**Σχολή Αρχιτεκτονικής, Πληροφορικής και Μηχανικής**

**Κωδικός Μαθήματος:** CN6035

**Εργασίας για το Μάθημα "Mobile & Distributed Systems"**

**Τίτλος Εργασίας: Ανάπτυξη Εφαρμογής Κράτησης Θέσεων σε Εστιατόριο μέσω Κινητής Συσκευής**

**Περιγραφή**

Η εργασία αποσκοπεί στην ανάπτυξη μιας εφαρμογής κινητού που επιτρέπει στους χρήστες να κάνουν κρατήσεις τραπεζιών σε εστιατόρια. Η εργασία απαιτεί τη συνεργασία και τον συνδυασμό frontend, backend και βάσης δεδομένων, ώστε οι φοιτητές να κατανοήσουν τις βασικές αρχές λειτουργίας ενός κατανεμημένου συστήματος.

**Απαιτήσεις Εργασίας**

**Frontend (React Native)**

* Δημιουργία εφαρμογής με React Native.
* Υλοποίηση των εξής λειτουργιών:
  + **Εγγραφή και Σύνδεση Χρήστη**:
    - Ο χρήστης πρέπει να μπορεί να εγγραφεί με email και password.
    - Η σύνδεση πρέπει να χρησιμοποιεί authentication token (JWT).
  + **Προβολή Λίστας Εστιατορίων**:
    - Εμφάνιση λίστας διαθέσιμων εστιατορίων από το backend.
    - Δυνατότητα αναζήτησης εστιατορίων με βάση την τοποθεσία ή το όνομα.
  + **Φόρμα Κράτησης**:
    - Επιλογή ημερομηνίας, ώρας και αριθμού ατόμων.
    - Αποστολή της κράτησης στο backend.
  + **Προφίλ Χρήστη**:
    - Προβολή ιστορικού κρατήσεων του χρήστη.
    - Διαγραφή ή τροποποίηση μελλοντικών κρατήσεων.

**Backend (Node.js & Express)**

* Δημιουργία REST API που να περιλαμβάνει:
  + **Endpoints**:
    - /register: Δημιουργία νέου χρήστη.
    - /login: Επιστροφή authentication token.
    - /restaurants: Επιστροφή λίστας εστιατορίων.
    - /reservations: Δημιουργία, τροποποίηση, ή διαγραφή κράτησης.
    - /user/reservations: Επιστροφή κρατήσεων συγκεκριμένου χρήστη.
  + **Middleware**:
    - Χρήση JWT για έλεγχο ταυτότητας.
  + **Σύνδεση με Βάση Δεδομένων**:
    - Επικοινωνία με MariaDB για ανάκτηση και αποθήκευση δεδομένων.

**Database (MariaDB)**

* Δημιουργία βάσης δεδομένων με τα εξής (ενδεικτικά):
  + **Πίνακας Χρηστών**:
    - user\_id, name, email, password (hashed).
  + **Πίνακας Εστιατορίων**:
    - restaurant\_id, name, location, description.
  + **Πίνακας Κρατήσεων**:
    - reservation\_id, user\_id, restaurant\_id, date, time, people\_count.

**Οδηγίες Υλοποίησης**

**Εργαλεία**

* **IDE**: Προτείνεται το **NetBeans** λόγω της ευκολίας του στην ολοκληρωμένη ανάπτυξη εφαρμογών με Node.js (μέσω plugins) και MariaDB.
* **Γλώσσες**: JavaScript (React Native & Node.js), SQL.
* **Πλατφόρμες**:
  + React Native για το frontend.
  + Node.js για το backend.
  + MariaDB για τη βάση δεδομένων.
* **Λοιπά Εργαλεία**:
  + Postman για δοκιμή των REST API endpoints.
  + GitHub για αποθετήριο και version control.

**Κριτήρια Αξιολόγησης**

**Υλοποίηση (70%)**

1. **Frontend (30%)**
   * **UI/UX**: Καθαρό και λειτουργικό περιβάλλον χρήστη.
     + 10%: Οπτική ομοιομορφία και χρήση βασικών κανόνων σχεδιασμού UI/UX.
   * **Λειτουργικότητα**:
     + 15%: Σωστή επικοινωνία με το backend για εγγραφή/σύνδεση, προβολή εστιατορίων και διαχείριση κρατήσεων.
     + 5%: Υποστήριξη ανατροφοδότησης χρήστη (π.χ., επιτυχία ή αποτυχία αιτημάτων).
2. **Backend (20%)**
   * **Αρχιτεκτονική**:
     + 10%: Ορθή δομή κώδικα, καθαρός διαχωρισμός routes, controllers και services.
   * **API Endpoints**:
     + 5%: Υποστήριξη ασφαλείας (JWT).
     + 5%: Ομαλή διασύνδεση με τη βάση δεδομένων.
3. **Database (20%)**
   * **Δομή**:
     + 10%: Καλά σχεδιασμένοι πίνακες με σωστές σχέσεις (primary/foreign keys).
   * **Ερωτήματα**:
     + 10%: Αποτελεσματική εκτέλεση ερωτημάτων CRUD.

**Παρουσίαση (30%)**

1. **Περιεχόμενο** (20%):
   * Σαφής περιγραφή στόχου της εφαρμογής (5%).
   * Παρουσίαση της αρχιτεκτονικής (5%).
   * Δείγματα χρήσης (screenshots ή live demo) (10%).
2. **Δεξιότητες Παρουσίασης** (10%):
   * Χρήση οπτικών βοηθημάτων (π.χ., PowerPoint) (5%).
   * Οργανωμένη και επαγγελματική παρουσίαση (5%).

**Παραδοτέα**

1. **Κώδικας**:
   * Ανεβασμένος σε αποθετήριο GitHub με README που περιλαμβάνει:
     + Οδηγίες εγκατάστασης.
     + Περιγραφή λειτουργικότητας.
2. **Παρουσίαση PowerPoint**:
   * Μέγιστος αριθμός διαφανειών: 12.
   * Μέγιστος χρόνος παρουσίασης: 10 λεπτά.

**Χρονοδιάγραμμα**

1. **Εβδομάδα 1-2**: Ανάλυση απαιτήσεων και σχεδιασμός βάσης δεδομένων.
2. **Εβδομάδα 3-7**: Ανάπτυξη backend και βασικών API endpoints.
3. **Εβδομάδα 8-11**: Δημιουργία frontend και διασύνδεση με backend.
4. **Εβδομάδα 12**: Παράδοση τελικού έργου και παρουσίασης.

RUBRIC:

|  |  |  |
| --- | --- | --- |
| **Κατηγορία** | **Κριτήριο Αξιολόγησης** | **Βαθμοί** |
| **Υλοποίηση (70%)** | | |
| **Frontend (30%)** | UI/UX σχεδιασμός, καθαρότητα διεπαφής, διαχείριση καταστάσεων και επικοινωνία με το backend | 10 |
| Σωστή λειτουργία αυθεντικοποίησης, κράτησης και διαχείρισης δεδομένων | 15 |
| Υποστήριξη μηνυμάτων επιβεβαίωσης και διαχείρισης λαθών | 5 |
| **Backend (20%)** | Ορθή δομή κώδικα, καθαρός διαχωρισμός routes, controllers και services | 10 |
| Ασφάλεια (JWT authentication, hashing passwords) και σωστή διαχείριση αιτημάτων | 5 |
| Ομαλή διασύνδεση με τη βάση δεδομένων | 5 |
| **Database (20%)** | Σωστή δομή πινάκων, κανονικοποίηση και σχέσεις μεταξύ των οντοτήτων | 10 |
| Αποτελεσματικά SQL queries, indexing και βελτιστοποίηση απόδοσης | 10 |
| **Παρουσίαση (30%)** | | |
| **Περιεχόμενο (20%)** | Σαφής περιγραφή των λειτουργιών της εφαρμογής | 5 |
| Ανάλυση αρχιτεκτονικής (frontend, backend, database) | 5 |
| Δείγματα χρήσης, παραδείγματα ή live demo | 10 |
| **Δεξιότητες Παρουσίασης (10%)** | Χρήση οπτικών βοηθημάτων (π.χ. PowerPoint) | 5 |
| Οργανωμένη και επαγγελματική παρουσίαση | 5 |
| **Σύνολο** | **Μέγιστη Βαθμολογία** | **100** |