Τεχνολογίες Εφαρμογών Διαδικτύου(ΤΕΔ)

Πλεύρης Κωσταντίνος 1115201400162

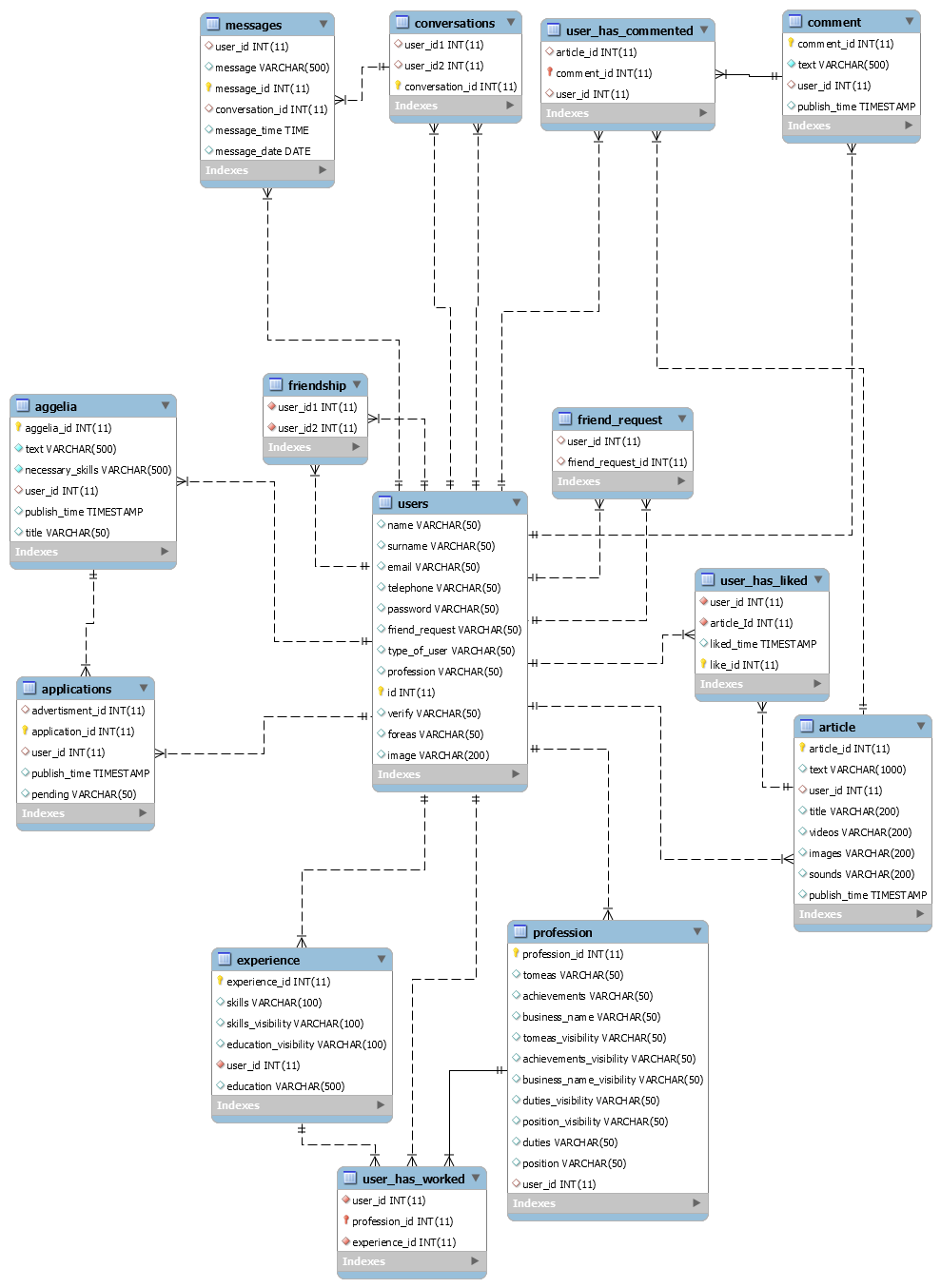
Βέργος Ηλίας 1115201400266

**Δομή άσκησης από το Front στο Back End**

To front κομμάτι αποτελείται από τα αρχεία .jsp που κρατούν το html κομμάτι,τα javascipt αρχεία καθώς επίσης και τα .css αρχεία που έχουν τις κλάσεις που χρησιμοποιούνται στο html.Oσο αναφορα το front κομμάτι χρησιμοποιήθηκε html,css,AngularJs,JSON,AJAX και κανονική javascript.Το css φτιάχτηκε εξ’ολοκλήρου από εμάς (δεν πήραμε τιποτα ετοιμο από καποιο framework όπως bootstrap ,κ.α.).

Το back κομμάτι αποτελείται από τα Servlets,την Βαση,τo JPA και τα Dao.

**Το σχημα της βασης είναι:**

****

Η βαση αποτελειται από 9 οντοτητες (users,profession,experience,messages,aggelia,applications,article ,conversation,comment).

1)**users**: Είναι το table με τα βασικα στοιχεια των χρηστων.

2)**profession** :Είναι το table που περιεχει τις πληροφοριες του χρηστη αναφορικα με τις παλιες δουλειες που εχει κανει στο παρελθον.Επισης υπαρχουν και τα πεδια για public και private για κάθε πεδιο.

3)**experience**: Είναι το table με την εκπαιδευση και τις ικανοτητες του χρηστη καθως και τα visibility πεδια.

4)**messages**: Είναι το table με τα μυνηματα που εχει κανει ο κάθε χρηστης.

5)**aggelia**:Είναι το table με τις αγγελιες που εχει δημοσιευσει ο κάθε χρηστης.

6)**applications**:Είναι το table με τις αιτησεις που εχει κανει κάθε χρηστης.Τα applications αυτά ,εχουν από default ,pending =True.Aυτο σημαινει ότι όταν ενας χρηστης κανει μια αιτηση για μια αγγελια και περιμενει για απαντηση τοτε εχουμε pending =true.Μολις ο δημιουργος της αγγελιας απαντησει θετικα ή αρνητικα τοτε το pending γινεται false και χρησιμοποειται μονο για τα στατιστικα που χρειαζομαστε για τον αλγοριθμο εμφανισης αγγελιων.Ετσι τα applications δεν σβηνονται,απλα αλλαζουν από pending =true σε pending =false εκτος και αν ο χρηστης που εκανε την αιτηση πατησει cancel apply ή αν ο διαχειριστης τον διαγραψει.

7)**article**:Είναι το table με τις δημοσιευσεις αρθρων.

8)**conversation**:Είναι το table που περιλαμβανει τις συνομιλιες των χρηστων.

9)**comment**:Είναι το table με τα σχολια των χρηστων σε αρθρα .

10)**user\_has\_liked:** Περιλαμβανει του χρηστες που εχουν like σε καποιο αρθρο καθως και την ωρα που εγινε το like.

Επισης η βαση για την σωστη χρηση της περιλαμβανει και 5 συσχετισεις(friendship,friend\_request,user\_has worked,user\_has\_commented).

**1)friendship:** Περιλαμβανει τους χρηστες και τους φιλους τους και το αντοθετο.Δηλαδη για κάθε χρηστη , η φιλια είναι αμφιδρομη πχ.user1 ->user2,user2->user1 🡪user1 is friend with user2.

**2)friend\_request:**Περιλαμβανει τους χρηστες που εκαναν αιτηση φιλιας και αυτους στους οποιους εκαναν την αιτηση φιλιας.

**3)user\_has\_worked:**Η συσχετιση αυτή περιλαμβανει τον χρηστη και ότι εχει να κανει με την επαγγελματικη του καριερα .Δηλαδη τον συνδεει με τo table της εκπαιδευσης και των ικανοτητων και με το table με τις προηγουμενες του δουλειες.

**4)user\_has\_commented:**Περιλαμβανει τα σχολια του χρηστη σε αρθρα.Συνδεει τα σχολια με τα αρθρα και τους χρηστες.

**Παρατηρησεις για τις τεχνολογιες που χρησιμοποιηθηκαν:**

1)Για την μεταφορα δεδομενων από τα Servlet στα jsp χρησιμοποιηθηκε η τεχνολογια JSON.Για την υλοποιηση του json χρησημοποιθηκαν καποιες ετοιμες βιβλιοθηκες που κατεβασαμε από το διαδικτυο και βρισκονται στον καταλογο WebContent/WEB-INF/lib/

2)Ο τροπος με τον οποιο αλληλεπιδρουν servlet και jsp είναι ο εξης:Αρχικα καλουμε στο url το .jsp,υστερα στο body του jsp υπαρχει μια συναρτηση που καλειται μεσω angular ng-init=loadPage() ,η οποια στελνει αιτημα get ασυγχρονο με ΑJAX στο servlet για να παρει τα απαραιτητα δεδομενα που χρειαζεται η σελιδα για να φορτωθει σωστα.Στην συνεχεια αναλογα με την σελιδα μπορει να σταλθουν και αλλα ασυγχρονα αιτηματα get ή post με AJAX ή με την κλασσικη μορφη της html φορμας.Για την σωστη αποστολη και ληψη στοιχειων από το jsp στο servlet χρησιμοποιουμε καταλληλες συνθηκες if που μας επιτετρεπουν να βλεπουμε ποιες από τα δεδομενα είναι κενα και ποια μεταφερουν υλικο.(Μπορει στο servlet να εχουμε συνθηκες από 10 διαφορετικα αιτηματα,γιαυτο χρησιμοπουμε if για να δουμε ποια είναι η σωστη αιτηση που εγινε στον server. )

3)Για τα παραγωγη του Xml αρχειου επρεπε με καποιον τροπο στα entities να βαλουμε ποιες από τις μεταβλητες δεν είναι απαραιτητες .Αυτο εγινε χρησιμοποιωντας το annotiation @XmlTransient.Για τον ιδιο λογο ,όμως τωρα για το JSON ,για να μην επιτρεπουμε την απειρη επαναληψη μεταβλητων χρησιμοποιησαμε τον annotiation @JsonIgnore.

4)Για τις σχεσεις μεταξυ των στοιχειων των entities χρησιμοποιηθηκαν τα annotiations (@OneToOne,@ManyToOne,@OneToMany,@ManyToMany)

5)Για την επικοινωνια servlet και βασης χρησιμοπουμε τα Dao και τον Εntity manager που μας επιτρεπει μεσω συναρτησεων να κανουμε τα καταλληλα sql ερωτηματα στην βαση και να παιρνουμε τα αποτελεσματα που θελουμε.

**Γενικες Παρατηρησεις:**

1)Για την συνδεση ενός χρηστη στον λογαριασμο του χρησιμοποιουνται cookies .

2)Το footer υπαρχει στις βασικες καρτελες που αναφερονται στην ασκηση.Περιλαμβανει το ονομα του site (Business World),τους σχεδιαστες της ασκησης (εμας τους 2 ) και 2 ασχετα πεδια που δεν κανουν τιποτα κλικαροντας τα ,απλα υπαρχουν για να γεμισει το footer.

3)Για την εισαγωγη του κωδικου στην βαση και την ανακτηση του χρησιμοποιουμε μια hash function που βρηκαμε ετοιμη στο internet (Hash Function MD5).Αυτο γινεται γιατι θεωρουμε πως ο κωδικος είναι ένα αππορητο στοιχειο που ουτε οσοι εχουν προσβαση στην βαση πρεπει να ξερουν.

4)Εχουμε φτιαξει καποιες βοηθητικες συναρτησεις όπως το ανεβασμα αρχειου στον υπολογιστη,την αποστολη email επιβεβαιωσης,την δημιουργια αντικειμενου json και καποιες άλλες βοηθητικες κλασεις για το xml ,και τους αλγοριθμους για τα recommended αρθρα και αγγελιες.

5)Τα στοιχεια των χρηστων από τα αρθρα αλλα και η εικονα προφιλ τους αποθηκευονται όλα σε φακελους μεσα στο workspace της ασκησης.Πιο συγκεκριμενα μεσα στο WebContent/resources/Database/.Eκει υπαρχει ένα φακελος με το id κάθε χρηστη που μπορει να περιλαμβανει 2 φακελους max.O 1ος φακελος είναι το images που περιεχει την εικονα προφιλ του και ο 2ος φακελος περιεχει τα id από τα αρθρα που πιθανον να εχει δημοσιευσει.Μεσα στον φακελο με τα id των αρθρων υπαρχουν 3 υποφακελοι max.Aυτοι μπορει να είναι τα videos ,photos,audios που μπορει να περιεχει κάθε αρθρο που δημοσιευει ο χρηστης.Τελος για να παρουμε τα στοιχεια αυτά από την βαση και να τα χρησιμοποιησουμε στο site υπαρχουν σχετικα πεδια στην βαση με τα paths των φακελων αυτων.

**Ερώτημα 1 (Υλοποιήθηκε):**

Για το 1ο ερώτημα φτιάξαμε μία σελίδα καλωσορίσματος όπου χρήστης μπορεί να πάρει κάποιες πληροφορίες για το site ,μπορεί επισης να κανει εγγραφη πατώντας το αντίστοιχο κουμπί και τελος μπορεί να κανει σύνδεση εφοσον πρωτα εχει φτιαξει ένα λογαριασμο και τον εχει επιβεβαιωσει από το email του.

**Ερώτημα 2 (Υλοποιήθηκε):**

Εχουμε υλοποιησει την σελιδα εγγραφης όπως μας ζητειται στην ασκηση και εχουμε προσθεσει 2 επιπλεον πεδια(Επαγγελμα και φορεας απασχολησης) τα οποια όμως σε αντιθεση με τα αλλα που ζητα η ασκηση δεν είναι υποχρεωτικα για τον χρηστη.

**Ερώτημα 3 (Υλοποιήθηκε)** **:**

Στην βαση υπαρχει ηδη ενας χρηστης με δικαιωματα διαχειριστη.Τα στοιχεια του για τυχον δοκιμες είναι τα παρακατω

username/email:admin@gmail.com

password: 1234

**Eρώτημα 4 (Υλοποιήθηκε):**

Στην σελίδα διαχείρισης ,υπάρχει μια λιστα με ολους τους χρηστες που εχουν κανει εγγραφη στο site.Για κάθε χρήστη ο διαχειριστης εχει την δυνατοτητα να βλεπει το profile του και τους φιλους αυτου του χρηστη.Επισης εχει την δυνατοτητα να διαγραψει έναν οποιονδηποτε χρήστη ή να εξαγει σε αρχειο XML τα δεδομενα ενός ή και παραπάνω χρηστών.Ο διαχειριστης δεν μπορει να στείλει μυνήματα στους χρηστες.Για την δημιουργια του XML αρχειου χρησιμοποιουμε μια κλαση(XmlClass.java) που περιεχει τα εξης στοιχεια του χρηστη (ονομα,επιθετο,id,email,τηλεφωνο,δουλεια,φορεας απασχολησης,τα αρθρα που εχει κανει like,την εμπειρεια του ,τα σχολια που εχει κανει σε αλλα αρθρα,τις αγγελιες και τα αρθρα που εχει δημοσιευσει),και αυτά είναι που εμφανιζονται στο xml αρχειο.Για την δημιουργια του XML χρησιμοπουμε τις εξης βιβλιοθηκες της java(javax.xml.bind.JAXBContext, javax.xml.bind.JAXBException,και javax.xml.bind.Marshaller) και ως κλαση για τον marshaller εχουμε φτιαξει μια κλαση(Users.java) που περιεχει μια λιστα από τις παραπανω κλασεις(ΧmlClass.java).To αρχειο αποθηκευεται μεσα σε καποιο φακελο στον υπολογιστη μας που του εχουμε ορισει εμεις.

**Ερώτημα 5 (Υλοποιηθηκε):**

Στην αρχικη σελιδα ,όπως και σε οποιαδηποτε σελιδα και να πλοηγηθει ο χρηστης μεσα στον λογαρισμο του πανω δεξια θα βρει το κουμπι και εξοδο από τον λογαριασμο του.Στο αριστερο μερος της αρχικης σελιδας εχουμε τους φιλους μας από πού μπορουμε με ένα κουμπι να στειλουμε μυνημα σε αυτους και με ένα άλλο να δουμε το προφιλ τους.Επισης υπαρχει πεδιο που ο χρηστης μπορει να συμπληρωσει καποια από τα στοιχεια του (Δεξιοτητες και Εκπαιδευση) ή αν θελει να υπαρχει κουμπι που τον στελνει στην σελιδα με τα προσωπικα του στοιχεια για να τα συμπληρωσει εκει.Στο κεντρικο μερος υπαρχει κουμπι οπου ο χρηστης πατα και μπορει να δημιουργησει καινουργιο αρθρο όπως η ασκηση ζητα(με εικονα,βιντεο ή ηχο).Επισης στο κεντρικο μερος εχει την δυνατοτητα να βλεπει τα αρθρα των φιλων του ή ότι αλγοριθμος του ερωτηματος 12 του προτεινει.Σε κάθε αρθρο που βλεπει ο χρηστης μπορει να κανει like ,comment να δει ποιος το δημοσιευσε και τι ωρα εγινε η δημοσιευση.Για τα βιντεο και ηχο χρησιμοποιηθηκαν τα html5 στοιχεια video και sound.

**Ερώτημα 6 (Υλοποιηθηκε):**

Στην σελιδα του Δικτυου του χρηστη ο χρηστης μπορει να δει σε μορφη πλεγματος τους φιλους του.Πατωντας να δει το προφιλ τους μπορει να τους στειλει μυνημα και να ξεκινησουν νεα συνομιλια και ακομα είναι σε θεση να δει και τους φιλους του φιλου του. Επισης μπορει να κανει αναζητηση στην μπαρα αναζητησης για την ευρεση νεων μελων.Η αναζητηση γινεται με βαση το ονομα ,το επιθετο ή και τα 2 μαζι.Για να γινει σωστη αναζητηση θα πρεπει τα στοιχεια που πληκτρολογουνται στην μπαρα αναζητησης να είναι ιδια με αυτά της βασης.Απο την λιστα με τα αποτελεσματα ο χρηστης μπορει να δει τα προφιλ των χρηστων ,χωρις όμως να μπορει να δει τους φιλους του χρηστη αυτου.Αν ο χρηστης αυτος που δεν είναι φιλος μας εχει επιλεξει να εχει όλα τα στοιχεια του ιδιωτικα τοτε δεν μπορουμε να δουμε τιποτα άλλο εκτος από το ονομα ,το επιθετο του και την εικονα του.

**Ερώτημα 7 (Υλοποιηθηκε):**

Στην καρτέλα με τις αγγελίες ,έχουμε το κεντρικο τμημα που περιεχει τις αγγελιες που ο χρηστης μπορει να εξετασει και να κανει αιτηση για την θεση και μπορει ακομα να φτιαξει μια δικια του αγγελια.Επισης υπαρχει και το αριστερο τμήμα της σελίδας οπου ο χρηστης βλεπει τις αιτησεις αλλων χρηστων για τις αγγελιες που ο ιδιος εχει αναρτησει.Εκει μπορει να δει το προφιλ του κάθε χρηστη πατωντας πανω στο ονομα του και υστερα μπορεί να επιλέξει αν θα δεχτει ή θα αρνηθει την αιτηση για την αγγελια που δημοσιευσε.Πατώντας αποδοχη ή απορριψη δεν γινεται τιποτα στο προγραμμα από αποψη λειτουργικοτητας ,το μονο που γινεται είναι μια αλλαγη στην βαση ότι ο χρηστης απαντησε στην αιτηση.Δεν βγαινει καποια ανακοινωση στον χρηστη ότι η αιτηση που εκανε αποριφθηκε ή ότι την αποδεχτηκαν.Οσο αναφορα την εμφανιση των αγγελιων,γινεται με τον εξης τροπο:Στο πρωτο κομματι εμφανιζουμε ολες τις αγγελιες των φιλων μας που μπορουμε να κανουμε αιτηση ή εχουμε κανει αιτηση και περιμενουμε απαντηση.Αν εχει δοθει απαντηση για μια αιτηση η αγγελια παυει να υπαρχει στο κεντρικο τμημα των αγγελιων.Το δευτερο κομματι βασιζεται κυριως στον αλγοριθμο που μας ζητειται.Για την υλοποιηση του αλγοριθμου παιρνουμε ολους τους χρηστες που δεν είναι φιλοι μας και ολες τις αγγελιες που εχουν δημοσιευτει.Αν ο συνδεδεμενος χρηστης εχει κανει αιτηση σε καποια από αυτές τις αγγελιες τοτε δινουμε σκορ σε αυτην την αγγελια ισο με 1,ολες οι υπολοιπες παιρνουν 0.Υστερα παιρνουμε ολες τις αγγελιες και συγκρινουμε τα skills κάθε αγγελιας με τα δικα μας skills.Aν υπαρχει κοινο skill τοτε στο σκορ της αγγελιας προσθετουμε το 2.Ετσι στο τελος, κάθε αγγελια που υπαρχει στην βαση εχει ένα σκορ.Εμεις παιρνουμε τις K=5(τυχαιο νουμερο) αγγελίες με το μεγαλυτερο σκορ.Αυτες οι αγγελιες είναι για εμας οι similar αγγελιες και με βαση αυτές θα παρουμε τις αγγελιες των χρηστων που δεν είναι φιλοι μας και οποια αγγελια από τις αγγελιες των μη φιλων μας εχει καποιο κοινο required skill τοτε την προτεινουμε στον συνδεδεμενο χρηστη.Επομενως οι αγγελιες που εμφανιζονται στον χρηστη είναι οι προτεινομενες (που προερχονται από τους χρηστες που δεν είναι φιλοι μας και λειτουργουν βαση αλγοριθμου) και οι αγγελιες των φιλων μας.Tα skills που βαζει ο χρηστης πρεπει να χωριζονται με κομμα(,).

**Ερωτημα 8 (Υλοποιηθηκε):**

Για την καρτελα με τις συζητησεις δεν υπαρχει κατι το διαφορετικο από αυτό που ζηταει η ασκηση.

**Ερωτημα 9 (Υλοποιηθηκε):**

Στις καρτελα με τις ειδοποιησεις όπως ζητα η ασκηση ,στο πανω μερος εχουμε τις αιτησεις φιλιας αν υπαρχουν οπου ο χρηστης μπορει να δει το προφιλ του χρηστη που του εκανε αιτηση πατωντας στο ονομα του και υστερα να δεχτει ή να αρνηθει την αιτηση φιλιας.Οσο αναφορα το κατω μερος ,εχουμε τις ειδοποιησεις για τα αρθρα που εχουμε δημοσιευσει(πρωτα οι ειδοποιησεις για τα likes και μετα οι ειδοπιησεις για τα comments).Πατώντας στην ειδοποιηση για το like που σου εγινε σε πεταει σε μια σελιδα με το αρθρο σου και όλα τα στοιχεια του(τιτλος,κειμενο,συννημενα,likes,comments).To ιδιο γινεται και στην περιπτωση των ειδοποιησεων των σχολιων μονο που εκει μπορεις πατωντας στην ειδοποιηση να δεις και ποιο από τα σχολια που φαινονται στο αρθρο είναι το σχολιο που κλικαρες.Τελος και για σχολια και για τα likes φαινεται με πιο εντονο χρωμα η τελευταια ειδοποιηση.

**Ερωτημα 10 (Υλοποιηθηκε):**

Στην καρτελα με τα προσωπικα στοιχεια από default(χωρις δλδ να συμπληρωσει κατι ο χρηστης ) υπαρχει το ονομα και το επιθετο του χρηστη ,η εικονα που εβαλε στην εγγραφη και τα κουμπια για να συμπληρωσει ότι πληροφορια χρειαστει.Υπαρχουν 3 κουμπια ,ένα για να συμπληρωσει τα skills του, ένα για να συμπληρωσει την εκπαιδευση του και ένα για να προσθετει τυχον παλιες του επαγγελματικες θεσεις(δουλεια,τομεας απασχολησης,ονομα επιχειρησης ,καθηκοντα,επιτευγματα).Το προηγουμενο κουμπι το ονομασαμε ‘προσθεσε επαγγελματικη εμπειρια’ και θεωρειται ότι προσθετωντας προηγουμενες δουλειες προσθετεις εμπειρια.Αν εχουν συμπληρωθει εκπαιδευση και ικανοτητες μπορεις να τα κανεις edit ανα πασα στιγμη από αυτην την σελιδα όπως επισης και τις προηγουμενες σου δουλειες.Τελος για όλα αυτά τα πεδια που μπορεις να συμπληρωσεις εχεις την δυνατοτητα να επιλεξεις αν θα είναι δημοσια (τα βλεπουν ολοι) ή ιδιωτικα(τα βλεπουν μονο οι φιλοι).Σημειωση:Οσο αναφορα τα προηγουμενα επαγγελματα ,αν ενας χρηστης εχει δηλωσει ως ιδιωτικο το ονομα της δουλειας και όλα τα υπολλειπα δημοσια τοτε δεν εμφανιζεται καν σαν στοιχειο στο dropdown menu με τις δουλειες.

**Ερωτημα 11 (Υλοποιηθηκε):**

Στις ρυθμισεις ο χρηστης μπορει να αλλαξει κωδικο και email οποτε θελει.

**Ερωτημα 12 (Βοnus)(Υλοποιηθηκε):**

Ο αλγοριθμος του ερωτηματος αυτου αρχικα παιρνει ολους τους χρηστες και όλα τα αρθρα που υπαρχουν στην βαση.Στην συνεχεια για κάθε χρηστη βλεπουμε που εχει κανει like ,comment και αναλογως προσθετουμε στο αντιστοιχο αρθρο έναν αριθμο.Οι αριθμοι αυτοι είναι οι εξης : 0 αν δεν εχει κανει ουτε like ουτε comment , 1 αν εχει κανει comment και όχι like ,2 αν εχει κανει like και όχι comment και 3 αν τα εχει κανει και τα δυο.Αν ο χρηστης δεν μπορει να δει καποια από τα αρθρα επειδη ο χρηστης που το δημοσιευσε δεν είναι φιλος του ειτε γιατι δεν του εχει προταθει ακομα ,τοτε το αντιστοιχο αρθρο παιρνει τιμε 10.Οσο μεγαλυτερη τιμη τοσο πιο μεγαλη θα βγει η ευκλειδια αποσταση και ετσι δεν θα επιλεγει,γιαυτο βαλαμε 10 και όχι 4.Υστερα αφου εχουμε βαθμολογησει κάθε αρθρο με βαση τον κάθε χρηστη,τοτε βρισκουμε την ευκλειδια αποσταση κάθε χρηστη από το συνδεδεμενο χρηστη.Αφου συγκεντρωσουμε ολες τις ευκλειδειες απόστασεις επιλεγουμε τις κ=2 πιο κοντικες σε εμας ,αυτές δηλαδη που εχουν την μικροτερη τιμη.Ετσι βρισκουμε τους 2 πιο κοντινους μας χρηστες από αποψη like και comment(Tους πιο similar σε εμας δηλαδη).Οταν βρουμε τους similar user παιρνουμε παλι όλα τα αρθρα και με βαση αυτους φτιαχνουμε ένα rating για κάθε αρθρο που θα μας χρησιμευσει στο να βρουμε τα πιο καταλληλα αρθρα για recommendation.To τυπο αυτό για το rating τον πηραμε από τις διαφανειες στο eclass.Εν συνεχεια αφου εχουμε φτιαξει τα ratings βρισκουμε τα κ=3 καλυτερα και τα εμφανιζουμε στην αρχικη σελιδα του χρηστη μαζι με τα αρθρα των φιλων μας.Σε περιπτωση που καποιος δεν εχει φιλους εμφανιζονται στην αρχικη μας σελιδα μονο τα δικα μας αρθρα.Αλλιως αν εχουμε φιλους δεν εμφανιζονται τα διμας αρθρα ,μονο των φιλων μας και τα recommended.