Πανεπιστήμιο Κρήτης
HY359 – Διαδικτυακός Προγραμματισμός (Web Programming)
Χειμερινό Εξάμηνο 2016
Διδάσκων: Παναγιώτης Παπαδάκος
Υπεύθυνος Βοηθός: Μαρία Στρατήγη

3η Σειρά Ασκήσεων

Διάρκεια: 4/11 – 14/11
Αξία: 8% του τελικού σας βαθμού
Θεματική ενότητα : **Servlets**

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

Αποτελεί μέρος του συνόλου των ασκήσεων που θα συνενωθούν στο τέλος.

Α. Εγγραφή χρήστη [40%]

Καλείστε να καλύψετε τις ανάγκες εγγραφής ενός χρήστη σε μια διαδικτυακή πλατφόρμα. Η φόρμα εγγραφής θα πρέπει να περιέχει τα εξής πεδία:

***Username:** Κείμενο τουλάχιστον 8 χαρακτήρων. Είναι μοναδικό.
***email:** Θα πρέπει να περιέχει μία μόνο εμφάνιση του χαρακτήρα '@' και τουλάχιστον μία εμφάνιση του χαρακτήρα '.' Κάθε χρήστης γράφεται μόνο μία φορά με το ίδιο email.
***Κωδικός χρήστη:** Κείμενο 6 εώς 10 χαρακτήρων. Κατά την πληκτρολόγηση θα πρέπει να εμφανίζονται * αντί των πραγματικών χαρακτήρων. Θα πρέπει να περιέχει τουλάχιστον ένα λατινικό χαρακτήρα, τουλάχιστον έναν αριθμό και τουλάχιστον ένα σύμβολο (π.χ. '#', '$', '%', κτλ.).
***Επιβεβαίωση κωδικού:** Πρέπει να είναι ίδιος με τον κωδικό στο πεδίο 2 (ώστε να αποφύγουμε λανθασμένη πληκτρολόγηση κωδικού από τον χρήστη).
***Όνομα:** Ένα κείμενο από 3 μέχρι 20 χαρακτήρες. Αποδεκτοί χαρακτήρες είναι οι λατινικοί και οι ελληνικοί χαρακτήρες.
***Επώνυμο:** Ένα κείμενο από 4 μέχρι 20 χαρακτήρες. Αποδεκτοί χαρακτήρες είναι οι λατινικοί και οι ελληνικοί χαρακτήρες.
***Ημερομηνία γέννησης:** Αποτελείται από την ημέρα, το μήνα, και το έτος και μπορούν να είναι drop-down menus ή απλό κείμενο. Αν είναι κείμενο πρέπει να δίνεται υποχρεωτικά στη μορφή HH/MM/XXXX (π.χ. 18/08/1984). Πρέπει να γίνεται έλεγχος για έγκυρη ημερομηνία. Επιπλέον δεν επιτρέπεται η εγγραφή σε παιδιά κάτω των 15 ετών.
***Φύλο:** Ένα radio button με τιμές 'Αγόρι', 'κορίτσι' και 'μη εφαρμόσιμο'.
***Χώρα:** Ένα drop-down μενού όπου θα εμφανίζονται όλες οι χώρες (με προεπιλεγμένη την Ελλάδα).
***Πόλη:** Ένα κείμενο από 2 εως 50 χαρακτήρες.
***Περισσότερες πληροφορίες:** Ελεύθερο κείμενο εώς 500 χαρακτήρες.
***Φωτογραφία:** Ένα κουμπί για επιλογή φωτογραφίας σαν avatar

Στο τέλος της φόρμας 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 σύμφωνα με τις παραπάνω περιγραφές και ότι το αρχείο που δόθηκε είναι εικόνα με μέγεθος μικρότερο από 2 MB. Δείτε στις διαλέξεις πως μπορείτε να κάνετε upload ένα αρχείο.
- Όσα πεδία αρχίζουν με * είναι υποχρεωτικά, δηλαδή ο χρήστης πρέπει να τα συμπληρώσει για να μπορέσει να εγγραφεί στο σύστημα.
- Σε περίπτωση που ο χρήστης δεν δώσει φωτογραφία, το σύστημα σας θα πρέπει χρησιμοποιεί μία προκαθορισμένη εικόνα.
- Σε περίπτωση που ο χρήστης δεν έχει συμπληρώσει κάποιο από τα υποχρεωτικά πεδία ή υπάρχουν μη έγκυρες τιμές σε αυτά, το σύστημα πρέπει να ειδοποιεί το χρήστη με τα κατάλληλα μηνύματα.
- 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 και το servlet θα πρέπει να επιστρέφει μια σελίδα με όλα τα στοιχεία που έδωσε ο χρήστης και το μήνυμα «Η εγγραφή σας πραγματοποιήθηκε επιτυχώς». Συγκεκριμένα, τα στοιχεία θα πρέπει να αποθηκεύονται σε κάποια δομή δεδομένων. Για παράδειγμα μπορείτε να αποθηκεύσετε τα στοιχεία σε ένα hashmap, με key το username του χρήστη

και value ένα instance μιας κλάσης π.χ. User που θα κρατά όλες τις παραπάνω πληροφορίες. Τη συγκεκριμένη δομή θα τη κρατάει το servlet για όλα τα requests. Είστε ελεύθεροι να εμφανίσετε αυτές τις πληροφορίες όπως επιθυμείτε.

- Κάποιοι από τους παραπάνω ελέγχους μπορούν να γίνουν στη μεριά του client (JavaScript/HTML5), ενώ οι υπόλοιποι πρέπει να γίνουν στη μεριά του server (servlets) ή και στους δύο. Για παράδειγμα για το username θα πρέπει να γίνει έλεγχος στη μεριά του server ότι δεν έχει ξαναγίνει εγγραφή του στη δομή που κρατά.
- **Καλείστε να υλοποιήσετε τα παραπάνω με χρήση ajax requests και όχι με forms.**

Β. Λειτουργικότητα για εγγεγραμμένο χρήστη [25%]

Ένας εγγεγραμμένος χρήστης θα πρέπει να μπορεί:

- να κάνει login, αλλιώς το σύστημα να του πετάει κατάλληλο μήνυμα
- να παραμένει συνδεδεμένος μέχρι να τελειώσει η συνεδρία με χρήση cookies (να μη χρειάζεται δηλαδή να κάνει login ξανά μέχρι να κάνει sign-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

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 προς εμφάνιση για τους επόμενους χρήστες 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 που εμφανίζει τα στοιχεία που έδωσε ο χρήστης ή να κατευθύνετε το χρήστη σε μια νέα σελίδα (windows.location), 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. Για παράδειγμα για τον chrome θα πρέπει να δώσετε την παράμετρο – **disable-xss-auditor** κατά την εκκίνηση του browser.
- 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.)

Σημειώσεις:

Μη ξεχνάτε τη χρήση του *"use strict";* για την *JavaScript*
Μπορείτε να χρησιμοποιήσετε το *jshint* *http://jshint.com/* για να βελτιώσετε την ποιότητα του *js* κωδικά σας.
Θα *πρέπει* να έχετε στήσει κάποιο *servlet container π.χ. tomcat/glassfish.* Προτείνεται η χρήση του *netbeans* όπου μπορείτε να επιλέξετε κάποιον από τους παραπάνω *containers* κατά την εγκατάσταση.

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, όπου θα περιέχεται ο κώδικας για κάθε άσκηση. **Θα πρέπει να φροντίσετε ότι όλα όσα έχετε κάνει έχουν γίνει σωστά commit και βρίσκοντα online στο bitbucket.**

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

Στις 00:00 της 15/11 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** (μόνο σε ειδικές περιπτώσεις σε συμφωνία με τον διδάσκοντα).

Αντιγραφή

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

Καλή εργασία