**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

|  |  |
| --- | --- |
|  | **ΕΘΝΙΚΟ & ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ** |

Τεχνολογίες Εφαρμογών Διαδικτύου

Project

**Παυλος Μιλασζεγουικζ** **1115200900093**

**Απόστολος Γιούλης 1115200900215**

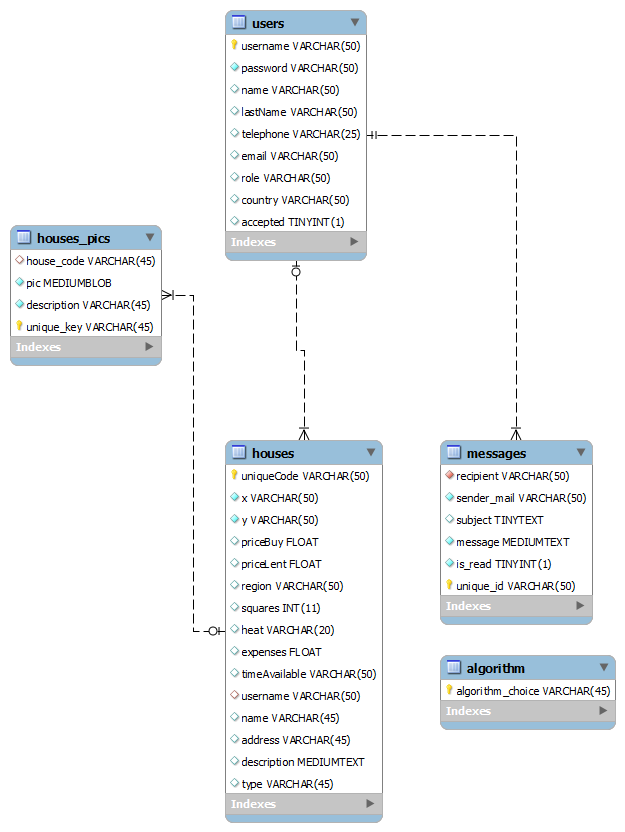
# Γενικά

* Έχουν Υλοποιηθεί όλες οι απαιτήσεις τις άσκησης
* Έγινε χρήση Bootstrap για καλύτερο οπτικό αποτέλεσμα.
* Έχει χρησιμοποιηθεί JavaScript για κάποιους ελέγχους καθώς και για τους χάρτες
* Έχει δοθεί η δυνατότητα στον χρήστη να σβήνει καταγεγραμμένα σπίτια
* Δοκιμάστηκε σε Ubuntu 14.04
* To πιστοποιητικό για το SLL πρέπει να βρίσκετε στο home directory του χρήστη. Για Windows πρέπει να γίνει σχετική αλλαγή στο server.xml
* Επειδή δεν παρατηρήθηκε καμία επιβάρυνση από την ssl όλες οι σελίδες του site την υλοποιούν.
* Όσον αφορά τα css έχουν σε μεγάλο βαθμό υλοποιηθεί σε ξεχωριστά αρχεία. Σε ορισμένες περιπτώσεις είναι Inline στην html σε απόπειρα περιορισμού css αρχείων.
* Γενικά η επικοινωνία με την βάση δες γίνεται στα Jsp παρά μόνο για selects. Η δημιουργία ιστοσελίδων δυναμικά από servlet αν και εξετάστηκε ως επιλογή δεν προτιμήθηκε.
* Στα jspf headers άρα κατ' επέκταση στα jsp και σε όλα τα servlets (τουλάχιστον σε αυτά που έβγαζε νόημα) γίνονται έλεγχοι για το session validity.
* Δεν γίνεται χρήση Hibernate. Η παρεχόμενη βιβλιογραφία αναφέρει ότι χρειάζεται για μεγάλες και πολύπλοκές βάσεις έτσι απορρίφθηκε. Επικοινωνούμε με την βάση με beans και τα database classes.
* Σε όλες της σελίδες υπάρχει στην κάτω αριστερή γωνία ένα βελάκι για την ευκολη μετάβαση στην αρχή της σελίδας.
* Στην εφαρμογή είναι προεγγατεστημένος ο χρήστης admin κάνοντας χρήση του παρεχόμενου Create Users.sql στο0 “/Sql create scripts/”

# Παραδοχές

* Δεν δίνεται η δυνατότητα στον χρήστη να αλλάξει δεδομένα του σπιτιού που αφορούν την τοποθεσία του. Θεωρήθηκε υπερβολική πληροφορία προς τον χρήστη. Για τον λόγο αυτό υπάρχει προειδοποιητικό μήνυμα κατά την εγγραφή σπιτιού στο σύστημα. Ο λόγος είναι ότι το σπίτι δεν γίνεται να αλλάξει τοποθεσία παρά μόνο άλλα γνωρίσματα. Σε περίπτωση λάθους ο χρήστης έχει την δυνατότητα σβήσει το σπίτι και να το επαναεκχωρήσει.
* Η αποστολή email δεν υλοποιήθηκε (η εκφώνηση έλεγε εκτός συστήματος) για την διευκόλυνση του χρήστη όμως δίνεται ένα κουμπί που ανοίγει την εφαρμογή ηλεκτρονικού ταχυδρομείου του χρήστη (π.χ. thunderbird) συμπληρώνοντας αυτόματα το πεδίο του παραλήπτη και του Subject.
* Δεν δίνεται στον χρήστη η δυνατότητα να αλλάξει τον αλγόριθμο. Μια τέτοια δυνατότητα θα ήταν υπερβολική για τον μέσο χρήστη. Μόνο χρήστης με admin Rights μπορεί να κάνει κάτι τέτοιο.
* Στην σελίδα παρουσίασης των αποτελεσμάτων μια αναζήτησης σπιτιών δεν έχει μπει λίστα με τα σπίτι παρά μόνο ο χάρτης με τα markers. Ο λόγος πίσω από αυτό είναι ότι σχεδιαστικά δεν βρήκαμε λύση ώστε και ο χάρτης να είναι αρκετά μεγάλος και να υπάρχει λίστα με τα σπίτια. Τέτοια λίστα έχει υλοποιηθεί στο UserManageHousePage.
* O χρήστης που έχει μόνο Visitor rights μπορεί μόνο να αλλάξει το προφίλ του και να ζητήσει περισσότερα rights
* Στα jsp το navigation bar είναι δυναμικό. Δηλαδή ανάλογα με τα δικαιώματα του εκάστοτε χρήστη εμφανίζονται τα απαιτούμενα κουμπιά μετάβασης στην σχετικη σελίδα

# Βάση Δεδομένων

* Όλα τα unique (unique\_key,unique\_id,uniqueCode) παράγονται από την συνάρτηση UUID.randomUUID() της Java που έγκειται μοναδικούς κωδικούς.

# Mini Guide

# Jsp

### General (Καθολική χρήση)

* Response.jsp Εμφανίζει διαγνωστικά μηνύματα και ανακατευθύνει τον χρήστη ανάλογα με τα στοιχεία που παρέχει ο κάθε servlet.

### Pre-Loggin (Στον φάκελο Web Pages καθώς ο οποιοσδήποτε μπορεί να έχει πρόσβαση σε αυτές)

* + - * index.jsp Αρχική σελίδα ο χρήστης μπορεί να κάνει Login ή signup. Τυχόν διαγνωστικά μνήματα εμφανίζονται και τον ανακατευθύνουν πάλι εκεί. Γίνεται έλεγχός και μέσω HTML και Server Side (ServletLogin) για την ορθή συμπλήρωση του πεδίου.
      * About.jsp Γενικές Πληροφορίες για την εταιρία
      * Contact.jsp Contact Page για τυχόν απορίες. Είναι εικονικό δεν έχει υλοποιηθεί.
      * Signup.jsp Ο χρήστης μπορεί να δημιουργήσεις account. Υπάρχουν έλεγχοι HTML5 (required inputs) καθώς και ένας έλεγχος javascript για το αν οι κωδικοί είναι ίδιοι. Οι έλεγχοι θα επαναληφθούν και Server Side (UserInserServlet)

### After Loggin (Στο WEB-INF) μπορεί κάποιος να είχε πρόσβαση σε αυτές μόνο μέσω Servlets. Ακόμα και αν γραφεί το full path στο url bar δεν θα είναι δυνατή η πρόσβαση.

### Admin

* **AdminHomePage.jsp** Κεντρική σελίδα του χρήστη με δικαιώματα admin. Δίνονται κάποια στατιστικά στοιχεία. Τα queries στην βάση γίνονται επί τις jsp και όχι μέσω servlet μιας και γίνονται μόνο selects. Αν ο χρήστης έχει και δικαιώματα απλού χρήστη μπορεί να συνεχίσει την περιήγηση του ως κανονικός χρήστης.
* **AdminAlgorithmChange.jsp** Δυνατότητα στον admin να αλλάξει το αλγόριθμο αναζήτησης μεταξύ του SAW και του TOPSIS χρήση (AdminSetAlgorithm)
* AdminPending Στην σελίδα αυτή ο admin μπορεί να εγκρίνει χρήστες ή να τους απορρίψει (χρήση AdminServletPending). Όλα τα στοιχεία του χρήστη εμφανίζονται στην λίστα.
* AdminUsrOverview O Admin μπορεί να διαγράψει χρήστες από το σύστημα.

### User

* **UserHomePage.jsp** Κεντρική σελίδα του χρήστη τον πληροφορεί για το πόσα μνήματα έχει να διαβάσει και πόσα σπίτια έχει εγγεγραμμένα στο σύστημα
* **UserUpdateProfile.jsp** Δίνεται η δυνατότητα αλλαγής στοιχείων του προφίλ του. Το username δεν αλλάζει.
* **UserSearchHousesResults.jsp** Παρουσιάζονται τα αποτελέσματα μιας αναζήτησης και τοποθετούνται ως στίγματα στον χάρτη.
* **UserManageHousePage. Jsp** Παρουσιάζονται σε έναν χρήστη τα σπίτια που έχει εγγεγραμμένα σε λίστα και σε χάρτη. Δίνεται η δυνατότατα είτε επιλέγοντας το στίγμα στον χάρτη είτε μέσω του σχετικού κομβιου στην λίστα να μεταβεί στην σελίδα του για περαιτέρω αλλαγές (HouseInfo.jsp)
* **UserInboxPage.jsp** Παρουσιάζονται τα μηνύματα στον χρήστη. με γκρι φόντο τα μη αναγνωσμένα με άσπρο τα αναγνωσμένα. Δίνεται η δυνατότατα να τα διαγράψει ή να τα διαβάσει και να απαντήσει (UserMessageReadReply.jsp)
* **UserMessageReadReply.jsp** Δίνεται η δυνατότατα να διαβάσει και να απαντήσει μέσω της τοπικής εφαρμογής ηλεκτρονικού ταχυδρομείου, το επιλεγμένο μήνυμα.
* **UserEnlistHouse.jsp** Σελίδα εγγραφής σπιτιού στο σύστημα. Οι φωτογραφίες μπορούν να προστεθούν μετά O χρήστης εισάγει τις συντεταγμένες ή την διεύθυνση στο σχετικό πεδίο στον χάρτη και συμπληρώνονται αυτόματα στον χάρτη ορισμένα πεδία. Το google maps API παρουσιάζει ασυνέχεια στην μορφή της διεύθυνσης που επιστρέφεται (geolocation) έτσι δίνεται η δυνατότατα στον χρήστη να αλλάξει τα πεδία αυτά.
* **UserContact.jsp** Μέσω του HousePresentation.jsp δίνεται η δυνατότητά στον αγοραστή/ ένοικο να επικοινωνήσει με μήνυμα στον πωλητή/ ενοικιαστή. To subject, ο προορισμός και ο αποστολέας είναι προκαθορισμένα.
* **UserBuyPage.jsp** Για αναζήτηση σπιτιών. Γίνονται πολλοί έλεγχοι όσον αφορά την εγκυρότητα των τιμών και εμφανίζονται σχετικά διαγνωστικά μηνύματα.
* **HousePresentation.jsp** Σελίδα παρουσίασης σπιτιού, εμφανίζονται οι πληροφορίες του σπιτιού καθώς και ένα slider (δεν έχει υλοποιηθεί από εμάς) με φωτογραφίες και την περιγραφή τους.
* **HouseInfo.jsp** Σελίδα που καλείτε από την UserManageHousePage. Jsp και δίνει την δυνατότατα στον χρήστη να αλλάξει στοιχεία του σπιτιού, να ανεβάσει φωτογραφίες και καθώς και να τις σβήσει.

# Source packages (Επεξήγηση με comments όπου χρειάζεται.)

### db (κλάσεις που υλοποιούν την επικοινωνία με την βάση)

### entities (κλάσεις που υλοποιούν τα beans)

### web (Servlet κλάσεις)