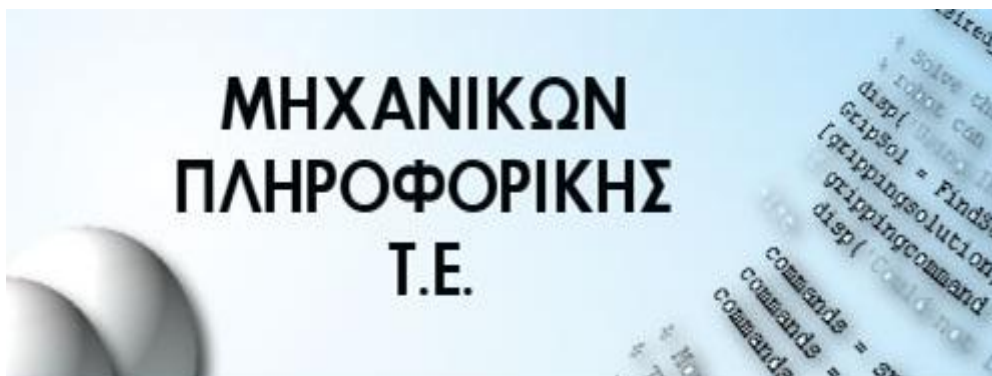


Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Ηπείρου



Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Μηχανικών Πληροφορικής Τ.Ε.

## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη βάσης δεδομένων σε cloud server με χρήση PHP

Σπουδάστρια

Κανέλα Σουργκούνη

Επιβλέπων Καθηγητής

Τάτσης Γεώργιος

Άρτα 2014

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

...../...../.....

Τόπος

Ημερομηνία

### **Επιτροπή Αξιολόγησης**

1. ....

Όνοματεπώνυμο Υπογραφή

2. ....

Όνοματεπώνυμο Υπογραφή

3. ....

Όνοματεπώνυμο Υπογραφή

**«Δηλώνω υπεύθυνα ότι το παρόν κείμενο αποτελεί προϊόν προσωπικής μελέτης και εργασίας και πως όλες οι πηγές που χρησιμοποιήθηκαν για τη συγγραφή της δηλώνονται σαφώς είτε στις παραπομπές είτε στη βιβλιογραφία. Γνωρίζω πως η λογοκλοπή αποτελεί σοβαρότατο παράπτωμα και είμαι ενήμερος/η για την επέλευση των νομίμων συνεπειών.»**

## ΠΕΡΙΛΗΨΗ

Αυτή η πτυχιακή υλοποιεί λειτουργίες μιας εφαρμογής cloud computing. Ο σκοπός της είναι να κάνει πιο γρήγορο και εύχρηστο τον διαμοιρασμό αρχείων ανάμεσα σε χρήστες που έχουν τη δυνατότητα να συνδεθούν στο διαδίκτυο. Συγκεκριμένα αυτή η πτυχιακή περιέχει :

- 1<sup>ο</sup> κεφάλαιο  
Επεξήγηση της έννοιας cloud computing και όλων των γλωσσών προγραμματισμού που χρησιμοποιήθηκαν για την υλοποίηση της ιστοσελίδας.
- 2<sup>ο</sup> κεφάλαιο  
Τα εργαλεία και ο σκοπός τους για την υλοποίηση της εφαρμογής.
- 3<sup>ο</sup> κεφάλαιο  
Ανάλυση της δομής εφαρμογής με βασικές λειτουργίες που χρησιμοποιήθηκαν για την κατασκευή της ιστοσελίδας.
- 4<sup>ο</sup> κεφάλαιο  
Λεπτομερής ανάλυση της λειτουργίας της εφαρμογής συμπεριλαμβανομένων του κώδικα και παραδείγματα φωτογραφιών από την εκτέλεση της εφαρμογής.

Οι λειτουργίες που υλοποιεί η εφαρμογή είναι :

- ανάλυση των υπηρεσιών του skybox
- Είσοδος στην εφαρμογή
- Δημιουργία λογαριασμού
- Upload και Download αρχείων
- Επεξεργασία αρχείων
- Δημιουργία-επεξεργασία φακέλου
- Διαμοιρασμός φακέλων μεταξύ χρηστών

## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΕΦΑΛΑΙΟ 1: ΔΙΑΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ.....	6
1.1 Τι είναι το skybox;.....	6
1.2 Γενικά για το cloud computing.....	7
1.3 Γλώσσα προγραμματισμού html.....	10
1.4 Γλώσσα προγραμματισμού php.....	13
ΚΕΦΑΛΑΙΟ 2: ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ.....	15
2.1 xampp.....	15
2.1.1 Πληροφορίες εγκατάστασης του XAMPP.....	17
2.2 Notepad ++.....	20
2.2.1 Εγκατάσταση του Notepad++.....	20
2.3 Mozilla.....	22
ΚΕΦΑΛΑΙΟ 3: ΔΟΜΗ ΕΦΑΡΜΟΓΗΣ.....	24
3.1 JavaScript και JQuery.....	24
3.2 Forms.....	25
3.2.1 Post – Get.....	26
3.2.2 Sessions.....	27
3.2.3 Αντικείμενα Φόρμας.....	27
3.3 Css.....	28
3.4 Εντολές MySQL.....	29
ΚΕΦΑΛΑΙΟ 4 : ΑΝΑΛΥΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ.....	31
4.1 ανάλυση των υπηρεσιών του skybox.....	31
4.2 Είσοδος στην εφαρμογή.....	32
4.3 Δημιουργία λογαριασμού.....	35
4.4 Upload και Download αρχείων.....	37
4.5 Επεξεργασία αρχείων.....	40
4.6 Δημιουργία-επεξεργασία φακέλου.....	43
4.7 Διαμοιρασμός φακέλων μεταξύ χρηστών.....	48
ΚΕΦΑΛΑΙΟ 5 : ΣΥΜΠΕΡΑΣΜΑΤΑ.....	51

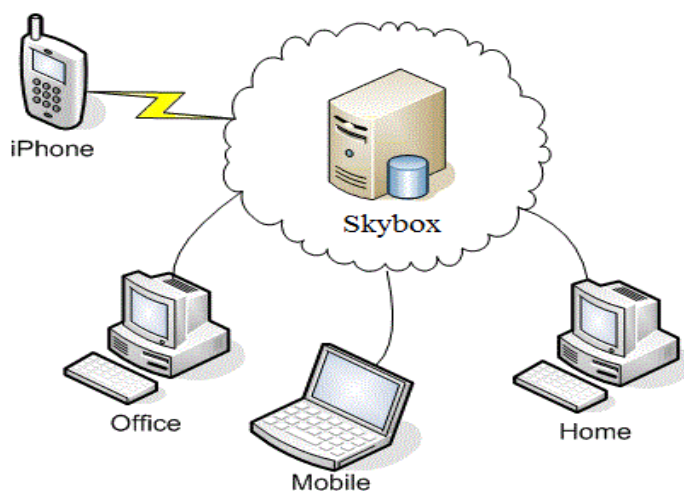
# ΚΕΦΑΛΑΙΟ 1<sup>ο</sup>

## Διαδικτυακές Εφαρμογές

### 1.1 Τι είναι το Skybox;

Η Skybox.gr είναι μια υπηρεσία όπου μπορείς να αποθηκεύσεις αρχεία και φακέλους σε servers στο internet. Δηλαδή μια υπηρεσία αποθήκευσης, απλή στην χρήση της και αν επιθυμείς μπορείς να ανακτήσεις τα δεδομένα σου οποιαδήποτε στιγμή από οποιοδήποτε τερματικό καθώς και να τα μοιραστείς με άλλους χρήστες.

Για να καταστεί αυτό δυνατό το μόνο που χρειάζεται ο χρήστης είναι να δημιουργήσει τον προσωπικό του λογαριασμό στη σελίδα Skybox.gr . Η δημιουργία του λογαριασμού είναι απλή και εύκολη, ο χρήστης δίνει το προσωπικό του email (που είναι μοναδικό) και έναν κωδικό ασφαλείας (password) και κατευθείαν εισέρχεται στην αρχική σελίδα. Έπειτα μπορεί να αρχίσει αμέσως την αποθήκευση των αρχείων του σ' αυτή. Έτσι έχουμε μια ιστοσελίδα με τα προσωπικά μας αρχεία όπου μπορούμε να τα ανακτήσουμε όποτε εμείς θελήσουμε αρκεί να έχουμε πρόσβαση στο διαδίκτυο.



Εικόνα 1.1 Skybox

Υπάρχουν πολλοί λόγοι για να χρησιμοποιήσει κανείς την υπηρεσία αυτή. Έστω ότι είσατε στο desktop pc της δουλειάς και θέλετε να συνεχίσετε την δουλειά στο laptop στο σπίτι σας. Αρχικά “ανεβάζετε” τα αρχεία που θέλετε στη σελίδα και έπειτα στο σπίτι μέσω του browser της επιλογής σας, πληκτρολογείτε την διεύθυνση της σελίδας Skybox.gr ,εισέρχετε στον προσωπικό σας λογαριασμό και κάνετε download τα αρχεία που επιθυμείτε να επεξεργαστείτε. Ακόμα μπορείτε να τη χρησιμοποιήσετε για online backup μερικών απαραίτητων αρχείων σας ή ακόμη και να τα μοιράσετε σε άλλους χρήστες αφού δημιουργήσετε μέσα στη σελίδα έναν κοινόχρηστο φάκελο.

Παραπάνω γίνεται αναφορά για κοινόχρηστους φακέλους, ο κάθε χρήστης μέλος της ιστοσελίδας μπορεί να επιλέξει αν επιθυμεί να μοιραστεί τα αρχεία του με άλλους χρήστες. Για να καταστεί αυτό δυνατό αρκεί να γνωρίζει το e-mail του άλλου χρήστη-μέλους της ιστοσελίδας και να του στείλει την αίτηση του για κοινή χρήση κάποιου φακέλου. Όταν γίνει αποδοχή της αίτησης από τον άλλον χρήστη και οι δύο αποκτούν τα ίδια δικαιώματα στον φάκελο. Αυτό σημαίνει πως όσοι μοιράζονται τον φάκελο μπορούν να δουν τα περιεχόμενά του, να τα κατεβάσουν-αποθηκεύσουν, να προσθέσουν άλλα καθώς και να τα διαγράψουν.

Τα υπέρ του είναι πως μπορείτε να διαχειρίζεστε τα αρχεία σας από οπουδήποτε και δεν χρειάζεται να εγκαταστήσετε κάποιο πρόγραμμα. Είναι γρήγορο, άμεσο, ασφαλές και αξιόπιστο.

Ένα μειονέκτημα είναι ότι χωρίς την πρόσβαση στο internet η χρήση είναι αδύνατη και πως οποιαδήποτε κακή σύνδεση θα καθυστερούσε την όποια εργασία του χρήστη.

## 1.2 Γενικά για το cloud computing;

*Τι είναι το cloud computing;*

Το **cloud computing** όπως το ορίζει το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας (**National Institute of Standards and Technology**) είναι ένα μοντέλο που επιτρέπει ευέλικτη, on-demand δικτυακή πρόσβαση σε ένα κοινόχρηστο σύνολο παραμετροποιήσιμων υπολογιστικών πόρων (όπως δίκτυα, servers, αποθηκευτικοί χώροι, εφαρμογές και υπηρεσίες), το οποίο μπορεί να τροφοδοτηθεί γρήγορα και να διατεθεί με ελάχιστη προσπάθεια διαχείρισης ή αλληλεπίδραση με τον πάροχο της υπηρεσίας. Αυτό το cloud μοντέλο προωθεί την διαθεσιμότητα και αποτελείται από πέντε βασικά χαρακτηριστικά, τρία μοντέλα παροχής υπηρεσιών, και τέσσερα μοντέλα ανάπτυξης.



Εικόνα 1.2 Cloud Computing

*Τα πέντε αυτά χαρακτηριστικά είναι :*

- **αυτοκαθορισμός των πόρων:** Οι χρήστες προβλέπουν μόνοι τους τις ανάγκες σε πόρους, όπως επιπλέον συστήματα (επεξεργαστική ικανότητα, λογισμικό, αποθήκευση) και πόρους δικτύου.
- **πληρωμή ανάλογα με την χρήση:** Οι χρήστες πληρώνουν τους πόρους που χρησιμοποιούν μόνο για όσο χρόνο τους χρειάζονται.
- **Πολυχρηστικότητα :** οι υπηρεσίες του cloud πρέπει να υποστηρίζουν διαφορετικές εφαρμογές. Οι εφαρμογές μπορούν να διαμοιράζονται πληροφορίες αλλά εκτελούνται απομονωμένα. Τέλος η ανάπτυξη μιας νέας εφαρμογής θα πρέπει να απαιτεί μικρή προσπάθεια.
- **ευρεία κλιμάκωση :** Το cloud παρέχει την δυνατότητα να ανέρθουν τα υπολογιστικά συστήματα σε δεκάδες χιλιάδες καθώς και την δυνατότητα να αυξηθεί μαζικά το εύρος ζώνης και ο αποθηκευτικός χώρος.
- **ελαστικότητα:** οι εφαρμογές πρέπει να είναι σε θέση να διαπραγματεύονται και να λαμβάνουν επιπλέον πόρους προκειμένου να καλύπτουν τις διαρκώς αυξανόμενες ανάγκες τους σε υπολογιστική ισχύ και αποθηκευτική δυνατότητα.

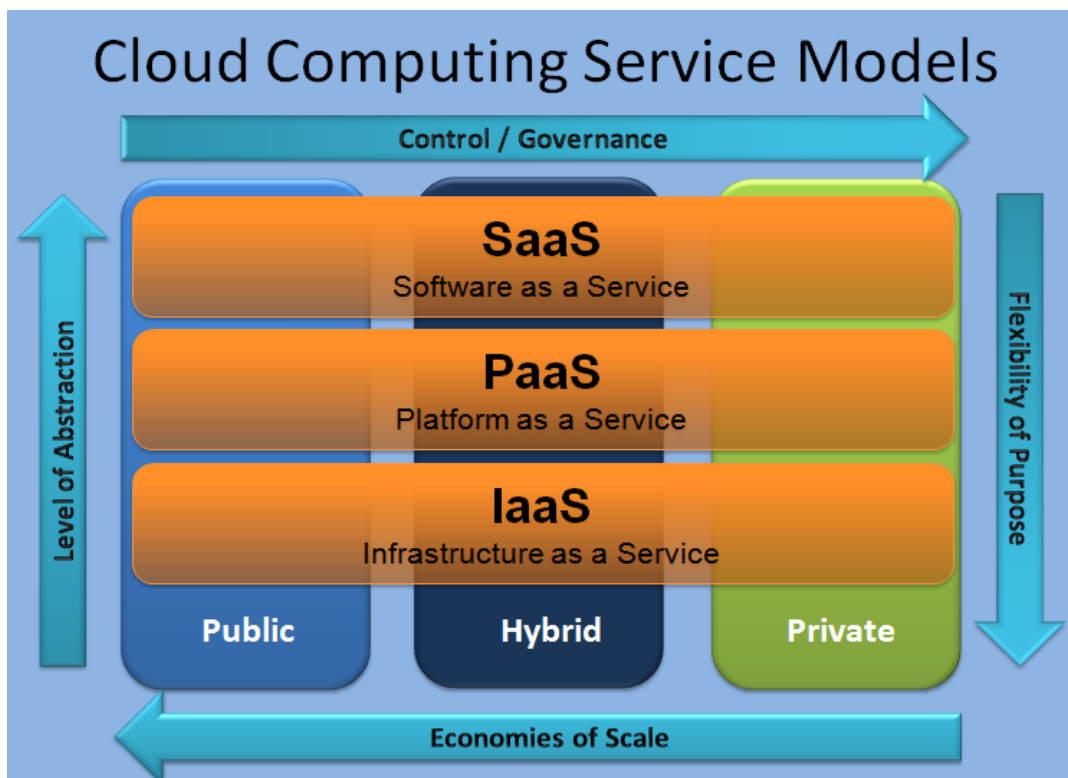
**Τα τρία μοντέλα παροχής υπηρεσιών :**

- **Cloud Software as a Service (SaaS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να χρησιμοποιεί τις εφαρμογές του παρόχου που τρέχουν σε μια cloud υποδομή. Οι εφαρμογές είναι προσβάσιμες από διάφορες client συσκευές μέσω ενός thin client interface, όπως ένα πρόγραμμα περιήγησης στο Web. Ο καταναλωτής δεν έχει τη διαχείριση ή τον έλεγχο της χρησιμοποιούμενης cloud υποδομής συμπεριλαμβανομένων των δικτύων, των servers, των λειτουργικών συστημάτων, των αποθηκευτικών μονάδων, ή ακόμα και μεμονωμένων δυνατοτήτων της εφαρμογής, με την πιθανή εξαίρεση κάποιων περιορισμένων user-specific ρυθμίσεων παραμετροποίησης των εφαρμογών.
- **Cloud Platform as a Service (PaaS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να αναπτύσσει πάνω στην cloud υποδομή εφαρμογές που έχει δημιουργήσει ή εφαρμογές που έχει αποκτήσει, οι οποίες έχουν δημιουργηθεί με χρήση γλωσσών προγραμματισμού και εργαλείων που υποστηρίζονται από τον πάροχο. Ο καταναλωτής δεν διαχειρίζεται ούτε ελέγχει τη σχετική cloud υποδομή που συμπεριλαμβάνει τα δίκτυα, τους servers, τα λειτουργικά συστήματα ή τα αποθηκευτικά μέσα, αλλά έχει τον έλεγχο των εφαρμογών που έχουν αναπτυχθεί, και ενδεχομένως, των παραμετροποιήσεων του περιβάλλοντος φιλοξενίας των εφαρμογών.
- **Cloud Infrastructure as a Service (IAAS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να μπορεί να δεσμεύσει προς χρήση επεξεργαστική ισχύ, αποθηκευτικά μέσα, δίκτυα, και άλλους θεμελιώδεις υπολογιστικούς πόρους, όπου ο καταναλωτής είναι σε θέση να αναπτύξει και να εκτελέσει αυθαίρετο λογισμικό, το οποίο μπορεί να περιλαμβάνει λειτουργικά συστήματα και εφαρμογές. Ο καταναλωτής δεν έχει τη διαχείριση ή τον έλεγχο της χρησιμοποιούμενης cloud υποδομής, αλλά έχει τον έλεγχο των λειτουργικών συστημάτων, των αποθηκευτικών μέσων, των εφαρμογών που έχουν αναπτυχθεί και πιθανόν κάποιον περιορισμένο έλεγχο επιλεγμένου εξοπλισμού δικτύωσης.



### Τέσσερα μοντέλα εφαρμογής cloud:

- **Community cloud:** Η cloud υποδομή μοιράζεται μεταξύ πολλών οργανισμών και υποστηρίζει μια συγκεκριμένη κοινότητα που έχει κοινές ανησυχίες (πχ. αποστολή, απαιτήσεις ασφαλείας, πολιτική και θέματα συμμόρφωσης). Η διαχείρισή της μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί να βρίσκεται εντός ή εκτός των εγκαταστάσεων του οργανισμού.
- **Private cloud:** Η cloud υποδομή λειτουργεί αποκλειστικά και μόνο για έναν. Η διαχείρισή της μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί να βρίσκεται εντός ή εκτός των εγκαταστάσεων του οργανισμού.
- **Hybrid cloud:** Ένα υβριδικό cloud αποτελείται από πολλαπλούς εσωτερικούς και /ή εξωτερικούς παρόχους clouds (private, community or public) τα οποία παραμένουν μοναδικές οντότητες, αλλά συνδέονται μεταξύ τους με τυποποιημένη ή αποκλειστική τεχνολογία που επιτρέπει τη φορητότητα δεδομένων και εφαρμογών (π.χ. εξισορρόπηση φόρτου εργασίας μεταξύ των clouds).
- **Public cloud:** Η cloud υποδομή διατίθεται στο ευρύ κοινό ή σε μια μεγάλη ομάδα εταιρειών και ανήκει σε έναν οργανισμό που πουλά υπηρεσίες cloud. Οι πόροι παρέχονται δυναμικά, σε μια αυτό-εξυπηρετούμενη βάση μέσω του internet, μέσω web εφαρμογών ή web υπηρεσιών, από έναν εξωτερικό τρίτο πάροχο που μοιράζει της πόρους και χρεώνει σύμφωνα με την χρήση.[1-3]



Εικόνα 1.3 Cloud Computing και μοντέλα παροχής υπηρεσιών.

### 1.3 Γλώσσα προγραμματισμού HTML.

Τι είναι η γλώσσα προγραμματισμού HTML;

Η **HTML** είναι το ακρωνύμιο των λέξεων **HyperText Markup Language** και είναι η βασική γλώσσα δόμησης σελίδων του Web. Επιτρέπει την ενσωμάτωση ήχου και εικόνων στις web σελίδες. Αρχικά είχε κατασκευασθεί με σκοπό μόνο την μορφοποίηση κειμένου, αλλά μεγάλωσε και ενσωμάτωσε σχεδιαστικές τεχνικές κ.α. Η γλώσσα χρησιμοποιεί ένα αριθμό από **tags** για την μορφοποίηση κειμένου οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από», μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (<head> και </head>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης. Ανάμεσα στις ετικέτες, μπορεί να τοποθετηθεί κείμενο, πίνακες, εικόνες κλπ. Η σήμανση της html αποτελείται εκτός από τα στοιχεία (tags), από τύπους δεδομένων, αναφορές χαρακτήρων και τις αναφορές οντοτήτων. Ωστόσο το συστατικό που ορίζει στον browser τον τρόπο εμφάνισης της σελίδας είναι η δήλωση τύπου εγγράφου. Όταν ένας Web Browser ανοίγει ένα αρχείο HTML τα στοιχεία μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας.[4]

Παρακάτω θα δούμε όλα τα tags :

- **<html>** δηλώνει την αρχή ενός αρχείου html και το tag **</html>** που δηλώνει το τέλος του αρχείου, ότι βρίσκεται ανάμεσά τους είναι τμήμα του εγγράφου html.
- **<head>** ,**</head>** δείχνει την αρχή και το τέλος της επικεφαλίδας ενός εγγράφου HTML document.
- **<title>** παρέχει στην σελίδα ένα τίτλο που θα εμφανίζεται στη γραμμή τίτλου του browser.
- Ανάμεσα στα **<body>** και **</body>** τοποθετούνται το κείμενο, οι εικόνες και ότι άλλο θέλουμε να περιλαμβάνει η σελίδα.
- **<body topmargin=25 leftmargin=30>**, ορίζονται τα περιθώρια της σελίδας για πάνω και αριστερά (αυτό σημαίνει ότι δεν εμφανίζονται κείμενο και εικόνες στο περιθώριο)
- **<br>**, δεν έχει tag τέλους. Είναι tag αλλαγής σειράς. Αν το θέσουμε στο μέσο κειμένου, το κείμενο αλλάζει σειρά.
- **<p>**, χρησιμοποιείται για την εισαγωγή παραγράφων. Μπορούμε να εισάγουμε ένα **<p>** πριν την αρχή της παραγράφου και ένα **</p>** μετά το τέλος. Μπορούμε να απλοποιήσουμε τη διαδικασία με το να εισάγουμε μόνο **<p>** ανάμεσα στις παραγράφους.
- **<pre>**, μπορούμε να θέσουμε το κείμενο ακριβώς στη θέση που θέλουμε. Είναι ο πιο εύκολος τρόπος αλλά το κείμενο εμφανίζεται σαν να έχει γραφεί με γραφομηχανή.
- **<h2>** Για την εισαγωγή μιας επικεφαλίδας πληκτρολογούμε **<h2>** μετά την επικεφαλίδα και τέλος το tag τέλους **</h2>**. Οι επικεφαλίδες μπορούν να στοιχηθούν αριστερά, στο κέντρο, ή δεξιά χρησιμοποιώντας την παράμετρο "align=" στο tag αρχής. Αν χρησιμοποιήσουμε πολλές διαφορετικές επικεφαλίδες το μέγεθος τους πρέπει να μειώνεται.

- **<blockquote> </blockquote>**, εισάγει αριστερό περιθώριο στο κείμενο που περιλαμβάνεται μεταξύ των tag αρχής και τέλους. Είναι πολύ αποτελεσματικό στο να καθιστά μία παράγραφο εμφανή.
- **<!--** , Σχόλιο (δεν εμφανίζονται στον Browser)
- **<b>** , Δημιουργία έντονων χαρακτήρων (**bold**)
- **<i>** , Δημιουργία πλάγιων χαρακτήρων (*Italic*)
- **<u>** , Υπογράμμιση χαρακτήρων(underline)
- **<tt>** , Χαρακτήρες γραφομηχανής (Typewriter)
- **<strike>** , Διακριτή διαγραφή (~~strike~~)
- **<sup>** , Εκθέτης
- **<sub>** , Δείκτης
- **<small>** , Μικρά γράμματα
- **<big>** , Μεγάλα γράμματα
- **<font>** , Μορφοποίηση γραμματοσειρά
- **<hr>** , Οριζόντια γραμμή
- **<a>** , Σύνδεσμος (link)
- **<img>** , Εισαγωγή εικόνας

## Δημιουργία αρχείων HTML

Η δημιουργία αρχείων HTML είναι πολύ απλή. Αρκεί να “τρέξουμε” έναν οποιοδήποτε διορθωτή *κειμένου text*, όπως το *Notepad* των Windows. Να γράψουμε τον κώδικα HTML που επιθυμούμε και να το αποθηκεύσουμε (σώσουμε) σε ένα αρχείο με κατάληξη **.htm** ή **.html**. Συνήθως αποθηκεύουμε με επέκταση **.html** όταν θα δημοσιεύσουμε τα αρχεία μας σε διακομιστή Unix. Αντίθετα ο εξυπηρετητής Windows NT καταλαβαίνει τα αρχεία και σαν **.htm** και σαν **.html**. [5]

Παράδειγμα γραφής της html :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello HTML</title>
  </head>
  <body>
    <p> Hello Greece </p>
  </body>
</html>
```

Το κείμενο ανάμεσα στο **<html>** και το **</html>** περιγράφει την ιστοσελίδα, και το κείμενο μεταξύ του **<body>** και του **</body>** είναι το ορατό μέρος της. Το σημασμένο κείμενο '**<title>Hello HTML</title>**' καθορίζει τον τίτλο που θα εμφανίζεται στην μπάρα τίτλου του browser.



Εικόνα 1.4 Επίσημη σήμανση της HTML

Τέλος, στον παρακάτω πίνακα παρουσιάζονται οι εκδόσεις HTML. [6]

<b>Έκδοση</b>	<b>Σχόλια</b>
HTML 2.0	Είχε στοιχεία που ήδη γνωρίζουμε, ωστόσο της έλειπαν μερικές από τις επεκτάσεις της Microsoft και της Netscape, ενώ δεν υποστήριζε πίνακες και ιδιότητες στοίχισης.
HTML 3	Έγινε φιλόδοξη προσπάθεια αναβάθμισης των χαρακτηριστικών και των βοηθημάτων της HTML από τον Dave Raggett. Η έκδοση 3.0 δεν ολοκληρώθηκε ποτέ.
HTML 3.2	Υποστήριζε πίνακες, εικόνες, επικεφαλίδες, ιδιότητες στοίχισης και κάποιες άλλες σχολαστικές λεπτομέρειες. Ωστόσο και από αυτή την έκδοση έλειπαν επεκτάσεις για τον Internet Explorer και τον Netscape όπως είναι για παράδειγμα τα Frames, Embed και τα Applets.
HTML 4.01	Τρέχον επίσημο πρότυπο. Υποστηρίζει τις περισσότερες απαραίτητες επεκτάσεις, και επιπλέον χαρακτηριστικά (έγγραφα σε πολλές γλώσσες, CSS, επιπλέον πίνακες, φόρμες και συμβατότητα με τη JavaScript).
HTML 5	Είναι αυτό που θα χρησιμοποιηθεί τα επόμενα χρόνια. Έχει αρκετά εντυπωσιακά νέα χαρακτηριστικά και είναι συμβατή με προηγούμενες εκδόσεις. Παρόλα αυτά, επειδή είναι ακόμα υπό ανάπτυξη ελάχιστοι browsers την υποστηρίζουν.

Πίνακας 1.1

## 1.4 Γλώσσα προγραμματισμού PHP.

Τι είναι η γλώσσα προγραμματισμού PHP ;

**Hypertext PreProcessor**, είναι μια γλώσσα συγγραφής σεναρίων που ενσωματώνεται μέσα στον κώδικα της HTML και εκτελείται στην πλευρά του server. Το μεγαλύτερο μέρος της PHP είναι εμπνευσμένο από τη C, PERL και JAVA. Ο σκοπός της είναι να δώσει τη δυνατότητα στους Web developers να δημιουργούν δυναμικά παραγόμενες ιστοσελίδες.

Διαχειρίζεται όλα τα αρχεία με επέκταση .php . Έχουμε κανονικά αρχεία HTML με ενσωματωμένα κομμάτια κώδικα PHP σε συγκεκριμένα σημεία. Το δυνατότερο κομμάτι και το πιο βασικό της PHP είναι η υποστήριξη που παρέχει σε μια ευρεία γκάμα από βάσεις δεδομένων. Αλλά και στο πιο βασικό της επίπεδο μπορεί να κάνει επεξεργασία των δεδομένων μιας φόρμας, δημιουργία δυναμικού περιεχομένου ιστοσελίδων ή αποστολή και λήψη cookies. [7]

**Βάσεις που υποστηρίζει :**

Adabas d	dBase	Empress	FilePro	Informix	InterBase	mSQL
mysql	Oracle	PostgreSQL	Solid	Sybase	Velocis	Unix dbm

Παράδειγμα γραφής της PHP :

```
<!DOCTYPE html>
<html>
  <head>
    <title>PHP Test</title>
  </head>
  <body>
    <?php echo '<p>Hello World</p>'; ?>
  </body>
</html>
```



Εικόνα 1.5 Επίσημη σήμανση PHP

Παρακάτω βλέπουμε τις εκδόσεις της PHP :

ΈΚΔΟΣΗ	Ημερομηνία κυκλοφορίας	Υποστήριξη μέχρι	Σχόλια
1.0	8 Ιουνίου 1995		Ονομάστηκε επίσημα "Personal Home Page Tools (PHP Tools)".
2.0	1 Νοεμβρίου 1997		Ονομάστηκε επίσημα "PHP/FI 2.0". Αυτή είναι η πρώτη έκδοση που θα μπορούσε να χαρακτηριστεί ως PHP, είναι μια αυτόνομη γλώσσα με πολλές δυνατότητες οι οποίες έχουν παραμείνει μέχρι σήμερα..
3.0	6 Ιουνίου 1998	20 Οκτωβρίου 2000	Την χρησιμοποιούν όλο και περισσότεροι προγραμματιστές. Οι Zeev Suraski και Andi Gutmans ξανάγραψαν τη βάση γι'αυτήν την έκδοση.
4.0	22 Μαΐου 2000	23 Ιανουαρίου 2001	Προστέθηκε πιο προηγμένο σύστημα δυο σταδίων ανάλυση/εκτέλεση (Zend)
4.1	10 Δεκεμβρίου 2001	12 Μαρτίου 2002	Εισήγαγε 'superglobals' (\$_GET, \$_POST, \$_SESSION, etc.)
4.2	22 Απριλίου 2002	6 Σεπτεμβρίου 2002	Από προεπιλογή είναι απενεργοποιημένα τα register_globals . Πλέον τα δεδομένα που λαμβάνονται από το δίκτυο δεν εισάγονται κατευθείαν στο global namespace, κλείνοντας έτσι πιθανές τρύπες ασφαλείας στις εφαρμογές.
4.3	27 Δεκεμβρίου 2002	31 Μαρτίου 2005	Εισήγαγε το περιβάλλον γραμμής εντολών (CLI), για να συμπληρώσει το CGI.
4.4	11 Ιουλίου 2005	7 Αυγούστου 2008	Διορθώθηκε ένα bug διαφθοράς της μνήμης, το οποίο απαιτούσε το σπάσιμο συμβατότητας με το δυαδικό σύστημα μαζί με επεκτάσεις που συντάχθηκαν κατά της έκδοσης PHP 4.3.x.
5.0	13 Ιουλίου 2004	5 Σεπτεμβρίου 2005	Zend μηχανισμός II με νέο μοντέλο εργασίας.
5.1	24 Νοεμβρίου 2005	24 Αυγούστου 2006	Βελτιώθηκε η απόδοση με την εισαγωγή μεταβλητών compiler στον ανασχεδιασμό της μηχανής του PHP. Προστέθηκε PHP Data Objects (PDO) ως ένα ενιαίο περιβάλλον για πρόσβαση σε βάσεις δεδομένων.
5.2	2 Νοεμβρίου 2006	6 Ιανουαρίου 2011	Ενεργοποίησε την επέκταση του φίλτρου από προεπιλογή. Μητρική JSON υποστήριξη.
5.3	30 Ιουνίου 2009	14 Αυγούστου 2014	Υποστήριξη Namespace, late static bindings, Jump label (limited goto), Native closures, Native PHP archives (phar), garbage collection για κυκλικές αναφορές, βελτιωμένη υποστήριξη των Windows, sqlite3, mysqlnd ως αντικατάσταση για το libmysql και ως βασική βιβλιοθήκη των επεκτάσεων που λειτουργούν με την MySQL, fileinfo ως αντικατάσταση για το mime_magic για την καλύτερη υποστήριξη MIME , επέκταση του Internationalization, και η απόσβεση της επέκτασης ereg.
5.4	1 March 2012	1 March 2015	Υποστήριξη Trait, Υποστήριξη σύνταξης μικρού πίνακα. Καταργήθηκαν τα στοιχεία: register_globals, safe_mode, allow_call_time_pass_reference, session_register(), session_unregister() and session_is_registered(). Ενσωματωμένος web server. Πολλές βελτιώσεις στις υπάρχουσες λειτουργίες, επιδόσεις και μειωμένες απαιτήσεις μνήμης.
5.5	20 June 2013	20 June 2016	Υποστήριξη των generators, στοιχεία για εξαιρέσεις χειρισμού, το OpCache είναι συνοδευόμενο στην επίσημη διανομή.
5.6	No date set	3 years after release	Συνεχής εκφράσεις scalar, συναρτήσεις variadic, argument unpacking, νέος εκθετικός τελεστής, επεκτάσεις use statement για συναρτήσεις και σταθερές, νέος phrdbg αποσφαλματωτής ως ενότητα SAPI, και άλλες μικρές επεμβάσεις.

## ΚΕΦΑΛΑΙΟ 2<sup>ο</sup>

### Προγραμματιστικά εργαλεία

Για την υλοποίηση του πρακτικού μέρους της εργασίας αυτής χρειάστηκε να εγκαταστήσω και να χρησιμοποιήσω κάποια προγράμματα τα οποία θα τα παρουσιάσω παρακάτω αναλυτικά. Ονομαστικά είναι τα εξής : XAMPP, NOTEPAD ++ 2.3, MySQL, Mozilla.

#### **2.1 XAMPP**

Το XAMPP είναι ένα πλήρες πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα, και ανεξαρτήτου πλατφόρμας.

Διατίθεται σε δύο εκδόσεις. Η μία είναι πιο ελαφριά και προορίζεται για ανάπτυξη εφαρμογών με την γλώσσα php. Τα εργαλεία που περιλαμβάνει είναι ο Apache Web Server, η γλώσσα PHP, η βάση δεδομένων MySQL με το περιβάλλον διαχείρισης phpMyAdmin και το πρωτόκολλο ασφαλείας OpenSSL. Η πλήρης έκδοση περιλαμβάνει επιπλέον την γλώσσα Perl, την εφαρμογή στατιστικών Webalizer, καθώς και τους servers Filezilla FTP, Tomcat και Mercury Mail Transport System. Τρέχει σε Microsoft Windows, Linux, Solaris, Mac OS X και χρησιμοποιείται ως πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων. Περιλαμβάνει δηλαδή όλα τα απαραίτητα εργαλεία για την φιλοξενία ιστοσελίδων και την ανάπτυξη διαδικτυακών εφαρμογών.

Με την βοήθειά του, η εγκατάσταση όλων αυτών των εργαλείων γίνεται εύκολη. Μέσα σε ελάχιστο χρόνο, έχουμε ένα πλήρες περιβάλλον για να δημιουργήσουμε τις εφαρμογές μας.[8]

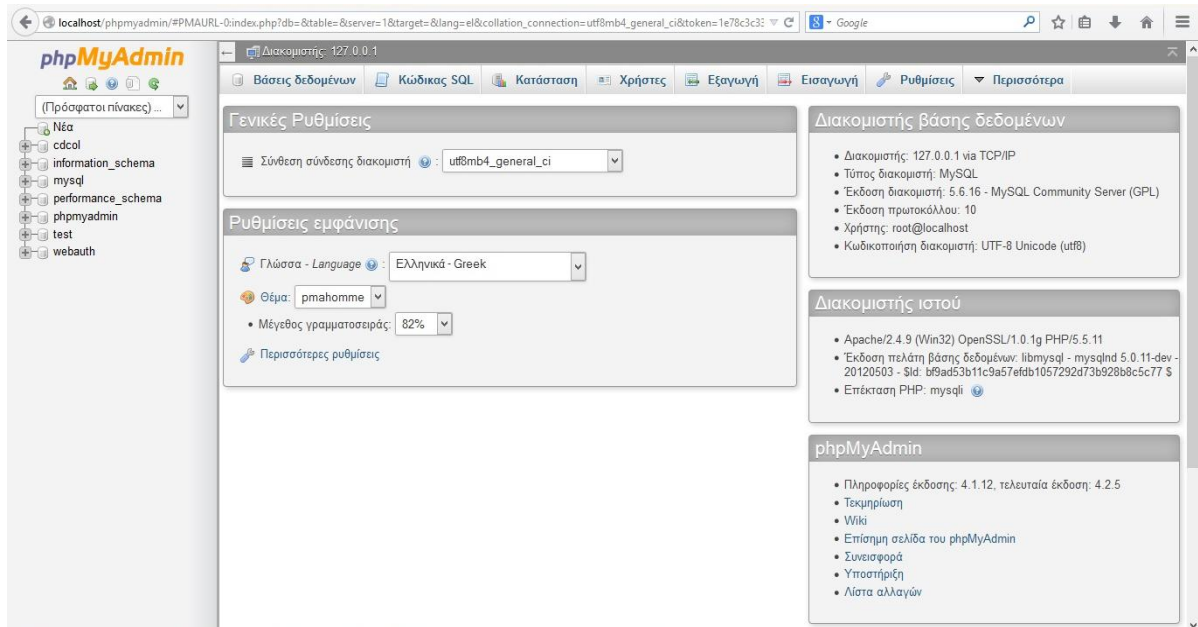
Από τα εργαλεία που μας παρέχει χρησιμοποιήσα τον **Apache Web Server** και τη βάση δεδομένων **MySQL**.

Ο **Apache** αρχικά σχεδιάστηκε για τους κεντρικούς υπολογιστές Unix. Αργότερα χρησιμοποιήθηκε για τα Windows και άλλα λειτουργικά συστήματα δικτύων (NOS =network operating systems). Είναι ελεύθερο λογισμικό όπου διανέμεται από το Apache Software Foundation που προωθεί τις διάφορες ελεύθερες και προηγμένες ανοικτές πηγές τεχνολογίας Ιστού (Web technologies). Ο ρόλος του είναι να αναμένει αιτήσεις από διάφορα προγράμματα – χρήστες (clients) όπως είναι ένας browser ενός χρήστη και στη συνέχεια να εξυπηρετεί αυτές τις αιτήσεις «σερβίροντας» τις σελίδες που ζητούν είτε απευθείας μέσω μιας ηλεκτρονικής διεύθυνσης (URL), είτε μέσω ενός συνδέσμου (link). Ο τρόπος με τον οποίο ο Apache εξυπηρετεί αυτές τις αιτήσεις, είναι σύμφωνος με τα πρότυπα που ορίζει το πρωτόκολλο HTTP (Hypertext Transfer Protocol).[9-11]

Η βάση δεδομένων **MySQL** είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων. Μας επιτρέπει να αποθηκεύουμε, να αναζητάμε, να ταξινομούμε και να ανακαλούμε τα δεδομένα αποτελεσματικά. Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα, για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL (

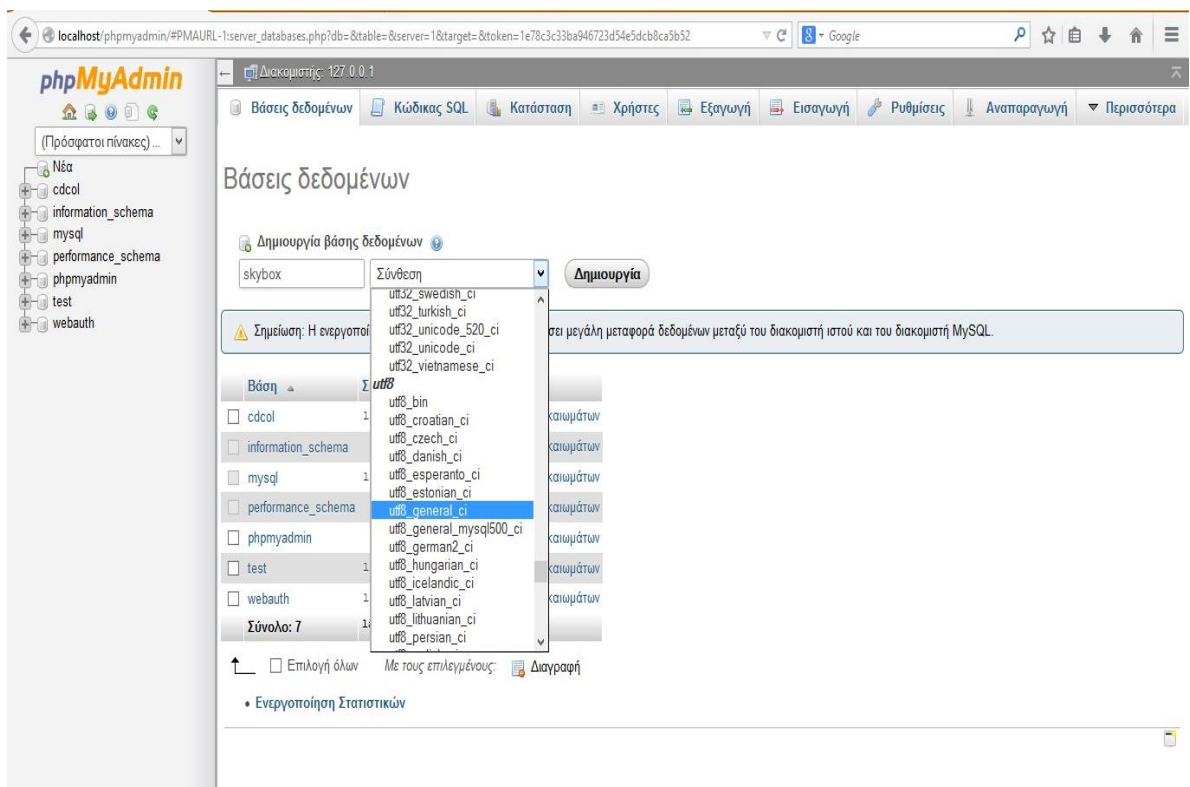
Structured Query Language ) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια.[12]

Συγκεκριμένα στο πρόγραμμα θα χρειαστεί να φτιάξω μια βάση οπότε μέσω του local host που το περιέχει τοxampp, δημιουργήσα την βάση μου.



Εικόνα 2.1 Δημιουργία της βάσης δεδομένων για το Skybox.

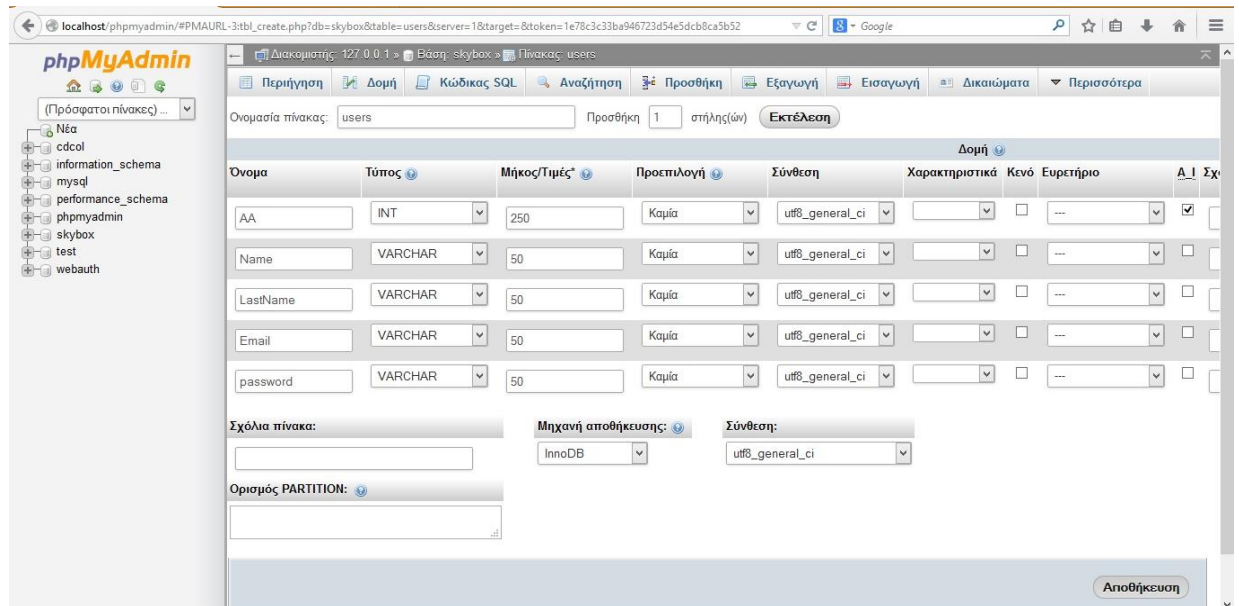
Έπειτα ρύθμισα τις παραμέτρους για τη σελίδα “Skybox”



Εικόνα 2.2 Ρύθμιση παραμέτρων για το Skybox.



Παρακάτω βλέπουμε τη δημιουργία ενός πίνακα με όνομα “users” και των κελιών του, όπου ρυθμίζουμε τον τύπο και το μήκος της μεταβλητής που θα καταχωρηθεί.

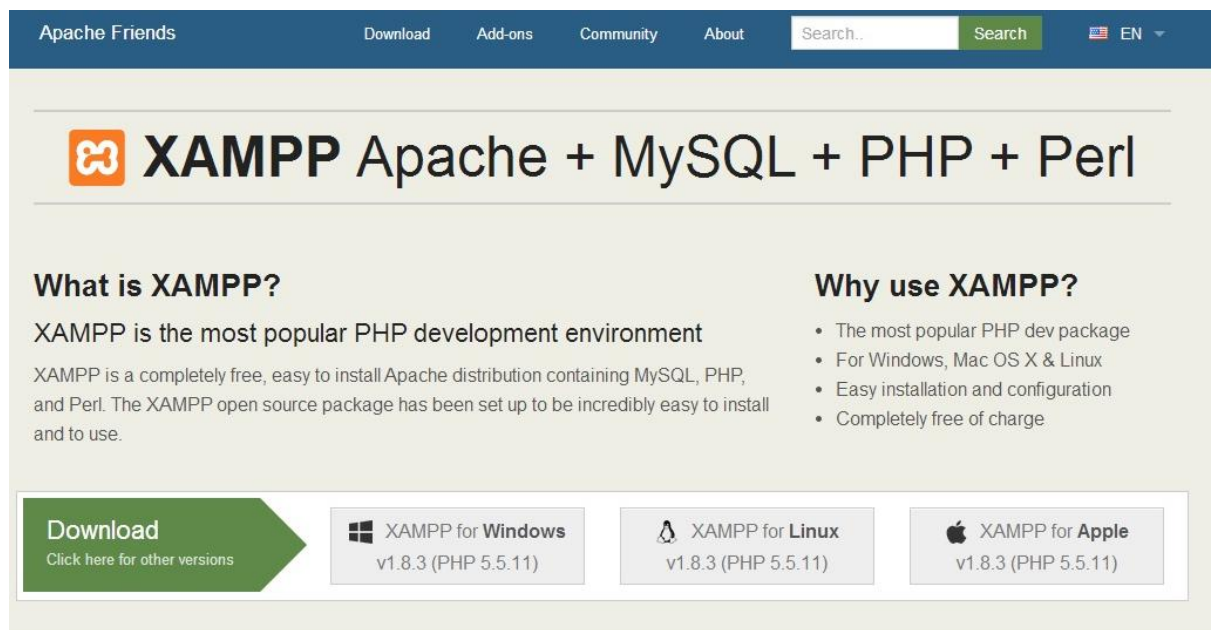


Εικόνα 2.3 Δημιουργία πίνακα στη βάση.

## 2.1.1 Πληροφορίες εγκατάστασης του προγράμματος XAMPP

### Βήμα 1<sup>ο</sup>

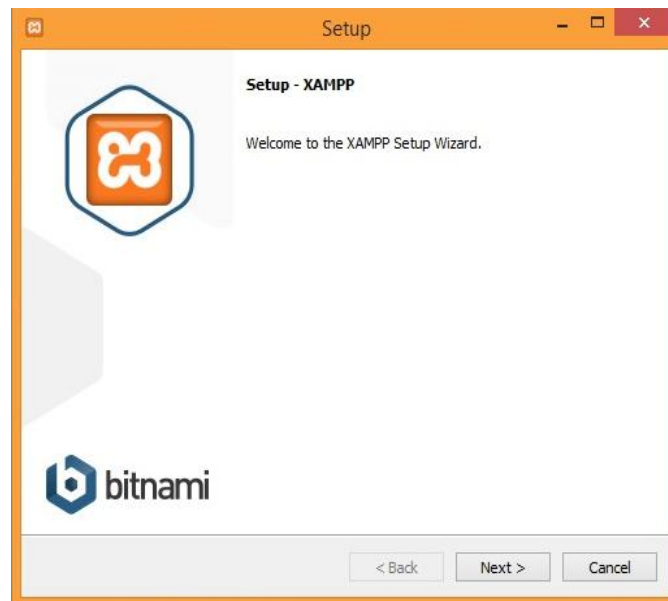
Για να κατεβάσουμε το πρόγραμμα μπαίνουμε στην επίσημη ιστοσελίδα του XAMPP (<https://www.apachefriends.org/index.html>). Στο παρακάτω παράθυρο επιλέγουμε για ποιά έκδοση το θέλουμε ( π.χ Windows ).



Εικόνα 2.4 Εγκατάσταση XAMPP

Βήμα 2<sup>ο</sup>

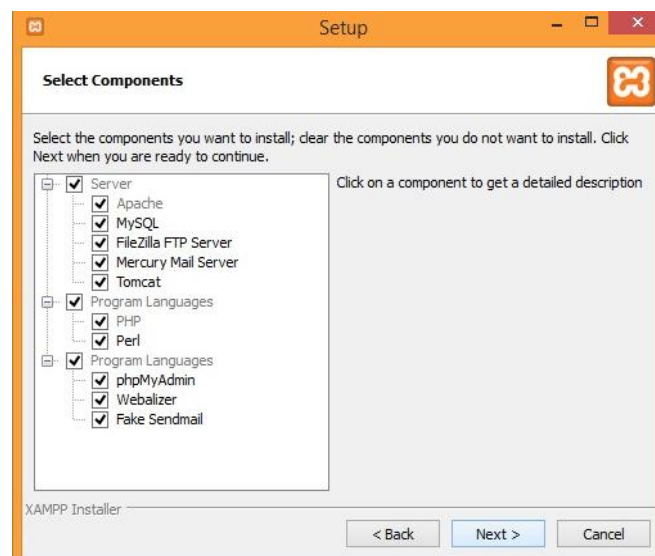
Εκτελούμε το set up επιλέγοντας Next>



Εικόνα 2.5 Εκτέλεση set up.

Βήμα 3<sup>ο</sup>

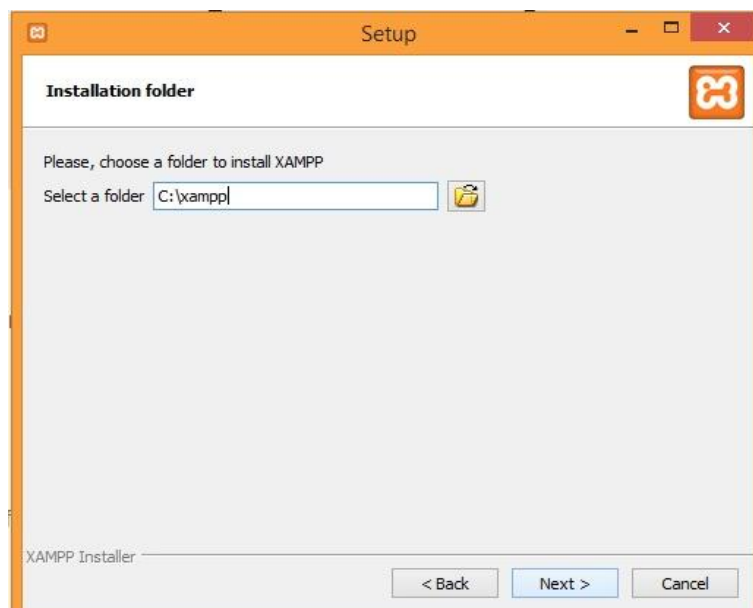
Επιλογή των στοιχείων προς εγκατάσταση (είναι ήδη επιλεγμένα) και πατάμε Next >



Εικόνα 2.6 Εκτέλεση set up

## Βήμα 4<sup>ο</sup>

Επιλέγουμε τη θέση που θέλουμε να αποθηκευτεί το πρόγραμμα.



Εικόνα 2.7 Εκτέλεση set up

## Βήμα 5<sup>ο</sup>

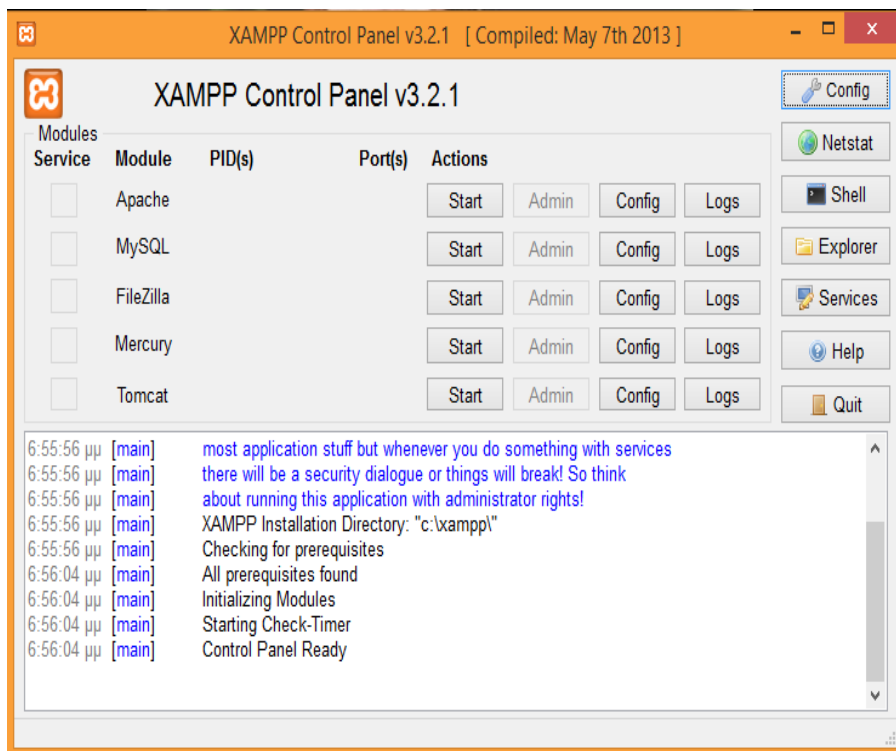
Επιτυχία εγκατάστασης και πατάμε Finish έχοντας επιλέξει το κουτάκι που μας προτείνει να ξεκινήσει το πρόγραμμα μετά το τέλος της εγκατάστασης.



Εικόνα 2.8 Εγκατάσταση set up

## Βήμα 6°

Εμφάνιση του **Control Panel**, όπου γίνεται ο έλεγχος και η διαχείριση όλων των εργαλείων. Από το παράθυρο αυτού του προγράμματος, βλέπουμε την κατάσταση όλων των εργαλείων. Επιπλέον, μπορούμε να εκκινήσουμε και να τερματίσουμε το κάθε ένα από αυτά ή να τροποποιήσουμε τις ρυθμίσεις του.



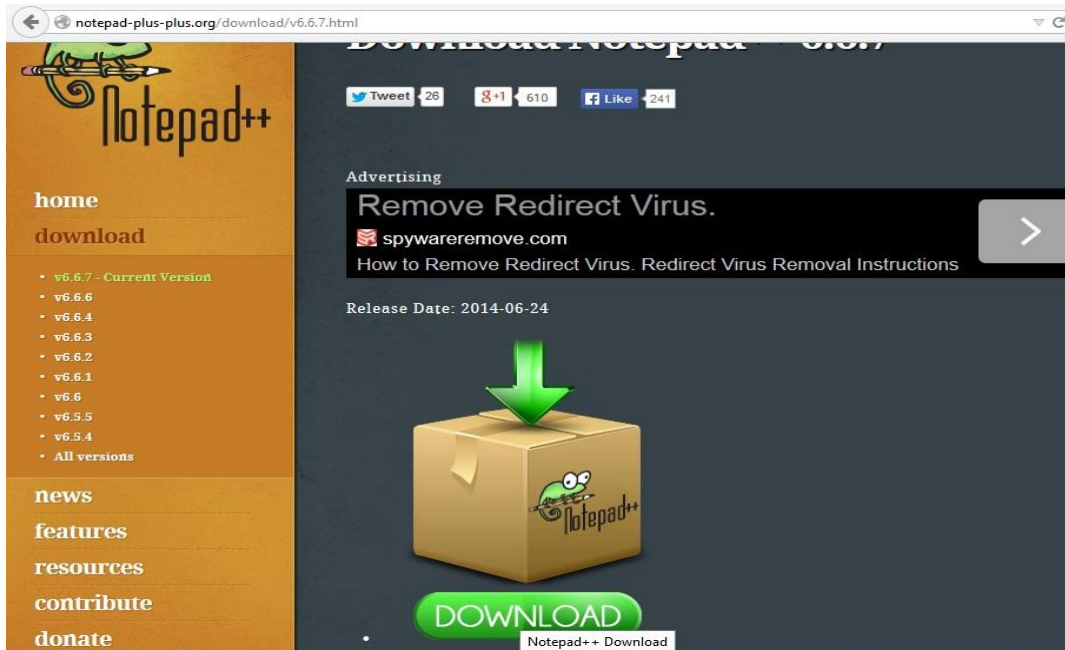
Εικόνα 2.9 Επιλογή προγράμματος εκκίνησης

## 2.2 Notepad ++ 2.3

Το Notepad++ διανέμεται ως ελεύθερο λογισμικό και είναι ένα εύχρηστο πρόγραμμα επεξεργασίας κειμένου και κώδικα για τα Windows. Διαθέτει καρτέλες επεξεργασίας κειμένου και επιτρέπει την εργασία σε πολλά ανοιχτά αρχεία στο ίδιο παράθυρο. Μπορείτε να το χρησιμοποιήσετε ακόμα και ως **source code editor** μιας και θα σας βοηθήσει να γράψετε ή να επεξεργαστείτε κώδικα σε **PHP,HTML** και δεκάδες ακόμα γλώσσες.[14]

### 2.2.1 Εγκατάσταση του Notepad++

Για να κατεβάσουμε το πρόγραμμα μπαίνουμε στην επίσημη ιστοσελίδα του Notepad++ (<http://notepad-plus-plus.org/download/v6.6.8.html>) και πατάμε το πράσινο “DOWNLOAD”.



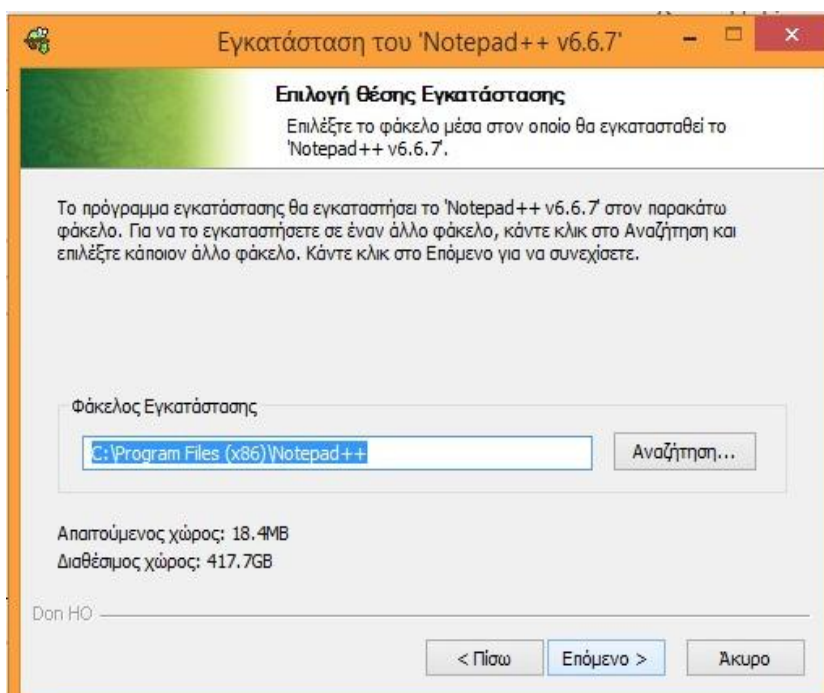
Εικόνα 2.10 Εγκατάσταση Notepad++

Έπειτα αφού κατεβάσει το πακέτο, πηγαίνω εκεί όπου αποθηκεύτηκε το αρχείο και το τρέχω. Αμέσως ξεκινά η εγκατάσταση. Πατάμε “Επόμενο”.



Εικόνα 2.11 Εγκατάσταση Notepad++

Έπειτα επιλέγουμε που θέλουμε να αποθηκευτεί το πρόγραμμα. Και πατάμε “επόμενο”.



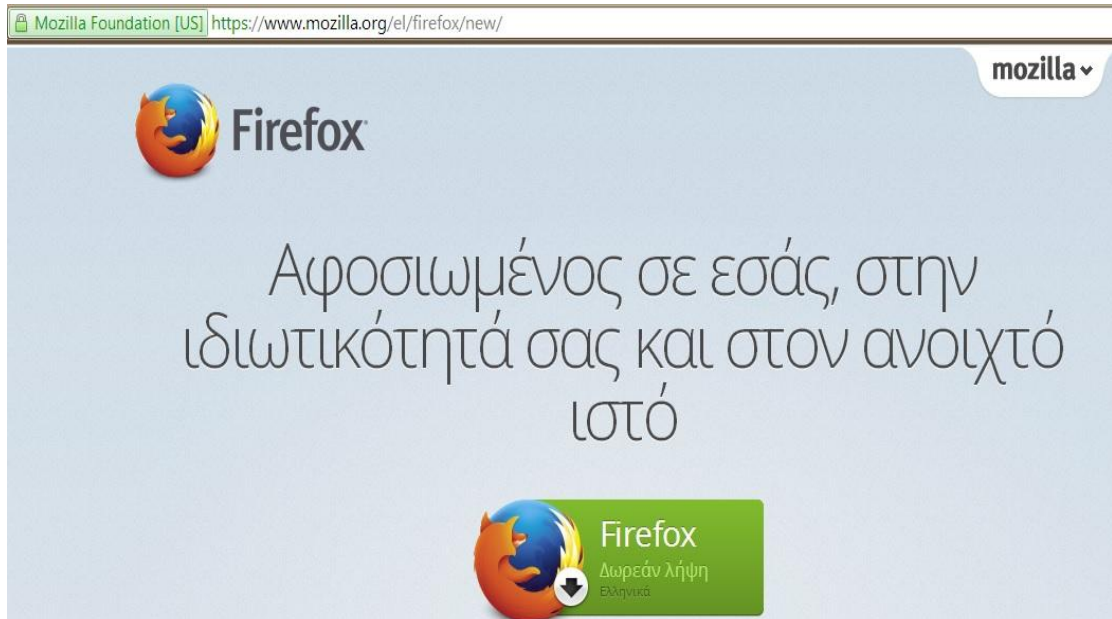
Εικόνα 2.12 Εγκατάσταση Notepad++

Τέλος, αφού το πρόγραμμα αποθηκεύτηκε με επιτυχία μπορούμε να ξεκινήσουμε να γράφουμε τον κώδικά μας απλώς επιλέγοντας αρχείο > νέο όπου μας ανοίγει μια νέα σελίδα προς επεξεργασία.

## 2.3 Mozilla

Ο Mozilla Firefox είναι ένας φυλλομετρητής του παγκόσμιου ιστού. Η εγκατάσταση είναι δωρεάν και κατατάσσεται στους πιο δημοφιλείς φυλλομετρητές (2<sup>ος</sup> σε θέση παγκοσμίως). Κέρδισε πολλούς χρήστες λόγω της πολύ καλής κατασκευής του (γρήγορος, φιλικός και αρκετά εξυπηρετικός) και βρίσκεται ένα βήμα μπροστά από τους υπόλοιπους browsers.

Στις λειτουργίες του Mozilla Firefox περιλαμβάνονται η φραγή των ανεπιθύμητων αναδυόμενων παραθύρων, η περιήγηση με τις καρτέλες, ο ορθογραφικός έλεγχος, οι ενεργοί σελιδοδείκτες, η ιδιωτική περιήγηση και ένα ενσωματωμένο πεδίο αναζήτησης με δυνατότητα επιλογής της μηχανής αναζήτησης που επιθυμούμε. Για περαιτέρω λειτουργίες υπάρχουν τα πρόσθετα (πχ. Adblock που κάνει φραγή διαφημίσεων). Η εγκατάσταση έγινε από την επίσημη σελίδα : <https://www.mozilla.org/el/firefox/new/> [15]



Εικόνα 2.13 Εγκατάσταση Mozilla Firefox.

Ωστόσο ο Mozilla χρειάστηκε για να ανοίγει τις σελίδες του προγράμματός μου. Αφού ανοίξω το control panel του xampp και επιλέξω start στον Apache και στο MySQL, ανοίγω τον browser και πληκτρολογώντας τη διεύθυνση localhost/xampp/ μου εμφανίζεται η αρχική σελίδα του xampp απ' όπου μπαίνω στην ιστοσελίδα που έφτιαξα.



Εικόνα 2.14 Αρχική σελίδα του xampp.

## ΚΕΦΑΛΑΙΟ 3ο

### Δομή εφαρμογής

#### 3.1 JavaScript και JQuery

Η γλώσσα προγραμματισμού **JavaScript** εμφανίστηκε το 1995 από τη Netscape με σκοπό να ζωντανέψει το τότε στατικό Web. Το αρχικό της όνομα ήταν Mocha έπειτα έγινε LiveScript και αργότερα για λόγους marketing μετονομάστηκε σε JavaScript (κυρίως γιατί επηρεάστηκε από τη γλώσσα Java). Είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική με ασθενείς τύπους και έχει αντικείμενα ως συναρτήσεις πρώτης τάξης.

Μερικά από τα πλεονεκτήματά της είναι:

1. Εύκολη στην εκμάθηση.
2. Δεν απαιτεί αγορά ειδικού λογισμικού (αρκεί το Notepad για την συγγραφή του κώδικα).
3. Άμεσα προγράμματα-αποτελέσματα χωρίς κόπο .Η συγγραφή του κώδικα είναι απλή υπόθεση. Οι εντολές εκτελούνται με τη σειρά που γράφονται.
4. Δεν απαιτεί μεταγλώττιση και μεταγλωττιστές.
5. Τρέχει σε όλα τα λειτουργικά συστήματα.

Μειονεκτήματα :

1. Μπορεί εύκολα να παρακαμφθεί, παρουσιάζοντας ζητήματα
2. Λειτουργικότητας σελίδας
3. Ασφάλειας
4. Αν δε σχεδιαστεί σωστά μπορεί να δυσκολέψει την πλοήγηση
5. Μπορεί να μεταφραστεί διαφορετικά από ποικίλους browsers
6. Μπορεί να αποτελέσει κενά ασφαλείας

Παράδειγμα χρήσης JavaScript στο πρόγραμμα :

```
<script>
var preEl ;
var orgBColor;
var orgTColor;
function HighLightTR(el,backColor,textColor){
    if(typeof(preEl)!="undefined"){
        preEl.bgColor=orgBColor;
        try{Change TextColor(preEl,orgTColor);}catch(e){;}
    }
    orgBColor = el.bgColor;
    orgTColor = el.style.color;
    el.bgColor=backColor;
}
</script>
```

Εξήγηση της παραπάνω γραμμής του κώδικα:

Κάθε φορά που πατάμε πάνω σε μια γραμμή του πίνακα αυτή αλλάζει χρώμα και μας δείχνει ποιά γραμμή επιλέξαμε.



Το **Jquery** είναι μια ελαφριά βιβλιοθήκη της JavaScript, συμβατή με όλους τους browsers, η οποία απλοποιεί την εκμάθηση και την χρήση της γλώσσας JavaScript (που χρησιμοποιείται για την δημιουργία ιστοσελίδων και web εφαρμογών. Με το JQuery μπορούμε να προσθέσουμε κίνηση-animation, να αυξήσουμε την διαδραστικότητα του χρήστη, να αλλάξουμε το περιεχόμενο της σελίδας χωρίς ο χρήστης να πρέπει να μεταφερθεί σε νέα σελίδα, να δημιουργήσουμε διάφορα εφέ και άλλα πολλά. Επίσης εξαλείφει το πρόβλημα συμβατότητας της JavaScript με τους διάφορους browsers της αγοράς. Κυκλοφόρησε το 2006 από τον John Resig και είναι ένα ελεύθερο λογισμικό-ανοιχτού κώδικα, αυτό σημαίνει ότι ο καθένας μπορεί να το χρησιμοποιήσει κατεβάζοντας μία από τις εκδόσεις από την επίσημη ιστοσελίδα του jQuery στην μορφή .js αρχείων. [16-19]

### 3.2 forms

Οι φόρμες μας δίνουν την δυνατότητα να μετασχηματίσουμε την ιστοσελίδα μας από μια μάζα πληροφοριών που ο επισκέπτης διαβάζει παθητικά σε μια αμφίδρομη επικοινωνία όπου έχει έναν πιο ενεργητικό ρόλο πληκτρολογώντας στοιχεία ή κάνοντας “κλικ” με το ποντίκι του σε διάφορες επιλογές.

*Η ετικέτα <form>*

Για να δημιουργήσουμε μια φόρμα χρησιμοποιούμε τις ετικέτες <form> και </form>. Οι κυριότερες ιδιότητες της ετικέτας είναι οι name, method και action. Με την ιδιότητα name δίνουμε ένα όνομα στις φόρμες που κατασκευάζουμε, ιδιαίτερα στις περιπτώσεις που έχουμε περισσότερες από μια φόρμες στην ίδια σελίδα. Η ιδιότητα action περιέχει το URL του αρχείου script στον Server το οποίο θα επεξεργαστεί τα στοιχεία της φόρμας. Το script είναι ένα πρόγραμμα το οποίο τρέχει στον Server και το οποίο μεταξύ των άλλων μπορεί να δέχεται σαν είσοδο δεδομένα τα οποία λαμβάνει ο browser του επισκέπτη (συνήθως από μια φόρμα). Το script, χρησιμοποιώντας μια script γλώσσα προγραμματισμού (π.χ. asp,php κτλ.) , επεξεργάζεται τα δεδομένα αυτά και έπειτα επιστρέφει στον browser μια HTML σελίδα. Η επεξεργασία αυτή μπορεί να είναι από απλές πράξεις μεταξύ των δεδομένων έως και αναζήτηση σε μια περίπλοκη Βάση Δεδομένων με βάση τα δεδομένα αυτά (ένα τέτοιο script περιέχει κατάλληλες εντολές ώστε να συνδεθεί σε μια Βάση Δεδομένων και να προσπελάσει αλλά και να προσθέσει και να διαγράψει εγγραφές). Η method καθορίζει τον τρόπο με τον οποίο στέλνονται τα δεδομένα της φόρμας στον Server που βρίσκεται το πρόγραμμα script που θα τα επεξεργαστεί και μπορεί να πάρει τις τιμές get ή post. [20-22]

Παράδειγμα χρήσης της ετικέτας <form> στην εφαρμογή του Skybox:

```
<html>
<body>
  <form action="LogIn.php" method="post">
    <input class="button" type="submit" value="LogIn" name="submission">
  </form>
</body>
</html>
```

### 3.2.1 Post – Get

Το πρωτόκολλο μεταφοράς υπερκειμένου (HTTP) έχει σχεδιαστεί ώστε να επιτρέπει την επικοινωνία μεταξύ πελατών (clients) και εξυπηρετητών (servers). Το HTTP λειτουργεί ως ένα πρωτόκολλο αίτησης-απόκρισης μεταξύ ενός πελάτη και διακομιστή. Ένας περιηγητής web μπορεί να είναι ο πελάτης, και μια εφαρμογή σε έναν υπολογιστή που φιλοξενεί μια ιστοσελίδα μπορεί να είναι ο server.

Στην HTML οι δύο πιο συχνές μέθοδοι που χρησιμοποιούνται για αιτήματα-αποκρίσεις μεταξύ ενός πελάτη και του διακομιστή είναι η Post και η Get. Η **Post** υποβάλλει δεδομένα για επεξεργασία σε ένα συγκεκριμένο πόρο, ενώ η **Get** ζητά δεδομένα από έναν καθορισμένο πόρο.

Μερικά από τα χαρακτηριστικά της μεθόδου Get είναι :

- Τα αιτήματα μπορούν να αποθηκεύονται προσωρινά
- Να παραμένουν στο ιστορικό του προγράμματος περιήγησης
- Να γίνουν σελιδοδείκτες
- Δεν πρέπει ποτέ να χρησιμοποιούνται όταν υπάρχουν ευαίσθητα δεδομένα
- Έχουν περιορισμό για το μέγεθος των δεδομένων
- Θα πρέπει να χρησιμοποιούνται μόνο για την ανάκτηση δεδομένων.

Αντίστοιχα και μερικά της μεθόδου Post :

- Τα αιτήματα δεν είναι ποτέ αποθηκευμένα
- Δεν παραμένουν στο ιστορικό του προγράμματος περιήγησης
- Δεν μπορούν να γίνουν σελιδοδείκτες
- Δεν έχουν περιορισμό για το μέγεθος των δεδομένων

Όταν ο χρήστης συμπληρώνει μια φόρμα και πατάει το κουμπί υποβολής τα δεδομένα της φόρμας αποστέλλονται για επεξεργασία σε ένα αρχείο PHP. Και η Get και η Post δημιουργούν έναν πίνακα που περιέχει όλα τα ζεύγη κλειδιού / τιμής, όπου τα κλειδιά είναι τα ονόματα των στοιχείων ελέγχου της φόρμας και οι τιμές είναι τα δεδομένα εισόδου του χρήστη.

Οι μέθοδοι Get και Post αντιμετωπίζονται σαν `$_GET` και `$_POST`, οι οποίες αποτελούν superglobals, πράγμα που σημαίνει ότι είναι προσβάσιμες πάντα ανεξάρτητα από το πεδίο εφαρμογής. Το `$_GET` είναι ένας πίνακας από μεταβλητές που περνάνε στο τρέχον script μέσω των παραμέτρων την γραμμής διεύθυνσης URL, ενώ το `$_POST` μέσω της μεθόδου HTTP POST. [23]

### 3.2.2 Sessions

Στο internet ο web server δεν γνωρίζει ποιος είναι ο κάθε χρήστης, επίσης πολλές φορές χρειάζεται να μεταφέρονται δεδομένα από τη μια ιστοσελίδα στην άλλη (π.χ. username). Γι' αυτό το λόγο υπάρχει ο μηχανισμός των sessions. Δημιουργούν ένα μοναδικό κωδικό (IDentifier) για κάθε επισκέπτη ώστε να σωθούν συγκεκριμένες μεταβλητές. Για να αρχίσουμε ένα session πρέπει να κληθεί η εντολή session\_start πριν την δήλωση του <html> tag. Και για να καταστρέψουμε ένα session χρησιμοποιούμε την εντολή session\_destroy, η οποία σβήνει όλα τα δεδομένα που είχαν αποθηκευτεί στο session. Η εντολή αυτή χρησιμοποιείται συνήθως όταν ο χρήστης κάνει logout.

Παράδειγμα :

```
<? php
    session_start();
    $_SESSION['var']=1;
?>
<html>
<body>
    <?php
        echo "ΔΕΔΟΜΕΝΑ=".SESSION['var'];
    ?>
</body>
</html>
```

### 3.2.3 Αντικείμενα Φόρμας

Με την ετικέτα <input> εισάγουμε τα περισσότερα στοιχεία της φόρμας. Οι κυριότερες ιδιότητες της ετικέτας είναι η type η οποία καθορίζει τον τύπο του στοιχείου της φόρμας (Πεδίο Κειμένου ή Περιοχή Κειμένου ή Κουμπί Επιλογών ή Κουτί Πολλαπλών Επιλογών ή Κουμπί), η name με την οποία δίνουμε ένα μοναδικό όνομα στο στοιχείο της φόρμας (δεν πρέπει να υπάρχουν στοιχεία φόρμας με τα ίδια ονόματα) και η value με την οποία δίνουμε μια αρχική τιμή στο στοιχείο της φόρμας. Η ετικέτα <input> δεν έχει ετικέτα τέλους. [24-25]

Στον παρακάτω πίνακα θα δούμε μερικά input και την λειτουργία τους.

<b>Text Fields</b>	Ορίζει ένα πεδίο όπου ο χρήστης μπορεί να εισάγει κείμενο.
<b>Password Field</b>	Ορίζει ένα πεδίο για τον κωδικό πρόσβασης.
<b>Radio Button</b>	Ορίζει ένα κουμπί.
<b>Checkboxes</b>	Ορίζει την πολλαπλή επιλογή στα κουτάκια.
<b>Submit Button</b>	Το κουμπί αυτό χρησιμοποιείται για να αποστέλλει τα δεδομένα της φόρμας σ' έναν διακομιστή.

Πίνακας 3.1

### 3.3 CSS

Η css είναι μια γλώσσα υπολογιστή που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί σε μια γλώσσα σήμανσης (html,xhtml). Είναι μια γλώσσα προορισμένη να αναπτύσσει “στιλιστικά” μια ιστοσελίδα. Διαμορφώνει πολλά χαρακτηριστικά, χρώματα και στοίχιση, επίσης δίνει περισσότερες δυνατότητες από την html. Οπότε για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της css κρίνεται απαραίτητη.

Πλεονεκτήματα της css:

- Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασσική HTML.
- Ευκολότερη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός ολόκληρου site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στο site.
- Μικρότερο μέγεθος αρχείου, δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.
- Καλύτερο SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο περιεχόμενο σκέτο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexin).
- Γρηγορότερες σελίδες. Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS, ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του site μας.

Παράδειγμα μορφοποίησης μίας σελίδας μέσω ενός αρχείου CSS

```
body{
    text-align:center;
    background-color:#61b3de;
    font-family:Arial, Helvetica, sans-serif;
    font-size:80%;
    color:#666;
}
```

Εξήγηση της παραπάνω γραμμής του κώδικα:

Το κομμάτι του κώδικα εφαρμόζεται σε σελίδες που είναι συνδεδεμένες με το συγκεκριμένο αρχείο CSS και μορφοποιεί την εμφάνιση της σελίδας. Συγκεκριμένα ορίζει την στοίχιση των αντικειμένων της σελίδας να είναι στο κέντρο, το χρώμα του παρασκήνιου της σελίδας να είναι μία απόχρωση του μπλε, η γραμματοσειρά να είναι τύπου Arial και το μέγεθός της να είναι το 80% από το κανονικό και τέλος το χρώμα της να είναι μία απόχρωση του γκρι.

### 3.4 Εντολές MySQL

Η MySQL είναι ένα σύστημα διαχείρισης βάσεων δεδομένων, εύκολο και γρήγορο στην χρήση του. Η βάση δεδομένων μας επιτρέπει να αποθηκεύουμε, να αναζητάμε, να ταξινομούμε και να ανακαλούμε δεδομένα αποτελεσματικά. Η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Αυτό προκύπτει από το γεγονός ότι ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα μας, για να μπορούν να δουλεύουν ταυτόχρονα πολλοί χρήστες, για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Χρησιμοποιεί την SQL γλώσσα η οποία είναι η τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων παγκόσμια. [26-27]

#### Εντολές

- `mysql_connect([servername], [username], [password]);` : αυτή η εντολή συνδέει την εφαρμογή με την βάση και περιέχει 3 παραμέτρους. Την παράμετρο `servername` που είναι το όνομα του server, την παράμετρο `username` που είναι το όνομα χρήστη και `password` είναι ο κωδικός του χρήστη της βάσης που χρησιμοποιεί η εφαρμογή.
- `mysql_close(<connection>);` : αυτή η εντολή τερματίζει οποιαδήποτε ανοιχτή σύνδεση.
- `mysql_query("CREATE DATABASE<όνομα>", <σύνδεση>);` : η εντολή αυτή δημιουργεί μια καινούργια βάση δεδομένων.
- `mysql_query("DROP DATABASE<όνομα>", <σύνδεση>);` : με την εντολή αυτή διαγράφουμε μια βάση δεδομένων.
- `mysql_select_db(<όνομα βάσης>, <σύνδεση βάσης>);` : η εντολή αυτή επιλέγει τη βάση που θα χρησιμοποιήσουμε.

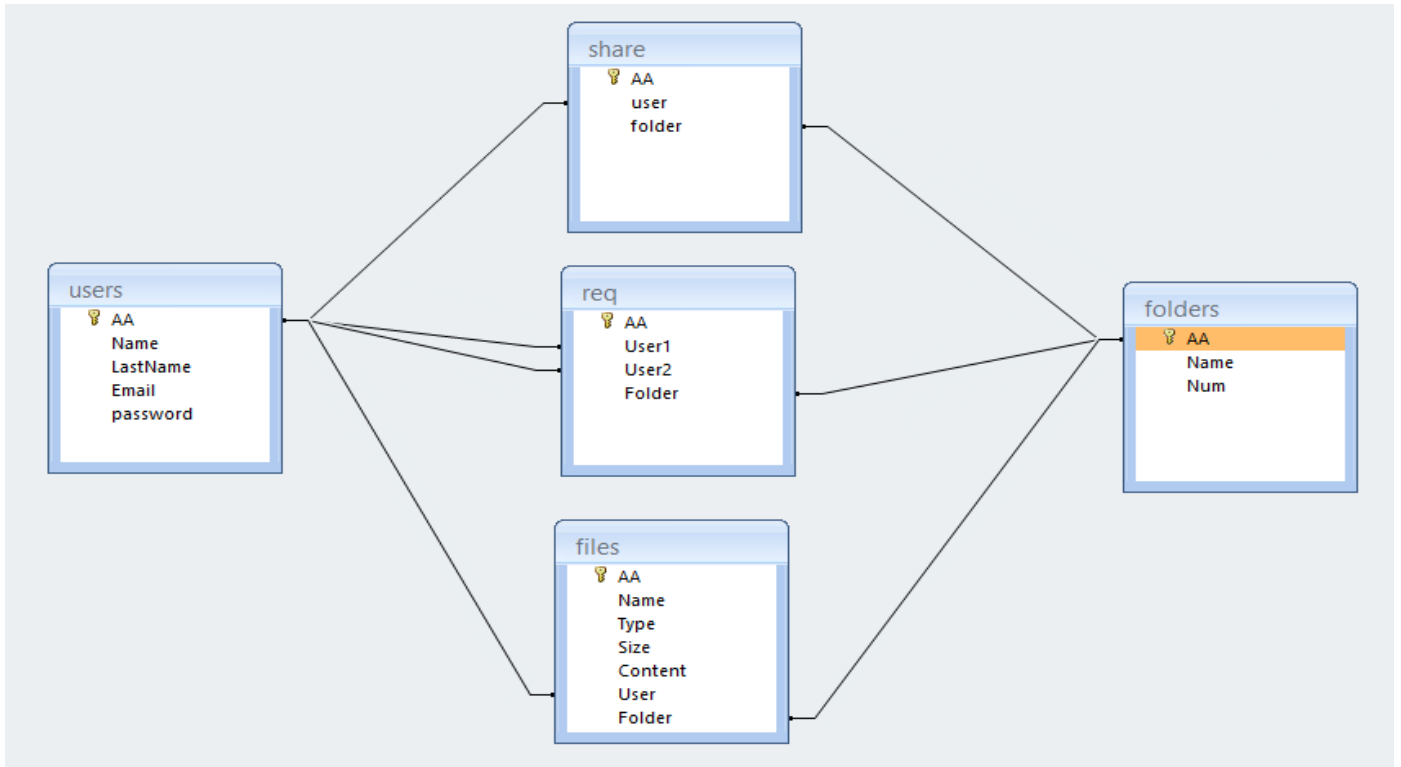
#### Παράδειγμα

Στο αρχείο `connectionDB.php` βρίσκεται ο κώδικας για την σύνδεση της βάσης με την εφαρμογή και καλείτε στην αρχή κάθε αρχείου `php` που επιθυμεί να αλληλεπιδράσει με την βάση δεδομένων της εφαρμογής.

```
<?php
    $con=mysql_connect("localhost","root","");
    if(!$con){
        die('Could not Connect to mysql:'.mysql_error());
    }
    mysql_selectdb("dropbox",$con);
?>
```

#### Εξήγηση της παραπάνω γραμμής του κώδικα:

Στην αρχή του κώδικα καλείτε η εντολή με την οποία κάνουμε την σύνδεση στην βάση δίνοντας τα στοιχεία του χρήστη Administrator. Έπειτα γίνεται έλεγχος άμα η σύνδεση επετεύχθητε και τέλος εκτελείτε η εντολή με την οποία διαλέγουμε την βάση της εφαρμογής που περιέχει όλα τα δεδομένα της.



Εικόνα 2.15 Διάγραμμα E-R της βάσης της εφαρμογής.

## ΚΕΦΑΛΑΙΟ 4<sup>ο</sup>

### Ανάλυση της Εφαρμογής

#### 4.1 ανάλυση των υπηρεσιών του skybox

Έχοντας αναφέρει όλα τα απαραίτητα εργαλεία για τον προγραμματισμό μιας εφαρμογής διαδικτύου και τον τρόπο με τον οποίο χρησιμοποιούνται αυτά ήρθε η ώρα να αναλύσουμε τις λειτουργίες που αναπτύξαμε στην εφαρμογή του skybox.

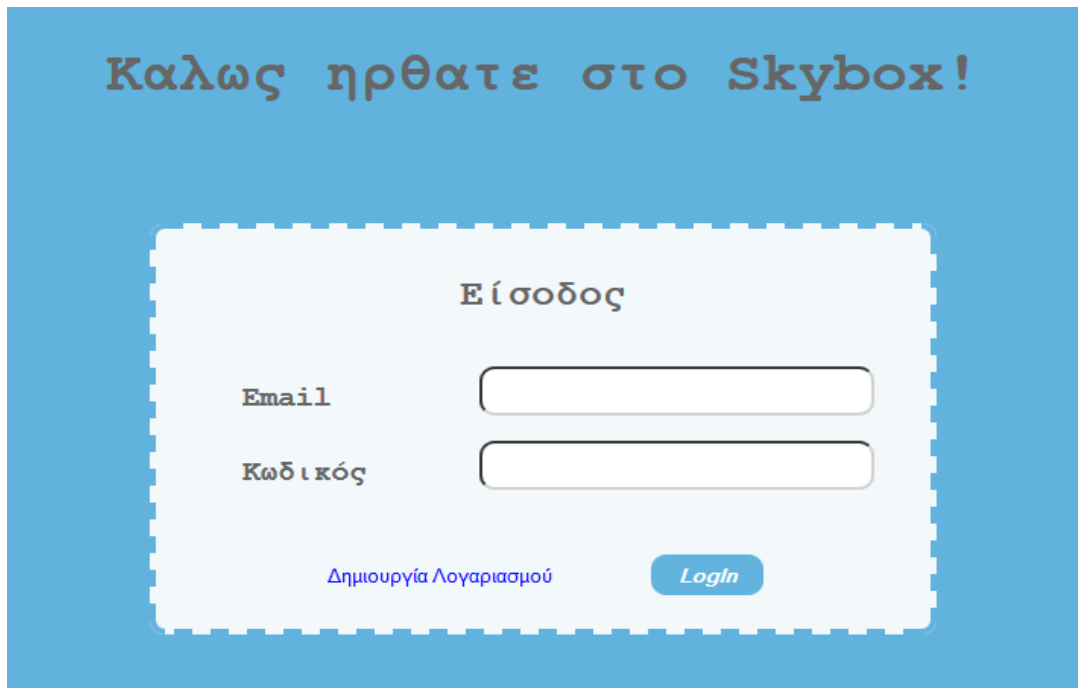
Αυτές οι λειτουργίες είναι :

- **Είσοδος στην εφαρμογή**  
Για να έχει πρόσβαση ένας χρήστης στην εφαρμογή πρέπει πρώτα να περάσει από τη διαδικασία της εισόδου στην εφαρμογή. Εφόσον έχει κάνει δημιουργία λογαριασμού δίνει στην εφαρμογή το e-mail και τον κωδικό που όρισε ο ίδιος και πατώντας το κουμπί “login” κάνει είσοδο στην εφαρμογή.
- **Δημιουργία λογαριασμού**  
Κάθε νέος χρήστης για να έχει πρόσβαση στην εφαρμογή πρέπει πρώτα να κάνει εγγραφή σ’αυτήν. Κατά τη διάρκεια της εγγραφής ο χρήστης εισάγει δεδομένα όπως το e-mail του, το όνομά του, το επώνυμό του και τον κωδικό που επιθυμεί για να κάνει είσοδο στην εφαρμογή.
- **Δημιουργία-επεξεργασία φακέλου**  
Ο χρήστης μπορεί να δημιουργήσει ένα φάκελο δίνοντάς του ένα όνομα. Στη συνέχεια μπορεί να μετονομάσει αυτό το φάκελο ή ακόμα και να τον διαγράψει εντελώς.
- **Upload και Download αρχείων**  
Κάθε χρήστης μπορεί να ανεβάσει αρχεία από τον υπολογιστή του στην εφαρμογή και να τα αποθηκεύσει είτε μέσα σε κάποιο φάκελο είτε κατευθείαν στον αρχικό κατάλογο. Επίσης έχει δικαίωμα και να κατεβάσει όποιο αρχείο επιθυμεί εφόσον έχει πρόσβαση σ’αυτό.
- **Επεξεργασία αρχείων**  
Όπως και στην επεξεργασία φακέλων έτσι και στην επεξεργασία αρχείων ο χρήστης έχει το δικαίωμα είτε να αλλάξει το όνομα κάποιου αρχείου είτε να το διαγράψει.
- **Διαμοιρασμός φακέλων μεταξύ χρηστών**  
Το πρόγραμμα προσφέρει και τη δυνατότητα να γίνει διαμοιρασμός φακέλων μεταξύ των χρηστών της εφαρμογής. Ένας χρήστης εφόσον έχει δημιουργήσει κάποιο φάκελο μπορεί να πατήσει στο κουμπί “share” και να ηλεκτρολογήσει το e-mail του χρήστη με τον οποίο επιθυμεί να μοιράζεται τα αρχεία που ανήκουν σ’αυτόν τον φάκελο.  
Για να ολοκληρωθεί ο διαμοιρασμός των φακέλων θα πρέπει με τη σειρά του ο παραλήπτης της αίτησης να την αποδεχτεί.

Εφόσον έχει ολοκληρωθεί η παραπάνω διαδικασία πλέον όσοι χρήστες διαμοιράζονται τον φάκελο έχουν ίσα δικαιώματα σ' αυτόν καθώς και στα αρχεία που περιέχονται μέσα στον φάκελο.

## 4.2 Είσοδος στην εφαρμογή

Η εμφάνιση της σελίδας όπου γίνεται είσοδος στην εφαρμογή έχει την εξής μορφή :



Εικόνα 4.1 Αρχική σελίδα της εφαρμογής.

Ο κώδικας html για την εμφάνιση της σελίδας εισόδου της εφαρμογής βρίσκεται στο αρχείο LogIn.html και είναι :

```
<html>
<head>
<title>SkyBox Είσοδος</title>
<link rel='stylesheet' type='text/css' href='LogIn.css' />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<script type="text/javascript">
function stopRKey(evt) {
  var evt = (evt) ? evt : ((event) ? event : null);
  var node = (evt.target) ? evt.target : ((evt.srcElement) ? evt.srcElement : null);
  if ((evt.keyCode == 13) && (node.type=="text")) {return false;}
  if ((evt.keyCode == 13) && (node.type=="password")) {return false;}
}

document.onkeypress = stopRKey;
</script>

</head>
<body>

<form action="LogIn.php" method="post">
<pre><Center> <p style="font-size:40px; font-weight:bold;">Καλως ηρθατε στο Skybox!</p></Center>
<div class="wrap">
```





```

        color: #fff;
        font-weight: bold;
        font-style: italic;
        padding: 6px 15px 5px;
        cursor: pointer;
    }
    .input{
        background-color: #fff;
        height: 35px;
        border-radius: 10px;
        border-color: #666;
    }
}

```

Τέλος ο κώδικας της εκτέλεσης των εντολών για την είσοδο του χρήστη στην εφαρμογή μέσω της σελίδας εισόδου βρίσκεται στο αρχείο LogIn.php και είναι :

```

<?php
session_start();
$access=false;

require'connectionDB.php';

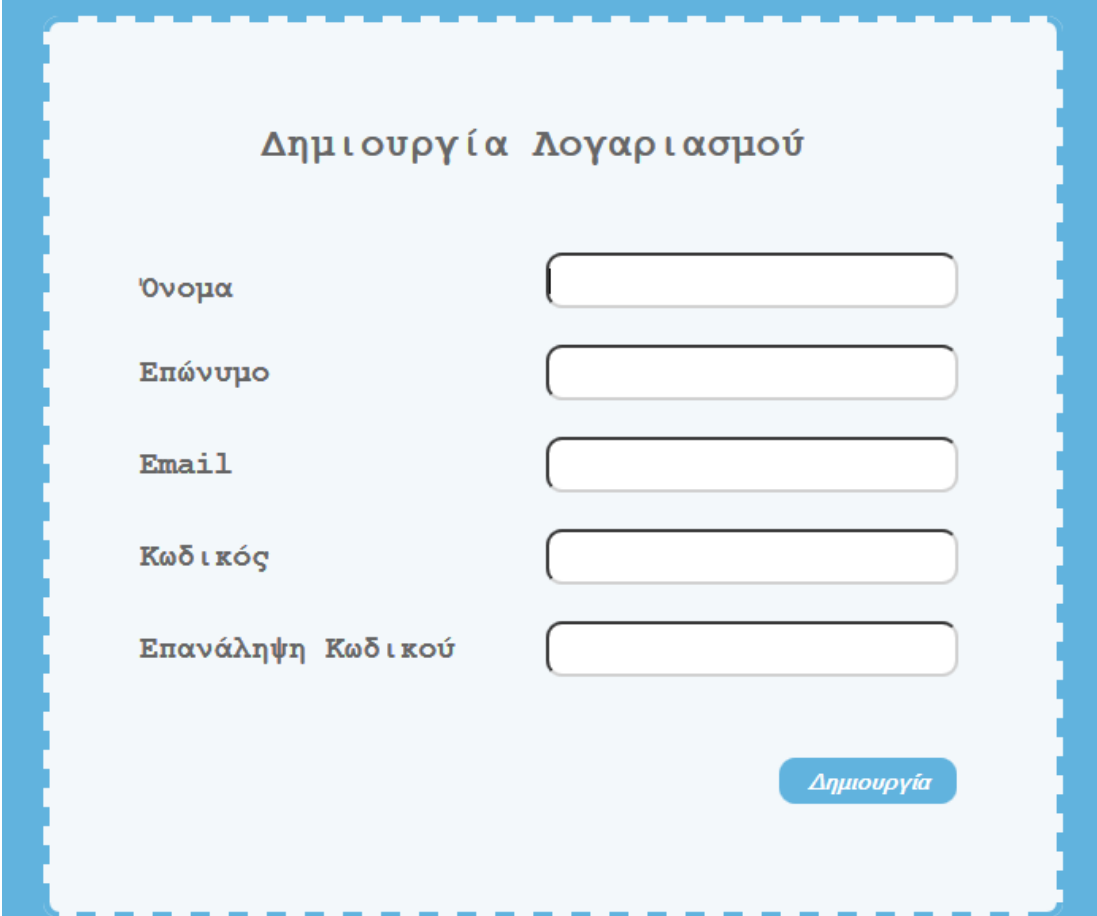
if ( $_POST["submission"] == "LogIn")
{
    $email=$_POST["email"];
    $pass=$_POST["password"];
    $result=mysql_query("SELECT * FROM users WHERE Email='$email' AND password='$pass'");
    $row=mysql_fetch_array($result);
    if($row)
        $access=true;
    if($access==true)
    {
        $_SESSION['log']=$row['AA'];
        include ("main.php");
    }
    else
    {
        echo file_get_contents("LogIn.html");
        echo "<p style=margin-top: -130px;'><CENTER><font color='red'> Λάθος Email ή Κωδικός!
</font></CENTER></p>";
    }
}
else
    echo file_get_contents("SignUp.html");

?>

```

### 4.3 Δημιουργία λογαριασμού

Για να επιτευχθεί η δημιουργία λογαριασμού νέου χρήστη στην εφαρμογή θα πρέπει ο χρήστης να συμπληρώσει τα στοιχεία του στην παρακάτω σελίδα.



The image shows a registration form with the following fields and a button:

- Όνομα
- Επώνυμο
- Email
- Κωδικός
- Επανάληψη Κωδικού
- Δημιουργία

Εικόνα 4.2 Δημιουργία νέου χρήστη στην εφαρμογή.

Η εμφάνιση της παραπάνω σελίδας οφείλεται στον κώδικα του αρχείου SignUp.html και είναι ο ακόλουθος :

```
<html>
<head>
<title>SkyBox Δημιουργία Λογαριασμού</title>
<link rel='stylesheet' type='text/css' href='LogIn.css' />
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
<script type="text/javascript">
function stopRKey(evt) {
    var evt = (evt) ? evt : ((event) ? event : null);
    var node = (evt.target) ? evt.target : ((evt.srcElement) ? evt.srcElement : null);
    if ((evt.keyCode == 13) && (node.type=="text")) {return false;}
    if ((evt.keyCode == 13) && (node.type=="password")) {return false;}
}

document.onkeypress = stopRKey;
</script>
</head>
<body bgcolor="#E6E6FA">
<form action="SignUp.php" method="post">
<pre><div class="wrap2">
<table width="600" align="center" border="3" cellpadding="3" cellspacing="1" bgcolor="transparent"
bordercolor="transparent">
```



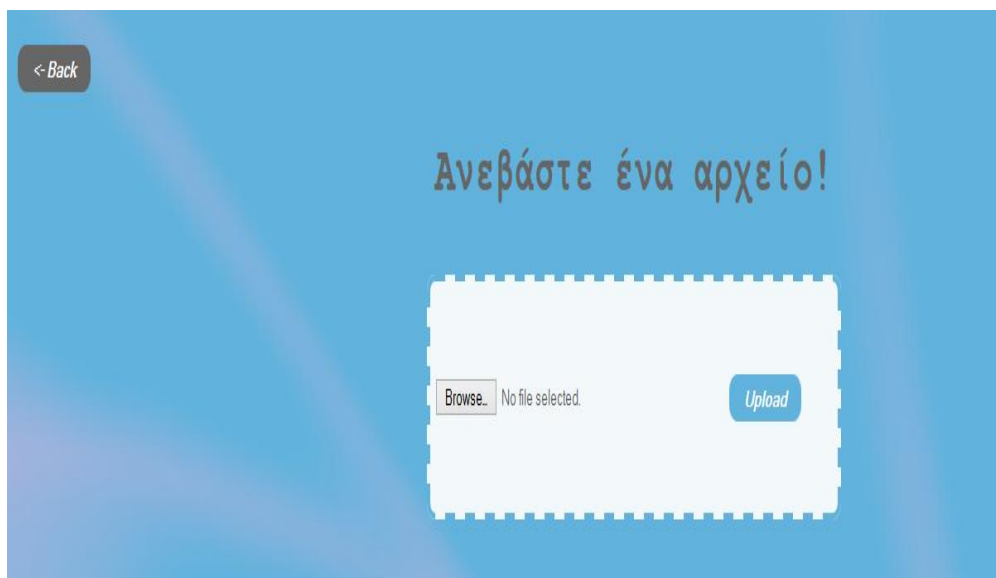
```

        echo file_get_contents("SignUp.html");
        echo "<p style='margin-top: -170px;'><CENTER><font color='red'> You must fill in all fields!
</font></CENTER></p>";
    }
    if($flag == true)
        if($password == $password2)
            $flag = true;
        else
        {
            $flag = false;
            echo file_get_contents("SignUp.html");
            echo "<p style='margin-top: -170px;'><CENTER><font color='red'> Passwords do not match!
</font></CENTER></p>";
        }
    if($flag == true)
    {
        mysql_query("INSERT INTO users (Name,LastName,Email,password)
        VALUES('$name','$lastname','$email','$password')");
        echo file_get_contents("LogIn.html");
        echo "<p style='margin-top: -405px;'><CENTER><font color='red'> Success!
</font></CENTER></p>";
    }
    ?>

```

#### 4.4 Upload και Download αρχείων

Για να ανεβάσει ένας χρήστης ένα αρχείο είτε σ'ένα φάκελο είτε στον αρχικό κατάλογο θα πρέπει να πατήσει το κουμπί με την ονομασία “upload” το οποίο και θα εμφανίσει το εξής παράθυρο :



Εικόνα 4.3 Upload αρχείων από τον Η/Υ.

Το αρχείο επεξεργασίας της παραπάνω σελίδας ονομάζεται upload.php, περιέχει τον κώδικα html και php για την εκτέλεση της λειτουργίας του upload και περιλαμβάνει τον ακόλουθο κώδικα :

```

<?php
$usr=$_GET['w1'];
$fold=$_GET['w2'];

```

```

?>

<html>
<head>
<title>SkyBox Upload</title>
<link rel='stylesheet' type='text/css' href='upload.css' />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<script type="text/javascript" src="//code.jquery.com/jquery-2.1.0.min.js"></script>
</head>
<body>
<pre>
<form action="back.php" method="post"><input name="<?php echo $usr ?>" type="submit" class="back"
value="<- Back" id= "back"></form>
<Center> <p style="font-size:40px; font-weight:bold;">Ανεβάζετε ένα αρχείο!</p></Center> <form
method="post" enctype="multipart/form-data">
<table width="350" border="0" cellpadding="1" cellspacing="1" class="wrap">
<tr>
<td width="246">
<input type="hidden" name="MAX_FILE_SIZE" value="2000000">
<input name="userfile" type="file" id="userfile">
</td>
<td width="80"><input name="upload" type="submit" class="button" id="upload" value=" Upload "></td>
</tr>
</table>
</form></pre>
</body>
</html>
<script>
function submitForm(){
document.getElementById('back').click();
}
</script>

<?php
require'connectionDB.php';
if(isset($_POST['upload']) && $_FILES['userfile']['size'] > 0)
{
$fileName = $_FILES['userfile']['name'];
$tmpName = $_FILES['userfile']['tmp_name'];
$fileSize = $_FILES['userfile']['size'];
$fileType = $_FILES['userfile']['type'];
$f = fopen($tmpName, 'r');
$content = fread($f, filesize($tmpName));
$content = addslashes($content);
fclose($f);
if(!get_magic_quotes_gpc())
{
$fileName = addslashes($fileName);
}

$query = "INSERT INTO files (Name, Size, Type, Content, User, Folder) ".
"VALUES ('$fileName', '$fileSize', '$fileType', '$content', '$usr', '$fold)";
mysql_query($query) or die('Error, query failed');
echo "<br>File $fileName uploaded<br>";
if($fold==0)
echo '<script type="text/javascript"> submitForm(); </script>';
else
echo '<script type="text/javascript"> window.location.href = "folder.php?AA=" + '.$fold.' + "&w1=" +
'.$usr.'; </script>';
}
?>

```

Αντίστοιχα το αρχείο μορφοποίησης της σελίδας και των αντικειμένων που περιέχονται σ' αυτήν βρίσκεται στη σελίδα upload.css και είναι :

```

body{
text-align:center;
background-image: url('background2.png');
background-repeat: no-repeat;
background-size:cover;
font-family:Arial, Helvetica, sans-serif;
font-size:80%;
color:#666;
}
.wrap{
background: #f3f8fb;
width:500px;
height:150px;
margin:0px auto;
border: 4px dashed #61b3de;
border-radius:10px;
padding: 5px 5px;
}
.button{
border-radius: 10px;
background-color: #61b3de;
border: 0;
color: #fff;
font-weight: bold;
font-style: italic;
padding: 6px 15px 5px;
cursor: pointer;
}
.back{
border-radius: 10px;
border: 0;
color: #fff;
font-weight: bold;
font-style: italic;
padding: 6px 15px 5px;
cursor: pointer;
background-color: #666;
margin-left: -1400px;
margin-top: 20px;
}

```

Η διαδικασία του download εφαρμόζεται κάθε φορά που ο χρήστης πατάει πάνω στο όνομα ενός αρχείου. Ο κώδικας αυτής της λειτουργίας βρίσκεται στο αρχείο download.php και είναι ο ακόλουθος :

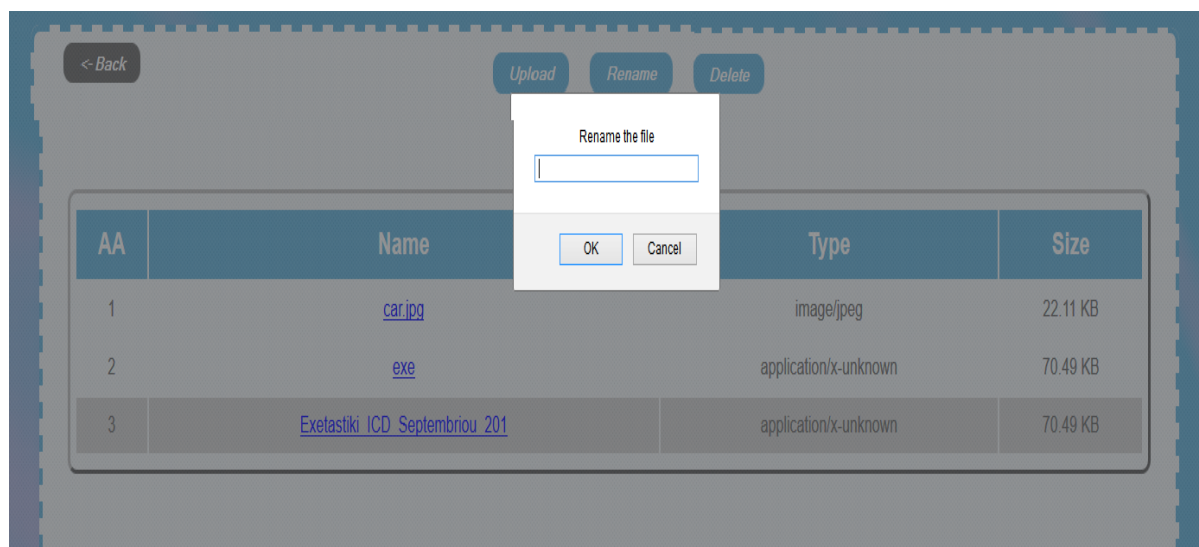
```

<?php
require'connectionDB.php';
//php code to download the clicked file from the table
if(isset($_GET['AA']))
{
// if id is set then get the file with the id from database
$AA = $_GET['AA'];
$query = "SELECT Name, Type, Size, Content " .
"FROM files WHERE AA = '$AA'";
$result = mysql_query($query) or die('Error, query failed');
list($Name, $Type, $Size, $Content) =mysql_fetch_array($result);
header("Content-length: $Size");
header("Content-type: $Type");
header("Content-Disposition: attachment; filename=$Name");
echo $content;
exit;
}
?>

```

## 4.5 Επεξεργασία αρχείων

Τα αρχεία που εμφανίζονται στον πίνακα όταν κάνει είσοδο ο χρήστης μπορούν να υποστούν επεξεργασία από τον ίδιο. Συγκεκριμένα ο χρήστης μπορεί να μετονομάσει το αρχείο ή να το διαγράψει. Η μετονομασία γίνεται πατώντας το κουμπί rename και η διαγραφή πατώντας το κουμπί delete εφόσον πρώτα έχει επιλέξει το αρχείο από τον πίνακα. Μόλις πατήσει το κουμπί rename θα εμφανιστεί ο παρακάτω πίνακας διαλόγου:



Εικόνα 4.4 Μετονομασία αρχείου.

Ο κώδικας για την εμφάνιση της παραπάνω σελίδας που υλοποιεί την εμφάνιση αποτελεσμάτων, το ανέβασμα, την μετονομασία και την διαγραφή αρχείων βρίσκεται στο παραπάνω αρχείο folder.php και είναι ο εξής :

```
<?php
if(isset($_GET['w1']))
{
    $saa=$_GET['w1'];
    require'connectionDB.php';
    $result=mysql_query("SELECT * FROM users WHERE AA='$saa'");
    $row=mysql_fetch_array($result);
    $fold= $_GET['AA'];
    $result2=mysql_query("SELECT Name FROM folders WHERE AA='$fold'");
    $row2=mysql_fetch_array($result2);
    $foldName=$row2['Name'];
}
else
{
    echo "You are not Logged In!";
    include ("LogIn.html");
    exit;
}
?>
<html>
<head>
<title>SkyBox <?php echo $foldName; ?></title>
<link rel='stylesheet' type='text/css' href='main.css' />
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
<meta content="charset=utf-8"/>
<script type="text/javascript" src="//code.jquery.com/jquery-2.1.0.min.js"></script>

<script>
var preEl ;
```



```

var orgBColor;
var orgTColor;
function HighLightTR(el, backColor,textColor){
  if(typeof(preEl)!='undefined') {
    preEl.bgColor=orgBColor;
    try{ ChangeTextColor(preEl,orgTColor);} catch(e){;}
  }
  orgBColor = el.bgColor;
  orgTColor = el.style.color;
  el.bgColor=backColor;
  try{ ChangeTextColor(el,textColor);} catch(e){;}
  preEl = el;
  document.getElementById('row').value=el.rowIndex;
}
function ChangeTextColor(a_obj,a_color){ ;
  for (i=0;i<a_obj.cells.length;i++)
    a_obj.cells(i).style.color=a_color;
}
</script>
<script>
function Rename() {
if(document.getElementById('row').value>0)
{
var user = "<?php echo $aa ?>";
var filename=document.getElementById('row').value;
var action = "rename";
var AA="<?php echo $fold ?>";
var name = prompt("Rename the file", "");
if (name != null)
window.location.href = "folderFunc.php?w1=" + user + "&w2=" + name + "&w3=" + action + "&w4=" + filename
+ "&w5=" + AA;
}
}
function Delete() {
if(document.getElementById('row').value>0)
{
var user = "<?php echo $aa ?>";
var filename=document.getElementById('row').value;
var action = "delete";
var AA="<?php echo $fold ?>";
window.location.href = "folderFunc.php?w1=" + user + "&w2=" + filename + "&w3=" + action + "&w4=" + AA;
}
}
function Upload() {
var user = "<?php echo $aa ?>";
var folder = "<?php echo $fold ?>";
window.location.href = "upload.php?w1=" + user + "&w2=" + folder;
}
function Exit() {
window.location.href = "LogIn.html";
}
</script>
</head>
<body>
<table class="tbl" border="0" align="right">
<tr>
<td><CENTER><p style="margin-bottom: 20px; margin-top: 0px;"><?php echo 'Welcome ' . $row['Name'] . '
.$row['LastName']; ?></p></CENTER></td>
</tr>
<tr>
<td><CENTER><input class="button" type="submit" value="Exit" name="submission"
onClick="Exit()"></CENTER></td>
</tr>
</table>

<div class="wrap"><pre>

```

```

<CENTER><input class="button" type="submit" value="Upload" name="submission" onClick="Upload()">
<input class="button" type="submit" value="Rename" name="submission" onClick="Rename()" id="rename">
<input class="button" type="submit" value="Delete" name="submission" onClick="Delete()"
id="delete"></CENTER>
<input type="submit" value="0" name="submission" id="row" style="color: transparent; background-color:
transparent; border: 0px;">
<CENTER><p style="font-size:28px; font-weight:bold; margin-top: -30px;"><?php echo $foldName;
?></p></CENTER>
<form action="back.php" method="post"><input name="<?php echo $aa ?>" type="submit" class="back"
value="<- Back" id="back"></form>
<table class="tbl2" id="mytable" align="center">
<tr bgcolor="#61b3de" height="50px">
<th><p style="color: white; font-size:22">AA</p></th>
<th><p style="color: white; font-size:22">Name</p></th>
<th><p style="color: white; font-size:22">Type</p></th>
<th><p style="color: white; font-size:22">Size</p></th>
</tr>
<?php
$num=1;
$query = "SELECT AA, Name, Type, Size FROM files Where User='$aa' And Folder='$fold'";
$result = mysql_query($query) or die('Error, query failed');
while(list($AA, $Name, $Type, $Size) = mysql_fetch_array($result))
{
$Size=$Size/1024;
$Size = floor(($Size*100))/100;
echo "<tr onClick='HighLightTR(this,'lightgrey','cc3333');\'' height='40px\''>";
echo "<td><CENTER>$num</td>";
echo "<td><CENTER><a href='download.php?AA=". $AA. "'>.$Name.</a></td>";
echo "<td><CENTER>$Type</td>";
echo "<td><CENTER>$Size KB</td>";
echo "</tr>";
$num++;
}
?>
</table>
</pre></div>
</body>
</html>
?>

```

Η υλοποίηση των παραπάνω λειτουργιών βρίσκεται στο αρχείο folderFunc.php και περιέχει τα εξής :

```

<?php
require'connectionDB.php';
$user=$_GET['w1'];
$action = $_GET['w3'];
if($action=="rename"){
$flag = $_GET['w4'];
$rename = $_GET['w2'];
$a = $_GET['w5'];
$query = "SELECT AA,Name FROM files Where User='$user' and Folder ='$a'";
$result= mysql_query($query);
$k=1;
while ( $row = mysql_fetch_array( $result ) )
{
if ( $k == $flag )
{
$a= $row['AA'];
$nm= $row['Name'];
}
$k++;
}
$pin=explode(".", $nm);
if(isset($pin[1]))
$rename=$rename.".".$pin[1];

```

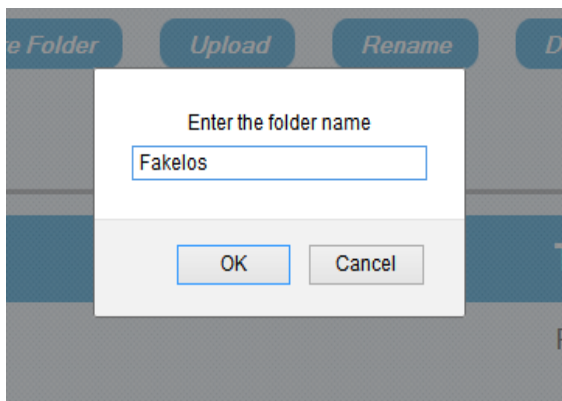
```

$update="UPDATE files SET Name = '$rename' where AA='$a'";
mysql_query($update);
}
elseif($action == "delete"){
$flag = $_GET['w2'];
$a = $_GET['w4'];
$query = "SELECT AA FROM files Where User='$user' and Folder='$a'";
$result = mysql_query($query);
$k=1;
while ( $row = mysql_fetch_array( $result ) )
{if ( $k == $flag )
$a= $row['AA'];
$k++;
}
$del="DELETE FROM files WHERE AA='$a'";
mysql_query($del);
}
session_start();
$_SESSION['log']=$user;
include ("main.php");
//echo '<script type="text/javascript"> window.location.href = "folder.php?AA=" + '.$a.' + "&w1=" + '.$user.';
?>

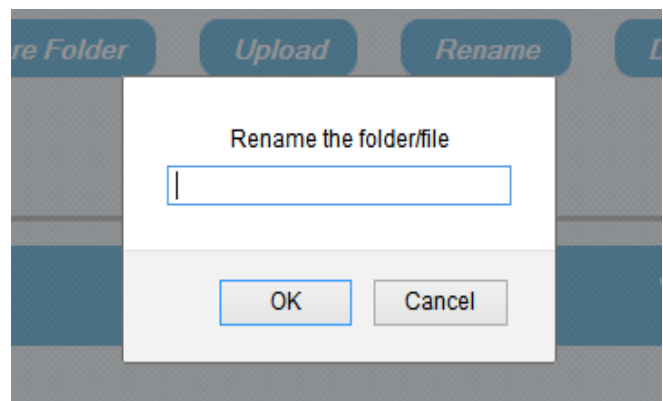
```

#### 4.6 Δημιουργία-επεξεργασία φακέλου

Κάθε χρήστης έχει τη δυνατότητα εκτός από την επεξεργασία αρχείων να επεξεργαστεί και φακέλους. Δηλαδή μπορεί να δημιουργήσει έναν φάκελο, να αλλάξει όνομα σε κάποιο φάκελο, ή να διαγράψει φακέλους που έχει δικαίωμα πρόσβασης. Για να δημιουργήσει κάποιο φάκελο πρέπει να πατήσει το κουμπί “create folder”, για να μετονομάσει πρέπει να πατήσει το κουμπί “rename” και για να διαγράψει κάποιο φάκελο το κουμπί “delete”. Για τις δύο τελευταίες λειτουργίες είναι απαραίτητο ο χρήστης πρώτα να επιλέξει το φάκελο από τον πίνακα.



Εικόνα 4.5 Δημιουργία φακέλου.



Εικόνα 4.6 Μετονομασία φακέλου

Ο κώδικας για την εμφάνιση των παραπάνω λειτουργιών βρίσκεται στο αρχείο main.php και έχει την εξής μορφή :

```

<?php
if(isset($_SESSION['log']))
{
$a=$_SESSION['log'];
require'connectionDB.php';

```

```

$result=mysql_query("SELECT * FROM users WHERE AA='$aa'");
$row=mysql_fetch_array($result);
unset($_SESSION['log']);
}
else
{
echo "You are not Logged In!";
include ("LogIn.html");
exit;
}
?>
<html>
<head>
<title>SkyBox</title>
<link rel='stylesheet' type='text/css' href='main.css' />
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
<meta content="charset=utf-8"/>
<script type="text/javascript" src="//code.jquery.com/jquery-2.1.0.min.js"></script>
<script>
var preEl ;
var orgBColor;
var orgTColor;
function HighLightTR(el, backColor,textColor){
if(typeof(preEl)!='undefined') {
preEl.bgColor=orgBColor;
try{ChangeTextColor(preEl,orgTColor);}catch(e){;}
}
orgBColor = el.bgColor;
orgTColor = el.style.color;
el.bgColor=backColor;
try{ChangeTextColor(el,textColor);}catch(e){;}
preEl = el;
document.getElementById('row').value=el.rowIndex;
if(el.rowIndex > document.getElementById('folderNum').value)
document.getElementById('share').disabled=true;
else
document.getElementById('share').disabled=false;
}
function ChangeTextColor(a_obj,a_color){ ;
for (i=0;i<a_obj.cells.length;i++)
a_obj.cells(i).style.color=a_color;
}
</script>
<script>
function Create() {
var user = "<?php echo $aa ?>";
var name = prompt("Enter the folder name", "");
var action = "create";
if (name != null)
window.location.href = "functions.php?w1=" + user + "&w2=" + name + "&w3=" + action;}
function Rename() {
if(document.getElementById('row').value>0)
{var user = "<?php echo $aa ?>";
var filename=document.getElementById('row').value;
var action = "rename";
var name = prompt("Rename the folder/file", "");
if (name != null)
window.location.href = "functions.php?w1=" + user + "&w2=" + name + "&w3=" + action + "&w4=" + filename}
}
function Delete() {
if(document.getElementById('row').value>0)
{var user = "<?php echo $aa ?>";
var filename=document.getElementById('row').value;
var action = "delete";
window.location.href = "functions.php?w1=" + user + "&w2=" + filename + "&w3=" + action;
}
}
}
}

```

```

function Upload() {
var user = "<?php echo $aa ?>";
var folder = "0";
window.location.href = "upload.php?w1=" + user + "&w2=" + folder;
}
function Share(){
if(document.getElementById('row').value>0){
var user = "<?php echo $aa ?>";
var filename=document.getElementById('row').value;
var email = prompt("Give the Users Email", "");
var action = "share";
if (email != null)
window.location.href = "functions.php?w1=" + user + "&w2=" + email+ "&w3=" + action + "&w4=" +
filename;}}
function Requests(){
var user = "<?php echo $aa ?>";
window.location.href = "requests.php?w1=" + user;
}
function Exit() {
window.location.href = "LogIn.html";
}
</script>
</head>
<body>
<table class="tbl" border="0" align="right">
<tr>
<td><CENTER><p style="margin-bottom: 20px; margin-top: 0px;"><?php echo 'Welcome ' . $row['Name'].
'. $row['LastName']; ?></p></CENTER></td>
</tr>
<tr>
<td><CENTER><input class="button" type="submit" value="Exit" name="submission"
onClick="Exit()"></CENTER></td>
</tr> </table>
<div class="wrap"><pre>
<CENTER><input class="button" type="submit" value="Create Folder" name="submission" onClick="Create()">
<input class="button" type="submit" value="Share Folder" name="submission" onClick="Share()" id="share">
<input class="button" type="submit" value="Upload" name="submission" onClick="Upload()"> <input
class="button" type="submit" value="Rename" name="submission" onClick="Rename()" id="rename"> <input
class="button" type="submit" value="Delete" name="submission" onClick="Delete()" id="delete"> <input
class="button" type="submit" value="Pendings" name="submission" onClick="Requests()"></CENTER>
<input type="submit" value="0" name="submission" id="row" style="color: transparent; background-color:
transparent; border: 0px;">
<table class="tbl2" id="mytable" align="center">
<tr bgcolor="#61b3de" height="50px">
<th><p style="color: white; font-size:22">AA</p></th>
<th><p style="color: white; font-size:22">Name</p></th>
<th><p style="color: white; font-size:22">Type</p></th>
<th><p style="color: white; font-size:22">Size</p></th>
</tr>
<?php
$query = "SELECT AA, Name, Num FROM folders";
$result = mysql_query($query) or die('Error, query failed');
$num=1;
while(list($AA, $Name, $Num) = mysql_fetch_array($result))
{if($Num == 1)
$num="Folder"
else
$num="Shared Folder";
echo "<tr onClick='\"HighLightTR(this,'lightgrey','cc3333');\" height=\"40px\">";
echo "<td><CENTER>$num</CENTER></td>";
echo "<td><CENTER><a href='folder.php?AA=".$AA."&w1=".$aa."'>.$Name.</a></CENTER></td>";
echo "<td><CENTER>$num</CENTER></td>";
echo "<td><CENTER>0</CENTER></td>";
echo "</tr>";
$num++;}
$folderNum=$num-1;
$query = "SELECT AA, Name,Type,Size FROM files Where User='$aa' And Folder=0";

```

```

$result = mysql_query($query) or die('Error, query failed');
while(list($AA, $Name, $Type, $Size) = mysql_fetch_array($result))
{
    $Size=$Size/1024;
    $Size = floor(($Size*100))/100;
    echo "<tr onClick='HighLightTR(this,'lightgrey','cc3333');\|" height=\|40px\|>";
    echo "<td><CENTER>$num</CENTER></td>";
    echo "<td><CENTER><a href='download.php?AA=".$AA.">".$Name."</a></CENTER></td>";
    echo "<td><CENTER>$Type</CENTER></td>";
    echo "<td><CENTER>$Size KB</CENTER></td>";
    echo "</tr>";
    $num++;
}
?>
</table>
<input class="button" type="submit" value="<?php echo $folderNum; ?>" name="submission" id="folderNum"
style="margin-top: -500px auto">
</pre></div>
</body>
</html>

```

Οι λειτουργίες του παραπάνω κώδικα υλοποιούνται στη σελίδα functions.php και περιέχουν :

```

<?php
require'connectionDB.php';
$user=$_GET['w1'];
$action = $_GET['w3'];
if($action=="create"){
    $name=$_GET['w2'];
    $query = "INSERT INTO folders (Name, Num ) VALUES ('$name', '1')";
    mysql_query($query) or die('Error, query failed')
}
elseif($action=="share"){
    $email= $_GET['w2'];
    $flag= $_GET['w4'];
    $query = "SELECT AA FROM folders";
    $result = mysql_query($query) or die('Error, query failed');
    $k=1;
    while ( $row = mysql_fetch_array( $result ) )
    {if ( $k == $flag )
    $a= $row['AA'];
    $k++;
    }
    $query = "SELECT AA FROM users WHERE Email='$email'";
    $result = mysql_query($query) or die('Error, query failed');
    $row = mysql_fetch_array( $result );
    $to=$row['AA'];
    $query2 = "INSERT INTO req (User1, User2, Folder ) VALUES ('$to', '$user', '$a')";
    mysql_query($query2) or die('Error, query failed');
}
elseif($action=="rename"){
    $flag = $_GET['w4'];
    $rename=$_GET['w2'];
    $query = "SELECT AA FROM folders";
    $result = mysql_query($query) or die('Error, query failed');
    $rn=mysql_num_fields($result)+1;
    //echo $rn;
    if ($rn > $flag-1)
    {$flag=$flag-1;
    $k=0;
    while ( $row = mysql_fetch_array( $result ) )
    {if ( $k == $flag )
    $a= $row['AA'];

```

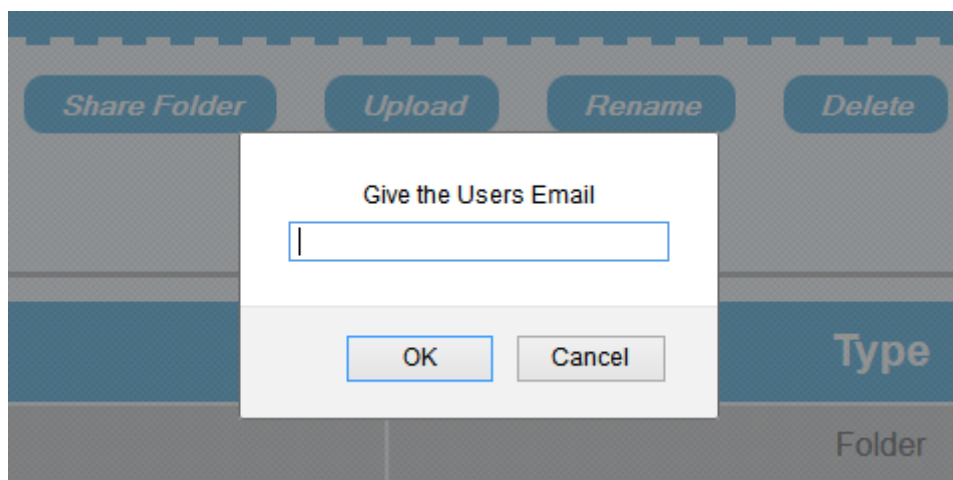
```

$k++;
}
$update="UPDATE folders SET Name = '$rename' where AA='$a'";
}
else{
$flag=$flag-$rn;
$query2 = "SELECT AA,Name FROM files Where User='$user'";
$result2 = mysql_query($query2);
$k=1;
while ( $row2 = mysql_fetch_array( $result2 ) )
{if ( $k == $flag )
{ $a= $row2['AA'];
$nm= $row2['Name'];
}
}
}
$pin=explode(".", $nm);
if(isset($pin[1]))
$rename=$rename.".".$pin[1];
$update="UPDATE files SET Name = '$rename' where AA='$a'";
}
mysql_query($update);
}
elseif($action == "delete"){
$flag = $_GET['w2'];
$query = "SELECT AA FROM folders";
$result = mysql_query($query) or die('Error, query failed');
$rn=mysql_num_fields($result);
//echo $rn;
if ($rn >= $flag)
{
$flag=$flag-1;
$k=0;
while ( $row = mysql_fetch_array( $result ) )
{
if ( $k == $flag )
$a= $row['AA'];
$k++;}
$del="DELETE FROM folders WHERE AA='$a'";
}
else
{
$flag=$flag-$rn;
$query2 = "SELECT AA FROM files Where User='$user'";
$result2 = mysql_query($query2);
$k=1;
while ( $row2 = mysql_fetch_array( $result2 ) )
{
if ( $k == $flag )
$a= $row2['AA'];
$k++;}
$del="DELETE FROM files WHERE AA='$a'";}
mysql_query($del);
}
session_start();
$_SESSION['log']=$user;
include ("main.php");
?>

```

## 4.7 Διαμοιρασμός φακέλων μεταξύ χρηστών

Η βασική λειτουργία που προσφέρει η εφαρμογή skybox είναι ο διαμοιρασμός φακέλων μεταξύ των χρηστών. Για να κάνει ο χρήστης κοινόχρηστο έναν φάκελο πρέπει πρώτα να τον επιλέξει από τον πίνακα και να πατήσει το κουμπί “share folder” το οποίο θα το εμφανίσει το παρακάτω παράθυρο διαλόγου, όπου θα πρέπει να εισάγει το e-mail του χρήστη με τον οποίο θέλει να μοιραστεί τον φάκελο.



Εικόνα 4.7 Αποστολή αιτήματος για κοινόχρηστο φάκελο

Μόλις ο χρήστης στον οποίο στάλθηκε το αίτημα για διαμοιρασμό κάποιου φακέλου κάνει είσοδο στην εφαρμογή θα πρέπει να πατήσει το κουμπί “Pendings” για να του εμφανιστούν όλα τα αιτήματα που εκκρεμούν. Αυτά τα αιτήματα μπορεί είτε να τα αποδεχτεί είτε να τα απορρήψει.



Εικόνα 4.8 Αιτήματα για διαμοιρασμό φακέλων.

Η εμφάνιση της παραπάνω εικόνας οφείλεται στο αρχείο requests.php το οποίο περιέχει τα εξής :

```
<?php
require'connectionDB.php';
if(isset($_GET['w1']))
{
$aa=$_GET['w1'];
}
else
{
```



```

echo "You are not Logged In!";
include ("LogIn.html");
exit;
}
?>
<html>
<head>
<title>SkyBox Requests</title>
<link rel='stylesheet' type='text/css' href='main.css' />
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
<meta content="charset=utf-8"/>
<script type="text/javascript" src="//code.jquery.com/jquery-2.1.0.min.js"></script>
<script>
function Accept() {
var user = "<?php echo $aa ?>";
var action = "accept";
var AA=document.getElementById('ac').name;
window.location.href = "answer.php?w1=" + user + "&w2=" + AA + "&w3=" + action;}
function Decline() {
var user = "<?php echo $aa ?>";
var action = "decline";
var AA=document.getElementById('de').name;
window.location.href = "answer.php?w1=" + user + "&w2=" + AA + "&w3=" + action;}
</script>
</head>
<body>
<div class="wrap"><pre>
<CENTER><p style="font-size:28px; font-weight:bold; margin-top: 20px;">Requests to Share
Folders</p></CENTER>
<form action="back.php" method="post"><input name="<?php echo $aa ?>" type="submit" class="back2"
value="<- Back" id="back"></form>
<table class="tbl3" id="mytable" align="center">
<tr bgcolor="#61b3de" height="50px">
<th><p style="color: white; font-size:22">AA</p></th>
<th><p style="color: white; font-size:22">Folder</p></th>
<th><p style="color: white; font-size:22">From</p></th>
<th><p style="color: white; font-size:22">Answer</p></th>
</tr>
<?php
$query = "SELECT AA, User1, User2, Folder FROM req Where User1='$aa'";
$result = mysql_query($query) or die('Error, query failed');
$num=1;
while(list($AA, $User1, $User2, $Folder) = mysql_fetch_array($result))
{
echo "<tr onClick='HighLightTR(this,'lightgrey','cc3333');\'' height='40px'">";
echo "<td><CENTER>$num</CENTER></td>";
$query2 = "SELECT AA, Name FROM folders Where AA='$Folder'";
$result2 = mysql_query($query2) or die('Error, query failed');
$row2 = mysql_fetch_array( $result2 );
$name=$row2['Name'];
echo "<td><CENTER>$fname</CENTER></td>";
$query3 = "SELECT AA, Email FROM users Where AA='$User2'";
$result3 = mysql_query($query3) or die('Error, query failed');
$row3 = mysql_fetch_array( $result3 );
$from=$row3['Email'];
echo "<td><CENTER>$from</CENTER></td>";
echo "<td><CENTER><input class='button' type='submit' value='Accept' name='$Folder' onClick='Accept()'
id='ac'> <input class='button' type='submit' value='Decline' name='$Folder' onClick='Decline()'
id='de'></CENTER></td>";
echo "</tr>";
$num++;
}
?>
</table>
</pre></div>
</body>
</html>

```

Και η υλοποίηση των παραπάνω λειτουργιών βρίσκεται στο αρχείο answer.php

```
<?php
require'connectionDB.php';
$user=$_GET['w1'];
$action = $_GET['w3'];
$fold = $_GET['w2'];
if($action == "accept"){
$query = "INSERT INTO share (user, folder ) VALUES ('$user', '$fold)";
mysql_query($query) or die('Error, query failed');
$del="DELETE FROM req WHERE User1='$user' AND Folder='$fold'";
mysql_query($del);
$query = "SELECT AA, Num FROM folders WHERE AA='$fold'";
$result = mysql_query($query) or die('Error, query failed');
$row = mysql_fetch_array( $result );
$n= $row['Num']+1;
$upd = "UPDATE folders Set Num='$n' WHERE AA='$fold'";
mysql_query($del);
}
elseif($action == "decline"){
$del="DELETE FROM req WHERE User1='$user' AND Folder='$fold'";
mysql_query($del) or die('Error, query failed');
}
session_start();
$_SESSION['log']=$user;
include ("main.php");
```

## **ΚΕΦΑΛΑΙΟ 5<sup>ο</sup>**

### **Συμπέρασμα**

Κατά την υλοποίηση των απαιτήσεων της εφαρμογής είδαμε πόσο απαιτητική είναι η δημιουργία ενός προγράμματος σε γλώσσα προγραμματισμού PHP, όλα τα απαραίτητα εργαλεία και η εγκατάσταση τους καθώς και όλες τις γνώσεις που έπρεπε να έχουμε για να αναπτύξουμε μια διαδικτυακή εφαρμογή.

Ένα από τα δυσκολότερα κομμάτια της πτυχιακής εργασίας ήταν η υλοποίηση της κοινής χρήσης αρχείων μεταξύ χρηστών της εφαρμογής λαμβάνοντας υπόψη όλες τις παραμέτρους που χρειάστηκαν για την δημιουργία και επεξεργασία των φακέλων .

Με εμπειριστατωμένα πειράματα παραδείγματα βλέπουμε ότι οι στόχοι της πτυχιακής εργασίας, που αναφέρθηκαν στην εισαγωγή του βιβλίου επιτεύχθηκαν. Παρόλα αυτά η εφαρμογή θα μπορούσε να εξελιχθεί περαιτέρω. Συγκεκριμένα :

- Να υπάρχει επεξεργασία στοιχείων προφίλ των χρηστών.
- Το επιτρεπτό μέγεθος αρχείων που μπορεί να ανεβάσει κάποιος χρήστης να αυξηθεί.
- Να δημιουργηθεί μια εφαρμογή Desktop και σε πλατφόρμες κινητών για να γίνει δυνατή η χρήση του προγράμματος χωρίς να χρειαστεί η είσοδος στην ιστοσελίδα.
- Να υπάρχει η δυνατότητα εισόδου στην εφαρμογή μέσω μιας ιστοσελίδας κοινωνικής δικτύωσης (Facebook) ώστε να μην χρειαστεί η δημιουργία νέου λογαριασμού.

## Βιβλιογραφία

- [1] <http://osarena.net/faqs/toses-cloud-ipiresies-pos-na-tis-organoso.html>
- [2] <http://www.nist.gov/index.html>
- [3] [http://en.wikipedia.org/wiki/Cloud\\_computing](http://en.wikipedia.org/wiki/Cloud_computing)
- [4] <http://el.wikipedia.org/wiki/HTML>
- [5] <http://pacific.jour.auth.gr/html/>
- [6] <http://www.web-resources.eu/archives/εκδόσεις-της-html>
- [7] <http://el.wikipedia.org/wiki/PHP>
- [8] <http://en.wikipedia.org/wiki/XAMPP>
- [9] <https://www.apachefriends.org/index.html>
- [10] <http://httpd.apache.org/>
- [11] [http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server)
- [12] <http://en.wikipedia.org/wiki/MySQL>
- [13] [http://el.wikibooks.org/wiki/Βασικές\\_γνώσεις\\_PHP\\_και\\_MySQL](http://el.wikibooks.org/wiki/Βασικές_γνώσεις_PHP_και_MySQL)
- [14] <http://en.wikipedia.org/wiki/Notepad>
- [15] <http://en.wikipedia.org/wiki/Mozilla>
- [16] <http://en.wikipedia.org/wiki/JavaScript>
- [17] [http://www.brenkoweb.com/m/www/design/design\\_jshistory.php](http://www.brenkoweb.com/m/www/design/design_jshistory.php)
- [18] [http://media.wiley.com/product\\_data/excerpt/88/07645790/0764579088.pdf](http://media.wiley.com/product_data/excerpt/88/07645790/0764579088.pdf)
- [19] <http://www.oracle.com/technetwork/java/intro-jquery-166845.html>
- [20] [http://www.w3schools.com/html/html\\_forms.asp](http://www.w3schools.com/html/html_forms.asp)
- [21] [http://en.wikipedia.org/wiki/Form\\_\(HTML\)](http://en.wikipedia.org/wiki/Form_(HTML))
- [22] <http://www.wlearn.gr/index.php/html-html-57>
- [23] [http://www.w3schools.com/tags/ref\\_httpmethods.asp](http://www.w3schools.com/tags/ref_httpmethods.asp)