Course Number: 420-DW4-AS

Course Title: Web Server Applications Development II

Session: Winter 2024

Group: 7270

FP Rental Management

Final Project Report

SUBMITTED TO

Teacher: QUANG HOANG CAO

BY

FRANCISCO RICARDO ANDRASCHKO - 2235113

2024-04-25

1. Project Description

FPRM (FP Rental Management) is a property rental management system developed using ASP.NET Core MVC technology. This project aims to simplify and optimize the rental management process for landlords, managers and tenants.

With an intuitive and friendly interface, FPRM offers several functionalities to meet the specific needs of each type of user:

- For Administrators/Owners:
  - Complete user management, including addition, editing, and deletion.
  - Control over all buildings, apartments, and rentals registered in the system.
  - Management of messages exchanged between users.
- For Managers:
  - Management of buildings and apartments available for rent.
  - Review and approval of pending appointment requests.
  - Efficient communication through internal messaging.
- For Tenants:
  - Viewing of buildings and apartments available for rent.
  - Scheduling visits to desired properties.
  - Tracking of their own appointments and communication with managers.

FPRM is designed to provide a personalized and secure experience for all users, with built-in authentication and access control capabilities.

This system is an indispensable tool to simplify real estate rental management, increase efficiency and improve the experience of both administrators and tenants.

2. Project Development

A) PHASE I - ANALYSIS

In this Phasys was necessary to read and understand all the customers requirements, then start the solution design. It is very important for the project progress and the final result. The requirements are:

Property Owner, Administrator:

- Create/Update/Delete/Search/List any property manager account
- Update/Delete/Search/List any potential tenant account
- Full control of the Web Site

Property Manager

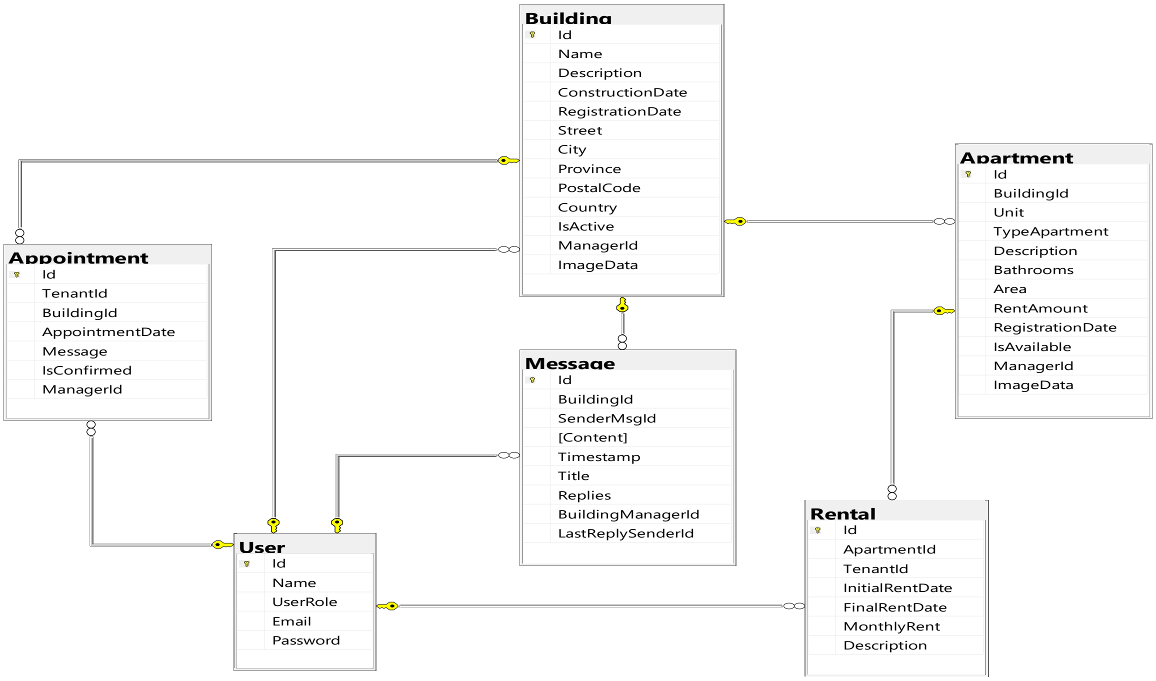
- Perform CRUD operations related to properties
- Perform CRUD operations related to apartments
- Keep track of apartments status
- Schedule potential tenants ‘appointments
- Respond to potential tenants ‘messages
- Report any event to the property owner when necessary

Potential Tenants

- Create an on-line account through Property Rental
- Management Web Site
- View any apartment suitable for their needs
- Make an appointment with the property manager
- Send necessary messages to the property manager

B) PHASE II - DESIGN

There are many different ways to start software design, but focusing on our common development, there are mainly two: Code First or Database First, but the end result needs to be the same.
In this case, Code First was used, which consists of designing the database using tools directly for the development environment. In the end, the result is the following diagram:



C) PHASE III - IMPLEMENTATION

Development Environment, Tools, Languages, Frameworks, among others, represents in most cases an exclusive option for the developer, but represents the present and future of the system.
With this in mind, the choice for this project was Asp.Net Core MVC, using Entity Framework, coding in C# and using Sql Server for the Database Server. Furthermore, for the front-end it is using HTML, CSS and JS.
This combo guarantees powerful software for developers, consistent, efficient, secure, scalable, adaptable and fully integrated with APIs and third-party tools, if necessary.
In the case of programming, the Code First approach ensures that business rules are followed at all times, as they are responsible for the structure. And even if the rules change during the coding process, this is not a problem, as the tools are prepared for changes, simply and quickly, keeping the code aligned with the business rules and ready for new changes at all times. This high performance keeps work moving forward, even in the face of head-on problems, always carrying out simple or complex routines in a robust manner.

D) PHASE IV (TESTING THE PROGRAM)

USER	FUNCTIONAL REQUIREMENT	RESULT/PROBLEMS
Administrator	Login	Successfully
Administrator	Create Users	Successfully
Administrator	Full Control(Irrestrict Access. Role: Administrator)	Successfully
Administrator	Received Messages Forwarded	Message was received, but the Reply is not Implemented.
Manager	Restrict Access (Role: Manager)	Successfully
Manager	CRUD operations in building (Restrict for your Management)	Successfully
Manager	CRUD operations in apartments (Restrict for your Management)	Image was including in the register, but the Image Collection presents problems and was disabled (keep just one image).
Manager	Receive and Accept/Reject Users Appointments (Change Status)	Successfully
Manager	Receive/Respond Messages	Successfully

Manager	Forward Messages/Events to Administrator	Successfully
Unlogged User	Open Application	Successfully
Unlogged User	Navigate to Active Buildings	Filter isActive is working, but it is not possible to hide the checkbox
Unlogged User	Navigate to Available Apartments	Filter isAvailable is working, but it is not possible to hide the checkbox
Unlogged User	Create User Tenant	Successfully
Tenant	Restrict Access (Role: Tenant)	Successfully
Tenant	List/View Active Buildings	Successfully
Tenant	List/View Available Apartments	Successfully
Tenant	Make Appointment	Successfully
Tenant	View Status Appointment	Successfully
Tenant	Send/Reply Message to Building Manager	Successfully
All Users	Search Building	Successfully
All Users	Search Apartments	Successfully
All Users	Logout	Successfully

Conclusion

During the IT Programmer course, many technologies were learned, but this semester, especially in this course, we are more immersed in the “real” technologies, in the language, in the tools and in the situations that we will find in the coding market, especially when using the Code First approach, even if the programmer is Full Stack, it is common for activities to be separated and in my opinion the DB is the first part that is separated.

Others important subjects we are using: Model-View-Controller (MVC), Dependency Injection, Middlewares, Pipelines, EF, Identity (mainly Roles and Claims) which improves my understanding of layers, integration and responsibilities. I really liked the course and started an extra course to improve my knowledge about Asp.Net and the new features and new architectures (SOLID, DDD, BLAZOR, among others).

Bibliography

Microsoft Documentation: https://docs.microsoft.com/

Microsoft Learn Asp.Net Core : https://learn.microsoft.com/en-us/aspnet/core/

Microsoft Learn LINQ: https://learn.microsoft.com/pt-br/dotnet/csharp/linq/

Eduardo Pires (Brazilian Microsoft MVP): https://www.desenvolvedor.io

Stack Overflow: https://stackoverflow.com/

Bootstrap: https://getbootstrap.com/

Html Templates: https://free-css.com/