PROPERTY RENALS

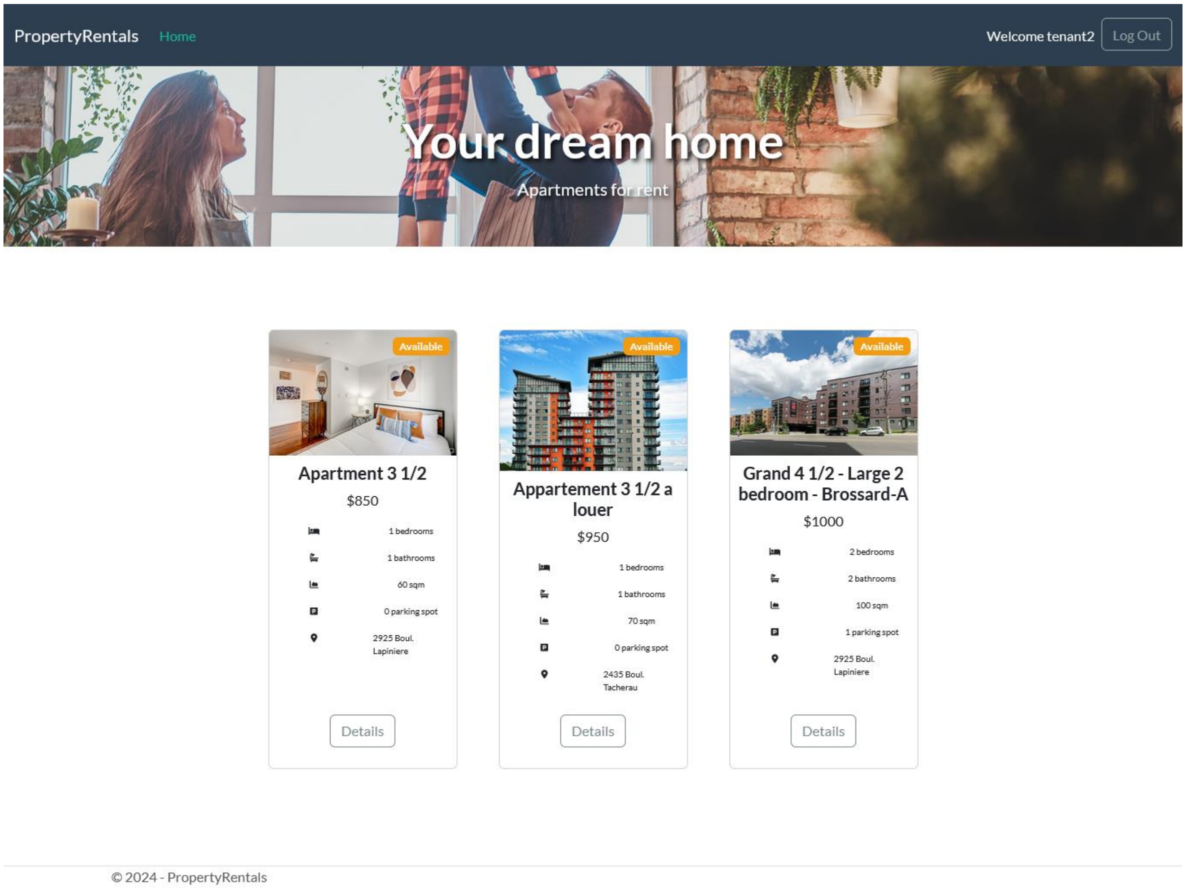
Course Number: 420-DW4-ASCourse Title: Web Server Applications Development II Session: Winter 2024Group: 7270

APRIL 25, 2024

ANDREA CAROLINA RUIZ DIAZ
Teacher: Quang Hoang Sao

PROJECT DESCRIPTION

Property Rental Management Web Site allows any potential tenant to search for an apartment suitable for his/her needs. The Web site also helps the property owner and the property manager to facilitate the management tasks of property rental.



Apartment 3 1/2
$850
1 bedrooms
1 bathrooms
60 sqm
0 parking spot
2925 Boul. Lapiniere

Appartement 3 1/2 a louer
$950
1 bedrooms
1 bathrooms
70 sqm
0 parking spot
2435 Boul. Tacherau

Grand 4 1/2 - Large 2 bedroom - Brossard-A
$1000
2 bedrooms
2 bathrooms
100 sqm
1 parking spot
2925 Boul. Lapiniere

© 2024 - PropertyRentals

PROJECT DEVELOPMENT

PHASE I ANALYSIS

Who are the users of this web app?: Tenants, Managers and Owner.

Property Owner:

- 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

List of tables

IDEAS	FINAL SOLUTION
Tenant	User
Buildings?	Tenant
Apartments	Manager
Appointments	Owner
Messages	Property
User	Apartments
Rentals?	Status
Employees	Appointments
Status? (rent)	Messages
Schedule?	Rentals
	Schedule
	Event
	Photos

PHASE II DESIGN

Relationships

Tables	Relationship
Users to Managers	One-to-One
Users to Tenants	One-to-One
Users to Owners	One-to-One
Users to Messages	One-to-Many
Users to Managers	One-to-Many
Users to Owners:	One-to-Many
Properties to Apartments	One-to-Many
Properties to Owners	Many-to-One
Properties to Managers	Many-to-One

Tenants to Rentals	One-to-many
Apartments to Rental	One-to-many
Appointments to Tenants	Many-to-One
Appointments to Managers	Many-to-One
Appointments to Apartments	Many-to-One
Appointments to Status	Many-to-One
Events to Apartments	Many-to-One
Events to Status	Many-to-One
Messages to Apartments	Many-to-One
Messages to Users	Many-to-One
Messages to Status	Many-to-One
Owners to Users	Many-to-One
Status to Apartments	One-to-Many
Photos to Apartments	Many-to-One

Required Types and Data Type

Table Status

Field	datatype	Design Notes	Example
StatusId (PK)	Number	Apartments (1. Available, 2.Rented, 3 Unavailable), Appointments(4 Pending,5. Confirmed, 6. Canceled)	1

Description	String	Apartments(Available, Unavalable, Rented), Appointments(Pending, Confirmed, Canceled)
		Events(7. New, 8. Pending,9 Solved).
		Messages(10. Read, 11. notRead)
			Available

Table User

Field	datatype	Design Notes	Example
UserId (PK)	int	auto generated	123456
Username	String	Unique	
password	String	Min 8 digit, numbers, letters, one caracter	Abcd3f8h
UserType	String	Tenant, Owner, Manager	Tenant

Table Owners

Field	datatype	Design Notes	Example
OwnerID(PK)	int	auto generated PK	1234
UserID	int	FK Users	123456
FirstName	String	Only string	Andrea
LastName	String	Only String	Ruiz
Email	String	Email type	asdf@mail.com
Phone	String	10 digit only numbers	4384658770

Table Managers

Field	datatype	Design Notes	Example

ManagerID(PK)	int	Autogenerated PK	1234
UserID	Number	FK Users	123456
FirstName	String	Only string	Andrea
LastName	String	Only String	Ruiz
Email	String	Email type	asdf@mail.com
Phone	Number	10 digit only numbers	4384658770

Table Tenant

Field	datatype	Design Notes	Example
TenantId(PK)	Number	Int auto generated PK	1234
UserID	Number	FK Users	123456
FirstName	String	Only string	Andrea
LastName	String	Only String	Ruiz
Email	String	Email type	asdf@mail.com
Phone	Number	10 digit only numbers	4384658770

Table Properties

Field	datatype	Design Notes	Example
PropertyCode(PK)	String	PK	BUL001
OwnerID	Number	FK Owners	1234
ManagerID	Number	FK Managers	1234
Name	String	6-floor	La Prairie Building
Description	String		6 floor building
Address	String		2323 Boul Teacherau
City	String		Brossard
State	String		Quebec
ZipCode	String	Zip code format	F5T6H6

Table Apartments

Field	datatype	Design Notes	Example
ApartmentId(PK)	int	PK , auto-generated	
ApartmentCode	String		601, A
PropertyCode	String	FK Properties	BUL001
Title	String		
Type	String	(3.5, 4.5, 5.5)	
Description	String		
Rent	Number		1200
StatusId		FK Status	1

Table Photos

Field	datatype	Design Notes	Example
PhotoId(PK)	Number	Autogenerated PK	12345678
ApartmentId	int	FK Apartments	1
PhotoLink	String		www.rental.com/photo1.jpg

Table Rentals

Field	datatype	Design Notes	Example
RentalId(PK)	Number	Autogenerated, PK	123456
ApartmentId	int	FK Apartments	601, A
TenantID	Number	FK Tenants	12345
RentalDate	Date		
EndContract	Date		

Table Appointments

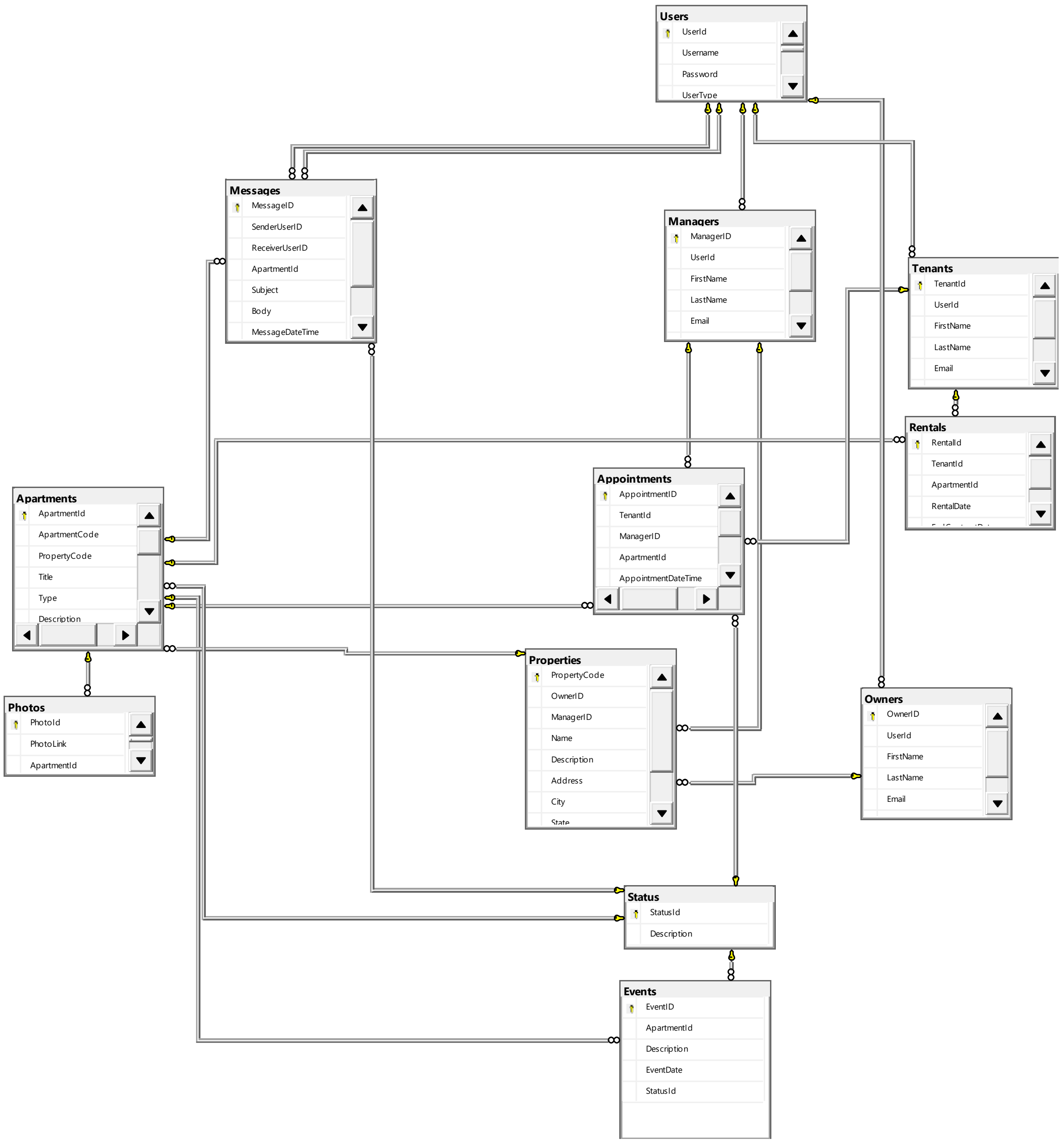
Field	datatype	Design Notes	Example
AppointmentID(PK)	Number	Autogenerated PK	12345678
TenantID	Number	FK Tenant	12345
ManagerID	Number	FK Mangers	1234
ApartmentiD		FK Apartments	
AppointmentDateTime	DateTime		
Description	String		
StatusId	number	(Pending, Confirmed, Canceled))	5

Table Messages

Field	datatype	Design Notes	Example
MessageID(PK)	Number	PK	1234567890
SenderUserID		FK User	
ReceiverUserID		FK User	
AppartmentId	Int	FK	
Subject	String		
Body	String		
MessageDateTime	DateTime		
StatusId			

Table Events

Field	datatype	Design Notes	Example
EventID(PK)	Number	Pk	1234567890
ApartmentID	int	FK	601, A
Description	String		
EventDate	Date		
StatusID	number	New o solved	8



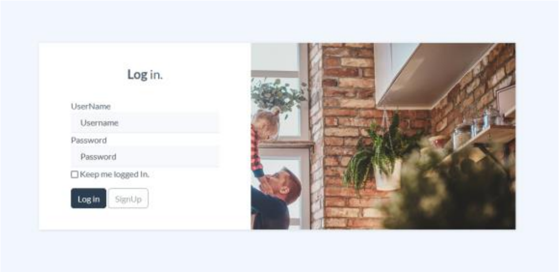
PHASE III IMPLEMENTATION

The implementation was made using ASP.Net core MVC for the server side, Bootstrap and JavaScript for client side.

1. Creation of the database on SQL Server Management Studio, and populate some initial data.
2. Creation of ASP.Net Core Web App MVC with Visual Studio 2022
3. Connect the Application to the Database using SQL Server authentication.

4. Instal the necessary packages: Microsoft Entity Framework Core, Sql Server, Tools and Thinktecture entity framework core relational.
5. Create the model from the database using the Scaffold-DbContext command in the Package Manager Console.
6. Move the connection string to the appsettings.json and edit the Program.cs to add the context.
7. Create the controllers and view necessary.
8. Start to develop the application:

Login Authentication/Authorization in ASP.NET CORE MVC



Log in.

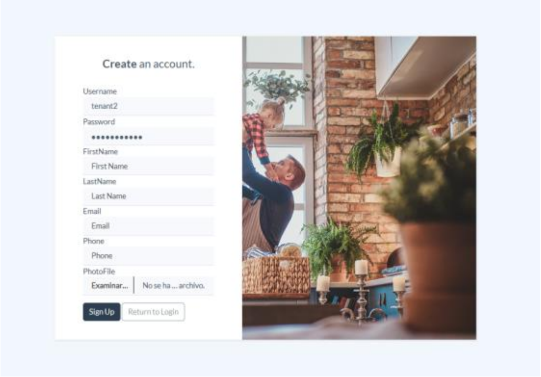
UserName

Username

Password

Password

□ Keep me logged In.



Create an account.

Username

tenant2

Password

••••••••••••

FirstName

First Name

LastName

Last Name

Email

Email

Phone

Phone

PhotoFile

Examinar... | No se ha ... archivo.

using Microsoft.AspNetCore.Authentication.Cookies;

```
builder.Services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationS
cheme)
    .AddCookie(option =>
    {
        option.LoginPath = "/Access/Login";
        option.ExpireTimeSpan = TimeSpan.FromMinutes(20);
    });


app.UseAuthentication();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Access}/{action=Login}/{id?}");
```

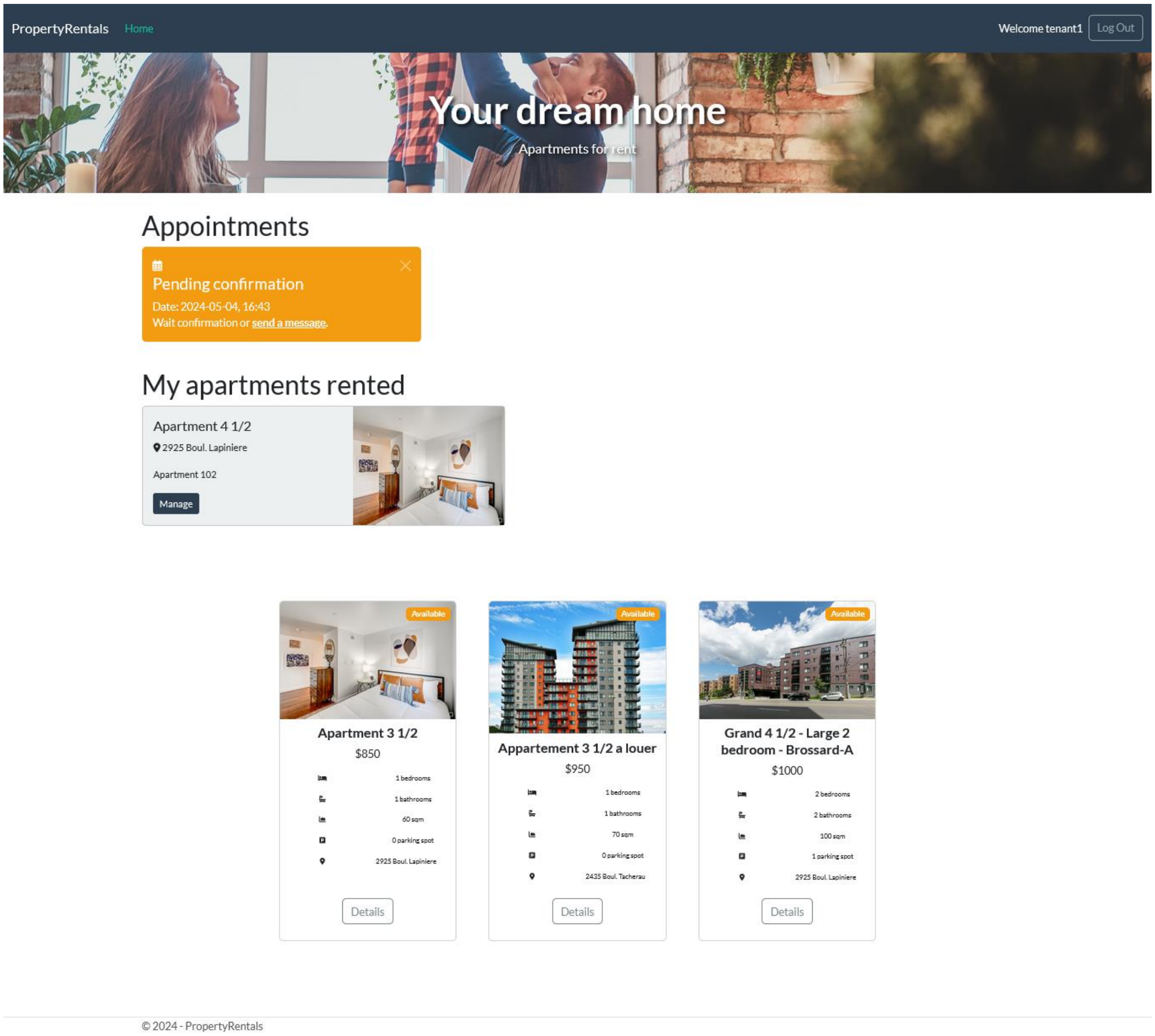
This code configures authentication using cookies in an ASP.NET Core application. It sets up cookie authentication with a login path of "/Access/Login" and an expiration time of 20 minutes. Then, it enables authentication in the application pipeline and sets a default route for MVC controllers, directing requests to the "Access" controller's "Login" action by default, with an optional "id" parameter.

The **Access controller** is responsible for handling user authentication-related actions in an ASP.NET Core application. It includes methods for user login and signup. The Login action checks if the user is already authenticated and redirects to the home page if so. The SignUp action validates user input for signup, including username, password, email, and phone number, and saves the new user to the database if all criteria are met. The Login action validates user credentials from the database and signs in the user using cookie-based authentication if the credentials are correct, storing user information in claims. It then redirects the user to the home page. If the login fails, appropriate error messages are displayed. The controller uses Entity Framework Core for database operations and includes basic input validation using regular expressions.

The **HomeController** is responsible for managing actions related to the home page of a property rental application. The controller includes an Index action, which retrieves user information, such as messages, appointments, and rented apartments, based on the user's role (Tenant, Manager, or Owner). It also handles authentication and authorization, ensuring that only authenticated users can access certain actions or views. Depending on the user's role, different data is retrieved and displayed on the home page. Additionally, the controller includes actions for privacy, logging out, and handling errors. The controller utilizes Entity Framework Core for database operations and integrates with ASP.NET Core's authentication and authorization features.

On the home is shown if the user has Messages not read, Appointments with their status, apartment events for owners and managers, apartments rented by the user tenant. And the list of apartments available to tenants or all the apartments with its status to Managers and Owners.

When user clicks on a Message, Appointment, event or apartment, is redirected to the Apartment details.



Your dream home

Apartments for rent

Appointments

Pending confirmation

Date: 2024-05-04, 16:43

Wait confirmation or send a message.

My apartments rented

Apartment 4 1/2

2925 Boul. Lapiniere

Apartment 102

Apartment 3 1/2

$850

1 bedrooms

1 bathrooms

60 sqm

0 parking spot

2925 Boul. Lapiniere

Appartement 3 1/2 a louer

$950

1 bedrooms

1 bathrooms

70 sqm

0 parking spot

2435 Boul. Tacherau

Grand 4 1/2 - Large 2 bedroom - Brossard-A

$1000

2 bedrooms

2 bathrooms

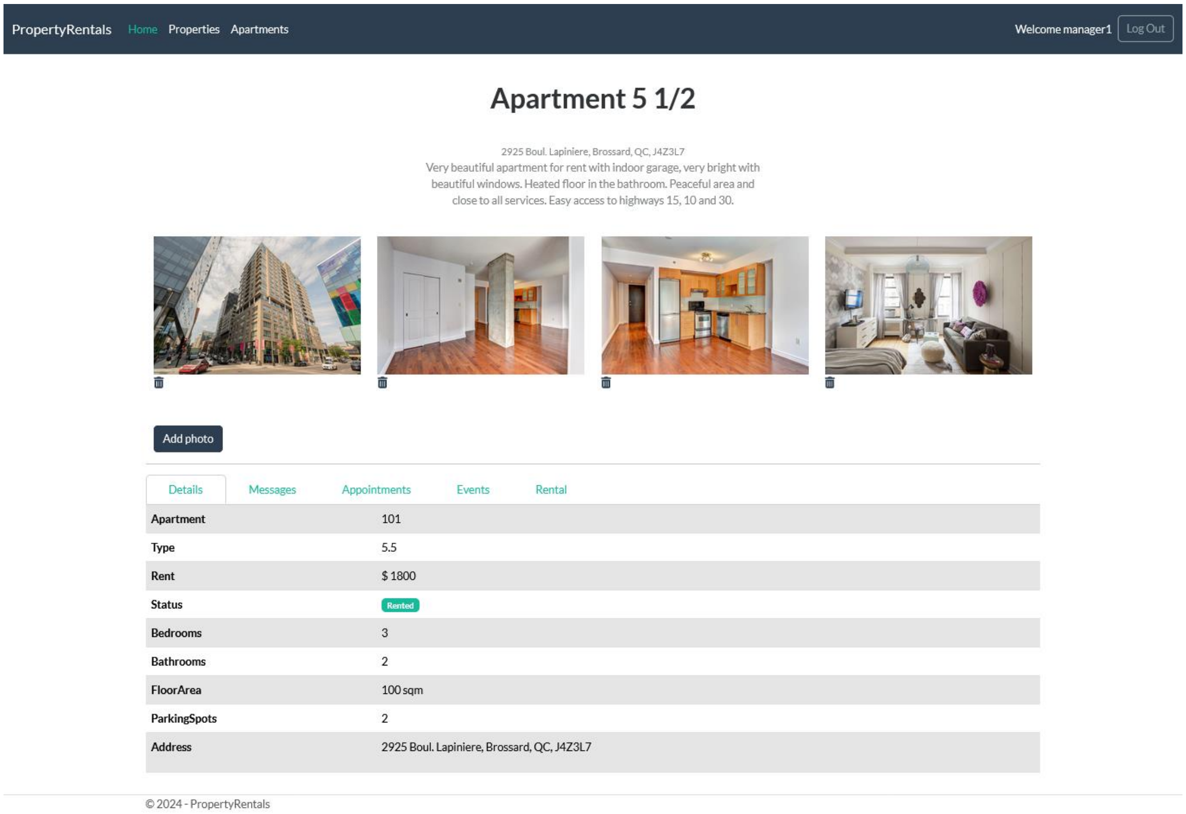
100 sqm

1 parking spot

2925 Boul. Lapiniere

© 2024 - PropertyRentals

The **ApartmentsController,** responsible for handling actions related to managing apartments in a property rental application. The controller is decorated with the [Authorize] attribute, ensuring that only authenticated users can access its actions. The controller includes actions for making an apartment available or unavailable, viewing apartment details, creating, editing, and deleting apartments. It also handles message retrieval and updates message status for tenants, as well as retrieves appointment details for managers and owners. Additionally, the controller utilizes Entity Framework Core for database operations and integrates with ASP.NET Core's authentication and authorization features.



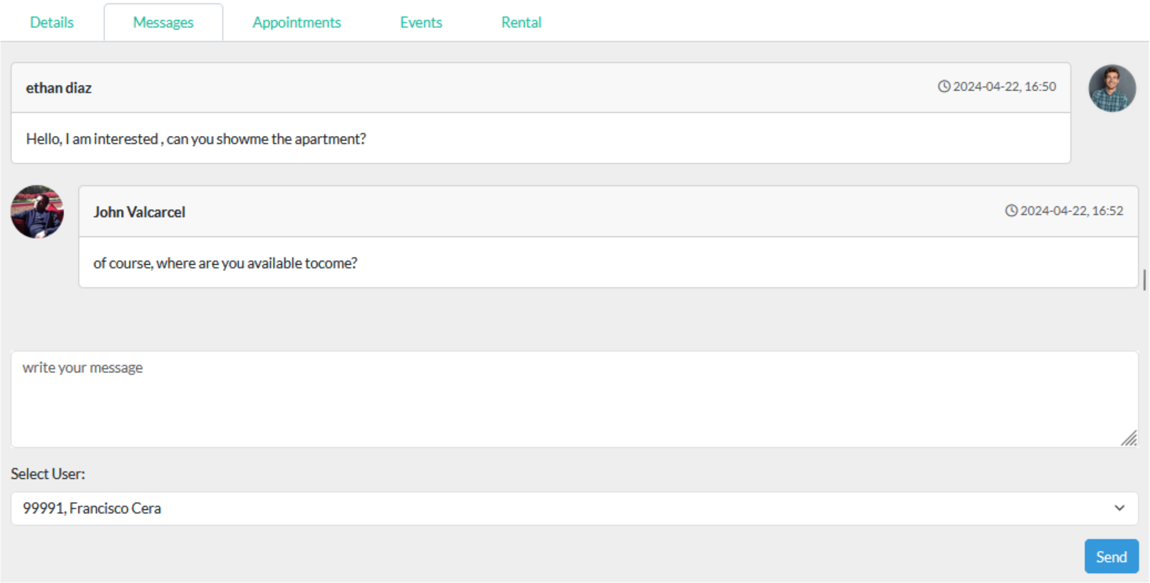
Apartment 5 1/2

2925 Boul. Lapiniere, Brossard, QC, J4Z3L7
Very beautiful apartment for rent with indoor garage, very bright with beautiful windows. Heated floor in the bathroom. Peaceful area and close to all services. Easy access to highways 15, 10 and 30.

Apartment	101
Type	5.5
Rent	$ 1800
Status	Rented
Bedrooms	3
Bathrooms	2
FloorArea	100 sqm
ParkingSpots	2
Address	2925 Boul. Lapiniere, Brossard, QC, J4Z3L7

© 2024 - PropertyRentals

The apartment details show the details of each apartment, and its photos that can be deleted or added by managers or owners, also some tabs show details, messages, appointments, events, and rentals depending on the user.



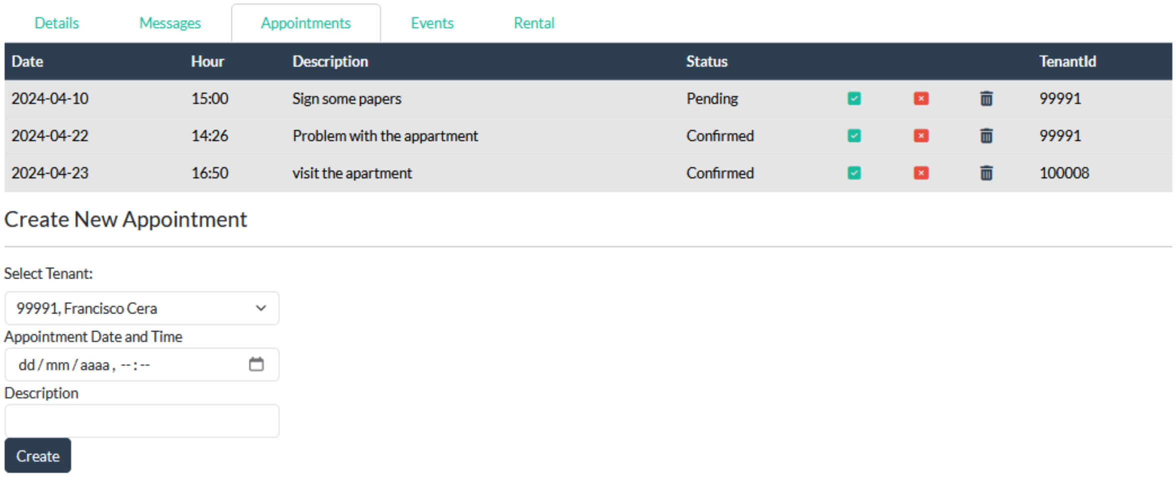
ethan diaz 2024-04-22, 16:50
Hello, I am interested , can you showme the apartment?

John Valcarcel 2024-04-22, 16:52
of course, where are you available tocome?

write your message

Select User:
99991, Francisco Cera

The messages are shown like a chat with the user photo and have the possibility of sending a message to the manager by default or to any Tenant if the user is logged as manager or owner.



Date	Hour	Description	Status				TenantId
2024-04-10	15:00	Sign some papers	Pending	☑	☒	🗑	99991
2024-04-22	14:26	Problem with the appartment	Confirmed	☑	☒	🗑	99991
2024-04-23	16:50	visit the apartment	Confirmed	☑	☒	🗑	100008

Create New Appointment

Select Tenant:

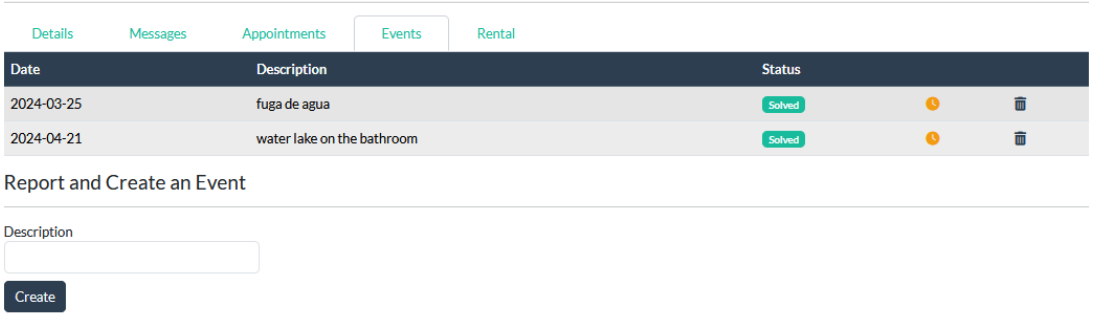
99991, Francisco Cera

Appointment Date and Time

dd / mm / aaaa , -- : --

Description

On the appointments tab are showed the appointments that the user have, with the possibility of confirming, cancel or delete the appointment. The user can also create a new appointment with the manager or the manager or owner can create a new appointment with a selected tenant..

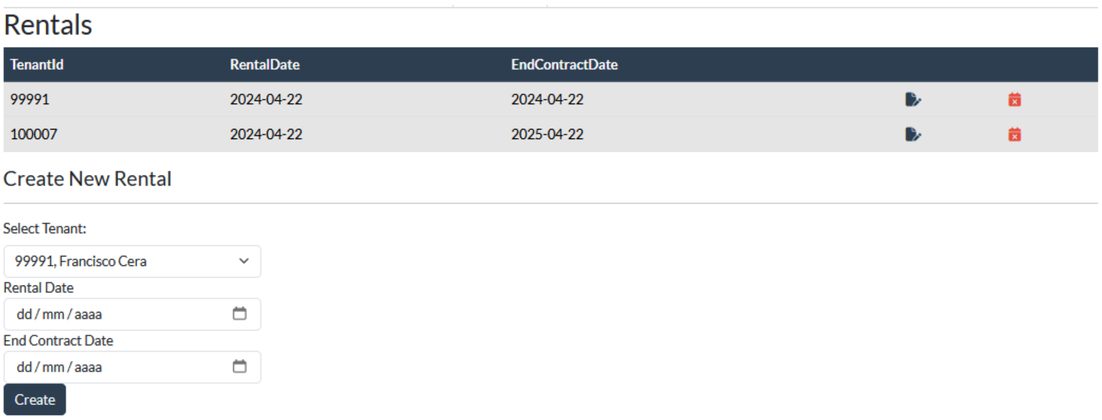


Date	Description	Status		
2024-03-25	fuga de agua	Solved	●	🗑
2024-04-21	water lake on the bathroom	Solved	●	🗑

Report and Create an Event

Description

The tab events show to managers and owners the list of events that the apartment has, with an option to mark it solve or pendint if is already solved, also it can be eliminated, and created.



Rentals

TenantId	RentalDate	EndContractDate		
99991	2024-04-22	2024-04-22	✎	🗑
100007	2024-04-22	2025-04-22	✎	🗑

Create New Rental

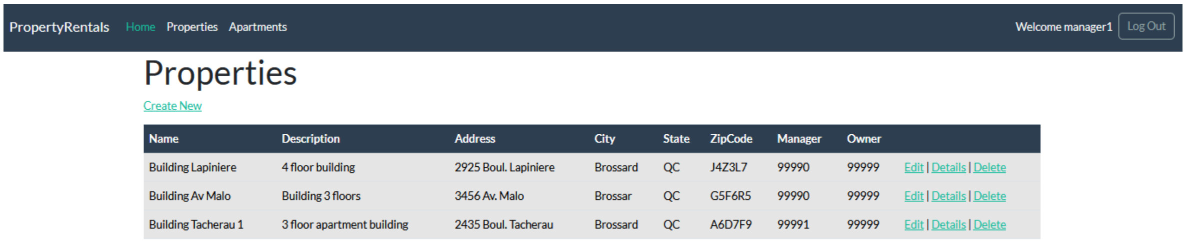
Select Tenant: 99991, Francisco Cera

Rental Date: dd / mm / aaaa

End Contract Date: dd / mm / aaaa

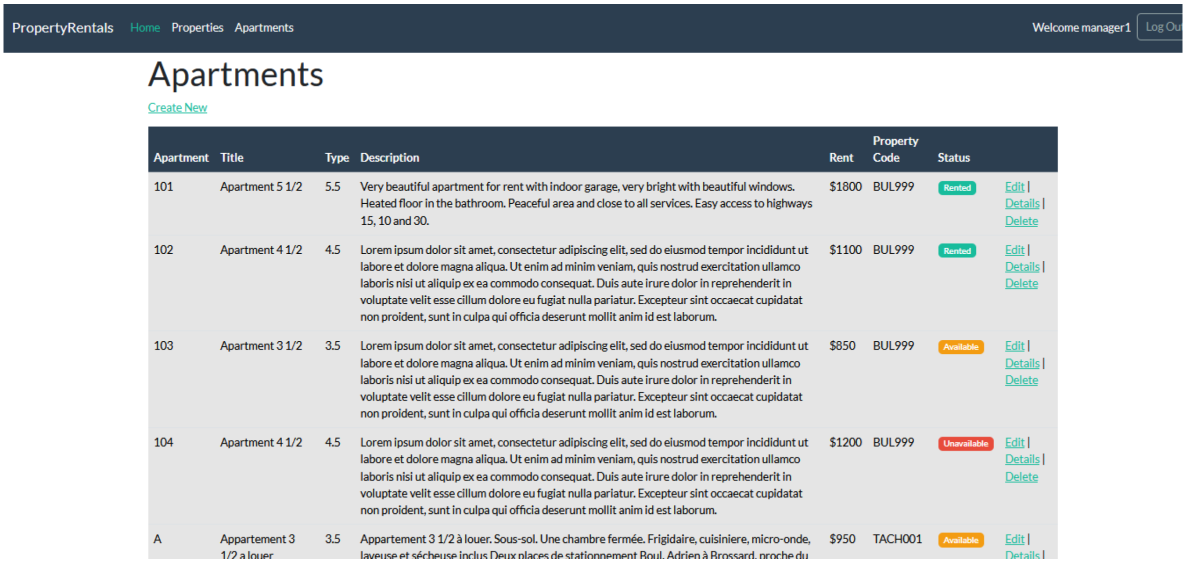
On the Rentals tab all the history of rentals can be showed with the possibility of editing the rental or delete it, also the manager or owner can create a new rental selected the tenant and dates.

The manager or owner can also manage the properties and apartments, they can list, create, edit, view details or delete.



Properties

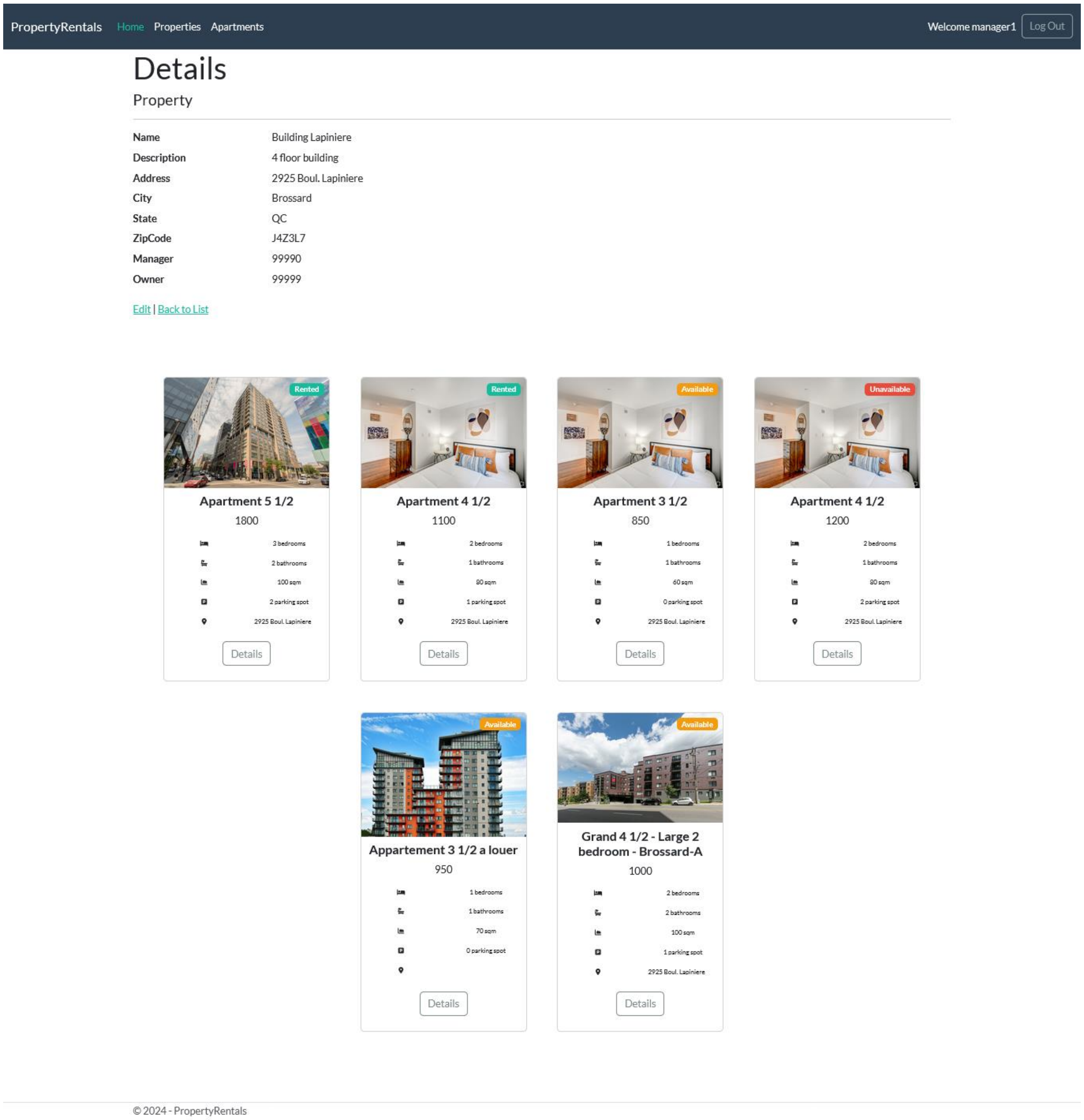
Name	Description	Address	City	State	ZipCode	Manager	Owner	
Building Lapiniere	4 floor building	2925 Boul. Lapiniere	Brossard	QC	J4Z3L7	99990	99999	Edit | Details | Delete
Building Av Malo	Building 3 floors	3456 Av. Malo	Brossar	QC	G5F6R5	99990	99999	Edit | Details | Delete
Building Tacherau 1	3 floor apartment building	2435 Boul. Tacherau	Brossard	QC	A6D7F9	99991	99999	Edit | Details | Delete



Apartments

Apartment	Title	Type	Description	Rent	Property Code	Status	
101	Apartment 5 1/2	5.5	Very beautiful apartment for rent with indoor garage, very bright with beautiful windows. Heated floor in the bathroom. Peaceful area and close to all services. Easy access to highways 15, 10 and 30.	$1800	BUL999	Rented	Edit | Details | Delete
102	Apartment 4 1/2	4.5	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	$1100	BUL999	Rented	Edit | Details | Delete
103	Apartment 3 1/2	3.5	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	$850	BUL999	Available	Edit | Details | Delete
104	Apartment 4 1/2	4.5	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	$1200	BUL999	Unavailable	Edit | Details | Delete
A	Appartement 3 1/2 a louer	3.5	Appartement 3 1/2 à louer. Sous-sol. Une chambre fermée. Frigidaire, cuisiniere, micro-onde, laveuse et sécheuse inclus Deux places de stationnement Boul. Adrien à Brossard, proche du	$950	TACH001	Available	Edit | Details |

The details of each property show the list of the apartments on the property.



Details

Property

Name	Building Lapiniere
Description	4 floor building
Address	2925 Boul. Lapiniere
City	Brossard
State	QC
ZipCode	J4Z3L7
Manager	99990
Owner	99999

Edit | Back to List

Rented
Apartment 5 1/2
1800
3 bedrooms
2 bathrooms
100 sqm
2 parking spot
2925 Boul. Lapiniere
Details

Rented
Apartment 4 1/2
1100
2 bedrooms
1 bathrooms
80 sqm
1 parking spot
2925 Boul. Lapiniere
Details

Available
Apartment 3 1/2
850
1 bedrooms
1 bathrooms
60 sqm
0 parking spot
2925 Boul. Lapiniere
Details

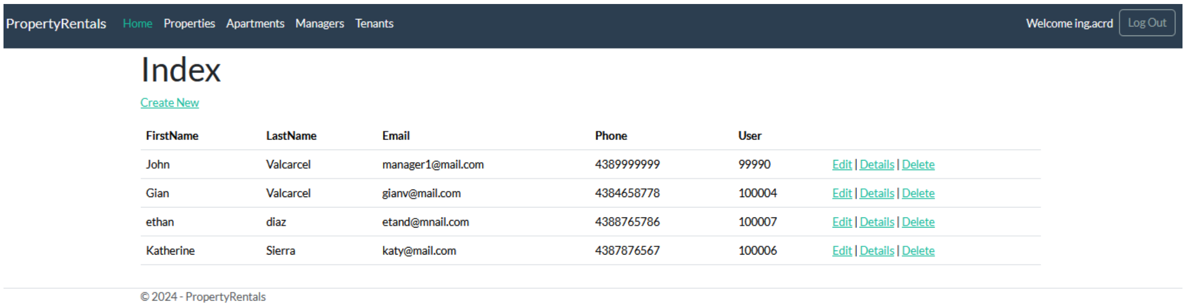
Unavailable
Apartment 4 1/2
1200
2 bedrooms
1 bathrooms
80 sqm
2 parking spot
2925 Boul. Lapiniere
Details

Available
Appartement 3 1/2 a louer
950
1 bedrooms
1 bathrooms
70 sqm
0 parking spot
Details

Available
Grand 4 1/2 - Large 2 bedroom - Brossard-A
1000
2 bedrooms
2 bathrooms
100 sqm
1 parking spot
2925 Boul. Lapiniere
Details

© 2024 - PropertyRentals

The owner also can manage the tenants, and managers. The tenants can be edited, deleted and see the details, and the managers can be created from a registered tenant, which is the user by default created at the moment of sign up. The managers can be edited, deleted and view the details.

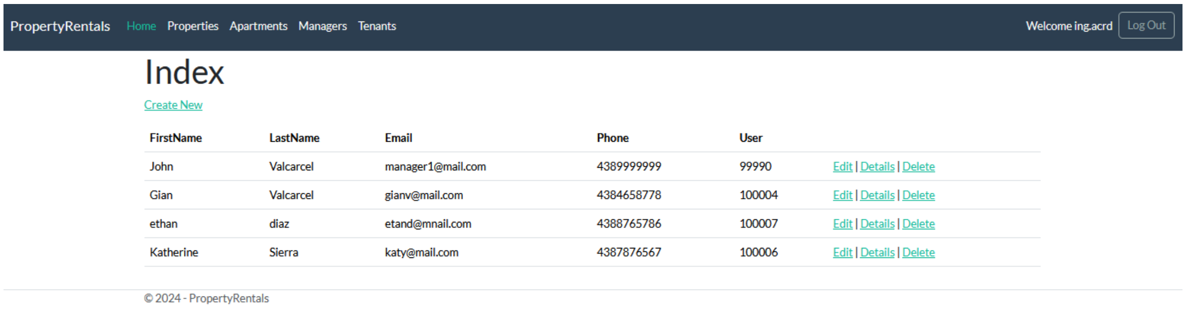


Index

Create New

FirstName	LastName	Email	Phone	User	
John	Valcarcel	manager1@mail.com	4389999999	99990	Edit | Details | Delete
Gian	Valcarcel	gianv@mail.com	4384658778	100004	Edit | Details | Delete
ethan	diaz	etand@mnail.com	4388765786	100007	Edit | Details | Delete
Katherine	Sierra	katy@mail.com	4387876567	100006	Edit | Details | Delete

© 2024 - PropertyRentals



Index

Create New

FirstName	LastName	Email	Phone	User	
John	Valcarcel	manager1@mail.com	4389999999	99990	Edit | Details | Delete
Gian	Valcarcel	gianv@mail.com	4384658778	100004	Edit | Details | Delete
ethan	diaz	etand@mnail.com	4388765786	100007	Edit | Details | Delete
Katherine	Sierra	katy@mail.com	4387876567	100006	Edit | Details | Delete

© 2024 - PropertyRentals

PHASE IV TESTING THE PROGRAM

User	Functional requirement	Test result/ problems
user	Sign in, Sign up	Successful
		If logging, after open the page using another browser, the application requires the login again.
Tenants	Access	Doesn't have access to controllers no authorized
	View any apartment suitable for their needs	Successful, the tenant can see all the available apartments from the home page.
	Make an appointment with the property manager	Successful, the tenant can make an appointment from the apartment details.
	Send necessary messages to the property manager	Successful, the tenant can send messages from the apartment details.
Manager	Perform CRUD operations related to properties	Successful, the manager can manage properties, from navigation bar
	Perform CRUD operations related to apartments	Successful, the manager can manage all the apartments,

		create, edit, delete, update from navigation bar.
	Keep track of apartments status	The status can be tracked from the home page, the apartment details and the list of apartments
	Schedule potential tenants appointments	Successful, the manager can make appointments from the apartment details.
	Respond to potential tenants 'messages	Successful, the manager can send messages to a specific Tenant, an answer their messages.
	Report any event to the property owner when necessary	Successful, the manager can report any event from the apartment details.
Owner	Create/Update/Delete/Search/List any property manager account	The owner can create, update, delete and see the list of the managers from the nav bar menu/Managers
	Update/Delete/Search/List any potential tenant account	The owner can update, delete and see the list of any potential tenant account , accessing from nav bar menu/Tenants
	Full control of the Web Site	The owner can do the same things that the manager does.

CONCLUSION

- The Property Rental Management Web Site using technologies like ASP.Net Core MVC on the server side and Bootstrap and JavaScript on the client side, ensures a robust and user-friendly experience.
- This project allowed to me put into practice all that I have learn in different courses and build a robust project that improves my skills and can be shown on my portfolio.

BIBLIOGRAPHY

- Bootstrap theme (https://bootswatch.com/flatly/)
- Bootstrap documentation (https://getbootstrap.com/docs/5.3/getting-started/introduction/)

- Font awesome(https://fontawesome.com/)
- Bootstrap snippets (https://bootsnipp.com/snippets/dla4e#google_vignette)
- Light box gallery bootstrap snippet (https://epicbootstrap.com/snippets/lightbox-gallery)
- Registration with photo bootstrap snippet(https://epicbootstrap.com/snippets/registration-form-with-photo)
- Login Authentication/Authorization in ASP.NET CORE 6 MVC (https://www.youtube.com/watch?v=uGoNCJf0t1g&t=1035s)