Πανεπιστήμιο Κρήτης
HY359 – Διαδικτυακός Προγραμματισμός (Web Programming)
Χειμερινό Εξάμηνο 2017
Διδάσκων: Παναγιώτης Παπαδάκος
Υπεύθυνος Βοηθός: Κωνσταντίνος Σγόντζος / Παναγιώτης Παπαδάκος

3η Σειρά Ασκήσεων

Διάρκεια: 15/11 – 29/11
Αξία: 8% του τελικού σας βαθμού
Θεματική ενότητα : **Servlets**

Άσκηση 1. [LQ] Εγγραφή & Λειτουργίες Εγγεγραμμένου Χρήστη [65%]

Αποτελεί μέρος του συνόλου των ασκήσεων για το sprint του Liquid Democracy

A. Εγγραφή χρήστη [40%]

Καλείστε να καλύψετε τις ανάγκες εγγραφής ενός χρήστη σε μια διαδικτυακή πλατφόρμα. Η φόρμα εγγραφής θα πρέπει να περιέχει τα πεδία εγγραφής που υλοποιήσατε στη πρώτη άσκηση αλλά πλέον η εγγραφή θα ολοκληρώνεται με τη δημιουργία ένος χρήστη στη μεριά του server και την αποθήκευση των στοιχείων του στην αντίστοιχη βάση δεδομένων.

Για να υλοποιήσετε τη συγκεκριμένη λειτουργικότητα θα πρέπει να χρησιμοποιήσετε τον κώδικα που σας έχει δοθεί για να αποθηκεύσετε τα εγγεγραμμένα μέλη στη βάση. Συγκεκριμένα πρέπει να κάνετε clone το repository https://bitbucket.org/papadako/lq.git, του οποίου αντίτυπο του project μπορείτε να χρησιμοποιήσετε σαν βάση για το δικό σας project (είναι maven web-app project).
ΠΡΟΣΟΧΗ! Δεν πρέπει να γίνει clone μέσα στο δικό σας git repository με τις ασκήσεις. Αντί αυτού μπορείτε να το κάνετε clone κάπου εξωτερικά και να πάρετε τα αρχεία που σας δίνονται για να τα χρησιμοποιήσετε στο δικό σας maven project. Αν το αντιγράψετε εσωτερικά στον φάκελο a3, φροντίστε να διαγράψετε το a3/lq/.git directory. Αν θέλετε να χρησιμοποιήσετε το ίδιο ως Μελλοντικά θα γίνονται αλλαγές στον συγκεκριμένο κώδικα ανάλογα με τις ανάγκες της εργασίας. Σε περίπτωση bugs/issues στον παραπάνω κώδικα, μπορείτε να ανοίγετε issues στον issue tracker του repository ώστε να παίρνετε και το credit!

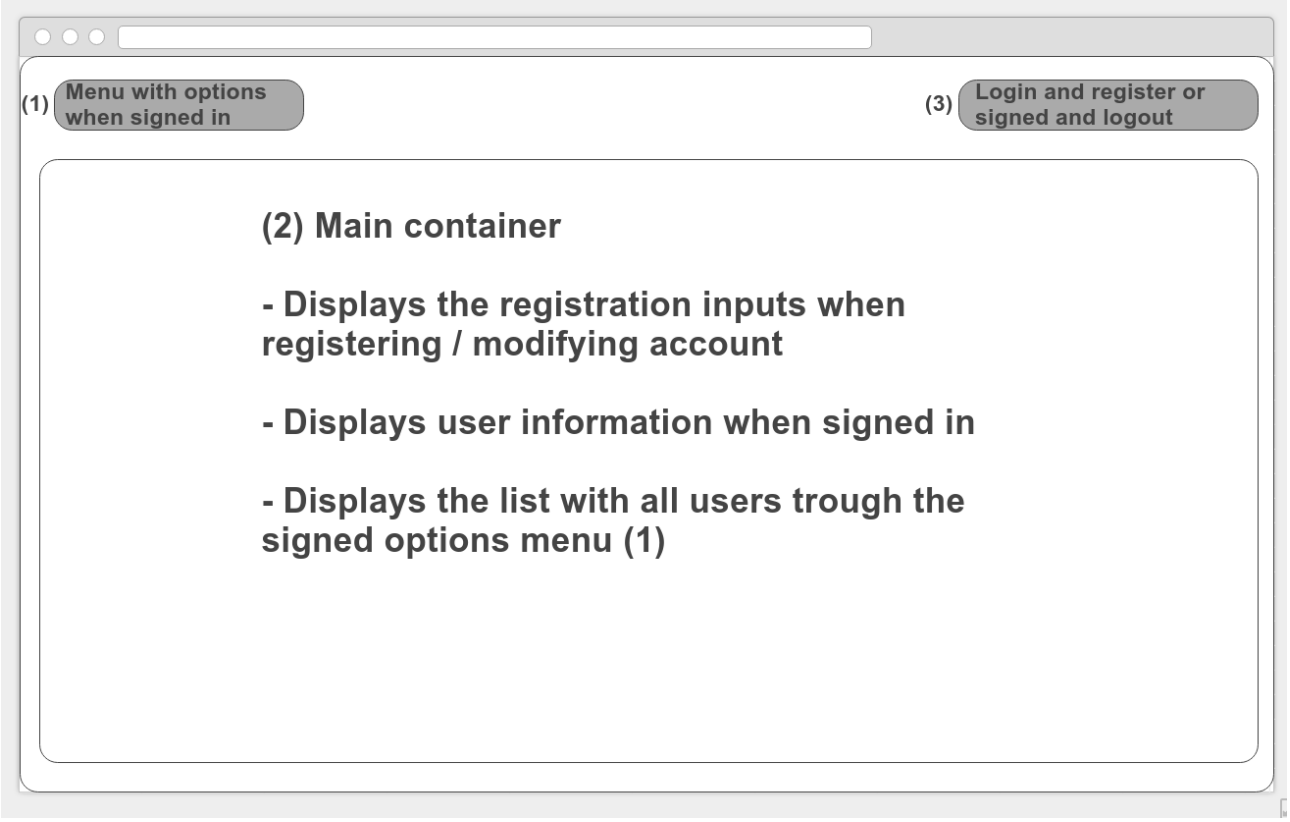
Στον καθένα σας έχουν δοθεί ατομικά credentials για τη σύνδεση στη βάση τα οποία μπορείτε να δείτε αν κάνετε pull στο repository που χρησιμοποιείτε. Αυτά τα credentials πρέπει να χρησιμοποιηθούν στην gr.csd.uoc.cs359.winter2017.lq.db.CS359DB για να μπορεί το exampleAPI που σας έχει δοθεί να παίξει.

Η κλάση UserDB είναι υπεύθυνη για λειτουργίες που σχετίζονται με τους χρήστες που κρατάμε στη βάση (addUser, getUser, updateUser, checkValidUserName, checkValidEmail). Επιπλέον υπάρχει η κλάση User που κρατάει πληροφορίες για κάποιον χρήστη.

Στο τέλος της φόρμας εγγραφής να υπάρχει το κουμπί Εγγραφή.

- Η μικροϋπηρεσία (servlet) που θα φτιάξετε θα πρέπει να ελέγχει ότι οι τιμές που έχει δώσει ο χρήστης είναι έγκυρες όπως και οι έλεγχοι που κάνατε στο front-end (μπορείτε να ξαναχρησιμοποιήσετε τα regex που φτιάξατε στην A1 και για τη μεριά του server). Διαφορετικά σε περίπτωση που υπάρχει κάποιο λάθος από τη μεριά του client θα πρέπει ο server να επιστρέφει το κατάλληλο status code και τα κατάλληλα μηνύματα λάθους (π.χ. Status code 400 – Bad Request και τα πεδία που δεν έχουν συμπληρωθεί σωστά).
- Σε περίπτωση που ο χρήστης δηλώσει ένα username το οποίο υπάρχει ήδη στη βάση, θα πρέπει να τυπωθεί ότι το συγκεκριμένο username χρησιμοποιείται ήδη. Αντίστοιχα και για το email. Αυτό θα γίνεται τη στιγμή που συμπληρώνεται το αντίστοιχο πεδίο.
- Εφόσον έχουν συμπληρωθεί όλα τα υποχρεωτικά πεδία με σωστό τρόπο και ο χρήστης κάνει click στο κουμπί "Εγγραφή", η διαδικασία εγγραφής πρέπει να ολοκληρωθεί επιτυχώς, σώζοντάς τα στη βάση. Το servlet θα πρέπει να επιστρέψει όλα τα στοιχεία που έδωσε ο χρήστης και το μήνυμα «Η εγγραφή σας πραγματοποιήθηκε επιτυχώς» στο μέρος της σελίδας όπου εμφανίζουμε το content (π.χ. στο κεντρικό div, δείτε το παρακάτω ενδεικτικό image).
- **Καλείστε να υλοποιήσετε τα παραπάνω με χρήση ajax requests (όχι με χρήση forms)!**
- Η σελίδα που θα φτιάξετε δεν θα πρέπει ποτέ να γίνεται refresh (SPA – single page application)

Ένα ενδεικτικό UI που μπορείτε να χρησιμοποιήσετε δίνετε παρακάτω



Β. Λειτουργικότητα για εγγεγραμμένο χρήστη [25%]

Ένας εγγεγραμμένος χρήστης θα πρέπει να μπορεί:

- να κάνει login, αλλιώς το σύστημα να του πετάει κατάλληλο μήνυμα μη επιτυχημένου login
- να κάνει logout
- να παραμένει συνδεδεμένος μέχρι να τελειώσει η συνεδρία - session (να μη χρειάζεται δηλαδή να κάνει login ξανά μέχρι να κάνει log-out).
- να δει τα στοιχεία του συγκεντρωτικά σε μια σελίδα και να μπορεί να τα αλλάξει,
- να δει μία λίστα με όλα τα εγγεγραμμένα μέλη

Καλείστε να υλοποιήσετε τα παραπάνω με χρήση ajax requests (όχι με χρήση forms)!
Η σελίδα που θα φτιάξετε δεν θα πρέπει ποτέ να γίνεται refresh (SPA – single page application)

Άσκηση 2. XSS – Cross site scripting [20%]

Το Cross Site Scripting (XSS) είναι μία μορφή επίθεσης που χρησιμοποιείται κυρίως σε εφαρμογές διαδικτύου. Ένα τυπικό σενάριο τέτοιου είδους επίθεσης είναι το εξής: Μία ιστοσελίδα επιτρέπει την εισαγωγή στοιχείων από το χρήστη (πχ. σχόλια) τα οποία στη συνέχεια δημοσιεύει. Εάν κάποιος χρήστης (ο επιτιθέμενος) εισάγει κάποιο script προς δημοσίευση αντί απλού κειμένου, αυτό τελικά θα αποτελεί μέρος του html και θα εκτελείται κανονικά μετά από κάθε επίσκεψη.

Σε αυτή την άσκηση αρχικά η σελίδα εγγραφής νέου μέλους θα είναι μη ασφαλής και θα κάνετε επίθεση. Συγκεκριμένα, αυτό μπορεί να γίνει με οποιοδήποτε από τα πεδία κειμένου. Σας ζητείται να:

- Εφαρμόσετε την επίθεση ώστε να αλλάξετε το χρώμα της σελίδας (CSS related) που εμφανίζει τα στοιχεία που έδωσε ο χρήστης και να εισάγετε ένα button στη σελίδα το οποίο στο onclick θα κάνει κάποιο action (π.χ alert ή να στέλνει το χρήστη σε μια εξωτερική σελίδα), και περιγράψτε σύντομα πώς πετύχατε την επίθεση. Προσοχή, οι τωρινές εκδόσεις των browsers έχουν εργαλεία τα οποία προσπαθούν να αποτρέψουν τέτοιου είδους επιθέσεις, οπότε θα πρέπει να απενεργοποιήσετε τη συγκεκριμένη λειτουργικότητα. Προτιμότερο είναι να χρησιμοποιείσετε τον firefox που δεν έχει κάποιο

default φιλτρο (αν έχετε όμως το noscript plugin ή άλλα παρεμφερή θα σας ειδοποιήσει για τέτοιου είδους XSS επιθέσεις, τους ελέγχους των οποίων μπορείτε να απενεργοποιήσετε).
- Διορθώστε την εφαρμογή ώστε να μην είναι πλέον ευάλωτη στην επίθεση.

Το υπόλοιπο 15% του βαθμού θα κατανεμηθεί βάσει των παρακάτω 3 κριτηρίων:

- **jshint, html validator, code quality – 5%:** θα κρίνεται από το αν η σελίδα σας δεν εμφανίζει λάθη/warnings στον jshint, στον validator, καθώς και στη γενική ποιότητα του κώδικά σας
- **ελκυστικότητα εμφάνισης σελίδων (στυλιστική συνέπεια) – 5%**
- **git – 5%:** θα κρίνεται από τη σωστή χρήση του git (π.χ. να υπάρχουν αρκετά commits που να περιγράφουν με σαφήνεια πως κάνατε την άσκηση, με κατανοητή περιγραφή, καθαρό ιστορικό, κτλ.)

Μη ξεχνάτε τη χρήση του "use strict"; για την JavaScript.

Το parse ενός JSON string γίνεται με χρήση της JSON.parse(str) που επιστρέφει το javascript object που αντιστοιχεί στο str, δεδομένου ότι το str είναι μια σωστή αναπαράσταση JSON .

Μπορείτε να χρησιμοποιήσετε κάποιον online linter όπως ο jshint http://jshint.com/ για να βελτιώσετε την ποιότητα του js κώδικά σας.

Προτείνεται η χρήση του netbeans σαν IDE. Επίσης προσπαθήστε να οργανώσετε με όμορφο τρόπο τα servlets που θα χρησιμοποιήσετε.

Τρόπος Παράδοσης

Οι ασκήσεις θα παραδίδονται μόνο μέσω git, σύμφωνα με τις οδηγίες που σας έχουν δοθεί. Συγκεκριμένα στο repository σας στο bitbucket το οποίο θα πρέπει να έχει γίνει ήδη share στο hy359, στο folder a3 θα πρέπει να υπάρχει ένα subfolder όπου θα περιέχεται το maven project της εργασίας σας με όνομα lq. Φροντίστε να δομήσετε όμορφα **Θα πρέπει να φροντίσετε ότι όλα όσα έχετε κάνει έχουν γίνει σωστά commit και βρίσκοντα online στο bitbucket. Αφού όλα είναι ok θα πρέπει να κάνετε tag την άσκηση σας "git tag a3" και να ανεβάσετε το tag και στο bitbucket μέσω της εντολής "git push --tags".**

Προγραμματίστε καλά το χρόνο σας και αποφύγετε να ασχοληθείτε με την εργασία τελευταία στιγμή!

Καλό είναι σαν τελευταίο έλεγχο πριν σταματήσετε να ασχολείστε με την εργασία να κάνετε clone το repository σας κάπου τοπικά και να ελέγξετε αν υπάρχει κάποιο πρόβλημα. Με αυτόν τον τρόπο άλλωστε θα βαθμολογηθείτε/εξεταστείτε.

Στις 23:59 της 29/11 θα γίνει αυτόματο pull από όλα τα repositories που έχουν γίνει share στο hy359 και βάσει αυτών θα βαθμολογηθείτε. Εκπρόθεσμες ασκήσεις **δεν θα γίνονται δεκτές**.

Αντιγραφή

Σε περίπτωση αντιγραφής θα μηδενίζονται άμεσα οι εργασίες όλων των εμπλεκόμενων.

Και λίγα λόγια για το μεράκι....

Θέσετε ως στόχο να υλοποιήσετε τις εργασίες σας με μεράκι και δημιουργικότητα και να αναδείξετε με όμορφο τρόπο το χρόνο που αφιερώσατε! Μην αφήνετε τα πράγματα στην τύχη και δώστε κάτι από τον εαυτό σας και το χαρακτήρα σας!

Καλή εργασία