



Τεχνολογικό εκπαιδευτικό Ίδρυμα Κρήτης
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών πληροφορικής



Πτυχιακή εργασία

Τίτλος: Εφαρμογή copy paste μέσω του IP

Πιπεράκη Παρασκευή (ΑΜ:2580)

Επιβλέπων καθηγητής: Μαλάμος Αθανάσιος

Επιτροπή αξιολόγησης: Μαλάμος Αθανάσιος,

Παναγιωτάκης Σπύρος

Καπετανάκης Κωνσταντίνος

Ημερομηνία παρουσίασης:21/03/2014

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη που μου παρείχαν ώστε να ολοκληρώσω τις σπουδές μου. Επίσης θα ήθελα να ευχαριστήσω τον Κ.Αθανάσιο Μαλάμο που μου έδωσε την δυνατότητα να εκπονήσω την πτυχιακή μου εργασία. Τέλος θα ήθελα να ευχαριστήσω τον Κώστα Καπετανάκη για την άμεση βοήθεια και την καθοδήγηση που μου παρείχε κατά την διάρκεια της πτυχιακής μου εργασίας.

Σύνοψη

Για αυτή τη πτυχιακή εργασία θα δημιουργηθεί ένας σύνθετος μηχανισμός web πλατφόρμας και stand-alone desktop application ο οποίος θα επικοινωνεί με λειτουργίες του συστήματος του χρήστη και με web-services θα γίνεται μεταφορά δεδομένων από χρήστη σε χρήστη. Μέσω της service θα γίνεται μεταφορά δεδομένων από κάποιο τερματικό, τα δεδομένα αυτά θα μεταφέρονται σε άλλα συνδεδεμένα τερματικά (μέσω της εφαρμογής). Πιο συγκεκριμένα μέσω της service θα κάνουμε copy από κάποιο pc αυτά θα μπαίνουν σε κάποιο Server για να είναι διαθέσιμα για paste στους άλλους υπολογιστές που έχουν συνάψει χειραψία, δηλαδή ο χρήστης που θα κάνει login ,θα έχει τη δυνατότητα να διαλέγει κάποιον υπολογιστή για να παίρνει τα copy στο δικό του σύστημα. Θα δημιουργούμε groups σε κάποια lan περιοχή για να κάνουμε τα copy paste με κάποιο συνδυασμό πλήκτρων control+c και control+v.

Λέξεις κλειδιά: Web service(Restful), Java, PHP

Abstract

For this diploma thesis, a complex mechanism of web platform and stand-alone desktop application will be created, which will communicate with operations of the user's system and will transfer data from user to user through web-services. Via this service, data transfer will occur from one terminal to other connected terminals (via this application). More specifically, via this service we will create a copy from a pc into a Server in order for it to be available for pasting to the other computers that have contracted handshake, As a result, the user who logs in, will be able to choose one computer in order to obtain the copied files in his own system. We will create groups in a lan domain in order to copy and paste the data with some combination of the keys control+c and control+v.

Keywords: Web service(Restful), Java, PHP

Πίνακας Περιεχομένων

1.Εισαγωγή.....	9
1.1περιληψη	9
1.2Κινητρο για τη διεξαγωγή της εργασίας	9
1.4Σκοπος και στόχοι της εργασίας.....	9
1.3Δομη της εργασίας.....	9
2. Απαιτήσεις.....	11
2.1Λειτουργικότητα1: Σύνδεση χρηστών με ip/port	11
2.2Λειτουργικότητα2: Χρήση ενός μενού για τις ενέργειες που προσφέρει η εφαρμογή ...	12
2.3Λειτουργικότητα3: Επικοινωνία των χρηστών για μεταφορά κειμένου με την χρήση ctrl+c και ctrl+v.Χρήση επιπλέον πλήκτρων.	13
2.4Λειτουργικότητα4: Δυνατότητα υλοποίησης και έλεγχο της εφαρμογής.....	13
Κεφάλαιο 3ο-Τεχνολογίες και εργαλεία ανάπτυξης πτυχιακής.....	14
3.1PHP	14
3.1.1 Γλώσσα PHP.....	14
3.1.2 Ιστορία της PHP	14
3.1.3 Σύνταξη της php	15
3.1.4 Τα αρχεία της PHP	15
3.1.5 Δύναμη της PHP.....	16
3.1.5 Αλλαγή HTML σε PHP	16
3.2.6 Σχόλια της PHP	16
3.2 Apache.....	17
3.3MySQL.....	17
3.4 SQL.....	18
3.4.1Τι είναι η SQL	18
3.4.2Δημιουργία πίνακα Βάσης Δεδομένων με SQL:	19
3.4.3Επιλογή δεδομένων από πίνακα.....	21
3.5 PhpMyAdmin.....	21
3.6 XAMPP	22
3.7 Html5	22
3.7.1 Τι είναι η <i>Html5</i>	22

3.7.2	Νέες δυνατότητες στην HTML5	23
3.7.3	CSS3	25
3.8	NetBeans IDE	25
3.9	Java	26
3.9.1	Χαρακτηριστικά της Java	27
3.9.2	Δημιουργία και εκτέλεση της Java.....	27
3.9.3	Εκτέλεση της Java με χρήση του NetBeans	28
3.9.4	Βιβλιοθήκες Κλάσεων της Java	29
3.9.5	Γραφικό Περιβάλλον Χρήστη(GUI-Grafical User Interface)	29
	Τα βασικά των γραφικών Διεπαφών:.....	29
3.9.6	GUI και Διασυνδέσεις ακρόασης Συμβάντων(Listeners).....	31
3.9.7	Treads.....	31
3.10	WebServices	32
3.10.1	RESTFull	32
	Τι είναι RESTFull υπηρεσίες.....	32
	Χαρακτηριστικά RESTFull	32
4.	Υλοποίηση.....	35
4.1	Δημιουργία βάσης Δεδομένων.....	35
4.2	Δημιουργία νέου project java κλάσεων.....	37
	Δημιουργία κλάσεων της εφαρμογής:	38
	Γραφικό περιβάλλον(GUI) στην κλάση Login	38
	Γραφικό περιβάλλον(GUI) στην κλάση Application	39
4.3	Κατασκευή WebService	42
	Δημιουργία σύνδεσης jdbc driver	43
	Δημιουργία server web service	43
	Δημιουργία Restful Java Client	47
4.4	Δημιουργία Web Interface(ιστοσελίδας)	47
5.	Ολοκλήρωση έργου	59
	Συμπεράσματα.....	63
	Παραρτήματα	64
	Παραρτημα1:Εγκατάσταση του Xampp	64
	Παραρτημα2: Εγκατάσταση Net beans.....	66
	BIBΛΙΟΓΡΑΦΙΑ	67
	1.Βιβλία	67

2. Διευθύνσεις url.....	67
-------------------------	----

Πίνακας Εικόνων

Εικόνα 1: Λειτουργικότητα1	12
Εικόνα 2: Λειτουργικότητα2	12
Εικόνα 3: Λειτουργικότητα2_ Διαχειριση πινάκων βάσης Δεδομένων.....	12
Εικόνα 4: Λειτουργικότητα2	13
Εικόνα 5:Συνδεση php με τη MySql.....	18
Εικόνα 6: Περιβάλλον phpMyAdmin	22
Εικόνα 7:Xampp.....	22
Εικόνα 8:Html5 to html	23
Εικόνα 9: Web σελίδα χωρίς τη χρήση css	25
Εικόνα 10: Web σελίδα με τη χρήση των css.....	25
Εικόνα 11: JavaApplication με χρήση NetBeans	26
Εικόνα 12:Εκτέλεση Java	27
Εικόνα 13:Example Java Application.....	29
Εικόνα 14: Ιεραρχία συστατικών της java	30
Εικόνα 15:Frame	30
Εικόνα 16:Listeners του πακέτου java.awt.event	31
Εικόνα 17:Statefull design	33
Εικόνα 18:Stateless Design	33
Εικόνα 19:RESTFul example σε JAVA μέσω NetBeans	34
Εικόνα 20:Δημιουργία Βάση δεδομένων.....	36
Εικόνα 21: Πίνακες στην βάση δεδομένων (appbase).....	37
Εικόνα 22: Δημιουργία νέου project βημα1.....	37
Εικόνα 23: Δημιουργία νέου project βημα2.....	37
Εικόνα 24: Δημιουργία κλάσης Application.....	38
Εικόνα 25: Login form.....	39
Εικόνα 26: Create.php σελίδα	39
Εικόνα 27:GUI	40
Εικόνα 28:Listeners	40
Εικόνα 29:Items in menubar	40
Εικόνα 30: Σχήμα επικοινωνίας(apache server, Mysql, Apache Tomcat Rest webServices).....	42
Εικόνα 31: Connection with jdbc	43
Εικόνα 32: Δημιουργία web application βήμα 1	43
Εικόνα 33: Δημιουργία web application βήμα 2	44
Εικόνα 34: Δημιουργία web application βήμα 3	44
Εικόνα 35: Restful web Services from patterns.....	45
Εικόνα 36 :Login service	45
Εικόνα 37: Μέθοδος Get της κλάσης Login	46
Εικόνα 38: Μέθοδος Get της κλάσης Logout	46
Εικόνα 39 :RESTFull Java Client.....	47
Εικόνα 40: Web Interface	47
Εικόνα 41: login.html σελίδα	51
Εικόνα 42: crud_processing.html σελίδα.....	52

Εικόνα 43 :Σελίδα crud_processing.php.....	52
Εικόνα 44: Αποτελέσματα crud_processing.php.....	58
Εικόνα 45: Login Form(Java).....	59
Εικόνα 46: Λάθος ip/port.....	60
Εικόνα 47: Λάθος όνομα/κωδικός.....	60
Εικόνα 48:Window χρήστη.....	60
Εικόνα 49: Εγκατάσταση Xampp βημα1.....	64
Εικόνα 50: Εγκατάσταση Xampp βημα2.....	64
Εικόνα 51: Εγκατάσταση Xampp βημα3.....	65
Εικόνα 52: Εγκατάσταση Xampp βημα4.....	65
Εικόνα 53: Εγκατάσταση Xampp βημα5.....	66
Εικόνα 54: Ενεργοποίηση Xampp.....	66
Εικόνα 55: Εγκατάσταση του NetBeans.....	66

Λίστα Πινάκων

Πίνακας 1: Βάσεις δεδομένων που υποστηρίζει η PHP.....	16
Πίνακας 2:users table.....	36
Πίνακας 3:copied_values table.....	36
Πίνακας 4: pasted_values table.....	36

1.Εισαγωγή

1.1περιληψη

Ο βασικός σκοπός της παρούσας πτυχιακής είναι η υλοποίηση μιας εφαρμογής μέσω της οποίας θα μπορεί μια κλειστή ομάδα χρηστών που ανήκουν στο ίδιο δίκτυο να μεταφέρουν απλό κείμενο με την χρήση των πλήκτρων $ctr+c$ (από ένα χρήστη) και $ctr+v$ (από τον άλλο χρήστη του δικτύου)

Η εφαρμογή δίνει τη δυνατότητα στους χρηστές να κάνουν αντιγραφή κειμένου μεταξύ τους με μια κίνηση ,δηλαδή με το πάτημα ενός πλήκτρου, να διατηρούν ιστορικό για τις αντιγραφές που έχουν γίνει και να μπορούν να τις πάρουν στο σύστημα τους, και τέλος να ενημερώνονται για την αντιγραφή που έχει υλοποιηθεί από τον κάθε χρήστη.

Ακόμα υπάρχει δυνατότητα διαχείρισης χρηστών (δημιουργία/διαγραφή του χρήστη και διαγραφή του ιστορικού.)

1.2Κινητρο για τη διεξαγωγή της εργασίας

Η τεχνολογία και το διαδίκτυο είναι ένα μεγάλο κομμάτι της ζωής μας το οποίο μπορεί να μας προσφέρει ιδέες για τη δημιουργία νέων εφαρμογών σύμφωνα με τις ήδη υπάρχουσες τεχνολογίες που υπάρχουν.

Έτσι θα ήταν πάρα πολύ χρήσιμο σε μια μικρή ομάδα εργαστηρίου να υπάρχει μια εφαρμογή στην οποία ένας μικρός αριθμός χρηστών θα μπορεί να μεταφέρει κείμενο μεταξύ τους με τη χρήση ενός πλήκτρου.

1.4Σκοπος και στόχοι της εργασίας

Στόχος της παρούσας εργασίας είναι η δημιουργία μιας εφαρμογής για την μεταφορά κειμένου σε ομάδες χρηστών.

Έχει σχεδιαστεί και δομηθεί έτσι ώστε η μεταφορά των δεδομένων να είναι πιο εύκολη και γρήγορη στην υλοποίηση της για κάθε χρήστη.

Σκοπός της συγκεκριμένης αναφοράς είναι να μάθουμε και να κατανοήσουμε τη λειτουργία και τα πλεονεκτήματα τα οποία προσφέρουν τα web services μέσω διαδικτύου και τη χρήση που μπορεί να προσφέρει η αντικειμενοστραφής γλωσσά προγραμματισμού(Java). Επίσης γίνεται ενημέρωση για όλες τις τεχνολογίες οι οποίες χρησιμοποιήθηκαν για να υλοποιηθεί η εφαρμογή.

1.3Δομη της εργασίας

Η συνέχεια της εργασίας είναι οργανωμένη ως εξής έγινε :

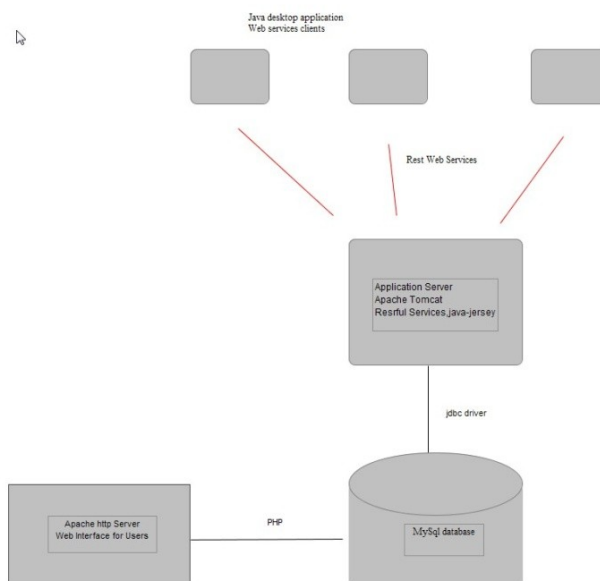
- Κεφάλαιο 2: Παρουσιάζονται οι απαιτήσεις και οι στόχοι της εφαρμογής
- Κεφάλαιο 3: Παρουσιάζονται οι κύριες τεχνολογίες και προγράμματα που χρησιμοποιήθηκαν.

- Κεφάλαιο 4: Παρουσιάζεται η υλοποίηση της εφαρμογής
- Κεφάλαιο 5: Παρουσιάζεται η ολοκλήρωση του έργου
- Τέλος παρουσιάζονται τα συμπεράσματα και οι μελλοντικές προοπτικές της εφαρμογής

2. Απαιτήσεις

Σε αυτό το κεφάλαιο θα αναφερθούν οι μέθοδοι ανάπτυξης και ο στόχος της εφαρμογής.

Για την υλοποίηση της εφαρμογής αυτής χρησιμοποιούνται οι τεχνολογίες RESTful services, Java και Php. Στο παρακάτω σχήμα φαίνεται η σύνδεση μεταξύ των τεχνολογιών αυτών ώστε να ολοκληρωθεί η μεταφορά και επικοινωνία των χρηστών.



Με βάση το σχήμα βλέπουμε ότι ο Apache http server μέσω της PHP συνδέεται με την βάση Δεδομένων. Με την σειρά της η βάση δεδομένων συνδέεται με τον server (Apache Tomcat) java-jersey με τον jdbc driver. Και αυτός με την σειρά του μέσω των Restful υπηρεσιών στέλνει τα δεδομένα στους clients. Με βάση αυτή την αρχιτεκτονική σκοπός είναι ο χρήστης που θα κάνει login να έχει τη δυνατότητα να διαλέγει κάποιον υπολογιστή για να παίρνει τα copy στο δικό του σύστημα ώστε να γίνεται η μεταφορά κειμένου από user σε user. Έτσι θα υπάρχει μια ομάδα χρηστών σε κάποια Lan περιοχή για να εκτελούνται τα copy paste με κάποιο συνδυασμό πλήκτρων control+c και control+v.

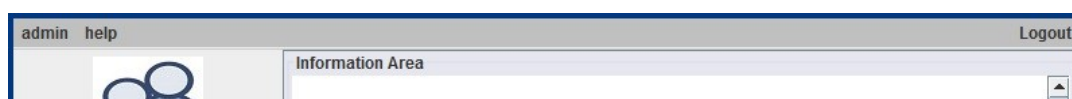
2.1 Λειτουργικότητα 1: Σύνδεση χρηστών με ip/port

Αρχικά σκοπός της εφαρμογής είναι να γίνεται σύνδεση χρηστών με τα στοιχεία τους και με συγκεκριμένη ip/port (Την ip/port του server). Έχοντας συνδεθεί στον server θα γίνεται σύνδεση της εφαρμογής και θα υπάρχουν οι συνδεδεμένοι χρήστες που έχουν κάνει εγγραφή.

Εικόνα 1: Λειτουργικότητα1

2.2 Λειτουργικότητα2: Χρήση ενός μενού για τις ενέργειες που προσφέρει η εφαρμογή

Όταν γίνει σύνδεση των χρηστών θα υπάρχει ένα μενού που θα μπορούν να το χρησιμοποιήσουν ανάλογα με το τι θέλουν να κάνουν.



Εικόνα 2: Λειτουργικότητα2

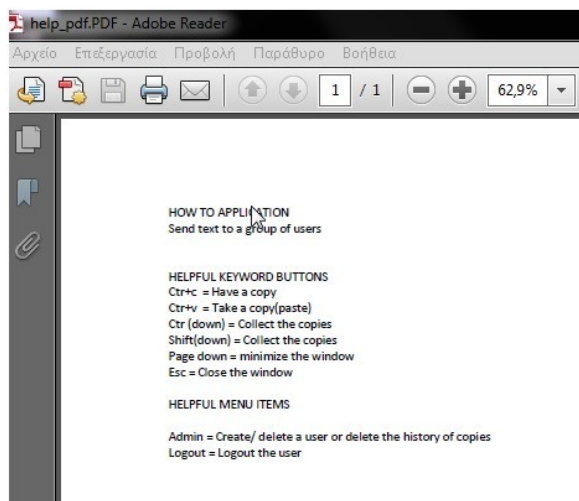
Με το κουμπί admin θα πρέπει να έχουν τη δυνατότητα να διαχειρίζονται τα copy και τους χρήστες που υπάρχουν στην εφαρμογή(Εικόνα 3)

username	Action
maria	Delete
giorgos	Delete
kostas	Delete
voula	Delete

User	Copy	Date	Action
voula	Welcome to application	2014-03-11 23:06:40	Delete

Εικόνα 3: Λειτουργικότητα2_ Διαχειριση πινάκων βάσης Δεδομένων

Στη συνέχεια θα μπορούν να ενημερώνονται με ένα pdf_file για την λειτουργία της εφαρμογής.



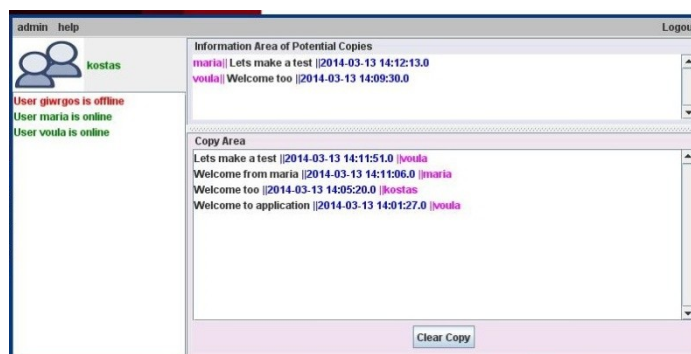
Τέλος με το κουμπί Logout θα μπορούν να κάνουν αποσύνδεση της εφαρμογής

2.3 Λειτουργικότητα3: Επικοινωνία των χρηστών για μεταφορά κειμένου με την χρήση ctrl+c και ctrl+v. Χρήση επιπλέον πλήκτρων.

Σε δεύτερο χρόνο οι χρήστες θα μπορούν μέσω τον πλήκτρων ctrl+c να κάνουν ένα copy, ctrl+v να παίρνουν το copy απευθείας, με ένα κλικ από το ποντίκι να παίρνουν τα προηγούμενα copy, ctr(down) να επιλέγουν πολλά copy, shift(down) να επιλέγουν μια ροη από copies, esc να αποσυνδέονται από την εφαρμογή και τέλος με το page down να κάνουν ελαχιστοποίηση του παραθύρου της εφαρμογής

2.4 Λειτουργικότητα4: Δυνατότητα υλοποίησης και έλεγχο της εφαρμογής

Οι χρήστες που έχουν εγγραφεί στην εφαρμογή θα έχουν τη δυνατότητα να μπορούν να βλέπουν ποιοι χρήστες είναι online, ποιοι χρήστες έχουν υλοποίηση ένα copy, και ποιοι χρήστες έχουν κάνει κάποια πιθανά paste.



Εικόνα 4: Λειτουργικότητα2

Κεφάλαιο 3ο-Τεχνολογίες και εργαλεία ανάπτυξης πτυχιακής

Σε αυτό το κεφάλαιο αναφέρονται οι τεχνολογίες και τα προγράμματα που χρησιμοποιήθηκαν για την υλοποίηση της πτυχιακής εργασίας. Συγκεκριμένα αναφέρονται:

- Η τεχνολογία PHP
- Ο Εξυπηρετητής Apache
- Η MySql
- PhpMyAdmin
- XAMPP
- Html5
- Java
- RESTful Services

3.1 PHP

3.1.1 Γλώσσα PHP

Η PHP (Hyper PreProcessor) είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μπορεί να εγκατασταθεί σχεδόν σε όλα τα λειτουργικά συστήματα όπως windows, Linux, Mac OS X, κλπ. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακοσμητή του Παγκόσμιου Ιστού(π.χ. Apache), ώστε να προβεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML. Ο κώδικας PHP μπορεί να συνδεθεί σε απομακρυσμένους υπολογιστές, και να θέσει ερωτήματα σε βάσεις δεδομένων. Στην συγκεκριμένη πτυχιακή κάνουμε χρήση της γλώσσας PHP για να συνδεθούμε με την βάση δεδομένων και να κάνουμε διαχείριση πινάκων

Βασικά στοιχεία της PHP:

- Το πρόγραμμα ξεκινά και τελειώνει με `<?php` και `<?`
- Το ερωτηματικό στο τέλος της πρότασης δηλώνει τέλος της εντολής
- Κάθε αρχείο πρέπει να έχει κατάληξη `.php`

3.1.2 Ιστορία της PHP

Η PHP ξεκίνησε το 1994 από τον φοιτητή Rasmus Lerdorf ο οποίος δημιούργησε ένα απλό Script με όνομα `php.cgi`. Η γλώσσα τότε ονομαζόταν PHP/FI (Personal Home Page/Form Interpreter). Στη συνέχεια υλοποιήθηκε και από τους Andi Gutmans και Zeev Suraski, έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP.

3.1.3 Σύνταξη της php

Η γλώσσα PHP είναι μια γλώσσα συγγραφής σεναρίων (scripting language) που ενσωματώνεται μέσα στον κώδικα της HTML και εκτελείται στην πλευρά του server (server-side scripting).

Ανταγωνιστικές τεχνολογίες της PHP είναι οι γλώσσες προγραμματισμού *ASP (Active Server Pages)* της εταιρείας Microsoft, *CFML (ColdFusion Markup Language)* της εταιρείας Allaire και *JSP (JavaServer Pages)* της εταιρείας Sun.

Η σύνταξη της PHP μοιάζει με την σύνταξη της Perl και της C. Ο σκοπός της γλώσσας είναι να δώσει τη δυνατότητα στους web developers να δημιουργούν δυναμικά παραγόμενες ιστοσελίδες.

Παράδειγμα χρήσης PHP :

```
<html>
<head>
<title> Παράδειγμα </title>
</head>
<body>
<?php echo "Γλώσσα PHP ?>
</body>
</html>
```

Σε αντίθεση με την Perl ή η C, αντί να γράψουμε ένα πρόγραμμα με πολλές εντολές για να δημιουργήσουμε κώδικα HTML, γράφουμε ένα HTML script με κάποιον ενσωματωμένο κώδικα για να κάνει κάτι, όπως στην συγκεκριμένη περίπτωση να εμφανίσει κάποιο κείμενο (μήνυμα). Ο κώδικας της PHP περικλείεται με ειδικά tags αρχής και τέλους για να μπορούμε να εισερχόμαστε και να εξερχόμαστε από το PHP mode.

Αυτό που κάνει την PHP να ξεχωρίζει από μια γλώσσα όπως η JavaScript, η οποία εκτελείται στην πλευρά του χρήστη (client-side), είναι ότι ο κώδικάς της εκτελείται στον server. Αν είχαμε σ' έναν server ένα script παρόμοιο με το παραπάνω, ο χρήστης (client) θα λάμβανε το αποτέλεσμα της εκτέλεσης αυτού του script, χωρίς να είναι σε θέση να γνωρίζει ποιος μπορεί να είναι ο αρχικός κώδικας.

Μπορούμε ακόμη να ρυθμίσουμε (configure) τον web server ώστε να επεξεργάζεται όλα τα HTML αρχεία με την PHP και τότε δεν θα υπάρχει πράγματι κανένας τρόπος να μάθουν οι χρήστες τον κώδικά μας

3.1.4 Τα αρχεία της PHP

Κάθε αρχείο της PHP έχει κατάληξη .php. Η γλώσσα PHP μπορεί να συνταθεί από ένα απλό επεξεργαστή κειμένου (NotePad ++), η από ειδικούς editors PHP οι όποιοι υπάρχουν στο

Διαδίκτυο. Για να γίνει η εκτέλεση των εντολών θα πρέπει να έχει εγκατασταθεί σε ένα εξυπηρετητή (Apache Tomcat, Microsoft IIS)

3.1.5 Δύναμη της PHP

Η PHP όπως και άλλα προγράμματα μπορεί να κάνει επεξεργασία δεδομένων μιας φόρμας , δημιουργία δυναμικού περιεχομένου ιστοσελίδων η αποστολή και λήψη cookies.

Το βασικό χαρακτηριστικό της PHP είναι η υποστήριξη που παρέχει σε μια ευρεία γκάμα από βάσεις δεδομένων. Στον παρακάτω πίνακα αναφέρονται οι βάσεις δεδομένων που μπορεί να υποστηρίξει.

Adabas D	dBase	Empress	FilePro	Informix	InterBase	mSQL
MySQL	Oracle	PostgreSQL	Solid	Sybase	Velocis	Unix dbm

Πίνακας 1: Βάσεις δεδομένων που υποστηρίζει η PHP

3.1.5 Αλλαγή HTML σε PHP

1.<? echo ("Είναι η απλούστερη \n"); ?>

2.<?php echo("Για XML έγγραφα \n"); ?>

3.<script language="php">

echo ("Σε μερικούς editors, όπως ο FrontPage, δεν αρέσουν οι εντολές επεξεργασίας");

</script>

4. <% echo (" "); %>

<%= \$variable; # Είναι μια συντόμευση για το "<?echo .." %>

3.2.6 Σχόλια της PHP

Η PHP χρησιμοποιεί τον ίδιο τρόπο σχολιασμού όπως η C, η C++ και το Unix shell. Για παράδειγμα :

```
<?php
```

```
echo "Σχολιο της php "; // Σχόλιο μίας γραμμής της C++
```

```
/* αυτό είναι ένα σχόλιο της C */
```

```
Echo "Σχόλιο της C";
```

```
Echo "Σχόλιο της shell"; # Σχόλιο της shell
```

```
?>
```

Τα σχόλια μίας γραμμής σχολιάζουν μέχρι το τέλος της γραμμής ή το τρέχον μπλοκ του PHP κώδικα, ανάλογα με το ποιο εμφανίζεται πρώτο.

<h1> Αυτό είναι ένα <?# Echo "απλό";?> παράδειγμα. </h1>

<p> Το header θα εμφανίσει το 'Αυτό είναι ένα παράδειγμα.' </p>

Πρέπει να είμαστε προσεκτικοί για να μην φωλιάζουμε (nest) τα σχόλια τύπου C.


```

<?php
/*
echo "Αυτό είναι ένα σχόλιο ";
/* Αυτό το σχόλιο δεν είναι σωστό */
*/
?>

```

3.2 Apache



Ο Apache HTTP ή Apache είναι ένας εξυπηρετητής παγκόσμιου ιστού (web) ο οποίος διατίθεται δωρεάν. Είναι προϊόν ενός έργου ανοιχτού κώδικα που διανέμει μια ποικιλία λογισμικών για το Διαδίκτυο σε οποιονδήποτε έχει τη δυνατότητα να τα μεταμορφώσει. Ο Apache μπορεί να εγκατασταθεί σε UNIX, LINUX και WINDOWS. Οι Διακοσμητές Ιστού Apache χρησιμοποιούνται συνήθως σε συνδυασμό με τη μηχανή βάσης δεδομένων MySQL, τη γλώσσα php και άλλες δημοφιλείς γλώσσες όπως Python και Perl.

3.3 MySQL

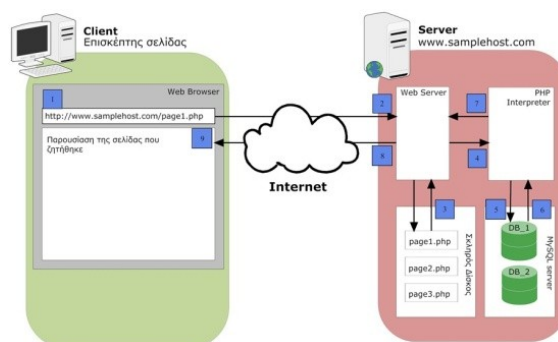


Η *MySQL* είναι ένα σύστημα σχεσιακών βάσεων δεδομένων (Relational database management system-RDBMS) και χρησιμοποιεί τη πιο διαδεδομένη γλώσσα ερωτημάτων την SQL. Το όνομα της το πήρε από την κόρη του Μόντυ Βιντένιους, την Μάι (My) και από την γλώσσα SQL (sql). Η Βάση Δεδομένων MySQL παρέχει υψηλή απόδοση, αξιοπιστία και εύκολη χρήση. Χρησιμοποιείται για τη δημιουργία διαδικτυακών χώρων με μεγάλο όγκο δεδομένων, κρίσιμων συστημάτων για τη λειτουργία εταιρικών εφαρμογών και πακέτων λογισμικού μεγάλων εταιριών. Βασίζεται στο LAMP (Linux, Apache, MySQL, PHP/Perl/Python) και λειτουργεί σε περισσότερες από 20 πλατφόρμες (Windows, Linux, OS/X, Netware και άλλες).

Με χρήση της γλώσσας PHP όπως αναφέρεται παραπάνω μπορούμε να συνδεθούμε σε κάποιο MySQL Server ώστε να πάρουμε τα δεδομένα από οποιαδήποτε βάση Δεδομένων, και να κάνουμε επεξεργασία δεδομένων (Εισαγωγή, ανανέωση και διαγραφή) σε πινάκες βάσεων.

Βήματα διαδικτυακής συναλλαγής Βάσεων δεδομένων:

- Ο web browser ενός χρήστη κάνει μια HTTP αίτηση για μια συγκεκριμένη διεύθυνση παγκόσμιου Ιστού
- Ο διακοσμητής(Apache Server) λαμβάνει την αίτηση για την σελίδα και την περνά στη μηχανή PHP προς επεξεργασία.
- Στη συνέχεια η μηχανή PHP μέσω μιας εντολής συνδέει τη βάση δεδομένων και εκτελεί το ερώτημα. Μέσω της PHP γίνεται σύνδεση με το MySQL διακοσμητή(server) .
- Ο MySQL διακοσμητής επεξεργάζεται το ερώτημα και στέλνει τα αποτελέσματα στη μηχανή PHP
- Η μηχανή PHP μετά την εκτέλεση των αποτελεσμάτων επιστρέφει στο διακοσμητή (Apache Server)
- Ο διακοσμητής(Apache) περνά ξανά την HTML στο φυλλομετρητή ώστε ο χρήστης να δει τα αποτελέσματα που ζήτησε.



Εικόνα 5:Συνδεση php με τη MySql

3.4 SQL

3.4.1 Τι είναι η SQL

Η SQL (Structured Query Language) χρησιμοποιείται για την δημιουργία, τη διαχείριση, την καταχώρηση δεδομένων και την άντληση δεδομένων από βάσεις δεδομένων και Σχεσιακές Βάσεις δεδομένων.

Για την καταχώρηση των δεδομένων που θέλουμε να διαχειριστούμε σε μια βάση δεδομένων είναι αναγκαίο η δημιουργία πινάκων. Οι βάσεις δεδομένων ονομάζονται σχεσιακές γιατί αποτελούνται από πολλούς πίνακες οι οποίοι συνδέονται μεταξύ τους.

Δημιουργία Βάσης Δεδομένων με SQL:

Η δημιουργία μιας βάσης δεδομένων με χρήση της SQL γίνεται με τον παρακάτω τρόπο→

```
CREATE DATABASE “όνομα_βάσης_δεδομένων”
```

Αν η βάση δεδομένων που θέλουμε να δημιουργήσουμε αφορά στην διαχείριση

Του ΤΕΙ πληροφορικής Ηρακλείου, η πιο αντιπροσωπευτική σύνταξη είναι η ακόλουθη:

```
CREATE DATABASE epp_teicrete
```

Διαγραφή Βάσης Δεδομένων με SQL:

Η ανάγκη διαγραφής της Βάσης δεδομένων όταν κάνουμε μια δοκιμαστική βάση και δεν μας είναι πλέον απαραίτητη εκτελούμε την παρακάτω εντολή→

```
DROP DATABASE “όνομα_βάσης_δεδομένων”
```

Αν θέλουμε να διαγράψουμε την Βάση Δεδομένων που δημιουργήσαμε προηγουμένως, εκτελούμε την εξής εντολή:

```
DROP DATABASE epp_teicrete
```

3.4.2 Δημιουργία πίνακα Βάσης Δεδομένων με SQL:

Για να μπορούμε να αποθηκεύσουμε τα δεδομένα της βάσης μας είναι ανάγκη η δημιουργία πινάκων. Κάθε πίνακας αποτελείται από επιμέρους στοιχεία που ονομάζονται πεδία. Στα πεδία αυτά αποθηκεύονται όλα τα δεδομένα που προσθέτουμε. Κάθε πεδίο, έχει ένα χαρακτηριστικό, το οποίο είναι ο τύπος του. Ανάλογα με τα δεδομένα που καταχωρούμε στο κάθε πεδίο, καθορίζουμε και τον τύπο του. Ο τύπος του πεδίου μπορεί να είναι χαρακτήρας (VARCHAR), ακέραιος αριθμός (INT), ημερομηνία (date).

Για να δημιουργήσουμε έναν πίνακα με την χρήση της γλώσσας SQL χρησιμοποιούμε την εντολή CREATE TABLE

Η σύνταξη της παραπάνω εντολής είναι η εξής:

```
CREATETABLE“όνομα_πίνακα”
```

```
(“πεδίο1” INT,
```

```
“πεδίο2” VARCHAR(20),
```

```
“πεδίο3” DATE)
```

Για παράδειγμα, θα μπορούσαμε να δημιουργήσουμε έναν πίνακα στον οποίον θα καταχωρούμε τους φοιτητές της σχολής TEI Πληροφορικής

```
CREATE TABLE students
```

```
(students_id INT,
```

```
name VARCHAR(20),
```

```
date_of_birth DATE)
```

Διαγραφή πίνακα Βάσης Δεδομένων με SQL:

Για να διαγράψουμε έναν πίνακα, χρησιμοποιούμε την εντολή DROP TABLE “όνομα_πίνακα”. Συγκεκριμένα η διαγραφή πίνακα φαίνεται παρακάτω

```
DROP TABLE students
```

Δημιουργία μη κενών πεδίων Βάσης Δεδομένων με SQL:

Μπορούμε να ορίσουμε να μην υπάρχουν κενά πεδία, δηλαδή ο χρήστης να πρέπει να εισάγει δεδομένα. Αυτό πραγματοποιείται με την εντολή NOT NULL

```
CREATE TABLE ‘όνομα πίνακα’
```

```
(“πεδίο1” INT NOT NULL,
```

πεδίο2” VARCHAR(20),
 πεδίο 3” DATE)

Στο παράδειγμα που χρησιμοποιούμε, το όνομα του φοιτητή θα πρέπει απαραίτητως να συμπληρώνεται, καθώς δεν μπορούμε να δημιουργήσουμε εγγραφή για κάποιον φοιτητή που δεν έχει όνομα.

```
CREATE TABLE students
(student_id INT,
name VARCHAR(20) NOT NULL,
date_of_birth DATE)
```

Ορισμός Πρωτεύοντος κλειδιού στον πίνακα

Σε κάθε πίνακα θα πρέπει να υπάρχει ένα πεδίο, που θα χαρακτηρίζει μοναδικά κάθε εγγραφή, αυτό μπορεί να γίνει είτε ορίζοντας ως ένα πεδίο UNIQUE είτε ορίζοντάς το ως πρωτεύον κλειδί. Ένα πεδίο που έχει οριστεί ως πρωτεύον κλειδί θα πρέπει πάντα σε μια εγγραφή το πεδίο αυτό να συμπληρώνεται.

Όταν ένα πεδίο είναι πρωτεύον κλειδί, αυτό σημαίνει ότι απαραίτητως πρέπει να είναι και μοναδικό (UNIQUE), το αντίστροφο όμως δεν ισχύει ,δηλαδή, όταν ένα πεδίο είναι μοναδικό (UNIQUE) δεν είναι απαραίτητως πρωτεύον κλειδί. Ένα πεδίο το ορίζουμε ως πρωτεύον κλειδί με τον εξής τρόπο:

```
CREATETABLE“όνομα_πίνακα”
(“πεδίο1” INT,
“πεδίο2” VARCHAR(20),
“πεδίο3” DATE,
PRIMARYKEY(πεδίο1))
```

Στο παράδειγμά μας, το πεδίο το οποίο μπορεί να χρησιμοποιηθεί ως πρωτεύον κλειδί είναι το student id.

```
CREATETABLE students
(student id INT
Name VARCHAR(20) NOT NULL
Date of birth
DATE PRIMARY KEY(student id))
```

Γενικά οι εντολές που μπορούν να εκτελεστούν είναι οι παρακάτω:

SELECT →Μπορεί να διαβάσει όλες τις στήλες (συμπεριλαμβανομένων αυτών που μπορεί να προστεθούν αργότερα με την εντολή ALTER TABLE).

INSERT →Εισάγει τιμές.

DELETE → Μπορεί να διαγράψει πλειάδες.

UPDATE(όνομα στήλης A)

REFERENCES (όνομα στήλης A) → Μπορεί να ορίσει ξένα κλειδιά (σε άλλους πίνακες) που αναφέρονται στην στήλη A.

3.4.3 Επιλογή δεδομένων από πίνακα

Η κύρια εντολή στην SQL για την επιλογή δεδομένων από τον κάθε πίνακα έχει την μορφή:

SELECT <λίστα από πεδία(είναι μια σειρά πεδίων που ανήκουν σε κάποιον πίνακα.)>

FROM <λίστα από πίνακες(Λίστα από πεδία)>

WHERE <συνθήκη(Συνθήκη είναι η συνθήκη που πρέπει να πληρούν οι πλειάδες των πινάκων που ανήκουν στην λίστα πινάκων, έτσι ώστε να επιλέξουμε τις τιμές από τα αντίστοιχα πεδία τα οποία υπάρχουν στο *SELECT*)>

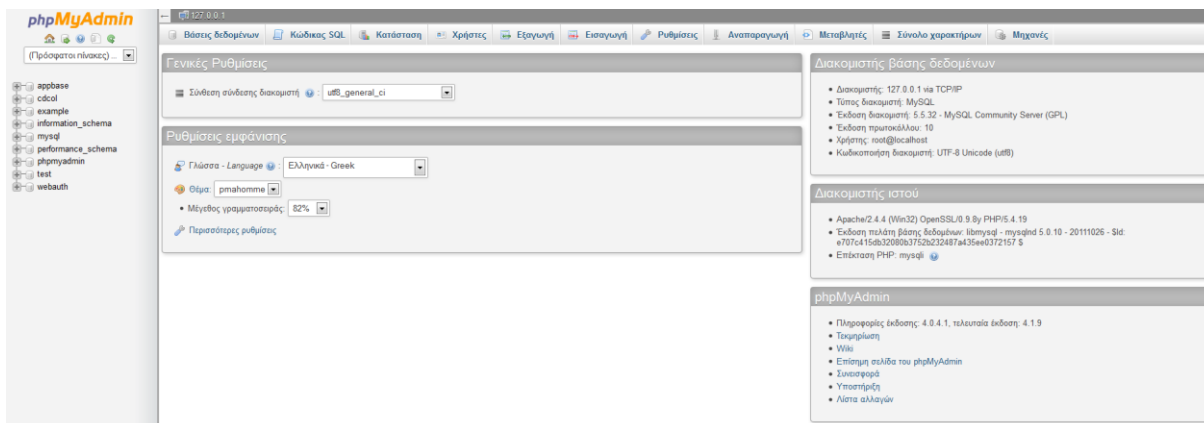
Για παράδειγμα αν θέλουμε να εμφανίσουμε στην οθόνη μας τα ονόματα των φοιτητών από τον πίνακα *students* τότε εκτελούμε την εντολή *select* ως εξής: *SELECT name FROM students*

Βασικές συναρτήσεις της SQL:

- *AVG()* Επιστρέφει την μέση τιμή
- *COUNT()* Επιστρέφει τον αριθμό των εγγραφών
- *FIRST()* Επιστρέφει την πρώτη τιμή
- *LAST()* Επιστρέφει την τελευταία τιμή
- *MAX()* Επιστρέφει την μέγιστη τιμή
- *MIN()* Επιστρέφει την ελάχιστη τιμή
- *SUM()* Επιστρέφει το σύνολο
- *UCASE()* Μετατρέπει σε κεφαλαία
- *LCASE()* Μετατρέπει σε μικρά
- *LEN()* Επιστρέφει το μέγεθος μια συμβολοσειράς
- *ROUND()* Στρογγυλοποιεί μια τιμή
- *NOW()* Επιστρέφει την τρέχουσα ώρα και ημερομηνία του συστήματος

3.5 PhpMyAdmin

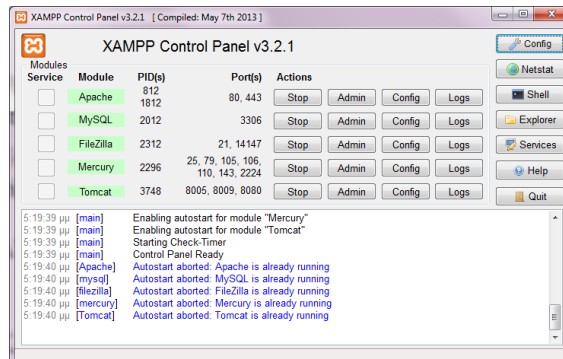
Το *PhpMyAdmin* δίνει τη δυνατότητα διαχείρισης βάσεων δεδομένων *MySQL* στο Διαδίκτυο. Μπορεί να διαχειριστεί κάποιο *mysql server* ή και απλές βάσεις δεδομένων. Το *PhpMyAdmin* παρέχει πολλές λειτουργίες μέσω γραφικής διεπαφής όπως πίνακες, πεδία, συσχετίσεις, ευρετήρια, χρήστες και δικαιώματα. Στην εικόνα 6 υπάρχει ένα στιγμιότυπο του *PhpMyAdmin*. Είναι ένα περιβάλλον που μπορούμε να δημιουργήσουμε βάσεις δεδομένων, πίνακες και να καταχωρήσουμε δεδομένα στους πίνακες μας.



Εικόνα 6: Περιβάλλον phpMyAdmin

3.6 XAMPP

Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού και ανεξαρτήτου πλατφόρμας. Παρέχει τον εξυπηρετητή http apache, τη βάση δεδομένων MySQL, και ένα διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Είναι διαθέσιμο σε πολλές πλατφόρμες (Windows, Linux, Solaris και Mac OS X).



Εικόνα 7: Xampp

3.7 Html5

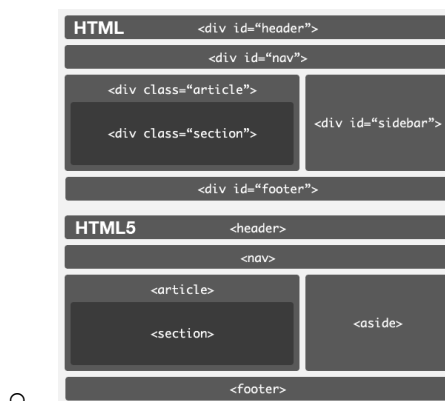
3.7.1 Τι είναι η Html5

Η *Html5* είναι μια επέκταση της γλώσσας προγραμματισμού ιστοσελίδων (HTML-HyperText Markup Language). Δημιουργήθηκε σε συνεργασία με την ομάδα Web Hypertext Application Technology Working Group (WHATWG) και την ομάδα World Wide Web Consortium (W3C). Ο σκοπός της δημιουργίας Html5 είναι η αναπαραγωγή ήχου και βίντεο σε κάποιο web site χωρίς τη χρήση κάποιου plug-in ή χρήσης κάποιου διαδικτυακού προγράμματος όπως το Adobe Flash, το Microsoft Silverlight, το Apache Pivot και τη Sun

JavaFX. Η Html5 είναι ακόμα υπό ανάπτυξη όμως υποστηρίζεται από πολλούς φυλλομέτρησες. Επίσης με χρήση του Html5 γίνεται δημιουργία mobile web sites τα οποία προσφέρουν περισσότερες ιδιότητες από ένα τυπικό mobile web site όπως το GPS και αναπαραγωγή βίντεο χωρίς τη χρήση προγραμμάτων.

Η Html5 σε σχέση με την Html υποστηρίζει:

- Το στοιχείο canvas για το drawing
- Αναπαραγωγή video και audio
- Νέα στοιχεία περιεχομένου όπως τα footer,header,nav και section
- Νέα στοιχεία δημιουργίας φόρμας όπως τα calendar,date,time,email,url και search



Εικόνα 8:Html5 to html

3.7.2 Νέες δυνατότητες στην HTML5

➤ Video στην HTML5

Με βάση την προηγούμενη τεχνολογία HTML για να δούμε ένα βίντεο ήταν απαραίτητη η εγκατάσταση κάποιου plug-in ή η εγκατάσταση κάποιου προγράμματος όπως το Flash. Με την δημιουργία της HTML5 η προβολή ενός βίντεο γίνεται αυτόματα χωρίς την προσθήκη καποιων πρόσθετων στοιχείων. Για την προσθήκη ενός βίντεο στην HTML5 εκτελούμε το παρακάτω HTML5 κώδικα:

```
<video src="movie.ogg" controls="controls">
</video>
```

Για να γίνει προβολή του μηνύματος σε browsers που δεν υποστηρίζουν HTML5 βίντεο γράφουμε το παρακάτω:

```
<video src="movie.ogg" width="320" height="240" controls="controls">
```

Ο φυλλομετρητής σας δεν υποστηρίζει το στοιχείο video της HTML5.

```
</video>
```

Τέλος μια πιο εξειδικευμένη χρήση του στοιχείου βίντεο είναι η εξής:

```
<video width="320" height="240" controls="controls">
<source src="movie.ogg" type="video/ogg">
<source src="movie.mp4" type="video/mp4">
Your browser does not support the video tag.
</video>
```

Οι ιδιότητες που προσφέρει η HTML5 για το βίντεο είναι οι εξής:

- autoplay: Το video θα ξεκινήσει αυτόματα μόλις φορτώσει
- controls: Θα εμφανίζονται ή όχι τα κουμπιά όπως το play
- height: Το ύψος σε pixels
- loop: Αν θα επαναλαμβάνεται το video μετά το τέλος του
- preload: Το video θα φορτώνεται μαζί με την σελίδα
- src: Το URL του video
- width: Το πλάτος σε pixels

➤ Audio στην HTML5

Το ίδιο ισχύει όπως και στο βίντεο και για τα audio clips. Έτσι η HTML5 μας ορίζει το στοιχείο audio για την αναπαραγωγή ήχων. Αυτό που πρέπει να γράψουμε είναι:

```
<audio src="song.ogg" controls="controls">
</audio>
```

Για να γίνει προβολή του μηνύματος σε browsers που δεν υποστηρίζουν HTML5 audio γράφουμε το παρακάτω:

```
<audio src="song.ogg" controls="controls">
```

Ο φυλλομετρητής σας δεν υποστηρίζει το στοιχείο audio της HTML5.

```
</audio>
```

Τέλος μια πιο εξειδικευμένη χρήση του στοιχείου audio είναι η εξής:

```
<audio controls="controls">
<source src="song.ogg" type="audio/ogg">
<source src="song.mp3" type="audio/mpeg">
```

Your browser does not support the audio element.

```
</audio>
```

Οι ιδιότητες που προσφέρει η HTML5 για τον ήχο είναι οι εξής:

- autoplay: ο ήχος θα ξεκινήσει αυτόματα μόλις φορτώσει
- controls: Θα εμφανίζονται ή όχι τα κουμπιά όπως το play
- loop: Αν θα επαναλαμβάνεται ο ήχος μετά το τέλος του
- preload: Ο ήχος θα φορτώνει μαζί με την σελίδα
- src: Το URL του ηχητικού κομματιού μας

3.7.3 CSS3

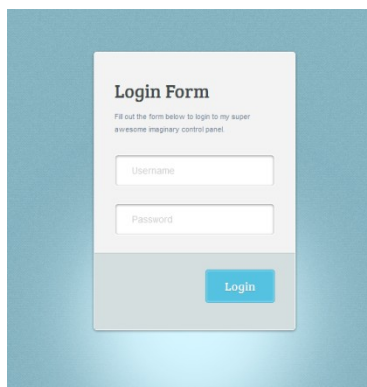
Το CSS3 είναι η 3^η έκδοση του **CSS (Cascade Style Sheet)** το οποίο δημιουργήθηκε για να έχουν οι σελίδες web μια συγκεκριμένη στοίχιση, χρώμα κ.α. Καθώς περνούσαν τα χρόνια, το CSS έγινε απαραίτητο εργαλείο αλλά και προαπαιτούμενος παράγοντας για να στηθεί μια ιστοσελίδα. Έτσι επεκτάθηκε και κατέληξε να γίνει μία πανίσχυρη γλώσσα αισθητικού design ιστοσελίδων.

Παράδειγμα μία ιστοσελίδα χωρίς CSS, έδειχνε κάπως έτσι:

Login Form

Fill out the form below to login to my super awesome imaginary control panel.

Εικόνα 9: Web σελίδα χωρίς τη χρήση css

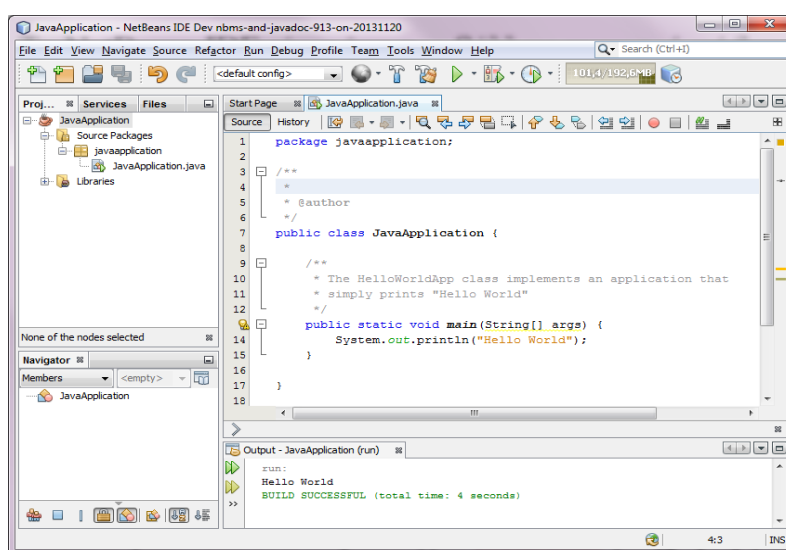


Εικόνα 10: Web σελίδα με τη χρήση των css

3.8 NetBeans IDE

Το NetBeans IDE (Integrated Development Environment) είναι ένα πλήρες περιβάλλον ανάπτυξης εφαρμογών ανοιχτού κώδικα. Ξεκίνησε το 1996 ως μαθηματικό πρόγραμμα με όνομα Xelfi από τον Faculty of Mathematics and Physics στο Charles University στην

Πράγα. Το 1997 ο Roman Stanek δημιούργησε μια εταιρία η οποία εμπορεύονταν το Netbeans IDE η οποία στη συνέχεια αγοράστηκε από την Sun Microsystems το 1999. Η Sun μετέτρεψε το NetBeans σε λογισμικό ανοιχτού κώδικα τον Ιούνιο του 2000 και συνεχίζει να είναι ο κύριος ανάδοχος. Το 2010 η Oracle ανέλαβε την εταιρία και το NetBeans. Το NetBeans δίνει τη δυνατότητα στους προγραμματιστές να κάνουν compile, debug και να αναπτύξουν προγράμματα. Είναι γραμμένο σε Java όμως υποστηρίζει όλες τις γλώσσες προγραμματισμού όπως Java, C/C++, php, Python, Javascript κ.α. Λειτουργεί σε όλες τις πλατφόρμες (Windows, Linux, mac, Os) στις οποίες έχει εγκατασταθεί JVM. Χρησιμοποιεί ένα μεγάλο αριθμό υπομονάδων (modules) τα οποία είναι εξαρτημένα μεταξύ τους και βοηθάνε στην επέκταση της λειτουργικότητας του NetBeans IDE. Στην εικόνα 11 φαίνεται ένα παράδειγμα NetBeans με χρήση γλώσσας Java.



Εικόνα 11: JavaApplication με χρήση NetBeans

3.9 Java

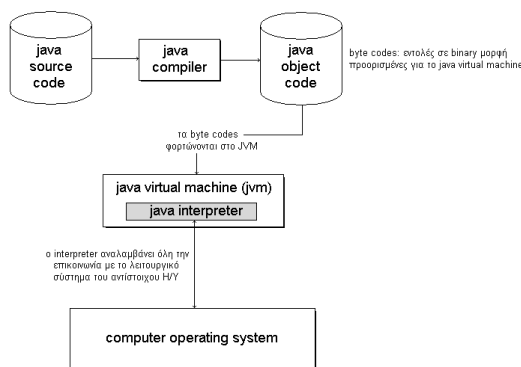
Η sun Microsystems χρηματοδότησε ένα εσωτερικό ερευνητικό έργο βασισμένη στη γλώσσα C++, της οποίας ο δημιουργός ήταν ο James Gosling και την ονόμασε Oak, από μία βελανιδιά (Oak) που ήταν έξω από το παράθυρο του στην Sun. Αργότερα ανακαλύφθηκε ότι υπήρχε ήδη μία γλώσσα με το ίδιο όνομα. Στη συνέχεια μία ομάδα ανθρώπων της Sun επισκέφθηκε ένα τοπικό καφενείο και προτάθηκε το όνομα Java (είδος καφέ) και παρέμεινε. Η Java χρησιμοποιήθηκε το 1993 για την προσθήκη δυναμικών περιεχομένων, όπως διαδραστικότητας και κινήσεων παραστάσεων σε ιστοσελίδες. Η Sun ανακοίνωσε επισήμως τη Java σε ένα συνέδριο τον Μάιο του 1995 και από τότε τράβηξε το ενδιαφέρον της επιχειρηματικής κοινότητας λόγω του μεγάλου ενδιαφέροντος για το web. Η Java χρησιμοποιείται σήμερα για τη δημιουργία δυναμικών ιστοσελίδων, για την ανάπτυξη εφαρμογών μεγάλης κλίμακας, για τη βελτίωση της λειτουργικότητας WEB διακοσμητών, για την δημιουργία εφαρμογών για καταναλωτικές συσκευές (π.χ. κινητά τηλέφωνα, pagers και προσωπικούς ηλεκτρονικούς βοηθούς) και για πολλούς άλλους σκοπούς.

3.9.1 Χαρακτηριστικά της Java

- Δυνατότητα δημιουργίας εφαρμογών applets(Προγράμματα που εκτελούνται σε μία ιστοσελίδα) και ανεξάρτητων εφαρμογών.
- Είναι interpreted γλώσσα. Αυτό σημαίνει ότι ο java compiler δεν παράγει εκτελέσιμο κώδικα αλλά μια μορφή ψευδοκώδικα (bytecode) ώστε να εκτελεστεί κάποιο αρχείο. Αντικειμενοστραφής(ομοιότητες εντολών με τη C++).
- Επιτρέπει την χρήση ασφαλών προγραμμάτων.
- Ανιχνεύει λάθη ή τυχόν προβλήματα κατά την εκτέλεση.
- Κατανεμημένη(distributed). Είναι κατάλληλη για εφαρμογές από το δίκτυο.
- Ασφαλής(secure).Η Java έχει σχεδιαστεί ώστε να ελαχιστοποιείται η προσβολή από κάποιο Applet.
- Υποστηρίζει πολυνηματική επεξεργασία(multithreaded processing). Συγκεκριμένα η Java υποστηρίζει τη χρήση πολλών threads με τη χρήση κάποιου χρονοδρομολογητή (scheduler).

3.9.2 Δημιουργία και εκτέλεση της Java

Για την εκτέλεση μιας Java εφαρμογής εκτελούνται 5φάσεις(Επεξεργασία, μεταγλώττιση, φόρτωση, επαλήθευση και εκτέλεση)



Εικόνα 12:Εκτέλεση Java

- Φάση1:Δημιουργία ενός προγράμματος

Για την δημιουργία ενός αρχείου Java χρησιμοποιείται κάποιο πρόγραμμα επεξεργασίας. Μέσω του προγράμματος επεξεργασίας πληκτρολογείται ο πηγαίος κώδικας Java και γίνεται αποθήκευση του αρχείου με την επέκταση. Java. Στα συστήματα linux χρησιμοποιούνται ευρέως τα vi και emacs ενώ στα windows υπάρχουν άλλα διαθέσιμα προγράμματα επεξεργασίας όπως το Notepad, το TextPad, το jEdit κ.α. Επίσης χρησιμοποιούνται ολοκληρωμένα περιβάλλοντα ανάπτυξης(integrated development environments-IDE) όπως το Netbeans, Eclipse, jBuilder, jCreator, BlueJ και JBuilder, τα οποία υποστηρίζουν επεξεργαστές για τη σύνταξη και επεξεργασία προγραμμάτων και αποσφαλματωτές(debugger) για τον εντοπισμό λογικών σφαλμάτων.

- Φάση 2:Μεταγλώττιση σε Bytecode

Για να γίνει μεταγλώττιση(compile) ενός Java αρχείου σε bytecode χρησιμοποιείται η εντολή javac. Για παράδειγμα ένα πρόγραμμα με όνομα hello. Java θα πρέπει να πληκτρολογηθεί στο παράθυρο εντολών η εντολή javac hello.java. Έτσι ο μεταγλωττιστής δημιουργεί ένα αρχείο που τελειώνει σε .class. Με αυτό τον τρόπο γίνεται μετάφραση του Java πηγαίου κώδικα σε bytecode τα οποία εκτελούνται από το Java Virtual Machine(JVM) ώστε να γίνει η διαδικασία εκτέλεσης(φάση5). Σε ολοκληρωμένα περιβάλλοντα ανάπτυξης(integrated development environments-IDE) η διαδικασία compile γίνεται από το ίδιο το πρόγραμμα. Το JVM (virtual machine) είναι μια εφαρμογή λογισμικού που χρησιμοποιεί η γλώσσα Java ώστε να λειτουργεί σε διαφορετικές πλατφόρμες(Windows,linux, Unix)

➤ Φάση 3:Φόρτωση ενός προγράμματος στη μνήμη

Το JVM χρησιμοποιεί κάποιον φορτωτή κλάσης (class loader) ο οποίος διαβάζει τα αρχεία .class που περιέχουν τα bytecode του προγράμματος και τα τοποθετεί στη μνήμη.

➤ Φάση 4: Επαλήθευση των bytecode

Σε αυτό το βήμα με την φόρτωση των κλάσεων γίνεται επαλήθευση των bytecode ώστε να ελεγχτεί ότι όλα τα bytecode είναι έγκυρα και δεν παραβιάζουν τους περιορισμούς ασφαλείας

➤ Φάση 5: Εκτέλεση

Για να εκτελεστεί το πρόγραμμα , το JVM διαβάζει τα bytecode και τα μεταγλωττίζει (τα μεταφράζει) σε μία γλώσσα που μπορεί να καταλάβει ο υπολογιστής.

3.9.3 Εκτέλεση της Java με χρήση του NetBeans

Στην Java έχουμε εκτελέσιμα προγράμματα με κατάληξη .jar τα οποία αποθηκεύονται στον φάκελο /usr/bin. Στην συγκεκριμένη εφαρμογή γίνεται η χρήση της γλώσσας Java με χρήση της πλατφόρμας NetBeans. Η εκτέλεση του αρχείου γίνεται με τα παρακάτω βήματα:

Class→Jar→Java Virtual Machine→Netbeans→Programming

Δημιουργίας Java Application με χρήση NetBeans

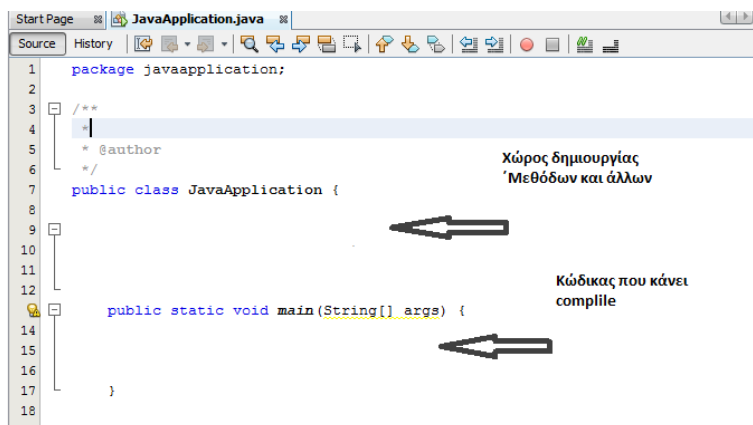
1.Δημιουργία Project

File > New Project >Java > Java Application > Next > Finish

2.Δημιουργία κλάσης

Δεξί κλικ στο project > New Java class

Δημιουργείτε ένα κενό project(Εικόνα 13) με κατάληξη .java



Εικόνα 13:Example Java Application

3.9.4 Βιβλιοθήκες Κλάσεων της Java

Η Java παρέχει βιβλιοθήκες κλάσεων Java γνωστές και ως Java API (Application Programming Interfaces-διασύνδεση προγραμματισμού εφαρμογών) οι οποίες είναι διαθέσιμες σε κάθε java περιβάλλον. Η Βιβλιοθήκη κλάσεων οργανώνεται σε πακέτα.

Βασικά πακέτα

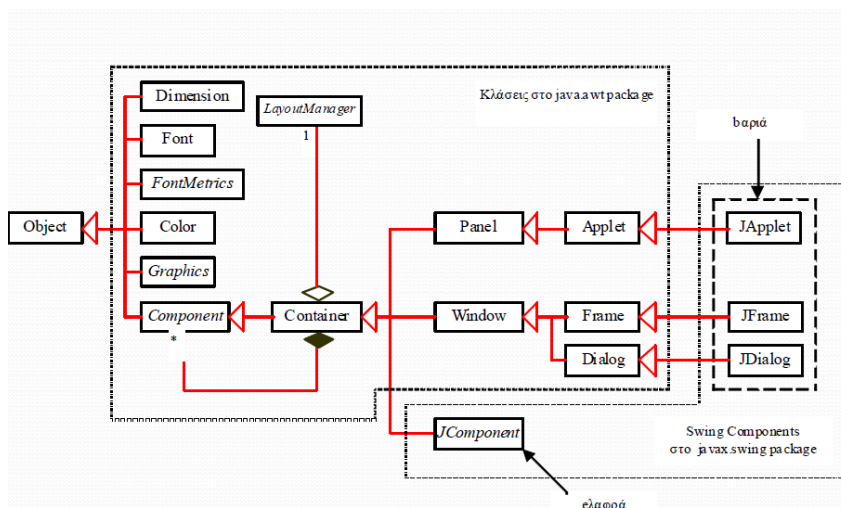
- ✓ Java.applet (Κλάσεις για την εφαρμογή Java Applets)
- ✓ Java.awt (Κλάσεις που υποστηρίζουν το γραφικό περιβάλλον GUI της Java)
- ✓ Java.io (Κλάσεις για λειτουργίες εισόδου/εξόδου)
- ✓ Java.lang (Κλάσεις που υποστηρίζουν τα βασικά χαρακτηριστικά της γλώσσας Java)
- ✓ Java.util (Κλάσεις για επιτέλεση βασικών εργασιών σε δεδομένα)
- ✓ Java.net (Κλάσεις που υποστηρίζουν λειτουργίες Internet)

3.9.5 Γραφικό Περιβάλλον Χρήστη (GUI-Gráfica User Interface)

Το Γραφικό Περιβάλλον Χρήστη της Java είναι ένα σύνολο γραφικών στοιχείων στο οποίο ο χρήστης μπορεί να αλληλεπιδρά με την εφαρμογή. Τα GUI δημιουργούνται από συστατικά GUI γνωστά και ως window gadgets. Ένα συστατικό GUI είναι ένα αντικείμενο με το οποίο ο χρήστης αλληλεπιδρά με οποιαδήποτε συσκευή εισόδου (πληκτρολόγιο, ποντίκι κ.α). Σε πολλά IDE υπάρχει δυνατότητα δημιουργίας GUI μέσω εργαλείων σχεδίασης χωρίς να χρειαστεί να γραφτεί κώδικας αφού δημιουργείται αυτόματα.

Τα βασικά των γραφικών Διεπαφών:

Στη java όσο αφορά το γραφικό περιβάλλον περιέχει πολλά συστατικά GUI που επιτρέπει στους δημιουργούς εφαρμογών να δημιουργούν δυνατά γραφικά περιβάλλοντα χρήστη. Τα περισσότερα έχουν γραφτεί και εμφανίζονται αποκλειστικά στη Java. Αποτελούν μέρος των θεμελιωδών Κλάσεων της java (Java Foundation Classes-JFC)-των βιβλιοθηκών της java για ανάπτυξη GUI σε διαφορετικές πλατφόρμες. Στο πακέτο javax.swing περιλαμβάνονται όλα τα στοιχεία (components) του περιβάλλον διεπαφής χρήστη (κουμπια, split pane, scrollPane και άλλα) Στην εικόνα 10 φαίνεται η ιεραρχία των συστατικών που υπάρχουν στη Java



Εικόνα 14: Ιεραρχία συστατικών της java

Όπως παρατηρούμε τα JApplet, JDialog και JFrame βρίσκονται ιεραρχικά ψηλότερα από κάθε υποδοχέα.

Τα Jframes(πλαίσια) είναι ένα παράθυρο το οποίο περιέχεται μέσα σε κάποιο άλλο παράθυρο. Και μπορεί να περιέχει όλα τα γραφικά components.

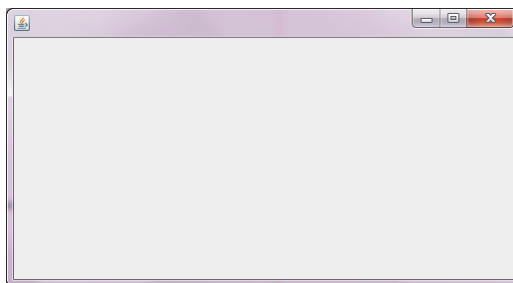
Παράδειγμα δημιουργίας frame

Για την δημιουργία του frame στη java χρειάζεται ο παρακάτω κώδικας

```
public class Main{
    public static void main(String[] args) {
        JFrame jFrame1 = new JFrame();

        jFrame1.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        jFrame1.pack();
        jFrame1.setVisible(true);
    }
}
```

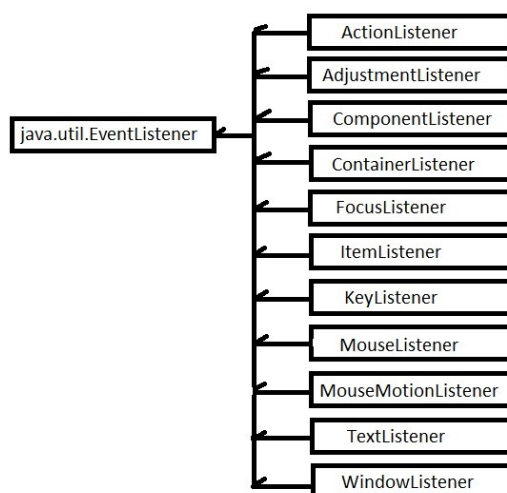
Το αποτέλεσμα φαίνεται στην εικόνα 15



Εικόνα 15:Frame

3.9.6 GUI και Διασυνδέσεις ακρόασης Συμβάντων(Listeners)

Όπως αναφέρθηκε ο χρήστης αλληλεπιδρά με το GUI. Αυτό επιτυγχάνεται με τη χρήση ακροατών συμβάντων. Οι ακροατές συμβάντων είναι κλάσεις που υλοποιούν συγκεκριμένες διασυνδέσεις(Listeners), ανάλογα με το είδος των συμβάντων που πρόκειται να χειριστούν. Για παράδειγμα μια κλάση ακροατής που προκαλείται από το πάτημα ενός πλήκτρου πρέπει να υλοποιεί τη διασύνδεση ActionListener. Κάθε διασύνδεση καθορίζει μια ή περισσότερες μεθόδους χειρισμού συμβάντων που πρέπει να δηλώνονται στην κλάση που υλοποιεί τη διασύνδεση. Οπότε κάθε κλάση που υλοποιεί μια διασύνδεση θα πρέπει να δηλώνει όλες τις abstract μεθόδους αυτής της διασύνδεσης για να μπορεί να δημιουργήσει αντικείμενα. Όταν λαμβάνει χώρα κάποιο συμβάν, το συστατικό GUI με το οποίο αλληλεπιδρά ο χρήστης ειδοποιεί τους listeners καλώντας την κατάλληλη μέθοδο χειρισμού. Ο ActionListener καλεί τη μέθοδο actionPerformed. Στην εικόνα 16 φαίνονται κάποιες διασυνδέσεις ακρόασης.



Εικόνα 16:Listeners του πακέτου java.awt.event

3.9.7 Treads

Τα Treads είναι διεργασίες που μπορούν να τρέξουν παράλληλα σε ένα πρόγραμμα. Πολλές γλώσσες προγραμματισμού όπως και η Java διαθέτουν εργαλεία για την υλοποίηση Treads γνωστές και ως multithreading languages. Στον παραδοσιακό προγραμματισμό όταν ένα πρόγραμμα εκτελείται ονομάζεται διεργασία(process) και οι εντολές του εκτελούνται σειριακά ή μια μετά την άλλη μέχρι το τέλος του. Δηλαδή σαν να εκτελείται σε μια διεργασία μόνο ένα Tread. Στις multithreading γλώσσες μπορούμε να ορίσουμε σε κάποια διεργασία να τρέχει ένα ή περισσότερα threads παράλληλα(ονομάζονται και lightweight processes) .Πιο συγκεκριμένα τα threads είναι διεργασίες που μπορούν να εκτελεσθούν παράλληλα μέσα στη αρχική διεργασία.

3.10 WebServices

Οι WebServices είναι μια τεχνολογία που επιτρέπει την επικοινωνία μεταξύ εφαρμογών μέσω δικτύου. Οι WeServices επικοινωνούν χρησιμοποιώντας τεχνολογίες όπως: XML,JSON και HTTP. Τα πιο διαδεδομένα WeServices είναι τα SOAP και RESTFul. Το SOAP βασίζεται στην δημιουργία ενός MXL envelope και την αποστολή του ως προς τον server του webservice. Αντίθετα το REST είναι απλούστερο καθώς απαιτεί τη κλήση ενός URL. Στην συγκεκριμένη εργασία θα αναφερθούμε στις υπηρεσίες RESTFull

3.10.1RESTFull

Τι είναι RESTFull υπηρεσίες

Οι RESTFull είναι web services που χρησιμοποιούν το πρότυπο REST(Representational State Transfer) για την υλοποίηση του. Χρησιμοποιεί ως Java API το JAX-RS (Representation State Transfer(REST)) το οποίο χρησιμοποιεί τους παραδοσιακούς μεθόδους αιτήματος/απόκρισης του διαδικτύου όπως τα GET και POST. Κάθε μέθοδος προσδιορίζεται από ένα μοναδικό URL έτσι όταν οπ διακοσμητής(server) λαμβάνει ένα αίτημα ξέρει ακριβώς ποια λειτουργία να εκτελέσει. Οι RESTFull web services σε αντίθεση με SOAP δεν περικλείονται σε φακέλους. Επίσης το REST δεν περιορίζεται να επιστρέφει δεδομένα σε μορφή XML. Μπορεί να γίνει χρήση πολλών μορφών όπως XML, JSON, HTML, απλό κείμενο και αρχεία πολυμέσων. Τα βασικά χαρακτηριστικά ενός μιας RESTFull Webservice αναφέρονται αναλυτικά παρακάτω.

Χαρακτηριστικά RESTFull

1.Χρήση μεθόδων για την επικοινωνία με το ένα μοναδικό uri

Το REST χρησιμοποιεί μεθόδους ώστε να κάνει ένα HTTP αίτημα. Οι μέθοδοι αυτοί είναι:

Η μέθοδος POST η οποία χρησιμοποιείται για την δημιουργία ενός πόρου στον server

Η μέθοδος GET η οποία χρησιμοποιείται για την ανάσυρση / ανάκτηση ενός πόρου

Η μέθοδος PUT για την αλλαγή της κατάστασης ενός πόρου ή την ενημέρωσή του

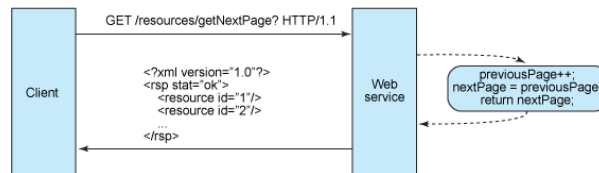
Η μέθοδος DELETE για την διαγραφή ή απομάκρυνση ενός πόρου

Κάθε μέθοδος σε μια RESTFull web service προσδιορίζεται από ένα μοναδικό URI(Uniform resource identifier).Κατά συνέπεια όταν ο εκτελεστής λαμβάνει ένα αίτημα ξέρει αμέσως ποια λειτουργία να εκτελέσει. Ένα URI εκφράζει ένα αντικείμενο στο οποίο παρέχει πρόσβαση η υπηρεσία μέσω ενός HTTP αιτήματος. Το είδος του αιτήματος καθορίζει την ενέργεια που θέλουμε να εφαρμόσουμε στο αντικείμενο αυτό και το περιεχόμενο του αιτήματος περιέχει διάφορες εξειδικεύσεις τις ενέργειας

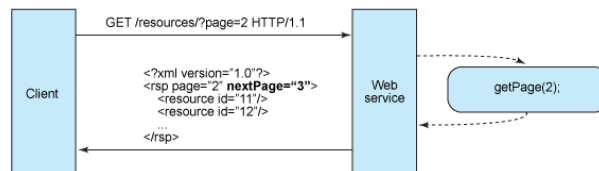
2.Είναι stateless

Για να γίνει πιο σαφές θα αναφερθούμε σε μια stetefull υπηρεσία(Εικόνα 17) μέσω της οποίας η μια κλήση αναφέρεται σε μια άλλη προγενέστερη κλήση. Για παράδειγμα όταν θέλει κάποιος χρήστης σε μια δικτυακή υπηρεσία να κάνει κλήση για login και μετά μια άλλη κλήση για ανάκτηση της πληροφορίας που επιθυμεί. Αυτή η υπηρεσία είναι statefull για να γίνει επεξεργασία της δεύτερης κλήσης η υπηρεσία θα πρέπει να θυμάται ότι αυτός ο

χρήστης έχει συνδεθεί μέσω της προηγούμενης κλήσης. Μια υπηρεσία RESTFull η οποία στηρίζει stateless υπηρεσίες(Εικόνα 18) περιλαμβάνει εντός του HTTP αιτήματος(header/body) όλη την πληροφορία(action,parameters,context,..) για την δημιουργία μιας απόκρισης στο αίτημα αυτό. Για αυτό το λόγο τα αιτήματα πρέπει να είναι ολοκληρωμένα και αυτόνομα ώστε όταν γίνεται επεξεργασία από τον server να μην χρειάζονται η ανάρτηση της προηγούμενης ή κάποιας άλλης πληροφορίας. Αυτή η υπηρεσία απαλλάσσει τον server να εκτελεί συγχρονισμό των δεδομένων μιας συνόδου με μια εξωτερική εφαρμογή.



Εικόνα 17:Statefull design



Εικόνα 18:Stateless Design

3.Απεικόνιση της δομής των καταλόγων σαν URIs

Οι χρήστες μπορούν να βρουν το δρόμο τους πιο εύκολα όταν υπάρχουν καλές URIs. Οι υπηρεσίες REST πρέπει να είναι προβλέψιμες και ευνόητες για κάθε χρήστη της υπηρεσίας και να απαιτείται καθόλου τεκμηρίωση ή εξήγηση σχετικά με τον πόρο προς τον οποίο δείχνει κάθε URI. Οι Restful μέσω του annotation @Path στο οποίο μπορείς να ορίσεις το URI για πρόσβαση στην web υπηρεσία ορίζουν τη δομή ενός URI παρόμοια με τον τρόπο δόμησης καταλόγων σε ένα σύστημα αρχείων. Για παράδειγμα η δομή ενός συγκεκριμένου URI έστω σε μια εταιρία προϊόντων θα μπορούσε να είναι:

`http://www.myservice.org/company/products/{product}`

`http://www.myservice.org/company/products/{product}/{product1}/{product2}`

Για τη σωστή δομή ενός URI σε μια υπηρεσία REST πρέπει να προσέξουμε τα παρακάτω:

- Απόκρυψη των επεκτάσεων που αφορούν την τεχνολογία προγραμματισμού στην πλευρά του server(.jsp, .php, .asp). Αυτό έχει σαν αποτέλεσμα όταν θελήσουμε να αλλάξουμε το script στο οποίο δείχνει το URI να μην χρειάζεται να αλλάξουμε και το URI
- Χρήση πάντα μικρών χαρακτήρων
- Αντικατάσταση κενών διαστημάτων(στα URIs) με παύλες ή κάτω παύλες

- Αποφύγει τον query strings
- Αντί για τη χρήση του κωδικού “404 Not found” ;όταν το URI στο οποίο στάλθηκε το αίτημα είναι μόνο τμήμα ενός αποδεκτού URI μπορεί να γίνει χρήση μιας default σελίδας σαν απόκριση.

4.Μεταφορά δεδομένων μέσω XML,JSON ή και των δυο

Μέσω Restful ο client μπορεί να επιλέξει τη μορφή των περιεχομένων της απόκρισης του server.Αυτό γίνεται με το να φτιάξουμε την υπηρεσία μας ώστε να κάνει χρήση του ενσωματωμένου HTTP Accept header όπου η τιμή του θα είναι ένας τύπος MIME. Οι πιθανοί τύποι μορφοποίησης δεδομένων σε μια Restful υπηρεσία είναι οι JSON(application/json), XML(application/xml) και XHTML(application/xhtml+xml).

Στην παρακάτω εικόνα φαίνεται ένα παράδειγμα δημιουργίας restful services με χρήση της γλώσσας java και του περιβάλλοντος NetBeans

```

    * @author voula
    */
    @Path("myservice")
    public class MyserviceResource {

        @Context
        private UriInfo context;

        /**
         * Creates a new instance of MyserviceResource
         */
        public MyserviceResource() {
        }

        /**
         * Retrieves representation of an instance of main.MyserviceResource
         * @return an instance of java.lang.String
         */
        @GET
        @Produces("text/plain")
        @Path("{name}")
        public String getText(@PathParam("name") String name) {
            return name;
        }

        @PUT
        @Consumes("text/plain")
        public void putText(String content) {
        }
    }

```

Εικόνα 19:RESTful example σε JAVA μέσω NetBeans

Η Βασική διαδρομή για την υπηρεσία είναι ο κατάλογος resources του έργου. Για να πάρουμε την παραπάνω υπηρεσία το πλήρες URI είναι:

<http://localhost:8080/MyServices/webresources/myservice/kostas>

4.Υλοποίηση

Για την υλοποίηση της εφαρμογής χρειάστηκε ο η δημιουργία και υλοποίηση ξεχωριστών τμημάτων. Αρχικά έγινε δημιουργία της βάσης Δεδομένων και των πινάκων που είναι απαραίτητα. Σε δεύτερο χρόνο έγινε δημιουργία της εφαρμογής με χρήση της γλώσσας Java. Στη συνέχεια έγινε η κατασκευή των web services με Restful υπηρεσίες ώστε να επιτευχτεί η επικοινωνία μεταξύ client/server. Τέλος έγινε η δημιουργία και ολοκλήρωση μιας web σελίδας η οποία με χρήση της τεχνολογίας PHP θα γίνεται σύνδεση με την Βάση Δεδομένων και θα εκτελούνται διάφορες λειτουργίες όπως Διαγραφή, δημιουργία.

4.1 Δημιουργία βάσης Δεδομένων

Η βάση μας πρέπει να κρατάει αρχείο για το τι συμβαίνει γενικά, ποιοι χρήστες υπάρχουν στην εφαρμογή ,ποιος χρήστης έχει κάνει εγγραφή, ποιος είναι διαθέσιμος,(δημιουργία ενός πίνακα) ποιο copy έχει γίνει, πότε εκτελέστηκε και από ποιον χρήστη (πίνακας 2) και τα διαθέσιμα copy και από ποιόν χρήστη έχουν εκτελεστεί καθώς και τα paste που έχουν γίνει(πίνακας 3)

Οπότε πρέπει να δώσουμε στοιχεία στη βάση για την εφαρμογή μας. Ας αρχίσουμε με τον πίνακα users:

- #id→ Το πρωτεύον κλειδί. Το αναγνωριστικό των χρηστών. Ακέραιος, Auto incr.
- Username→ Το όνομα των χρηστών που θα κάνουν χρήση της εφαρμογής, Varchar(20)
- Passcode→ τον κωδικό τον οποίο θα συνδέονται οι χρήστες. Varchar(20)
- Online→θα εμφανίζει το status των χρηστών αν είναι συνδεδεμένοι ή όχι

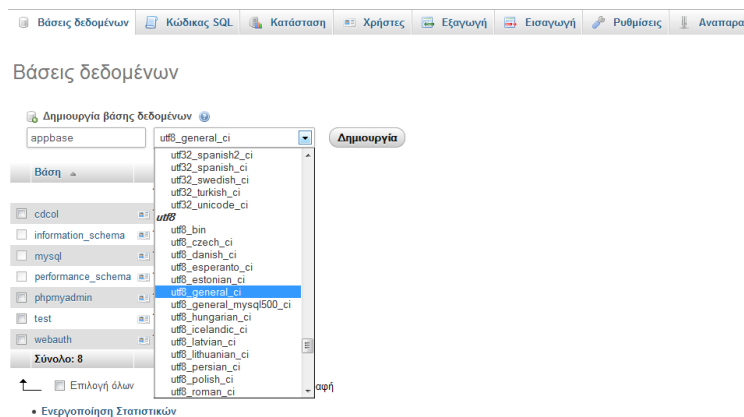
Στη συνέχεια έγινε δημιουργία του πίνακα copied_values

- #id→ Το πρωτεύον κλειδί. Το αναγνωριστικό των χρηστών. Ακέραιος, Auto incr
- Copy→ αποθηκεύει τα copy που γίνονται από τους χρήστες.text
- User→τον χρήστη που εκτελεί το copy. Varchar(20)
- When→Την ημερομηνία και ώρα που εκτελέστηκε το copy.timestamp

Τέλος μένει η δημιουργία του πίνακα paste

- #id→ Το πρωτεύον κλειδί. Το αναγνωριστικό των χρηστών. Ακέραιος, Auto incr
- paste→ αποθηκεύει τα paste που γίνονται από τους χρήστες. Varchar(200)
- User→τον χρήστη που εκτελεί το paste. Varchar(20)
- When→Την ημερομηνία και ώρα που εκτελέστηκε το paste.timestamp

Γράφουμε σε ένα browser <http://localhost/phpmyadmin>. πάμε Databases-->Create a database ώστε να δημιουργήσουμε μια βάση δεδομένων και στη συνέχεια επιλέγουμε κωδικοποίησης χαρακτήρων. Η συγκεκριμένη βάση έχει όνομα appbase



Εικόνα 20: Δημιουργία Βάση δεδομένων

Στην συνέχεια έγινε δημιουργία πινάκων

#	Name	Type	Length	Collation	Default	A_I	Notes
1	id	INT	10		null	yes	Primary Key
2	username	VARCHAR	20	Utf_general_ci	null		
3	passcode	VARCHAR	10	Utf_generafmp4_ci	null		
4	online	SMALLINT	6		Όπως ορίστηκε(0)		

Πίνακας 2: users table

#	Name	Type	Length	Collation	Default	A_I	Notes
1	id	INT	10		null	yes	Primary Key
2	copy	TEXT		Utf_general_ci	null		
3	user	VARCHAR	20	Utf_genera_ci	null		
4	when	TIMESTAMP			Current_timestamp		

Πίνακας 3: copied_values table

#	Name	Type	Length	Collation	Default	A_I	Notes
1	id	INT	10		null	yes	Primary Key
2	pasted	VARCHAR	200	Utf_general_ci	null		
3	user	VARCHAR	20	Utf_genera_ci	null		
4	when	TIMESTAMP			Current_timestamp		

Πίνακας 4: pasted_values table

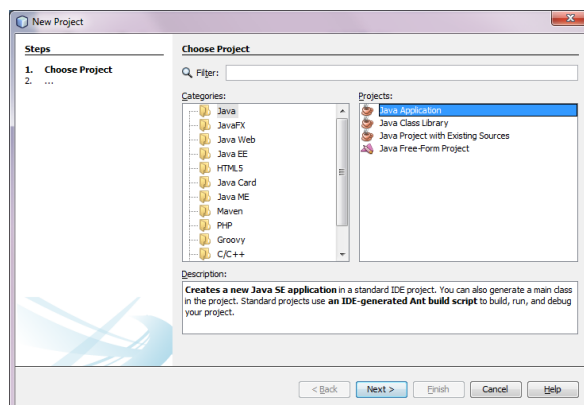
Στον παρακάτω πίνακα φαίνονται οι πίνακες της βάσης arpbase που έχουν δημιουργηθεί.

Πίνακας	Ενέργεια	Εγγραφές	Τύπος	Σύνθεση	Μέγεθος	Περίσσεια
copied_values	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~1	InnoDB	utf8_general_ci	1.6 KB	-
pasted_values	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~3	InnoDB	utf8_general_ci	1.6 KB	-
users	Περιήγηση Δομή Αναζήτηση Προσθήκη Αδειασμα Διαγραφή	~5	InnoDB	utf8mb4_general_ci	32 KB	-
3 πίνακες	Σύνολο		9 InnoDB	utf8mb4_general_ci	64 KB	0 B

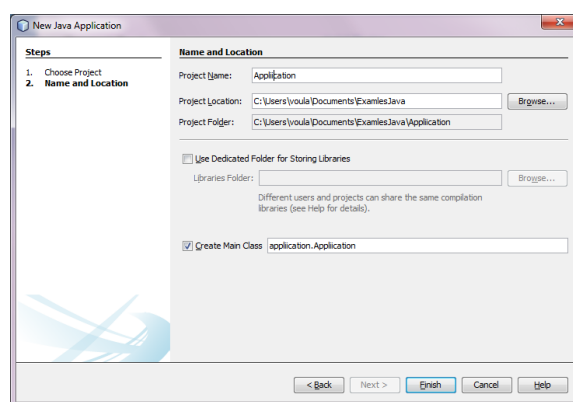
Εικόνα 21: Πίνακες στην βάση δεδομένων (arpbase)

4.2 Δημιουργία νέου project java κλάσεων

Για την δημιουργία νέου project ακλούθησα τα βήματα δημιουργίας νέου project (files→New Project), όπως φαίνεται στις εικόνες.



Εικόνα 22: Δημιουργία νέου project βημα1



Εικόνα 23: Δημιουργία νέου project βημα2

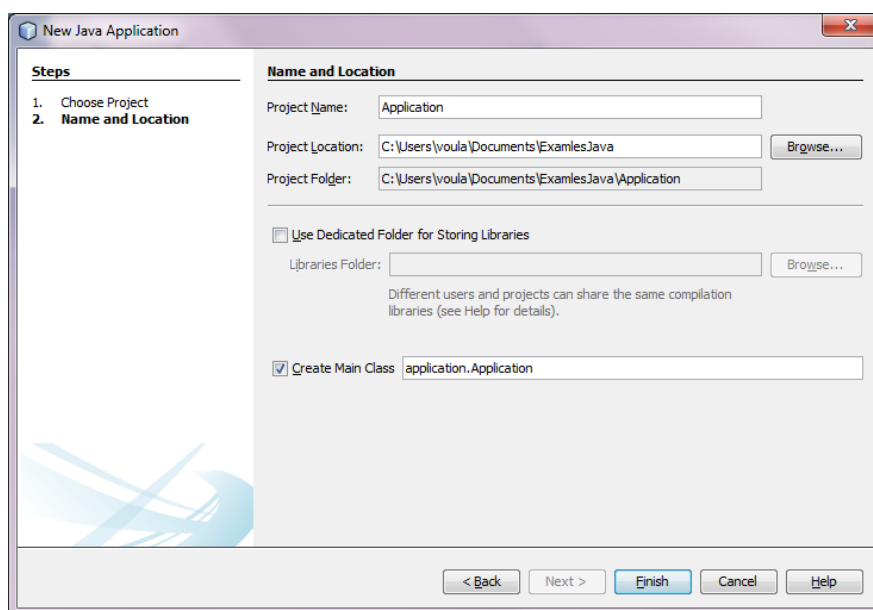
Έχοντας δημιουργήσει ένα project με όνομα Application ονόμασα ένα package με το ίδιο όνομα στο οποίο δημιούργησα τις βασικές κλάσεις της εφαρμογής, ένα package με όνομα home στο οποίο δημιουργήθηκαν οι κλάσεις με τους Restful clients και ένα package με όνομα new package το οποίο περιέχει εικόνες και αρχεία που χρησιμοποιήθηκαν

Δημιουργία κλάσεων της εφαρμογής:

Για την εφαρμογή μας δημιουργήθηκαν 5 κλάσεις στο package Application.

- Η κλάση Application (Application.java)(Περιέχει το γραφικό κομμάτι της εφαρμογής και τις κατάλληλες μεθόδους που χρειάστηκαν.
- Η κλάση Login(login.java)(Περιέχει τη φόρμα login)
- Η κλάση MyValue(δίνει δικαίωμα πρόσβασης στο API των instances id,copy,user,date)
- Η κλάση OthersPaste(δίνει δικαίωμα πρόσβασης στο API των instances id,copy,user,date)
- Η κλάση User(δίνει δικαίωμα πρόσβασης στο API των instances name και online)

Για να γίνει η δημιουργία των κλάσεων στο package application έκανα δεξί κλικ και δημιουργία νέας κλάσης java και στη συνέχεια έγινε επιλογή του ονόματος(application) και της τοποθεσίας της κλάσης όπως φαίνεται στην εικόνα18.



Εικόνα 24: Δημιουργία κλάσης Application

Με τον ίδιο τρόπο δημιουργήθηκαν και οι υπόλοιπες 4 κλάσεις.

Γραφικό περιβάλλον(GUI) στην κλάση Login

Η κλάση login περιέχει μια φόρμα η οποία περιέχει ένα label για να μπορεί ο χρήστης να επιλεγεί την ip και ένα label για την θύρα που θέλει ώστε να συνδέεται με τον server. Επίσης μέσω αυτής της φόρμας μπορεί να βάλει το όνομα και το κωδικό του ώστε να συνδεθεί μέσω της βάσης δεδομένων και να έχει δικαίωμα χρήσης της εφαρμογής. Εάν δεν έχει κάνει εγγραφή στη βάση δεδομένων τότε μπορεί με το κουμπί create να εγγραφεί στη βάση.

Έχοντας λοιπόν δημιουργήσει την κλάση login δημιουργούμε ένα frame στο οποίο καταχωρούμε όλα τα αντικείμενα και τις ενέργειες που εκτελούν. Στην εικόνα 19 φαίνεται το αποτέλεσμα της κλάσης Login

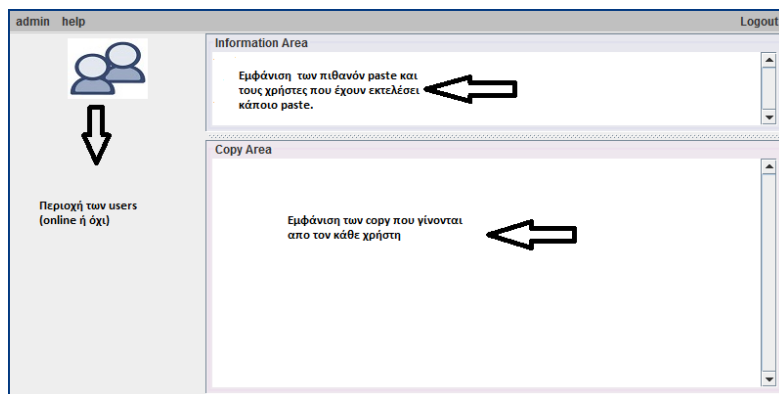
Εικόνα 25: Login form

Επίσης το κουμπί create μας παραπέμπει στην παρακάτω σελίδα (εικόνα 26)

Εικόνα 26: Create.php σελίδα

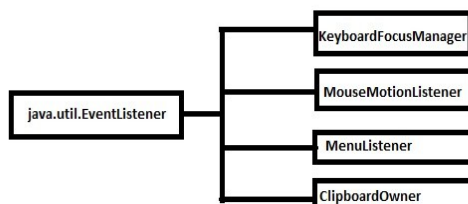
Γραφικό περιβάλλον(GUI) στην κλάση Application

Για την δημιουργία της εφαρμογής έγινε η κατασκευή ενός γραφικού περιβάλλοντος(GUI:Graphical User Interface) με χρήση της γλώσσας JAVA και του περιβάλλοντος NetBeans IDE. Στο συγκεκριμένο GUI έγινε διαχωρισμός σε panels ώστε να καθοριστεί η θέση των αντικειμένων που χρησιμοποιήσαμε. Χρειαζόμαστε ένα panel(panel1) στο οποίο έχει τοποθετηθεί ένα αντικείμενο(component) menubar το οποίο περιέχει τα αντικείμενα (admin,help,και logout).Το panel2 έχει δημιουργηθεί έτσι ώστε να εμφανίζονται οι χρήστες(users) που έχουν εγγραφεί στην εφαρμογή. Τέλος στο panel 3 έχει προστεθεί ένα splitpane component το οποίο περιέχει τα πιθανά paste που έχουν γίνει(Αριστερή θέση) και τα copy που έχουν εκτελεστεί από τους συνδεδεμένους χρήστες(Δεξιά θέση)



Εικόνα 27:GUI

Επιπλέον έχουν προστεθεί κάποια συμβάντα όταν ο χρήστης αλληλεπιδρά με το GUI.(Εικόνα 28)



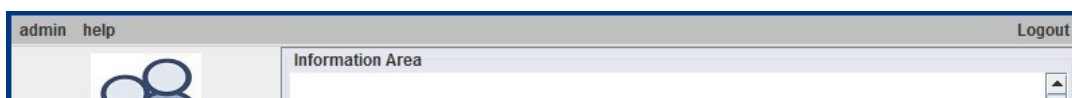
Εικόνα 28:Listeners

Αρχικά χρησιμοποιήθηκε ο MouseMotionListener ο οποίος δίνει κίνηση στο παράθυρο της εφαρμογής μέσω της κλάσης mouseDragged()→

```

@Override
public void mouseDragged(MouseEvent e) {
    frame.setLocation(frame.getX() + e.getX(), frame.getY() + e.getY());
}
  
```

Ο Menu Listener χρησιμοποιήθηκε για την εκτέλεση των στοιχείων που υπάρχουν στο menu(menuubar). Εικόνα 29



Εικόνα 29:Items in menubar

Στη συνέχεια έγινε χρήση του ClipboardOwner ώστε να δέχεται το copy που γίνεται από τον κάθε χρήστη.

```
public void gainOwnership(Transferable t) {
    systemClipboard.setContents(t, this);
}

public void setClipboardContents(String aString) {
    StringSelection stringSelection = new StringSelection(aString);
    Clipboard clipboard = Toolkit.getDefaultToolkit().getSystemClipboard();
    clipboard.setContents(stringSelection, this);
}

@Override
public void lostOwnership(Clipboard clipboard, Transferable contents) {
    try {
        this.sleep(200);
    } catch (InterruptedException e1) {
        e1.printStackTrace();
    }
    try {
        copy = (String) clipboard.getData(DataFlavor.stringFlavor);

        home.GetCopy gp = new GetCopy();
        gp.getText(username, copy);

    } catch (UnsupportedFlavorException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
    gainOwnership(contents);
}
```

Τέλος χρησιμοποιήθηκε ο KeyboardFocusManager ο οποίος καλεί μια μέθοδο ώστε να ενεργοποιείτε η εκτέλεση των πλήκτρων (esc=κλείσιμο της εφαρμογής, page down=ελαχιστοποίηση της εφαρμογής) από το πληκτρολόγιο.

```

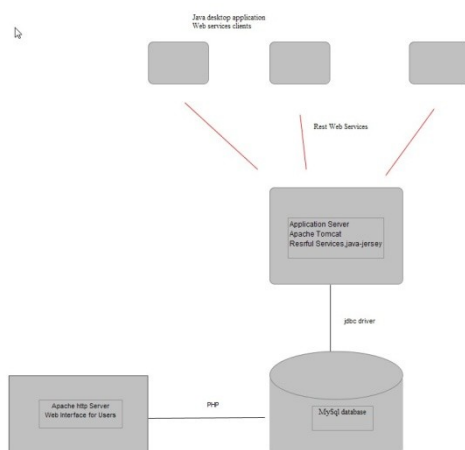
public boolean dispatchKeyEvent(KeyEvent e) {
    if (e.getID() == KeyEvent.KEY_PRESSED) {
        if (e.getKeyCode() == KeyEvent.VK_ESCAPE) {
            frame.dispose();//close the gui
            System.exit(0);//close the application

        } else if (e.getKeyCode() == KeyEvent.VK_PAGE_UP) {
            frame.setExtendedState(Frame.MAXIMIZED_BOTH);
        }
    } else if (e.getID() == KeyEvent.KEY_RELEASED) {
        if (e.getKeyCode() == KeyEvent.VK_PAGE_DOWN) {
            frame.setState(Frame.ICONIFIED);
        }
    } else if (e.getID() == KeyEvent.KEY_TYPED) {
        if (e.getKeyCode() == KeyEvent.VK_PAGE_UP) {
            frame.setExtendedState(Frame.MAXIMIZED_BOTH);
        }
    }
}

```

4.3 Κατασκευή Webservice

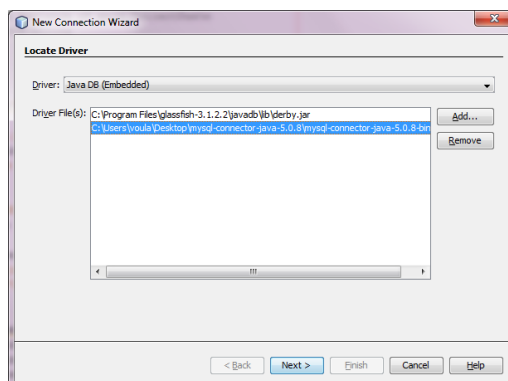
Στην παρακάτω εικόνα φαίνεται ένα σχεδιάγραμμα για την σύνδεση και την επικοινωνία των services ,της Βάσης Δεδομένων και του Apache(web Interface).



Εικόνα 30: Σχήμα επικοινωνίας(apache server, Mysql, Apache Tomcat Rest webServices)

Δημιουργία σύνδεσης jdbc driver

Αρχικά έγινε δημιουργία μιας νέας σύνδεσης μέσω του driver jdbc ώστε να πραγματοποιηθεί επικοινωνία της γλώσσας Java με την Βάση Δεδομένων μας (appbase).
Εικόνα 17



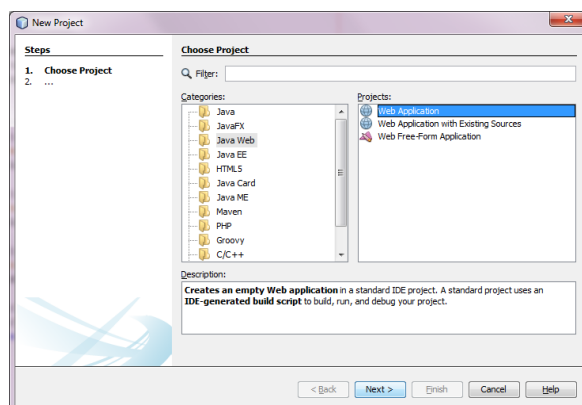
Εικόνα 31: Connection with jdbc

Δημιουργία server web service

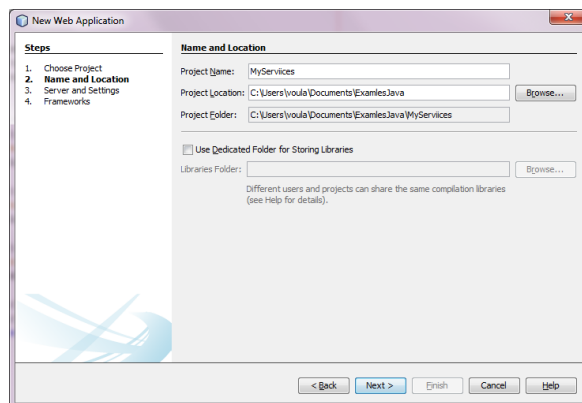
Το βασικό κομμάτι όλης της εφαρμογής ώστε να μπορούν οι χρήστες να επικοινωνήσουν με τον server είναι η κατασκευή του Server. Στη συγκεκριμένη εφαρμογή έγινε δημιουργία του server με χρήση των restfull services στις οποίες το NetBeans παρέχει έτοιμες βιβλιοθήκες χωρίς την προσθήκη κάποιου plug in.

Αρχικά δημιουργούμε ένα νέο project ως web εφαρμογή.

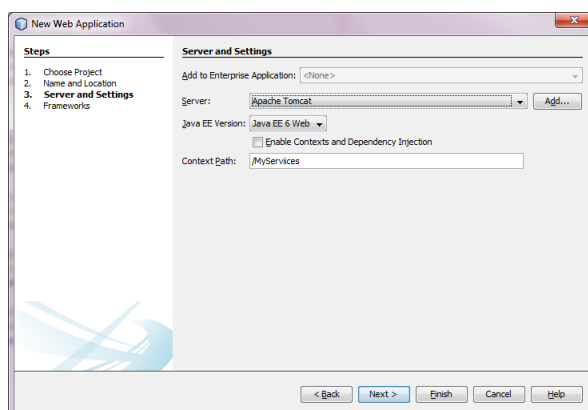
Πάμε File → New Project → Java Web (web application) → Ορίζουμε όνομα της εφαρμογής (MyServices) και στο επόμενο βήμα επιλεγούμε τον Apache Tomcat ως server. Η διαδικασία για την ολοκλήρωση μιας web εφαρμογής φαίνεται στις παρακάτω εικόνες.



Εικόνα 32: Δημιουργία web application βήμα 1



Εικόνα 33: Δημιουργία web application βήμα 2

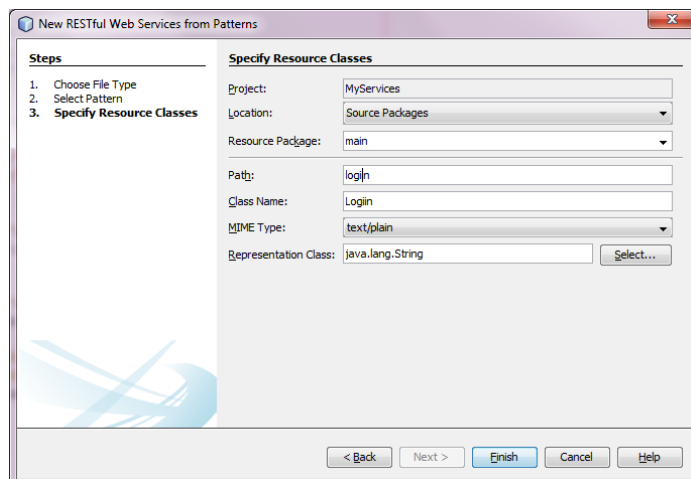


Εικόνα 34: Δημιουργία web application βήμα 3

Ετσι δημιουργείτε μια web εφαρμογή στην οποία δημιουργήθηκαν τα κατάλληλα Restful web services που χρειάζονται για την υλοποίηση της εφαρμογής. Συγκεκριμένα χρειάστηκε η δημιουργία του login (παίρνει τα username και passcode από τον πίνακα users ώστε να γίνει πραγματοποίηση login από το χρήστη), Logout(Για την απενεργοποίηση των χρηστών από την εφαρμογή), GetCopy (καταχωρεί τα copy και τους users στη βάση), GetPaste (επιστέφει στους users το αποτέλεσμα από getCopy), GetUsers, ShowPaste, CollectPaste.

Παράδειγμα δημιουργίας του restful service για login στη βάση δεδομένων(όνομα login)

Αφού έχουμε δημιουργήσει μια web application πατάμε δεξί κλικ στο project →New (RESTful Web Services From Patterns)→Next→και έπειτα βάζουμε ως path το όνομα login(όνομα URI για την πρόσβαση στη web εφαρμογή ,ως class name το Login και επιλέγουμε text/plain τον τύπο των MIME Type. (Εικόνα 35)



Εικόνα 35: Restful web Services from patterns

Παρακάτω φαίνεται η κλάση login(Εικόνα 36) και η μέθοδος get(Εικόνα 37) που χρησιμοποιήσαμε για την ανάκτηση των πεδίων username και passcode από την βάση δεδομένων (arphbase). Με την ίδια λογική δημιουργήθηκαν και τα υπόλοιπα webservice ανάλογα με τις ενέργειες που θα κάνει το καθένα.

```

package main;

import javax.ws.rs.core.Context;
import javax.ws.rs.core.UriInfo;
import javax.ws.rs.PathParam;
import javax.ws.rs.Consumes;
import javax.ws.rs.PUT;
import javax.ws.rs.Path;
import javax.ws.rs.GET;
import javax.ws.rs.Produces;

/**
 * REST Web Service
 *
 * @author voula
 */
@Path("login")
public class Login {
    @Context
    private UriInfo context;

    /**
     * Creates a new instance of Loginn
     */
    public Login () {
    }

    /**
     * Retrieves representation of an instance of main.Loginn
     * @return an instance of java.lang.String
     */
    @GET
    @Produces("text/plain")
    public String getText() {
        //TODO return proper representation object
        throw new UnsupportedOperationException();
    }

    /**
     * PUT method for updating or creating an instance of Loginn
     * @param content representation for the resource
     * @return an HTTP response with content of the updated or created resource
     */
    @PUT
    @Consumes("text/plain")
    public void putText(String content) {
    }
}

```

Εικόνα 36 :Login service

```

@GET
@Produces("text/plain")
@Path("/{username}/{password}")
public String login(@PathParam("username") String username, @PathParam("password") String password) {
    try {
        System.out.println(username + " " + password);
        Class.forName("com.mysql.jdbc.Driver");// enabling a Java application to interact with a database
    } catch (ClassNotFoundException e) {
        System.out.println("Where is your MySQL JDBC Driver?");
        e.printStackTrace();
    }
    boolean loginResult = false;
    System.out.println("MySQL JDBC Driver Registered!");
    // create a connection for the database
    Connection connection = null;
    try {
        // take the connection from jdbc --address of the database
        connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/appbase?zeroDateTimeBehavior=convertToNull", "root", "");
        String select = ("SELECT 1 FROM users where username = ? AND passcode = ?");// sql string
        PreparedStatement selectSQL = connection.prepareStatement(select);// give to the connection the sql statement
        // set the sql parameters
        selectSQL.setString(1, username);
        selectSQL.setString(2, password);
        ResultSet results = selectSQL.executeQuery(); // execute sql query and take the results
        while (results.next()) { // process results one row at a time
            loginResult = true;
        }
        selectSQL.close();// dont forget to close the statement
        if (loginResult) { // if the login succeeded
            // update the user to be online
            String insertTableSQL = "UPDATE users SET ONLINE = 1 WHERE USERNAME = ?";
            // prepare the statement
            PreparedStatement preparedStatement = connection.prepareStatement(insertTableSQL);
            preparedStatement.setString(1, username);
            preparedStatement.executeUpdate();
            preparedStatement.close();
        }
        connection.close(); // close connection at last
    } catch (SQLException e) {
        System.out.println("Connection Failed! Check output console");
        e.printStackTrace();
    }
    if (loginResult) {
        return ("login ok");
    } else {
        return ("Failed to make connection!");
    }
}

```

Εικόνα 37: Μέθοδος Get της κλάσης Login

```

@GET
@Produces("text/plain")
@Path("/{username}/{password}")
public String login(@PathParam("username") String username, @PathParam("password") String password) {
    try {
        System.out.println(username + " " + password);
        Class.forName("com.mysql.jdbc.Driver");// enabling a Java application to interact with a database
    } catch (ClassNotFoundException e) {
        System.out.println("Where is your MySQL JDBC Driver?");
        e.printStackTrace();
    }
    boolean loginResult = false;
    System.out.println("MySQL JDBC Driver Registered!");
    // create a connection for the database
    Connection connection = null;
    try {
        // take the connection from jdbc --address of the database
        connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/appbase?zeroDateTimeBehavior=convertToNull", "root", "");
        String select = ("SELECT 1 FROM users where username = ? AND passcode = ?");// sql string
        PreparedStatement selectSQL = connection.prepareStatement(select);// give to the connection the sql statement
        // set the sql parameters
        selectSQL.setString(1, username);
        selectSQL.setString(2, password);
        ResultSet results = selectSQL.executeQuery(); // execute sql query and take the results
        while (results.next()) { // process results one row at a time
            loginResult = true;
        }
        selectSQL.close();// dont forget to close the statement
        if (loginResult) { // if the login succeeded
            // update the user to be online
            String insertTableSQL = "UPDATE users SET ONLINE = 1 WHERE USERNAME = ?";
            // prepare the statement
            PreparedStatement preparedStatement = connection.prepareStatement(insertTableSQL);
            preparedStatement.setString(1, username);
            preparedStatement.executeUpdate();
            preparedStatement.close();
        }
        connection.close(); // close connection at last
    } catch (SQLException e) {
        System.out.println("Connection Failed! Check output console");
        e.printStackTrace();
    }
    if (loginResult) {
        return ("login ok");
    } else {
        return ("Failed to make connection!");
    }
}

```

Εικόνα 38: Μέθοδος Get της κλάσης Logout

Δημιουργία Restful Java Client

Στη συνέχεια έγινε η δημιουργία Restful services ώστε να γίνεται η επικοινωνία των clients με τον server. Στο project με όνομα Application στο package home (όπως αναφέρθηκε παραπάνω) δημιούργησα όσες κλάσεις έχουν δημιουργηθεί στον server ώστε να γίνεται επικοινωνία μεταξύ τους

Παράδειγμα δημιουργίας java client restful στη κλάση του login του server

Σε ένα διαφορετικό package με όνομα (clientservice) πατάμε δεξί κλικ→ RESTful Java Client→Γράφουμε όνομα το όνομα της κλάσης(Login) και επιλέγουμε από το select REST Resources το MyService→Login

```
import javax.ws.rs.ClientErrorException;
import javax.ws.rs.client.Client;
import javax.ws.rs.client.WebTarget;

/**
 * Jersey REST client generated for REST resource:Login [login]<br>
 * USAGE:
 * <pre>
 *   Login client = new Login();
 *   Object response = client.XXX(...);
 *   // do whatever with response
 *   client.close();
 * </pre>
 * @author voula
 */
public class Login {

    private WebTarget webTarget;
    private Client client;
    private static final String BASE_URI = "http://localhost:8080/MyServices/webresources";

    public Login() {

        client = javax.ws.rs.client.ClientBuilder.newClient();
        webTarget = client.target(BASE_URI).path("login");
    }

    public void putText(Object requestEntity) throws ClientErrorException {
        webTarget.request(javax.ws.rs.core.MediaType.TEXT_PLAIN).put(javax.ws.rs.client.Entity.entity(requestEntity, javax.ws.rs.core.MediaType.TEXT_PLAIN));
    }

    public String login(String username, String password) throws ClientErrorException {
        WebTarget resource = webTarget;
        resource = resource.path(java.text.MessageFormat.format("/{0}/{1}", new Object[] {username, password}));
        return resource.request(javax.ws.rs.core.MediaType.TEXT_PLAIN).get(String.class);
    }

    public void close() {
        client.close();
    }
}

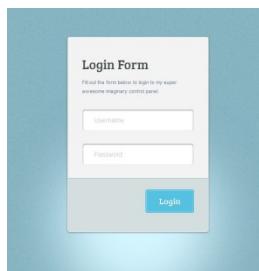
```

Εικόνα 39 :RESTFull Java Client

Η ίδια διαδικασία επαναλήφθηκε για όλα τα RESTFull Java Client

4.4 Δημιουργία Web Interface(ιστοσελίδας)

Αρχικά κατέβασα μια έτοιμη πλατφόρμα html5 και css3 όπου έγινε επεξεργασία της σελίδας και των css με χρήση του NotePad ++



Εικόνα 40: Web Interface

Στη συνέχεια δημιούργησα το φάκελο που θα φιλοξενήσει το Site το οποίο ονόμασα **webApplication** και ένα φάκελο **“Images”** όπου θα υπάρχουν οι εικόνες του web Interface.

Έχοντας το Notepad ++ ανοιχτό έγινε η δημιουργία μιας νέας σελίδας με όνομα crud_processing.html στην οποία αρχικά έγραψα τα βασικά στοιχεία για την δημιουργία μιας Ιστοσελίδας. Επίσης με το link → `link rel="stylesheet" href="stylesheet.css">` του δίνουμε μορφοποίηση από το ίδιο υπάρχον αρχείο css με όνομα stylesheet.

```
<!doctype html>
<head>
  <meta charset="utf-8">
  <title>Our First HTML5 Page</title>
  <meta name="description" content="Welcome to my basic template.">
  <link rel="stylesheet" href="stylesheet.css">
</head>

<body>
</body>
</html?>
```

Ανάλυση βασικών css που χρησιμοποιήθηκαν:

Για την σωστή τοποθέτηση των στοιχείων της σελίδας μας χρησιμοποιήθηκε ένα ευρύ φάσμα css. Ένα βασικό css style που χρησιμοποιήθηκε είναι τι χρώμα και μέγεθος θα έχει η σελίδα μας και η τοποθέτηση της φόρμας.

```
body {
    background: url('http://www.demo.amitjakhu.com/login-form/images/bg.png');
    font-family: "HelveticaNeue-Light", "Helvetica Neue Light", "Helvetica Neue",
    Helvetica, Arial, "Lucida Grande", sans-serif;
    font-weight:300;
    text-align: left;
    text-decoration: none;
    height: 500px;
}
#wrapper {
    /* Center wrapper perfectly */
    width: 300px;
    height: 400px;
    margin: 70px auto;
}
```

Το css με όνομα body χρησιμοποιείται σε όλη τη σελίδα. Για το τμήμα Body, χρησιμοποιούμε γραμματοσειρά HelveticaNeue-Light, αν δεν την έχει το σύστημα που τρέχει το Site, την Helvetica Neue Light και τέλος αν συμβεί το ίδιο τις υπόλοιπες(Helvetica Neue",

Helvetica, Arial, "Lucida Grande", sans-serif). Του λέμε το font-weight να είναι μεγέθους 300 και το ύψος(height:500 px). Το βασικό χρώμα του background είναι μια εικόνα με απόκριση χρώματος μπλε. Το #wrapper χρησιμοποιείται συγκεκριμένο element,την φόρμα login με πλάτος 300px και ύψος 400px και με margin 70px δηλαδή έγινε μετακίνηση της φόρμας κατά 70px προς τα δεξιά. Οπότε βάζουμε πριν την δημιουργία της φόρμας στο login αρχείο το <div id="wrapper"> </div>. Το ίδιο εκτελούμε για όλα τα css που θα χρησιμοποιήσουμε.

Επίσης το παρακάτω το .login-form δημιουργεί ένα box για τη φόρμα. Το πλάτος θα είναι 300px και η θέση ως relative σημαίνει ότι όλα τα στοιχεία θα είναι στη θέση αυτή. Το χρώμα της φόρμας έχει οριστεί λευκό.

```
.login-form {
    width: 300px;
    margin: 0 auto;
    position: relative;

    background: #f3f3f3;
    border: 1px solid #fff;
    border-radius: 5px;

    box-shadow: 0 1px 3px rgba(0,0,0,0.5);
    -moz-box-shadow: 0 1px 3px rgba(0,0,0,0.5);
    -webkit-box-shadow: 0 1px 3px rgba(0,0,0,0.5);
}
```

Login Form

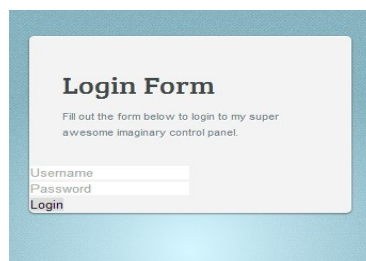
Fill out the form below to login to my super awesome imaginary control panel.

Username

 Password

 Login

Πριν τη χρήση συγκεκριμένων cuss



The image shows the same login form as above, but styled with CSS. The form is centered within a light blue gradient box. The text is bold and black. The input fields are white with a light gray border. The login button is a small, dark square with the word 'Login' in white.

Μετά τη χρήση συγκεκριμένων cuss

Στη συνέχεια η κλάση `.content { padding: 0 30px 25px 30px;}` καθορίζει το διάστημα ανάμεσα στο `element border` and the `element content`. Πιο συγκεκριμένα έχει οριστεί ως `.login-form .content`. Επίσης χρησιμοποιήθηκε η κλάση `.login-form .content .input` η οποία δημιουργεί τα `boxes` των `username` και `password`. Το διάστημα ανάμεσα στο `element border` and the `element content` είναι `15px` και `25px`.

```
.login-form .content .input {
  width: 188px;
  padding: 15px 25px;

  font-family: "HelveticaNeue-Light", "Helvetica Neue Light", "Helvetica Neue", Helvetica, Arial, "Lucida Grande", sans-serif;
  font-weight: 400;
  font-size: 14px;
  color: #9d9e9e;
  text-shadow: 1px 1px 0 rgba(256,256,256,1.0);

  background: #fff;
  border: 1px solid #fff;
  border-radius: 5px;

  box-shadow: inset 0 1px 3px rgba(0,0,0,0.50);
  -moz-box-shadow: inset 0 1px 3px rgba(0,0,0,0.50);
  -webkit-box-shadow: inset 0 1px 3px rgba(0,0,0,0.50);
}
```

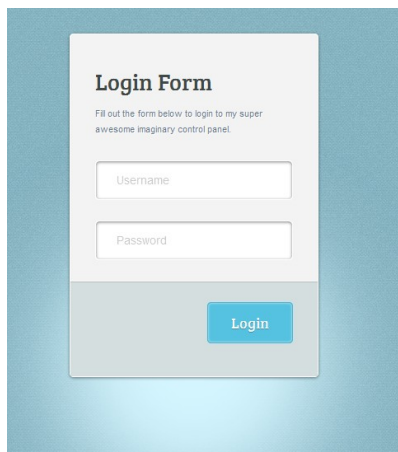
Οπότε σύμφωνα με το παραπάνω `style sheet` η φόρμα που δημιουργείτε φαίνεται στην παρακάτω εικόνα

Επιπλέον προστέθηκαν επιπλέον `css` για τη δημιουργία και στοίχιση των κουμπιών και την τοποθέτηση των εικόνων και άλλα. Τα ίδια σχεδόν `css` χρησιμοποιήθηκαν και στις άλλες σελίδες που έχουν δημιουργηθεί.

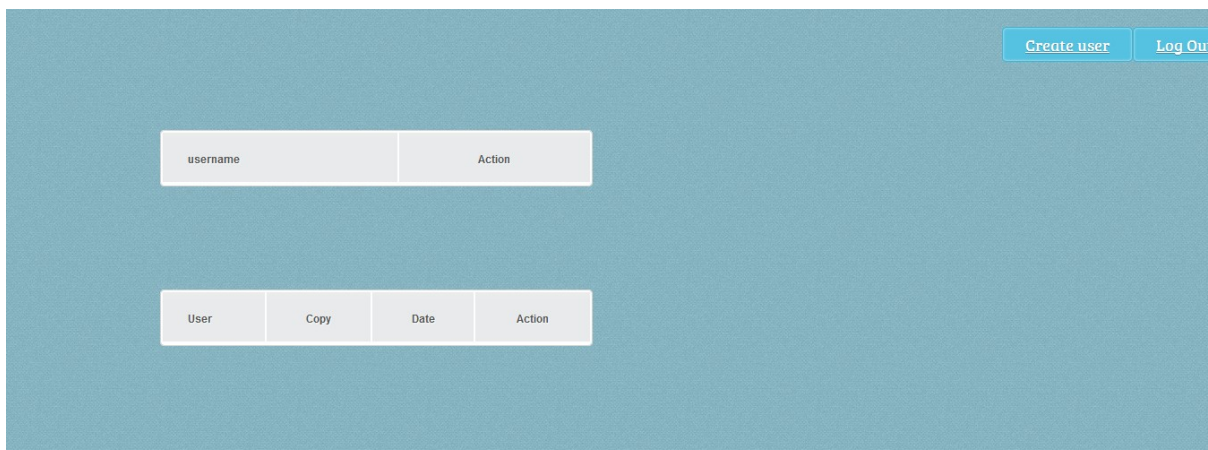
Έχοντας παρουσιάσει τα σημαντικότερα `css` παρακάτω φαίνεται η δημιουργία `html` κώδικα την φόρμας και την τοποθέτηση κάποιων `css` στη σελίδα `login.html`

```
<body>
<div id="wrapper">
  <form name="login-form" class="login-form" method="post">
    <div class="header">
      <h1>Login Form</h1>
      <span>Fill out the form below to login to my super awesome imaginary control panel.</span>
    </div>
    <div class="content">
      <input name="username" type="text" class="input username" placeholder="Username" />
      <div class="user-icon"></div>
      <input name="password" type="password" class="input password" placeholder="Password" />
      <div class="pass-icon"></div>
    </div>
    <div class="footer">
      <input type="submit" name="submit" value="Login" class="button" />
    </div>
  </form>
</div>
```

Με την κατάλληλη τοποθέτηση των css οι σελίδες μας τροποποιήθηκαν όπως φαίνεται στις εικόνες που ακολουθούν



Εικόνα 41: login.html σελίδα

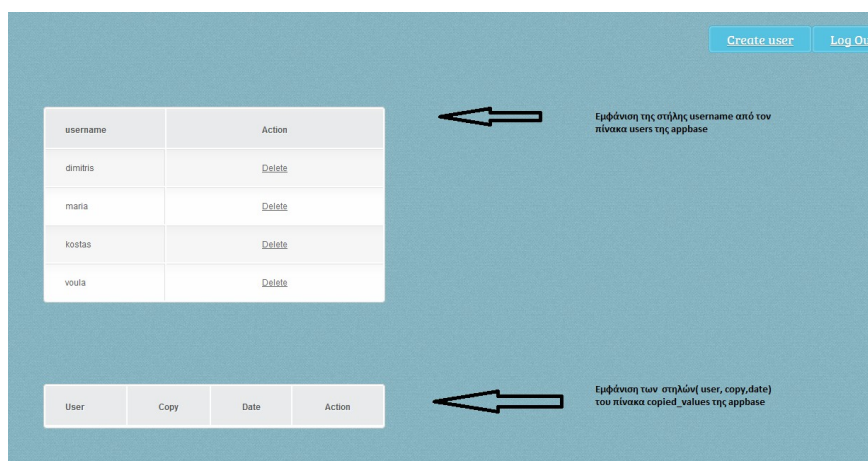


Εικόνα 42: crud_processing.html σελίδα

Χρήση γλώσσας php

Έπειτα για να γίνει σύνδεση με την βάση δεδομένων χρησιμοποίησα τον Apache http server και τη τεχνολογία php. Για να γίνει μετατροπή website από html σε php έπρεπε να γίνει μετατροπή του αρχείου μας login.html σε login.php και του αρχείου crud_processing.html σε crud_processing.php. Επίσης μετέφερα τον φάκελο webApplication στο path //htdocs του server .

Σύμφωνα με την τεχνολογία php η crud_processing σελίδα περιέχει τα στοιχεία των πινάκων users και pasted_values από την βάση δεδομένων ώστε να μπορεί να γίνει επεξεργασία και δημιουργία χρηστών.



Εικόνα 43 :Σελίδα crud_processing.php

Η υλοποίηση και σύνδεση των δύο σελίδων (login.php και crud_processing) έγινε ως εξής:

Σελίδα login.php

Για την σελίδα login.php χρειάζονται 2php αρχεία το config.php και το lock.php, το Το αρχείο →config.php περιέχει τον κώδικα όπου γίνεται η σύνδεση στη βάση μου(appbase). Το αρχείο →lock.php ελέγχει το password και το username και αν είναι σωστά μας συνδέει με τη σελίδα crud_processing.php όπου μπορεί να γίνει επεξεργασία χρηστών

```

Config.php file
<?php
    $mysql_hostname = "localhost";
    $mysql_user = "root";
    $mysql_password = "";
    $mysql_database = "appbase";
    $bd = mysql_connect($mysql_hostname, $mysql_user, $mysql_password)

    or die("Opps some thing went wrong");
    mysql_select_db($mysql_database, $bd) or die("Opps some thing went wrong");
?>

```

```

Lock.php file
<?php
    include('config.php');
    session_start();
    $user_check=$_SESSION['login_user'];

    $ses_sql=mysql_query("select      username      from      appbase      where
username='$user_check' ");

    $row=mysql_fetch_array($ses_sql);

    $login_session=$row['username'];

    if(!isset($login_session))
    {
        header("Location: crud_processing.php");
    }
?>

```

Σελίδα crud_processing.php

Για την σελίδα crud_processing.php χρειάστηκαν 2 php αρχεία το database.php ,delete.php, delete1.php, create.php, logout.php. Το αρχείο database το παίρνουμε με την εντολή include όπως και όλα τα αρχεία της php. Το οποίο κάνει έλεγχο για την βάση και αν υπάρχει μας συνδέει με τους πίνακες users και copied values ώστε να μας εμφανίσει όλες τις εγγραφές που έχουν καταγραφή στη βάση(appbase). Στη συνέχεια σε κάθε πίνακα με το include προσθέτουμε τα αρχεία delete.php για τον πίνακα users και delete1.php για τον πίνακα copied_values αντίστοιχα ώστε να μπορεί να γίνει η διαγραφή κάθε εγγραφής του πίνακα. Τέλος στη συγκεκριμένη σελίδα υπάρχει δυο κουμπιά, το κουμπί create το οποίο μας παραπέμπει στη σελίδα create.php στην οποία μπορεί να γίνει δημιουργία χρηστών και το κουμπί logout που γίνεται αποσύνδεση από την βάση.

```

Database.php
<?php
class Database
{
    private static $dbName = 'appbase' ;
    private static $dbHost = 'localhost' ;
    private static $dbUsername = 'root';
    private static $dbUserPassword = "";
    private static $cont = null;
    public function __construct() {
        exit("Init function is not allowed");
    }
    public static function connect()
    {
        // One connection through whole application
        if ( null == self::$cont )
        {
            try
            {
                self::$cont = new PDO( "mysql:host=".self::$dbHost.";dbname=".self::$dbName,
                self::$dbUsername, self::$dbUserPassword);
            }
            catch(PDOException $e)
            {
                die($e->getMessage());
            }
        }
        return self::$cont;
    }

    public static function disconnect()
    {
        self::$cont = null;
    }
}

```

Delete.php file

```
<?php
    require 'database.php';
    $id = 0;

    if ( !empty($_GET['id'])) {
        $id = $_REQUEST['id'];
    }

    if ( !empty($_POST)) {
        // keep track post values
        $id = $_POST['id'];

        // delete data
        $pdo = Database::connect();
        $pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
        $sql = "DELETE FROM users WHERE id = ?";
        $q = $pdo->prepare($sql);
        $q->execute(array($id));
        Database::disconnect();
        header("Location: crud_processing.php");

    }
?>
```

Logout.php file

```
<?php
    session_start();
    if(session_destroy())
    {
        header("Location: login.php");
    }
?>
```

Create.php file

```
<?php
require 'database.php';

if ( !empty($_POST) ) {

    // keep track validation errors
    $usernameError = null;
    $passcodeError = null;

    // keep track post values
    $username = $_POST['username'];
    $passcode = $_POST['passcode'];
    // validate input
    $valid = true;
    if (empty($username)) {
        $usernameError = 'Please enter Name';
        $valid = false;
    }

    if (empty($passcode)) {
        $passcodeError = 'Please enter passcode';
        $valid = false;
    }

    // insert data
    if ($valid) {
        $pdo = Database::connect();
        $pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
        $sql = "INSERT INTO users(username,passcode)
values(?, ?)";

        $q = $pdo->prepare($sql);
        $q->execute(array($username,$passcode));
        Database::disconnect();
        header("Location: crud_processing.php");
    }
}

?>
```

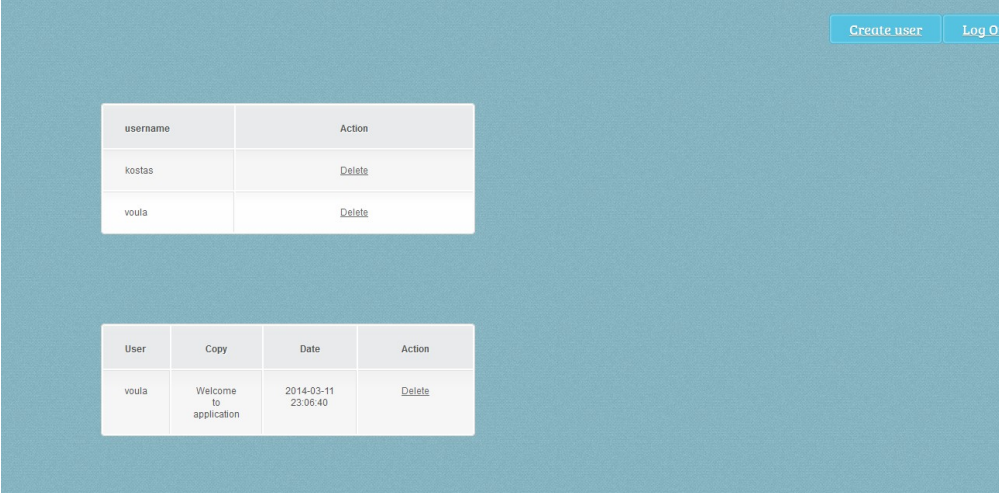

Έχοντας παρουσιάσει την δημιουργία σελίδων(δημιουργία φόρμας και τοποθέτηση css) και τη βασική δομή της γλώσσας php στο παρακάτω πλαίσιο φαίνεται η δημιουργία ενός πίνακα στο αρχείο crud_processing.php ο οποίος περιέχει στη στήλη με όνομα username και τη στήλη με όνομα action. Με χρήση της PHP κάνουμε σύνδεση με την βάση δεδομένων(include 'database.php') και παίρνουμε όλα τα στοιχεία του πίνακα users(\$sql = 'SELECT * FROM users ORDER BY id DESC'). Συγκεκριμένα εδώ λέμε να εμφανίσει μόνο τις εγγραφές του πεδίου username του πίνακα users

```

<body>
    <p><a href="logout.php" class="button"> Log Out</a></p>
    <p><a href="create.php" class="button"> Create user</a> </p>
    <div id="content">
        <table>
            <thead>
                <tr>
                    <th>username</th>
                    <th>Action</th>
                </tr>
            </thead>
            <tbody>
                <?php
                    include 'database.php';
                    $pdo = Database::connect();
                    $sql = 'SELECT * FROM users ORDER BY id DESC';
                    foreach ($pdo->query($sql) as $row) {
                        echo '<tr>';
                        echo '<td>'. $row['username'] . '</td>';
                        echo '<td width=250>';
                        echo '<a class="btn btn-danger"
href="delete.php?id='.$row['id'].'">Delete</a>';
                        echo '</td>';
                        echo '</tr>';
                    }
                    Database::disconnect();
                ?>
            </tbody>
        </table>
    </div>

```

Με την ολοκλήρωση του κατάλληλου κώδικα php στην crud_processing η σελίδα θα συνδέεται με την βάση και θα εμφανίζει τους χρήστες που έχουν εγγραφεί και τα copy που έχουν γίνει (Εικόνα 38)



The screenshot shows a web application interface with a teal background. In the top right corner, there are two buttons: "Create user" and "Log Out". Below these buttons, there are two tables. The first table lists users with their usernames and delete actions. The second table lists copies with columns for User, Copy, Date, and Action.

username	Action
kostas	Delete
voula	Delete

User	Copy	Date	Action
voula	Welcome to application	2014-03-11 23:06:40	Delete

Εικόνα 44: Αποτελέσματα crud_processing.php

Στη συνέχεια το αρχείο create.php και delete.php εκτελούν τη δημιουργία/διαγραφή χρηστών και τη διαγραφή των copy που έχουν γίνει.

5. Ολοκλήρωση έργου

Σύμφωνα με την επιτυχή υλοποίηση της εφαρμογής copy paste μέσω id παρακάτω περιγράφο πώς τρέχουμε την εφαρμογής μας και πως λειτουργεί.

Αρχικά πρέπει να υπάρχει ο server μας σε έναν υπολογιστή ο οποίος θα τρέχει συνέχεια ώστε να μπορούν οι clients να επικοινωνήσουν μαζί τους. Οπότε χρησιμοποιούμε έναν υπολογιστή στον οποίο έχουμε εγκαταστήσει όλα τα απαραίτητα εργαλεία (apache tomcat, Apache, jdbc) που χρειάζονται ώστε να γίνει επικοινωνία στον server.

Στη συνέχεια παίρνουμε μόνο την εφαρμογή από τα αρχεία του NetBeans και την κάνουμε επικόλληση σε όποιον χρήστη θέλει να λειτουργήσει την εφαρμογή. Ανοίγουμε με την εντολή cmd τη γραμμή εντολών και εκτελούμε μόνο το αρχείο jar από τον φάκελο dist της εφαρμογής. Πιο αναλυτικά:

Ανοίγουμε τον φάκελο Application → dist → και κάνουμε drop and drag το αρχείο στη γραμμή εντολών πατάμε enter και τότε θα μας εμφανίσει το παρακάτω παράθυρο (Εικόνα 45):

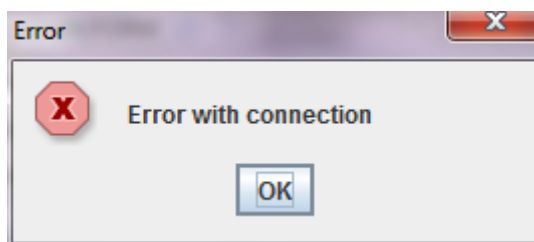
!Προσοχή αν βγάζει σφάλμα τότε δεν μπορεί να διαβάσει τα αρχεία jar. Τότε κάνετε τα εξής βήματα: Δεξί κλικ στον εικονίδιο υπολογιστής μου → Ρυθμίσεις συστήματος για προχωρημένους → Για προχωρημένους → Μεταβλητές περιβάλλοντος → και στη μεταβλητή συστήματος στο path πατάμε επεξεργασία και βάζουμε ένα ερωτηματικό στο τέλος ώστε να βάλουμε το path της java (C:\Program Files\Java\jre7) πατάμε ok και στη συνέχεια μπορούμε να ξανατρέξουμε την εφαρμογή μας.



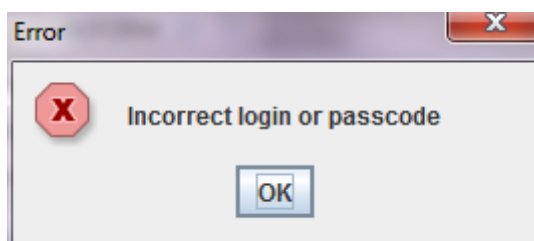
Server IP : Port	10.0.24.18	: 8081
Username:	voula	
Passcode:	•••	
SUBMIT		CREATE

Εικόνα 45: Login Form (Java)

Οι χρήστες έχουν τη δυνατότητα να επιλέγουν την ip και την θύρα του server με τον οποίο θα συνδεθούν. Στη συνέχεια με βάση του ονόματος τους και τον κωδικό τους να κάνουν login. Αν βάλουν μια διαφορετική ip ή τα στοιχεία τους δε υπάρχουν, τότε τους εμφανίζεται ένα παραθυράκι λάθους για κάθε περίπτωση και δεν έχουν δυνατότητα σύνδεσης στην εφαρμογή. Επίσης μπορούν να εγγραφούν στη Βάση με το κουμπί create.

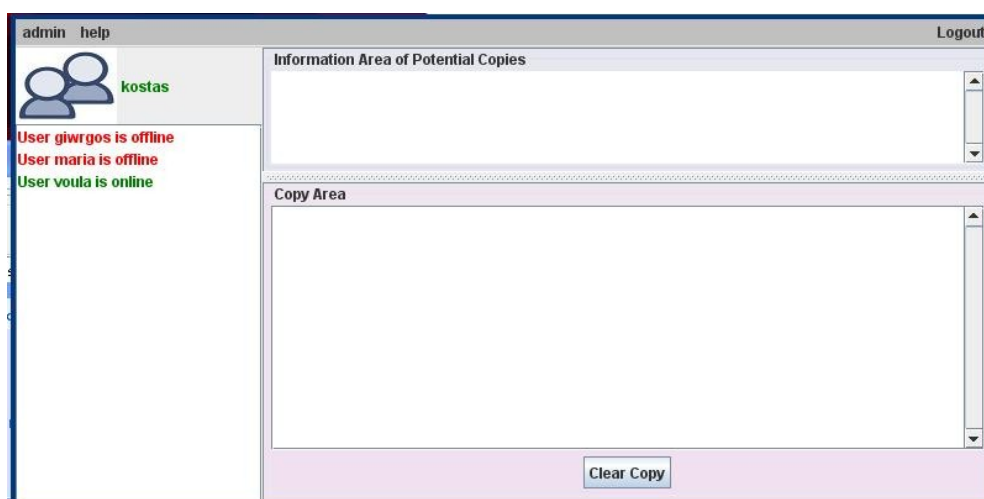


Εικόνα 46: Λάθος ip/port



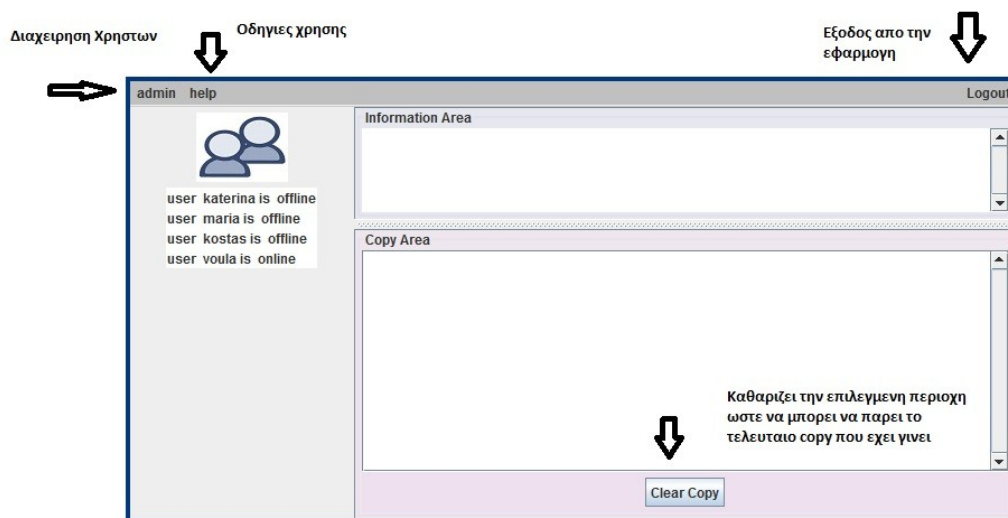
Εικόνα 47: Λάθος όνομα/κωδικός

Όταν ο χρήστης κάνει login τότε έχουν δικαίωμα συμμετοχής στην εφαρμογή copy/paste με τους χρηστές που υπάρχουν στη λίστα(Εικόνα 48)

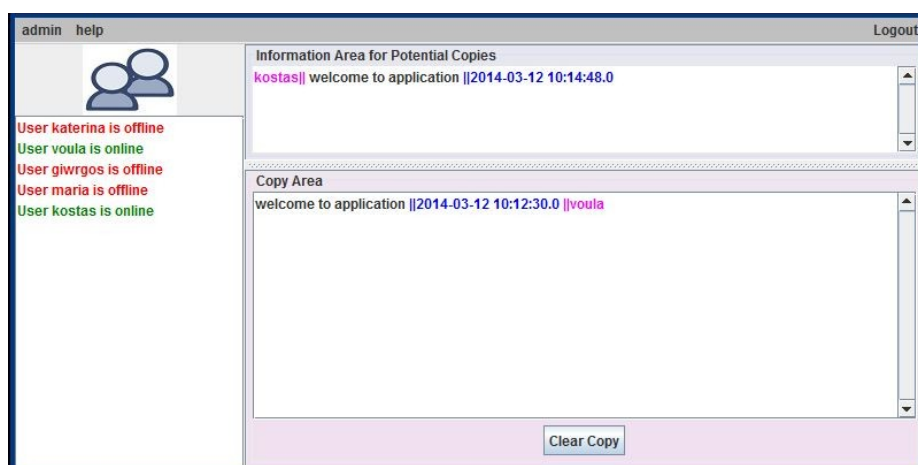


Εικόνα 48: Window χρήστη

Στο συγκεκριμένο παράθυρο ο χρήστης έχει δικαίωμα με το κουμπί app να κάνει διαχείριση χρηστών, με το κουμπί help να δει πως λειτουργεί η εφαρμογή και με το κουμπί logout να κάνει αποσύνδεση.

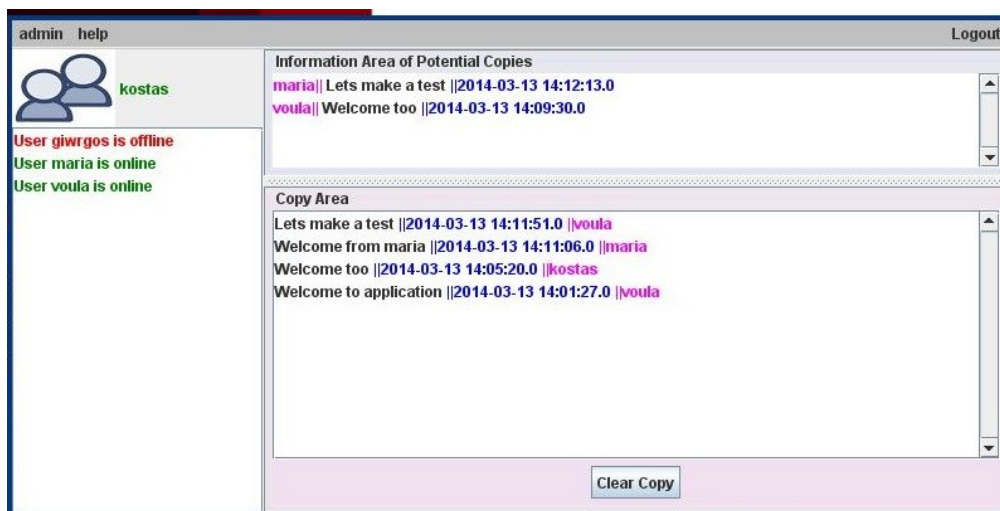


Στη συνέχεια κάνει αντιγραφή ένα κείμενο με `ctrl+c`. Για παράδειγμα κάνει αντιγραφή το κείμενο `welcome to application` και χωρίς να κάνει κάτι ο άλλος χρήστης που έχει συνδεθεί μπορεί να το πάρει με ένα `ctrl+v`.



Επιπλέον με το `ctrl (down)` θα μπορεί να πάρει μια σειρά από τα `copy` που έχουν γίνει και με το `shift(down)` θα μπορεί να παίρνει μια ροή από τα `copies` που υπάρχουν στην εφαρμογή.

Στην παρακάτω εικόνα φαίνεται η σύνδεση 3 χρηστών στην οποία ο χρήστης Kostas παίρνει με το `ctrl+v` το τελευταίο `copy` που έχει εκτελεστεί από το χρήστη voula



!Τα test της εφαρμογής πραγματοποιήθηκαν στο εργαστήριο Πολυμέσων (media) του ΤΕΙ Κρήτης

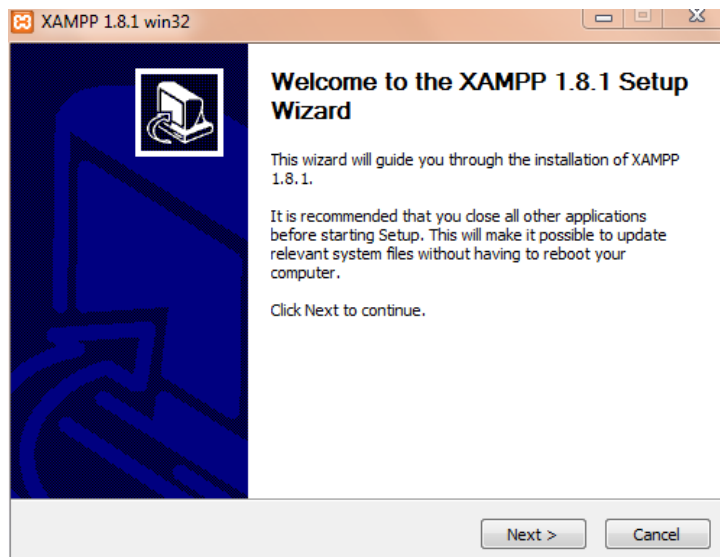
Συμπεράσματα

Με την ολοκλήρωση του έργου αυτού, βλέπουμε ένα νέο παράθυρο ευκαιριών μέσα από την “νεαρή” αλλά πολλά υποσχόμενη τεχνολογία. Μπορεί μελλοντικά ο κόσμος να στραφεί στην πιο γρήγορη και εύκολη μεταφορά κειμένου αφού μέσω των webs services και άλλων τεχνολογιών υπάρχει δυνατότητα ανάπτυξης στον τομέα της επικοινωνίας και μεταφοράς δεδομένων. Με την ολοκλήρωση της συγκεκριμένης εφαρμογής καταλήξαμε ότι υπήρχαν κάποια προβλήματα. Ένα από αυτά ήταν ότι δεν μπορούσαμε να κάνουμε εύκολη χρήση του πληκτρολογίου. Η java μπορεί να ακούσει το πληκτρολόγιο μόνο αν αλληλεπιδρά με το συγκεκριμένο γραφικό περιβάλλον που χρησιμοποιεί. Έτσι δεν μπορούσαμε να κάνουμε αντικατάσταση άλλων πλήκτρων με τα πλήκτρα `ctr+c` και `ctr+v` αφού δεν θα είχαν αποτέλεσμα μόνο εάν ήταν ενεργό το παράθυρο της εφαρμογής. Έτσι ήταν εξίσου δύσκολο να μπορούμε να πάρουμε πίσω το `copy` που έχει γίνει αφού δεν μπορούσαμε να πιάσει σε ποιο αντικείμενο θα γίνεται το `paste`. Με βάση αυτά τα προβλήματα καταλήξαμε ότι έπρεπε να μετατραπεί όλη η εφαρμογή μας σε web service ή να γίνει χρήση άλλης γλώσσας προγραμματισμού όπως η dotnet. Τέλος όπως έχει αναφερθεί η συγκεκριμένη εφαρμογή δίνει τη δυνατότητα στους χρήστες να κάνουν αντιγραφή και επικόλληση μόνο κείμενο έτσι οι δυνατότητες εξέλιξης της εφαρμογής θα μπορούσε να είναι η μεταφορά άλλων τύπων δεδομένων όπως μεταφορά εικόνων και αρχείων.

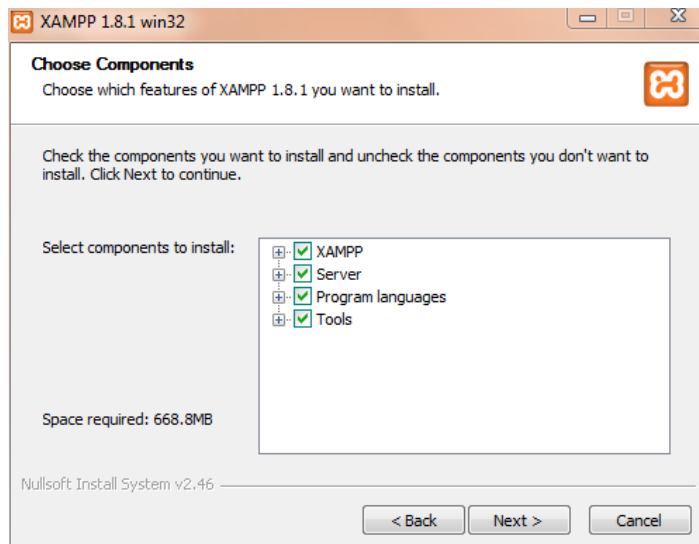
Παράρτηματα

Παράρτημα1:Εγκατάσταση του Xampp

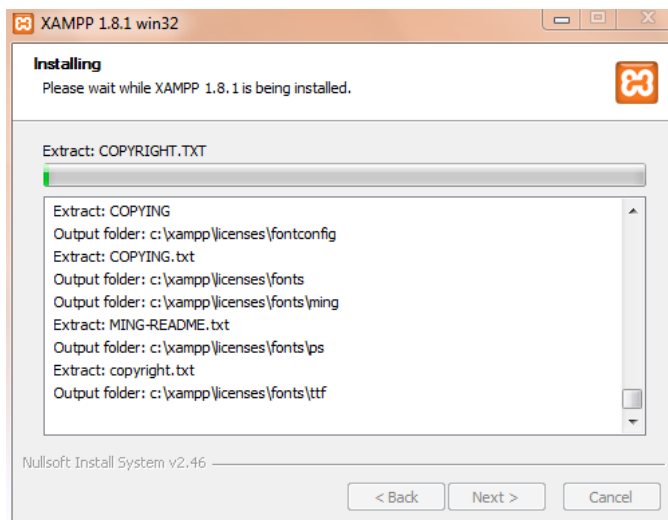
Για να την εγκατάσταση του Xampp εκτέλεσα τα παρακάτω βήματα:



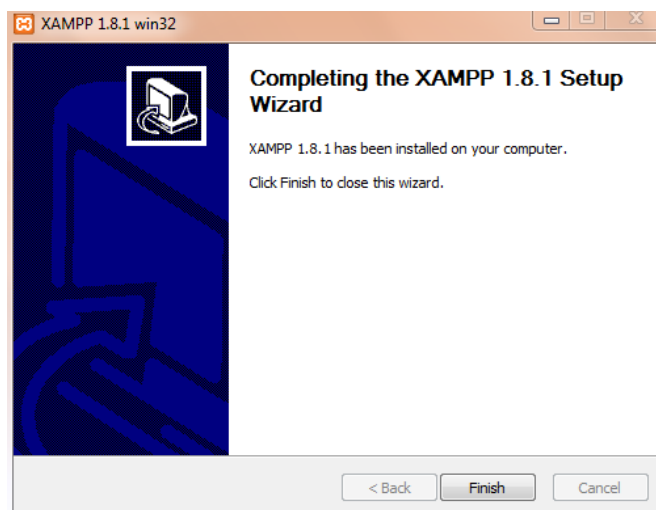
Εικόνα 49: Εγκατάσταση Xampp βημα1



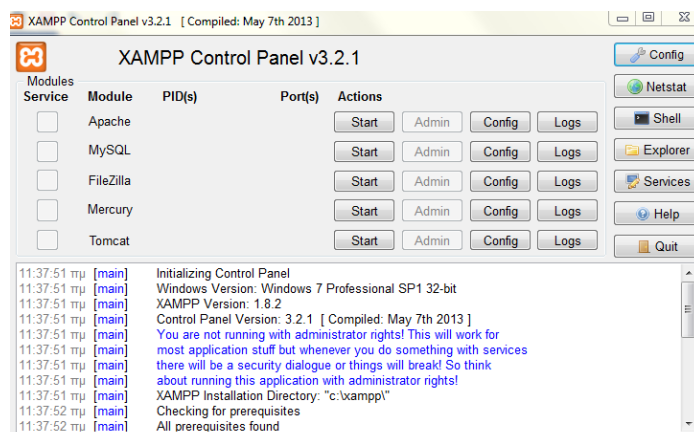
Εικόνα 50: Εγκατάσταση Xampp βημα2



Εικόνα 51: Εγκατάσταση Xampp βημα3

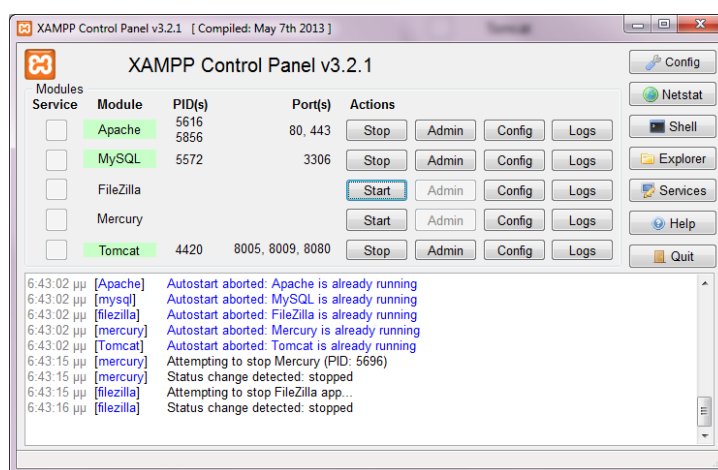


Εικόνα 52: Εγκατάσταση Xampp βημα4



Εικόνα 53: Εγκατάσταση Χαμpp βημας

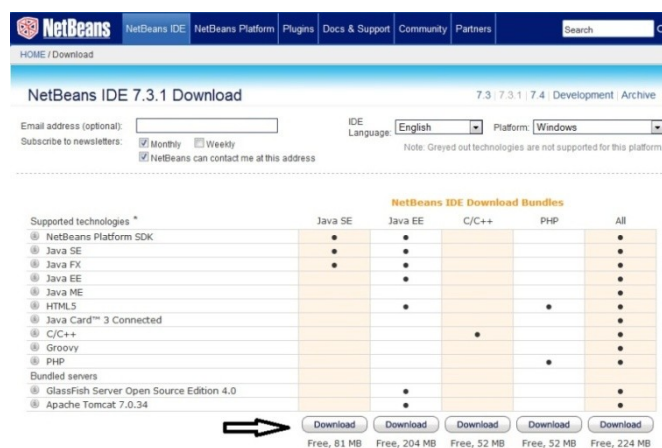
Αφού έχουμε εκτελέσει όλα τα βήματα εγκατάστασης θέτουμε τον Apache, MySql και τον Tomcat σε λειτουργία



Εικόνα 54: Ενεργοποίηση Χαμpp

Παράρτημα2: Εγκατάσταση Net beans

Για την δημιουργία και εκτέλεση της εφαρμογής έγινε εγκατάσταση της 7.3.1 έκδοσης του Net Beans



Εικόνα 55: Εγκατάσταση του NetBeans

Η εγκατάσταση ήταν εύκολη ακολουθώντας τα βήματα εγκατάστασης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1.Βιβλία

Darrel Ince, Κατανεμημένες εφαρμογές και ηλεκτρονικό εμπόριο,2007,σελ 204

Paul Deitel,Harvey Deitel, Java Προγραμματισμός, 2010, σελ8,9

2.Διευθύνσεις url

2.1 Wikipedia:2 <http://el.wikipedia.org>

2.2 Netbeans: <https://netbeans.org/>

2.3 Wikibooks: <http://el.wikibooks.org>

2.4 Ebooks: <http://www.ebooks4greeks.gr/>

2.5 Docs oracle: <http://docs.oracle.com>