

# Habitat Connect

Laura Rodríguez López  
Department of Computer Science  
Hanyang University  
Madrid, Spain  
lpostlaura@gmail.com

Christian Alexander Hernandez  
Department of Computer Science  
Hanyang University  
Dallas, United States  
ch407541@gmail.com

Marcel Plesky  
Department of Information Systems  
Hanyang University  
Zürich, Switzerland  
marcel.plesky68@gmail.com

**Abstract**—Habitat Connect is a web application designed for residents of Hanyang Habitat, our apartment complex, to stay connected, regardless of their cultural backgrounds. Our platform offers features like an announcement board, event calendar, washer checker, and maintenance request page. Residents can easily receive important updates, plan events, check laundry availability, and report maintenance concerns – all within a user-friendly interface. With Habitat Connect, we hope to foster a stronger sense of community and make everyday tasks more convenient for everyone in Hanyang Habitat.

## ROLE ASSIGNMENTS

In the table below, we can find a detailed breakdown of the roles and responsibilities assigned to each team member throughout the duration of the project.

Role	Name	Task Description
User/Customer	Laura	<ul style="list-style-type: none"><li>- Uses the software</li><li>- Establishes the conditions and features that the software must satisfy (requirements)</li><li>- Provides feedback</li><li>- Approves change requests</li></ul>
Software Developer	Christian	<ul style="list-style-type: none"><li>- Works with user/customer to gather their needs</li><li>- Designs and develops the software</li><li>- Tests and debugs the software</li></ul>
Development Manager	Marcel	<ul style="list-style-type: none"><li>- Supervises the software development process</li><li>- Facilitates communication between user/customer and software developer</li><li>- Manages tools and resources</li><li>- Monitors progress and quality of the software</li></ul>

## I. INTRODUCTION

### A. Motivation

When international students arrive at Hanyang University, they often seek affordable and short-term housing options. One of the most popular accommodations are Goshiwons like Hanyang Habitat. As members of this team currently living in Habitat, we would like to provide residents with a new tool that caters to their needs.

Living in Hanyang Habitat has given us firsthand insight into the challenges faced by exchange students. With approximately 70 international residents that come from more than 15 countries, it's evident that online communication platforms

vary widely. For instance, Europeans prefer applications like WhatsApp or Instagram, while North Americans rely heavily on text messaging. In contrast, WeChat is very popular in certain Asian countries like China or Indonesia. In South Korea, KakaoTalk is the go-to messaging app, but its unique features and interface can pose a learning curve for newcomers, especially for those that do not speak Korean.

### B. Problem Statement

Given these cultural differences regarding communication tools, we asked ourselves: which platform should Hanyang Habitat residents use to effectively communicate with one another? Our proposed solution is Habitat Connect, an announcement board with a user-friendly English interface.

As well as serving as an online communication tool, Habitat Connect offers more features that focus on the needs of Habitat residents. This includes a laundry checker that provides real-time updates on washer availability, an event and birthday calendar to keep everyone informed about upcoming celebrations, and an easy-to-use maintenance request system, ensuring a direct line of communication between the tenants and the apartment manager.

Habitat Connect ensures that all residents, regardless of their language and online backgrounds, can comfortably navigate and utilize the tool. This idea comes from a desire to foster a sense of community among students in Hanyang Habitat. We aim to enhance the living experience of international students, making their time at Hanyang University more comfortable and rewarding.

### C. Research on related software

#### WHATSAPP

WhatsApp is a widely used messaging application that offers instant messaging, voice and video calls, file-sharing and even a poll-system that allows voting for user given options. WhatsApp is used in family, friends and even work environments to stay connected and up to date with your contacts.

#### KAKAOTALK

Just like WhatsApp, KakaoTalk is a popular messaging application. It offers even more features than WhatsApp like mobile payment and other various app services making it a multifunctional communication platform.

## DISCORD

Discord is known for its features like text and voice chat, creating and managing channels, and building communities around specific interests or topics. It's often used for both casual and structured discussions, making it a versatile option for promoting communication and collaboration in a variety of settings, including apartment buildings and shared living spaces.

## GOOGLE CALENDAR

Google Calendar is a user-friendly online calendar application from Google. It allows users to create, manage and share events, streamlining planning. Moreover, users can also import birthdays from their contacts, making it easier to remember important dates. Google integration and various viewing options make it a valuable time management and organization tool.

## FAMILYWALL

FamilyWall is an application designed to help families and individuals organize their lives, share information, and manage family-related activities. It allows users to create shared calendars, to-do lists, shopping lists, and share photos, as well as important information with family members or within their respective groups.

## LAUNDRY APPLICATIONS

Some laundromats like Samsung and LG have developed their own mobile apps to check the availability of washing machines and dryers.

## ZENDESK

Zendesk is a popular ticket management system used for customer support and issue tracking. It helps organize and track customer inquiries and support tickets. With Zendesk, businesses can provide better customer service.

## II. REQUIREMENT ANALYSIS

### A. Sign up/Log in

If users are not registered on the database, they must sign up with an email address and a password. Moreover, to complete the registration process, it is necessary to provide a username, birthdate, and Habitat's room number. Registered users can log in with their username and password.

### B. Recent announcements

Recent Announcements is the main page. The user is able to see all announcements ever posted, except those deleted by their corresponding authors. Regarding announcements, the following actions are available for the user:

- Post an announcement: user can write text and insert images in an announcement. Once they choose to post the announcement, it will be available in the Recent announcements page to every user registered in Habitat Connect. The ownership of the announcement is registered as well.

- Edit an announcement: users can edit their posted announcements.
- Delete an announcement: users can delete their posted announcements.
- Comment on an announcement: registered users can comment on posted announcements. Comments can only contain ASCII characters. Furthermore, the ownership of the comment is available for every user to see.

### C. Calendar view

The system will provide a calendar view for users that automatically displays other users' birthdays based on their sign-up information. The calendar view will be monthly, and it can be switched between months with the help of navigation arrows on each side of the calendar. Users should be able to click on a specific date on the calendar to view the list of birthdays for that date, including the names of the users celebrating their birthdays.

### D. Laundry checker

This function will be on the left side of the main page in the form of an icon, and once clicked, users will be taken to another page where they will see each washer, which is assigned a unique number. The status of each washer is displayed in the Status column, with different colors to make it visually clear: IN USE is displayed in red, AVAILABLE is displayed in green, and FINISHED is displayed in white. Visually, it will be in a table-like format for easy understanding. Each washer has a set of action buttons next to it. The available actions are:

- AVAILABLE (green button): users can click this button when they are going to use the washer. Once clicked, the washer's status will change to IN USE, in the color red.
- FINISHED (white button): once a user is done using the washer, they will click a button that will change the status back to AVAILABLE, displayed in green.

There may be additional features such as timers to display how long a washer has been in use, as well as user-friendly tooltips or information icons that can provide additional information or instructions.

### E. Maintenance requests

Under the laundry checker icon, we can find the Maintenance Request icon. Once it is clicked, users will be taken to a new page. There will be two text boxes to create a Ticket: the first textbox will be the ticket name (for example "No Hot Water"), and the second text box will be the description of the maintenance issue with a maximum of 200 characters. Once the ticket is set up, there will be a Submit Ticket button at the bottom of the page. Once clicked, an email will automatically be sent to the apartment manager's private email. The page for the user after clicking the Submit Ticket button will also change. In the first half of the page, we will find a message saying "Your ticket has been submitted!". Under that will be the status of the ticket which can be one of two options: Under Review (which will automatically be selected), and Ticket Completed. Once the user's request has been completed in

person, they can click the Ticket Completed option on the Maintenance Request page, which will update the user's page back to the original create a ticket page.

#### *F. Loading page*

Upon opening our website, a 3 second screen with just our logo will appear on the screen. This is the general convention for websites, we can see an example of this in X or discord. After 3 seconds, the page will load into the recent announcements page. Moreover, the implementation of this loading screen is not merely a matter of aesthetic preference; it serves a functional purpose as well. The brief delay allows essential elements of the website to load efficiently, resulting in a smoother and more responsive browsing experience for our users. This loading screen will only be seen upon reloading the site, not when switching between the different functions. The reason for this is for the user's convenience, as they won't need to wait for the page to load every time they want to use a new function.

#### *G. Notifications*

Users will only receive notifications regarding the status of their maintenance tickets, and the events that are occurring the day of. They can see their notifications on the top right of your screen, where the profile is located. Next to the profile, there will be a bell icon with a hand number, signifying how many notifications the user has received. Once the bell is clicked, users will see their notifications which again, can either be the ticket status or the events that are occurring that day. Users can click the Clear button to remove the notifications. However, once done, they cannot go back to see what was there. This will be the extent of the notifications, as we do not want to overflow it with other unnecessary added notifications.

#### *H. User authentication*

In order to ensure secure privacy is for all of our users, we will put in place an authentication function to ensure that the person logging in is the correct user. If someone is trying to log into their account with a registered user's email, and fails the password 5 times, that user will be locked out of trying to login the account for 1 hour. An email will be sent to the email that the unauthorized user tried to log in to, and if they click "NOT ME", that unauthorized user's ip will be banned from the website for a day, and the user will be forced to change their password. Another form of authentication we will use is an addition of a two step system. When a user logs in with their correct email and password, they will be asked to enter in a code, a code of which can be found in their email or by phone (depending on the one the user wants to use). This addition can be added in this way:

- Profile – Top right of screen
- Settings – Under profile once profile is clicked.
- User Authentication – One of the options in setting will be "User Authentication"
- Add Authentication – Once clicked, the user will see a confirmation popup saying: "Are you sure you want

to add Authentication?" and the options "Yes" or "No". Once "Yes" is clicked, the function will be in place.

#### *I. User feedback Support*

In order to manage the website without bugs, we need a substantial amount of feedback from users. This is a function that most every high level website has in order to enhance user experience. At the top right of the screen, where "Profile" is, one of the options under above "Settings" is a "Report Bug" option. Once clicked, the user will be taken to a separate page, where similar to an email, will have a Title and description text box with a maximum of 300 characters. Once both are filled and the user clicks "Submit", an email will automatically be sent to an admin regarding the bug. This will also be used as a "Support Ticket", whenever someone needs to get in contact with an admin regarding an issue they have about their account.

#### *J. User profile options*

The user profile is a fundamental component of an application, serving as a repository of user-specific information and a platform for users to manage and personalize their experience within the system. The user profile typically encompasses several key options and functionalities designed to cater to the diverse needs and preferences of the application's users. However, to keep the concept of this web application, the changes the user can make to their profile is limited. On the "Profile" section and "Edit" subsection, the user will be taken to a new page, where they can apply one of 3 options regarding their account. The first option is to switch to anonymous mode, where their room number and name will be hidden. With this, the user is still able to read the announcement chat and use the washer checker, however they cannot type in the announcement chat or create a maintenance manager ticket. The second option is to edit their username, as upon signing up for our website, their real names are used for the users accounts. The users only have the option to change their usernames once a month. The last option they have is to change their profile pictures. There is no limit to this option and the users can change it as many times as they please.

#### *K. User roles and permissions*

Our application will only need two roles. One being a member role given to tenants, which will have little to no permissions regarding what they can change to the website. The second role is admin, which will be given to the apartment manager as well as the website developers.

### **III. DEVELOPMENT ENVIRONMENT**

#### *A. Choice of software development platform*

Given that Habitat Connect is a web application, the choice of operating system is quite flexible. That is, we can choose any operating system that supports web development tools. To be more specific, our working environment consists of the following:

Name	Version	Description
Windows	11 Home	<ul style="list-style-type: none"> <li>- User-friendly environment with an updated UI</li> <li>- Wide range of web development tools and IDEs</li> </ul>
macOS		<ul style="list-style-type: none"> <li>- Unix-based</li> <li>- Offers a powerful terminal for web development</li> </ul>

Moreover, our development process will make use of some tools that can be found below.

- 1) **Svelte**: Svelte is a JavaScript library that streamlines the process of building user interfaces. It compiles components directly into efficient, imperative code. This leads to faster rendering and a smoother user experience.
- 2) **Bootstrap**: Bootstrap is a front-end framework that provides pre-designed, responsive, and customizable UI components. It allows us to create appealing and web applications with ease.
- 3) **Axios**: Axios is a promise-based HTTP client for the browser and Node.js. It simplifies making HTTP requests, handling responses, and managing asynchronous tasks in JavaScript. We will mainly use it for fetching data from APIs.
- 4) **MongoDB**: MongoDB is a NoSQL database that provides a flexible, document-based data structure. It's designed for handling large amounts of unstructured data, making it suitable for dynamic web applications.
- 5) **Visual Studio Code**: Visual Studio Code is a powerful code editor that supports various programming languages. Furthermore, this IDE offers features like syntax highlighting, debugging, extensions, and Git integration.
- 6) **Git/GitHub**: Git is a version control system that allows multiple developers to collaborate on a project. On the other hand, GitHub is a web-based platform that provides Git repositories, collaboration tools, and hosting services. It facilitates efficient team collaboration and code management.
- 7) **Google Drive**: Google Drive is a cloud-based storage and collaboration platform. It allows us to store files, share documents, and collaborate in real-time. It's useful for storing mock documents that will be pushed to GitHub.
- 8) **Overleaf**: Overleaf is an online platform for collaborative document editing. It simplifies the process of creating and editing scientific and technical documents, so it's very popular among researchers and academics. We will make use of it to create our LaTeX documents.

Regarding the choice of programming language, we will be using the following:

- 1) **JavaScript**: JS is a high-level programming language primarily known for its use in web development. It allows us to add interactivity and dynamic behavior to web pages. With Node.js, it can also be used for server-side development.

- 2) **HTML**: HTML is the standard language used for creating web pages. It provides the structure and content of a webpage, using elements like headings, paragraphs, and links.
- 3) **CSS**: CSS is a language used for styling web pages. It controls the visual presentation of HTML elements, including aspects like layout, colors, fonts, and animations.
- 4) **Python**: Python is a user-friendly and versatile programming language known for its simplicity and readability.

All software development tools are free of use. Furthermore, since Windows 11 and macOS are already in use, we will not count them as costs.

#### B. Software in use

#### C. Task distribution

The responsibilities of each member are yet to be determined. However, we are thinking of splitting the workload in the following categories:

- **Management**: responsible for checking up on each team member work and coordinating the overall work to meet due dates
- **Documentation**: responsible for writing the pdf and creating the LaTeX files
- **Front-end**: responsible for developing the front-end side of the software
- **Back-end**: responsible for developing the back-end side of the software

Our idea is that each team member will choose which skills they want to improve best, as long as the work is fairly divided. We believe that not dividing the tasks will result in an inefficient and not realistic way of software development.

## IV. SPECIFICATIONS

### REFERENCES

- [1] Facebook for Developers, "WhatsApp Business API - Documentation," Facebook for Developers. [Online]. Available: <https://developers.facebook.com/docs/whatsapp/>
- [2] "KakaoTalk - Services," Kakao Corporation. [Online]. Available: <https://www.kakaocorp.com/page/service/service/KakaoTalk?lang=en>
- [3] "GitHub - Discord," GitHub. [Online]. Available: <https://github.com/discord>
- [4] "Google Calendar API Documentation," Google Developers. [Online]. Available: <https://developers.google.com/calendar>
- [5] "CI Hackathon July 2020 - GitHub Repository," GitHub. [Online]. Available: [https://github.com/maliahavlicek/ci\\_hackathon\\_july\\_2020](https://github.com/maliahavlicek/ci_hackathon_july_2020)
- [6] "Zendesk - GitHub Repository," GitHub. [Online]. Available: <https://github.com/zendesk>
- [7] "Svelte - GitHub Repository," GitHub. [Online]. Available: <https://github.com/sveltejs/svelte>
- [8] "Bootstrap - GitHub Repository," GitHub. [Online]. Available: <https://github.com/twbs/bootstrap>

- [9] "Axios - GitHub Repository," GitHub. [Online]. Available: <https://github.com/axios/axios>
- [10] "MongoDB," MongoDB. [Online]. Available: <https://www.mongodb.com>
- [11] "Visual Studio Code - GitHub Repository," GitHub. [Online]. Available: <https://github.com/microsoft/vscode>
- [12] "Git," Official Git Website. [Online]. Available: <https://git-scm.com/>
- [13] "Overleaf - Documentation," Overleaf. [Online]. Available: <https://www.overleaf.com/learn>
- [14] "JavaScript Documentation," Mozilla Developer Network (MDN). [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [15] "HTML Documentation," Mozilla Developer Network (MDN). [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTML>
- [16] "CSS Documentation," Mozilla Developer Network (MDN). [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/CSS>
- [17] "General Python FAQ," Python.org. [Online]. Available: <https://docs.python.org/3/faq/general.html>