**Aksel Cruses Mini Project 312323247009**

**Bill Order Management System**

The Bill Order Management System is a web-based application designed to manage the lifecycle of bill orders. It includes features for creating, viewing, editing, approving, issuing, and checking orders. The system is role-based, with different functionalities available for Directors, Managers, Store personnel, and Checkers.

**Key Components**

**Frontend**

- Technology: HTML, CSS, JavaScript

|  |  |
| --- | --- |
| Page | Description |
| index.html | Login page for different roles (Director, Manager, Store, Checker). |
| home.html | Home page for Directors to create and view orders. |
| newOrder.html | Form for creating new orders. |
| viewOrder.html | List of all orders with options to view, edit, or delete. |
| editOrder.html | Form for editing existing orders. |
| managerView.html | List of orders for Managers to approve or reject. |
| managerViewDetails.html | Detailed view of an order for Managers to approve or reject. |
| storeView.html | List of orders for Store personnel to issue or mark as out of stock. |
| storeViewDetails.html | Detailed view of an order for Store personnel. |
| checkView.html | List of orders for Checkers to verify. |
| checkViewDetails.html | Detailed view of an order for Checkers to verify. |

**Backend**

- File: server.js

- Technology: Node.js with Express.js

- Database: MySQL

**-** API Endpoints: to fetch data’s

**Scripts:**

- newOrder.js: Handles the creation of new orders.

- editOrder.js: Handles editing of existing orders.

- viewOrder.js: Displays a list of all orders.

- managerView.js: Displays orders for Managers to approve or reject.

- storeView.js: Displays orders for Store personnel to issue or mark as out of stock.

- checkView.js: Displays orders for Checkers to verify.

- viewOrderDetails.js, managerViewDetails.js, storeViewDetails.js, checkViewDetails.js: Handle detailed views for respective roles.

**Roles and Credentials**

|  |  |
| --- | --- |
| Role | Description |
| Director | Create, view, edit, and delete orders. |
| Manager | Approve or reject orders. |
| Store | Issue orders or mark them as out of stock. |
| Checker | Verify orders and mark missing items. |

**Features**

1. Login System

- Role-based login with predefined credentials.

- Redirects users to their respective dashboards.

2. Order Management

- Create Orders: Directors can create new orders with detailed information.

- View Orders: All roles can view orders relevant to their responsibilities.

- Edit Orders: Directors can edit existing orders.

- Delete Orders: Directors can delete orders.

3. Approval Workflow

- Managers can approve or reject orders with reasons.

- Store personnel can issue orders or mark them as out of stock.

- Checkers can verify orders and mark missing items.

4. Export and Print

- Export order details to Excel files.

- Print order details directly from the browser.

**Setup Instructions**

1. Prerequisites

* Node.js installed on your system.
* MySQL database server installed and running.

2. Installation

1. Clone the repository.
2. Navigate to the project directory.
3. Install dependencies:

**npm install**

3. Database Setup

1. Import the [billorderdb.sql](vscode-file://vscode-app/c:/Users/aksel/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html) file into your MySQL server:

**mysql -u [username] -p [database\_name] < billorderdb.sql**

1. Update the database connection details in [server.js](vscode-file://vscode-app/c:/Users/aksel/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html):

**const db = mysql.createConnection({**

**host: 'your-database-host',**

**user: 'your-database-username',**

**password: 'your-database-password',**

**database: 'billorderdb',**

**});**

4. Run the Server

Start the server:

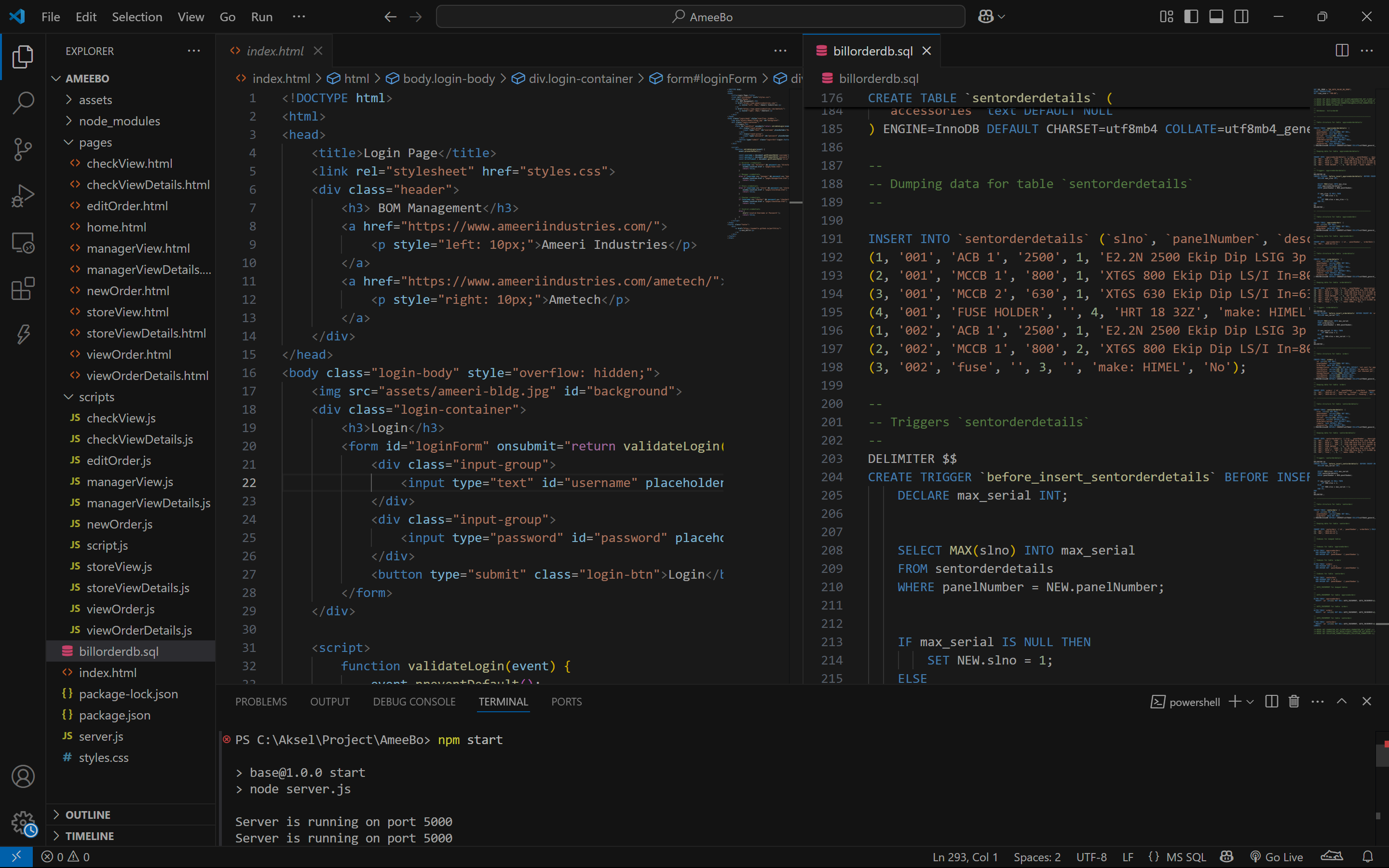
**npm start**

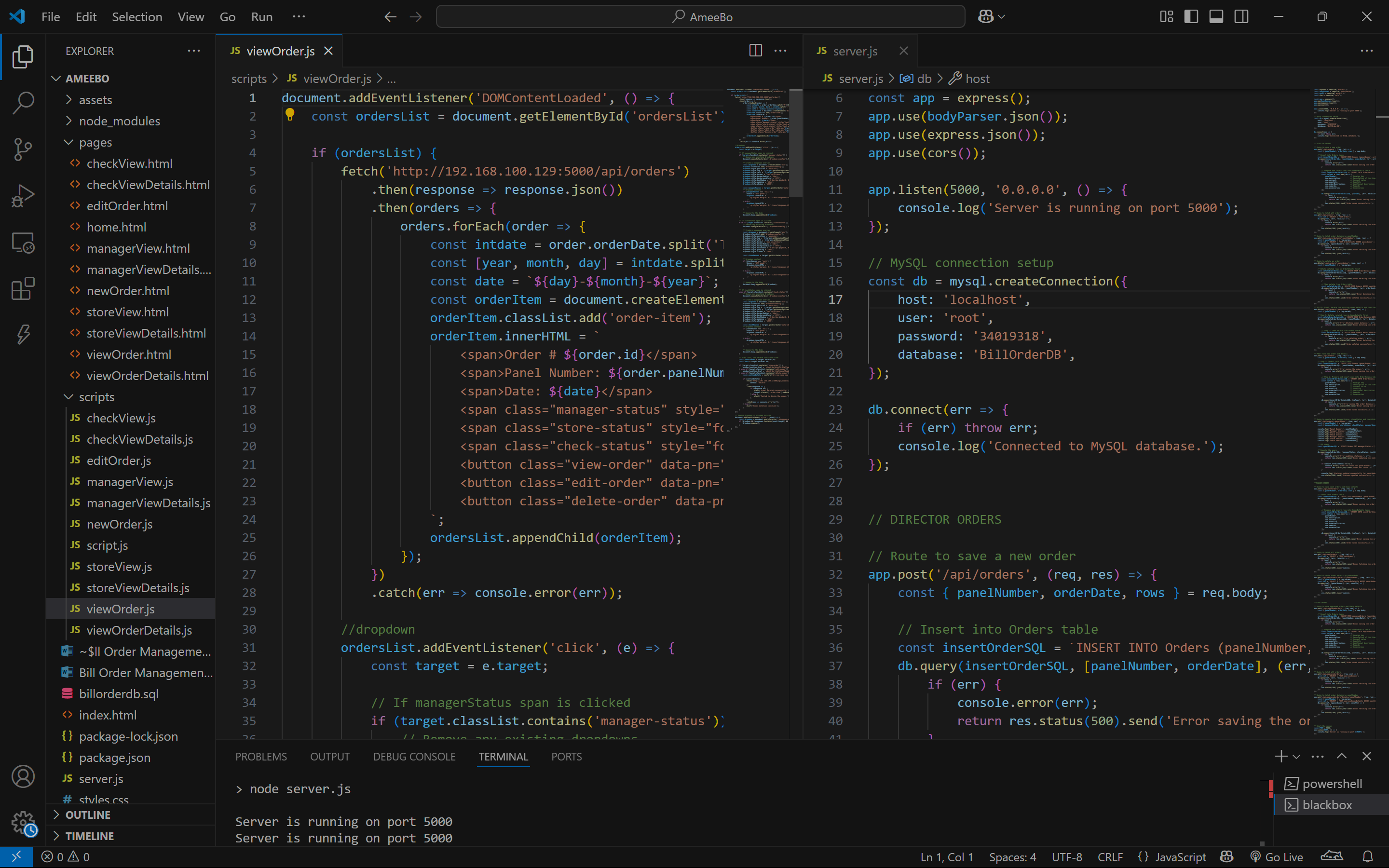
The server will run on http://localhost:5000.

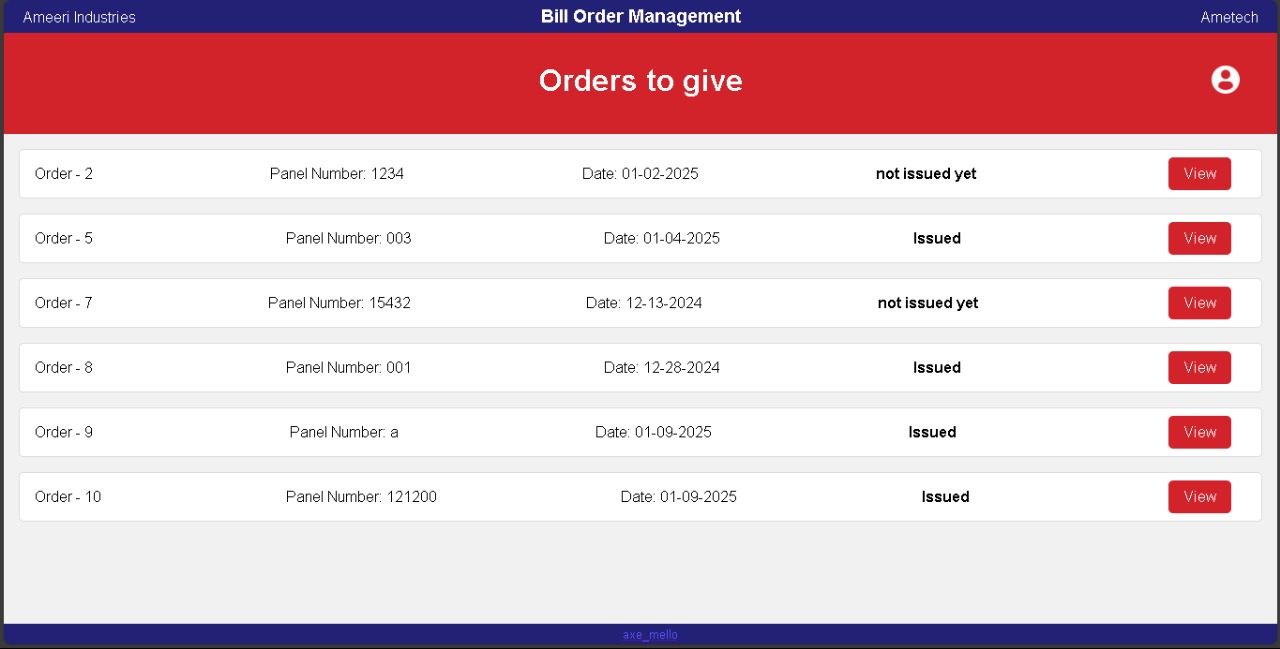
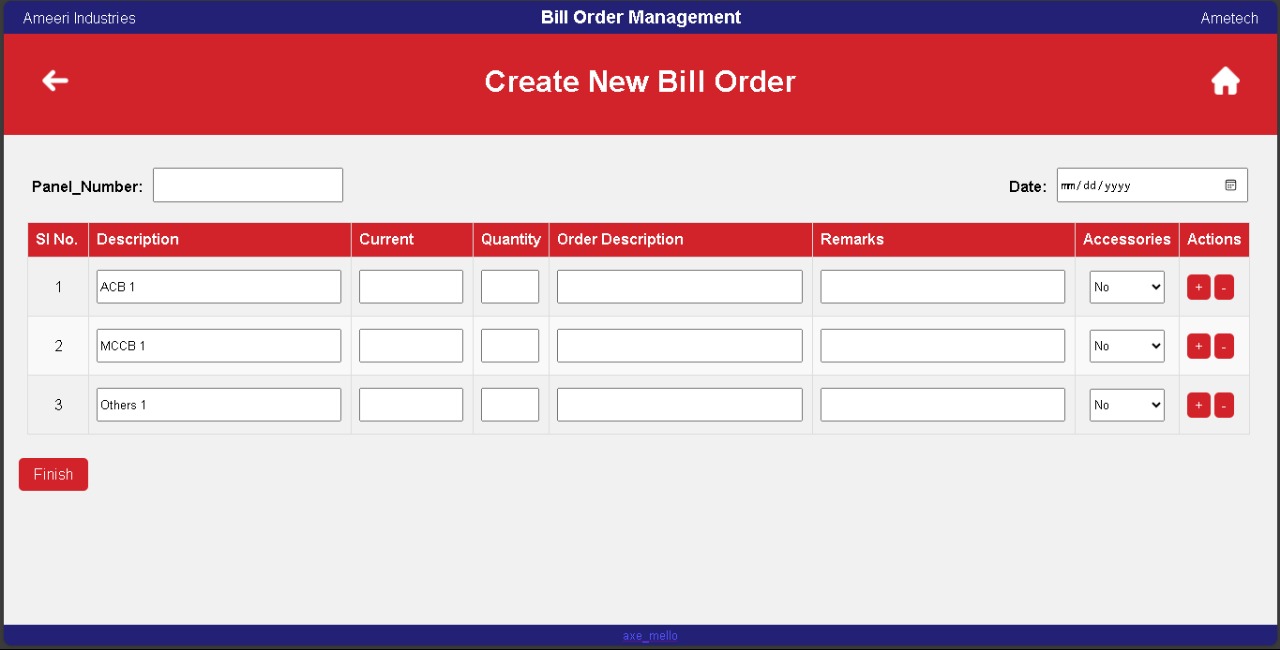
**5. Access the Application**

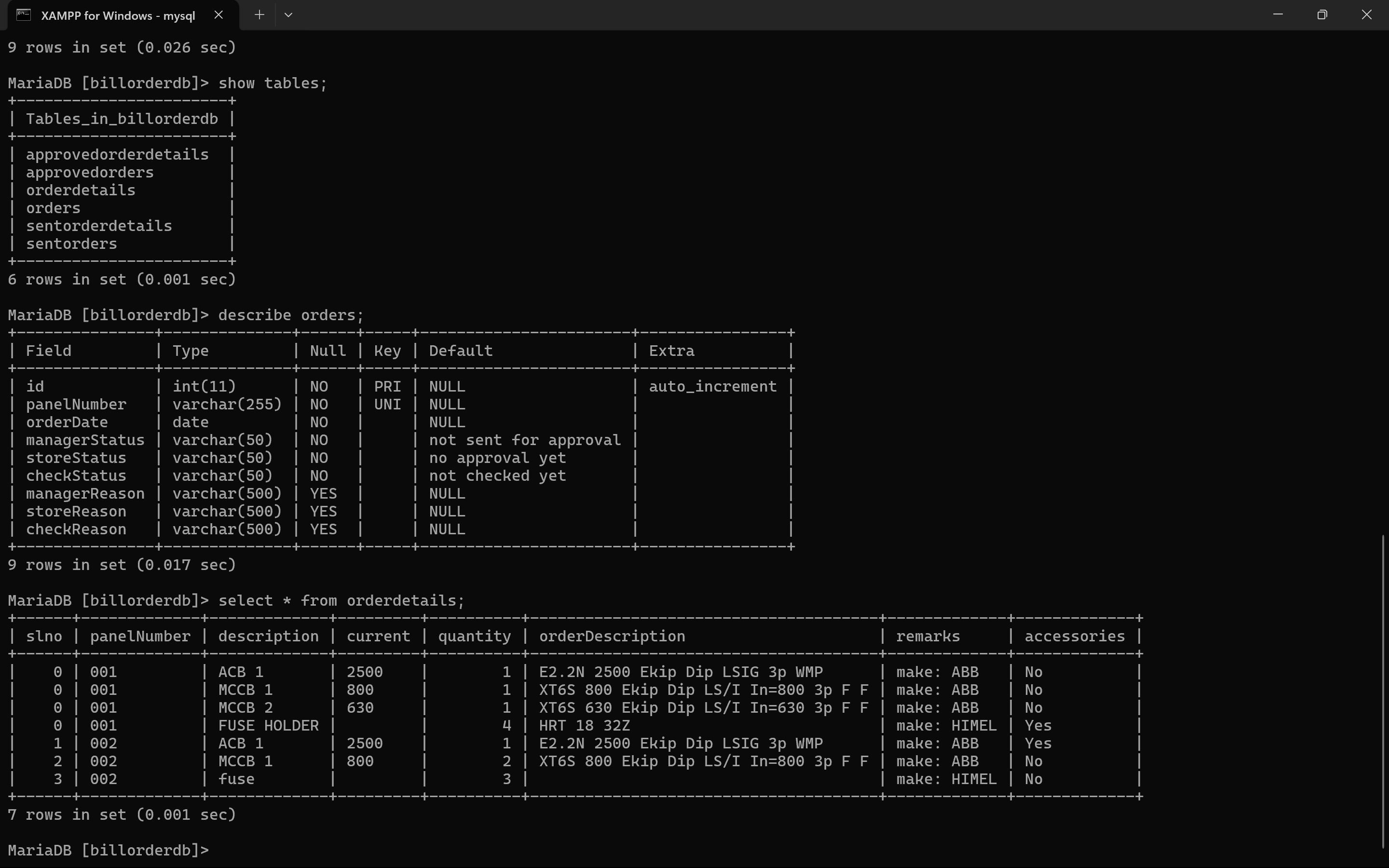
Open [index.html](vscode-file://vscode-app/c:/Users/aksel/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html) in your browser to access the login page.

**Screenshots:**

****

****

****

****

**Conclusion**

The Bill Order Management System is a robust, role-based web application designed to streamline the management of orders across various roles, including Directors, Managers, Store personnel, and Checkers. The project integrates a Node.js backend with a MySQL database and a responsive frontend built with HTML, CSS, and JavaScript.