

DOCUMENTATION

Στο 1^ο ατομικό project υλοποίησα μία δυναμική σελίδα χρησιμοποιώντας Java, Html, Css, MySql. Σκοπός της είναι τεστάρει τον πίνακα pro_user στην βάση δεδομένων bids4job που θα χρησιμοποιηθεί στην ομαδική εργασία. Στην σελίδα ο χρήστης έχει την δυνατότητα να τεστάρει 6 queries για τον συγκεκριμένο πίνακα (create(), update(), remove(), findOne(), findFromLocation(), findAll()). Ο χρήστης έχει άμεση πρόσβαση, μέσω της σελίδας, με την βάση δεδομένων. Παρακάτω υπάρχει αναλυτικά η λειτουργία της σελίδας.

DataBase (bids4job)

Η βάση δεδομένων bids4job υλοποιήθηκε στο MySQL Workbench και αποτελείτε από 5 πίνακες.

pro_user (pro_user_ID(primary key) - first_name - last_name - location - profession)

Πίνακας που χαρακτηρίζει τους επαγγελματίες χρήστες της εφαρμογής.

simple_user (simple_user_ID(primary key) - first_name - last_name - location)

Πίνακας που χαρακτηρίζει τους χρήστες της εφαρμογής.

bid (bid_ID(primary key) - task_ID(foreign key) - pro_user_ID(foreign key) - amount - bid_time)

Πίνακας που χαρακτηρίζει τις προσφορές των επαγγελματιών χρηστών σε σχέση με ένα συγκεκριμένο task κάποιου χρήστη της εφαρμογής.

task (task_ID(primary key) - deadline - work_field - simple_user_ID(foreign key))

Πίνακας που χαρακτηρίζει τα tasks των χρηστών στην εφαρμογή.

contract (contract_ID(primary key) - bid_ID(foreign key) - task_ID(foreign key) - contact_time)

Πίνακας που χαρακτηρίζει τα συμβόλαια που κλείνονται, αφού ο χρήστης δεχτεί κάποιο bid στο task του χρήστη.

BackEnd

Το λειτουργικό σύστημα της εφαρμογής είναι γραμμένο σε java και υλοποιήθηκε με Eclipse

Παρακάτω φαίνονται αναλυτικά τα πακέτα και οι κλάσεις του συστήματος.

src → domain → ProfessionalUser.java
 → dao → ProfessionalUserDao.java
 → DaoUtils.java
 → service → ProfessionalUserService.java
 → servlets → CreateController.java
 → UpdateController.java
 → RemoveController.java
 → findOneController.java
 → findFromLocationController.java
 → findAllController.java

Στο πακέτο domain υπάρχει η κλάση **ProfessionalUser.java**, η οποία είναι ο σκελετός του αντικειμένου ProfessionalUser. Υλοποιεί τους constructors / getters / setters και 3 πολύ βασικές μεθόδους.

Στο πακέτο dao υπάρχει η κλάση **DaoUtils.java** η οποία είναι υπεύθυνη για την σύνδεση του λειτουργικού μας συστήματος με την βάση δεδομένων. Επίσης υπάρχει η κλάση **ProfessionalUserDao.java** η οποία υλοποιεί κυρίως 6 μεθόδους που υλοποιούν τα 6 αντίστοιχα queries (create - update - remove - findAll - findFromLocation - findOne)

Στο πακέτο service υπάρχει η κλάση **ProfessionalUserService.java** η οποία καλεί όλες τις μεθόδους της κλάσης ProfessionalUserDao.

Στο πακέτο servlets υπάρχουν οι κλάσεις **CreateController.java**, **UpdateController.java**, **RemoteController.java**, **findOneController.java**, **FindFromLocationController.java**, **FindAllController.java** οι οποίες είναι servlets που δέχονται POST και GET actions από το index.jsp και αναλόγως το action και το Submit value απαντάνε στα ...Results.jsp αρχεία.

FrontEnd

Το FrontEnd κομμάτι είναι γραμμένο σε .jsp σελίδες και στατικές .html σελίδες.

Στον φάκελο WebContent υπάρχει ο φάκελος css που περιέχει τα .css αρχεία, η βιβλιοθήκη της MySQL – Java και όλα τα .jsp / .html αρχεία.

header.html : Στατικό html αρχείο που είναι υπεύθυνο για τον σκελετό του index.jsp

headerResults.html : Στατικό html αρχείο που είναι υπεύθυνο για τους σκελετούς όλων των .jsp αρχείων για τα αποτελέσματα.

footer.html : Στατικό html αρχείο που είναι υπεύθυνο για τον σκελετό όλων των .jsp αρχείων.

index.jsp : Είναι υπεύθυνο για τις 6 φόρμες της αρχικής σελίδας. Κάθε φόρμα έχει δικό της submit value, έτσι το servlet controller.java ελέγχει ποια λειτουργία πρέπει να κάνει.

createResults.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία create().

updateResults.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία update().

removeResults.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία remove().

findOneResults.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία findOne().

findFromLocationResults.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία findFromLocation().

findAll.jsp : Είναι υπεύθυνο για τα αποτελέσματα που θα δείξει στον χρήστη αν επιλέξει την λειτουργία FindAll().

errorPage.jsp : Είναι υπεύθυνο να τυπώνει το σωστό μήνυμα λάθους, όποτε αυτό χρειάζεται.

Σημείωση: Κάποιοι Professional Users δεν μπορούν να διαγραφούν(Professional User ID: 1-4,6,10-18), γιατί υπάρχουν σαν foreign keys σε κάποια bids, και δεν έχουμε αποφασίσει ακόμα πως θα το χειριστούμε στην ομαδική εργασία.