

Μάθημα: ΥΠΗΡΕΣΙΟΣΤΡΕΦΕΣ ΛΟΓΙΣΜΙΚΟ
Αναπληρωτής Καθηγητής Ευάγγελος Σακκόπουλος
Ακαδημαϊκό Έτος 2024-2025

Θέμα εργασίας: «Διαχείριση και σύνθεση ομάδων εργαστηρίων» & Κατασκευή REST API

Η εργασία έχει δύο θέματα και πρέπει να παραδοθούν και τα δύο.

Θέμα 1:

Διαχείριση portal εργαστηριακών ασκήσεων

Ζητείται να γίνει ένα αλληλεπιδραστικό υπηρεσιοστρεφές λογισμικό διαχείρισης και σύνθεσης ομάδων εργασίας. Πρόκειται για ένα σύστημα στο οποίο θα μπορεί κανείς να αναζητήσει και να συνομιλήσει χρήστες που έχουν τα ίδια hobby-ενδιαφέροντα, ίδια μαθήματα ή να διαμορφώσουν ομάδες εργασίας.

Συγκεκριμένα ζητούνται τα παρακάτω:

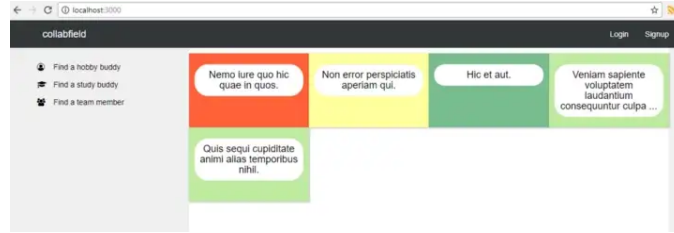
ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ

1. ΔΥΝΑΤΟΤΗΤΑ ΝΑ ΑΠΟΣΤΕΛΝΕΤΕ POSTS ΝΑ ΑΝΑΖΗΤΕΪΤΕ POSTS ΚΑΙ ΝΑ ΤΑ ΚΑΤΗΓΟΡΙΟΠΟΙΕΪΤΕ ΓΙΑ ΠΟΛΛΑΠΛΟΥΣ ΧΡΗΣΤΕΣ
2. ΔΥΝΑΤΟΤΗΤΑ ΝΑ ΠΡΟΣΤΙΘΕΝΤΑΙ ΧΡΗΣΤΕΣ ΩΣ ΠΡΟΣΩΠΙΚΕΣ ΕΠΑΦΕΣ ΕΝΟΣ ΧΡΗΣΤΗ
3. ΑΥΘΕΝΤΙΚΟΠΟΙΗΣΗ (ΜΕ ΤΟ DEVISE Ή ΑΝΤΙΣΤΟΙΧΟ)
 - 3.1. ΑΥΘΕΝΤΙΚΟΠΟΙΗΣΗ ΜΕ ΧΡΗΣΗ FACEBOOK Ή GOOGLE Ή TWITTER IDP
 - 3.2. ΑΥΘΕΝΤΙΚΟΠΟΙΗΣΗ ΜΕ BASIC AUTHENTICATION ΤΟΠΙΚΑ ΣΤΟ ΣΥΣΤΗΜΑ ΣΑΣ
4. ΑΜΕΣΑ ΜΗΝΥΜΑΤΑ (ΡΟΡUP ΠΑΡΑΘΥΡΟ ΓΙΑ ΧΩΡΙΣΤΑ ΜΗΝΥΜΑΤΑ) ΜΕ ΤΗ ΔΥΝΑΤΟΤΗΤΑ ΑΠΟΣΤΟΛΗΣ ΠΡΟΣΩΠΙΚΩΝ ΜΗΝΥΜΑΤΩΝ ΣΕ ΑΛΛΟ ΧΡΗΣΤΗ Ή ΠΡΟΣ ΟΜΑΔΑ ΠΟΛΛΩΝ ΧΡΗΣΤΩΝ
5. ΆΜΕΣΗ ΕΝΗΜΕΡΩΣΗ ΜΕ NOTIFICATION

Β) Συνοδευτικά εγχειρίδια

Η εφαρμογή θα πρέπει να συνοδεύεται από Εγχειρίδιο χρήστη (user manual)

- **Η ομάδα θα πρέπει να καταχωρήσει τον κώδικά της σε σύστημα *git* και να παρουσιάσει τον σχετικό κώδικα κατά την εξέταση. Σημειώνεται ότι κάθε μέλος της ομάδας θα πρέπει να είναι μέλος του *project github* ή *gitlab* που θα παραδοθεί και να έχει συμμετάσχει ενεργά με ανάρτηση σχετικού κώδικα (δείτε [σύντομο οδηγό github](#) & <http://git-class.gr/>)**
- **Μπορείτε να χρησιμοποιήσετε για καθοδήγηση την υλοποίηση του <https://github.com/domagude/collabfield> και την [παρουσίαση του](#) για να υλοποιήσετε την εργασία σας σύμφωνα με το παράδειγμα (στο δικό σας github).**



Θέμα 2:

Δημιουργία REST API.

Δημιουργήστε ένα REST API με χρήση RAILS με τις παρακάτω λειτουργίες για μία [λίστα εκκρεμοτήτων](#) (todo) η οποία έχει λεπτομέρειες για κάθε todo (list of todo items).

Προσοχή: Η ανάπτυξή σας θα πρέπει να περιλαμβάνει τα κατάλληλα test (test driven development) καθώς και δοκιμή των test με httpie:

Θα ενεργοποιήσετε τα ακόλουθα endpoint για τις λειτουργικότητες αντίστοιχα.

Endpoint	Functionality
POST /signup	Signup
POST /auth/login	Login
GET /auth/logout	Logout
GET /todos	List all todos and todo items
POST /todos	Create a new todo
GET /todos/:id	Get a todo
PUT /todos/:id	Update a todo
DELETE /todos/:id	Delete a todo and its items
GET /todos/:id/items/:iid	Get a todo item
POST /todos/:id/items	Create a new todo item
PUT /todos/:id/items/:iid	Update a todo item
DELETE /todos/:id/items/:iid	Delete a todo item

- Η εφαρμογή θα συνοδεύεται από Εγχειρίδιο χρήσης (user manual) τύπου openapi [swagger](#)

- Η εργασία και τα δύο θέματα και θα πρέπει να γίνει από ομάδες των 1-2 ατόμων.
- Η υλοποίηση της εργασίας να γίνει με χρήση Ruby on Rails
- Η εργασία σας να εγκατασταθεί στο fly.io ή στο gitpod ή render η άλλο cloud της επιλογής σας
- Η εργασία μπορεί εναλλακτική να υλοποιηθεί με nodejs ή .net core
- Η ομάδα θα πρέπει να καταχωρήσει τον κώδικά της στο σύστημα github ή gitlab και να παρουσιάσει τον σχετικό κώδικα κατά την εξέταση. (δείτε [σύντομο οδηγό github](#) & <http://git-class.gr/>)
- Η ισχύς της παρούσας εργασίας είναι για το ακαδημαϊκό έτος 2024-2025

Η παράδοση της εργασίας θα πραγματοποιηθεί στο thales στην ημερομηνία που αναφέρεται στο σύστημα. Θα περιλάβετε όλο το project σας καθώς και μία σύντομη αναφορά σε μορφή DOC/ODT/PDF με το εγχειρίδιο χρήσης ή το openapi. Θα περιλάβετε το ονοματεπώνυμο και το ΑΜ όλων των μελών της ομάδα σας. Η εξέταση της εργασίας θα πραγματοποιηθεί με οδηγίες που θα σας ανακοινωθούν.

Καλή Επιτυχία!