Simple Homes

Software Design

Class: S3-CB02

Joan Krastanov
Supervisor: Tim Kurvers & Marcel Boelaars
Eindhoven, 10-3-2021

Definitions, Acronyms and Abbreviations

Term	*Description*

[these terms should be ordered alphabetically]

Table of Contents

1 Introduction

1.1 Document Purpose

This is a Software Design document explaining how the Simple Homes web application will be developed and prepared for deployment. It will also provide information about the principles that will be followed and the frameworks that will be used in order to complete the project.

1.2 Document Overview

Chapter 2 will give a general description of the functionality, context, and design of the Simple Homes project.

Chapter 3 will provide a C1, C2 and C3 architecture diagrams and explain the components and their relations including rationale for selecting the architecture described including critical issues and trade/offs that were considered.

Chapter 4 provides the functionality of the system from the user's perspective and explain how the end user will be able to use your system to complete all the expected features and the feedback information that will be displayed for the user.

2 System Overview

The Simple Homes project is a web application designed with the intent of helping people in the Netherlands with finding a place to live by buying it or renting it out. It will also help homeowners find reliable and responsible tenants for their properties.

An email service will be set up which will notify the users when a property with their desired details has been uploaded to the system. That way the ones who have set up the service will have the ability to quickly get in contact with the homeowners which will make the searching process a lot easier for them.

3 System Architecture

3.1 Architectural Design

C1 Diagram - System Context diagram

Provides the software system which is going to be created and how it relates to the users that are expected to be using it.

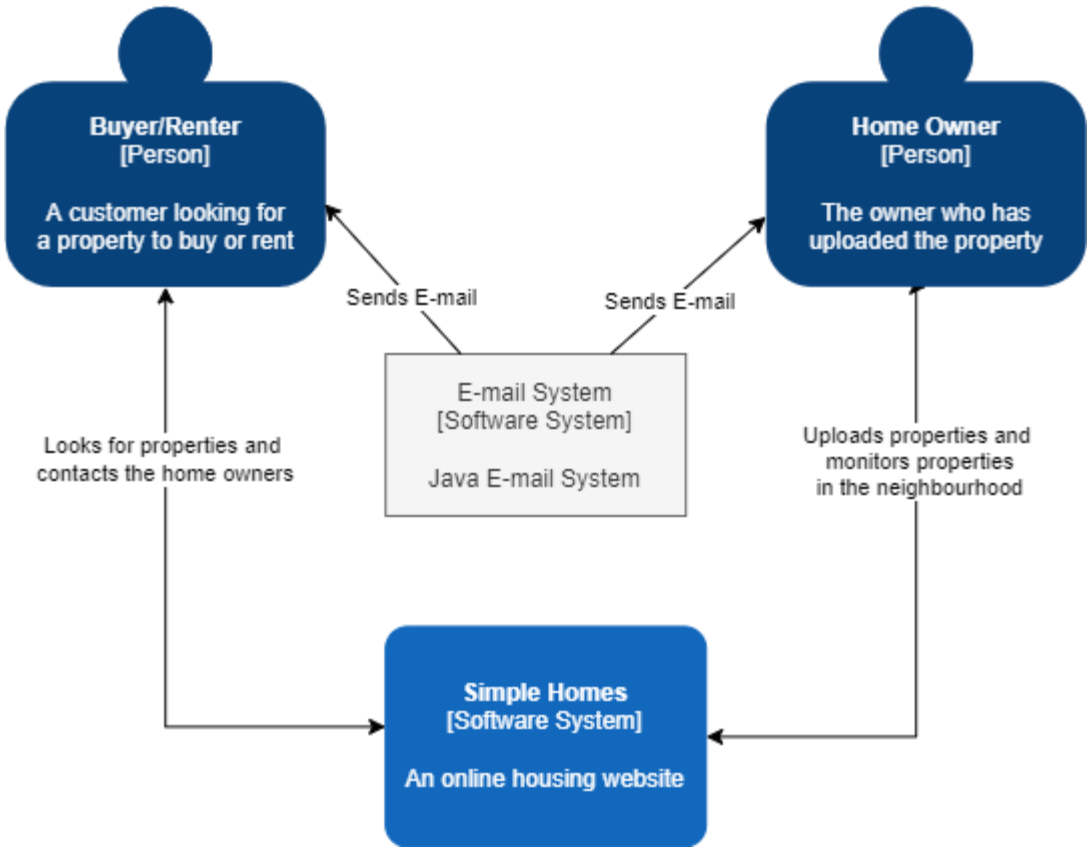


Figure 1 - C1 Diagram

C2 Diagram- Container diagram

Shows the containers with the technology decisions within the software system and explains their general purpose.

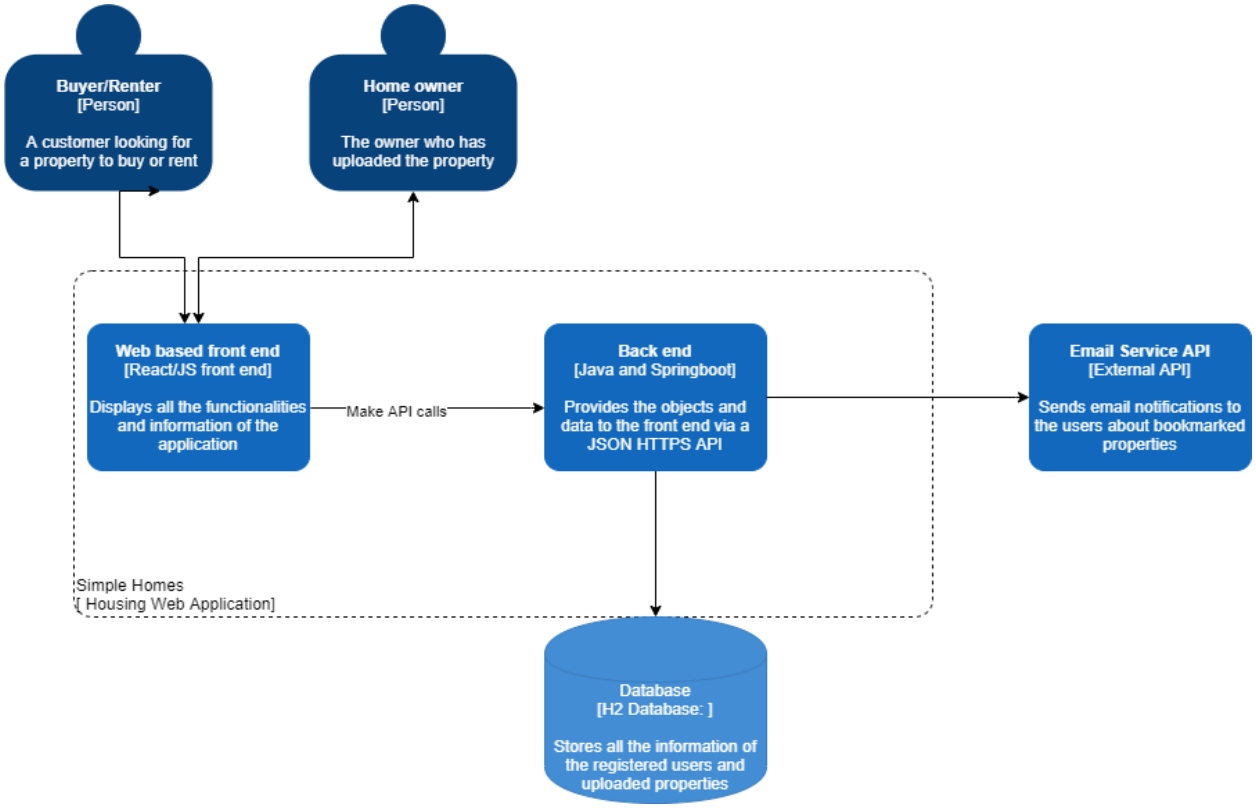


Figure 2 - C2 Diagram

C3 Diagram - Component diagram

The diagram zooms in on the software container and provides a more in-depth overview of the structure and architecture that supports the software solution as separate components.

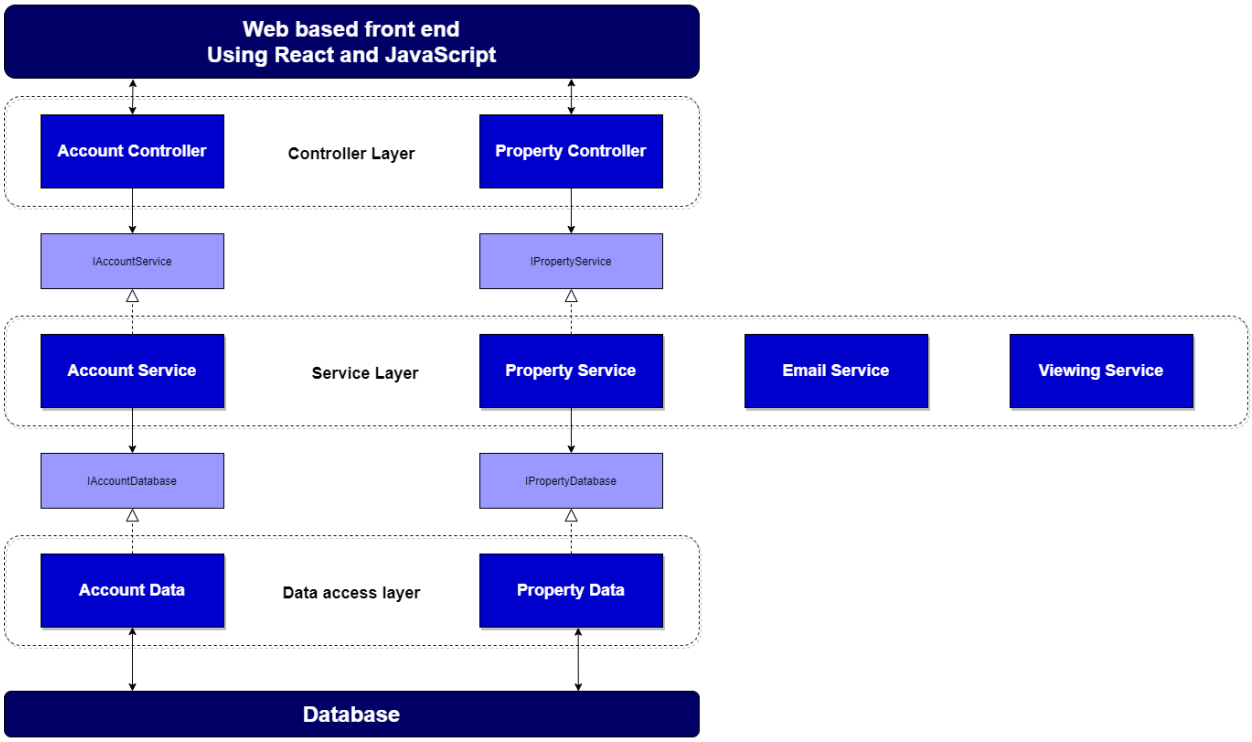


Figure 3 - C3 Diagram

C4 Diagram - Code diagram

The diagram zooms into an individual component, showing how that component is implemented

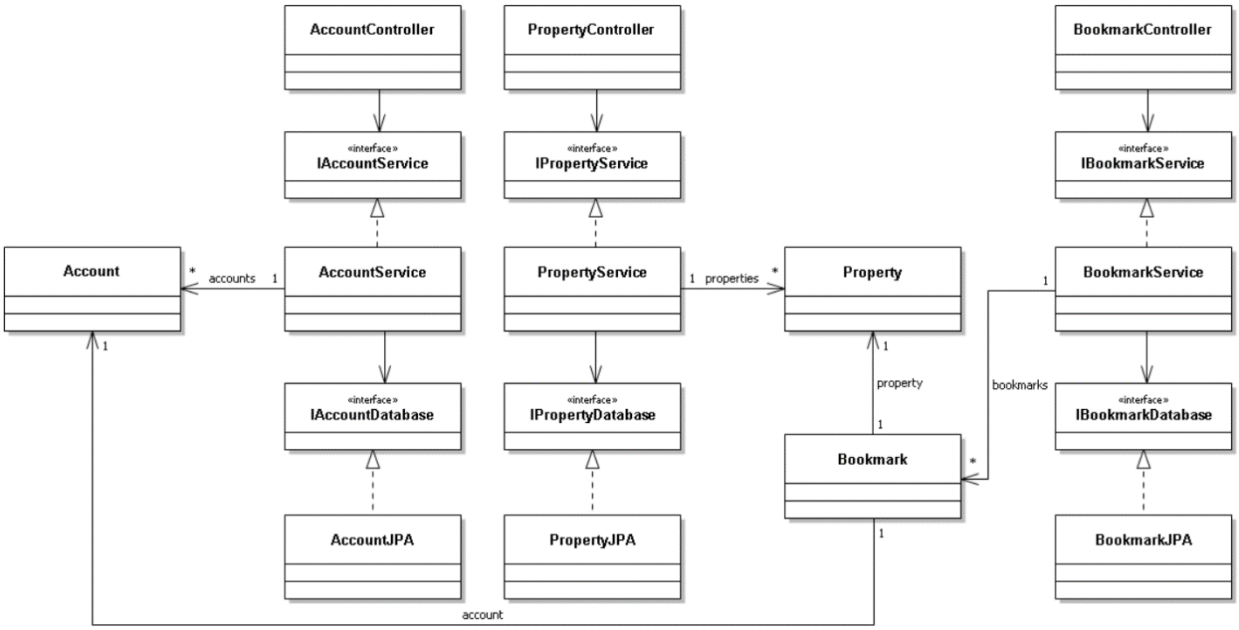


Figure 4 - C4 Diagram

3.2 Design Rationale

The project will be following the standard three-tier architecture consisting of a presentation layer, a business layer, and a data access layer.

The presentation layer consists of the controller classes which are used to interact with the front end by receiving API calls and responding with the appropriate data.

The business layer can be regarded as the brain of the application as it contains all the logic and calculations of the project and in putting it in a separate layer we ensure that the end user cannot interact with it directly, therefore making the application more secure.

The data access layer is what is going to be interacting with the database by retrieving and sending data to the business layer and the database respectively. Its sole purpose is to receive and send data without performing any calculations on its own.

Ensuring the three-tier architecture is a key component of the project since in doing so, we ensure that there will be no data leaks or security issues with the application since the end user will have access only to their data and will only be able to modify what is intended.

The framework which is going to be used is Spring Boot which help with the process by reducing the development time and increasing the productivity by saving time which can later be used for additional features.

4 GUI

4.1 General Overview

The web application will be focused on two different users: the seekers and the homeowners.

The seekers will be the ones looking for properties. They will have different filtering methods which can be used to filter out properties that are not of interest. They will also have the ability to directly search for desired properties by inputting a street name, city or even a postal code. Lastly, the customers will be able to display some basic information about themselves that will be on display in their profiles.

The homeowners will be able to do everything that a regular seeker is since they are all technically users, but the homeowners will also have the ability to upload a property.

4.2 User Interface Action

The user can use the search bar to quickly search for properties based on their location (city, street name or postal code).

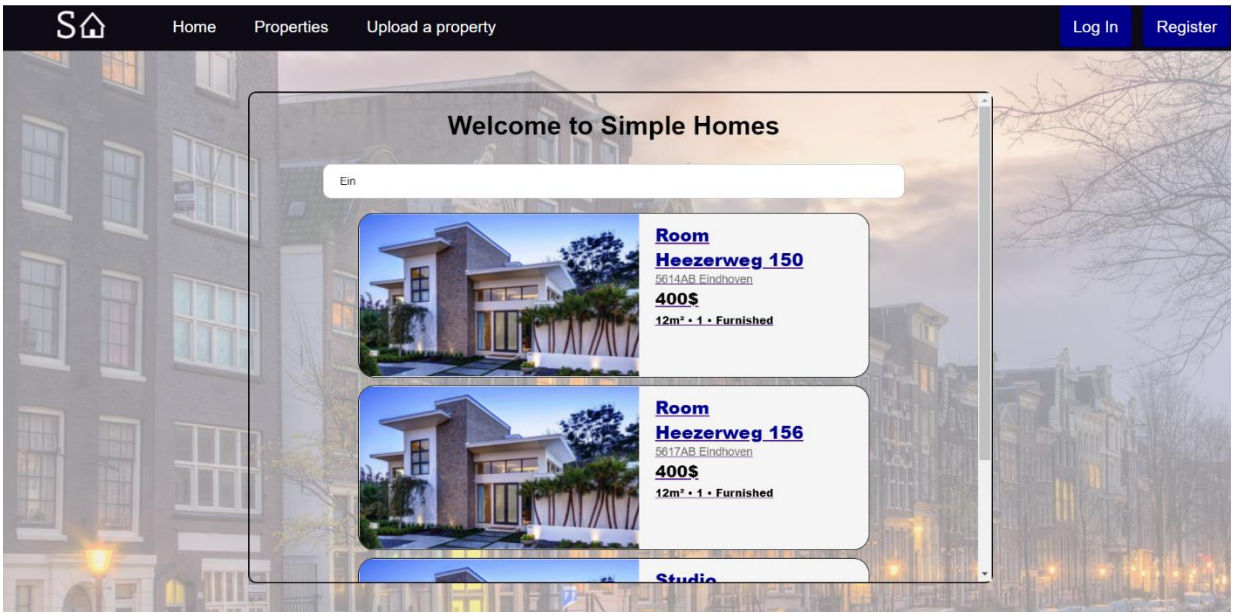


Figure 5 - Main Page of Website

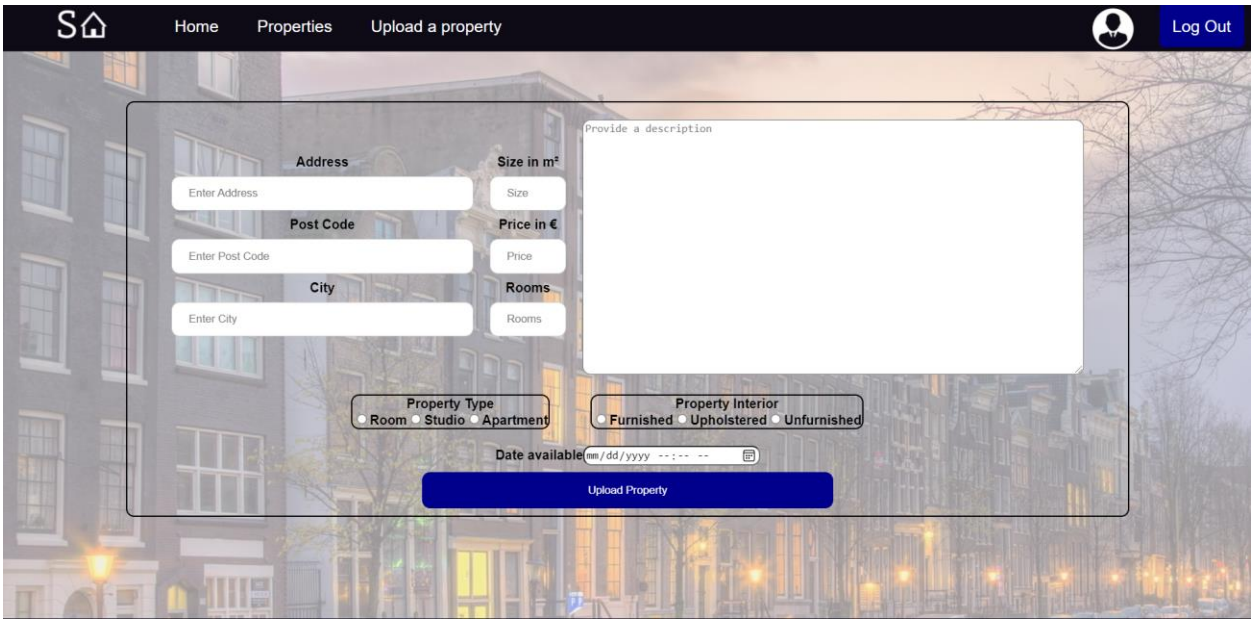


Figure 2 - Upload Property Page

A homeowner can use this page to upload a property to the website by providing the information required in the textboxes. After the "Upload Property" button has been pressed and the data has been validated, the property is uploaded to the website.

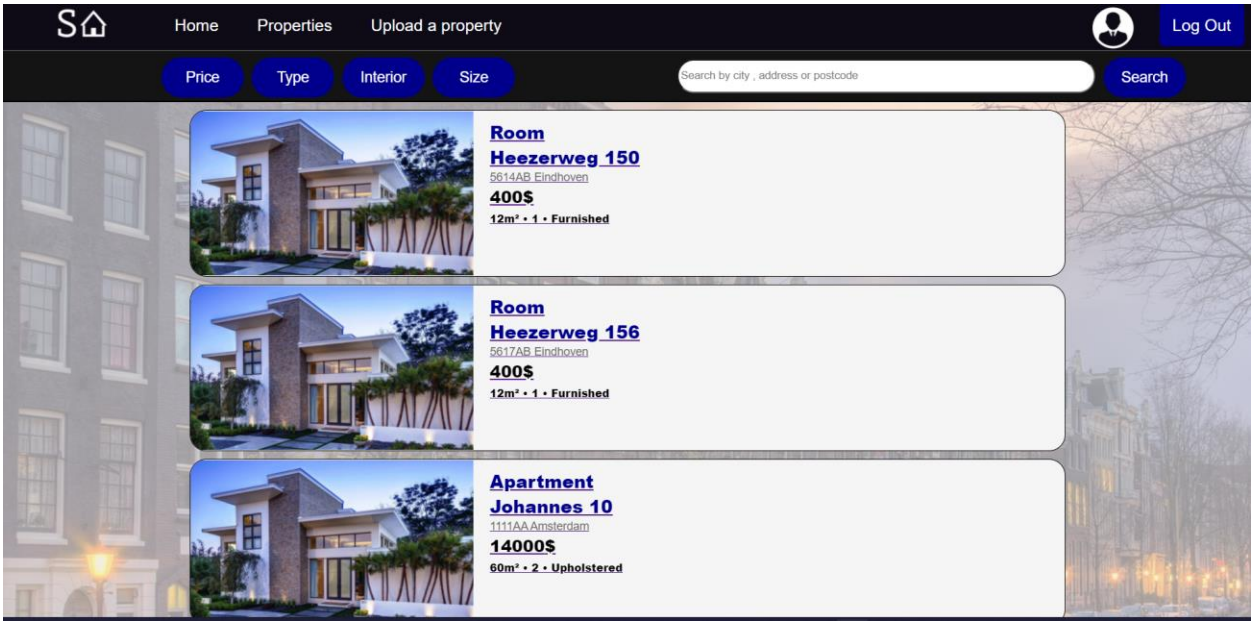


Figure 3 - View Properties Page

Each user can search for a property based on a number of different filters (price, property type (e.g. apartment, studio, etc.), interior (e.g. furnished, upholstered, etc.), size and location).

5 Email Service API

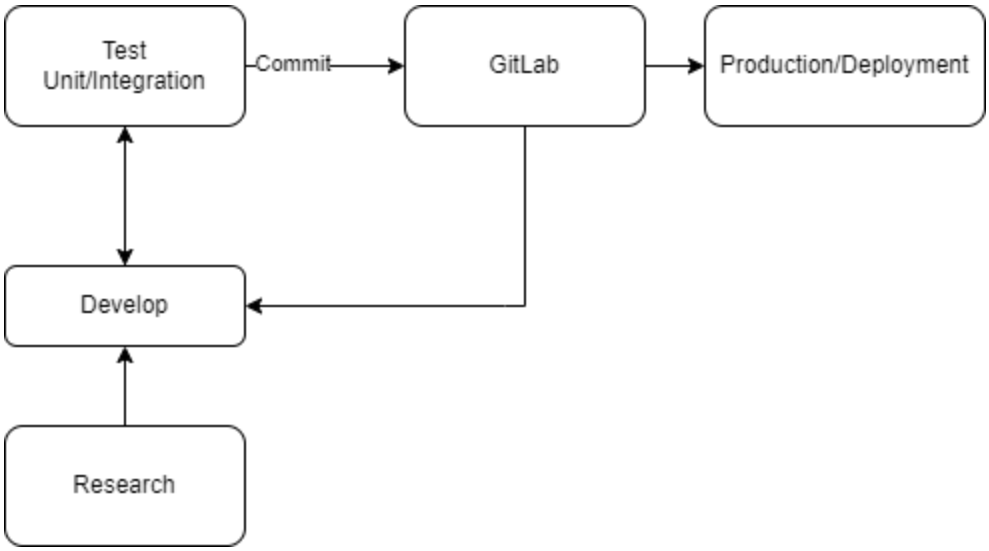
The email service shown in the Figure 1 - C1 Diagram will be a function that notifies the users when a property they have bookmarked has been updated or taken.

Since the project back end is created in Java a lot of different APIs and libraries can be used to accomplish that task. I have decided to use Simple Java Mail since at first glance it looks fairly simple to use so it shouldn't be that hard to implement properly.

The reason I picked that particular API is, because when I did my research about which is the best email API to use with Java, Simple Java Mail kept showing up. I also see that it is very well documented which should make the learning period easier and quicker.

Hopefully with the implementation of this functionality, the web application will have a feature that sets it apart from other housing websites which, in turn, will make it a unique and more convenient place for home-seekers to come.

6 CI Setup Diagram



Working on this project has taught me is to always start by doing me research related to the particular topic I am working on.

After that comes the development phase where all the code is written for the front end and the back end. Since we are working Agile, the development phase consists of picking one user story at a time and finishing it completely before choosing another one.

The following step is the testing stage where the new features are tested with Unit and Integration tests to figure out if the work as intended and do not cause the application to break in any way.

After that the code is pushed to a GitLab repository where other team members can use it to do their work (Not applicable for the individual project).

After the code has been uploaded to Git a new phase of development can begin, or if the project is complete, it can be deployed as a fully functioning product.