ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Σχεδίαση Εφαρμογών και Υπηρεσιών Διαδικτύου

Διδάσκων: Απόστολος Γκάμας (Διδάσκων –ΠΔ 407/80)

3η Άσκηση

Σκοπός

Σκοπός αυτής της άσκησης είναι η εξοικείωση με την γλώσσα PHP η οποία αναφέρθηκε στην θεωρία. Σκοπός της άσκησης είναι η χρήση των δυνατοτήτων διαχείρισης συνόδου (session), διαχείρισης φορμών (POST/GET) και πρόσβασης σε βάσεις δεδομένων για την δημιουργία δυναμικών εφαρμογών πάνω από το Διαδίκτυο. Οι φοιτητές θα δημιουργήσουν μια υπηρεσία διαχείρισης χρηστών.

Ημερομηνίες

Οι αναφορές των ασκήσεων θα πρέπει να αποσταλούν στον διδάσκοντα με την χρήση του e-class μέχρι την **Παρασκευή 11/01/2008**.

Περιγραφή – Ζητούμενα

Ζητούμενο της άσκησης είναι η δημιουργία μια εφαρμογής διαχείρισης χρηστών (όπου θεωρητικά θα διαχειρίζεται του χρήστες μια υπηρεσίας). Για κάθε χρήστη θα πρέπει να αποθηκεύονται τα παρακάτω στοιχεία:

- Login: όχι μικρότερο από 4 χαρακτήρες και όχι μεγαλύτερο από 10 χαρακτήρες και μοναδικό για κάθε χρήστη (υποχρεωτικό).
- Password: όχι μικρότερο από 4 χαρακτήρες (υποχρεωτικό).
- E-mail: Θα πρέπει να περιέχει το σύμβολο @ (υποχρεωτικό).
- Ονοματεπώνυμο: Προαιρετικό στοιχείο.
- Διαχειριστής: Το στοιχείο αυτό θα καθορίζει εάν ο χρήστης είναι διαχειριστής ή όχι (υποχρεωτικό)

Δικαίωμα πρόσβασης στην υπηρεσία την οποία θα δημιουργήσετε θα έχουν μόνο οι χρήστες με την ιδιότητα του Διαχειριστή. Οι διαχειριστές θα μπορούν να επιτελέσουν τις παρακάτω λειτουργίες:

- Είσοδος στην υπηρεσία (login) μετά από επιτυχημένη εισαγωγή συνδυασμού (login/password). Μετά την επιτυχημένη πιστοποίηση του διαχειριστή θα προωθείται στην σελίδα με την λίστα των χρηστών. Μετά από αποτυχημένη πιστοποίηση του διαχειριστή θα οδηγείται ξανά στην σελίδα εισόδου στην υπηρεσία (login). Στην σελίδα login θα πρέπει να παρέχεται η δυνατότητα η εφαρμογή να θυμάται το login του χρήστη (θα πρέπει ο χρήστης να μπορεί να επιλέγει την λειτουργία αυτή μέσα από ένα check-box) (υλοποίηση με χρήση Cookie).

- Λίστα χρηστών: (Σε αυτή την σελίδα οδηγείται ο διαχειριστής μετά από επιτυχημένο login και αποτελεί την κεντρική σελίδα της εφαρμογής). Πρόσβαση στην λίστα των χρηστών όπου θα εμφανίζονται όλοι οι χρήστες ταξινομημένοι σύμφωνα με το login τους και θα υπάρχει δίπλα από κάθε login σύνδεσμοι για διαγραφή του συγκεκριμένου χρήστη (μετά από επιβεβαίωση), αλλαγή των στοιχείων του συγκεκριμένου χρήστη (μέσα από άλλη σελίδα) και προβολή των στοιχείων του συγκεκριμένου χρήστη (μέσα από άλλη σελίδα).
- Προσθήκη χρήστη μέσα από φόρμα.
- Διαγραφή χρήστη (επιλέγοντας τον χρήστη από την λίστα χρηστών)
- Αλλαγή στοιχείων χρήστη μέσα από φόρμα (επιλέγοντας τον χρήστη από την λίστα χρηστών). Οι διαχειριστές θα μπορούν να αλλάξουν τα παρακάτω στοιχεία ενός χρήστη: Password, E-mail, Ονοματεπώνυμο, Διαχειριστής.
- Προβολή στοιχείων χρήστη (εκτός του password) (επιλέγοντας τον χρήστη από την λίστα χρηστών).
- Έξοδος από την υπηρεσία (logout).

Η εφαρμογή θα πρέπει να έχει τα εξής χαρακτηριστικά:

- Αποθήκευση των πληροφοριών σε βάση δεδομένων mySQL.
- Όλες οι σελίδες θα πρέπει να προστατεύονται από ασφάλεια με την έννοια ότι ο χρήστης θα πρέπει να έχει κάνει login πριν προσπελάσει κάποια σελίδα της εφαρμογής (υλοποίηση με χρήση session). Σε περίπτωση που ένας χρήστης επιχειρήσει να προσπελάσει μια σελίδα της εφαρμογής, χωρίς προηγουμένως να έχει login να οδηγείται στην σελίδα εισόδου στην υπηρεσία (login).
- Τα password θα πρέπει να αποθηκεύονται κρυπτογραφημένα στην βάση δεδομένων.
- Θα πρέπει να γίνονται οι απαραίτητοι έλεγχοι και σε περίπτωση λάθους να επιστρέφονται τα αντίστοιχα μηνύματα (πχ. Υπάρχει ήδη χρήστης με αυτό το login παρακαλώ επιλέξτε άλλο login) κλπ. Ο έλεγχος μπορεί να γίνεται είτε μόνο στο εξυπηρετητή ή είτε και στο πελάτη και τον εξυπηρετητή.

Για την υλοποίηση των διάφορων χαρακτηριστικών προτείνονται τα παρακάτω:

- Για την υλοποίηση του μηχανισμού ασφαλείας προτείνεται η χρήση session ως εξής:
    - Μετά από ένα πετυχημένο login θέτουμε μια session μεταβλητή η οποίο σηματοδοτεί το πετυχημένο login.
    - Στην αρχή κάθε σελίδας καλούμε μια συνάρτηση η οποία ελέγχει την ύπαρξη της σχετική session μεταβλητής, εάν αυτή υπάρχει η εκτέλεση της σελίδας συνεχίζει κανονικά ενώ σε αντίθετη περίπτωση ο χρήσης γίνεται redirect στην σελίδα login.
    - Όταν ο χρήστης κάνει logout καταστρέφουμε το session.
- Η υλοποίηση της δυνατότητας η εφαρμογή να θυμάται το login του χρήστη προτείνεται να υλοποιηθεί με χρήση Cookie. Η ημερομηνία και ο χρόνος του τελευταίου επιτυχημένου Login προτείνεται να αποθηκεύεται σε ένα Cookie στον υπολογιστή του χρήστη.
- Προσθήκη χρήστη μέσα από φόρμα.
- Για την κρυπτογράφηση του password προτείνεται η χρήση της συνάρτησης password στο επίπεδο της SQL.
- Για την υλοποίηση του μηχανισμού redirect προτείνεται η χρήση της συνάρτησης header της php.

- Για την αποφυγή προβλημάτων με το input του χρήστη προτείνεται η χρήση των συναρτήσεων addslashes και stripslashes της php.
- Προτείνεται η επαναχρησιμοποίηση κώδικα με την χρήση ενός αρχείου που θα γίνεται include σε όλα τα php αρχεία και θα περιέχει τις κοινές λειτουργίες σε μορφή συναρτήσεων (πχ. Σύνδεση με την βάση, έλεγχος αν ο χρήστης έχει κάνει login κλπ).

Το παραδοτέο της άσκησης θα είναι οι html σελίδες και τα php scripts οι οποίες θα περιέχουν τον κώδικα PHP (με σχόλια στα βασικά σημεία του κώδικα) καθώς και μια σύντομη αναφορά σχετικά με τα scripts τα οποία έχετε υλοποιήσει και τις λειτουργίες που αυτά επιτελούν

Παρατηρήσεις

1. Περισσότερες διευκρινήσεις θα δοθούν στο εργαστήριο.
2. Κατά την δημιουργία της βάσης δεδομένων μέσα από το SQL script προσθέστε και ένα χρήστη τύπου διαχειριστή για την αρχική λειτουργία της εφαρμογής (βλέπε παράρτημα Β).

Παράρτημα Α: PHP references

Περισσότερες πληροφορίες για την PHP μπορείτε να βρείτε στο παρακάτω URL: http://www.php.net

Παράρτημα Β: SQL Script δημιουργίας βάσης δεδομένων

```
CREATE TABLE `users` (`login` VARCHAR( 10 ) NOT NULL , `password` VARCHAR( 100 ) NOT NULL , `email` VARCHAR( 100 ) NOT NULL , `name` VARCHAR( 100 ) , `administrator` INT NOT NULL , PRIMARY KEY ( `login` ) ) ;

INSERT INTO `users` ( `login` , `password` , `email` , `name` , `administrator` ) VALUES ('admin', password('admin'), 'admin@test.com', 'First Admin', '1');
```