**Course Number: Course Number: 420-DW4-AS**

**Course Title: Web Server Applications Development II**

**Session: Autumn 2024**

**Group: 7278**

FINAL PROJECT REPORT

# Easy Property Rental

2024/12/04

SUBMITTED

TO

QUANG HOANG CAO

BY

AIDAN MONK - 202233954

**I. PROJECT DESCRIPTION**

Easy Property Rental (what I chose to name my project) is an ASP.net Core application created to view and manage rental properties on a user friendly website. The program is usable and designed for property owners, property managers, and existing and potential tenants. I used a Database First design approach for this project.

**II. PROJECT DEVELOPMENT**

**PHASE I ANALYSIS**

1. **Read the project’s requirements very carefully; make sure you understand them properly.**

* The requirements were analyzed to understand the scope of features required for property owners, property managers, and tenants to interact with the system.
* Features such as CRUD operations for properties, tenant management, apartment booking, and notification systems were identified.

1. **Functional Requirements Document**

* **Property Owner**: Manage buildings, view tenants, and approve property manager requests.
* **Property Manager**: Manage apartments, assign tenants to buildings, and handle requests.
* **Tenant**: View apartments, make requests, and communicate with property managers.

**PHASE II DESIGN**

* Designed the application using ASP.NET Core MVC, Entity Framework for database management, and built a user-friendly interface for different roles (Property Owner, Property Manager, and Tenant).
* Database schema was structured to support relationships between buildings, apartments, tenants, and property managers.
* Designed for responsiveness to be usable on both desktop and mobile devices.

**PHASE III IMPLEMENTATION**

* Implemented the project using ASP.NET Core MVC with:
* Controllers and views for managing buildings, apartments, tenants, and users.
* Authentication and role-based authorization for Property Owners, Property Managers, and Tenants.
* Real-time data management using Entity Framework and SQL Server.
* Developed responsive and interactive UIs for managing data.
* Ensured secure login and session management for users based on roles.

**PHASE IV TESTING THE PROGRAM**

Hints: In Table Format

|  |  |  |
| --- | --- | --- |
| User | Functional Requirement | Test Result / problems |
| Property Owner | Perform CRUD operations for property managers | Success |
| Property Owner | Perform CRUD operations for Buildings | Success |
| Property Owner | Perform CRUD operations for Apartments | Success |
| Property Owner | Login, Registration of accounts | Success |
| Property Owner | Messages | Fail, issues encountered due to database design |
| Property Manager | Perform CRUD operations for Tenants | Success |
| Property Manager | Perform CRUD operations for Buildings | Success |
| Property Manager | Perform CRUD operations for Apartments | Success |
| Property Manager | Login, Registration of accounts | Success |
| Property Manager | Messages | Fail, issues encountered due to database design |
| Tenant | View Apartment info | Success |
| Tenant | Messages | Fail, issues encountered due to database design |
| Tenant | Appointments | Fail |
| Tenant | Login, Registration of accounts | Success |
| All Users | View Available Apartments | Success |
| All Users | Filter Apartments with search fields | Success |
| All Users | View Apartment details | Success |
| All Users | Website design, navigation between pages | Success |

**III. CONCLUSION**

Through the testing and debugging I endured in creating this project, aside from gaining experience in developing asp.net web applications, I learned the importance of prudent planning and designing **before** starting work on the project. My initial database structure was not sufficient to accomplish all of the tasks outlined in this project, and as a result the quality and time it took to develop it suffered tremendously. Next time I will surely spend more time in the planning phase.

**IV. BIBLIOGRAPHY**

StackOverflow, Youtube, chatgpt for design questions