

AdvDatabaseProject_CRUD

Author:	Alberto Orlando Mafla Prado
Student Number:	10147804
Module Title:	Advanced Database System
Module Code:	QHO541
Tutors Name:	Dr Ajmal Gharib

GitHub Repository [A0M0305/AdvDatabaseProject_CRUD](https://github.com/A0M0305/AdvDatabaseProject_CRUD)

Introduction

As part of my job as system administrator I have a responsibility of the daily management of our Cinema software "Vista". As I administrate the different level of databases one of my struggles is the management of a rational database which requires separating the data. Which in term requires information to be stored in different areas of the program. This can be time consuming as requires lots of opening, closing different windows to manage a user or item.

To solve this issue, I want to create a much simpler program that allows me to access and view all information as a User Management system minimising the closing of windows.

System Overview

Backend

Dependencies

```
"dependencies": {  
  "axios": "^0.26.1",  
  "bcrypt": "^5.0.1",  
  "body-parser": "^1.19.2",  
  "cors": "^2.8.5",  
  "dotenv": "^16.0.0",  
  "ejs": "^3.1.6",  
  "express": "^4.17.3",  
  "express-basic-auth": "^1.2.1",  
  "jsonwebtoken": "^8.5.1",  
  "mongoose": "^6.2.7",  
  "morgan": "^1.10.0",  
  "nodemon": "^2.0.15"  
}
```

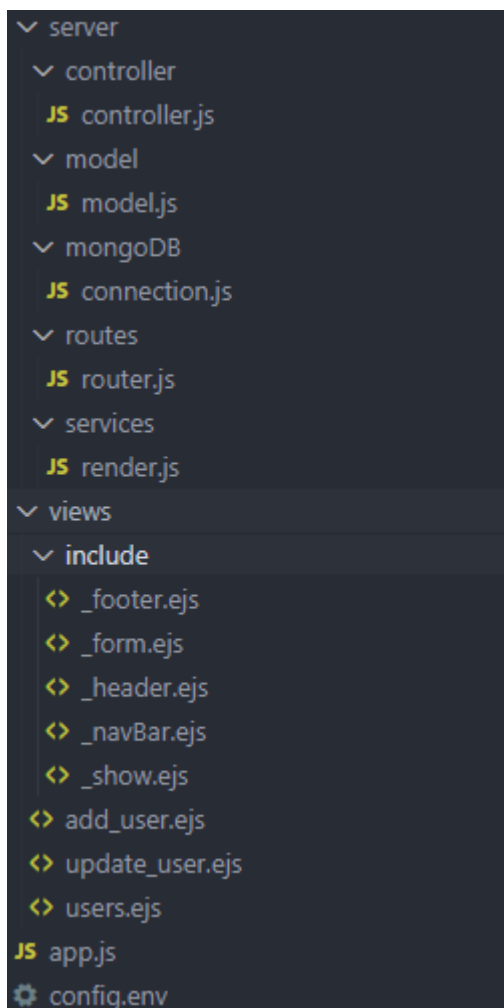
<>

- nodeJs
- express
- mongoose - allows me to access my MongoDB

- morgan - to help log messages when we request
- Nodemon - Helps to restart the server when we make changes to the project (saves from pressing refresh)
- ejs - the template engine I used to develop the project via HTML and allows for JavaScript
- body-parser - this allows us to serialise the data and access form data using body properties
- dot env - this module allows me to separate the "secret" from the source code. This is for future development of the project where I can have other people collaborating with me but not sharing passcodes, just the code.
- A module I came across from research that allows easier request access to the library in express applications

Project Structure

- Using the MVC (model, view, controller) pattern design
- This helps separate the application data and the business logic from the views.
- The controller mediates between the model and view
 - Model folder contains the MongoDB data. This will perform the data validation, process data, create mongo scheme
 - Controller folder will deal with the user request for resources from the server.



How to run the project

- Type on terminal "npm start".
- This will kick start nodemon and you will have a link to the localhost page 5033.

- press ctrl and rightclick to open up the webpage

Front End

Key Design Decisions

Database Design

Security and Scalability

Conclusions