

Immobi-Tec

Where Your Home Search Ends, And Your Story Begins.

**Table of Contents**

[***General Introduction*** 1](#_Toc134628782)

[***Chapter I: Preliminary Study*** 2](#_Toc134628783)

[**I.** **Introduction** 3](#_Toc134628784)

[**II.** **Presentation of the company** 3](#_Toc134628785)

[**III.** **Preliminary Study** 4](#_Toc134628786)

[**1.** **Problematic** 4](#_Toc134628787)

[**2.** **State of the art** 4](#_Toc134628788)

[**3.** **Critique** 5](#_Toc134628789)

[**4.** **Proposed Solution** 5](#_Toc134628790)

[**5.** **Goals** 5](#_Toc134628791)

[**IV.** **Conclusion** 6](#_Toc134628792)

[***Chapter II: Requirements specification*** 7](#_Toc134628793)

[**I.** **Introduction** 8](#_Toc134628794)

[**II.** **Functional requirement** 8](#_Toc134628795)

[**III.** **Nonfunctional requirement** 10](#_Toc134628796)

[**IV.** **Actor identification** 10](#_Toc134628797)

[**V.** **Use Case diagrams** 11](#_Toc134628798)

[**VI.** **Sequence diagram** 12](#_Toc134628799)

[**VII.** **Class diagram** 13](#_Toc134628800)

[**VIII.** **Architecture** 14](#_Toc134628801)

[**1.** **Technical architecture:** 14](#_Toc134628802)

[**2.** **Logical architecture:** 14](#_Toc134628803)

[**IX.** **Conclusion** 15](#_Toc134628804)

[***Chapter III: Implementation*** 16](#_Toc134628805)

[**I.** **Introduction** 17](#_Toc134628806)

[**II.** **Languages and Framework** 17](#_Toc134628807)

[**III.** **Methodology: The Unified Process** 17](#_Toc134628808)

[**I.** **Mockups** 18](#_Toc134628809)

[**IV.** **SDG’S** 19](#_Toc134628810)

[**V.** **Conclusion** 19](#_Toc134628811)

[***Chapter IV: Backend Development*** 20](#_Toc134628812)

[**I.** **Introduction** 21](#_Toc134628813)

[**II.** **Environment Setup** 21](#_Toc134628814)

[**III.** **Backend Architecture Design** 21](#_Toc134628815)

[**IV.** **Implementation of the Backend** 22](#_Toc134628816)

[**V.** **Testing API** 23](#_Toc134628817)

[**VI.** **Development** 27](#_Toc134628818)

[**VII.** **Conclusion** 27](#_Toc134628819)

[***Chapter V: Data Analysis*** 28](#_Toc134628820)

[**I.** **Introduction** 29](#_Toc134628821)

[**II.** **Business Understanding** 29](#_Toc134628822)

[**III.** **Data Understanding** 29](#_Toc134628823)

[**IV.** **DATA PREPARATION** 36](#_Toc134628824)

[**V.** **MODELING** 37](#_Toc134628825)

[**1.** **PREDICTION:** 37](#_Toc134628826)

[**2.** **SEGMENTATION** 39](#_Toc134628827)

[**3.** **Scoring** 46](#_Toc134628828)

[**VI.** **Conclusion** 47](#_Toc134628829)

[***Chapter VI: Front End Development*** 48](#_Toc134628830)

[**I.** **Introduction** 49](#_Toc134628831)

[**II.** **Environment Setup** 49](#_Toc134628834)

[**III.** **Implementation of the backend application** 49](#_Toc134628835)

[**IV.** **Interfaces** 50](#_Toc134628836)

[**V.** **Conclusion** 51](#_Toc134628837)

[***General Conclusion*** 52](#_Toc134628838)

**List of Figures**

[Figure 1 : Esprit-Tech logo 3](#_Toc134628839)

[Figure 2 : Global Use Case Diagram 11](#_Toc134628840)

[Figure 3 : Use Case Diagram of the User Management 12](#_Toc134628841)

[Figure 4 : Sequence Diagram of 'Authentication' 12](#_Toc134628842)

[Figure 5 : Class Diagram 13](#_Toc134628843)

[Figure 6 : Technical architecture 14](#_Toc134628844)

[Figure 7 : Logical architecture 14](#_Toc134628845)

[Figure 8 : Forum Prototype 18](#_Toc134628846)

[Figure 9 : Announcement prototype 18](#_Toc134628847)

[Figure 10 : Login Interface 50](#_Toc134628848)

[Figure 11 : List Product Interface 50](#_Toc134628849)

# ***General Introduction***

The real estate industry is one of the most dynamic and complex sectors of the global economy. The demand for real estate properties is constantly growing, and with the advent of technology, the way people buy, sell, and rent properties has drastically changed. Real estate websites have emerged as a popular platform that provides people with an easy and convenient way to browse, buy, sell, or rent properties.

This report will focus on a particular real estate website that offers a comprehensive range of features and services for its users. The website enables people to not only rent or buy real estate properties but also to sell their own property. Additionally, the website has features that allow users to predict the price of their property, manage dorms, participate in forums, auctions, and even provides ecommerce and service sections.

The website's comprehensive approach to real estate allows it to cater to a broad range of customers. The platform provides services to people who are looking for a new home or a property investment, as well as to those who are interested in selling their property or managing their dorm. Additionally, the website's auction feature allows people to buy or sell properties in a competitive and transparent environment, while the forum section enables users to discuss real estate trends and issues.

The report will provide an in-depth analysis of the website's key features, including its user interface, search functionality, property listings, pricing algorithms, dorm management, forum, auction, and ecommerce sections. The report will also evaluate the website's overall performance and user experience.

Finally, the report will highlight the website's strengths and weaknesses, and provide recommendations for potential improvements.

# ***Chapter I: Preliminary Study***

****

## **Introduction**

A preliminary study is a crucial step in any project, including the real estate website project discussed in this report. It helps to identify the objectives, scope, and requirements of the project, as well as potential challenges and opportunities.

The study aimed to gather information about the real estate industry and the needs of potential users. It assessed the competition, analyzed market trends, and evaluated the feasibility of the project. The findings were used to define the project's objectives and scope and develop a plan for its implementation.

Overall, the preliminary study was essential in ensuring the success of the real estate website project. It provided valuable insights into the market, user needs, and technical requirements, which helped to guide the project's development and implementation. This chapter will provide a detailed analysis of the preliminary study and its findings..

## **Presentation of the company**

**ESPRIT-Tech** was launched in 2010, it is the Research-Development-Innovation (RDI) structure at ESPRIT.

Since the establishment of training programs, the RDI activity at ESPRIT has become one of the priorities in the strategic choices.

Thanks to this entity, teacher-researchers at ESPRIT are interconnected and can benefit from funding and support opportunities, including grants, RDI training, partnership development and contract negotiation, as well as to commercialization ESPRIT-Tech develops strategies and future directions for RDI activities, in close collaboration with relevant stakeholders. In addition, it gives particular priority to applied research and innovation, without excluding fundamental research.



Figure 1 : Esprit-Tech logo

## **Preliminary Study**

### **Problematic**

The highly competitive nature of the real estate industry is one of the key problematic areas for the real estate website project.

The industry is crowded with many players, including established real estate agents and brokers, as well as new startups and online platforms. These players offer a range of services and have established a strong foothold in the market.

This presents significant challenges for the real estate website project in terms of attracting and retaining users.

To overcome this challenge, the project must differentiate itself by providing unique and valuable features and services that are not available on other platforms.

Additionally, the project must invest in effective marketing and branding strategies to raise awareness and establish a strong presence in the market. By addressing these challenges, the real estate website project can compete effectively in the industry and achieve its objectives.

### **State of the art**

There are plenty of real estate platforms that exist nowadays which will allow us to identify the problems that exist in them and their benefits.

In this part, we present some nationally and internationally known platforms based on their user interface, search efficiency, inventory size, accuracy, customer support, and reliability. We have chosen to present, in what follows, the best-known and most-used among these applications:

Nationally, the most used platform is:

* Immobiler.tn:

Immobilier.tn is a real estate platform in Tunisia , it allows the user to consult thousands of sale/rental offers, to sell and rent apartments, villas, farms, land, houses, and commercial premises, to post an ad, and to discover services.

Internationally, we can mention sites such as:

* Immobilier-france.fr:

Immobilier-France.fr was launched in 2006 and it is the main search engine for real estate in France: houses, apartments, building plots, garages, offices, shops, industries, new houses, and new apartments.

* Seloger.com:

The SeLoger Group is the marketplace of real estate portals in France that has been supporting and helping French people in the realization of all their real estate projects for 30 years.

The strong point of these is their price prediction system, their detailed search and filter criteria, and their user-friendly interfaces

### **Critique**

After analyzing the current situation, we realized that there are many factors that can make it easy for buyers, sellers, realtors, and landlords to find each other quickly and conveniently. One is to have software that connects all customers in one place and helps organize and improve efficiency.

The second is to meet user needs and maintain the authenticity of the product through a good user experience. The systems analyzed in the previous section certainly allow the management of estates but do not take into account the relationship between the seller and the buyer.

In addition, we noticed that there are some reported bugs with the filter function and the saved property function.

### **Proposed Solution**

Our goal is to create a real estate web application that can meet our needs. The solution will not only contain the basic features of a real estate application as seen in the critique of the current state.

But also, it will provide the client and the property owner with a communication tool that will allow them to communicate conveniently.

The application aims to offer clear information on each property to make for a smooth and pleasing user experience, and in the vain of making the said user experience even simpler, we aim to set up a discussion Forum.

### **Goals**

To achieve the desired outcomes, the preliminary study for the real estate website project was conducted with the following goals in mind.

* To provide users with a personalized experience on the website and streamline the registration and login process.
* To allow users to bid on and purchase properties or products in an efficient and transparent manner.
* To manage and list properties effectively, including property details, images, and other relevant information.
* To provide a range of services for users related to real estate, such as property management, maintenance, and making appointments.
* To manage dormitory properties effectively, including occupancy rates, rent collection, and reservations.
* To provide a secure and convenient payment method for users to pay for services, products, or properties.
* To provide a platform for users to buy and sell a variety of products related to real estate or other industries, which can also generate revenue for the website.
* To provide a platform for users to discuss real estate topics, ask questions, and share their experiences.

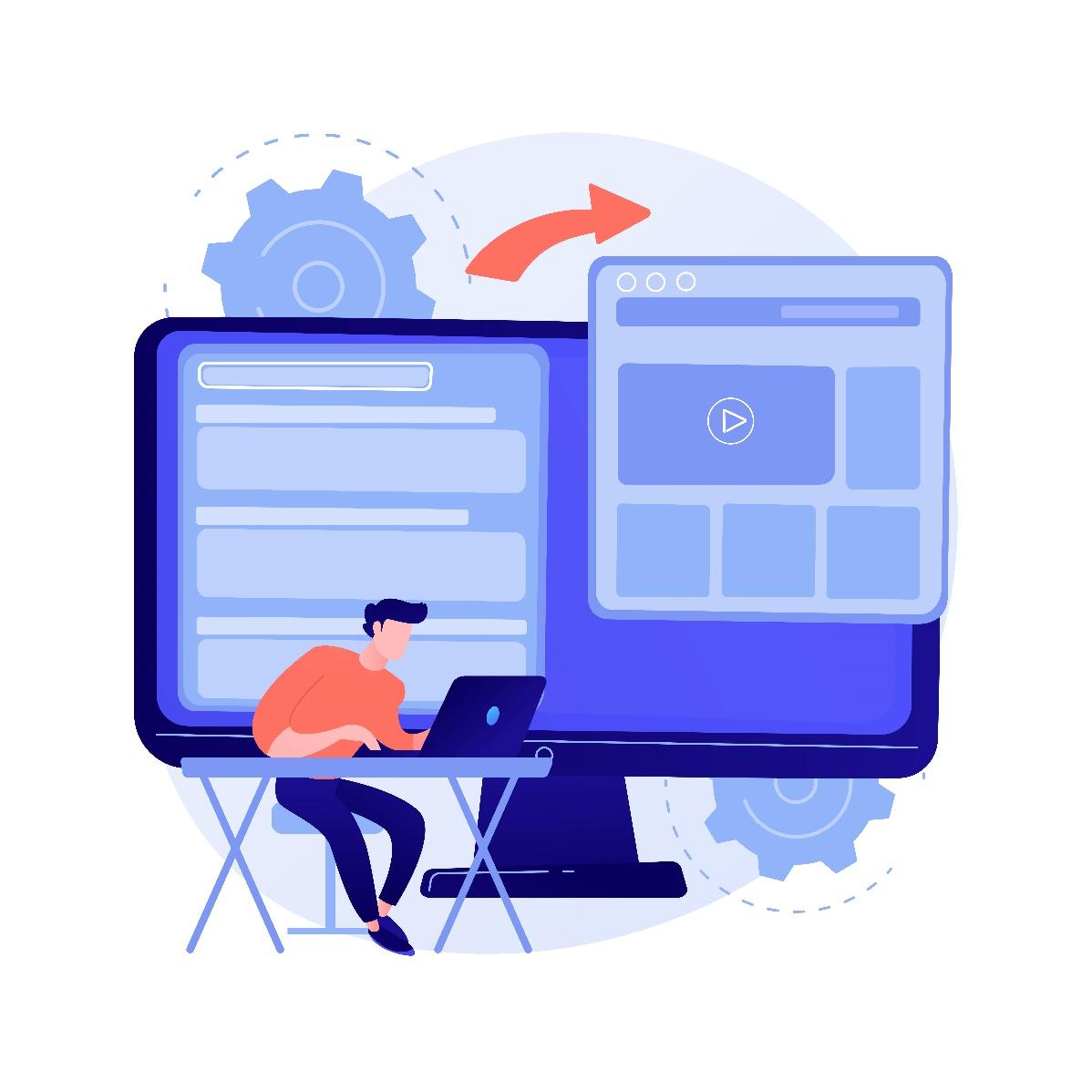
## **Conclusion**

The preliminary study was a crucial step in ensuring the success of the real estate website project. It provided valuable insights into the real estate industry, user needs, and technical requirements, which helped to guide the project's development and implementation.

The study identified the highly competitive nature of the real estate industry as a key problematic area for the project. To overcome this challenge, the project must differentiate itself by providing unique and valuable features and services and invest in effective marketing and branding strategies.

Overall, the preliminary study provided a solid foundation for the real estate website project, ensuring that it meets the needs of users and achieves its objectives. The insights gained from the study will guide the project's development and implementation, making it a success in the competitive real estate industry.

# ***Chapter II: Requirements specification***



## **Introduction**

The requirements specification phase is a crucial step in any project as it helps to define the specific needs and functionalities that the project must deliver. In the case of the real estate website project, this phase aims to identify and specify the features and functionalities that the website must offer to meet the needs of its users.

This chapter will provide a detailed analysis of the requirements specification phase for the real estate website project. It will outline the objectives of the phase, describe the methodologies used to gather and document requirements, and provide a detailed list of the requirements identified. Overall, the requirements specification phase is critical in ensuring that the real estate website project meets the needs of its users and delivers on its objectives.

## **Functional requirement**

Functional requirements, which describe the specific features and functionalities that the real estate website must provide to its users, will be identified and specified in this part to ensure that the project meets its objectives and meets the needs of its users.

Our application must allow the users to:

* Authenticate: Any user already registered in the system can authenticate and access the application using the email and password provided.
* User Management: The administrator of the application must be able to:
* Add a new user;
* Modify the role of a specific user;
* Delete users.
* Role Management: Only the administrator of the application must be able to:
* Add role.
* Modify role.
* Delete role.
* Property Management:
* The user will be able to add a new property.
* The user will be able to modify a property.
* The user will be able to delete a property.
* Announcement Management:
* The user will be able to consult the details of a specific announcement.
* The user will be able to add a new announcement.
* The user will be able to modify or delete an announcement.
* The user will be able to boost an announcement.
* The administrator will be able to validate or refuse an announcement.
* Store Management:
* The user will be able to consult all the products.
* The user will be able to place an order and to follow his orders.
* The user will be able to add a new product.
* The user will be able to modify a product.
* The user will be able to delete a product.
* Post Management:
* The user will be able to consult posts, add a new post, modify it, or delete it.
* The user will be able to interact with like, dislike, or comments on a post.
* Comment Management: The user will be able to consult the comments of a specific post, modify it, or delete it.
* Service Management:
* The user will be able to offer a service (moving furniture, painting houses...), modify it, or delete it.
* The user will be able to hire other users who offer services.
* The user will be able to make an appointment.
* Dorm Management:
* The user will be able to add dorm room to be rented.
* The user will be able to make online reservations.
* The user can give a feedback on the room he tried.
* Auction Management:
* The user will be able to set up an auction for their property.
* The user will be able to participate in existing auctions.
* Management of purchase/rental requests:
* The user will be able to apply to purchase/rent a property
* The user will be able to boost his proposal (optional)

## **Nonfunctional requirement**

Nonfunctional requirements, such as performance, scalability, security, and usability, will also be identified and specified in this part to ensure that the real estate website project meets the highest quality standards.

* Security: Using Spring Security will secure the authentication and protect data against attacks.
* Ergonomics: Producing a user-friendly application will allow clear and easy use for a broad audience of users.
* Performance: The system will return fast results.
* Reliability: The application must always be able to function correctly and the data provided must be reliable.

## **Actor identification**

An actor is a role played by anything that interacts with our system. It can be a physical person or even another system.

In addition, an actor can directly consult and/or modify the actions of our project by sending or receiving messages likely to carry data. In this project, the role specifies the right of access of each actor to system functionality.

These roles vary and depend on Esprit-Tech. For that, we have chosen six actors for our project.

The following are the actors of our system:

* The administrator has access to almost all the features of our application in addition to role management.
* The Visitor: The visitor is a non-registered user; he can only consult the announcements and register.
* The Subscriber: The subscriber is a registered user; he has a profile that will allow him to sell or purchase a property and to access the other various features of the application.
* The Agency Owner: The Agency Owner is the representative of an agency; he provides to the user the properties of his agency.
* The Dorm Owner: The dorm owner provides the subscribers with rooms that they can reserve.
* The Service Provider: The service provider is a normal user that can provide a type of service for other users

## **Use Case diagrams**

To visually represent the system's requirements and interactions, the use case diagrams will be used in this chapter.

* Global Use Case Diagram

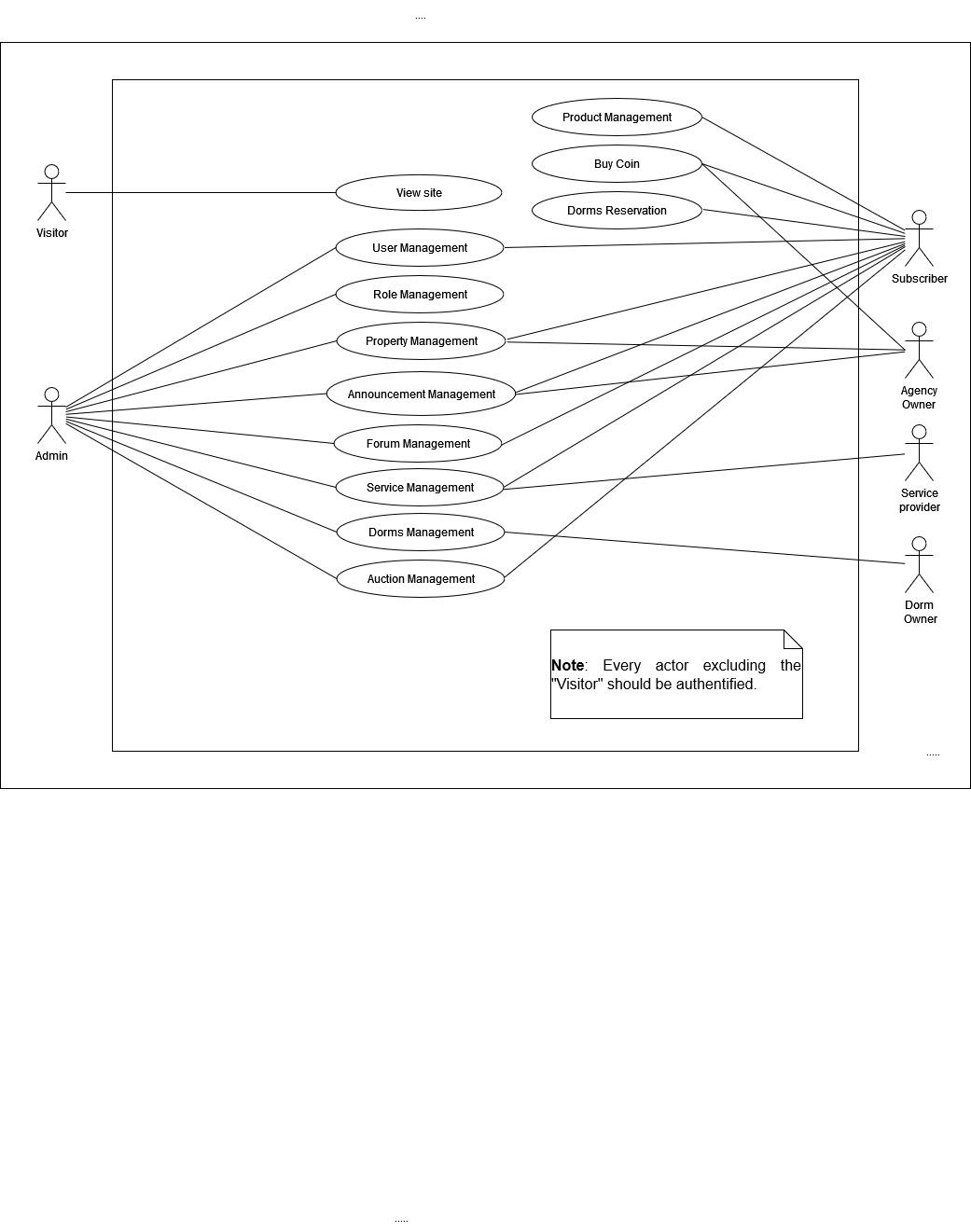
****

Figure 2 : Global Use Case Diagram

* Use Case Diagram of the User Management

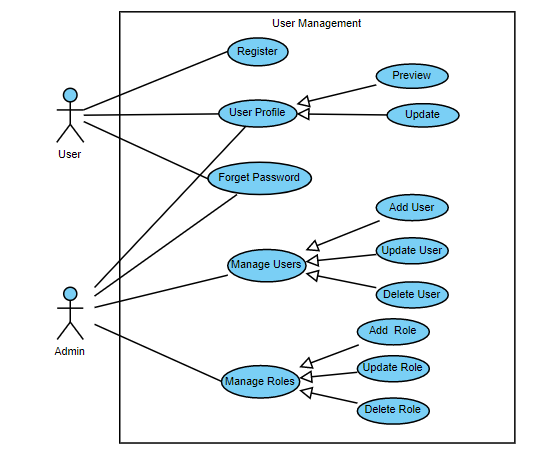


Figure 3 : Use Case Diagram of the User Management

## **Sequence diagram**

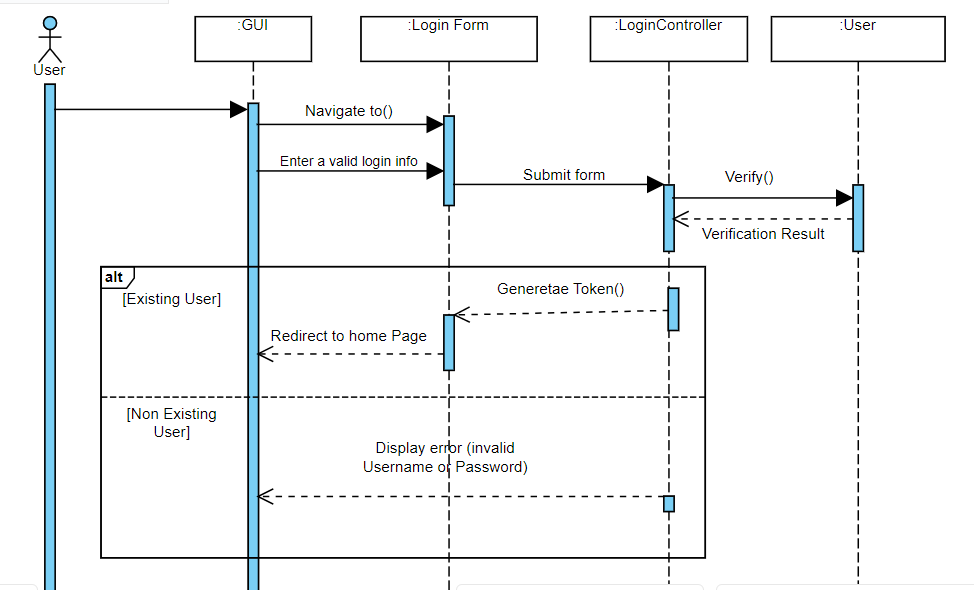
****

Figure 4 : Sequence Diagram of 'Authentication'

## **Class diagram**

Additionally, a class diagram will be used to illustrate the relationships between the system's classes and their attributes, methods, and associations.

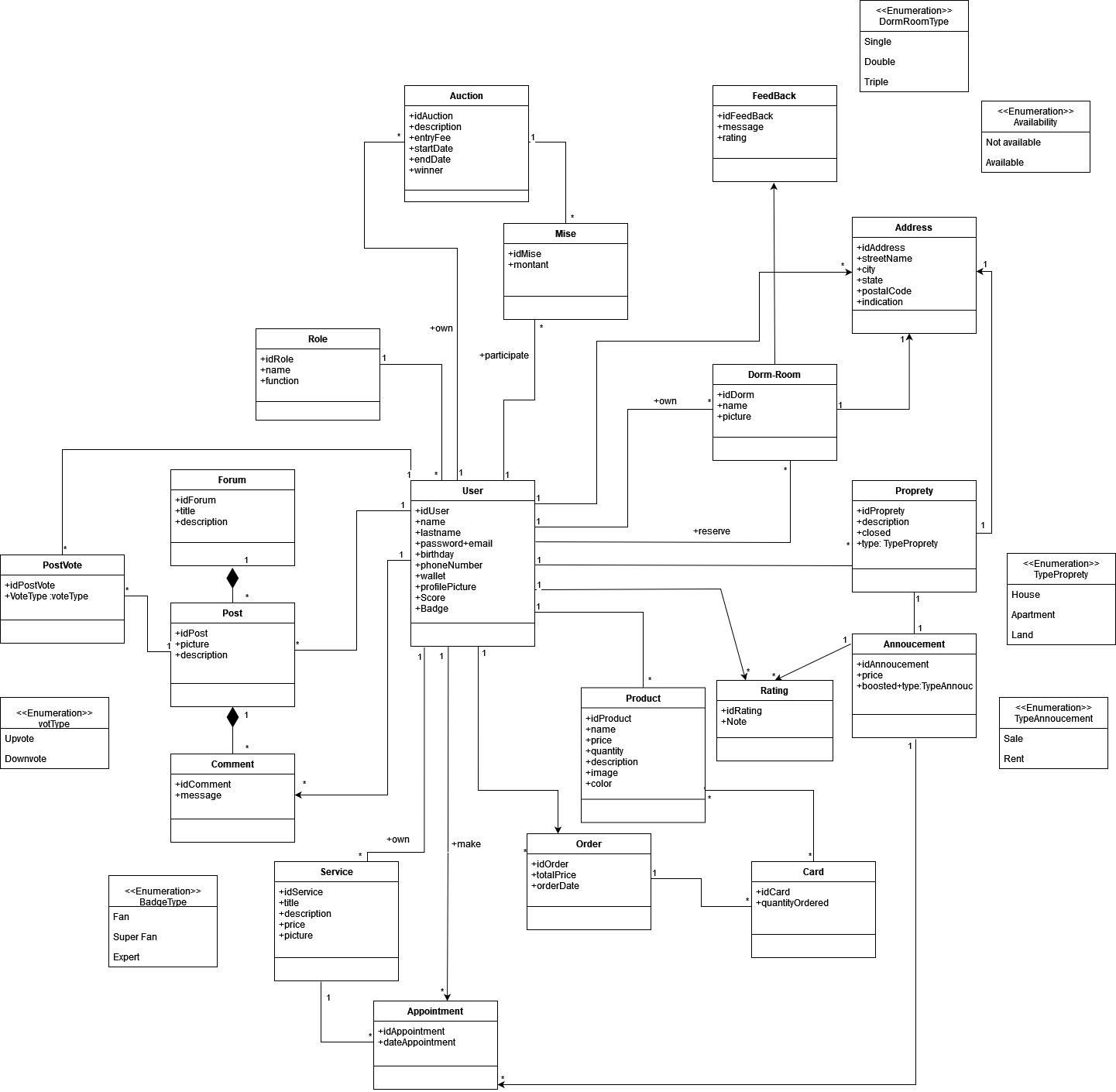
****

Figure 5 : Class Diagram

## **Architecture**

### **Technical architecture:**

The technical architecture of the real estate website project will also be described in this part, outlining the overall system's structure, components, and their interactions to achieve the project's objectives.

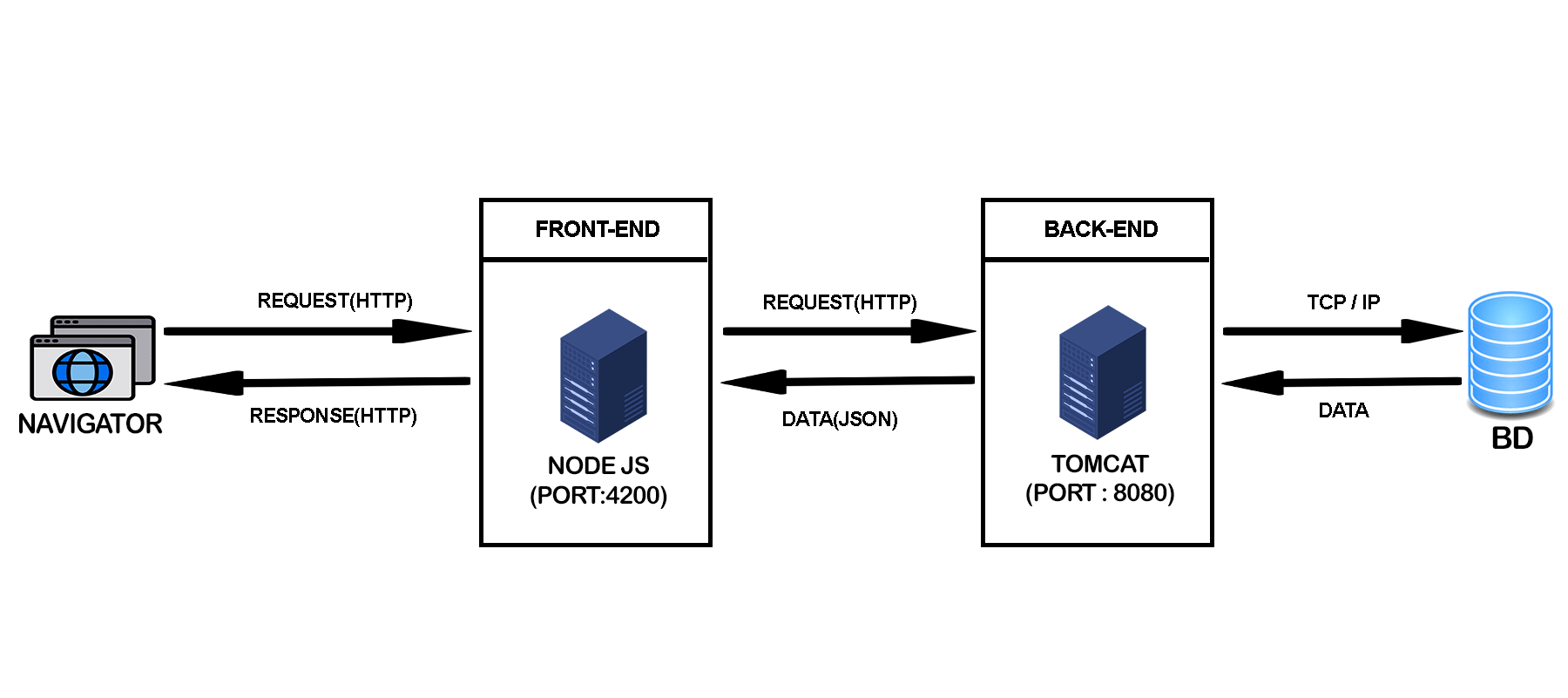
****

Figure 6 : Technical architecture

### **Logical architecture:**

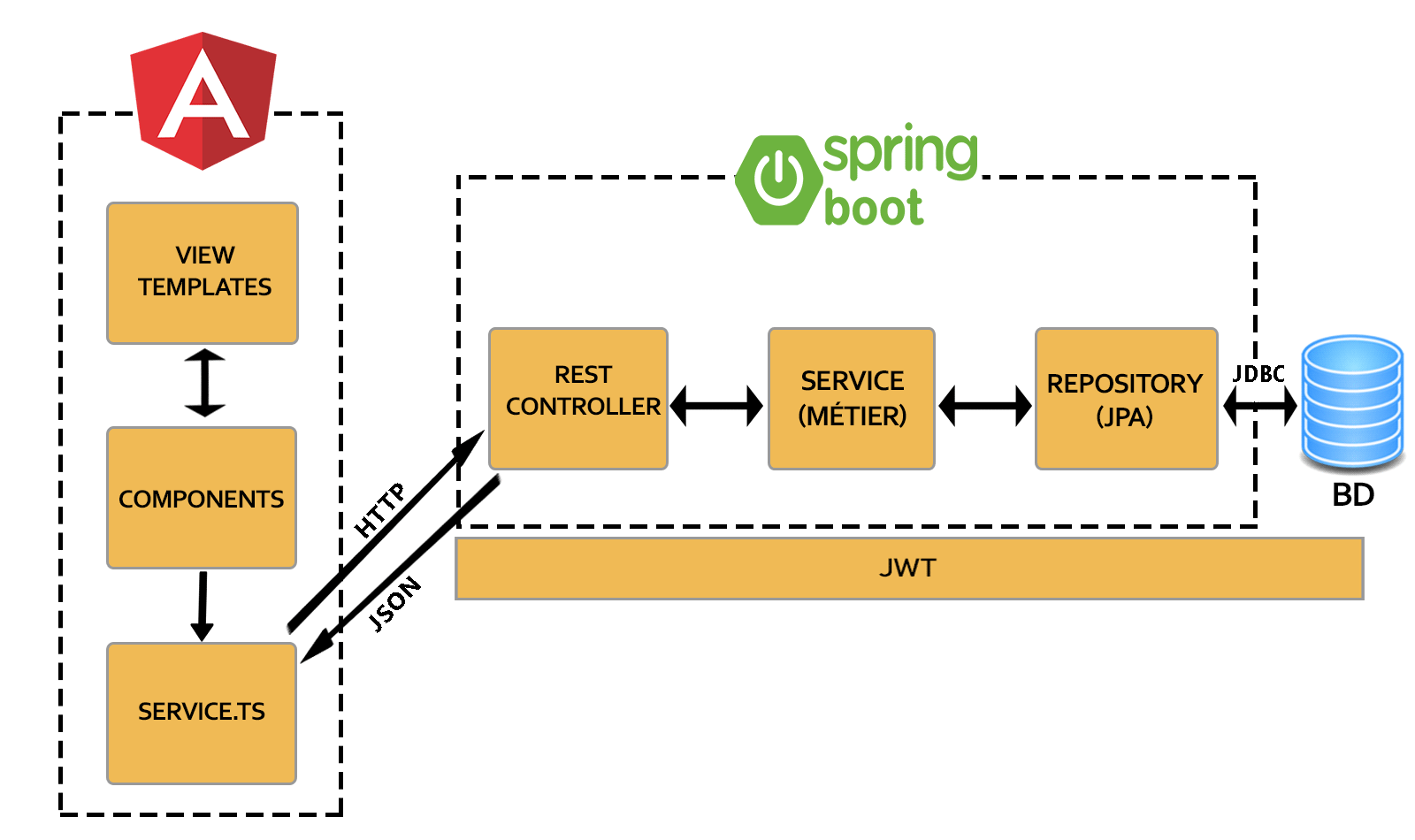
Furthermore, the logical architecture of the system, including the various layers and components that make up the system's functionality, will be described in detail in this part. ****

Figure 7 : Logical architecture

## **Conclusion**

In conclusion, the requirements specification chapter is a critical step in ensuring the success of the real estate website project.

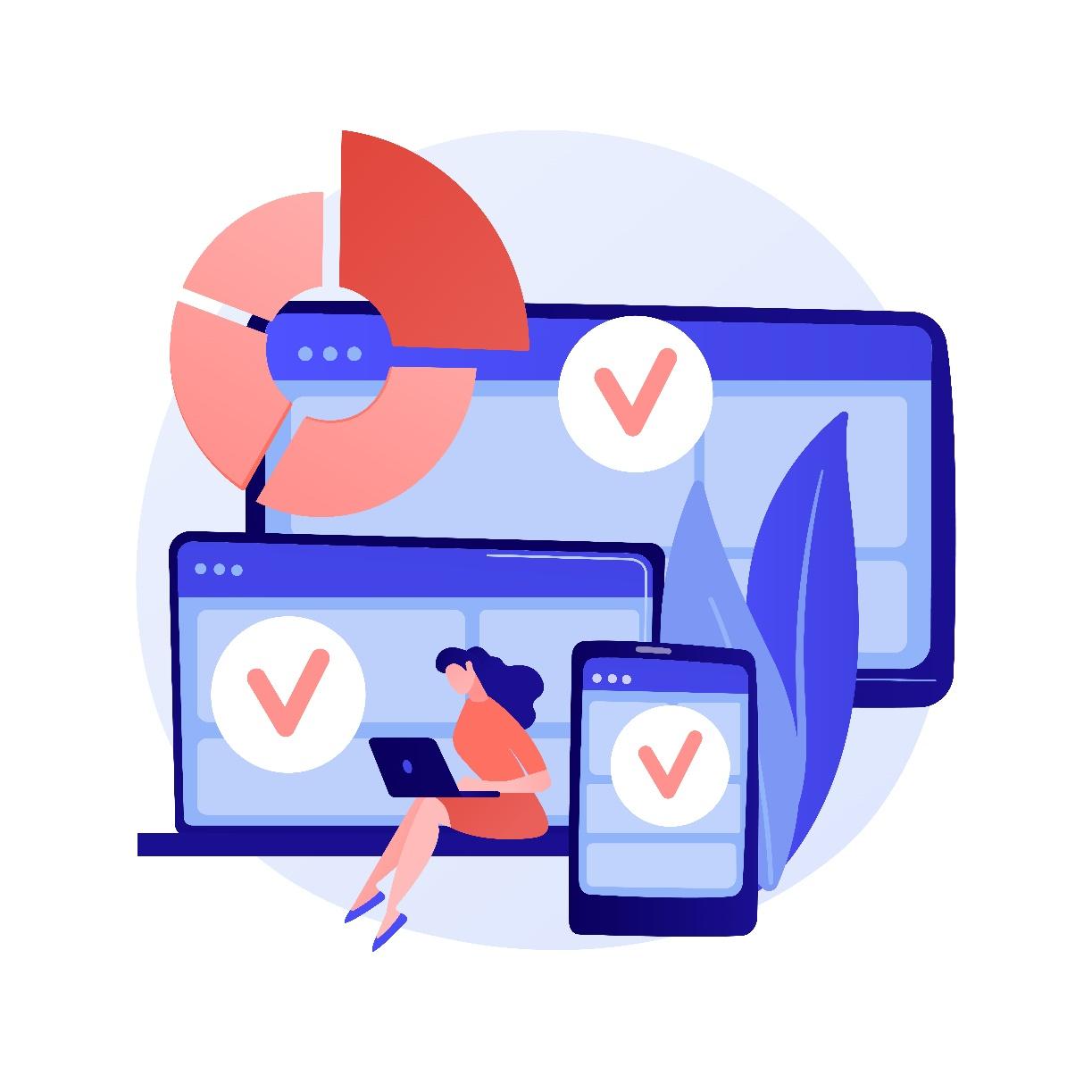
Through this chapter, we have identified and specified both functional and non-functional requirements, which will help to ensure that the final product meets user expectations and delivers a high-quality system.

Additionally, we have identified the various actors involved in the system and developed use case and class diagrams to help capture their interactions with the system.

The technical and logical architecture of the system have also been described, providing a solid foundation for the development and deployment of the system.

Overall, this chapter is crucial in laying the foundation for the successful implementation and deployment of the real estate website project, ensuring that the final product meets the project's objectives and delivers a high-quality system that meets the needs of its users.

# ***Chapter III: Implementation***

******

## **Introduction**

The “Implementation” chapter is a crucial section of the real estate website project report that provides an in-depth overview of the technical aspects of the system's development.

In this chapter, the various languages and frameworks used to develop the system will be presented, as well as the methodology employed during the development process.

Additionally, the chapter will outline the logical and technical architecture of the system, along with detailed descriptions of the various components and their interactions. Furthermore, mockups will be presented to provide a visual representation of the website's design and layout.

By providing this detailed technical information, this chapter serves as a critical reference for future developers and stakeholders who may need to understand the project's technical underpinnings.

## **Languages and Framework**

To develop the real estate website project, various languages and frameworks were used to ensure that the system meets the technical requirements and is scalable, secure, and efficient.

For the back-end development: Spring.

* Spring

For the front-end development: Angular.

* Angular

Database: A hosted MySQL database on Docker or a MySQL database in local.

## **Methodology: The Unified Process**

To ensure that the project was developed efficiently and effectively, a specific methodology was employed during the project's development, which will be outlined in this part.

The Unified Software Development Process (USDP) is a software development methodology that integrates the best practices from several other methodologies to create a streamlined and efficient process.

USDP emphasizes a focus on collaboration, communication, and continuous improvement, with the goal of delivering high-quality software in a timely and cost-effective manner.

The use of USDP can result in many advantages, including improved project outcomes, increased productivity, reduced development costs, and improved software quality.

### **Mockups**

Mockups, which are a visual representation of the website's design and layout, will also be presented in this chapter to provide a clear and accurate depiction of the website's interface and user experience.

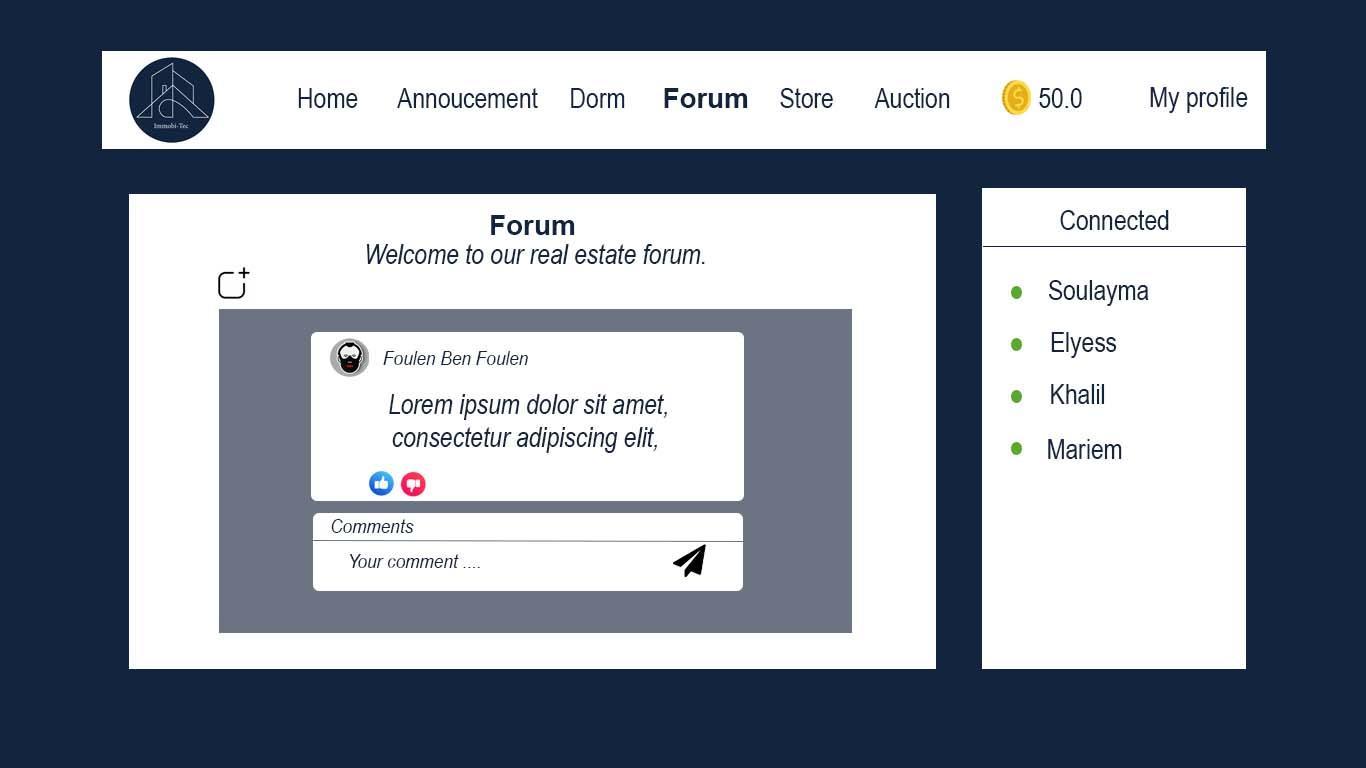
****

Figure 8 : Forum Prototype

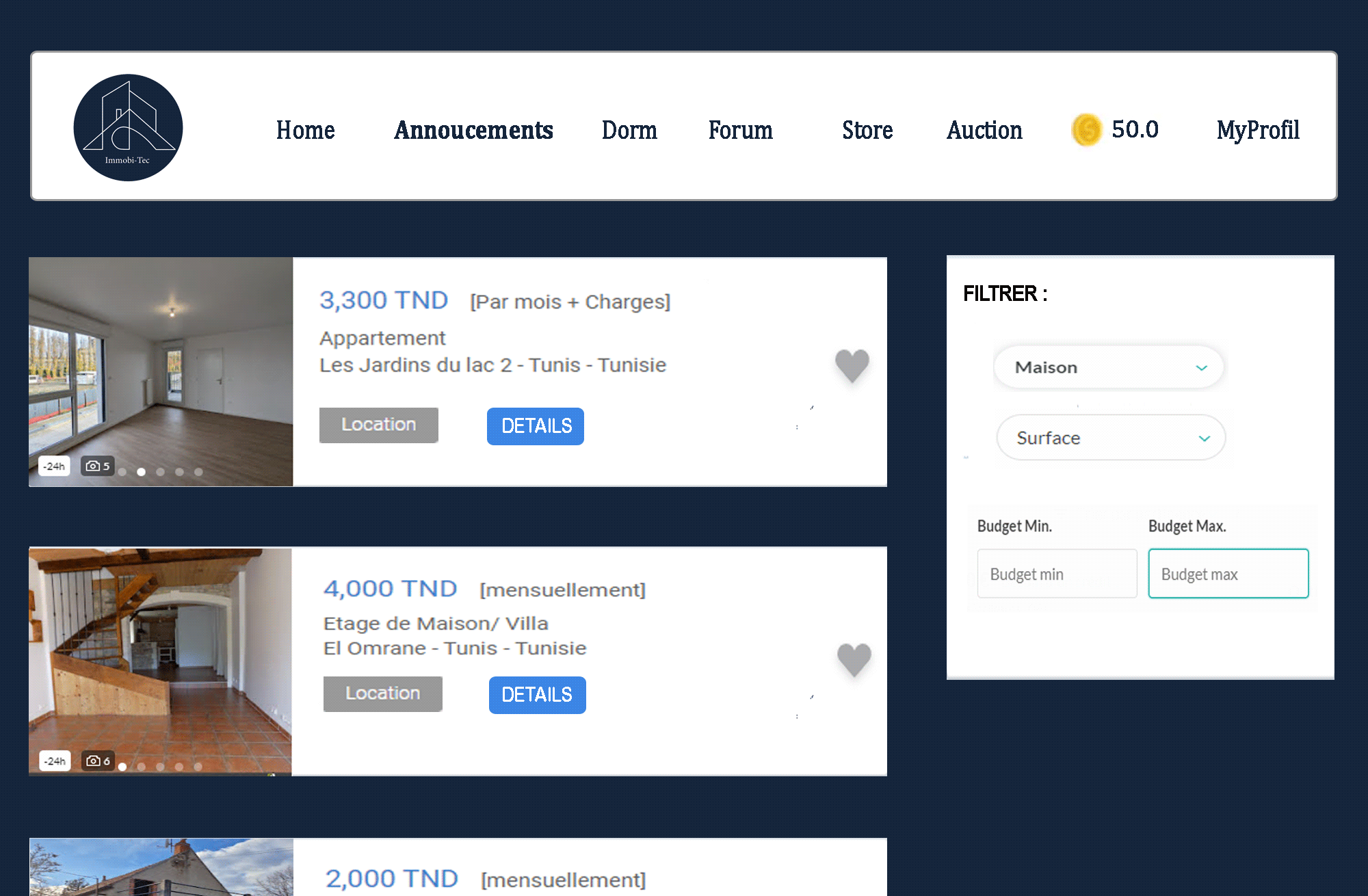
****

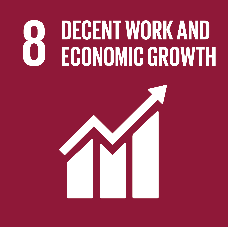
Figure 9 : Announcement prototype

## **SDG’S**

In addition to meeting user and technical requirements, the real estate website project also aims to contribute to sustainable development by addressing a specific sustainable development goal, which will be outlined in this part.

The primary sustainable development goal that the real estate website project aims to contribute to is SDG number eight: Decent Work and Economic Growth, which focuses on promoting sustained, inclusive, and sustainable economic growth, full and productive employment, and decent work for all.

In addition to SDG number eight, the real estate website project also aims to contribute to SDG number eleven: Sustainable cities and communities, which focuses on making cities and human settlements inclusive, safe, resilient, and sustainable. Specifically, the project aims to promote sustainable urbanization by providing a platform for individuals and businesses to buy, sell, and rent properties in a sustainable and responsible manner.



## **Conclusion**

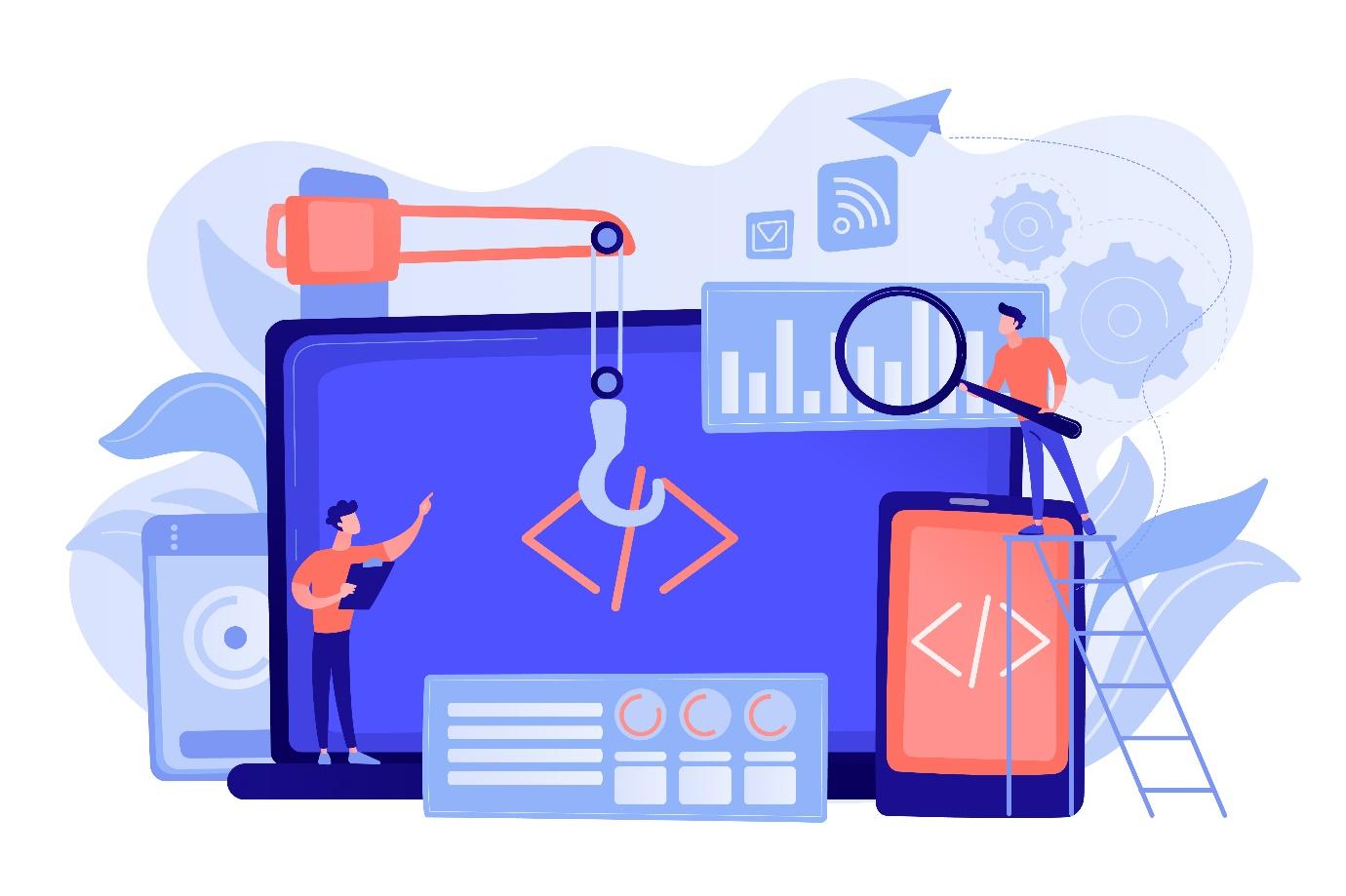
In conclusion, the "Implementation" chapter provides a comprehensive overview of the technical aspects of the real estate website project.

By detailing the various languages and frameworks used, the methodology followed, and the logical and technical architecture of the system, this chapter serves as a valuable reference for developers and stakeholders alike.

The mockups presented in this chapter also provide a clear and accurate depiction of the website's design and layout, which can help guide future development and design decisions.

Overall, this chapter showcases the technical prowess and attention to detail that went into the development of the real estate website project, and highlights the critical role that technical expertise plays in the successful implementation of complex systems.

# ***Chapter IV: Backend Development***

******

## **Introduction**

The backend development phase is a crucial aspect of any software development project, and it is particularly important for the real estate website project. In this chapter, we will focus on the development of the backend components of the system using the Spring Framework.

The Spring Framework is a widely used and highly regarded open-source framework for Java-based web applications. It provides a robust, flexible, and scalable platform for building enterprise-grade web applications that can handle large amounts of traffic and data.

In this chapter, we will explore the various components of the Spring Framework and how they can be used to develop the backend of the real estate website project. We will also discuss the database design and implementation, API development, and server-side programming languages and frameworks used to create a powerful and efficient backend system.

Overall, this chapter will provide a detailed overview of the backend development process and how it contributes to the overall success of the real estate website project.

## **Environment Setup**

This chapter covers the backend development phase of the project using Spring Boot. The first step involves installing the Java Development Kit (JDK) version 1.8 in our case, which is necessary to run Spring Boot applications. Next, we install an Integrated Development Environment (IDE) such as IntelliJ IDEA or Eclipse to provide a development environment for writing and testing the Spring Boot application.

The final step is to configure the Spring Framework and its dependencies, such as Spring Data JPA and Spring Security. This includes setting up the project structure, configuration files, and build tools like Maven in our case. Once the environment is configured, we can start building the backend of the real estate website.

## **Backend Architecture Design**

The backend architecture of our real estate application includes the use of different technologies and tools. For database management, we have the option to use MySQL locally, a MySQL database in a Docker container, or a MySQL database hosted in Azure. The choice of the database management system depends on the specific needs of the application and the scalability requirements.

In addition to the database management system, we also used Docker to containerize the application and its dependencies, allowing for easier deployment and management across different environments. We have also integrated various Spring modules, such as Spring Boot and Spring Data JPA, to facilitate the development and deployment of the application.

To ensure secure user authentication and authorization, we have implemented Spring Security in the backend architecture. This includes features such as password encryption, role-based access control, and session management.

Furthermore, we have integrated various third-party APIs and services, such as email notification APIs, payment gateways, and file storage services, to enhance the functionality of the application.

Overall, the backend architecture of our real estate application is designed to support the complex functionalities required by the application while ensuring efficient performance, security, and scalability. By leveraging different technologies and tools, we have created a robust and flexible architecture that can meet the changing needs of the application and its users.

## **Implementation of the Backend**

The development of the backend involves the implementation of the functionalities specified during the design phase. This involves the creation of different layers of the application, including controllers, services, DAOs, and others.

The controller layer is responsible for handling incoming HTTP requests and returning the appropriate HTTP responses. We have implemented this layer using Spring MVC, which provides powerful features such as request mapping, request validation, and others.

The service layer provides the business logic for the application. We have implemented this layer using the Spring Framework, which allows us to easily manage dependencies and perform transactions.

The DAO layer is responsible for interacting with the database to store and retrieve data. We have used Spring Data JPA, which provides an easy-to-use interface for working with the database and eliminates the need for writing complex SQL queries.

In addition to the above layers, we have also implemented other components such as exception handlers, security filters, and others to ensure efficient and secure functioning of the application.

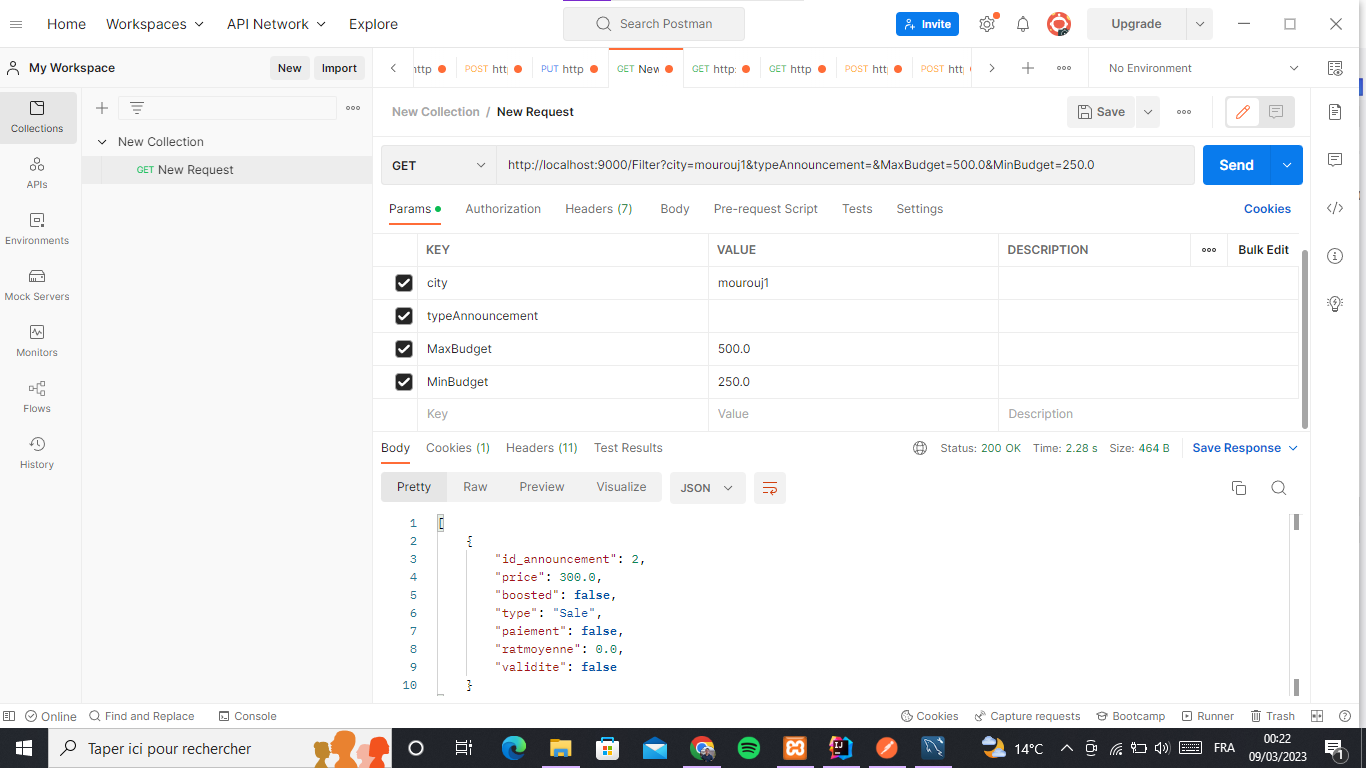
To ensure the quality of the code, we have implemented both unit tests and integration tests. Unit tests are used to test individual components of the application, while integration tests are used to test the interaction between different components.

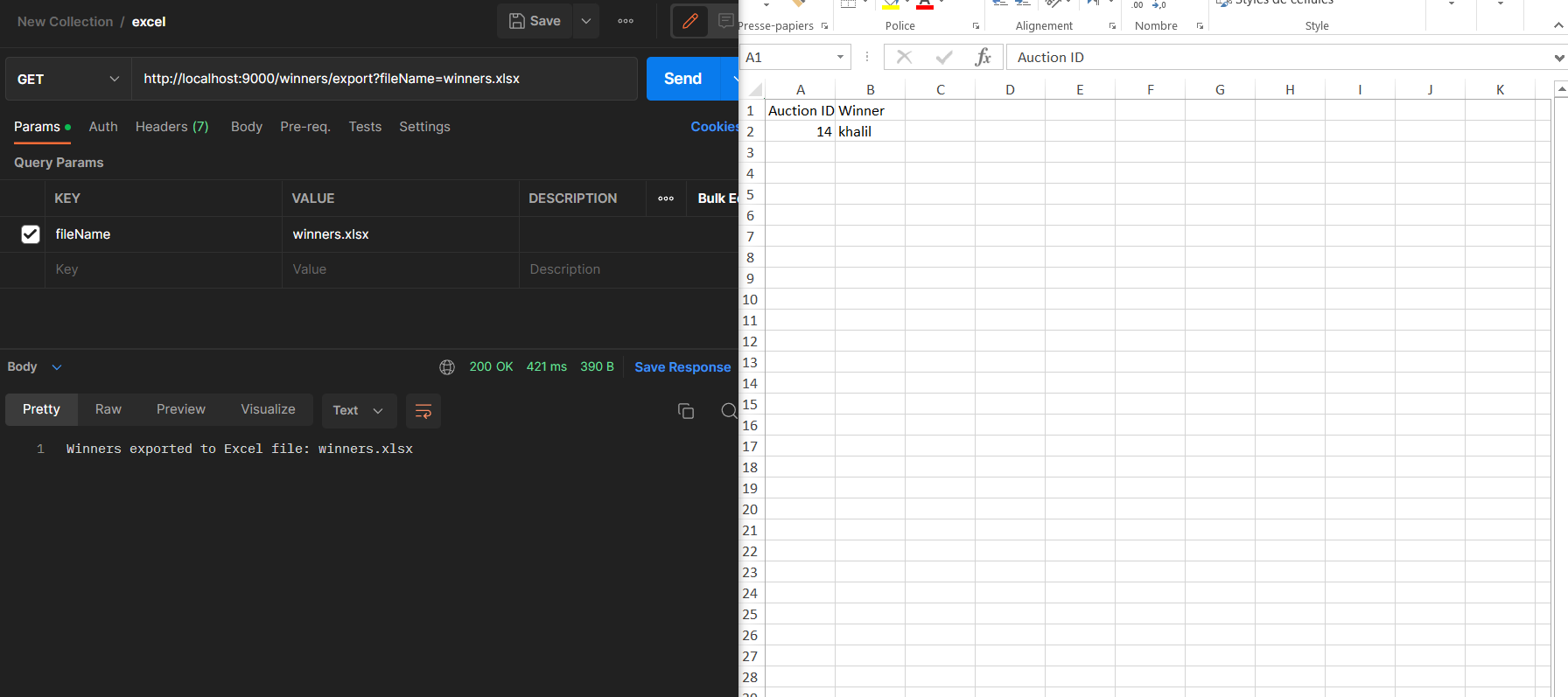
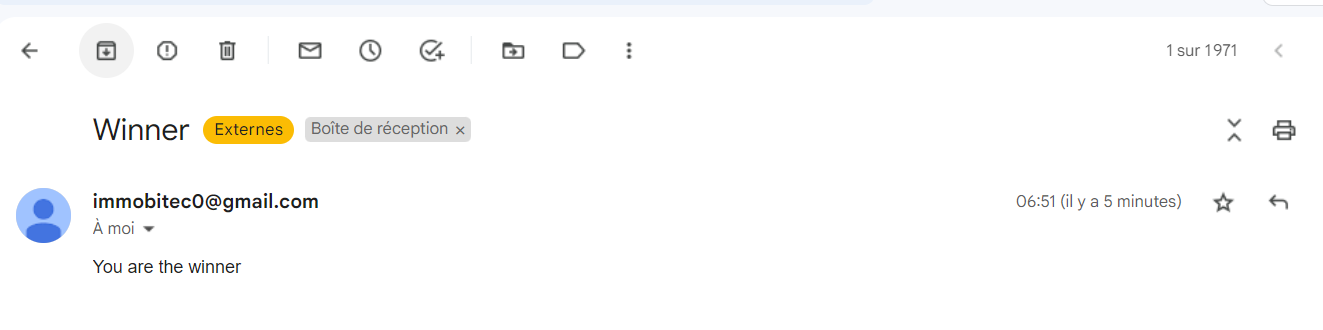
Overall, the development of the backend involves the implementation of the different layers and components of the application using various frameworks and libraries. It also involves thorough testing to ensure the efficient and secure functioning of the application.

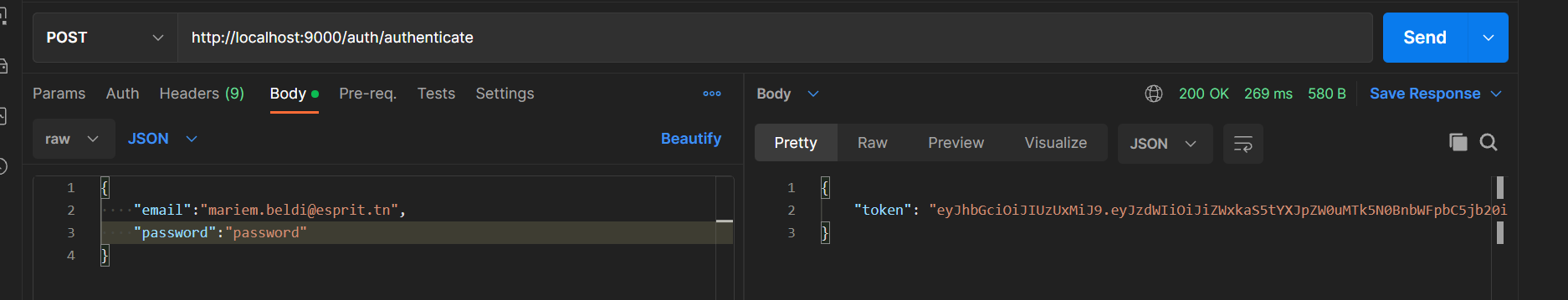
## **Testing API**

In software development, testing is a crucial step to ensure that the software is working as expected and meets the requirements. One popular tool for testing APIs is Postman. Postman provides a user-friendly interface for making API requests and testing their responses.

To demonstrate the use of Postman for API testing, we have captured some screenshots of sample requests and responses. In the following section, we will use these screenshots to illustrate how Postman can be used to test APIs and validate their responses.

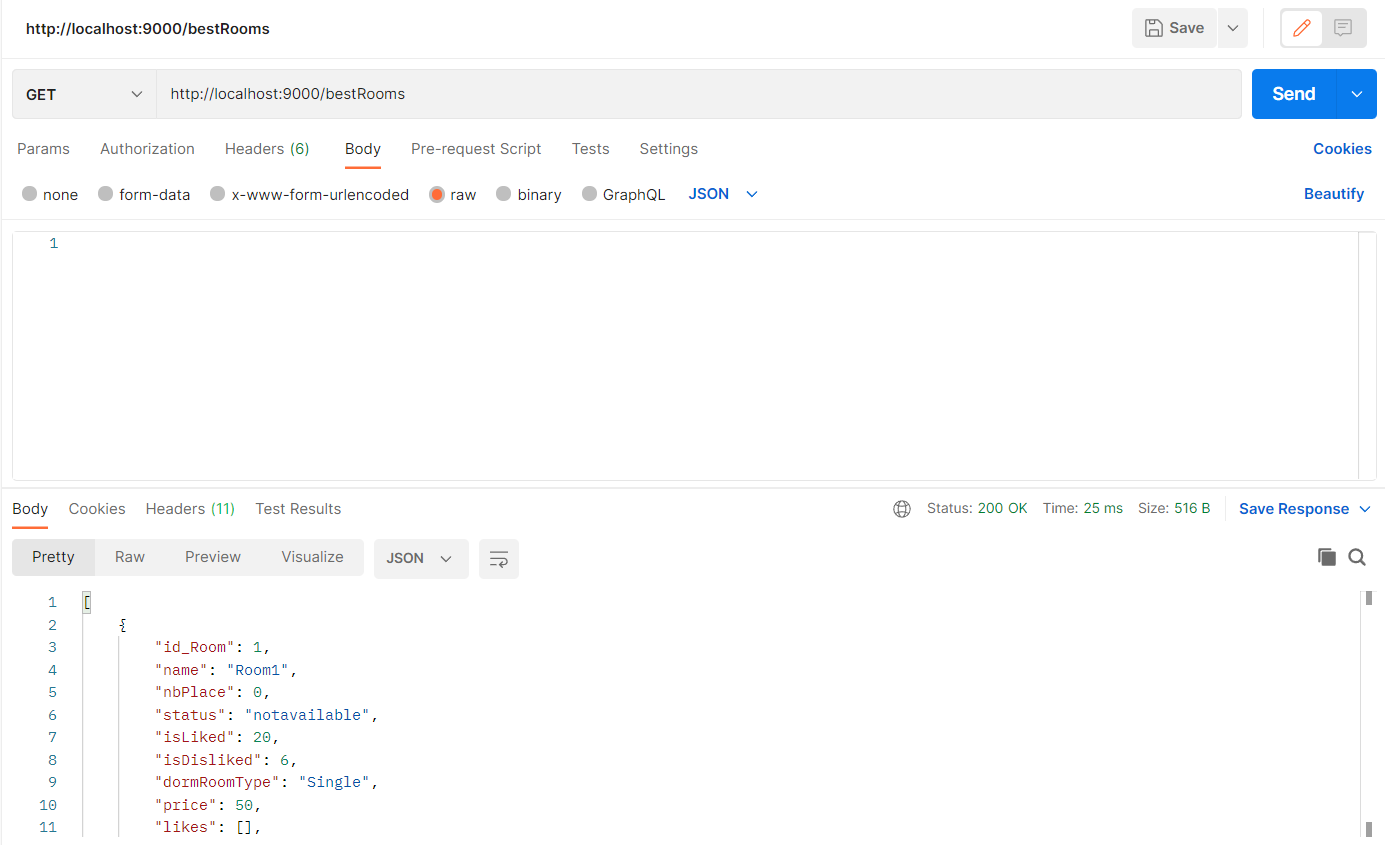


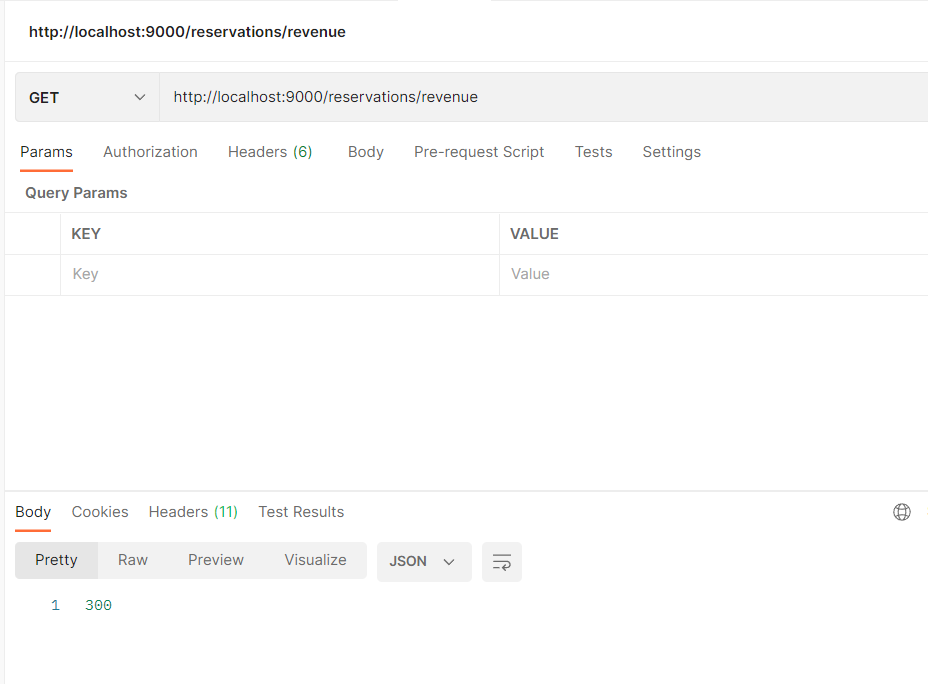


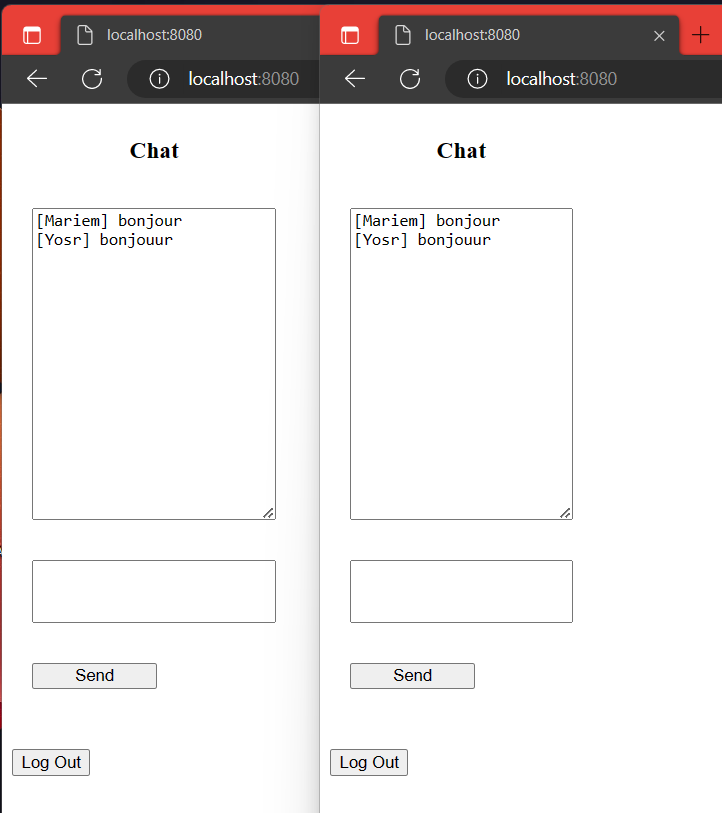


Service and appointment:

Dorm Rooms and Reservations:







### **Development**

Maven is a build automation tool for Java projects that manages dependencies and builds Java projects. Our POM file contains the configuration and metadata for our project called "Immobi-Tec". It specifies the project's group ID, artifact ID, and version, as well as its dependencies, including Spring Boot, Spring Security, MySQL Connector/J, Apache POI, and others.

The POM file also includes the version of Java required for the project, which is 1.8 in this case. Additionally, it includes dependencies for various libraries such as ModelMapper, SimpleCaptcha, JJWT, openpdf, Mapbox SDK Services, and Stripe Java, among others.

Finally, there are also test dependencies and optional dependencies such as Lombok. The project uses Springfox Swagger 2 and Swagger UI for API documentation.

## **Conclusion**

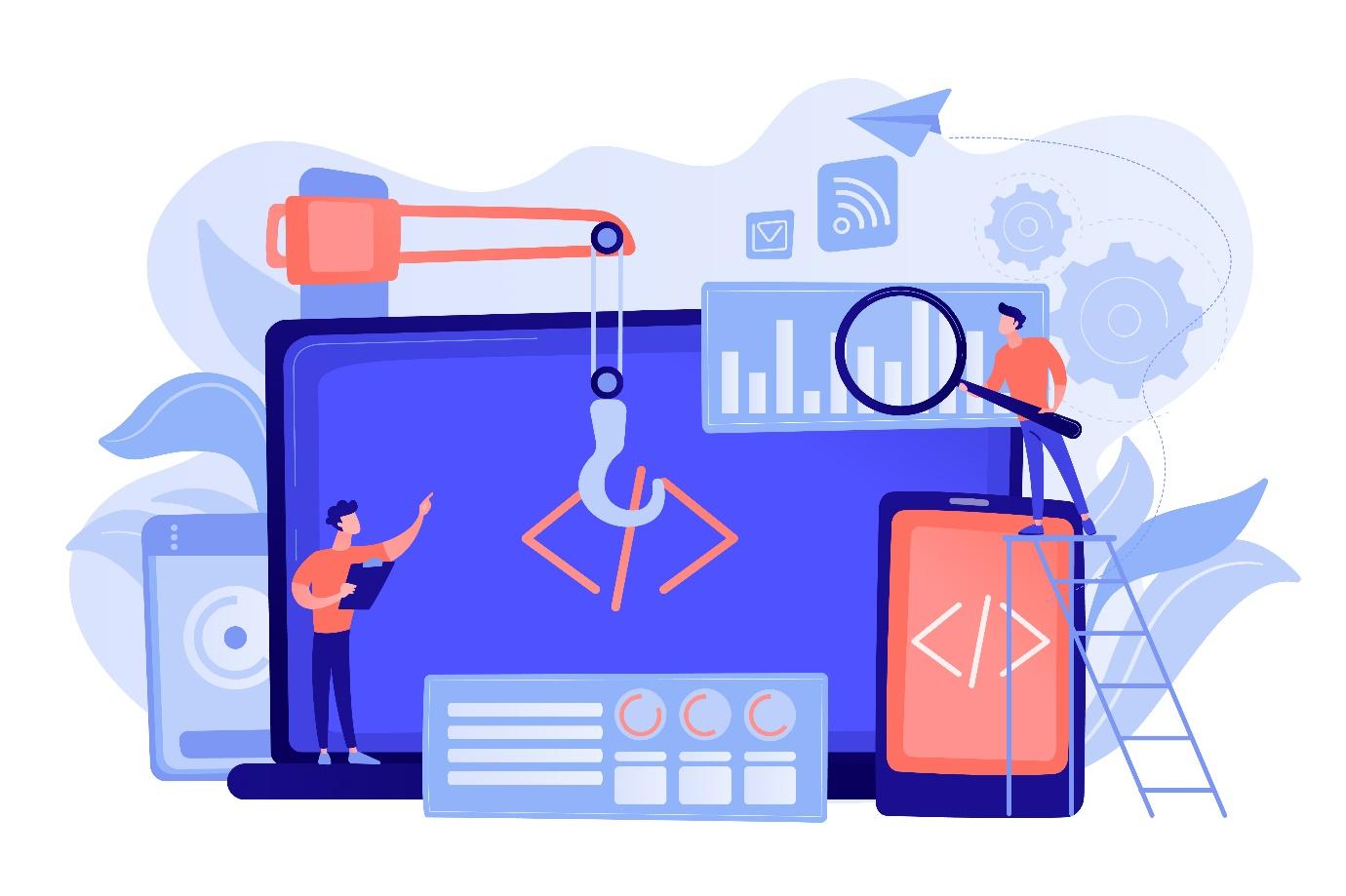
In conclusion, the backend development phase of the real estate website project has been a critical aspect of building a robust and scalable system. The Spring Framework has provided a solid foundation for the development of the backend components, including database design and implementation, API development, and server-side programming. The use of Spring Boot has allowed for a rapid and efficient development process, enabling us to deliver high-quality code in a timely manner.

Furthermore, the dependencies used in the development process have played a significant role in the success of the backend system. We have used a range of libraries and tools, including Maven for dependency management and Hibernate for object-relational mapping. These dependencies have helped to ensure the reliability, maintainability, and scalability of the backend system.

Part of the development process involved the implementation of the various backend features, such as user authentication and authorization, property listing management, and search functionality. We also implemented a RESTful API to allow for easy integration with the frontend components of the system.

Overall, the backend development phase has been a crucial part of the real estate website project, providing a solid foundation for the overall success of the system. The use of Spring Framework and its various components, as well as the dependencies and development tools used, have enabled us to build a reliable and scalable backend system that meets the functional and non-functional requirements of the project.

# ***Chapter V: Data Analysis***

******

## **Introduction**

Data mining is the process of extracting useful and insightful information from large volumes of data. It is a critical component of modern data analytics, and it has become increasingly important in a wide range of industries. The goal of data mining is to discover hidden patterns and relationships in data, which can then be used to make informed decisions and predictions. This chapter will provide an overview of data mining techniques, including data preprocessing, association rule mining, clustering, classification, and regression analysis. We will also discuss some of the common tools and platforms used in data mining, as well as some of the ethical and legal considerations associated with the practice.

## **Business Understanding**

The real estate market is a crucial sector of the economy, and being able to predict real estate prices accurately is essential for both buyers and sellers.

In this data science project, we aim to use the CRISP-DM methodology to develop a predictive model that can forecast real estate prices based on various factors such as location, size, amenities, and other relevant variables. Additionally, we intend to segment the properties into clusters based on their characteristics, enabling us to identify different groups of properties and develop specific strategies to target each segment.

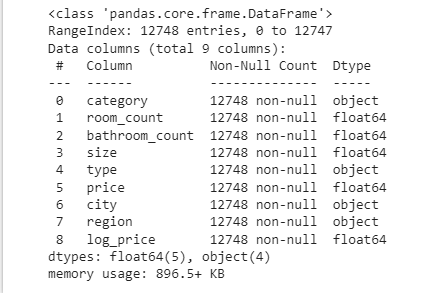
We will also develop a scoring system that assigns scores to each property, providing a quick and easy way for buyers to assess the quality and value of a given property. The insights obtained from this project will be useful for real estate agents, buyers, and sellers, allowing them to make more informed decisions based on data-driven insights.

Furthermore, the predictive model developed in this project could be extended to other real estate markets, providing valuable insights into property trends and helping stakeholders to stay ahead of the curve.

## **Data Understanding**

As shown in the screenshot, our data contains 12,748 rows and 9 columns. The columns are described as follows:

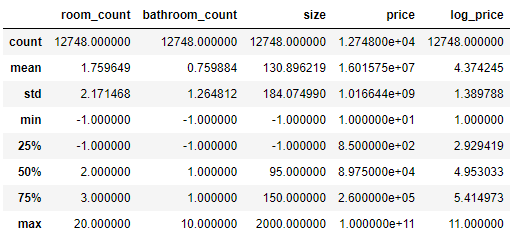
* Category: This column indicates the category of the real estate, such as apartments or villas.
* Room Count: This column indicates the number of rooms in a property.
* Bathroom Count: This column indicates the number of bathrooms in a property.
* Size: This column indicates the size of the property.
* Type: This column indicates whether a property is for sale or for rent.
* Price: This column indicates the price of a property.
* City: This column indicates the city where the property is located.
* Region: This column indicates the region where the property is located.
* Log\_price: This column indicates the logarithmic value of the price.



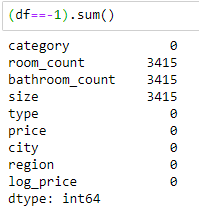
This screenshot displays the first five rows of our dataset, providing us with a glimpse of the information stored within them.



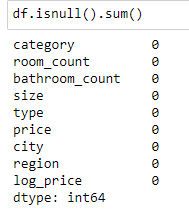
This set of screenshots provides us with a more detailed understanding of the numerical columns in our dataset. One significant observation is that some values in the data are represented as -1, which could potentially lead to issues in our analysis.



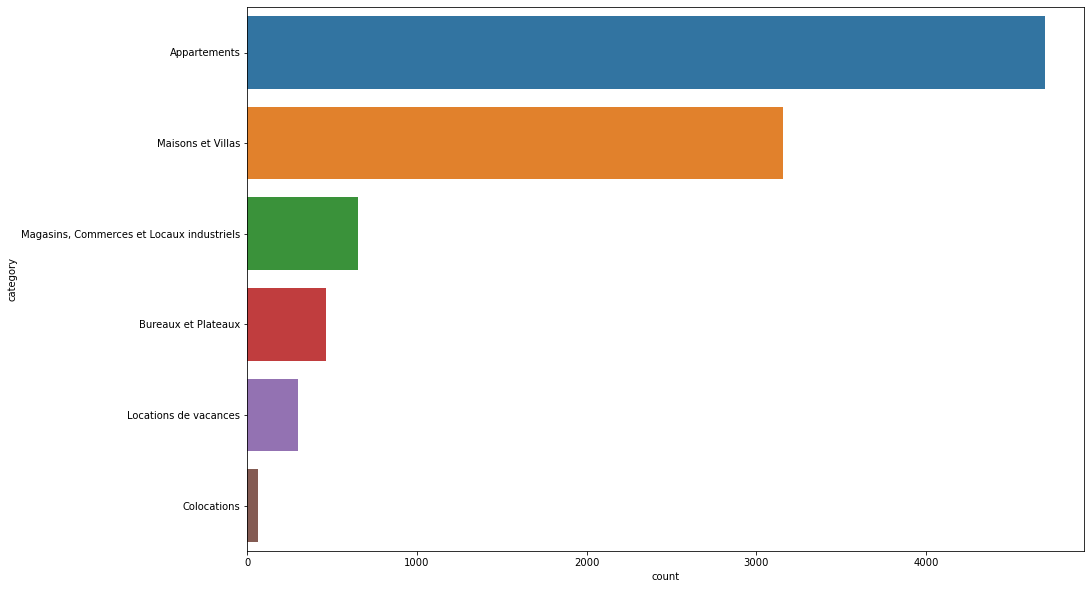
We performed a sum of the “-1” values in our data and obtained the upper screenshot. The results indicate that there are 3415 instances of “-1” values across our numerical columns.



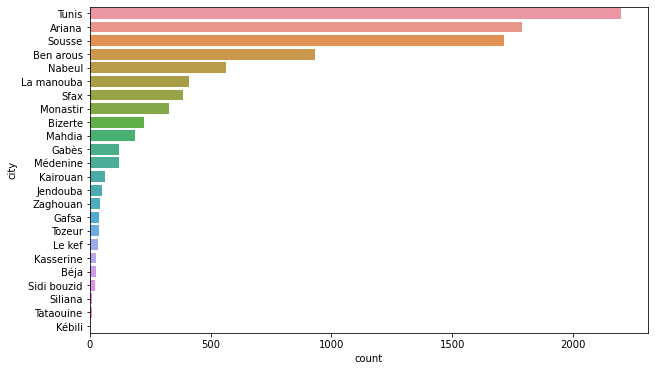
We calculated the number of null values in our dataset and found that all columns have a sum of 0, as shown in the screenshot.



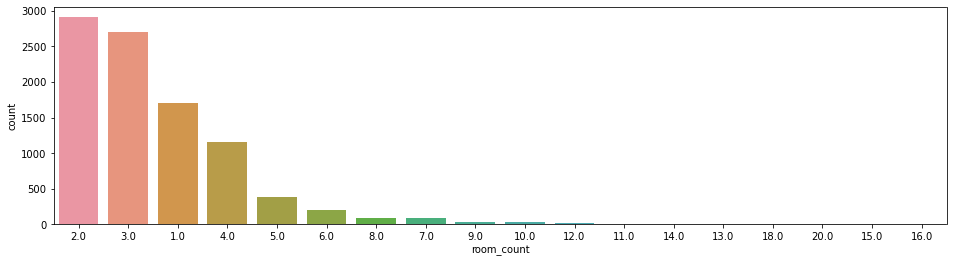
This figure shows a countplot of the various categories of real estate in our dataset. As illustrated, the majority of the properties are apartments, followed by houses and villas.



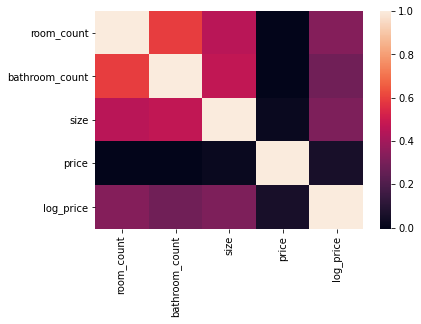
This countplot depicts the distribution of real estates across different cities and regions. The top three regions with the highest number of real estates are Tunis, Ariana, and Sousse governorates.



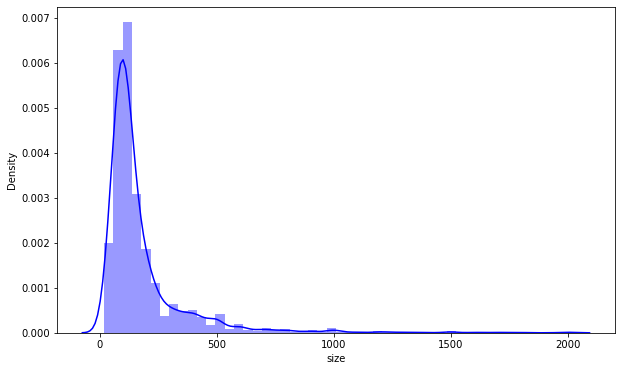
This countplot provides insight into the distribution of the number of rooms per real estate in our dataset. As can be seen, the majority of the properties have either 2 or 3 rooms, followed by single-room properties.



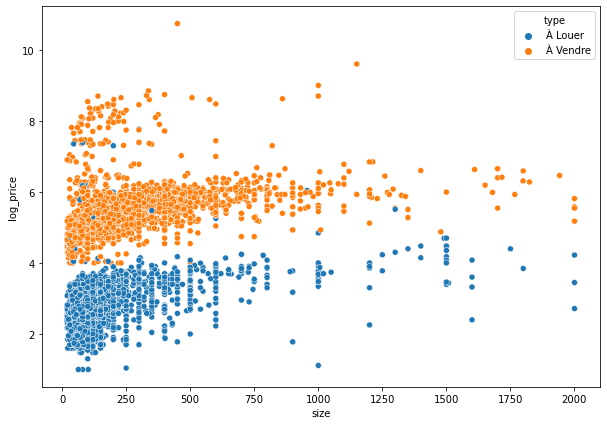
After conducting the preliminary data visualization, we moved on to creating a correlation map to plot pairs of correlated data. The correlation map revealed a strong correlation between size and log\_price, which we took into consideration as we started plotting the density-plot of the size column.



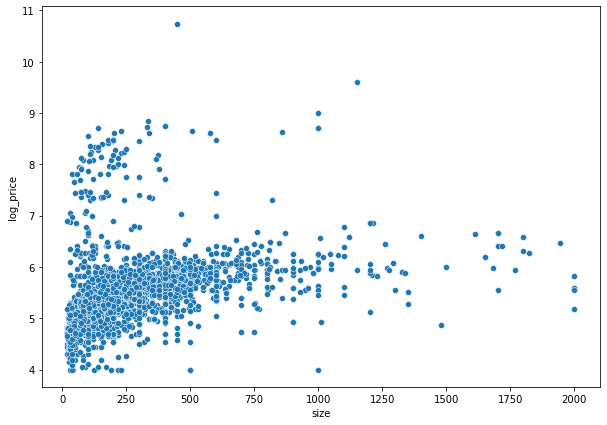
This scatter plot shows a positive linear relationship between size and log\_price, meaning that as the size of a property increases, the log\_price also tends to increase. We can also see some outliers in the plot, which could be due to other factors affecting the price of a property, such as location or condition.



As expected, the plot shows that properties for sale have higher prices than properties for rent. However, we cannot draw further conclusions from this plot alone, so we proceeded to split the data into two parts: one for rented properties and another for sold properties.



This plot shows the relationship between the size and logarithmic price of the "For Sale" category, where we can observe a logarithmic relationship with some outliers.



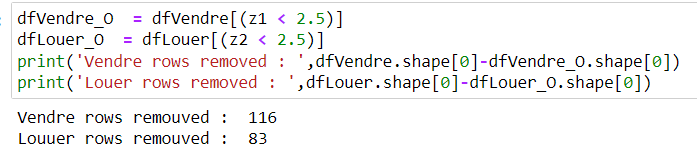
This plot shows the relationship between the size and log\_price of the "For Rent" category, which appears to be more linear than the "For Rent" category plot.



## **DATA PREPARATION**

* ***Deleting Outliers :***

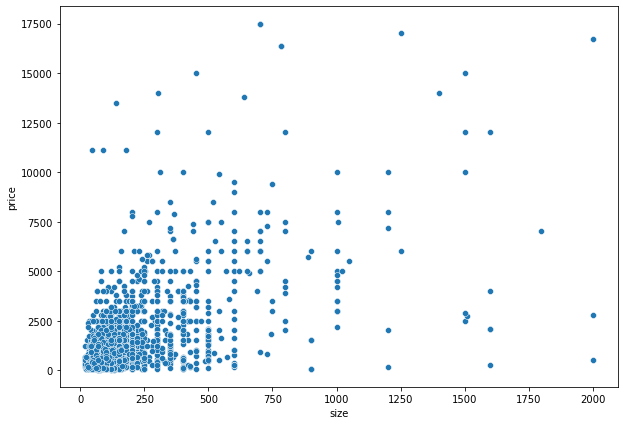
To remove outliers from the dataset, we employed the Z-score method, which resulted in the following outcomes: (2.5 seemed the best value for sigma)



We revisited the (size x log\_price) countplots after removing the outliers using the Zscore method. The resulting plots are presented below:



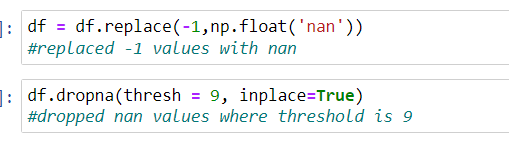
PROPERTIES FOR SALE



PROPERTIES FOR RENT.

* ***Null Values treatment:***

As mentioned previously, we discovered several instances of ‘-1’ values in our dataset. In order to address this issue, we applied a threshold of 9 and removed all rows containing such values.



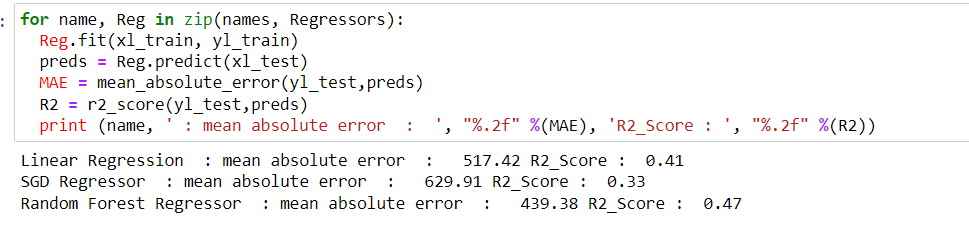
## **MODELING**

1. **PREDICTION:**

We employed regression analysis to predict the property prices and evaluated the performance of three different models:

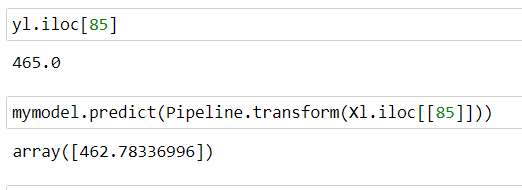


We selected the most suitable regression algorithm based on the R2 score and MAE:



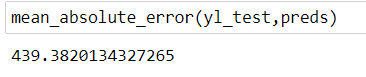
* The Random Forest Regressor had the best scores among the two models, therefore, we chose it.

After fitting, training, and testing the model, we obtained a highly satisfactory prediction with a high R2 score and a low MAE, indicating the model's accuracy in predicting property prices.



* The real value is 465 and our predicted value is 462.8.
* ***EVALUATION:***

The final Mean Absolute Error (MAE) of the Random Forest Regressor model was 439.4, which is a very satisfying result. It is worth noting that the prediction was performed on three different datasets, namely the Sales dataset, the Rent dataset, and a combined dataset. Among these datasets, the Rent dataset yielded the best results in terms of MAE.



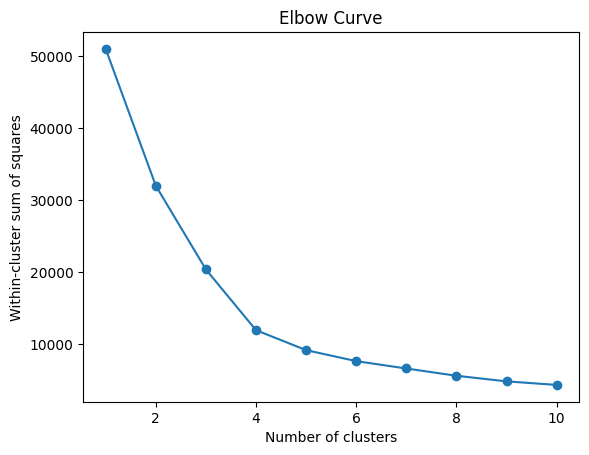
1. **SEGMENTATION**

We applied two segmentation models, K-means and ACH, to each of the three datasets. The results differed depending on the dataset.

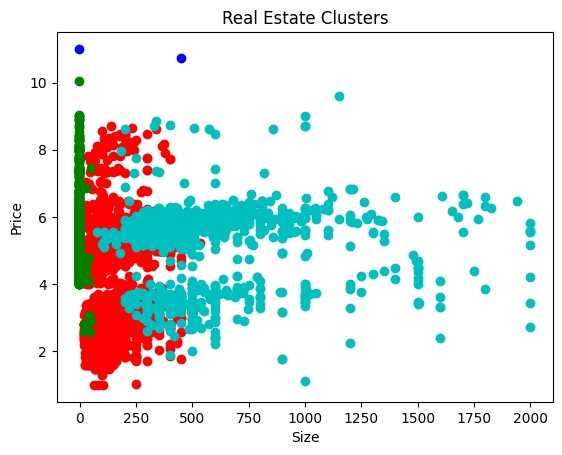
* ***SEGMENTATION ON A COMBINED DATASET:***

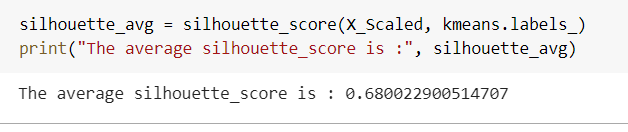
In this section, we will discuss a dataset that includes both properties for rent and properties for sale.

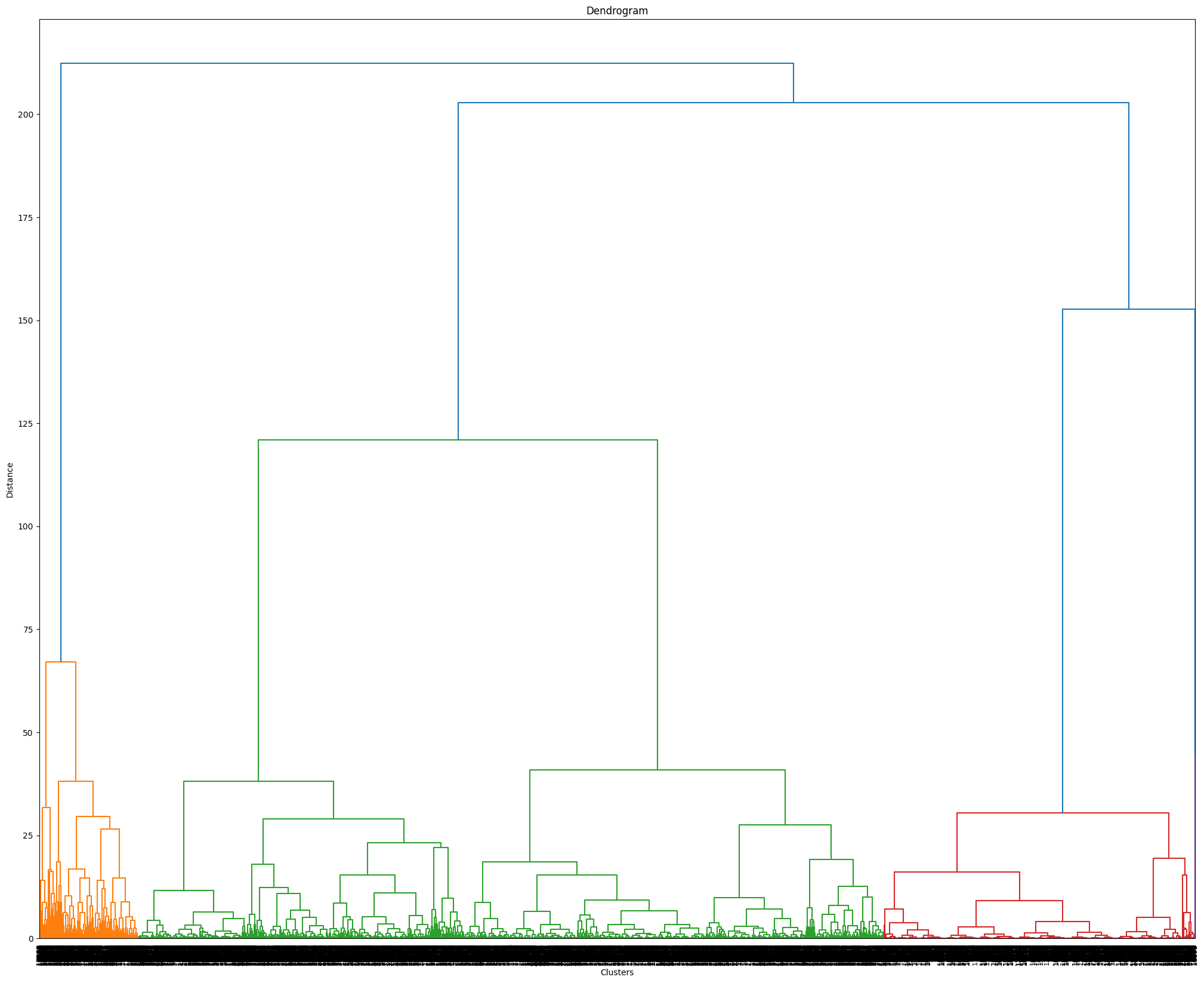
We began by plotting an elbow curve to determine the optimal number of potential clusters in our dataset, and we determined that four clusters were appropriate.



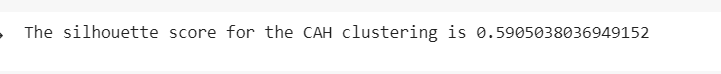
These are the four clusters we obtained using the segmentation model. The average silhouette score for this method is as follows:







This is our dendrogram (ACH algorithm) which had the following silhouette score :



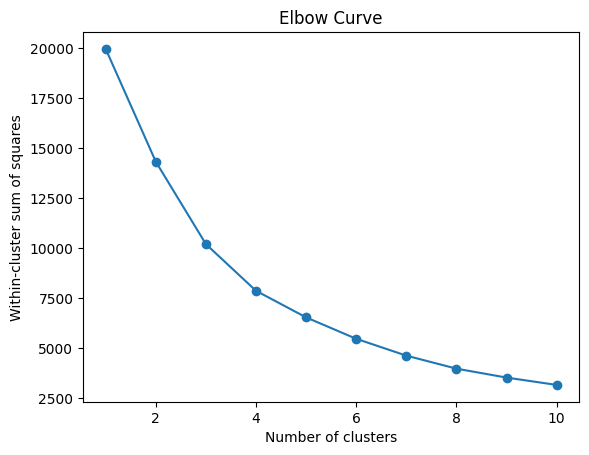
***CONCLUSION :***

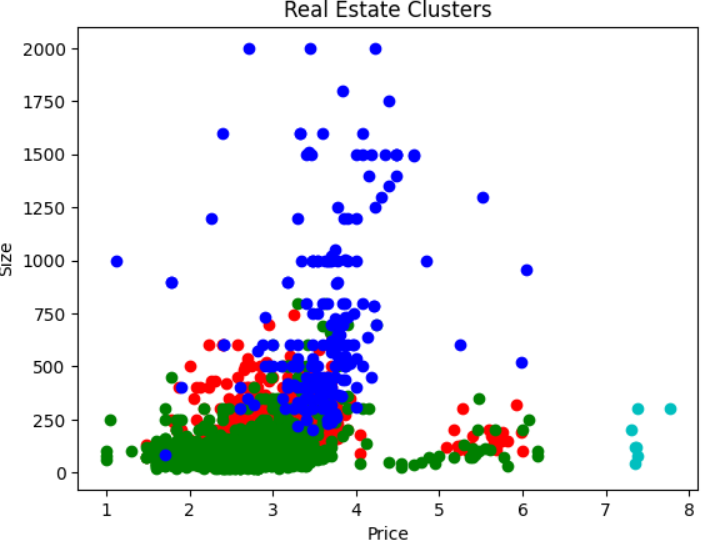
as per the combined dataset, we can see that the K-means algorithm works better.

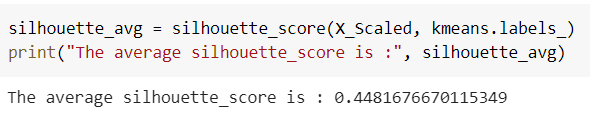
* ***Rent Dataset:***

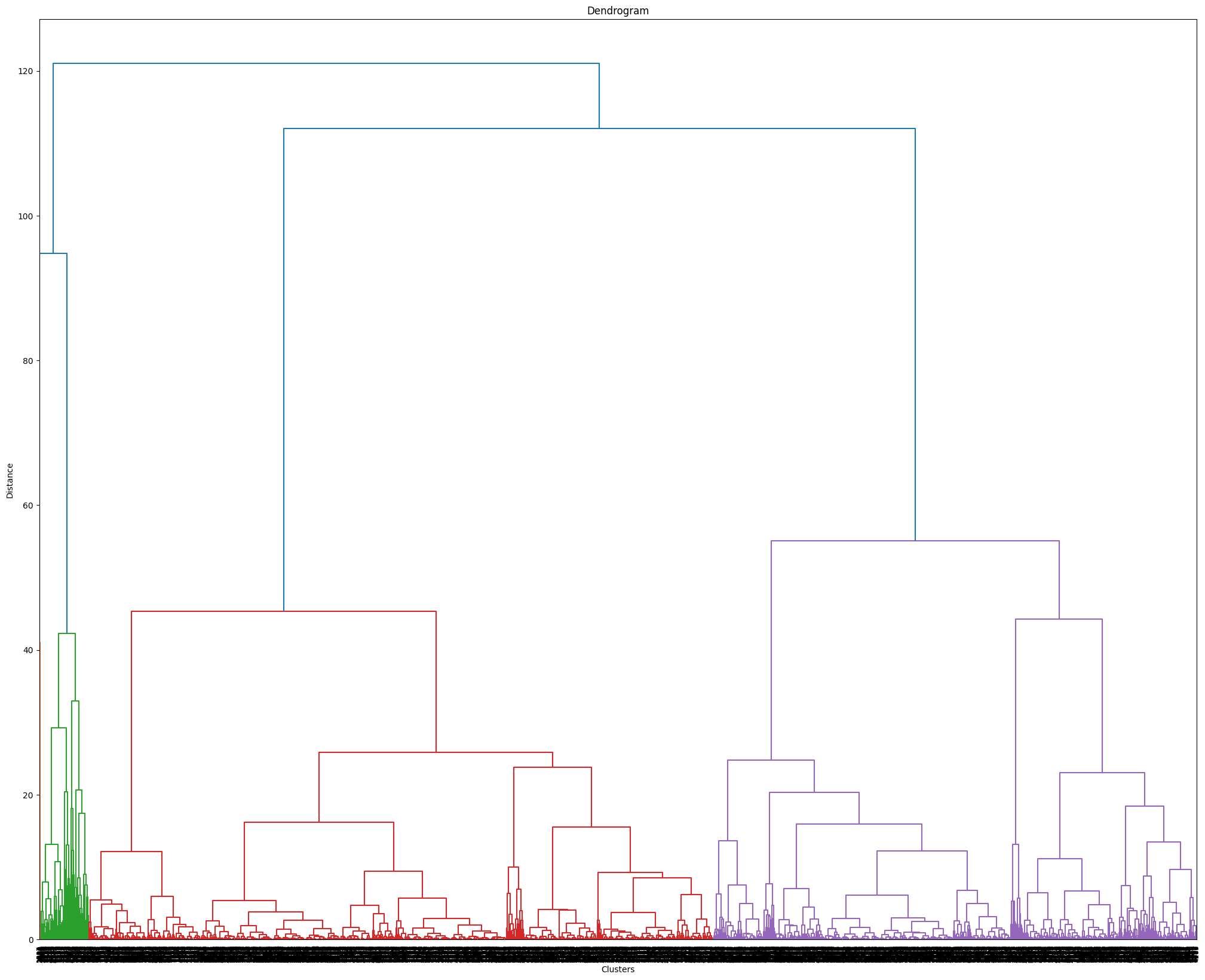
In this section, we will discuss the analysis of a dataset that contains properties for rent.

We started by plotting the elbow curve to get an idea of the optimal number of clusters for our dataset, and the curve indicated that 4 clusters would be a good choice.





After running the K-means algorithm, we ended up with this plot, which appears to be unsatisfying. This is confirmed by the silhouette score, which is the following:



As per the ACH method, we got this plot which scored significantly better than K-means :

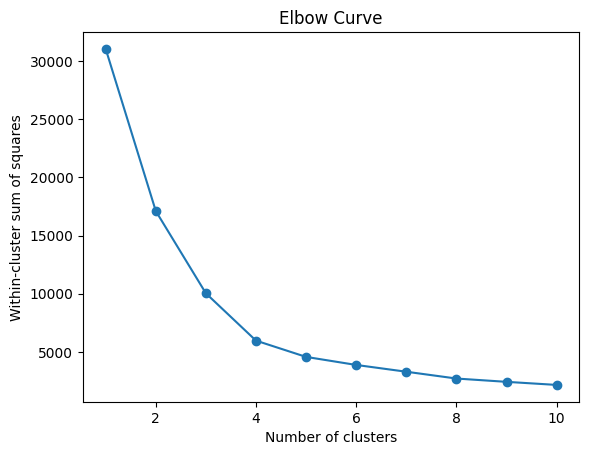


***CONCLUSION :***

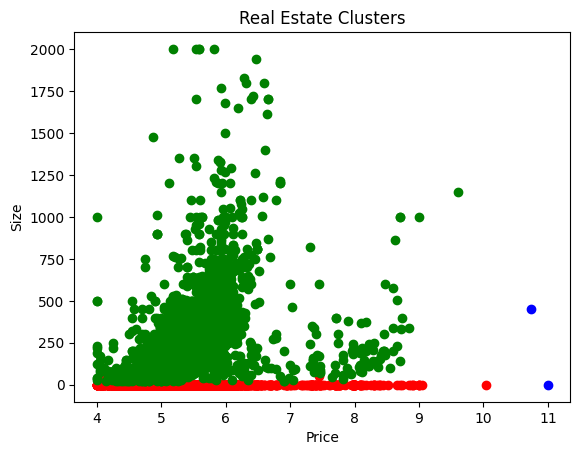
For the rent dataset, the ACH method worked better.

* ***Sales Dataset.***

In this section, we will discuss the analysis of a dataset that contains properties for sale.



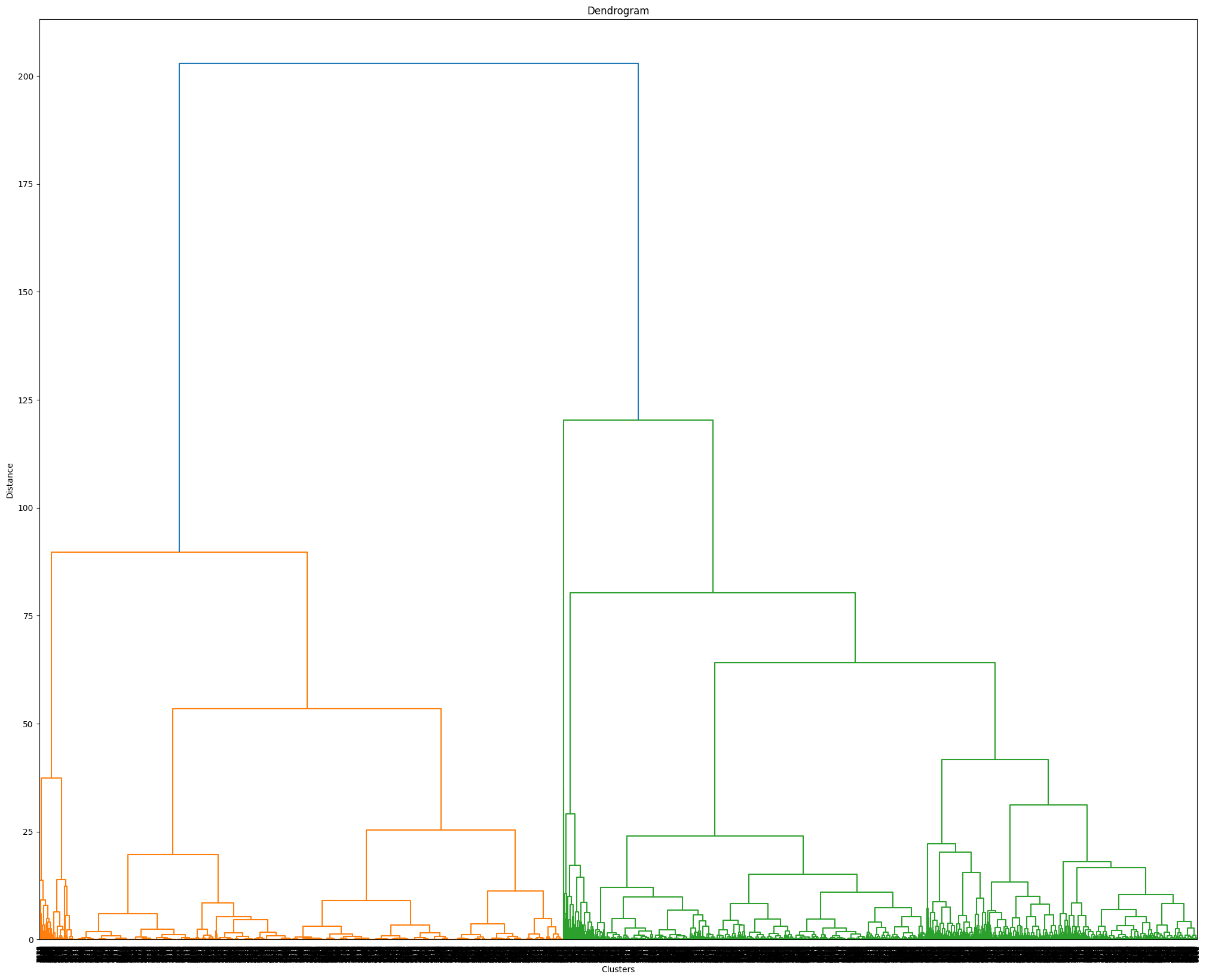
We started by plotting an elbow curve, and the number 4 was selected as the optimal number of clusters.



we can see a significant improvement in this plot, which shows clear clusters, this is confirmed by its silhouette score:



As per the ACH algorithm, we got this plot :



However, this is not satisfying and will not be retained due to its low silhouette score:



***CONCLUSION:***

The K-Means algorithm worked better than the ACH algorithm for this dataset.

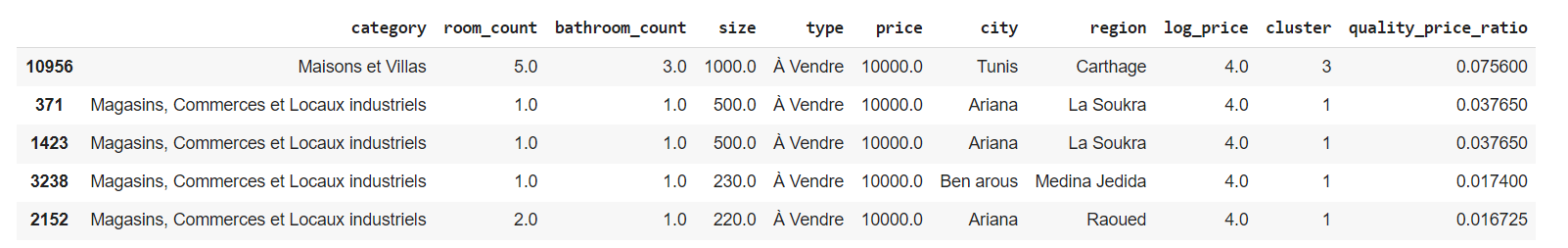
### **Scoring**

We developed an algorithm to calculate the quality-price ratio of any given property and added it as a new column in the dataset. The algorithm is as follows:

Examples of the scoring in practice:



this table shows the top 5 best properties to rent.



this table shows the top 5 best properties to buy.

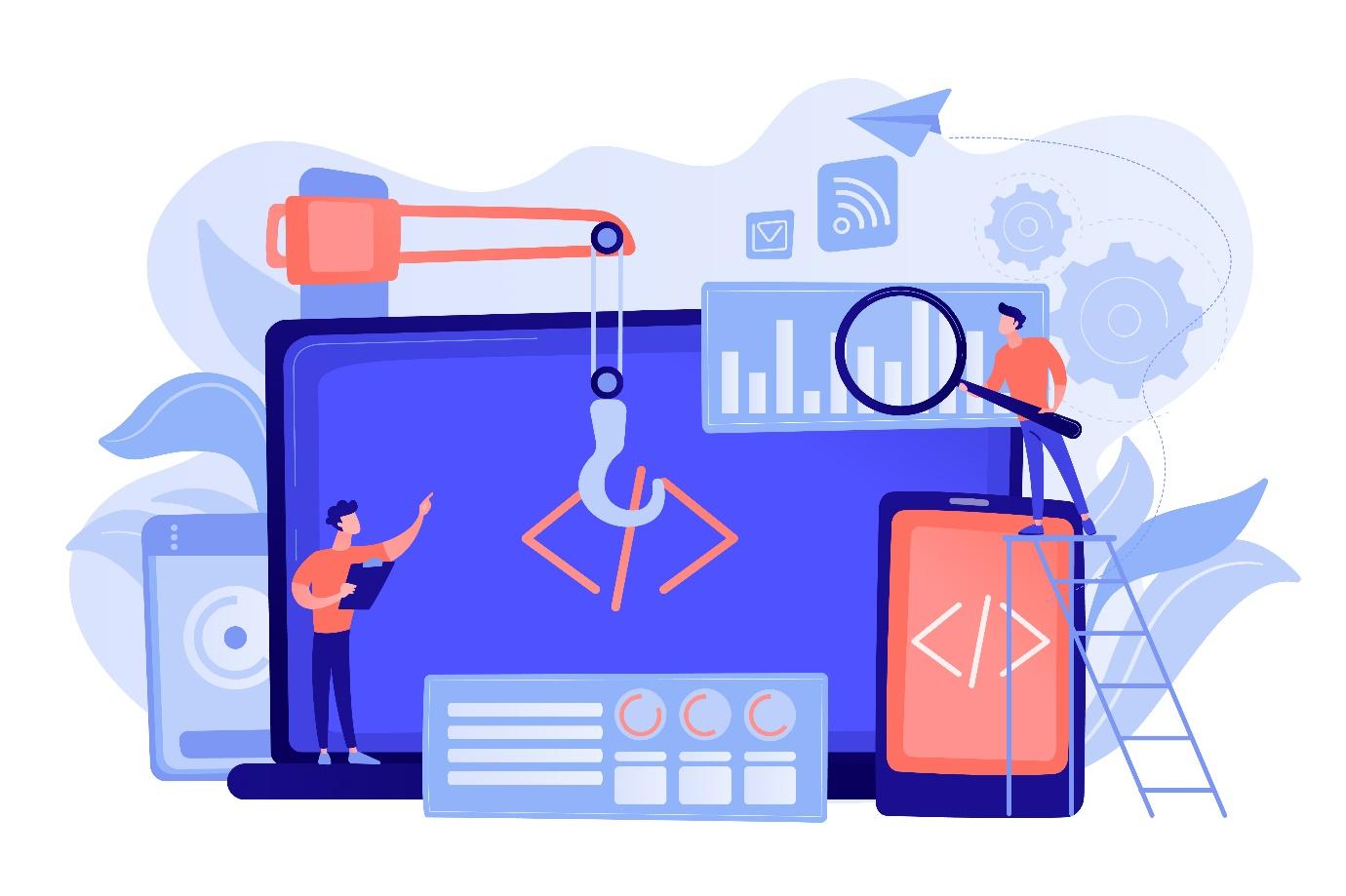
## **Conclusion**

In conclusion, we have analyzed and explored three datasets containing properties for sale, properties for rent, and a combined dataset. We have performed data cleaning, exploratory data analysis, regression analysis, and segmentation using K-means and ACH algorithms. We have also developed an algorithm to calculate the quality-price ratio for each property in the dataset.

Our findings show that the Rent dataset yielded the best results in terms of regression analysis and segmentation. Additionally, the K-means algorithm produced better results for the combined dataset than the ACH algorithm. The quality-price ratio algorithm can be used as an additional metric to help customers make informed decisions when purchasing or renting a property.

Overall, this analysis can be useful for individuals and companies interested in the real estate market, providing insights and potential strategies for pricing and segmenting properties.

# ***Chapter VI: Front End Development***

******

## **Introduction**

The development of modern web applications requires a combination of frontend and backend technologies. In this report chapter, we will explore the frontend development with Angular and the implementation of the server-side application with Spring Boot. By combining these two technologies, we can build a full-stack web application that offers a seamless user experience and a scalable backend. In this report chapter, we will discuss the steps involved in setting up an Angular project and integrating it with a Spring Boot backend. We will also explore how to create RESTful APIs, handle HTTP requests and responses, and secure our endpoints using Spring Security. **Finally, we will delve into the interfaces of our application and how to design them to meet the needs of the users.**

By the end of this report chapter, we will have a clear understanding of how we developed our full-stack web application using Angular and Spring Boot.

## **Environment Setup**

Setting up the environment for an Angular frontend project is a crucial step that we, as developers, need to perform to start building our applications. We installed Node.js, Angular CLI, and created a new Angular project. Once the project was created, we could start serving the application using the development server provided by Angular CLI. This allowed us to see our changes in real-time and quickly iterate on our application. With the environment set up, we focused on creating the frontend components and business logic for our application. By following the steps outlined in this process, we ensured that our Angular frontend project was set up correctly and ready for development.

## **Implementation of the backend application**

*To implement a Spring Boot application as a server side in our frontend Angular application,* we used Angular's HttpClient module to make HTTP requests to our Spring Boot backend and receive responses. We wrote services in Angular to encapsulate the HTTP requests and responses and expose them to our Angular components.

We also made use of Spring Security to secure our API endpoints and ensure that only authorized users could access sensitive data. This involved defining authentication and authorization mechanisms and configuring them in our Spring Boot application.

By implementing our Spring Boot application as the server side in our Angular frontend application, we were able to build a full-stack web application with a responsive user interface and a scalable and secure backend.

## **Interfaces**

This screenshot displays the login interface where the user can authenticate using his email address and password, register, reset his password and login with his google account.

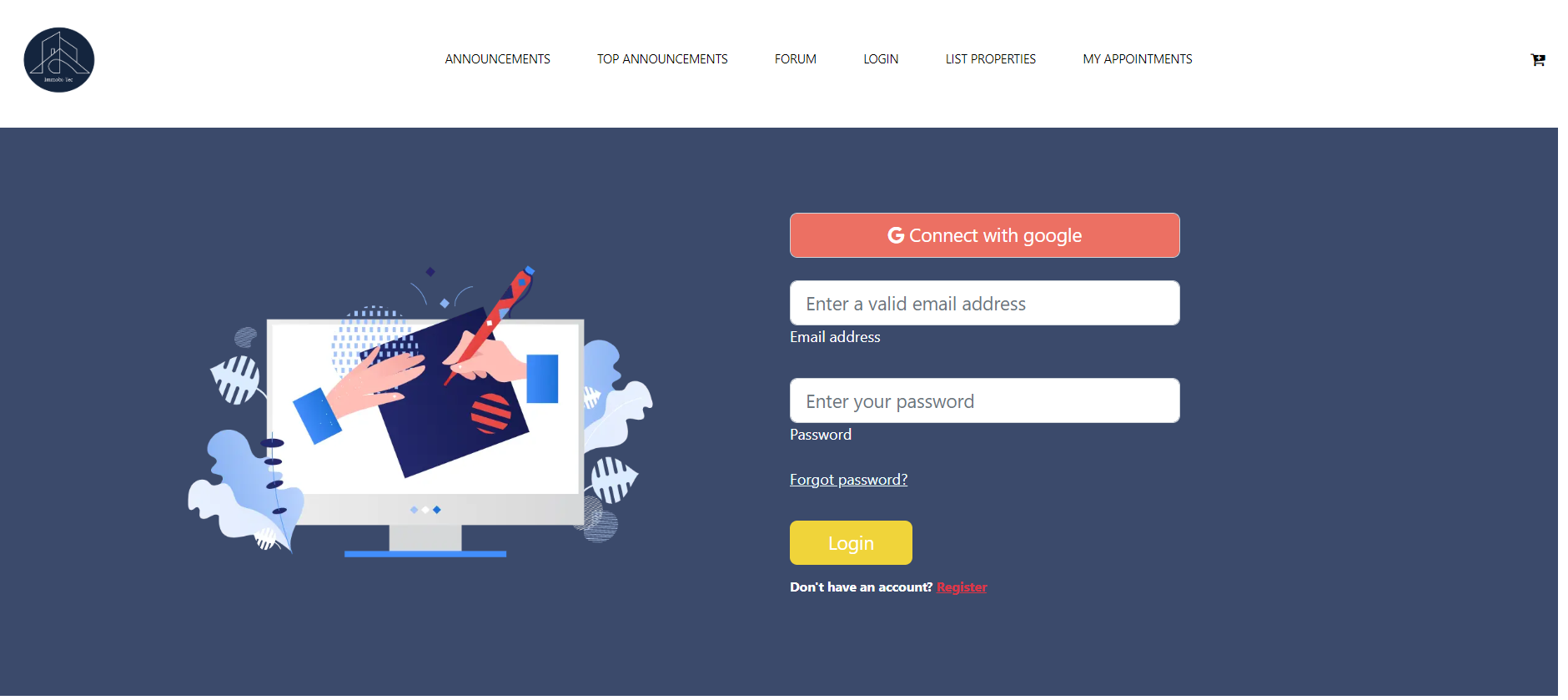


Figure 10 : Login Interface

This screenshot displays the list of product, where the user can add a promo or can order a product

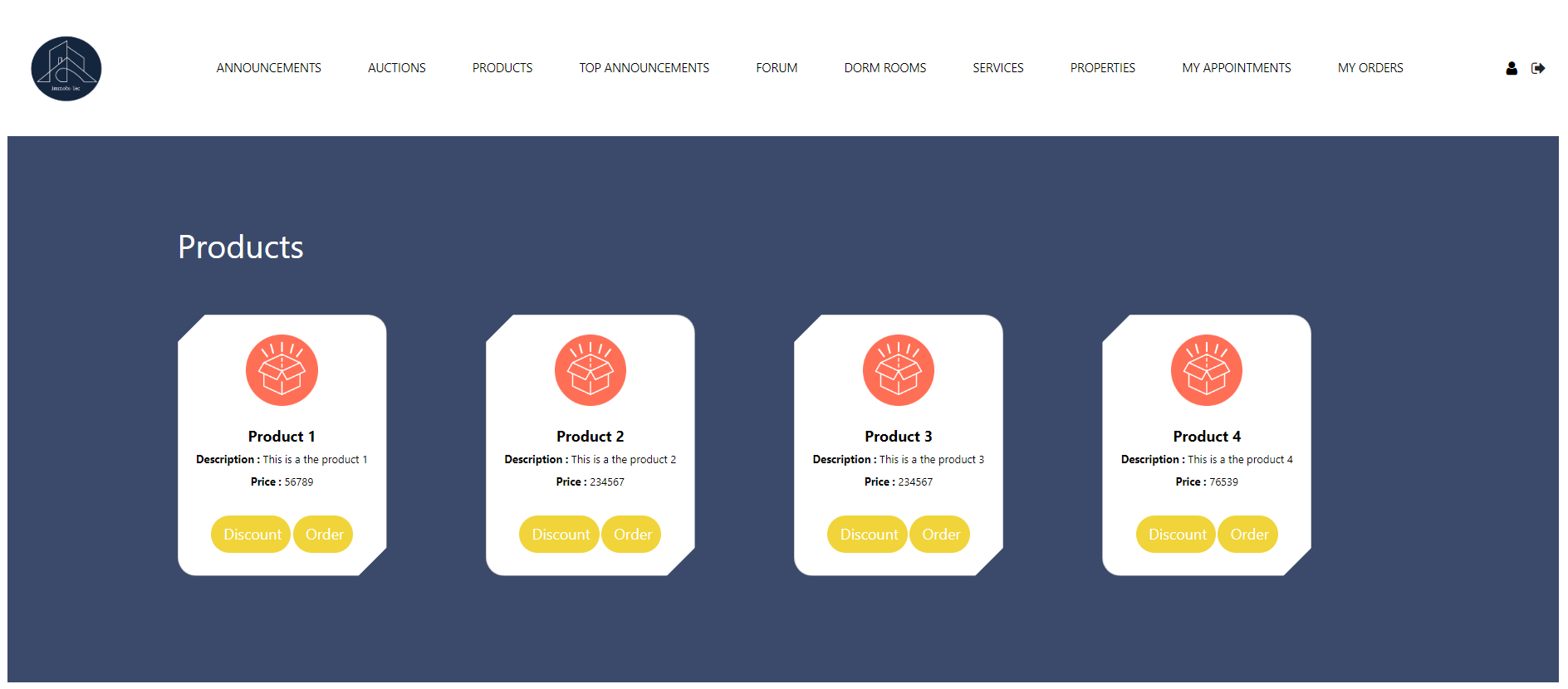


Figure 11 : List Product Interface

This screenshot shows the announcement details where the connected user can rate the announcement and he can calculate the distance and durance between the property position

And a chosen position by the map.

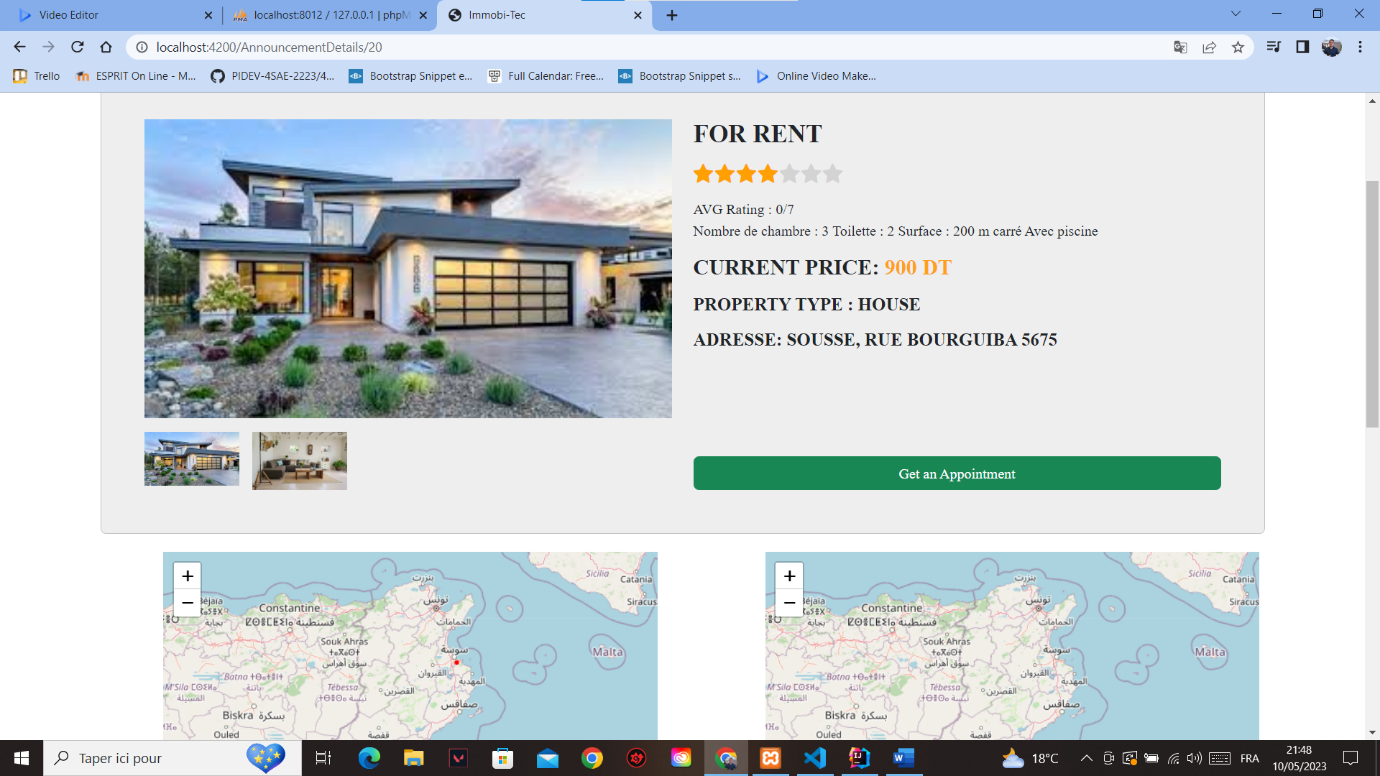


Figure 12: Annoncement Details

This screenshot displays the list of dorm rooms where the user can reserve a room and also he can like or dislike a room.

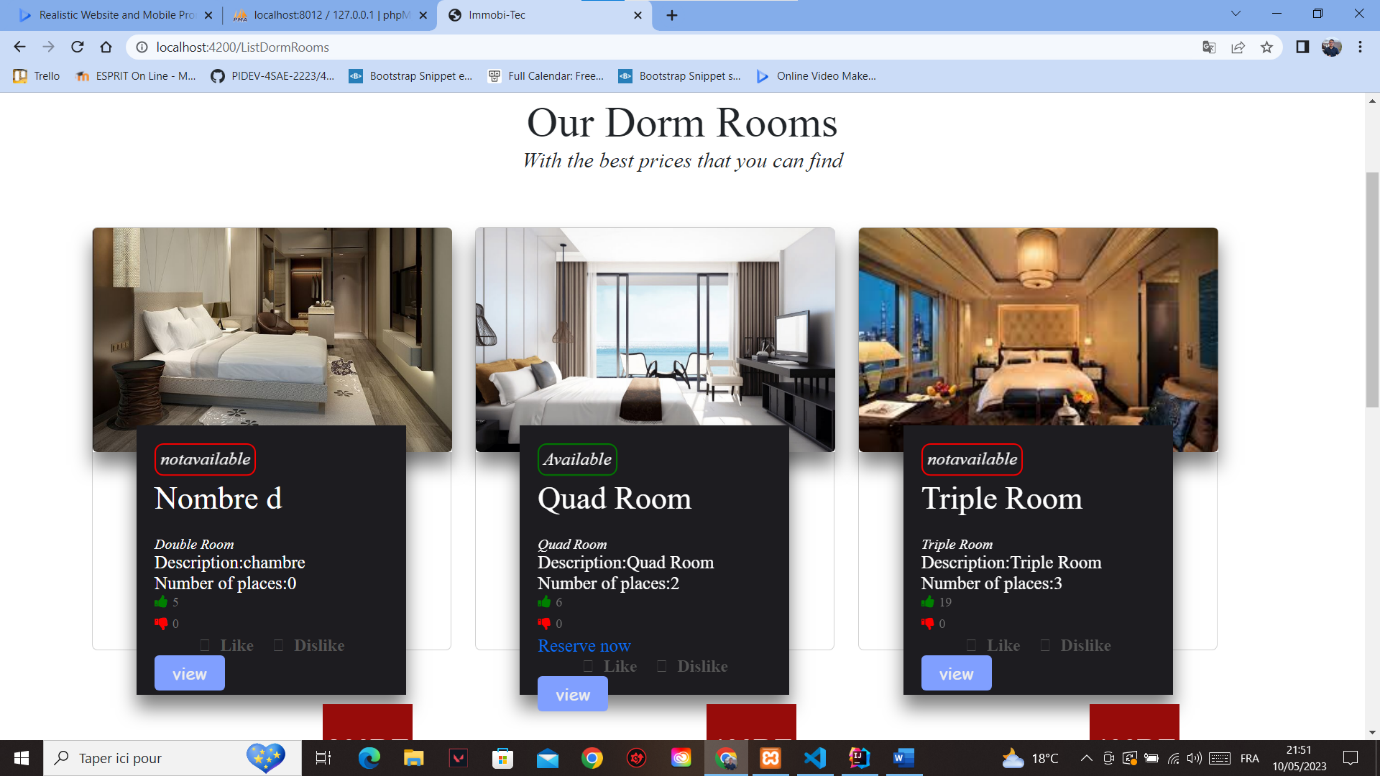
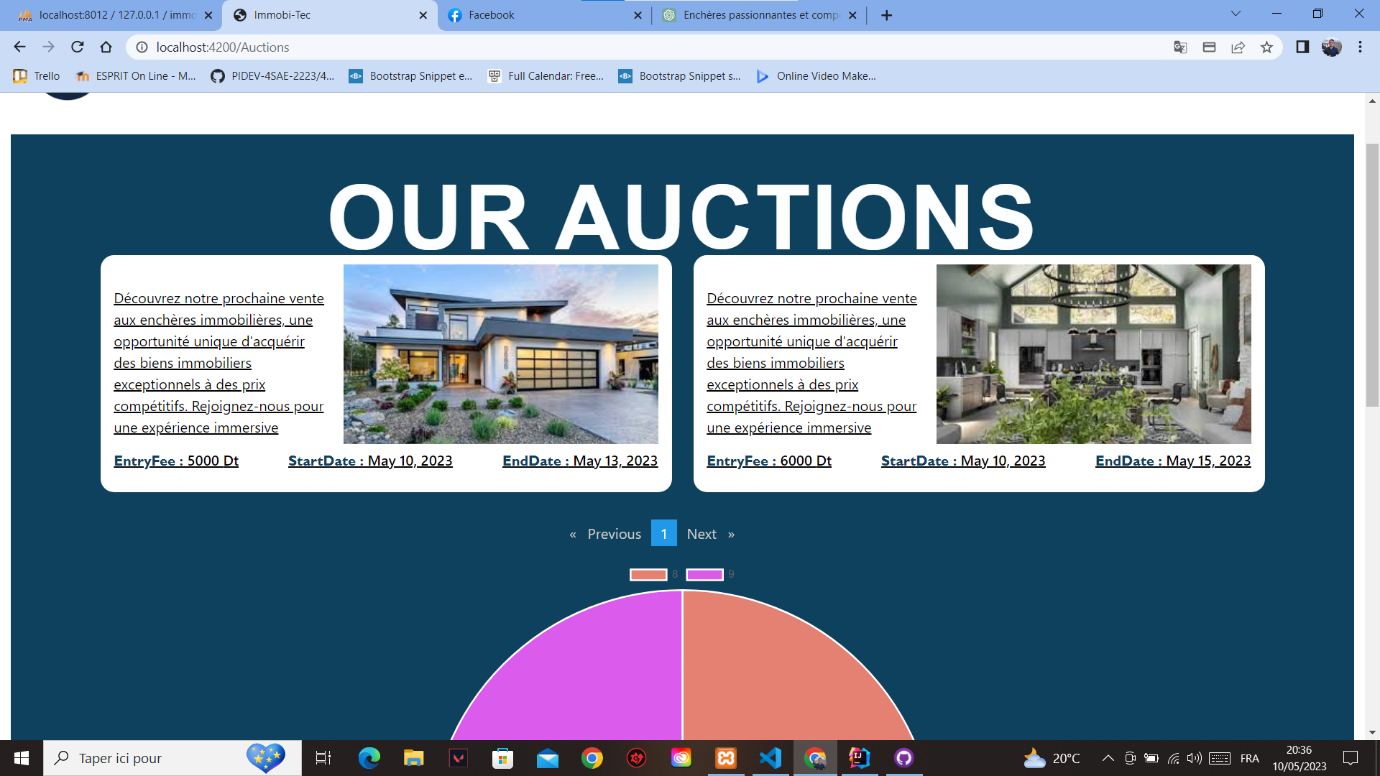
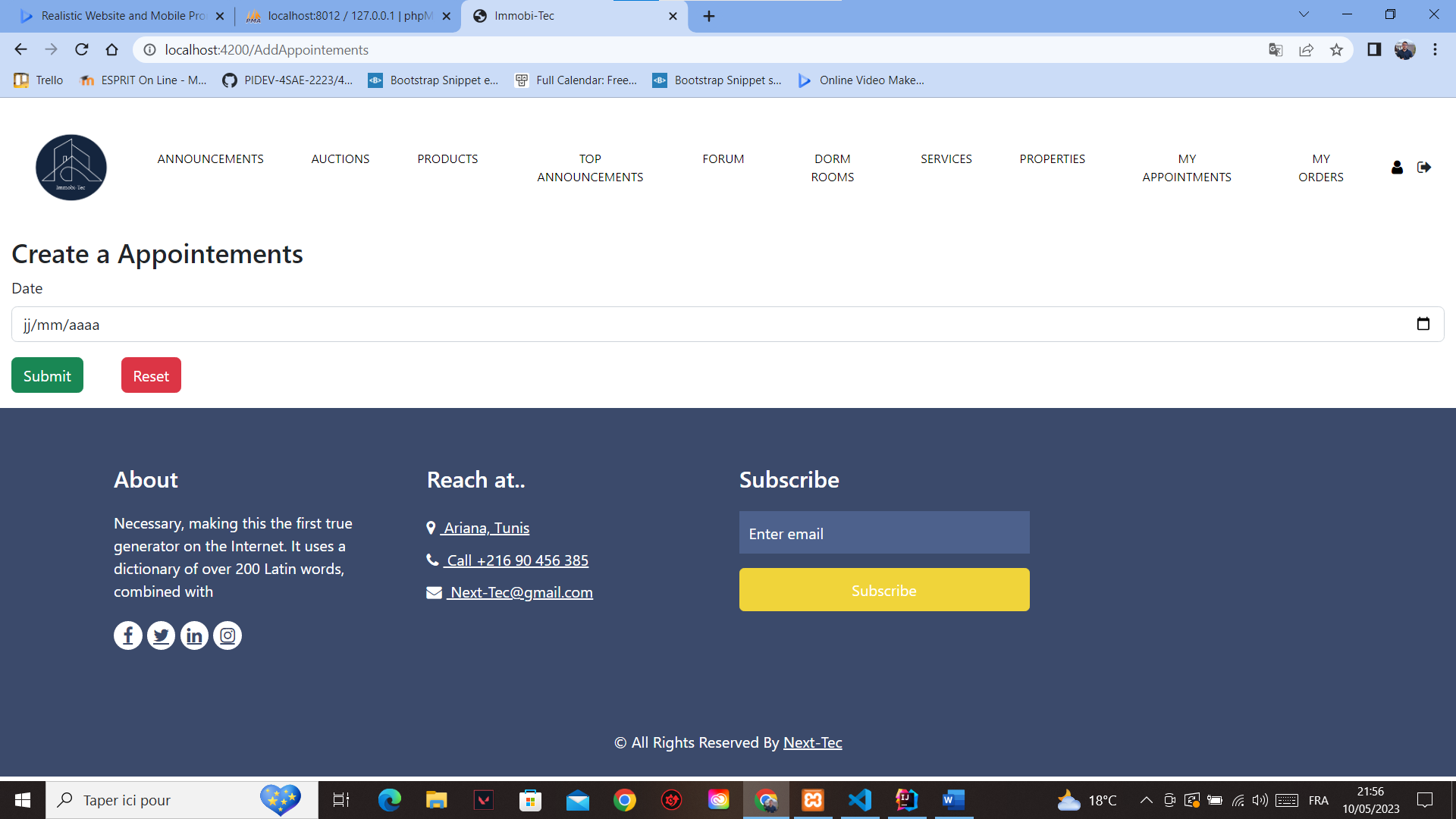


Figure 13: Dorm Room List

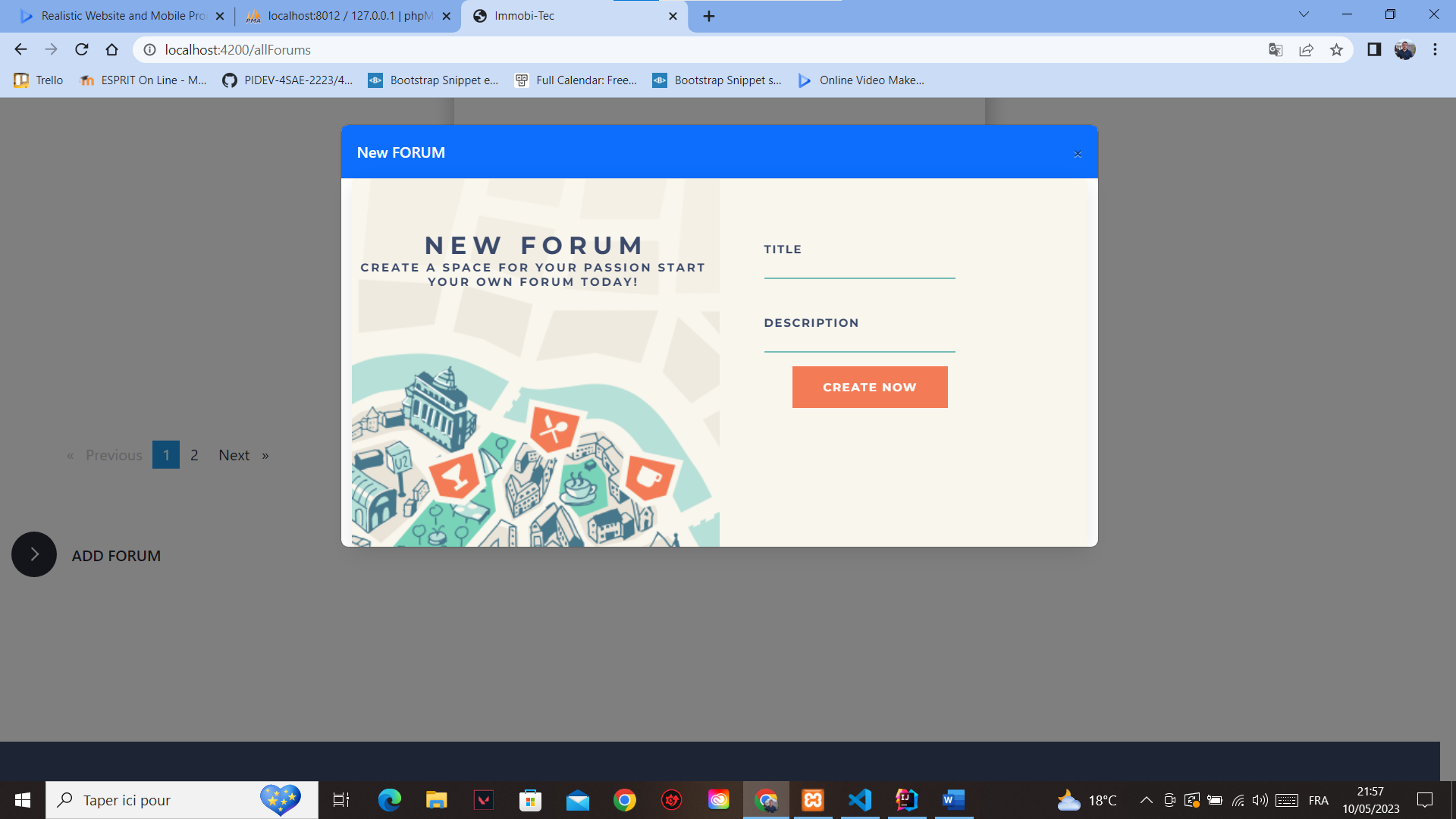
This screenshot shows the list of auctions and some statistics where the user can enter one of the auctions and add his mise.



This screenshot shows the interface of adding an appointment on a selected service.



This screenshot displays the interface of adding a forum. After adding a forum the user can add a post on a selected forum and also he can comment a post too.



## **Conclusion**

In conclusion, the combination of Angular and Spring Boot provides a powerful platform for developing full-stack web applications. By leveraging the strengths of each technology, we can create web applications that are both responsive and scalable. Through this report chapter, we have covered the steps involved in setting up an Angular project and integrating it with a Spring Boot backend. We have also explored the fundamentals of RESTful API creation, HTTP request and response handling, and endpoint security using Spring Security. Additionally, we have discussed the importance of intuitive interface design and development to ensure a seamless user experience. By mastering these concepts, developers can build robust and engaging web applications that meet the needs of modern users.

## ***General Conclusion***

In this report, we presented a comprehensive analysis of a real estate dataset that contains properties for rent and sale. We started by discussing the requirements specification for the development of a web application that would allow users to browse and filter these properties based on various criteria. For the front-end development, we chose Angular, a popular JavaScript framework that provides robust tools for building dynamic and responsive web applications. For the back-end development, we opted for Spring, a powerful Java framework that enables the creation of scalable and secure web services.

In the data mining chapter, we explored various techniques for data preprocessing and analysis. We discussed the importance of data cleaning and feature engineering, and we presented different methods for outlier detection and removal. We also used regression algorithms to predict property prices and segmentation models to cluster properties based on their attributes. Our analysis revealed interesting patterns and insights about the real estate market, and we showcased our results using visualizations and performance metrics.

Finally, we presented the implementation details of our web application and how it integrates with the data mining algorithms we developed. Our application allows users to search for properties based on different criteria such as location, price range, property type, and more. It also provides features for favoriting properties, sharing them on social media, and contacting the property owner. We believe that our web application provides a valuable resource for anyone looking to rent or buy a property, and we hope that our analysis can help inform their decisions.

Overall, this project has been a challenging and rewarding experience that allowed us to showcase our skills in web development, data mining, and machine learning. We learned a lot about the real estate industry and how to extract valuable insights from complex datasets. We hope that our report can serve as a useful reference for anyone interested in similar projects, and we look forward to exploring new and exciting opportunities in the future.