

WeCasa

Business Requirements Document

Team HAGS JP

Team Lead:

Allison Austin

Team Members:

Githel Lynn Suico

Haley Nguyen

Joshua Quibin

Judy Li

Matthew Chung

<https://github.com/githelsui/WeCasa>

Date Submitted: March 13, 2023

BRD Version Table

Version	Description	Date
1.0	Initial Business Requirements <ul style="list-style-type: none"> - Project Overview <ul style="list-style-type: none"> - Document Purpose - Problem - Solution - Target Customers - Competition - Scope - Features - System Failures - Glossary - References 	10/5/2022
1.1	Content Improvements <ul style="list-style-type: none"> - Scope <ul style="list-style-type: none"> - Product-Wide Requirements - System Failures <ul style="list-style-type: none"> - Formatting 	12/05/2022
1.2	Content Improvements <ul style="list-style-type: none"> - Removed implementation details - Claims-based Authorization <ul style="list-style-type: none"> - Added claims checks in Appendix 	12/06/2022
1.3	Content Improvements <ul style="list-style-type: none"> - Provided more detail for metrics and behavior of features - Added more Non-functional Requirements 	1/11/2023
1.4	Content Improvements <ul style="list-style-type: none"> - Core Components 	2/7/2023
2.0	Document Reformatting <ul style="list-style-type: none"> - Renamed 'Target Customers' section to 'Target Audience/Scope' and combined with 'Scope' section - Split up 'Features' Section into 'Core Component Features' and 'App Specific Features' <ul style="list-style-type: none"> - Recategorized Core Component Features <ul style="list-style-type: none"> - Security - User Administration - System Observability - Moved Authorization Section - Core Component 	2/10/2023

	Features > Security Content Improvements <ul style="list-style-type: none">- Logging- Account Recovery- User Management- Added 'Archiving' Section	
2.1	Content Improvements <ul style="list-style-type: none">- Added 'Group Management' Section- Failure Handling<ul style="list-style-type: none">- Authentication- Logout- Logging- Budget Sharing- Calendar- Calendar- Incomplete Task Summary- FAILURE HANDLING ERROR TABLE	2/20/2023

Table Of Contents

BRD Version Table	2
Table Of Contents	4
Project Overview	6
Document Purpose	6
Problem	6
Solution	6
Target Audience/Scope	6
Pain Points	7
Competition	7
Home Organization Apps	7
Social Media Groups/Community Apps	8
Task Management Apps	8
Product-Wide Requirements	8
Functional	8
Non-Functional	8
Core Component Features	9
Security	9
Authentication	9
Authorization	11
Logout	13
User Administration	14
Account Creation/Registration	14
Account Recovery	15
Account Deletion	17
User Management	18
User Access Control	20
System Observability	21
Analytics	21
Logging	24
Archiving	26
User Feedback	28
App Specific Features	30
App Settings	30
Group Management	31
Budget Sharing Bar/Bill List	33
Bulletin Board	35
Calendar	36

Chore List	38
Grocery List	40
Reminders	41
Nudging	42
Incomplete Task Summary	43
Circular Progress Bar	44
Photo/Documentation Uploads	45
Error Handling	47
System Failure	47
FAILURE HANDLING ERROR TABLE	47
Appendix A	69
Claims Checks	69
Glossary	70
References	71

Project Overview

Document Purpose

The purpose of the Business Requirement Documents (BRD) is to gather all business requirements and criteria for success necessary to build WeCasa as per client's needs and requirements. This document focuses on defining the failures, successes, and specifications of WeCasa's system and features, as well as scenarios and solutions for failures. This document should be referenced in relation to the high level design, and project plan.

Problem

Roommates do not have a designated one-stop app that can manage all the necessary organization required in their shared housing situation, resulting in problems of stress and tension due to a lack of coordination and miscommunication.

Solution

The value of our app is its ability to effectively solve the mentioned pain points that arise from living with roommates.

As a solution to these problems, our app is a dependable toolbox for organizing all aspects of shared housing management to relieve the stress and tension of living with roommates.

Target Audience/Scope

WeCasa is designed to make the entire roommate experience go as seamlessly as possible. Anyone from students, to working professionals, to even family members can utilize the app to help streamline daily tasks. Whether or not one feels like their roommates work well with each other, our app will help strengthen weak points in communication. The app is designed for anyone with roommates, but it is especially helpful to those who feel like they live in a disorganized home and could use a third-party to help with that.

Our single-page-application is targeted initially to the US market. As such, it will only support American English (en-US) at launch and comply with California privacy laws. And as our app grows, we would like to expand this to other markets such as Canada and Europe. However, this will not initially be supported.

Furthermore, our product will follow the Web Content Accessibility Guidelines (WCAG, 2022) in order to make our application's content more accessible to our audience.

Pain Points

Several core problems and issues exist for those living with roommates that our app will address:

- **Disorganization:** Roommates often share lists of chores, maintenance dates, and many other items to be managed. For example, it can be difficult to manage whose turn it is to do a specific chore or when to complete it by.
- **Miscommunication:** Roommates will create assumptions that may result in future disagreements unless properly handled with transparency. Therefore, the app's features aim to encourage compromise and communication between roommates.
- **Aversion to Confrontation:** Roommates will run into disagreements and have trouble effectively mitigating the situation, due to an inability to initiate the confrontation. The app will have features that incentivize users to initiate confrontation.
- **Burden of Managing Multiple Apps:** Several task and home management apps exist on the market to help roommates. However, it is overwhelming for the user to sign up and keep track of multiple accounts.

Competition

One of the primary goals of our app is to combine features of several existing apps into one. Our main competitors consist of applications that target both organization and communication within a group. WeCasa derives its core components from a variety of these apps while also providing distinct, original features to its users.

Home Organization Apps

Include gamified chore or task management features for families living under the same roof. Examples of these apps include OurHome, Nipto, and RoomMate. The

target audience for these apps is typically children, and the apps focus on incentivization through a reward system. OurCasa targets adults, and does not use a reward or point system to motivate its users.

Social Media Groups/Community Apps

Include group forum or chat room applications that allow users to share information, content, and/or events. Examples of these apps include Facebook Groups, WhatsApp, GroupMe, Slack, and Discord. While the target audience of these apps intersect with WeCasa, these apps focus on communication rather than organization.

Task Management Apps

Include project management, organization, and note-taking software. Examples of these apps include Trello and Notion. While the target audience for these apps intersect with WeCasa, these apps aim to reach the general audience, rather than being tailored to shared living spaces. Our app is created specifically with roommates in mind, which allows our features to be more defined and useful. Additionally, these apps typically charge a premium to unlock key features and storage, and we hope to provide a solution at no cost.

Product-Wide Requirements

Functional

- **Browser:** The primary browser that will allow users to access the full functionality of our app will be Chrome Version 104.X and above. We chose to target Chrome because this is the most popular browser (Oberlo, 2022). W3C validation will allow our app to be accessible on smaller screens such as mobile devices. (LoginRadius, 2020).

Non-Functional

- **Accessibility:** Our app will utilize the Web Content Accessibility Guidelines (WCAG) as the set of standards for digital accessibility (WCAG, 2022). WeCasa will provide the following accessibility features in all versions of the application:
 - Alternative text for graphics and media

- Clear form labels
- Meeting a color contrast ratio of at least 4:5:1 for normal text and 3:1 for large text (18 point or larger, or 14-18 point and bolded)
- Disabling key traps

Core Component Features

Security

Authentication

- **Description:** A mechanism for identifying a valid user of the system.
- **Functional Requirements**
 - System shall validate the user.
 - A maximum of 5 failed authentication attempts within 24 hours is allowed for the same account before the account is disabled. temporarily for 24 hours or until the user recovers the password.
 - After 24 hours, fail count resets to 0.
 - Account recovery must be performed by the Systems Administrator.
 - Every failed authentication attempt will be logged into a secure database.
 - In case of any failure to authenticate the user, the user must be notified of the reason.
- **Non-functional Requirements**
 - Authentication must take a maximum of 3 seconds.
 - After successfully logging in, the user must be directed to the home page in a maximum of 3 seconds.
- **Required Input:**
 - The user must provide a valid security credential whenever attempting to authenticate with the system
 - Valid usernames must meet the character requirements
 - Minimum of 8 characters
 - a-z
 - 0-9
 - (Optional) The following special characters: . - @
 - Valid passwords must meet the character requirements
 - Minimum of 8 characters

- One of each of the following characters:
 - a-z
 - A-Z
 - 0-9
 - ., @!-
- **Preconditions:**
 1. The user must be in the Login view.
 2. The user must have an existing account.
- **Success Postconditions**
 1. The user successfully logs in
 2. The user is redirected to the user's home page.
 3. The user is notified of invalid email/password with every failed login attempt.
 4. The user is notified of the last attempt before the account is disabled and locked.
 5. If the user's account is disabled, the user is notified to contact the Support for account recovery.
- **Failure Scenarios**
 1. The user abruptly exits from the authentication process.
 2. The user leaves any of the required fields empty.
 3. The user inputs invalid username.
 4. The user input invalid password.
 5. The user is not notified of invalid login attempt.
 6. The user is not notified of disabled account.
 7. Failed attempts are not logged by the system.
 8. The user successfully inputs valid credentials but is still unable to log in.
- **Failure Handling**
 1. Authentication process does not complete. User is prompted to restart the authentication process.
 2. The user is notified to input all required fields.
 3. The user is notified of invalid email/username and to follow the proper guidelines.
 4. The user is notified of invalid password and to follow the proper guidelines.
 5. Error is logged by the system and the user is asked to reload the page and try again.

6. Error is logged by the database and the user is prompted to submit a User Feedback ticket. (see User Feedback)
7. Error is logged by the database and the user is asked to reload the page and try again or attempt account recovery. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)
8. Error is logged by the database and the user is asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)

Authorization

- **Description:** A mechanism for restricting access to protected resources (e.g. functionalities, data, and views) to only valid users
- **Functional Requirements:**
 - Claims-based authorization references the claim value of the user requesting the resource. Access to resources will be based on whether the user possesses a claim with the correct value required by the resource's security policy. See Appendix A for the complete list of claims checks.
 - Users will split up into two main categories
 - System Administrator: Has full read and write permissions for all users. This user has the ability to disable/enable accounts, recover accounts, and monitor the logs and analytics dashboard. (see User Management).
 - Generic User: Any user that is not the system administrator. The 3 permission levels are in effect for these users, and this type of user is created with the Account Creation feature Unauthorized access attempts will be logged by the system (see Logging)
 - Unauthorized users will be prevented from viewing/interacting with any protected views and executing any protected functionalities
- **Non Functional Requirements:**
 - The error rate of users' authorization requests must not exceed 3%
 - Authorization requests must be completed within 5 seconds
 - Failed authorization requests made by a single user will be notified to the system administrators when the number of requests exceeds 20 failures within an hour.
- **Preconditions:**

1. The user must have an existing account.
2. Users must have an active authenticated session
- **Successful Postconditions:**
 1. Access is granted to users attempting to interact with protected functionalities within their authorization scope
 2. Access is granted to users attempting to access protected data within their authorization scope, allowing them perform read operations
 3. Access is granted to users attempting to modify protected data within their authorization scope, allowing them to perform read operations
 4. Access is granted to users attempting to access protected views within their authorization scope, automatically navigating them to view
- **Failure Scenarios:**
 1. Unauthorized access attempts fail to be logged by the system
 2. User attempts to access protected functionalities/data/views outside of their authorization scope
 3. User attempts to modify protected data outside of their authorization scope.
 4. Sensitive protected data that is not within a user's read scope is accessible via a view within their authorization scope.
 5. Sensitive protected data that is not within a user's write scope is accessible via a view within their authorization scope.
- **Failure Handling:**
 1. Failure to log unauthorized access attempts will prompt a failure system log.
 2. Attempts to access protected functionalities/data/views outside of authorization scope will be logged by the system and the user will be prompted with a system message.
 3. Attempts to modify protected data outside of authorization scope will be logged by the system and the user will be prompted with a system message.
 4. Attempts to view protected data outside of authorization scope will be logged by the system, the user will be prompted with a system message, and protected data will be hidden from view
 5. Attempts to modify protected data outside of write scope will be logged by the system and the user will be prompted with a system message.

Logout

- **Description:** This feature allows users to securely log out of their accounts on one device.
- **Functional Requirements:**
 - System shall log the user out of the system.
 - The current logged in user session will end after 5 seconds of invoking the logout feature.
 - The user will be redirected to the system's home page after logging out.
 - All logouts will be logged to a secure database.
 - System failure from invoking the logout feature will not cause the system to go offline.
- **Non-functional Requirements:**
 - Logout process must take a maximum of 5 seconds.
 - Redirection to the system home page must take a maximum of 3 seconds.
- **Required Input:**
 - The user selects the Logout option.
- **Preconditions:**
 1. The user must have an existing account.
 2. The user must be logged in.
- **Successful Postcondition:**
 1. A message is displayed to the user to confirm whether they are sure to logout.
 2. The user is logged out and the active session ends.
 3. The user is redirected to the system's home page.
 4. A confirmation message is displayed stating that they have successfully logged out.
- **Failure Scenarios:**
 1. The user abruptly exits from the logout request.
 2. The user is not logged out and the session remains active.
 3. Confirmation message is not displayed to the user.
 4. The user is not redirected to the system's home page.
 5. Logout is not logged by system.
- **Failure Handling:**
 1. Logout process does not complete. User is required to restart the logout process.

2. Error is logged by the database and the user is asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)
3. Display view error is logged by the system.
4. Display view error is logged by the system.
5. Data store error is logged by the system.

User Administration

Account Creation/Registration

- **Description:** Allows for users to create new accounts within the WeCasa system.
- **Functional Requirements:**
 - For registering an account with WeCasa, users need to provide a unique email address and a password to use when logging into their account.
 - Email address must be valid.
 - A one-time password will be used.
 - This string of characters will be displayed to the user to use once when they first log in.
 - This one-time password will be associated with the user until it is used or expires after 2 minutes. After it is used, the user will be prompted to enter their own password for subsequent use. A user is not “enabled” until they log in with their one-time password.
 - Valid password would consist of the following:
 - Minimum 8 characters
 - One of each characters:
 - A-Z
 - a-z
 - ., @!
 - All user accounts must be stored in the persistent data store.
 - **Non- functional Requirements:**
 - New account data should update in database in less than 5 seconds
 - The error rate of user’s creating a new account must not exceed 5%

- **Required Input:** The following information must be passed in order for a new account to be created:
 1. A valid email
 2. First name
 3. Last name
 4. A valid password
 5. A valid OTP verification consisting of the following:
 - a. 10 character length
 - b. A-Z
 - c. A-z
 - d. 009
 - e. .-@
- **Preconditions:**
 1. User must not be logged in
 2. User must select “Create an Account”
- **Successful Postconditions:**
 1. A new user account is created
 2. The new user data is stored in the database
 3. Verification email is sent to the user’s email
 4. User enters the correct OTP
- **Failure Scenarios:**
 1. The user abandons the registration and exits
 2. User does not receive confirmation email
 3. The new user data is not logged into the database
- **Failure Handling:**
 1. Registration process does not complete. User is required to restart the account creation process.
 2. If the logging feature indicates that the user has not validated their email, when the user returns to the WeCasa site, a message will be prompted to them to check their email/resend the confirmation email.
 3. The system prompts automatic logging re-attempt. After 3 subsequent failed attempts, a data store error is logged by the system. User is prompted via email to re-register their account.

Account Recovery

- **Description:** Allows for users to regain access to an active or disabled account in the WeCasa system.

- **Functional Requirements:**
 - The user must provide assigned username and valid OTP in order to submit an account recovery request and to receive a secure link.
 - An authorized system admin must be able to view the latest account recovery request for all non-admin users.
 - Once a successful account recovery is authorized by the system admin, the user will be able to be authenticated into the system.
- **Non- functional Requirements:**
 - The update within the database must be less than 5 seconds.
 - The error rate of user's creating a new account must not exceed 5%
- **Required Input:**
 1. Request for Account Recovery
 2. Input of user's assigned username
- **Preconditions:**
 1. User must not be logged in.
 2. User must have a valid account in the WeCasa system.
 3. User must be in the Account Recovery view.
- **Successful Postconditions:**
 1. User provides assigned username and valid OTP.
 2. The request is made available within 5 seconds of invocation.
 3. The affected user regains access to the system within 5 seconds of invocation.
- **Failure Scenarios:**
 1. User provides an invalid username
 2. User provides an invalid OTP
 3. OTP does not match
 4. An authorized system admin completes the account recovery for the user, but the user does not regain access to the system within 5 seconds
- **Failure Handling:**
 1. In the case the user provides an invalid username, a message will prompt the user to enter in a new username .
 2. In the case the user provides an invalid OTP, a message will prompt the user to retry.
 3. User is prompted to re-enter OTP or request a new code.
 4. Business error is logged by the system.

Account Deletion

- **Description:** Allows for users to delete an user account within the WeCasa system.
- **Functional Requirements:**
 - All data (excluding logs) of this user will be removed from the system database.
 - All personal identifiable information is also removed from the system
 - Only a system administrator can delete another system administrator account.
- **Non- functional Requirements:**
 - Update the database in less than 3 seconds
 - Error rates of user's deleting accounts must not exceed 10%
 - User must confirm their selection of account deletion.
- **Required Input:**
 1. Selection of Delete account
 2. Confirmation of Action
 3. Confirmation of permanent deletion
 4. Correct password of the user.
- **Preconditions:**
 1. User must have a valid WeCasa account
 2. User must be in a logged in session
 3. User must be in the Account Setting View.
- **Successful Postconditions:**
 1. All stored personal identifiable information of the user must be deleted
 2. All logs and other data of this user must be removed from the system.
 3. Upon confirmation of account deletion action the user is redirected to login view.
 4. A system message displays "Account deletion successful"
- **Failure Scenarios:**
 1. The user chooses to delete their accounts and confirms action, however the system doesn't delete both personal identifiable data and user account data.
 2. Data is not permanently deleted from the system.
 3. A system message confirming account deletion is not shown or if the wrong message is shown instead.
 4. The user is not automatically redirected to the login view after account deletion.

- **Failure Handling:**

1. Failed attempts to delete an account will be logged by the system. After 3 subsequent failed attempts, a data store debug error will be logged to the system and the user will be prompted with a system message and asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)
2. Failed attempts to permanently delete the user data will be logged by the system. After 3 subsequent failed attempts, a data store debug error will be logged to the system, and the user will be prompted with a system message and asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)
3. Failed attempts to display system messages will be logged by the system. After 3 subsequent failed attempts, a view debug error will be logged to the system.
4. Display view error is logged by the system.

User Management

- **Description:** A system for WeCasa System Administrators to manage existing users along with their WeCasa group spaces
- **Functional Requirements**
 - System Administrators are the only users within the system with access to the User Management view
 - System Administrators are the only users with access to the entire system, and must have access to view/modify any user accounts within the system along with their associated profile data, including other System Administrators
 - At least one System Administrator must exist within the system at any given time
 - System Administrators are the only users that can create other system administrator accounts
 - System Administrators will have access to the following management operations
 - Create Account
 - Update Account
 - Delete Account
 - Disable Account

- Enable Account
- Creating Groups
- Update Groups
- Delete Groups
- Disable Groups
- Enable Groups
- Bulk Operations will need to adhere to additional constraints:
 - Multiple operations (e.g. all the same or mixed) within the same request
 - Maximum of 10K operations per request
 - Requests can be made through an uploaded file extract
 - File extract cannot be greater than 2GB in size
- **Non-functional Requirements**
 - All user management operations should be completed within 5 seconds of execution
 - Bulk operations should be completed within 60 seconds
 - System failures from this feature must not result in the system going offline
- **Preconditions:**
 1. User must have an active authenticated session
 2. User must be on user management view
 3. User must be a system administrator
- **Successful Postcondition**
 1. User is able to perform any single UM operation within 5 seconds upon invocation. A system message displays “UM operation was successful”
 2. User is able to perform less than 10K UM operations in bulk within 60 seconds. A system message displays “Bulk UM operation was successful”
 3. User is able to perform 10K UM operations in bulk within 60 seconds. A system message displays “Bulk UM operation was successful”
- **Failure Scenarios**
 1. Single UM operation takes longer than 5 seconds
 2. Bulk UM operations takes longer than 60 seconds
 3. Single UM operation completes within 5 seconds, but no system message is shown or inaccurate system message is shown
 4. Bulk UM operations completes within 60 seconds, but no system message is shown or inaccurate system message is shown

5. Single UM operation completes within 5 seconds, with system message “UM operation was successful” shown, but latest data is not written to data store
 6. Bulk UM operations completes within 60 seconds, with system message “Bulk UM operation was successful” shown, but latest data is not written to data store
 7. 10K Bulk UM operations completes takes longer than 60 seconds
 8. 10K Bulk UM operations completes within 60s seconds, but no system message is shown or inaccurate system message is shown
 9. 10K Bulk UM operations completes within 60 seconds, with system message “Bulk UM operation was successful” shown, but latest data is not written to data store
- **Failure Handling:**
 1. User administrators will be notified in the event of any internal failures related to UM operations
 2. Any internal failures related to UM operations will be recorded via a system failure log (see Logging)

User Access Control

- **Description:** This feature allows members to control the read and edit permissions for each space. This feature will implement 3 different levels of permission for read and edit access to certain features.
- **Functional Requirements**
 - The system successfully allows users to access/edit data within their permission level.
 - The system successfully bars users from accessing/editing data above their permission level.
 - To add users to a space, a member of the shared space can generate a link to join the space. This link can be shared with members that want to join. This link expires after 24 hours or after it is used (whichever happens first).
 - To remove users from a group, members of a space can select the user they want to remove from the member list.
 - This user will no longer have read or edit access to the space, and they will no longer receive notifications from that space.
 - Members that are removed will not receive a notification that they have been removed from the space.

- All changes to user permissions and members of a space will be logged to a database in the form of user activity or errors (if any). Any errors in generating links will also be logged.
- **Non-functional Requirements**
 - When a user is added and/or removed from the space, the database should update within 3 seconds.
 - Generated join links will be sent via email within 5 seconds.
 - System administrators will be notified in the event a terminated user accesses/edits a space after removal.
- **Preconditions:**
 1. The user must have an active authenticated session (logged in).
 2. The user must have access to the shared space they want to edit.
- **Successful Postcondition:**
 1. All elements of the user's permission level are visible and editable
 2. Elements not included in the user's permission level or not accessible in the application, and therefore, are not editable.
- **Failure Scenarios:**
 1. System fails to withhold content/external user information from user
 2. Unauthorized user has access to a space
- **Failure Handling:**
 1. If any successful postconditions fail, the affected users will be notified via email of the failure scenario.

System Observability

Analytics

- **Description:** This feature will be used by the Business/Development team to lead future work on WeCasa. This summary includes various metrics related to the usage of the app, and provides insight on what features to expand/condense, overall engagement, retention, customer service, and overall app growth.
- **Functional Requirements**
 - All of the following Key Performance Indicators (KPIs) are visible in the summary and have adjustable time frames for viewing (1 week, 1 month, 3 months, 6 months) unless specified otherwise.
 - *Logins Per Day*

- Span of 3 months
 - Trend chart
- *Registrations Per Day*
 - Span of 3 months
 - Trend Chart
- *DAU (Daily Active Users)*
 - How many users are active each day
- *Sessions Per User (SPU)*
 - Average number of sessions initiated by each user
 - Total time spent across sessions / Total number of sessions
- *Most Used Features*
 - Percentage of user interaction with each feature
 - User interaction includes viewing and clicking on elements in a feature
 - Data from interactions come from logs
- *Error Rate*
 - Number of requests that result in an error / total number of requests
 - Time frames: 12 hours, 1 day, 3 days, 1 week, 1 month
- *Retention Rate*
 - Percentage of users who still use WeCasa after a certain time period after registering an account
 - Time frames: 1 day, 3 days, 1 week, 2 weeks, and 1 month
- *Customer service ratings*
 - Satisfaction scores from WeCasa customer service follow-up emails (see User Feedback)
- *User ratings from reviews*
 - Ratings (1-5) from users who submitted feedback through the User Feedback feature
- Dashboard is view-only and is meant for staff to take insight on app usage.
- Dashboard gathers aggregated data every 60 seconds and automatically updates all charts.
 - Metrics are displayed either graphically or with charts.
 - Certain data can be shown with specific time frames.
- **Non-functional Requirements**

- View of the Analytics dashboard must be loaded within 15 seconds of navigation.
- Edits (CRUD) to the analytics database should register within 3 seconds.
- Analytics database queries should return a result within 3 seconds.
- Data will be displayed in the charts within 3 seconds.
 - Includes chart updates after new time frame selections.
- System failures from this feature must not result in the system going offline.
- **Preconditions:**
 1. Users must have an active authenticated session.
 2. Users must have valid credentials to access WeCasa staff analytics dashboard (they must be a System Administrator)
 3. Users must be on the Analytics Dashboard view.
 4. The database must be active.
 5. The database must be accessible by the system.
- **Successful Postcondition**
 1. Dashboard is viewable by system administrator.
 2. The charts load within 15 seconds.
 3. All metrics automatically refresh every 60 seconds.
- **Failure Scenarios**
 1. Analytics are not being saved into the database.
 2. Analytics being written to the database are not persisted within 3 seconds
 3. Retrieving analytics from the database takes longer than 3 seconds.
 4. Dashboard metrics do not load within 3 seconds.
 5. Dashboard feature is not visible to system administrators and/or it is visible to generic users.
- **Failure Handling**
 1. Failed attempts to save analytics will be logged by the system. After 3 subsequent failed attempts, a data store debug error will be logged to the system.
 2. A server warning error will be logged to the system.
 3. A data store warning will be logged to the system.
 4. A view warning will be logged to the system.
 5. A business error will be logged to the system.

Logging

- **Description:** This feature allows the system to log normal user operations, system failures and exceptions into a secure database that only system administrators can access.
- **Functional Requirements**
 - The system successfully writes logs to the database.
 - The system administrator will be notified after 150 error logs.
 - Only the system administrator has permission to edit (CRUD) logs.
 - Logging occurs:
 - After all requests to the server.
 - When users login and logout unsuccessfully (wrong login information).
 - During events that can potentially become an error (exceed time limits, databases nearing capacity).
 - After every error condition (failed API calls, internal errors).
 - Log Archive will consist of:
 - Log Category/Type
 - View - These logs capture information related to the user interface and user interactions within the application.
 - These logs can be used for tracking user behavior, optimizing the user experience, and identifying usability issues
 - Ex. which buttons were clicked, and other user activity.
 - Business - These logs capture information related to the business logic of the application.
 - These logs can be used for debugging, troubleshooting, and auditing the application's business logic.
 - Ex. which business rules were triggered, which workflows were executed, and other application-specific details.
 - Server - These logs capture information related to the server-side operations of the application.
 - These logs can be used for monitoring the performance and health of the server, identifying

errors or problems within the server, and tracking the usage of server resources.

- Ex. which requests were received, which responses were sent, and other server-related activity.
- Data - These logs capture information related to the data used by the application.
 - These logs can be used for auditing, debugging, and troubleshooting data-related issues within the application.
 - Ex. which data was input, which data was output, and which transformations were performed on the data.
- Data Store - These logs capture information related to the storage and retrieval of data within a database or other data storage system..
 - These logs can be used for monitoring the performance and health of the data store, identifying errors or problems with the data storage system, and tracking changes to the data over time.
 - Ex. which queries were run, and other data store-related activity
- Log Level
 - Info - for tracking flow of system
 - Debug - for tracking key information crucial to maintainers of the system
 - Warning - for tracking events that may lead to system failures
 - Error - for tracking system errors
- User information (username, device).
- Group information (group name).
- User Operation/Activity (features that the user interacts with or requests data from).
- Timestamps (UTC timestamp that the user interacted with that activity).
- Inactivity frame (time between user activity).
- **Non-functional Requirements**

- Logs should write to the database within 3 seconds. If a log fails to show up in the database, it will try again in an hour. This will be attempted 2 more times.
- Database queries should return a result within 3 seconds.
- Logs will be automatically deleted after 2 years.
- The system administrator will be notified after 150 error logs within 5 seconds.
- Edits (CRUD) to the log archive should register within 3 seconds.
- Logs will not include PII (Personal identifiable Information).
- **Preconditions:**
 1. A secure database should exist for logging purposes only.
 2. The database must be persistent and active.
- **Successful Postcondition**
 1. All Logging entries must be present. No logs should be missing.
 2. Logging entries being saved to the data store must be accurate.
 3. All queries to the database should return all logs within the specified fields.
- **Failure Scenarios:**
 1. Logs are missing from the database.
 2. Logs contain missing or incorrect information.
 3. Logs aren't deleted after 2 years.
 4. The log cannot be queried or returned results are wrong.
- **Failure Handling:**
 1. Failed attempts to log to the system. After 3 subsequent failed attempts, the system will attempt to log a data warning error and prompt an email alert to the WeCasa administrators.
 2. A data debug error will be logged to the system.
 3. A data store error will be logged to the system.
 4. A server warning will be logged to the system.

Archiving

- **Description:** An internal mechanism for offloading log entries to preserve system resources
- **Functional Requirements:**
 - Archival process must execute every 00:00:00 AM (local time) on 1st of the month, offloading any log entries older than 30 days
 - Prior to archiving, log entries must be consolidated and compressed

- Offloaded log entries must be saved to a separate persistent data store location
- Offloaded entries will be subsequently removed from the system after successful archival
- **Non-Functional Requirements:**
 - Archival process must complete within 60 seconds upon invocation
 - System failures from this feature must not result in the system going offline
- **Preconditions:**
 1. Persistent data store must be active
 2. Persistent data store must be accessible by the system
 3. Archival destination location must have storage capacity
- **Successful Postcondition:**
 1. Archival process executes at 00:00:00 AM (local time) on the 1st of the month.
 2. All log entries older than 30 days are consolidated, compressed and relocated to another persistent data store location.
 3. All archived logs are removed from the system.
 4. The entire archival process completes within 60 seconds upon invocation.
- **Failure Scenarios:**
 1. Archival process did not start at 00:00:00 AM
 2. Archival process started at 00:00:00 AM, but not local time
 3. Archival process started at 00:00:00 AM (local time), but not on the 1st of the month
 4. 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
 5. 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
 6. 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.
 7. 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.

8. 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
 9. 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 is relocated to another location, but archived logs are not removed from the system.
 10. Archival process took longer than 60 seconds to complete upon invocation.
- **Failure Handling**
 1. Any internal failures related to the archival process will be recorded via a system failure log (see Failure Handling)

User Feedback

- **Description:** This feature allows users to communicate issues or positive feedback with the WeCasa team.
- **Functional Requirements**
 - An option to submit feedback is visible to the user.
 - When prompted, a form will appear where the user can specify the feedback type (issue or review) and type a message and give a rating (1-5 with intervals of 1/2).
 - Responses from users are sent to the WeCasa team's email.
 - All responses labeled as issues will be followed up within one week asking users how their customer service experience was.
 - Feedback submission will be logged to a database in the form of activity and/or errors (if any).
 - Satisfaction scores (1-5 with intervals of 1/2) obtained through follow up emails are used for monitoring and assessing WeCasa customer service.
- **Non-functional Requirements**
 - The popup window appears within 2 seconds.
 - The message written by the user is no more than 200 characters in length.
 - Once the user selects "submit", the message should indicate a successful/unsuccessful submission within 3 seconds.
 - Response to user's feedback appears in the user's inbox within 1 minute.

- **Required Input:**
 - The user must navigate and select the “feedback” option
 - The user must provide a typed message
 - Valid messages will consist of the following:
 - Must be at minimum 8 characters
 - A-z, A-Z
 - 0-9
 - !@#\$%^&*()
- **Successful Postcondition**
 1. Once the feedback is submitted, a note will appear indicating that the feedback was submitted successfully or unsuccessfully and the feedback window closes.
 2. The feedback is sent to the WeCasa team email and the message appears in full.
 3. Replies to the user’s feedback message are sent to the email associated with the user’s WeCasa account.
- **Failure Scenarios**
 1. The user abandons the form and exits.
 2. User feedback does not appear in the WeCasa team email.
 3. Response to user feedback does not appear in the user’s inbox.
 4. The feedback window does not restrict the user from typing a message greater than 200 characters.
 5. The user’s feedback is not logged into the database.
- **Failure Handling**
 1. Authentication process does not complete. User is expected to resubmit their ticket.
 2. A server warning will be logged by the system prompting the administration team to manually reach out to the user regarding User Feedback resubmission.
 3. A server warning will be logged by the system prompting the administration team to manually reach out to the user regarding User Feedback response.
 4. A view error will be logged by the system.
 5. A data warning will be logged by the system.

App Specific Features

App Settings

- **Description:** A system for users to edit the settings of the WeCasa app to fit their specific needs
- **Functional Requirements:**
 - Users have access to toggle on/off the following settings
 - Mobile Notifications (via SMS)
 - Email Notifications
- **Non-Functional Requirements:**
 - Changes to App Settings Toggles (On/Off) will persist to the database within 5 seconds.
 - The popup window appears after 2 seconds of saving changes.
- **Preconditions:**
 1. The user must have an active authenticated session (logged in)
- **Successful Postconditions:**
 1. The user successfully toggles mobile/email notifications On/Off
 2. The user receives app notifications via mobile/email if toggled 'On'.
 3. The user does not receive any app notifications via mobile/email if toggled 'Off'.
 4. Upon saving account settings changes, the user is prompted with a system confirmation message.
- **Failure Scenarios:**
 1. Changes to app settings are not persisted to the database within 5 seconds
 2. The user does not receive app notifications via mobile/email if toggled 'On'.
 3. The user receives app notifications via mobile/email if toggled 'Off'.
 4. System confirmation message is not prompted to the user within 2 seconds.
- **Failure Handling:**
 1. Server warning error is logged by the system.
 2. User is prompted to submit a User Feedback ticket.
 3. User is prompted to submit a User Feedback ticket.
 4. Business error is logged by the system

Group Management

- **Description:** A system for users to create new groups and manage existing group spaces, along with actions pertaining to inviting/deleting members.
- **Functional Requirements:**
 - Individual groups will have a user limit of 10 members
 - Users will be allowed to join a maximum of 3 groups at any given time - A dropdown navigation bar will be implemented to cycle between assigned groups
 - The user who creates a new group will be automatically assigned the role of 'Group Owner' - Within Group Management Settings, this role can be transferred to another group member.
 - A group owner must be assigned to a group at all times
 - In the event that a group owner's account is deleted/disabled, a group member will be automatically assigned to the role of 'Group Owner'
 - Group Owners will have access to the following the following settings within the Group Management
 - Editing the Group Name
 - Valid Input: a-z, A-Z, 0-9, 60 characters limit
 - Editing the Group Photo
 - Inviting/Deleting group members (see User Access Control)
 - A confirmation message will be prompted to the user prior to deletion of a group member
 - Toggling Group Features On/Off
 - Disabling/Deleting the Group
 - A confirmation message will be prompted to the user prior to deletion of a group
 - Within Group Management Settings, an invite link can be generated to add new group members (see User Access Control)
 - Generated invite links can be sent directly to users via email, or sent manually by users after being generated
 - Invite links will redirect unauthenticated users to a sign in page or prompt new users to create an account
 - The following group features can be toggled On/Off - when features are toggled Off, they will be hidden from the group space for all users
 - Budget Sharing Bar/Bill List
 - Bulletin Board

- Chore List
 - Grocery List
 - Calendar
 - Circular Progress Bar
- Users who are not assigned to the 'Group Owner' role will only have view access for the following settings:
 - Group Name
 - Group Photo
 - Group Members
 - Features Toggled 'On'
- Within Group Management Settings, any user has the option to leave an assigned group
 - Users assigned to the 'Group Owner' role will be prompted to transfer the role over to another user before they are allowed access to leave the group
 - A confirmation message will be prompted to the user prior to leaving a group
- **Non-functional Requirements:**
 - Cycling between different groups will update within 10 seconds
 - 'Group Owner' role transfers will update within 5 seconds
 - Invite links will be generated within 5 seconds
 - Edits to the Group Name/Photo will update within 5 seconds
 - Toggling group features On/Off will update within 5 seconds
- **Preconditions:**
 1. The user must have an active authenticated session (logged in)
 2. The user must be assigned to the group in order to view Group Management Settings
 3. The user must be assigned the 'Group Owner' role to edit Group Management Settings
- **Successful Postcondition:**
 1. 'Group Owner' is able to perform any Group Management operations within 5 seconds upon invocation
- **Failure Scenarios:**
 1. Group Management operations take longer than 5 seconds
 2. Users that are not assigned the 'Group Owner' role are able to perform Group Management operations
 3. 'Group Owner' role is not transferred to another user

- **Failure Handling:**

1. Server warning error is logged by the system.
2. Business warning error is logged by the system.
3. Business warning error is logged by the system, prompting administrators to manually assign a new Group Owner.

Budget Sharing Bar/Bill List

- **Description:** A budgeting system that allows roommates to keep track of shared expenses and balances.
- **Functional Requirements**
 - The budget bar functionalities:
 - Budget values have no upper-bound limit, however, progression towards the budget may not exceed the defined budget resulting in a negative budget readout.
 - The bill list is visible when the user interacts with the user icons.
 - The number of sections are determined by the number of users in the group that have spent money that month. Each section will be color coordinated with predefined colors.
 - The Bill List will show the following information:
 - The date
 - The bill name
 - The bill reporter
 - The bill amount
 - The bill status - 'UNPAID', 'PAID'
 - A receipt icon (if the user uploaded a receipt)
 - The Bill List (Current) enables users to:
 - Add a bill by prompting the user to:
 - (Required User Input) Inputting the name of bill
 - Valid Input: a-z, A-Z, 0-9, 60 characters limit
 - (Optional User Input) Input a description
 - Valid Input: a-z, A-Z, 0-9, 2000 characters limit
 - (Required User Input) Inputting how much is the bill
 - Based on U.S. monetary system (decimal system)
 - Valid Input: Floating point values only (\$X.XX)
 - (Optional User Input) Checking a box if the bill is to be repeated in the next period
 - Valid input: Boolean value only

- (Optional User Input) Select the members to split the bill with. Bill will be split equally between all members.
 - Valid input: Boolean value only
 - (Optional User Input) Upload a receipt.
 - Valid Input: .PNG, .JPEG, .HEIF
 - Update the status of payment.
 - View uploaded receipts.
 - View information related to uploaded bills. Pre-existing bills can only be edited by the user who initially reported the bill.
- The Bill List (History):
 - Shows and allows users to restore deleted bills within 24 hours
 - Shows and allows users to restore 'PAID' bills from the previous month
- Only users with an active authentication session and access to the space can utilize this feature.
- Changes to the budget sharing bar and adding bills to the transaction list will be logged to a database in the form of activity and/or errors (if any).
- A confirmation message will be prompted to the user prior to the deletion of a bill.
- **Non-functional Requirements**
 - When user wants to add a bill, a form appears within 3 seconds
 - When the user submits the form, the budget bar and the transaction list should update within 3 seconds.
 - The information should be recorded in the bills database within 5 seconds.
 - Deleted bills can only be restored within 24 hours of deletion, after which they will be permanently deleted from the system.
- **Preconditions**
 1. The user must have an active authenticated session (logged in).
 2. The user must have access to the shared space they want to edit.
- **Successful Postcondition**
 1. All transactions reflect in the bar and the transactions list accurately and within the maximum allotted time.
 2. All submitted information should be recorded in the bills database.
- **Failure Scenarios**
 1. The form doesn't appear when requested.

2. The user abandons the form and exits.
 3. The budget bar doesn't update after a bill is submitted.
 4. The transaction list doesn't update after a bill is submitted
 5. The budget bar doesn't refresh on the 1st of every month.
- **Failure Handling:**
 1. A view error is logged by the system and the user is asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback).
 2. If the user exits the form, data will not be recorded in the database.
 3. If the transaction list is not successfully updated, the user will not be notified but the error will be logged.

Bulletin Board

- **Description**
 - A space for roommates to pin common information (e.g., Wi-Fi name and password).
- **Functional Requirements**
 - All members of the group can edit, add or delete sticky notes on the board.
 - Users can add a sticky note on the bulletin board
 - Users can also delete sticky notes
 - The app will ask for confirmation before deleting the sticky note
 - Users will be able to undo the latest note deletion
 - The user will be able to add the following to a sticky note:
 - (Optional User Input) Text
 - Valid input: String limited to 500 characters
 - (Optional User Input) Photos
 - Valid input: Compatible image file types: .PNG, .JPEG, .HEIF
 - Only users with an active authentication session and access to the space can utilize this feature.
 - Changes to the bulletin board will be logged to a database in the form of activity and/or errors (if any).
 - Deleted bulletin points can be reverted, restoring them to their original place on the bulletin board.
 - A confirmation message will be prompted to the user prior to the deletion of a bulletin point.

- **Non-functional Requirements**
 - Bulletin Board view will load within 3 seconds
 - New bullet points will buffer within 3 seconds
 - Deletion of bullet points will buffer within 3 seconds
 - Reverting a deleted bullet point will buffer within 3 seconds
 - Deleted bulletin points can only be restored within 24 hours of deletion, after which they will be permanently deleted from the system.
- **Preconditions**
 1. The user must have an active authenticated session (logged in).
 2. The user must have access to the shared space they want to edit
- **Successful Postconditions**
 1. The Bulletin Board is viewable.
 2. The user can edit the contents of the board.
 3. The board can hold text and images.
- **Failure Scenarios**
 1. The user exceeds the character limit.
 2. The user attempts to add invalid image type.
- **Failure Handling:**
 1. The user is asked to stay within character limit.
 2. The user is asked to only upload valid image types.

Calendar

- **Description:** A visual representation of scheduled events. This is especially handy for trash pickup day, rent due dates and appliance maintenance dates.
- **Functional Requirements**
 - The user should be able to navigate to the Calendar view for a particular group
 - The Calendar will display the current month.
 - Calendar view can be changed from month, week, and day.
 - The current day will be outlined and its tasks will be displayed chronologically on a sidebar by default
 - Events on a particular date can be shown.
 - Upon creating a new event, the user will be prompted to:
 - (Required User Input) Input a name for the event
 - Valid input: a-z, A-Z, 0-9, 60 characters limit
 - (Optional User Input) Assign an event a color.
 - Valid input: Blue, Orange, Yellow, Red, Green

- (Optional User Input) Input a time of occurrence for the event
 - Valid input: Integer values only
 - (Optional User Input) Check if event will be repeated daily, weekly, annually
 - Valid input: Categorical array
 - (Optional User Input) Check if the event will be private or public (defaults to public)
 - Valid input: Boolean value only
 - (Optional User Input) Input a time before the event for reminders via email
 - Valid input: Integer values only, must be larger than current date
 - (Optional User Input) Input description for event
 - Valid input: String limited to 250 characters
- Events can be created, edited, and deleted
- Changes to the calendar and adding events will be logged to a database in the form of activity and/or errors (if any).
- Deleted calendar events can be reverted, restoring them to their original place on the bulletin board.
- A confirmation message will be prompted to the user prior to the deletion of a calendar event.
- **Non-functional Requirements**
 - The calendar view should load within a maximum of 3 seconds
 - New calendar events will buffer within 3 seconds
 - Deletion of calendar events will buffer within 3 seconds
 - Reverting a deleted calendar event will buffer within 3 seconds
 - Deleted calendar events can only be restored within 24 hours of deletion, after which they will be permanently deleted from the system.
- **Preconditions**
 1. The user must be logged in to their own account.
 2. The user must be in the Calendar view.
- **Success Postconditions**
 1. If a user creates/edits an event, an event update should be seen by allowed parties.
 2. The user receives a reminder in accordance with their reminder time selection.
- **Failure Scenarios**

1. The calendar view does not load.
 2. The user is unable to create/update an event.
 3. The user does not receive a message notification reminder.
 4. The event is updated/created, however other users cannot view.
 5. The user abruptly leaves the system while creating/updating events.
- **Failure Handling**
 1. A view error is logged by the system and the user is asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback).
 2. A view error is logged by the system and the user is asked to reload the page and try again. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback).
 3. A server warning is logged by the system. User is prompted to submit a User Feedback ticket. (See App Settings/User Feedback)
 4. A view warning is logged by the system
 5. Calendar event process does not complete. User is expected to resubmit event.

Chore List

- **Description:** A task board to split up chores amongst housemates.
- **Functional Requirements**
 - The chore list will display as a weekly list of chores separated by dates. This list will start on Monday.
 - The user can select between Current To-Do's and To-Do History views on the page for their viewing options.
 - The user can enter a new chore into the list with the following required inputs:
 - (Required User Input) Input chore name
 - Valid input: a-z, A-Z, 0-9, 60 character limit
 - (Required User Input) Input time to reset
 - Valid input: Integer values only, must be larger than current date
 - (Optional User Input) Input chore notes
 - Valid input: String limited to 250 characters
 - (Optional User Input) Input assignments to a specific person
 - Valid input: Array value only
 - (Optional User Input) Check if the chore is repeated

- Valid input: Boolean value only
 - The user can mark a task as completed.
 - If a chore isn't done by the due date, it is added to the assigned user's next chore list.
 - Chores are unassigned if someone is uninvited from the group.
 - Every week, an even number of tasks are randomly split between each user.
 - Changes to the chore list will be logged to a database in the form of activity and/or errors (if any).
- **Non-functional Requirements**
 - Chore List View will be loaded within 3 seconds.
 - New chores will buffer within 2 seconds.
 - Deletion of chores will buffer within 2 seconds.
 - Completion of a chore will buffer within 2 seconds.
 - Reassigning a chore will buffer within 2 seconds.
 - Switching to a different week of chores will buffer within 2 seconds.
 - Weekly number of assigned tasks will change depending on the current number of roommates.
- **Preconditions**
 1. The user must be logged in to their own account.
 2. The user must be at the Chore List View.
- **Successful Postconditions**
 1. Interacting with the 'TO-DO' section will display a list of current chores.
 2. Interacting with the 'History' section will display a list of completed chores.
 3. When a user checks a task as completed:
 - The task will be removed from TO-DO and added to History
 - The color bar will reflect its status and change from red to green (green for complete, red for incomplete)
- **Failure Scenarios**
 1. The user abruptly exits from saving a new chore.
 2. The user attempts to leave required fields empty.
 3. The user exceeds the maximum character limit for a new chore name.
- **Failure Handling**
 1. Chore list process does not complete. User is expected to resubmit chore.
 2. The user is notified to input all required fields and try again.

3. The user is notified to try again with limited characters.

Grocery List

- **Description:** A shared list designed to efficiently complete grocery shopping.
- **Functional Requirements**
 - A user is able to navigate to the Grocery List view for a particular group
 - A user has the ability to add grocery item with the following inputs:
 - (Required User Input) Input grocery item name
 - Valid input: a-z, A-Z, 0-9, 60 character limit
 - (Optional User Input) Input Optional notes
 - Valid input: String limited to 250 characters
 - (Optional User Input) Input assignments to a specific person
 - Valid input: Array value only
 - The user is able to delete an item on the list that has already been purchased
 - Clearing the list will clear all crossed and uncrossed items
 - Items can be deleted from the list
 - The list instantly syncs up changes across all user accounts.
 - Changes to the grocery list will be logged to a database in the form of activity and/or errors (if any).
- **Non-functional Requirements**
 - Grocery List View will be loaded within 3 seconds.
 - New grocery items will buffer within 2 seconds.
 - Deletion of grocery items will buffer within 2 seconds.
 - Purchase of grocery items will buffer within 2 seconds.
- **Preconditions**
 1. The user must be logged in to their own account.
 2. The user must be at the Grocery List View.
- **Success Postconditions**
 1. The user is able to add grocery item name
 2. The user is able to delete a grocery item
 - The deleted item is visible in the list, but is crossed out
- **Failure Scenarios**
 1. The user abruptly exits from saving a new grocery item.
 2. The user attempts to leave required fields empty.
 3. The user exceeds the maximum character limit for the new grocery item name.

- **Failure Handling**

1. Grocery list process does not complete. User is expected to resubmit grocery item.
2. The user is notified to input all required fields and try again.
3. The user is notified to try again with limited characters.

Reminders

- **Description:** An easy way to get notified of scheduled events labeled on the calendar.

- **Functional Requirements**

- Reminder settings are configurable upon creation/editing of any of the aforementioned features:
 - Budget Sharing Bar/Bills
 - Bulletin Posts
 - Calendar Events
 - Chores
- Reminders will be sent via email.
- Reminders can be configured for the following time intervals:
 - One Week in Advance
 - 1 Day in Advance
 - 30 Minutes in Advance
- Reminder settings can be configured for recurring events:
 - Daily
 - Weekly
 - Bi-weekly
 - Monthly
 - Annually
- Adding or changing reminders will be logged to a database in the form of activity and/or errors (if any).

- **Non-functional Requirements**

- Saving configuration for reminders must buffer within 3 seconds.
- Reminders must be sent promptly according to the user's scheduled time settings.

- **Preconditions**

1. To configure reminders, users must be logged into their own accounts.
2. To receive reminders, users do not have to be logged in.

3. The user must have access to the space to receive reminders for that space.
 4. The user must have configured reminders to true when creating/editing events.
 5. Reminders will be configured to true as default.
- **Success Postconditions**
 1. The user receives an email at the scheduled time they configured the reminder for.
 - **Failure Scenarios**
 1. Reminder has not been sent via email at the scheduled time it has been configured for.
 - **Failure Handling**
 1. A server warning is logged by the system. User is prompted to submit a User Feedback ticket. (See App Settings/User Feedback)

Nudging

- **Description**
 - A friendly reminder sent to roommates who haven't completed their assigned tasks in time by another user.
- **Functional Requirements**
 - The nudging option will be available on all task cards other than the current user's.
 - Users can 'nudge' a roommate, sending the assignee a friendly notification to complete that task.
 - The user will have predefined messages to send to the assignee.
 - Assignee will be notified through email and once they login.
 - Only users with an active authentication session and access to the space can utilize this feature.
 - Users do not have to be logged in to receive a Nudge.
 - Sending a nudge will be logged to a database in the form of activity and/or errors (if any).
 - Users can only nudge one assignee per day.
- **Non-functional Requirements**
 - Saving the nudge to send should buffer within 3 seconds.
 - The assignee should receive their nudge within a minute of being sent.
- **Preconditions**

1. To send a nudge, the user must be logged in and have access to a space.
 2. Users do not have to be logged in to receive a nudge.
 3. The user must be in the Chore List view.
 4. The task being nudged must be incomplete.
- **Success Postconditions**
 1. The user should have a confirmation that the nudge is sent.
 2. Assignee should have an email and be notified of the nudge when they log in.
 - **Failure Scenarios**
 1. Assignee is not notified of a nudge via email or when they log in.
 2. Nudge option is disabled even though the task is incomplete.
 - **Failure Handling**
 1. Business error is logged by the database and the user is asked to reload the page and try again or attempt account recovery. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)
 2. Error is logged by the database and the user is asked to reload the page and try again or attempt account recovery. For further assistance, user is prompted to submit a User Feedback ticket. (see User Feedback)

Incomplete Task Summary

- **Description:** A summary of incomplete tasks is automatically generated and sent to all roommates via text/email notification.
- **Functional Requirements**
 - Comprehensive list of users with incomplete tasks from the past week (Monday - Sunday) is compiled.
 - Comprehensive list of incomplete tasks from the past week (Monday - Sunday) is compiled.
 - Only users with an active authentication session and access to the space can receive this notification and/or edit this feature.
 - Users do not have to be logged in to receive the Incomplete Task Summary.
 - All users within a space should receive the Incomplete Task Summary.
 - This feature will be logged to a database in the form of errors (if any).
- **Non-functional Requirements**

- Notification will be sent at 12AM (UTC) every Sunday, any tasks left incomplete will appear in the summary.
- Notification must appear within 3 seconds of its scheduled time.
- **Preconditions**
 1. The user must have access to a space.
 2. Tasks from the shared chore list must be left incomplete by the end of the week.
- **Successful Postconditions**
 1. The notification is sent successfully.
 2. All incomplete tasks appear in summary.
- **Failure Scenarios**
 1. Notification fails to be sent via email at the scheduled time it has been configured for.
- **Failure Handling**
 1. A server warning is logged by the system. User is prompted to submit a User Feedback ticket. (See App Settings/User Feedback)

Circular Progress Bar

- **Description:** This feature reflects the percentage of assigned chores a user completes onto a progress bar.
- **Functional Requirements**
 - If a user doesn't complete their tasks by the due date, it will negatively impact their scores on this bar:
$$\# \text{ of incomplete tasks} / (\# \text{ of incomplete tasks} + \# \text{ of complete tasks}).$$
 - If the user completes their tasks, the bar will progress:
$$\# \text{ of complete tasks} / (\# \text{ of incomplete tasks} + \# \text{ of complete tasks}).$$
 - The scores will be updated whenever a task is complete or a task is overdue.
 - The bar will refresh on the first of every month.
 - Changes to the tracking bar will be logged to a database in the form of activity and/or errors (if any).
- **Non-functional Requirements**
 - The bar updates within 3 seconds when a user marks a chore as complete or if a chore becomes overdue.
- **Preconditions**
 1. The user must be logged in to their own account.
 2. The user must be in the Circular Progress Bar view.

- **Successful Postconditions**

1. The bar recalculates and updates when a user marks a chore as complete or if a chore becomes overdue.
2. The bar refreshes visually every month.
3. The status of the bar and any changes made to the bar are successfully logged into the appropriate database.

- **Failure Scenarios**

1. The bar doesn't update visually when a chore is marked complete, or a chore becomes overdue.
2. The bar doesn't refresh visually every month.
3. The bar status and changes to the bar are not logged to the appropriate database.

- **Failure Handling**

1. View warning is logged by the system.
2. View warning is logged by the system.
3. Data store error is logged by the system.

Photo/Documentation Uploads

- **Description:** Users can upload files such as lease documentation, rental agreements, property photos, receipts, etc.

- **Functional Requirements**

- Users have the ability to upload a new file from their device
- Users are able to configure valid file types:
 - Photo File Type: PNG, JPEG, and GIF
 - Document File Type: PDF, DOC, DOCX, HTML, TXT
- Users are able to configure valid file size limit
 - 10 MB File Size Limit
- Users are able to delete a file uploaded to the storage only if they uploaded the file.
- Files can be retrieved and previewed by all users.
- Files can be downloaded from storage to a user's device.
- File storage syncs with any changes across all user accounts.
- Only the user who uploaded the file has permission to delete it.
- Adding/removing a photo/document will be logged to a database in the form of activity and/or errors (if any).
- Deleted photos/documents can be reverted, restoring them to their original place in the storage.

- A confirmation message will be prompted to the user prior to the deletion of a photo/document.
- **Non-functional Requirements**
 - Storage will load within 10 seconds.
 - File previews will load within 10 seconds.
 - Files will be stored within the database 10 seconds after user upload.
 - Deleted photos/documents can only be restored within 24 hours of deletion, after which they will be permanently deleted from the system.
- **Preconditions**
 1. The user must be logged in to their account.
 2. The user must have access to a space.
 3. The user must be in the file upload view.
 4. Uploaded file does not exceed storage space limit/file upload limit
 - a. Storage Limit: 15 GB
 - b. File Upload Limit: 10 MB
 5. Uploaded file is of valid file type
 - a. Photo File Type: PNG, JPEG, and GIF
 - b. Document File Type: PDF, DOC, DOCX, HTML, TXT
- **Successful Postconditions**
 1. File is uploaded to storage.
 2. Files are accessible by any user within the group.
 3. When a user uploads a file, the activity is logged into a database.
- **Failure Scenarios**
 1. File cannot be uploaded due to invalid file type.
 2. File cannot be uploaded due to invalid file size.
 3. File cannot be uploaded due to insufficient storage space.
 4. File cannot be retrieved from database due to connection error.
 5. User does not have permission to access the file.
- **Failure Handling**
 1. File upload is aborted. User is prompted with an error message based on failure scenario.

Error Handling

System Failure

- **Description:** The following section details how system failure scenarios will be handled. The Failure Handling Error Table provides further details regarding failure handling for previously outlined Core Component/App Specific Features.
- **Functional Requirements:**
 - The user must be shown an error page.
 - The user must be notified of the reason behind the system failure, and a follow up action should be recommended through the error page.
- **Non-Functional Requirements:**
 - The error given to the user must be given in an user-friendly manner.
 - Does not use technical language, details system specifications, or communicate security risks
 - Provide an action for resolution
 - Contact WeCasa Support
 - The error page must be displayed within 3 seconds.
- **Preconditions:**
 1. User is logged in and is on any page of the website.
- **Successful Postconditions:**
 - The error page is displayed.
- **Failure Scenarios:**
 1. Database goes offline.
 2. Internet connection fails.
- **Failure Handling**
 1. User is notified through an error page that servers unexpectedly crashed and is asked to login later.
 2. User is notified through an error page that the internet connection is lost and is asked to reconnect to the internet.
 3. Users are notified through an error page that a connection error occurred and all entries are automatically aborted.

FAILURE HANDLING ERROR TABLE

Error ID	Scenario	Error Message	Log Category/Type	Log Level
[Security] Authentication				
AUTHN-1	The user abruptly exits from the authentication process	Previous authentication attempt failed. Please try again.	N/A	N/A
AUTHN-2	The user leaves any of the required fields empty.	Please fill out all required fields	N/A	N/A
AUTHN-3	The user inputs invalid username.	Invalid Input. Please enter valid input. Valid usernames must meet the character requirements Minimum of 8 characters a-z 0-9 (Optional) The following special characters: . - @	N/A	N/A
AUTHN-4	The user input invalid password.	Invalid Input. Please enter valid input. Minimum of 8 characters One of each of the following	N/A	N/A

		characters: a-z A-Z 0-9 .,@!-		
AUTHN-5	The user is not notified of invalid login attempt.	N/A	View	Error
AUTHN-6	The user is not notified of disabled account.	N/A	View	Error
AUTHN-7	Failed attempts are not logged by the system.	N/A	Business	Error
AUTHN-8	The user successfully inputs valid credentials but is still unable to log in.	N/A	Data	Error
[Security] Authorization				
AUTHZ-1	Unauthorized access attempts fail to be logged by the system	N/A	Business	Error
AUTHZ-2	User attempts to access protected functionalities/ data/views	Unauthorized Access	Business	Warning

	outside of their authorization scope			
AUTHZ-3	User attempts to modify protected data outside of their authorization scope.	Unauthorized Access to Data	Data	Warning
AUTHZ-4	Sensitive protected data that is not within a user's read scope is accessible via a view within their authorization scope.	Unauthorized Access to Data	Data	Warning
AUTHZ-5	Sensitive protected data that is not within a user's write scope is accessible via a view within their authorization scope.	Unauthorized Access to Data	Data	Warning
[Security] Logout				
OUT-1	The user abruptly exits from the logout request.	Previous logout attempt failed. Please try again.	N/A	N/A

OUT-2	The user is not logged out and the session remains active.	Logout failed. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	Business	Error
OUT-3	Confirmation message is not displayed to the user.	N/A	View	Error
OUT-4	The user is not redirected to the system's home page.	N/A	View	Error
OUT-5	Logout is not logged by system.	N/A	Data Store	Error
[User Administration] Account Creation/Registration				
REG-1	The user abandons the registration and exits	N/A	N/A	N/A
REG-2	User does not receive confirmation email	Email has not been confirmed, please check your inbox for a confirmation email. Did not receive a confirmation email? Resend	Business	Error

REG-3	The new user data is not logged into the database	N/A	Data Store	Error
[User Administration] Account Recovery				
AR-1	User provides an invalid username	Invalid Input. Please enter valid input. Valid usernames must meet the character requirements Minimum of 8 characters a-z 0-9 (Optional) The following special characters: . - @	N/A	N/A
AR-2	User provides an invalid OTP	Invalid Input. Please enter valid input. A valid OTP verification consisting of the following: 10 character length A-Z A-z 009 .-@	N/A	N/A
AR-3	OTP does not match	OTP does not match. Please re-enter OTP or request a	N/A	N/A

		new code.		
AR-4	An authorized system admin completes the account recovery for the user, but the user does not regain access to the system within 5 seconds	N/A	Business	Error
[User Administration] Account Deletion				
AD-1	The user chooses to delete their accounts and confirms action, however the system doesn't delete both personal identifiable date and user account data.	Account deletion error. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	Data / Data Store	Error / Debug
AD-2	Data is not permanently deleted from the system.	Account deletion error. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	Data / Data Store	Error / Debug
AD-3	A system	N/A	View	Error / Debug

	message confirming account deletion is not shown or if the wrong message is shown instead.			
AD-4	The user is not automatically redirected to the login view after account deletion.	N/A	View	Error
[User Administration] User Management				
UM-1	Single UM operation takes longer than 5 seconds	N/A	Data Store	Error
UM-2	Bulk UM operation takes longer than 60 seconds	N/A	Data Store	Error
UM-3	Single UM operation completes within 5 seconds, but no system message is shown or inaccurate system message is shown	N/A	View	Error

UM-4	Bulk UM operations completes within 60 seconds, but no system message is shown or inaccurate system message is shown	N/A	View	Error
UM-5	Single UM operation completes within 5 seconds, with system message “UM operation was successful” shown, but latest data is not written to data store	N/A	Data	Error
UM-6	Bulk UM operations completes within 60 seconds, with system message “Bulk UM operation was successful” shown, but latest data is not written to data store	N/A	Data	Error

UM-7	10K Bulk UM operations completes takes longer than 60 seconds	N/A	Data Store	Error
UM-8	10K Bulk UM operations completes within 60s seconds, but no system message is shown or inaccurate system message is shown	N/A	View	Error
UM-9	10K Bulk UM operations completes within 60 seconds, with system message “Bulk UM operation was successful” shown, but latest data is not written to data store	N/A	Data	Error
[User Administration] User Access Control				
UAC-1	System fails to withhold content/external user information from user	N/A	Business	Error

UAC-2	Unauthorized user has access to a space	N/A	Business	Warning
[System Observability] Analytics				
ANA-1	Analytics are not being saved into the database.	N/A	Data / Data Store	Error / Debug
ANA-2	Analytics being written to the database are not persisted within 3 seconds	N/A	Server	Warning
ANA-3	Retrieving analytics from the database takes longer than 3 seconds.	N/A	Server	Warning
ANA-4	Dashboard metrics do not load within 3 seconds.	N/A	View	Error
ANA-5	Dashboard feature is not visible to system administrators and/or it is visible to generic users.	N/A	Business	Warning
[System Observability] Logging				
LOG-1	Logs are	N/A	Data	Warning

	missing from the database.			
LOG-2	Logs contain missing or incorrect information.	N/A	Data	Debug
LOG-3	Logs aren't deleted after 2 years	N/A	Data	Error
LOG-4	The log cannot be queried or returned results are wrong	N/A	Server	Warning
[System Observability] Archiving				
ARC-1	Archival process did not start at 00:00:00 AM	N/A	Business	Error
ARC-2	Archival process started at 00:00:00 AM, but not local time	N/A	Business	Error
ARC-3	Archival process started at 00:00:00 AM (local time), but not on the 1st of the month	N/A	Business	Error
ARC-4	Archival process started at	N/A	Data	Error

	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			
ARC-5	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	N/A	Data	Error
ARC-6	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.	N/A	Data	Error
ARC-7	Archival process started at 00:00:00 AM (local time) on the 1st of the month. All log	N/A	Data	Error

	entries older than 30 days are consolidated, but are not compressed.			
ARC-8	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	N/A	Data	Error
ARC-9	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 is relocated to another location, but archived logs	N/A	Data	Error

	are not removed from the system.			
ARC-10	Archival process took longer than 60 seconds to complete upon invocation.	N/A	Business	Error
[System Observability] User Feedback				
UF-1	The user abandons the form and exits.	N/A	N/A	N/A
UF-2	User feedback does not appear in the WeCasa team email.	User Feedback submission failure.	Server	Warning
UF-3	Response to user feedback does not appear in the user's inbox.	Failure issuing User Feedback response. Please reach out to User manually.	Server	Warning
UF-4	The feedback window does not restrict the user from typing a message greater than 200 characters.	N/A	View	Error
UF-5	The user's feedback is not logged into the	N/A	Data	Warning

	database.			
[App Specific Features] App Settings				
AS-1	Changes to app settings are not persisted to the database within 5 seconds	N/A	Server	Warning
AS-2	The user does not receive app notifications via mobile/email if toggled 'On'.	Notifications have been toggled "On". Not receiving notifications? For further assistance, please submit a User Feedback ticket.	Business	Error
AS-3	The user receives app notifications via mobile/email if toggled 'Off'.	Notifications have been toggled "Off" Still receiving notifications? For further assistance, please submit a User Feedback ticket.	Business	Error
AS-4	System confirmation message is not prompted to the user	N/A	Business	Error

	within 2 seconds.			
[App Specific Features] Group Management				
GM-1	Group Management operations take longer than 5 seconds	N/A	Server	Warning
GM-2	Users that are not assigned the 'Group Owner' role are able to perform Group Management operations	N/A	Business	Warning
GM-3	'Group Owner' role is not transferred to another user	N/A	Business/Data	Warning
[App Specific Features] Budget Sharing Bar/Bill List				
BUD-1	The form doesn't appear when requested.	Error loading budget bar. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	View	Error
	The user abandons the form and exits.	N/A	N/A	N/A

BUD-2	The budget bar doesn't update after a bill is submitted.	N/A	View	Error
BUD-3	The transaction list doesn't update after a bill is submitted	N/A	View	Error
BUD-4	The budget bar doesn't refresh on the 1st of every month.	N/A	Business	Error
[App Specific Features] Bulletin Board				
BB-1	The user exceeds the character limit.	Invalid Input - Character Limit Exceeded.	N/A	N/A
BB-2	The user attempts to add invalid image type.	Invalid File Type.	N/A	N/A
[App Specific Features] Calendar				
CAL-1	The calendar view does not load.	Error loading calendar. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	View	Error

CAL-2	The user is unable to create/update an event.	Error loading calendar. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	Business	Error
CAL-3	The user does not receive a message notification reminder.	N/A	Server	Warning
CAL-4	The event is updated/created, however other users cannot view.	N/A	View	Warning
CAL-5	The user abruptly leaves the system while creating/updating events.	N/A	N/A	N/A
[App Specific Features] Chore List				
CL-1	The user abruptly exits from saving a new chore.	N/A	N/A	N/A
CL-2	The user attempts to leave required fields empty.	Invalid Input - Please fill out all required fields.	N/A	N/A
CL-3	The user	Invalid Input -	N/A	N/A

	exceeds the maximum character limit for a new chore name.	Maximum Character Limit Exceeded		
[App Specific Features] Grocery List				
GL-1	The user abruptly exits from saving a new grocery item.	N/A	N/A	N/A
GL-2	The user attempts to leave required fields empty.	Invalid Input - Please fill out all required fields.	Business	Warning
GL-3	The user exceeds the maximum character limit for the new grocery item name.	Invalid Input - Maximum Character Limit Exceeded	Business	Warning
[App Specific Features] Reminders				
REM-1	Reminder has not been sent via email at the scheduled time it has been configured for.	N/A	Server	Warning
[App Specific Features] Nudging				
NDG-1	Assignee is not notified of a nudge via email or when they log in.	Error loading nudge. Please reload the page and try again. For	Business	Error

		further assistance, please submit a User Feedback ticket.		
NDG-2	Nudge option is disabled even though the task is incomplete.	Error loading nudge. Please reload the page and try again. For further assistance, please submit a User Feedback ticket.	View	Error
[App Specific Features] Incomplete Task Summary				
ITS-1	Notification fails to be sent via email at the scheduled time it has been configured for.	N/A	Server	Error
[App Specific Features] Circular Progress Bar				
CPB-1	The bar doesn't update visually when a chore is marked complete, or a chore becomes overdue.	N/A	View	Warning
CPB-2	The bar	N/A	View	Warning

	doesn't refresh visually every month.			
CPB-3	The bar status and changes to the bar are not logged to the appropriate database.	N/A	Data Store	Error
[App Specific Features] Photo/Documentation Uploads				
UPL-1	File cannot be uploaded due to invalid file type.	Error: Invalid File Type	N/A	N/A
UPL-2	File cannot be uploaded due to invalid file size.	Error: Invalid File Size	N/A	N/A
UPL-3	File cannot be uploaded due to insufficient storage space.	Error: Insufficient Storage Space	N/A	N/A
UPL-4	File cannot be retrieved from database due to connection error.	Error: Connection lost! Please try again later.	N/A	N/A
UPL-5	User does not have permission to access the file.	Error: Permission denied.	N/A	N/A

Appendix A

Claims Checks

Claim	Description
The user is disabled.	This user has not activated their one-time password and does not have a profile. This user does not have access to any spaces or features.
The user has an active authentication session.	This user has created a profile and successfully logged in. They can view their home page and potentially any space that they have access to. This user can view and change their account settings.
The user has access to a space	This user can view and interact with the features of the space they have access to with an authenticated session. This user can add/remove members of the space they have access to. This user can receive notifications from this space (even if they are logged out).
The user is logged as a systems administrator.	This user has read and write access to all information of all generic user accounts. This user can view the analytics dashboard and has the ability to disable users. This user does not have write access to log directories (<i>Security Tips - Apache HTTP Server, 2022</i>).

Glossary

Term	Definition
Deleted bills	Persisted bills that have been deleted manually by the user.
KPI	Key Performance Indicator
Private information	Any user information that is only accessible and editable from that user. This includes the user's email and phone number.
Public information	Any user information that members in a shared space can view. This includes the user's profile icon and the user's name.
Referral Rate	Metric for how many invites get sent out for new users
Retention Rate	Percentage of users who continue engaging with the app over time (time intervals: 30 days, 7 days)
UAC	User Access Control

References

Jack. (2021, November 24). Does the ADA Require Mobile Websites and Apps to be

Accessible? Boia.org; Bureau of Internet Accessibility.

<https://www.boia.org/blog/does-the-ada-require-mobile-websites-and-apps-to-be-accessible>

Michael. (2018, October). The Basics and Importance of Color Contrast for Web

Accessibility. Boia.org; Bureau of Internet Accessibility.

<https://www.boia.org/blog/the-basics-and-importance-of-color-contrast-for-web-accessibility>

Most Popular Web Browsers in 2022 [Sep '22 Update] | Oberlo. (2022). Oberlo.com.

<https://www.oberlo.com/statistics/browser-market-share>.

Security Tips - Apache HTTP Server Version 2.4. (2022). Apache.org.

https://httpd.apache.org/docs/2.4/misc/security_tips.html

Vatanak Vong. (2022, October 17). Core Components Requirements. CSULB.

W3C Validation: What is it and why to use it? | LoginRadius Blog. (2020).

Loginradius.com.

<https://www.loginradius.com/blog/engineering/w3c-validation/>

Web Content Accessibility Guidelines (WCAG) 2.1. (2018, June 5). W3.org.

<https://www.w3.org/TR/WCAG21/>