# WARESPACE: A WEB, MOBILE AND DESKTOP APPLICATION FOR FREE SPACE FINDING AND LISTING, WITH BUILT IN LOCATION PINNING

ASPE, EUGENE LANCE
NAVAL, LIANA JOY
VIBAR, REGINA JULIANE

# THE NEEDS DOCUMENT

# OF

# WARESPACE: A WEB, MOBILE AND DESKTOP APPLICATION FOR FREE SPACE FINDING AND LISTING, WITH BUILT IN LOCATION PINNING

## 1. INTRODUCTION

The client is a small business owner who has difficulties finding convenient, and reasonably priced storage for his goods. He states that his place is small, and it is not sufficient to store his products. Likewise, there are also individuals who find it difficult to make use of their extra space, leaving it to pile up dirt and dust. And aside from putting up expensive advertisements and posting their spaces on social media sites, they do not have an easier and more convenient way to market their spaces.

As such, this project aims to create a solution to the problems encountered by individuals who have vacant spaces and those in need of space. It provides hosts a way to market their properties by listing them in the application and for interested renters to easily see and access.

## 2. MISSION STATEMENT

The mission of this project is to provide the following:

1. A system that allows individuals who have extra storage space and wants to monetize their property to list their space for rent.

This system allows these individuals to be able to market their spaces to a wide range of people, making it easy for them to find clients that would rent the space.

2. A system that allows individuals who are in need of free space to locate/find the suitable place for them.

The application provides easy access to a wide variety of spaces, enabling renters to choose the most suitable place for them and their needs.

3. A system that is easily accessible and usable for all regardless of computer experience.

The system must provide a user-friendly and learnable interface that can cater even the most unsophisticated users, encouraging them to actively use the system to list or find spaces.

3. TECHNICAL OBJECTIVES

| **Technical Objectives** | **Performance Measures** |
|---|---|
| Create a Mobile and Web application that allows users to register an account. | A registration and login module must be present.<br><br>Users must be able to login and out of the system (Test 1).<br><br>Test 1: The test consists of 2 sets, one for invalid passwords and one for valid passwords. Each set consists of 100 different combinations of passwords. The invalid set must have 0% success rate while the valid set must have a 100% success rate. |
| Create a Mobile and Web application that allows registered hosts to post their available spaces for rent. | Must include a space recording module.<br><br>Only registered host users can access the space recording module (Test 2.1, Test 2.2).<br><br>Test 2.1: The test would have two sets of population, the first set consists of a group of users that will use the module to post valid spaces for rent, while the second set consists of users that will use the module to post invalid spaces |

| | |
|---|---|
| | for rent. The first set, the valid group, must have a 100% success rate in posting their spaces for rent while the second set, the invalid group, should have a 0% success rate of posting their space.<br><br>Test 2.2: The spaces that were posted by the first set of Test 2.1 must be seen by different renters publicly. |

| | |
|---|---|
| Create a Mobile and Web application that allows registered renters to rent available spaces. | Should provide a simple ecommerce module.<br><br>Includes a simple and advanced database search module (Survey 1, Test 3.1).<br><br>Provides a map showing the available spaces for rent based on the location of the renter (Survey 2, Test 3.2).<br><br>Survey 1: System Usability Scale Survey of Database Search Module - The survey would ask different users whether the simple function is enough for them or if they prefer having the option of using an advanced search to further narrow down the results of their search. The developers would use the System Usability Scale and specifically tailor it to test whether the users prefer the option of advance search or just the simple search.<br><br>According to SUS having a score of 68 is considered to be average, thus the survey must at least have a score of 68.<br><br>Test 3.1: There should be 100 test scenarios 50 of which |

| | |
|---|---|
| | searches for an item that is in the database and 50 searches for something that is not in the database. The 50 scenarios that contain the item in the database should have a 100% success rate when searching a term related to it while the other 50 scenarios must have a 0% success rate.<br><br>Survey 2: System Usability Scale Survey of Map Module - The survey would ask different users whether the Map Module is usable and helpful. The developers would use the System Usability Scale and specifically tailor it to test whether the users find the Map Module usable.<br><br>According to SUS having a score of 68 is considered to be average, thus the survey must at least have a score of 68.<br><br>Test 3.2: There should be 50 location pinning instances and all 50 should be properly pinned within a 5% margin of error. |
| Create a desktop application that easily allows hosts to monitor the | Presence of a dashboard containing the lists of spaces for |

| | |
|---|---|
| status of the rented space and monitor all the current space listing they have. | rent, renters, and duration of their stay (Survey 3).<br><br>Survey 3: System Usability Scale Survey of Dashboard - The survey would ask different users whether the dashboard present in the desktop application is useful and usable. To achieve this, the developers would use the System Usability Scale and specifically tailor it to test the usability of the desktop dashboard.<br><br>According to SUS having a score of 68 is considered to be average, thus the survey must at least have a score of 68. |
| Create a front-end using usable and customizable 3rd party templates. | Usability survey results should at least have an “acceptable” rating (Survey 4).<br><br>Survey 4: System Usability Scale Survey of Front-End Design - The survey would ask different users whether the interface of the web is usable. To achieve this, the developers would use the System Usability Scale and specifically tailor it to test whether the front-end is usable. |

| | |
|---|---|
| | According to SUS having a score of 68 is considered to be average, thus the survey must at least have a score of 68. |
| Create a module for users to edit their existing account information. I.e whether they are a “host” or a “renter” or their account password or their account name. | The system must be able to change the data embedded in the database according to the data that the user wants to change (Test 4.1).<br><br>Presence of "Switch to Hosting" and "Switch to Renting" button that allows users to easily switch user types (Test 4.2).<br><br>In addition, it must also have a “Logout” button that logs them out of the application (Test 4.3).<br><br>Test 4.1: The test consists of 50 different scenarios consisting of different combinations of information to update. 25 of these combinations are valid while the rest of 25 are invalid. The 25 valid combinations must have a 100% success rate of updating the information while the other 25 invalid combinations must have a 0% success rate of updating.<br><br>Test 4.2: The test consists of 15 |

| | |
|---|---|
| | different users testing out whether they can switch between host and renter. The test would last 10 minutes and they would switch back and forth during that time. During the duration of the test, users must not encounter renter elements when they are in host mode and vice versa.<br><br>Test 4.3: The test consists of 15 different users testing out whether they can safely logout. The test would last 10 minutes and they would login and logout during that time. During the duration of the test, users must not encounter any errors with regards to logging in and logging out and at the same time, they must not have access to the application once they are logged out. |
| The web, mobile and desktop application must have minimal downtime. | The server that the website is being hosted on must only have an average of 3 hours of downtime per month. This data is based on websitesetup.org. This is to maximize the time that the website is accessible by all users. |

## 4. SCOPE AND LIMITATIONS

This project aims to provide individuals who need storage space and people with extra storage space with a solution to both of their problems through an application where they can list and find spaces suitable for them. The application caters to two types of users: the renter and the host. The host is able to list their extra spaces for rent through the application, while the renter can locate and choose any available spaces. By default, logging into a registered account will show the interface built for the renter, and to become a host, they can simply press a "switch to host" button in the application. To go back as a renter, the user can just press the "switch to renter" button in the user account setting module. It is also available on three different platforms: web, mobile, and desktop. Unlike the other two platforms (mobile and web), where both users (host and renter) are catered to, the desktop version of the application will be specifically built for the host’s and renter’s information management.

However, the application will not be able to provide communication between the host and renter through the app, but it will provide the host's contact information. Additionally, if the renter wants to check the site/space before renting, the system will not provide a schedule on when it is possible to do so. The host and renter must arrange the schedule themselves. Additionally, the system will not be liable for any damage done to the property during the stay of a renter. All further action regarding this case will solely depend on the host's action. Lastly, the system will not be able to provide security measures (e.g. cctv) for the safety of the items.

The project will take approximately 2 months to complete, unless any unforeseen circumstances occur that may lead to a delay in its completion.

CONCEPTUAL FUNCTIONAL MODEL

OF

WARESPACE: A WEB, MOBILE AND DESKTOP APPLICATION FOR FREE SPACE FINDING AND LISTING, WITH BUILT IN LOCATION PINNING AND MESSAGING

**Technical Objective 1: Create a system that allows users to register an account.**

OPERATIONAL SCENARIO 1: User Identification

There are instances where multiple users have the same name, same credentials or there may be multiple users accessing an application within the same computer. Normally, there will be no way of distinguishing between the different users, user a can act as user b and so on. Having users register an account makes it so that they have their unique identification within the application which helps not only the users but also the hosts and administrators differentiating who the users are.

OPERATIONAL SCENARIO 2: Users Needs to Log Out to Access Different Roles

In the instance that a user that is registered as a host, needed to rent a space, they are unable to do so if the account they have is

registered only as a host account. Therefore only one account is needed to be registered. There should be a way for users to switch between viewing the application as a renter and viewing the application as a host.

**Technical Objective 2: Create a Mobile and Web application that allows registered hosts to post their available spaces for rent.**

OPERATIONAL SCENARIO 1: Minimizing Excess Space/Wasted Free Space

There exists a certain group of people that is an owner of a certain space, may it be a garage, a vacant apartment or some sort of warehouse that are currently not being used by anyone and they themselves have no use for the space. To minimize the spaces that are being wasted, the application provides a way for these owners to post their own spaces to the application which are then visible to the public. In that way, someone interested in renting their space may take notice and contact the owner, therefore benefitting both the owner and the renter.

OPERATIONAL SCENARIO 2: Advertising Spaces for Rent

There also exists a group of people who want their spaces to be rented, the likes of boarding houses, apartments, hotels and other similar spaces but cannot find a way to find renters for various reasons. One could be that they are overshadowed by other more known spaces for rent. The application then can serve as an avenue for them to advertise their own space for rent. They can provide the necessary information such as the location, the price, pictures of the space. This way, they may gain renters who are specifically interested in their space.

**Technical Objective 3: Create a Mobile and Web application that allows registered renters to rent available spaces.**

OPERATIONAL SCENARIO 1: Renters Looking for Temporary Lodging

Students, Workers, Tourists and other groups of people who are not residents of a specific location and are just either visiting or they have a business in the area are usually searching for places that they temporarily stay in. The application can provide them information on the available spaces that they can lodge on and the renters can easily identify the spaces that are fit for their specific condition. If they are on a tight budget, they can search the available spaces within a specific range. If they need to lodge in a specific area, they can easily search through the available spaces using the map function of the application.

OPERATIONAL SCENARIO 2: Small Business Owners Looking for Temporary Storage

Since most small businesses are just starting out, they often lack the space needed to store their goods. They also often lack the proper budget to buy a proper storage warehouse and most spaces that are available are too big for the small business that they currently have. In order to solve this particular problem, the application provides an easy way for them to find a location that is suitable for their business which they can use as a temporary storage for their goods until they find or buy a more dedicated space for storage.

**Technical Objective 4: Create a desktop application that easily allows hosts to monitor the status of the rented space and monitor all the current space listing.**

OPERATIONAL SCENARIO 1: Keep Tabs on Listed Spaces

If a host has multiple spaces for rent, it can be quite confusing and takes an extra effort in keeping tabs on all of the spaces. They might get confused as to which spaces are rented and which spaces are available. The desktop application removes the need for manual checking of the spaces that an owner has and instead provides a centralized place to access and monitor the spaces that an owner has listed or are currently being rented.

OPERATIONAL SCENARIO 2: Minimize Free Space Downtime

If a particular space that was listed by a host already arrived at the end of the renter's duration of stay, they want to list it again as available as soon as possible. However, taking down the listing and posting it again is inconvenient and can increase the downtime of the availability of the space for rent. In the desktop application the host can easily mark the listing they have as rented and they can set it if they want to automatically mark it as available once the duration of stay of the renter has ended. This minimizes the downtime of the space hosts have for rent.

**Technical Objective 5: Create a front-end using usable and customizable 3rd party templates.**

OPERATIONAL SCENARIO 1: User Friendly

Applications with confusing user interface and design often leads to less users using the application. Therefore, the front-end of the application affects the motivation of users to keep using the application. Users must be able to easily recognize and navigate the interface of the website, mobile and desktop application. In addition, the users using the application should easily adapt and learn the interface.

OPERATIONAL SCENARIO 2: Interchangeable Interface

If a user is already familiar with the interface of the mobile app, then using the web app and the desktop app should pose no problem and vice versa.

**Technical Objective 6: Create a module for users to edit their existing account information. i.e whether they are a “host” or a “renter” or both, their account password or their account name.**

OPERATIONAL SCENARIO 1: Misinput/Typographical Error

If a user at the time of registration has made a mistake on their credentials and was not able to change it, the only way for them to fix it is to create a new account with the correct credentials. This poses problems. (1) It is inconvenient to create an entirely new account and repeat all the steps needed to correct a single mistake. (2) There is a waste of data as the account that will not be used will still exist within the database. To solve this, there must be a way for users to change their credentials if it is necessary to do so.

OPERATIONAL SCENARIO 2: Lost Phone Number/Change of Contact Method

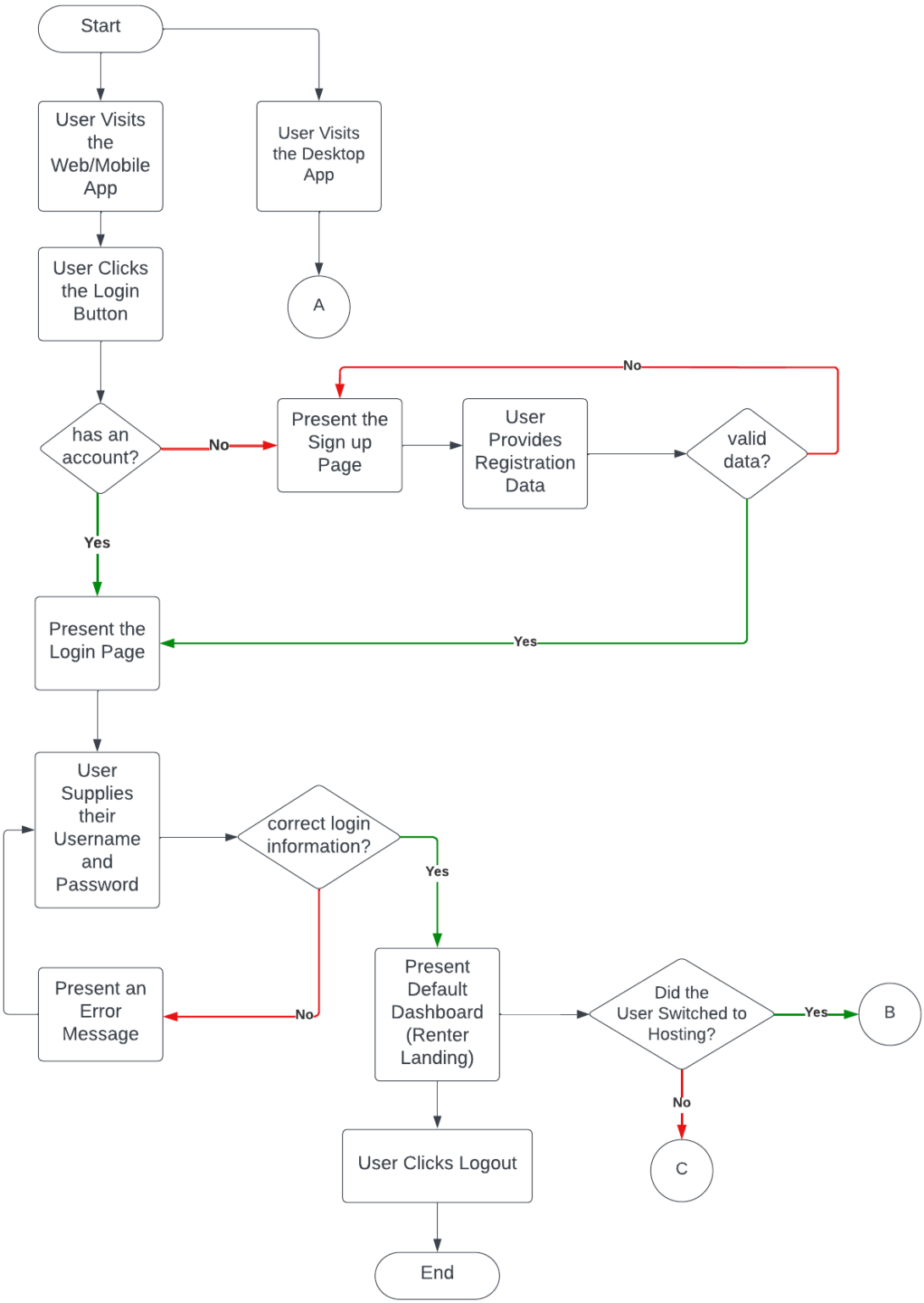
In the instance that a user loses their phone number or needs to change their contact method, they should be able to do so as if only outdated information is on their account, possible renters will not be able to properly contact the host and vice versa.

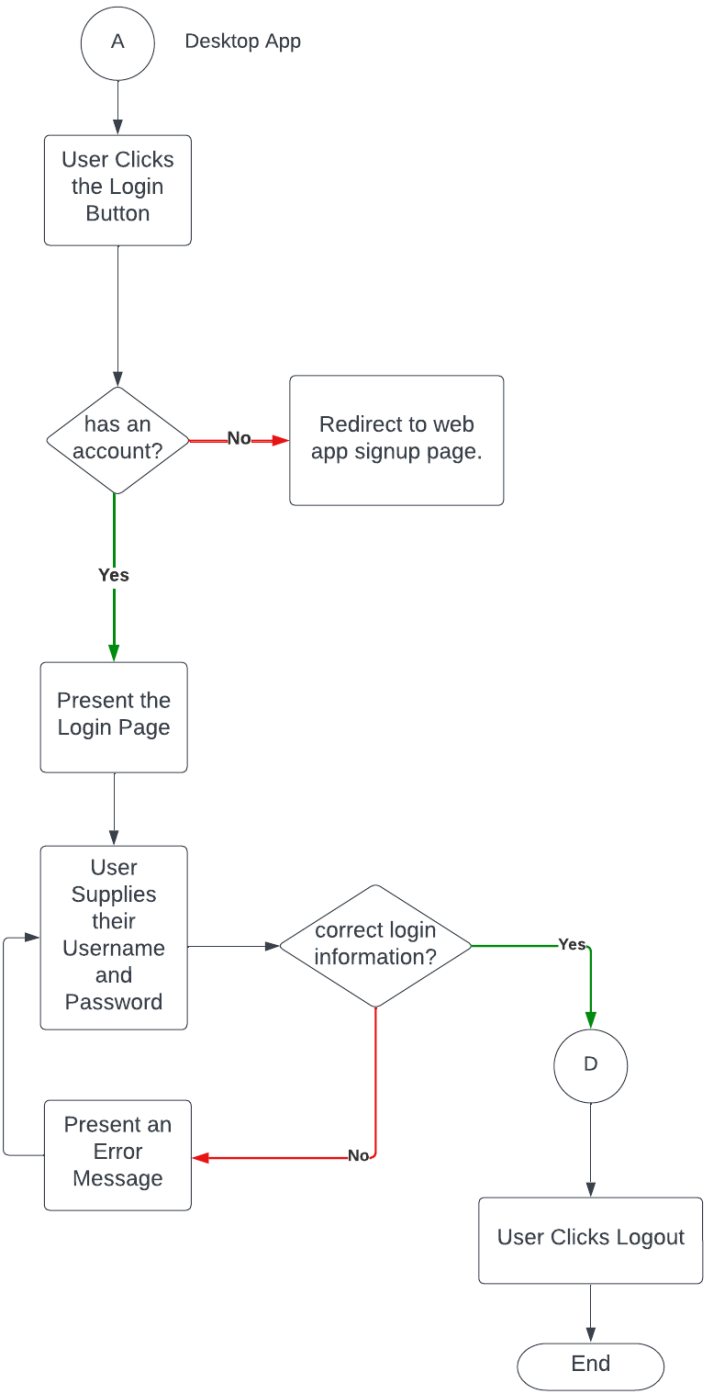
**Technical Objective 7: The web, mobile and desktop application must have minimal downtime.**

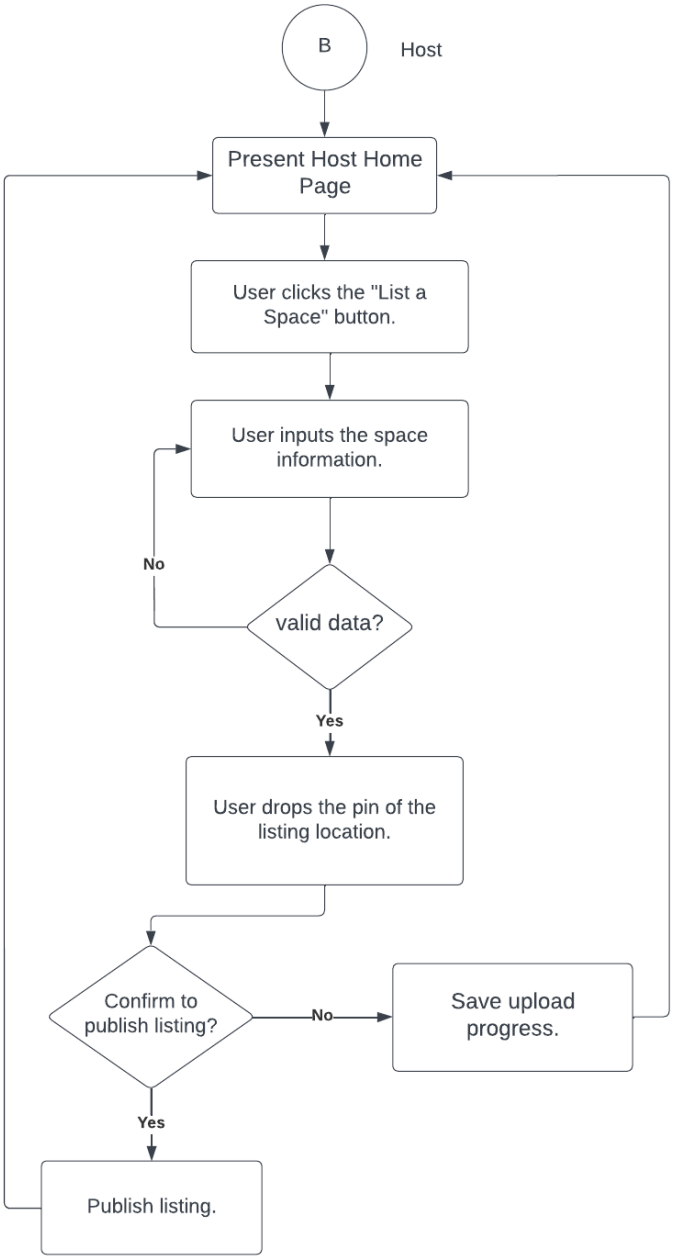
OPERATIONAL SCENARIO 1: Unforeseen Emergency

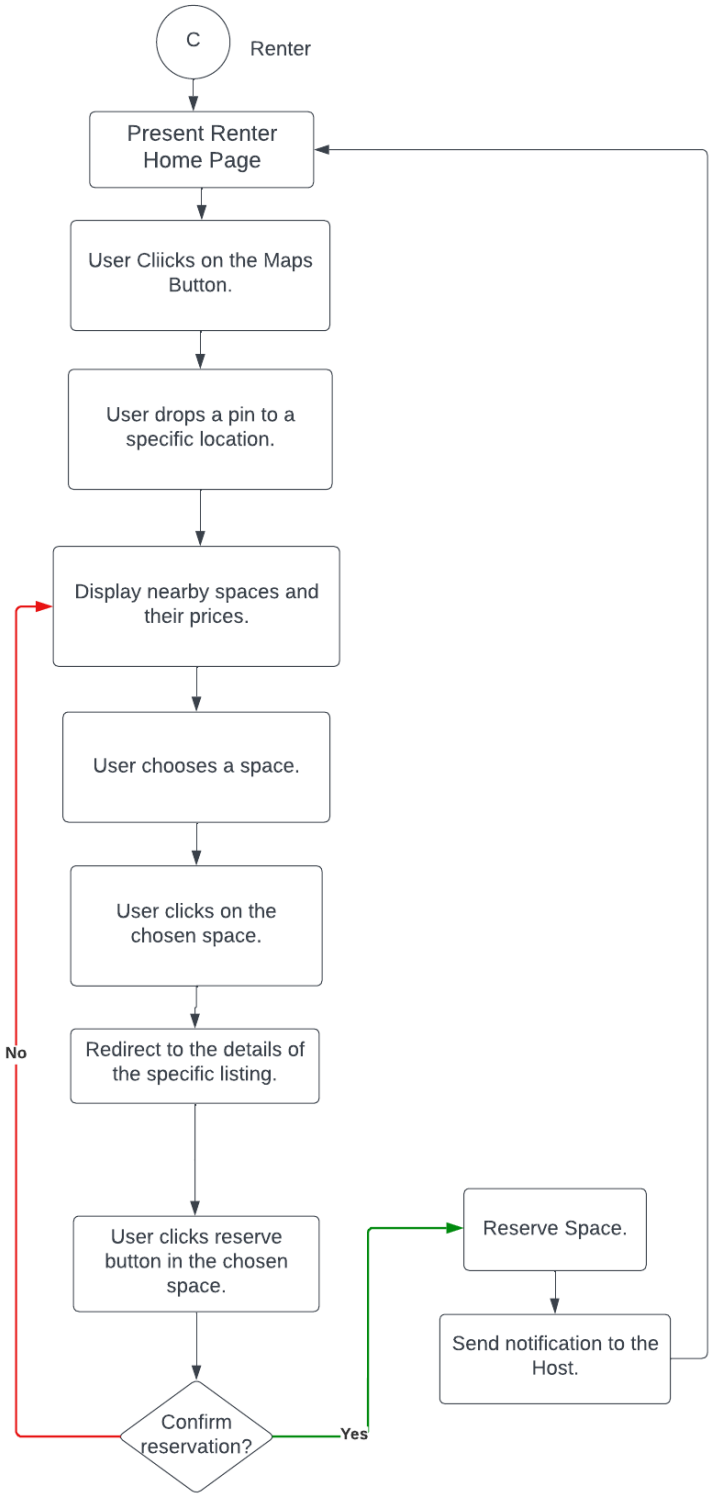
If in the instance there is someone suddenly needing a space to rent in the middle of the night they must be able to access the application. This also applies to any time of the day, users must be able to access the application whenever they need to do so. In order to achieve this, the application itself must have minimal downtime and if there will be a downtime, there should be an announcement to all users notifying them beforehand i.e for maintenance.

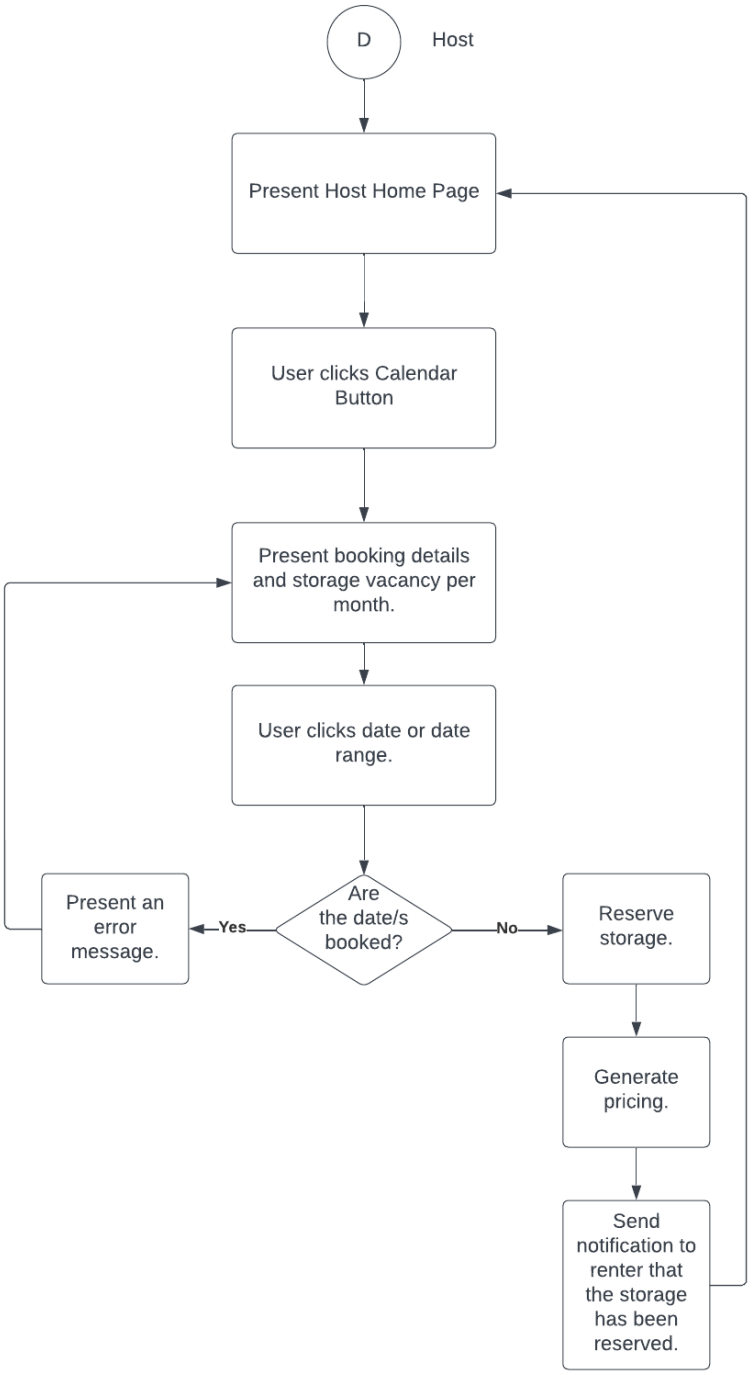
# System Activity Diagram



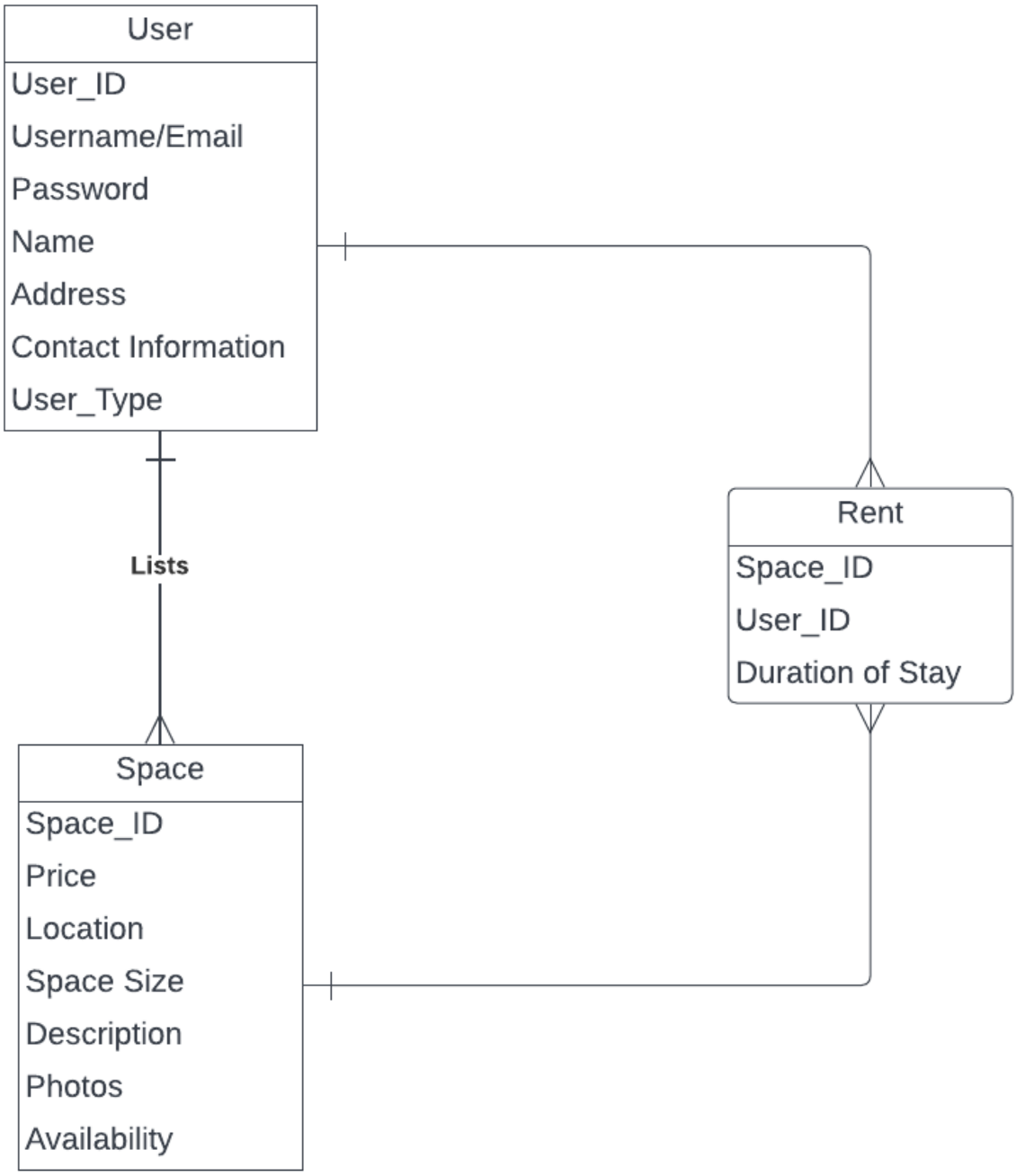

A
Desktop App
User Clicks the Login Button
has an account?
No
Redirect to web app signup page.
Yes
Present the Login Page
User Supplies their Username and Password
correct login information?
Yes
D
Present an Error Message
No
User Clicks Logout
End


B
Host
Present Host Home Page
User clicks the "List a Space" button.
User inputs the space information.
No
valid data?
Yes
User drops the pin of the listing location.
Confirm to publish listing?
No
Save upload progress.
Yes
Publish listing.


C
Renter
Present Renter Home Page
User Cliicks on the Maps Button.
User drops a pin to a specific location.
Display nearby spaces and their prices.
User chooses a space.
User clicks on the chosen space.
Redirect to the details of the specific listing.
User clicks reserve button in the chosen space.
Confirm reservation?
No
Yes
Reserve Space.
Send notification to the Host.

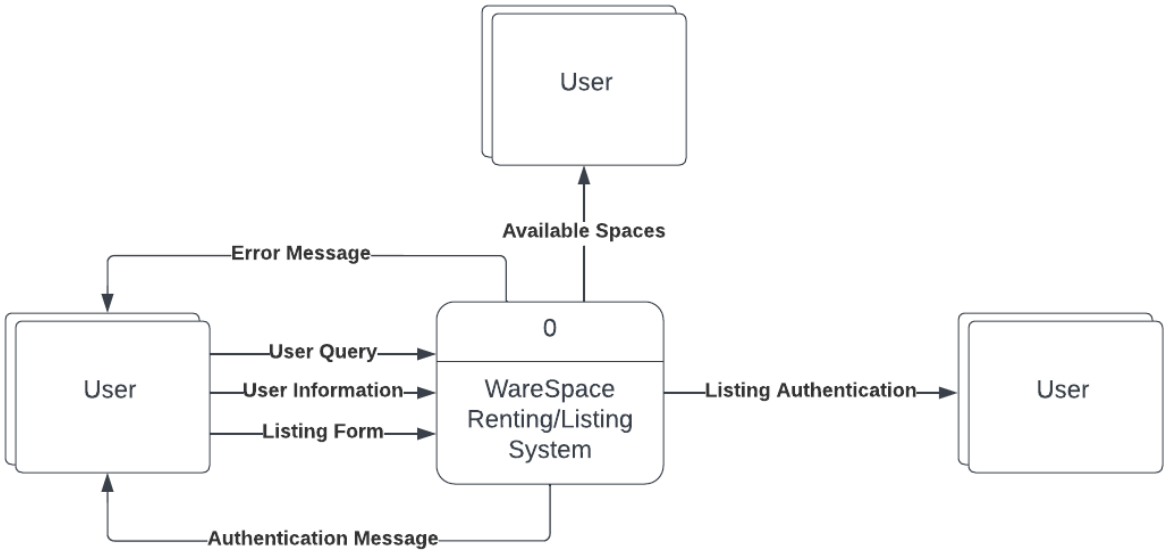

D
Host
Present Host Home Page
User clicks Calendar Button
Present booking details and storage vacancy per month.
User clicks date or date range.
Are the date/s booked?
Yes
Present an error message.
No
Reserve storage.
Generate pricing.
Send notification to renter that the storage has been reserved.

**ERD**

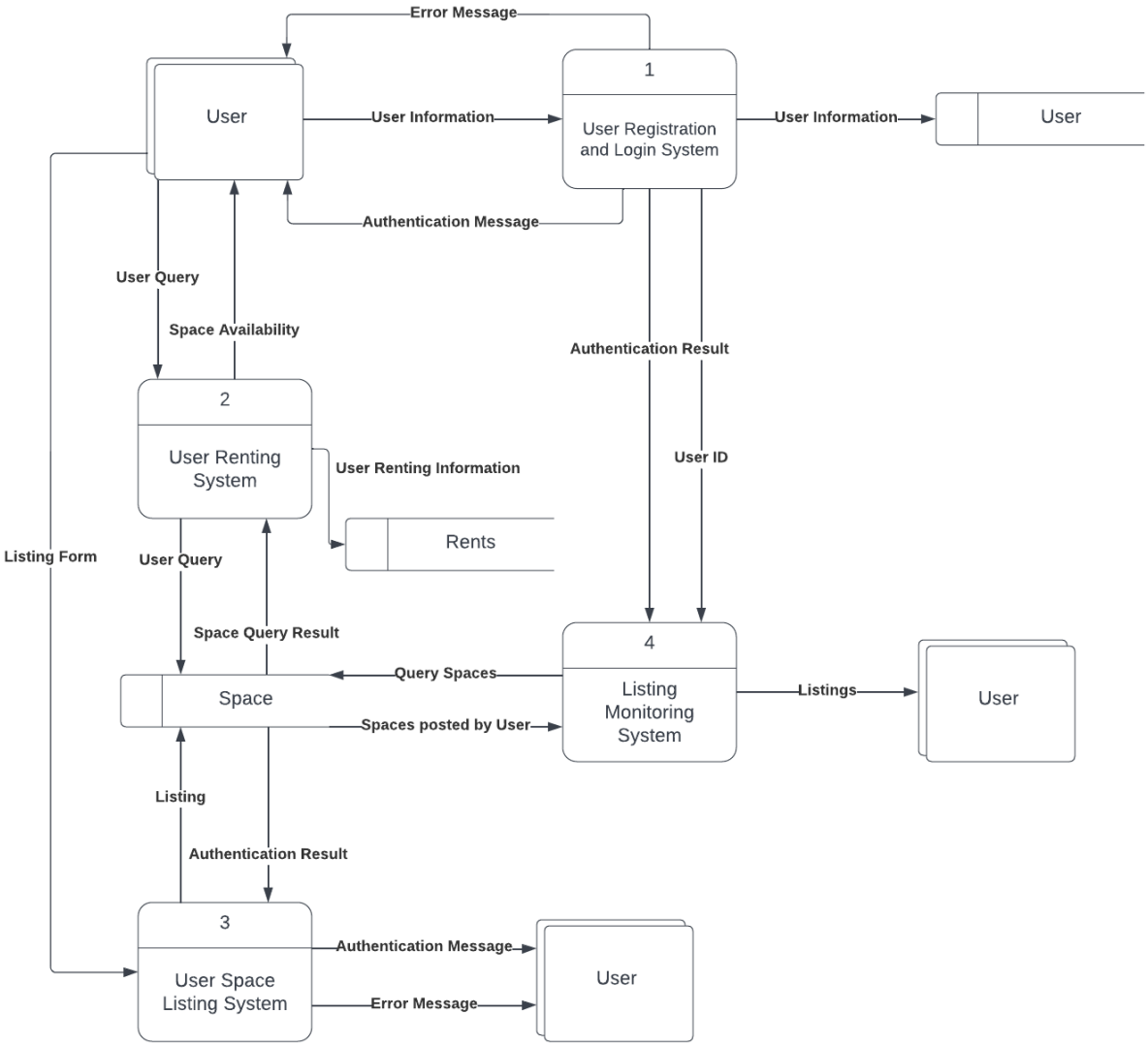

User
User_ID
Username/Email
Password
Name
Address
Contact Information
User_Type
Lists
Rent
Space_ID
User_ID
Duration of Stay
Space
Space_ID
Price
Location
Space Size
Description
Photos
Availability

## Data Flow Diagram

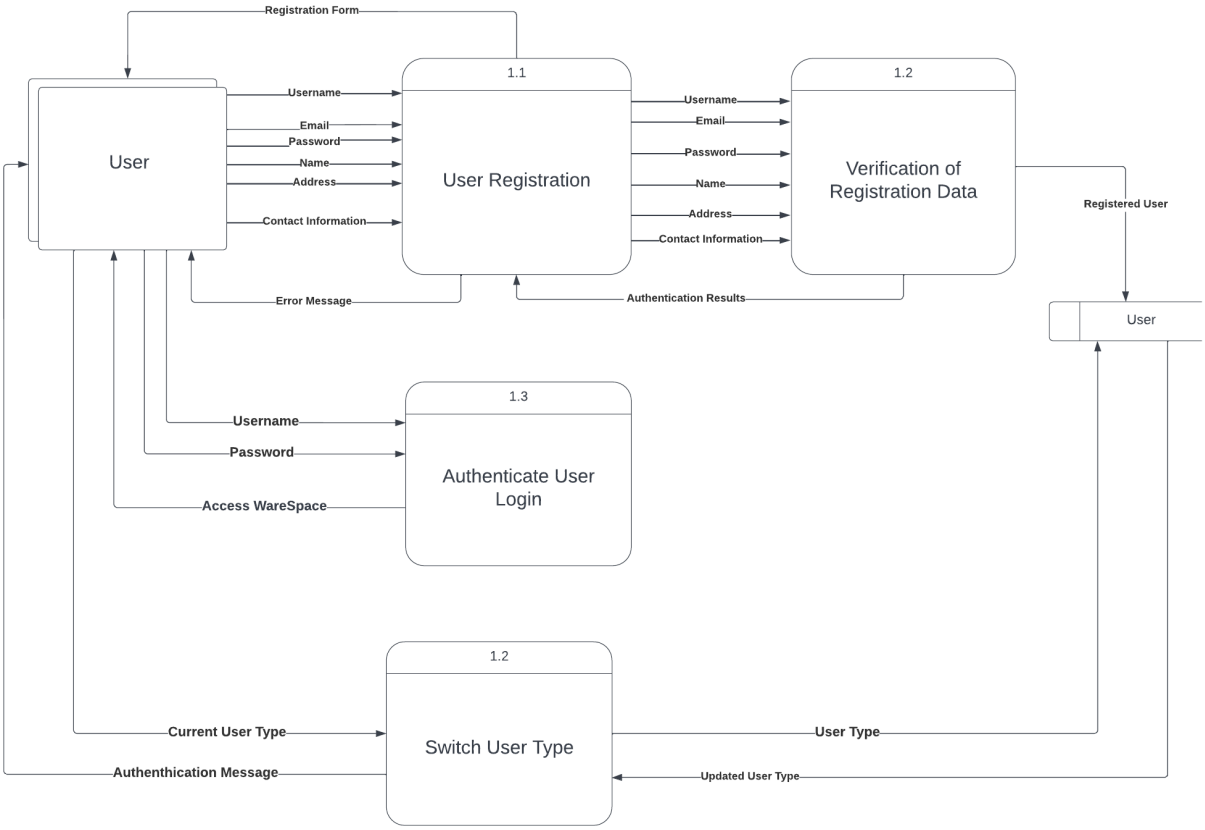
Context Level



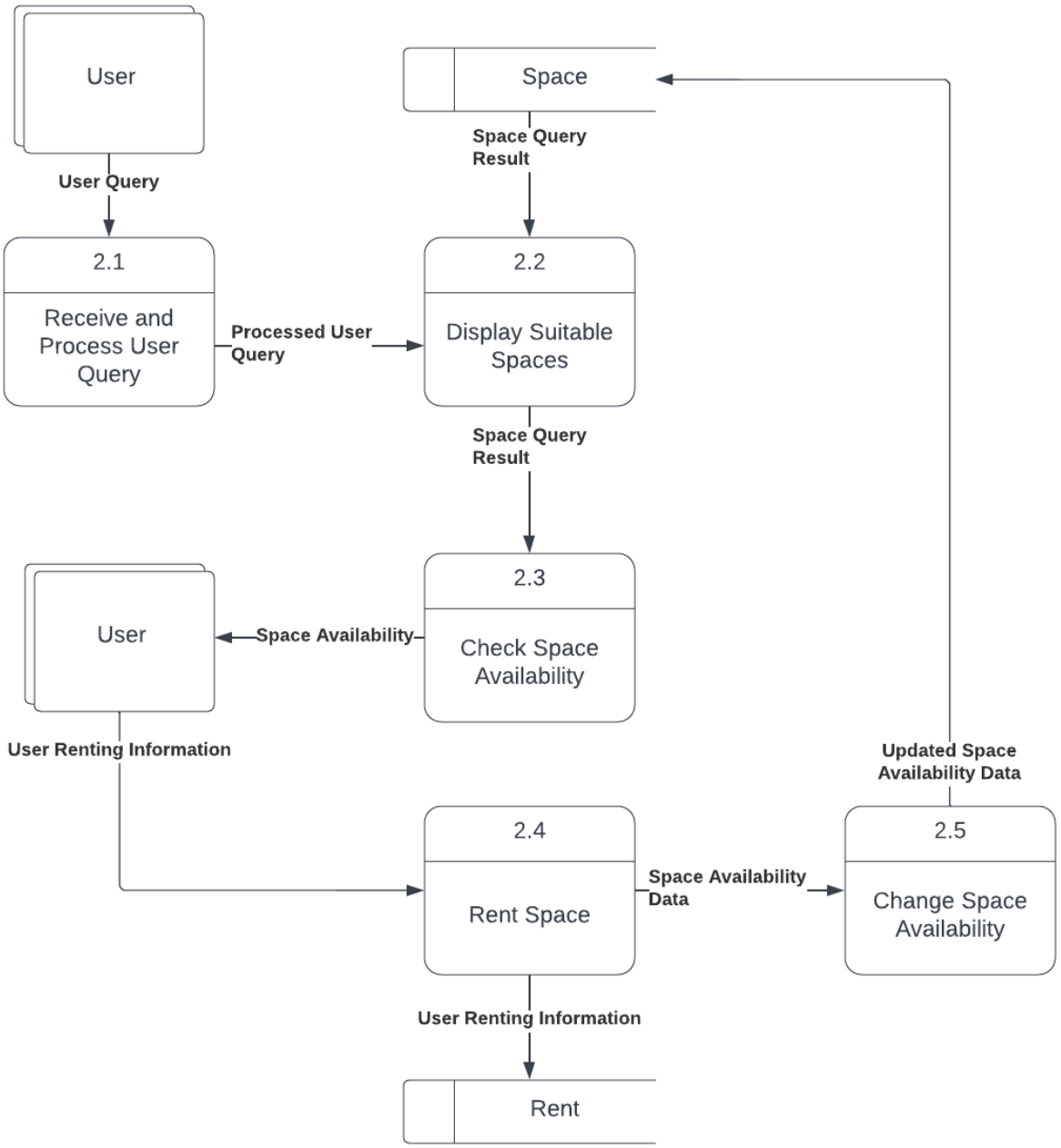
# Diagram 0


Error Message
User
User Information
1
User Registration and Login System
User Information
User
Authentication Message
User Query
Space Availability
Authentication Result
2
User Renting System
User Renting Information
Rents
User ID
Listing Form
User Query
Space Query Result
4
Query Spaces
Space
Listing Monitoring System
Listings
User
Spaces posted by User
Listing
Authentication Result
3
Authentication Message
User Space Listing System
User
Error Message

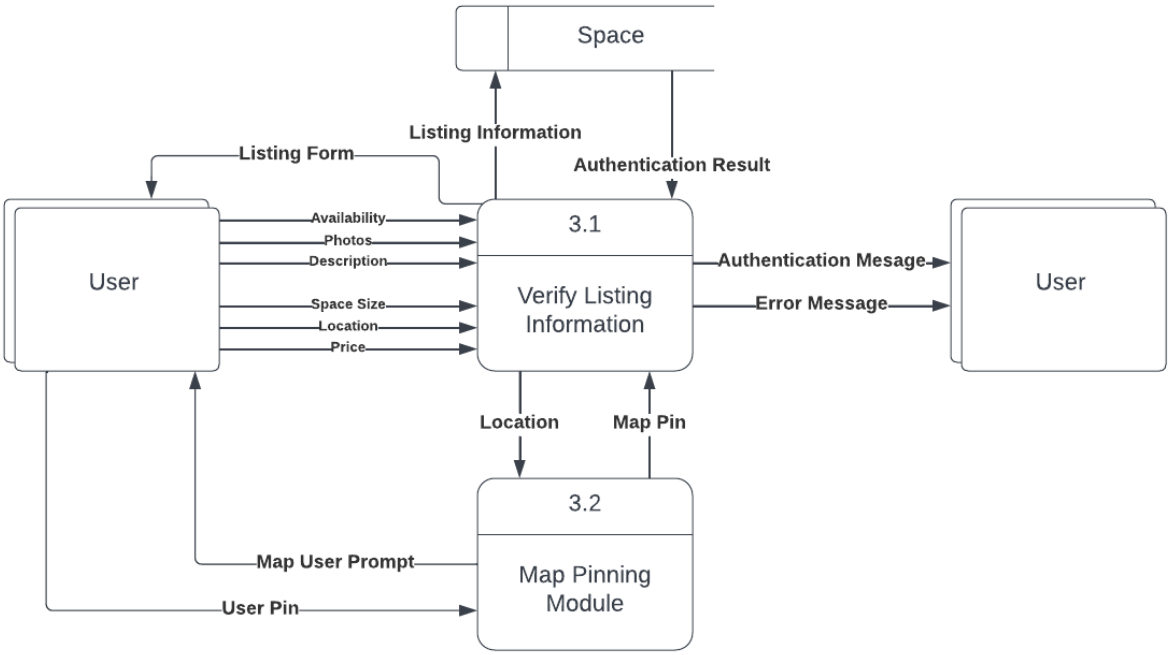
# 1. User Registration and Login System



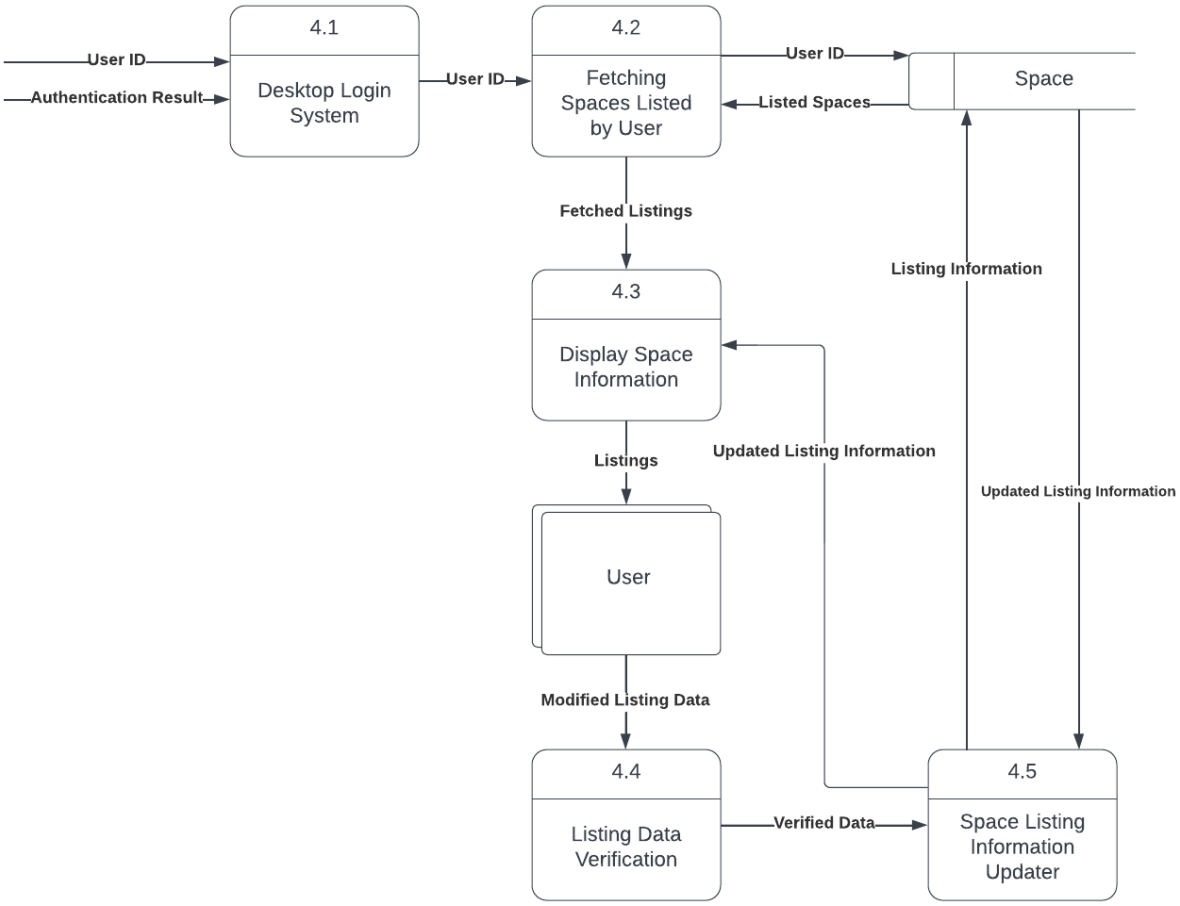
## 2. User Renting System



# 3. User Listing System

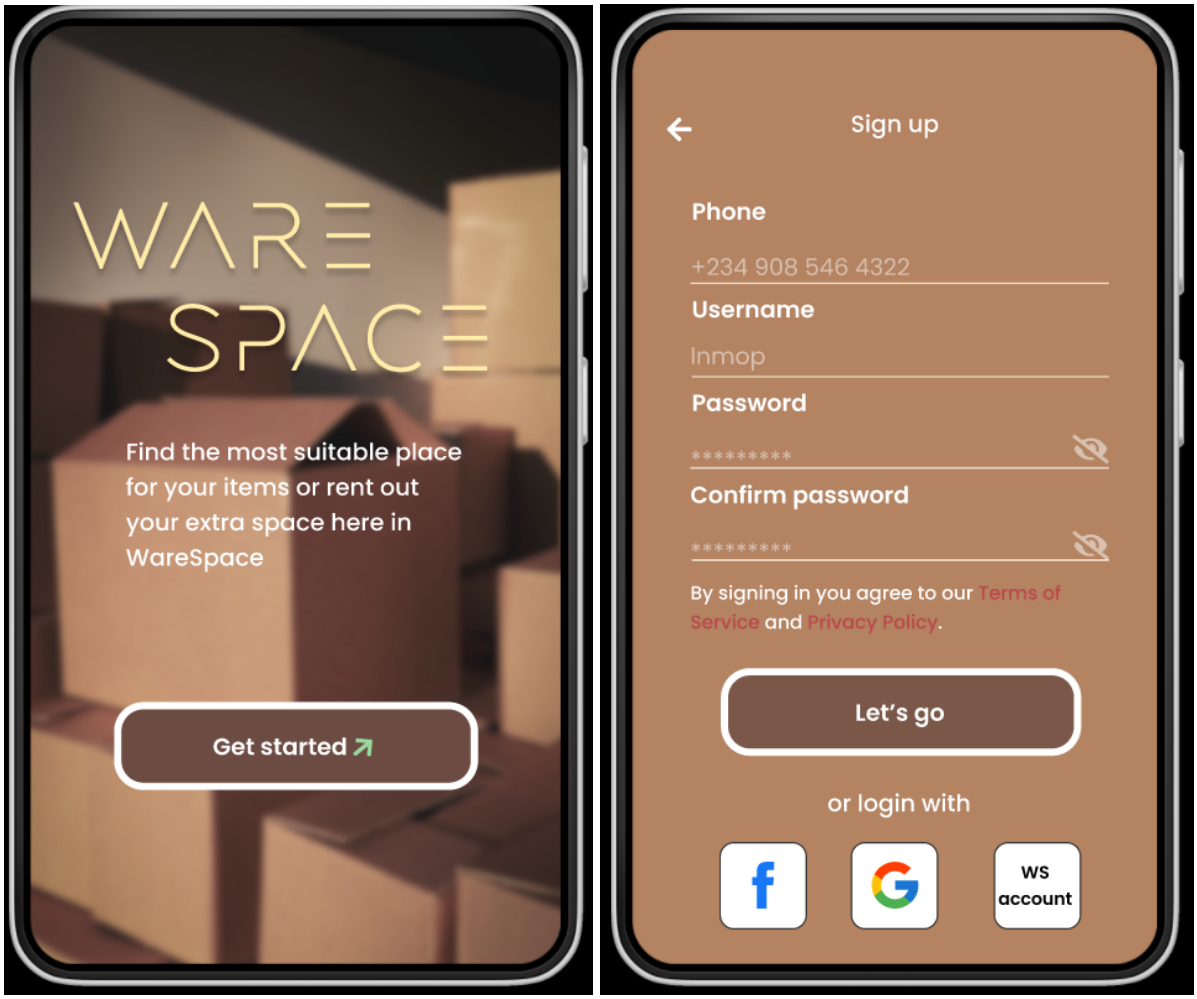


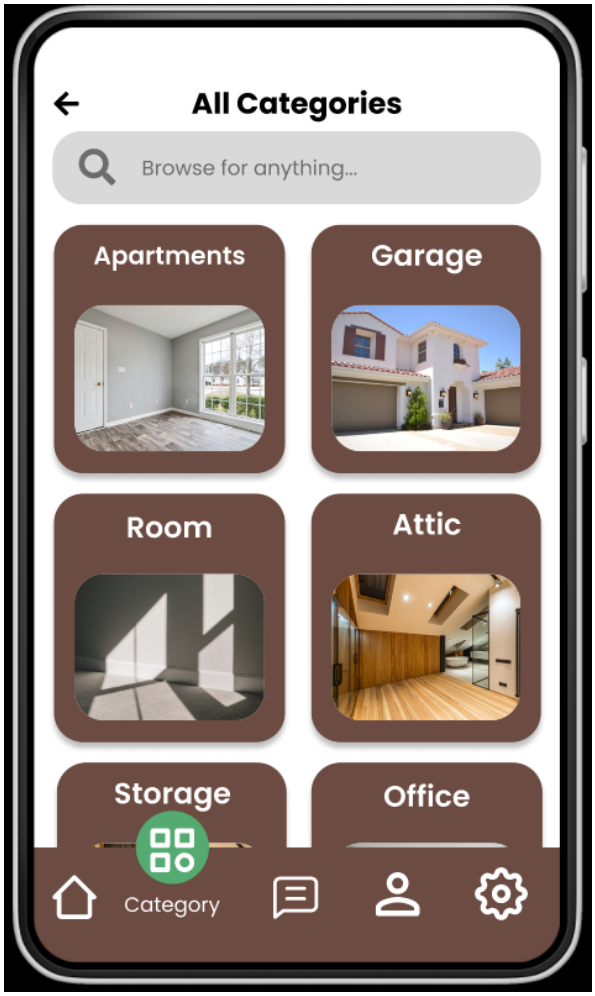
# 4. Listing Monitoring System

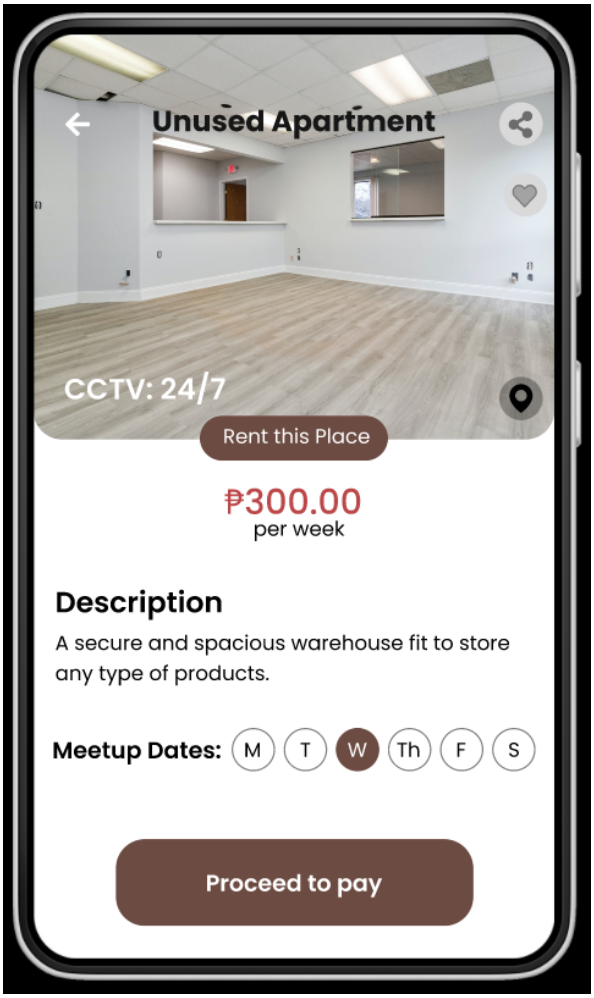


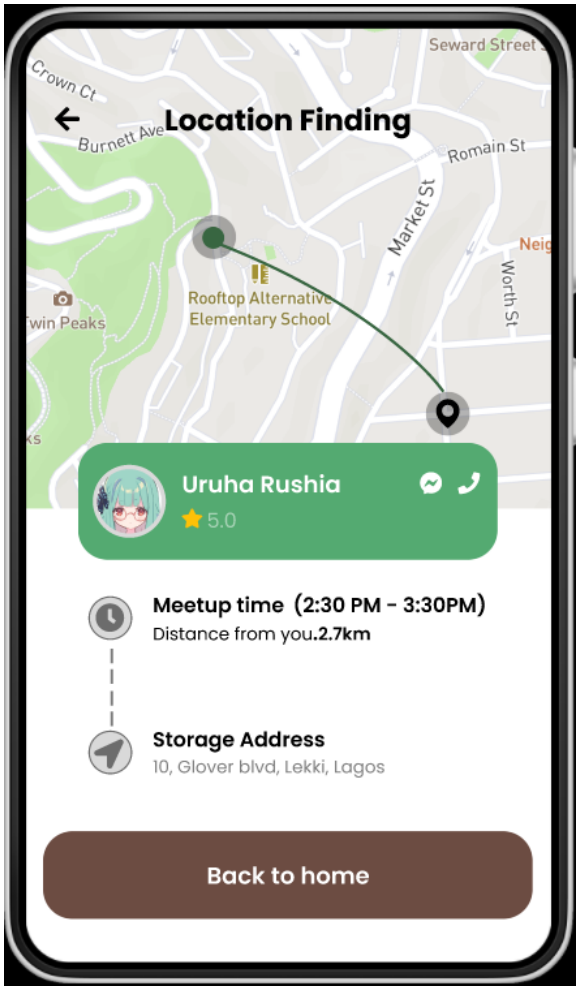
# Use-Case Diagram

## Mock Screenshots

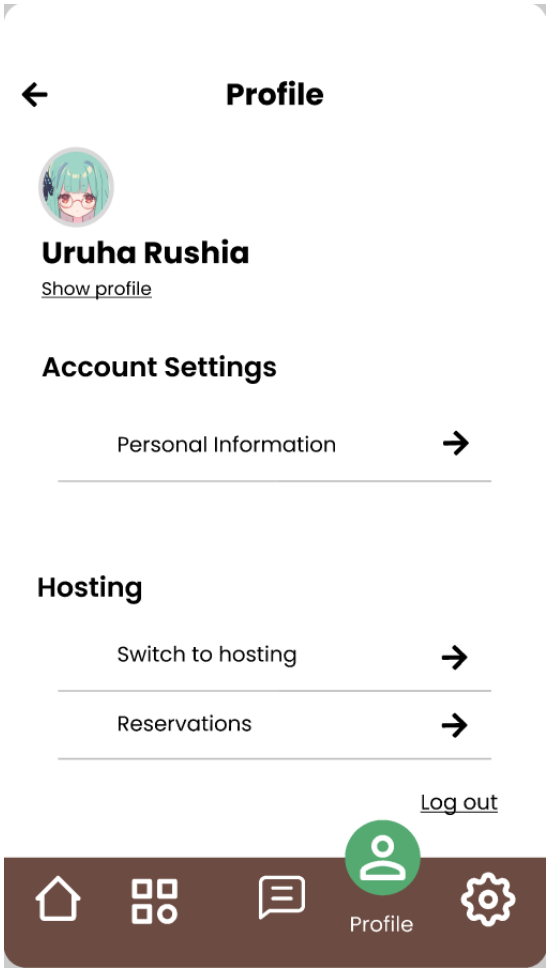



All Categories
Browse for anything...
Apartments
Garage
Room
Attic
Storage
Office
Category


Unused Apartment
CCTV: 24/7
Rent this Place
₱300.00
per week
Description
A secure and spacious warehouse fit to store any type of products.
Meetup Dates: M T W Th F S
Proceed to pay


Location Finding
Seward Street
Crown Ct
Burnett Ave
Romain St
Market St
Worth St
Rooftop Alternative
Elementary School
Twin Peaks
Uruha Rushia
5.0
Meetup time (2:30 PM - 3:30PM)
Distance from you.2.7km
Storage Address
10, Glover blvd, Lekki, Lagos
Back to home


Name of Your Place
Upload This Place
₱00.00
per week
Description
"Type Your Description Here"
Floor Area 103 m²
Location City, State/Province/Region
Proceed to listing


Profile
Uruha Rushia
Show profile
Account Settings
Personal Information
Hosting
Switch to hosting
Reservations
Log out
Profile

PHYSICAL ALLOCATION MODEL

OF

WARESPACE: A WEB, MOBILE AND DESKTOP APPLICATION FOR FREE SPACE FINDING AND LISTING, WITH BUILT IN LOCATION PINNING AND MESSAGING

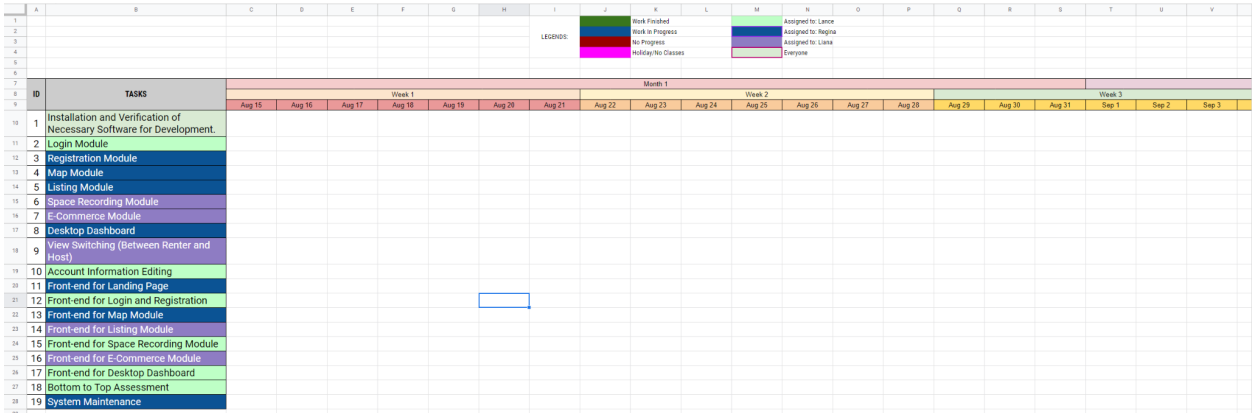
| Task | Assigned To | Approximation of Time Needed |
|---|---|---|
| Installation and Verification of Necessary Software for Development. | Everyone | 1 Day |
| Database | Lance | 2 hours |
| Login Module | Lance | 1 day |
| Registration Module | Regina | 1 day |
| Map Module | Regina | 2 days |
| Listing Module | Regina | 1 day |
| Space Recording Module | Liana | 2 days |
| E-Commerce Module | Liana | 2 days |
| Desktop Dashboard | Regina | 2 days |
| View Switching (Between Renter and Host) | Liana | 1 day |

| | | |
|---|---|---|
| Account Information Editing | Lance | 1 day |
| Front-end for Landing Page | Regina | 1 day |
| Front-end for Login and Registration | Lance | 1 day |
| Front-end for Map Module | Regina | 1 day |
| Front-end for Listing Module | Liana | 1 day |
| Front-end for Space Recording Module | Lance | 1 day |
| Front-end for E-Commerce Module | Liana | 1 day |
| Front-end for Desktop Dashboard | Lance | 1 day |
| Bottom to Top Assessment | Lance | 2 days |
| System Maintenance | Regina | 3 hours |

# GANTT CHART

# OF

# WARESPACE: A WEB, MOBILE AND DESKTOP APPLICATION FOR FREE SPACE FINDING AND LISTING, WITH BUILT IN LOCATION PINNING AND MESSAGING



| | ID | TASKS | Month 1 | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Week 1 | | | | | | | Week 2 | | | | | | | Week 3 | | | | | |
| | | | Aug 15 | Aug 16 | Aug 17 | Aug 18 | Aug 19 | Aug 20 | Aug 21 | Aug 22 | Aug 23 | Aug 24 | Aug 25 | Aug 26 | Aug 27 | Aug 28 | Aug 29 | Aug 30 | Aug 31 | Sep 1 | Sep 2 | Sep 3 |
| | 1 | Installation and Verification of Necessary Software for Development. | | | | | | | | | | | | | | | | | | | | |
| | 2 | Login Module | | | | | | | | | | | | | | | | | | | | |
| | 3 | Registration Module | | | | | | | | | | | | | | | | | | | | |
| | 4 | Map Module | | | | | | | | | | | | | | | | | | | | |
| | 5 | Listing Module | | | | | | | | | | | | | | | | | | | | |
| | 6 | Space Recording Module | | | | | | | | | | | | | | | | | | | | |
| | 7 | E-Commerce Module | | | | | | | | | | | | | | | | | | | | |
| | 8 | Desktop Dashboard | | | | | | | | | | | | | | | | | | | | |
| | 9 | View Switching (Between Renter and Host) | | | | | | | | | | | | | | | | | | | | |
| | 10 | Account Information Editing | | | | | | | | | | | | | | | | | | | | |
| | 11 | Front-end for Landing Page | | | | | | | | | | | | | | | | | | | | |
| | 12 | Front-end for Login and Registration | | | | | | | | | | | | | | | | | | | | |
| | 13 | Front-end for Map Module | | | | | | | | | | | | | | | | | | | | |
| | 14 | Front-end for Listing Module | | | | | | | | | | | | | | | | | | | | |
| | 15 | Front-end for Space Recording Module | | | | | | | | | | | | | | | | | | | | |
| | 16 | Front-end for E-Commerce Module | | | | | | | | | | | | | | | | | | | | |
| | 17 | Front-end for Desktop Dashboard | | | | | | | | | | | | | | | | | | | | |
| | 18 | Bottom to Top Assessment | | | | | | | | | | | | | | | | | | | | |
| | 19 | System Maintenance | | | | | | | | | | | | | | | | | | | | |

LEGENDS:
- Work Finished
- Work In Progress
- No Progress
- Holiday/No Classes
- Assigned to: Lance
- Assigned to: Regina
- Assigned to: Liana
- Everyone

The Gantt Chart that will be used in this project is in a separate file: https://docs.google.com/spreadsheets/d/1Co5HLHrezrJ1GETRr19p1UqRV-9WKL9RfSQ6Mpe-fIs/edit?usp=sharing