

Design & Testing Document

Project 7: Hot-Desking App - Admin Side

University of British Columbia Okanagan

COSC 499, Summer 2022



OKANAGAN

Damyn Filipuzzi: damynfilipuzzi@gmail.com

Ravi Bullock: ravioliasb@gmail.com

Serena Chen: serenachiyin@gmail.com

Suyash Shingare: suyashshingare@gmail.com

1. Project Description	3
2. UML and User Groups	3
2.1 User Groups	3
2.2 UML Diagram	4
2.2 Use Case Breakdown	5
3. System Architecture	18
3.1 ER Diagram	19
3.2 Data-Flow-Diagram	20
3.3 Sequence Diagram	21
3.3.1 Manage Workspace Sequence Diagram	22
4. UI mockups	23
5. Technical specification	28
6. Test plan	29
6.1 Unit Testing	29
6.2 Integration Testing	29
6.3 System Testing	29
6.4 Acceptance Testing	29
6.5 User Testing	29

1. Project Description

The objective of this project is to develop an application that will allow students, staff, and visitors of the University of British Columbia to seek and reserve available desks on their campus. The development of the application is in an effort to resolve the problem of individuals not being able to find available spaces to work and study on their campus. We will be developing the administrative component of the application that allows administrators to define spaces (buildings, floors, rooms, and desks) around their campus. Administrators will also be able to define the resources within these spaces, assign usage rules to users, and provide the users with the times when these spaces are available. Furthermore, administrators will be able to manage users by assigning user roles and relationships, deleting users, and changing user permissions. Lastly, administrators will be able to access usage statistics of resources to gain feedback on how space is being used and improve space management in the future.

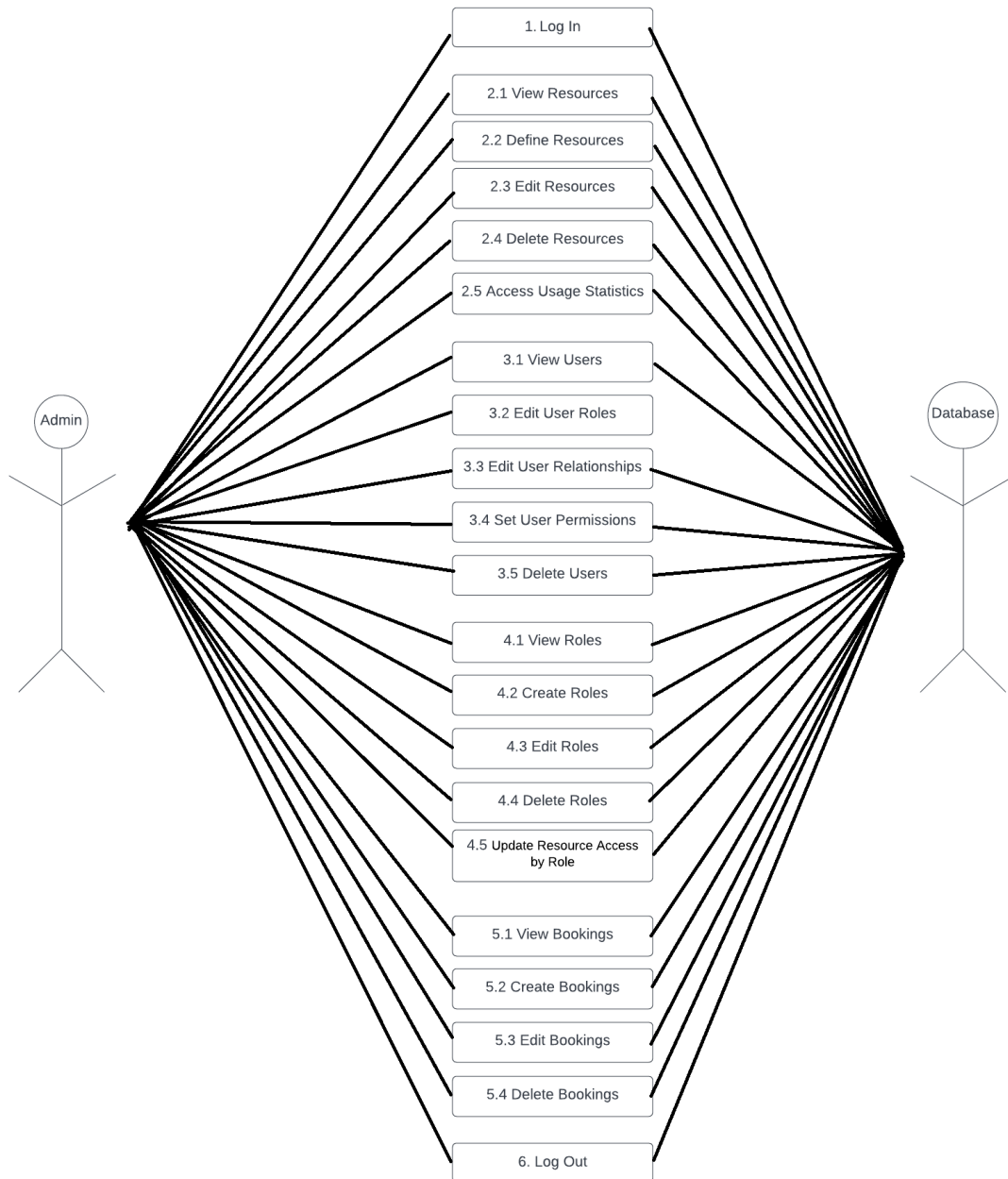
2. UML and User Groups

2.1 User Groups

Administrators: This will be the only user group. All administrators using the administrative component of the application will have the same privileges and permissions. There will be no super administrators or sub-administrators. We assume that administrators have low technical literacy.

2.2 UML Diagram

This Use-Case UML Diagram shows the expected usage scenarios between the administrator and the application. The administrator will be able to perform the tasks shown in the use-case descriptions and these tasks will be associated with the system's database. The interactions between the two actors are shown in the UML diagram below.



2.2 Use Case Breakdown

Use Case ID	Use Case Name	Actor(s)
1	Log in	Administrator
2	Manage Resources	Administrator
2.1	View Resources	Administrator
2.2	Define Resources	Administrator
2.3	Edit Resources	Administrator
2.4	Delete Resources	Administrator
2.5	Access Usage Statistics	Administrator
3	Manage Users	Administrator
3.1	View Users	Administrator
3.2	Edit User Roles	Administrator
3.3	Edit User Relationships	Administrator
3.4	Set User Permissions	Administrator
3.5	Delete Users	Administrator
4	Manage Roles	Administrator
4.1	View Roles	Administrator
4.2	Create Roles	Administrator
4.3	Edit Roles	Administrator
4.4	Delete Roles	Administrator
4.5	Update Resource Access by Role	Administrator
5	Manage Bookings	Administrator
5.1	View Bookings	Administrator
5.2	Create Bookings	Administrator
5.3	Edit Bookings	Administrator
5.4	Delete Bookings	Administrator
6	Log Out	Administrator

Use Case ID:	1
Use Case:	Log in
Actor(s):	Administrator
Description:	Administrators will be able to log into the system with their authorized account using username and password
Pre-Condition:	<ul style="list-style-type: none"> • Administrator is human • Administrator has an account • Administrator is logged out
Post-Condition:	<ul style="list-style-type: none"> • Administrator is logged in • Administrator will be able to access admin panel
Flow:	<ol style="list-style-type: none"> 1. Administrator will click on website 2. Administrator will input their username 3. Administrator will input their password 4. System will check database to verify that credentials are valid 5. Administrator will gain access to Admin Panel

Use Case ID	2
Use Case	Manage Resources
Actors	Administrator
Description	Administrators will be able to manage all buildings, rooms, desks, and resources
Pre-Condition:	<ul style="list-style-type: none"> • Administrator is logged in • Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> • Administrator manages buildings, rooms, desks, and resources • View or edit campuses, buildings, floors, rooms, and desks
Flow	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel 3. Administrator will click on Manage Resources button

Use Case ID	2.1
Use Case	View Resources
Actors	Administrator
Description	Administrators will be able to upload blueprints for the campus map, floor plans of buildings, and the arrangement of desks.
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator view previously defined Resources
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Resources button System will show a list of resources in database

Use Case ID	2.2
Use Case	Define Resources
Actors	Administrator
Description	Administrators will be able to upload blueprints for the campus map, floor plans of buildings, and the arrangement of desks.
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator defines spaces based on their resources and location Add image file to the rooms table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Resources button Administrator will click on Define Resource button System adds new resources to database

Use Case ID	2.3
Use Case	Edit Resource

Actors	Administrator
Description	Administrators will be able to edit the building plans, floor plans, and room plans
Pre-Condition:	<ul style="list-style-type: none"> • Administrator is logged in • Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> • Administrator changes previously defined resources • Updates one or many tables from the campus, buildings, floors, rooms, or desks tables
Flow:	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel 3. Administrator will click on Manage Resources button 4. Administrator will click on Edit Resource button 5. System edits resources in database

Use Case ID	2.4
Use Case	Delete Resource
Actors	Administrator
Description	Administrator will be able to delete plans of campuses, buildings, floors, and rooms
Pre-Condition:	<ul style="list-style-type: none"> • User is logged in • User accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> • Resource removed • Empty space created • Deletes one or many tables from the campus, buildings, floors, rooms, or desks table
Flow:	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel 3. Administrator will click on Manage Resources button 4. Administrator will click on Delete Resource button 5. System deletes resource from database

Use Case ID	2.5
Use Case	Access Usage Statistics

Actors	Administrator
Description	Administrators will be able to see the usage statistics of resources
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator views tables of usage statistics of each resource, user, role or department
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Campus button Administrator will click on Access Usage Statistics button System releases usage data from database in data

Use Case ID	3
Use Case	Manage Users
Actors	Administrator
Description	Administrators will be able to change user roles, relationships, and permissions. They will also be able to ban and delete users.
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator edits information on users Update users table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Users button

Use Case ID	3.1
Use Case	View Users
Actors	Administrator
Description	<ul style="list-style-type: none"> Administrators will be able to view users and their information
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in

	<ul style="list-style-type: none"> Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrators views users in a table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Users button System shows a list of users in database

Use Case ID	3.2
Use Case	Edit User Roles
Actors	Administrator
Description	Administrator can edit user role as being administrator, undergraduate student, graduate student, or staff member
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator edits user roles Update user_roles table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Users button Administrator will search for a certain user Administrator will click on Edit User Role button System updates roles in database

Use Case ID	3.3
Use Case	Edit User Relationships
Actors	Administrator
Description	Administrator can edit user relationships, such as if they have a supervisor
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator edits user relationships Update users table

Flow:	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel 3. Administrator will click on Manage Users button 4. Administrator will search for certain user 5. Administrator will click on Edit User Relationship button 6. System edits relationship of user in database
--------------	---

Use Case ID	3.4
Use Case	Set User Permissions
Actors	Administrator
Description	Administrator can set user permissions (ban/unban users)
Pre-Condition:	<ul style="list-style-type: none"> • Administrator is logged in • Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> • Administrator bans users • Administrator unbans users
Flow:	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel 3. Administrator will click on Manage Users button 4. Administrator will search for certain user 5. Administrator will click on Edit User Permissions button 6. System updates user permissions in database

Use Case ID	3.5
Use Case	Delete Users
Actors	Administrator
Description	Administrator can delete users from the database
Pre-Condition:	<ul style="list-style-type: none"> • Administrator is logged in • Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> • Administrator deletes user from database • Delete user from users table
Flow:	<ol style="list-style-type: none"> 1. Administrator will log in 2. Administrator will access Admin Panel

	<ol style="list-style-type: none"> Administrator will click on Manage Users button Administrator will search for certain user Administrator will click on Delete User button System deletes user from database
--	--

Use Case ID	4
Use Case	Manage Roles
Actors	Administrator
Description	Administrator can change the permissions that each role of admin, undergraduate student, graduate student, and staff member has
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Admin grants permissions based on user roles Updates roles in database
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Role Permissions button

Use Case ID	4.1
Use Case	View Roles
Actors	Administrator
Description	Administrator will be able to see the available roles
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will see the roles that are available to be assigned to users
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Roles button System shows list of roles in database

Use Case ID	4.2
Use Case	Create Roles
Actors	Administrator
Description	Administrator will be able to create new roles for users
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will create new types of roles to be assigned to users System adds role to table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Roles button Administrator will click on Create Role button System creates new role in database

Use Case ID	4.3
Use Case	Edit Roles
Actors	Administrator
Description	Administrator will be able to edit roles and their capabilities
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will edit previously created roles System updates roles table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Roles button Administrator will search for certain role Administrator will click on Edit Role button System edits selected role in database

Use Case ID	4.4
Use Case	Delete Roles
Actors	Administrator
Description	Administrator will be able to delete previously created roles
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will delete roles from the list of roles
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Roles button Administrator will search for certain role Administrator will click on Delete Role button System deletes role from database

Use Case ID	4.5
Use Case	Update Resource Access by Role
Actors	Administrator
Description	Administrator is able to restrict users based on their roles from Resources
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will restrict access to Resources based on the user's role System updates roles table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Roles button Administrator will search for certain role Administrator will click on Update Resource Access button System updates resource in database

Use Case ID	5
--------------------	---

Use Case	Manage Bookings
Actors	Administrator
Description	Administrator will be able to view and manage bookings made by users
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will manage bookings Add or update entry in user_desks table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Bookings button

Use Case ID	5.1
Use Case	View Bookings
Actors	Administrator
Description	Administrator will be able to view bookings made by users
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will view all the information on bookings made by users
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Bookings button System lists the bookings in database

Use Case ID	5.2
--------------------	-----

Use Case	Create Bookings
Actors	Administrator
Description	Administrator will be able to create bookings for users if the users have difficulty doing the task themselves
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will create a new booking Store entry in user_desks table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Bookings button Administrator will search for certain user Administrator will click on Create Booking button System adds booking to database

Use Case ID	5.3
Use Case	Edit Bookings
Actors	Administrator
Description	Administrator will be able to edit bookings in database
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will edit previously created bookings Update entry in user_desks table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Bookings button Administrator will search for certain booking Administrator will click on Edit Booking button System edits certain booking in database

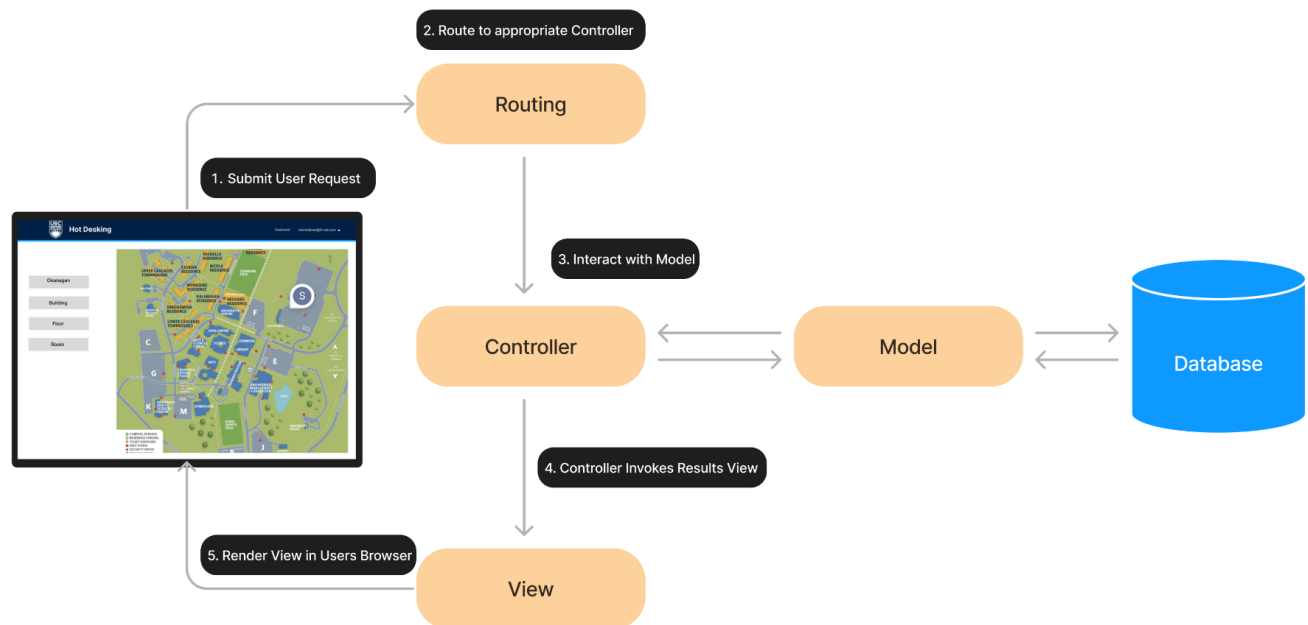
Use Case ID	5.4
Use Case	Delete Bookings
Actors	Administrator
Description	Administrator will be able to delete bookings in database
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in Administrator accesses Admin Panel
Post-Condition:	<ul style="list-style-type: none"> Administrator will delete previously created bookings Delete entry in user_desks table
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will access Admin Panel Administrator will click on Manage Bookings button Administrator will search for certain booking Administrator will click on Delete Booking button System deletes certain booking from database

Use Case ID	6
Use Case	Log Out
Actors	Administrator
Description	Administrator will be able to log out of system
Pre-Condition:	<ul style="list-style-type: none"> Administrator is logged in
Post-Condition:	<ul style="list-style-type: none"> Administrator logs out of system
Flow:	<ol style="list-style-type: none"> Administrator will log in Administrator will press Log Out button

3. System Architecture

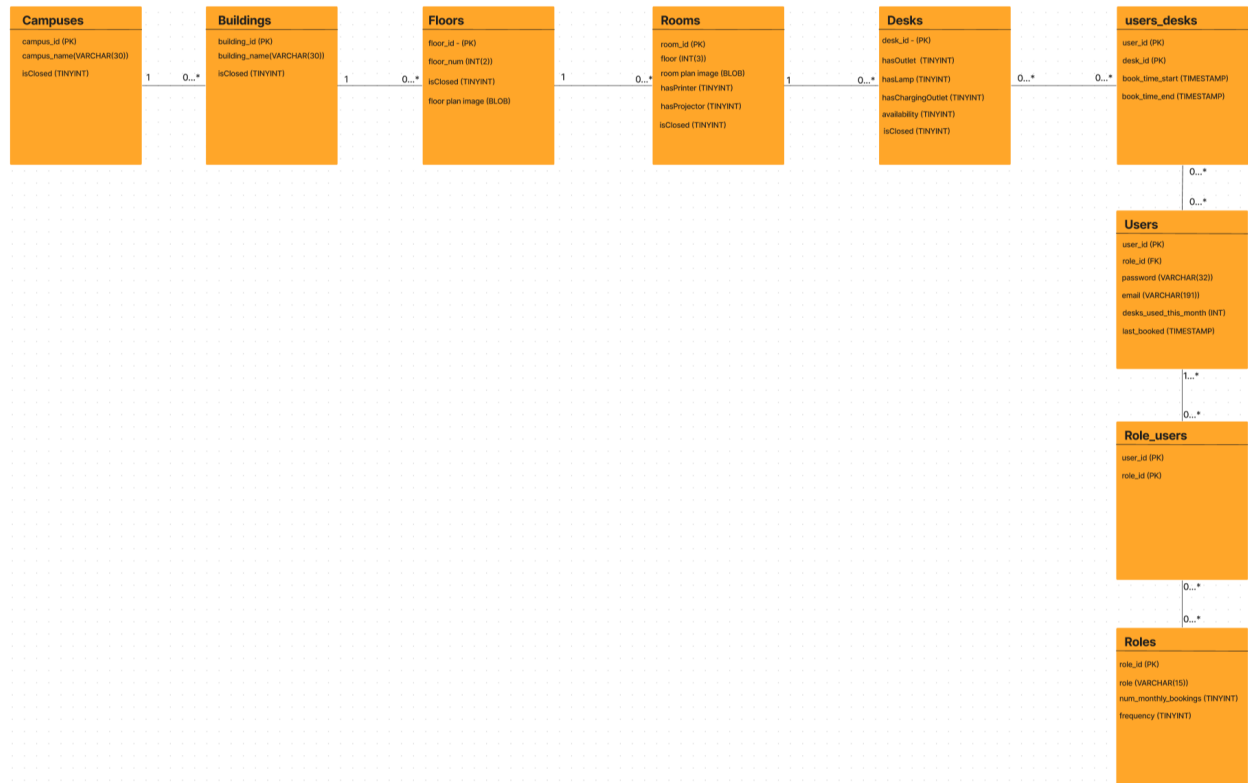
System Architecture Diagram

The system architecture diagram shows the various components of the Model-View-Controller architecture and how they communicate.



3.1 ER Diagram

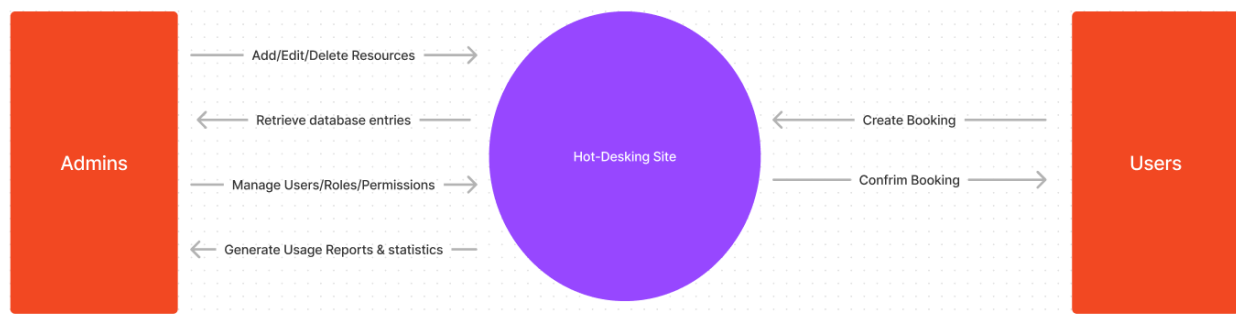
This diagram dictates how data is represented as objects and their relations in the relational schema



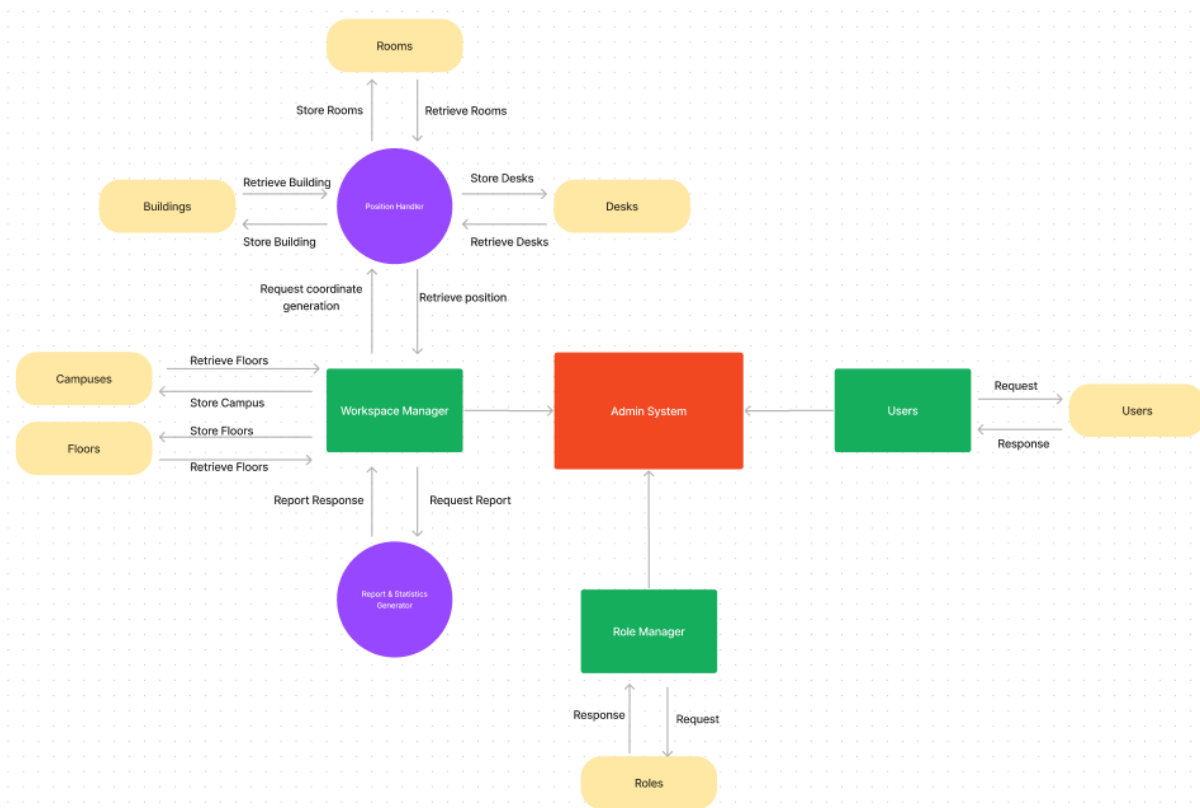
3.2 Data-Flow-Diagram

This diagram dictates how the users actions will interact with the various systems in the application

Level 0:

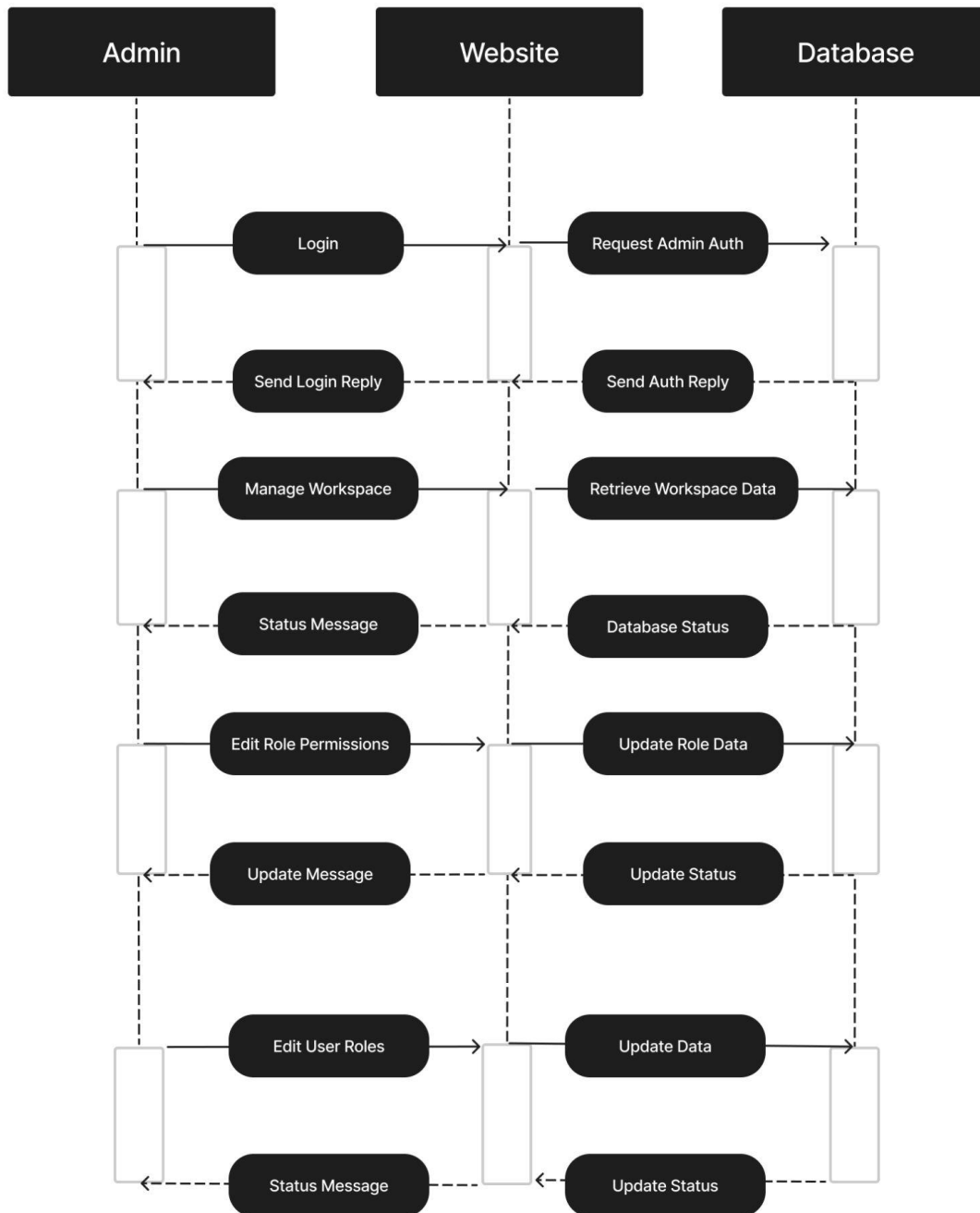


Level 1:



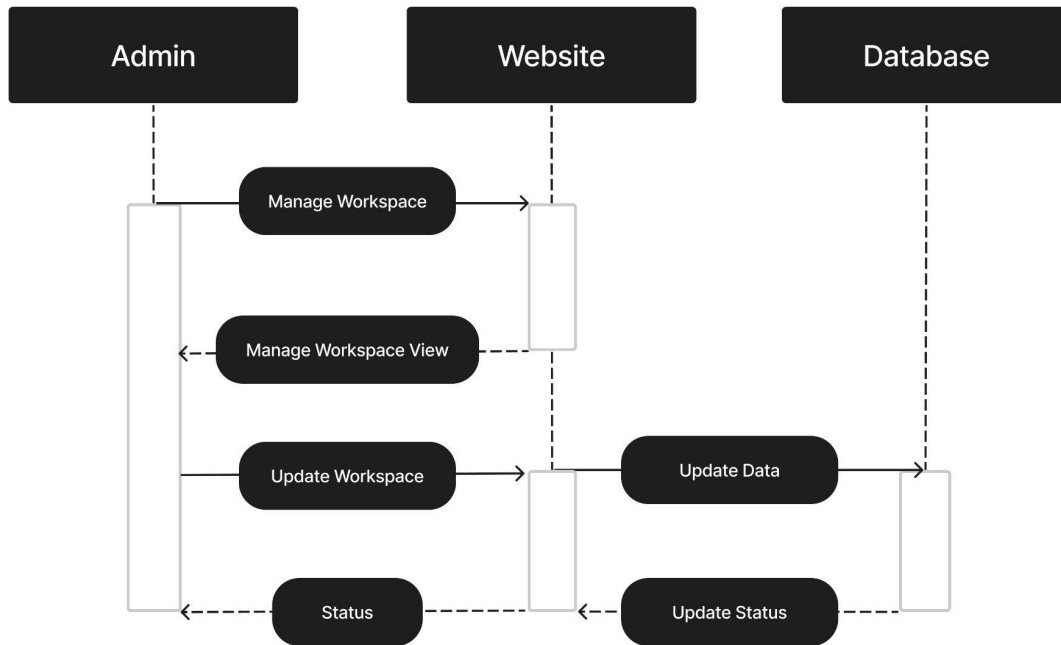
3.3 Sequence Diagram

This diagram shows how the admins interactions are relayed through the system and the timeline of processes that occur through different components.




3.3.1 Manage Workspace Sequence Diagram

This diagram expands on the Manage Workspace event timeline, showing the communication and processes occurring in more detail.



4. UI mockups

Login Page

**Hot Desking**[Login](#)[Register](#)

Login

E-Mail Address

Password

☐ Remember Me

Login

[Forgot Your Password?](#)



Main Page

- [Manage Users](#)
- [Manage Roles](#)
- [Manage Resources](#)
- [Manage Bookings](#)
- [Usage Reports](#)



Users

	Name	Role			
1	Bob	Graduate Student	Edit	Delete	View
2	Dylan	Undergrad Student	Edit	Delete	View
3	Elora	Faculty	Edit	Delete	View
4	James	Faculty	Edit	Delete	View
5	Jordyn	Undergrad Student	Edit	Delete	View
6	Elizabeth	Graduate Student	Edit	Delete	View
7	Patrick	Graduate Student	Edit	Delete	View
8	Judas	Faculty	Edit	Delete	View
9	Lucca	Faculty	Edit	Delete	View
10	Lidia	Graduate Student	Edit	Delete	View



Hot Desking

Dashboard

AdminEmail@Email.com ▲

Okanagan

Building

Floor

Room





Ground-Level Floor Plan

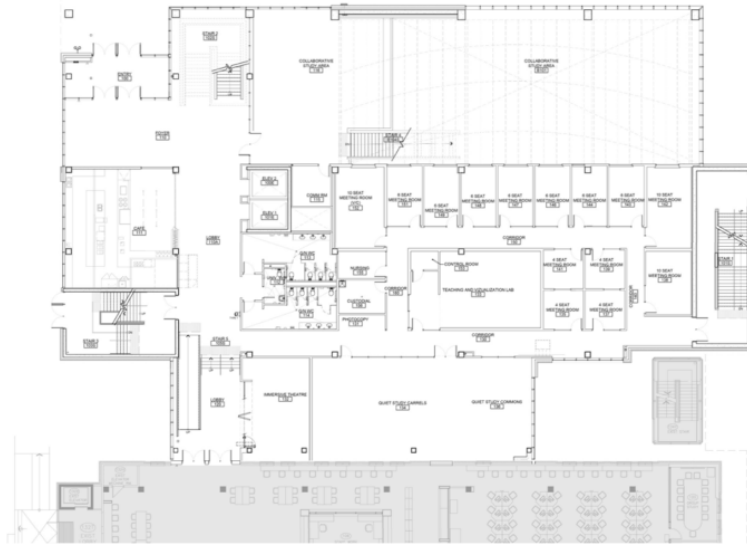
Okanagan

Commons

Floor 0

Room

100
121
122
136



Okanagan

Commons

Floor 0

136

Desk 1

Desk 2

Desk 3

Desk 4

5. Technical specification

- Laravel (Framework)
- Nginx
- PHP
- PHPUNIT(Framework Testing)
- PostgreSQL
- Bootstrap (Front-end Styling)
- Docker
- HTML
- CSS
- JS
- AJAX
- JQUERY

6. Test plan

6.1 Unit Testing

PHPUnit will be used for unit testing. This will be done throughout the development of the system. For example, unit testing on the database will be used to check that the database queries match the expected output.

6.2 Integration Testing

Integration testing will be done when each component is being integrated. For example, authentication will be tested with Laravel.

6.3 System Testing

System testing will be done after the system is built to ensure the system meets the required list of functional requirements and non-functional requirements. For example, the system's response time is tested with Selenium.

6.4 Acceptance Testing

Acceptance testing will be done frequently throughout the development of the system. For example, the client will do functional testing to make sure the system meets their expectations.

6.5 User Testing

Feedback will be collected after the system is built. For example, users will provide feedback on whether they can successfully complete tasks such as "add resources to a room".