

Business Requirements Document

October 5, 2022

Issued by:

Algorithmic Alchemist

Team Lead

Sierra Harris

Team Members

Bryant Lam

Abhay Solanki

Faisal Al Muharrami

Tania Adame

David Chan

Github: https://github.com/abhay772/AA_Senior_Project/

Version History

Version #	Date	Reason for Change
Version 1	10/5/2022	Original Document
Version 2	11/19/2022	Fixing according to feedback after submission
Version 3	12/6/2022	Separate the classes
Version 4	12/7/2022	Add core components and requirements

Table of Contents

Version History	2
Overview	4
Business Goal	4
Project Overview	5
Project Constraints	5
Stakeholders Identification	6
Business Requirements	6
Use Cases	7
References:	59

Overview

The business requirements document will go over functional and non-functional requirements, and how the system will behave in case of failure. The business problem this product will solve and the goals we are trying to attain with the product as well as business rules and criteria of success that must be followed while keeping in mind the risks and constraints that this project may entail. All this information will be listed on the following pages.

Business Problem

Most property management software provides property owners, managers, and landlords the foundation to keep track of their finances and business needs. These applications are tailored toward a property's residential bookkeeping and tenant management. However, what these applications lack, and overlook is arguably the most important factor of property management, the physical property.

Property management software mainly focuses on the accounting aspect of a business rather than the property itself. Property managers and owners lack a foundation to effectively manage and improve their property appearance and value. Software for property management today focuses on residential businesses, bookkeeping, and connecting maintenance companies with property managers. For instance, the top property manager application, Buildium allows its users to keep track of rental payments, vacancies, and accounting (*Buildium*). However, you need to attract and retain tenants first and foremost. For instance, Hales Property Managements published an article titled “8 ways to make your rental property more appealing to prospective tenants” in 2021 with each listing pertaining to the property’s appearance and upkeep. On the other hand, Angie’s, an application that connects property managers to pros in their area, allows consumers to keep track of their open projects and compare quotes(*How does Angi Work*). However, Property Manager to Go will allow for the common property owner and property managers to keep track of their properties, maintenance services, and renovation projects, manage and assist in the scheduling of services, and provide quick estimates for property value.

Business Goal

Our goal is to provide property management software that will efficiently and thoroughly assist its users, with the elevation of property value as its objective, an overlooked outlook within property management. The value of our product is in providing a foundation that prioritizes increasing property value by organizing all property services management into one platform.

The software will focus on digitizing and organizing services that property managers use frequently for property upkeep, in addition to organizing business transactions and needs. While providing insight on how the property value changes before and after all transactions and assisting in the scheduling, budgeting, and planning of value-increasing services. For instance, our software will help with the planning of future services by providing a change in property value for mere service inquiries and after-service completion. Our software will also

accommodate each user by recommending service schedule dates and comparing estimates for budgeting. Lastly, our software will provide a feature to give services a rating based on performance, which will tell maintenance and property services companies what they can improve on and what they are doing well, improving their overall quality of work.

The appearance and value of a property are equally as important for reasons such as tenant attraction and retention, in addition to simply wanting a well-maintained home. What the industry lacks is a one-stop-shop application for property owner's to efficiently and thoroughly manage their property while overseeing their change in property value. Thus, Algorithmic Alchemist proposes Property Manager Go, a web application to meet these needs and provide a foundation for all property services management one room at a time.

Project Overview

Algorithmic Alchemist aims to deliver an application that allows property managers to create profiles to manage their services, plan maintenance and renovation projects, evaluate their property, and schedule appointments with service providers. We also aim to provide a foundation for service providers to promote their companies or organizations and acquire more customers by including their profiles in a service search list and presenting their service ratings. Companies will also receive assistance in managing their service request from property managers with the request manager. The application will allow all users access to their user profiles, a dashboard, a calendar, user management, and document storage.

Project Constraints

- At the initial deployment, the app will only cover Contractors, Service Providers, and Properties in **California**.
- At the initial deployment, the app will only support **US-English**.
- At the initial stage, the product will only be accessible from the **web**.
- **Payment Processing** is not going to be supported.

Stakeholders Identification

The stakeholders include those that will benefit from or will be affected by the application. Stakeholders include property owners, landlords, property managers, maintenance companies, and service providers.

Users:

- Property Managers are users that own or manage a property, such as a home or a commercial property such as an office building or retail store.
- Service providers are users that provide property services such as
 - Landscaping, street sweeping, window washing, janitorial, or plumbing.
 - Renovation for upgrades to a bedroom, bathroom, or kitchen like tiles or flooring, appliances, and fixtures.
 - And the remodeling of rooms.

Business Requirements

Property Managers require a single platform to manage their properties. Algorithmic Alchemist brings forward an alternative application called Property Manager to Go or PMtoGo to satisfy and effectively accommodate property owners and managers while promoting service providers and their services.

Algorithmic Alchemist proposes the PMtoGo application to..

- Increase revenue and outreach for service providers
- Improve customer service and quality of work for service providers
- Simplify property management and assist with planning and budgeting
- Increase property value

Use Cases

Features for all Users:

Use Case ID:	1
Use Case Name:	Account Creation
Actors:	- All Users
Business Rule:	- User's email is assigned as their unique username
Description:	- Non logged in users are able to create a new account with PMTOGO.
Preconditions:	<ul style="list-style-type: none">- User must not have an active authenticated session- User must be on account creation view
Functional Requirements:	<ul style="list-style-type: none">- The user should be able to provide a valid email and password to register an account.- The user should be able to provide a full name, date of birth, and address, which is stored in their profile.- All user accounts must be stored in a persistent data store.- The user is assigned their email as the system-wide unique username.- The user provides a password for requesting OTP, and upon entering a valid OTP they can successfully register their account.
Postconditions:	<ul style="list-style-type: none">- The user is redirected to log in.
Nonfunctional Requirements:	<ul style="list-style-type: none">- System administrators cannot be created using the Account Creation feature.- System failures from this feature must not result in the system going offline- Password must be a minimum of 8 characters- Password can be a maximum of 16 characters- Valid characters will consist of the following:<ul style="list-style-type: none">a. blank spaceb. a-zc. A-Zd. 0-9e. ., @!-- The password must be censored with a * for every character..- Users must be notified within 5 seconds if they entered invalid credentials.- The user can choose to "show password" when signing up.

	<ul style="list-style-type: none"> - User is redirected to the homepage in under 5 seconds.
Success cases:	<ul style="list-style-type: none"> - User registers with a valid email and valid password. The system is able to assign a system-wide unique username. A system message displays "Account created successfully" within 5 seconds of invoking the registration process. The system provides the username to the user.
Failure cases:	<ul style="list-style-type: none"> - User registers with an invalid email. A system message displays "Invalid email provided. Retry again or contact system administrator" or no system message. The account is not created. - User registers with an invalid password. A system message displays "Invalid password provided. Retry again or contact system administrator" or no system message, Account is not created. - User registers with a valid email and valid password. The system was unable to assign a system-wide username. A system message displays "Unable to assign username. Retry again or contact system administrator". The account is not created. - User registers with a valid email and valid password. The system was able to assign a system-wide username. The entire process took longer than 5 seconds. A system log entry is recorded. The account is created. - The user is able to provide a valid password and email however is not automatically redirected to the homepage. The operation takes longer than 6 seconds. - The user provides an invalid password. The system prompts the message, "Invalid Password". - The system does not censor user password input with a "*" .

Use Case ID:	2
Use Case Name:	Login
Actors:	<ul style="list-style-type: none"> - All Users
Business Rule:	<ul style="list-style-type: none"> - Users must not be already logged in.
Description:	<ul style="list-style-type: none"> - Allowing users to login, who do not already have an authenticated session.
Preconditions:	<ul style="list-style-type: none"> - Users must not already have an active authenticated session

	<p>with the system on the current device, otherwise authentication is not possible.</p> <ul style="list-style-type: none"> - Users must be on login view or attempting to access a protected resource as defined in Authorization.
Functional Requirements:	<ul style="list-style-type: none"> - The user should be able to provide a valid email and password to login into an account. - The user is provided an OTP, upon entering a valid OTP they can successfully login to their account. - Upon authentication the user should be directed to the home page.
Postconditions:	<ul style="list-style-type: none"> - The user is redirected to the home page, with an authenticated session.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - System failures from this feature must not result in the system going offline - The password must be censored with a * for every character.. - Users must be notified within 5 seconds if they entered invalid credentials. - The user can choose to “show password” when signing up. - User is redirected to the homepage in under 5 seconds.
Success cases:	<ul style="list-style-type: none"> - User enters a valid email and valid password. Receiving an OTP, and upon entering a valid OTP, is directed to the home page with an authenticated session. - If the user is already authenticated, the user should not be able to reach the login view.
Failure cases:	<ul style="list-style-type: none"> - User enters valid security credentials, but is not directed to the home view. - User enters valid security credentials, but is directed to a view other than the home view. - User enters an invalid email or password. A system message displays “Invalid email or password provided. Retry again or contact system administrator” or no system message. The user is not authenticated. - User registers with a valid email and valid password. But does not enter a valid OTP. “Invalid username or password provided. Retry again or contact system administrator if issue persists”. The user is not authenticated. - Users enter valid security credentials. The entire process took longer than 5 seconds. A system log entry is recorded. - The system does not censor user password input with a “*”. - User submits valid security credentials for a disabled account. A system message displays “Account disabled. Perform account recovery or contact system admin”. The failure attempt is recorded accurately. - User submits valid security credentials for a disabled account.

	A system message displays “Account disabled. Perform account recovery or contact system admin”. The failure attempt is not recorded accurately. The system attempts to log that the failure attempt did not complete successfully.
--	--

Use Case ID:	3
Use Case Name:	Login - Protected Resources
Actors:	- All Users
Business Rule:	- The user must not be already logged in.
Description:	- User must be on login view or attempting to access a protected resource as defined in Authorization
Preconditions:	<ul style="list-style-type: none"> - The user must not already have an active authenticated session with the system on the current device, otherwise authentication is not possible. - The user must be on login view or attempting to access a protected resource as defined in Authorization.
Functional Requirements:	<ul style="list-style-type: none"> - The user should be able to provide a valid email and password. - The user is provided an OTP, upon entering a valid OTP they can successfully login to their account. - Upon authentication the user should be directed to the resource view they were trying to access.
Postconditions:	- The user is redirected to the resource view, with an authenticated session.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - System failures from this feature must not result in the system going offline - The password must be censored with a * for every character.. - Users must be notified within 5 seconds if they entered invalid credentials. - The user can choose to “show password” when signing up. - User is redirected to the resource view in under 5 seconds.
Success cases:	<ul style="list-style-type: none"> - User enters a valid email and valid password. Receiving an OTP, and upon entering a valid OTP, is directed to previously restricted resource view with an authenticated session. - If the user is already authenticated, the user should not be able to reach the login view.

Failure cases:	<ul style="list-style-type: none"> - User enters valid security credentials, but is not directed to the resource view. - User enters valid security credentials, but is directed to a view other than resource view. - User enters an invalid email or password. A system message displays "Invalid email or password provided. Retry again or contact system administrator" or no system message. The user is not authenticated. - User registers with a valid email and valid password. But does not enter a valid OTP. "Invalid username or password provided. Retry again or contact system administrator if issue persists". The user is not authenticated. - The user enters valid security credentials. The entire process took longer than 5 seconds. A system log entry is recorded. - The system does not censor user password input with a "*". - User submits valid security credentials for a disabled account. A system message displays "Account disabled. Perform account recovery or contact system admin". The failure attempt is recorded accurately. - User submits valid security credentials for a disabled account. A system message displays "Account disabled. Perform account recovery or contact system admin". The failure attempt is not recorded accurately. The system attempts to log that the failure attempt did not complete successfully.
----------------	---

Use Case ID:	4
Use Case Name:	Authorization - Use Functionality
Actors:	<ul style="list-style-type: none"> - All Users
Business Rule:	<ul style="list-style-type: none"> - The user must not be already logged in. - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity (i.e., anonymous user)
Description:	<ul style="list-style-type: none"> - Allowing users to access functionality based on their role.
Preconditions:	<ul style="list-style-type: none"> - User must be authenticated to enforce user-specific restrictions - User account must be active
Functional Requirements:	<ul style="list-style-type: none"> - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of

	user's identity (i.e., anonymous user) <ul style="list-style-type: none"> - The operation and timestamp of each unauthorized access will be recorded by the system - The system must prevent unauthorized users from executing any protected functionality -
Postconditions:	<ul style="list-style-type: none"> - The user is able to see the resource if they have the correct role.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - There should be 4 types of roles in the system: <ol style="list-style-type: none"> Anonymous User (AnonUser) Property Manager (PMUser) Service Provider (SPUser) Admin - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User attempts to access a protected functionality within authorization scope. Access is granted to perform functionality.
Failure cases:	<ul style="list-style-type: none"> - Unauthorized access is not recorded by system when authorization fails. A system log of failure is attempted. - User attempts to access a protected functionality outside of authorization scope. Access is denied and a system message displays "Unauthorized access".

Use Case ID:	5
Use Case Name:	Authorization - Read Data
Actors:	<ul style="list-style-type: none"> - All Users
Business Rule:	<ul style="list-style-type: none"> - The user must not be already logged in. - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity (i.e., anonymous user)
Description:	<ul style="list-style-type: none"> - Allowing users to read data based on their role.
Preconditions:	<ul style="list-style-type: none"> - User must be authenticated to enforce user-specific restrictions - User account must be active
Functional Requirements:	<ul style="list-style-type: none"> - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of

	user's identity (i.e., anonymous user) <ul style="list-style-type: none"> - The operation and timestamp of each unauthorized access will be recorded by the system - The system must prevent unauthorized users from reading any protected data - Any user access modifications should be active upon the next successful authentication by user
Postconditions:	<ul style="list-style-type: none"> - The user is able to read protected data if they have the correct role.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - There should be 4 types of roles in the system: <ol style="list-style-type: none"> Anonymous User (AnonUser) Property Manager (PMUser) Service Provider (SPUser) Admin - System failures from this feature must not result in the system going offline
Success cases:	<ul style="list-style-type: none"> - User attempts to access protected data within authorization scope. Access is granted to perform read operations.
Failure cases:	<ul style="list-style-type: none"> - Unauthorized access is not recorded by system when authorization fails. A system log of failure is attempted. - User attempts to access protected data outside of authorization scope. Access is denied and a system message displays "Unauthorized access to data".

Use Case ID:	6
Use Case Name:	Authorization - Modify Data
Actors:	<ul style="list-style-type: none"> - All Users
Business Rule:	<ul style="list-style-type: none"> - The user must not be already logged in. - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity (i.e., anonymous user)
Description:	<ul style="list-style-type: none"> - Allowing users to modify data based on their role.
Preconditions:	<ul style="list-style-type: none"> - User must be authenticated to enforce user-specific restrictions - User account must be active
Functional Requirements:	<ul style="list-style-type: none"> - By default, unauthenticated users will only be given access to

	<p>resources or functionalities that does not require knowledge of user's identity (i.e., anonymous user)</p> <ul style="list-style-type: none"> - The operation and timestamp of each unauthorized access will be recorded by the system - The system must prevent unauthorized users from modify any protected data - Any user access modifications should be active upon the next successful authentication by user -
Postconditions:	<ul style="list-style-type: none"> - The user is able to modify protected data if they have the correct role.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - There should be 4 types of roles in the system: <ul style="list-style-type: none"> a. Anonymous User (AnonUser) b. Property Manager (PMUser) c. Service Provider (SPUser) d. Admin - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User attempts to modify protected data within authorization scope. Access is granted to perform write operations.
Failure cases:	<ul style="list-style-type: none"> - Unauthorized access is not recorded by system when authorization fails. A system log of failure is attempted. - User attempts to modify protected data outside of authorization scope. Access is denied and a system message displays "Unauthorized access to data".

Use Case ID:	7
Use Case Name:	Authorization - Protected Views
Actors:	<ul style="list-style-type: none"> - All Users
Business Rule:	<ul style="list-style-type: none"> - The user must not be already logged in. - By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity (i.e., anonymous user)
Description:	<ul style="list-style-type: none"> - Allowing users to access protected views based on their role.
Preconditions:	<ul style="list-style-type: none"> - User must be authenticated to enforce user-specific restrictions - User account must be active

Functional Requirements:	<ul style="list-style-type: none"> - The operation and timestamp of each unauthorized access will be recorded by the system - User attempts to access protected views within authorization scope. Access is granted to the view. User is automatically navigated to view. - The system must prevent unauthorized users from viewing or interacting with any protected views.
Postconditions:	<ul style="list-style-type: none"> - The user is able to access protected views if they have the correct role.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - There should be 4 types of roles in the system: <ul style="list-style-type: none"> a. Anonymous User (AnonUser) b. Property Manager (PMUser) c. Service Provider (SPUser) d. Admin - System failures from this feature must not result in the system going offline
Success cases:	<ul style="list-style-type: none"> - User attempts to access protected views within authorization scope. Access is granted to the view. User is automatically navigated to view.
Failure cases:	<ul style="list-style-type: none"> - Unauthorized access is not recorded by system when authorization fails. A system log of failure is attempted. - User attempts to access protected views outside of authorization scope. Access is denied and a system message displays "Unauthorized access to view". - User attempts to access protected views within authorization scope, but contains protected data that is not within read scope. Access is granted to the view. Upon completion of automatic navigation to view, a system message displays "Unauthorized access to data" with protected data not visible within the view. - User attempts to access protected views within authorization scope, but contains protected data that is not within write scope. Access is granted to the view. Upon completion of automatic navigation to view, protected data is visible within the view. Attempts to modify the data will result in a system message that displays "Unauthorized access to data"

Use Case ID:	8
Use Case Name:	Logout
Actors:	<ul style="list-style-type: none"> - All authenticated users

Business Rule:	- Users must be already logged in.
Description:	- Allowing users to exit their authenticated session.
Preconditions:	<ul style="list-style-type: none"> - User must be in an authenticated session - User must be on a view with the logout option
Functional Requirements:	<ul style="list-style-type: none"> - The user will be navigated to the home view of the system upon successful completion -
Postconditions:	- User is able exit their authenticated session, and redirected to the home view
Nonfunctional Requirements:	<ul style="list-style-type: none"> - The current active session on the device will end within 5 seconds upon invocation - System failures from this feature must not result in the system going offline
Success cases:	<ul style="list-style-type: none"> - User performs a logout request. The active session ends. The user is automatically navigated to the default home view of the system with the default culture settings. A system message displays “Logout successfully” upon completion of automatic navigation to home view. The logout process is completed within 5 seconds upon invocation. - If the user is not authenticated, then the system should not allow log out.
Failure cases:	<ul style="list-style-type: none"> - User performs a logout request. The active session has ended. The user is not automatically navigated to the default home view. A system message displays “Logout operation error” or no message is shown - User performs a logout request. The active session has ended. The user is automatically navigated to the default home view, but not set to the default culture settings. A system message displays “Logout operation error” or no message is shown - The logout process takes longer than 5 seconds.

Use Case ID:	9
Use Case Name:	Account Deletion
Actors:	- All registered users
Business Rule:	- Only a system administrator account can delete another

	system administrator account.
Description:	<ul style="list-style-type: none"> - Users are able to delete their profile
Preconditions:	<ul style="list-style-type: none"> - User has an authenticated session
Functional Requirements:	<ul style="list-style-type: none"> - A user should be able to enter their password, and delete all PII related to this account. - Upon successful deletion a status message is displayed, "Account deletion successful". Upon acknowledgment of the system message, the user is directed to the home page.
Postconditions:	<ul style="list-style-type: none"> - The user is logged out and cannot login with the deleted email.
Nonfunctional Requirements:	<ul style="list-style-type: none"> - Account deletion is irreversible. - System failures from this feature must not result in the system going offline. - Users must be notified within 6 seconds if they entered invalid credentials. - User is redirected to the homepage in under 5 seconds. - After 3 attempts to log in the account will be locked and only the option of "Forgot password" will be available.
Success cases:	<ul style="list-style-type: none"> - User chooses to delete the account and confirms action. All PII data and user account data are permanently deleted from the system. A system message displays "Account deletion successful". Upon acknowledgment of the system message, the user is automatically navigated to the home view with default language and culture settings.
Failure cases:	<ul style="list-style-type: none"> - User does not enter a valid password. - User enters a valid password, but the account is not deleted. - User enters a valid password, but the profile is not deleted. - User enters a valid password, but all of PII data is not deleted. - The user is able to recover deleted accounts. - A system message is not shown or the wrong message is shown after all PII data and user account data is permanently deleted from the system - The user is unable to acknowledge the system message "Account deletion successful" after the successful data deletion. - The user is not automatically redirected to the default home view of the system. - The user is automatically redirected to the default home view, but the default language and culture setting is not shown.

Use Case ID	10
Use Case Name	Account Recovery Request
Actor	- All registered users
Business Rules	- Users must be able to request an account recovery when their account when its disabled
Description	- When an account is disabled a user would need to use the account recovery function to gain access to it. The user would need to enter a new password.
Preconditions	<ul style="list-style-type: none"> - User must not have an active authenticated session on the device, otherwise the user is unable to perform the operation - User must be on account recovery view
Functional	<ul style="list-style-type: none"> - The user must provide valid email and valid OTP to submit account recovery request - Upon successful account recovery by authorized system admin, the user will be able to authenticate into the system.
Postconditions	- The user is able to set a new password for their account.
Nonfunctional	<ul style="list-style-type: none"> - Users must always be presented with the “Forget password” during the whole login process(before they login). - The user can choose to “show password” when changing password. - Recovery email is sent to the user's email address within 5 seconds. - Request is made available to authorized system admin users within 5 seconds. - A system message displays “Account recovery request sent” within 5 seconds of invocation of request. - A system message displays “Account recovery completed successfully for the user” within 5 seconds of invocation. - Affected user regains access to the system within 5 seconds of invocation. - System failures from this feature must not result in the system going offline
Success Cases	- User provides assigned username and valid OTP. Request is made available to authorized system admin users within 5 seconds. A system message displays “Account recovery request sent” within 5 seconds of invocation of request.
Failure Cases	<ul style="list-style-type: none"> - User provides invalid username. A system message displays “Invalid username or OTP provided. Retry again or contact system administrator” - User provides valid username, but invalid OTP. A system

	<p>message displays “Invalid username or OTP provided. Retry again or contact system administrator”</p> <ul style="list-style-type: none"> - User provides valid username and valid OTP. Request is not available to authorized system admin users. - User provides valid username and valid OTP. Request is available to authorized system admin users. System message does not display within 5 seconds on invocation.
--	--

Use Case ID	11
Use Case Name	Account Recovery Request Resolution
Actor	- Admins
Business Rules	- Admins should be able to accept/deny account recovery requests
Description	- When an account is disabled a user would need to use the account recovery function to gain access to it. The user would need to enter a new password.
Preconditions	<ul style="list-style-type: none"> - User must have an active authenticated session on the device as an admin, otherwise the user is unable to perform the operation - User must be on account recovery requests view
Functional	<ul style="list-style-type: none"> - Admins can view all account recovery requests - Admins can accept/deny account recovery requests - Admins receive notifications for a new request.
Postconditions	- Admins are able to view and resolve account recovery requests
Nonfunctional	<ul style="list-style-type: none"> - Request is made available to authorized system admin users within 5 seconds. - A system message displays “Account recovery request received” within 5 seconds of invocation of request. - A system message displays “Account recovery completed successfully for the user” within 5 seconds of invocation. - A request should disappear in 5 seconds after resolution. - System failures from this feature must not result in the system going offline
Success Cases	- An authorized system admin completes account recovery for users. A system message displays “Account recovery completed successfully for the user” within 5 seconds of invocation. Affected user regains access to the system within 5

	seconds of invocation.
Failure Cases	<ul style="list-style-type: none"> - User provides valid username and valid OTP. Request is not available to authorized system admin users. - User provides valid username and valid OTP. Request is available to authorized system admin users. System message does not display within 5 seconds on invocation. - An authorized system admin completes account recovery for users. System message does not display within 5 seconds on invocation. - An authorized system admin completes account recovery for users. System message does display within 5 seconds on invocation. Affected user does not regain access. - An authorized system admin completes account recovery for users. System message does display within 5 seconds on invocation. Affected user does not regain access within 5 seconds.

Use Case ID:	12
Use Case Name:	User Management - Create Account
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system
Description:	<ul style="list-style-type: none"> - Allows administrators to create an account.
Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session - User must be on user management view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - A system administrator is able to create an user account - All operations are applied to a persistent data store

Postconditions:	<ul style="list-style-type: none"> - User account created by a system administrator is saved to a persistent data storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - A system message displays “Account creation was successful”, upon successful modification of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User is able to create an account within 5 seconds upon invocation. A system message displays “Account creation was successful”
Failure cases:	<ul style="list-style-type: none"> - Creation of the account takes longer than 5 seconds - Creation of the account completes within 5 seconds, but no system message is shown or inaccurate system message is shown - Creation of the account completes within 5 seconds, with system message “Creation of the account was successful” shown, but latest data is not written to data store

Use Case ID:	13
Use Case Name:	User Management - Update Account
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system
Description:	<ul style="list-style-type: none"> - Allows administrators to update an account.
Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session - User must be on user management view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - A system administrator is able to update an user account

	<ul style="list-style-type: none"> - All operations are applied to a persistent data store
Postconditions:	<ul style="list-style-type: none"> - Modifications to a user account by a system administrator is saved to a persistent data storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - A system message displays “Account update was successful”, upon successful modification of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User is able to modify an account within 5 seconds upon invocation. A system message displays “Account update was successful”
Failure cases:	<ul style="list-style-type: none"> - Modification of an account takes longer than 5 seconds - Modification of an account completes within 5 seconds, but no system message is shown or inaccurate system message is shown - Modification of an account completes within 5 seconds, with system message “Modification of the account was successful” shown, but latest data is not written to data store

Use Case ID:	14
Use Case Name:	User Management - Delete Account
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system
Description:	<ul style="list-style-type: none"> - Allows administrators to delete an account.
Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session - User must be on user management view - User must be a system administrator

Functional Requirements:	<ul style="list-style-type: none"> - A system administrator is able to delete an user account - All operations are applied to a persistent data store
Postconditions:	<ul style="list-style-type: none"> - Deletion of a user account by a system administrator is saved to a persistent data storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - A system message displays "Account deletion was successful", upon successful deletion of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User is able to delete an account within 5 seconds upon invocation. A system message displays "Account deletion was successful"
Failure cases:	<ul style="list-style-type: none"> - Deletion of an account takes longer than 5 seconds - Deletion of an account completes within 5 seconds, but no system message is shown or inaccurate system message is shown - Deletion of an account completes within 5 seconds, with system message "Deletion of the account was successful" shown, but latest data is not written to data store

Use Case ID:	15
Use Case Name:	User Management - Disable Account
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system - Any account disabled for more than 2 years will be deleted
Description:	<ul style="list-style-type: none"> - Allows administrators to disable an account.
Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session

	<ul style="list-style-type: none"> - User must be on user management view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - A system administrator is able to disable an user account - All operations are applied to a persistent data store
Postconditions:	<ul style="list-style-type: none"> - Disabling of a user account by a system administrator is saved to a persistent data storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - A system message displays "Account disabling was successful", upon successful disabling of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User is able to disable an account within 5 seconds upon invocation. A system message displays "Account disabling was successful"
Failure cases:	<ul style="list-style-type: none"> - Disabling of an account takes longer than 5 seconds - Disabling of an account completes within 5 seconds, but no system message is shown or inaccurate system message is shown - Disabling of an account completes within 5 seconds, with system message "Disabling of the account was successful" shown, but latest data is not written to data store

Use Case ID:	16
Use Case Name:	User Management - Enable Account
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system
Description:	<ul style="list-style-type: none"> - Allows administrators to enable an account.

Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session - User must be on user management view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - A system administrator is able to enable an user account - All operations are applied to a persistent data store
Postconditions:	<ul style="list-style-type: none"> - Enabling of a user account by a system administrator is saved to a persistent data storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - A system message displays “Account enabling was successful”, upon successful enabling of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - User is able to enable an account within 5 seconds upon invocation. A system message displays “Account enabling was successful”
Failure cases:	<ul style="list-style-type: none"> - Enabling of an account takes longer than 5 seconds - Enabling of an account completes within 5 seconds, but no system message is shown or inaccurate system message is shown - Enabling of an account completes within 5 seconds, with system message “Enabling of the account was successful” shown, but latest data is not written to data store

Use Case ID:	17
Use Case Name:	User Management - Bulk Operations
Actors:	<ul style="list-style-type: none"> - All admins
Business Rule:	<ul style="list-style-type: none"> - Only system administrator have access to the User Management view - The system must have at least one system administrator account with total system access at all times - Only a system administrator account can create other system administrator accounts - All single and bulk operations must be able to affect any user account/profile attribute within the system - The system administrator will have access to view and modify all accounts and their associated user profile data within the system - Maximum of 10K user management operations per request are

	allowed
Description:	<ul style="list-style-type: none"> - Allows administrators to perform 10k or less user management operations
Preconditions:	<ul style="list-style-type: none"> - User must have an active authenticated session - User must be on user management view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - User should be able to execute multiple user management operations (e.g. all the same or mixed) within the same request - Requests can be made through an uploaded file extract - User should be able to create a system administrator account
Postconditions:	<ul style="list-style-type: none"> - All the bulk operations take place and result are stored in a persistent storage
Nonfunctional Requirements:	<ul style="list-style-type: none"> - File extract cannot be greater than 2GB in size - Operation should be completed within 60 seconds - A system message displays “Bulk user management operation was successful”, upon successful enabling of the account. - System failures from this feature must not result in the system going offline.
Success cases:	<ul style="list-style-type: none"> - Users are able to perform 10K or less user management operations in bulk within 60 seconds. A system message displays “Bulk user management operation was successful”
Failure cases:	<ul style="list-style-type: none"> - Bulk user management operations takes longer than 60 seconds - Bulk user management operations completes within 60 seconds, but no system message is shown or inaccurate system message is shown - Bulk user management operations completes within 60 seconds, with system message “Bulk user management operation was successful” shown, but latest data is not written to data store - 10K Bulk user management operations completes takes longer than 60 seconds - 10K Bulk user management operations completes within 60s seconds, but no system message is shown or inaccurate system message is shown - 10K Bulk user management operations completes within 60 seconds, with system message “Bulk user management operation was successful” shown, but latest data is not written to data store

Use Case ID:	18
Use Case Name:	Usage Analysis Dashboard
Actors:	- All admins
Business Rule:	<ul style="list-style-type: none"> - Key Performance Indicators (KPIs) <ul style="list-style-type: none"> o The number of logins per day within the span of 3 months (trend chart) o The number of registrations per day within the span of 3 month (trend chart) o The number of scheduled services per day within the span of 3 months (trend chart) o The number of completed services per day within the span of 3 months (trend chart) o The number of DIY views per day within the span of 3 months (trend chart)
Description:	- A dashboard to visualize user activity within the system
Preconditions:	<ul style="list-style-type: none"> - Persistent data store must be active - Persistent data store must accessible by the system - User must have an active authenticated session on the device - User must be on Usage Analysis Dashboard view - User must be a system administrator
Functional Requirements:	<ul style="list-style-type: none"> - All data must be fetched from a persistent data store - User should be able to view all the Key Performance Indicators (KPIs)
Nonfunctional Requirements:	<ul style="list-style-type: none"> - All KPI data must be automatically refreshed in intervals of 60 seconds - The view must load within 15 seconds upon completion of navigation. - System failures from this feature must not result in the system going offline
Success cases:	- User is able to navigate to the view. The view loads within 15 seconds. All KPIs automatically refreshes data within 60 seconds.
Failure cases:	<ul style="list-style-type: none"> - User is unable to navigate to the page, but is a system administrator - User is able to navigate to the page, but the view does not load within 15 seconds upon navigation completion. - The user is able to navigate to the page, view loads within 15 seconds, but no KPI data is refreshed. - The user is able to navigate to the page, view loads within 15

	seconds, but not all KPI data is refreshed. - The user is able to navigate to the page, view loads within 15 seconds, but all KPI refresh takes longer than 60 seconds.
--	--

Use Case ID:	19
Use Case Name:	Logging - System Success
Actors:	- The system
Business Rules:	<ul style="list-style-type: none"> - The log created must be immutable. - The log entry must be stored into the persistent datastores. - The log entry must contain a UTC timestamp, log level, user performing operation, a category, and a description or message. <ul style="list-style-type: none"> - Valid Levels are Info, Debug, Warning, Error - Valid Categories are View, Business, Server, Data, Datastore - While logging it must not block other processes.
Description:	<ul style="list-style-type: none"> - The system will log activity not done by users -
Preconditions:	<ul style="list-style-type: none"> - There must be a persistent data store active to log. - The persistent data store must be accessible by the system. - The persistent data store must have storage capacity to add a log entry
Functional	-
Postconditions:	- The system logs with the correct information and saves it to the data store.
Nonfunctional	-
Success Cases:	- The system logs system success events
Failure Cases:	<ul style="list-style-type: none"> - Logging process took over 5 seconds to complete upon invocation. - The user is blocked from interacting with the system while logging. - The logging process completes under 5 seconds, but was not able to save to a persistent data store. - The logging process completes under 5 seconds, but did not correctly save the right information to the data store.

	- Old logs are mutable and can be changed.
--	--

Use Case ID:	20
Use Case Name:	Logging - System Failures
Actors:	- The system
Business Rules:	<ul style="list-style-type: none"> - The log created must be immutable. - The log entry must be stored into the persistent datastores. - The log entry must contain a UTC timestamp, log level, user performing operation, a category, and a description or message. <ul style="list-style-type: none"> - Valid Levels are Info, Debug, Warning, Error - Valid Categories are View, Business, Server, Data, Datastore - While logging it must not block other processes.
Description:	-
Preconditions:	<ul style="list-style-type: none"> - There must be a persistent data store active to log. - The persistent data store must be accessible by the system. - The persistent data store must have storage capacity to add a log entry
Functional	-
Postconditions:	- The system logs with the correct information and saves it to the data store.
Nonfunctional	-
Success Cases:	<ul style="list-style-type: none"> - The system logs system failure events -
Failure Cases:	<ul style="list-style-type: none"> - Logging process took over 5 seconds to complete upon invocation. - The user is blocked from interacting with the system while logging. - The logging process completes under 5 seconds, but was not able to save to a persistent data store. - The logging process completes under 5 seconds, but did not correctly save the right information to the data store. - Old logs are mutable and can be changed.

Use Case ID:	21
--------------	-----------

Use Case Name:	Logging - User Success
Actors:	- The system
Business Rules:	<ul style="list-style-type: none"> - The log created must be immutable. - The log entry must be stored into the persistent datastores. - The log entry must contain a UTC timestamp, log level, user performing operation, a category, and a description or message. <ul style="list-style-type: none"> - Valid Levels are Info, Debug, Warning, Error - Valid Categories are View, Business, Server, Data, Datastore - While logging it must not block other processes.
Description:	- The logging system will keep track of
Preconditions:	<ul style="list-style-type: none"> - There must be a persistent data store active to log. - The persistent data store must be accessible by the system. - The persistent data store must have storage capacity to add a log entry
Functional:	-
Postconditions:	- The system logs with the correct information and saves it to the data store.
Nonfunctional	-
Success Cases:	- The system logs user success events
Failure Cases:	<ul style="list-style-type: none"> - Logging process took over 5 seconds to complete upon invocation. - The user is blocked from interacting with the system while logging. - The logging process completes under 5 seconds, but was not able to save to a persistent data store. - The logging process completes under 5 seconds, but did not correctly save the right information to the data store. - Old logs are mutable and can be changed.

Use Case ID:	22
Use Case Name:	Logging - User failures
Actors:	- The system
Business Rules:	- The log created must be immutable.

	<ul style="list-style-type: none"> - The log entry must be stored into the persistent datastores. - The log entry must contain a UTC timestamp, log level, user performing operation, a category, and a description or message. <ul style="list-style-type: none"> - Valid Levels are Info, Debug, Warning, Error - Valid Categories are View, Business, Server, Data, Datastore - While logging it must not block other processes.
Description:	<ul style="list-style-type: none"> - The logging system will keep track of all the failed attempts caused by the user.
Preconditions:	<ul style="list-style-type: none"> - There must be a persistent data store active to log. - The persistent data store must be accessible by the system. - The persistent data store must have storage capacity to add a log entry
Functional:	-
Postconditions:	<ul style="list-style-type: none"> - The system logs with the correct information and saves it to the data store.
Nonfunctional	<ul style="list-style-type: none"> - The logging process should not take longer than 5 seconds.
Success Cases:	<ul style="list-style-type: none"> - The system logs user failure events
Failure Cases:	<ul style="list-style-type: none"> - Logging process took over 5 seconds to complete upon invocation. - The user is blocked from interacting with the system while logging. - The logging process completes under 5 seconds, but was not able to save to a persistent data store. - The logging process completes under 5 seconds, but did not correctly save the right information to the data store. - Old logs are mutable and can be changed.

Use Case ID:	23
Use Case Name:	Archiving
Actors:	<ul style="list-style-type: none"> - The system
Description:	<ul style="list-style-type: none"> - Helps in preserving system resources by offloading log entries
Preconditions:	<ul style="list-style-type: none"> - Persistent data store must be active - Persistent data store must accessible by the system - Archival destination location must have storage capacity

Property Manager Functional	<ul style="list-style-type: none"> - Archival process must consolidate and compress entries being archived - Archival process must offload entries to another location - Archival process must remove offloaded entries from the system after successful archival
Postconditions:	<ul style="list-style-type: none"> - All log entries are archived in a different location, and are removed from the system.
Nonfunctional	<ul style="list-style-type: none"> - Archival process must execute every 00:00:00 AM (local time) on 1st of the month - Archival process must only offload log entries that are older than 30 days - Archival process must complete within 60 seconds upon invocation - System failures from this feature must not result in the system going offline
Success Cases:	<ul style="list-style-type: none"> - Archival process executes at 00:00:00 AM (local time) on the 1st of the month. All log entries older than 30 days are consolidated, compressed and relocated to another location. All archived logs are removed from the system. The entire archival process completes within 60 seconds upon invocation.
Failure Cases:	<ul style="list-style-type: none"> - Archival process did not start at 00:00:00 AM - Archival process started at 00:00:00 AM, but not local time - Archival process started at 00:00:00 AM (local time), but not on the 1st of the month - Archival process started at 00:00:00 AM (local time) on the 1st of the month, but did not archive any log entries even though there are logs older than 30 days - Archival process started at 00:00:00 AM (local time) on the 1st of the month, but did not archive all log entries older than 30 days - Archival process started at 00:00:00 AM (local time) on the 1st of the month. All log entries older than 30 days are not consolidated. - Archival process started at 00:00:00 AM (local time) on the 1st of the month. All log entries older than 30 days are consolidated, but are not compressed. - Archival process started at 00:00:00 AM (local time) on the 1st of the month. All log entries older than 30 days are consolidated and compressed, but is not relocated to another location - Archival process started at 00:00:00 AM (local time) on the 1st of the month. All log entries older than 30 days are consolidated, compressed and are relocated to another location, but archived logs are not removed from the system. - Archival process took longer than 60 seconds to complete upon

	invocation.
--	-------------

Use Case ID:	24
Use Case Name:	User Profile - Property Manager
Actors:	- Property Manager
Business Rules:	<ul style="list-style-type: none"> - There should be only one profile per account. - Only the admins and authorized users can change the profile information.
Description:	- Users will be able to see their account history of past services and projects and their ratings.
Preconditions:	- The user has logged into the website and been taken to the homepage with access to their associated features.
Property Manager Functional	<ul style="list-style-type: none"> - Display their past and current services done and service providers worked with and the ratings of that service - Users will be able to view their history from their profile - The property managers will be presented with their past ratings, the service they had done and service provider that provided it in the "View History" option
Postconditions:	- The user is able to see their ratings of services or service ratings and past projects and services they provided or were given.
Nonfunctional	<ul style="list-style-type: none"> - The user's history will be presented in a neatly organized list. - The list will start with the latest services going down to the oldest services. - The history shows the past 50 services. - The format of the date displayed should be formatted as Month/Day/Year
Success Cases:	- A user with services done navigates to "View History" and sees displayed the service done, the service provider, and the rating given.
Failure Cases:	<ul style="list-style-type: none"> - User navigates to "View History" and sees nothing displayed after having a service done. - Users cannot see their rating for service.

Use Case ID	25
-------------	-----------

Use Case Name	User Profile - Service Provider
Actors	- Service providers
Business Rules	- The user should be able to access their account history
Description	- The User Profile should hold the user's history of appointments, services, and rating
Preconditions	- The user has logged into the website and been taken to the homepage with access to their associated features.
Functional	<ul style="list-style-type: none"> - Display their past and current services they provide, the property manager and property address they are servicing - Users will be able to view their history from their profile - Users will be presented with their past ratings, and properties that they provided a service to using the "View History" option
Postconditions	- User has access to their profiles history
Nonfunctional	<ul style="list-style-type: none"> - The user's history will be presented in a neat organized list. - The list will start with the latest services going down to the oldest services. - The history shows the past 50 services. - The format of the date displayed should be formatted as Month/Day/Year
Success Cases	- A user with services done navigates to "View History" and sees displayed the service done, the service provider and ratings given.
Failure Cases	<ul style="list-style-type: none"> - User navigates to "View History" and sees nothing displayed after having a service done. - The user cannot see their rating for a service.

Use Case ID:	26
Use Case Name:	Calendar -Add Event
Actors:	- All Users
Business Rule:	<ul style="list-style-type: none"> - Users must have an account with a valid email. - Calendar displays users appointments with a date and time.
Description:	- This feature allows property manager user to add event such as scheduling appointments and future projects
Preconditions:	- The user has successfully logged into the website and is

	directed to the homepage where they can select the calendar feature.
Property Manager Functional	<ul style="list-style-type: none"> - Allow the user to “add” an appointment <ul style="list-style-type: none"> a. Require the date, time and location of appointment - Display upcoming appointments - Display ongoing services and projects - The calendar displays the upcoming 30 days.
Postconditions:	<ul style="list-style-type: none"> - An appointment is scheduled, modified or deleted and the proper individuals are notified.
Nonfunctional	<ul style="list-style-type: none"> - User are notified of status update messages within 3 seconds - The calendar lists upcoming appointments, ongoing services and projects. - Calendar will also be limited to 5 appointments per day. - Date format should be “Month/Day/Year”
Success Cases:	<ul style="list-style-type: none"> - User modifies an appointment. System accurately shows the modified appointment. Sidebar is updated with the correct information for the appointment. - User goes to make an appointment. User enters the date of appointment. System updates the user’s calendar.
Failure Cases:	<ul style="list-style-type: none"> - User navigates to “add” on the calendar. User enters the date of appointment. System does not update the user’s calendar. - The user navigates to “modify” on the calendar. After the user input system does not change the modify new content on to the appointment. - The user with scheduled appointments navigates to “Calendar”. Calendar displays no scheduled appointments.

Use Case ID	27
Use Case Name	Calendar - Delete Event
Actor	<ul style="list-style-type: none"> - All Users
Business Rules	<ul style="list-style-type: none"> - Users must have an account with a valid email. - Calendar displays users appointments with a data and time
Description	<ul style="list-style-type: none"> - This feature allows users to delete an events such as appointments and projects
Preconditions	<ul style="list-style-type: none"> - The user has successfully logged into the website and is directed to the homepage where they can select the calendar feature.

Functional	<ul style="list-style-type: none"> - Allow the user to “cancel” an appointment - Allow the user to delete projects from calendar - Notify all parties of cancellation
PostConditions	<ul style="list-style-type: none"> - An appointment is scheduled, modified or deleted and the proper individuals are notified.
NonFunctional	<ul style="list-style-type: none"> - Other user party will be notified within 10 seconds when a cancellation of appointment were to happen. - System will be updated within 5 seconds of a deletion of an event.
Success Cases	<ul style="list-style-type: none"> - User deletes the appointment. Calendar is updated by the system. User no longer sees the appointment on the calendar. - User deletes a project. Calendar is updated by the system. User no longer sees the project displayed on the calendar.
Failure Cases	<ul style="list-style-type: none"> - User deletes an appointment. Calendar is not updated by the system. User still sees canceled appointment on the calendar. - Users' deletion notification takes more then 10 seconds. - User deleted a project. Calendar is not updated by the system. User is still able to see project displayed on the calendar.

Use Case ID	28
Use Case Name	Calendar - Modify Event
Actor	<ul style="list-style-type: none"> - All Users
Business Rules	<ul style="list-style-type: none"> - User should be able to make appointments and have access to a calendar - Calendar displays users appointments with a date and time.
Description	<ul style="list-style-type: none"> - This feature allows users to modify appointments or project descriptions
Preconditions	<ul style="list-style-type: none"> - The user has successfully logged into the website and is directed to the homepage where they can select the calendar feature.
Functional	<ul style="list-style-type: none"> - Allow the user to “modify” an appointments date, time and location - Notify all parties of modification of event - Allow the user to modify a project's description.
Postconditions	<ul style="list-style-type: none"> - An appointment is scheduled, modified or deleted and the proper individuals are notified.

Nonfunctional	<ul style="list-style-type: none"> - Notification of modification of an event is sent within 10 seconds. - System updates any modification of an event within 5 seconds
Success Cases	<ul style="list-style-type: none"> - User modifies an appointment. System accurately shows the modified appointment. Sidebar is updated with the correct information for the appointment. - System notifies users when there has been a modification to an appointment. System sends this notification in 6 seconds.
Failure Cases	<ul style="list-style-type: none"> - The user navigates to “modify” on the calendar. After the user input system does not change the modify new content on to the appointment. - Modification notifications take 60 seconds to send. - System does not notify users of modification to an appointment.

Use Case ID	29
Use Case Name:	Document Storage - Search Document
Actors:	- All Users
Business Rule:	- Users must have an account with a valid email.
Description:	- This feature allows users to share files and search up stored files using an optical text search algorithm.
Preconditions:	- The user has successfully logged into the website and navigated to their user profile. The user navigates to files.
Functional:	<ul style="list-style-type: none"> - User is able to search for documents - Use optical text search algorithm - Display if a document is signed - Display if a document is shared and with who
Postconditions:	<ul style="list-style-type: none"> - A document is retrieved from the database and shown to the user. - If no document matches then it returns a notification bubble.
Nonfunctional	<ul style="list-style-type: none"> - The user should be able to search for a document using the optical text search algorithm. - Searching for a document using optical text search algorithm should be within 3 seconds - File uploaded into the system should be within 3 seconds.

Success Cases:	<ul style="list-style-type: none"> - The correct documents are displayed after a search. - There are no matching documents so nothing is shown.
Failure Cases:	<ul style="list-style-type: none"> - The document is in the system but not displayed after search. - The document is not in the system but another document is shown after the search. - The user can search for documents they are not allowed to see. - The system gets stuck in a loop searching.

Use Case ID	30
Use Case Name	Document Storage - Upload Document
Actor	- All Users
Business Rule	<ul style="list-style-type: none"> - Documents being uploaded should be in pdf format only. - The user is given a maximum storage amount of 128MB.
Description	- The user should be able to upload a document and share it with other users.
Precondition	- The user successfully signed in to their account and navigated to document storage
Functional	<ul style="list-style-type: none"> - User should be able to upload documents - The user can specify if a document needs to be signed. - Users can share documents with other users. - The user is shown an status update message for uploaded or signed documents
Postcondition	<ul style="list-style-type: none"> - A document is uploaded to the database and shown to the user. - A document can be marked with the option to sign. - If the user selected a recipient then it should automatically send it to that user.
Nonfunctional	<ul style="list-style-type: none"> - File uploaded into the system should be within 3 seconds. - A file uploaded that needs to be signed is given a prefix of "[Signature Needed]" in the document name.
Success Cases	<ul style="list-style-type: none"> - User navigates to "Upload". User uploads a signed document to the document storage. System prompts a message "Uploaded Successfully" - User uploads a document to the document storage to be signed. The user shares the document to another user's account. The recipient user receives documents into their document storage.
Failure Cases	- User navigates to "Upload". User uploads a signed document to the document storage. System does not prompt a message.

	<ul style="list-style-type: none"> - User uploads a document. System prompts a message “Uploaded Successfully” however the document was not stored properly and is unable to be found. - A document was uploaded that was marked to be signed but a “[Signature Needed]” prefix was not added in the name.
--	--

Use Case ID	31
Use Case Name	Document Storage - Sign Document
Actor	- All Users
Business Rule	- The user should be able to sign important documents
Description	- The system will allow the user to digitally sign a document and add the date and time of signature
Precondition	<ul style="list-style-type: none"> - The user has successfully signed into their account and navigated to document storage - The user sees a sign document request and clicks on it.
Functional	<ul style="list-style-type: none"> - Users are able to sign documents that are marked to be signed. - The user’s signature must not be blank. - System records the date and time of the signature - The user is shown an status update message for uploaded or signed documents - Users should be able to mark files that need to be signed using a check box.
Postcondition	- A document is signed and shown to the user.
Nonfunctional	<ul style="list-style-type: none"> - The signature is shown with time and date stamps. - The user will be notified via email within 5 seconds of request. - Documents that need to be sign are marked as unsigned - Unsigned documents are shown at the top of the document storage list
Success Case	- Signed documents are marked as signed and the name prefix is changed to “[Signed]” and the signed copy is uploaded to the user that signed it.
Failure Case	<ul style="list-style-type: none"> - The user signs the document but then the signature is not saved. - The user does a blank signature and the document is still saved as signed. - The document is signed and the name prefix was not changed to “[Signed]”

Use Case ID	32
Use Case Name	Document Storage - Delete Document
Actor	- All Users
Business Rule	- The user should be able to delete only documents that they uploaded.
Description	- The system will allow the user who uploaded the document to be able to delete the document.
Precondition	- The user has successfully signed into their account and navigated to document storage and then selected a document that they uploaded.
Functional	<ul style="list-style-type: none"> - The user must click on a document that they uploaded - The user must then click delete documents and the system removes the document from the
Postcondition	- The document is deleted from the users and in the database.
Nonfunctional	<ul style="list-style-type: none"> - The document should be deleted in under 5 seconds. - The document should be
Success Case	- The user requests to delete a document that they uploaded and it is deleted from the database.
Failure Case	<ul style="list-style-type: none"> - The user requested to delete a document but it was not deleted from the database. - The user who uploaded the document is unable to see the button to delete the document. - A user who did not create the document is able to delete the document.

Use Case ID:	33
Use Case Name:	Dashboard - Property Manager
Actors:	- Property Manager
Business Rules:	- Users who are property managers will have a dashboard that displays upcoming appointments, services, and current property value.
Description:	- This feature allows the user to view their nearest upcoming appointments, services and rating

Preconditions:	<ul style="list-style-type: none"> - The user has navigated to the website and successfully logged into the system and directed to the homepage.
Functional	<ul style="list-style-type: none"> - Display the rating of an service provider - Display the nearest upcoming appointment - Display the current property value estimation of a property manager - Display nearest appointment and service details <ul style="list-style-type: none"> a. Date and Time - User should be able to view their dashboard on the homepage
Postconditions:	<ul style="list-style-type: none"> - The user can see their upcoming appointments, rating and current property value for property managers.
Nonfunctional	<ul style="list-style-type: none"> - The dashboard displays important information like appointments, services, and ratings precisely and clearly. - The rating will be shown as a number #/5 - Date should be formatted at "Month/Day/Year" - Time should be in Pacific Standard time, formatted as "00:00" - Dashboard will be updated daily
Success Cases:	<ul style="list-style-type: none"> - Property manager user successfully logs into their account. User then navigates to their homepage. The system shows a user dashboard with user ratings, upcoming appointments, and the current property value estimation.
Failure Cases:	<ul style="list-style-type: none"> - User successfully logs into their account. User then navigates to the homepage. The system shows the user dashboard with upcoming appointments, however appointments do not show date and time of the appointment. - The rating on the dashboard does not display in the format of a number out of 5 ; "#/5".

Use Case ID	34
Use Case Name	Dashboard - Service Manager
Actor	<ul style="list-style-type: none"> - Service Providers
Business Rule	<ul style="list-style-type: none"> - Users who are service providers will have a dashboard that displays upcoming appointments, and the rating they received from their employer.

Description	<ul style="list-style-type: none"> - This feature allows the user to view their nearest upcoming appointments, services and rating
Precondition	<ul style="list-style-type: none"> - The user has navigated to the website and successfully logged into the system and directed to the homepage.
Functional	<ul style="list-style-type: none"> - Display the rating of an service provider - Display the nearest upcoming appointment - Display the current property value estimation of a property manager - Display nearest appointment and service details <ul style="list-style-type: none"> a. Date and Time - User should be able to view their dashboard on the homepage
Postcondition	<ul style="list-style-type: none"> - User dashboard shows their correct upcoming appointments, service and ratings
Nonfunctional	<ul style="list-style-type: none"> - The dashboard is neat and organized and displays important information clearly and precisely. - The rating will be shown as a number #/5 - Date should be formatted at "Month/Day/Year" - Time should be in Pacific Standard time, formatted as "00:00" - Dashboard will be updated daily
Success Case	<ul style="list-style-type: none"> - Service provider users successfully log into their account. User then navigates to their homepage. The system shows the user dashboard with their service ratings total, and upcoming appointments.
Failure Case	<ul style="list-style-type: none"> - User successfully logs into their account. User then navigates to the homepage. The system shows the user dashboard with upcoming appointments, however appointments do not show date and time of the appointment. - The rating on the dashboard does not display in the format of a number out of 5 ; "#/5".

Property Manager Features:

Use Case ID:	35
Use Case Name	Property Evaluation
Actors:	<ul style="list-style-type: none"> - Property Managers
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager, with a property. - Users will have to accept location permission. - The user is allowed 100 characters for room descriptions.

Description:	<ul style="list-style-type: none"> - This feature will allow users to get an estimated property value by prompting the user to answer questions regarding the condition, size, bath, and bedroom count of their home.
Preconditions:	<ul style="list-style-type: none"> - The user has navigated to the website and successfully logged into the system and selected the Property Evaluation feature from the homepage.
Functional	<ul style="list-style-type: none"> - Calculate the estimated user property value - Display the calculated property value in dollars - Only accept US metric system measurements - Questions can be answered in the form of numbers or as a drop down menu with specific conditions. - User should be able to enter descriptions of their property outdoors - User should be able to enter descriptions of their property indoors - Users are required to enter the number and bedrooms and baths.
Postconditions:	<ul style="list-style-type: none"> - The estimated property value is displayed to the user along with the details of what the value was based on.
Nonfunctional	<ul style="list-style-type: none"> - The generated value will be clearly shown in an output box. - The questions are easy to read and understand if not then an explanation will be provided when appropriate. - Calculating the estimated property value should be within 5 seconds. - Descriptions are written in Arial for readability for users.
Success Cases:	<ul style="list-style-type: none"> - Property managers are prompted with questions about their property. Their profile is updated with details about their property and their calculated value of the property. - Calculated value of the property shows value in US dollars.
Failure Cases:	<ul style="list-style-type: none"> - User navigates to the property evaluation feature. User answers all questions available. The system does not give the user their estimation of the property. - The user navigates to the property evaluation feature. The user is not able to see the questions displayed on the page.

Use Case ID	36
Use Case Name	Maintenance and Renovation - Compare
Actor	<ul style="list-style-type: none"> - Property Managers.

Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property. - The service title must be under 50 characters. - The service description must be under 250 characters.
Description	<ul style="list-style-type: none"> - This service will show a list of available services for the users to compare information.
Preconditions	<ul style="list-style-type: none"> - The user has successfully logged into the website and selected the Maintenance and Renovation feature from the homepage. - The user is a property manager.
Functional	<ul style="list-style-type: none"> - The user must enter a name of a type of service which must be under 50 characters.. - The system must create and display a list of services. - Each service must show a price estimate, title, and description. - If there are no available services then the system must pop up a notification notifying the user.
Postcondition	<ul style="list-style-type: none"> - User sees a list of available services. - If no available services then the user sees a notification bubble saying there are none available.
Nonfunctional	<ul style="list-style-type: none"> - The system must create the list within 5 seconds. - The notification bubble must be created in under 5 seconds. - The entire title and description must be viewable.
Success Case	<ul style="list-style-type: none"> - The user enters a service name and if there are matches gets a list of services related to what they entered. - The user enters a service name and if there are no matches it returns a notification for it.
Failure Case	<ul style="list-style-type: none"> - The user enters a service name and does not get a list of services even though they exist. - The user enters a service name and it does not have any matches but nothing appears. - The user enters a service name and returns a list of completely different services that don't match the entered name.

Use Case ID	37
Use Case Name	Maintenance and Renovation - Estimate Project
Actor	<ul style="list-style-type: none"> - Property Managers
Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property.
Description	<ul style="list-style-type: none"> - This service will give the users an estimate of how much their property value will increase or decrease if a particular service is done to their property

Preconditions	<ul style="list-style-type: none"> - User has successfully logged into the account and navigated to the Maintenance and Renovation page. Users have select a service
Functional	<ul style="list-style-type: none"> - Calculates the property value using property evaluation and the selected service
Postcondition	<ul style="list-style-type: none"> - Property value estimate is display to the user
Nonfunctional	<ul style="list-style-type: none"> - Property value estimation is displayed to the user in US dollars - Property value estimation is added project plan report
Success Case	<ul style="list-style-type: none"> - The property value is calculated and displayed to the user
Failure Case	<ul style="list-style-type: none"> - Property value is not calculated and displayed to the user - Property value is calculated and not displayed to the user

Use Case ID	38
Use Case Name	Maintenance and Renovation - Price Charts
Actor	<ul style="list-style-type: none"> - Property Managers
Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property.
Description	<ul style="list-style-type: none"> - Price charts displays to the user the price of an project, service or item over a certain period of time
Preconditions	<ul style="list-style-type: none"> - User has successfully logged in and has navigated to the Maintenance and Renovation page
Functional	<ul style="list-style-type: none"> - Gather price information over a selected period of time - Graph price information - Display Price chart to user
Postcondition	<ul style="list-style-type: none"> - Price chart is displayed
Nonfunctional	<ul style="list-style-type: none"> - The period of time for the price chart can be adjusted - The price chart is display as a line graph - The price and period of time on opposing axis
Success Case	<ul style="list-style-type: none"> - The price chart gathers information concerning the correct object - The price chart is created and display
Failure Case	<ul style="list-style-type: none"> - The price chart is developed but not displayed - The price chart shows incorrect information. - The chart

Use Case ID	39
Use Case Name	Maintenance and Renovation - Add Project
Actor	- Property Managers
Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property. - The user must be logged into their accounts. - The user will be limited to having 3 different projects simultaneously.
Description	- Users will be able to create project plans and view them on their account
Preconditions	- User is successfully logged in and navigates to their projects in the Maintenance and Renovation section
Functional	<ul style="list-style-type: none"> - The user will be able to create a project plan by inputting the date, time, service and budget. - User will be able to choose from services within their budget
Postcondition	- A project plan will be will be created and displayed to the user
Nonfunctional	<ul style="list-style-type: none"> - Users will be able to sort services by rating and/or price - Users project will be added to their profile within 5 seconds
Success Case	- Users project is added to account within 5 seconds and displayed to the user
Failure Case	<ul style="list-style-type: none"> - User project is added to system within 5 seconds but not displayed to the user - User project is not added within 5 seconds but displayed to the user - User project is not added and not displayed to the user. User receives "Unable to Add Project" error message

Use Case ID	40
Use Case Name	Maintenance and Renovation - Edit Project
Actor	- Property Managers
Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property. - The user must be logged into their accounts. - The user must be the creator of the project.
Description	- The user will be able to edit a project that they created such as the date, time, service provider, service, and budget.

Preconditions	<ul style="list-style-type: none"> - User is successfully logged into their account and navigates to Maintenance and Renovation project
Functional	<ul style="list-style-type: none"> - User should be able to edit existing projects - The user can edit the future services required for the project. - Users can edit the date and time they plan to have services. - Users will be increase or decrease their project budget
Postcondition	<ul style="list-style-type: none"> - Users project plan is updated in the system
Nonfunctional	<ul style="list-style-type: none"> - User modifications are updated within 5 seconds and displayed to the user
Success Case	<ul style="list-style-type: none"> - User project modifications are updated and displayed to the user within 5 seconds
Failure Case	<ul style="list-style-type: none"> - User project modifications are updated within 5 seconds but not displayed to the user. - User project modifications are not updated and user is displayed old data - User project modifications are not updated and not displayed to the user - User receives an “Unsuccessful Update” error message

Use Case ID	41
Use Case Name	Maintenance and Renovation - Delete Project
Actor	<ul style="list-style-type: none"> - Property Managers
Business Rule	<ul style="list-style-type: none"> - User must be a property manager, with a property. - The user must be logged into their accounts. - The user must be the creator of the project.
Description	<ul style="list-style-type: none"> - The user will be able to delete a project that they created.
Preconditions	<ul style="list-style-type: none"> - User has successfully logged into their account and navigated to the Maintenance and Renovation option. - User has one more projects
Functional	<ul style="list-style-type: none"> - Users project plan and data is deleted from the database
Postcondition	<ul style="list-style-type: none"> - Project is deleted from the user's account. - The user can no longer access deleted projects. - User receives and “Project Delete Successful” message
Nonfunctional	<ul style="list-style-type: none"> - User is able to delete project using the “Delete Project” option

	<ul style="list-style-type: none"> - User project is deleted within 5 seconds
Success Case	<ul style="list-style-type: none"> - User project is deleted within 5 seconds and receives an “Project Delete Successful” message - The user does not have the rights to delete a project and can not delete the project.
Failure Case	<ul style="list-style-type: none"> - User project is not deleted within 5 seconds and does not receive success message - User project could not be deleted and receives an “Unable to Delete” message - User can delete a project that they should not be able to delete.

Use Case ID:	42
Use Case Name:	Service Management - Service Request
Actors:	<ul style="list-style-type: none"> - Property Manager
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager with a known address.
Description:	<ul style="list-style-type: none"> - This feature will allow the user to manage all their services with service providers such as requesting, canceling, or changing the frequency of service.
Preconditions:	<ul style="list-style-type: none"> - The user has successfully logged in to the website, and selected the Service Management feature from the homepage.
Service Request Functional	<ul style="list-style-type: none"> - Display services and service providers list - Allow the user to request services - Notifies service provider of request - Allow users to enter additional comments
Postconditions:	<ul style="list-style-type: none"> - A service request is created and sent, waiting for the service provider’s response.
Nonfunctional	<ul style="list-style-type: none"> - The search bar should complete the search within 5 seconds. - Notification to property manager should be within 5 seconds - Notification to service provider should be within 5 seconds - Updating service data should take within 3 seconds. - Comment text box should be in Arial - Comment text back only allows 150 characters
Success Cases:	<ul style="list-style-type: none"> - The user wants to add a service. User chooses the service and the system updates user services. - The system

Failure Cases:	<ul style="list-style-type: none"> - User chooses a service and the system does not update the amount of services the user has. - User goes to search for a service and it takes longer than 5 seconds. - The user successfully is able to rate the service that was performed. System does not update the service provider's rating.
----------------	--

Use Case ID:	43
Use Case Name:	Service Management - Frequency Change
Actors:	<ul style="list-style-type: none"> - Property Manager
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager with a known address.
Description:	<ul style="list-style-type: none"> - This feature will allow the user to manage all their services with service providers such as requesting, canceling, or changing the frequency of service. In addition to rating services, and seeing all of their services.
Preconditions:	<ul style="list-style-type: none"> - The user has successfully logged in to the website, was directed to the homepage and selected the Service Management feature from the homepage.
Frequency Change & Cancellation Functional	<ul style="list-style-type: none"> - Display the users current services - Allow the user to Cancel or change the frequency of an service - Notify the property manager - Update service database
Postconditions:	<ul style="list-style-type: none"> - The service request or change is sent to the service provider and the service rating is uploaded.
Nonfunctional	<ul style="list-style-type: none"> - The search bar should complete the search within 5 seconds. - Database should record user rating within 3 seconds. - Notification to property manager should be within 5 seconds - Notification to service provider should be within 5 seconds - Updating service data should take within 3 seconds. - Comment text box should be in Arial
Success Cases:	<ul style="list-style-type: none"> - User is successfully able to rate the service that was performed. System updates the service provider's rating. - User navigates to a service change. Service change is sent to the service provider. User confirms service change. System updates user services. - The user wants to add a service. User chooses the service and

	the system updates user services.
Failure Cases:	<ul style="list-style-type: none"> - User chooses a service and the system does not update the amount of services the user has. - User goes to search for a service and it takes longer than 5 seconds. - The user successfully is able to rate the service that was performed. System does not update the service provider's rating.

Use Case ID:	44
Use Case Name:	Service Management - Rating
Actors:	<ul style="list-style-type: none"> - Property Manager
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager with a known address.
Description:	<ul style="list-style-type: none"> - This feature will allow the user to manage all their services with service providers such as requesting, canceling, or changing the frequency of service. In addition to rating services, and seeing all of their services.
Preconditions:	<ul style="list-style-type: none"> - The user has successfully logged in to the website, was directed to the homepage and selected the Service Management feature from the homepage.
Ratings Functional	<ul style="list-style-type: none"> - Allow the user rate a service - Display a likert scale - Record user rating in the database - Property Managers can enter comments on the service provider.
Postconditions:	<ul style="list-style-type: none"> - The service request or change is sent to the service provider and the service rating is uploaded.
Nonfunctional	<ul style="list-style-type: none"> - The search bar should complete the search within 5 seconds. - Database should record user rating within 3 seconds. - Notification to property manager should be within 5 seconds - Notification to service provider should be within 5 seconds - Updating service data should take within 3 seconds. - Comment text box should be in Arial - Comment is limited to 240 words.
Success Cases:	<ul style="list-style-type: none"> - User is successfully able to rate the service that was performed. System updates the service provider's rating. - User navigates to a service change. Service change is sent to the service provider. User confirms service change. System updates user services.

	<ul style="list-style-type: none"> - The user wants to add a service. User chooses the service and the system updates user services.
Failure Cases:	<ul style="list-style-type: none"> - User chooses a service and the system does not update the amount of services the user has. - User goes to search for a service and it takes longer than 5 seconds. - The user successfully is able to rate the service that was performed. System does not update the service provider's rating.

Use Case ID:	45
Use Case Name:	Neighborhood Crime Alerts - View Alerts
Actors:	<ul style="list-style-type: none"> - User is Property Managers
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager, with a property. - User property location is needed for crime alert notifications.
Description:	<ul style="list-style-type: none"> - The system will allow users to view crime on the Neighborhood Crime Alert.
Preconditions:	<ul style="list-style-type: none"> - The user is a Property Manager and has successfully logged into the website, was directed to the homepage, and selected the Neighborhood Crime Alerts feature.
Functional:	<ul style="list-style-type: none"> - The system must make a popup bubble when a user hovers their mouse above a dot on the map and the bubble when the cursor is removed. - Users can only see crime alerts within a 25 mile radius of their property. - The user must see the location of where the crime occurred. - The user must see the time of when the crime happened.
Postconditions:	<ul style="list-style-type: none"> - The system shows a map with all current crime alerts
Nonfunctional:	<ul style="list-style-type: none"> - Alert notification for crime reports should be sent out within 5 seconds. - Location should be formatted as absolute location: "street, city, zip code" - Description box is in Arial for legibility. - Time is formatted as hours:minutes: "00:00"
Successful Cases:	<ul style="list-style-type: none"> - User navigates to the crime map. When the user hovers over the map, the user is able to see the crime that was committed with

	the data and time.
Failure Cases:	<ul style="list-style-type: none"> - User has successfully logged into the website, was directed to the homepage, and selected the Neighborhood Crime Alerts feature but the system does not respond. - User clicks a crime alert and sees no information. - User is not logged in but can still access crime alerts. - The user can see crime alerts outside their property radius.

Use Case ID:	46
Use Case Name:	Neighborhood Crime Alerts - Add Alert
Actors:	<ul style="list-style-type: none"> - User is Property Managers
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager, with a property. - User must be the creator of the crime alert. - Users are allowed 150 characters per crime alert description. - Users are allowed a maximum of 2 posts within 24 hours. - The 24-hour restriction timer starts after the first post. - A crime alert post must be automatically deleted after 24 hours.
Description:	<ul style="list-style-type: none"> - The system will allow users to report crimes to the Neighborhood Crime Alerts for others to see.
Preconditions:	<ul style="list-style-type: none"> - The user is a Property Manager and has successfully logged into the website, was directed to the homepage and selected the Neighborhood Crime Alerts feature.
Functional:	<ul style="list-style-type: none"> - The user must enter a location of where the crime occurred. - The user must enter a time of when the crime happened. - The user must enter a description of the crime. - The user must enter a title of the crime.
Postconditions:	<ul style="list-style-type: none"> - The crime alert is publicly posted and other users within a radius are automatically notified. - After 24 hours the post expires and is automatically deleted from the crime map.
Nonfunctional:	<ul style="list-style-type: none"> - Alert notification for crime reports should be sent out within 5 seconds. - The system will limit the title to 50 characters. - The system will limit the description to 150 characters. - User must add a location in the format of: "street, city, zip code" - User must add a time formatted as hours:minutes: "00:00"
Successful Cases:	<ul style="list-style-type: none"> - User navigates to the crime map. Then click on the add crime alert button and fill out the required information correctly. The

	<p>user then clicks the submit crime alert button and the crime map is successfully updated.</p> <ul style="list-style-type: none"> - 24 hours have passed since the crime alert was posted and the post gets automatically deleted.
Failure Cases:	<ul style="list-style-type: none"> - User successfully reports the crime. The system does not update the crime map. - The user incorrectly filled out a crime alert form or left parts blank but the crime alert was still created. - The user correctly fills out a crime alert form and it creates the crime alert but it does not show the correct information. - It has passed 24 hours and the crime alert has not been automatically deleted.

Use Case ID:	47
Use Case Name:	Neighborhood Crime Alerts - Delete Alert
Actors:	<ul style="list-style-type: none"> - User is Property Managers
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager, with a property. - User must be the creator of the crime alert.
Description:	<ul style="list-style-type: none"> - This feature will allow users to delete crimes alerts that they created that have not expired yet.
Preconditions:	<ul style="list-style-type: none"> - The user is a Property Manager and has successfully logged into the website, was directed to the homepage and selected the Neighborhood Crime Alerts feature.
Functional:	<ul style="list-style-type: none"> - The system must delete the post from the crime map when the creator of the post requests it. - The system should notify the post creator when the post have been successfully deleted - The system must allow only the creator of the post to delete their own posts.
Postconditions:	<ul style="list-style-type: none"> - The user successfully deletes their crime alert post and it is removed from the crime map.
Nonfunctional:	<ul style="list-style-type: none"> - The system must delete the post from the crime map within 5 seconds. - The system should notify the post creator when the post have been successfully deleted within 5 seconds
Successful Cases:	<ul style="list-style-type: none"> - User clicks edit post then clicks delete post. Then the post is deleted and removed from the crime map by the system.
Failure Cases:	<ul style="list-style-type: none"> - User clicks edit post and does not see the delete post button.

	<ul style="list-style-type: none"> - User clicks edit post then clicks delete post but the post is not deleted by the system. - The user successfully deletes their post but it is still showing on the crime map.
--	--

Use Case ID:	48
Use Case Name:	Neighborhood Crime Alerts - Edit Alert
Actors:	<ul style="list-style-type: none"> - User is Property Managers
Business Rules:	<ul style="list-style-type: none"> - User must be a property manager, with a property. - User must be the creator of the crime alert.
Description:	<ul style="list-style-type: none"> - This feature will allow users to edit their own crime alert posts that have not yet expired.
Preconditions:	<ul style="list-style-type: none"> - The user is a Property Manager and has successfully logged into the website, was directed to the homepage, and selected the Neighborhood Crime Alerts feature.
Functional:	<ul style="list-style-type: none"> - The user must be able to change the title of the crime alert. - The user must be able to change the description of the crime alert. - The user must be able to change the location of the crime alert. - The user must be able to change the time of the crime alert. - The system must not reset the 24 hour automatic delete function.
Postconditions:	<ul style="list-style-type: none"> - The user navigates to their crime alert post and presses the edit button.
Nonfunctional:	<ul style="list-style-type: none"> - Crime alert posts should be updated to the crime alert map within 5 seconds. - The system will limit the title to 50 characters. - The system will limit the description to 150 characters. - User must add a location in the format of: "street, city, zip code" - User must add a time formatted as hours:minutes: "00:00"
Successful Cases:	<ul style="list-style-type: none"> - The user is able to click edit crime alerts and change the data. Then the system updates the crime alert map. - The user is able to click edit crime alerts and change the data but cancels changes. Then the system does not update the crime alert map. - The user is able to click edit crime alerts and does not change the data and clicks saves changes. Then the system does not update the crime alert map.

Failure Cases:	<ul style="list-style-type: none"> - The user clicks edit crime alert button but is not navigated to edit the crime alert. - The user clicks edit crime alert button and edit the post but it is not updated by the system. - The system allows a user who was not the creator to edit the post.
----------------	---

Use Case ID:	49
Use Case Name:	DIY Manager - Upload
Actors:	<ul style="list-style-type: none"> - Property Managers - Service Providers
Business Rules:	<ul style="list-style-type: none"> - The user must be logged in. - User's maximum storage capacity is 128MB. - The upload video quality is restricted to 720p or lower.
Description:	<ul style="list-style-type: none"> - This feature will allow users to upload Do It Yourself tutorials on maintenance and renovation projects for other users to see. Users can save these tutorials and add the project to their calendars.
Preconditions:	<ul style="list-style-type: none"> - The user has successfully logged into the website and selected the DIY feature from the homepage.
Add DIY tutorial Functional	<ul style="list-style-type: none"> - Allow the user to upload DIY tutorial videos - Allow the user to enter DIY steps and materials - User can add DIY tutorials using the "Add DIY" option - User should be able to enter a description and title of the DIY project - The user is shown a status update message for their DIY tutorial post.
Postconditions:	<ul style="list-style-type: none"> - The DIY tutorial is uploaded
Nonfunctional:	<ul style="list-style-type: none"> - The status message should be displayed within 3 seconds. - Text should be displayed in Arial for the description of the DIY video. - If the upload was successful "Upload Successful" is displayed.
Success Cases:	<ul style="list-style-type: none"> - System uploads the video successfully.
Failure Cases:	<ul style="list-style-type: none"> - System does not successfully upload the video. - System successfully uploads the video with no description, - System successfully uploads the video with no sound,

	<ul style="list-style-type: none"> - System successfully uploads the video with no title - System successfully uploads the video but cannot be found through search, - System successfully uploads the video but the total storage used by the user is more than the allocated storage. - A status message after an uploading attempt is not shown, - A status message after an uploading attempt is shown, but with wrong information,
--	--

Use Case ID:	50
Use Case Name:	DIY Manager - Search
Actors:	<ul style="list-style-type: none"> - Property Managers - Service Providers
Business Rules:	<ul style="list-style-type: none"> - Users can use this function without logging in. - Users are able to search for videos using keywords. - Users are able to play searched videos. - Only relevant videos are shown, - relevance is defined by whether or not the title of search results contains the keyword.
Description:	<ul style="list-style-type: none"> - This feature will allow users to search for Do It Yourself tutorials on maintenance and renovation projects uploaded by other users.
Preconditions:	<ul style="list-style-type: none"> - The user has selected the DIY feature from the homepage.
Search DIY tutorial Functional	<ul style="list-style-type: none"> - Users are able to enter keywords in the search bar. - Users are able to search for videos with entered keywords. - Users are able to play a video for their search results. - Users are able to read the description of the video.
Postconditions:	<ul style="list-style-type: none"> - The user was able to click on a search result, and watch a DIY video.
Nonfunctional:	<ul style="list-style-type: none"> - Only 10 search results should be displayed per page. - The DIY tutorial search should be under 5 seconds. - Text should be displayed in Arial for the description of the DIY video.
Success Cases:	<ul style="list-style-type: none"> - The user is able to search for and watch a video from the search result using keywords.
Failure Cases:	<ul style="list-style-type: none"> - No search results are displayed, - Search results are displayed but, with missing titles, - Search results are displayed but, are not relevant to a keyword,

	<ul style="list-style-type: none"> - Search results are displayed but, there are more than 10 search results per page, - Search results are displayed but, there are less than 10 videos per page, while the total search results are more than 10 - The user is not able to type in the search bar. - The user is not able to click on a search result. - The user is able to click on a video but no video is played.
--	--

Use Case ID:	51
Use Case Name:	DIY Manager - Bookmark
Actors:	<ul style="list-style-type: none"> - Property Managers - Service Providers
Business Rules:	<ul style="list-style-type: none"> - Users are able to bookmark a video.
Description:	<ul style="list-style-type: none"> - Users can save these tutorials and add the project to their calendars.
Preconditions:	<ul style="list-style-type: none"> - The user has successfully logged in and searched for a video using the search function.
Bookmark DIY tutorial Functional	<ul style="list-style-type: none"> - Users are able to bookmark a video
Postconditions:	<ul style="list-style-type: none"> - The DIY tutorial is bookmarked to the default bookmark list.
Nonfunctional:	<ul style="list-style-type: none"> - Bookmarks are by default saved to the "ToDo" bookmark list. - Users can have a total of 30 bookmarks.
Success Cases:	<ul style="list-style-type: none"> - The user is able to bookmark a video to the default list.
Failure Cases:	<ul style="list-style-type: none"> - Video is not bookmarked, - Video is bookmarked but, is not saved to the default list, - Video is bookmarked but it exceeds the total number of bookmarks allowed.

Service Provider Features:

Use Case ID:	52
--------------	-----------

Use Case Name:	Request Management
Actors:	- Service Providers
Business Rules:	- The user must be logged in as a service provider.
Description:	- This feature allows service providers to manage their requests from property managers.
Preconditions:	- The user has successfully logged in as a service provider and selected the Request Management feature.
Functional	<ul style="list-style-type: none"> - Display service request - Allow the user to accept or decline a request - All parties are notified when a request is accepted and when an appointment date is scheduled. - The user should be able to view all of their service requests
Postconditions:	- If a service request is accepted, a service is added to the user's list of services.
Nonfunctional:	<ul style="list-style-type: none"> - Request notifications to service providers should be within 5 seconds of the property manager's request. - Appointment dates should be formatted as month/day/year: "00/00/0000" - Service requests can be sorted from most recent to older or older to most recent requests.
Success Cases:	- The user accepts or declines a request and the system notifies both parties and displays the request on the schedule.
Failure Cases	<ul style="list-style-type: none"> - A service request is not displayed. - A service request is displayed but does not have the correct information. - Request is not accepted/declined. - Request is accepted/declined but resolved the opposite way. - Request is accepted/declined but is still displayed. - A notification is not sent to the requestor. - A notification is sent to the requestor, but with the wrong results. - A notification is sent to the requestor, but with the wrong date. - All service requests are not displayed. - All service requests are displayed, but with wrong information.

References:

- Asana. "Business Requirements Document Template: 7 Components [2022] • Asana." *Asana*, 6 July 2022, <https://asana.com/resources/business-requirements-document-template>.
- Blake, Sean. "Use Cases vs. User Stories: How They Differ and When to Use Them." *Easy Agile*, 8 June 2021, <https://www.easyagile.com/blog/use-cases-vs-user-stories/#:~:text=User%20stories%20are%20simple%2C%20short,cases%20contain%20much%20more%20content>.
- Karaleise, director. *YouTube*, YouTube, 7 May 2020, <https://www.youtube.com/watch?v=llXFQjas73U>. Accessed 5 Oct. 2022.
- "Search NYU." *Business+Requirement+Document*, *NYU Search*, 2018, <https://search.nyu.edu/s/search.html?f.All%7CAllDocumentsFill=All&query=business%2Brequirement%2Bdocument&collection=nyu-all-meta-v02>. Accessed 5 Oct. 2022.
- "Top Tips for Writing the Perfect Business Requirements Document." *Tips for Writing Business Requirements Documents* | *Lucidchart Blog*, Lucidchart, 29 Apr. 2020, <https://www.lucidchart.com/blog/tips-for-a-perfect-business-requirements-document>.
- Vong, Vatanak. E-mail to AA. 17 October 2022.
- YouTube*, YouTube, 27 July 2022, <https://www.youtube.com/watch?v=jX6s9A6SCG8>. Accessed 5 Oct. 2022.
- YouTube*, YouTube, 31 May 2020, <https://www.youtube.com/watch?v=QYkkDIXE8Ys>. Accessed 5 Oct. 2022.
- Vong, Vatanak. *Core Components Requirements*, 17 Oct. 2022.