Database Documentation for the Library Management System

Agenda:

1) DB Related Design Decisions.
2) DB Setup Steps.

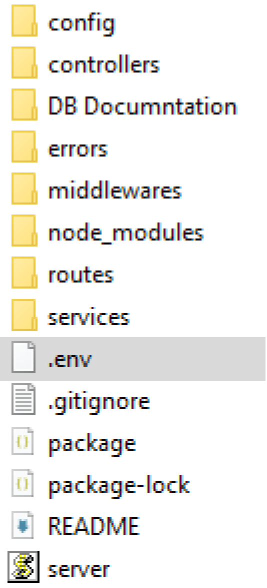
1. DB Design Decisions:

- The DB schema is composed of 4 tables, each for a certain purpose:

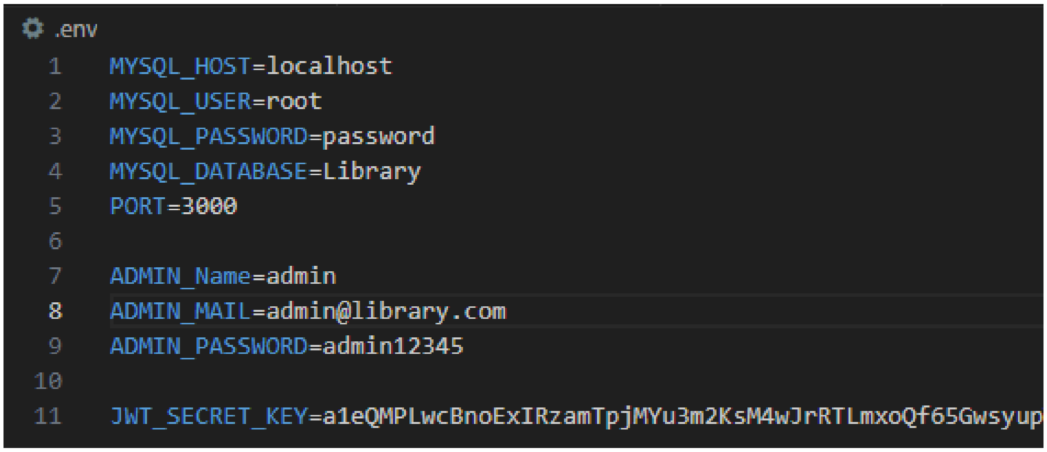
Table	Purpose
User	Stores necessary data related to all types of system users, including admin, librarians, and borrowers.
Book	Stores necessary data related to all books in the library.
Borrow	Stores necessary data related to the borrowing process.
passkey	Stores pass-keys generated during borrowing / returning books.

- The *role* field stored in User table is used in creating JWT which is used in most of requests to grant access to the API user according to the role.
- The name attribute of User is split into 2 atomic fields *first_name* and *last_name* (1NF).
- Passwords are hashed and stored in User table to provide security in case of DB leackage.
- The *Last_login_at* field stored in User table is used in creating JWT together with user email and role. This is to guarantee the uniqueness of JWT generated by the Backend server for each logged-in user.
- The shelf-location attribute of Book is split into 3 atomic fields *section*, *bay_number*, and *shelf_number* (1NF).
- The *return_date* field stored in borrow table is used to signal if the borrower has returned the book or not.
- Though the relation between user and passkey is one-to-one but it is split into 2 tables as the passkey is generated for borrowers only (2NF).
- Index on book author is created to speed up queries specifying only author field, as the composite index on title and author will not be useful in this case, since it requires the title column to also be part of the query.
- All Indices on time-related fields are created in DESC manner as the user would be more interested in newer data.

2. DB Setup Steps:
   - Just Run the associated setupScript.sql
   - The system depends on the presence of an admin user who is authorized to do specific tasks such as registering librarians, …
     - There is no need to insert a record for this admin user while setup.
     - Just attach an .env file in the application source code as shown below and the admin user record would be inserted once the server starts:
       - Make sure the admin name has no digits, just letters, spaces and hyphes. Otherwise, the admin record would be not updatable.
       - Make sure to update The admin password once the server starts for security issues.



config
controllers
DB Documntation
errors
middlewares
node_modules
routes
services
.env
.gitignore
package
package-lock
README
server



.env

```
MYSQL_HOST=localhost
MYSQL_USER=root
MYSQL_PASSWORD=password
MYSQL_DATABASE=Library
PORT=3000

ADMIN_Name=admin
ADMIN_MAIL=admin@library.com
ADMIN_PASSWORD=admin12345

JWT_SECRET_KEY=a1eQMPLwcBnoExIRzamTpjMYu3m2KsM4wJrRTLmxoQf65Gwsyup
```