

Μάθημα: Προηγμένος Προγραμματισμός Full Stack JavaScript Frameworks Εξάμηνο: 1^ο Ατομική Απαλλακτική εργασία 8 μονάδων

Θέμα:

Ο Δήμαρχος μιας πόλης, πλούσιας σε ιστορία και αξιοθέατα, αποφάσισε να εισέλθει στον κόσμο του Έξυπνου Τουρισμού. Με στόχο να προσφέρει στους κατοίκους και τους επισκέπτες μια ολοκληρωμένη εμπειρία, ανακοίνωσε την ανάπτυξη μιας ιστοσελίδας, μιας mobile εφαρμογής και μιας desktop εφαρμογής που θα επιτρέπει στους χρήστες (τουρίστες ή κατοίκους) να συμβάλλουν στον έξυπνο τουρισμό, αποθηκεύοντας σε μια βάση δεδομένων τα αγαπημένα τους μέρη και παρέχοντας πληροφορίες για τοπικά αξιοθέατα.

Στα πλαίσια αυτού του φιλόδοξου σχεδίου, καλούμαστε να αναπτύξουμε ένα API, το οποίο θα υποστηρίζει τις βασικές λειτουργίες διαχείρισης δεδομένων (Create, Read, Update, Delete - CRUD). Το API αυτό θα διασφαλίζει την ομαλή λειτουργία του εκάστοτε συστήματος, επιτρέποντας τη διαχείριση των χρηστών και των σημείων ενδιαφέροντος (POI).

Λειτουργίες του API:

▪ Εγγραφή Χρήστη:

Ένας νέος χρήστης εγγράφεται στο σύστημα, καταχωρώντας το όνομά του, την ηλικία και το email του. Το API ελέγχει αν υπάρχει ήδη χρήστης με το ίδιο email και αν ο χρήστης είναι κάτω των 18 ετών. Ο νέος χρήστης αποθηκεύεται στο collection "Users" της βάσης.

▪ Προσθήκη Σημείου Ενδιαφέροντος (POI):

Υποστηρίζεται η αποθήκευση ενός νέου σημείου ενδιαφέροντος στη βάση, όπως ένα μουσείο, επιλέγοντας το όνομα, την κατηγορία, τη διεύθυνση και προαιρετικά μια φωτογραφία. Το POI αποθηκεύεται στο collection "POIs" της βάσης.

- **Ανάγνωση Πληροφοριών Χρήστη:**

Ο χρήστης του API μπορεί να διαβάσει τις πληροφορίες ενός χρήστη ή όλων. Με το API, μπορεί να αναζητήσει συγκεκριμένο χρήστη βάσει του email του και να βλέπει μόνο το όνομα του.

- **Ανακάλυψη Σημείων Ενδιαφέροντος (POI):**

Οι χρήστες μπορούν να δουν όλα τα σημεία ενδιαφέροντος στην πόλη. Με το API, μπορούν να αναζητήσουν ένα συγκεκριμένο POI βάσει παραμέτρων όπως η κατηγορία.

- **Ενημέρωση Πληροφοριών Χρήστη/ Σημείων ενδιαφέροντος**

- **Διαγραφή Χρήστη ή POI:**

Υποστηρίζεται η διαγραφή ενός χρήστη ή ενός POI

Πιο αναλυτικά, για την παρούσα εργασία, σας ανατίθεται η υλοποίηση ενός API που θα διαχειρίζεται δύο διαφορετικά collections σε μια MongoDB βάση δεδομένων.

Collections:

1. Χρήστες (Users):

- Πεδία: Όνομα, Ηλικία, Email (υποχρεωτικά)
- Το email αποθηκεύεται με μικρά γράμματα στη βάση, ανεξάρτητα από την είσοδο του χρήστη.
- Έλεγχος για υπάρχον email κατά τη δημιουργία, με επιστροφή κατάλληλου status σε περίπτωση διπλοεγγραφής.
- Έλεγχος ηλικίας <18 με **χρήση middleware**.

2. Σημεία Ενδιαφέροντος (Points of Interest - POIs):

- Πεδία: Όνομα, Κατηγορία, Διεύθυνση, Γεωγραφικές Συντεταγμένες
- Φωτογραφία (προαιρετικό)
- Καμία περιοριστική απαίτηση ως προς τα πεδία.

Λειτουργίες API:

1. Create:

- Χρήστες: Αποθήκευση νέου χρήστη στη βάση με επιστροφή κατάλληλου status.
- POI: Αποθήκευση νέου σημείου ενδιαφέροντος με επιστροφή κατάλληλου status.

2. Read:

- Χρήστες:
 - Όλων των εγγράφων των χρηστών.

- Ενός χρήστη βάσει του email του (εμφάνιση μόνο του ονόματος).
- POI:
 - Όλων των σημείων ενδιαφέροντος.
 - Ενός σημείου ενδιαφέροντος βάσει παραμέτρου στο request.
- 3. **Update:**
 - Ενημέρωση ενός χρήστη βάσει του email του.
 - Ενημέρωση ενός σημείου ενδιαφέροντος βάσει του id του.
- 4. **Delete:**
 - Διαγραφή ενός χρήστη βάσει του email του.
 - Διαγραφή ενός σημείου ενδιαφέροντος βάσει του id του.

Σημείωση: Πέρα από τα υποχρεωτικά πεδία, μπορείτε να προσθέσετε ότι άλλο κρίνετε χρήσιμο για να αποθηκεύεται στη βάση στο κάθε collections.

Επιπλέον:

- Για τη χρήση του παρόντος API, **δεν** απαιτείται αυθεντικοποίηση (authentication)
- **Δεν** απαιτείται να δημιουργηθεί front end.
- Για τη βάση καλείστε να χρησιμοποιήσετε την **MongoDB και τη mongoose**
- Το project σας θα πρέπει να είναι οργανωμένο σε **MVC αρχιτεκτονική**. Μπορείτε να επιλέξετε ή τον τρόπο που είδαμε στα μαθήματα, ή οποιονδήποτε άλλο.
- Η εφαρμογή θα πρέπει να περιέχει **ένα package.json** αρχείο. Εκεί θα ορίζεται από που γίνεται η εκτέλεση της εφαρμογής. Στο terminal εκτελώντας την εντολή **npm start** θα πρέπει να γίνεται εκκίνηση της εφαρμογής.
- Να χρησιμοποιούνται **μεταβλητές περιβάλλοντος** στην εφαρμογή σας.
- Μπορείτε να αποστείλετε το postman collection (προαιρετικό)
- Μαζί με τον κώδικα καλείστε να παραδώσετε και ένα **έγγραφο** στο οποίο θα περιλαμβάνονται ως απαντήσεις στα παραπάνω ζητούμενα, τα requests που κάνετε, και τα responses που λαμβάνετε.

Σημαντικό:

- ✓ Σε περίπτωση που δεν καταφέρετε να υλοποιήσετε ολόκληρη την εργασία, μπορείτε να στείλετε μέρος της (το οποίο βέβαια να είναι λειτουργικό) και θα βαθμολογηθείτε για αυτό!
- ✓ Το παραδοτέο σας θα είναι ένα αρχείο .zip το οποίο θα περιλαμβάνει τον κώδικα, το έγγραφο και ότι άλλο κρίνετε απαραίτητο (πχ postman collection) το οποίο πρέπει να το ανεβάσετε στο χώρο των εργασιών του e-class εντός της προθεσμίας.

Καλή επιτυχία!