Eurasian National University named after L.N. Gumilyov

Faculty of «Information Technologies»

Department of «Information Systems»

****

**Project work:** «Rental housing management system»

**by disciplines**  «Object-oriented programming»

**Student:** Iskakov Y. K.

**Group:** IS

**Checked by:** prof. Zhukabayeva. T. K.

Abstract

This project is the final part of assessing the student on the disciplines of Object-oriented programming, by applying the acquired knowledge and skills in practice by java, where the student successfully demonstrated them when writing code, constructing flowcharts, creating a database, etc.

This project primarily focuses on maintaining house renting and establishing a connection between renters and clients. The system is designed to streamline these functions and allows to update the current data. Users can log into the system as a client and offer their suggestion.

We have developed a system that enables end-users to rent a house easily. In this project, we will discuss the implementation and development of the system and explain how it works in detail.

Acknowledgment

At the beginning, we would like to express our deepest gratitude to the prof. Tamara Kokenova, lecturer of this discipline. To do a small project, and finish it successfully, where we use tools of Information technologies and programming languages.

I tried to create a fully functional Java project, where I could take on a lot of IT roles – projects like these increase student’s knowledge and academic effectiveness.

Table of Contents

[Introduction 5](#_Toc164537847)

[**1.1 Background of the project** 5](#_Toc164537848)

[**1.2 The purpose of the project** 5](#_Toc164537849)

[**1.3 Project objectives** 5](#_Toc164537850)

[Methodology 6](#_Toc164537851)

[**3.1 Project planning using Jira** 6](#_Toc164537852)

[**3.2 System Design and Architecture** 7](#_Toc164537853)

[**3.3 Flowcharts and UML of future system** 7](#_Toc164537854)

[**3.4 PostgreSQL database** 8](#_Toc164537855)

[System implementation 11](#_Toc164537856)

[**4.1 Database connection** 11](#_Toc164537857)

[**4.2 Swing library** 11](#_Toc164537858)

[Results and discussions 12](#_Toc164537859)

[**5.1** **About system functionality** 12](#_Toc164537860)

[Conclusion 16](#_Toc164537861)

[Glossary 17](#_Toc164537862)

[References 18](#_Toc164537863)

**Design and Implementation of a pharmacy management system**

# Introduction

## **Background of the project**

## The need for a more structured and effective platform to oversee and improve house rental actions provided rise to the House Rental Management System. With demographic shifts and development on the rise, an efficient system to link tenants to landlords has become essential. Honesty, effectiveness, and convenience are often lack in traditional house hunting and rental methods. By offering a digital solution that makes the process easier for all parties involved, this project seeks to close this gap. By using technology, this system wants to emphasise all rental listings onto a single platform, simplifying the process for both homeowners and renters by allowing homeowners to easily connect with potential tenants.

## **The purpose of the project**

The House Rental Management System's primary aim is to provide a thorough, user-friendly platform that makes renting out houses easier. By enabling users to filter listings based on particular criteria like location, price, and number of rooms, this system seeks to make the process of looking for rental properties easier.

## **Project objectives**

User-friendly Interface: Using Java Swing, create a simple and attractive interface that is easy to use for all kinds of users.

Database Integration: Use PostgreSQL to build a strong backend that securely manages and stores user profiles, data from transactions, and house listings.

Functionality: Offer basic functions like user registration, login, filtering houses, making offers, purchasing properties, and price-based listing sorting.

# Methodology

## **3.1 Project planning using Jira**

Every project has its own start and end, but sometimes they don’t have precise beginning and ending dates (Lock, 2013). All projects are temporary and undertaken to create a product, service, or result, the period between the start and end of the project is referred to as the project life cycle (Figure 1.3). But it's relatively small to explain how the project works and identify phases. In that case, we use the Gantt chart. That can be used to set out the phases of a large capital project against the total life history timescale.

**Figure 1.**

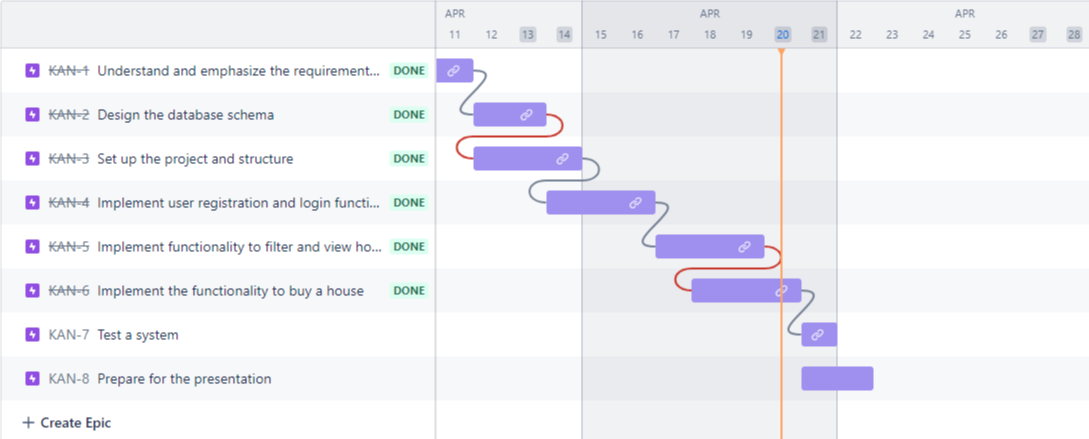
A project life cycle

Изображение выглядит как текст, диаграмма, круг, Шрифт

Автоматически созданное описание

To develop the project schedule, the project manager analyzes the project scope and purposes which helps him define the project deliverables. Based on it, we develop a milestone schedule, that establishes key dates to finish the project on time. For more complex, or just to plan succeeded projects, a more detailed schedule is required, which is shown in Table 1. As you can see, we have divided project tasks into three main blocks. After we create child issues.

**Figure 1.** Timeline of the project



*Note: This data will be shown in Gantts chart.*

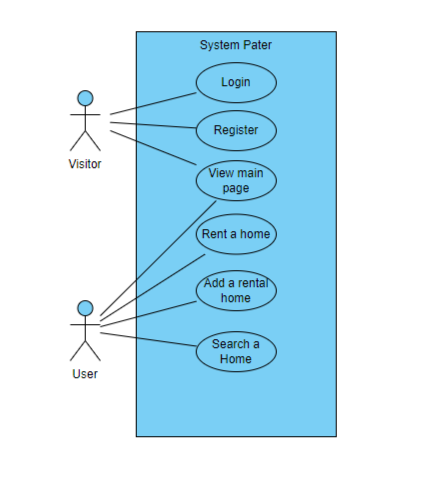
Although the project scope is a primary compass for developing it, it clarifies the project deliverables. The project plan lists activities that are needed to accomplish the work by dates. The more detailed planning, the more activities that are identified to accomplish the work. For this case, we use a widely used software development tool user for project management – ‘Jira6’. Particularly those practicing agile methodologies such as Scrum or Kanban, where we use Kanban agile (figure2).

## **3.2 System Design and Architecture**

### **3.3 Flowcharts and UML of future system**

**Figure 2**

A use case diagram



*Note: Model was developed using Visual Paradigm7*

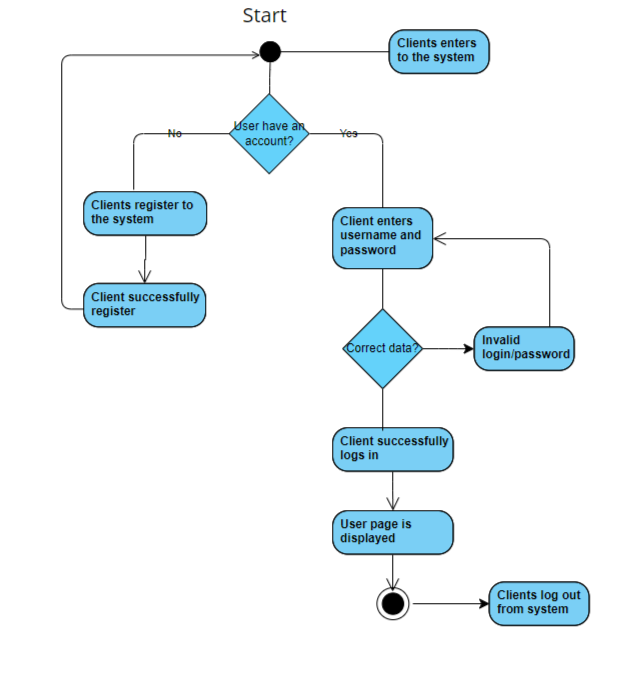
A flowchart is a type of diagram that represents an algorithm, works, and processes that display your system. It is also defined as step approach to solving a task. This is illustrating a solution model to given problem. They are especially used in analyzing, designing, or to manage processes.

First, a login request needs to make with a username and password, check from database. If they are valid, the login will be successful. Here, users will get individual profile access to rents. The user could read or add any other information that stores in the database, furthermore actions (figure 2)

Now, display user login activity using this model. This simple model almost describes most of systems.

**Figure 3**

Login activity of the client



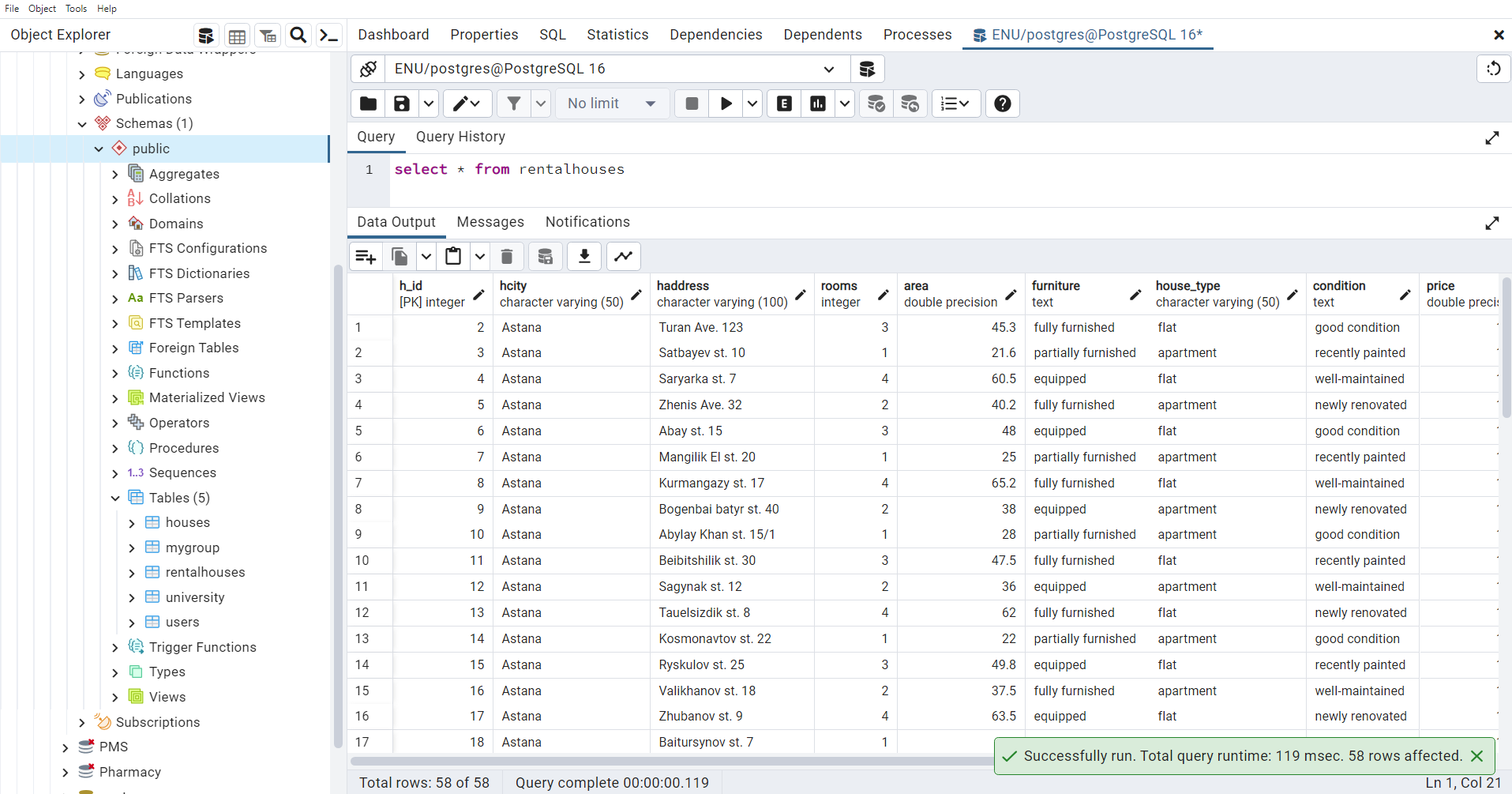
*Note: Model was developed using Visual Paradigm*

## **3.4 PostgreSQL database**

A database is an organized collection of structured information, or data, typically stored in the computer system. Ok, we found that its place where we store our data.

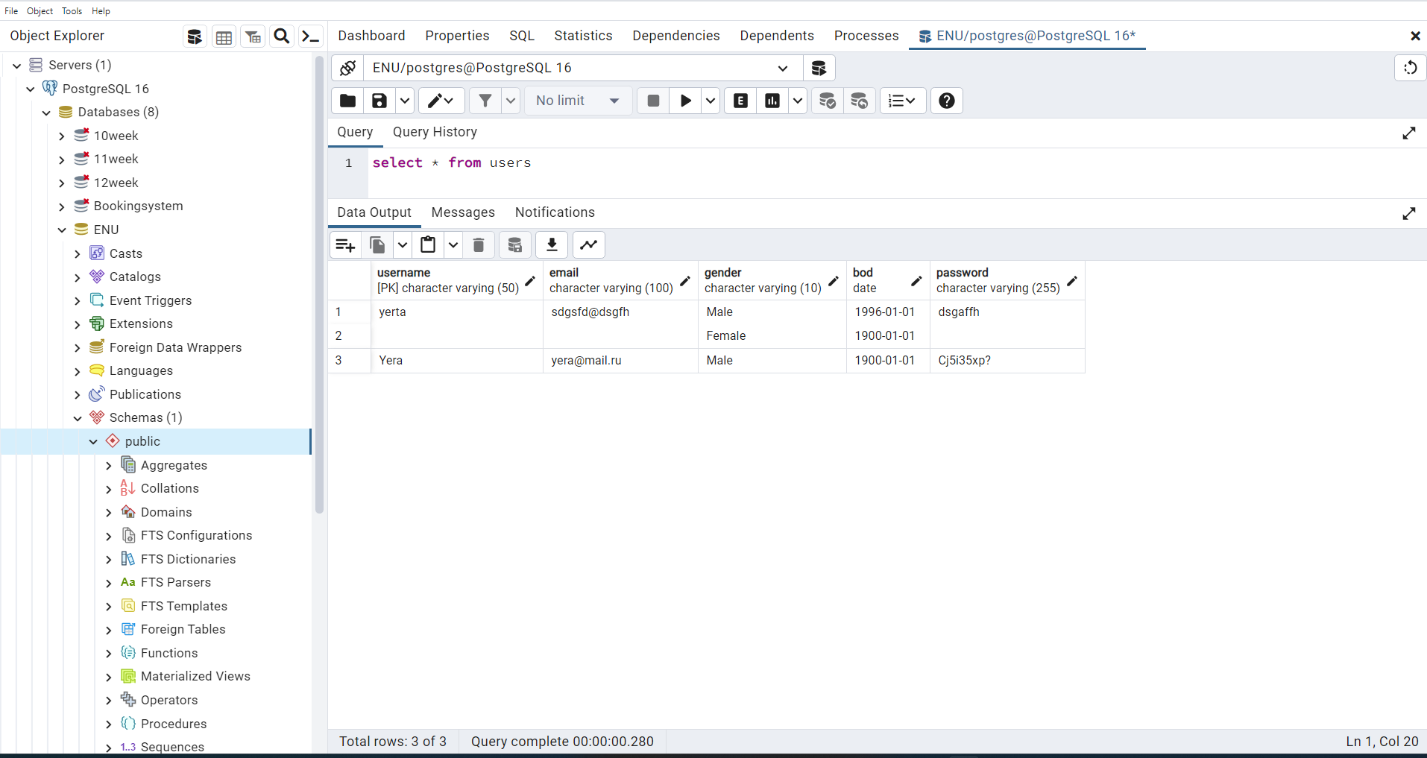
Most common, and user-friendly, free open sourced, has many useful features for developers database – PostgreSQL (figure 8). Compared with others, user could find and fix the error, after mistake is detected.

**Figure 4**

****

Interface of PostgreSQl database, rentalhouses table

**Figure 5**

****

Interface of PostgreSQl database, users table

*Note: For displaying functionality was created an option.*

To create a table in PostgreSQL, we use the ‘Create table’ statement. For example, lets create table called userdata:

Create table users (

User\_id Serial primary key,

Username varchar,

Email varchar,

Password varchar,

);

How to fill table with data? To fill the table with data, we use ‘Insert into’ statement.

Insert into userdata (username, email, password) Values (‘Aniya’, ‘San04@gmail.com’, ‘Reks23’

# System implementation

## **4.1 Database connection**

**Figure 6.1**

Function to connect database

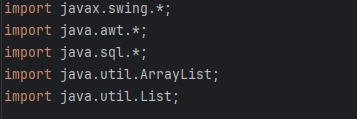
A computer screen shot of a program code

Description automatically generated

## **4.2 Swing library**

**Figure 6.2**

Libraries to import

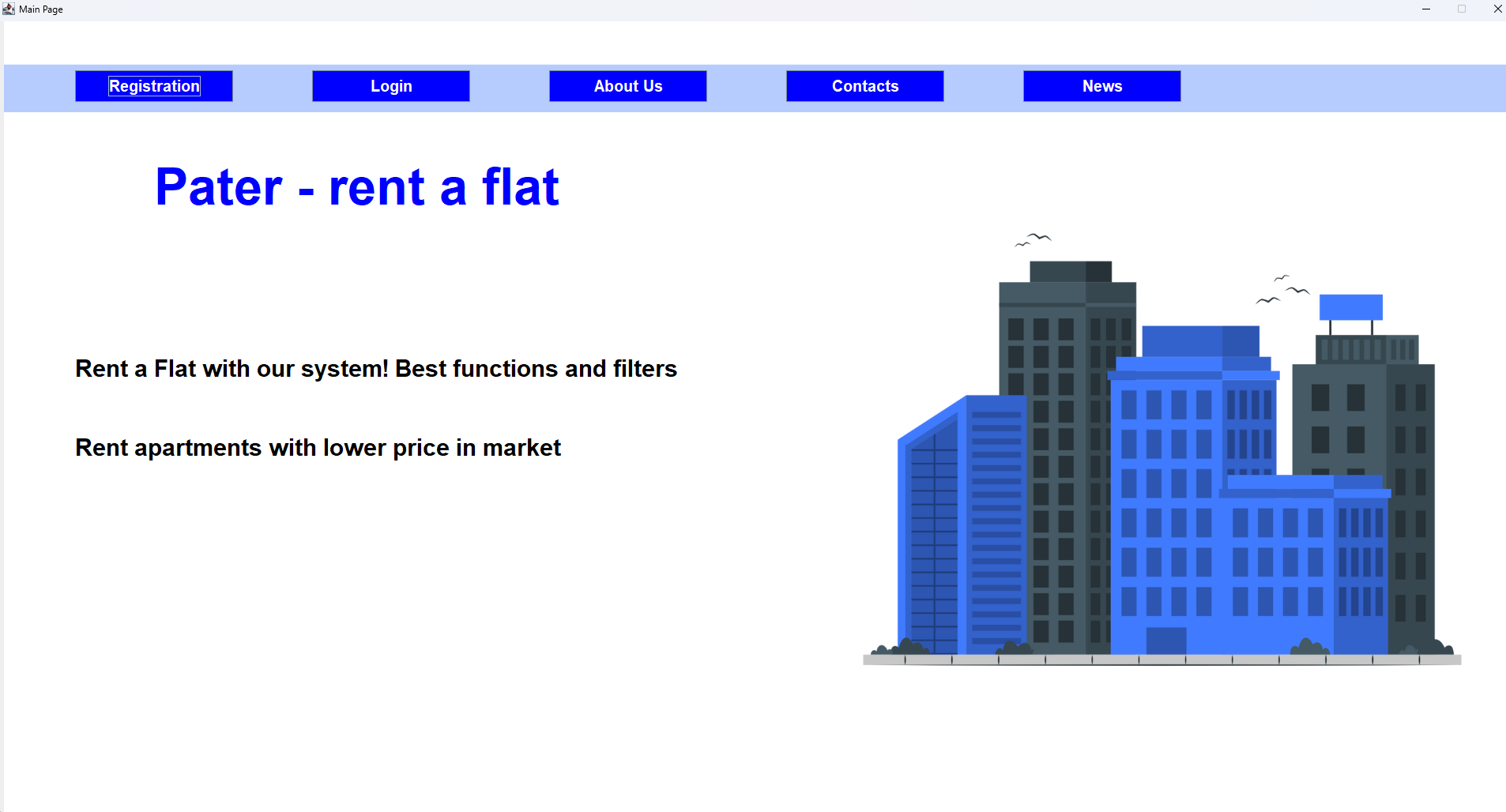


# Results and discussions

## **About system functionality**

**Figure 7**

Interface of the main page



Let’s consider now, the main page. Mostly functional page in this system.

1. Includes buttons for registration, login, and so on buttons.

- Navigable to other pages.

- Where user logs to the system and go to other pages.

**Figure 8**

Interface of the registration page

A computer screen with a login screen

Description automatically generated

**Figure 9**

Interface of the user page.

A computer screen with a login box

Description automatically generated

Lets consider now, login and registration page.

1. Login in functionality that enable user to enter his/her data.

* If the data match those in the database, user directed into next page.
* If the data are invalid or missing, appropriate error message are displayed.

1. Register.

* User can input their name, email, password, date of birth, gender etc and confirm the password. So while program check that all fields are required.

**Figure 10**

Interface of the About Us page.

A screenshot of a computer

Description automatically generated

Lets consider now, About us page.

1. About us page include information about our system.

**Figure 11**

Interface of the user page.

A screenshot of a computer

Description automatically generated

The provided page is an application for a house rental management system. It allows clients to filter and view available rental houses based on criteria such as street name, price, and number of rooms. Users can also submit their own house offers and buy houses listed on the platform. The application connects to a PostgreSQL database to store and retrieve house data.

# Conclusion

Projects are chosen for a variety of reasons and not all of them are apparent. The project manager (author of this work) must understand why a project was selected over other choices to work toward it. As project managers, we follow three main sections: explaining the purpose of the project, outlining workflow responsibilities, and presenting the completed work.

The project described a thorough effort to develop a productive and easy-to-use rent-a-house management system, combining the advantages of Java swing library for the graphical user interface and PostgreSQL database tools.

Main goal was to create a strong online system for ‘Pater’ that will enhance the management of data and operations in online systems. We have reviewed literature and past successes to determine the best path to reach our objective. We used Swing, PostgreSQL tools to create this system. We learned through implementation how to use codes correctly, what to add to the system, and even how to deploy this project.

We put all our attention into practice, where we learned through, but after it changed mistakes. Our essential element was to create a functional system that, that in result works

# Glossary

1. GUI are visual representation of communication that facilitates interaction with electronic devices.

2. PostgreSQL is a free and open-source relational database management system that emphasizes extensibility and SQL compliance.

3. Java Swing is the most common GUI library used to create GUI.

4. The Unified Modeling Language (UML) is a graphical language used to model, document, and visualize a system’s architecture.

5. Jira is a software development tool that is used for project management and issue tracking

6. Visual Paradigm is a tool used for creating and managing the UML, other cases

# References

1. Lock, D. (2013). Project management. Routledge.

2.Schmitz, A. (2012) Beginning Project Management. Creative commons.

3. Stored data in the github https://github.com/Yerassyl04/Rent-a-house-system

4. Stored project in the platform jira https://iskakovk2016.atlassian.net/jira/software/projects/KAN/boards/1/timeline?shared=&atlOrigin=eyJpIjoiZGMwYzU3MDQ5NzllNGJhZThjNjNkOGNiZDVlNzQ2NzAiLCJwIjoiaiJ9