**Report Εφαρμογής Δανεισμού Βιβλίων**

**Εξώφυλλο**

**Τίτλος:** Εφαρμογή Δανεισμού Βιβλίων

**Συγγραφέας:** Σταμούλης Σπυρίδων 03775, Μαργαριτης Πανταζης

**Ημερομηνία:** 17 Ιανουαρίου 2025

**Περιγραφή:**

Η παρούσα εργασία αφορά την ανάπτυξη μιας εφαρμογής διαχείρισης βιβλιοθήκης με χρήση Node.js, Express.js και PostgreSQL. Η εφαρμογή επιτρέπει στους χρήστες να δανείζονται βιβλία, να προσθέτουν κριτικές και στους διαχειριστές να διαχειρίζονται τα διαθέσιμα βιβλία και τα δεδομένα δανεισμού.

**Περιεχόμενα**

1. Εισαγωγή
2. Αρχιτεκτονική της Εφαρμογής
3. Βάση Δεδομένων
4. Λειτουργικότητα Εφαρμογής
   * 4.1 Εγγραφή Χρήστη
   * 4.2 Σύνδεση Χρήστη
   * 4.3 Προβολή Βιβλίων
   * 4.4 Προσθήκη Κριτικής
   * 4.5 Προβολή Κριτικής
   * 4.6 Δανεισμός Βιβλίων (Διαχειριστής)
   * 4.7 Επιστροφή Βιβλίων (Διαχειριστής)
5. Ελεγχος σφαλματων χρηστη
6. Βιβλιογραφία
7. Λίστα Αρχείων
8. Οδηγίες Εγκατάστασης και Χρήσης

**Λίστα Αρχείων**

* **server.js:** Περιέχει τον κώδικα του διακομιστή Node.js και τη λογική της εφαρμογής.
* **index.html:** Η αρχική σελίδα της εφαρμογής, με φόρμες εγγραφής και σύνδεσης.
* **book-lending.html:** Σελίδα προβολής διαθέσιμων βιβλίων.
* **book-details.html:** Σελίδα με λεπτομέρειες βιβλίου και κριτικές.
* **admin.html:** Σελίδα διαχείρισης για διαχειριστές (Καταγραφή δανεισμών και επιστροφων και λιστα με δανεισμενα βιβλια).
* **style.css:** Αρχείο CSS με στυλ για την εμφάνιση της εφαρμογής.

**1. Εισαγωγή**

Η εφαρμογή διαχείρισης βιβλιοθήκης που αναπτύχθηκε, αυτοματοποιεί τον δανεισμό βιβλίων και ενημερώνει τους χρήστες για τη διαθεσιμότητά τους. Προσφέρει λειτουργίες εγγραφής/σύνδεσης, προβολής βιβλίων, προσθήκης κριτικών και διαχείρισης από διαχειριστές. Αναπτύχθηκε με Node.js, Express.js και PostgreSQL, εξασφαλίζοντας αξιοπιστία και επεκτασιμότητα. Στα επόμενα κεφάλαια, θα αναλυθεί η αρχιτεκτονική, η βάση δεδομένων, οι λειτουργίες και οι οδηγίες χρήσης.

**2. Αρχιτεκτονική της Εφαρμογής**

Η εφαρμογή βασίζεται στην αρχιτεκτονική client-server. Ο client (πρόγραμμα περιήγησης) επικοινωνεί με τον server (Node.js) μέσω HTTP requests. Ο server διαχειρίζεται τη λογική της εφαρμογής και αλληλεπιδρά με τη βάση δεδομένων PostgreSQL.

**3. Βάση Δεδομένων**

Η βάση δεδομένων PostgreSQL χρησιμοποιείται για την αποθήκευση των δεδομένων της εφαρμογής. Περιλαμβάνει τους πίνακες:

* **Users:** Αποθηκεύει τα στοιχεία των χρηστών (UserID, Username, Password, IsAdmin).
* **Books:** Αποθηκεύει τα στοιχεία των βιβλίων (bookId, Title, Author).
* **Reviews:** Αποθηκεύει τις κριτικές των χρηστών για τα βιβλία (ReviewID, bookId, UserID, ReviewText).
* **LentBooks:** Αποθηκεύει τα δεδομένα δανεισμού (LentBookID, bookId, UserID, LentAt).

**4. Λειτουργικότητα Εφαρμογής**

**4.1 Εγγραφή Χρήστη**

Οι χρήστες μπορούν να εγγραφούν στην εφαρμογή παρέχοντας όνομα χρήστη και κωδικό πρόσβασης.

**4.2 Σύνδεση Χρήστη**

Οι χρήστες μπορούν να συνδεθούν στην εφαρμογή με τα στοιχεία που δημιούργησαν κατά την εγγραφή.

**4.3 Προβολή Βιβλίων**

Οι χρήστες μπορούν να δουν τη λίστα με τα διαθέσιμα βιβλία.

**4.4 Προσθήκη Κριτικής**

Οι χρήστες μπορούν να προσθέσουν κριτικές για τα βιβλία που έχουν διαβάσει.

**4.5 Δανεισμός Βιβλίων (Διαχειριστής)**

Οι διαχειριστές μπορούν να σημειώσουν ένα βιβλίο ως δανεισμένο, επιλέγοντας το βιβλίο και τον χρήστη που το δανείζεται.

**4.6 Επιστροφή Βιβλίων (Διαχειριστής)**

Οι διαχειριστές μπορούν να σημειώσουν ένα βιβλίο ως επιστραμμένο.

**4.7 Διαγραφή Βιβλίων (Διαχειριστής)**

Οι διαχειριστές μπορούν να διαγράψουν βιβλία από τη βάση δεδομένων.

**5. Οδηγίες Εγκατάστασης και Χρήσης**

1. **Εγκατάσταση PostgreSQL:** Εγκαταστήστε τον PostgreSQL και δημιουργήστε μια βάση δεδομένων με το όνομα "mydb".
2. **Εκτέλεση SQL script:** Εκτελέστε το SQL script για τη δημιουργία των πινάκων στη βάση δεδομένων.
3. **Εγκατάσταση Node.js:** Εγκαταστήστε το Node.js και τις απαραίτητες βιβλιοθήκες (express, pg, bcrypt).
4. **Ρύθμιση διακομιστή:** Ρυθμίστε τα στοιχεία σύνδεσης με τη βάση δεδομένων στο αρχείο server.js.
5. **Εκκίνηση διακομιστή:** Εκτελέστε την εντολή node server.js για να ξεκινήσει ο διακομιστής.
6. **Πρόσβαση στην εφαρμογή:** Ανοίξτε ένα πρόγραμμα περιήγησης και πλοηγηθείτε στη διεύθυνση http://localhost:3000.

**6. Επικοινωνια με χρηστη**

Εντάξει, ας δούμε μια πρόταση που επεξηγεί το κάθε μήνυμα που εμφανίζεται στην εφαρμογή:

* **Επιτυχής εγγραφή:**
  + "User registered successfully": Ο χρήστης εγγράφηκε με επιτυχία.
* **Σφάλματα κατά την εγγραφή:**
  + "Username and password are required": Πρέπει να συμπληρώσετε και τα πεδία "Όνομα χρήστη" και "Κωδικός πρόσβασης".
  + "Registration failed: [συγκεκριμένο μήνυμα σφάλματος από τον server]": Η εγγραφή απέτυχε λόγω κάποιου προβλήματος, με συγκεκριμένη επεξήγηση από τον server.
  + "Failed to register. Please try again later.": Η εγγραφή απέτυχε, δοκιμάστε ξανά αργότερα (Κάποιο πρόβλημα με την σύνδεση λογικά).
  + “Username already exists”: O χρήστης είναι ήδη εγγεγραμμένος.
* **Επιτυχής σύνδεση:**
  + "Login successful": Η σύνδεση πραγματοποιήθηκε με επιτυχία.
* **Σφάλματα κατά τη σύνδεση:**
  + "Username and password are required": Πρέπει να συμπληρώσετε και τα πεδία "Όνομα χρήστη" και "Κωδικός πρόσβασης".
  + "User not found": Δεν βρέθηκε χρήστης με αυτό το όνομα χρήστη.
  + "Invalid username or password": Το όνομα χρήστη ή ο κωδικός πρόσβασης είναι λάθος.
  + "Login failed. Please try again later.": Η σύνδεση απέτυχε, δοκιμάστε ξανά αργότερα.
* **Επιτυχής προσθήκη βιβλίου:**
  + "Book added successfully": Το βιβλίο προστέθηκε με επιτυχία.
* **Σφάλματα κατά την προσθήκη βιβλίου:**
  + "Title and author are required": Πρέπει να συμπληρώσετε και τα πεδία "Τίτλος" και "Συγγραφέας".
  + "Failed to add book": Η προσθήκη του βιβλίου απέτυχε.
  + "Error adding book: [συγκεκριμένο μήνυμα σφάλματος]": Η προσθήκη του βιβλίου απέτυχε λόγω κάποιου προβλήματος, με συγκεκριμένη επεξήγηση.
* **Σφάλματα κατά την προβολή λεπτομερειών βιβλίου:**
  + "Error fetching book details. Please try again later.": Αποτυχία ανάκτησης των λεπτομερειών του βιβλίου. Δοκιμάστε ξανά αργότερα.
  + "Error fetching book reviews. Please try again later.": Αποτυχία ανάκτησης των κριτικών του βιβλίου. Δοκιμάστε ξανά αργότερα.
* **Επιτυχής προσθήκη κριτικής:**
  + "Review added successfully": Η κριτική προστέθηκε με επιτυχία.
* **Σφάλματα κατά την προσθήκη κριτικής:**
  + "Review text is required": Πρέπει να γράψετε κείμενο για την κριτική.
  + "[συγκεκριμένο μήνυμα σφάλματος] or 'Failed to add review'": Η προσθήκη της κριτικής απέτυχε λόγω κάποιου προβλήματος.
  + "An error occurred while adding the review.": Παρουσιάστηκε σφάλμα κατά την προσθήκη της κριτικής.
* **Σφάλματα κατά την ανάκτηση δανεισμένων βιβλίων:**
  + "Error fetching lent books. Please try again later.": Αποτυχία ανάκτησης της λίστας με τα δανεισμένα βιβλία. Δοκιμάστε ξανά αργότερα.
* **Επιτυχής μαρκάρισμα βιβλίου ως δανεισμένου:**
  + "Book marked as lent": Το βιβλίο μαρκαρίστηκε ως δανεισμένο.
* **Σφάλματα κατά το μαρκάρισμα βιβλίου ως δανεισμένου:**
  + "Book not found": Δεν βρέθηκε βιβλίο με αυτό το ID.
  + "User not found": Δεν βρέθηκε χρήστης με αυτό το όνομα.
  + "Book is already lent out": Το βιβλίο είναι ήδη δανεισμένο.
  + "Failed to mark book as lent": Αποτυχία μαρκαρίσματος του βιβλίου ως δανεισμένου.
  + "An error occurred while marking the book as lent.": Παρουσιάστηκε σφάλμα κατά το μαρκάρισμα του βιβλίου ως δανεισμένου.
* **Επιτυχής μαρκάρισμα βιβλίου ως επιστραμμένου:**
  + "Book marked as returned": Το βιβλίο μαρκαρίστηκε ως επιστραμμένο.
* **Σφάλματα κατά το μαρκάρισμα βιβλίου ως επιστραμμένου:**
  + "Failed to mark book as returned": Αποτυχία μαρκαρίσματος του βιβλίου ως επιστραμμένου.

**7. Βιβλιογραφία**

* Node.js documentation: [https://nodejs.org/en/docs/](https://www.google.com/url?sa=E&source=gmail&q=https://nodejs.org/en/docs/)
* Express.js documentation: [https://expressjs.com/](https://www.google.com/url?sa=E&source=gmail&q=https://expressjs.com/)
* PostgreSQL documentation: [https://www.postgresql.org/docs/](https://www.google.com/url?sa=E&source=gmail&q=https://www.postgresql.org/docs/)
* We used AI for help connecting postgre with the app
* **Οδηγίες Εγκατάστασης και Εκτέλεσης**

Για την εγκατάσταση και εκτέλεση της εφαρμογής ακολουθήστε τα παρακάτω βήματα:

**1. Εγκατάσταση PostgreSQL:**

* Κατεβάστε την τελευταία έκδοση του PostgreSQL από την επίσημη ιστοσελίδα: [https://www.postgresql.org/download/](https://www.google.com/url?sa=E&source=gmail&q=https://www.google.com/url?sa=E%26source=gmail%26q=https://www.postgresql.org/download/" \t "_blank)
* Ακολουθήστε τις οδηγίες του installer για να ολοκληρώσετε την εγκατάσταση.
* Κατά την εγκατάσταση, ορίστε έναν κωδικό πρόσβασης για τον χρήστη "postgres" και **σημειώστε τον**.

**2. Δημιουργία Βάσης Δεδομένων:**

* Ανοίξτε το psql πληκτρολογώντας psql -U postgres στο τερματικό.
* Εισάγετε τον κωδικό πρόσβασης που ορίσατε.
* Δημιουργήστε μια νέα βάση δεδομένων με όνομα "mydb" εκτελώντας την εντολή:

*CREATE DATABASE mydb;*

* Συνδεθείτε στη νέα βάση δεδομένων:

*\c mydb*

* Εκτελέστε το SQL script που παρέχεται για τη δημιουργία των πινάκων (schema.sql).

**3. Εγκατάσταση Node.js modules:**

* Ανοίξτε ένα τερματικό στον φάκελο του project.
* Εκτελέστε την εντολή «npm install express pg bcrypt» για να εγκαταστήσετε τις απαραίτητες βιβλιοθήκες.
* Σε περίπτωση που λείπει κάποιο άλλο πακέτο μπορεί να γίνει εγκατάσταση με npm install …

**4. Εκτέλεση της εφαρμογής:**

* Βεβαιωθείτε ότι τα στοιχεία σύνδεσης με τη βάση δεδομένων στο αρχείο server.js είναι σωστά.
* Εκτελέστε την εντολή node server.js για να ξεκινήσει ο server (στο μονοπάτι του project).
* Ανοίξτε ένα πρόγραμμα περιήγησης και πλοηγηθείτε στη διεύθυνση http://localhost:3000.

**Σημείωση:** Βεβαιωθείτε ότι έχετε εγκατεστημένο το Node.js και το npm (Node Package Manager) στον υπολογιστή σας. **Επίσης στο αρχείο server.js ο χρήστης είναι σωσμένος ως postgres, με κωδικό: 1234, database name: mydb και port: 5432.** Μπορείτε να τα αλλάξετε όπως επιθυμείτε.