



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ  
ΚΡΗΤΗΣ**

**Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Εφαρμοσμένης Πληροφορικής &  
Πολυμέσων**

## **«Εκπαιδευτική Εφαρμογή m-learning»**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

της

**Ζαχαρώ-Ευαγγελία Ανδριώτη**

**AM 1444**

Εισηγητές:

Μιχάλης Κλεισαρχάκης

Αθανάσιος Μαλάμος

Ηράκλειο, Οκτώβριος 2010

Copyright © Ζαχαρώ-Ευαγγελία Ανδριώτη, 2010

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς την συγγραφέα.

## Περίληψη

Αντικείμενο της παρούσας πτυχιακής εργασίας είναι ο σχεδιασμός, η ανάπτυξη και η υλοποίηση ενός εκπαιδευτικού προγράμματος (υπό τη μορφή «παιχνιδιού») μέσω συσκευών κινητών τηλεφώνων με τη χρήση των τεχνολογιών Java 2 Micro Edition και Bluetooth.

Πρόκειται για μια εφαρμογή που απευθύνεται σε μαθητές Γυμνασίου και δίνει την δυνατότητα στον καθηγητή που βρίσκεται μέσα στην τάξη να μεταφέρει μια εφαρμογή και πιο συγκεκριμένα ένα εκπαιδευτικό παιχνίδι από τον υπολογιστή του στα κινητά τηλέφωνα των μαθητών μέσω της τεχνολογίας Bluetooth.

Αυτό το εκπαιδευτικό παιχνίδι αποτελείται από ερωτήσεις και απαντήσεις που δίνει την δυνατότητα στον μαθητή/ στην μαθήτριά να ελέγξει τις γνώσεις του/της σ' ένα συγκεκριμένο μάθημα και να αξιολογηθεί πάνω σ' αυτό.

Τέλος, μας δίνεται η ευκαιρία να ασχοληθούμε και να μάθουμε για το νέο και γρήγορα αναπτυσσόμενο πεδίο του mobile learning.

Λέξεις κλειδιά: mobile-learning, Java 2 Micro Edition, MIDlets, Bluetooth, SQL

## Abstract

The objective of this diploma thesis is the design, the development and the implementation of an educational game for devices such as mobile phones using both Java 2 Micro Edition technology and Bluetooth technology.

This application applies to junior high school students. It gives the opportunity to the professor, who is in the classroom to transfer an application (and to be more specific an educational game) from his/her desktop computer (or laptop) to students' mobiles phones via Bluetooth technology.

This educational game comprises with questions and answers and gives the student the opportunity to check his/her knowledge on a particular course and to be scored.

Finally, this diploma thesis makes us acquainted with the new and fast developing topic of the mobile learning.

Key words: mobile-learning, Java 2 Micro Edition, MIDlets, Bluetooth, SQL

## Ευχαριστίες

Η παρούσα πτυχιακή εργασία έχει θέμα «Εκπαιδευτική Εφαρμογή m-learning».

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου Μιχάλη Κλεισαρχάκη που μου έδωσε την ευκαιρία να ασχοληθώ με το παραπάνω θέμα και να εμπλουτίσω τις γνώσεις μου πάνω στις τεχνολογίες των κινητών τηλεφώνων.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένεια μου που είναι δίπλα μου σε κάθε βήμα που κάνω, την στήριξη της όλα αυτά τα χρόνια και για την συμπαράσταση της όλον αυτόν τον καιρό που έγραφα την πτυχιακή μου εργασία.

Τέλος, θα ήθελα να ευχαριστήσω τον συμφοιτητή μου τον Κώστα Γρεβενίτη για την ηθική και ψυχολογική υποστήριξη του.

Οκτώβριος 2010,  
Ζαχαρώ-Ευαγγελία Ανδριώτη

## Περιεχόμενα

Κεφάλαιο 1 <sup>ο</sup> .....	14
1.1 Στόχος της πτυχιακής εργασίας.....	14
1.2 Σε ποιους απευθύνεται η εργασία.....	14
1.3 Λειτουργία του εκπαιδευτικού παιχνιδιού.....	14
1.4 Περιγραφή της εφαρμογής.....	15
1.5 Οργάνωση Τόμου.....	16
Κεφάλαιο 2 <sup>ο</sup> .....	18
Εισαγωγή.....	18
Ενότητα 1 <sup>η</sup> .....	19
1.1 Ιστορία των Ασυρμάτων Επικοινωνιών.....	19
1.2 Ιστορία της Κινητής Τηλεφωνίας.....	19
1.2.1 Κινητή Τηλεφωνία (0G).....	20
1.2.2 Πρώτη Γενιά (1G).....	20
1.2.3 Δεύτερη Γενιά (2G).....	21
1.2.4 Τρίτη Γενιά (3G).....	23
1.2.5 Τέταρτη Γενιά (4G).....	23
Ενότητα 2 <sup>η</sup> .....	24
2.1 Εισαγωγή στη Γλώσσα Αντικειμενοστραφούς Προγραμματισμού JAVA.....	24
2.2 Εισαγωγή στη Java 2 Platform.....	26
2.3 Τεχνολογία J2ME.....	27
2.4 MIDlets.....	36
Ενότητα 3 <sup>η</sup> .....	40

3.1 Bluetooth.....	40
3.2 Java APIs για Ασύρματη Τεχνολογία Bluetooth JSR-82.....	43
Ενότητα 4 <sup>η</sup> .....	45
4.1 Βάσεις Δεδομένων.....	45
Ενότητα 5 <sup>η</sup> .....	51
5.1 Netbeans IDE.....	51
5.2 Series 40 Platform.....	52
Ενότητα 6 <sup>η</sup> .....	55
6.1 Συνεργατική Μάθηση μέσω Κινητών Συσκευών (Mobile Learning).....	55
6.2 Μάθηση μέσω Κινητών Συσκευών.....	56
6.3 Σχεδιασμός Εφαρμογών για Συνεργατική Μάθηση μέσω Κινητών Συσκευών.....	58
Βιβλιογραφία για το 2 <sup>ο</sup> Κεφάλαιο.....	62
Κεφάλαιο 3 <sup>ο</sup> .....	65
Εισαγωγή.....	65
Μέρος 1 <sup>ο</sup> .....	66
Βιβλιογραφία για το 1 <sup>ο</sup> Μέρος.....	91
Μέρος 2 <sup>ο</sup> .....	92
3.2.1 Εγκατάσταση Bluetooth 2.0.....	93
3.2.2 Χαρακτηριστικά κινητού τηλεφώνου.....	97
3.2.3 Download πλατφόρμες NOKIA & SONY ERICSSON.....	97
3.2.4 Εγκατάσταση των πλατφόρμων NOKIA & SONY ERICSSON στον υπολογιστή & στο πρόγραμμα.....	98
3.2.4.1α Εγκατάσταση της πλατφόρμας NOKIA στον υπολογιστή.....	98
3.2.4.1β Εγκατάσταση της πλατφόρμας NOKIA στο πρόγραμμα Netbeans.....	98
3.2.4.2α Εγκατάσταση της πλατφόρμας SONY ERICSSON στον υπολογιστή.....	104
3.2.4.2β Εγκατάσταση της πλατφόρμας SONY ERICSSON στο	

πρόγραμμα Netbeans.....	104
3.2.5 Download open-source κώδικα από το forum της NOKIA.....	109
3.2.6 Ανάλυση του τελικού κώδικα γραμμή προς γραμμή.....	112
3.2.7 Εφαρμογή του κώδικα με χρήση εξομοιωτή (emulator) της Sun Java Wireless Toolkit και εμφάνιση αποτελεσμάτων.....	127
3.2.8 Σύνδεση NOKIA 5300 με τον υπολογιστή μέσω Bluetooth και εμφάνιση των αποτελεσμάτων χρησιμοποιώντας τον εξομοιωτή (emulator) της NOKIA.....	133
Βιβλιογραφία για το 3 <sup>ο</sup> Κεφάλαιο.....	142
Κεφάλαιο 4 <sup>ο</sup> .....	144
4.1 Σύνοψη.....	144
4.2 m-learning versus e-learning.....	145
4.3 Από το e-learning στο m-learning.....	146
4.4 Συμπεράσματα - Προοπτική.....	146
Βιβλιογραφία για το 4 <sup>ο</sup> Κεφάλαιο.....	147



# Κεφάλαιο 1<sup>ο</sup>

## 1.1 Στόχος της πτυχιακής εργασίας

Η παρούσα πτυχιακή εργασία αφορά μια εκπαιδευτική εφαρμογή του mobile learning. Στόχος της πτυχιακής μας εργασίας είναι να φτιάξουμε ένα εκπαιδευτικό παιχνίδι το οποίο στη συνέχεια θα μπορεί να μεταφερθεί από τον υπολογιστή στο κινητό τηλέφωνο ενός μαθητή/μιας μαθήτριας μέσω της τεχνολογίας Bluetooth.

## 1.2 Σε ποιους απευθύνεται η εργασία

Επειδή, θέλουμε αυτή η εφαρμογή να χρησιμοποιηθεί από μικρές ηλικίες αλλά να έχουν τα παιδιά στην κατοχή τους κινητό τηλέφωνο, συμπεραίνουμε ότι αυτή η εφαρμογή θα απευθύνεται σε παιδιά ηλικίας 13-15 χρόνων (δηλαδή σε μαθητές Γυμνασίου).

Η εφαρμογή μας όπως αναφέραμε παραπάνω θα έχει εκπαιδευτικό χαρακτήρα. Το θέμα που θα υποστηρίξει διαλέξαμε να είναι ερωτήσεις και απαντήσεις που αφορούν το μάθημα των ηλεκτρονικών υπολογιστών.

## 1.3 Η φύση του εκπαιδευτικού παιχνιδιού

Το παιχνίδι θα αποτελείται από ερωτήσεις εκπαιδευτικού χαρακτήρα. Σε κάθε ερώτηση ο μαθητής/η μαθήτρια θα πρέπει να δίνει μία απάντηση η οποία και θα αξιολογείται.

Οι απαντήσεις του παιχνιδιού μπορεί να είναι τύπου πολλαπλής επιλογής, δηλαδή ο χρήστης θα έχει να επιλέξει μία απάντηση ανάμεσα από 2 μέχρι και 4 πιθανές απαντήσεις. Το αποτέλεσμα (σωστό-λάθος) κάθε απάντησης θα γίνεται γνωστό στον

χρήστη. Στο τέλος, θα δίδεται το συνολικό αποτέλεσμα με τη μορφή βαθμολογίας σαν ποσοστού των σωστών από το σύνολο των απαντήσεων.

## 1.4 Περιγραφή της εφαρμογής

Καταρχήν, θα φτιάξουμε μια Βάση Δεδομένων. Στη Βάση Δεδομένων θα αποθηκεύσουμε 100 ερωτήσεις. Η κάθε ερώτηση θα αποτελείται από:

- τον Αύξοντα Αριθμό της ερώτησης (δηλαδή ο αριθμός που αντιστοιχεί σε κάθε ερώτηση),
- τις πιθανές απαντήσεις που μπορεί να έχει (μια ερώτηση μπορεί να έχει από δύο μέχρι τέσσερις πιθανές απαντήσεις),
- τη σωστή απάντηση
- και έναν αριθμό ο οποίος αντιπροσωπεύει το μάθημα.

Το επόμενο βήμα είναι να συνδέσουμε τη βάση μας. Στη συνέχεια, θα χρησιμοποιήσουμε το πρόγραμμα Netbeans IDE που είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης για να φτιάξουμε ένα πρόγραμμα το οποίο θα κάνει τα εξής:

Θα διαβάσει τη Βάση Δεδομένων και θα επιλέγει τυχαία (random) 10 ερωτήσεις μαζί με τις πιθανές απαντήσεις τους, τον A/A της ερώτησης και την σωστή απάντηση. Μετά, θα αποθηκεύονται σε ένα txt αρχείο με το όνομα Game.txt ο A/A της ερώτησης, η ερώτηση και οι πιθανές απαντήσεις. Σε ένα άλλο txt αρχείο με το όνομα Answers.txt θα αποθηκεύσουμε μόνο τις σωστές απαντήσεις.

Στη συνέχεια, θα φτιάξουμε μια εφαρμογή για κινητά τηλέφωνα όπου θα χρησιμοποιήσουμε τα παραπάνω αρχεία που δημιουργήσαμε. Θα χρησιμοποιήσουμε την Java ME και πιο συγκεκριμένα θα χρησιμοποιήσουμε MIDlets για να υλοποιήσουμε την εφαρμογή. Η εφαρμογή θα κάνει τα ακόλουθα:

Όταν ο μαθητής/η μαθήτρια θα πατάει το κουμπί Read, θα εμφανίζει το αρχείο Game.txt στην οθόνη του κινητού τηλεφώνου. Στην οθόνη θα εμφανιστούν σε μια φόρμα οι ερωτήσεις, όπου η κάθε ερώτηση θα εμφανίζεται με τις πιθανές απαντήσεις της. Κάτω από τις πιθανές απαντήσεις θα έχουμε ένα text field. Ο μαθητής/η μαθήτρια θα συμπληρώνει στο text field την απάντηση που θεωρεί σωστή.

Έχοντας ο μαθητής/η μαθήτρια απαντήσει σε όλες τις ερωτήσεις, στη συνέχεια πατάει το κουμπί Load. Εμφανίζεται μια νέα φόρμα στην οθόνη με τις απαντήσεις που είχε δώσει ο μαθητής/η μαθήτρια. Συγχρόνως, το πρόγραμμα συγκρίνει τις απαντήσεις που έδωσε ο μαθητής/η μαθήτρια με τις απαντήσεις που βρίσκονται στο

αρχείο Answers.txt. Από τη σύγκριση αυτή, εμφανίζουμε κάτω από τις απαντήσεις του μαθητή/της μαθήτριας, το αποτέλεσμα για την κάθε ερώτηση ξεχωριστά γράφοντας αν η απάντηση που έδωσε ο μαθητής/η μαθήτρια είναι σωστή ή λάθος. Στο τέλος, θα εμφανίζεται το αποτέλεσμα του μαθητή/της μαθήτριας, δηλαδή πόσο τοις εκατό επιτυχία είχε. Η κάθε σωστή ερώτηση βαθμολογείται με +10%. Η κάθε λάθος απάντηση δεν βαθμολογείται.

## 1.5 Οργάνωση Τόμου

Η πτυχιακή μας εργασία έχει οργανωθεί σε τέσσερα κεφάλαια.

Στο παρόν **πρώτο κεφάλαιο**, αναφερόμαστε στον στόχο της πτυχιακής μας εργασίας και σε ποιους απευθύνεται. Επίσης, αναφερόμαστε στη φύση και δομή της πτυχιακής μας εργασίας, στη λειτουργικότητα της καθώς και στην περιγραφή της εφαρμογής της.

Το **δεύτερο κεφάλαιο**, είναι το θεωρητικό κομμάτι της εργασίας μας. Αναφέρεται στις έννοιες, ορισμούς και ανάλυση όλων των αντικειμένων που αφορούν την εφαρμογή μας. Αποτελείται από έξι ενότητες.

- Η πρώτη ενότητα αφορά τα ιστορικά στοιχεία της κινητής τηλεφωνίας, των ασυρμάτων επικοινωνιών, και τις τέσσερις γενιές κινητών τηλεφώνων.
- Η δεύτερη ενότητα αφορά για την γλώσσα αντικειμενοστραφούς προγραμματισμού Java, Java 2 Platform, J2ME και MIDlets.
- Η τρίτη ενότητα ασχολείται με το Bluetooth και το JSR-82.
- Η τέταρτη ενότητα αναφέρεται στις Βάσεις Δεδομένων και πιο συγκεκριμένα στη Microsoft Access, στη γλώσσα SQL και στις σχέσεις πινάκων.
- Η πέμπτη ενότητα αναφέρεται στο ολοκληρωμένο περιβάλλον ανάπτυξης Netbeans IDE, και στις μια από τις δυο πλατφόρμες που θα χρησιμοποιήσουμε. Αφορά την πλατφόρμα της NOKIA, Series 40 Platform.
- Η έκτη ενότητα αφορά στη Συνεργατική Μάθηση μέσω Κινητών Συσκευών - Mobile Learning.

Το **τρίτο κεφάλαιο** περιέχει τα τεχνικά χαρακτηριστικά της εργασίας μας. Χωρίζεται σε δύο μέρη. Στο πρώτο μέρος, περιγράφουμε τον σχεδιασμό και την υλοποίηση της Βάσης Δεδομένων, που απαιτεί η εφαρμογή μας. Επίσης, φτιάχνουμε και περιγράφουμε το πρόγραμμα που θα συνδέεται με και θα χρησιμοποιεί τη Βάση

Δεδομένων. Στο δεύτερο μέρος, ασχολούμαστε με τη δημιουργία νέου project χρησιμοποιώντας MIDlets και στη συνέχεια, εφαρμόζουμε την εφαρμογή στο κινητό μας τηλέφωνο.

Το **τέταρτο κεφάλαιο** συνοψίζει τα βασικά στάδια της ανάπτυξης της εφαρμογής και δείχνει τον διεπιστημονικό της χαρακτήρα. Τέλος, παραθέτουμε τα συμπεράσματα μας και συζητούμε τις προοπτικές που διανοίγονται με την συνεργατική μάθηση μέσω κινητών συσκευών (m-learning) και σε σχέση με την κλασική διδασκαλία μέσω ηλεκτρονικού υπολογιστή (e-learning).

# Κεφάλαιο 2<sup>ο</sup>

## Εισαγωγή

Σε αυτό το κεφάλαιο θα αναφερθούμε στις έννοιες, ορισμούς και ανάλυση όλων των αντικειμένων που αφορούν την εφαρμογή μας. Αυτό θα μας βοηθήσει στην καλύτερη κατανόηση της εφαρμογής που θα έχουμε να σχεδιάσουμε και να υλοποιήσουμε στο επόμενο κεφάλαιο (Κεφάλαιο 3<sup>ο</sup>).

Το παρόν κεφάλαιο χωρίζεται σε 6 Ενότητες.

- Η πρώτη ενότητα αφορά τα ιστορικά στοιχεία της κινητής τηλεφωνίας, των ασυρμάτων επικοινωνιών, και τις τέσσερις γενιές κινητών τηλεφώνων.
- Η δεύτερη ενότητα αφορά την γλώσσα αντικειμενοστραφούς προγραμματισμού Java, Java 2 Platform, J2ME και MIDlets.
- Η τρίτη ενότητα ασχολείται με το Bluetooth και το JSR-82.
- Η τέταρτη ενότητα αναφέρεται στις Βάσεις Δεδομένων και πιο συγκεκριμένα στη Microsoft Access, στη γλώσσα SQL και στις σχέσεις πινάκων.
- Η πέμπτη ενότητα αναφέρεται στο ολοκληρωμένο περιβάλλον ανάπτυξης Netbeans IDE, και στις μια από τις δυο πλατφόρμες που θα χρησιμοποιήσουμε. Αφορά την πλατφόρμα της NOKIA, Series 40 Platform.
- Η έκτη ενότητα αφορά στη Συνεργατική Μάθηση μέσω Κινητών Συσκευών - Mobile Learning.

# Ενότητα 1<sup>η</sup>

## 1.1 Ιστορία των Ασύρματων Επικοινωνιών

Η ιστορία των ασύρματων επικοινωνιών ξεκινούν από πολύ παλιά. Τα πρώτα βήματα της ασύρματης επικοινωνίας ξεκινούν από το 1886, όταν ο Heinrich Rudolf Hertz ανακάλυψε τη διπολική κεραία. Στη συνέχεια, όπως καταγράφεται από την ιστορία, το 1893 ο Nikola Tesla κατασκεύασε το πρώτο ασύρματο σύστημα επικοινωνίας. Το 1895 χρησιμοποίησε τα σήματα Morse σε ακτίνα 80 km.

Στη συνέχεια, ο Alexander Stepanovich Popov πέτυχε την μετάδοση ραδιοκυμάτων μεταξύ κοντινών κτιρίων το 1896. Μετά, το 1898 πέτυχε την μετάδοση ραδιοκυμάτων στα 6 μίλια και τέλος το 1899 πέτυχε την μετάδοση ραδιοκυμάτων στα 30 μίλια. Έπειτα, την σκυτάλη την παίρνει ο Guglielmo Marconi που ήταν εκείνος που έστειλε το πρώτο υπερατλαντικό σήμα το 1901. Την ίδια χρονιά γίνεται και η πρώτη μετάδοση ήχου από τον Reginald Fessenden. Έκανε την πρώτη αμφίδρομη υπερατλαντική ασύρματη επικοινωνία το 1906 και την ίδια χρονιά έγινε η πρώτη ραδιοφωνική εκπομπή με μουσικό περιεχόμενο.

Το 1900-1940 υπήρχε ευρεία χρήση ασύρματου τηλέγραφου και ιδιαίτερα από τον 1<sup>ο</sup> Παγκόσμιο Πόλεμο. Επίσης, το 1920-1930 αναπτύχθηκαν οι ραδιοφωνικές εκπομπές. [1]

## 1.2 Ιστορία της Κινητής Τηλεφωνίας

Από το 1940 και μετά παρατηρούμε την ανάπτυξη και την εξέλιξη της κινητής τηλεφωνίας. Οι πρώτες προσπάθειες έγιναν μετά τον 2<sup>ο</sup> Παγκόσμιο Πόλεμο από την Σουηδία, την Φιλανδία και την Αμερική.

Στις 3 Απριλίου 1973, στην Νέα Υόρκη, δημιουργήθηκε το πρώτο σύγχρονο κινητό τηλέφωνο με τον κωδικό MotorolaDynaTAC. Ο εφευρέτης του κινητού τηλεφώνου είναι ο Dr. Martin Cooper από την Motorola. Η συσκευή έμοιαζε με φορητό ασύρματο ύψους 25 εκατοστών και βάρους περίπου ενός κιλού. Η πρώτη κλήση που έκανε ήταν στον ανταγωνιστή του από την Bell Labs, τον Joel Engel όπου του είπε: «Joel, σου τηλεφωνώ από ένα κινητό τηλέφωνο. Από ένα πραγματικό κινητό τηλέφωνο». Όμως, λίγα χρόνια αργότερα, το 1978, η Bell κατασκεύασε δοκιμαστικά το πρώτο δίκτυο κινητής τηλεφωνίας. Στις αρχές της δεκαετίας 1980 δημιουργήθηκε

το πρώτο αυτοματοποιημένο δίκτυο κινητής τηλεφωνίας στη Σκανδιναβία. Μέχρι τα τέλη της δεκαετίας του 80 είχαν εγκατασταθεί ογκώδη κινητά τηλέφωνα σε αυτοκίνητα. [5]

Κοιτάζοντας πίσω στο χρόνο μπορούμε να δούμε την εξέλιξη των κινητών τηλεφώνων και την αντιστοίχιση τους στην αντίστοιχη γενιά στην οποία αναπτύχθηκαν.

### 1.2.1 Κινητή Τηλεφωνία (0G)

Το 1946 έγινε η εγκατάσταση του πρώτου συστήματος τηλεφωνίας από αυτοκίνητα στο Saint Louis. Το σύστημα αυτό χρησιμοποιούσε ένα μεγάλο πομπό στην κορυφή ενός ψηλού κτιρίου και είχε ένα μόνο κανάλι, το οποίο χρησιμοποιείτο τόσο για αποστολή όσο και για λήψη. Για να μιλήσει ο χρήστης, έπρεπε να πατήσει ένα πλήκτρο το οποίο ενεργοποιούσε τον πομπό και απενεργοποιούσε το δέκτη. Αυτό το σύστημα ονομάζεται πίεσε για να μιλήσεις (push-to-talk system) και εγκαταστάθηκε σε πολλές πόλεις από τα τέλη της δεκαετίας του 1950. Την δεκαετία του 1960 εγκαταστάθηκε το Βελτιωμένο Σύστημα Κινητής Τηλεφωνίας ή IMTS (Improved Mobile Telephone System). Σε αυτό το σύστημα χρησιμοποιείτο ένα πομπό υψηλής ισχύος (200 watt) στην κορυφή ενός λόφου και είχε δυο συχνότητες, μια για αποστολή και μια για λήψη. Το IMTS υποστήριζε 23 κανάλια, τα οποία εκτείνονταν από τα 150 MHz μέχρι τα 450MHz. Το πρόβλημα που παρουσιάστηκε ήταν στο περιορισμένο αριθμό καναλιών με αποτέλεσμα τα συστήματα να κορεστούν πολύ γρήγορα. [7]

### 1.2.2 Πρώτη Γενιά (1G)

Η πρώτη γενιά συστημάτων κινητής τηλεπικοινωνίας εμφανίστηκε τη δεκαετία του 1980. Η πρώτη γενιά δεν ήταν η αρχή των κινητών τηλεπικοινωνιών επειδή από πριν υπήρχαν αρκετά ραδιο-συστήματα κινητών όπως διαβάσαμε προηγουμένως (1.2.1). Τα πρώιμα αυτά δίκτυα είχαν πολύ λιγότερες δυνατότητες από ότι μεταγενέστερα τα κυψελωτά δίκτυα. Επίσης, υπήρχε πρόβλημα στην υποστήριξη της κινητικότητας των χρηστών.

Στα κυψελωτά κινητά δίκτυα, η κάλυψη της περιοχής χωρίζεται σε μικρά κελιά. Συνεπώς οι ίδιες συχνότητες μπορούν να χρησιμοποιηθούν αρκετές φορές στο δίκτυο

χωρίς να υπάρχουν φαινόμενα παρεμβολής. Αυτό κατά συνέπεια αυξάνει την δυνατότητα του δικτύου.

Η πρώτη γενιά χρησιμοποίησε τεχνικές αναλογικής μετάδοσης για την κίνηση (traffic), που ήταν σχεδόν αποκλειστικά φωνή. Δεν υπήρχε κάποιο πρότυπο που να επικράτησε, αντιθέτως υπήρχαν αρκετά. Τα πιο επιτυχημένα πρότυπα ήταν το Nordic Mobile Telephone (NMT) το οποίο αναπτύχθηκε στην Ευρώπη, το Total Access Communications System (TACS) το οποίο αναπτύχθηκε στην Αγγλία και το Advanced Mobile Phone Service (AMPS) το οποίο αναπτύχθηκε στην Η.Π.Α το 1982 και σχεδιάστηκε από τη Bell Labs. Υπήρξαν και άλλα πρότυπα που αναπτύχθηκαν και χρησιμοποιήθηκαν αποκλειστικά σε μια χώρα όπως το πρότυπο C-Netz στη Δυτική Γερμανία και το πρότυπο Radiocomm 2000 στη Γαλλία.

Πρέπει να επισημάνουμε ότι παρόλο που στην σημερινή εποχή όλος ο κόσμος έχει επικεντρωθεί στα δίκτυα τρίτης γενιάς (3G), υπάρχουν συστήματα πρώτης γενιάς που χρησιμοποιούνται ακόμα και σήμερα. Ωστόσο, σε χώρες με προχωρημένη τηλεπικοινωνιακή υποδομή, τα συστήματα της πρώτης γενιάς σύντομα ή αν δεν έχει ήδη γίνει, θα κλείσουν καθώς σπαταλούν πολύτιμο φάσμα συχνοτήτων που θα μπορούσε να χρησιμοποιηθεί πιο αποτελεσματικά σε καινούργια ψηφιακά δίκτυα. [4]

### 1.2.3 Δεύτερη Γενιά (2G)

Η δεύτερη γενιά (2G) κυβελωτών συστημάτων χρησιμοποιεί ψηφιακή ραδιο μετάδοση της κίνησης. Για αυτό το λόγο, η διαχωριστική γραμμή μεταξύ πρώτης και δεύτερης γενιάς συστημάτων είναι προφανής: είναι ο διαχωρισμός αναλογικού/ψηφιακού. Τα δίκτυα δεύτερης γενιάς (2G) έχουν περισσότερες δυνατότητες από τα συστήματα της πρώτης γενιάς. Ένα κανάλι συχνοτήτων χωρίζεται ταυτόχρονα σε αρκετούς χρήστες (είτε με διαίρεση κώδικα είτε με διαίρεση χρόνου). Επίσης, έχουμε ιεραρχικές δομές κελιών, στα οποία η περιοχή κάλυψης χωρίζεται σε macrocells, microcells και picocells επαυξάνοντας ακόμη περισσότερο τις δυνατότητες του δικτύου.

Υπάρχουν τέσσερα κύρια πρότυπα για τα συστήματα δεύτερης γενιάς: είναι το Global System for Mobile (GSM) και τα παράγωγα του:

- Digital AMPS (D-AMPS)
- Code-division multiple access (CDMA) IS-95
- Personal digital cellular (PDC)



Το GSM είναι κατά πολύ το πιο επιτυχημένο και ευρέως χρησιμοποιημένο σύστημα της δεύτερης γενιάς. Αρχικά σχεδιάστηκε ως ένα πρότυπο ευρωπαϊκό το οποίο γρήγορα υιοθετήθηκε σε ολόκληρο τον κόσμο. Μόνο στην Αμερική δεν έχει επικρατήσει το GSM. Στην Βόρειο Αμερική, το Personal Communication System-1900 (PCS-1900 ή αλλιώς GSM-1900) έχει κερδίσει κάποιο έδαφος ενώ στη Νότια Αμερική και στη Χιλή έχει ευρεία κάλυψη το πρότυπο GSM. Παρόλο αυτά, το 2001 στη Βόρειο Αμερική η κοινότητα του Time Division multiple Access (TDMA) αποφάσισε να υιοθετήσει το wideband CDMA (WCDMA) σύστημα στο Third Generation Partnership Project (3GPP). Το WCDMA υποστηρίζει 3G τεχνολογία. Ως ενδιάμεση λύση μέχρι να ξεκινήσει να λειτουργεί το σύστημα WCDMA, πολλά συστήματα IS-136 άλλαξαν σε GSM/GPRS (General Packet Radio Service).

Το βασικό GSM σύστημα χρησιμοποιεί τη ζώνη συχνοτήτων των 900MHz αλλά υπάρχουν αρκετά παράγωγα του από τα οποία δυο είναι τα πιο σημαντικά. Το πρώτο είναι το Digital Cellular System 1800 (γνωστό ως GSM-1800) και το δεύτερο είναι το GSM-1900. Το GSM-1900 χρησιμοποιείται μόνο στη Βόρειο Αμερική και Χιλή ενώ το GSM-1800 χρησιμοποιείται σε άλλες περιοχές σε ολόκληρο τον κόσμο. Ο πρωταρχικός λόγος για την καινούργια ζώνη συχνοτήτων ήταν η έλλειψη χωρητικότητας στη ζώνη των 900MHz. Τελευταία, η European Telecommunications Standards Institute (ETSI) έχει αναπτύξει τα πρότυπα GSM-400 και GSM-800. [4]

Όσο αφορά τα κινητά τηλέφωνα, είναι πιο ελαφριά (100-200 γραμμάρια) με μικρότερες μπαταρίες. Επίσης, έχουν καλύτερη ποιότητα ήχου και καλύτερη προστασία από την ακτινοβολία. Οι υπηρεσίες που προσφέρει αυτή η γενιά είναι η υποστήριξη αποστολής σύντομων μηνυμάτων (SMS), αμφίδρομα δεδομένα μέχρι και 9.6 kbps, fax και εκπομπή κυψέλης (Cell Broadcast). [5]

### 1.2.4 Τρίτη Γενιά (3G)

Η γρήγορη ανάπτυξη των κινητών τηλεπικοινωνιών ήταν μια από τις πιο αξιοσημείωτες επιτυχημένες ιστορίες την δεκαετία του '90. Τα δίκτυα 2G άρχισαν να λειτουργούν στην αρχή της δεκαετίας (το πρώτο GSM δίκτυο λειτούργησε το 1991 στη Φιλανδία) και συνεχίζουν να εξαπλώνονται και να εξελίσσονται. Το Σεπτέμβριο του 2002 υπήρχαν 460 GSM δίκτυα παγκοσμίως και όλα μαζί εξυπηρετούσαν 747.5 εκατομμύρια συνδρομητές. Την ίδια χρονιά το ίδρυμα ETSI είχε ήδη ξεκινήσει την προτυποποίηση για την επόμενη γενιά δικτύων κινητής τηλεπικοινωνίας. Το καινούργιο σύστημα ονομάστηκε Universal Mobile Telecommunications System (UMTS). Η ανάπτυξη του 3G δεν έγινε μόνο από το ETSI. Υπήρξαν και πολλοί άλλοι οργανισμοί και ερευνητικά προγράμματα που είχαν τον ίδιο σκοπό. [4]

Ο βασικός στόχος της ανάπτυξης των κινητών δικτύων τρίτης γενιάς είναι η παροχή των κινητών υπηρεσιών «οπουδήποτε» και «κάθε στιγμή». Αυτό σημαίνει ότι ένας χρήστης κινητών δικτύων τρίτης γενιάς μπορεί να μετακινείται οπουδήποτε και να εξυπηρετείται ακόμα και σε περιοχές όπου δεν υπάρχει κάλυψη από συστήματα τρίτης γενιάς αλλά υπάρχουν άλλου είδους ασύρματα δίκτυα. Για την ακρίβεια, ο χρήστης θα μπορεί να εξυπηρετείται από οικιακά ασύρματα συστήματα, από άλλα κυβελωτά κινητά δίκτυα καθώς και από δορυφορικά δίκτυα. [3]

Επιπλέον, οι παρεχόμενες υπηρεσίες επεκτείνονται σε υπηρεσίες διαδικτύου και σε υπηρεσίες πολυμέσων με υψηλούς ρυθμούς μετάδοσης (προβλέπονται ρυθμοί που ξεκινούν από τα 144 Kbps και φτάνουν ακόμα και σε ρυθμούς της τάξης των Mbps). Με τον όρο υπηρεσίες πολυμέσων αναφερόμαστε σε υπηρεσίες κατά τις οποίες υπάρχει συνδυασμός εικόνας, ήχου και κειμένου σε ένα διαρκώς μεταβαλλόμενο ψηφιακό περιβάλλον. Τέλος, θα πρέπει να αναφερθούν τα επικρατέστερα, προς το παρόν, συστήματα τρίτης γενιάς τα οποία είναι: το UMTS (Ευρώπη), το CDMA2000 (Βόρεια Αμερική) και το NTT Docomo (Ιαπωνία). [3]

### 1.2.5 Τέταρτη Γενιά (4G)

Τα κινητά τηλέφωνα 4<sup>ης</sup> γενιάς είναι το μέλλον της κινητής τηλεφωνίας. Τα χαρακτηριστικά της 4<sup>ης</sup> γενιάς κινητών τηλεφώνων είναι η IP λύση που δίνεται όπου φωνή, δεδομένα και πολυμέσα θα παρέχονται οποτεδήποτε και οπουδήποτε. Η τέταρτη γενιά έχει αρχίσει ήδη να εμφανίζεται σιγά σιγά (ήδη κάνει τα πρώτα της βήματα στις Η.Π.Α, τις σκανδιναβικές χώρες, την Νοτιοανατολική Ασία και

Ιαπωνία). Οι χρήστες κινητών τηλεφώνων θα έχουν πιο γρήγορη πρόσβαση στο Διαδίκτυο, γεγονός που αναμένεται να δώσει μια νέα δυναμική στην αγορά των κινητών τηλεφωνίας τόσο σε υπηρεσίες (μετεωρολογικές προβλέψεις, GPS) όσο και σε περιεχόμενο (ειδήσεις, ραδιόφωνο, παιχνίδια). Τα κινητά τηλέφωνα αυτής της γενιάς μπορούμε να πούμε ότι μοιάζουν περισσότερο με μικρούς υπολογιστές με απεριόριστες δυνατότητες. Οι επεξεργαστές τους φτάνουν σε συχνότητα το 1GHz, οι οθόνες τους είναι 3 και 4 ιντσών και διαθέτουν υψηλή ποιότητα και ευκρίνεια, οι κάμερες ξεπερνούν τα 5 Megapixels, η σύνδεση στο διαδίκτυο γίνεται με κάθε δυνατό τρόπο, υπάρχει σύγκλιση μεταξύ ασύρματης και ενσύρματης τεχνολογίας, η πρόσβαση στις ιστοσελίδες κοινωνικής διαδικτύωσης (Facebook και Twitter) έχει τα χαρακτηριστικά της αμεσότητας και της ταχύτητας. Τέλος, μπορεί να παρέχει ασύρματη ευρυζωνική πρόσβαση, Multimedia Messaging Service, video chat, mobile TV, υψηλής πιστότητας τηλεοπτικό περιεχόμενο (HDTV) και ψηφιακή μετάδοση video (DVB) και υψηλή ασφάλεια. [2]

## Ενότητα 2<sup>η</sup>

### 2.1 Εισαγωγή στη Γλώσσα Αντικειμενοστραφούς

#### Προγραμματισμού JAVA

Στις αρχές του 1991, στη Sun αναζητούσαν το κατάλληλο εργαλείο για να αποτελέσει την πλατφόρμα ανάπτυξης λογισμικού σε μικρο-συσκευές (έξυπνες οικιακές συσκευές έως πολύπλοκα συστήματα παραγωγής γραφικών) με αποτέλεσμα το 1995 η Sun Microsystems να παρουσιάσει μια νέα γλώσσα αντικειμενοστραφούς προγραμματισμού, την Java, η οποία σχεδιάστηκε αρχικά από τον James Gosling που θεωρείται ο «πατέρας» της Java.

Η Java αρχικά είχε προταθεί σαν μια τεχνολογία για τον εμπλουτισμό ιστοθέσεων με προγράμματα τα οποία εκτελούνταν σε εφαρμογές περιήγησης (web browsers). Είναι η πρώτη που κατάφερε να συμπεριλάβει ήχο και κίνηση σε μια ιστοσελίδα και επιπλέον επιτρέπει στους χρήστες να αλληλεπιδρούν (interact) με την ιστοσελίδα.

Σήμερα είναι πιο πιθανό να τη βρει κανείς σε servers, να αποτελεί τη βάση δυναμικών Web εφαρμογών που υποστηρίζονται από σχεσιακές βάσεις δεδομένων σε ορισμένες από τις μεγαλύτερες ιστοθέσεις στο Web.

Αυτό που κάνει ανταγωνιστική την Java είναι ότι μπορεί να εφαρμοστεί σε οποιονδήποτε browser.

### ***Δυνατότητες της JAVA***

- Ήχος ο οποίος εκτελείται όποτε ο χρήστης φορτώνει μία σελίδα
- Μουσική που παίζει στο background μιας σελίδας
- Δημιουργία κινουμένων σχεδίων
- Βίντεο
- Παιχνίδια με πολυμέσα

### ***Πλεονεκτήματα της JAVA***

- Είναι απλή.
  - Είναι αντικειμενοστραφής, δηλαδή τα πάντα στη Java είναι είτε κλάση, είτε μέθοδος ή αντικείμενο.
  - Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουν ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Αυτό επιτεύχθηκε με την ανάπτυξη της Εικονικής Μηχανής (Virtual Machine ή VM ).
  - Είναι ασφαλής.
  - Είναι πολυνηματική, δηλαδή ένα απλό πρόγραμμα σε Java μπορεί να κάνει πολλά, διαφορετικά προγράμματα ανεξάρτητα και αλληλεπιδρώντα.
- [10],[11],[12]

## 2.2 Εισαγωγή στη Java 2 Platform

Η Java 2 Platform έχει τρία χαρακτηριστικά στοιχεία:

1. Η γλώσσα προγραμματισμού Java είναι συντακτικά ίδια με την C++ αλλά διαφέρει σε βασικά στοιχεία. Η C++ χρησιμοποιεί μη ασφαλείς δείκτες (pointers) και οι προγραμματιστές είναι υπεύθυνοι να κατανέμουν και να ελευθερώνουν την μνήμη. Αντιθέτως, η Java χρησιμοποιεί ασφαλή τύπου στοιχείων αναφορών (object references) και η μνήμη που δεν χρησιμοποιείται αυτόματα αναμορφώνεται. Επιπλέον, η γλώσσα προγραμματισμού Java αποφεύγει την πολλαπλή κληρονομικότητα χρησιμοποιώντας interfaces για καλύτερη δομή του προγράμματος.
2. Η Εικονική Μηχανή (Virtual Machine) θέτει τα «θεμέλια» της πλατφόρμας της Java. Η αρχιτεκτονική προσφέρει μερικά ελκυστικά χαρακτηριστικά: Η εικονική μηχανή μπορεί να υλοποιηθεί για να «τρέξει» για μια ποικιλία από λειτουργικά συστήματα και hardware. Επίσης, προσφέρει πολύ καλό έλεγχο για ασφαλή εκτέλεση του κώδικα.
3. Τέλος, τα APIs (Application Programming Interfaces) βρίσκονται εκτός της πλατφόρμας της Java και μπορούν να υποστηρίξουν σχεδόν οποιαδήποτε εφαρμογή που θέλουμε. Για παράδειγμα από το user interface μέχρι την κρυπτογραφία. [13]

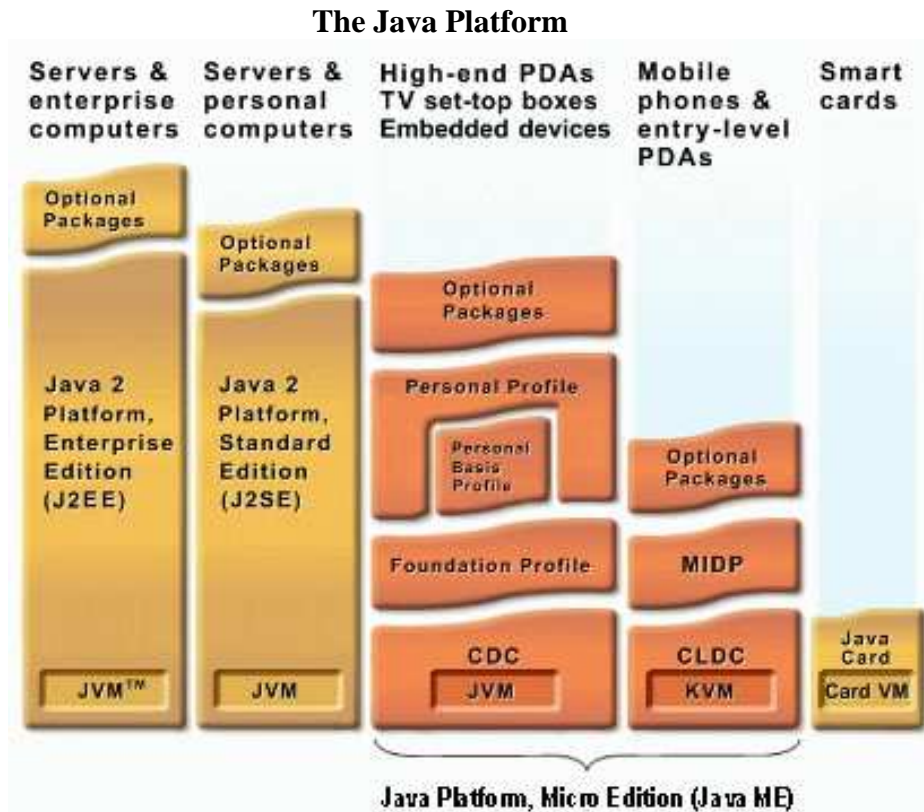
Ο συνδυασμός της γλώσσας προγραμματισμού Java, της Εικονική Μηχανή και των Java APIs συνθέτουν την Java πλατφόρμα. Επιπλέον, η Java πλατφόρμα είναι σχεδιασμένη να περιγράφει μια μεγάλη γκάμα hardware υπολογιστών, από έξυπνες κάρτες (smart cards) μέχρι και enterprise servers.

Επομένως, η Java πλατφόρμα χωρίζεται σε τρεις τεχνολογίες:

- Java 2, Standard Edition (J2SE) έχει σχεδιαστεί για desktop υπολογιστές. Συνήθως «τρέχει» σε OS X, Linux, Solaris, Microsoft Windows
- Java 2, Enterprise Edition (J2EE) αφορά εταιρικές εφαρμογές. Έχει βασιστεί στην J2SE έχοντας όμως προσθέσει APIs για server-side computing.
- Java 2, Micro Edition (J2ME) είναι ένα σύνολο τεχνολογιών και προδιαγραφών που αναπτύχθηκε για μικρές συσκευές όπως κινητά τηλέφωνα, κτλ. Η J2ME χρησιμοποιεί υποσύνολα από τα στοιχεία της J2SE όπως μικρότερες εικονικές

μηχανές και APIs. [13] Για αυτό θα περιοριστούμε στην λεπτομερή ανάπτυξη μόνο της J2ME.

**Εικόνα 1:** *Java Πλατφόρμα (Βιβλιογραφία:[14])*



## 2.3 Τεχνολογία J2ME

Η τεχνολογία Java ME δημιουργήθηκε αρχικά για να αντιμετωπίσει τους περιορισμούς που συνδέονταν με την ανάπτυξη εφαρμογών για μικρές συσκευές. Για αυτό τον λόγο, η Sun δημιούργησε την Java ME τεχνολογία με χαρακτηριστικά που μπορούν να αποδώσουν σε περιορισμένο περιβάλλον και να αναπτυχθούν Java εφαρμογές που να «τρέχουν» σε μικρές συσκευές με περιορισμένη μνήμη και δυνατότητες προβολής και ισχύος (display και power capacity). [16]

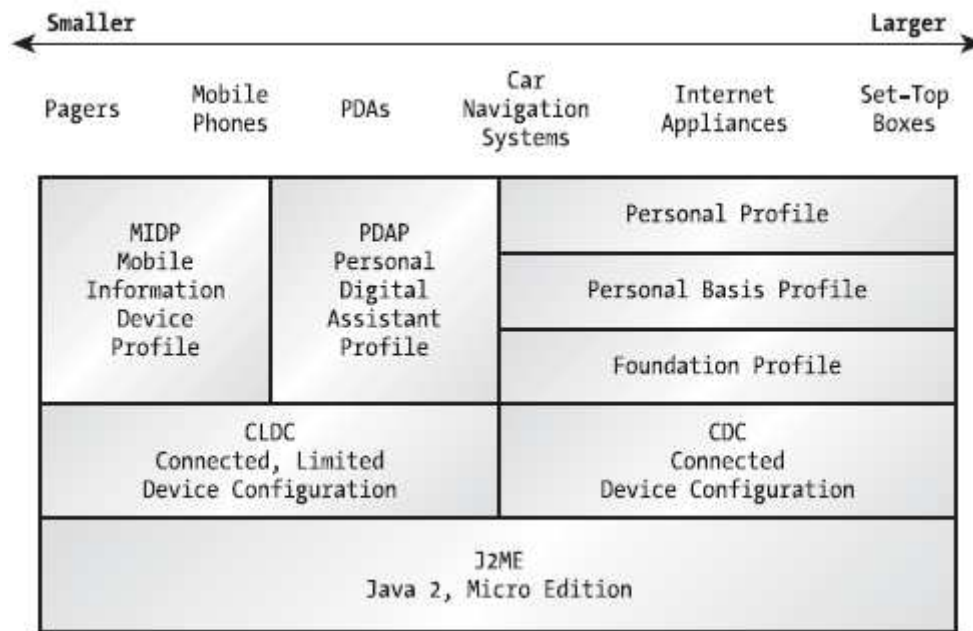
Η J2ME επιτρέπει σε χρήστες να «κατεβάσουν» εφαρμογές λογισμικού (software applications) - που συνήθως λέγονται MIDlets- σε συσκευές όπως κινητά τηλέφωνα και PDAs. Οι Java εφαρμογές για κινητές συσκευές συμπεριλαμβάνουν παιχνίδια, εργαλεία για εμπορικές εφαρμογές και Διαδίκτυο (business tools και Internet utilities). [17]

Η J2ME χωρίζεται σε *configurations*, *profiles* και *optional APIs*, τα οποία προσφέρουν συγκεκριμένες πληροφορίες για τα APIs και διαφορετικές οικογένειες συσκευών.

- **Configuration (Διαμόρφωση)** έχει σχεδιαστεί για ένα συγκεκριμένο μοντέλο συσκευής που βασίζεται σε περιορισμένη μνήμη και ισχύ επεξεργαστή (processor power). Καθορίζει την Εικονική Μηχανή που μπορεί εύκολα να εγκατασταθεί σε συσκευές που υποστηρίζουν το configuration. Επιπλέον, ορίζει ένα αυστηρό υποσύνολο της Java 2 Platform, Standard Edition (J2SE) APIs που θα χρησιμοποιηθεί στην πλατφόρμα, και επιπρόσθετα APIs που μπορεί να είναι απαραίτητα. Οι κατασκευαστές μικρών συσκευών είναι υπεύθυνοι για την εγκατάσταση ενός συγκεκριμένου configuration στις συσκευές τους.
- **Profile (Προφίλ)** είναι πιο συγκεκριμένα από τα configurations. Ένα profile βασίζεται σε configuration και παρέχει επιπλέον APIs όπως user interface, persistent storage και οτιδήποτε άλλο χρειάζεται για να σχεδιάσει και να αναπτύξει εφαρμογές που να «τρέχουν» σε συσκευή.
- **Optional (επιλεκτικά) APIs** ορίζουν επιπλέον συγκεκριμένη λειτουργικότητα που μπορεί να συμπεριληφθεί σε ένα συγκεκριμένο configuration (or profile). Όλη η συλλογή που περιλαμβάνει configuration, profile και optional APIs και που υλοποιείται σε μια συσκευή ονομάζεται *stack* (στοίβα). Για παράδειγμα, μια πιθανή μελλοντική stack συσκευή μπορεί να είναι CLDC / MIDP + Mobile Media API. [15]

Στην παρακάτω εικόνα βλέπουμε ένα πίνακα με τα πιο συνηθισμένα configurations και profiles στη J2ME. Παρατηρούμε για κάθε συσκευή από την πιο μικρή μέχρι την πιο μεγάλη ποια configurations και ποια profiles χρησιμοποιούνται. Παρακάτω θα τα δούμε και πιο αναλυτικά. [15]

**Εικόνα 2:** *Common J2ME profiles and configurations*



**Πίνακας 1.** Παρακάτω βλέπουμε μερικά από *J2ME Configurations, Profiles, and Optional APIs* που χρησιμοποιούνται για την ανάπτυξη εφαρμογών.

Τα configurations και profiles πρωτοεμφανίζονται ως Java Specification Requests (JSRs). Μπορούμε να δούμε όλα τα JSRs στο site: <http://jcp.org/jsr/all/>

<b>Configurations</b>	
<b>JSR</b>	<b>Name</b>
30	Connected, Limited Device Configuration (CLDC) 1.0
139	Connected, Limited Device Configuration (CLDC) 1.1
36	Connected Device Configuration 1.0.1
218	Connected Device Configuration 1.1
<b>Profiles</b>	
<b>JSR</b>	<b>Name</b>
37	Mobile Information Device Profile 1.0
118	Mobile Information Device Profile 2.0
75	PDA Profile 1.0
46	Foundation Profile 1.0
129	Personal Basis Profile 1.0
<b>Optional APIs</b>	
<b>JSR</b>	<b>Name</b>
75	PDA Optional Packages for J2ME



82	Java APIs for Bluetooth
135	Mobile Media API 1.1
184	Mobile 3D Graphics

Πριν συνεχίσουμε θα πρέπει να σημειώσουμε ότι τα configurations και τα profiles στη J2ME καθορίζονται κυρίως με γνώμονα την χωρητικότητα μνήμης. Συνήθως, ορίζεται μια ελάχιστη χωρητικότητα μνήμης στη ROM και RAM. Για μικρές συσκευές, έχουμε δυο είδη μνημών: Η πρώτη είναι **volatile (ασταθής) μνήμη** που σημαίνει ότι οι συσκευές δεν έχουν σταθερή μνήμη. Σε αυτήν την περίπτωση, η volatile μνήμη δεν διατηρεί τις πληροφορίες όταν απενεργοποιηθεί η συσκευή. Η δεύτερη είναι η **nonvolatile (σταθερή) μνήμη** όπου η μνήμη παραμένει σταθερή. Αυτό σημαίνει ότι, η nonvolatile μνήμη μπορεί να αποθηκεύσει τις πληροφορίες και να τις κρατήσει ανέπαφες όταν ενεργοποιήσουμε και απενεργοποιήσουμε την συσκευή. [15]

### **Configurations**

Configurations (ή αλλιώς διαμορφώσεις) καθορίζουν μια Εικονική Μηχανή και δίνουν λεπτομέρειες για ένα βασικό σύνολο από APIs που μπορούν να χρησιμοποιηθούν με συγκεκριμένες συσκευές. Μέχρι και σήμερα υπάρχουν δυο κατηγορίες συσκευών: (i) Connected Device Configuration (CDC) και (ii) Connected Limited Device Configuration (CLDC). [15]

### **Connected Device Configuration (CDC)**

Μια συσκευή CDC έχει το μίνιμουμ 512 KB read-only memory (ROM) , 256 KB random-access memory (RAM) και μία σύνδεση δικτύου. Είναι υποσύνολο της J2SE, που περιλαμβάνει όλες σχεδόν τις βιβλιοθήκες που δεν σχετίζονται με GUI. Το CDC έχει σχεδιαστεί για συσκευές όπως τηλεόραση set-top boxes, συστήματα πλοήγησης αυτοκινήτου και high-end PDAs. [15]

Η διαμόρφωση CDC έχει τρία διαφορετικά ορισμένα προφίλ (profiles)

- The Foundation Profile (JSR 219)
- The Personal Basis Profile (JSR 217) και
- The Personal Profile (JSR 216) [16]

### ***Connected Limited Device Configuration (CLDC)***

Η CLDC είναι η διαμόρφωση που θα ασχοληθούμε με αυτήν την εργασία γιατί περιλαμβάνει κινητά τηλέφωνα, pagers, PDAs και άλλες συσκευές αυτού του μεγέθους. Η CLDC απευθύνεται σε μικρότερες συσκευές από αυτές που απευθύνονται στη CDC. Οι συσκευές CLDC έχουν περιορισμένη μνήμη, περιορισμένο display, περιορισμένη CPU power, περιορισμένο display size, περιορισμένο input, περιορισμένη διάρκεια ζωής της μπαταρίας και περιορισμένη σύνδεση δικτύου.

Η διαμόρφωση CLDC έχει σχεδιαστεί για συσκευές με 160 KB μέχρι 512 KB συνολικής μνήμης, συμπεριλαμβανομένου ένα μίνιμουμ με 160 KB ROM και 32 KB RAM που είναι διαθέσιμο για την Java πλατφόρμα.

Η υλοποίηση της CLDC διαμόρφωση βασίζεται σε μια μικρή εικονική μηχανή της (JVM) που λέγεται KVM (Kilo Virtual Machine). Η KVM είναι ένα ολοκληρωμένο περιβάλλον λειτουργίας (Java Runtime Environment - JRE ) που σχεδιάστηκε για μικρές συσκευές. Έχει μικρό μέγεθος δηλαδή της τάξης 40-80K, το οποίο είναι κατάλληλο για κινητά, pagers, και PDAs. Το KVM μπορεί να «τρέξει» σε οποιοδήποτε σύστημα που έχει 16bit/32bit επεξεργαστή και 160-512K συνολικής μνήμης.

Η KVM δηλαδή η εικονική μηχανή της J2ME δεν υποστηρίζει ορισμένα χαρακτηριστικά όπως η JVM στη J2SE.

- Δεν υποστηρίζει αριθμούς κινητής υποδιαστολής.
  - Η Object Finalization δεν είναι διαθέσιμη στη CLDC και κατά επέκταση στο MIDP. Αυτό σημαίνει ότι δεν υπάρχει μέθοδος Object.finalize().
  - Η CLDC δεν υποστηρίζει Reflection API.
  - Δεν μπορούν να προστεθούν native methods στο ολοκληρωμένο περιβάλλον λειτουργίας. Όλη η λειτουργικότητα των native methods είναι τοποθετημένη μέσα στη KVM. Οι εφαρμογές δεν μπορούν να ορίσουν καινούργιες native methods.
  - Η CLDC/MIDP δεν υποστηρίζει thread groups και daemon threads.
  - Η KVM συμπεριλαμβάνει μόνο ένα υποσύνολο από standard bytecode verifier. Αυτό σημαίνει ότι η εργασία για την επιβεβαίωση κλάσεων έχει χωριστεί ανάμεσα στη CLDC συσκευή και σε εξωτερικούς μηχανισμούς. Αυτό όμως, δημιουργεί σοβαρά προβλήματα - επιπτώσεις στην ασφάλεια.
- [15],[18]

### **Profiles (Προφίλ)**

Ένα προφίλ δημιουργείται με βάση μια διαμόρφωση αλλά προσθέτει πιο συγκεκριμένα APIs για να δημιουργήσει ένα ολοκληρωμένο περιβάλλον λειτουργίας εφαρμογών. [6]

### **Profiles που χρησιμοποιούνται σήμερα**

Υπάρχουν αρκετά διαφορετικά προφίλ που έχουν αναπτυχθεί από την Java Community Process. Μερικά από αυτά είναι το Foundation Profile, το *Personal Basis Profile*, το *Personal Profile* και το *PDA Profile*.

Το *Foundation Profile* είναι μια διαμόρφωση για συσκευές που μπορούν να υποστηρίξουν ένα περιβάλλον δικτύου στην J2ME. Δεν υποστηρίζει user interface αλλά με την πρόσθεση άλλων προφίλ με διαδοχικά επίπεδα πάνω από το Foundation Profile μπορεί να υποστηρίξει user interface και άλλες λειτουργικότητες.

Πάνω από το Foundation Profile ιεραρχικά βρίσκονται το *Personal Basis Profile* και το *Personal Profile*. Ο συνδυασμός του CDC + Foundation Profile + Personal Basis Profile + Personal Profile έχει σχεδιαστεί για την επόμενη γενιά του PersonalJava application runtime environment.

Το *PDA Profile* (PDAP) το οποίο έχει σχεδιαστεί σε CLDC, έχει σχεδιαστεί για palmtop συσκευές με μίνιμουμ 512KB σε συνδυασμό ROM και RAM μαζί (και maximum 16MB). [15]

### **Mobile Information Device Profile (MIDP)**

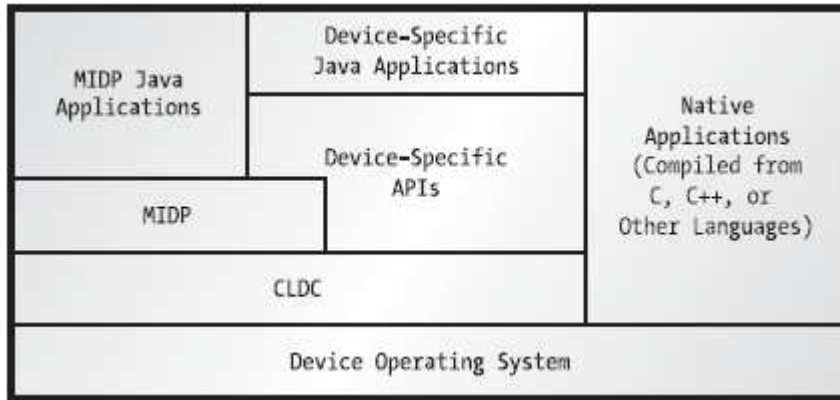
Σε αυτή την εργασία θα επικεντρωθούμε στο Mobile Information Device Profile (MIDP) που απευθύνεται σε κινητά τηλέφωνα. Το MIDP έχει τα παρακάτω χαρακτηριστικά:

- Να έχει μίνιμουμ 256KB ROM μνήμη για την υλοποίηση MIDP.
- Να έχει μίνιμουμ 128KB RAM μνήμη για Java runtime heap.
- Πρέπει να έχει μίνιμουμ 8KB nonvolatile εγγράψιμη μνήμη για persistent data
- Η οθόνη πρέπει να είναι τουλάχιστον 96x54 pixels.
- Να υπάρχει χωρητικότητα εισόδου δεδομένων από τον χρήστη, είτε με keypad, είτε με πληκτρολόγιο ή με οθόνη αφής.
- Να υπάρχει αμφίδρομη σύνδεση δικτύου. [15]

Υπάρχουν δύο εκδόσεις MIDP

- MIDP 1.0 (JSR 37)
- MIDP 2.0 (JSR 118)

**Εικόνα 3:** *MIDP Software Components*



### **Λειτουργία MIDP**

Οι περιοχές που καλύπτει το MIDP είναι οι παρακάτω:

- Διαχείριση κύκλου ζωής μιας εφαρμογής (κλάσεις και μέθοδοι για έναρξη, παύση και καταστροφή εφαρμογών).
- User interface και γεγονότα (κλάσεις και interfaces για δημιουργία αντικειμένων GUI).
- Συνδεσιμότητα δικτύου (επεκτείνει τις κλάσεις συνδεσιμότητας του CLDC για να επιστρέψει συνδέσεις HTTP).
- Αποθήκευση δεδομένων στη συσκευή (υλοποιεί σύστημα διαχείρισης βάσεων δεδομένων με χρήση εγγραφών). [6]

### **Χαρακτηριστικά των δυο εκδόσεων MIDP**

Η MIDP 2.0 είναι συμβατή με την MIDP 1.0. Καταρχήν, όλες οι τωρινές διαθέσιμες συσκευές υποστηρίζουν MIDP 2.0 και όλες οι καινούργιες συσκευές που θα διατεθούν στην αγορά επίσης θα υποστηρίζουν MIDP 2.0. Επίσης, η MIDP 2.0 υποστηρίζει multimedia εφαρμογές, ένα καινούργιο παιχνίδι user interface API και σύνδεση HTTPS. Επιπλέον, η MIDP 2.0 βελτιώνει και ενισχύει το προφίλ με διάφορους τρόπους (threading, ασφάλεια, κλπ.). Τέλος, οι εφαρμογές που γράφονται σε MIDP ονομάζονται Midlets. [6],[15]

### ***Πλεονεκτήματα MIDP***

Το MIDP εμφανίζεται σε μια κρίσιμη στιγμή που οι συσκευές όπως για παράδειγμα τα κινητά τηλέφωνα εξελίσσονται πολύ γρήγορα και κατακτούν όλο και περισσότερο την αγορά. Οι συσκευές MIDP καταφέρνουν να συνδυάσουν την ισχύ του επεξεργαστή (processing power), τη διαθέσιμη μνήμη και την σύνδεση Διαδικτύου (Internet). Αυτό έχει ως αποτέλεσμα να παρουσιάσουν μια ελκυστική πλατφόρμα για κινητά τηλέφωνα με εφαρμογές δικτύου.

Ένα από τα πλεονεκτήματα που δημιουργείται με την χρησιμοποίηση της Java έναντι άλλων εργαλείων για την ανάπτυξη εφαρμογών για μικρές συσκευές είναι η *φορητότητα* (portability). Μπορούμε να γράψουμε εφαρμογές με γλώσσες προγραμματισμού όπως η C ή C++ αλλά το αποτέλεσμα θα ήταν να χρησιμοποιηθεί η εφαρμογή αυτή για μια μόνο πλατφόρμα. Αντιθέτως, μια εφαρμογή που έχει γραφτεί με MIDP APIs είναι αμέσως φορητή δηλαδή μπορεί να χρησιμοποιηθεί για οποιαδήποτε MIDP συσκευή.

Τέλος, άλλο ένα πλεονέκτημα που χρησιμοποιούμε Java για μικρές συσκευές είναι η ασφάλεια. Η Java είναι γνωστή για την ικανότητα της να «τρέχει» downloaded code με ασφάλεια όπως τα applets. [15]

### ***MIDP Packages***

Τα APIs που είναι διαθέσιμα σε μια MIDP εφαρμογή προέρχονται από packages του CLDC και του MIDP. Το CLDC ορίζει τον πυρήνα των APIs που προέρχεται κυρίως από τη J2SE. [15]

### ***Πίνακας 2: MIDP packages***

<b>CLDC 1.1</b>	<b>MIDP 2.0</b>
java.io	javax.microedition.lcdui
java.lang	javax.microedition.midlet
java.security	javax.microedition.rms
java.util	

Στον παραπάνω πίνακα παρατηρούμε τις κυριότερες βιβλιοθήκες του CLDC 1.1 (JSR 139) και του MIDP 2.0 (JSR 118) που χρησιμοποιούμε στη J2ME.

Παρακάτω θα αναλύσουμε περιληπτικά που χρησιμοποιείται η κάθε μια βιβλιοθήκη.

### Από τη CLDC 1.1

**java.io:** Εξασφαλίζει για το σύστημα την είσοδο και έξοδο δεδομένων μέσω των ροών δεδομένων (data streams), serialization και file system. [20]

**java.lang:** Εξασφαλίζει κλάσεις οι οποίες είναι θεμελιώδεις για το σχεδιασμό της γλώσσας προγραμματισμού Java. Οι πιο σημαντικές κλάσεις είναι τύπου Object. Αυτές αποτελούν την κύρια βάση της ιεραρχίας των κλάσεων και τις διασυνδέσεις κλάσεων (interfaces) οι οποίες αντιπροσωπεύουν κλάσεις σε run time.[21]

**java.util:** Περιέχει συλλογές πλαισίου εργασίας (framework), legacy collection classes, event model, ευκολίες για τον ορισμό ημερομηνίας & ώρας, διεθνοποίηση (internationalization) και κλάσεις διαφόρων χρήσεων (miscellaneous utility classes). [22]

**java.security:** Εξασφαλίζει κλάσεις και διασυνδέσεις (interfaces) για την ασφάλεια του πλαισίου εργασίας (security framework). [23]

### Από το MIDP 2.0

**javax.microedition.lcdui:** Το UI API εξασφαλίζει ένα σετ από χαρακτηριστικά για την υλοποίηση των διασυνδέσεων χρηστών (user interfaces) για εφαρμογές MIDP. [25]

**javax.microedition.midlet:** Τα MIDlet packages ορίζουν τις εφαρμογές MIDP και τις αλληλεπιδράσεις μεταξύ της εφαρμογής και του περιβάλλοντος στην οποία «τρέχει» η εφαρμογή. Ένα MIDlet είναι μια εφαρμογή MIDP. Ο κύκλος ζωής ενός MIDlet ορίζει το πρωτόκολλο μεταξύ ενός MIDlet και του περιβάλλοντος του. [19]

**javax.microedition.rms:** Το Mobile Information Device Profile εξασφαλίζει ένα μηχανισμό για MIDlets για τη συνεχή αποθήκευση δεδομένων και για να μπορεί να γίνει μετά η ανάκτηση τους. Αυτός ο συνεχής μηχανισμός αποθήκευσης έχει μοντελοποιηθεί ως μία απλή βάση δεδομένων record oriented (simple record oriented database) και έχει ονομαστεί the Record Management System. [24]

## 2.4 MIDlets

Όπως σ' όλα τα MIDlets, χρησιμοποιείται η βιβλιοθήκη `javax.microedition.midlet.MIDlet` η οποία είναι η βασική κλάση για όλες τις εφαρμογές MIDP. Επίσης, υπάρχει ειδικό λογισμικό στη συσκευή που ονομάζεται Java Application Manager (JAM) ή Application Management Software (AMS) που επιτρέπει στον χρήστη να ελέγχει την διαδικασία εγκατάστασης, όταν «τρέχει» η εφαρμογή και όταν γίνεται αφαίρεση των MIDlets. Όταν ο χρήστης επιλέξει να «τρέξει» μια εφαρμογή MIDlet, είναι το λογισμικό JAM που δημιουργεί ένα instance της κλάσης MIDlet και με το οποίο τρέχει τις μεθόδους πάνω σε αυτό.

Η σειρά των μεθόδων που θα καλεστούν στην υποκλάση του MIDlet ορίζεται από τον κύκλο ζωής του MIDlet. Το λογισμικό JAM θα καλέσει τις μεθόδους στην εφαρμογή MIDlet για να δηλώσει τις μεταβάσεις από τη μια κατάσταση στην άλλη. Οι μέθοδοι `startApp()`, `pauseApp()`, `destroyApp()` και ο constructor του MIDlet που «τρέχει» εκείνη την ώρα είναι όλα μέρος του κύκλου ζωής του MIDlet. [15]

### ***Ο κύκλος ζωής ενός MIDlet***

Οι MIDP εφαρμογές παρουσιάζονται μέσω instances (στιγμιότυπων) της κλάσης `javax.microedition.midlet.MIDlet`. Τα MIDlets έχουν ένα συγκεκριμένο κύκλο ζωής που απεικονίζουν τις μεθόδους και την συμπεριφορά της κλάσης MIDlet.

Ο application manager είναι εκείνο που ελέγχει την εγκατάσταση, την εκτέλεση και το κύκλο ζωής των MIDlets. Τα MIDlets δεν έχουν καθόλου πρόσβαση στο application manager. Γίνεται εγκατάσταση ενός MIDlet μετακινώντας τα αρχεία κλάσεων (class files) στη συσκευή. Τα αρχεία κλάσεων τοποθετούνται όλα μαζί σε ένα JAR (Java Archive) αρχείο. Επίσης, το συνοδεύει ένα άλλο αρχείο (με κατάληξη `.jad`) που περιγράφει τα περιεχόμενα του JAR αρχείου.

Ένα MIDlet εκτελεί τις επόμενες καταστάσεις:

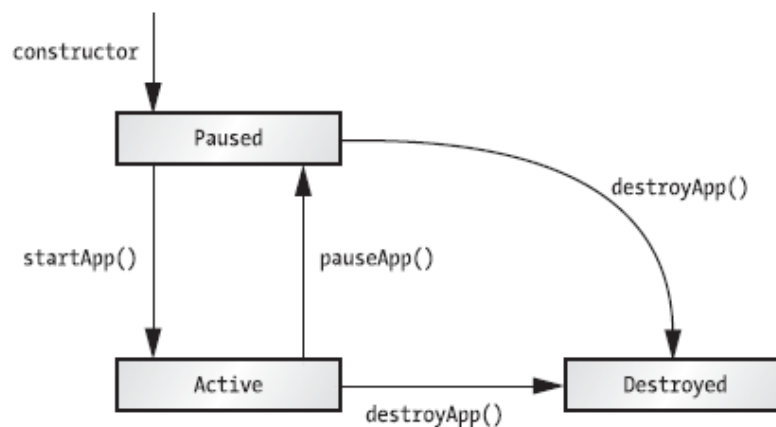
- Όταν το MIDlet είναι έτοιμο να εκτελεστεί, δημιουργείται ένα instance. Ο constructor του MIDlet εκτελείται και το MIDlet βρίσκεται σε Paused state.
- Στην επόμενη φάση, το MIDlet εισέρχεται σε Active state όταν ο application manager καλέσει τη `startApp()`.
- Ενώ το MIDlet είναι Active, ο application manager μπορεί να αναστείλει την εκτέλεση της εφαρμογής καλώντας `pauseApp()`. Αυτό έχει ως συνέπεια το

MIDlet να ξαναγυρίσει στη Paused state. Επίσης, ένα MIDlet μπορεί να θέσει αυτό καθεαυτό σε Paused state καλώντας τη μέθοδο `notifyPaused()`.

- Ενώ το MIDlet βρίσκεται σε Paused state, ο application manager μπορεί να καλέσει τη μέθοδο `startApp()` για να επαναφέρει το MIDlet πίσω στην Active state.
- Ο application manager μπορεί να τερματίσει την εκτέλεση του MIDlet καλώντας τη μέθοδο `destroyApp()`, όπου το MIDlet καταστρέφεται και υπομονετικά περιμένει για το garbage collection. Ένα MIDlet μπορεί να καταστρέψει αυτό καθεαυτό καλώντας τη μέθοδο `notifyDestroyed()`. [15]

Η παρακάτω εικόνα δείχνει τις καταστάσεις ενός MIDlet και τις μεταβιβάσεις καταστάσεων που γίνονται μεταξύ τους. Δηλαδή, το πέρασμα από τη μια κατάσταση σε μια άλλη κατάσταση. [15]

**Εικόνα 4:** Ο κύκλος ζωής ενός MIDlet





### **Βασικός κώδικας ενός MIDlet**

Το παρακάτω πρόγραμμα “Hello World” παρουσιάζει τη βασική δομή των MIDlets.

```

6  import javax.microedition.midlet.*;
   import javax.microedition.lcdui.*;
8
9
10 public class HelloWorld extends MIDlet {
11     public void startApp() {
12     }
13
14     public void pauseApp() {
15     }
16
17     public void destroyApp(boolean unconditional) {
18     }
19 }
20

```

### **JAR & JAD Αρχεία**

Για να μπορούμε να αναπτύξουμε μια εφαρμογή θα πρέπει τα αρχεία να είναι όλα συγκεντρωμένα σε ένα package. Αυτό το αρχείο ονομάζεται Java Archive (JAR).

Το JAR αρχείο έχει κατάληξη \*.jar. Κάθε JAR συμπεριλαμβάνει ένα manifest file, META-INF\MANIFEST.MF, το οποίο περιγράφει τα περιεχόμενα του αρχείου. Το manifest file πρέπει να περιλαμβάνει επιπλέον πληροφορίες. Οι επιπλέον πληροφορίες για το ολοκληρωμένο περιβάλλον MIDP είναι σημαντικές όπως το MIDlet class name και οι εκδόσεις των CLDC και MIDP. [15]

Για παράδειγμα, το αρχείο manifest.mf που βρίσκεται στο φάκελο META-INF του παραπάνω κώδικα “Hello World” περιέχει τις παρακάτω πληροφορίες οι οποίες είναι:

MIDlet-1: Midlet, , HelloWorld

MIDlet-Vendor: Vendor

MIDlet-Name: mobile

MIDlet-Version: 1.0

MicroEdition-Configuration: CLDC-1.1

MicroEdition-Profile: MIDP-2.0

Ένα JAR file μπορεί να έχει περισσότερες από μια MIDlet class το οποίο είναι γνωστό ως MIDlet suite.

### *Δημιουργώντας ένα MIDlet Descriptor*

Χρειαζόμαστε ένα επιπλέον αρχείο. Πρέπει να δημιουργηθεί ένα application descriptor file. Αυτό το αρχείο περιέχει πολλές όμοιες πληροφορίες με αυτές που βρίσκονται στο αρχείο MIDlet JAR manifest. Δεν συμπεριλαμβάνεται στο JAR αρχείο. Είναι ένα text αρχείο με κατάληξη .jad. [15] Τα σημαντικότερα περιεχόμενα που περιλαμβάνει ένα JAD αρχείο είναι το όνομα του προγράμματος, τις απαιτούμενες εκδόσεις του CLDC και MIDP και την έκδοση του midlet για να μπορεί να κάνει updates, κλπ.

Παρακάτω βλέπουμε αναλυτικά τι περιλαμβάνει ένα JAD αρχείο του παραπάνω κώδικα “Hello World”.

### **Mobile.jad**

MIDlet-1: Midlet, , HelloWorld

MIDlet-Jar-Size: 821

MIDlet-Jar-URL: mobile.jar

MIDlet-Name: mobile

MIDlet-Vendor: Vendor

MIDlet-Version: 1.0

MicroEdition-Configuration: CLDC-1.1

MicroEdition-Profile: MIDP-2.0

# Ενότητα 3<sup>η</sup>

## 3.1 Bluetooth

### *Εισαγωγή στις Ασύρματες Επικοινωνίες (Wireless Communications)*

Ζούμε στην εποχή της πληροφόρησης. Αυτός ο όρος έχει επικρατήσει εξαιτίας της μεγάλης ανάπτυξης των υπολογιστικών συσκευών που δίνουν την δυνατότητα μεταφοράς μεγάλων ποσοτήτων δεδομένων μεταξύ υπολογιστών χρησιμοποιώντας ενσύρματη και ασύρματη τεχνολογία. Επίσης, ο κλάδος της επικοινωνίας αναπτύσσεται ραγδαία και δημιουργούνται εφαρμογές που μπορούν να εφαρμοστούν και στους υπολογιστές. [27]

Συγκεκριμένα, οι ασύρματες επικοινωνίες είναι ένα τεράστιο πεδίο, περικλείοντας τα πάντα από την εκπομπή σήματος ραδιοφώνου και τηλεόρασης μέχρι κινητά τηλέφωνα και δορυφορικές επικοινωνίες. [28]

Ως ασύρματο δίκτυο χαρακτηρίζεται το τηλεπικοινωνιακό δίκτυο, συνήθως τηλεφωνικό ή δίκτυο υπολογιστών, το οποίο χρησιμοποιεί, ραδιοκύματα ως φορείς πληροφορίας. Τα δεδομένα μεταφέρονται μέσω ηλεκτρομαγνητικών κυμάτων, με συχνότητα φέροντος η οποία εξαρτάται κάθε φορά από τον ρυθμό μετάδοσης που απαιτείται να υποστηρίξει το δίκτυο. [26]

Στα ασύρματα δίκτυα εντάσσονται τα δίκτυα κινητής τηλεφωνίας, οι δορυφορικές επικοινωνίες, τα ασύρματα δίκτυα ευρείας περιοχής (WWAN), τα ασύρματα μητροπολιτικά δίκτυα (WMAN), τα ασύρματα τοπικά δίκτυα (WLAN) και τα ασύρματα προσωπικά δίκτυα (WPAN). [26] Εμείς, θα αναλύσουμε το ασύρματο προσωπικό δίκτυο (WPAN) και πιο συγκεκριμένα το Bluetooth.

### *Ασύρματα Προσωπικά Δίκτυα (WPAN)*

Ο όρος ασύρματα προσωπικά δίκτυα (Wireless Personal Area Networks-WPAN) είναι σχετικά σύγχρονος όρος και αναφέρεται στις σύγχρονες τεχνολογίες οι οποίες επιτρέπουν την ασύρματη διασύνδεση και επικοινωνία σε αποστάσεις λίγων μέτρων φορητών προσωπικών συσκευών όπως είναι τα κινητά τηλέφωνα, τα PDA's και οι Ultra Mobile υπολογιστές μεταξύ πολλών άλλων. Η επικοινωνία αυτή επιτρέπει στις συσκευές αυτές υπηρεσίες όπως ανταλλαγή αρχείων, διαμοίραση εφαρμογών, άμεση επικοινωνία κλπ. Τα περισσότερα δίκτυα του συγκεκριμένου τύπου βασίζονται στο πρότυπο Bluetooth που θα περιγράψουμε αναλυτικά παρακάτω. [29]

## **BLUETOOTH**

Το Bluetooth είναι ένα βιομηχανικό πρότυπο για ασύρματα προσωπικά δίκτυα το οποίο επιτρέπει τη σύνδεση και επικοινωνία σε μια πλειάδα συσκευών όπως κινητά τηλέφωνα, laptop, εκτυπωτές, ψηφιακές μηχανές κλπ. μέσω μιας ασφαλούς ραδιοσυχνότητας. Το πρότυπο Bluetooth αναπτύχθηκε από την ομάδα Special Interest Group ή SIG η οποία αποτελούνταν από τις εταιρείες Ericsson, IBM, Intel, Nokia και Toshiba. Λειτουργεί στα 2.4GHz (πρότυπο 802.11b) και προσφέρει φωνητική επικοινωνία και μετάδοση δεδομένων. Το πρότυπο αυτό σχεδιάστηκε έχοντας υπόψη την χαμηλή κατανάλωση ρεύματος και την δημιουργία συσκευών λήψης/μετάδοσης οι οποίες θα έχουν πολύ μικρό μέγεθος και χαμηλό κόστος. [6],[29]

Η βασική μονάδα ενός συστήματος Bluetooth είναι ένα μικροσκοπικό δίκτυο (piconet), το οποίο αποτελείται από έναν κόμβο κυρίου (master) και μέχρι επτά ενεργούς κόμβους υπηρέτη (slaves) μέσα σε μια απόσταση 10 μέτρων. Επίσης, το Bluetooth υποστηρίζει συνδέσεις μεταξύ piconets: Όταν ένας κύριος κόμβος (master) ενός piconet γίνεται κόμβος υπηρέτη (slave) σε ένα άλλο piconet, τότε το piconet προσφέρει μια σύνδεση μεταξύ τους. [7],[30]

Η ραδιοσυχνότητα στην οποία λειτουργεί το Bluetooth είναι η ίδια με τα πρότυπα Wi-Fi αλλά το Bluetooth χρησιμοποιεί πολύ μικρότερη ισχύ και διαφορετικούς τρόπους πολύπλεξης του σήματος. Ως αποτέλεσμα, είναι αποτελεσματικό σε αποστάσεις που τυπικά κινούνται από 1m έως 10m και σε χαμηλότερες ταχύτητες. Επιπλέον, βρίσκει διαφορετικές εφαρμογές από τα πρότυπα Wi-Fi καθώς το δεύτερο θεωρείται ένας τύπος ασύρματου Ethernet ενώ το Bluetooth θεωρείται μιας μορφής ασύρματης USB σύνδεσης. [29]

Το πρότυπο Bluetooth χρησιμοποιείται τυπικά στις εξής εφαρμογές:

- Έλεγχος και επικοινωνία μεταξύ κινητών τηλεφώνων και περιφερειακών συσκευών (πχ. headset)
- Ασύρματη επικοινωνία μεταξύ υπολογιστών και περιφερειακών συσκευών όπως ποντίκι, πληκτρολόγιο και εκτυπωτές.
- Αντικατάσταση των επικοινωνιών που τυπικά χρησιμοποιούσαν την υπέρυθη ακτινοβολία (IrDA). Για παράδειγμα, κάποιες σύγχρονες κονσόλες παιχνιδιών χρησιμοποιούν το Bluetooth για την επικοινωνία με τα χειριστήρια που διαθέτουν.

- Dial-up πρόσβαση στο Internet μέσω κινητών τηλεφώνων για υπολογιστές που διαθέτουν Bluetooth. [29]

Το τελευταίο πρότυπο Bluetooth είναι το 2.0 το οποίο επιτρέπει θεωρητικά ταχύτητες μέχρι τα 2.1Mbps ενώ η διαμόρφωση του σήματος γίνεται με μια τεχνική παρόμοια με την Frequency Hopping Spread Spectrum η οποία παρέχει ασφάλεια και προστασία από παρεμβολές. Σε επίπεδο προστασίας της επικοινωνίας το πρότυπο ενσωματώνει δυνατές κρυπτογραφικές μεθόδους καθώς και δυναμική δημιουργία κλειδιών για την προστασία των δεδομένων. [29]

### ***Γιατί επιλέγουμε Java τεχνολογία για συσκευές Bluetooth;***

Ξεκινάμε με ένα παράδειγμα, όπου δυο άτομα έχουν το ίδιο μοντέλο κινητού τηλεφώνου με τεχνολογία Bluetooth αλλά θέλουν να το χρησιμοποιήσουν για διαφορετικό σκοπό ο καθένας. Το ένα άτομο θέλει να κάνει download video παιχνίδια στο κινητό του τηλέφωνο και να χρησιμοποιεί το κινητό του ως τηλεχειριστήριο για την τηλεόραση. Αντίθετα, το άλλο άτομο μπορεί να θέλει να χρησιμοποιήσει το ίδιο μοντέλο κινητού για να ξεκλειδώσει τις πόρτες του αυτοκινήτου ή να ανοίξει και να κλείσει την πόρτα του γκαράζ.

Με βάση το παραπάνω παράδειγμα, ένας τρόπος για να πετύχουν και τα δυο άτομα τον στόχο τους είναι να μπορέσουν να κάνουν download Bluetooth εφαρμογές στα personal organizers και στα κινητά τηλέφωνα. Για να γίνει αυτό, χρειάζεται ένα standard API που επιτρέπει στους προγραμματιστές να γράψουν εφαρμογές Bluetooth που «τρέχουν» σε πολλές hardware πλατφόρμες. Για να ορίσουμε αυτό το standard API, η γλώσσα προγραμματισμού Java είναι η ιδανική επιλογή. Ένα Java API δίνει την δυνατότητα στις εφαρμογές να «τρέξουν» σε διαφορετικούς τύπους hardware και λειτουργικών συστημάτων. Επιπλέον, εκτός από τη φορητότητα (portability), η Java προσφέρει και αρκετά άλλα πλεονεκτήματα:

- Γρήγορος σχεδιασμός & ανάπτυξη των εφαρμογών.
- Η ικανότητα να εξαπλώνεται δυναμικά η λειτουργικότητα ενός προγράμματος κατά την διάρκεια εκτέλεσης του, «φορτώνοντας» τις κλάσεις σε run time.
- Η επιβεβαίωση αρχείων κλάσεων και χαρακτηριστικών ασφαλείας που προσφέρουν προστασία ενάντια σε μη επιθυμητές (malicious) εφαρμογές.

- Μεγάλη κοινότητα από προγραμματιστές. Ο αριθμός των ανθρώπων που προγραμματίζουν σε Java συνεχώς αυξάνεται.

Για τους παραπάνω λόγους, χρησιμοποιούμε την γλώσσα προγραμματισμού Java για την ανάπτυξη standard API για Bluetooth wireless technology. [27]

## 3.2 Java APIs για Ασύρματη Τεχνολογία Bluetooth JSR-82

Η Java 2 Platform, Micro Edition (J2ME) και η τεχνολογία Bluetooth είναι δυο από τις πιο εντυπωσιακές προτάσεις στην ασύρματη βιομηχανία σήμερα. Η J2ME υποστηρίζει το μότο «write once run anywhere» και γι' αυτό εντείνει την παραγωγική ανάπτυξη. Από την άλλη το Bluetooth είναι μια παγκόσμια μικρού εύρους ασύρματης standard συνδεσιμότητας για ηλεκτρονικές συσκευές και κινητά τηλέφωνα. Το Bluetooth επιτρέπει σε συσκευές να επικοινωνούν μεταξύ τους ασύρματα και η J2ME μας επιτρέπει να γράφουμε κατά παραγγελία εφαρμογές και να αναπτύξουμε τις εφαρμογές αυτές σε κινητές συσκευές.

Ενώ το Bluetooth hardware έχει προοδεύσει, δεν έχει υπάρξει κανένας τρόπος τυποποίησης για την ανάπτυξη Bluetooth εφαρμογών, μέχρι που εμφανίστηκε το JSR-82. [31] Για την ανάπτυξη του λογισμικού Bluetooth σε Java πλατφόρμα, η JCP όρισε το JSR-82 ως το standard API for Bluetooth Wireless Technology (JABWT). Το API ορίστηκε ως optional package για την Java ME συσκευή βασισμένο στο CLDC configuration. [27]. Κρύβει την πολυπλοκότητα του Bluetooth protocol stack πίσω από ένα σετ από Java APIs που μας επιτρέπουν να επικεντρωνόμαστε στην ανάπτυξη εφαρμογών παρά σε low-level λεπτομέρειες του Bluetooth. Το JSR-82 βασίζεται στην έκδοση 1.1 του Bluetooth Specification.

Όπως όλα τα JSRs, τα Java APIs για το Bluetooth έχουν αναπτυχθεί μέσω της Java Community Process (JCP). Το JCP αποτελείται από ένα γκρουπ από ειδικούς, του οποίου τα μέλη αντιπροσωπεύονται από 20 εταιρείες. Το JSR-82 αποτελείται από δυο optional packages: (i) the core Bluetooth API και (ii) the Object Exchange (OBEX) API. Το τελευταίο μπορεί να χρησιμοποιηθεί χωρίς το πρώτο.

Τα Java APIs για Bluetooth στοχεύει συσκευές με τα παρακάτω χαρακτηριστικά.

- 512K μίνιμουμ της συνολικής διαθέσιμης μνήμης
- Σύνδεση δικτύου ασύρματου Bluetooth

- Compliant υλοποίηση του J2ME Connected Limited Device Configuration. [31]

### **Χαρακτηριστικά του JSR 82**

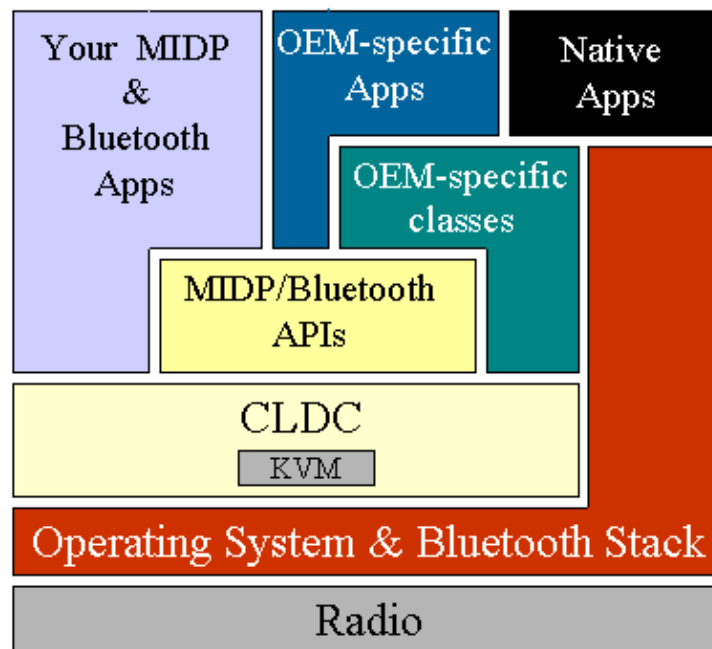
Το API προορίζεται να προσφέρει τις παρακάτω χαρακτηριστικά

- Δήλωση υπηρεσιών
- Εύρεση συσκευών και υπηρεσιών
- Εγκατάσταση συνδέσεων RFCOMM, L2CAP και OBEX μεταξύ συσκευών.
- Χρησιμοποιώντας αυτές τις συνδέσεις, αποστολή και λήψη δεδομένων (δεν υποστηρίζεται η επικοινωνία με φωνή)
- Διαχείριση και έλεγχος στις συνδέσεις επικοινωνίας
- Προσφέρει ασφάλεια για αυτές τις δραστηριότητες [31]

Τα APIs έχουν σχεδιαστεί με τέτοιο τρόπο που οι κατασκευαστές (developers) μπορούν να χρησιμοποιήσουν την γλώσσα προγραμματισμού Java για να σχεδιάσουν καινούργια Bluetooth profiles πάνω στο API εφόσον το core specification δεν αλλάξει. Το JSR-82 περιλαμβάνει APIs για OBEX και L2CAP έτσι ώστε τα μελλοντικά Bluetooth profiles να μπορούν να υλοποιηθούν με Java.

Στην παρακάτω εικόνα βλέπουμε που ορίζονται τα APIs σε αυτό το specification τα οποία περιλαμβάνονται σε μια CLDC/MIDP αρχιτεκτονική. [31]

**Εικόνα 5:** High-level Αρχιτεκτονική της J2ME CLDC/MIDP και Bluetooth



# Ενότητα 4<sup>η</sup>

## 4.1 Βάσεις Δεδομένων

Καταρχήν, πριν προχωρήσουμε στην έννοια της Βάσης Δεδομένων θα πρέπει να ξεχωρίσουμε τις έννοιες δεδομένα (data) και πληροφορία (information).

Ως **δεδομένα**, ορίζονται στοιχεία, έννοιες και οδηγίες, τυποποιημένα σε καθορισμένη μορφή, κατάλληλα για επεξεργασία από ανθρώπους ή μηχανές. Η **πληροφορία** εξάγεται από τα δεδομένα και χρησιμοποιείται στην επίλυση προβλημάτων.

Ένα βασικό χαρακτηριστικό των δεδομένων είναι ότι είναι κατάλληλα για να αποθηκευτούν σε ηλεκτρονικό υπολογιστή. Αυτό οδηγεί στην δημιουργία και χρήση Βάσεων Δεδομένων (databases).

Άρα, η **Βάση Δεδομένων** είναι μια αυστηρά τυποποιημένη συλλογή από σχετικά μεταξύ τους δεδομένα, που παρέχει έναν αυτόματο, κεντρικό και κοινό τρόπο χειρισμού τους.

Η σχετικότητα των δεδομένων είναι σημαντική για τον ορισμό μιας Βάσης Δεδομένων. Δεδομένα που δε σχετίζονται μεταξύ τους και απλά έχουν αποθηκευτεί σε έναν ηλεκτρονικό υπολογιστή δεν αποτελούν μια Βάση Δεδομένων. Μια Βάση Δεδομένων πρέπει να αντικατοπτρίζει ένα περιβάλλον του πραγματικού κόσμου. Τα δεδομένα που αποθηκεύονται στη Βάση Δεδομένων πρέπει να έχουν λογική συνέχεια και νόημα. Η Βάση Δεδομένων έχει ένα σκοπό. Υλοποιείται για να απεικονίσει στον υπολογιστή ένα πρόβλημα και να διευκολύνει τη λύση του. Αυτό συνήθως σημαίνει ότι τα δεδομένα που χειρίζεται δεν είναι στατικά αλλά δυναμικά, δηλαδή αλλάζουν συνεχώς. [32]

### *Συστήματα Διαχείρισης Βάσεων Δεδομένων*

Η εξέλιξη των Βάσεων Δεδομένων και οι ανάγκες για δημιουργία όλο και περισσότερων Βάσεων Δεδομένων, οδήγησαν στη δημιουργία των Συστημάτων Διαχείρισης Βάσεων Δεδομένων (Database Management Systems ή DBMS). Το Σύστημα Διαχείρισης Βάσεων Δεδομένων είναι ένα εργαλείο το οποίο διευκολύνει τους χρήστες να εργάζονται με Βάσεις Δεδομένων. Με τη χρήση του Συστήματος Διαχείρισης Βάσης Δεδομένων οι χρήστες μπορούν να κατασκευάσουν και να χρησιμοποιήσουν Βάσεις Δεδομένων.



Άρα, **Συστήματα Διαχείρισης Βάσεων Δεδομένων** είναι το λογισμικό το οποίο επιτρέπει στους χρήστες να δημιουργούν και να χρησιμοποιούν Βάσεις Δεδομένων.

Ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων κατά κανόνα «φιλοξενεί» πολλές Βάσεις Δεδομένων που έχουν κατασκευαστεί από διαφορετικούς χρήστες.

Οι δυνατότητες που παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων στους χρήστες είναι:

- Ορισμός της Βάσης Δεδομένων. Ο χρήστης μπορεί να καθορίσει το μοντέλο της Βάσης Δεδομένων, να ορίσει τους τύπους δεδομένων που θα χρησιμοποιήσει και να ελέγξει τη Βάση Δεδομένων χωρίς να προχωρήσει σε κατασκευή της.
- Κατασκευή της Βάσης Δεδομένων. Οι τύποι των δεδομένων και τα δεδομένα αποθηκεύονται στο υλικό (hardware) του ηλεκτρονικού υπολογιστή με διαδικασίες που ελέγχονται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων και δεν απασχολούν τον χρήστη.
- Διαγραφή της Βάσης Δεδομένων. Ο χρήστης αποφασίζει για τον τερματισμό μιας Βάσης Δεδομένων και την απομάκρυνση των δεδομένων από το υλικό.
- Χρήση της Βάσης Δεδομένων. Ο χρήστης είτε χειρίζεται τα δεδομένα (εισάγει νέα δεδομένα, τροποποιεί δεδομένα, ή διαγράφει δεδομένα), είτε υποβάλει ερωτήσεις στη Βάση Δεδομένων με στόχο την εξαγωγή πληροφοριών. [32]

#### ***Τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων***

Μια Βάση Δεδομένων μπορεί να υλοποιηθεί και αυτόνομα χωρίς να χρησιμοποιηθεί ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων. Όμως, κατά κανόνα οι παροχές ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων είναι αυτές που τελικά ωθούν τους χρήστες να το χρησιμοποιήσουν. Ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων παρέχει:

- Ευκολία στη σχεδίαση και υλοποίηση
- Επίπεδα χρηστών και έλεγχος πρόσβασης
- Προστασία από βλάβες υλικού
- Δυνατότητα περιγραφής περιορισμών ορθότητας
- Δυνατότητα ελέγχου πλεονασμών [32]

***Τι παρέχει η βάση δεδομένων***

- Ταυτόχρονη προσπέλαση
- Ταχύτατη εξαγωγή απαντήσεων
- Ευελιξία
- Υψηλή ποιότητα δεδομένων [32]

***Εισαγωγή στην Microsoft Access***

Η Microsoft Access είναι ένα από τα πιο δημοφιλή προγράμματα διαχείρισης βάσεων δεδομένων που κυκλοφορούν στην αγορά. Η μεγάλη διάδοση της τα τελευταία χρόνια, οφείλεται στην απλότητα, λειτουργικότητα και ευκολία στη χρήση της, καθώς και στη δυνατότητα της να δημιουργεί εφαρμογές διαχείρισης βάσεων δεδομένων σε σχετικά μικρό χρονικό διάστημα. Η Microsoft Access επιτρέπει τη δημιουργία βάσεων που στηρίζονται στο σχεσιακό μοντέλο (relational database model). [34]

Η Access είναι ένα επιτραπέζιο (Desktop) σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS), η οποία λειτουργεί στο γραφικό περιβάλλον των Windows. Διαθέτει ένα ολοκληρωμένο σύστημα ανάπτυξης εφαρμογών με σκοπό την αυτοματοποίηση των εργασιών. Ο όρος επιτραπέζιο σύστημα αναφέρεται προκειμένου να τονιστεί η διαφορά από άλλα μεγάλα συστήματα διαχείρισης βάσεων δεδομένων τα οποία λειτουργούν σε περιβάλλον πελάτη/ διακομιστή (client/server), όπως για παράδειγμα ο SQL server, MySQL, Oracle κλπ, τα οποία απευθύνονται σε επιχειρήσεις ή οργανισμούς.

Η Access δίνει τη δυνατότητα σε ένα χρήστη να δημιουργήσει εύκολα μια ολοκληρωμένη εφαρμογή δίχως να χρειαστεί τις περισσότερες φορές να γράψει ούτε μια γραμμή κώδικα προγράμματος. Η βοήθεια στην Access είναι πολύ εξελεγχμένη και σημαντική. Εκτός από τη βοήθεια υπάρχουν οι λεγόμενοι Οδηγοί ή Μάγοι (Wizards) που μπορούν να μας καθοδηγήσουν εύκολα για να δημιουργήσουμε Πίνακες, Εκθέσεις, Φόρμες ή ακόμα και μια νέα Βάση Δεδομένων.

Η Access έχει την δυνατότητα να επικοινωνεί και να χειρίζεται δεδομένα από αρχεία κειμένου, λογιστικά φύλλα, αρχεία από τις πιο δημοφιλείς βάσεις δεδομένων. Κάθε εφαρμογή της Access αποθηκεύεται σε ένα μοναδικό αρχείο με προέκταση .mdb. Ανοίγοντας ένα αρχείο βάσης δεδομένων ενεργοποιούνται όλοι οι πίνακες, τα ερωτήματα, οι φόρμες, οι εκθέσεις, οι μακροεντολές και οι λειτουργικές μονάδες. [33]

### **Σχεσιακή Βάση Δεδομένων**

Η Σχεσιακή Βάση Δεδομένων (Relational Database) είναι μια οργανωμένη συλλογή πληροφοριών (δεδομένων), η οποία βασίζεται στο σχεσιακό μοντέλο. Η οργάνωση των πληροφοριών πρέπει να γίνεται με τέτοιο τρόπο ώστε να εντοπίζονται εύκολα και γρήγορα από κάποιο χρήστη. Οι πληροφορίες αυτές είναι οργανωμένες σε Πίνακες (Tables). Κάθε πίνακας έχει πληροφορίες που έχουν σχέση με ένα μόνο θέμα και αποτελείται από σειρές και στήλες οι οποίες είναι προσβάσιμες οριζόντια ή κατακόρυφα.

Οι στήλες ή πεδία (fields) των πινάκων περιέχουν διαφορετικά είδη πληροφοριών για το ίδιο όμως θέμα. Οι γραμμές ή εγγραφές (records) περιγράφουν τις ιδιότητες μιας περίπτωσης του θέματος. Κάθε εγγραφή περιέχει πληροφορίες οι οποίες είναι συσχετισμένες με ένα μοναδικό θέμα. Στις σχεσιακές βάσεις δεδομένων οι εγγραφές σε έναν πίνακα πρέπει να διαφέρουν τουλάχιστον ως προς ένα πεδίο. Αυτό επιτυγχάνεται με τη χρήση πεδίων που ορίζονται ως κλειδιά (keys). Η Access διαθέτει έναν ειδικό τύπο πεδίου, που ονομάζεται Αυτόματη Αρίθμηση (Autonumber), με το οποίο μπορούμε να ορίσουμε ένα πρωτεύον κλειδί (primary key), με το οποίο επιτυγχάνεται αυτόματα η αποφυγή διπλοεγγραφών. Οι πληροφορίες συνδέονται μεταξύ τους με σχέσεις που απορρέουν από τα κοινά πεδία διαφορετικών πινάκων. Τα κοινά πεδία παίρνουν τις τιμές τους από κοινά πεδία τιμών και η ύπαρξη μιας κοινής τιμής καθορίζει και μια σχέση μεταξύ των γραμμών διαφορετικών πινάκων. [33]

Η Access χρησιμοποιεί την SQL (Structured query language – Δομημένη Γλώσσα Ερωτημάτων) που αποτελεί σήμερα την πιο διαδεδομένη γλώσσα διαχείρισης σχεσιακών βάσεων δεδομένων. [35] Ο χρήστης έχει τη δυνατότητα να δημιουργήσει μόνος του τα ερωτήματα προς τη βάση γράφοντας κατευθείαν κώδικα σε SQL, αλλά αν το επιθυμεί, μπορεί να το κάνει μέσα από ένα εύχρηστο περιβάλλον αυτοματοποιημένης δημιουργίας ερωτημάτων, το οποίο, ζητά από το χρήστη να καθορίσει τις πληροφορίες που θέλει να ανακτήσει, και στη συνέχεια, δημιουργεί τον κώδικα SQL από μόνο του. [34]

***Η SQL παρέχει δυνατότητες για:***

- Τον ορισμό, τη διαγραφή και τη μεταβολή πινάκων και κλειδιών
- Τη σύνταξη ερωτήσεων (queries)
- Την εισαγωγή, διαγραφή και μεταβολή στοιχείων
- Τον ορισμό όψεων (views) πάνω στα δεδομένα
- Τον ορισμό δικαιωμάτων πρόσβασης
- Τον έλεγχο της ακεραιότητας των στοιχείων
- Τον έλεγχο συναλλαγών (transaction) [35]

***Χαρακτηριστικά των σχεσιακών βάσεων δεδομένων***

- Σχέση: όλες οι πληροφορίες που συσχετίζονται με ένα μοναδικό θέμα (πίνακας).
- Ιδιότητα: κάποια συγκεκριμένη πληροφορία που είναι σχετική με το θέμα (πεδίο του πίνακα).
- Αντιστοίχιση: με ποιον τρόπο οι πληροφορίες μιας σχέσης συσχετίζονται με τις πληροφορίες μιας άλλης σχέσης (ένα προς πολλά, ένα προς ένα, πολλά προς πολλά).
- Ένωση: σύνδεση πινάκων και ερωτημάτων σε πίνακες [33]

***Τύποι σχέσεων πινάκων***

- **Ένα προς πολλά (one-to-many):** Είναι η πιο συνηθισμένη σχέση που συναντάμε στην Access. Συνδέει μία εγγραφή του γονικού πίνακα (parent table) με μηδέν, μία ή περισσότερες εγγραφές του θυγατρικού (child) πίνακα. Αυτό σημαίνει ότι για κάθε εγγραφή του πρώτου πίνακα, υπάρχουν πολλές συσχετισμένες εγγραφές στον δεύτερο, αλλά σε κάθε εγγραφή του δεύτερου πίνακα αντιστοιχεί μόνο μία εγγραφή του πρώτου.
- **Ένα προς ένα (one-to-one):** Κάθε εγγραφή του ενός πίνακα αντιστοιχεί μία και μόνο μία εγγραφή του άλλου πίνακα και το αντίστροφο. Συχνά χρησιμοποιείται για να σπάμε ένα πίνακα στον οποίο περιέχονται πολλά πεδία σε δύο ή περισσότερους μικρότερους πίνακες.
- **Πολλά-προς-πολλά (many-to-many):** Αυτή η σχέση συνδέει μία ή περισσότερες εγγραφές ενός πίνακα με μία ή περισσότερες εγγραφές ενός άλλου πίνακα. Γι' αυτό, θα πρέπει να δημιουργήσουμε έναν ξεχωριστό πίνακα

διασταύρωσης (intersection table) για να μπορέσουμε να χειριστούμε τη σχέση αποτελεσματικά. Αν δεν το κάνουμε με αυτό τον τρόπο, τότε η Access χαρακτηρίζει την σχέση αόριστη. [36],[37]

### ***Java Database Connectivity***

Το Java Database Connectivity (JDBC) είναι ένα σύνολο κλάσεων που χρησιμοποιούνται για την ανάπτυξη εφαρμογών με αρχιτεκτονική client/server (πελάτη/διακομιστή), οι οποίες μπορούν να δουλεύουν με βάσεις δεδομένων προερχόμενες από λογισμικό των Microsoft, Sybase, Informix, και άλλες πηγές.

Με το JDBC, μπορούμε να χρησιμοποιούμε τις ίδιες μεθόδους και κλάσεις σε προγράμματα Java για να διαβάζουμε και να γράφουμε εγγραφές, καθώς για να εκτελούμε άλλα είδη εργασιών που απαιτούν την προσπέλαση μιας βάσης δεδομένων. Μια κλάση η οποία αποκαλείται πρόγραμμα οδήγησης (driver) λειτουργεί σαν γέφυρα με την προέλευση της βάσης δεδομένων.

Η Java DB, η βάση δεδομένων που περιλαμβάνεται στην Java 6, υποστηρίζει την γλώσσα SQL και κάνει πιο απλή τη χρήση εφαρμογών σχεσιακών βάσεων δεδομένων. Στις εφαρμογές βάσεων δεδομένων, μια αίτηση για ανάκτηση εγγραφών από μια βάση δεδομένων αποκαλείται ερώτημα (query). Χρησιμοποιώντας την SQL, μπορούμε να στέλνουμε πολύπλοκα ερωτήματα σε μια βάση δεδομένων και να ανακτάμε τις εγγραφές που μας ενδιαφέρουν με την σειρά που εμείς καθορίζουμε. Το **ODBC** (Open Database Connectivity) είναι το κοινό περιβάλλον διασύνδεσης που χρησιμοποιεί η Microsoft για την προσπέλαση βάσεων δεδομένων SQL.

Συμπερασματικά, βλέπουμε πως μέσα από ένα πρόγραμμα Java μπορούμε να συνδεθούμε με αρκετές μορφές σχεσιακών βάσεων δεδομένων, χρησιμοποιώντας το JDBC ή ODBC και την γλώσσα SQL μια γλώσσα που έχει καθιερωθεί σαν πρότυπο για την ανάγνωση, την εγγραφή και την διαχείριση βάσεων δεδομένων. [12]

# Ενότητα 5<sup>η</sup>

## 5.1 Netbeans IDE

Το **Netbeans** είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE – Integrated Development Environment) για κατασκευαστές λογιστικού (software developers). Με τα εργαλεία που μας παρέχει μπορούμε να δημιουργήσουμε επιτραπέζιες (desktop), enterprise, web και mobile εφαρμογές με την Java πλατφόρμα. [41]

Πιο συγκεκριμένα, εμείς θα ασχοληθούμε με το Mobile Application. Οι περισσότερες εφαρμογές κινητών τηλεφώνων χρησιμοποιούν **Java Micro Edition** (Java ME) πλατφόρμα, η οποία έχει αναπτυχθεί για μικρές συσκευές όπως κινητά τηλέφωνα, αν και τώρα χρησιμοποιούνται για μεγαλύτερη γκάμα συσκευών. [40] Το Netbeans μας προσφέρει για περιβάλλον εργασίας & ανάπτυξης εφαρμογής το **Sun Java Wireless Toolkit**.

Το Sun Java Wireless Toolkit (γνωστό και ως J2ME Wireless Toolkit) είναι ένα σετ από εργαλεία για τη δημιουργία εφαρμογών Java που «τρέχουν» σε συσκευές με Java Technology for Wireless Industry (JTWI, JSR 185) και Mobile Service Architecture (MSA, JSR 248). Περιέχει build tools, utilities και συσκευή εξομοιωτή (device emulator). [43] Θα το χρησιμοποιήσουμε για την σχεδίαση & ανάπτυξη εφαρμογής για το κινητό τηλέφωνο. Για την εργασία μας, χρησιμοποιήσαμε την έκδοση 2.5.2. Επειδή, η J2ME Wireless Toolkit 2.5.2 περιέχει 4 κύριες συσκευές εξομοιωτών (emulators), εμείς, σε αυτήν την εργασία επιλέξαμε το DefaultColorPhone η οποία είναι μια μη πραγματική συσκευή κινητού τηλεφώνου με 240x320 pixel color screen. [15] Ο εξομοιωτής (emulator) μας δίνει την δυνατότητα να εκτελέσουμε και να κάνουμε τεστ την εφαρμογή πριν την μεταφέρουμε στο κινητό μας τηλέφωνο.

Για την εργασία μας, θα χρησιμοποιήσουμε το Netbeans IDE 6.1. Πριν χρησιμοποιήσουμε το πρόγραμμα αυτό, θα πρέπει πρώτα να εγκαταστήσουμε το **JDK** (Java Development Kit) που είναι το περιβάλλον ανάπτυξης για την δημιουργία εφαρμογών Java. Εμείς έχουμε εγκαταστήσει το JDK 1.6 Update 7. Το JDK περιέχει το JVM (Java Virtual Machine), Java βιβλιοθήκες, εργαλεία όπως το jar και javadoc. [42]

Το Netbeans, το JDK και το Sun Java Wireless Toolkit διατίθενται δωρεάν στις παρακάτω διευθύνσεις:

<http://netbeans.org/downloads/index.html>

<http://www.oracle.com/technetwork/java/javase/downloads/jdk6-jsp-136632.html>

<http://www.oracle.com/technetwork/java/download-135801.html>

(\* Οι παραπάνω διευθύνσεις σας παραθέτουν στις τελευταίες εκδόσεις των προγραμμάτων που κυκλοφορούν.)

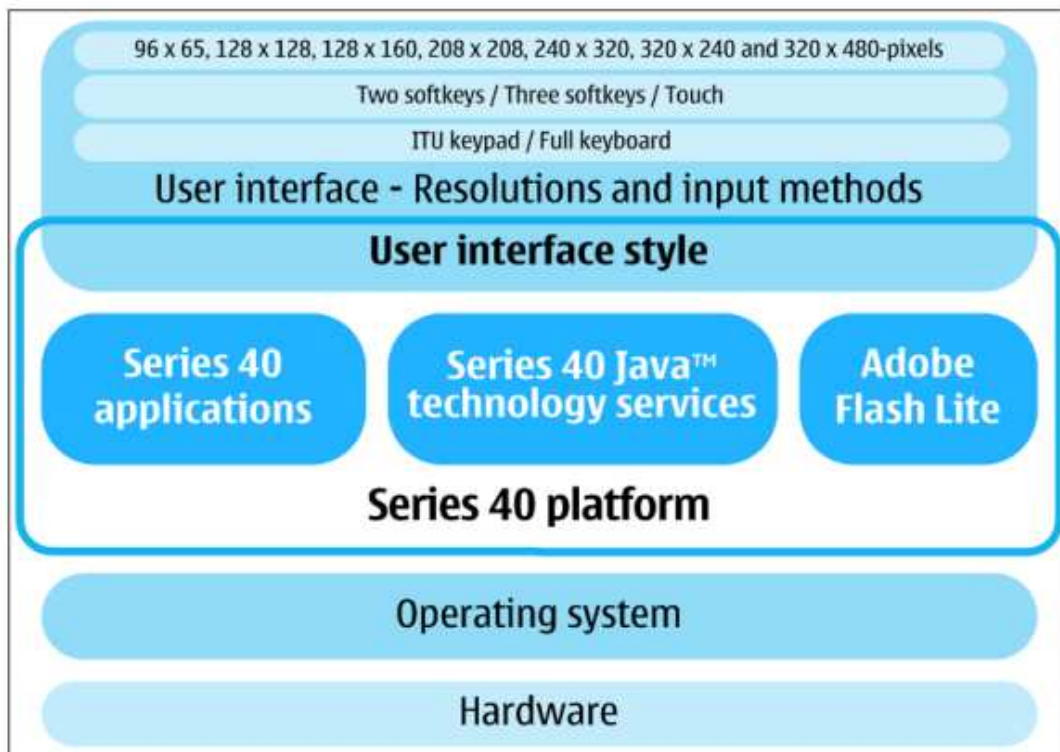
(\* \* Αν στο Netbeans δεν υπάρχει το Sun Java Wireless Toolkit τότε μπορείτε να το κάνετε download δωρεάν και να το εγκαταστήσετε).

Το Netbeans μπορεί να εγκαταστήσει εκτός από το Sun Java Wireless Toolkit και πλατφόρμες κινητών τηλεφώνων της NOKIA και SONY ERICSSON. Έτσι μας δίνεται η δυνατότητα μέσω του Netbeans να σχεδιάσουμε και να αναπτύξουμε την εφαρμογή μας και στο κινητό τηλέφωνο της NOKIA και της SONY ERICSSON. Παρακάτω, θα αναλύσουμε μόνο την πλατφόρμα της NOKIA.

## 5.2 Series 40 Platform

Η πλατφόρμα Series 40 έχει υλοποιηθεί χρησιμοποιώντας το λειτουργικό σύστημα της NOKIA και προσφέρει στους developers την ευκαιρία να δημιουργήσουν εφαρμογές χρησιμοποιώντας Java ME APIs και Adobe Flash Lite. Η πλατφόρμα Series 40 έχει σχεδιαστεί για μια μεγάλη γκάμα συσκευών. Υπάρχουν ποικιλίες για το user interfaces, APIs και λογισμικό συσκευής.

Όλες οι συσκευές που βασίζονται στην πλατφόρμα Series 40 μοιράζονται μια κοινή αρχιτεκτονική και αυτό μπορούμε να το δούμε στην παρακάτω εικόνα.

**Εικόνα 6**

Η αρχιτεκτονική της συσκευής αποτελείται από το hardware, λειτουργικό σύστημα το οποίο προσφέρει βασικές υπηρεσίες στην πλατφόρμα, και η πλατφόρμα Series 40 η οποία αποτελείται από:

- Series 40 εφαρμογές – Προσφέρει λειτουργικότητα στον χρήστη και περιλαμβάνει
  - Εφαρμογές επικοινωνίας όπως τηλέφωνο, μηνύματα και internet browser
  - Εφαρμογές media όπως image viewer, κάμερα, music player, voice recorder και ραδιόφωνο FM.
  - Personal Information Manager (PIM) που έχει ημερολόγιο, tasks και contacts applications.
- Series 40 Java, technology services – Η Java τεχνολογία υλοποιείται μέσα στην πλατφόρμα
- Adobe Flash Lite – Οι υπηρεσίες που προσφέρουν την δυνατότητα να «τρέξουν» Flash Lite εφαρμογές.(σε επιλεγμένες εκδόσεις)
- User interface style: ανάλυση (resolution) και μέθοδοι εισαγωγής δεδομένων (input methods)



Η πλατφόρμα Series 40, μπορεί να ενσωματωθεί σε συσκευές που υποστηρίζουν ένα εύρος από standards ασύρματων δικτύων.

- Όλες οι συσκευές Series 40 υποστηρίζουν είτε GSM και GPRS δίκτυα είτε CDMA δίκτυα, προσφέροντας συμβατότητα με δίκτυα κινητών τηλεφώνων σε όλο τον κόσμο.
- Μερικές συσκευές υποστηρίζουν EDGE δίκτυα και 3G Universal Mobile Telecommunications (UMTS) δίκτυα για γρήγορη ασύρματη μετάδοση και δεδομένων.

Επιπλέον, οι Series 40 συσκευές μπορούν να προσφέρουν ένα ή περισσότερα πρωτόκολλα συνδεσιμότητας τοπικών δικτύων (local network - connectivity protocols) τα οποία είναι: Ασύρματη τεχνολογία Bluetooth, Universal Serial Bus (USB), IrDA και WLAN χρησιμοποιώντας Unlicensed Mobile Access (UMA) technology. [38]

Στην εργασία μας, θα χρησιμοποιήσουμε την πλατφόρμα Series 40 SDK, 3<sup>rd</sup> Edition Feature Pack 2 επειδή το κινητό υποστηρίζει αυτή την πλατφόρμα.

Αυτό το feature pack προσφέρει:

-> Java ME και συμπεριλαμβάνει:

- MIDP 2.0 και CLDC 1.1
- PDA Optional Packages for the J2ME Platform (JSR 75), including PIM and FC.
- Java APIs for Bluetooth (JSR 82), excluding support for OBEX
- Wireless Messaging API (WMA) (JSR 120)
- Mobile Media API (MMAPI) (JSR 135), providing the ability to play back MIDI and tone, sampled audio, RTSP streaming, and music progressive sound files, as well as perform video and image rendering.
- J2ME Web Services Specification (JSR 172), implementing the XML parsing package.
- Security and Trust Services API for J2ME (JSR 177), implementing the SATSA-APDU package.
- Mobile 3D Graphics API for J2ME (JSR 184).
- Wireless Messaging API 2.0 (JSR 205).
- Scalable 2D Vector Graphics API for J2ME (JSR 226)

- The Nokia UI API.
- Browser: Nokia Web Browser (offering WAP 2.0 and XHTML and HTML over TCP/IP)
- Flash Lite 2.0 support
- Supported screen resolutions: 128 x 160 pixels and 240 x 320 pixels. [39]

To Series 40 SDK, 3<sup>rd</sup> Edition Feature Pack 2 περιέχει επίσης τη συσκευή εξομοιωτή (emulator) του κινητού μας τηλεφώνου που έχουμε στην πραγματικότητα, δηλαδή το κινητό NOKIA 5300.

## Ενότητα 6<sup>η</sup>

### 6.1 Συνεργατική Μάθηση μέσω Κινητών Συσκευών (Mobile Learning) [8]&[9]

Η έλευση της κοινωνίας της γνώσης (knowledge society) δημιουργεί νέα δεδομένα για την μάθηση: η παγκόσμια ψηφιακή οικονομία απαιτεί ένα ευέλικτο εργατικό δυναμικό, που ενημερώνεται διαρκώς για τη νέα γνώση που παράγεται με γρήγορους ρυθμούς και οι πολίτες της κοινωνίας της γνώσης χρειάζεται να (επαν)εκπαιδούνται διαρκώς ώστε να είναι ανταγωνιστικοί, αλλά και να αξιοποιούν τις νέες δυνατότητες για την προσωπική τους εξέλιξη.

Η ραγδαία εξέλιξη των τεχνολογιών μάθησης (learning technologies) – που αξιοποιούν την αντίστοιχη εξέλιξη των τεχνολογιών πληροφορικής και επικοινωνιών (ΤΠΕ) – προσφέρει μια σειρά από νέες δυνατότητες για την κάλυψη των παραπάνω αναγκών: εφαρμογές και υπηρεσίες (συνεργατικής) εκπαίδευσης και κατάρτισης είναι διαθέσιμες σε όλους (anyone, anytime, anyplace), π.χ. μέσω δικτυακών περιβαλλόντων μάθησης (web-based learning environments), συστημάτων διαχείρισης μάθησης (learning management systems), συστημάτων διαχείρισης μαθησιακού περιεχομένου (learning content management systems), κλπ.

Από την άλλη πλευρά όμως, οι περισσότερες από τις υπάρχουσες τεχνολογικές λύσεις υλοποιούν περιβάλλοντα μάθησης που δεν ξεφεύγουν πολύ από τα «παραδοσιακά» πρότυπα: βασίζονται στην έννοια ότι ένας (ή πολλοί) εκπαιδευτές

υποστηρίζουν τους εκπαιδευόμενους στην απόκτηση συγκεκριμένης γνώσης (μέσω συγκεκριμένου μαθησιακού περιεχομένου, δραστηριοτήτων, κλπ), που μπορεί να αξιολογηθεί με συγκεκριμένους τρόπους, κλπ.

Οι πρόσφατες καινοτομίες στις ΤΠΕ επιτρέπουν τη διαφοροποίηση από το μοντέλο αυτό: η μάθηση μπορεί να λαμβάνει χώρα μέσω κινητών συσκευών, χωρίς κανένα περιορισμό χώρου ή χρόνου. Έτσι η μάθηση μπορεί να «αποδεσμευτεί» από το περιβάλλον της τάξης, και να πραγματοποιηθεί σε οποιοδήποτε περιβάλλον (στη φύση, σε ένα δάσος, σε ένα εργαστήριο εκτός σχολείου, κλπ), παράλληλα με πραγματικές και «αυθεντικές» δραστηριότητες (outdoor activities).

## 6.2 Η Μάθηση μέσω Κινητών Συσκευών

Η μάθηση μέσω κινητών συσκευών (ΜΚΣ) μπορεί να οριστεί απλά ως κάθε μορφή μάθησης που αξιοποιεί τις δυνατότητες που προσφέρουν οι κινητές (mobile) και ασύρματες (wireless) τεχνολογίες και συσκευές, όπως Wi-Fi, Bluetooth, multi-hop wireless LAN, GPS, GSM, GPRS, 3G και δορυφορικά συστήματα, κινητά τηλέφωνα, PDAs, φορητοί υπολογιστές, κλπ. Θα μπορούσε δηλαδή η ΜΚΣ να διαφοροποιηθεί μόνο από την αξιοποίηση συγκεκριμένων τεχνολογιών και συσκευών. Αν δούμε όμως τη μάθηση από την πλευρά του χρήστη-εκπαιδευόμενου, η σημαντική διαφορά έγκειται στο γεγονός ότι η μάθηση μπορεί να λαμβάνει χώρα οπουδήποτε: για παράδειγμα, οι μαθητές μπορούν να διαβάσουν τα μαθήματά τους στο σχολικό λεωφορείο, οι γιατροί να αποκτήσουν νέες γνώσεις στις συζητήσεις στους διαδρόμους του νοσοκομείου, κλπ. Γενικά, υπάρχουν αμέτρητες περιπτώσεις μάθησης καθώς οι άνθρωποι κινούνται. Κατά συνέπεια, ο ορισμός της ΜΚΣ πρέπει να περιλάβει και τις δύο αυτές διαστάσεις, δηλαδή της τεχνολογίας και του χρήστη :

### ***Ορισμός Μάθησης μέσω Κινητών Συσκευών***

- Είναι κάθε μορφή μάθησης που πραγματοποιείται χωρίς ο εκπαιδευόμενος να χρειάζεται να βρίσκεται σε προκαθορισμένα σημεία
- Αξιοποιεί τις δυνατότητες που προσφέρουν οι ασύρματες φορητές τεχνολογίες και συσκευές

Οι κινητές συσκευές διαθέτουν μια σειρά από χαρακτηριστικά που τις καθιστούν ελκυστικές για την εκπαίδευση, όπως:

- Είναι φθηνές, σε σχέση με τους υπολογιστές, και μπορούν να μεταφερθούν εύκολα
- Προσφέρουν τη δυνατότητα για ubiquitous («πανταχού παρόν») computing
- Προσφέρουν πρόσβαση σε πληροφορίες και προωθούν την ανάπτυξη του ψηφιακού αλφαριθμητισμού (Information literacy)
- Διευκολύνουν τα άτομα με ειδικές ανάγκες

Επίσης, οι κινητές συσκευές προσφέρουν μια σειρά από πλεονεκτήματα λόγω της φύσης τους, όπως:

- Κοινωνική διαδραστικότητα (social interactivity): η ανταλλαγή δεδομένων και η συνεργασία με άλλους εκπαιδευόμενους μπορεί να συμβεί πρόσωπο-με-πρόσωπο
- Ευαισθησία στο γενικότερο πλαίσιο της μάθησης (context sensitivity): οι κινητές συσκευές μπορούν και να συγκεντρώσουν αλλά και να απαντήσουν σε πραγματικά ή προσομοιωμένα δεδομένα που είναι μοναδικά για την παρούσα θέση, περιβάλλον και χρόνο
- Συνδεσιμότητα (connectivity): είναι δυνατή η δημιουργία ενός τοπικού δικτύου με τη σύνδεση κινητών συσκευών σε συσκευές συλλογής δεδομένων, σε άλλες συσκευές, ή σε ένα κοινό δίκτυο
- Ατομικότητα (individuality): η βοήθεια που παρέχεται για περίπλοκες δραστηριότητες μπορεί να προσαρμόζεται στον εκάστοτε μαθητευόμενο.

Σύμφωνα με έρευνες, οι κινητές συσκευές παρουσιάζουν τα ακόλουθα πλεονεκτήματα, αναφορικά με τη χρησιμοποίησή τους στη διαδικασία της μάθησης:

#### ***Βασικά Πλεονεκτήματα των Κινητών Συσκευών στη Μάθηση***

- Ερέθισμα, κίνητρο (motivational stimulus)
- Ευκολία χρήσης (ease of storage and portability)
- Ευκολία και βελτίωση γραπτών εργασιών (improved and easier written work)
- Καλύτερη γνώση για τους υπολογιστές (increased knowledge of computers)
- Διαθεσιμότητα (readily available at all times)

- Εύρος (offered a range of useful functions)
- Διασκέδαση (fun)
- Ευελιξία (flexibility and use outside classroom)
- Κατοχή (personal ownership)
- Ανεξάρτητη εργασία (independent working)
- Καλύτερη πρόσβαση (greater computer access)
- Ενδιαφέρον (more interesting than desktop machine)

Από την άλλη πλευρά, η χρησιμοποίηση κινητών συσκευών μπορεί να δημιουργήσει μια σειρά από προβλήματα, όπως:

#### ***Προβλήματα των Κινητών Συσκευών στη Μάθηση***

- Κόστος: ειδικά λόγω της ραγδαίας εξέλιξης, που απαξιώνει γρήγορα την αντίστοιχη τεχνολογία
- Ευαισθησία: ειδικά για σχολικά περιβάλλοντα
- Τεχνικά προβλήματα: μπαταρία, σύνδεση, συμβατό λογισμικό
- Έλλειψη εκπαιδευτικού λογισμικού

## **6.3 Σχεδιασμός Εφαρμογών για Συνεργατική Μάθηση μέσω Κινητών Συσκευών**

Ο σχεδιασμός εφαρμογών για συνεργατική μάθηση μέσω κινητών συσκευών περιλαμβάνει μια σειρά από παράγοντες που πρέπει να ληφθούν υπόψη.

### ***Βασικές Διαστάσεις για το Σχεδιασμό Περιβαλλόντων Συνεργατικής ΜΚΣ***

#### **1. Δομή των μαθησιακών δραστηριοτήτων που πρόκειται να υποστηριχθούν**

Η σχεδίαση των μαθησιακών δραστηριοτήτων για εφαρμογές συνεργατικής κινητής μάθησης υπόκειται στις ίδιες βασικές αρχές όπως οι γενικότερες εφαρμογές συνεργατικής μάθησης. Η βασική συμβουλή εδώ είναι «σχεδιάστε για την μάθηση και όχι για την χρήση της τεχνολογίας». Συχνά (κινητές) συσκευές υιοθετούνται στη μαθησιακή/διδασκτική διαδικασία χωρίς να προσθέτουν και χωρίς να βελτιώνουν τη μαθησιακή εμπειρία. Εκείνο που πρέπει να θυμάται πάντα ο σχεδιαστής είναι ότι η χρήση της τεχνολογίας δεν είναι αυτοσκοπός, αλλά ένα μέσο που καθιστά εφικτές

δραστηριότητες που είναι αδύνατον να εκτελεστούν με άλλο τρόπο, ή που βελτιώνει την απόδοση του μαθητή ή και της ομάδας.

## 2. Επιλογή δικτύου υποδομής (network infrastructure)

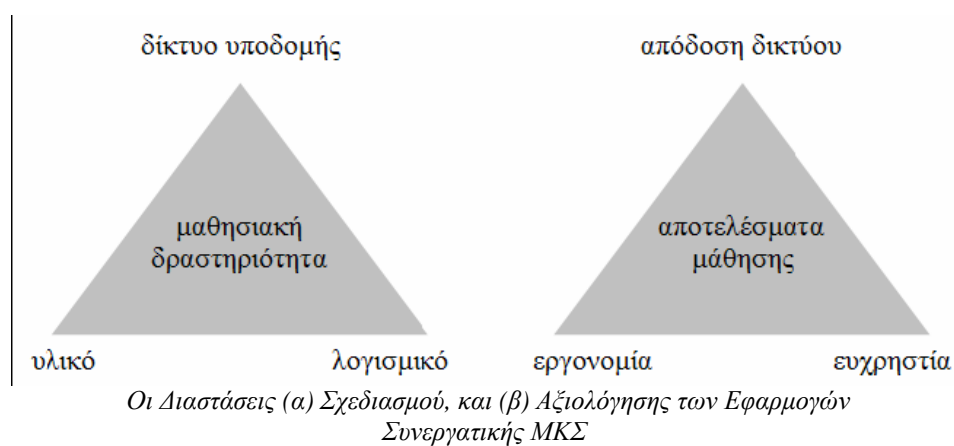
- Υποδομή δικτύων επικοινωνίας
  - δορυφορικά δίκτυα, δίκτυα κινητής τηλεφωνίας, ασύρματα τοπικά δίκτυα και ad hoc προσωπικά δίκτυα
  - Παράμετροι που λαμβάνονται υπόψη
    1. ο αριθμός των χρηστών
    2. η ακτίνα σύνδεσης (range)
    3. οι ανάγκες συνδεσιμότητας και πρόσβασης σε δεδομένα
    4. ο τόπος/χώρος χρήσης
    5. η παρουσία άλλων συσκευών που μπορεί να δημιουργούν «θόρυβο»
    6. οι ανάγκες ασφάλειας
    7. το κόστος για τον παροχέα ή και για τον χρήστη
    8. και η τοπολογία του δικτύου
- Υποδομή για εφαρμογές που αποκρίνονται με βάση τη θέση του χρήστη (positioning systems)
  - υπέρυθρες ακτίνες (infrared), λύσεις δικτύου όπως Bluetooth, WiFi και ultra wide band, συστήματα ραδιοσυχνότητας όπως GPS και RFID (Radio Frequency Identification tags), και υβριδικά συστήματα Ραδιοσυχνότητας, π.χ. που κάνουν χρήση υπερήχων
  - Παράγοντες που λαμβάνονται υπόψη:
    1. ανάγκες ακριβείας θέσεως (π.χ. διαφορετικές ανάγκες για χρήση σε εσωτερικό χώρο ή σε εξωτερικό),
    2. ανάγκες συνεχούς παρακολούθησης της θέσης του χρήστη (real time positioning)
    3. και το αν η ενημέρωση θέσης γίνεται αυτόματα ή κατόπιν δήλωσης της θέσης από το χρήστη

### **3. Επιλογή του υλικού (κινητές συσκευές, τεχνολογίες, κλπ)**

- Laptop/notebook, tables, personal digital assistants (PDAs), και κινητά τηλέφωνα (συμβατικά ή smart phones)
- Η επιλογή ενός είδους συσκευής εξαρτάται
  - από τις ανάγκες για υπολογιστική δύναμη (processing power)
  - για διάρκεια μπαταρίας
  - ανθεκτικότητα (robustness)
  - καθώς και από παράγοντες που έχουν να κάνουν με θέματα εργονομίας (π.χ. τη στάση του χρήστη κατά τη χρήση,
    1. δηλαδή εάν έχει ελεύθερο ένα ή και τα δύο χέρια
    2. εάν κινείται σωματικά ή είναι στάσιμος
    3. πόσο βάρος μπορεί να σηκώσει, και άλλα ζητήματα εύκολης μεταφοράς
    4. τον τόπο, χρόνο και τα ιδιαίτερα χαρακτηριστικά του περιβάλλοντος, όπως η θερμοκρασία και ο θόρυβος).
  - το μοντέλο διαχείρισης ιδιαίτερα στην περίπτωση που το υλικό παρέχεται από το εκπαιδευτικό ίδρυμα, διευκρινίζοντας:
    1. θα παράσχουμε ατομικές συσκευές ή συσκευές για ομάδες χρηστών;
    2. θα μπορούν οι μαθητές να πάρουν τις συσκευές μαζί τους έξω από το χώρο του σχολείου ή όχι;
    3. ποιος θα είναι υπεύθυνος για τυχόν βλάβες; κλπ.

### **4. σχεδίαση του λογισμικού**

- Σχεδιασμού της διεπιφάνειας χρήσης
- Του συστήματος βοήθειας
- Και της αλληλεπίδρασης

**Εικόνα 7**

Όπως φαίνεται στην παραπάνω εικόνα, στο (α)η επιλογή δικτύου υποδομής, υλικού και λογισμικού είναι αλληλοεξαρτώμενες και συνυφασμένες με τη σχεδίαση των μαθησιακών δραστηριοτήτων. Στο (β) παρουσιάζει τις διαστάσεις αξιολόγησης συστημάτων συνεργατικής ΜΚΣ σε αντιστοιχία με τις διαστάσεις σχεδίασης.



## Βιβλιογραφία για το 2<sup>ο</sup> ΚΕΦΑΛΑΙΟ

- [1]. <http://pages.cs.aueb.gr/~toumpis/courses/ECE453/history.pdf>
- [2]. <http://www.tanea.gr/default.asp?pid=2&artid=4560874&ct=2>
- [3]. [http://nemertes.lis.upatras.gr/dspace/bitstream/123456789/515/1/Nimertis\\_Papazois%28m%29.pdf](http://nemertes.lis.upatras.gr/dspace/bitstream/123456789/515/1/Nimertis_Papazois%28m%29.pdf)
- [4]. Βιβλίο: Introduction to 3G Mobile Communications 2<sup>nd</sup> edition  
Joha Korhonen, Artech House Mobile Communications Series
- [5]. Εφαρμογές Κινητής Τεχνολογίας – Εισαγωγή (Mobile01.pdf)  
Σταύρος Καμμάς, Τμήμα Πολιτισμικής Τεχνολογίας και Επικοινωνίας  
Πανεπιστήμιο Αιγαίου
- [6]. Εφαρμογές Κινητής Τεχνολογίας – Τεχνολογίες και Χρήση Κινητών Συσκευών (Mobile02.pdf)  
Σταύρος Καμμάς, Τμήμα Πολιτισμικής Τεχνολογίας και Επικοινωνίας  
Πανεπιστήμιο Αιγαίου
- [7]. Βιβλίο: Δίκτυα Υπολογιστών  
Andrew S.Tanenbaum  
Τέταρτη Αμερικάνικη Έκδοση  
Εκδόσεις Κλειδάριθμος
- [8]. <http://karagian.users.uth.gr/cscl/14-Karagiannidis-Vavoula.pdf>
- [9]. Συνεργατική μάθηση μέσω κινητών συσκευών – Mobile learning: Διάλεξη 9  
Σταύρος Καμμάς  
(07-Mobile Learning.ppt)
- [10]. <http://el.wikipedia.org/wiki/Java>
- [11]. <http://www.it.uom.gr/project/java/tutorial.htm>
- [12]. Βιβλίο: Πλήρες Εγχειρίδιο της Java 6,  
Rogers Cadenhead, Laura Lemay,  
Εκδόσεις: Μ. Γκιούρδας
- [13]. <http://developers.sun.com/mobility/getstart/>
- [14]. <http://www.oracle.com/technetwork/java/javame/tech/index.html>
- [15]. Βιβλίο: Beginning J2ME From Novice to Professional  
Jonathank Knudsen and Sing Li  
Third Edition, Apress
- [16]. <http://java.sun.com/javame/technology/index.jsp>

- [17]. <http://hamilton.bell.ac.uk/swdev3and4/lecturenotes/j2me.pdf>
- [18]. Βιβλίο: Wireless Programming with J2ME-Cracking the code  
Dreamtech Software Team  
Hungry Minds
- [19].  
<http://java.sun.com/javame/reference/apis/jsr037/javax/microedition/midlet/package-summary.html>
- [20]. <http://download.oracle.com/javase/1.4.2/docs/api/java/io/package-summary.html>
- [21]. <http://download.oracle.com/javase/1.4.2/docs/api/java/lang/package-summary.html>
- [22]. <http://download.oracle.com/javase/1.4.2/docs/api/java/util/package-summary.html>
- [23]. <http://download.oracle.com/javase/1.4.2/docs/api/java/security/package-summary.html>
- [24].  
<http://java.sun.com/javame/reference/apis/jsr118/javax/microedition/rms/package-summary.html>
- [25]. <http://www.cs.aau.dk/~dahl/diverse/j2me-midp-api/javax/microedition/lcdtui/package-summary.html>
- [26]. [http://el.wikipedia.org/wiki/%CE%91%CF%83%CF%8D%CF%81%CE%BC%CE%B1%CF%84%CE%BF\\_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF](http://el.wikipedia.org/wiki/%CE%91%CF%83%CF%8D%CF%81%CE%BC%CE%B1%CF%84%CE%BF_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF)
- [27]. Βιβλίο: Bluetooth Application Programming with the JAVA APIs  
Essentials Edition  
Timothy J. Thompson, Paul J. Kline, C Bala Kumar  
Morgan Kaufmann Publishers
- [28]. <http://developers.sun.com/mobility/getstart/>
- [29]. [http://de.teikav.edu.gr/telematics/pdf/3o\\_Meros\\_Asymata\\_thlematikh.pdf](http://de.teikav.edu.gr/telematics/pdf/3o_Meros_Asymata_thlematikh.pdf)
- [30]. <http://developers.sun.com/mobility/apis/articles/bluetoothintro/index.html>
- [31]. <http://developers.sun.com/mobility/midp/articles/bluetooth2/>
- [32]. Βιβλίο: Βάσεις Δεδομένων  
Μιχαήλ Ξένος, Δημήτριος Χριστοδουλάκης  
Ελληνικό Ανοικτό Πανεπιστήμιο, Σχολή Θετικών Επιστημών & Τεχνολογίας  
<http://www.free-ebooks.gr/gr/e-rafi.php?id=159>

- [33]. <http://www.pre.aegean.gr/Documents/StuffFiles/tsolak/Access01.pdf>
- [34]. <http://www.neural.uom.gr/Documents/DataBases/chapter7.pdf>
- [35]. <http://dmst.aueb.gr/dds/c3/sql/rel.htm>
- [36]. <http://www.msaccess.gr/index.php/el/access-tutorials/41-access->
- [37]. <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorial-Office-Access.pdf>
- [38]. [http://wiki.forum.nokia.com/index.php/Series\\_40](http://wiki.forum.nokia.com/index.php/Series_40)
- [39]. [http://wiki.forum.nokia.com/index.php/Series\\_40: Editions and Feature Packs](http://wiki.forum.nokia.com/index.php/Series_40:_Editions_and_Feature_Packs)
- [40]. <http://netbeans.org/kb/trails/mobility.html>
- [41]. <http://netbeans.org/community/releases/69/>
- [42]. Αρχείο .ppt downloaded from Google:  
**Εισαγωγή στη Java, Μέρος Α´**,  
Βασίλης Παπαταξιάρχης  
Μάθημα: Αντικειμενοστραφής Προγραμματισμός  
Διδάσκουσα: Ιζαμπώ Καράλη  
Τμήμα Πληροφορικής και Τηλεπικοινωνιών  
Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
- [43]. <http://www.oracle.com/technetwork/java/download-135801.html>

# Κεφάλαιο 3<sup>ο</sup>

## Εισαγωγή

Το κεφάλαιο αυτό αναφέρεται στα τεχνικά χαρακτηριστικά (σχεδιασμό, ανάλυση, προγραμματισμό) και στην εφαρμογή της πτυχιακής μας εργασίας. Αναπτύσσεται σε δύο μέρη:

- Στο πρώτο μέρος περιγράφουμε τον σχεδιασμό και την υλοποίηση της Βάσης Δεδομένων. Επίσης, φτιάχνουμε και περιγράφουμε το πρόγραμμα που θα συνδέεται και θα χρησιμοποιεί τη Βάση Δεδομένων. Εκτελώντας SQL ερώτημα δημιουργούμε δυο txt αρχεία που μας είναι απαραίτητα για την συνέχεια.
- Στο δεύτερο μέρος, θα ασχοληθούμε με τη δημιουργία νέου project χρησιμοποιώντας MIDlets. Έχοντας ολοκληρώσει τον κώδικα, δημιουργούμε το jar αρχείο της εφαρμογής και το μεταφέρουμε στο κινητό μας τηλέφωνο μέσω της τεχνολογίας Bluetooth.

# Μέρος 1<sup>ο</sup>

Χρησιμοποιήσαμε το πρόγραμμα Microsoft Office Access 2003 για να φτιάξουμε τη Βάση Δεδομένων. Δημιουργήσαμε μια καινούργια Βάση Δεδομένων και σκοπός μας ήταν να φτιάξουμε δυο πίνακες.

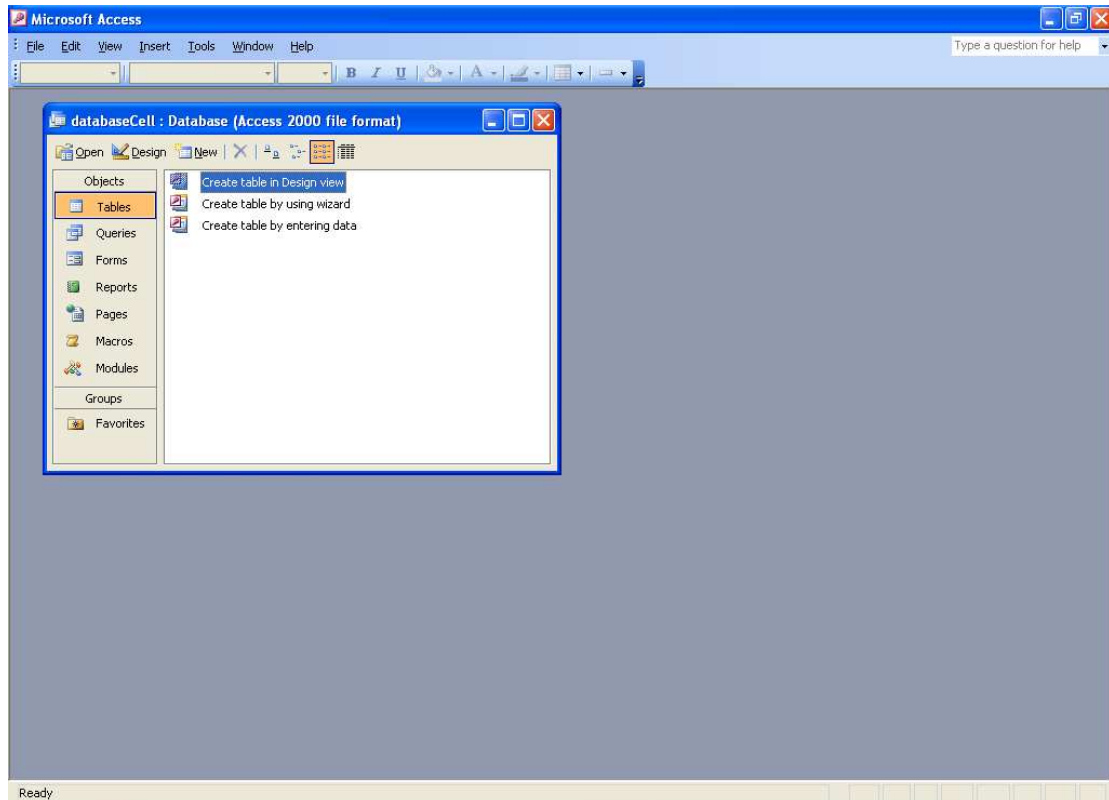
Ο πρώτος πίνακας θα αφορά τον αριθμό της ερώτησης, την ερώτηση, τις τέσσερις πιθανές (προτεινόμενες προς επιλογή) απαντήσεις στην ερώτηση αυτή, την σωστή απάντηση και τον αριθμό-δείκτη, που αντιστοιχεί στο αντίστοιχο μάθημα το σχετικό με την ερώτηση (δηλαδή το μάθημα που είναι καταχωρημένη η ερώτηση). Δηλαδή, η ερώτηση μπορεί να αφορά το μάθημα των Υπολογιστών ή των Μαθηματικών ή της Φυσικής.

Ο δεύτερος πίνακας έχει δυο στήλες. Η μια στήλη θα αντιστοιχεί στον αριθμό του μαθήματος και η δεύτερη στήλη στο όνομα του μαθήματος.

Παρακάτω βλέπουμε την διαδικασία που ακολουθήσαμε για την δημιουργία της Βάσης Δεδομένων μας.

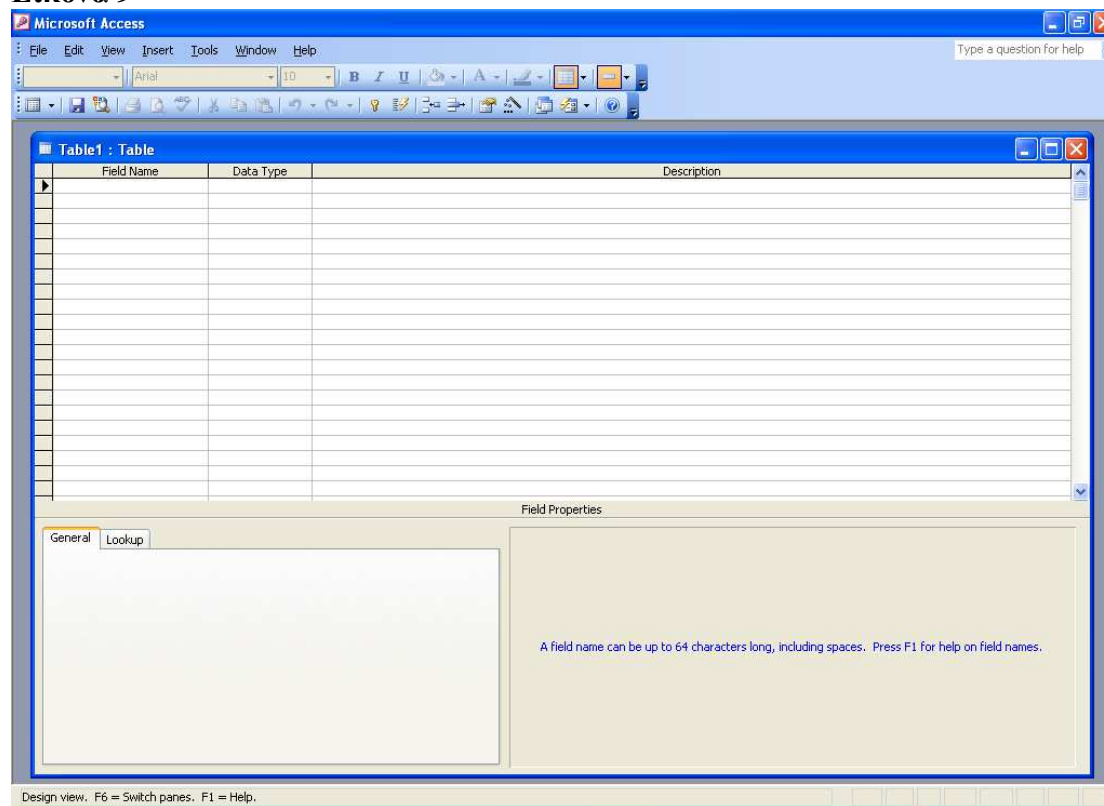
Δημιουργήσαμε ένα καινούργιο αρχείο και το ονομάσαμε databaseCell.mdb

## Εικόνα 8



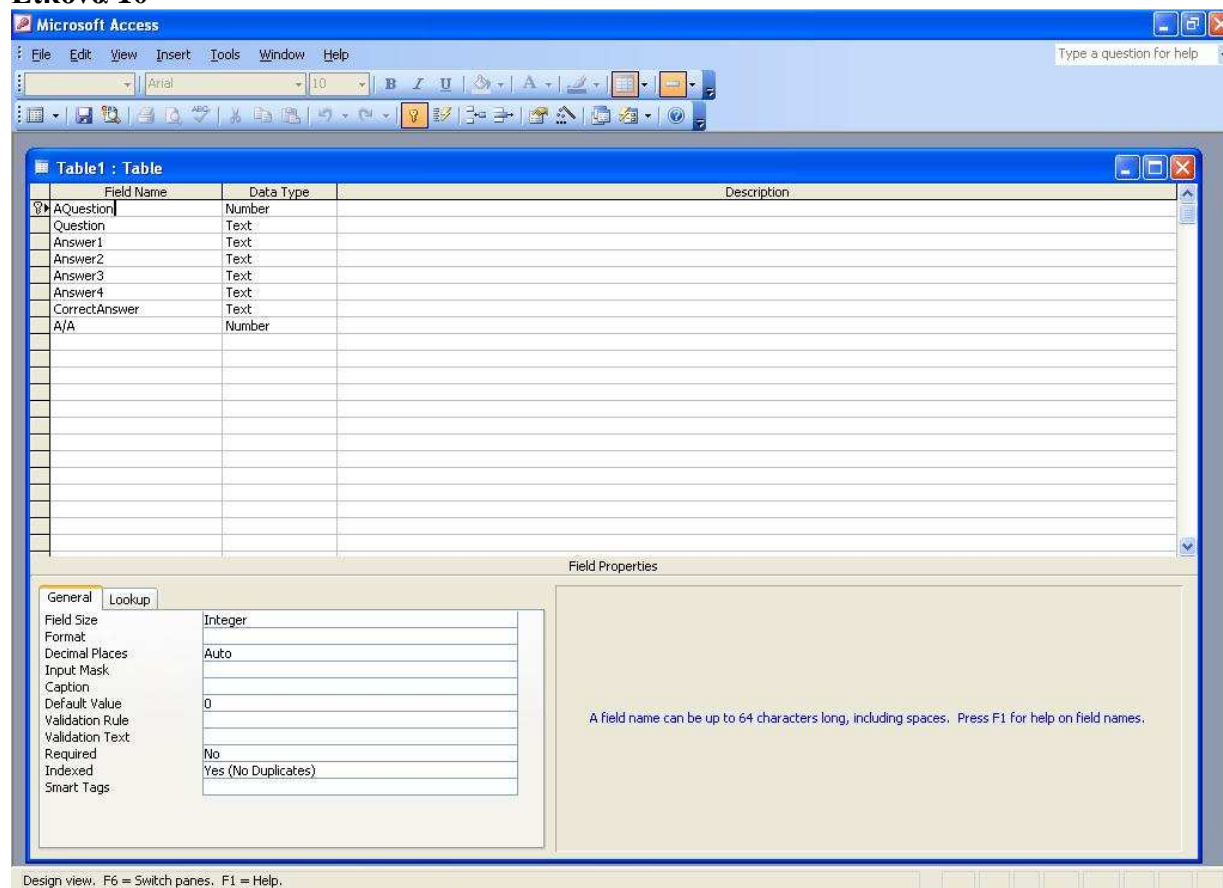
Έχοντας επιλέξει το Create table in Design view όπως βλέπουμε παραπάνω, στη συνέχεια, κάνουμε κλικ στο Design και μας εμφανίζει την παρακάτω καρτέλα.

### Εικόνα 9



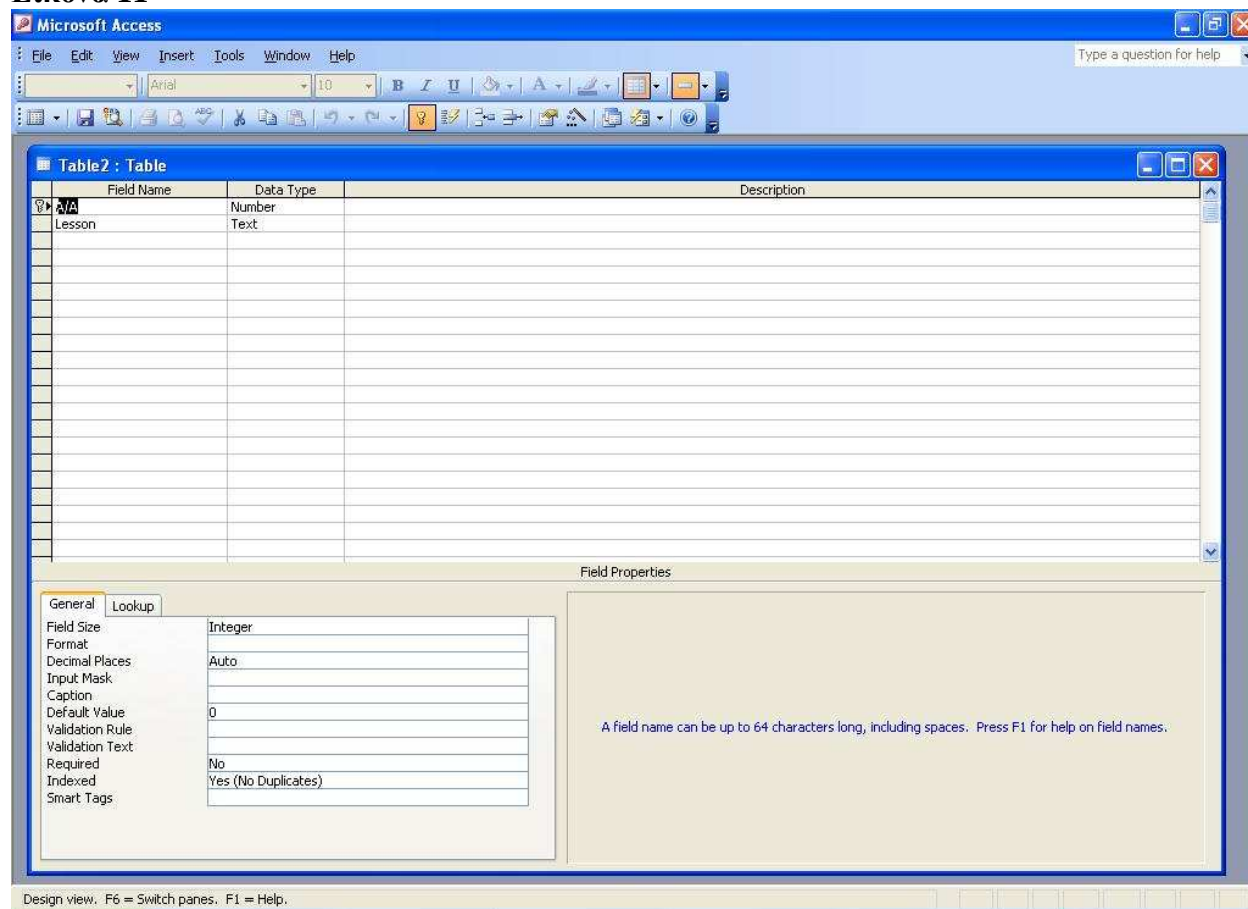
Σ' αυτή την καρτέλα ορίζουμε τις στήλες και τα περιεχόμενα τους για τον πρώτο πίνακα. Δηλαδή, στο Field Name γράφουμε το όνομα της στήλης, στο Data Type γράφουμε τύπους δεδομένων δηλαδή αν το περιεχόμενο της στήλης είναι text, number, date κλπ. Στη συνέχεια, ορίζουμε το πρωτεύον κλειδί. Πρωτεύον κλειδί για τον πρώτο πίνακα είναι το AQuestion. Τέλος, αποθηκεύουμε τον πίνακα με το όνομα Table1. Παρακάτω βλέπουμε συμπληρωμένο τον πίνακα.

### Εικόνα 10



Στη συνέχεια, δημιουργούμε με τον ίδιο τρόπο και τον δεύτερο πίνακα και τον ονομάζουμε Table2. Παρακάτω βλέπουμε συμπληρωμένο και τον δεύτερο πίνακα.

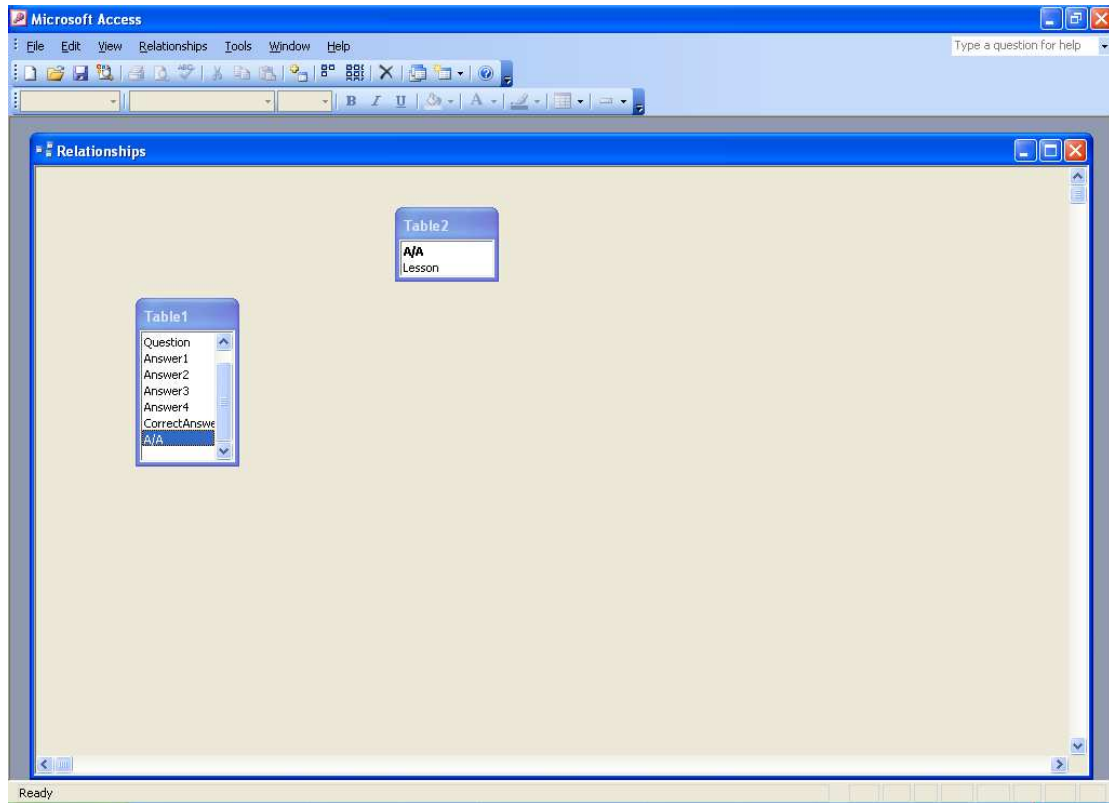
**Εικόνα 11**





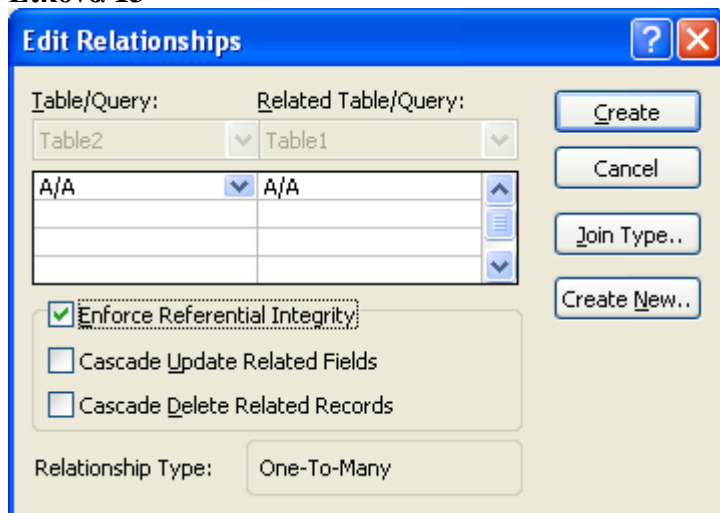
Έχοντας ορίσει τους δυο πίνακες θα πρέπει να δούμε στη συνέχεια την σχέση μεταξύ των δύο πινάκων. Η επεξεργασία και ο ορισμός των σχέσεων γίνεται από την επιλογή Tools -> Relationships.

**Εικόνα 12**



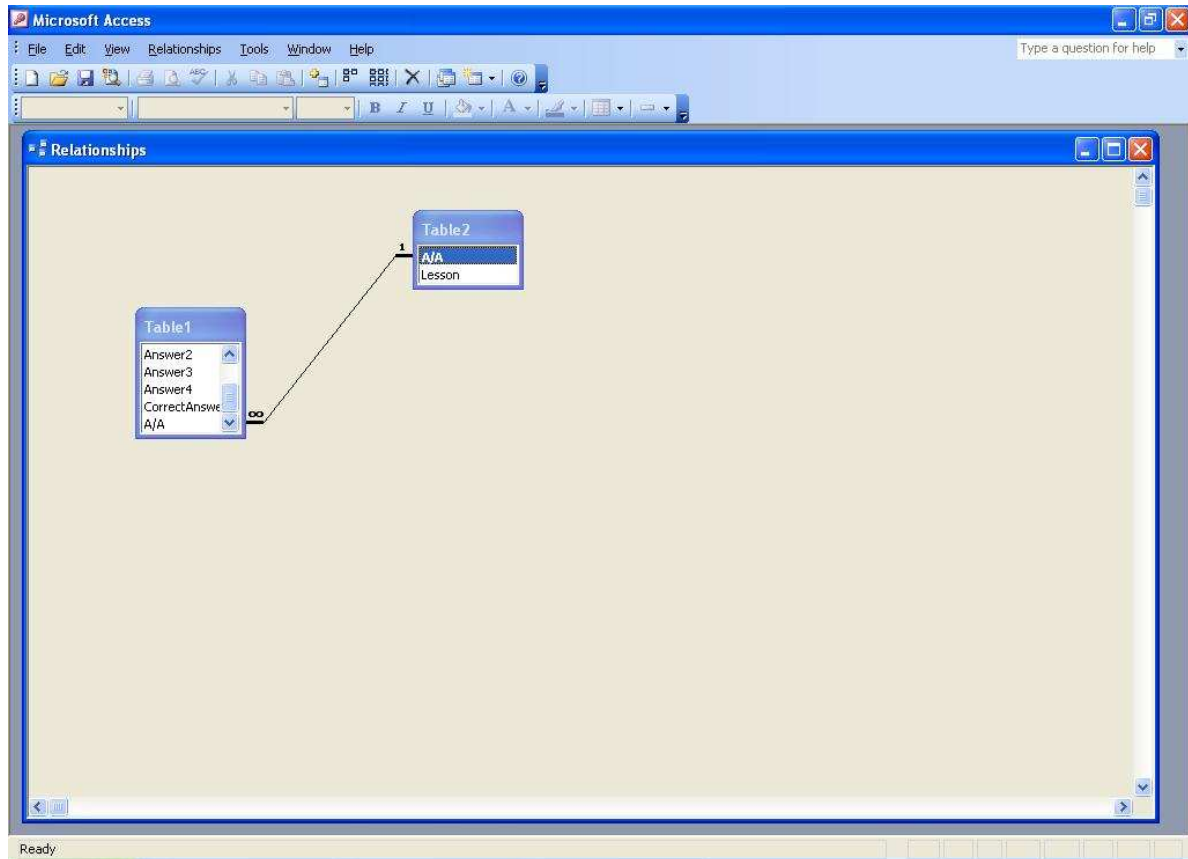
Για να κάνουμε σύνδεση των δυο πινάκων επιλέξαμε από το πίνακα Table1 το A/A και το ενώσαμε με το A/A του πίνακα Table2. Μας εμφανίζεται το παράθυρο Edit Relationships και μας εμφανίζει τον παρακάτω πίνακα.

**Εικόνα 13**



Τσεκάρουμε το πρώτο κουτάκι Enforce Referential Integrity. Η σχέση που προκύπτει από τους δυο παραπάνω πίνακες είναι ένα προς πολλά. Τέλος, πατάμε το κουμπί Create και βλέπουμε το αποτέλεσμα στην παρακάτω εικόνα.

**Εικόνα 14**



Έχοντας τελειώσει την δημιουργία των δύο πινάκων μπορούμε να «γεμίσουμε» την Βάση Δεδομένων μας με τις ερωτήσεις, τις πιθανές απαντήσεις, τις σωστές απαντήσεις και για ποιο μάθημα απευθύνεται. Για να «γεμίσουμε» την Βάση μας θα πρέπει να έχουμε κάνει ένα κλικ πάνω στον πίνακα Table1 για να τον ενεργοποιήσουμε και στη συνέχεια να κάνουμε κλικ στο Open. Ισχύει η ίδια διαδικασία και για τον πίνακα Table2.

Πριν ξεκινήσουμε να προγραμματίζουμε σε Java και συνδέσουμε το πρόγραμμα μας με την Βάση, μπορούμε να κάνουμε ένα τεστ για να δούμε αν δουλεύει σωστά η Βάση Δεδομένων που φτιάξαμε χρησιμοποιώντας την δομημένη γλώσσα ερωτημάτων **SQL** (Structured Query Language). Η διαδικασία είναι η εξής:

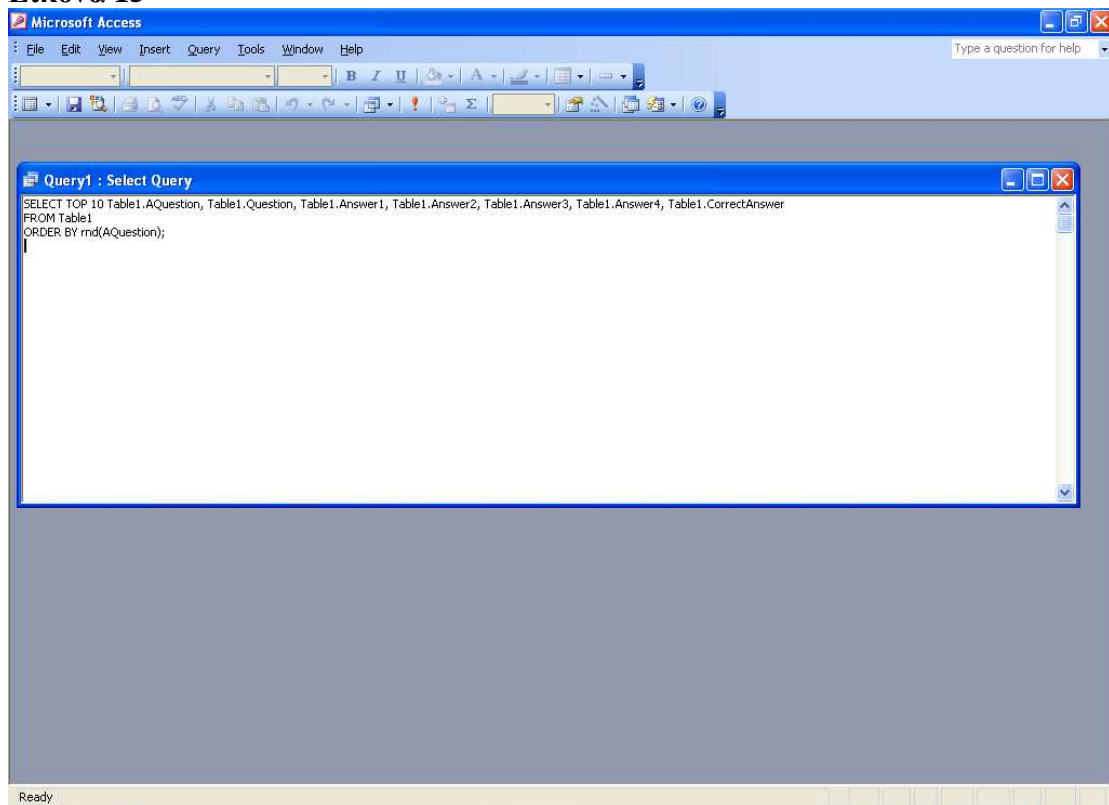
Από την αριστερή στήλη που λέγεται Objects επιλέγουμε την δεύτερη επιλογή που λέγεται Queries. Στη συνέχεια κάνουμε κλικ στο Design. Έπειτα, κλείνουμε το παράθυρο που μας εμφανίζεται και από το View επιλέγουμε SQL View.

Στη συνέχεια, γράφουμε το παρακάτω ερώτημα. (Αυτό που βλέπουμε και στην παρακάτω εικόνα).

```
SELECT TOP 10 Table1.AQuestion, Table1.Question, Table1.Answer1,
Table1.Answer2, Table1.Answer3, Table1.Answer4, Table1.CorrectAnswer
FROM Table1
ORDER BY rnd(AQuestion);
```

Με την παραπάνω εντολή επιλέγουμε 10 φορές τα πεδία AQuestion, Question, Answer1, Answer2, Answer3, Answer4, CorrectAnswer από τον πίνακα Table1 ταξινομημένα με τυχαίο τρόπο ως προς τον αριθμό της ερώτησης (AQuestion)

### Εικόνα 15



Έπειτα, αποθηκεύουμε το ερώτημα και στη συνέχεια κάνουμε κλικ στο Open και βλέπουμε τα αποτελέσματα. Τα αποτελέσματα είναι σωστά και μπορούμε να συνεχίσουμε το project μας.

Πριν ασχοληθούμε με προγραμματισμό πρέπει να συνδέσουμε την βάση μας. Το **JDBC** (Java Database Connectivity), είναι μια βιβλιοθήκη κλάσεων που συνδέουν προγράμματα Java με σχεσιακές βάσεις δεδομένων (relational databases).

Η βιβλιοθήκη JDBC περιλαμβάνει κλάσεις για όλες τις κοινές εργασίες που σχετίζονται με την χρήση βάσεων δεδομένων:

- Υλοποίηση μιας σύνδεσης με μια βάση δεδομένων.
- Δημιουργία μιας πρότασης με την χρήση της SQL.
- Εκτέλεση αυτού του ερωτήματος SQL στην βάση δεδομένων
- Εμφάνιση των εγγραφών που επιστρέφει το ερώτημα

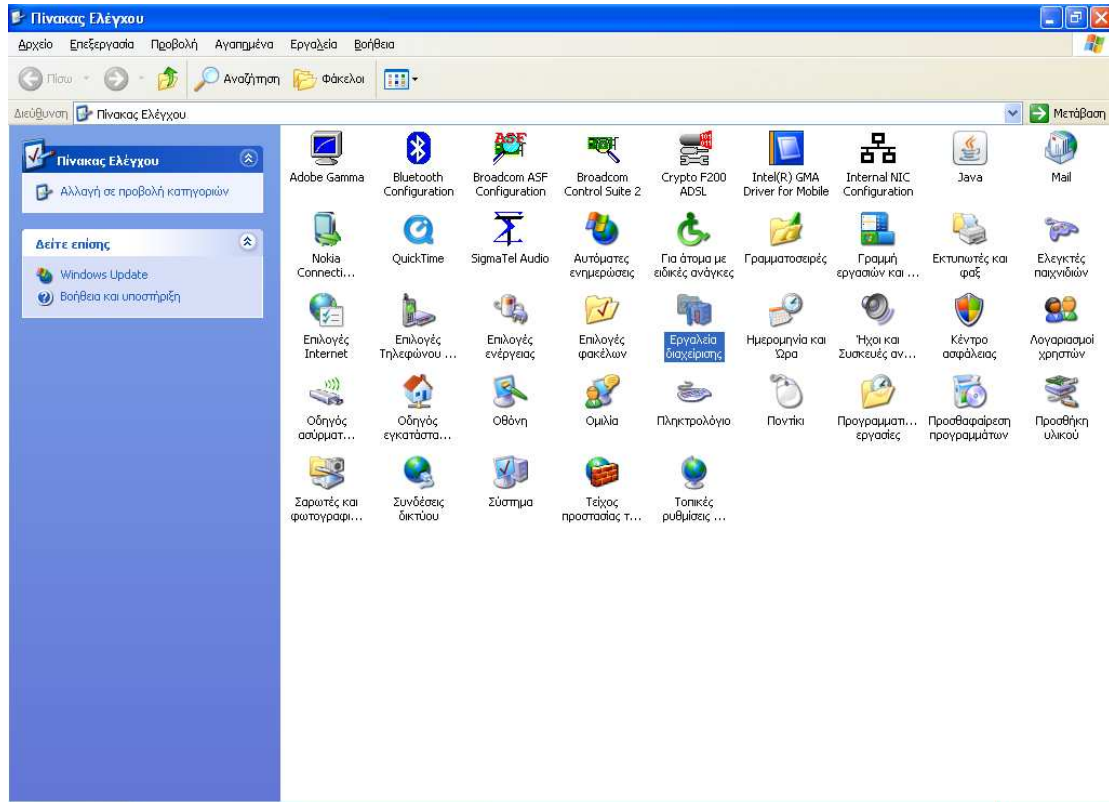
Όλες αυτές οι κλάσεις του JDBC αποτελούν μέρος του πακέτου java.sql.

Το **ODBC** (Open Database Connectivity) είναι το κοινό περιβάλλον διασύνδεσης που χρησιμοποιεί η Microsoft για την προσπέλαση βάσεων δεδομένων SQL, και η διαχείριση του σε ένα σύστημα με τα Windows γίνεται με το ODBC Data Source Administrator.

Παρακάτω είναι τα βήματα που ακολουθήσαμε για την δημιουργία και την εγκατάσταση του ODBC για την Microsoft Access έτσι ώστε στη συνέχεια να γίνει η σύνδεση με την βάση μας databaseCell.mdb.

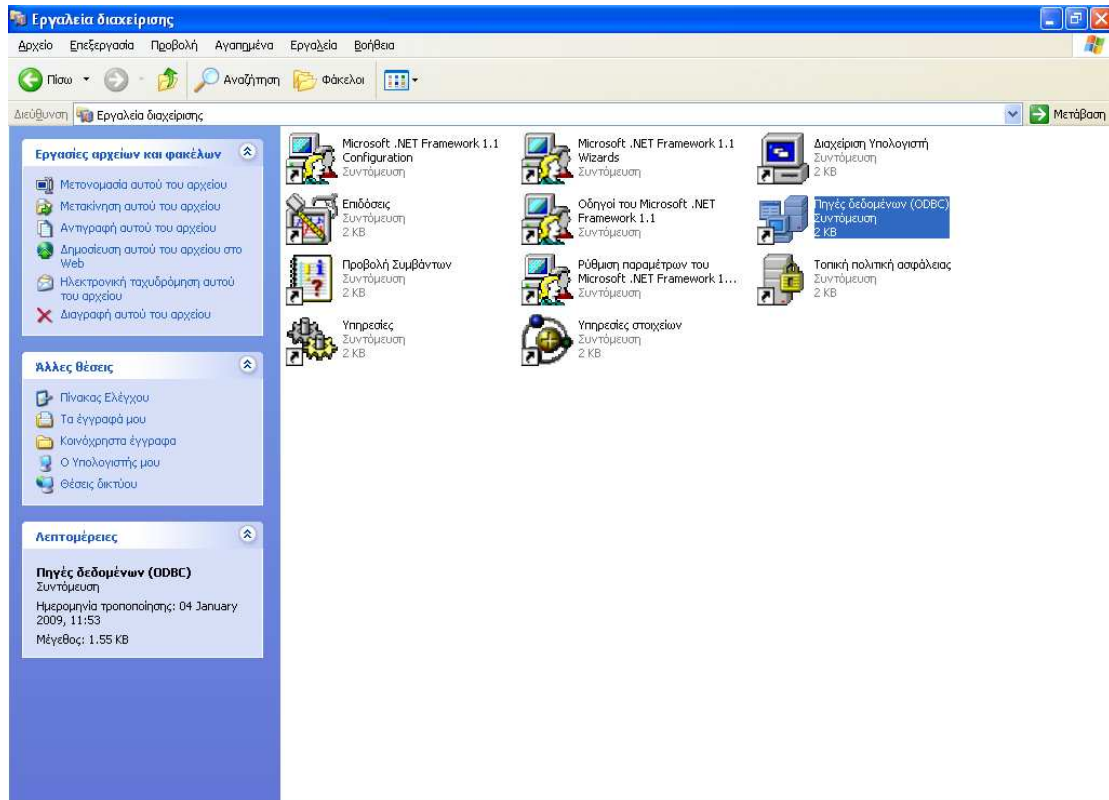
Στα Windows XP επιλέγουμε Έναρξη -> Πίνακας Ελέγχου και μας ανοίγει το παρακάτω παράθυρο.

**Εικόνα 16**



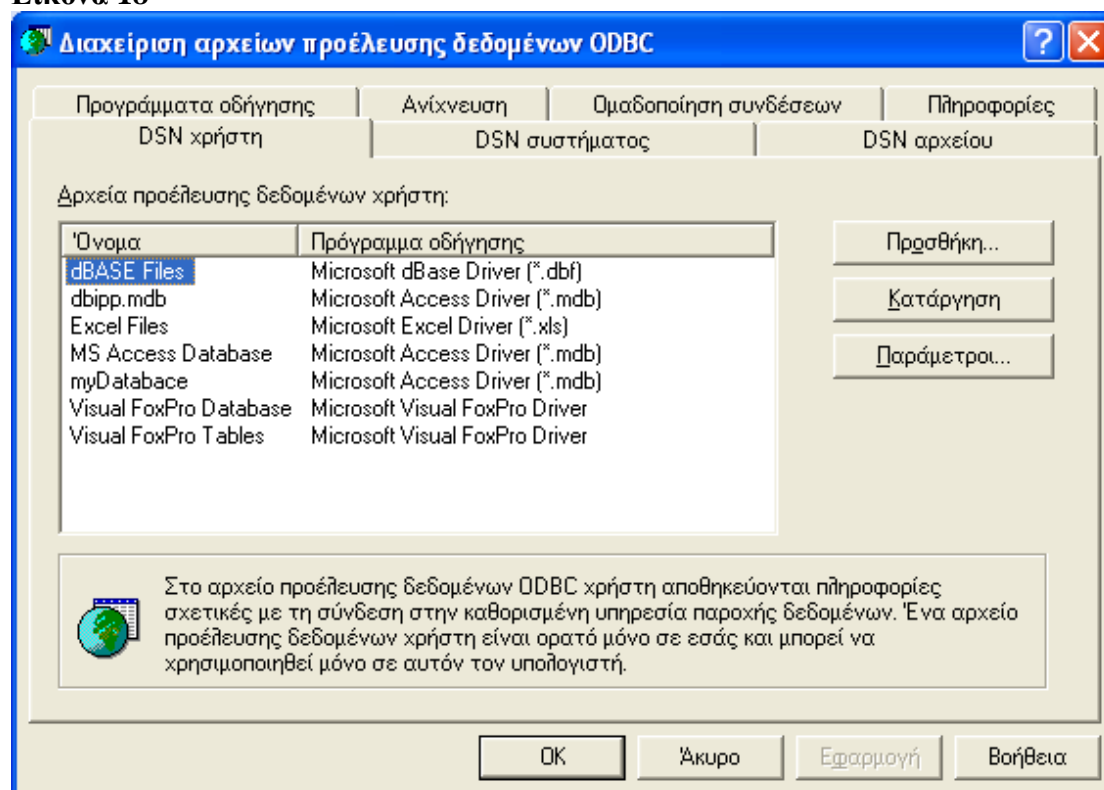
Στη συνέχεια, κάνουμε διπλό κλικ στα Εργαλεία Διαχείρισης και μας εμφανίζεται το παρακάτω παράθυρο.

**Εικόνα 17**



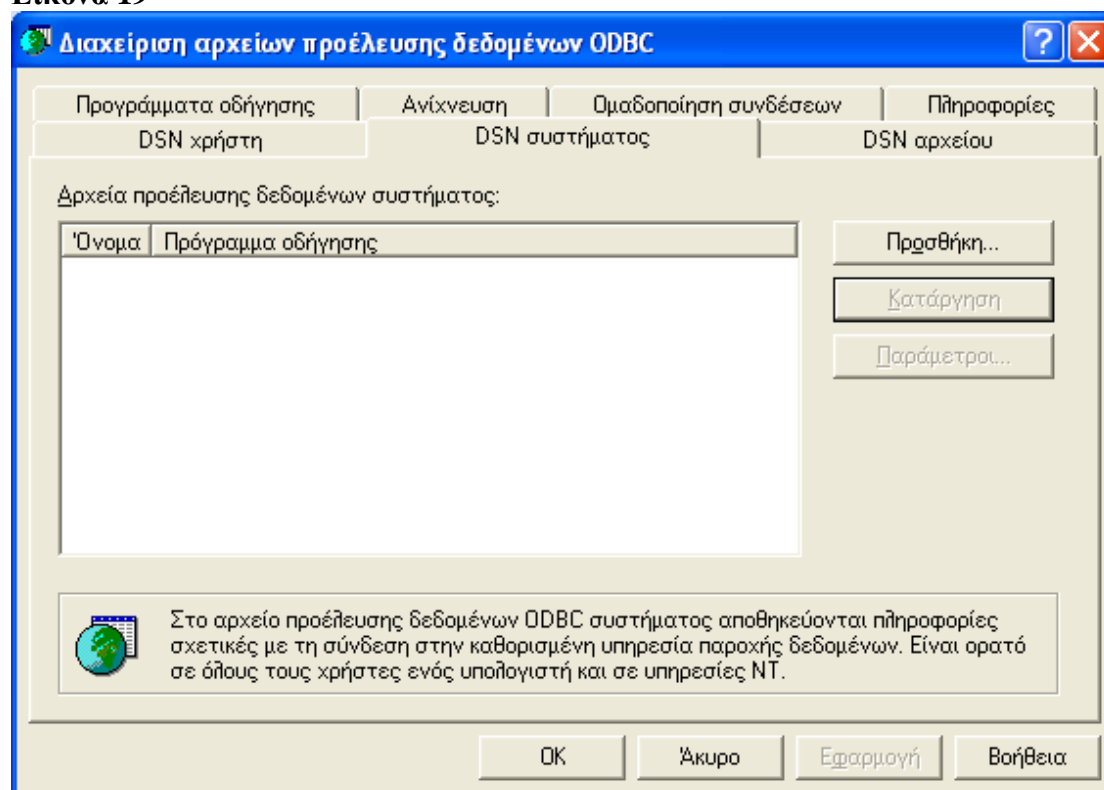
Μετά, κάνουμε διπλό κλικ στο «Πηγές Δεδομένων (ODBC)» και μας εμφανίζει το παρακάτω παράθυρο.

**Εικόνα 18**



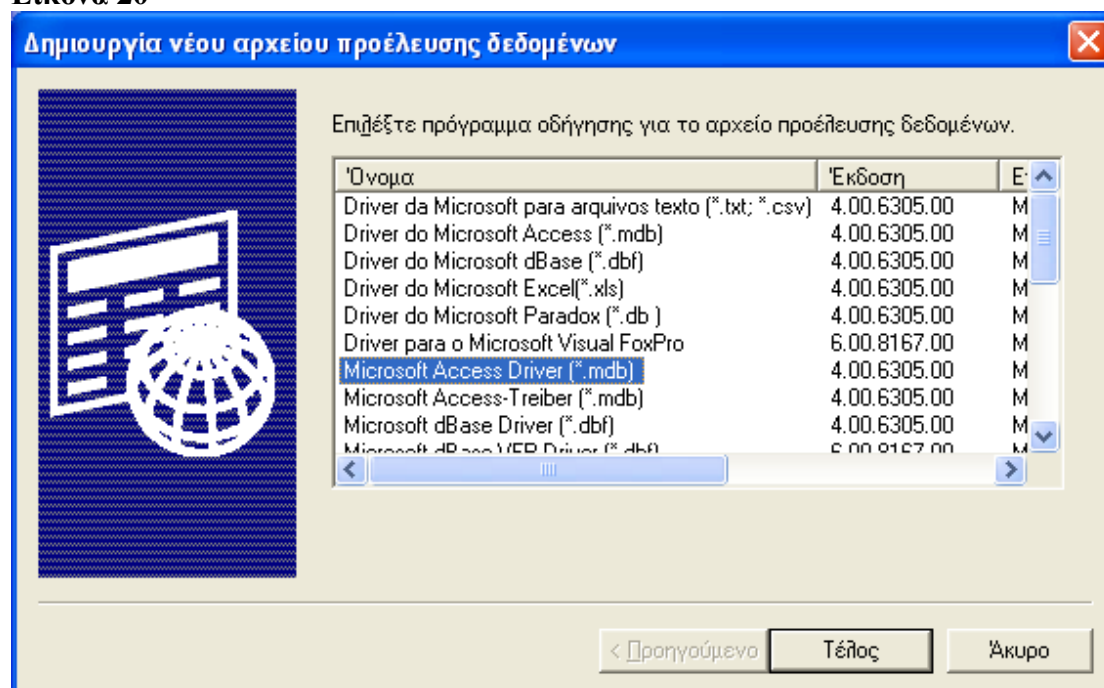
Κάνουμε κλικ στην καρτέλα DSN συστήματος και βλέπουμε το παρακάτω παράθυρο.

**Εικόνα 19**



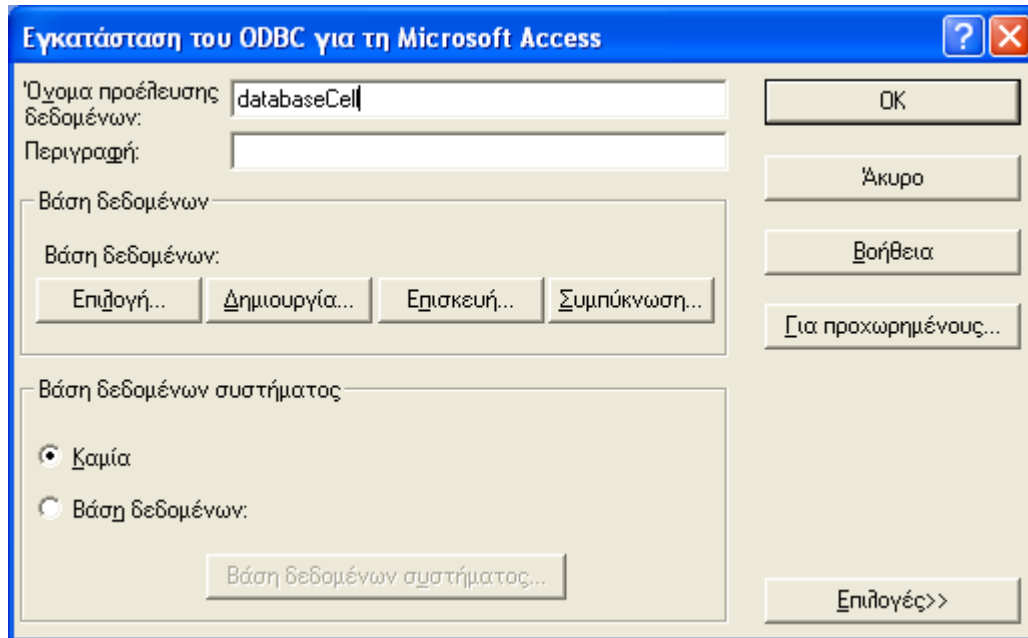
Στη συνέχεια, κάνουμε κλικ στο κουμπί «Προσθήκη».

**Εικόνα 20**



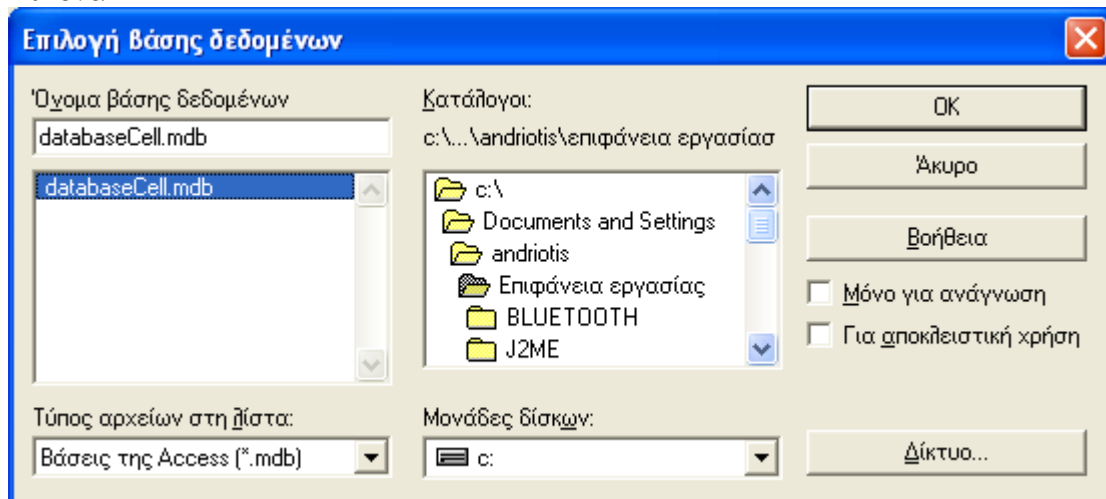
Εμφανίζεται το παραπάνω παράθυρο «Δημιουργία νέου αρχείου προέλευσης δεδομένων» και επιλέγουμε Microsoft Access Driver (\*.mdb) και κάνουμε κλικ στο κουμπί «Τέλος» και μας εμφανίζεται το παρακάτω παράθυρο.

**Εικόνα 21**



Μετά, γράφουμε στο «Όνομα προέλευσης δεδομένων» το αρχείο της βάσης μας που έχουμε φτιάξει, δηλαδή databaseCell. Στη συνέχεια, κάνουμε κλικ στο κουμπί «Επιλογή» της Βάσης Δεδομένων και βλέπουμε το παρακάτω παράθυρο.

**Εικόνα 22**



Επιλέγουμε από τον κατάλογο το φάκελο στο οποίο βρίσκεται αποθηκευμένη η βάση δεδομένων. Εμφανίζεται το όνομα της βάσης δεδομένων πάνω αριστερά στο παράθυρο και μόλις το επιλέξουμε πατάμε το κουμπί OK.



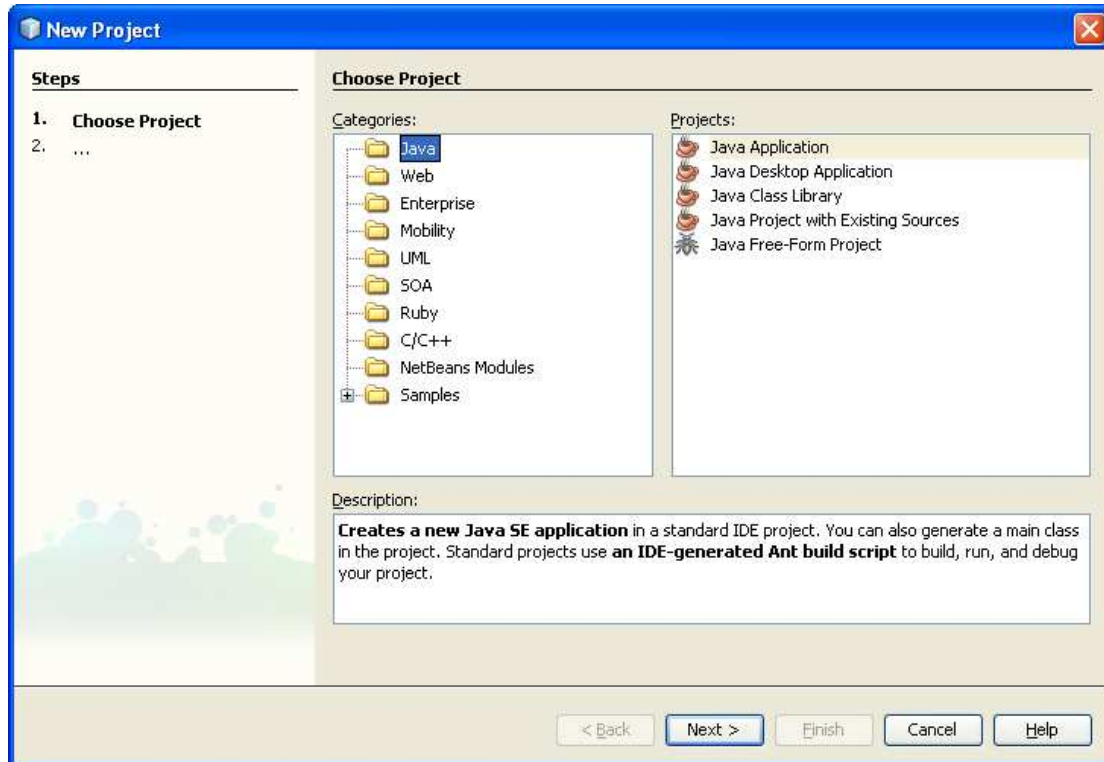
Τέλος, πατάμε το κουμπί OK όπως γυρίζουμε πίσω στα προηγούμενα παράθυρα που είναι ακόμα ενεργοποιημένα. Η σύνδεση μας με τη βάση είναι επιτυχής!

Στη συνέχεια, χρησιμοποιώντας το πρόγραμμα Netbeans 6.1 αναπτύξαμε ένα πρόγραμμα που θα συνδέεται καταρχήν με τη Βάση Δεδομένων που έχουμε φτιάξει. Η δουλειά του προγράμματος αυτού είναι να διαλέγει με τυχαίο τρόπο (random) 10 από τις 100 ερωτήσεις που βρίσκονται αποθηκευμένες στη βάση μαζί με τις τέσσερις πιθανές απαντήσεις για την κάθε ερώτηση. Οι ερωτήσεις και οι πιθανές απαντήσεις αποθηκεύονται όλα μαζί σε ένα txt αρχείο με το όνομα Game.txt. Επίσης, για να ξεχωρίσουμε την κάθε ερώτηση με τις τέσσερις πιθανές απαντήσεις της έχουμε τοποθετήσει ένα αστερίσκο (\*) για διαχωριστικό. Τέλος, φυσικά για κάθε ερώτηση υπάρχει και η σωστή απάντηση. Έτσι λοιπόν, όταν επιλέγονται οι 10 πιθανές ερωτήσεις αυτομάτως σε ένα δεύτερο txt αρχείο με το όνομα Answers.txt αποθηκεύονται μόνο οι σωστές απαντήσεις.

Παρακάτω βλέπουμε τα βήματα για τη δημιουργία του προγράμματος και στη συνέχεια τον κώδικα που αναπτύξαμε συνοδευόμενο από σχόλια.

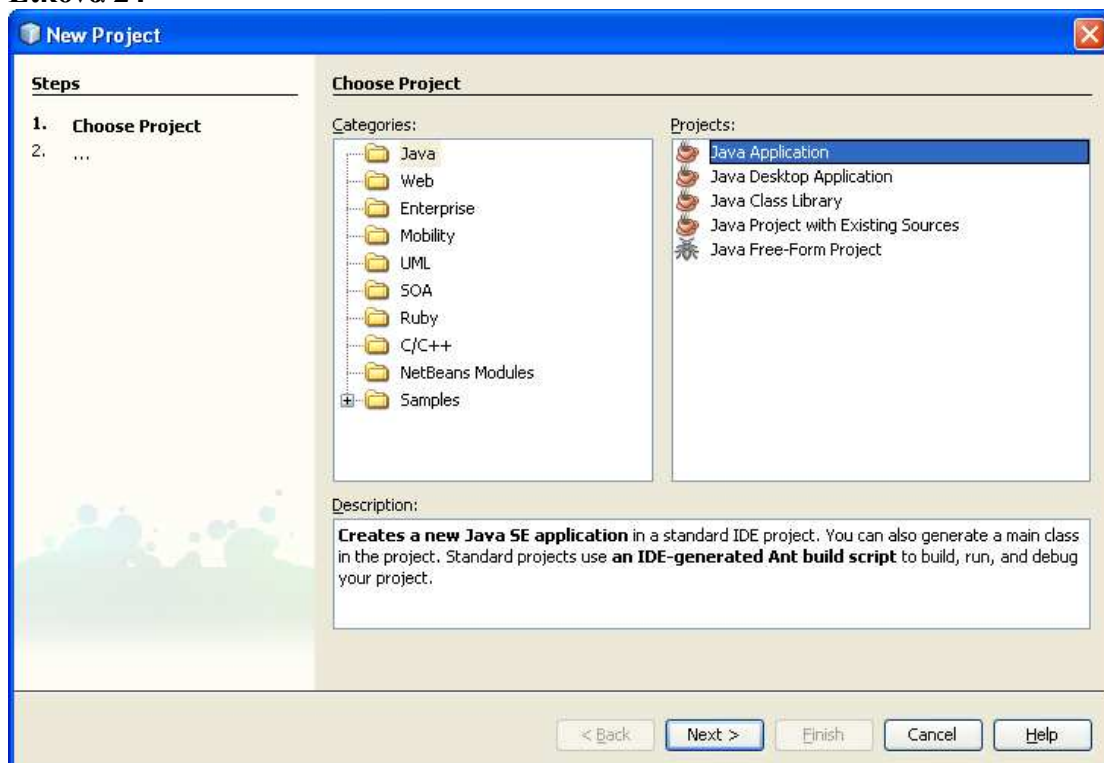
**Πρώτο βήμα:** Ανοίγουμε το πρόγραμμα Netbeans 6.1. Στη συνέχεια, πηγαίνουμε πάνω αριστερά και επιλέγουμε File -> New Project για να δημιουργήσουμε ένα νέο project. Μετά, εμφανίζεται η παρακάτω καρτέλα και επιλέγουμε από την πρώτη στήλη που λέγεται Categories την πρώτη επιλογή που είναι η Java (το βλέπουμε που είναι σε μπλε πλαίσιο).

Εικόνα 23



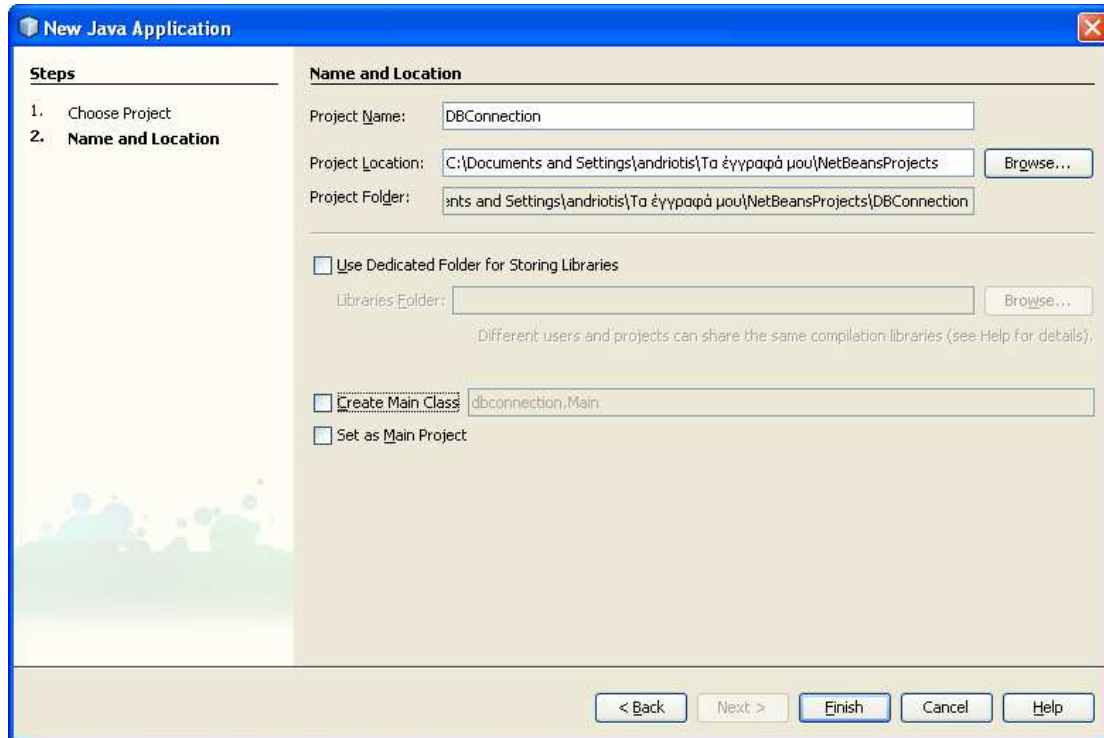
Από τη δεύτερη στήλη που είναι τα Projects επιλέγουμε την πρώτη επιλογή που είναι το Java Application (αυτό που είναι σε μπλε πλαίσιο). Στη συνέχεια, κάνουμε κλικ στο κουμπί «Next».

Εικόνα 24



**Δεύτερο βήμα:** Στη συνέχεια, εμφανίζεται η παρακάτω καρτέλα που σε αυτό το βήμα γράφουμε το όνομα του project και που θέλουμε να το αποθηκεύσουμε. Επίσης, δεν πρέπει να τσεκάρουμε τα δυο τελευταία κουτάκια: Το Create Main Class και το Set as Main Project. Μετά κάνουμε κλικ στο κουμπί «Finish».

**Εικόνα 25**



Μόλις τελειώσουμε την παραπάνω διαδικασία δημιουργείται ένα καινούργιο project με το όνομα DBConnection. Κάνουμε κλικ στο + που βρίσκεται μπροστά στο όνομα του project που έχουμε δημιουργήσει και βλέπουμε τα αποτελέσματα στην παρακάτω εικόνα.

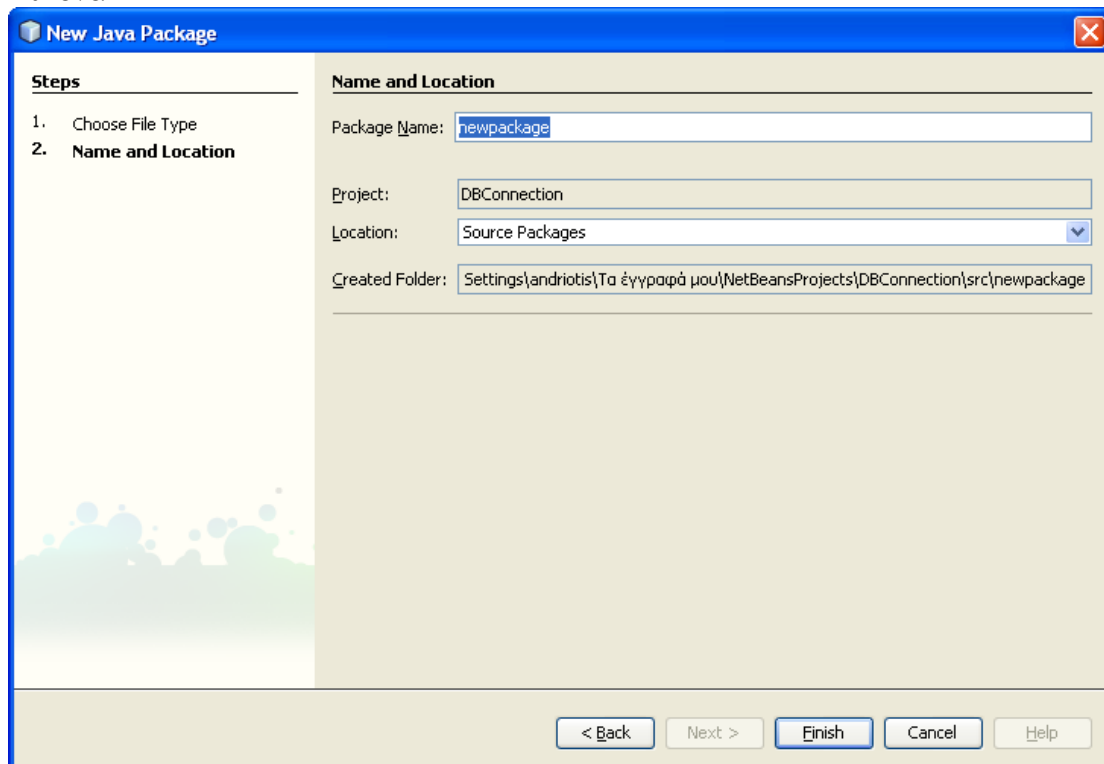
**Εικόνα 26**



Στη συνέχεια, στο Source Packages θα δημιουργήσουμε ένα καινούργιο πακέτο που μέσα θα έχουμε αποθηκεύσει ένα JFrame Form αρχείο με κατάληξη .java όπου θα γράψουμε τον πηγαίο κώδικα.

**Τρίτο βήμα:** Κάνουμε δεξί κλικ πάνω στο Source Packages. Επιλέγουμε New -> Java Package και μας εμφανίζεται η παρακάτω καρτέλα. Ονομάζουμε το πακέτο newpackage και είναι για το DBConnection Project. Επιλέγουμε στο Location, το πακέτο να τοποθετηθεί στο Source Packages και στη συνέχεια βλέπουμε που αποθηκεύτηκε το πακέτο. Τέλος, κάνουμε κλικ στο κουμπί «Finish».

**Εικόνα 27**

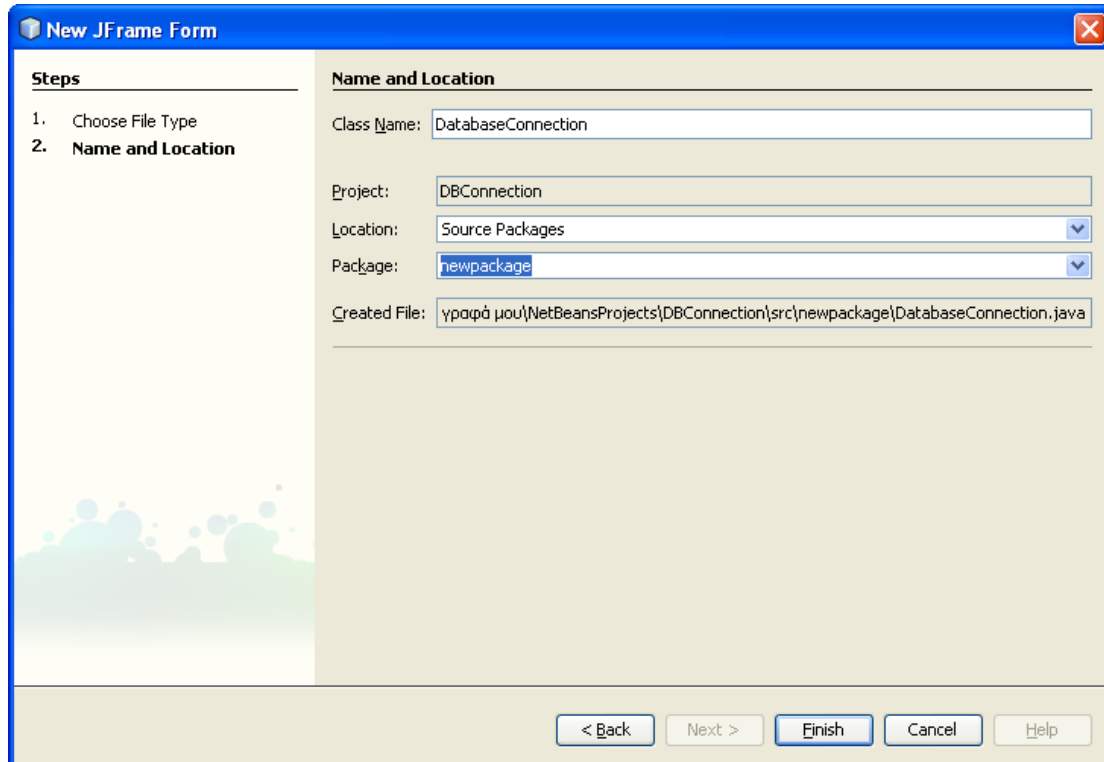


Παρακάτω βλέπουμε την δημιουργία του πακέτου newpackage και την τοποθέτηση του στο Source Packages.

**Εικόνα 28**



**Τέταρτο βήμα:** Στη συνέχεια, κάνουμε ξανά δεξί κλικ στο Source Packages και επιλέγουμε New -> JFrame Form και εμφανίζεται η παρακάτω καρτέλα. Ονομάζουμε το αρχείο DatabaseConnection που είναι του project DBConnection και έχουμε επιλέξει να βρίσκεται στο Source Packages και συγκεκριμένα στο πακέτο που δημιουργήσαμε πριν, δηλαδή, στο newpackage. Επιπλέον, βλέπουμε που έχει αποθηκευτεί το αρχείο αυτό. Τέλος, κάνουμε κλικ στο κουμπί «Finish».

**Εικόνα 29**

Παρακάτω βλέπουμε την δημιουργία του αρχείου DatabaseConnection.java και που έχει αποθηκευτεί στο newpackage.

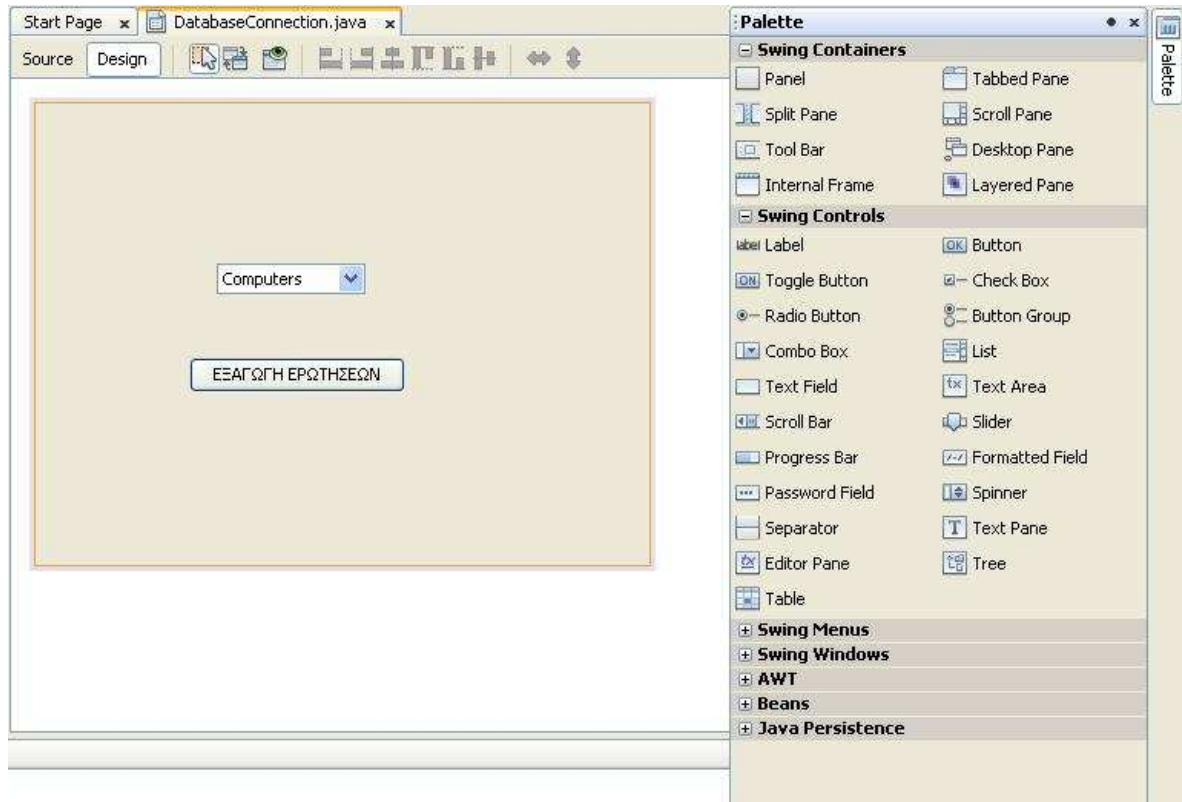
**Εικόνα 30**

Τώρα είμαστε έτοιμοι να γράψουμε τον πηγαίο κώδικα.

Καταρχήν, έχουμε ορίσει ένα JFrame Form και από την παλέτα επιλέγουμε και τοποθετούμε μέσα στη φόρμα (drag and drop) ένα Combo Box και ένα Button. Στη συνέχεια, πατάμε δεξί κλικ πάνω στο Combo Box και επιλέγουμε Properties. Μας εμφανίζεται μια καρτέλα και εκεί που γράφει model δίπλα γράφουμε Υπολογιστές, Μαθηματικά, Φυσική. (Οι επιλογές που θα έχουμε στο Combo Box). Τέλος, πατάμε Close. Για το κουμπί Button θα κάνουμε δεξί κλικ και θα επιλέξουμε ξανά Properties. Μας εμφανίζεται μια καρτέλα και εκεί που γράφει text γράφουμε δίπλα «ΕΞΑΓΩΓΗ

ΕΡΩΤΗΣΕΩΝ» (Αυτό θα εμφανίζεται πάνω στο κουμπί όπως φαίνεται παρακάτω). Τέλος, πατάμε Close.

**Εικόνα 31**



Επειδή, χρησιμοποιούμε JFrame Form, αυτό έχει ως αποτέλεσμα να μας προσφέρεται έτοιμος ο βασικός κώδικας. Έτσι, όταν πατήσουμε το κουμπί Source, μας εμφανίζει τον κώδικα που έχει δημιουργηθεί από την τοποθέτηση των δύο αντικειμένων (Combo Box & Button). Εμείς αυτό που έχουμε να κάνουμε είναι να προσθέσουμε τις εντολές που χρειαζόμαστε για να φτιάξουμε το project μας.

Στην αρχή, θα κάνουμε **import** τις παρακάτω βιβλιοθήκες που θα χρειαστούμε

```
import java.sql.*; // Πακέτο που αφορά τη Βάση Δεδομένων
import java.io.*; // Πακέτο που αφορά Input/Output δεδομένα
import java.util.Random; // Πακέτο που αφορά επιλογή τυχαίου αριθμού
```

Οι δηλώσεις **import** τοποθετούνται στην αρχή του αρχείου μας, πριν από οποιουσδήποτε ορισμούς κλάσεων αλλά μετά από τη δήλωση του πακέτου όπως βλέπουμε στην παρακάτω εικόνα.

```

8   package newpackage;
9
10  import java.sql.*;
11  import java.io.*;
12  import java.util.Random;
13

```

Στη συνέχεια, στην 22<sup>η</sup> σειρά έχουμε προσθέσει την παρακάτω γραμμή κώδικα  
`super("Database Connection");` // στη γραμμή τίτλου θα εμφανιστεί ο τίτλος  
`// Database Connection`

```

18  public class DatabaseConnection extends javax.swing.JFrame {
19
20  /** Creates new form DatabaseConnection */
21  public DatabaseConnection() {
22      super("Database Connection");
23      initComponents();
24  }

```

Συνεχίζουμε κάνοντας δεξί κλικ πάνω στο Combo Box και επιλέγουμε  
 Events -> Action -> actionPerformed

Μόλις κάνουμε αυτή την επιλογή μας παραθέτει στον κώδικα στο σημείο που αφορά το Combo Box και γράφουμε το παρακάτω κώδικα.

```

153 private void jComboBox1ActionPerformed(java.awt.event.ActionEvent evt) {
154
155     String t=(String)jComboBox1.getSelectedItem();
156
157
158     if(t.equals("Υπολογιστές"))
159     {
160         jComboBox1.setSelectedIndex(0);
161     }
162     else if(t.equals("Μαθηματικά"))
163     {
164         jComboBox1.setSelectedIndex(1);
165     }
166     else if(t.equals("Φυσική"))
167     {
168         jComboBox1.setSelectedIndex(2);
169     }
170
171
172 }

```

Η μέθοδος `getSelectedItem()` επιστρέφει το κείμενο του επιλεγμένου στοιχείου της λίστας. Η μέθοδος `equals()` συγκρίνει αν δυο αντικείμενα String είναι ίσα. Άρα

παραπάνω έχουμε την σύγκριση και την επιλογή ενός από τις τρεις επιλογές του προσφέρει το combo box. Επιλέγουμε ή Υπολογιστές ή Μαθηματικά ή Φυσική.

Συνεχίζουμε κάνοντας δεξί κλικ πάνω στο Button που το έχουμε ονομάσει «ΕΞΑΓΩΓΗ ΕΡΩΤΗΣΕΩΝ» και επιλέγουμε

Events -> Action -> actionPerformed

Μόλις κάνουμε αυτή την επιλογή, μας παραθέτει στον κώδικα στο σημείο που αφορά το Button και γράφουμε το παρακάτω κώδικα.

```

81 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
82     Connection myC=null;
83     Statement mySt=null;
84     ResultSet myRs=null;
85
86
87     try{
88         Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
89         myC=DriverManager.getConnection("jdbc:odbc:databaseCell");
90         mySt=myC.createStatement();
91
92         Random rand = new Random();
93
94         int[] ids = new int[10];
95
96         for (int i = 0; i < 10; i++)
97         {
98             ids[i] = rand.nextInt(100) + 1;
99
100            if (i!=0)
101            {
102                if (ids[i - 1] == ids[i])
103                {
104                    ids[i] = rand.nextInt(100) + 1;
105                }
106            }
107        }
108
109        myRs = mySt.executeQuery("SELECT TOP 10 AQuestion, Question, Answer1, Answer2, " +
110            "Answer3, Answer4, CorrectAnswer FROM Table1 WHERE AQuestion IN (" + ids[0] + "," + ids[1]
111            + "," + ids[2] + "," + ids[3] + "," + ids[4] + "," + ids[5]
112            + "," + ids[6] + "," + ids[7] + "," + ids[8] + "," + ids[9] + ");");
113
114
115
116
117        PrintWriter x = new PrintWriter(new FileOutputStream("Game.txt"));
118        PrintWriter x2 = new PrintWriter(new FileOutputStream("Answers.txt"));
119
120
121        while (myRs.next())
122        {
123
124            String Answer1 = myRs.getString("Answer1");

```



```

125         String Answer2 = myRs.getString("Answer2");
126         String Answer3 = myRs.getString("Answer3");
127         String Answer4 = myRs.getString("Answer4");
128         String Question = myRs.getString("Question");
129         String AQuestion = myRs.getString("AQuestion");
130         String CorrectAnswer = myRs.getString("CorrectAnswer");
131
132
133
134         x.println(AQuestion + " " + Question);
135         x.println("a."+Answer1);
136         x.println("b."+Answer2);
137         x.println("c."+Answer3);
138         x.println("d."+Answer4);
139         x.println("**");
140
141         x2.print(CorrectAnswer);
142     }
143     x.close();
144     x2.close();
145
146 } catch (Exception e) {
147     System.out.println("Error" + e.toString());
148 }
149
150
151 }

```

Μερικά σχόλια για τον παραπάνω κώδικα:

**Γραμμή 82:** Δημιουργία αντικειμένου Connection με το όνομα myC. Χρησιμοποιείται για την σύνδεση με τη βάση και αρχικοποιούμε την μεταβλητή ίσον με null.

**Γραμμή 83:** Δημιουργία αντικειμένου Statement με το όνομα mySt. Χρησιμοποιείται για να αποστέλλει την εντολή SQL στη βάση δεδομένων και να την εκτελεί, επιστρέφοντας τα αποτελέσματα με συγκεκριμένη μορφή που διευκολύνει την επεξεργασία τους και αρχικοποιούμε την μεταβλητή ίσον με null.

**Γραμμή 84:** Δημιουργία αντικειμένου ResultSet με το όνομα myRs, η οποία περιέχει το αποτέλεσμα της εντολής SELECT. Αρχικοποιούμε την μεταβλητή ίσον με null.

**Γραμμή 88:** Όλες οι εφαρμογές που χρησιμοποιούν μία προέλευση δεδομένων ODBC χρησιμοποιούν το sun.jdbc.odbc.JdbcOdbcDriver, το πρόγραμμα οδήγησης για την γέφυρα JDBC-ODBC που περιλαμβάνεται στην Java. Άρα, η εντολή αυτή είναι η φόρτωση του οδηγού (driver) που θα επικοινωνεί με τη βάση.

**Γραμμή 89:** Σύνδεση με τη βάση δεδομένων. Για την υλοποίηση της σύνδεσης χρησιμοποιούμε την μέθοδο getConnection(url) της κλάσης DriverManager. Αυτή η

μέθοδος επιστρέφει μια αναφορά προς ένα αντικείμενο Connection το οποίο αντιπροσωπεύει μια ενεργή σύνδεση δεδομένων. Επειδή χρησιμοποιούμε για οδηγό JDBC-ODBC τότε το url θα αρχίσει με “jdbc:odbc:”. Το υπόλοιπο τμήμα της διεύθυνσης είναι το όνομα της βάσης. Δηλαδή στην δική μας περίπτωση είναι databaseCell. Άρα το url θα είναι “jdbc:odbc:databaseCell”.

**Γραμμή 90:** Μια πρόταση SQL αντιπροσωπεύεται από ένα αντικείμενο Statement. Επειδή, η Statement είναι μια διασύνδεση (interface), δεν είναι δυνατή η άμεση δημιουργία αντικειμένων από αυτήν. Ωστόσο, ένα αντικείμενο το οποίο υλοποιεί αυτή τη διασύνδεση επιστρέφεται από τη μέθοδο createStatement() ενός αντικειμένου Connection.

**Γραμμή 92:** Δημιουργία αντικειμένου Random. Η μεταβλητή rand είναι δείκτης αναφοράς στο αντίστοιχο αντικείμενο που δημιουργήθηκε με τη new.

**Γραμμή 94:** Δημιουργία ακέραιου πίνακα 10 θέσεων με το όνομα ids.

**Γραμμές 96-107:** Βρόγχος επανάληψης που θα εκτελεστεί 10 φορές. Σε κάθε μία θέση του πίνακα ids θα αποθηκεύεται ένας τυχαίος αριθμός από το 1 μέχρι το 100.

**Γραμμές 109-112:** Έχοντας ένα αντικείμενο Statement μπορούμε να το χρησιμοποιήσουμε για να εκτελέσουμε ένα ερώτημα SQL καλώντας τη μέθοδο executeQuery(String) αυτού του αντικειμένου. Το όρισμα String θα πρέπει να είναι ένα ερώτημα SQL διατυπωμένο με τη σύνταξη αυτής της γλώσσας. Η μέθοδος executeQuery() επιστρέφει ένα αντικείμενο ResultSet (σύνολο αποτελεσμάτων) το οποίο περιέχει όλες τις εγγραφές που ανακτήθηκαν από την προέλευση δεδομένων. Όταν επιστρέφεται ένα αντικείμενο ResultSet από την executeQuery(), αυτό τοποθετείται στην πρώτη εγγραφή που έχει ανακτηθεί. Άρα, στις γραμμές 109-112 επιλέγουμε τις 10 πρώτες εγγραφές από τις στήλες AQuestion, Question, Answer1, Answer2, Answer3, Answer4, CorrectAnswer από τον πίνακα Table1 όπου το AQuestion να ισούται με τον αριθμό που είναι αποθηκευμένο σε κάθε θέση του πίνακα ids.

**Γραμμή 117:** Δημιουργούμε ένα txt αρχείο με το όνομα Game.txt. Σε αυτό το αρχείο θα αποθηκεύσουμε τις 10 ερωτήσεις, τον αριθμό της ερώτησης και τις 4 πιθανές απαντήσεις. Αποθηκεύουμε τα περιεχόμενα του αρχείου με την εξής δομή: Στην αρχή, εμφανίζεται ο αριθμός της ερώτησης και ερώτηση. Στην επόμενη γραμμή εμφανίζεται η πρώτη απάντηση, στην επόμενη γραμμή η δεύτερη απάντηση, στην επόμενη γραμμή η τρίτη απάντηση και στην επόμενη γραμμή η τέταρτη απάντηση.

Μετά, έχουμε ορίσει να εμφανίζεται στη επόμενη γραμμή ένας αστερίσκος (\*). Αυτό το κάναμε για να ξεχωρίζουν οι ερωτήσεις με τις πιθανές απαντήσεις τους.

**Γραμμή 118:** Δημιουργούμε ένα txt αρχείο με το όνομα Answers.txt. Σε αυτό το αρχείο θα αποθηκεύσουμε τις σωστές απαντήσεις των ερωτήσεων που έχουν επιλεγεί. Αποθηκεύουμε τις απαντήσεις τη μια δίπλα στην άλλη, σαν ένα string.

**Γραμμές 121-142:** Η μεταβλητή myRs περιέχει το αποτέλεσμα της εντολής SELECT, το οποίο μπορούμε να επεξεργαστούμε. Η επεξεργασία όμως γίνεται γραμμή προς γραμμή και για να μπορούμε να μεταφερόμαστε στην επόμενη γραμμή θα χρησιμοποιούμε την μέθοδο next() του αντικειμένου ResultSet. Με την εκτέλεση της μεθόδου next() μετακινούμε έναν εικονικό δρομέα (cursor) στα αποτελέσματα του πίνακα και κάθε φορά επεξεργαζόμαστε τη γραμμή στην οποία βρίσκεται ο δρομέας (που καλείται στη τρέχουσα γραμμή). Η αρχική τρέχουσα γραμμή είναι πάνω από την πρώτη γραμμή των αποτελεσμάτων που επιστρέφονται με το ResultSet (που με την σειρά τους εξαρτώνται από την εντολή SQL που εκτελούμε με την executeQuery()). Αυτή η μέθοδος επιστρέφει μια τιμή false όταν επιχειρήσουμε να προχωρήσουμε πέρα από το τέλος ενός συνόλου αποτελεσμάτων. Επίσης, η getString() του ResultSet επιστρέφει το String που είναι αποθηκευμένο στο προσδιοριζόμενο όνομα πεδίου. Άρα, στις γραμμές 121-142 έχουμε έναν βρόγχο επανάληψης όπου παίρνουμε τα αποτελέσματα από την βάση και αποθηκεύουμε τα αποτελέσματα AQuestion, Question, Answer1, Answer2, Answer3, Answer4 στο αρχείο Game.txt ενώ στο αρχείο Answers.txt αποθηκεύουμε τα αποτελέσματα από την στήλη CorrectAnswer.

**Γραμμή 143:** Η μέθοδος close() κλείνει το stream εξόδου αρχείου Game.txt.

**Γραμμή 144:** Η μέθοδος close() κλείνει το stream εξόδου αρχείου Answers.txt.

**Γραμμή 146:** Στον κώδικας μας έχουμε χρησιμοποιήσει try-catch η οποία μας δίνει την δυνατότητα όταν δεν εκτελεστεί σωστά ο κώδικας να παγιδεύσει την εξαίρεση και να μας εμφανίσει που βρίσκεται το λάθος. Η μέθοδος toString() μας εμφανίζει ένα String με μια μικρή περιγραφή της εξαίρεσης.

Έχοντας ολοκληρώσει τον κώδικα μας κάνουμε δεξί κλικ στο DatabaseConnection.java και επιλέγουμε Compile. Παρατηρούμε όπως βλέπουμε παρακάτω ότι τα αποτελέσματα είναι επιτυχή.

**Εικόνα 32**

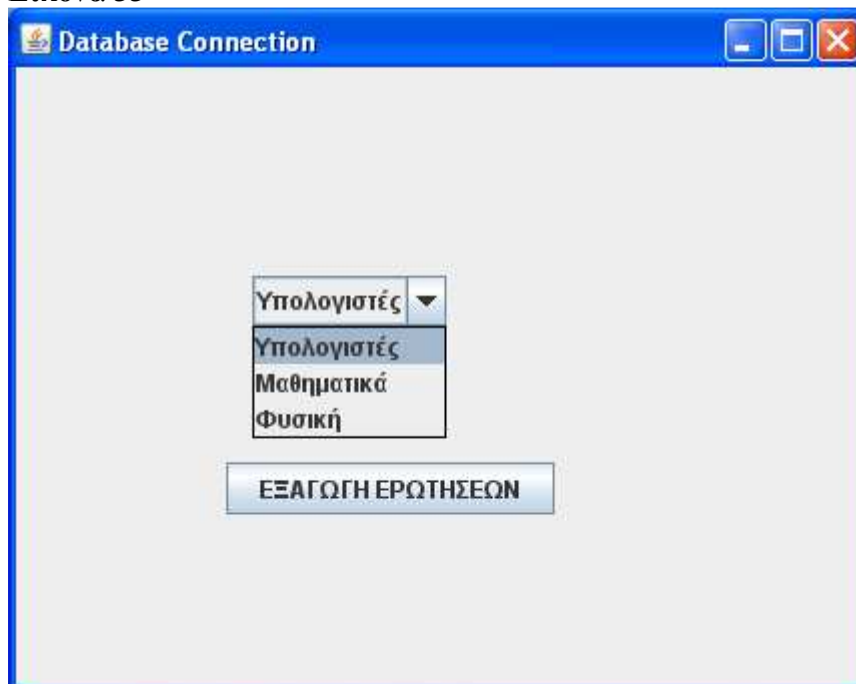
```

Output - DBConnection (compile-single)
init:
deps-jar:
Compiling 1 source file to C:\Documents and Settings\andriotis\Τα έγγραφά μου\NetBeansProjects\DBConnection\build\classes
compile-single:
BUILD SUCCESSFUL (total time: 2 seconds)

```

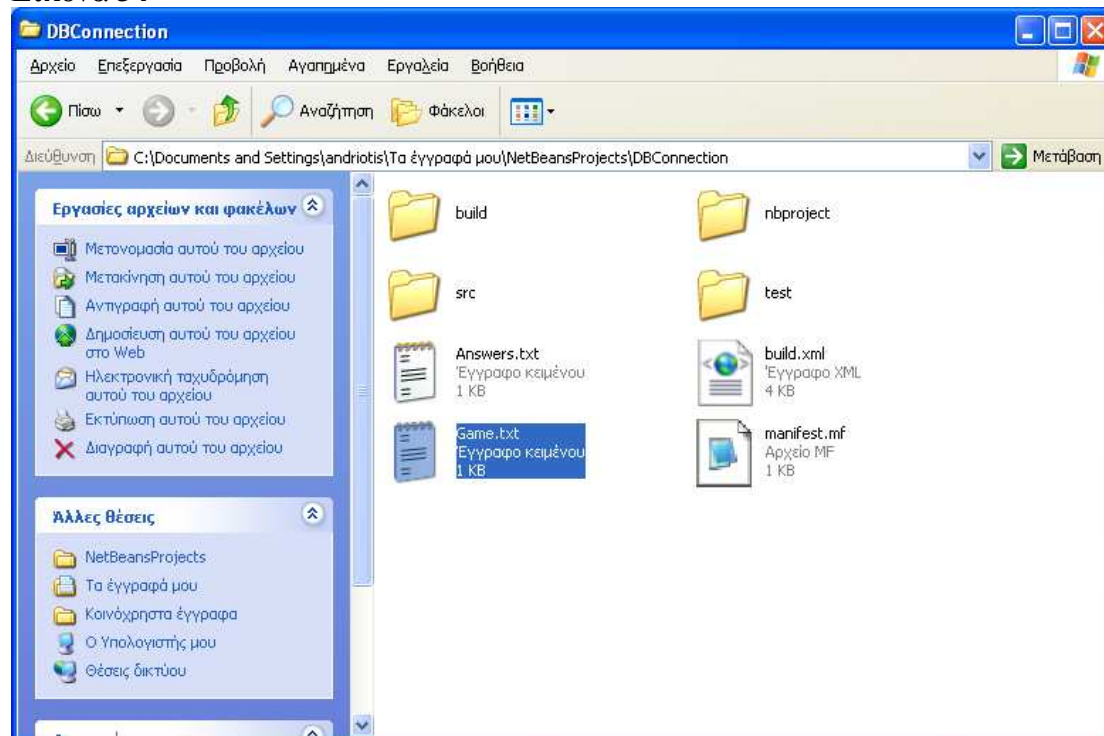
Τέλος, κάνουμε ξανά δεξί κλικ στο DatabaseConnection.java και επιλέγουμε Run File και βλέπουμε ότι λειτουργεί με επιτυχία ο κώδικας μας.

Παρακάτω βλέπουμε το παράθυρο που φτιάξαμε όπου επιλέγουμε ένα από τα τρία μαθήματα και στην συνέχεια πατάμε το κουμπί ΕΞΑΓΩΓΗ ΕΡΩΤΗΣΕΩΝ.

**Εικόνα 33**

Στο φάκελο NetbeansProjects που έχουμε αποθηκεύσει τον αρχείο DBConnection βλέπουμε την δημιουργία των δυο txt αρχείων Game.txt και Answers.txt και τα δεδομένα που έχουν αποθηκευτεί με επιτυχία.

**Εικόνα 34**



## Βιβλιογραφία για το 1<sup>ο</sup> Μέρος

- [1]. Βιβλίο: Πλήρες Εγχειρίδιο της JAVA 6  
Συγγραφείς: Rogers Cadenhead, Laura Lemay  
Πέμπτη Έκδοση – Εκδόσεις: Μ. Γκιούρδας
- [2].Βιβλίο: Βάσεις Δεδομένων Access  
Συγγραφέας: Χατζάκης Ηλίας  
Ηράκλειο 2004, Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης
- [3]. <http://aetos.it.teithe.gr/~constant/javaI/javadocs/Course-Java-Notes.pdf>
- [4]. <http://download.oracle.com/javase/1.4.2/docs/api/java/util/Random.html>
- [5]. <http://leepoint.net/notes-java/algorithms/random/random-api.html>

## Μέρος 2<sup>ο</sup>

Όπως προαναφέραμε, στο Μέρος 2<sup>ο</sup> θα ασχοληθούμε με τη δημιουργία νέου project χρησιμοποιώντας MIDlets. Πιο συγκεκριμένα, μόλις φτιάξουμε την εφαρμογή στη συνέχεια, δημιουργούμε το JAR αρχείο της εφαρμογής. Μετά, το JAR αρχείο το μεταφέρουμε στο κινητό μας τηλέφωνο μέσω της τεχνολογίας Bluetooth. Η διαδικασία που θα ακολουθήσουμε στο δεύτερο μέρος του project μας είναι η παρακάτω:

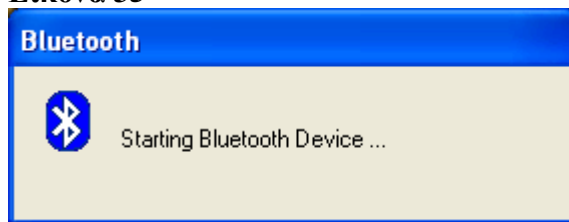
- Καταρχήν, θα εγκαταστήσουμε Mini Bluetooth v2.0 + EDR, USB-BT21 της ASUS, επειδή ο υπολογιστής μας δεν έχει ενσωματωμένη την τεχνολογία Bluetooth.
- Επίσης, βλέπουμε τα χαρακτηριστικά του κινητού τηλεφώνου που θα χρησιμοποιήσουμε στο project μας έτσι ώστε να δούμε ποια πλατφόρμα θα κάνουμε download.
- Στη συνέχεια, θα πρέπει να κάνουμε download τις πλατφόρμες δύο διαφορετικών κινητών τηλεφώνων. Η μία πλατφόρμα είναι της εταιρείας NOKIA και η άλλη της εταιρείας SONY ERICSSON.
- Στη συνέχεια κάνουμε εγκατάσταση τις δύο πλατφόρμες πρώτα στον υπολογιστή μας και στη συνέχεια στο πρόγραμμα μας Netbeans 6.1.
- Επίσης, από το forum της NOKIA θα χρησιμοποιήσουμε ένα κομμάτι κώδικα (open-source). Χρησιμοποιώντας τον κώδικα αυτό θα φτιάξουμε τη δική μας εφαρμογή.
- Επίσης, θα χρησιμοποιήσουμε το πρόγραμμα Netbeans 6.1 και θα χρησιμοποιήσουμε MIDlets για να φτιάξουμε την εφαρμογή μας. Στη συνέχεια, θα αναλύσουμε τον κώδικα γραμμή προς γραμμή.
- Τέλος, εφαρμόζουμε τον κώδικα χρησιμοποιώντας τον εξομοιωτή (emulator) της Sun Java Wireless Toolkit και εμφανίζουμε τα αποτελέσματα.
- Ολοκληρώνουμε το κεφάλαιο αυτό συνδέοντας το κινητό τηλέφωνο NOKIA 5300 με τον υπολογιστή μέσω Bluetooth και εμφανίζοντας τα αποτελέσματα χρησιμοποιώντας το εξομοιωτή (emulator) της NOKIA.

### 3.2.1 Εγκατάσταση Bluetooth 2.0

Ακολουθούμε τα βήματα της εγκατάστασης του Bluetooth 2.0. Μόλις γίνει η εγκατάσταση στον υπολογιστή, στη συνέχεια ενεργοποιούμε τη συσκευή Bluetooth για την αναζήτηση συσκευών κινητών τηλεφώνων και την σύνδεση τους με τον υπολογιστή. Ακολουθούμε τα παρακάτω βήματα.

Καταρχήν, από το My Bluetooth Places επιλέγουμε Add a Bluetooth Device και εμφανίζεται το παρακάτω παράθυρο.

**Εικόνα 35**



Στη συνέχεια, εμφανίζεται το παρακάτω παράθυρο που θα ξεκινήσει για την αναζήτηση κινητού τηλεφώνου. Πατάμε το κουμπί Επόμενο.

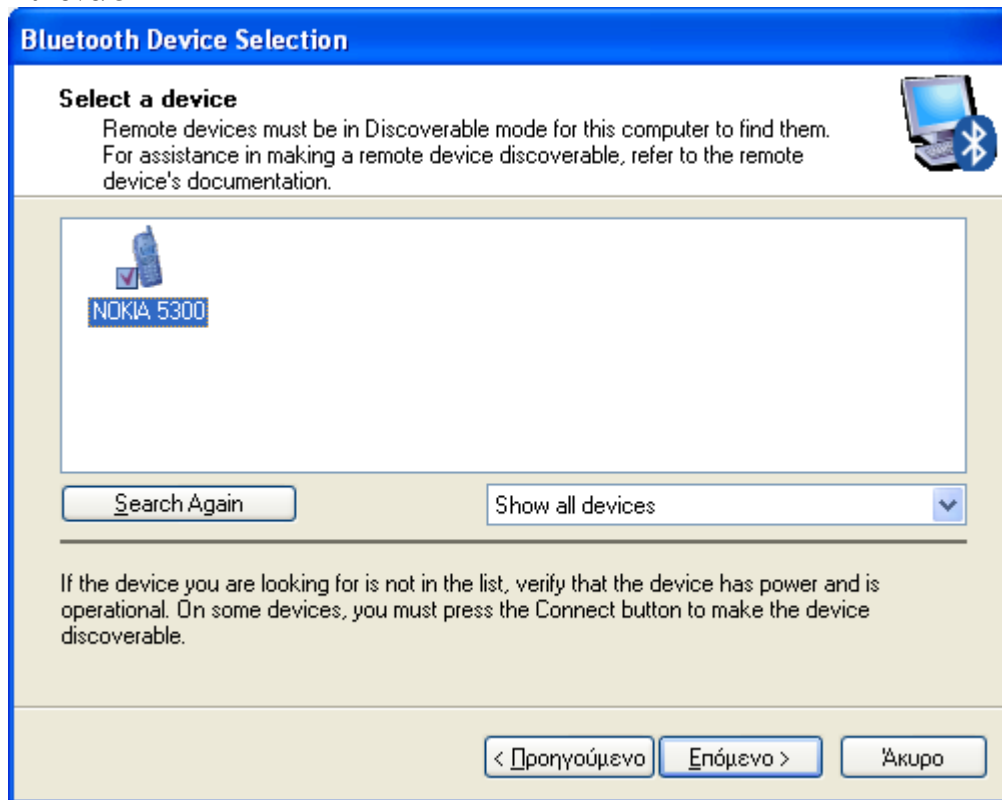
**Εικόνα 36**



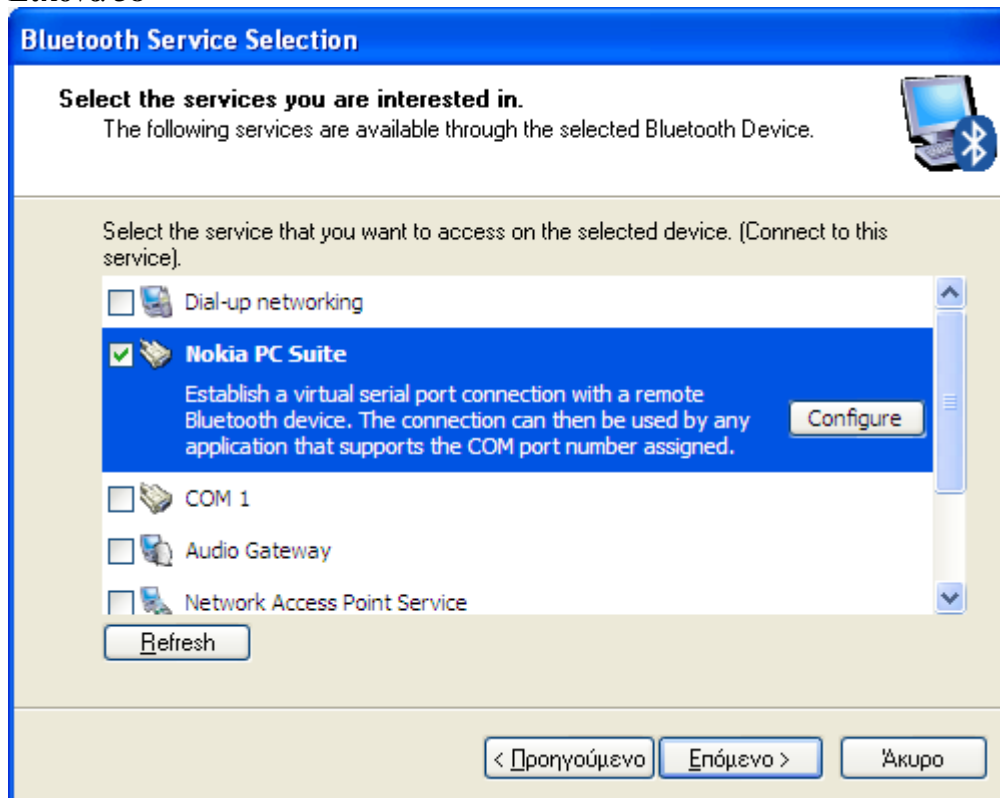


Μετά, εμφανίζεται το παρακάτω παράθυρο που γίνεται αναζήτηση της συσκευής κινητού τηλεφώνου. Στη συνέχεια, επιλέγουμε την συσκευή που θα χρησιμοποιήσουμε στο project μας. Πατάμε το κουμπί Επόμενο για να συνεχίσουμε.

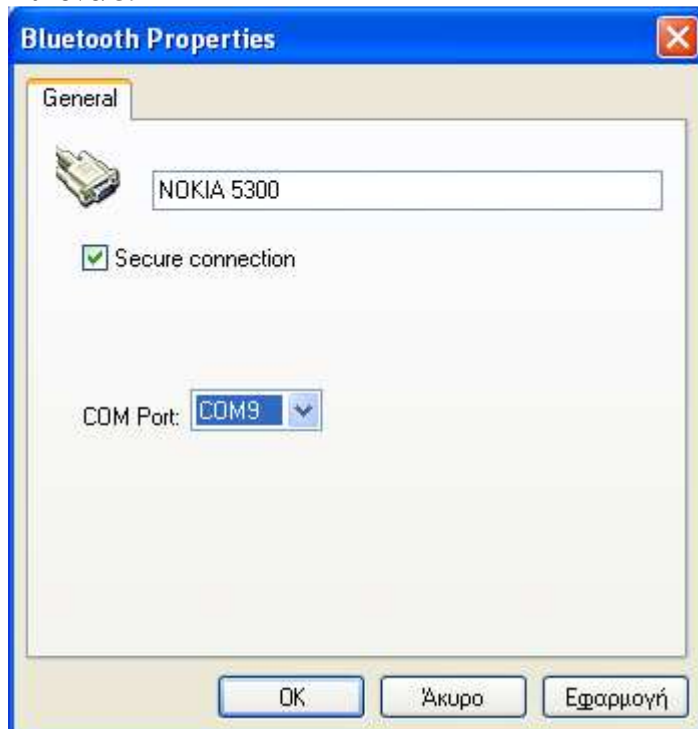
**Εικόνα 37**



Εμφανίζεται το παρακάτω παράθυρο που μας δίνει την δυνατότητα να επιλέξουμε υπηρεσίες. Εμείς θα επιλέξουμε το Nokia PC Suite. Με την επιλογή αυτή θα εγκατασταθεί μια εικονική σειριακή θύρα, η οποία θα συνδεθεί με μια κινητή Bluetooth συσκευή. Στη συνέχεια, κάνουμε κλικ στο Configure.

**Εικόνα 38**

Γράφουμε με ποιο όνομα θέλουμε να αναγνωρίζουμε την συσκευή. Επίσης, τσεκάρουμε την ασφαλή σύνδεση. Κάνουμε κλικ στο OK.

**Εικόνα 39**

Επιστρέφουμε πίσω στο παράθυρο της Εικόνας 28 και πατάμε το κουμπί Επόμενο.  
Τσεκάρουμε το κουτάκι κάτω και η σύνδεση μας έχει ολοκληρωθεί με επιτυχία.  
Πατάμε το κουμπί Τέλος.

**Εικόνα 40**



Έχοντας ολοκληρώσει με επιτυχία την εγκατάσταση μας και εξασφαλίζοντας ότι ο υπολογιστής μας αναγνωρίζει την συσκευή κινητού τηλεφώνου μας, μπορούμε να συνεχίσουμε στο επόμενο βήμα.

### 3.2.2 Χαρακτηριστικά κινητού τηλεφώνου

Τα χαρακτηριστικά του κινητού τηλεφώνου μπορούμε να τα διαβάσουμε στο παρακάτω site:

[http://www.forum.nokia.com/Devices/Device\\_specifications/5300\\_XpressMusic/](http://www.forum.nokia.com/Devices/Device_specifications/5300_XpressMusic/)

Η συγκεκριμένη συσκευή χρησιμοποιεί την πλατφόρμα Series 40 3<sup>rd</sup> Edition, Feature Pack 2. Επίσης, υποστηρίζει την Java τεχνολογία και πιο συγκεκριμένα τα παρακάτω JSRs που μας είναι απαραίτητα για το project μας.

JSR 139 Connected, Limited Device Configuration (CLDC) 1.1

JSR 118 MIDP 2.0

JSR 185 Java™ Technology for Wireless Industry

JSR 75 FileConnection and PIM API

JSR 82 Bluetooth API

JSR 135 Mobile Media API

Για περισσότερες λεπτομέρειες μπορούμε να ανατρέξουμε στην παραπάνω διεύθυνση.

### 3.2.3 Download πλατφόρμες NOKIA & SONY

#### ERICSSON

Το επόμενο βήμα είναι το download των δυο πλατφόρμων. Εμείς θα χρησιμοποιήσουμε την πλατφόρμα της NOKIA για να επαληθεύσουμε την εφαρμογή μας και να δούμε ότι δουλεύει στο κινητό τηλέφωνο χωρίς προβλήματα.

Η πλατφόρμα της NOKIA ονομάζεται Series 40 3<sup>rd</sup> Edition, Feature Pack 2.

Η πλατφόρμα της SONY ERICSSON λέγεται Sony Ericsson SDK for the Java ME platform 2.5.0.4.

Μπορούμε να κάνουμε download την πλατφόρμα της NOKIA από το παρακάτω link

[http://www.forum.nokia.com/info/sw.nokia.com/id/cc48f9a1-f5cf-447b-bdba-c4d41b3d05ce/Series\\_40\\_Platform\\_SDKs.html](http://www.forum.nokia.com/info/sw.nokia.com/id/cc48f9a1-f5cf-447b-bdba-c4d41b3d05ce/Series_40_Platform_SDKs.html)

Για την πλατφόρμα της SONY ERICSSON κάνουμε download από το παρακάτω link

<http://developer.sonyericsson.com/wportal/devworld/downloads/download/dw-102090-semcjavameclcdsdk2504?cc=gb&lc=en>

## 3.2.4 Εγκατάσταση των πλατφόρμων NOKIA & SONY ERICSSON στον υπολογιστή & στο πρόγραμμα

### 3.2.4.1α Εγκατάσταση της πλατφόρμας NOKIA στον υπολογιστή

Για να κάνουμε την εγκατάσταση της πλατφόρμας, εγκαταστήσαμε πρώτα το Java SE Runtime Environment (JRE) 5.0. Επίσης, για να χρησιμοποιήσουμε το Netbeans 6.1 με το SDK θα πρέπει να έχουμε εγκαταστήσει και το Java SE Development Kit (JDK).

Έχοντας ολοκληρώσει την παραπάνω διαδικασία μπορούμε να εγκαταστήσουμε το Series 40 SDK, 3<sup>rd</sup> Edition Feature Pack 2. Η διαδικασία έχει ως εξής:

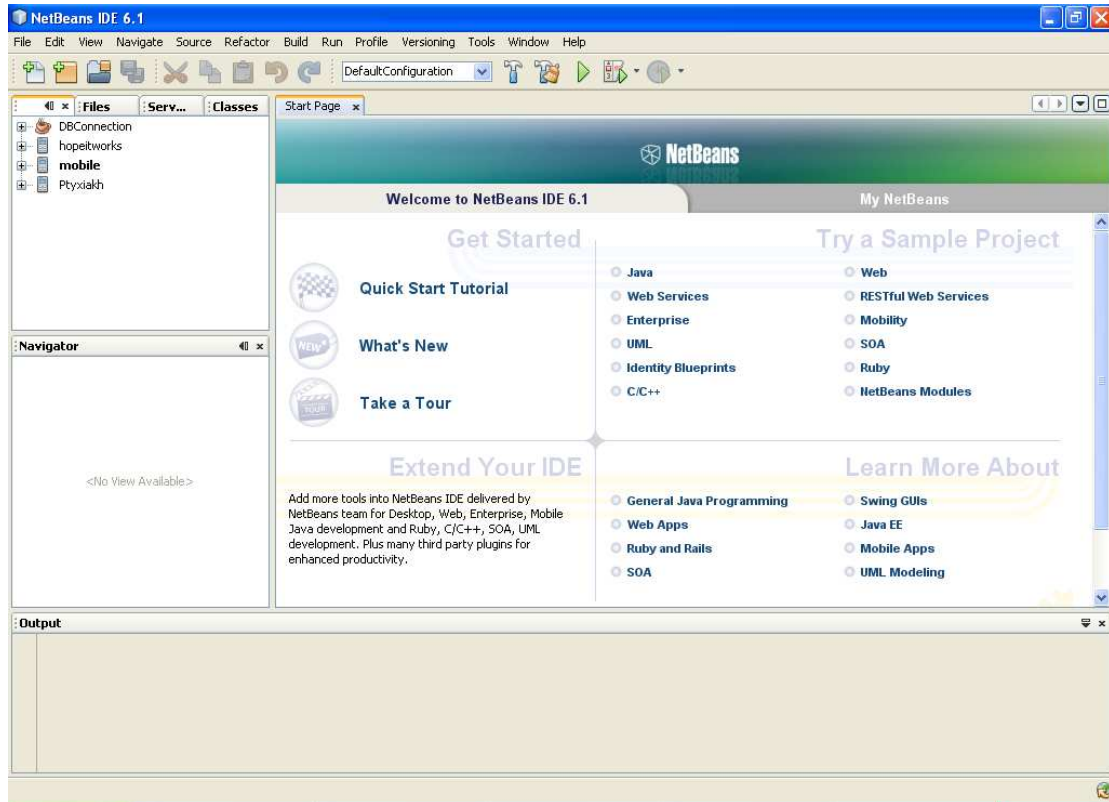
- Έχοντας κάνει download το zip αρχείο, στη συνέχεια κάνουμε δεξί κλικ στο zip αρχείο και επιλέγουμε Extract Here για να μπορέσουμε να χρησιμοποιήσουμε τα αρχεία.
- Στη συνέχεια, κάνουμε διπλό κλικ στο αρχείο S40\_SDK\_3rd\_Edition\_FP2\_installer\_em.exe. Η διαδικασία εγκατάσταση ξεκινάει.
- Κατά τη διάρκεια της εγκατάστασης, επιλέγουμε που θα αποθηκεύσουμε το αρχείο. Εμείς, επιλέγουμε να το αποθηκεύσουμε στην παρακάτω διεύθυνση C:\Nokia\Devices\S40\_SDK\_3rd\_Edition\_Feature\_Pack\_2.

Η εγκατάσταση έγινε με επιτυχία. Για να φορτώσουμε το SDK μέσα σε ένα περιβάλλον IDE και πιο συγκεκριμένα στο Netbeans 6.1 θα ακολουθήσουμε την παρακάτω διαδικασία.

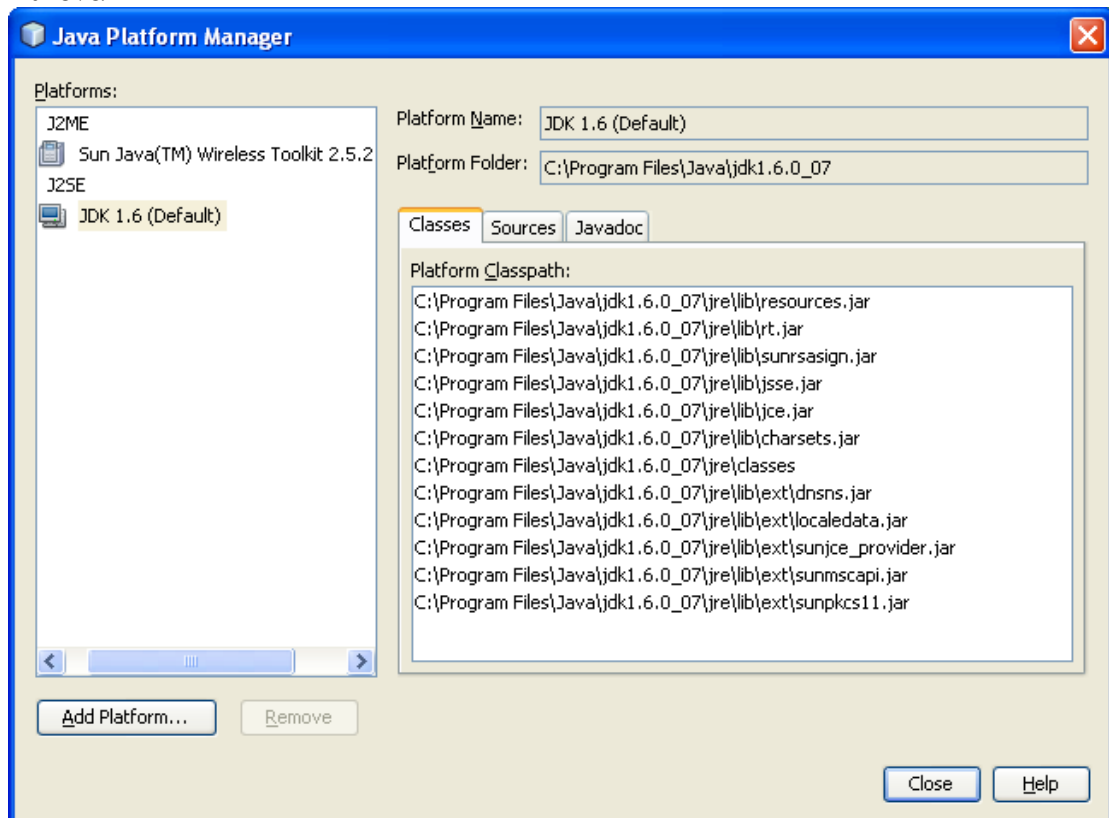
### 3.2.4.1β Εγκατάσταση της πλατφόρμας NOKIA στο πρόγραμμα Netbeans

Στη συνέχεια, κάνουμε την εγκατάσταση της πλατφόρμας NOKIA στο πρόγραμμα Netbeans 6.1.

Ανοίγουμε το πρόγραμμα Netbeans όπως βλέπουμε στην παρακάτω εικόνα, πηγαίνουμε Tools και επιλέγουμε Java Platforms.

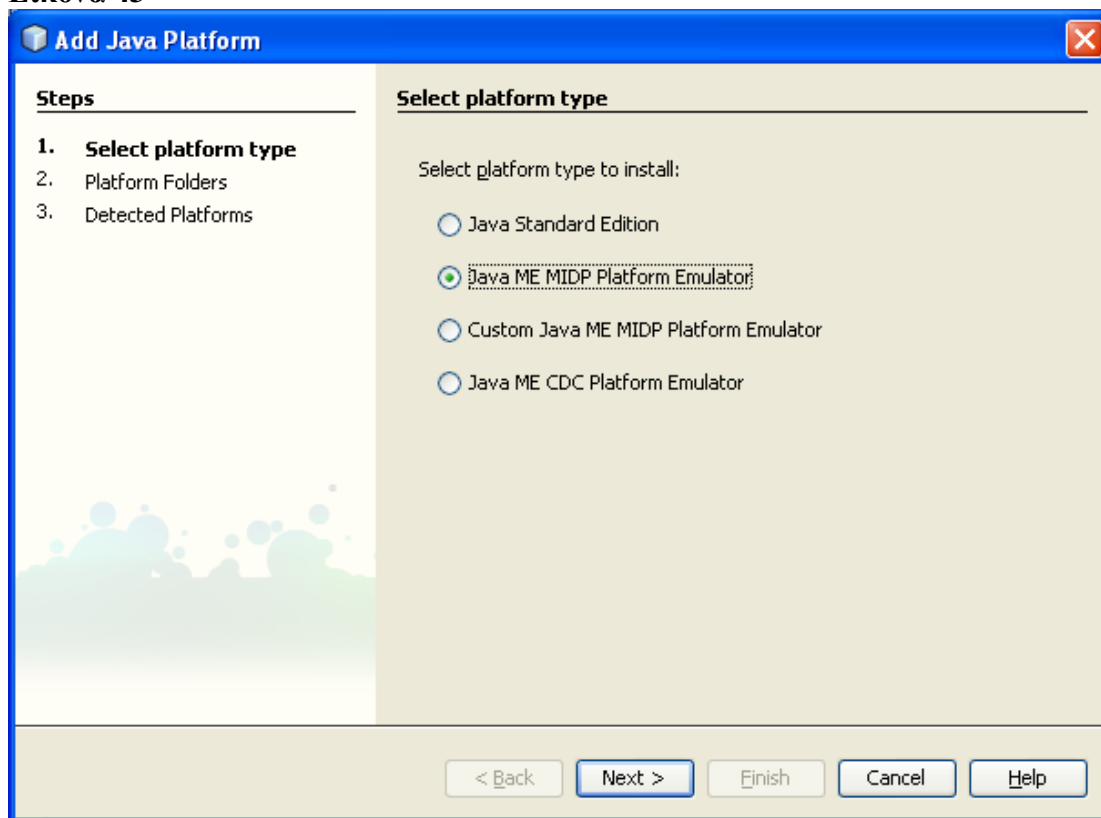
**Εικόνα 41**

Μας εμφανίζεται το παρακάτω παράθυρο και πατάμε το κουμπί Add Platform.

**Εικόνα 42**

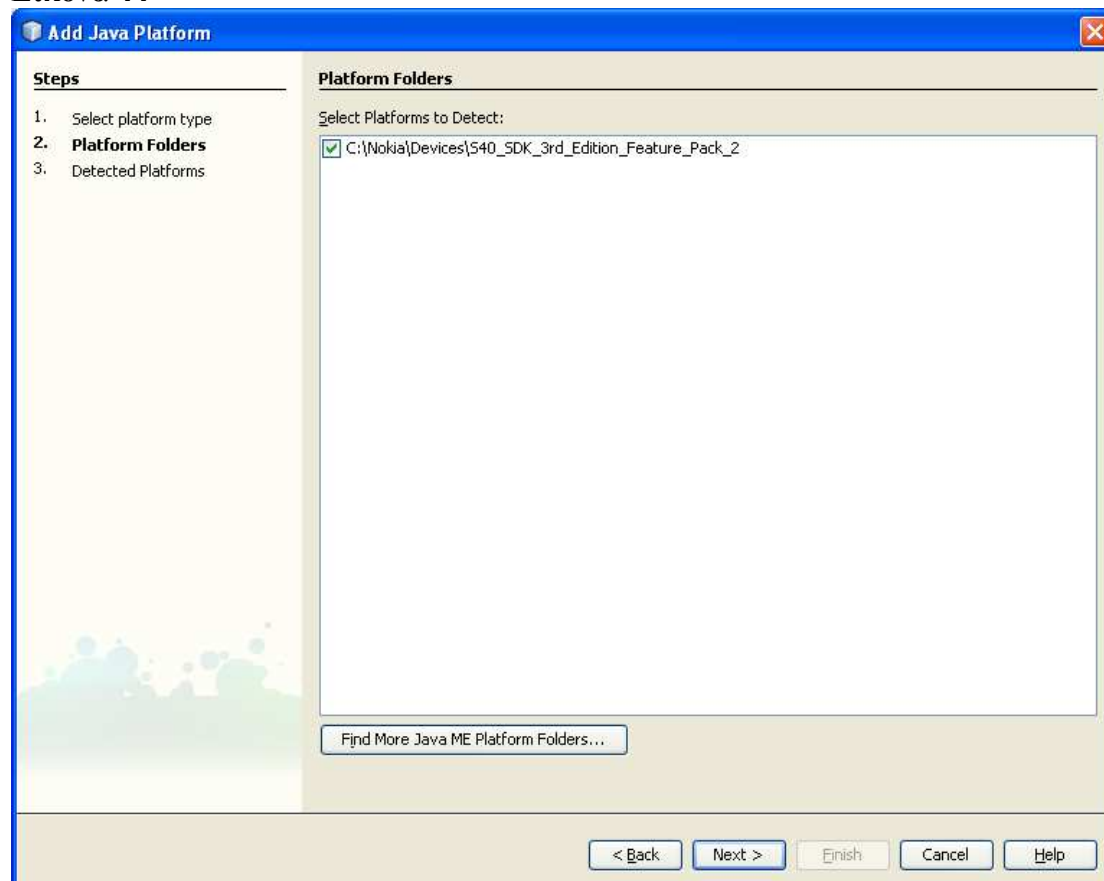
Στο παρακάτω παράθυρο ξεκινά η διαδικασία εγκατάστασης της πλατφόρμας της NOKIA στο πρόγραμμα μας. Διαλέγουμε την δεύτερη επιλογή, Java ME MIDP Platform Emulator. Πατάμε το κουμπί Next για να συνεχίσουμε παρακάτω.

**Εικόνα 43**



Σε αυτό το παράθυρο παρατηρούμε ότι το IDE εντοπίζει τις πλατφόρμες που είναι διαθέσιμες. Οι διαθέσιμες πλατφόρμες εμφανίζονται στη δεξιά μεριά. Τσεκάρουμε το κουτάκι που δείχνει στο S40\_SDK\_3<sup>rd</sup>\_Edition\_Feature\_Pack\_2. Στη συνέχεια, πατάμε το κουμπί Next.

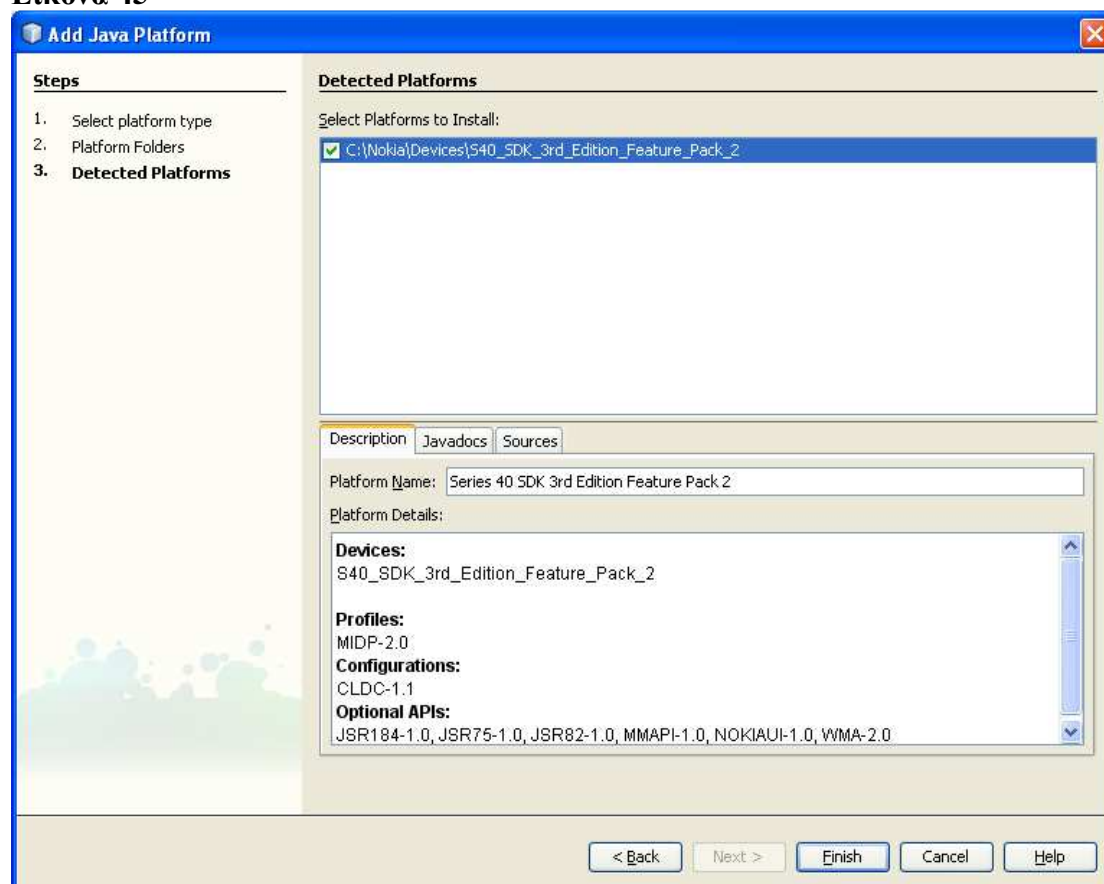
#### Εικόνα 44





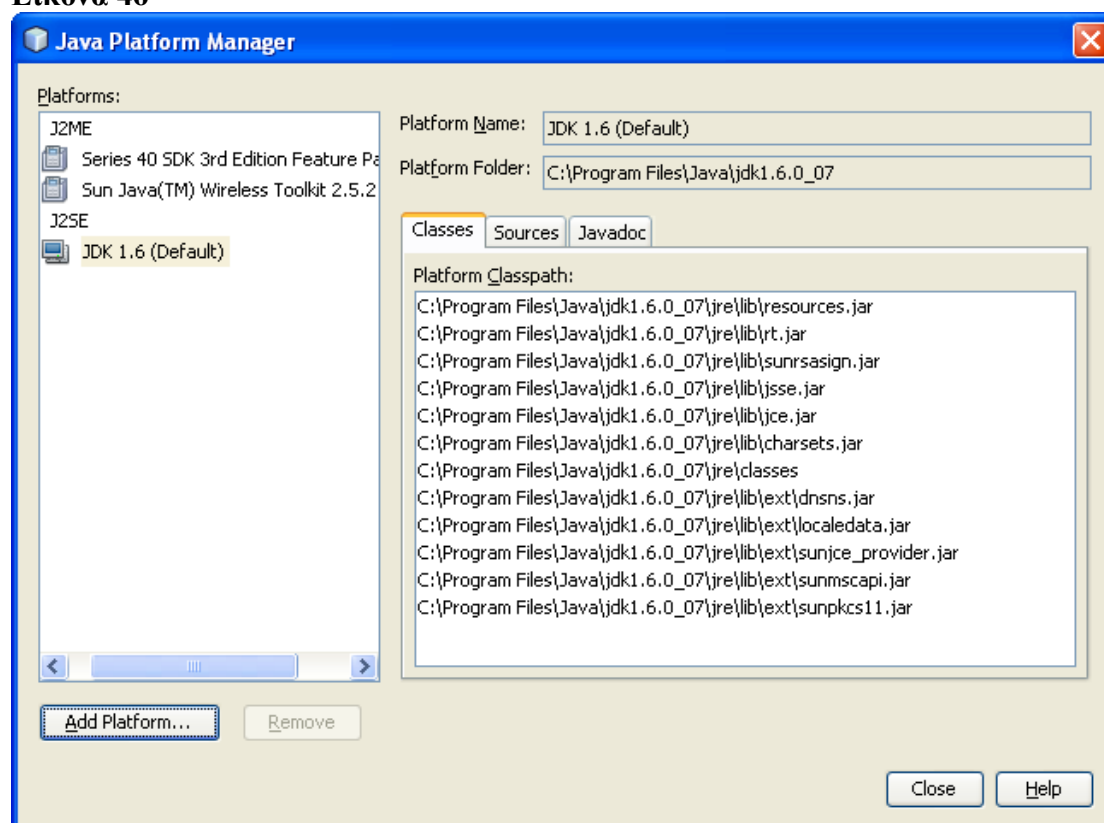
Εμφανίζεται το παρακάτω παράθυρο, το οποίο μας δείχνει λεπτομέρειες από το επιλεγμένο SDK. Στη συνέχεια, πατάμε το κουμπί Finish.

**Εικόνα 45**



Τέλος, βλέπουμε ότι επιστρέψαμε στην αρχική καρτέλα (Java Platform Manager) και παρατηρούμε ότι στην αριστερή μεριά, το SDK έχει προστεθεί στις πλατφόρμες J2ME. Η εγκατάσταση της πλατφόρμας της NOKIA είναι επιτυχής.

**Εικόνα 46**



### 3.2.4.2α Εγκατάσταση της πλατφόρμας SONY ERICSSON στον υπολογιστή

Πρέπει να κάνουμε εγκατάσταση της πλατφόρμας της SONY ERICSSON SDK 2.5.0.4 for the Java ME Platform στον υπολογιστή μας. Η διαδικασία που ακολουθήσαμε για την εγκατάσταση είναι η εξής:

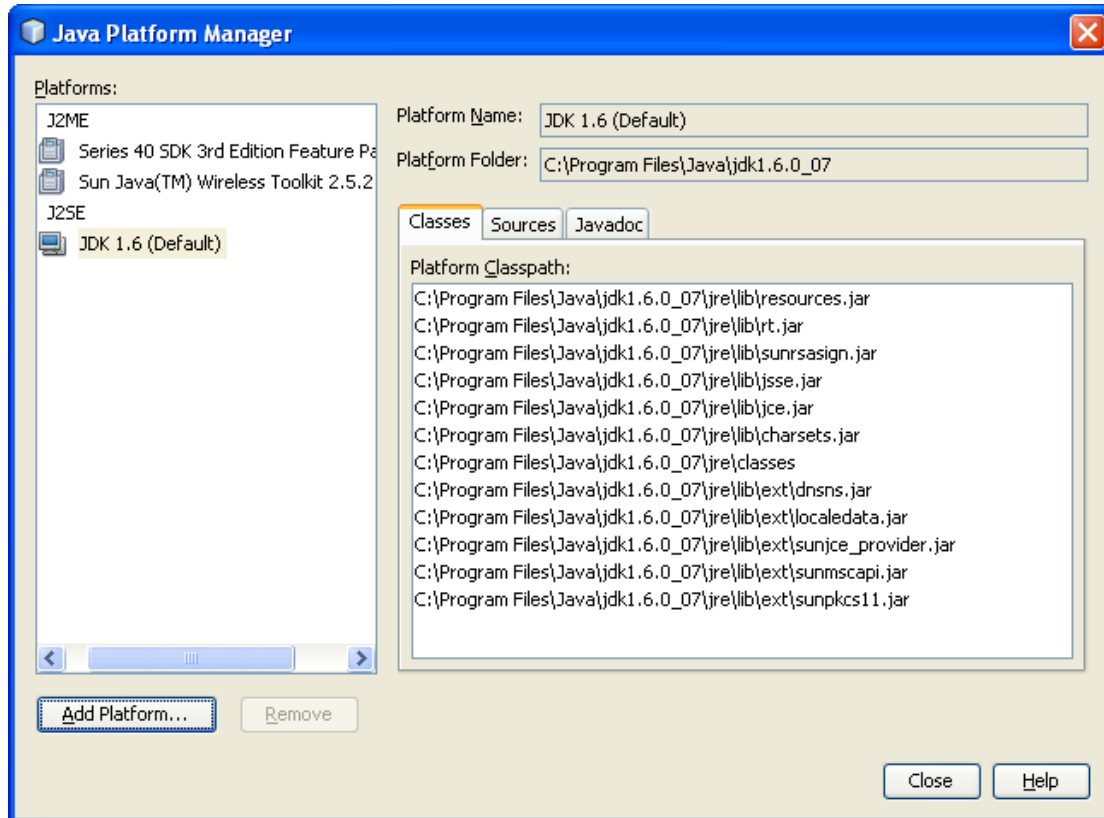
- Έχοντας κάνει download το zip αρχείο, στη συνέχεια κάνουμε δεξί κλικ στο zip αρχείο και επιλέγουμε Extract Here για να μπορέσουμε να χρησιμοποιήσουμε τα αρχεία.
- Μετά, κάνουμε διπλό κλικ στο αρχείο semc\_java\_me\_cldc\_sdk.2-5-0-4.exe για να ξεκινήσει η εγκατάσταση.
- Κατά τη διάρκεια της εγκατάστασης, επιλέγουμε που θα αποθηκεύσουμε το αρχείο. Εμείς, επιλέγουμε να το αποθηκεύσουμε στην παρακάτω διεύθυνση C:\SonyEricsson\JavaME\_SDK\_CLDC.

Η εγκατάσταση έγινε με επιτυχία. Για να φορτώσουμε το SDK μέσα σε ένα περιβάλλον IDE και πιο συγκεκριμένα στο Netbeans 6.1 θα ακολουθήσουμε την παρακάτω διαδικασία.

### 3.2.4.2β Εγκατάσταση της πλατφόρμας SONY ERICSSON στο πρόγραμμα Netbeans

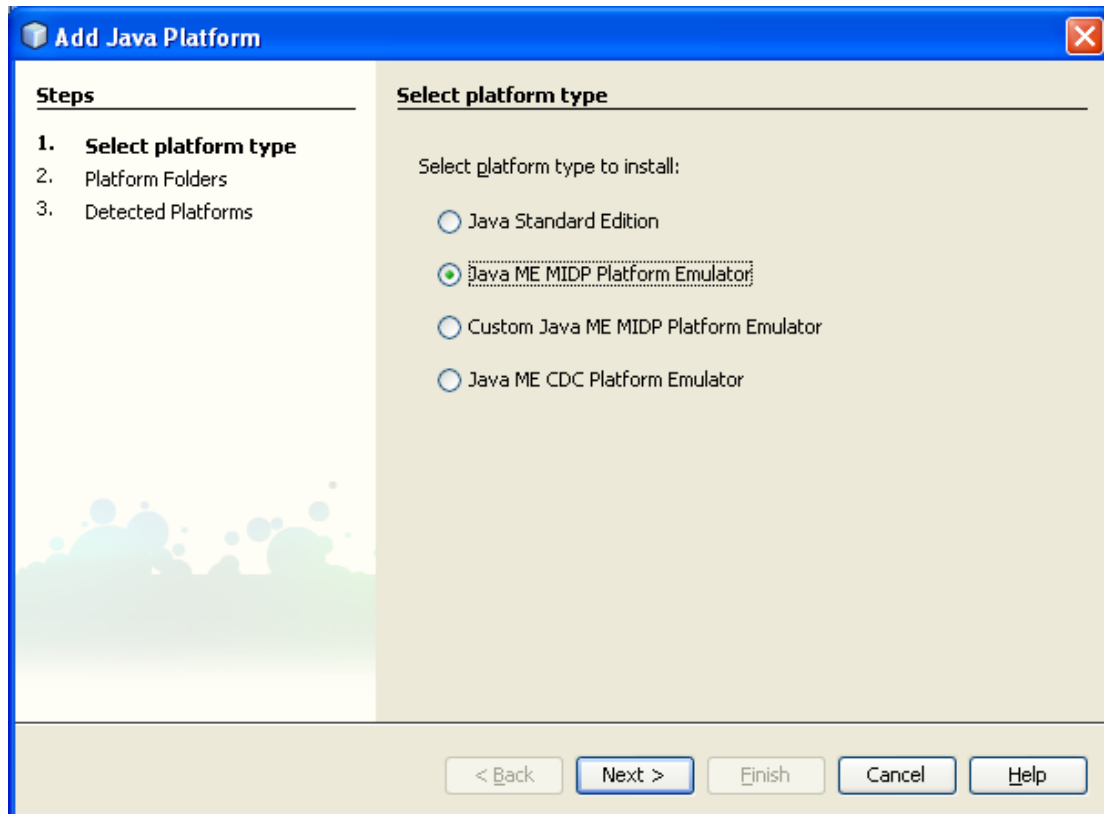
Ξεκινάμε την διαδικασία εγκατάστασης ανοίγοντας το πρόγραμμα Netbeans. Από το Tools επιλέγουμε Java Platforms και εμφανίζεται το παρακάτω παράθυρο.

Εικόνα 47



Πατάμε το κουμπί Add Platform και εμφανίζεται το παρακάτω παράθυρο.

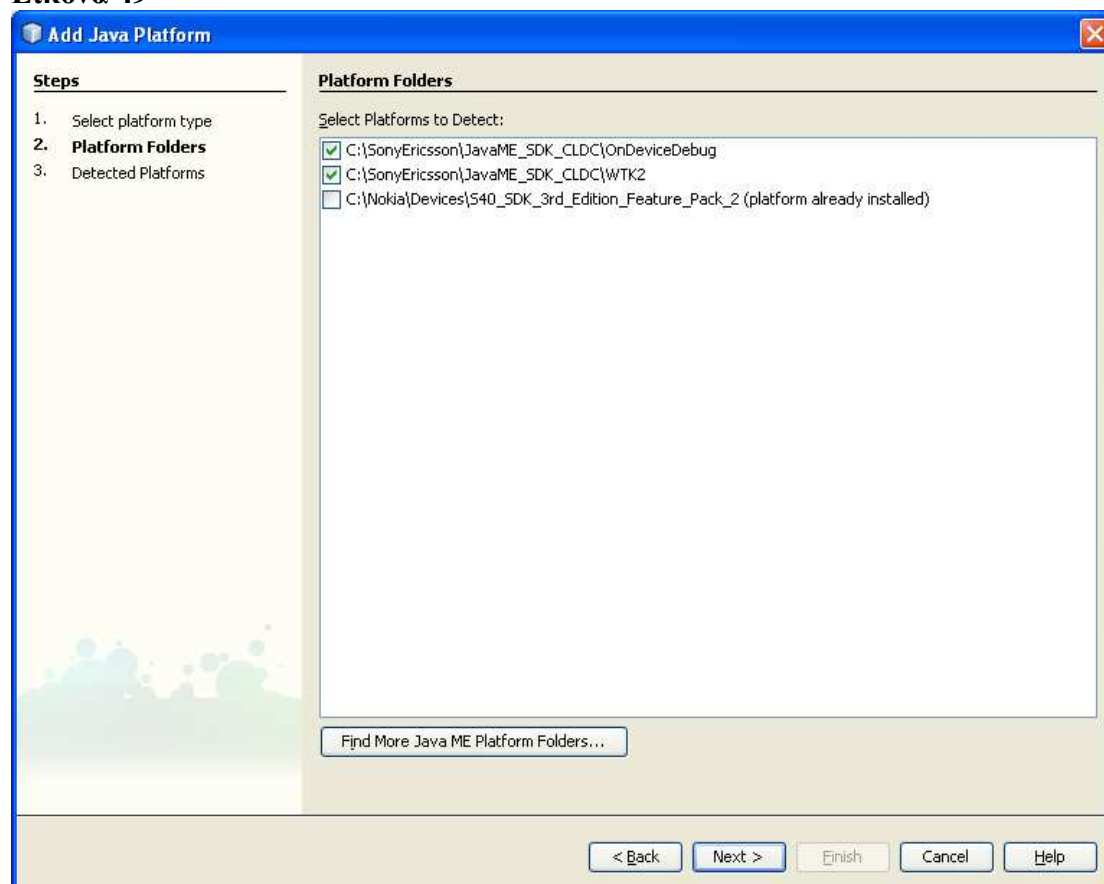
Εικόνα 48



Διαλέγουμε την δεύτερη επιλογή, Java ME MIDP Platform Emulator. Μετά, πατάμε το κουμπί Next.

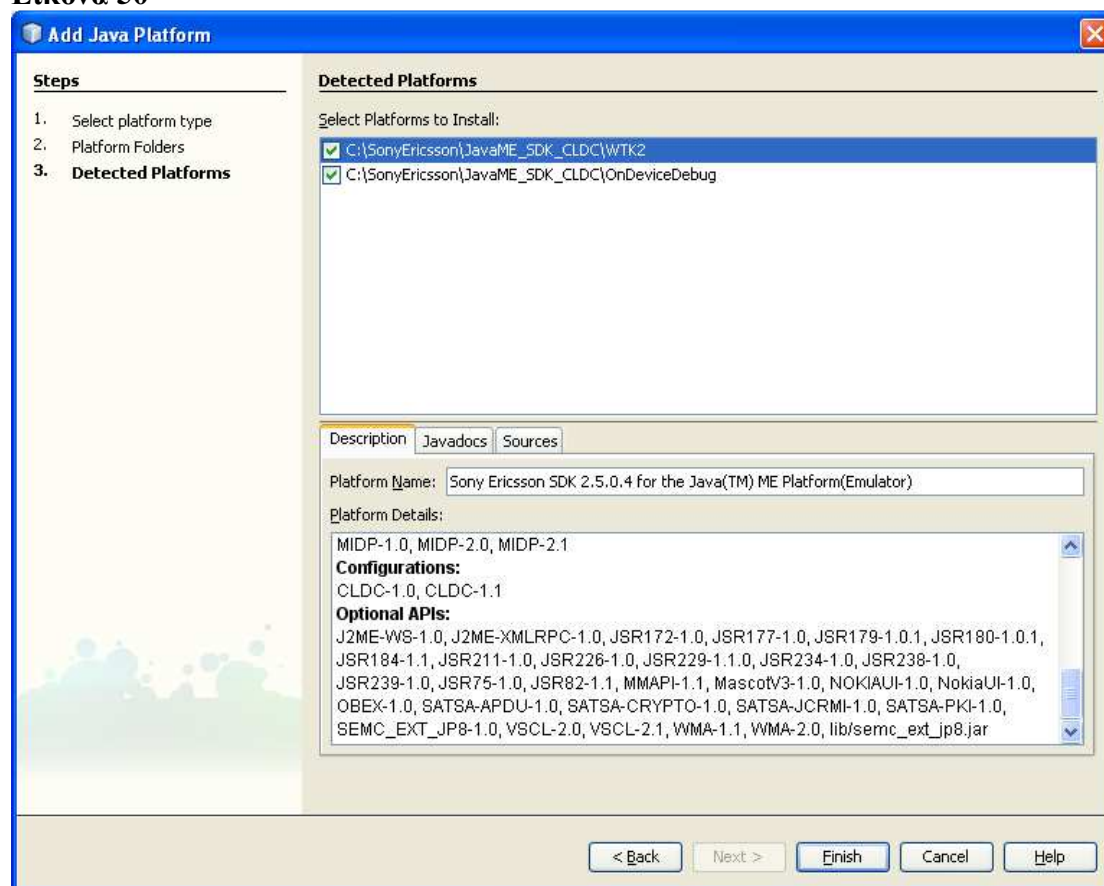
Τσεκάρουμε μόνο τα δύο πρώτα κουτάκια που αφορούν την πλατφόρμα της SONY ERICSSON. Στη συνέχεια, πατάμε το κουμπί Next.

### Εικόνα 49

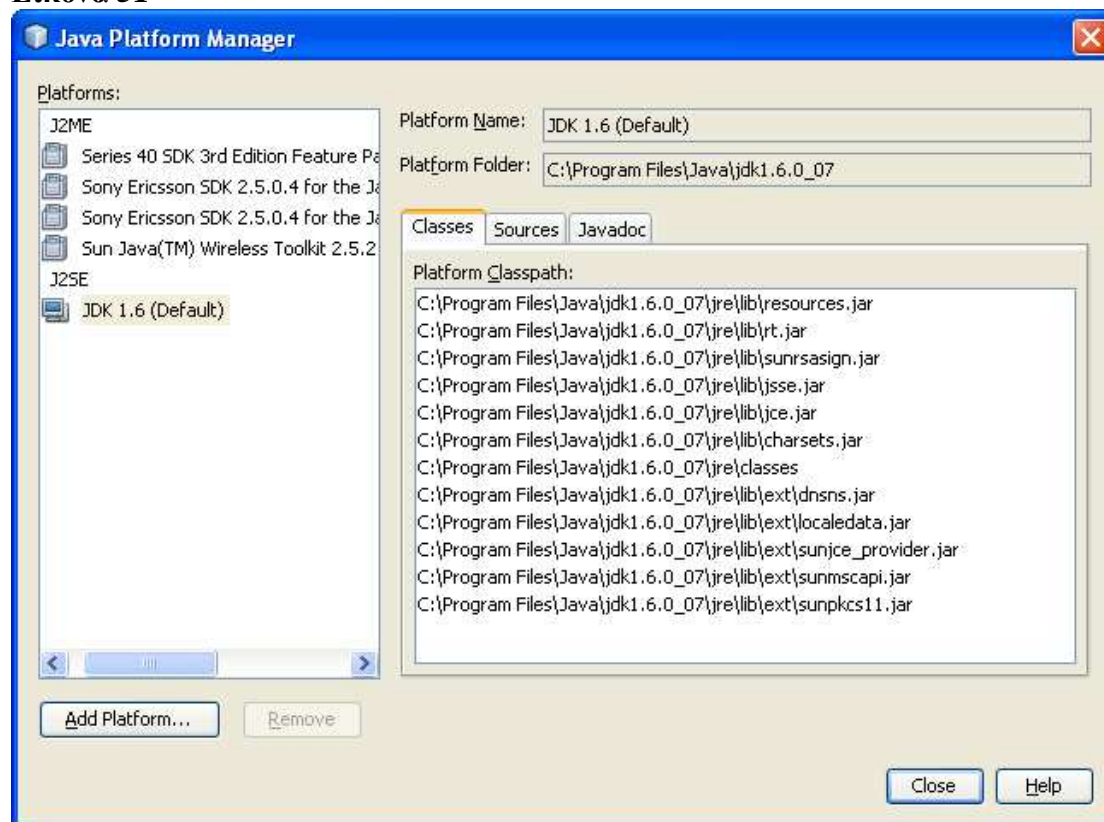


Παρακάτω βλέπουμε τις λεπτομέρειες που περιγράφουν την πλατφόρμα της SONY ERICSSON. Στη συνέχεια, πατάμε το κουμπί Finish.

### Εικόνα 50



Τέλος, βλέπουμε ότι η εγκατάσταση της πλατφόρμας της SONY ERICSSON έγινε με επιτυχία όπως βλέπουμε στο παρακάτω παράθυρο. Πατάμε το κουμπί Close για να κλείσουμε το παράθυρο.

**Εικόνα 51**

## 3.2.5 Download open-source κώδικα από το forum της NOKIA

Για να υλοποιήσουμε το project μας, χρησιμοποιήσαμε open-source κώδικα από το forum της NOKIA. Τροποποιώντας τον παρακάτω κώδικα μπορέσαμε να δημιουργήσουμε τον δικό μας κώδικα για το project μας. Ο κώδικας που χρησιμοποιήσαμε βρίσκεται στην παρακάτω διεύθυνση:

[http://wiki.forum.nokia.com/index.php/CS001006 -  
\\_Reading a text file line by line](http://wiki.forum.nokia.com/index.php/CS001006_-_Reading_a_text_file_line_by_line)

Παρακάτω βλέπουμε τον κώδικα από το forum της NOKIA

```
import java.io.IOException;
import java.io.InputStreamReader;
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Form;
import javax.microedition.midlet.MIDlet;

public class ReaderMIDlet extends MIDlet implements CommandListener {
    private Form mainForm;
    private Command exitCommand;

    /**
     * Constructor. Constructs the object and initializes
     displayables.
     */
    public ReaderMIDlet() {
        mainForm = new Form("Example MIDlet");

        exitCommand = new Command("Exit", Command.EXIT, 1);
        mainForm.addCommand(exitCommand);
        mainForm.setCommandListener(this);

        try {
            readFile("text.txt");
        } catch (IOException ex) {
            // TODO: Exception handling
        }
    }

    /**
     * Outputs the specified file onto the form.
     * @throws java.io.IOException if an exception occurs when
     reading the
     * file
     */
    private void readFile(String filename) throws IOException {
```



```

        InputStreamReader reader = new InputStreamReader(
            getClass().getResourceAsStream(filename));
        String line = null;
        // Read a single line from the file. null represents the EOF.
        while ((line = readLine(reader)) != null) {
            // Append the read line to the main form with a linefeed
            ('\n')
            mainForm.append(line + "\n");
        }
        reader.close();
    }

    /**
     * Reads a single line using the specified reader.
     * @throws java.io.IOException if an exception occurs when
    reading the
     * line
     */
    private String readLine(InputStreamReader reader) throws
    IOException {
        // Test whether the end of file has been reached. If so,
        return null.
        int readChar = reader.read();
        if (readChar == -1) {
            return null;
        }
        StringBuffer string = new StringBuffer("");
        // Read until end of file or new line
        while (readChar != -1 && readChar != '\n') {
            // Append the read character to the string. Some
            operating systems
            // such as Microsoft Windows prepend newline character
            ('\n') with
            // carriage return ('\r'). This is part of the newline
            character
            // and therefore an exception that should not be appended
            to the
            // string.
            if (readChar != '\r') {
                string.append((char) readChar);
            }
            // Read the next character
            readChar = reader.read();
        }
        return string.toString();
    }

    /**
     * From MIDlet.
     * Called when the MIDlet is started.
     */
    public void startApp() {
        // The initial display is the first form
        Display.getDisplay(this).setCurrent(mainForm);
    }

    /**
     * From MIDlet.
     * Called to signal the MIDlet to enter the Paused state.
     */
    public void pauseApp() {

```

```
        // No implementation required
    }

    /**
     * From MIDlet.
     * Called to signal the MIDlet to terminate.
     * @param unconditional whether the MIDlet has to be
unconditionally
     * terminated
     */
    public void destroyApp(boolean unconditional) {
        // No implementation required
    }

    /**
     * From CommandListener.
     * Called by the system to indicate that a command has been
invoked on a
     * particular displayable.
     * @param command the command that was invoked
     * @param displayable the displayable where the command was
invoked
     */
    public void commandAction(Command command, Displayable
displayable) {
        if (command == exitCommand) {
            // Exit the MIDlet
            notifyDestroyed();
        }
    }
}
```

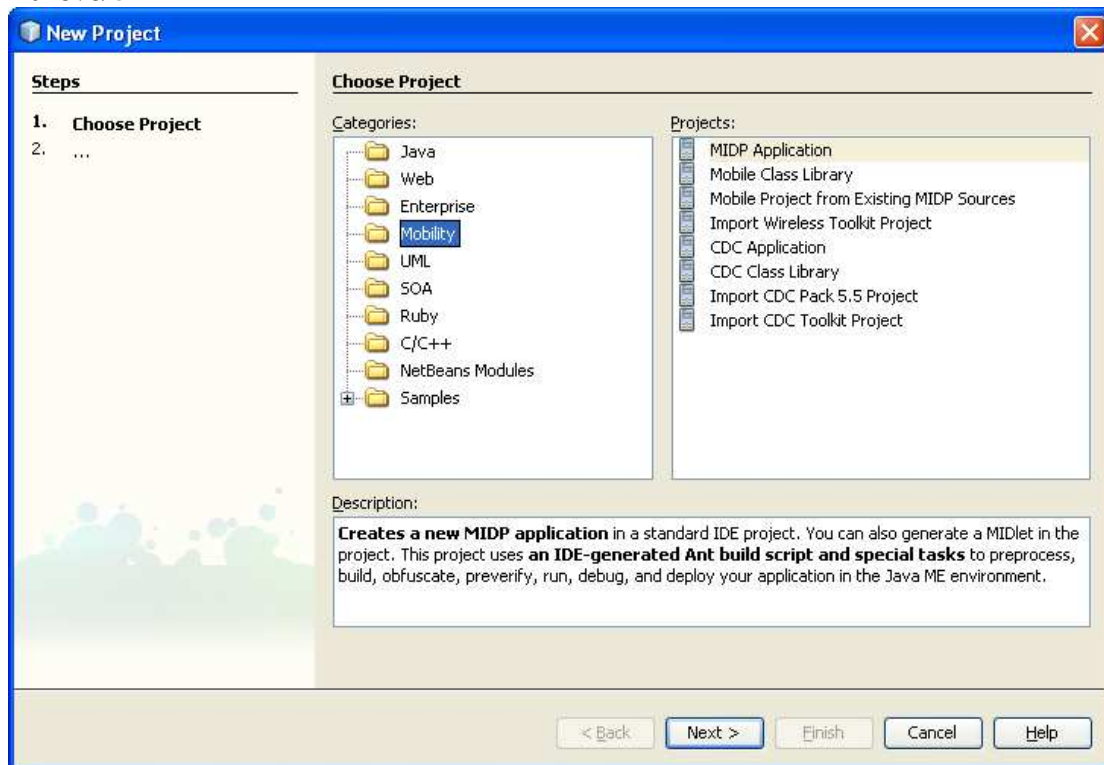
### 3.2.6 Ανάλυση του τελικού κώδικα γραμμή προς γραμμή

Έχοντας κάνει εγκατάσταση το Bluetooth Software της ASUS στον υπολογιστή μας και την εγκατάσταση των δύο πλατφόρμων είμαστε έτοιμοι να γράψουμε τον κώδικα μας.

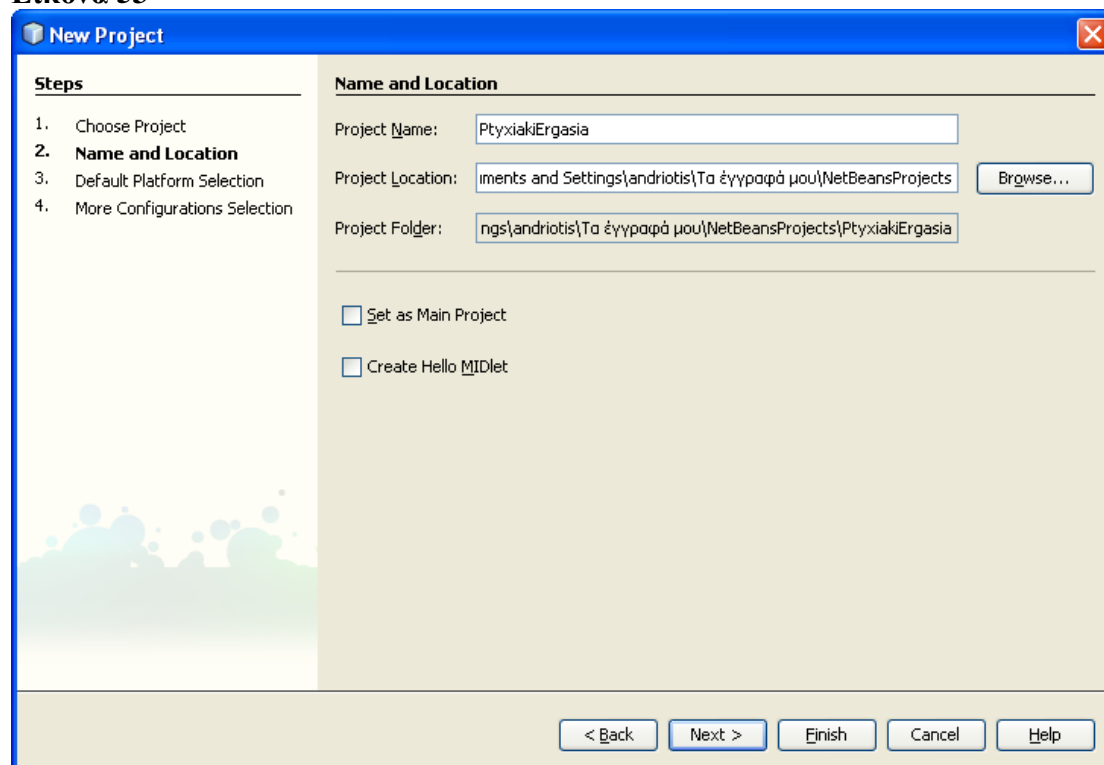
Καταρχήν, ανοίγουμε το πρόγραμμα Netbeans 6.1 και δημιουργούμε ένα καινούργιο project. Παρακάτω βλέπουμε την διαδικασία που ακολουθήσαμε:

Από το File επιλέγουμε New Project και εμφανίζεται το παρακάτω παράθυρο.

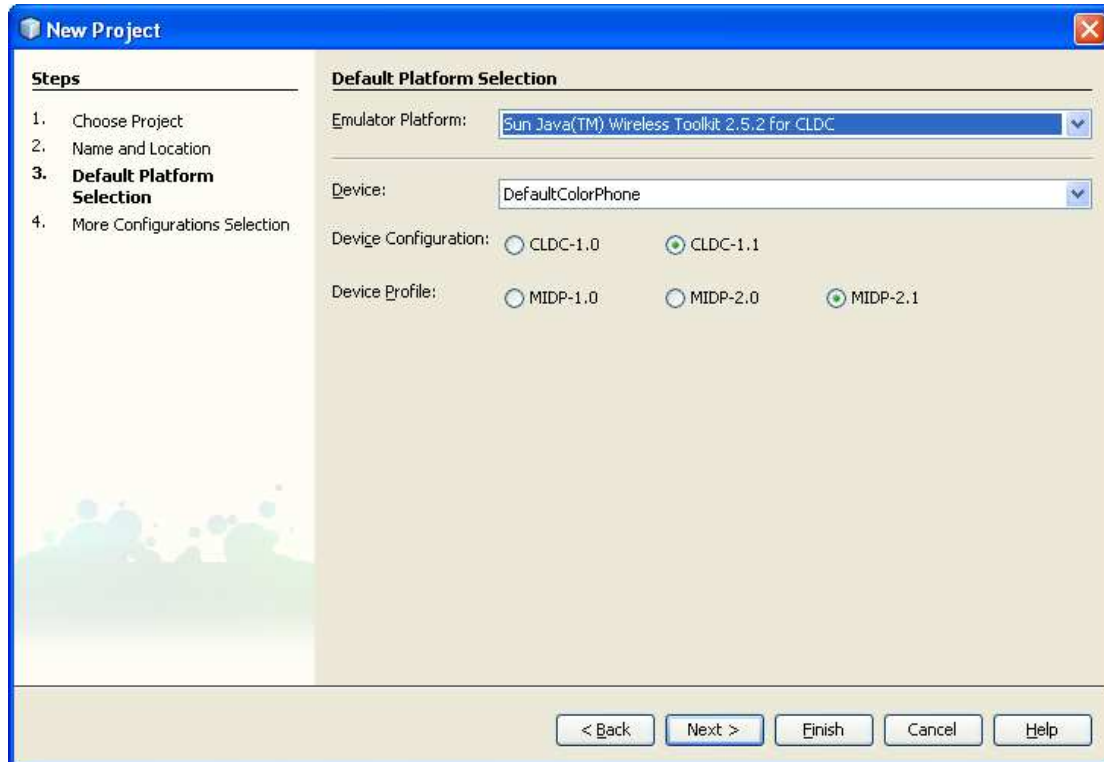
Εικόνα 52



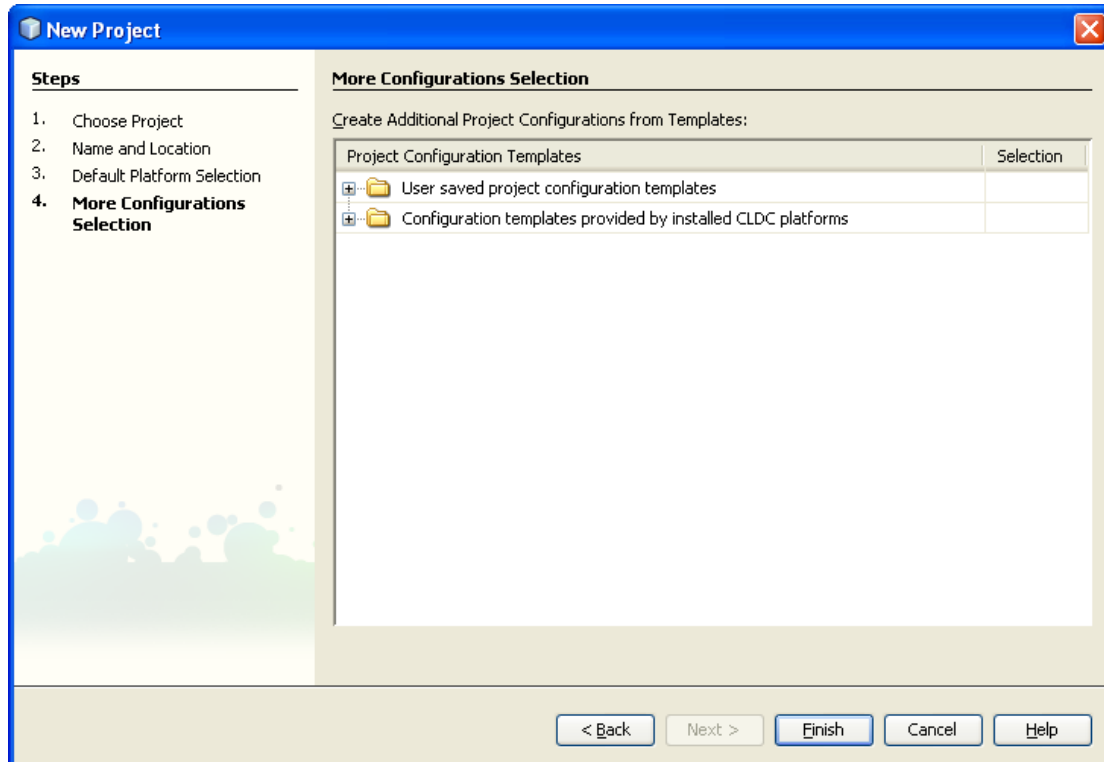
Από την πρώτη στήλη που λέγεται Categories επιλέγουμε το φάκελο Mobility. Από την δεύτερη στήλη με το όνομα Projects επιλέγουμε την πρώτη επιλογή, δηλαδή το MIDP Application. Στη συνέχεια, πατάμε το κουμπί Next και μας εμφανίζεται το παρακάτω παράθυρο.

**Εικόνα 53**

Στο παραπάνω παράθυρο ορίζουμε ότι το όνομα του Project Name θα είναι PtyxiakiErgasia. Τα Project Location και Project Folder δείχνουν σε ποιο μέρος είναι αποθηκευμένο το αρχείο μας. Τέλος, δεν τσεκάρουμε τα δύο κουτάκια Set as Main Project και το Create HelloMIDlet. Στη συνέχεια, πατάμε το κουμπί Next.

**Εικόνα 54**

Στο παραπάνω παράθυρο ορίζουμε για Emulator Platform το Sun Java(TM) Wireless Toolkit 2.5.2 for CLDC. Επιλέγουμε επίσης για Device (Συσκευή) το DefaultColorPhone. Τέλος, έχοντας επιλέξει το Emulator Platform ορίζονται αυτομάτως το Device Configuration και το Device Profile τα οποία είναι CLDC-1.1 και MIDP-2.1 αντίστοιχα. Πατάμε το κουμπί Next για να συνεχίσουμε παρακάτω.

**Εικόνα 55**

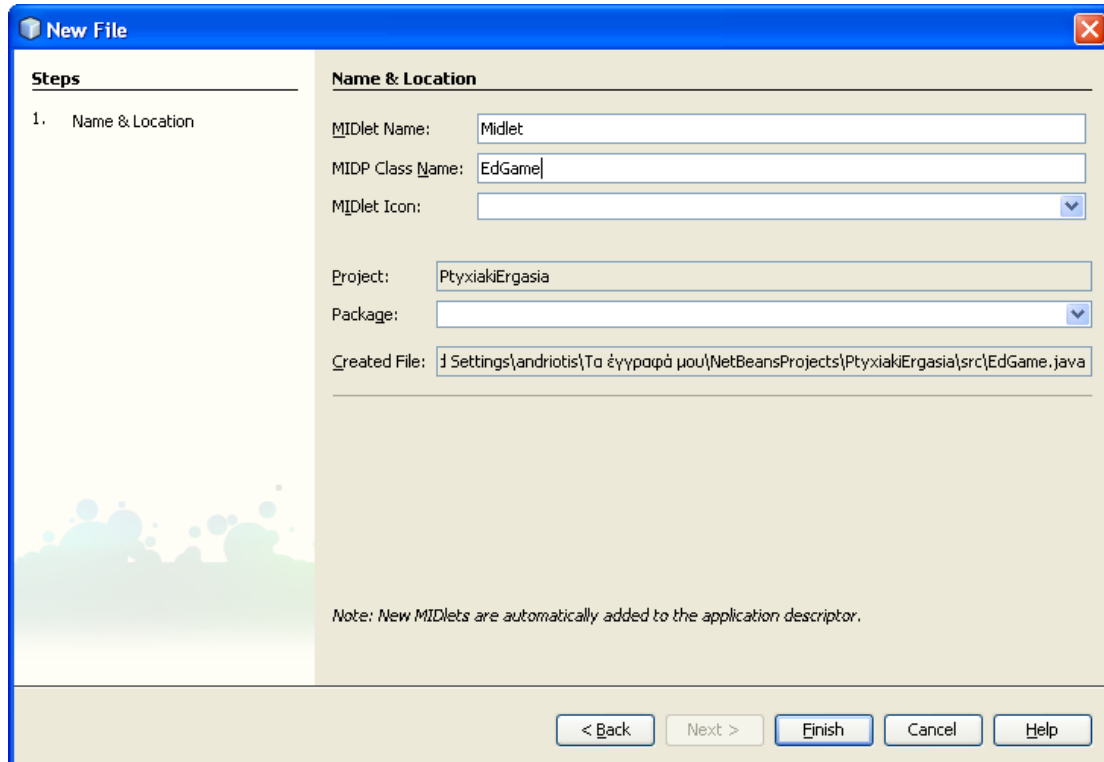
Δεν αλλάζουμε κάτι σε αυτό το παράθυρο. Πατάμε το κουμπί Finish για να δημιουργήσουμε το καινούργιο μας project.

Παρακάτω βλέπουμε την δημιουργία του νέου μας project.



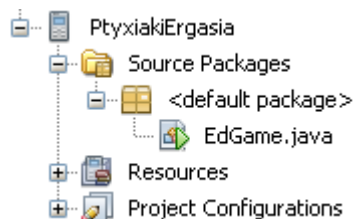
Στη συνέχεια, κάνουμε δεξί κλικ στο Source Packages και επιλέγουμε New -> MIDlet...

Ανοίγει το παρακάτω παράθυρο.

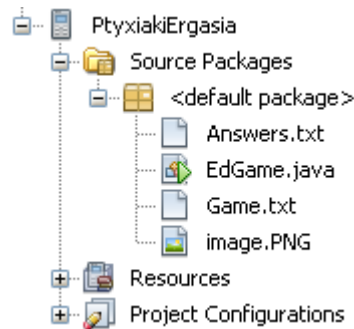
**Εικόνα 56**

Στο παραπάνω παράθυρο ορίζουμε το όνομα του MIDlet. Το ονομάζουμε EdGame και πατάμε το κουμπί Finish.

Βλέπουμε παρακάτω την δημιουργία του αρχείου EdGame.java στο Source Packages.



Στη συνέχεια, προσθέτουμε στο Source Packages τα δύο txt αρχεία που δημιουργήσαμε στο 1<sup>ο</sup> μέρος, στο πρόγραμμα DBConnection. Τα αρχεία είναι Game.txt και Answers.txt. Επίσης, στο Source Packages προσθέτουμε και μια εικόνα με το όνομα image.PNG. Έτσι βλέπουμε παρακάτω όλα τα αρχεία που έχουμε τοποθετήσει στο Source Packages του project PtyxiakiErgasia.



Στο αρχείο Midlet.java είναι το αρχείο εκείνο που θα γράψουμε τον κώδικα μας. Στην αρχή, όταν δημιουργείται το αρχείο ενός MIDlet εμφανίζεται ο παρακάτω κώδικας. Είναι ο κύκλος ζωής ενός MIDlet που είδαμε και αναλύσαμε στην Εισαγωγή.

```

1  /*
2   * To change this template, choose Tools | Templates
3   * and open the template in the editor.
4   */
5
6  import javax.microedition.midlet.*;
7  import javax.microedition.lcdui.*;
8
9  /**
10   * @author andriotis
11   */
12  public class EdGame extends MIDlet {
13      public void startApp() {
14      }
15
16      public void pauseApp() {
17      }
18
19      public void destroyApp(boolean unconditional) {
20      }
21  }
22

```

Θα αρχίσουμε να γράφουμε τον τελικό κώδικα και θα χρησιμοποιήσουμε και τον κώδικα που «κατεβάσαμε» από το forum της NOKIA. Παρακάτω βλέπουμε ολοκληρωμένο των κώδικα μας.



```
1  /*
2   * To change this template, choose Tools | Templates
3   * and open the template in the editor.
4   */
5
6  import javax.microedition.lcdui.Command;
7  import javax.microedition.lcdui.CommandListener;
8  import javax.microedition.lcdui.Display;
9  import javax.microedition.lcdui.Displayable;
10 import javax.microedition.lcdui.Form;
11 import javax.microedition.lcdui.TextField;
12 import javax.microedition.midlet.MIDlet;
13 import java.io.*;
14 import javax.microedition.lcdui.Image;
15 import javax.microedition.lcdui.ImageItem;
16
17 /**
18  * @author Andrioti Zacharo-Evangelia
19  */
20 public class Midlet extends MIDlet implements CommandListener{
21
22     //Δημιουργία αντικειμένων των κλάσεων Form,Command,TextField,Display, Image
23     //και ImageItem
24     private Form mainForm;
25     private Command exitCommand;
26     private Command read;
27     private Command load;
28     private Command back;
29     private TextField text;
30     private Display mDisplay;
31     private Form form;
32     private Image image;
33     private ImageItem imageItem;
34
35
36     public EdGame () {
37
38         //Δημιουργία της mainForm
39         mainForm = new Form("Example MIDlet");
40         //Δημιουργία του κουμπιού Read
41         read = new Command("Read", Command.SCREEN, 1);
```

```

42 //Δημιουργία του κουμπιου Exit
43 exitCommand = new Command("Exit", Command.EXIT, 1);
44 //Δημιουργία του κουμπιου Load
45 load = new Command("Load", Command.OK, 1);
46 //Δημιουργία του κουμπιου Back
47 back = new Command("Back", Command.BACK, 2);
48
49
50 try
51 {
52     // Δημιουργία της εικόνας
53     image=Image.createImage("/image.PNG");
54     imageItem = new ImageItem("Πτυχιακή Εργασία της Ζαχαρώ-Ευαγγελία " +
55         "Ανδριώτη AM 1444",
56         image, ImageItem.LAYOUT_CENTER, "image");
57 }catch(Exception e){
58     e.getMessage();
59 }
60
61 //Εμφάνιση-τοποθέτηση του ImageItem και των commands στην mainForm
62 mainForm.append(imageItem);
63 mainForm.addCommand(exitCommand);
64 mainForm.addCommand(read);
65 mainForm.addCommand(load);
66 mainForm.setCommandListener(this);
67
68 }
69
70 private void readFile(String filename) throws IOException {
71     InputStreamReader reader = new InputStreamReader(
72         getClass().getResourceAsStream(filename), "UTF-8");
73     String line = null;
74
75     // Read a single line from the file.
76     while ((line = readLine(reader)) != null) {
77
78         // Append the read line to the main form with a linefeed ('\n')
79         mainForm.append(line + "\n");
80
81     }
82     reader.close();
83 }
84
85
86 /**
87  * Reads a single line using the specified reader.
88  * @throws java.io.IOException if an exception occurs when reading the
89  * line
90  */
91 private String readLine(InputStreamReader reader) throws IOException {
92
93     // Test whether the end of file has been reached. If so, return null.
94     int readChar = reader.read();
95     if (readChar == -1) {
96         return null;
97     }
98     StringBuffer string = new StringBuffer("");
99

```

```

100 // Read until end of file or new line
101 while (readChar != -1 && readChar != '\n') {
102
103     while(readChar == '*')
104     {
105         //create textfield
106         text = new TextField("Απόννηση: " , "", 10, TextField.ANY);
107
108         text.setMaxSize(1);
109
110         mainForm.append(text);
111         readChar = reader.read();
112     }
113
114     // Append the read character to the string. Some operating systems
115     // such as Microsoft Windows prepend newline character ('\n') with
116     // carriage return ('\r'). This is part of the newline character
117     // and therefore an exception that should not be appended to the
118     // string.
119
120     if (readChar != '\r') {
121         string.append( (char) readChar);
122     }
123
124     // Read the next character
125     readChar = reader.read();
126
127 }
128 return string.toString();
129 }
130
131
132
133 private String readLineText(InputStreamReader readAnswer) throws IOException {
134
135     // Test whether the end of file has been reached. If so, return null.
136     int readChar = readAnswer.read();
137     if (readChar == -1) {
138         return null;
139     }
140     StringBuffer string = new StringBuffer("");
141
142     // Read until end of file
143     while (readChar != -1 )
144     {
145         string.append( (char) readChar);
146
147         // Read the next character
148         readChar = readAnswer.read();
149
150     }
151
152     return string.toString();
153 }
154

```

```
155
156  /**
157   * From MIDlet.
158   * Called when the MIDlet is started.
159   */
160  public void startApp() {
161      // The initial display is the first form
162
163      mDisplay = Display.getDisplay(this);
164      mainForm.setCommandListener(this);
165      mDisplay.setCurrent(mainForm);
166
167  }
168
169
170  /**
171   * From MIDlet.
172   * Called to signal the MIDlet to enter the Paused state.
173   */
174  public void pauseApp() {
175      // No implementation required
176  }
177
178  /**
179   * From MIDlet.
180   * Called to signal the MIDlet to terminate.
181   * @param unconditional whether the MIDlet has to be unconditionally
182   * terminated
183   */
184  public void destroyApp(boolean unconditional) {
185      // No implementation required
186  }
187
188
189
190  /**
191   * From CommandListener.
192   * Called by the system to indicate that a command has been invoked on a
193   * particular displayable.
194   * @param command the command that was invoked
195   * @param displayable the displayable where the command was invoked
```

```

196 L      */
197      public void commandAction(Command command, Displayable displayable) {
198          if(command==read)
199          {
200              MainForm.delete(0);
201              try
202              {
203                  readFile("Game.txt");
204              }
205              catch (IOException ex)
206              {
207                  ex.getMessage();
208              }
209          }
210
211          if (command==load)
212          {
213              mDisplay = Display.getDisplay(this);
214              form = new Form("Καταχώρηση Απαντήσεων");
215          try
216          {
217              {
218
219                  InputStreamReader readAnswer = new InputStreamReader(
220                      getClass().getResourceAsStream("Answers.txt"));
221
222
223                  String line = null;
224
225                  int counter,i=0,s=0;
226                  String n=null;
227                  String pin[]=new String[10];
228                  String mat[]=new String[10];
229
230                  //Αποθήκευση των απαντήσεων του χρήστη στον πίνακα pin
231                  for ( counter=0; counter<mainForm.size(); counter++)
232                  {
233                      if ( i<10 && mainForm.get( counter) instanceof TextField )
234                      {
235                          n = (( TextField)mainForm.get(counter)).getString();
236                          pin[i]=n;
237
238                          form.append("Απάντηση "+i+": " + pin[i] + "\n");
239
240                          i++;
241                      }
242                  }
243
244                  //Αποθήκευση των περιεχομένων του αρχείου answers.txt στον πίνακα mat
245                  while ((line = readLineText(readAnswer)) != null)
246                  {
247                      for(s=0;s<10;s++)
248                      {
249                          mat[s]=line.substring(s, s+1);
250                      }
251                  }
252
253                  int count=0;
254
255                  //Σύγκριση πινάκων
256                  for (i=0;i<10;i++)
257                  {
258                      if (pin[i].equals(mat[i]))

```

```

259         {
260             form.append("Η απάντηση " + i + " είναι σωστή\n");
261             count=count+10;
262         }
263     else
264     {
265         form.append("Η απάντηση " + i + " είναι λάθος\n");
266         count=count+0;
267     }
268
269     }
270     form.append("Τα αποτελέσματα είναι: "+count+"/100");
271
272     readAnswer.close();
273
274     }
275     catch (IOException ex)
276     {
277         ex.getMessage();
278     }
279
280
281     mDisplay.setCurrent(form);
282
283     form.addCommand(back);
284     form.setCommandListener(this);
285     }
286     if (command == exitCommand)
287     {
288         // Exit the MIDlet
289         notifyDestroyed();
290     }
291     if (command == back)
292     {
293         mDisplay.setCurrent(mainForm);
294     }
295     }
296     }
297

```

Μερικά σχόλια για τον παραπάνω κώδικα

**Γραμμές 6-15:** Όπως έχουμε προαναφέρει στο κεφάλαιο της Εισαγωγής θα δουλέψουμε με midlets. Για αυτό τον λόγο εισάγουμε τις αντίστοιχες βιβλιοθήκες οι οποίες είναι: javax.microedition.midlet και javax.microedition.lcdui. Τα πακέτα αυτά δίνουν δυνατότητα στην εφαρμογή να μπορεί να «τρέξει» σε κινητό τηλέφωνο. Επίσης, χρησιμοποιήσαμε το πακέτο java.io. Όπως βλέπουμε και στο πρόγραμμα από το πακέτο javax.microedition.lcdui θα χρησιμοποιήσουμε τις παρακάτω κλάσεις Command, CommandListener, Display, Displayable, Form, TextField, Image, ImageItem.

**Γραμμή 20:** Η κλάση EdGame είναι υποκλάση της MIDlet και υλοποιεί (implements) μια διασύνδεση, την CommandListener.

**Γραμμές 22-33:** Δημιουργούμε τα αντικείμενα: mainForm της κλάσης Form, exitCommand της κλάσης Command, read της κλάσης Command, load της κλάσης Command, back της κλάσης Command, text της κλάσης TextField, mDisplay της κλάσης Display, form της κλάσης Form, image της κλάσης Image, imageItem της κλάσης ImageItem. Όλα τα ορίζουμε private.

**Γραμμές 36-68:** Δημιουργία constructor. Τον constructor τον χρησιμοποιούμε για να αρχικοποιήσουμε τα αντικείμενα, τα οποία στη συνέχεια θα τα εμφανίσουμε στην οθόνη του κινητού. Για να τοποθετήσουμε αντικείμενα στην οθόνη χρησιμοποιούμε μια Form. Ορίζουμε μια Form (φόρμα) που λέγεται mainForm και παίρνει για όρισμα την επικεφαλίδα της φόρμας, δηλαδή Educational Game. Στις **γραμμές 40-47** ορίζουμε 4 Commands (εντολές). Τα commands εμφανίζονται στο κάτω μέρος της οθόνης και έτσι δίνει την δυνατότητα στον χρήστη να επιλέξει την εντολή που θέλει για μια εφαρμογή. Σε αυτή την περίπτωση έχουμε ένα command με το όνομα read, ένα command με το όνομα load, ένα command με το όνομα exitCommand και ένα command με το όνομα back. Τα commands παίρνουν τρία ορίσματα: Το πρώτο είναι το όνομα με το οποίο θα εμφανίζεται στην οθόνη του κινητού, το δεύτερο όρισμα είναι ο τύπος, και το τρίτο όρισμα είναι η προτεραιότητα. Στις **γραμμές 50-59** η κλάση Image χρησιμοποιείται για να αποθηκεύει δεδομένα γραφικών εικόνων. Θα πρέπει να φορτώνονται οι εικόνες σε PNG format. Χρησιμοποιούμε την μέθοδο createImage() της κλάσης Image. Η ImageItem είναι ένα item που περιέχει μια εικόνα. Δημιουργούμε ένα Image από το όνομα του αρχείου της εικόνας, το οποίο αρχείο θα πρέπει να το αποθηκεύσουμε στο Source Package έτσι ώστε μετά να τοποθετηθεί μέσα στο JAR αρχείο που περιέχει το MIDlet που έχουμε φτιάξει. Στη συνέχεια, δημιουργούμε ένα αντικείμενο της κλάσης ImageItem με το όνομα imageItem το οποίο παίρνει για ορίσματα τον τίτλο που θα εμφανίζεται πάνω από την εικόνα, το αντικείμενο της κλάσης Image, η τοποθέτηση της εικόνας στη οθόνη του κινητού θα είναι στο κέντρο και το τελευταίο όρισμα είναι το όνομα του αρχείου που περιέχει την εικόνα. Στις **γραμμές 62-66** έχουμε την εμφάνιση-τοποθέτηση του ImageItem και των commands στη mainForm. Χρησιμοποιώντας τις μεθόδους append() και addCommand() μπορούμε να εμφανίσουμε την εικόνα και τις εντολές στην οθόνη. Επίσης χρησιμοποιούμε τη μέθοδο setCommandListener για την λειτουργία των commands στην mainForm.

**Γραμμές 70-83:** Χρησιμοποιούμε μια μέθοδο με το όνομα `readFile` που παίρνει για όρισμα το όνομα του `txt` αρχείου που θέλουμε να διαβάσει. Με αυτή τη μέθοδο διαβάζουμε κάθε φορά μια γραμμή από το αρχείο και ελέγχουμε αν ισούται με `null`. Για να γίνει ο έλεγχος καλούμε μια άλλη μέθοδο την `readLine` (θα δούμε την λειτουργία της παρακάτω). Αν είναι διάφορο του `null` τότε θα εμφανίζεται η γραμμή στην οθόνη. Αυτή η διαδικασία θα γίνεται μέχρι ότου η γραμμή θα ισούται με `null`. Δηλαδή, μέχρι τέλους του αρχείου. Στη γραμμή 82 έχουμε την έξοδο από το αρχείο μας. Τέλος, Για να διαβάσουμε το αρχείο χρησιμοποιήσαμε `UTF-8 format`.

**Γραμμές 91-129:** Έχουμε την μέθοδο `readLine` η οποία ελέγχει καταρχήν αν έχουμε φτάσει στο τέλος του αρχείου ή όχι. Αν ναι, επιστρέφει `null`. Αν όχι, συνεχίζουμε και μπαίνουμε σε έναν επαναληπτικό βρόγχο που διαβάζει μέχρι το τέλος του αρχείου ή μιας καινούργιας γραμμής. Αν διαβάσει τον χαρακτήρα `*` (αστερίσκο) τότε στη θέση του αστερίσκου δημιουργούμε ένα `text field`. Τοποθετούμε το `text field` στην `mainForm` για να το εμφανίσουμε στην οθόνη του κινητού. Μετά διαβάζουμε τον επόμενο χαρακτήρα. Συνεχίζουμε στον εξωτερικό βρόγχο και ελέγχουμε τον χαρακτήρα να είναι διάφορος του `\r` και μετατρέπουμε τον χαρακτήρα σε `String`. Διαβάζουμε στη συνέχεια τον επόμενο χαρακτήρα. Βγαίνουμε από τον βρόγχο και επιστρέφουμε το `String`.

**Γραμμές 133-153:** Η μέθοδος `readLineText` διαβάζει το `txt` αρχείο `Answers.txt`. Διαβάζει το αρχείο όπου έχουν αποθηκευτεί οι σωστές απαντήσεις και επιστρέφει πίσω τα αποτελέσματα σε ένα `string`.

**Γραμμές 160-167:** Όταν το `MIDlet` ξεκινάει για πρώτη φορά, καλείται η μέθοδος `startApp()`. Στη μέθοδο αυτή, ενεργοποιείται ουσιαστικά η εφαρμογή. Εμφανίζεται η `mainForm` με τα αντικείμενα στην οθόνη του κινητού και επίσης τα `commands` ενεργοποιούνται.

**Γραμμές 174-176:** Μέθοδος `pauseApp()`. Δεν απαιτείται υλοποίηση της μεθόδου.

**Γραμμές 184-186:** Μέθοδος `destroyApp()`. Δεν απαιτείται υλοποίηση της μεθόδου.

**Γραμμές 197-295:** Το `commandAction` παίρνει δυο ορίσματα: το ένα είναι `command` και το άλλο είναι ένα `displayable`. Στη **γραμμή 198**, αν το `command` ισούται με την εντολή `read` τότε καλούμε την μέθοδο `readFile` παίρνοντας για όρισμα το αρχείο `Game.txt`. Αν δεν εκτελεστεί σωστά ο κώδικας τότε θα μας εμφανίσει την εξαίρεση `IOException`. Στη **γραμμή 200** χρησιμοποιούμε τη μέθοδο `delete` η οποία διαγράφει-αφαιρεί την εικόνα από την οθόνη. Στη **γραμμή 212** αν το `command` ισούται με `load` τότε να εμφανίσει στην οθόνη την καινούργια φόρμα. Στη συνέχεια, να διαβάσει το



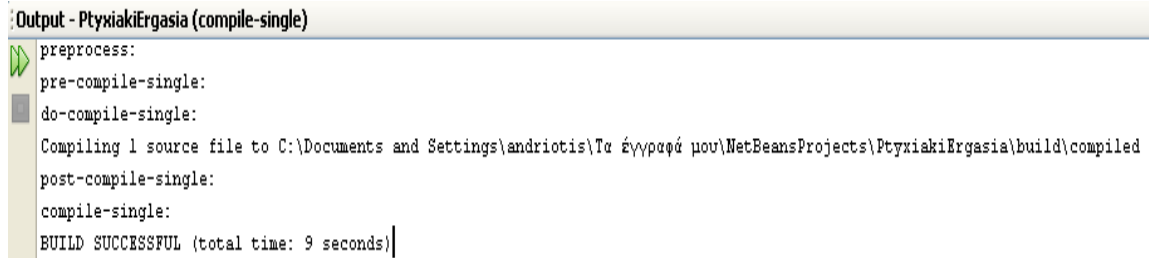
αρχείο Answers.txt. Ορίζουμε δυο String μεταβλητές με το όνομα line και η άλλη με το όνομα n και η κάθε μια ισούται με null. Επίσης, ορίζουμε μια ακέραια μεταβλητή με το όνομα counter, ορίζουμε δυο ακέραιες μεταβλητές i=0 και s=0 και τέλος ορίζουμε δυο String πίνακες που ο κάθε πίνακας θα έχει 10 θέσεις. Ο ένας πίνακας λέγεται pin και ο άλλος mat. Στη συνέχεια, μπαίνουμε σε ένα βρόγχο επανάληψης. Σε αυτόν τον βρόγχο ο χρήστης έχοντας απαντήσει στις ερωτήσεις συμπληρώνοντας τα αντίστοιχα text fields στη συνέχεια οι απαντήσεις αυτές αποθηκεύονται με τη σειρά σε ένα πίνακα pin. Ο επόμενος βρόγχος επανάληψης αποθηκεύει τις σωστές απαντήσεις από το αρχείο Answers.txt σε έναν πίνακα με το όνομα mat. Στη συνέχεια, συγκρίνουμε τους δυο πίνακες. Αν τα περιεχόμενα είναι ίδια τότε ο χρήστης έδωσε σωστή απάντηση και ο μετρητής αποτελεσμάτων αυξάνεται +10. Αν όχι, τότε ο χρήστης απάντησε λάθος και ο μετρητής αποτελεσμάτων παραμένει με την ίδια τιμή. Στο τέλος, εμφανίζεται το αποτέλεσμα του χρήστη, δηλαδή πόσο τοις εκατό επιτυχία είχε στις ερωτήσεις. Ο παραπάνω κώδικας γράφτηκε μέσα σε μια try-catch. Παρακάτω στον κώδικα έχουμε τις εντολές για την εμφάνιση της δεύτερης φόρμας και την τοποθέτηση του κουμπιού εντολής Back. Επίσης, χρησιμοποιούμε την μέθοδο setCommandListener για να εκτελεστούν τα commands και στη δεύτερη φόρμα. Στις **γραμμές 286-290**, αν το command ισούται με την εντολή exitCommand τότε γίνεται η έξοδος από το MIDlet (από την εφαρμογή) καλώντας την μέθοδο notifyDestroyed().

Τέλος, στις **γραμμές 291-294**, αν το command ισούται με την εντολή back τότε εμφανίζεται η προηγούμενη φόρμα (Form) στην οθόνη.

### 3.2.7 Εφαρμογή του κώδικα με χρήση εξομοιωτή (emulator) της Sun Java Wireless Toolkit και εμφάνιση αποτελεσμάτων

Έχοντας ολοκληρώσει τον κώδικα, κάνουμε δεξί κλικ στο EdGame.java και επιλέγουμε Compile File. Η διαδικασία είναι επιτυχής όπως βλέπουμε παρακάτω.

#### Εικόνα 57



```

Output - PtychiakiErgasia (compile-single)
preprocess:
pre-compile-single:
do-compile-single:
Compiling 1 source file to C:\Documents and Settings\andriotis\Τα έγγραφά μου\NetBeansProjects\PtychiakiErgasia\build\compiled
post-compile-single:
compile-single:
BUILD SUCCESSFUL (total time: 9 seconds)

```

Στη συνέχεια, κάνουμε δεξί κλικ στο αρχείο PtychiakiErgasia και επιλέγουμε Run για να «τρέξουμε» την εφαρμογή και να δούμε τα αποτελέσματα. Η εφαρμογή θα «τρέξει» χρησιμοποιώντας τον εξομοιωτή (emulator) της Sun Java Wireless Toolkit.

#### «Τρέχοντας» την εφαρμογή - Περιγραφή

Στην εικόνα 58 πατάμε στον κουμπί Launch για να «φορτώσουμε» την εφαρμογή μας. Όπως βλέπουμε στην εικόνα 59 η εφαρμογή μας αρχίζει και «τρέχει». Κάτω αριστερά υπάρχει το κουμπί Exit που είναι για την έξοδο μας από την εφαρμογή και δεξιά υπάρχει το κουμπί Menu που όταν το πατήσουμε μας δίνονται δυο εντολές όπως βλέπουμε στην εικόνα 60. Εμείς επιλέγουμε το πρώτο, το Read. Στη συνέχεια, εμφανίζεται η φόρμα Educational Game που έχει τις ερωτήσεις με τις πιθανές απαντήσεις και από κάτω το text field όπου γράφουμε την απάντηση. Στην εικόνα 62 απαντάμε στις ερωτήσεις και μόλις ολοκληρώσουμε, επιλέγουμε από το Menu την εντολή Load όπως βλέπουμε στην εικόνα 63. Στην εικόνα 64 παρατηρούμε ότι εμφανίζεται νέα φόρμα «Καταχώρηση Απαντήσεων». Βλέπουμε τις απαντήσεις που έχουμε δώσει και ακριβώς παρακάτω βλέπουμε αν η απάντηση που δώσαμε είναι σωστή ή λάθος. Αυτό φαίνεται καλύτερα στην εικόνα 65. Επιπλέον, στην τελευταία σειρά βλέπουμε και το αποτέλεσμα μας. Είχαμε τρεις λάθος απαντήσεις με αποτέλεσμα να έχουμε 70% επιτυχία. Μετά, πατάμε το κουμπί Back και γυρίζουμε

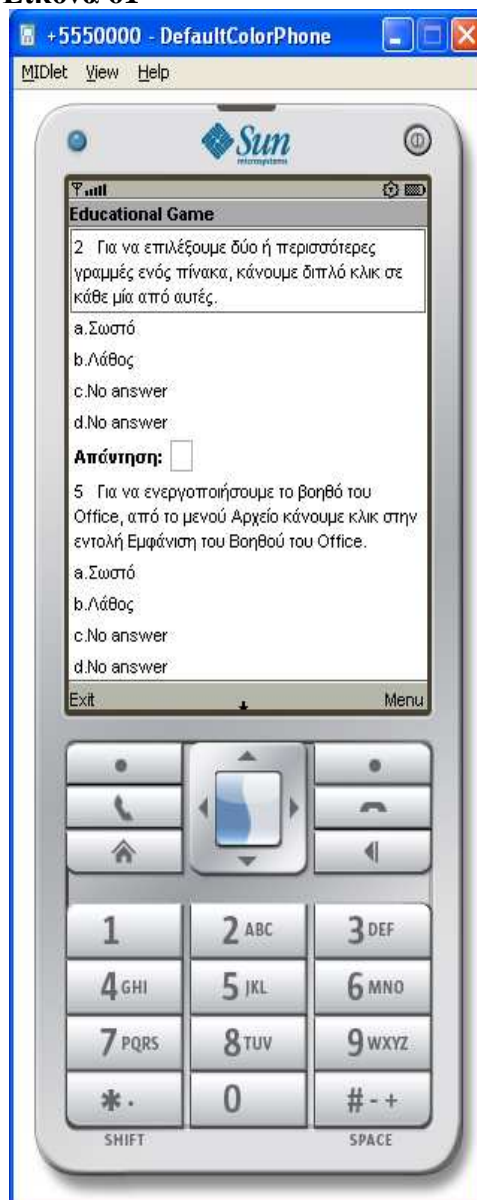
πίσω στην προηγούμενη φόρμα με τις ερωτήσεις όπως βλέπουμε στην εικόνα 66. Τέλος, πατάμε το κουμπί Exit για να βγούμε από την εφαρμογή (Εικόνα 67).

**Εικόνα 58****Εικόνα 59**

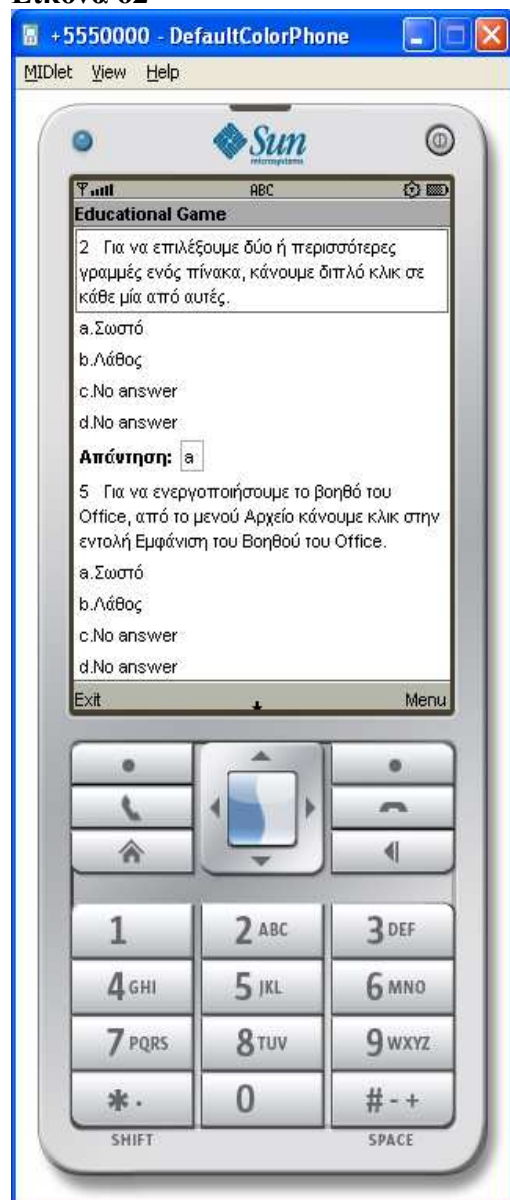
**Εικόνα 60**



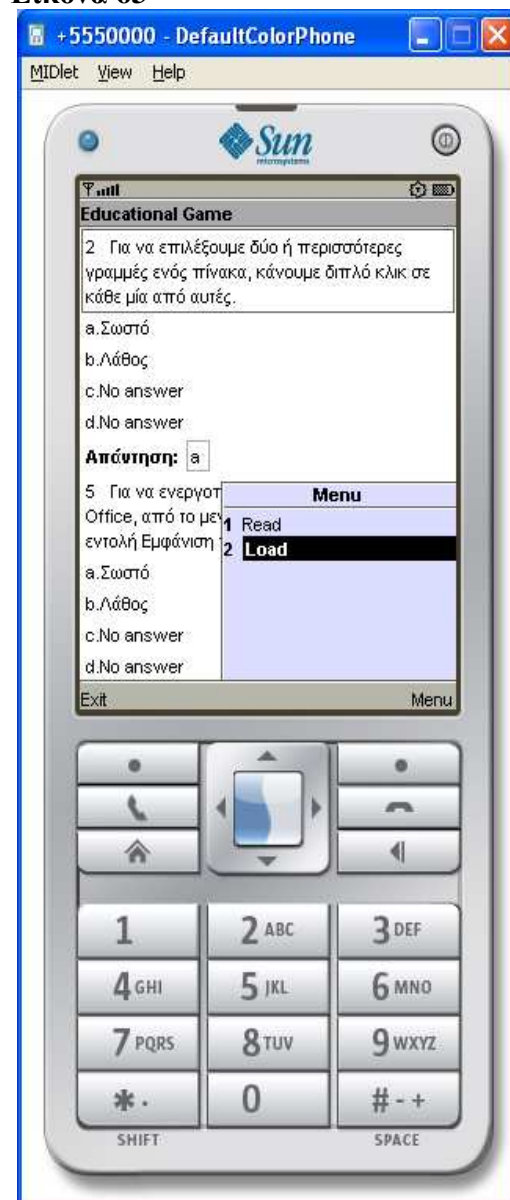
**Εικόνα 61**



**Εικόνα 62**



**Εικόνα 63**



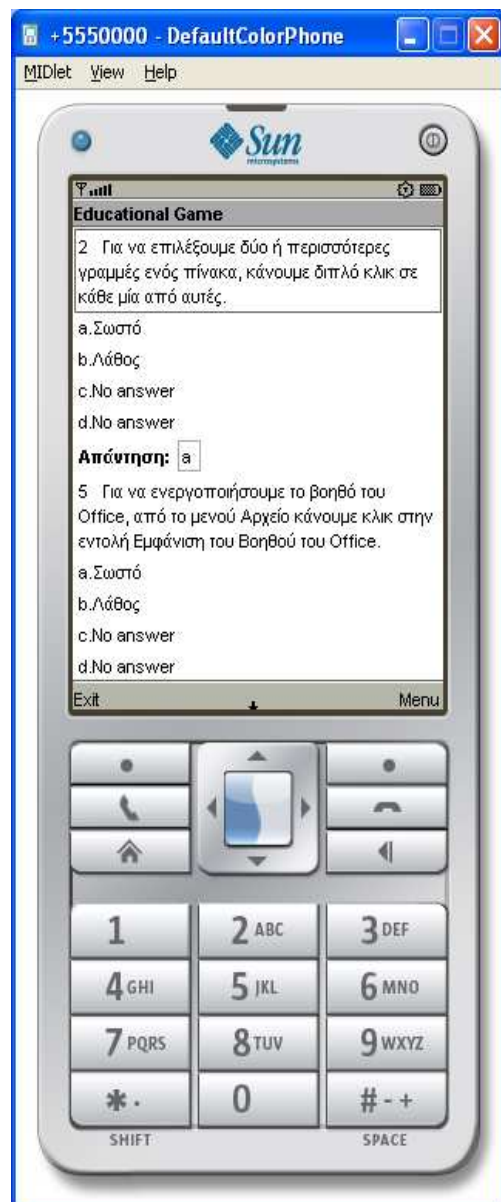
**Εικόνα 64**



**Εικόνα 65**



**Εικόνα 66**



**Εικόνα 67**



### 3.2.8 Σύνδεση NOKIA 5300 με τον υπολογιστή μέσω Bluetooth και εμφάνιση των αποτελεσμάτων χρησιμοποιώντας τον εξομοιωτή (emulator) της NOKIA

Έχοντας κάνει δοκιμή την εφαρμογή τώρα είμαστε έτοιμοι να μεταφέρουμε την εφαρμογή μας στο κινητό μας τηλέφωνο. Θα κάνουμε παρουσίαση σε ένα από τα δύο κινητά τηλέφωνα που έχουμε στη διάθεση μας. Επιλέξαμε να περιγράψουμε την διαδικασία με το κινητό της NOKIA. Ισχύει η ίδια διαδικασία για το κινητό της SONY ERICSSON.

Καταρχήν, ενεργοποιούμε το Bluetooth στο κινητό μας τηλέφωνο. Από το Menu του κινητού μας επιλέγουμε

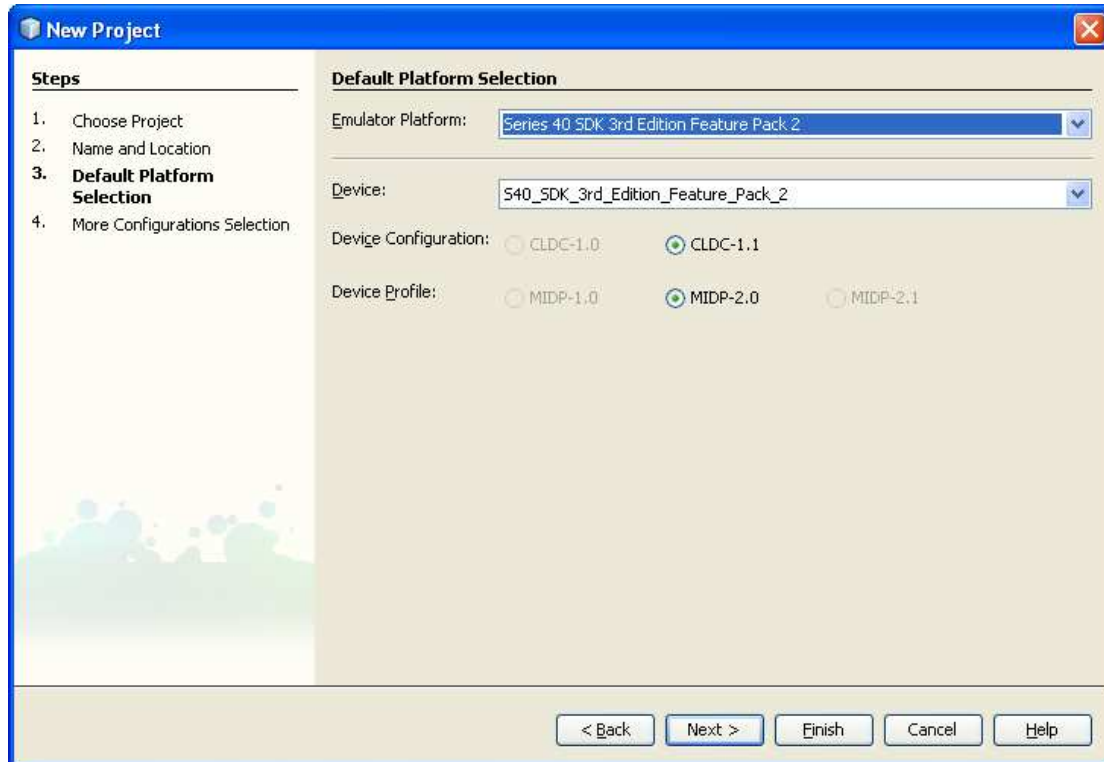
Ρυθμίσεις -> Συνδεσιμότητα -> Bluetooth και το ενεργοποιούμε

Μετά, ενεργοποιούμε το Bluetooth στον υπολογιστή μας. Κάνουμε δεξί κλικ στο εικονίδιο του Bluetooth και επιλέγουμε Start the Bluetooth Device.

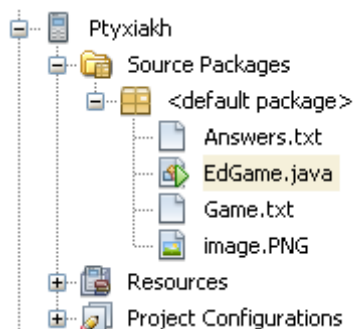
Στη συνέχεια, πηγαίνουμε στο πρόγραμμα μας DBConnection και «τρέχουμε» την εφαρμογή. Παίρνουμε τα δύο αρχεία που χρειαζόμαστε, Game.txt και Answers.txt.

Μετά, φτιάχνουμε ένα νέο midlet αρχείο με το όνομα Ptychiakh και στο νέο project αλλάζουμε μόνο την παρακάτω καρτέλα που αφορά την πλατφόρμα. Τώρα θα επιλέξουμε την πλατφόρμα της NOKIA.



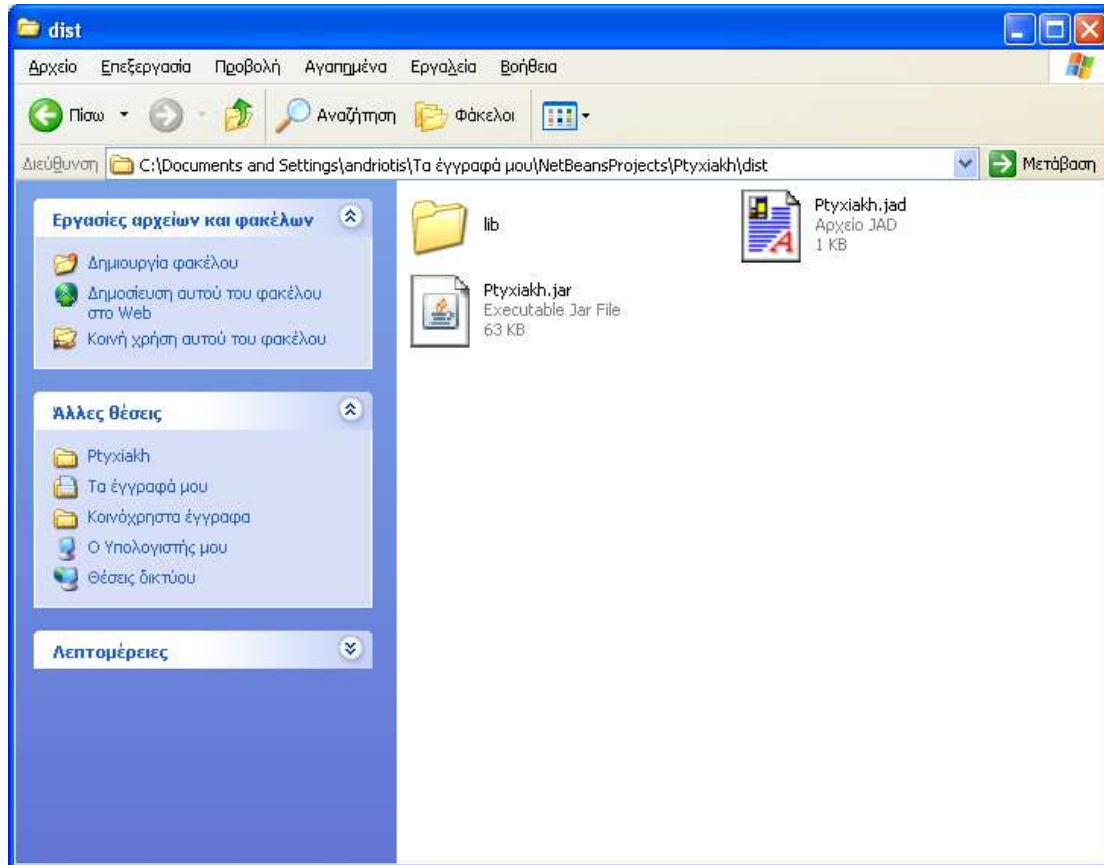
**Εικόνα 68**

Στη συνέχεια, φτιάχνουμε το midlet αρχείο. Ο κώδικας παραμένει ο ίδιος. Δεν αλλάζουμε τίποτα. Τα αρχεία Game.txt και Answers.txt, τα μεταφέρουμε στο Source Packages του midlet αρχείου Ptyxiakh. Επίσης, προσθέτουμε στο Source Package την εικόνα image.PNG.

**Εικόνα 69**

Κάνουμε compile και είναι επιτυχής. Στη συνέχεια, κάνουμε Build και επιλέγουμε Build Main Project για να δημιουργήσουμε τα αρχεία JAR και JAD. (Άλλος τρόπος είναι να κάνουμε δεξί κλικ στο project Ptyxiakh και επιλέγουμε Build.)

Στο φάκελο που αποθηκεύουμε το project, στο φάκελο dist βλέπουμε τα αρχεία JAR και JAD που δημιουργήθηκαν. Βλέπε εικόνα 58.

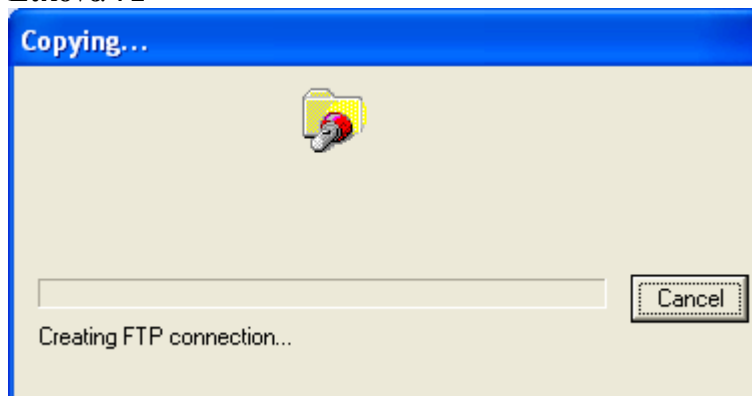
**Εικόνα 70**

Πάνω στο αρχείο Ptychiakh.jar κάνουμε δεξί κλικ και επιλέγουμε

Αποστολή προς -> Bluetooth -> NOKIA 5300

Στη συνέχεια, εμφανίζεται το παρακάτω παράθυρο που δείχνει ότι δημιουργεί μια FTP σύνδεση μεταξύ υπολογιστή και κινητού τηλεφώνου και συγχρόνως στην οθόνη του κινητού τηλεφώνου εμφανίζεται το παρακάτω μήνυμα.

«Ανταλλαγή δεδομένων με; XAROULA» και εμείς επιλέγουμε: «Αποδοχή». (Όπου XAROULA το όνομα του υπολογιστή!)

**Εικόνα 71**

Το αρχείο μεταφέρεται από τον υπολογιστή στο κινητό τηλέφωνο.

Η σύνδεση είναι επιτυχής και το αρχείο έχει μεταφερθεί στο κινητό μας τηλέφωνο!

### **«Τρέχοντας» την εφαρμογή - Περιγραφή**

Σε αυτή την περίπτωση χρησιμοποιούμε τον εξομοιωτή (emulator) του κινητού NOKIA 5300. Βλέπουμε πως θα «τρέχει» η εφαρμογή σε ένα πραγματικό κινητό τηλέφωνο. Στην εικόνα 72 έχουμε «φορτώσει» την εφαρμογή. Η τοποθέτηση των κουμπιών αλλάζει ανάλογα με το κινητό. Εδώ όταν πατήσουμε το κουμπί Options επιλέγουμε Read πατώντας το κουμπί Select (Εικόνα 73). Στην Εικόνα 74 βλέπουμε τις 10 τυχαίες ερωτήσεις και στην Εικόνα 75 συμπληρώνουμε τα text fields. Στις Εικόνες 76,77,78 και 79 βλέπουμε την δεύτερη φόρμα με τις απαντήσεις που έχουμε δώσει εμείς και παρακάτω βλέπουμε ποια απάντηση είναι σωστή και ποια λάθος. Στο τέλος, βλέπουμε και το αποτέλεσμα μας. Πατώντας το κουμπί Back επιστρέφουμε πίσω στην προηγούμενη φόρμα (Εικόνα 80) και πατώντας το κουμπί Exit βγαίνουμε από την εφαρμογή (Εικόνα 81).

**Εικόνα 72**



**Εικόνα 73**



**Εικόνα 74**



**Εικόνα 75**



**Εικόνα 76**



**Εικόνα 77**



**Εικόνα 78**



**Εικόνα 79**



**Εικόνα 80**



**Εικόνα 81**





## Βιβλιογραφία για το 2<sup>ο</sup> Μέρος

- [1] Βιβλίο: Beginning J2ME From Novice to Professional  
Συγγραφείς: Jonathank Knudsen and Sing Li  
Apress, Third Edition
- [2] Βιβλίο: Πλήρες Εγχειρίδιο της JAVA 6  
Συγγραφείς: Rogers Cadenhead, Laura Lemay  
Πέμπτη Έκδοση – Εκδόσεις: Μ. Γκιούρδας
- [3] Introduction to using Java J2ME , Bob Dickerson, October 2005  
<http://homepages.feis.herts.ac.uk/~comqrgd/docs/java-j2me/j2me.pdf>
- [4] 11. Wireless Java with J2ME  
<http://hamilton.bell.ac.uk/swdev3and4/lecturenotes/j2me.pdf>
- [5] [http://www.forum.nokia.com/Devices/Device\\_specifications/5300\\_XpressMusic/](http://www.forum.nokia.com/Devices/Device_specifications/5300_XpressMusic/)
- [6] [http://wiki.forum.nokia.com/index.php/CS001006\\_-\\_Reading\\_a\\_text\\_file\\_line\\_by\\_line](http://wiki.forum.nokia.com/index.php/CS001006_-_Reading_a_text_file_line_by_line)
- [7] [http://www.forum.nokia.com/Develop/Java/Getting\\_started/](http://www.forum.nokia.com/Develop/Java/Getting_started/)
- [8] [http://www.forum.nokia.com/info/sw.nokia.com/id/cc48f9a1-f5cf-447b-bdba-c4d41b3d05ce/Series\\_40\\_Platform\\_SDKs.html](http://www.forum.nokia.com/info/sw.nokia.com/id/cc48f9a1-f5cf-447b-bdba-c4d41b3d05ce/Series_40_Platform_SDKs.html)
- [9] <http://developer.sonyericsson.com/wportal/devworld/article/java-sdk-versionhistory>
- [10] <http://developer.sonyericsson.com/wportal/devworld/downloads/download/dw-102090-semcjavameclcdsdk2504?cc=gb&lc=en>
- [11] [http://wiki.forum.nokia.com/index.php/Series\\_40:\\_Editions\\_and\\_Feature\\_Packs#Series\\_40\\_3rd\\_Edition\\_Feature\\_Pack\\_2](http://wiki.forum.nokia.com/index.php/Series_40:_Editions_and_Feature_Packs#Series_40_3rd_Edition_Feature_Pack_2)
- [12] [http://www.forum.nokia.com/Devices/Series\\_40/](http://www.forum.nokia.com/Devices/Series_40/)
- [13] <http://www.forum.nokia.com/Develop/Java/Tools/>
- [14] NOKIA, Series 40 Software Developer Kit, 3<sup>rd</sup> Edition Feature Pack 2  
Installation and Configuration Guide, December 2006
- [15] <http://download.oracle.com/javame/config/cldc/ref-impl/midp1.0/jsr037/javax/microedition/midlet/package-summary.html>
- [16] [http://ants.dif.um.es/~felixgm/docencia/j2me/javadoc/javax/microedition/lcdui/package-summary.html#package\\_description](http://ants.dif.um.es/~felixgm/docencia/j2me/javadoc/javax/microedition/lcdui/package-summary.html#package_description)
- [17] [http://www.forum.nokia.com/Develop/Java/Getting\\_started/](http://www.forum.nokia.com/Develop/Java/Getting_started/)

[18] <http://www.forum.nokia.com/Develop/Java/>

[19] <http://web.mac.com/wheresjbob/iWeb/wheresjbob.com/Codelog/935F766B-4D9F-4B84-9E2A-E88B3D48D381.html>

# Κεφάλαιο 4<sup>ο</sup>

## 4.1 Σύνοψη

Η εργασία μας αυτή αναφέρεται στον σχεδιασμό και στην ανάπτυξη μιας εφαρμογής της συνεργατικής μάθησης μέσω κινητών συσκευών (m-learning). Δείχνει αφενός μεν την πολυπλοκότητα αλλά και αφετέρου την απλή χρήση της από μαθητές και καθηγητές. Η πολυπλοκότητα της ανάπτυξης της εφαρμογής που αναπτύξαμε δείχνει ακόμη την διεπιστημονική της βάση, αφού συνδυάζει γνώσεις πολλών πεδίων και απαιτεί την συνεργασία μιας ομάδας ερευνητών με εξειδικεύσεις σε σύγχρονα θέματα τεχνολογίας, επιστήμης και διδακτικής.

Θεωρώντας δεδομένη την ανάπτυξη του hardware των κινητών τηλεφώνων, η ανάπτυξη της συνεργατικής μάθησης μέσω κινητών συσκευών βασίζεται στον έξυπνο και αποτελεσματικό συνδυασμό γνώσεων που αφορούν:

1. Τις δυνατότητες της προσφερόμενης τεχνολογίας κινητών τηλεφώνων.
2. Βασικές γνώσεις από τα μαθήματα υπολογιστών (π.χ. βάσεις δεδομένων).
3. Το υπάρχον λογισμικό και την δυνατότητα ελεύθερης απόκτησής του.
4. Ειδικές γνώσεις του μαθήματος που απαιτεί η εφαρμογή (για την δημιουργία των απαιτούμενων βάσεων δεδομένων – ερωτήσεις και απαντήσεις).
5. Εξοικείωση με την υλοποίηση τέτοιου τρόπου διδασκαλίας.

Το m-learning στην κατεύθυνση αυτή, μπορεί να δανειστεί και να κάνει χρήση πολλών εφαρμογών που έχουν αναπτυχθεί με την κλασική διδασκαλία μέσω ηλεκτρονικού υπολογιστή (e-learning). Ίσως επί του παρόντος ο συνδυασμός του m-learning με το e-learning να είναι απαραίτητος. Φαίνεται όμως ότι η ανάπτυξη της τεχνολογίας να εξασφαλίζει την δυναμική στο m-learning για τον εκτοπισμό του e-learning.

## 4.2 m-learning versus e-learning

Θα μπορούσαμε να χαρακτηρίσουμε την συνεργατική μάθηση μέσω κινητών τηλεφώνων (m-learning) σαν την φυσική εξέλιξη της από αρκετό καιρό επιχειρούμενης διδασκαλίας μέσω του κλασικού ηλεκτρονικού υπολογιστή (e-learning), που βασιζόταν (και εξακολουθεί να βασίζεται) στην αρχή «κάθε μαθητής και υπολογιστής». Γίνεται όμως αμέσως φανερό από μια γρήγορη σύγκριση ότι το κινητό τηλέφωνο στη σημερινή του μορφή παρέχει μία εναλλακτική λύση στην προοπτική της διδασκαλίας μέσω υπολογιστή η οποία έχει σημαντικά πλεονεκτήματα. Ειδικότερα η συνεργατική μάθηση μέσω κινητών συσκευών:

1 Αναβαθμίζει την **στατική** μέθοδο διδασκαλίας με τον κλασικό υπολογιστή (e-learning) που θέλει τους μαθητές καθηλωμένους μπροστά στην οθόνη του υπολογιστή σε αίθουσες κατάλληλα διαμορφωμένες για τον σκοπό αυτό. Το κινητό τηλέφωνο αναβαθμίζει τη μέθοδο αυτή της διδασκαλίας σε **δυναμική**, που μπορεί να εφαρμοστεί σε οποιοδήποτε μέρος, στην κλασική αίθουσα διδασκαλίας, στο εργαστήριο, στην εκπαιδευτική εκδρομή, στο σπίτι, κλπ.

2. Ελαχιστοποιεί ή και μηδενίζει το κόστος της διδασκαλίας με τον υπολογιστή, μια και όλοι οι μαθητές είναι εφοδιασμένοι με κινητό τηλέφωνο με αποτέλεσμα να υπάρχει εξοικονόμηση αιθουσών διδασκαλίας αλλά και οικονομικών πόρων που απαιτεί η διδασκαλία με την χρήση των κλασικών υπολογιστών. Με κάποιο τρόπο, το κόστος της διδασκαλίας με τον υπολογιστή μεταφέρεται σιωπηλά στη τσέπη των γονέων που έτσι κι αλλιώς δεν φαίνεται να αρνούνται τον εφοδιασμό των παιδιών τους με κινητό τηλέφωνο.

3. Μειώνει κατά πολύ τις απαιτήσεις σε ειδικευμένο διδακτικό προσωπικό για την διδασκαλία με υπολογιστή αφού πια τώρα έχει γενικευθεί η χρήση του κινητού τηλεφώνου αλλά και γιατί διάφορες εφαρμογές της συνεργατικής μάθησης μέσω κινητών συσκευών θα μπορούσαν να εξασφαλίζονται από μια κεντρική διεύθυνση που θα φρόντιζε για την διανομή τους σ' όλα τα σχολεία. Τούτο θα μπορούσε να επιτευχθεί με την δημιουργία μιας κεντρικής «βιβλιοθήκης εφαρμογών» οι οποίες είτε θα εξασφαλιζόνταν από το ελεύθερο εμπόριο είτε από τις δημιουργίες ενός μόνιμου προσωπικού αφιερωμένου στην ανάπτυξη τέτοιων εφαρμογών. Σ' αυτή την περίπτωση, τα σχετιζόμενα Τμήματα των Ανωτάτων Εκπαιδευτικών Ιδρυμάτων θα είχαν πολλά να προσφέρουν.

### 4.3 Από το e-learning στο m-learning

Η παρούσα εργασία μας επιχειρεί ένα σημαντικό βήμα και θεωρούμε ότι βάζει ένα πετραδάκι στην ανάπτυξη του m-learning. Η εργασία μας αυτή μπορεί να ειπωθεί σαν ένας κρίκος στις προσπάθειες που έγιναν και θα γίνουν στην κατεύθυνση αυτή. Έχουμε ήδη δει τα αρχικές προσπάθειες της ανάπτυξης του m-learning. Ενδεικτικά, αναφέρουμε το podcasting μια αναβαθμισμένη έκδοση των ηχογραφημένων διαλέξεων που χρησιμοποίησαν πολλά Αμερικανικά πανεπιστήμια. Και όπως επιβεβαίωσε ένα πείραμα που έγινε στο πανεπιστήμιο του Sydney οι φοιτητές ένιωθαν ότι επωφελούνταν μαθησιακά από αυτόν τον τρόπο διδασκαλίας και βρήκαν θετικό το γεγονός ότι μπορούσαν να κάνουν download τα αρχεία στις κινητές τους συσκευές από οπουδήποτε και οποιαδήποτε ώρα.[2] Σαν άλλο πρόδρομο του m-learning αναφέρουμε την adaptive course caching strategy (ACCS) που προτάθηκε για να ξεπεραστεί το πρόβλημα της ανεπαρκούς κάλυψης της ασύρματης επικοινωνίας καθώς και της μικρής υπολογιστικής ισχύος και του μειωμένου διαδικτυακού bandwidth των προηγούμενων κινητών τηλεφώνων. Η ACCS βασίζεται στο SCORM Sequence & Navigation specification για ένα περιβάλλον που αφορά το mobile learning. Για την υλοποίηση αυτής της έρευνας χρησιμοποιήθηκε το Pocket SCORM και η έρευνα έδειξε ότι η ACCS μειώνει τις απαιτήσεις για επανασυνδέσεις Διαδικτύου [3].

### 4.4 Συμπεράσματα - Προοπτική

Η συνεργατική μάθηση μέσω κινητών συσκευών είναι ένας πολύ σημαντικός τομέας έρευνας και ανάπτυξης: μπορεί να προσφέρει νέες μορφές επικοινωνίας, συνεργασίας βρίσκεται ακόμη στην αρχή της εξέλιξης του. Αυτό είναι άλλωστε φυσικό, μια και οι κινητές συσκευές και τεχνολογίες διαδόθηκαν ευρέως (και σε οικονομική τιμή) μόλις τα τελευταία χρόνια. Κατά συνέπεια, δεν υπάρχουν ακόμη διαθέσιμες πολλές εμπορικές εκπαιδευτικές εφαρμογές.

Περιμένουμε μια ραγδαία εξέλιξη στα περιβάλλοντα συνεργατικής μάθησης μέσω κινητών συσκευών στα αμέσως επόμενα χρόνια, καθώς οι τεχνολογίες θα παρέχουν ολοένα και περισσότερο εύρος με λιγότερο κόστος. Η πλήρης αξιοποίηση των κινητών τεχνολογιών στη συνεργατική μάθηση δεν αναμένεται πάντως να υλοποιηθεί άμεσα. Ο λόγος είναι ότι η θεωρητική βάση πάνω στην οποία θα στηριχθεί αυτή η ανάπτυξη αναμένεται να χρειαστεί περισσότερο χρόνο για να ωριμάσει. Οι

περισσότερες «κλασικές» θεωρίες και μοντέλα μάθησης θεωρούσαν ως δεδομένο για αρκετές δεκαετίες την «παραδοσιακή τάξη»: ένας εκπαιδευτής βοηθά πολλούς εκπαιδευόμενους να αποκτήσουν συγκεκριμένη γνώση (μέσω μαθησιακού περιεχομένου, δραστηριοτήτων, κλπ), που μπορεί να αξιολογηθεί με βάση συγκεκριμένα μέσα. Όπως αναφέρουν οι Falk και Dierking, «ό,τι γνωρίζουμε μέχρι σήμερα αναφορικά με τη μάθηση προέρχεται από μελέτες, είτε σε αίθουσες διδασκαλίας, είτε σε εργαστήρια ψυχολογίας. Κατά συνέπεια, (η απομάκρυνση από την παραδοσιακή τάξη) μπορεί να μην αποτελεί την κατάλληλη βάση για τη μάθηση που πραγματοποιείται έξω από τα περιβάλλοντα αυτά». [1]

Η εργασία μας αυτή, δείχνει μια δυνατότητα εφαρμογής της συνεργατικής μάθησης μέσω κινητών συσκευών. Με την λεπτομερή περιγραφή που επιχειρήσαμε, γίνονται κατανοητές οι απαιτήσεις για την ανάπτυξη της μεθόδου αυτής. Και αυτές οι απαιτήσεις αφορούν αποκλειστικά τους σχεδιαστές και τους κατασκευαστές των εφαρμογών. Δεν αφορούν τον διδάσκοντα ή τον διδασκόμενο. Η λεπτομερής περιγραφή της εργασίας μας δίνει την δυνατότητα να εντοπιστούν τα σημεία εκείνα στα οποία μπορούν να γίνουν βελτιώσεις που όπως είναι φυσικό θα επιβάλλονται από την ταχεία βελτίωση και αναβάθμιση των κινητών τηλεφώνων από τη μια μεριά και από την ανάπτυξη πιο αναβαθμισμένου λογισμικού από την άλλη.

Αυτή η εργασία μπορεί να αποτελέσει τη βάση για να σχεδιαστούν και να αναπτυχθούν καινούργιες εφαρμογές με περισσότερα χαρακτηριστικά. Τα κινητά εξελίσσονται συνεχώς και τα κινητά της 4<sup>ης</sup> γενιάς μπορούν να ανταποκριθούν λόγω των δυνατοτήτων τους στις απαιτήσεις των εφαρμογών του m-learning.

## Βιβλιογραφία για το 4<sup>ο</sup> Κεφάλαιο

- [1]. <http://karagian.users.uth.gr/cscl/14-Karagiannidis-Vavoula.pdf>
- [2]. “Using short podcasts to reinforce lectures”  
Steve Clark, Lucy Taylor, Mark Westcott  
[http://sydney.edu.au/science/uniserve\\_science/pubs/procs/2007/08.pdf](http://sydney.edu.au/science/uniserve_science/pubs/procs/2007/08.pdf)
- [3]. “An adaptive Cashing Strategy for m-Learning Based on SCORM Sequencing and Navigation”  
Hsuan Pu Chang, Timothy k. Shih, Qing Li, Chun-Chia Wang, Te-Hua Wang, Louis R. Chao  
<http://www.springerlink.com/content/h1p3863223330808/>

*Τέλος*