Project Title

Rental management system for teachers (REFT)

Project's team members:

البرنامج	اسم الطالب	الرقم الأكاديمي	م
IT	Karim Allah Saad Saad Elbanan	802321980	.1
IT	Osman Mamdouh Ramadan	802267541	.2
IT	Mahmoud Refaat Osman	802324202	.3
CS	Abdulrahman Ali Ali Almowafy	803091743	.4
CS	Maher Ahmed Ali Alshahbor	806334998	.5
IT	Ahmed Mohamed Gamal Elmasarawy	800167476	.6
CS	Muaaz Reyad Hasan Abdelwahed	806329005	.7
IS	Muaaz Shaaban Abdelwahed	802324808	.8

Under the supervision of:

1. Dr./ Ahmed Mohamed Rabie

2023

Acknowledgment

We want to thank our parents for always supporting us during our project. Your encouragement means a lot, and we couldn't have done it without you.

A big thank you to D. Ahmed Mohammed Rabea for guiding us through the project. Your help and expertise made a real difference in our learning.

This achievement is a team effort, and we're grateful for the support of our parents and D. Ahmed Mohammed Rabea

Thank you

Abstract

It is obvious that our country, Egypt, is adopting the new technologies to keep up with technological revolution in the world. We believe that modern technologies should be applied in all educational fields, because better education is central to human happiness and well-being. It also makes an important contribution to economic progress, as educated populations are Conscious, more productive, and earn more. So, our proposed system will target the education field. We are going to build a complete environment to meet the teacher's needs beginning from searching for the appropriate place for him, communicating with the owner of the place and completing the reservation process while ensuring security. The system has been developed using React, Node.js and a renting website. So, "Rental management system for teachers" is the title of our project and we call the system REFT.

Keywords: React; Node.js; Renting Website; Security; Teacher.

Table of Contents

Contents

I. List of Figures

II.   List of Tables

1- Introduction

1.1 Overview

We are currently living in the era of technology, which plays a significant role in the progress of nations towards an era characterized by precision, efficiency, and saving time and effort in all areas of life, whether it is in engineering, medicine, agriculture, industry, or education. Our topic specifically focuses on the educational aspect because all other fields depend on it. It is the cornerstone or solid foundation upon which all other fields rely.

In this chapter, we will provide an introduction to our proposed project. First, in section 1.2 and section 1.3 we discuss the motivation and the objectives of our project respectively. Next, in section 1.4, we present the scope of the intended project. Then, Section 1.5 shows timeline of our project. Finally, section 1.6 provides a document organization including a brief description about the context of each chapter.

1.2 Problem Definition

Egyptian families spend very large sums of money for private lessons for their children, even though government education is free and e-learning is available in an unprecedented manner.

one of the most prominent reasons for the high cost of private lessons is that teachers rent places for large sums of money to give lessons to their students.

We will create a website that allows the teacher to rent the place, but according to the number of hours he needs to teach

Consequently, the cost will be reduced for the teacher and thus for the Egyptian family, allowing them to spend the amount saved on other matters.

1.3 Project Objectives

We will build a website that helps the teacher find the best place for him to give private lessons at a much lower price than the current situation and at the same time increase the profits of the owner of the place. This is a list of the most prominent goals of the project.

1) It allows the owner of the place to share the location and photos and the cost per hour of the place

2) The place owner can determine the number of rooms to be rented and the appropriate timing for renting and even the appropriate days in the week

3) The teacher can search for available places near him

4) The teacher can rent a specific number of hours per month according to his personal needs

5) The place owner can determine the payment method, whether online or by physical methods

1.4 Project Scope

The intended system aims to help the teacher find the appropriate place for him to give educational lessons by controlling the area of the place, the capacity of students, the cost of rent, and the appropriate location. In addition, it allows the owner to publish his property on our website with the needed information, such as the location, pictures of the place, the rental price, and the available days and hours without further trouble on both sides to find the appropriate option for them.

1.5 Project Timeline

#	Task Name	Start	Finish
1	Background research	20-10-2023	30-10-2023
2	Determine objectives	01-11-2023	20-11-2023
3	Review related work and papers	21-11-2023	10-12-2023
4	Determine needed components	11-12-2023	22-12-2023
5	Design diagrams	1-01-2024	7-1-2024
6	Interface design for the mobile application	1-01-2024	7-1-2024
7	Presentation for phase 1	1-01-2024	7-1-2024

Table1.1 project timeline

1.6 Document organization

This document consists of six chapters in addition to one appendix. A brief description about the contents of each chapter is given in the following paragraphs:

Chapter 1: - Introduction, provides an introduction to the project, overview, definition, objectives and scope of project.

Chapter 2: - Literature Review, provides the reader with an overview of the previous related work, common technologies used, and the relation between our work and the relevant work.

Chapter 3: - System Analysis, includes the analysis of existing system, system requirements, use requirements, system architecture, development methodology languages used in our system.

Chapter 4: - System Design, provides the system design including class diagram design and interface design.

Chapter 5: - System implementation, shows the process of mapping design into implementation, sample application code, system testing, results of the investigation, and goals achieved.

Chapter 6: - Results and Future Work, displays the results of the project, the next release, and future additions.

2- Background and Related Work

In this chapter, we provide a background of the tools and techniques needed to build our system. We also review the previous related work to our system and the common technologies that are used. At the end of the chapter, we provide a comparison between the system we are going to build and the related systems.

2.1 Background

The system we are going to build will depend on Google Map API and web application to operate. The following subsections provide a brief background information about tools, and techniques needed to build our system including programming languages and frameworks. There is a total of two major parts of the project plan, which are the frontend and backend parts. In this scientific research project HTML, CSS, Tailwind CSS, JavaScript, and React are used for the frontend design of the system. MongoDB and Express are used for creating and designing the database and Backend part. For writing all the codes Visual Studio Code editor is used.

2.1.1 HTML

HTML stands for Hypertext Markup Language which allows the user to make and structure sections, paragraphs, headings, titles, line breaks, add media, links, blockquotes, etc. for websites and applications.

2.1.2 CSS

CSS stands for Cascading Style Sheets which is a simple design language intended to simplify the method of constructing this website presentable. It's designed to enable the separation of presentation and content, including layout, colors, spacing, padding, fonts, and so on.

2.1.3 Java Script

JavaScript is a scripting or dynamic computer programming language that allows the implementation of complex features on web pages and client-side scripts to interact with the user and make dynamic web pages.

2.1.4 Tailwind CSS

Tailwind CSS's official documentation describes it as a "utility-first CSS framework" with classes equipped to build custom UI designs directly in the users' markup. It is handy to implement inline styling to rustle up a stunning UI without writing any CSS.

2.1.5 React

React is the library for web and native user interfaces. Build user interfaces out of individual pieces called components written in JavaScript.

2.1.6 MongoDB

MongoDB is a popular NoSQL database management system. It is known for its flexibility, scalability, and ability to handle large amounts of data. MongoDB uses a document-oriented model, where data is stored in flexible, JSON-like documents. This allows for easy manipulation and querying of data. It is widely used in various industries for building modern, scalable applications.

2.1.7 Express

Express.js is a fast and minimalist web application framework for Node.js. It provides a simple and flexible way to build web applications and APIs. With Express.js, you can easily handle HTTP requests, define routes, and implement middleware for tasks like authentication and error handling. It has a robust ecosystem of plugins and middleware that can be used to extend its functionality. Express.js is widely adopted and is a popular choice for building server-side applications with Node.js.

2.2 Review of Relevant Work

The following are some of the research and papers that study Renting or reserving places for various purposes:

- Ikuomola A. J. and Asefon M.P. A Secured Mobile Cloud-Based House Rental Management System. In this paper, a secured mobile cloud-based house rental management system is designed. The system provides landlord/agent/tenant/prospective tenant with information on house records. These records can be shared with other users like making an advertisement for an apartment that is available. It also provides a payment system and wallet engine which allows easy means of making transactions.

Landlords' powers in the application:
- Listing properties with details and photos
- Managing rental agreements and leases electronically
- Processing rent payments online
- Communicating directly with tenants through the app
- Tracking maintenance requests and assigning work orders
- Accessing reports and analytics about their rentals

Tenants' powers in the application:

- Searching for rental properties based on their preferences

- Applying for rentals online

- Viewing and signing rental agreements electronically

- Making rent payments online

- Submitting maintenance requests and tracking their status

- Communicating directly with landlords through the app
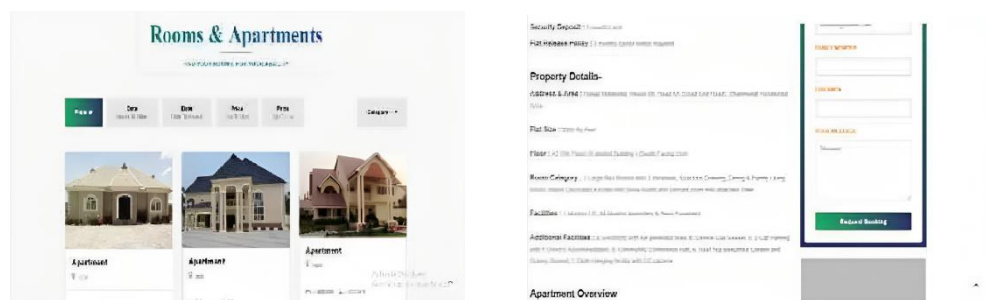
Figures 2.1 and 2.2

-Mandeep Katre (Professor), Kumar Abhay Gupta, Shipra Katiyar, Saumya Shahi, Shreya Awasthi. A Review Paper on PG Recommendation System.

In this paper rental website development is discussed. Through this website, tenants can rent properties and landlords can upload their property details for rent. This website will help users to give or take rent houses without dealing with flat brokers to finally reach an agreement that suits the interests of all parties. The packaging and shifting services facility will help users shift valuable things safely to their destinations.

The mechanism of action of the application is:

I. Search by preferred location, currency, distance, etc.
II. We provide all hostels, PG, Rental rooms, etc.
III. Students can hire a teacher close to their studies.
IV. Daily, weekly, and monthly visit justice.
V. Provides quality Tiffin daily, weekly, and monthly service.

A. Admin Module

1) Dashboard: In this section, the administrator can see all the details briefly like country total, city total, total owner, and PG theme.

2) Region / City: In this section, the administrator can manage status (add / update).
3) Record Owner: In this section, the administrator can view and edit the registered owner.

B. PG Owner Module

1) Dashboard: In this section, the owner can view all the details briefly like pg. listed list, complete booking, new booking, total guaranteed booking, and total canceled booking.
2) List Your PG: In this section, the owner can list their page.
3) Received Request Booking: At this stage, the administrator may view the new booking and is entitled to a guaranteed booking.

C. User module
Registered users can do the following work

1) Book a page.
2) Update his profile.
3) Change the password.
4) The registered user can regain his or her password.
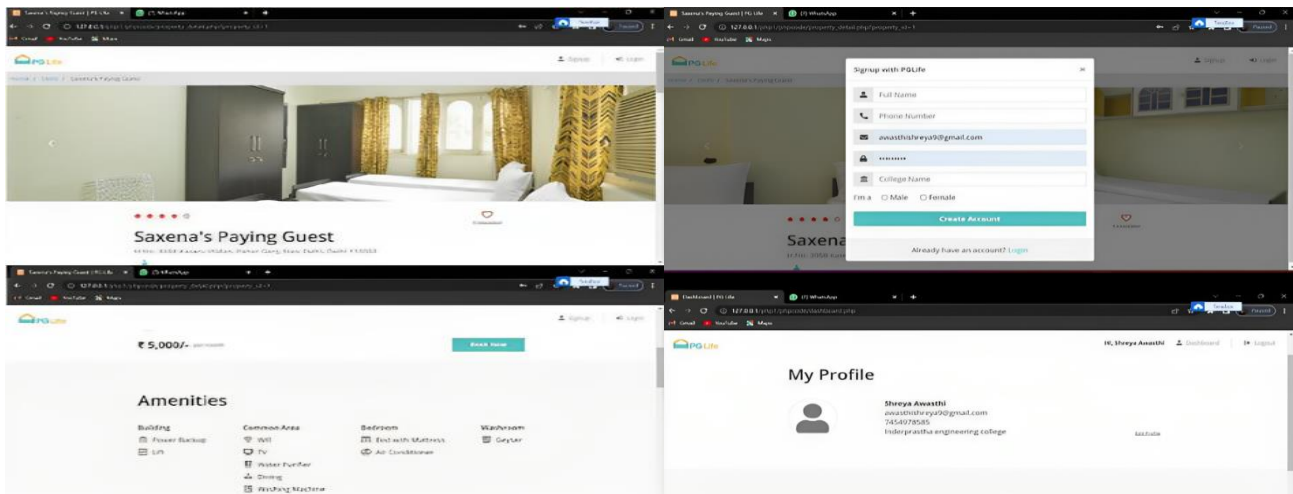
Figures 2.3 and 2.4

-Murahari Prithvi Yash, Chinmay Choudhary, Akanksha Lakra, Swati Dewangan. RentoAxis: Android App for Paying Guest Management.

This paper presents the design and implementation of an Android-based Paying Guest (PG) management app named RentoAxis. The proposed

system comprises an android application where (i) the PG owners can advertise the PG details by registering into the application. (ii) the user (common people) can search for the PG, apartments, and houses all over Durg and Raipur. (iii) the customers can also search the PG information based on some parameters such as nearby colleges, markets, etc. In other words, this application will help PG owners improve the management of their community and the tasks related to it.

A. Administrator Module:
• Manages all the information and has access rights.
• Check the details of the PG owner as well as the customer.
• View/Delete the details related to PG, owner, and user.

B. Owner Module:
• The PG Owner will follow the registration procedure by using identity proofs.
• They can easily add details of the Apartment, Rooms, and Paying Guests and can update or delete the details according to the need.

C. User Module:
• The user will also register by their unique ID.
• Users can search for hostel rooms, paying guests, etc., and get the room details, room rent, address of the room, pictures of the room, etc. across the city.
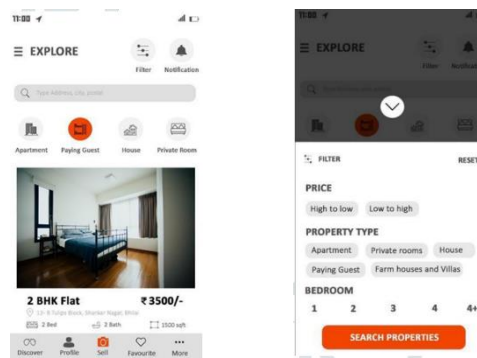
This app works in three phases:

o Phase 1- RentoAxis: Login Screen

o Phase 2- RentoAxis: PG list and Filter Option

o Phase 3- Accommodation Details and Reservation

Stages of application:

Figure 2.5

The Snapshot (a) shows the users can sign up or log in to link their data to this email and facilitate the browsing process in the application.

Figures 2.6 and 2.7

Snapshots (b) and (c) show the client searching for a room based on his personal preferences.
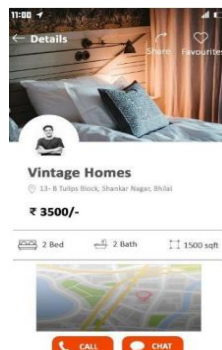

Figure 2.8

The snapshot (d) shows the client booking the room in which he will stay after settling on the price and number of rooms.

2.3 Relationship between the Relevant Work and Our Work

The previously mentioned related work shows that designed systems that study the Rental Management Systems and Paying Guest Management but have some drawbacks we will overcome them with the following:

1- Fills a specific market need: Our project addresses the unique needs of teachers seeking affordable lesson spaces, which is not addressed by other platforms.

2- Reduced cost for teachers: By connecting teachers directly with space owners, our project eliminates the need for middlemen and reduces the cost of renting spaces.

3- Increased profits for space owners: Our project allow space owners to set their prices and reach a wider audience of potential renters.

4- Secure and convenient platform: Our project provides a secure platform for booking and managing rentals, as well as a system for reviews and ratings.

5- Limited focus: These platforms are not designed specifically for private lessons, leading to a cluttered search experience for teachers and potentially unsuitable spaces.

6- High cost: Renting spaces through other applications can be expensive, particularly for teachers who need them regularly.

7- Lack of specific features: These platforms lack features tailored to private lessons, such as booking specific hours, managing student schedules, or providing feedback.

8- Diversification of income streams: Renting spaces for private lessons can be an additional income stream for space owners, alongside any existing rental activities.

9- Reduced risk of cancellations: A secure booking system and clear cancellation policies can help teachers avoid lost income due to last-minute cancellations.

2.4 Summary

The previous sections show the needed background information to understand how we are going to build a system that can provide places for private lessons while saving the time and effort of teachers in searching for specific locations to teach the subjects. Section 2.1 provides information on software tools, and techniques needed to develop the web application and how to employ the technologies to suit the functions of the application. Section 2.2 discusses relevant work based on research and papers that study Rental Management Systems and Paying Guest Management. The first app intended to provide landlord/agent/tenant/prospective tenant with information on house records, second app is a website that will help users to give or take rent houses without dealing with flat brokers to finally reach an agreement that suits the interests of all parties, last app is an application for helping PG owners to improve the management of their community and the tasks related to it. In section 2.3, we discuss the relationship between our intended project and the related work. We seek to build a platform that facilitates affordable private lessons while increasing profits for space owners., Also, provides a secure platform for booking and managing rentals, as well as a system for reviews and ratings.

3- System Analysis and Design

In this chapter we provide a detailed knowledge about our system including functional requirements, non-functional requirements, user requirements, system architecture, use case and sequence diagrams. We can divide the process of analysis into three parts, which are determining requirements, determining system architecture, and determining development methodology. Section 3.2, Section 3.3, and Section 3.4 will discuss those three parts. In the end of this chapter, we provide information about the tools and languages we needed to use to build our system.

3.1 System Requirements

System requirements are the needed configurations for the system to operate efficiently. The next three subsections will discuss the functional requirements, Nonfunctional requirements, and user requirements.

3.1.1 Functional Requirements

In systems engineering, functional requirements are directly concerned with the system services, where a function is described as a specification of behavior. In this subsection, we list the functions required in our system. We, also, provide a description for each function

3.1.2 Non-functional requirements

The system needs to operate efficiently and meet the requirements. Any failure of the components of the systems may lead to one or more of functions stopping or being misused. A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system.

#	Functional Requirement	Description
1	Allow the owner to share the location and photos	The owner of the place can share the geographical location of the place, upload pictures of the rooms he wants to rent, and also determine the appropriate financial cost for one hour of rent.
2	Allow owner to specify the number of rooms and appropriate timings	The owner of the place determines the number of rooms he wants to share on the site and also determines the number of hours during which he wants the rent to be made
3	Allows the teacher to search for nearby places	The site helps the teacher by showing him the nearest geographical places The teacher can also search for available places and hours in any city he wants to search for
4	Permit teacher to rent a specific number of hours according to his need	The teacher can rent as many hours as he wants on the days of the week he wants as long as it is available on the website
5	Online payment available	The teacher can choose the payment method, either online or in the traditional way
6	Add some laws and conditions	The owner of the place can set some conditions in addition to the general site rules, and the teacher must adhere to those conditions, and if he violates them, he is subject to fines.

Table 3.1: functional requirements for System

#	Non-functional requirements	
1	Availability	ensure availability by using redundant servers
2	Performance	optimize code, minimize loading time
3	Capacity	Low storage space consumption.
4	Recoverability	System response within a few seconds.
5	Security	there is authentications and encryption
6	Scalability	monitor for scalable growth utilize cloud infrastructure
7	Usability	Easy to use and understand for different users

Table 3.2 Non-functional requirements

3.1.3 User requirements:

The user of this site is in fact two users, the owner of the place (the lessor) who wants to rent one or more rooms to teachers according to the time that suits him, and the other user is the teacher who is looking for a suitable place to reserve for a number of hours per month at a price suitable for him.

Place owner requirements

1) Login to the system
2) Share the geographical location of the place
3) Determine the number of rooms
4) Determine appropriate timings
5) Determine the payment method
6) Add an electronic wallet number
7) Determine the money per one hour
8) Establish special conditions for it
9) Delete the place

The Teacher Requirements

1) login to the system
2) search for the nearest places location
3) Search for places in other cities
4) Choose a suitable place for him
5) Choose the appropriate room for the number of students he has
6) Choose the number of hours he wants
7) Choose a payment method
8) Cancel reservation

3.3 Development Methodology

After we knew the basic structure of the system. We are going to view all of its functions, the relation between them, and the Activity diagram in the following subsections.

3.3.1 Use Case Diagrams

First, with a use case diagram, we will specify the expected behavior (what) of the system, not the exact method of making it happen (how). This helps us to design the system from the end user's perspective. Figure 3.3 shows the use case diagram where the system boundary is the EduNet and the actor is the place owner and the teacher.
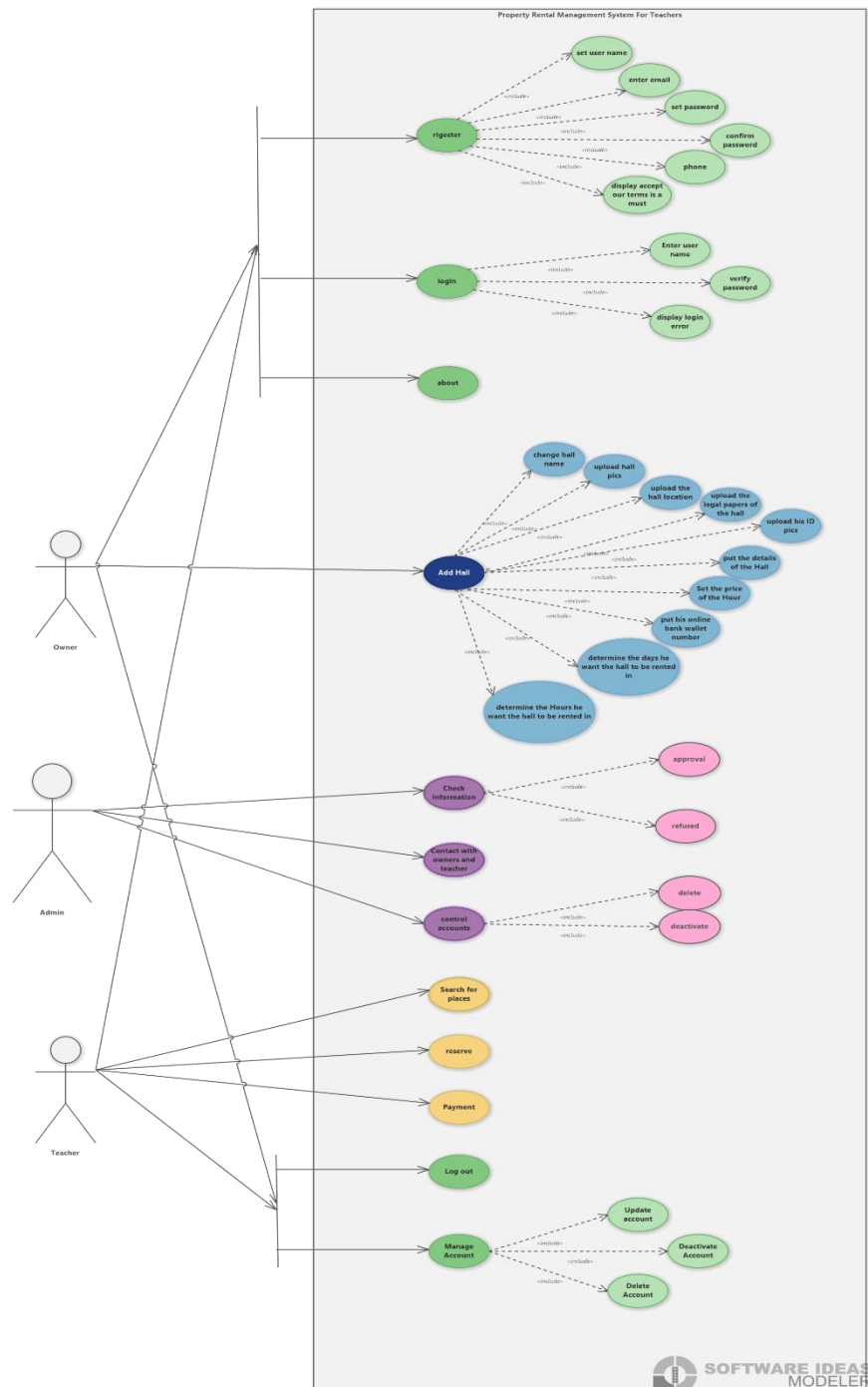
Figure 3.1 Use Case diagram

3.4 Tools and Languages

Developing our software application can be divided into main two parts, which are the design part and the implementation part. The design part involves designing diagrams and designing user interface of the site. The implementation part involves programming languages, IDEs, frameworks, and libraries. The following list shows the needed tools for the software development and a brief description about their usages:

1. **Software Ideas Modeler** – it is used to draw the UML diagrams.

2. **Figma** – it is used to design user interfaces and prototypes.

3. **Visual studio code –** it's used to write the programming codes

3.5 Summary

In this chapter we provide the reader with detailed knowledge about our system. Section 3.1 provides system requirements Which is divided into functional and non-functional requirements, and user requirements which specify some different specifications for users, Section 3.2 includes system architecture which describe its major components, their relationships, and how they interact with each other, Section 3.3 provides development methodology which includes UML diagrams that shows the details of how the system will function. In the end of the chapter, we listed the needed tools for us to build the system.

4- System Architecture

4.1 class diagram

Class diagrams are the main building blocks of every object-oriented method. The class diagram can be used to show the classes, relationships, interface, association, and collaboration. UML is standardized in class diagrams. Since classes are the building block of an application that is based on OOPs, so as the class diagram has an appropriate structure to represent the classes, inheritance, relationships, and everything that OOPs have in their context. It describes various kinds of objects and the static relationship between them.
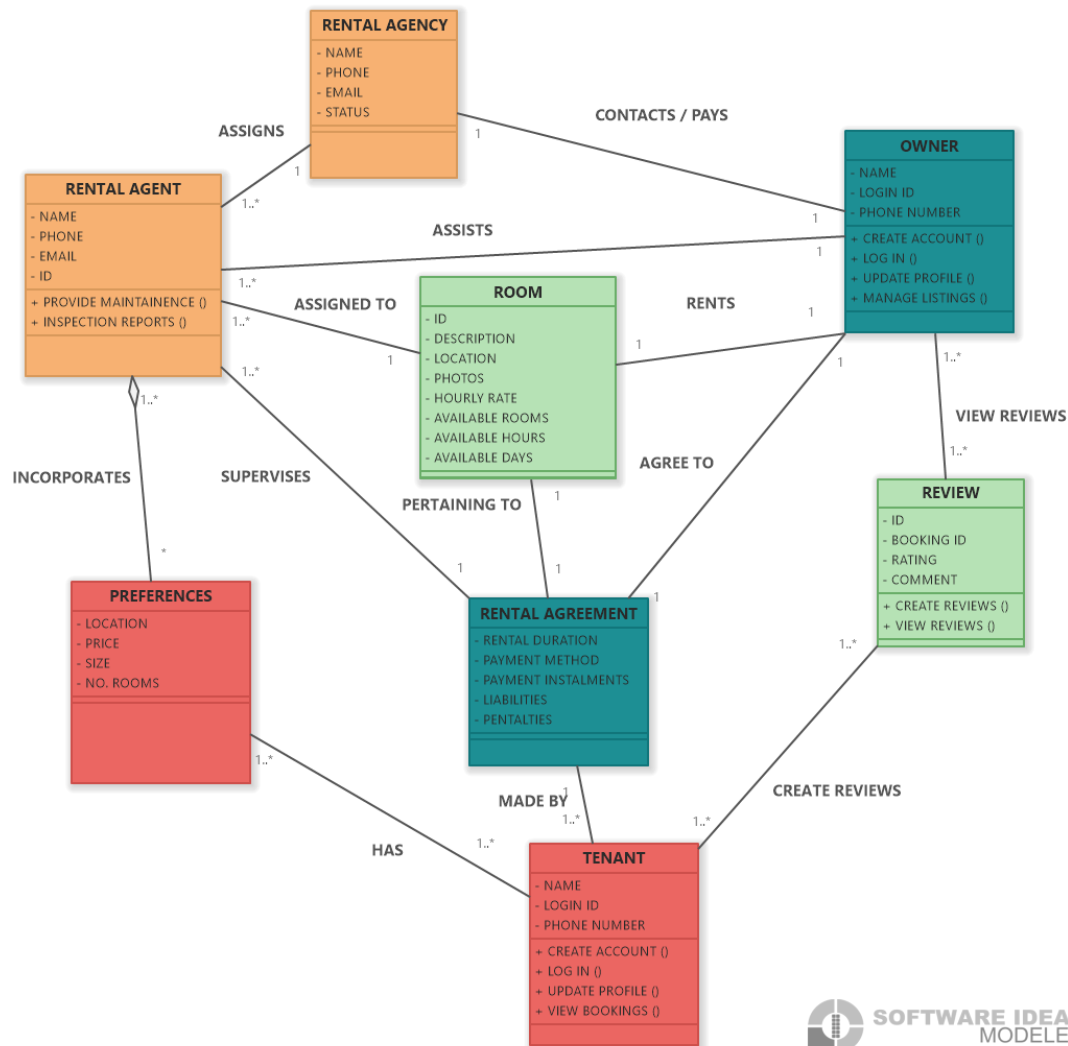
Figure 4.1 class diagram

4.2 Activity Diagram

We use Activity Diagrams to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on the condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram. UML models basically three types of diagrams, namely, structure diagrams, interaction diagrams, and behavior diagrams. An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modelling where their primary use is to depict the dynamic aspects of a system. An activity diagram is very similar to a flowchart. So let us understand if an activity diagrams or flowcharts are any different:
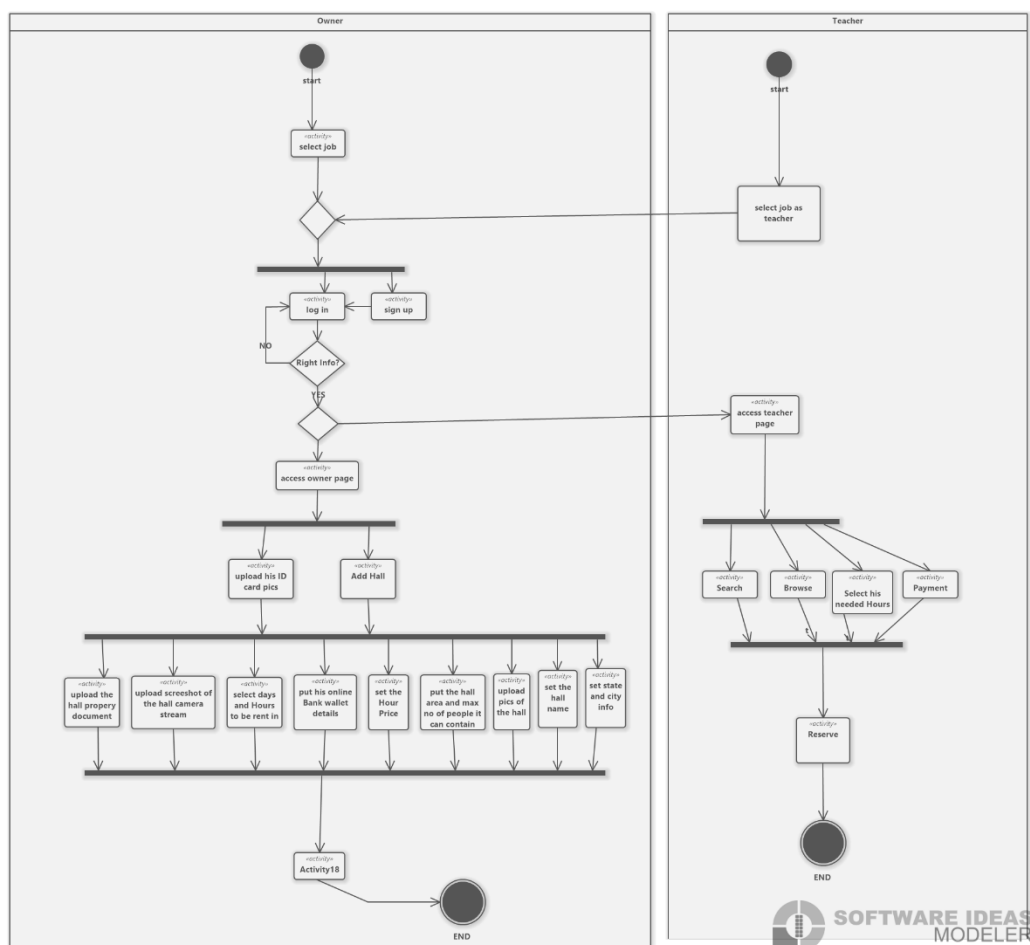

Figure 4.2 Activity diagram

4.3 System Architecture

1. Frontend:

• User Interface (UI): The frontend should have a user-friendly interface for both property owners and teachers.

• Search and Filters: Implement a robust search functionality with filters like location, size, amenities, etc.

• User Authentication: Secure registration and login system for teachers and property owners.

• Property Listings: Display property listings with details, pricing, and availability.

• Booking System: Enable teachers to submit booking requests, and property owners to manage and confirm bookings.

2. Backend:

• Server: Host your backend on a server. Popular choices include AWS, Google Cloud, or Azure.

• Backend Framework: Use a backend framework like Django (Python), Ruby on Rails (Ruby), or Express.js (Node.js).

• Database: Store property information, user details, and bookings in a relational database (e.g., PostgreSQL, MySQL).

• Authentication: Implement a secure authentication system for user login and authorization.

• APIs: Create APIs to handle communication between the frontend and backend.

3. User Management:

• User Profiles: Allow users to create and manage their profiles.

• Messaging System: Implement a messaging system for communication between teachers and property owners.

4. Listing Management:

• Property Upload: Create a feature for property owners to upload details, images, and availability of their places.

• Content Management System (CMS): Implement a CMS for property owners to manage their listings easily.

5. Booking System:

• Reservation Handling: Develop a system for handling booking requests and confirmations.

• Calendar Integration: Integrate a calendar system to show property availability.

6. Payments:

• Payment Gateway: Integrate a secure payment gateway (e.g., Stripe, PayPal) for handling transactions.

• Pricing System: Implement a pricing system based on property features, location, and duration of stay.

7. Notifications:

• Email Notifications: Send email notifications for account verification, booking confirmations, and other relevant updates.

• In-App Notifications: Implement in-app notifications for real-time updates.

8. Security:

• SSL Certification: Ensure secure data transmission with SSL.

• Data Encryption: Encrypt sensitive data, especially user credentials and payment information.

9. Analytics and Monitoring:

• Analytics Tools: Integrate tools like Google Analytics to track user behavior.

• Error Monitoring: Implement a system to monitor and log errors for quick troubleshooting.

10. Scaling:

• Scalable Infrastructure: Design the architecture to scale horizontally as the user base grows.

Remember, this is a high-level overview, and the specific technologies you choose will depend on your preferences, expertise, and other project requirements. Additionally, consider legal aspects, such as terms of service and privacy policies, to protect both property owners and teachers using your platform.

5- Practical Implementation Details

Interface Design

We seek to minimize time and efforts needed by the user to do some action. So, the interface of the application should be clear, easy to use, easy to understand, and smooth. The following subsections show the designed views of the website and description of each view.
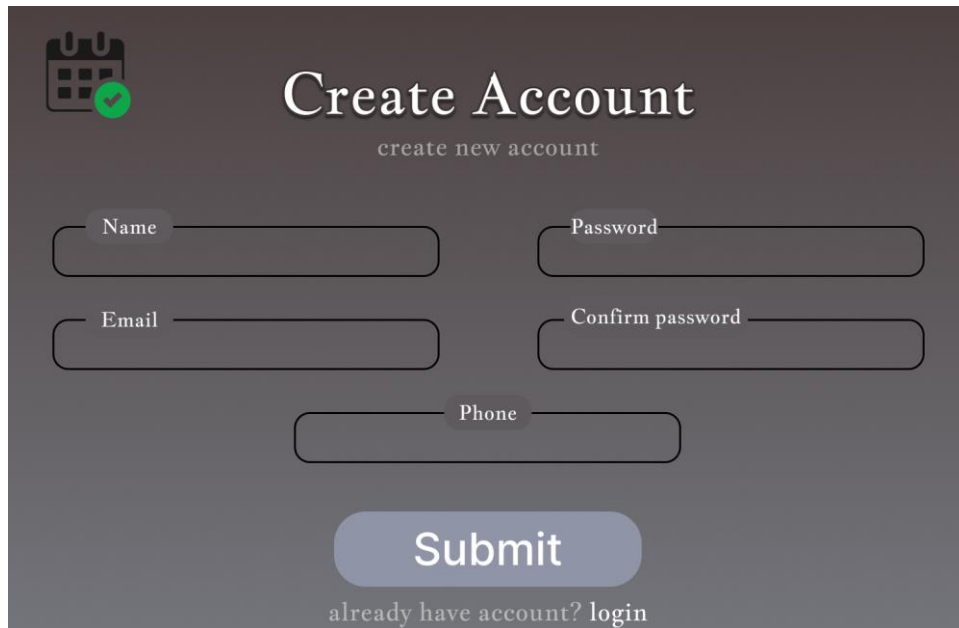
5.1 splash screen

The name of the site and its promotional phrase appear here, as well as a Start button for the beginning of the site

5.2 Signup page(register):

1- **First input field: used for entering name.**

2- **Second input field: used for entering Email.**

3- **Third input field: used for entering phone number.**

4- **Fourth input field: used for entering password.**

5- **Fifth input field: used for entering password conformation.**

6- **Signup button: Confirm registration process.**
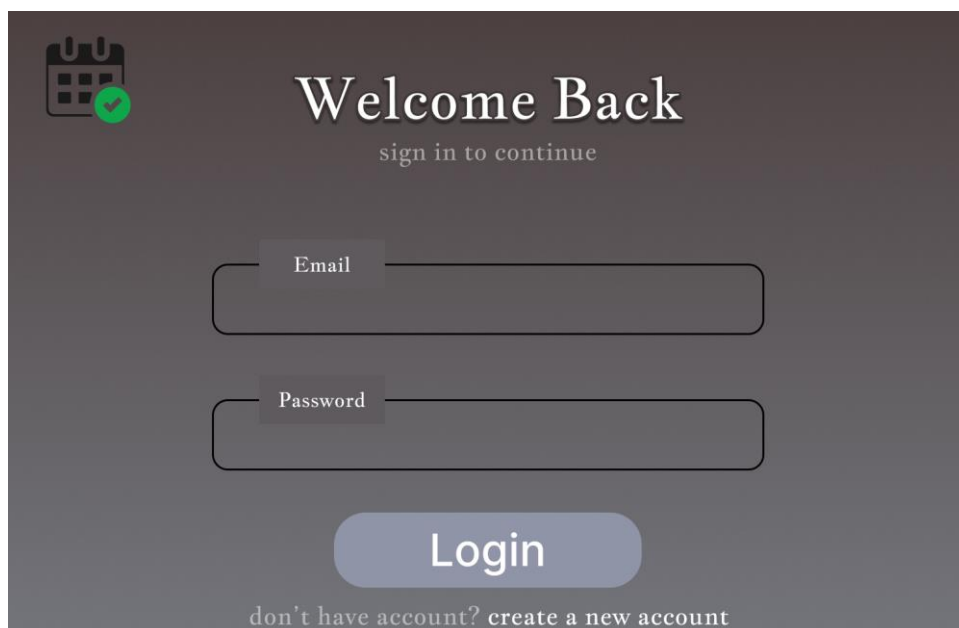
5.3 Log in Page

1- First input field: used for entering registered Email.

2- Second input field: used for entering password.

3- Login button: Confirm access to your account.

5.4 introductory page

There are two users on our site the teacher and the place owner, so each of them will have a different home page from the other. Therefore, it is important that we have an introductory page for the user to determine his job, whether as a teacher or property owner.
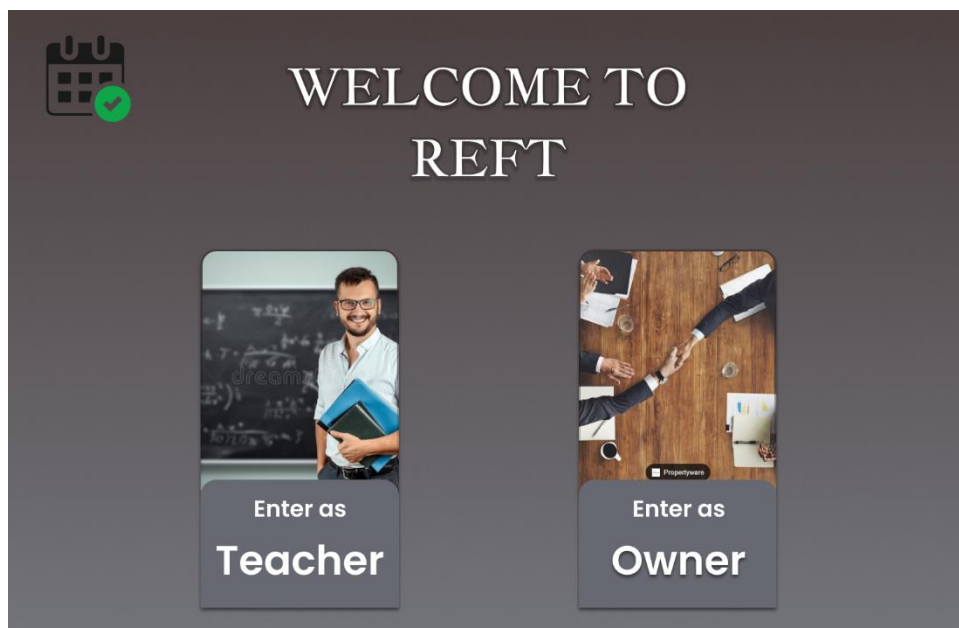
5.5 Main page for Owner

The home page of the place owner, which shows the places he has previously added, or by clicking on the (Add a hall) icon to turn it into a page where he fills out the site data to be reviewed by the administrator and then uploaded to the site.
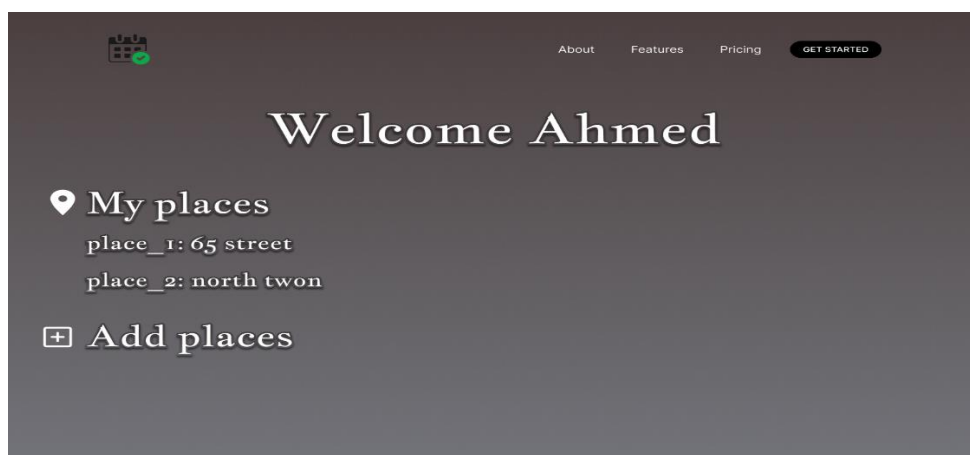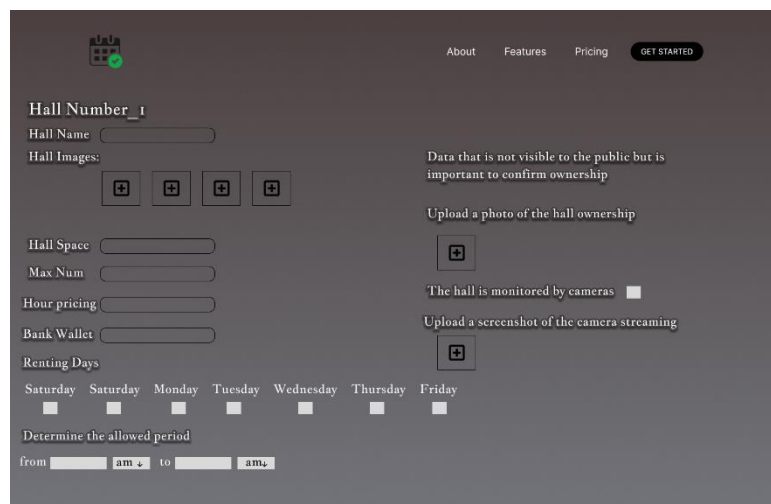
5.6 Add Hall page

Through this page, the owner can fill out some data and then upload it to the Revit website

There is some information the owner must fill to share his property

Such as:

1- First input field: used for entering hall name.

2- Second input field: used for upload hall pics.

3- Third input field: used for setting the hall space.

4- Fourth input field: used for setting the max number of attendees.

5- Fifth input field: used for determining the hour price.

6- sixth input field: used for putting his own bank wallet numbers.

7- seventh input field: used for determining the days he want the hall to be rented in

8- eighth input field: used for determining the allowed period of times at which the property is available to be rented.

9- ninth input field: used for uploading pics of ownership papers of that property.

10-tenth input field: is a necessary checkbox, it is very important to confirm the presence of surveillance cameras in the place, as it is not allowed to raise the hall without surveillance cameras

11-eleventh input field: uploading screenshots of surveillance cameras streams.

6- Results and Future Work

Results

The Reft website aims to allow teachers to rent suitable places for them to teach to students by renting the number of hours the teacher needs per day throughout the semester or the entire year, which contributes to reducing the teacher's expenses and thus reducing the prices of private lessons in our country, Egypt.

At the same time, it also aims to increase the profits of the property owner or hall owner by providing a larger number of teachers in the same place but distributed over the hours of the day.

Future work

There is a lot of work to be done to make Reft a great educational platform and the first thought for teachers and owners of educational properties.

These are some of the things that we will work on developing in the future.

1- Create an artificial intelligence bot that recommends places to the teacher based on questions he asks the teacher

2- Create a community for teachers to communicate with each other

III. Appendices

Appendix A

DVD Contents

The DVD of the project contains the following:

1. Full code of the web application

2. Documentation of the project in PDF format

3. Presentation of the project

Please note that you'll need to learn React and JavaScript programming languages to understand the code. You can view the official documentation of each language on its website on the internet or you can watch videos if you like (i.e. YouTube tutorial). You will also need to install Visual Studio Code on your machines to run the code.

IV. References

[1] Ikuomola A. J. and Asefon M.P. A Secured Mobile Cloud-based House Rental Management System. https://www.researchgate.net/publication/340926278_A_Secured_Mobile_Cloud-Based_House_Rental_Management_System

[2] Mandeep Katre (Professor), Kumar Abhay Gupta, Shipra Katiyar, Saumya Shahi, Shreya Awasthi. A Review Paper on PG Recommendation System. https://www.researchgate.net/publication/370944421_A_Review_Paper_on_PG_Recommendation_System

[3] Murahari Prithvi Yash, Chinmay Choudhary, Akanksha Lakra, Swati Dewangan. RentoAxis: Android App for Paying Guest Management. https://www.jetir.org/papers/JETIR1812A23.pdf

[4] F. H. Hassan,"Heuristic search methods and cellular automata modeling for layout design ," Ph.D dissertation, Sch. of Info. Sys, Comp. and Math., Brunel Univ., UK, 2013.

[5] S. Sarmady, F. Haron, and A. Z. H. Talib, "Modelling groups of pedestrians in least effort crowd movements using cellular automata," in Proc. 2009 2nd Asia International Conference on Modelling & Simulation, Bali, Indonesia, 2009, pp. 520-525.

[6] V. J. Blue, and J. L. Adler, "Cellular automata micro-simulation of bi-directional pedestrian flows," J. Transportation Research, pp. 135-141, 2000.

[7] Malan, R., & Bredemeyer, D. (2001). Functional requirements and use cases. Bredemeyer Consulting. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.436.4773&rep=rep1&type=p df

[8] Dinh-Trong, T. T., Ghosh, S., & France, R. B. (2006, November). A systematic approach to generate inputs to test UML design models. In 2006 17th International Symposium on Software Reliability Engineering (pp. 95-104). IEEE. https://ieeexplore.ieee.org/abstract/document/4021975

[9] React https://react.dev/blog/2023/03/16/introducing-react-dev

[10] Tailwind CSS
https://tailwindcss.com/

[11] Node.js
https://nodejs.org/

[11] Express.js
https://expressjs.com/