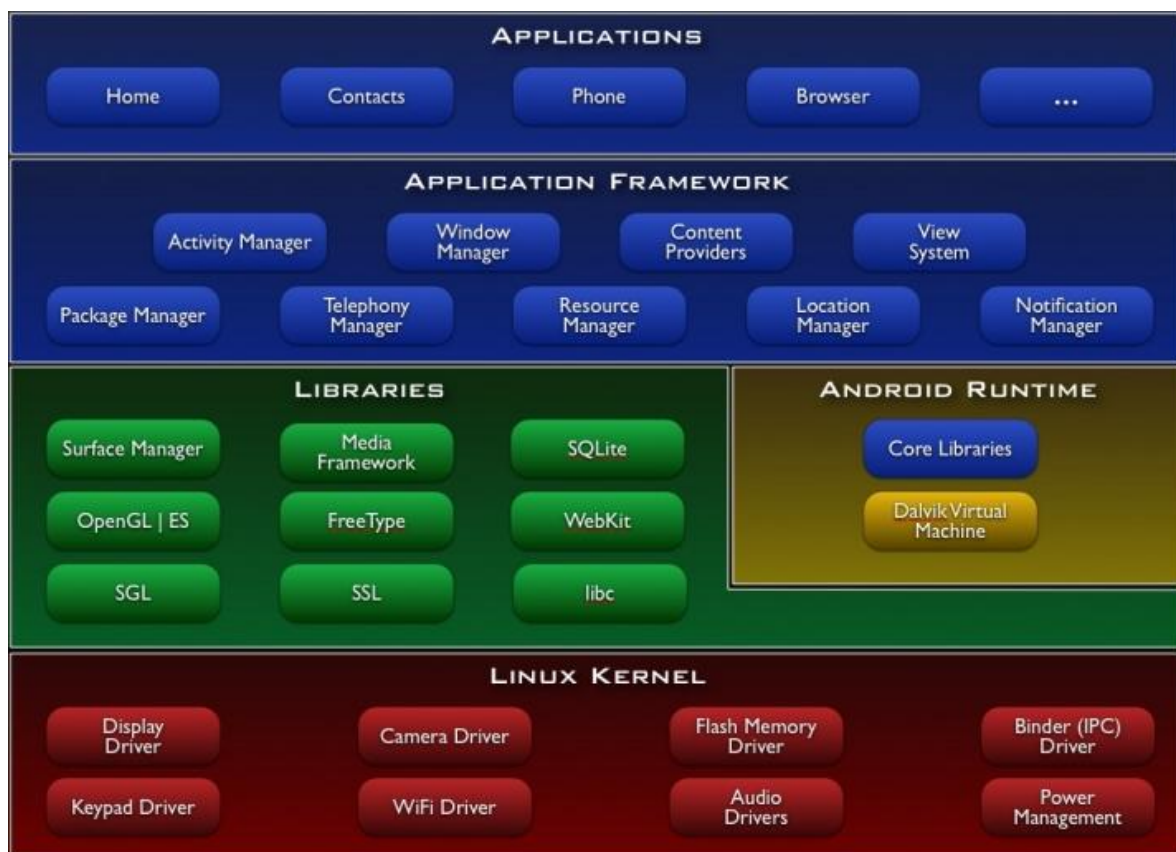
Όπως φαίνεται στον Πίνακα 1.1 και τα δύο μοντέλα έχουν ως Λειτουργικό Σύστημα το *Android*. Το *Android* αποτελεί μια στοίβα λογισμικού (*software stack*) για κινητές συσκευές η οποία περιλαμβάνει το Λειτουργικό Σύστημα, το Ενδιάμεσο Λογισμικό (*middleware*) και Εφαρμογές κλειδί (*key applications*). Την ανάπτυξη του λογισμικού αγόρασε η *Google Inc.* το 2005. Το *Android* έχει ως βάση για το Λειτουργικό του Σύστημα τον πυρήνα (*kernel*) των *Linux*[15]. Έτσι ο πυρήνας είναι προγραμματισμένος σε γλώσσα προγραμματισμού *C*[16], ενώ η διαπροσωπεία του χρήστη σε *Java*[17] όπως φαίνεται στο Σχήμα 1.3 .



Σχήμα 1.3 : Διαπροσωπεία Χρήστη στα *Android* με *Java*

Η *Google* έκδωσε το μεγαλύτερο κομμάτι του κώδικα των *Android* κάτω από την άδεια *Apache* [18] η οποία απευθύνεται σε δωρεάν λογισμικό ανοιχτού πηγαίου κώδικα (*open source*). Έτσι οποιοσδήποτε προγραμματιστής μπορεί να προγραμματίσει εφαρμογές σε

*Java* οι οποίες τρέχουν πάνω από ένα αντικειμενοστραφές (*object oriented*) πλαίσιο εφαρμογών πάνω από κάποιες βιβλιοθήκες πυρήνα της *Java* οι οποίες τρέχουν στην εικονική μηχανή (*virtual machine*) *Dalvik* [ref], όπως φαίνεται στο Σχήμα 1.4.



Σχήμα 1.4 : Η αρχιτεκτονική σε ένα σύστημα με *Android* [1]

## 1.2 Υποκίνηση Εργασίας

Η χρήση των *Smartphones* σήμερα ολοένα και αυξάνεται καθώς η αγορά όσον αφορά τις παραδοσιακές κινητές συσκευές άρχισε να μικραίνει προωθώντας ταυτόχρονα τα *Smartphones*. Σήμερα το Λειτουργικό Σύστημα *Android* έχει τις **μεγαλύτερες πωλήσεις παγκόσμια** [20,21] όσο αφορά πλατφόρμες για *Smartphones*. Επακόλουθο είναι η αύξηση

στην δημιουργία εφαρμογών για τέτοιες συσκευές. Το *Android* έχει μια αρκετά μεγάλη κοινότητα από προγραμματιστές οι οποίοι γράφουν εφαρμογές οι οποίες επεκτείνουν την λειτουργικότητα των συσκευών. Υπάρχουν προς το παρόν πάνω από **διακόσιες πενήντα χιλιάδες** (250 000) [22] τέτοιες εφαρμογές οι οποίες διατίθενται στο κοινό μέσω του *Android Market* [24], ένα *online* κατάστημα εφαρμογών το οποίο διαχειρίζεται η *Google*. Όπως φαίνεται και στο [23] από το *Android Market* έχουν κατεβαστεί πάνω από **τέσσερα εκατομμύρια** φορές εφαρμογές χωρίς να μετράμε αυτές που αποκτήθηκαν από άλλες υπηρεσίες, ενώ έχουμε ανά μέσω όρο περίπου **χίλιες** [23] (1000) καινούργιες εφαρμογές ημερησίως.

Αναγνωρίζοντας την ανάγκη για αυτοματοποίηση της διαδικασίας ανάπτυξης, ελέγχου και διαχείρισης τέτοιων εφαρμογών αλλά και την ανάγκη της παροχή της δυνατότητας αυτής σε προγραμματιστές και λαμβάνοντας υπόψη το γεγονός ότι δεν υπάρχει κάτι προς το παρόν, το SmartNet δίνει λύση. Οποιοσδήποτε προγραμματιστής μπορεί με απλά μια εγγραφή να κρατήσει κινητές συσκευές *Android* εύκολα και γρήγορα μέσω μιας Διαδικτυακής διαπροσωπείας και να αναπτύξει και να ελέγξει με αυτοματοποιημένο τρόπο την εφαρμογή του σε πραγματικές συσκευές.

### 1.3 Περίγραμμα εργασίας

Στο πρώτο κεφάλαιο παρουσιάσαμε το βασικό υπόβαθρο για καλύτερη κατανόηση αλλά και την υποκίνηση που υπάρχει για την δημιουργία ενός εύκολου τρόπου ανάπτυξης, ελέγχου και διαχείρισης εφαρμογών που απευθύνονται σε *Smartphones* και συγκεκριμένα *Smartphones* με *Android* Λειτουργικό Σύστημα. Στο δεύτερο κεφάλαιο θα παρουσιάσουμε σχετική δουλειά που έχει γίνει και θα δώσουμε το τεχνολογικό υπόβαθρο που χρησιμοποιήθηκε για την δημιουργία του SmartNet. Στο τρίτο κεφάλαιο της εργασίας αυτής θα παρουσιάσουμε την αρχιτεκτονική του SmartNet και τα επιμέρους συστατικά του συστήματος. Στο τέταρτο κεφάλαιο θα παρουσιάσουμε το Σύστημα Διαπροσωπείας του SmartNet και θα επεξηγήσουμε τις βασικές του λειτουργίες. Στο πέμπτο κεφάλαιο θα γίνει παρουσίαση της πειραματικής λειτουργίας της υποδομής του SmartNet και τέλος στο κεφάλαιο έξι θα παραθέσουμε τα τελικά συμπεράσματα και μελλοντική δουλειά που

μπορεί να γίνει στο SmartNet. Στο Παράρτημα Α παρουσιάζεται Οδηγός Εγκατάστασης του SmartNet, στο Παράρτημα Β ο πηγαίος κώδικας και τέλος στο Παράρτημα Γ ο ψηφιακός δίσκος με τα απαραίτητα αρχεία.

# Κεφάλαιο 2

## Σχετική Δουλειά και Τεχνολογικό Υπόβαθρο

---

2.1 MoteLab	10
2.2 WISEBED	12
2.3 Remote Device Access	14
2.4 Amazon Elastic Compute Cloud	17
2.5 Τεχνολογικό Υπόβαθρο	18

---

Στο κεφάλαιο αυτό θα παρουσιάσουμε σχετική δουλειά που έχει γίνει και συγκεκριμένα το *MoteLab* και το *Wireless Sensor Network Testbed (WISEBED)*. Τα δύο αυτά συστήματα έχουν να κάνουν με Ασύρματα Δίκτυα Αισθητήρων και θα αναπτυχθούν πιο κάτω. Επίσης μελετήθηκε το *Remote Device Access* από την *Nokia* και το *Amazon Elastic Compute Cloud (Amazon EC2)* από την *Amazon*. Τέλος σε αυτό το κεφάλαιο θα παρουσιάσουμε το τεχνολογικό υπόβαθρο που χρησιμοποιήθηκε για την ανάπτυξη του *SmartNet*.

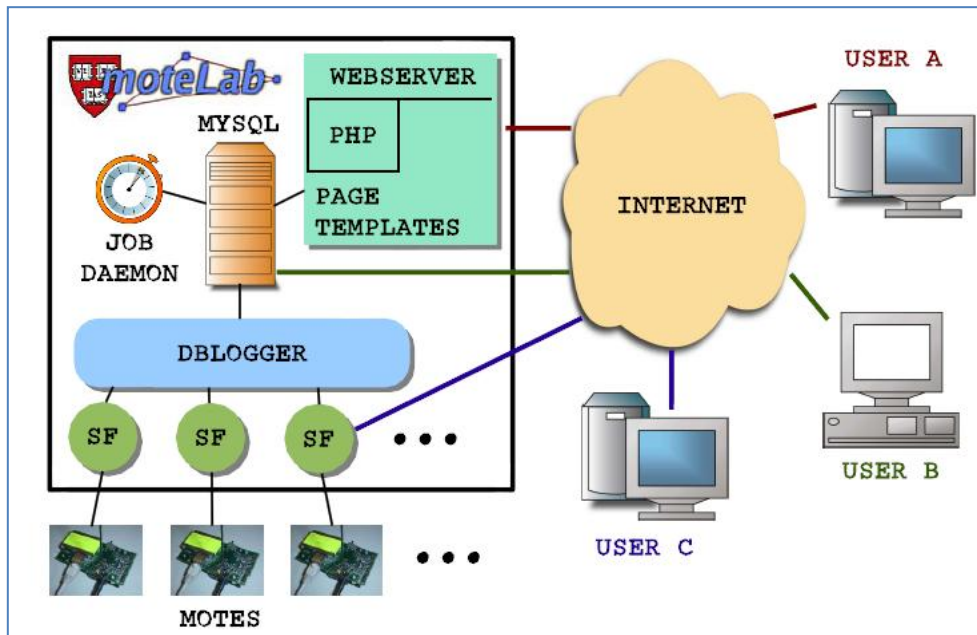


## 2.1 MoteLab

Καθώς τα ασύρματα δίκτυα αισθητήρων αναπτύσσονται ολοένα και περισσότερο δίνοντας ολοένα και μεγαλύτερο εύρος στο ερευνητικό πεδίο όσον αφορά την επιστήμη της Πληροφορικής υπάρχουν αρκετές δυσκολίες σχετικά με την ανάπτυξη και την απασφαλμάτωση εφαρμογών σε ρεαλιστικά μεγάλα δίκτυα οι οποίες δεν είχαν επιλυθεί.

Ο έλεγχος στα ασύρματα δίκτυα αισθητήρων είναι αρκετά χρονοβόρος, αφού η διαδικασία τοποθέτησης του δικτύου σε ένα ρεαλιστικό περιβάλλον απαιτεί επαναπρογραμματισμό όλων των αισθητήρων ένα-ένα. Εάν και υπάρχει πληθώρα από εργαλεία για αξιολόγηση μεγάλων δικτύων αισθητήρων μέσω προσομοίωσης [25][26], μόνο μια πραγματική πλατφόρμα δοκιμών ασύρματων δικτύων αισθητήρων μπορεί να προσφέρει τον ρεαλισμό που χρειάζεται για να κατανοήσει κανείς έννοιες κλειδιά για την λειτουργία του δικτύου όπως τους περιορισμούς στους πόρους και στην ενέργεια, αλλά και την απώλεια επικοινωνίας (*communication loss*).

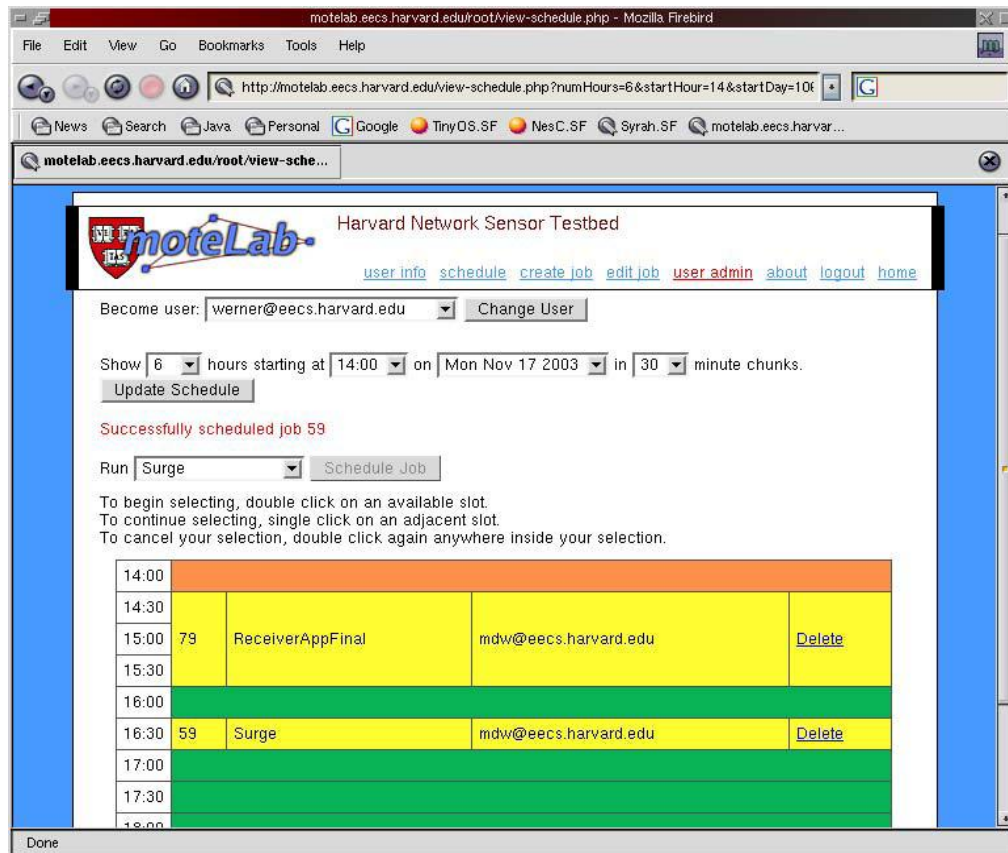
Για να επιλύσουν τα προβλήματα αυτά το πανεπιστήμιο του **Harvard** [27] προχώρησε στον σχεδιασμό και υλοποίηση της πλατφόρμας “*MoteLab*”. Το *MoteLab* μέσω μιας Διαδικτυακής διαπροσωπείας επιτρέπει σε χρήστες τον έλεγχο ασύρματων δικτύων αισθητήρων σε πραγματικές συνθήκες. Συγκεκριμένα επιτρέπει στους χρήστες την δημιουργία και δρομολόγηση πειραμάτων, αυτοματοποιώντας τον επαναπρογραμματισμό των αισθητήρων, ενώ αποθηκεύει δεδομένα από τα συγκεκριμένα πειράματα σε μια βάση δεδομένων (*MySQL*[28]). Οι χρήστες έχουν πρόσβαση στα δεδομένα για τα δικά τους πειράματα είτε μέσω της διαπροσωπείας στο Διαδίκτυο είτε απευθείας από την βάση δεδομένων (Σχήμα 2.1).



Σχήμα 2.1 : Αρχιτεκτονική όψη του MoteLab [7]

Το *MoteLab* έχει εγκατασταθεί σε ένα δίκτυο με τριάντα (30) ασύρματους αισθητήρες οι οποίοι βρίσκονται κατανεμημένοι σε τρις ορόφους σε κτίρια στο Πανεπιστήμιο του *Harvard*. Στο Σχήμα 2.1 μπορούμε να διακρίνουμε τέσσερα συστατικά της αρχιτεκτονικής του *MoteLab*:

- **MySQL Βάση Δεδομένων**: Αποθηκεύει δεδομένα τα οποία συλλέγονται κατά την εκτέλεση των πειραμάτων, πληροφορίες που χρειάζονται για την δημιουργία του περιεχομένου της σελίδας και την κατάσταση των αισθητήρων.
- **Web Interface** : Σελίδες οι οποίες δημιουργούνται δυναμικά μέσω *PHP scripts*, παρέχουν στον χρήστη μια διαπροσωπεία μέσω του διαδικτύου για δημιουργία και χρονοπρογραμματισμό εργασιών (*jobs*), συλλογή δεδομένων και ρυθμίσεων που έχουν να κάνουν με το σύστημα (Σχήμα 2.2).
- **DBLogger** : Πρόγραμμα γραμμένο σε *Java* υπεύθυνο για την συλλογή και ανάλυση των δεδομένων που επιστρέφονται από τα πειράματα.
- **Job Demon** : *Script* γραμμένο σε *Pearl* το οποίο τρέχει σαν *cron job* και είναι υπεύθυνο για την εκκίνηση και τερματισμό εργασιών.

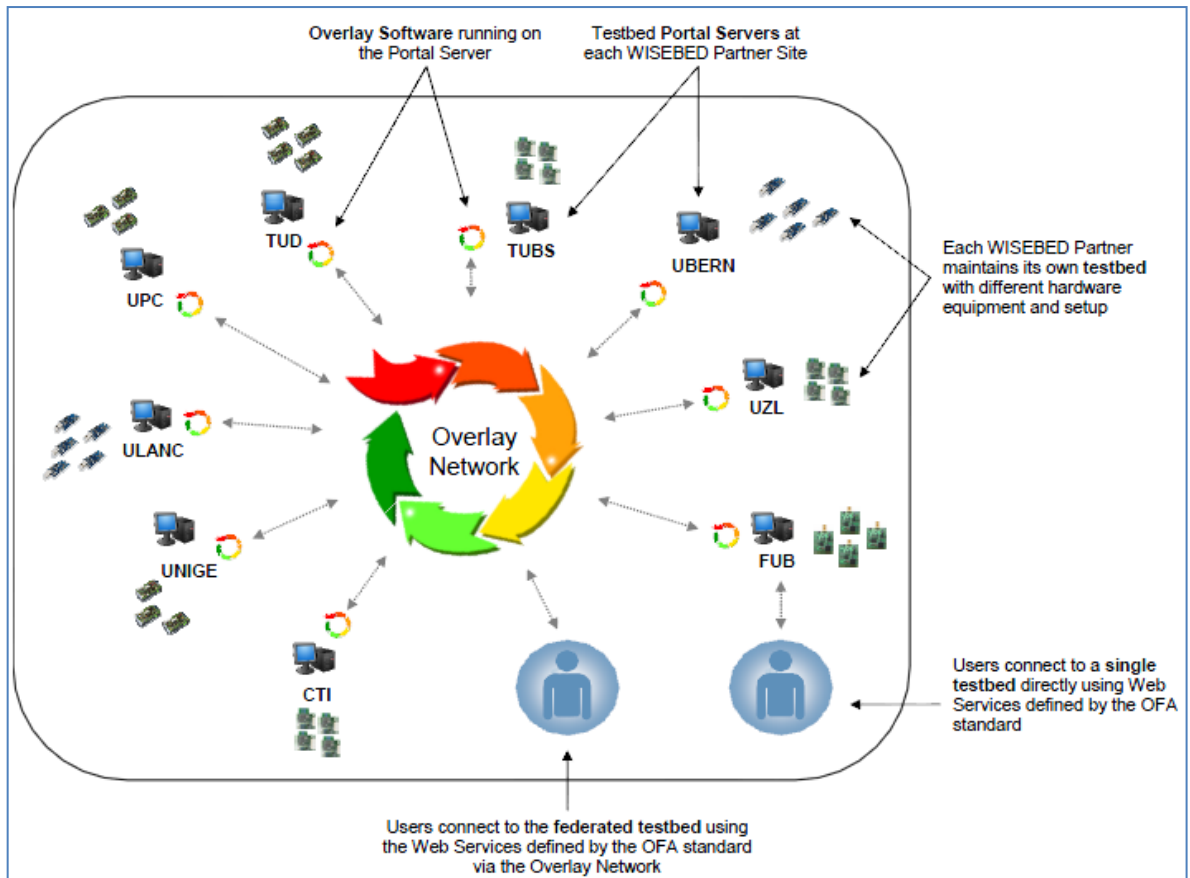


Σχήμα 2.2 : Η Διαδικτυακή Διαπροσωπεία του MoteLab [7]

## 2.2 WISEBED

Το *WISEBED* (*Wireless Sensor Network Testbeds*) είναι ένα Ευρωπαϊκό πρόγραμμα το οποίο έχει ως στόχο την εγκαθίδρυση ενός πανευρωπαϊκού ασύρματου δικτύου αισθητήρων το οποίο θα είναι προσβάσιμο από Ευρωπαίους ερευνητές για όλα τα πειραματικά πεδία. Για την επίτευξη του συγκεκριμένου στόχου συνένωσαν και επέκτειναν ήδη υπάρχουσες δοκιμαστικές πλατφόρμες (*testbeds*) που βρίσκονται στην Ευρώπη δημιουργώντας έτσι μια ομοσπονδία από καταναμημένα εργαστήρια για δοκιμές.

Συγκεκριμένα μελετήθηκε η αρχιτεκτονική του *WISEBED* η οποία και παρουσιάζεται πιο κάτω και φαίνεται στο Σχήμα 2.3.



Σχήμα 2.3 : Γενική Αρχιτεκτονική του WISEBED[8]

Ουσιαστικά η αρχιτεκτονική του WISEBED βασίζεται σε μια ιεραρχία από επίπεδα, όπου κάθε επίπεδο αποτελείται από μία ή περισσότερες όμοιες συσκευές (*peers*). Κάθε μια από αυτές τις συσκευές μπορεί να είναι ένας παραδοσιακά δικτυωμένος επεξεργαστής ή μια ασύρματη συσκευή αισθητήρων. Μπορούμε να διακρίνουμε τρία επίπεδα όπου το κάθε ένα έχει τον δικό του συγκεκριμένο ρόλο στο σύστημα:

- Το κατώτατο επίπεδο περιέχει το ασύρματους κόμβους αισθητήρων οι οποίοι μπορούν να τρέχουν το *iSense* [29], το *Contiki* [30], το *TinyOS* [31] ή οποιοδήποτε άλλο υπάρχον λειτουργικό σύστημα. Οι συσκευές αυτές σχηματίζουν έτσι ασύρματα δίκτυα τα οποία αποτελούν τις δοκιμαστικές πλατφόρμες των Ασύρματων Δικτύων Αισθητήρων.

- Στο πιο πάνω επίπεδο οι δοκιμαστικές πλατφόρμες κάθε συνέταιρου (π.χ *TUD*, *UPC*, *ULAN*, *CTI* κτλ. όπως φαίνονται στο *Σχήμα 2.3*) ελέγχονται από τους λεγόμενους *Portal Servers* οι οποίοι παρέχουν πρόσβαση και διεπαφές για διαχείριση και λειτουργία τους. Οι χρήστες μπορούν να ενωθούν σε μια δοκιμαστική πλατφόρμα εάν επιθυμεί απευθείας μέσω το Διαδικτύου έχοντας πρόσβαση στην διαπροσωπεία που παρέχεται από τον συγκεκριμένο *Portal Server*. Το μόνο που χρειάζεται είναι η δημόσια IP διεύθυνση του εν λόγω *Portal Server*.
- Τέλος στο ανώτερο επίπεδο οι *Portal Servers* κάθε δοκιμαστικής πλατφόρμας είναι συνδεδεμένοι μεταξύ τους με ένα επικαλυμμένο δίκτυο δίνοντας έτσι την δυνατότητα στους *peers* που είναι ενωμένοι στο δίκτυο αυτό να έχουν πρόσβαση σε ένα ή περισσότερους *Portal Servers*. Έτσι μπορούν να χρησιμοποιηθούν πολλαπλές δοκιμαστικές πλατφόρμες κατανεμημένα.

### 2.3 Remote Device Access








Το *Remote Device Access (RDA)* αποτελεί μια υπηρεσία η οποία έχει δημιουργηθεί από την *Nokia* με κύριο σκοπό την παροχή βοήθειας στην ανάπτυξη και έλεγχο εφαρμογών για κινητές συσκευές *Nokia*. Επίσης δημιουργεί ευκαιρίες για απασχόληση προγραμματιστών ενώ συνενώνει ταυτόχρονα τους προγραμματιστές κινητών συσκευών *Nokia* σε ένα κοινό προγραμματιστικό πλαίσιο.

Συγκεκριμένα το *RDA* επιτρέπει στους προγραμματιστές να ελέγχουν τις εφαρμογές και τις υπηρεσίες τους που απευθύνονται σε κινητές συσκευές όπου το Λειτουργικό Σύστημα είναι ένα από τα δύο που προσφέρει η *Nokia* για *Smartphones (Symbian [32] ή Maemo [33])* απομακρυσμένα σε πραγματικές συσκευές (*Σχήμα 2.4*). Το κύριο χαρακτηριστικό της υπηρεσίας αυτής είναι ο απομακρυσμένος έλεγχος μιας συσκευής, εγκατάσταση και εκτέλεση μιας εφαρμογής αλλά και η μεταφορά αρχείων από και προς την συσκευή. Για να χρησιμοποιήσει κάποιος την υπηρεσία αυτή χρειάζεται να εγγραφεί στα *forums* της *Nokia* και να έχει ήδη εγκατεστημένο το *Java Web Start [34]*.



Σχήμα 2.4 : Η δομή του *Remote Device Access* από την *Nokia* [9]

Εδώ μελετήθηκε κυρίως ο τρόπος με τον οποίο διαχειρίζονται η συσκευές (κράτηση και απελευθέρωση συσκευών), η διαπροσωπεία του χρήστη αλλά και οι λειτουργίες που παρέχονται. Όπως βλέπουμε στο Σχήμα 2.5 ο χρήστης μπορεί να κρατήσει μια συσκευή για συγκεκριμένο χρονικό διάστημα και να την ξεκινήσει. Επίσης ο χρήστης μπορεί να κρατήσει μια συσκευή για μελλοντική χρήση, για αυτό και ενημερώνεται για το πόσο χρόνο είναι διαθέσιμη μια συγκεκριμένη συσκευή.

 <p><b>Nokia 5230</b> 352014047231803 +358447147420 SW 21.0.004</p>	 <p><b>Nokia 5320 XpressMusic</b> 354826020021572 +358504877263 SW 5.16</p>	 <p><b>Nokia 5530 XpressMusic</b> 359368036340766 +358447835592 SW 31.0.005</p> <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;">           Reserve for 15 minutes            Reserve for 30 minutes            Reserve for 45 minutes         </div> <p style="text-align: center;">Available for 13 minutes</p> <p style="text-align: center; border: 1px solid gray; padding: 2px 10px;">Start</p> <p style="text-align: center; color: red;">Make later reservation</p>
 <p><b>Nokia 5800 XpressMusic</b> 354182020013056 +358447147414 SW 40.0.005</p>	 <p><b>Nokia 5800 XpressMusic</b> 352012046013966 +358447147491 SW 50.0.005</p>	
 <p><b>Nokia E70</b> 357580000766980 +358504858033 SW 3.0633.09.04</p>	 <p><b>Nokia 5800 XpressMusic</b> 352012046014337 +358447147429 SW 50.0.005</p>	

Σχήμα 2.5 : Κράτηση και εκκίνηση συσκευής στο *RDA* [9]



**Σχήμα 2.6 : Διαπροσωπεία Χρήστη στο RDA**

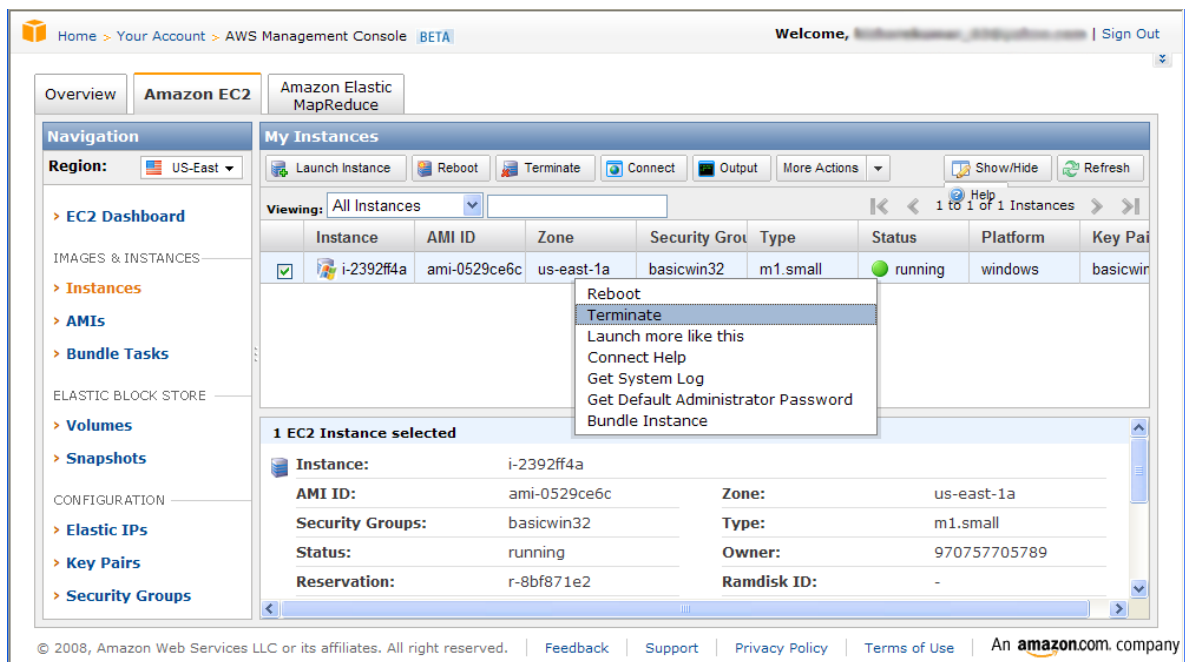
Στο πιο πάνω σχήμα (Σχήμα 2.6) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το RDA. Στο σχήμα υπάρχουν περιοχές που έχουν απαριθμηθεί (από το 1 μέχρι και το 6). Ακολουθεί περιγραφή των αριθμημένων περιοχών καθώς και η λειτουργία τους:

- 1. Διαχείριση Αρχείων :** Επιτρέπει την μεταφορά αρχείων από και προς την κινητή συσκευή.
- 2. Εγκατάσταση Λογισμικού :** Επιτρέπει την εγκατάσταση λογισμικού στην κινητή συσκευή.
- 3. Λήψη στιγμιότυπου οθόνης :** Λαμβάνει και αποθηκεύει ένα στιγμιότυπο της οθόνης της κινητής συσκευής.
- 4. Λήψη video :** Λαμβάνει και αποθηκεύει video της οθόνης της κινητής συσκευής.

5. **Επιλογή Ποιότητας** : Επιτρέπει στον χρήστη να επιλέξει την ποιότητα της εικόνας που παρουσιάζεται σε αυτόν.
6. **Οθόνη Κινητού** : Αντιπροσωπεύει την οθόνη της κινητής συσκευής στην οποία ο χρήστης ενώνεται
7. **Χειριστήριο Κινητού** : Επιτρέπει στον χρήστη να ελέγχει την κινητή συσκευή.

## 2.4 Amazon Elastic Compute Cloud

Το *Amazon Elastic Compute Cloud (Amazon EC2)* είναι μια διαδικτυακή υπηρεσία η οποία προσφέρει υπολογιστική ισχύ στο νέφος. Η απλή διαπροσωπεία (Σχήμα 2.6) που παρέχεται επιτρέπει στον χρήστη να κρατήσει και να ρυθμίσει τις μηχανές που θέλει σύμφωνα με τις ανάγκες του.



Σχήμα 2.7 : Διαπροσωπεία Χρήστη του *Amazon EC2* [10]

Το *Amazon EC2* προσφέρει ένα εικονικό υπολογιστικό περιβάλλον το οποίο σου επιτρέπει να χρησιμοποιήσεις την διαδικτυακή διαπροσωπεία για να “ανεβάσεις” διαφορετικά



στιγμιότυπα με διαφορετικά λειτουργικά συστήματα και να φορτώσεις σε αυτά τις εφαρμογές που θέλεις να τρέξεις.

Για να χρησιμοποιήσει κάποιος το *Amazon EC2* πρέπει να περάσει από τα πιο κάτω βήματα:

- Επιλογή μιας προκαθορισμένης “εικόνας” έτσι ώστε να αρχίσει να την χρησιμοποιεί ή δημιουργία ενός *Amazon Machine Image (AMI)* το οποίο περιέχει τις εφαρμογές, τις βιβλιοθήκες, τα δεδομένα και τις σχετικές ρυθμίσεις που επιθυμεί.
- Ρύθμιση της ασφάλειας και της διαδικτυακής πρόσβασης που θέλει να παρέχεται στο στιγμιότυπο του *Amazon EC2*.
- Επιλογή ποιους τύπους στιγμιότυπων των μηχανών και του λειτουργικού συστήματος που χρειάζεται. Στην συνέχεια μπορεί να ξεκινήσει, να τερματίσει και να παρακολουθήσει όσα στιγμιότυπα των *AMI* χρειάζεται χρησιμοποιώντας τις διαδικτυακές υπηρεσίες που προσφέρονται από διάφορα εργαλεία.
- Τέλος μπορεί να επιλέξει εάν θέλει να τρέχει η εφαρμογή του από πολλαπλές τοποθεσίες, να κάνει χρήση στατικού IP ή να προσκολλήσει σε κάθε στιγμιότυπο μόνιμες αποθηκευτικές μονάδες.

## 2.5 Τεχνολογικό Υπόβαθρο

Στη συνέχεια θα παρουσιάσουμε το Τεχνολογικό Υπόβαθρο το οποίο χρησιμοποιήθηκε για την υλοποίηση του *SmartNet*.

### 2.5.1 Android Debug Bridge (ADB)

Το *Android Debug Bridge (ADB)* είναι ένα ευέλικτο εργαλείο το οποίο επιτρέπει την διαχείριση της κατάστασης είτε ενός στιγμιότυπου σε ένα προσομοιωτή είτε μιας συσκευής η οποία τρέχει **Android** μέσω του ηλεκτρονικού υπολογιστή. Αποτελεί ένα πρόγραμμα εξυπηρετητή-πελάτη (server-client) και συμπεριλαμβάνει τρία συστατικά:

- Ένα *client*, ο οποίος τρέχει στην μηχανή που αναπτύσσεται η εφαρμογή

- Ένα *server* ο οποίος τρέχει σαν διεργασία στην μηχανή που αναπτύσσεται η εφαρμογή. Ο εξυπηρετητής αυτός είναι υπεύθυνος για την διαχείριση της επικοινωνίας μεταξύ του *client* και του *adb daemon* που τρέχει στον εξομοιωτή ή την συσκευή.
- Ένα *daemon* ο οποίος τρέχει σαν διεργασία στο υπόβαθρο σε κάθε εξομοιωτή ή συσκευή.

Όταν ξεκινήσουμε ένα *adb client*, ο *client* πρώτα ελέγχει εάν ήδη τρέχει κάποια διεργασία εξυπηρετητή. Εάν δεν τρέχει τότε ξεκινάει μια. Όταν ο εξυπηρετητής ξεκινήσει δεσμεύει την *TCP* θύρα 5037 και περιμένει εντολές οι οποίες στέλλονται από τους *adb clients* (όλοι οι *adb clients* χρησιμοποιούν την θύρα 5037 για να επικοινωνούν με τον *adb server*).

Ο εξυπηρετητής στη συνέχεια εγκαθιδρύει σύνδεση με όλα τα στιγμιότυπα των συσκευών και των εξομοιωτών που τρέχουν. Για να εντοπίσει συσκευές και εξομοιωτές σαρώνει όλες τις μονές θύρες από το 5555 μέχρι και το 5585 αφού αυτό είναι το εύρος που χρησιμοποιείται από τις συσκευές και τους εξομοιωτές. Ακολουθούν παραδείγματα εκτέλεσης βασικών εντολών του *ADB* για καλύτερη κατανόηση της λειτουργίας του.

Για να δούμε ποιές συσκευές/προσομοιωτές είναι ενωμένοι στο μηχάνημα εκτελούμε την πιο κάτω εντολή και παίρνουμε το ακόλουθο αποτέλεσμα:

```
$ adb devices
List of devices attached
SH0APPL01201    device
HT99TL902572   device
SH0APPL01284   device
emulator-5554   device
```

Όπως βλέπουμε στο σύστημα υπάρχουν τρεις συσκευές (*Android*) ενωμένες και ένας προσομοιωτής (*emulator-5554*). Για την κάθε συσκευή παρουσιάζεται ο μοναδικός της αριθμός. Μπορούμε να εκτελέσουμε διάφορες ενέργειες οι οποίες μας παρέρχονται μέσω του *ADB*, όπως για παράδειγμα εγκατάσταση μιας εφαρμογής. Εάν υπάρχουν περισσότερες

από μια συσκευές/προσομοιωτές ενωμένοι πρέπει να καθορίσουμε σε μια συσκευή/προσομοιωτή θέλουμε να εκτελέσουμε μια επιθυμητή ενέργεια με την πιο κάτω εντολή:

```
$ adb -s <serialNumber> <command>
```

Εάν υποθέσουμε ότι θέλουμε να εγκαταστήσουμε στην πρώτη συσκευή μια εφαρμογή με το όνομα *HelloWorld.apk* πρέπει πρώτα να επισημάνουμε την συσκευή στην οποία θέλουμε να εκτελέσουμε την ενέργεια κάνοντας χρήση του διακόπτη `-s` όπως φαίνεται πιο κάτω:

```
$ adb -s SH0APPL01201 install HelloWorld.apk
```

Για να πάρει ο χρήστης μια λίστα με όλες τις εντολές τις οποίες παρέχει ο *ADB* μπορεί να εκτελέσει την πιο κάτω εντολή:

```
$ adb -help
```

### 2.5.2 *JavaScript / jQuery*

*JavaScript* είναι μια αντικειμενοστραφής scripting γλώσσα η οποία χρησιμοποιείται παραδοσιακά από τους *web browsers* έτσι ώστε να παρέχουν πιο πλούσιες διαπροσωπείες χρήστη και δυναμικές ιστοσελίδες.

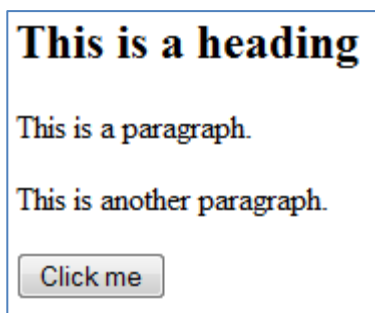
Το *jQuery* είναι μια βιβλιοθήκη της JavaScript η οποία σχεδιαστική για να απλοποιήσει την διαδικασία προγραμματισμού της *HTML* στην πλευρά του *client*. Σήμερα αποτελεί την πιο δημοφιλή βιβλιοθήκη της *JavaScript*[35][36]. Το *jQuery* είναι δωρεάν λογισμικό ανοικτού πηγαίου κώδικα. Η σύνταξη της εν λόγω βιβλιοθήκης σχεδιάστηκε έτσι ώστε να κάνει πιο εύκολη την μετακίνηση σε ένα έγγραφο, την επιλογή στοιχείων

DOM [37], την δημιουργία *animation*, χειρισμού συμβάντων, και την ανάπτυξη *AJAX* [38] εφαρμογών. Το *jQuery* προσφέρει επίσης την δυνατότητα στους προγραμματιστές να δημιουργήσουν *plugins* πάνω από την βιβλιοθήκη της *JavaScript*. Ακολουθεί παράδειγμα του *jQuery* για καλύτερη κατανόηση.

Ο κώδικας στο Σχήμα 2.8 έχει σαν αποτέλεσμα την παράγωγή του HTML αρχείου που παρουσιάζεται στο Σχήμα 2.9.

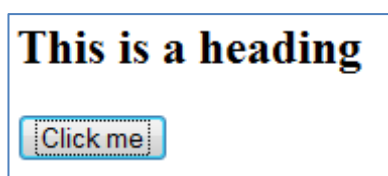
```
1. <html>
2. <head>
3. <script type="text/javascript" src="jquery.js"></script>
4. <script type="text/javascript">
5. $(document).ready(function() {
6.   $("button").click(function() {
7.     $("p").hide();
8.   });
9. });
10. </script>
11. </head>
12.
13. <body>
14. <h2>This is a heading</h2>
15. <p>This is a paragraph.</p>
16. <p>This is another paragraph.</p>
17. <button>Click me</button>
18. </body>
19. </html>
```

Σχήμα 2.8 : Παράδειγμα Κώδικα *jQuery*



Σχήμα 2.9 : Αρχείο *HTML* του κώδικα στο Σχήμα 2.8

Στην γραμμή 3 στο *Σχήμα 2.8* καθορίζουμε την χρήση του *jQuery*. Σκοπός του παραδείγματος είναι η παρουσίαση της εύκολης επιλογής στοιχείων *DOM* και της εκτέλεσης ενεργειών που παρέχει το *jQuery*. Όπως φαίνεται στην γραμμή 6 ( `$(“button”).click(function(){` ) στο ίδιο σχήμα μέσω του `$(“button”)` επιλέγουμε εύκολα και απλά τα στοιχεία τύπου “button” και ορίζουμε μια συνάρτηση η οποία εκτελείται όταν γίνει click στο εν λόγω στοιχείο μέσω της `click(function){`. Η συνάρτηση η οποία φαίνεται στην γραμμή υπ’ αριθμό 7 επιλέγει όλα τα στοιχεία τύπου “p” και εκτελεί σε αυτά την ενέργεια `hide()` η οποία αποκρύπτει από το *HTML* αρχείο τα επιλεγμένα στοιχεία. Μετά το πάτημα του κουμπιού “Click Me” που φαίνεται στο *Σχήμα 2.9*, έχουμε το αποτέλεσμα που παρουσιάζεται στο *Σχήμα 2.10*. Οι γραμμές 15 και 16 αποκρύπτονται δυναμικά χωρίς την επαναφόρτωση ολόκληρου του αρχείου αφού είναι τύπου “p”.



**Σχήμα 2.10 :** Αρχείο *HTML* μετά το πάτημα του Click Me

### 2.5.3 *PHP / MySQL*

Η *MySQL* είναι ένα σύστημα διαχείρισης σχεσιακής βάση που τρέχει σαν εξυπηρετητής παρέχοντας έτσι πρόσβαση σε πολλούς χρήστες σε πολλές βάσεις ταυτόχρονα.

Η *PHP* (*Hypertext Preprocessor*) είναι και αυτή μια *scripting* γλώσσα η οποία δημιουργήθηκε αρχικά για ανάπτυξη δυναμικών ιστοσελίδων. Για να το πετύχει αυτό ο κώδικας της *PHP* είναι ενσωματωμένος στον πηγαίο κώδικα του *HTML* αρχείου και ερμηνεύεται από ένα διαδικτυακό εξυπηρετητή ο οποίος έχει ήδη εγκατεστημένο τον επεξεργαστή της *PHP* ο οποίος με την σειρά του δημιουργεί την ιστοσελίδα.

#### 2.5.4 Java

Η **Java** είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού η οποία αναπτύχθηκε από την **Sun Microsystems** [39]. Είναι αρκετά παρόμοια με την **C** και την **C++** όσο αφορά την σύνταξή της αλλά προσφέρει ένα πιο απλό μοντέλο αντικειμένων και λιγότερες δυνατότητες χαμηλού επιπέδου. Οι εφαρμογές της **Java** γίνονται *compiled* σε *bytecode* (**class file**) το οποίο μπορεί να τρέξει σε οποιαδήποτε **Εικονική Μηχανή Java** (**Java Virtual Machine – JVM**) ανεξάρτητα της αρχιτεκτονικής του ηλεκτρονικού υπολογιστή (Σχήμα 2.11). Η **Java** χαρακτηρίζεται από τα εξής: απλή, αντικειμενοστραφής, συμβατή με δικτυακά πρωτόκολλα, ουδέτερη της υποκείμενης αρχιτεκτονικής, φορητή ασφαλής, υψηλής απόδοσης, δυναμική, σταθερή, interpreted και multithreaded. Σκοπός της είναι να επιτρέπει στους προγραμματιστές να γράφουν μία φορά τον κώδικα και να τρέχει παντού.



Σχήμα 2.11 : Τρόπος Λειτουργίας της **Java**

# Κεφάλαιο 3

## Η Αρχιτεκτονική του *SmartNet*

---

3.1 Επισκόπηση Αρχιτεκτονικής <i>SmartNet</i>	25
3.2 Συστατικά Συστήματος	29

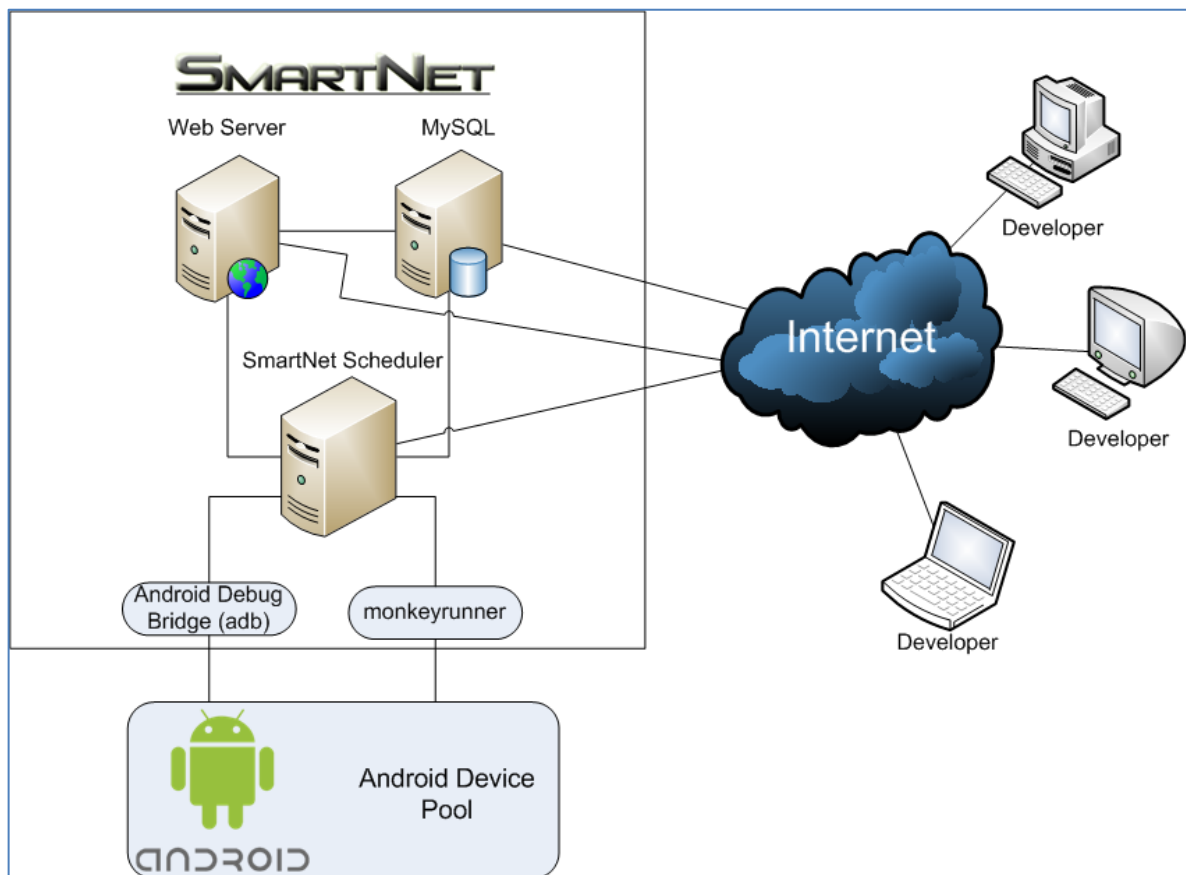
---

Στο προηγούμενο κεφάλαιο παρουσιάστηκε η σχετική δουλειά που έχει γίνει όπως επίσης και το τεχνολογικό υπόβαθρο, για τις τεχνολογίες που χρησιμοποιήθηκαν για την δημιουργία του *SmartNet* με σκοπό την ευκολότερη κατανόηση του παρόντος κεφαλαίου. Σε αυτό το κεφάλαιο αρχικά θα γίνει επισκόπηση της αρχιτεκτονικής του *SmartNet* με μια σύντομη περιγραφή για το ρόλο του κάθε συστατικού και στην συνέχεια θα γίνει πιο λεπτομερής ανάλυση των επιμέρους συστατικών του.

### 3.1 Επισκόπηση Αρχιτεκτονικής SmartNet

Το *SmartNet* απαρτίζεται από ένα σύνολο εργαλείων λογισμικού για την διαχείριση μιας ή πολλαπλών συσκευών *Android* με σκοπό την γρηγορότερη και ευκολότερη ανάπτυξη εφαρμογών. Ένας κεντρικός εξυπηρετητής είναι υπεύθυνος για την δέσμευση και αποδέσμευση των συσκευών, την διαχείρισή τους, την αποθήκευση δεδομένων και αρχείων καθώς επίσης και για την παροχή διαδικτυακής διαπροσωπείας στους χρήστες.

Το *SmartNet* αποτελείται από αρκετά διαφορετικά συστατικά λογισμικού (Σχήμα 3.1) για τα οποία ακολουθεί σύντομη περιγραφή.



Σχήμα 3.1 : Αρχιτεκτονική Όψη του *SmartNet*



Τα κύρια κομμάτια λογισμικού που απαρτίζουν το *SmartNet* είναι τα ακόλουθα:

- **MySQL Βάση Δεδομένων (Backend)** : Αποθηκεύει πληροφορίες για την δυναμική δημιουργία των ιστοσελίδων, για την κατάσταση των κινητών συσκευών, και πληροφορίες που σχετίζονται με την διαχείριση και την λειτουργία του *SmartNet* .
- **SmartNet Scheduler** : Πρόγραμμα πολυνηματικού πελάτη-εξυπηρετητή (*multithreaded client-server*) γραμμένο σε *Java* το οποίο είναι υπεύθυνο για την εκτέλεση ενεργειών στις κινητές συσκευές.
- **Android Debug Bridge (adb)** : Το εργαλείο το οποίο μας επιτρέπει επικοινωνία μεταξύ του εξυπηρετητή και των κινητών συσκευών *Android*.
- **Monkeyrunner** : Το εργαλείο το οποίο μας επιτρέπει την διαχείριση των κινητών συσκευών *Android*.
- **Web Server** : Παρέχει στον χρήστη την διαδικτυακή διαπροσωπεία για την δέσμευση, αποδέσμευση και διαχείριση των κινητών συσκευών καθώς επίσης και την μεταφορά αρχείων.

Στο Σχήμα 3.1 φαίνεται η επικοινωνία που υπάρχει μεταξύ των συστατικών λογισμικού που μόλις αναφέραμε. Στην παράγραφο αυτή ακολουθεί περιγραφή του υλικού που χρησιμοποιήθηκε για την υλοποίηση του *SmartNet*.

### 3.1.1 Υλικό που χρησιμοποιήθηκε

Για εξυπηρετητής χρησιμοποιήθηκε το *Lenovo ThinkPad T400* και παρουσιάζεται στον Πίνακα 3.1. Οι συσκευές που χρησιμοποιήθηκαν για την ανάπτυξη και έλεγχο του *SmartNet* είναι όπως προαναφέραμε *HTC Desire* και *HTC Hero* και διατίθενται προς κράτηση παρουσιάζονται στον Πίνακα 3.2.

## Lenovo ThinkPad T400 with Lenovo Docking Station 2504



### ThinkPad T400 Specs

- **Processor:** Intel Core 2 Duo T9600 (2.80GHz, 1066MHz FSB, 6MB Cache)
- **Memory:** 2GB DDR3 RAM
- **Storage:** 160GB HDD (7200rpm)
- **Operating System:** Ubuntu 10.10

### Docking Station Ports

- 4x Hi-Speed USB 2.0 ( 4 PIN USB Type A )
- 1x audio / video - DisplayPort - 20 pin DisplayPort
- 1x display / video - VGA - 15 pin HD D-Sub (HD-15)
- 1x display / video - DVI-Digital - 24 pin digital DVI
- 1x microphone - input - mini-phone 3.5mm
- 1x headphones - output - mini-phone stereo 3.5 mm
- 1x network - Ethernet 10Base-T/100Base-TX/1000Base-T - RJ-45

Πίνακας 3.1 : Το μηχάνημα που χρησιμοποιήθηκε ως εξυπηρετητής

HTC Desire	
 <p>The image shows the HTC Desire smartphone. The screen displays a digital clock showing 10:08 AM, the date Tuesday, February 16, and weather information for Barcelona: Mostly Sunny with a temperature of 12°C. Below the weather are icons for Messages, Mail, Internet, and Camera. At the bottom of the screen is a dock with a Phone icon and a plus sign. The physical buttons (power, volume, and navigation) are visible at the bottom of the device.</p>	<ul style="list-style-type: none"> <li>• <b>Processor:</b> 1 GHz Scorpion processor, Adreno 200 GPU, Qualcomm QSD8250 Snapdragon chipset</li> <li>• <b>Memory:</b> 576 MB RAM 512 MB ROM</li> <li>• <b>Storage:</b> microSD 4GB</li> <li>• <b>Operating System:</b> Android OS, v2.1 (Éclair)</li> </ul>
HTC Hero	
 <p>The image shows the HTC Hero smartphone. The screen displays two analog clock faces for New York and London. Below the clocks are icons for Messages, Mail, Internet, and Calendar. At the bottom of the screen is a dock with a Phone icon and a plus sign. The physical buttons (power, volume, and navigation) are visible at the bottom of the device.</p>	<ul style="list-style-type: none"> <li>• <b>Processor:</b> 528 MHz ARM 11 processor, Adreno 130 GPU, Qualcomm MSM7200A chipset</li> <li>• <b>Memory:</b> 288 MB RAM 512 MB ROM</li> <li>• <b>Storage:</b> microSD 4GB</li> <li>• <b>Operating System:</b> Android OS, v1.5 (Cupcake)</li> </ul>

Πίνακας 3.2 : Οι Συσκευές που χρησιμοποιήθηκαν

## 3.2 Συστατικά συστήματος

Στην παράγραφο αυτή θα παρουσιάσουμε πιο αναλυτικά τα συστατικά του συστήματος και του ρόλου που έχουν στο σύστημα.

### 3.2.1 MySQL Βάση Δεδομένων (Backend)

Η *MySQL* βάση δεδομένων χρησιμοποιείται για την αποθήκευση όλων των πληροφοριών που χρειάζεται το *SmartNet* για να δουλέψει. Συγκεκριμένα αποθηκεύει τις πληροφορίες του κάθε χρήστη ο οποίος εγγράφεται μέσω της Διαδικτυακής Διαπροσωπείας έτσι ώστε να υπάρχει ελεγχόμενη πρόσβαση στην υπηρεσία του *SmartNet*. Επίσης έχει αποθηκευμένες πληροφορίες που σχετίζονται με τις συσκευές που βρίσκονται ενωμένες στον εξυπηρετητή, όπως την ταυτότητα της συσκευής, το μοντέλο, τεχνικά χαρακτηριστικά και μια γενική περιγραφή. Επίσης κρατείται η διαθεσιμότητα της κάθε συσκευής. Με την δημιουργία μιας κράτησης συσκευής, αποθηκεύονται πληροφορίες όπως την μοναδική ταυτότητα του χρήστη που έκανε την κράτηση, η ταυτότητα της συσκευής την οποία επιθυμεί να κρατήσει αλλά και το διάστημα των ημερομηνιών που θα γίνει η κράτηση. Στην βάση υπάρχουν αρκετές αποθηκευμένες διαδικασίες (*stored procedures*) οι οποίες βοηθούν στην διαχείριση των δεδομένων κάνοντας έτσι πιο γρήγορη την λειτουργία της. Τα δεδομένα αυτά χρησιμοποιούνται για την δημιουργία της ιστοσελίδας δυναμικά παρέχοντας έτσι μια πιο πλούσια και διαδραστική διαδικτυακή διαπροσωπεία για τον χρήστη.

Συγκεκριμένα η Βάση Δεδομένων αποτελείται από τρεις πίνακες περιγραφή των οποίων παρουσιάζεται στον ακόλουθο πίνακα.

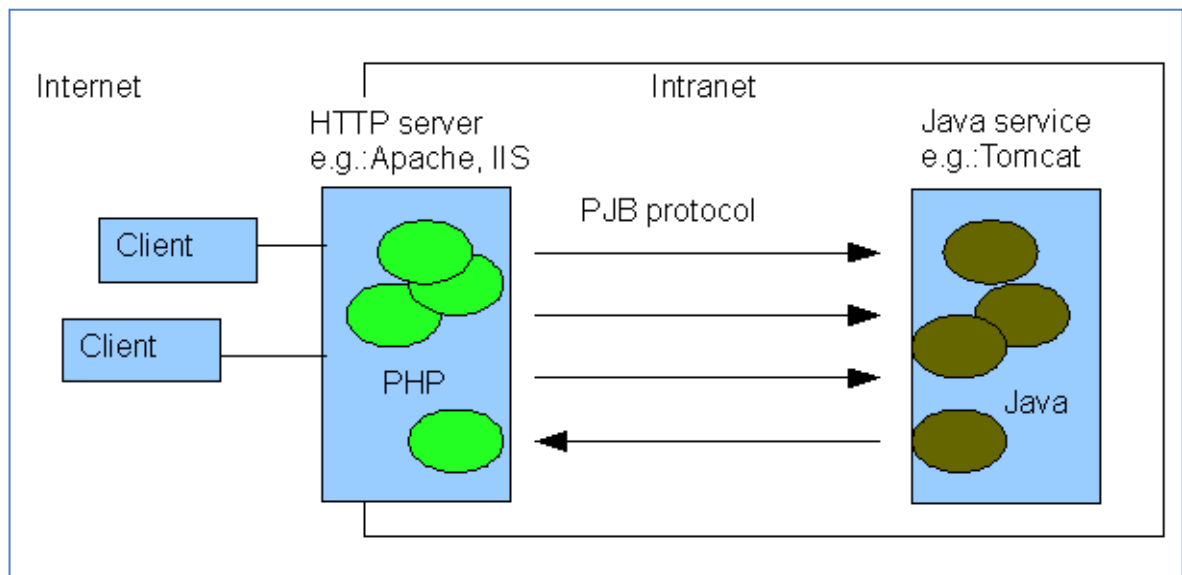
<b>Πίνακας Members</b>		
<b>Περιγραφή</b>	<b>Τύπος</b>	<b>Μέγεθος (σε bytes)</b>
1. Μοναδική Ταυτότητα Χρήστη	INTEGER	4
2. Όνομα Χρήστη	VARCHAR(100)	100
3. Επίθετο Χρήστη	VARCHAR(100)	100
4. Συνθηματικό Χρήστη	VARCHAR(100)	100
5. Κωδικός Εισόδου Χρήστη	VARCHAR(32)	32
6. Administrator	BOOLEAN	1
<b>Συνολικό Μέγεθος Εγγραφής:</b>		<b>337 bytes</b>
<b>Πίνακας Devices</b>		
1. Ταυτότητα Συσκευής	CHAR(12)	12
2. Διαθεσιμότητα Συσκευής	TINYINT(4)	1
3. Μοντέλο Συσκευής	VARCHAR(50)	50
4. Μνήμη RAM	CHAR(20)	20
5. Μνήμη ROM	CHAR(20)	20
6. Επεκτάσιμη Μνήμη	CHAR(80)	80
7. Συχνότητα CPU	CHAR (10)	10
8. Πλατφόρμα	CHAR (80)	80
9. Σένσορες που διαθέτει	VARCHAR(200)	200
<b>Συνολικό Μέγεθος Εγγραφής:</b>		<b>473 bytes</b>
<b>Πίνακας ReservedDevices</b>		
1. Ταυτότητα Κράτησης	INTEGER	4
2. Ταυτότητα Συσκευής	CHAR(12)	12
3. Ταυτότητα Χρήστη Κράτησης	INTEGER	4
4. Ημερομηνία Έναρξης Κράτησης	DATETIME	8
5. Ημερομηνία Λήξης Κράτησης	DATETIME	8
<b>Συνολικό Μέγεθος Εγγραφής:</b>		<b>36 bytes</b>
<b>Πίνακας Registration</b>		
1. Κωδικός Εγγραφής	VARCHAR(12)	12

2. Εγκυρότητα	TINYINT	1
Συνολικό Μέγεθος Εγγραφής:		13 bytes

**Πίνακας 3.3 : Μορφή Εγγραφών των Πινάκων της Βάσης Δεδομένων στο SmartNet**

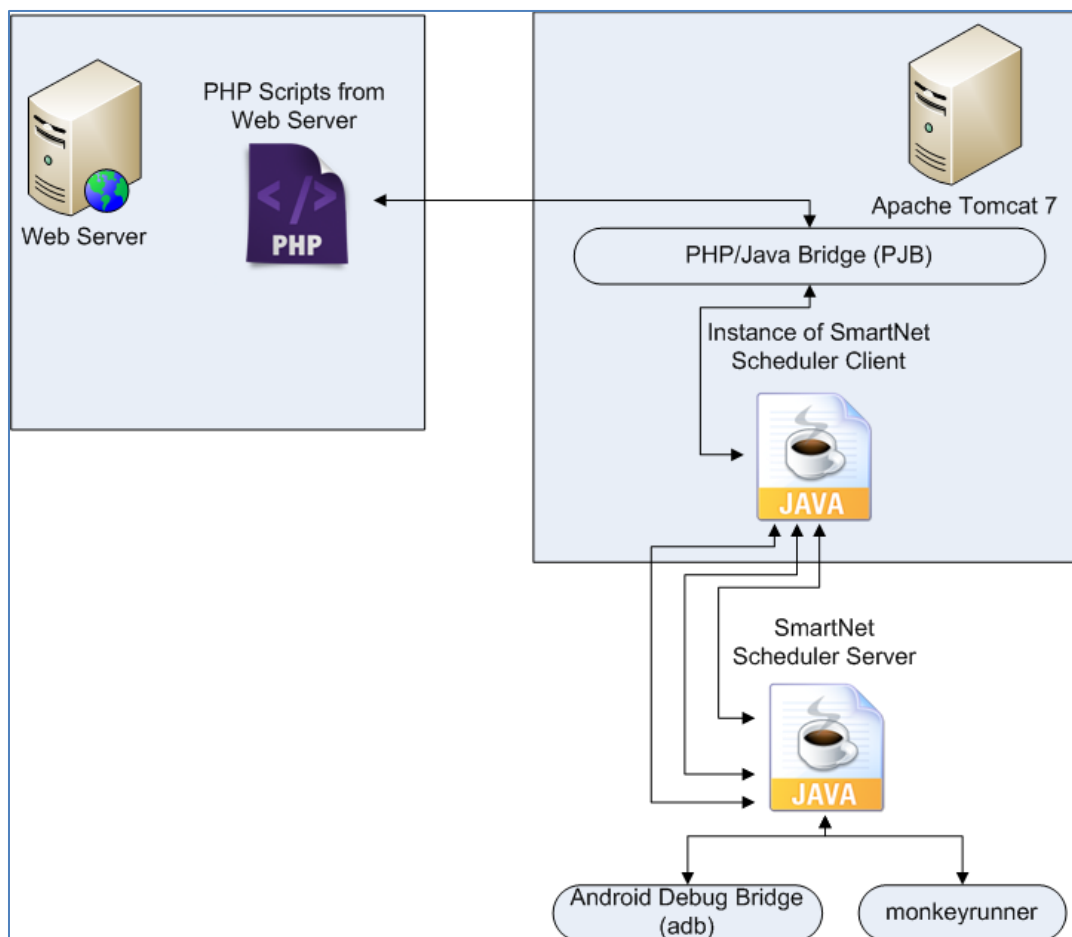
### 3.2.2 SmartNet Scheduler

Πρόγραμμα πολυνηματικού (*multithreaded*) *client-server* το οποίο υλοποιήσαμε στην *Java*. Είναι υπεύθυνο για την εκτέλεση ενεργειών στις κινητές συσκευές. Συγκεκριμένα μας επιτρέπει την επικοινωνία του *Web Server* μέσω *PHP scripts* με το *Android Debug Bridge*. Για να το πετύχουμε αυτό, το κομμάτι του πελάτη (*client*) τρέχει σαν *servlet* [40] σε ένα άλλο εξυπηρετητή *Apache Tomcat™ 7* [41]. Στον εν λόγω εξυπηρετητή υπάρχει επίσης σαν *servlet* εγκατεστημένη μια γέφυρα (*bridge*) επικοινωνίας μεταξύ *PHP* και *Java* (*PHP/Java Bridge* ή *PJB* [42]) όπως φαίνεται στο Σχήμα 3.2.



**Σχήμα 3.2 : Το PHP/Java Bridge επιτρέπει επικοινωνία της PHP με την Java**

Μέσω της γέφυρας αυτής τα *PHP scripts* τα οποία εκτελούνται όταν ο χρήστης θέλει να εκτελέσει μια ενέργεια στις κινητές συσκευές δημιουργούν στιγμιότυπα (*instances*) ενός πελάτη του *SmartNet Scheduler*. Ο πελάτης συνδέεται στον εξυπηρετητή του *SmartNet Scheduler* μεταφέροντας τις απαραίτητες πληροφορίες που χρειάζεται το *Android Debug Bridge* ή το *monkeyrunner* για να εκτελέσει την επιθυμητή ενέργεια στις επιλεγμένες κινητές συσκευές (Σχήμα 3.3). Στη συνέχεια οι αιτήσεις για εκτέλεση των συγκεκριμένων ενεργειών μπαίνουν σε μια *First In First Out (FIFO)* ουρά. Η ουρά αυτή καταναλώνεται από ένα ξεχωριστό νήμα το οποίο εκτελεί σειριακά έτσι τις επιθυμητές ενέργειες μέσω του *ADB/monkeyrunner*. Η σειριακή εκτέλεση των ενεργειών αποσκοπεί στην εκτέλεση μόνο ενός στιγμιότυπου του *ADB/monkeyrunner* αποτρέποντας έτσι κατακλυσμό τους εξυπηρετητή με αιτήσεις που θα οδηγούσαν σε άνοιγμα διαφορετικών στιγμιότυπων τους.



**Σχήμα 3.3 : Αρχιτεκτονική Όψη του SmartNet Scheduler**

### 3.2.3 *Android Debug Bridge (ADB)*

Το εργαλείο αυτό περιγράφεται στο **Κεφάλαιο 2 : Σχετική Δουλειά και Τεχνολογικό Υπόβαθρο** στην υποπαράγραφο **2.5.1**. Όπως φαίνεται στο *Σχήμα 3.1* και με μεγαλύτερη λεπτομέρεια στο *Σχήμα 3.3* το Android Debug Bridge είναι υπεύθυνο για την επικοινωνία των κινητών συσκευών *Android* και του *SmartNet Scheduler* τον οποίο περιγράψαμε στην προηγούμενη υποπαράγραφο.

### 3.2.4 *Monkeyrunner*

Το monkeyrunner είναι ένα εργαλείο το οποίο παρέχει ένα *API (Application Programming Interface)* για συγγραφή προγραμμάτων τα οποία ελέγχουν μια *Android* συσκευή ή προσομοιωτή. Μέσω του monkeyrunner ο χρήστης μπορεί να γράψει ένα πρόγραμμα στην γλώσσα προγραμματισμού ***Python*** και μέσω αυτού να τρέξει μια εφαρμογή και να στείλει τα πατήματα των πλήκτρων που επιθυμεί έτσι ώστε να εκτελέσει μια συγκεκριμένη ενέργεια στην εν λόγω εφαρμογή. Με αυτό τον τρόπο το *SmartNet* παρέχει την δυνατότητα εκτέλεσης εφαρμογών που έχει ήδη εγκαταστήσει ο χρήστης για την εκτέλεση δοκιμών που επιθυμεί. Περιγραφή της ακριβής λειτουργίας του στο *SmartNet* όπως και παράδειγμα κώδικα παρατίθεται στο **Κεφάλαιο 4 : Το Σύστημα Διαπροσωπείας του SmartNet** στην υποπαράγραφο **4.5**.

### 3.2.5 *Web Server*

Παρέχει στον χρήστη την διαδικτυακή διαπροσωπεία για την δέσμευση, αποδέσμευση και διαχείριση των κινητών συσκευών καθώς επίσης και την μεταφορά αρχείων. Το *SmartNet* χρησιμοποιεί την *PHP*, για να παρέχει δυναμικό περιεχόμενο στην ιστοσελίδα και την *JavaScript* και *jQuery* δίνοντας στον χρήστη μια πιο διαδραστική εμπειρία, εμπλουτίζοντας ταυτόχρονα την διαπροσωπεία που παρέχετε στον χρήστη. Αυτό επιτρέπει στον χρήστη να αποκτήσει πρόσβαση στο *SmartNet* ανεξαρτήτως της πλατφόρμας που χρησιμοποιεί. Μετά την εγγραφή και είσοδο, ο χρήστης έχει πρόσβαση στις πιο κάτω λειτουργίες:



- **Οικοσελίδα (Homepage)** : Παρουσιάζει πληροφορίες σχετικά με ήδη υπάρχον κρατήσεις που έχουν γίνει από τον χρήστη.
- **Δημιουργία Κράτησης** : Ο χρήστης μπορεί να επιλέξει ανάμεσα σε δύο τρόπους για να κάνει κράτηση των συσκευών. Μπορεί να επιλέξει να κάνει μελλοντική κράτηση σε αριθμό συσκευών ή να κάνει κράτηση την παρούσα στιγμή. Εάν επιλέξει να κάνει κράτηση την παρούσα στιγμή η κράτηση ισχύει μέχρι το τέλος της συγκεκριμένης μέρας. Σε περίπτωση που επιλέξει να κάνει κράτηση για μελλοντική χρήση επιλέγει από μια αυτοματοποιημένη διαδικασία κατά την οποία το παρέχονται οδηγίες τις ημερομηνίες που επιθυμεί να κάνει την κράτηση αλλά και τις συσκευές που επιθυμεί.
- **Διαχείριση Συσκευών** : Ο χρήστης μπορεί να επιλέξει συσκευές στις οποίες έχει ενεργή κράτηση και να εκτελέσει μια σειρά ενεργειών σε αυτές, όπως μεταφορά αρχείων, εγκατάσταση προγραμμάτων και απεγκατάσταση προγραμμάτων, απελευθέρωση των συσκευών, λήψη στιγμιότυπου οθόνης των συσκευών και εκτέλεση προγραμμάτων.
- **Οδηγίες**: Παρέχει στον χρήστη οδηγίες σχετικά με το πώς μπορεί να τρέξει το δικό του πρόγραμμα στις συσκευές. Επίσης των ενημερώνει για ο πού μπορεί να εντοπίσει τα στιγμιότυπα οθόνης των συσκευών που έχει βγάλει.
- **Διαχειριστής Αρχείων** : Ο χρήστης μπορεί να διαχειριστεί αρχεία στον εξυπηρετητή, σε ένα δικό του προσωπικό χώρο.

# Κεφάλαιο 4

## Το Σύστημα Διαπροσωπείας του SmartNet

---

4.1 Εγγραφή στο <i>SmartNet</i>	36
4.2 Εισαγωγή στο <i>SmartNet</i>	37
4.3 Κράτηση Συσκευών	38
4.4 Χρονοπρογραμματισμός Κράτησης Συσκευών	40
4.5 Διαχείριση Συσκευών	41
4.6 Οδηγίες	47
4.7 Διαχειριστής Αρχείων	47
4.8 Γενικά Χαρακτηριστικά	49
4.9 Παραγωγή Κλειδιών Εγγραφής ( Admin )	50

---

Στο προηγούμενο κεφάλαιο έγινε επισκόπηση της αρχιτεκτονικής του *SmartNet* με μια σύντομη περιγραφή για το ρόλο του κάθε συστατικού ακολουθούμενη από μια πιο λεπτομερή ανάλυση των επιμέρους συστατικών του. Σε αυτό το κεφάλαιο θα παρουσιαστεί το Σύστημα Διαπροσωπείας του *SmartNet* επεξηγώντας τις δυνατότητες που παρέχει αλλά και τον τρόπο λειτουργίας του.

#### 4.1 Εγγραφή στο SmartNet



### SmartNet Testbench Registration

First Name	<input type="text" value="Demo"/>	1
Last Name	<input type="text" value="User"/>	2
Login	<input type="text" value="demo"/>	3
Password	<input type="password" value="...."/>	4
Confirm Password	<input type="password" value="...."/>	5
Registration Code	<input type="text" value="t6fzq21q3zct"/>	6

Σχήμα 4.1 : Εγγραφή στο *SmartNet*

Στο πιο πάνω σχήμα (Σχήμα 4.1) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για να εγγραφεί στο σύστημα. Με την εγγραφή του χρήστη δημιουργείται προσωπικός χώρος για την διαχείριση αρχείων στον εξυπηρετητή και τα στοιχεία του αποθηκεύονται στην βάση για την διαδικασία της εισαγωγής του στο σύστημα. Στο σχήμα υπάρχουν περιοχές που έχουν απαριθμηθεί (από το 1 μέχρι και το 5). Ακολουθεί περιγραφή των αριθμημένων περιοχών καθώς και σύντομη επεξήγησή τους:

1. **Όνομα** : Το όνομα του χρήστη.
2. **Επίθετο** : Το επίθετο του χρήστη.
3. **Συνθηματικό Είσοδου** : Το συνθηματικό που θα χρησιμοποιεί ο χρήστης για την είσοδό του στο σύστημα.
4. **Κωδικός** : Ο κωδικός που θα χρησιμοποιεί ο χρήστης για την είσοδό του στο σύστημα.

5. **Επιβεβαίωση κωδικού** : Ο κωδικός που θα χρησιμοποιεί ο χρήστης για την είσοδό του στο σύστημα.
6. **Κωδικός Εγγραφής** : Ο κωδικός που πρέπει να δώσει ο χρήστης για να εγγραφεί στο σύστημα. Τον κωδικό αυτό πρέπει να τον ζητήσει και να τον παραλάβει από κάποιο administrator έτσι ώστε να εγγραφεί. Ο εν λόγω κωδικός ισχύει μόνο για μια εγγραφή, μετά την χρήση του ακυρώνεται.

#### 4.2 Εισαγωγή στο SmartNet



**Σχήμα 4.2 : Εισαγωγή στο SmartNet**

Στο πιο πάνω σχήμα (Σχήμα 4.2) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για την εισαγωγή του στο σύστημα, αφού περάσει από την διαδικασία της εγγραφής. Τα στοιχεία που δίνει ελέγχονται μέσω της *MySQL* Βάσης Δεδομένων για την εγκυρότητά τους δυναμικά. Σε περίπτωση που ο χρήστης αποτύχει να

δώσει ορθά συνθηματικά για την είσοδό του ενημερώνεται με κατάλληλο μήνυμα και καλείται να εισάγει τα ορθά συνθηματικά ή να εγγραφεί στο σύστημα εάν δεν το έχει πράξει ήδη.

### 4.3 Κράτηση Συσκευών

The screenshot displays the 'SmartNet Demo' interface. At the top, there is a navigation bar with tabs: 'Device Reservation', 'Schedule Device Reservation', 'Device Management', 'Instructions', 'File Manager', and 'Logout'. Below the navigation bar, a green banner reads 'Welcome to SmartNet Demo'. A button labeled 'Make a new Reservation' is visible. The main content area is titled 'Available Devices now!' and lists two device models:

- HTC Desire 2** (labeled 1):
  - Storage (labeled 3):
    - RAM: 576 MB
    - ROM: 512 MB
    - Expansion Slot: microSD memory card (SD 2.0 compatible)
  - CPU Processing Speed: 1 GHz (labeled 4)
  - Platform: Android 2.1 (Eclair) with HTC Sense
  - Sensors: \* G-Sensor \* Digital compass \* Proximity sensor \* Ambient light sensor (labeled 5)
  - Available HTC Desire Devices now: 2 (labeled 6)
  - Devices to Reserve: 0 (labeled 7)
- HTC Hero** (labeled 2):
  - Storage:
    - RAM: 288 MB
    - ROM: 512 MB
    - Expansion Slot: microSD memory card (SD 2.0 compatible)
  - CPU Processing Speed: 528 MHz
  - Platform: Android 2.1 (Eclair) with HTC Sense
  - Sensors: \* G-sensor \* Digital Compass
  - Available HTC Hero Devices now: 1
  - Devices to Reserve: 0

At the bottom right, there is a 'Reserve now' button (labeled 8).

Σχήμα 4.3 : Κράτηση Συσκευών στο *SmartNet*

Στο πιο πάνω σχήμα (Σχήμα 4.3) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για την άμεση κράτηση συσκευών. Κατά την εισαγωγή του στο σύστημα, δυναμικά μέσω της *PHP* το σύστημα αντλεί πληροφορίες από την Βάση Δεδομένων για τις διαθέσιμες συσκευές την συγκεκριμένη στιγμή και τις παρουσιάζει στον χρήστη δίνοντας του την δυνατότητα να επιλέξει το είδος και τον αριθμό των συσκευών που θέλει να κρατήσει. Στο σχήμα υπάρχουν περιοχές που έχουν απαριθμηθεί (από το 1

μέχρι και το 8). Ακολουθεί περιγραφή των αριθμημένων περιοχών καθώς και η σκοπιμότητά τους:

- 1. Φωτογραφία Συσκευής :** Φωτογραφία της διαθέσιμης συσκευής δίνοντας στον χρήστη να καταλάβει αμέσως πια συσκευή παρουσιάζεται.
- 2. Όνομα Συσκευής :** Το όνομα του μοντέλου της συσκευής.
- 3. Χωρητικότητα :** Πληροφορίες που αφορούν την χωρητικότητα της συσκευής. Συγκεκριμένα παρουσιάζεται η χωρητικότητα της Μνήμης Τυχαίας Προσπέλασης (*RAM*), της Μνήμης Μόνο Διαβάσματος (*ROM*) και της μνήμης που παρέχεται από την θέση επέκτασης. Έτσι δίνεται στον χρήστη η δυνατότητα να επιλέξει συσκευές που καλύπτουν τις ανάγκες του.
- 4. Ταχύτητα της Κεντρικής Μονάδας Επεξεργασίας :** Η ταχύτητα της Κεντρικής Μονάδας Επεξεργασίας (*CPU*) της συσκευής. Αποσκοπεί και πάλι στην παροχή της δυνατότητας επιλογής συσκευής από τον χρήστη η οποία να καλύπτει τις ανάγκες του.
- 5. Πλατφόρμα και Αισθητήρες :** Το είδος της πλατφόρμας που τρέχει η συσκευή και οι αισθητήρες που διαθέτει. Αποσκοπεί και πάλι σε ότι αποσκοπεί το σημείο 3 και 4.
- 6. Διαθέσιμες Συσκευές :** Παρουσιάζει τον αριθμό το διαθέσιμων συσκευών την δεδομένη στιγμή που μπορεί να κρατήσει ο χρήστης.
- 7. Αριθμός Συσκευών για Κράτηση :** Μέσω ενός *drop down* μενού ο χρήστης μπορεί να επιλέξει τον αριθμό των συσκευών που θέλει να κρατήσει.
- 8. Κράτηση :** Εκτελεί την επιθυμητή κράτηση. Σε περίπτωση που για οποιοδήποτε λόγο αποτύχει ο χρήστης ενημερώνεται με κατάλληλο μήνυμα, όπως και στην περίπτωση επιτυχούς κράτησης.

#### 4.4 Χρονοπρογραμματισμός Κράτησης Συσκευών



**Σχήμα 4.4 : Χρονοπρογραμματισμός Κράτησης Συσκευών στο *SmartNet***

Στο πιο πάνω σχήμα (Σχήμα 4.4) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για χρονοπρογραμματισμό κράτησης συσκευών. Ο χρήστης έχει την δυνατότητα μέσω ενός διαδραστικού ημερολογίου να επιλέξει το διάστημα των ημερομηνιών κατά τις οποίες επιθυμεί να κάνει μια κράτηση. Σε περίπτωση που επιλέξει μη έγκυρο διάστημα του ζητείται να επιλέξει ξανά. Στην συνέχεια επιλέγει το “*Get Available Devices*” και του επιστρέφεται μια λίστα με τις διαθέσιμες συσκευές στο διάστημα που έχει επιλέξει. Τέλος επιλέγει τις συσκευές που θέλει να κρατήσει και εκτελεί την κράτηση. Οι συγκεκριμένες συσκευές δεν παρουσιάζονται σαν ενεργές κρατήσεις στον πίνακα ενεργών κρατήσεων (τον οποίο θα δούμε πιο κάτω) του χρήστη και μπορούν να κρατηθούν από άλλους χρήστες στο διάστημα που δεν υπάρχει χρονοπρογραμματισμένη κράτηση.

## 4.5 Διαχείριση Συσκευών

# SMARTNET

**Scheduled Reservations: 2**

Id	From Date	To Date	Model	Device Id
220	2011-05-21 00:00:00	2011-05-21 23:59:59	HTC Desire	SH0APPL01201
219	2011-05-21 00:00:00	2011-05-21 23:59:59	HTC Desire	SH0APPL01284

**Active Reservations**

Res. Id	Device Id	Model	Ram	Rom	CPU	Platform
228	SH0APPL01284	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
229	SH0APPL01201	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense

Σχήμα 4.5 : Διαχείριση Συσκευών στο *SmartNet*

Στο Σχήμα 4.5 παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για την διαχείριση των συσκευών που έχει κρατήσει. Μέσω της διαπροσωπείας αυτής μπορεί ασύγχρονα να εκτελέσει ενέργειες ενώ τα δεδομένα ανανεώνονται δυναμικά. Στο σχήμα υπάρχουν περιοχές που έχουν απαριθμηθεί (από το 1 μέχρι και το 11). Ακολουθεί περιγραφή των αριθμημένων περιοχών καθώς και η λειτουργία τους:

- 1. Χρονοπρογραμματισμένες Κρατήσεις :** Σε αυτή την περιοχή παρουσιάζονται πληροφορίες σχετικά με κρατήσεις τις οποίες έχει κάνει ο χρήστης χρησιμοποιώντας την λειτουργία που αναφέραμε στην υποπαράγραφο 4.4 και δεν είναι ακόμα ενεργές.
- 2. Ανέβασμα Αρχείου :** Παρέχει την δυνατότητα στον χρήστη να ανεβάσει ασύγχρονα μέσω του *jQuery* αρχεία. Τα αρχεία ανεβαίνουν όχι μόνο στις επιλεγμένες κινητές συσκευές αλλά και στον προσωπικό του χώρο πάνω στον οποίο έχει πρόσβαση μέσω του Διαχειριστή Αρχείων τον οποίο θα δούμε αργότερα. Στον χρήστη επιτρέπεται το ανέβασμα όλου των τύπων αρχείων. Ο χρήστης



ενημερώνεται στο τέλος της ενέργειας για το αποτέλεσμα της (σε περίπτωση αποτυχίας αλλά και επιτυχίας).

- 3. Εγκατάσταση αρχείου :** Ο χρήστης έχει την δυνατότητα να εγκαταστήσει στις επιλεγμένες συσκευές αρχεία τύπου **.apk** (*Android Package*), εξοικονομώντας έτσι χρόνο και κόπο. Αφού κάνει την συγκεκριμένη επιλογή, ένας *file browser* ο οποίος του επιτρέπει την επιλογή μόνον αρχείων με επέκταση *.apk* ανοίγει. Τα αρχεία τα οποία εγκαθίστανται στις συσκευές ανεβάζονται επίσης στον προσωπικό χώρο του χρήστη για τυχόν μεταγενέστερη χρήση. Στο τέλος της ενέργειας ο χρήστης ενημερώνεται για το αποτέλεσμα της ενέργειας με κατάλληλο μήνυμα όπως φαίνεται στο *Σχήμα 4.6*.



The screenshot displays the SmartNet web interface. At the top right, a notification box states "File installed successfully...". The main navigation bar includes "Device Reservation", "Schedule Device Reservation", "Device Management", "Instructions", "File Manager", and "Logout". The "Device Management" section is active, showing "SmartNet: Device Management" and "Scheduled Reservations: 2". A table lists two reservations:

Id	From Date	To Date	Model	Device Id
220	2011-05-21 00:00:00	2011-05-21 23:59:59	HTC Desire	SH0APPL01201
219	2011-05-21 00:00:00	2011-05-21 23:59:59	HTC Desire	SH0APPL01284

Below the table are buttons for "Upload File", "Install File", "Uninstall File", "Take Screenshot", "Release Device", "Run Program", "Show Files", "Get File", "Reboot Device", and "Show Screen". An "Active Reservations" table is also visible:

Res. Id	Device Id	Model	Ram	Rom	CPU	Platform	
<input checked="" type="checkbox"/>	228	SH0APPL01284	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
<input checked="" type="checkbox"/>	229	SH0APPL01201	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
<input type="checkbox"/>	230	HT99TL902572	HTC Hero	288 MB	512 MB	528 MHz	Android 1.5 (Cupcake)

Page 1 of 3 is shown at the bottom.

**Σχήμα 4.6 :** Εγκατάσταση αρχείου στο *SmartNet*

- 4. Απεγκατάσταση αρχείου :** Ο χρήστης έχει την δυνατότητα να απεγκαταστήσει αρχεία από τις επιλεγμένες συσκευές. Αφού επιλέξει την απεγκατάσταση του ζητείται να εισάγει το όνομα του πακέτου που θέλει να απεγκαταστήσει. Ο χρήστης ενημερώνεται κατάλληλα σε περίπτωση σφάλματος αλλά και επιτυχίας με το πέρας της ενέργειας.
- 5. Λήψη Στιγμιότυπου Οθόνης :** Ο χρήστης μπορεί ανά πάσα στιγμή να λάβει στιγμιότυπο της οθόνης των συσκευών που επιθυμεί. Τα στιγμιότυπα αποθηκεύονται και μπορούν να ανακτηθούν από τον προσωπικό χώρο του χρήστη

μέσω του **Διαχειριστή Αρχείων**. Με αυτό τον τρόπο ο χρήστης θα μπορούσε για παράδειγμα να πάρει τα αποτελέσματα μετά από το τέλος της εκτέλεσης ενός προγράμματος, εάν το πρόγραμμα παρουσιάζει τα αποτελέσματα στην οθόνη ή να ελέγξει την κατάσταση ενός προγράμματος.

6. **Απελευθέρωση συσκευής** : Δίνει στον χρήστη την δυνατότητα να απελευθερώσει μια συσκευή την οποία δεν χρειάζεται πλέον καθιστώντας την έτσι διαθέσιμη για τους υπόλοιπους χρήστες. Η συσκευή αμέσως αφαιρείται από τον πίνακα με τις ενεργές κρατήσεις του χρήστη και δεν μπορεί πλέον να την διαχειριστεί.
7. **Εκτέλεση προγράμματος** : Παρέχει στον χρήστη την δυνατότητα εκτέλεσης κάποιας εφαρμογής που έχει ήδη εγκαταστήσει στις συσκευές. Συγκεκριμένα μέσω του *monkeyrunner* [42] μπορεί να ετοιμάσει ένα εξειδικευμένο αρχείο για την εφαρμογή του το οποίο περιέχει τις ενέργειες που θέλει να εκτελέσει στην εφαρμογή του. Το αρχείο αυτό είναι γραμμένο σε γλώσσα προγραμματισμού *Python*. Στον **Διαχειριστή Αρχείων** κάθε χρήστη υπάρχει ένα πρότυπο (*template*) τέτοιου αρχείου με το όνομα “*template.py*” (Σχήμα 4.7) το οποίο ο χρήστης καλείται να ανακτήσει και να τροποποιήσει έτσι ώστε να ανταποκρίνεται στην δική του εφαρμογή. Το αρχείο αυτό πρέπει να ανεβαστεί από τον χρήστη μέσω του **Διαχειριστή Αρχείου**. Στην συνέχεια εκτελώντας την ενέργεια “*Run Program*” από το μενού με τις ενέργειες του *SmartNet* ζητείται από τον χρήστη να εισάγει το όνομα του αρχείου που δημιούργησε έτσι ώστε να εκτελεστεί.

```

1 #####
2 # DO NO EDIT FROM THIS POINT
3 #####
4
5 # Imports the monkeyrunner modules used by this program
6 from com.android.monkeyrunner import MonkeyRunner, MonkeyDevice
7 import optparse
8 parser = optparse.OptionParser("")
9 parser.add_option("-d", "--device", dest="device", type="string")
10
11 (options, args) = parser.parse_args()
12 device = options.device
13
14
15 # Connects to the current device, returning a MonkeyDevice object
16 device = MonkeyRunner.waitForConnection(10000, device)
17
18
19 #####
20 # WRITE YOUR CODE BELOW THIS POINT
21 #####
22
23
24 # Sets a variable with the package's internal name (EDIT THIS)
25 package =          # eg. package = 'cs.ucy.ac.cy'
26
27 # Sets a variable with the name of an Activity in the package (EDIT THIS)
28 activity =         # eg. activity = 'cs.ucy.ac.cy.SmartTraceThreaded'
29
30 # Sets the name of the component to start (DO NOT EDIT THIS)
31 runComponent = package + '/' + activity
32
33 # Runs the component (DO NOT EDIT THIS)
34 device.startActivity(component=runComponent)
35
36
37 # WRITE THE KEY EVENTS THAT YOU WANT TO EXECUTE ON YOUR APPLICATION
38 # e.g. device.press('DPAD_DOWN', 'DOWN_AND_UP')
39
40
41
42
43
44 #####

```

Σχήμα 4.7 : Το αρχείο “*template.py*”

- 8. Προβολή Αρχείων :** Ο χρήστης μπορεί να δει τα αρχεία που υπάρχουν στον προσωπικό του χώρο πάνω στις συσκευές. Πατώντας στην προβολή αρχείων δημιουργείται ένα αρχείο με το όνομα “ListingOfFiles\_xx.xx.xxxx-xx.xx.xx\_DevId-xxxxxxxxxxxxxxxx.txt” στον **Διαχειριστή Αρχείων (File Manager)** του, όπου παρουσιάζει τα αρχεία που βρίσκονται στον προσωπικό του χώρο στην επιλεγμένη συσκευή. Χρησιμοποιείται σε συνδυασμό με την επόμενη λειτουργία “**Λήψη Αρχείου**”
- 9. Λήψη Αρχείου :** Ο χρήστης μπορεί να λάβει ένα αρχείο από την επιλεγμένη συσκευή. Πατώντας στην Λήψη Αρχείου ο χρήστης καλείται να εισάγει το όνομα του αρχείου (το οποίο μπορεί να πάρει με την **Προβολή Αρχείων**) το οποίο θέλει να πάρει από την συσκευή . Εάν το αρχείο υπάρχει στην εν λόγω συσκευή το αρχείο τοποθετείται στον προσωπικό του χώρο στον εξυπηρετητή και έχει πρόσβαση σε αυτό μέσω του **Διαχειριστή Αρχείων**.
- 10. Επανεκκίνηση Συσκευής :** Ο χρήστης μπορεί να επανεκκινήσει μια ή περισσότερες συσκευές. Οποιοσδήποτε ενέργειες γίνονται στις συσκευές την συγκεκριμένη στιγμή καθυστερούν την εκτέλεσή τους μέχρι να ξεκινήσουν οι συσκευές.
- 11. Παρουσίαση Οθόνης :** Ο χρήστης μπορεί να δει τις οθόνες των επιλεγμένων συσκευών σε νέα παράθυρα. Η οθόνη ανανεώνεται κάθε τριάντα δευτερόλεπτα.
- 12. Επιλογή Συσκευών :** Ο χρήστης μπορεί να επιλέξει σε ποιές συσκευές θέλει να εκτελέσει συγκεκριμένες ενέργειες από τον πίνακα με τις ενεργές κρατήσεις. Εάν δεν επιλέξει καμία συσκευή και προσπαθήσει να εκτελέσει οποιαδήποτε ενέργεια παίρνει ανάλογο μήνυμα λάθους.
- 13. Πληροφορίες Ενεργών Κρατήσεων.** Στην περιοχή αυτή παρουσιάζονται πληροφορίες για την κάθε ενεργή κράτηση που έχει εκτελέσει ο χρήστης. Συγκεκριμένα παρουσιάζεται η μοναδική ταυτότητα κράτησης, η μοναδική ταυτότητα της συσκευής, το μοντέλο της συσκευής και τεχνικά χαρακτηριστικά της, όπως η *RAM*, *ROM*, *CPU* αλλά και η πλατφόρμα στην οποία τρέχει η συσκευή. Κάθε στήλη του πίνακα ενεργών κρατήσεων μπορεί να χρησιμοποιηθεί για να ταξινομηθεί τόσο σε φθίνουσα όσο και αύξουσα σειρά ολόκληρος ο πίνακας ενεργών κρατήσεων επιλέγοντας το όνομα της στήλης. Επίσης στον χρήστη

παρέχεται η δυνατότητα αναζήτησης και φιλτραρίσματος του πίνακα με οποιοδήποτε πεδίο του χρησιμοποιώντας λογικούς τελεστές όπως φαίνεται στο Σχήμα 4.8.

# SMARTNET

The screenshot shows the 'SmartNet: Device Management' interface. At the top, there are navigation tabs: 'Device Reservation', 'Schedule Device Reservation', 'Device Management' (highlighted), 'Instructions', 'File Manager', and 'Logout'. Below the tabs, the page title 'SmartNet: Device Management' is displayed. Underneath, it says 'Scheduled Reservations: 2'. A table lists two reservations with columns for Id, From Date, To Date, Model, and Device Id. Below this, there is a row of action buttons: 'Upload File', 'Install File', 'Uninstall File', 'Take Screenshot', 'Release Device', 'Run Program', 'Show Files', 'Get File', 'Reboot Device', and 'Show Screen'. The main section is titled 'Active Reservations' and contains a table with columns: Res. Id, Device Id, Model, Ram, Rom, CPU, and Platform. A search dropdown menu is open over the table, showing a search box with 'greater' entered and a 'Find' button. The dropdown menu lists the following fields: Res. Id, Device Id, Model, Ram (highlighted), Rom, CPU, and Platform. The table shows three rows of data, with the first two rows selected.

Res. Id	Device Id	Model	Ram	Rom	CPU	Platform
228	SH0APPL01284	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
229	SH0APPL01201	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
230	HT99TL902572	HTC Hero	288 MB	512 MB	528 MHz	Android 1.5 (Cupcake)

**Σχήμα 4.8 : Αναζήτηση και φιλτράρισμα στον Πίνακα Ενεργών Κρατήσεων στο SmartNet**

Εάν ο χρήστης είναι administrator έχει την επιλογή να διαχειριστεί όλες τις συσκευές και να δει ποιες είναι διαθέσιμες και ποιες όχι όπως φαίνεται στο Σχήμα 4.9 .

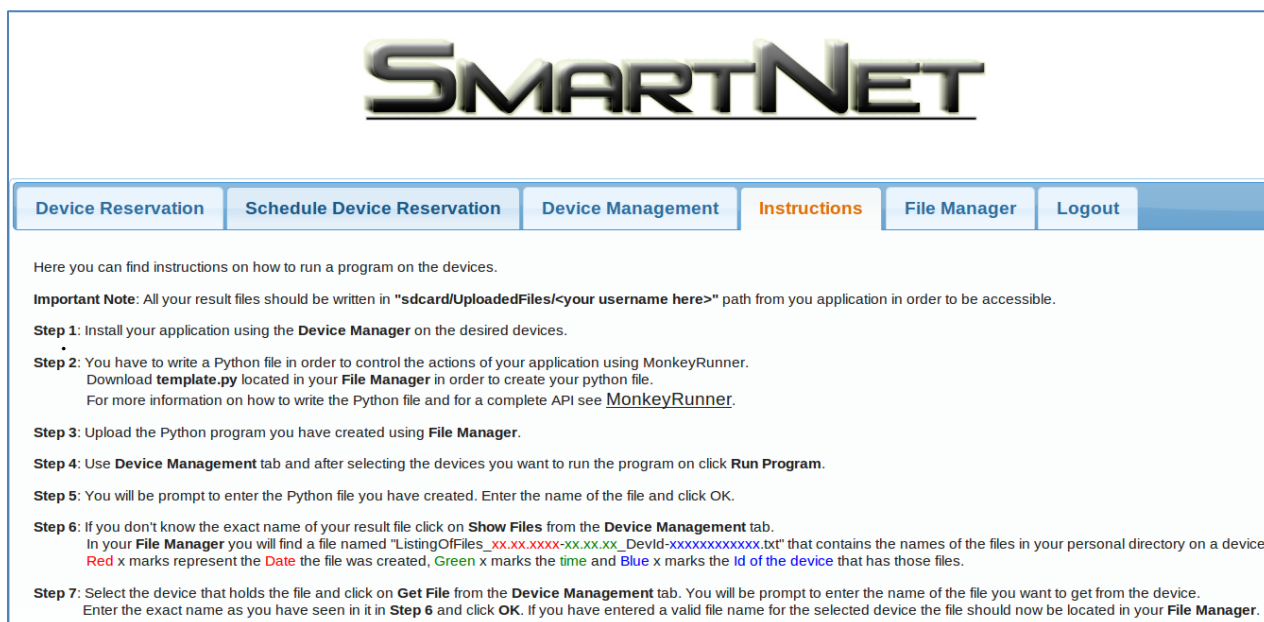
The screenshot shows the 'SmartNet: Device Management' interface. At the top, there are navigation tabs: 'Device Reservation', 'Schedule Device Reservation', 'Device Management' (highlighted), 'Instructions', 'File Manager', and 'Logout'. Below the tabs, the page title 'SmartNet: Device Management' is displayed. Underneath, it says 'Scheduled Reservations: 0'. A row of action buttons is visible: 'Upload File', 'Install File', 'Uninstall File', 'Take Screenshot', 'Run Program', 'Show Files', 'Get File', 'Reboot Device', and 'Show Screen'. The main section is titled 'All Devices: Admin Mode' and contains a table with columns: Available Now, Device Id, Model, Ram, Rom, CPU, and Platform. The table shows three rows of data. Below the table, there is a pagination control showing 'Page 1 of 0'.

Available Now	Device Id	Model	Ram	Rom	CPU	Platform
0	SH0APPL01284	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
0	SH0APPL01201	HTC Desire	576 MB	512 MB	1 GHz	Android 2.1 (Eclair) with HTC Sense
1	HT99TL902572	HTC Hero	288 MB	512 MB	528 MHz	Android 1.5 (Cupcake)

**Σχήμα 4.9 : Διαχείριση Συσκευών από Administrator στο SmartNet**

## 4.6 Οδηγίες

Παρέχει οδηγίες σχετικά με το πώς μπορεί ένας χρήστης να τρέξει μια δική του εφαρμογή στο *SmartNet* όπως φαίνεται στο *Σχήμα 4.10*.



The screenshot shows the SmartNet web interface. At the top, the 'SMARTNET' logo is displayed. Below the logo is a navigation menu with tabs for 'Device Reservation', 'Schedule Device Reservation', 'Device Management', 'Instructions' (highlighted in orange), 'File Manager', and 'Logout'. The main content area contains the following text:

Here you can find instructions on how to run a program on the devices.

**Important Note:** All your result files should be written in "sdcard/UploadedFiles/<your username here>" path from you application in order to be accessible.

**Step 1:** Install your application using the **Device Manager** on the desired devices.

**Step 2:** You have to write a Python file in order to control the actions of your application using MonkeyRunner. Download **template.py** located in your **File Manager** in order to create your python file. For more information on how to write the Python file and for a complete API see [MonkeyRunner](#).

**Step 3:** Upload the Python program you have created using **File Manager**.

**Step 4:** Use **Device Management** tab and after selecting the devices you want to run the program on click **Run Program**.

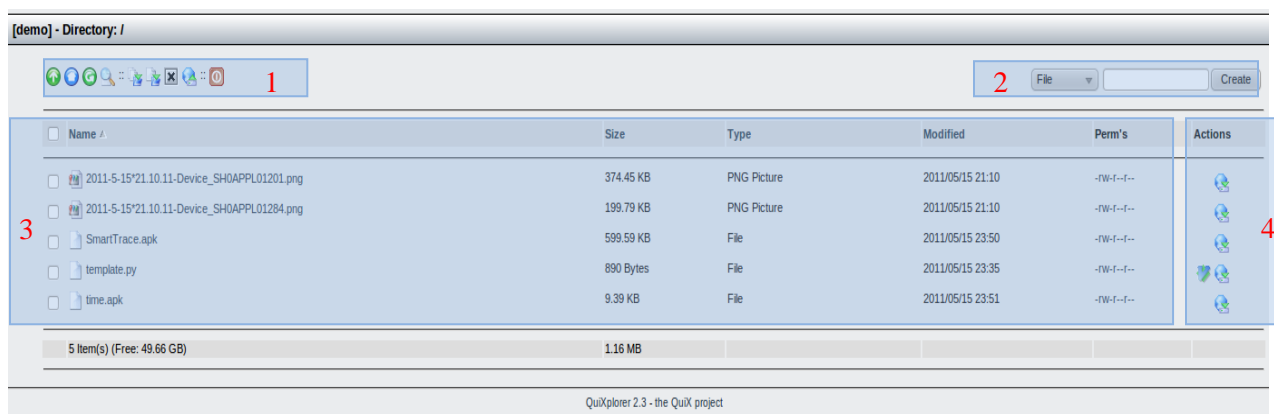
**Step 5:** You will be prompt to enter the Python file you have created. Enter the name of the file and click OK.

**Step 6:** If you don't know the exact name of your result file click on **Show Files** from the **Device Management** tab. In your **File Manager** you will find a file named "ListingOfFiles\_xx.xx.xxx-xx.xx.xx\_Devid-xxxxxxxxxxxx.txt" that contains the names of the files in your personal directory on a device. **Red x** marks represent the **Date** the file was created, **Green x** marks the **time** and **Blue x** marks the **Id of the device** that has those files.

**Step 7:** Select the device that holds the file and click on **Get File** from the **Device Management** tab. You will be prompt to enter the name of the file you want to get from the device. Enter the exact name as you have seen in it in **Step 6** and click **OK**. If you have entered a valid file name for the selected device the file should now be located in your **File Manager**.

Σχήμα 4.10 : Οδηγίες για εκτέλεση εφαρμογής στο *SmartNet*

## 4.7 Διαχειριστής Αρχείων



The screenshot shows the SmartNet File Manager interface. At the top, there is a toolbar with various icons and a 'File' dropdown menu. Below the toolbar is a table listing files in a directory. The table has columns for Name, Size, Type, Modified, Perm's, and Actions. The files listed are:

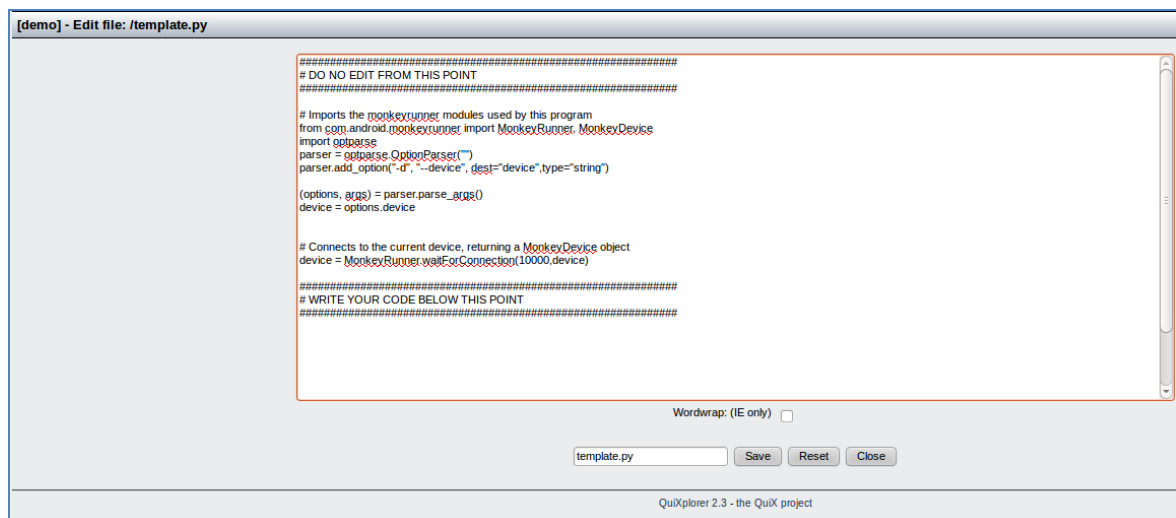
Name	Size	Type	Modified	Perm's	Actions
2011-5-15*21.10.11-Device_SH0APPL01201.png	374.45 KB	PNG Picture	2011/05/15 21:10	-rw-r--r--	[Actions]
2011-5-15*21.10.11-Device_SH0APPL01284.png	199.79 KB	PNG Picture	2011/05/15 21:10	-rw-r--r--	[Actions]
SmartTrace.apk	599.59 KB	File	2011/05/15 23:50	-rw-r--r--	[Actions]
template.py	890 Bytes	File	2011/05/15 23:35	-rw-r--r--	[Actions]
tme.apk	9.39 KB	File	2011/05/15 23:51	-rw-r--r--	[Actions]

At the bottom of the table, it shows '5 Item(s) (Free: 49.66 GB)' and '1.16 MB'. The footer of the interface reads 'Quixplorer 2.3 - the Quix project'.

Σχήμα 4.11 : Ο Διαχειριστής Αρχείων στο *SmartNet*

Στο πιο πάνω σχήμα (Σχήμα 4.11) παρουσιάζεται η διαπροσωπεία που παρέχεται στον χρήστη από το *SmartNet* για την διαχείριση των αρχείων. Μέσω της διαπροσωπείας αυτής μπορεί να εκτελέσει διάφορες ενέργειες σε αρχεία που βρίσκονται στον προσωπικό του χώρο. Στο σχήμα υπάρχουν περιοχές που έχουν απαριθμηθεί (από το 1 μέχρι και το 4). Ακολουθεί περιγραφή των αριθμημένων περιοχών καθώς και η λειτουργία τους:

- 1. Μενού Ενέργειών :** Σε αυτή την περιοχή παρουσιάζονται ενέργειες που μπορεί να εκτελέσει ο χρήστης μέσω του Διαχειριστή Αρχείων. Συγκεκριμένα ο χρήστης μπορεί να μετακινηθεί ένα επίπεδο πάνω από το επίπεδο αρχείων που βρίσκεται, να επιστρέψει στον αρχικό φάκελό του (*home directory*), να αναζητήσει ένα αρχείο, να αντιγράψει να διαγράψει να ανεβάσει αλλά και να μετακινήσει ένα αρχείο.
- 2. Δημιουργία Αρχείου/Φακέλου :** Ο χρήστης μπορεί να δημιουργήσει ένα καινούριο αρχείο ή φάκελο στην τοποθεσία και με το όνομα που επιθυμεί.
- 3. Αρχεία Χρήστη :** Στην περιοχή αυτή παρουσιάζονται τα αρχεία του χρήστη. Τα στιγμιότυπα οθόνων που παίρνει ο χρήστης αποθηκεύονται και μπορούν να ανακτηθούν από εδώ. Επίσης το πρότυπο αρχείο της *Python* το οποίο πρέπει να χρησιμοποιηθεί από τον χρήστη για να τρέξει την δική του εφαρμογή βρίσκεται εδώ εξ' ορισμού από την δημιουργία λογαριασμού στο *SmartNet*. Εδώ μπορεί να βρει και να ανακτήσει επίσης όλα τα αρχεία τα οποία μετέφερε και εγκατέστησε στις συσκευές που κράτησε. Συγκεκριμένα παρουσιάζεται το όνομα κάθε αρχείου, το μέγεθός του, ο τύπος του, η ημερομηνία που τροποποιήθηκε και τα δικαιώματα που έχει σε κάθε αρχείο. Πατώντας στα δικαιώματα έχει την επιλογή να θέσει τα δικαιώματα (*read, write, execute*) του αρχείου για κάθε μια από τις τρεις ομάδες (*owner, group, public*).
- 4. Ενέργειες αρχείων :** Από εδώ ο χρήστης έχει την δυνατότητα να ανακτήσει ένα αρχείο όπως επίσης να το τροποποιήσει *online* σε περίπτωση που είναι αρχείο κειμένου όπως φαίνεται στο Σχήμα 4.12.



**Σχήμα 4.12 :** Τροποποίηση αρχείων *online* μέσω του *File Manager* στο *SmartNet*

#### 4.8 Γενικά Χαρακτηριστικά

Το μενού πλοήγησης του *SmartNet* έχει δύο χαρακτηριστικά που αξίζει να σημειωθούν αφού δίνουν στο *SmartNet* μια πιο διαδραστική αίσθηση στον χρήστη. Συγκεκριμένα το μενού είναι συμπτυσσόμενο όπως φέρεται στο *Σχήμα 4.13*. Πατώντας δύο φορές σε οποιαδήποτε επιλογή στο μενού πλοήγησης το μενού συμπύσσεται και πατώντας ξανά μια φορά σε οποιαδήποτε επιλογή ανοίγει.



**Σχήμα 4.13 :** Συμπτυσσόμενο μενού πλοήγησης στο *SmartNet*



Το δεύτερο χαρακτηριστικό που αξίζει να σημειωθεί είναι ότι ο χρήστης μπορεί να μετακινήσει τις επιλογές που έχει στο μενού πλοήγησης χρησιμοποιώντας *drag and drop* και να τις επανατοποθετήσει όπως αυτός επιθυμεί προσαρμόζοντας έτσι το μενού σε ότι θεωρεί αυτός πιο εύχρηστο όπως φαίνεται στο Σχήμα 4.14.



**Σχήμα 4.14 :** Αναπροσαρμογή του μενού πλοήγησης στο *SmartNet*

#### **4.9 Παραγωγή Κλειδιών Εγγραφής ( Admin )**

Εάν ο χρήστης είναι administrator του δίνεται η επιλογή παραγωγής κλειδιών εγγραφής (Σχήμα 4.15). Επίσης βλέπει τα υπάρχον κλειδιά εγγραφής τα οποία είναι έγκυρα. Επιλέγοντας την επιλογή “Generate Registration Codes” ζητείται από τον administrator να εισάγει τον αριθμό των κλειδιών εγγραφής που θέλει να παράγει και στην συνέχεια εμφανίζονται τα εν λόγω κλειδιά στην λίστα με τα έγκυρα κλειδιά.



**Σχήμα 4.15 :** Κλειδιά εγγραφής και παραγωγή κλειδιών από administrator στο *SmartNet*

## Κεφάλαιο 5

### Πειραματική Λειτουργία της Υποδομής SmartNet

---

5.1 Πειραματική Διαδικασία	52
5.2 <i>SmartTrace</i>	53
5.3 Πειραματικά Αποτελέσματα	54

---

Στο προηγούμενο κεφάλαιο έγινε παρουσίαση του Συστήματος Διαπροσωπείας του *SmartNet* ενώ επεξηγήθηκαν οι δυνατότητες που παρέχει και ο τρόπος με τον οποίο λειτουργεί. Στο παρών κεφάλαιο αρχικά θα αναλύσουμε την πειραματική διαδικασία που θα ακολουθήθηκε για την εκτέλεση των πειραμάτων. Στην συνέχεια θα παρουσιάσουμε την εφαρμογή *SmartTrace* η οποία χρησιμοποιήθηκε για την εκτέλεση της πειραματικής λειτουργίας του *SmartNet*. Τέλος θα παρουσιάσουμε τα πειραματικά αποτελέσματα τα οποία εξήχθησαν βάση της πειραματικής λειτουργίας.

## 5.1 Πειραματική Διαδικασία

Μετρήσαμε τον χρόνο που χρειάζεται για να εκτελέσει κάποιος ενέργειες στις κινητές συσκευές πρώτα χειρονακτικά και μετά με την χρήση του SmartNet. Με αυτό τον τρόπο βλέπουμε εάν όντως η χρήση του παρέχει διευκολύνσεις και εξοικονομά χρόνο. Οι ενέργειες που εκτελέσαμε απαρτίζουν πραγματικό σενάριο εκτέλεσης της εφαρμογής του SmartTrace [44] (το οποίο περιγράφουμε στην επόμενη υποπαράγραφο) μέσω του SmartNet και χειρονακτικά. Συγκεκριμένα πάρθηκαν μετρήσεις για το πόσο χρόνο χρειάζεται για να εγκαταστήσει ένας χρήστης το SmartTrace, να ανεβάσει τέσσερα αρχεία στις συσκευές απαραίτητα για την λειτουργία του και να το ξεκινήσει εκτελώντας μια ενέργεια. Τέλος πήραμε μετρήσεις για τον χρόνο απεγκατάσταση του SmartTrace.

## 5.2 SmartTrace

Το *SmartTrace* [44] έχει δημιουργηθεί στα πλαίσια της διπλωματικής εργασίας του συμφοιτητή και φίλου Κώστα Κωνσταντίνου. Συγκεκριμένα το *SmartTrace* παρουσιάζεται ένα καινοτόμο πλαίσιο αναζήτησης ομοιότητας σε χώρο-χρονικά δεδομένα, χωρίς να αποκαλύπτεται κανένα στοιχείο για τους συμμετάσχοντες χρήστες. Το πλαίσιο που έχει υλοποιηθεί, το *SmartTrace*, εκμεταλλεύεται την ευκαιρία και την συμμετοχή των χρηστών για να απαντηθούν άμεσα οι ερωτήσεις της μορφής: Βρες τα αντικείμενα τα οποία ακολουθούν μια κίνηση όμοια με ένα αντικείμενο Q, όπου Q είναι κάποια τροχιά εισόδου. Το *SmartTrace* στηρίζεται στο αποκλειστικά μοντέλο αποθήκευσης δεδομένων, όπου χωρικά δεδομένα αποθηκεύονται τοπικά στην έξυπνη κινητή συσκευή για επίδοση και απόκρυψη πληροφοριών. Το *SmartTrace* μετά αναπτύσσει έναν αποδοτικό top-k αλγόριθμο επερώτησης, ο οποίος εκμεταλλεύεται τα κατανεμημένα μετρά ομοιότητας τροχείων, ανθεκτικό στον χωρικό και χρονικό θόρυβο, καταφέροντας να πάρει την πιο σχετική απάντηση με το Q γρήγορα και αποδοτικά.

Στην εν λόγω διπλωματική εργασία δίνεται η δυνατότητα επερώτησης με αποστολή τροχιών από συντεταγμένες (*GPS coordinates*) αλλά και από ασύρματα σημεία πρόσβασης συνδυασμένα με την ισχύ του σήματος (*Wi-Fi access point with received signal strength*). Επίσης έχουν υλοποιηθεί δυο τρόποι επερώτησης αντίστοιχα και για τις δυο πτυχές :

Αλληλεπιδραστικό μοντέλο επερώτησης, όπου οι συσκευές χειρίζονται από του χρήστες για να αναγνωρίσουν ποιος κινείται πιο όμοια με τον κόμβο που έχει κάνει την επερώτηση.

Μοντέλο επερώτησης βάση αρχείου, όπου μπορεί να γίνει επερώτηση μεγάλης κλίμακας μέσω των αποθηκευμένων αρχείων με σκοπό την εύρεση των K πιο κοινών τροχιών γρήγορα και αποδοτικά.

Η όλη υλοποίηση έγινε πάνω σε συσκευές Android. Συγκεκριμένα πάνω σε HTC desire, HTC Wildfire και HTC Hero. Το Android SDK χρησιμοποιήθηκε για την υλοποίηση των αλγορίθμων και η αντικειμενοστραφείς γλώσσά προγραμματισμού Java για την υλοποίηση των εξυπηρετητών.

### 5.3 Πειραματικά Αποτελέσματα

Παρουσιάζονται οι χρόνοι που χρειάστηκε ένα άτομο χειρονακτικά για την εκτέλεση ενεργειών σε δύο συσκευές και οι χρόνοι για την εκτέλεση των ίδιων ενεργειών με την χρήση του SmartNet (Πίνακας 5.1).

Ενέργεια	Χρόνος Χειρονακτικά (sec)	Χρόνος με <i>SmartNet</i> (sec)
Εγκατάσταση αρχείου μεγέθους 280.12 KB ( <i>SmartTrace</i> )	176.3	4.8
Ανέβασμα τεσσάρων αρχείων συνολικού 80.9 KB	229.7	16.7
Έναρξη Προγράμματος και εκτέλεση ενεργειών ( <i>SmartTrace</i> )	5.3	19.4
Απεγκατάσταση Αρχείου	245.4	6
<b>Συνολικός Χρόνος</b>	<b>665.7</b>	<b>46.9</b>

**Πίνακας 5.1 : Αποτελέσματα Χρόνων της Πειραματικής Λειτουργίας του SmartNet**

Οι πιο πάνω χρόνοι στην περίπτωση αύξησης του αριθμού των συσκευών αυξάνονται εκθετικά εάν πρόκειται για χειρονακτική διαχείριση ενώ για το SmartNet παραμένουν σχεδόν σταθερά. Αυτό συμβαίνει γιατί ένα άτομο εκτελεί σειριακά ενέργειες σε κάθε συσκευή ενώ το SmartNet εκτελεί τις ενέργειες προγραμματιστικά με ελάχιστη διαφορά χρόνου σε κάθε συσκευή.

# Κεφάλαιο 6

## Συμπέρασμα και Μελλοντική Δουλειά

---

6.1 Συμπεράσματα	55
6.2 Μελλοντική Δουλειά	55

---

### 6.1 Συμπεράσματα

Μετά την πειραματική διαδικασία που περιγράψαμε στο προηγούμενο κεφάλαιο, είδαμε ότι το *SmartNet* δουλεύει ορθά και παρέχει ένα πλήθος ενεργειών για εύκολη διαχείριση πολλαπλών συσκευών Android. Η υποδομή που δημιουργήσαμε και η αρχιτεκτονική του *SmartNet* το καθιστά επεκτάσιμο με πολύ λίγη προσπάθεια. Επίσης είδαμε την χρησιμότητα του σε πραγματικές συνθήκες αφού ο χρόνος διεκπεραίωσης ενεργειών σε πολλαπλές συσκευές αποτελεί μια αρκετά χρονοβόρα λειτουργία την οποία το *SmartNet* μειώνει. Στο πείραμα που εκτελέσαμε είδαμε ότι το *SmartNet* ήταν δεκατέσσερις φορές πιο γρήγορο από την χειρονακτική διαχείριση. Τέλος όσο περισσότερες συσκευές έχουμε να διαχειριστούμε ταυτόχρονα ο χρόνος διεκπεραίωσης ενεργειών χειρονακτικά αυξάνεται με πολύ ψηλούς ρυθμούς είναι στο *SmartNet* ο χρόνος δεν μεταβάλλεται σε τόσο βαθμό.

### 6.2 Μελλοντική Δουλειά

Εάν και το σύστημα δουλεύει ορθά, θα μπορούσαμε να εκτελέσουμε περισσότερα πειράματα σε αυτό. Επίσης στο μέλλον θα μπορούσαμε να βλέπουμε τις οθόνες των συσκευών σε πραγματικό χρόνο. Τέλος θα μπορούσαμε να εισάγουμε περισσότερες λειτουργίες στο σύστημα, πράγμα αρκετά εύκολο λόγω της αρχιτεκτονικής που χρησιμοποιήσαμε.

## Βιβλιογραφία

- [1] **Android.** <http://www.android.com/>
- [2] **HTC Desire.** <http://www.htc.com/www/product/desire/overview.html>
- [3] **HTC Hero.** <http://www.htc.com/www/product/hero/overview.html>
- [4] **HTML.** <http://www.webopedia.com/TERM/H/html.html>
- [5] **PHP.** <http://www.php.net/>
- [6] **Android** **Debug** **Bridge.**  
<http://developer.android.com/guide/developing/tools/adb.html>
- [7] **Geoffrey Werner-Allen, Patrick Swieskowski, and Matt Welsh,** “MoteLab: A Wireless Sensor Network Testbed.” In: In Proceedings Fourth International Conference on Information Processing in Sensor Networks (IPSN'05). IEEE (2005) 438-488
- [8] **WISEBED.** <http://www.wisebed.eu/>
- [9] **Remote** **Device** **Access.**  
[http://www.forum.nokia.com/Devices/Remote\\_device\\_access/](http://www.forum.nokia.com/Devices/Remote_device_access/)
- [10] **Amazon EC2.** <http://aws.amazon.com/ec2/>
- [11] **NIST.** <http://www.nist.gov/>
- [12] **Smartphone.**  
[http://cellphones.about.com/od/smartphonebasics/a/what\\_is\\_smart.htm](http://cellphones.about.com/od/smartphonebasics/a/what_is_smart.htm)
- [14] **HTC.** <http://www.htc.com/>
- [15] **Linux.** <http://www.linux.com/>
- [16] **B. W. Kernighan, D. M. Ritchie,** “The C programming language” Prentice-Hall, Inc. Upper Saddle River, NJ, USA ©1978. ISBN:0-13-110163-3
- [17] **Java.** <http://www.java.com/en/>
- [18] **Apache Software License.** <http://source.android.com/source/licenses.html>
- [19] **DalvikVM.** <http://www.dalvikvm.com/>
- [20] **Tarmo Virki and Sinead Carew,** "Google topples Symbian from smartphones top spot". Reuters. <http://uk.reuters.com/article/2011/01/31/oukin-uk-google-nokia-idUKTRE70U1YT20110131>

- [21] **Canalys**, "Google's Android becomes the world's leading smart phone platform". Canalys research release: r2011013. 31 January 2011. <http://www.canalys.com/pr/2011/r2011013.html>
- [22] **Andrew Kameka**, "Android has 150k apps, 350k daily activations, and more notes from Eric Schmidt's MWC keynote", 15 February 2011. <http://androinica.com/>
- [23] **Android Stats**. <http://www.androlib.com/appstats.aspx>
- [24] **Android Market**. <https://market.android.com/>
- [25] **P. Levis, N. Lee, M. Welsh, and D. Culler**, "TOSSIM: Accurate and scalable simulation of entire TinyOS applications," in Proc. the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003), November 2003.
- [26] **V. Shnayder, M. Hempstead, B. rong Chen, G. Werner-Allen, and M. Welsh**, "Simulating the power consumption of large-scale sensor network applications," in Proc. the 2nd international conference on Embedded networked sensor systems (SenSys 2004), November 2004.
- [27] **Harvard University**. <http://www.harvard.edu/>
- [28] **MySQL**. <http://www.mysql.com/>
- [29] **iSense**. <http://www.coalesenses.com/index.php?page=isense-software>
- [30] **Contiki**. <http://www.sics.se/contiki/>
- [31] **TinyOS**. <http://www.tinyos.net/>
- [32] **Symbian**. <http://symbian.nokia.com/>
- [33] **Maemo**. <http://maemo.nokia.com/>
- [34] **Java** **Web** **Start**.  
<http://download.oracle.com/javase/6/docs/technotes/guides/javaws/>
- [35] **jQuery Usage Statistics**. <http://trends.builtwith.com/javascript/JQuery>
- [36] **Usage of JavaScript libraries for websites**.  
[http://w3techs.com/technologies/overview/javascript\\_library/all](http://w3techs.com/technologies/overview/javascript_library/all)
- [37] **DOM**. <http://www.w3.org/TR/WD-DOM/introduction.html>
- [38] **Jesse James Garrett**, "Ajax: A New Approach to Web Applications", <http://www.adaptivepath.com/>
- [39] **Sun Microsystems**. <http://www.sun.com/>



- [40] **Servlet.** [http://java.sun.com/j2ee/tutorial/1\\_3-fcs/doc/Servlets2.html](http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/Servlets2.html)
- [41] **Apache Tomcat 7.** <http://www.tomcat7.com/>
- [42] **PJB.** <http://php-java-bridge.sourceforge.net/pjb/>
- [43] **monkeyrunner.**  
[http://developer.android.com/guide/developing/tools/monkeyrunner\\_concepts.html](http://developer.android.com/guide/developing/tools/monkeyrunner_concepts.html)
- [44] **Costa C., Laoudias C., Zeinalipour-Yazti D. and Gunopulos D.,** “Smart-Trace: Finding Similar Trajectories in Smartphone Networks without Disclosing the Traces”, Demo paper to appear in IEEE ICDE’11.

## Παράρτημα Α

### Οδηγός Εγκατάστασης SmartNet

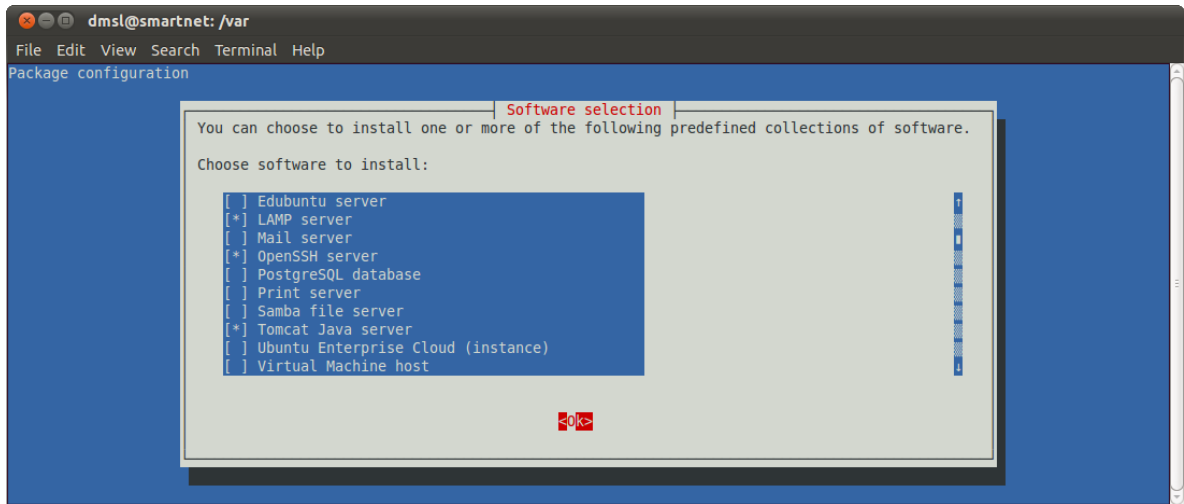
Στο Παράρτημα θα περιγράψουμε τον τρόπο με τον οποίο μπορούμε να εγκαταστήσουμε σε ένα εξυπηρετητή το SmartNet. Συγκεκριμένα για το παράδειγμα χρησιμοποιήσαμε εξυπηρετητή με λειτουργικό σύστημα Ubuntu Linux Server Edition v. 10.10.

Αρχικά πρέπει να εγκαταστήσουμε κάποια προγράμματα τα οποία είναι αναγκαία για την λειτουργία του SmartNet. Συγκεκριμένα πρέπει να εγκαταστήσουμε: Java, PHP, MySQL, Apache2, Tomcat7,

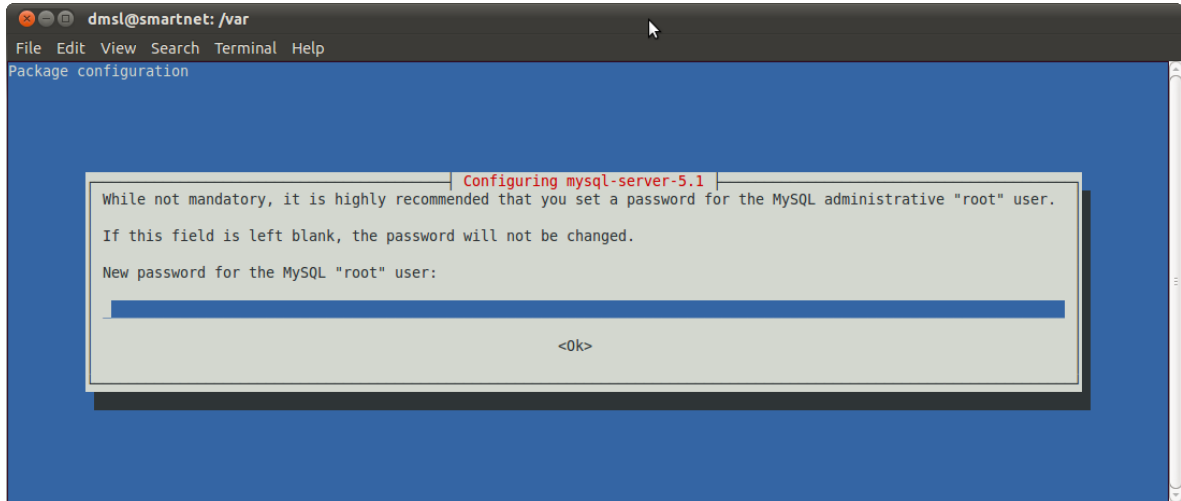
Για την εγκατάσταση των Apache2, Tomcat7, MySQL, Java και PHP πληκτρολογούμε σε ένα τερματικό (Applications >Accessories >Terminal) την ακόλουθη εντολή:

```
$ sudo taskel
```

Στην συνέχεια στο παράθυρο που παρουσιάζεται επιλέγουμε από το μενού τις επιλογές LAMP server και Tomcat Java server και μετά OK όπως φαίνεται στο πιο κάτω σχήμα.



Κατά την εγκατάσταση της MySQL εισάγουμε τον επιθυμητό κωδικό πρόσβασης για τον “root” λογαριασμό και συνεχίζουμε την εγκατάσταση.



Όταν τελειώσει η εγκατάσταση πρέπει να ανεβάσουμε τα απαραίτητα αρχεία στον εξυπηρετητή. Για να το πετύχουμε αυτό ανεβάζουμε το αρχείο “SmartNetDirectory.tar” το οποίο βρίσκεται στο CD. Για να το πετύχουμε αυτό γράφουμε την πιο κάτω εντολή

```
$ sftp root@smartnet
```

Στη συνέχεια εισάγουμε τον κωδικό για να ενωθούμε και πληκτρολογούμε την ακόλουθη εντολή

```
$ put <local path> <remote path>
```

Όπου το <local path> είναι το μονοπάτι στο τοπικό αρχείο “SmartNetDirectory.tar” και <remote path> ο απομακρυσμένος φάκελος /var/www/.

Από την σελίδα <http://developer.android.com/sdk/index.html> κατεβάζουμε το **Android SDK** επιλέγοντας την έκδοση για το Linux.

Με τον ίδιο τρόπο ανεβάζουμε το αρχείο “Server.tar” το οποίο επίσης βρίσκεται στο CD αλλά και το αρχείο του Android SDK το οποίο κατεβάσαμε νωρίτερα. Με την ακόλουθη εντολή αποσυμπιέζουμε και τα τρία συμπιεσμένα αρχεία τύπου tar που ανεβάσαμε.

```
$ tar -xvf <filename> <extract path here>
```

Το αρχείο SmartNetDirectory.tar το αποσυμπιέζουμε στον φάκελο /var/www/ ενώ τα υπόλοιπα τα αποσυμπιέζουμε σε τοποθεσία της επιλογής μας (π.χ. /home/dmsl/). Στον φάκελο **Server/Scripts** βρίσκεται το αρχείο “**SmartNet.sql**” το οποίο κάνουμε import στην βάση δεδομένων με την πιο κάτω ακολουθία εντολών.

```
$ sudo mysql -u <user> -p<password>
.....
$mysql> CREATE DATABASE MobNet;
$mysql>quit
$sudo mysql -u <user> -p<pass> MobNet < SmartNet.sql
```

Επίσης στον φάκελο “**Server**” υπάρχει ο φάκελος “**JavaBridge**” τον οποίο μετακινούμε στον φάκελο /usr/share/tomacat7/webapps .

Στο αρχείο **config.php** το οποίο βρίσκεται στον φάκελο /var/www/ υπάρχουν ρυθμίσεις που πρέπει να αλλάξουμε σύμφωνα με τις τοποθεσίες των εργαλείων **adb** και **monkeyrunner** τα οποία βρίσκονται στο φάκελο **AndroidSDK/android-sdk-linux\_86/tools** τον οποίο ανεβάσαμε και αποσυμπιέσαμε ήδη.

Στη συνέχεια αφού ενώσουμε τις συσκευές τρέχουμε τις ακόλουθες εντολές δεδομένου ότι βρισκόμαστε στο **AndroidSDK/android-sdk-linux\_86/tools** .

```
$ sudo ./adb kill-server
$ sudo ./adb start-server
$ sudo ./adb devices
```

Στο τέλος παίρνουμε μια λίστα με τις συσκευές που βρίσκονται ενωμένες. Τρέχουμε επίσης το αρχείο **autoScreens.sh** που βρίσκεται στο φάκελο **Server/scripts**. Τέλος τρέχουμε τον εξυπηρετητή του **SmartNet Scheduler** ο οποίος βρίσκεται στον φάκελο **Server/SmartNet Scheduler** με την ακόλουθη εντολή

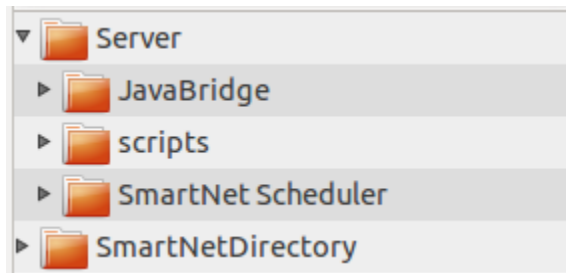
```
$ java Multiserver
```

Έτσι το SmartNet είναι έτοιμο για να τρέξει.

## Παράρτημα Β

### Πηγαίος Κώδικας

Ακολουθεί ο πηγαίος κώδικας του *SmartNet* και τα αναγκαία αρχεία για εγκατάσταση του *SmartNet* τα οποία βρίσκονται στο CD που συνοδεύει την παρούσα διπλωματική εργασία κάτω από τον φάκελο Source Code που έχει την εξής μορφή:



Στο φάκελο

- **Java Bridge:** Βρίσκονται τα αρχεία για το PHP/Java Bridge που τρέχουν σαν servlet στον Tomcat7 εξυπηρετητή καθώς και τα client αρχεία του SmartNet Scheduler που τρέχουν επίσης σαν servlet.
- **scripts:** Βρίσκεται το αρχείο που πρέπει να τρέχει για την λήψη στιγμιότυπων οθόνης των κινητών συσκευών καθώς επίσης και το script για την εισαγωγή της Βάσης Δεδομένων του SmartNet.
- **SmartNet Scheduler:** Βρίσκεται η υλοποίηση σε Java του SmartNet Scheduler
- **SmartNetDirectory:** Βρίσκεται η υλοποίηση του SmartNet.

**Παράρτημα Γ**  
**Ψηφιακός Δίσκος**

