4

Software Requirements Specification

for

CoHab

Version 1.0 approved

Prepared by Nieky Allen
Joe Hernandez
Christian Lanzer
Sarah Mehmedi
Shan-e-Ali H. Shah

COMP 330-001

March 1, 2016

Table of Contents

Table of Contents

Introduction

1.1 Purpose
1.2 Document Conventions
1.3 Intended Audience and Reading Suggestions
1.4 Product Scope
1.5 References

2. Overall Description

2.1 Product Perspective
2.2 Product Functions
2.3 User Classes and Characteristics
2.4 Operating Environment
2.5 Design and Implementation Constraints
2.6 User Documentation
2.7 Assumptions and Dependencies

3. External Interface Requirements

3.1 User Interfaces
3.2 Hardware Interfaces
3.3 Software Interfaces

4. System Features

5. Other Nonfunctional Requirements

5.1 Performance Requirements
5.2 Safety/Security Requirements
5.3 Software Quality Attributes

Appendix: Analysis Models

A.1 Use Case Diagram
A.2 UML Class Diagram

1. Introduction

1.1 Purpose

The purpose of this document is to provide the software requirement specification report for the CoHab mobile application. This document contains the scope, functional and non-functional requirements of the project, and the guidelines for the development team to design the system.

1.2 Document Conventions

The document will use Times New Roman font for all text. Section headings will be in 18pt **bold** font, and numerically ordered. Subsection headings will be in 14pt **bold** font and numerically designated as part of the section of which they are constituent. The bodies of sections will have 12 pt font, and will elaborate upon the heading or subheading, sometimes with the use of lists, bulleted, numbered, or otherwise sequenced.

1.3 Intended Audience and Reading Suggestions

1. Development Team: This document should serve as a guide for the development team to understand the software's requirements and the specifics of those requirements' implementations. The application's development team should read this document in its entirety to understand the guidelines for the product's development.
2. Clients: This document elaborates on user expectations of the application, software and hardware specifications, and provides a guide detailing the technical information about the system and the interaction of its constituent parts and dependencies. Clients are recommended to read section 2 in its entirety.

1.4 Product Scope

CoHab is a native iOS application. At its most basic level it serves as a personal organizer application that can share one pool of data and read/write access over multiple users. Its primary function is to serve as a small closed user network with a shared calendar among connected users for organization and scheduling purposes. Its secondary function is to act as a public notice board among local users within an area. The purpose of the primary function is to provide a convenient organization and communication system for cohabitant users, specifically in terms of their interdependent finances, and shared responsibilities. More explicitly, through the primary function we hope to provide a system for roommates within a single unit who will all use their own individual user access on a shared account to (1) track user-input bills, expenditures and activities on a calendar, (2) set user assignments to a bill/expenditure/activity on the calendar, (3) set status updates on a bill/expenditure/activity on the calendar, and (4) send notifications to other users within the account regarding a bill/expenditure/activity on the calendar. Through the secondary function we hope to provide a tool for increased in-cohabitation-unit, and inter-unit communication

by means of a closed network forum board service for in-cohabitation-unit users and an open network forum board service for inter-unit accounts in the area. Scope and features for software determined by: (1) development team initial product inception and design, (2) application design and feature assessment and suggestions by cohabitant potential users through development-team-created online survey, (3) development team conversations with three clients representative of the product's three major user classes (see section 2.3), and (4) use cases developed with potential user input.

I. Potential User #1: Maria Folvarska - mfolvarska12@gmail.com
   A. Lives in a house, the neighborhood is very quiet
   B. She would love to communicate with the community and get to know the people that live nearby
   C. She knows of people who have roommate problems and this app would totally help with resolving those problems (late payments/not doing chores)
   D. Mentioned having a way of confirming payments/tasks would be useful
   E. Wants a mobile app so it can be used by her personally because she has to share a laptop with her siblings

II. Potential User #2: Alex Hobt - ahobt@luc.edu
   A. Lives in an apartment with two roommates
   B. As a renter who is more of acquaintances than friends with her roommates, he liked the idea of a management system to communicate with his roommates in simple ways about responsibilities and expenses

III. Potential User #3: Zindy Marquez - zmarquez@luc.edu
   A. Lives in apartment-style, 5-person suite in a residence hall on a private college campus
   B. She immediately thought this was a very useful app to be developing
   C. From tense situations and conversations she's dealt with amongst her roommates, she felt a more simple and slightly formal way to interact with her roommates about serious matters would be very beneficial
   D. As an advertising/PR major, she also said he had marketing ideas and offered to help

1.5 References

[1] IEEE Std 1233, 1998 Edition, IEEE Guide for Developing System Requirements Speifications.
[2] Allen, N., Hernandez, J., Lanzer, C., Mehmedi, S., Shah, S. H., "Cohabitation Convenience." Survey, 29 February, 2016

2. Overall Description

2.1 Product Perspective

This mobile application is currently a new system in development. It was conceived as a tool for roommates who share bills and responsibilities to have joint access to a calendar that can track these elements and provide live feedback on their status, such as a bill being paid, or unpaid, or in-progress, or unlikely to be paid. Our application is a response to the increasing trend of prolonged cohabitation in the United States, and a lack of applications available to sufficiently organize the financial and domestic aspects of cohabitation.

This system was conceived as two parts, the first being a mobile application and the second being a web portal. The mobile application will be used to find other accounts in the area for the open network public message board functionality. The web portal will be used for managing bill/expenditure/activity status, due date, user assignment and information about the users and the system as a whole.

To search for other accounts, and existing public message board instances of the application in the area, the mobile application will need to communicate with a GPS application in the mobile phone. The GPS will provide the mobile application with the location of the user. This functionality will be embedded into the application so that other functions in the application may be accessed without any transition between applications.

The system will potentially be taking in a great deal of data from multiple users to an account. To store and reference that data the web portal part of the system will communicate with a database. The web portal will be able to add and modify data. All communication will take place over the internet.

2.2 Product Functions

With the mobile application a user will be able to set up a named group for their shared living space, and send out invites to the group to their cohabitant users, who have also downloaded the application and set up personal user accounts. Group membership will be structured similar to the mobile messaging application WhatsApp, where group access may only come from an existing group member's invite, but anyone may leave the group at any time. The users will be able to use the tools provided by the application to communicate about and organize the financial and domestic aspects of cohabitation.

All users on a shared account will be able to (1) add bills/expenditures/activities on a shared calendar for the group, (2) assign group user(s) onto a bill/expenditure/activity on the shared calendar, (3) update the status of a bill/expenditure/activity, (4) notify other users on the shared account about the status of the bill/expenditure/activity, (5) have access to separate views, in a

variety of data sorting options, of their personal responsibility schedule, and the shared pool of responsibilities, and (6) send messages to any user on the shared account.

Additionally users will be able to use the application to post to a public message board with their chosen profile name shared by all other nearby accounts of the application.

I. Users will be able to add bills/expenditures/activities on a shared calendar for the group
II. Users will be able to assign fellow group user(s) onto a bill/expenditure/activity on the shared calendar.
III. Users will be able to update the status of a bill/expenditure/activity within their personal list, or in the shared collection of responsibilities.
IV. Users will be able to notify other users on the shared account about the status of the bill/expenditure/activity.
V. Users will be able to have access to separate views, in a variety of data sorting options, of their personal responsibility schedule, and the shared pool of responsibilities.
VI. Users will be able to post 140 character messages to a private group forum that automatically deletes messages a week after initial posting.
VII. Users will be able to post 140 character messages to a public forum shared among any users of the application within 100,000 square feet of their GPS location, which will automatically delete any message 24 days after initial posting.
VIII. Users will be able to flag messages on the private or public message board for deletion for inappropriate content. Based on how many users accessing the feature, a majority user flag rate will result in the message being removed.
IX. Users within a group will be able to view all members connected to their group from their settings/group information tab.
X. Users within a group will be able to vote for the removal of a group member from the group. A majority vote will result in group expulsion.

2.3 User Classes and Characteristics

There are three types of users that interact with the system: 1) cohabitants of a living space, and 2.) proprietors/staff of the living space, and 3.) users who don't share an immediate living space, but are cohabitants of a neighborhood or area.

1.) Cohabitants: these users will use either part, or all of the functionality of the system, depending on their organizational needs.

2.) Proprietors/staff: these users will only use the public, open network message board service, most likely to address maintenance, security, complaint and suggestions postings, or to broadcast important information/announcements about the living space. These users may potentially use the organizational aspect of the application simply as a personal organizer, or to organize duties within their administration of the property by inviting employees to a constructed group.

3.) Households: These users, much like user type 2, are most likely to only use the open network to communicate with neighbors who also have the application. They may also potentially use the application as personal organizers, or invite other users to a group to manage domestic affairs, such as chores.

2.4 Operating Environment

The system is planned for Apple's iOS9 for mobile phones and tablets, with hopes of extending to Android 4.0 "Ice Cream Sandwich" in the future. All platforms must support internet connection and GPS.

2.5 Design and Implementation Constraints

The application is constrained by the need for an Internet connection. Since the application will store, track and model each user's calendar data by fetching, adding and changing data from the database over the Internet, an Internet connection is required. (Might research other ways to store data efficiently on each user's phone and still make it accessible across account)

The application is also constrained by the interface of the system to phone or tablet's GPS system. Depending on the features and implementation system of multiple GPS services, it'll be unlikely that the interface will be the same for all of them.

2.6 User Documentation

Several components of documentation will be made available with the application to make usability convenient. This documentation includes: in-app tutorials, help documents referencable through user interface, a support website for additional tutorials, FAQ's and question posting.

2.7 Assumptions and Dependencies

One assumption is that the product will be exclusively used on mobile phones and tablets that are up to specification. The application may not work as intended, crash, or provide inaccurate data and tracking if the platform does not have the hardware resources available.

A second assumption relies on the GPS of all platforms working uniformly. The application will need to be specifically adjusted to each phone or tablet if they have different interfaces for GPS, which would change the requirements significantly.

3. External Interface Requirements

3.1 User Interfaces

If the user is using the application for the first time, they should arrive at the log-in page upon booting the application. If the user hasn't registered for an account as yet, they should be able to do that through this log-in page.

If the user isn't a new user, they should be taken to a home screen that acts as the hub of the application's available functionality. Here the user will choose the specific functionality they'd like to employ between the calendar, their schedule, their list of tasks, list of bills, the messaging boards, account history, and user profile. Every user will have a profile page where they can edit their displayed name, e-mail address, phone number, address and password.

When a user selects the calendar, a calendar with the current month should replace the home screen. This calendar screen will display bills/expenditures/activities added by all users on the shared account, color coded by status. There will be available selections to add/delete bills/expenditures/activities. Selection of a specific activity on the calendar will switch the view to that specific activity where options will be available to do any of the following to the activity: edit/delete/update status/remind assigned user/ask assigned user for update. The calendar view can also filter postings to show only the bills/expenditures/activities of a specific selected user. Though the calendar defaults to the current month, past and future months can be swiped to.

On the home screen, selecting the message board should prompt the user for a choice of either the public, open-network, area-wide message board, or the private messaging board for users on the shared account. Both will have the same interface and functionality, allowing users to view postings marked important, add postings, or report postings for deletion based on inappropriate or offensive content.

(Mock-ups of GUI to be added later)

3.2 Hardware Interfaces

Since the mobile phone or tablet's operating system and the web server will handle the hardware connection to the database, and every phone/tablet's operating system manages the GPS, the application itself has no designated hardware apart from a standard iOS device.

3.3 Software Interfaces

The application communicates with the database to both read and change data. The application also communicates with the GPS application to get account location.

4. System Features (Functional Requirements)

When the user opens the app, they will be prompted for a user-name and password for logging in, or can click the 'Sign Up' button near the bottom. Selecting 'Sign Up' prompts the user to enter a user-name, e-mail address, and password. There is more account information to be collected, but you can opt to fill the rest out later. A verification e-mail is sent to the e-mail address provided to authenticate the user. After successfully creating an account, the user is taken to the 'Home' page.

Upon a successful log-in they will be taken to the 'Home' page where they can choose between options including, but not limited to (Home menu options not finalized): My Schedule, Tasks, Bills,

Calendar, Group/Cohabitants, Building Forum, and Settings. If the user is creating a new account, then upon the successful completion they will be brought to the Home page where they can select options they have access to (Group & Building Forum availability contingent on group membership and user location, respectively).

The functionality of My Schedule, Tasks, and Bills will be fairly similar and managed in a list-like way where each user can add and/or delete events, duties, and upcoming expenses respectively. The views of these functions are equipped with a 'Filter' option to see only the user's additions or all additions by the group. All of these can be assigned to different members within the Cohabitant group, but can be edited by anyone within the group (apart from personal events added to My Schedule). All will have statuses of some kind such as 'in-progress,' 'completed,' 'urgent,' or 'past due' with some overlap, but exclusivity will be integral in some cases.

The functionality of Calendar is quite simple and almost serves as the culmination of the three aforementioned features. Upon its selection, a Calendar view will appear in the 'Monthly' view, but can be set to 'Daily' or 'Weekly' as well. The Calendar will show all the events, duties (if assigned a "complete by' date or scheduled time), and bills added by the group members, color coded by status. It will also be equipped with a 'Filter' function to view all of the additions by a certain user in the Cohabitant group. All Calendar items can be selected, redirecting the user to the item's page where they can view and/or edit the details/status of the item, or even send a reminder to the person who is responsible for said item. The reminder will be received by the other user via push notification.

The two main functions remaining, Group and Building Forum, are set up as a group chat or discussion board where cohabitants can openly communicate with their group members (roommates) or their entire apartment building/complex, or residence hall. The Group's messages are private and must be accessed by an invitation to the group via a passcode. The Building Forum, however, is only active when the user verifies they live in their respective building or complex. The user's location is verified via GPS location matching to an address the user enters as part of their account information.

The last option on the 'Home' menu is Settings where the user can edit account information and preferences. This information includes a user's address and group information, such as the group passcode for inviting people to your Cohabitant group. Upon entering a resident address the user is prompted to provide permission for the app to use their location services to verify their location. The Settings feature also allows a user to disconnect from a Group or Building Forum, or even deactivate their account.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The application should be able to receive and relay information to the users in a quick and useful manner. There should be little to no delays if possible. When a new user is trying to create an account for the first time, the form should appear quickly and the form should also catch any input errors from the user. Once a user has signed in there should be a quick and easy way for the user to join a group as well as create a group. They will also be put into a building forum where every member of the building can participate in. The forum part of the board should be easily accessible and let the user see the current items and add anything. The user should also be able to flag posts to keep the forum as friendly as possible. The user should also have a way of viewing, creating, and editing tasks. Along with tasks, the user should be able to create bills that are due and have the ability to mark them as paid, unpaid, and will pay soon. There will also be a way for the user to confirm their bills or tasks by taking a picture and all should be smooth and quick. The user should also have a way of viewing and editing their account information in a quick manner. When the user is signing on, viewing a post, posting to a group, joining a group, and editing information there should be little to no delays on retrieval of the information. A user in a group should be able to add, delete, edit, request status, and assign certain tasks and bills to the people in the current group they are posting in. There user experience should be as smooth and updated as possible so that they can have the best experience possible.

5.2 Safety & Security Requirements

The user information will be securely transmitted to a server and saved there. Along with it, the location and account information for bill payments of each user will be secure.

The privacy of the user's entered data will be taken into consideration and will be kept private from everyone. The user identity authentication will be based on the location of the user as well as other options as given by the owners. A user would have the ability to post publicly only if they are in the group and is a verified member of the community. Information the user is giving and receiving will be monitored by the community and will be flagged if inappropriate items are found. Overall, the privacy and safety of our members is important and will be monitored strictly.

5.3 Software Quality Attributes

Availability

The application should be available at all times and if the internet gets disconnected while a user is trying to post something it would save and ask the user to try again when there's internet. While offline, the user should be able to view old posts that have been loaded at an earlier time.

Adaptability

The iOS application will need users to adapt to new concepts along with the users adapting, the developers will also need to adapt to what the majority of users prefer. With every new application, there is always some sort of learning curve for both parties.

Correctness

The information on the application would most likely be as correct as possible, however there may be flaws in what users posts. So the correctness is really dependant on the user.

Flexibility

The application is flexible to a majority of iOS users and will be available online and offline.

Interoperability

As the application is being updated, the user should not experience any serious interruptions. The application will contain interfaces that are easy to understand and that will make it easier to work with other systems or products.

Maintainability

Any errors or problems that are faced will be addressed as soon as possible to maintain the smoothness of the application.

Reliability

The application will be as reliable based on how the user uses the tools provided. Overall, it should be a great way for people to interact and get things organized.

Testability

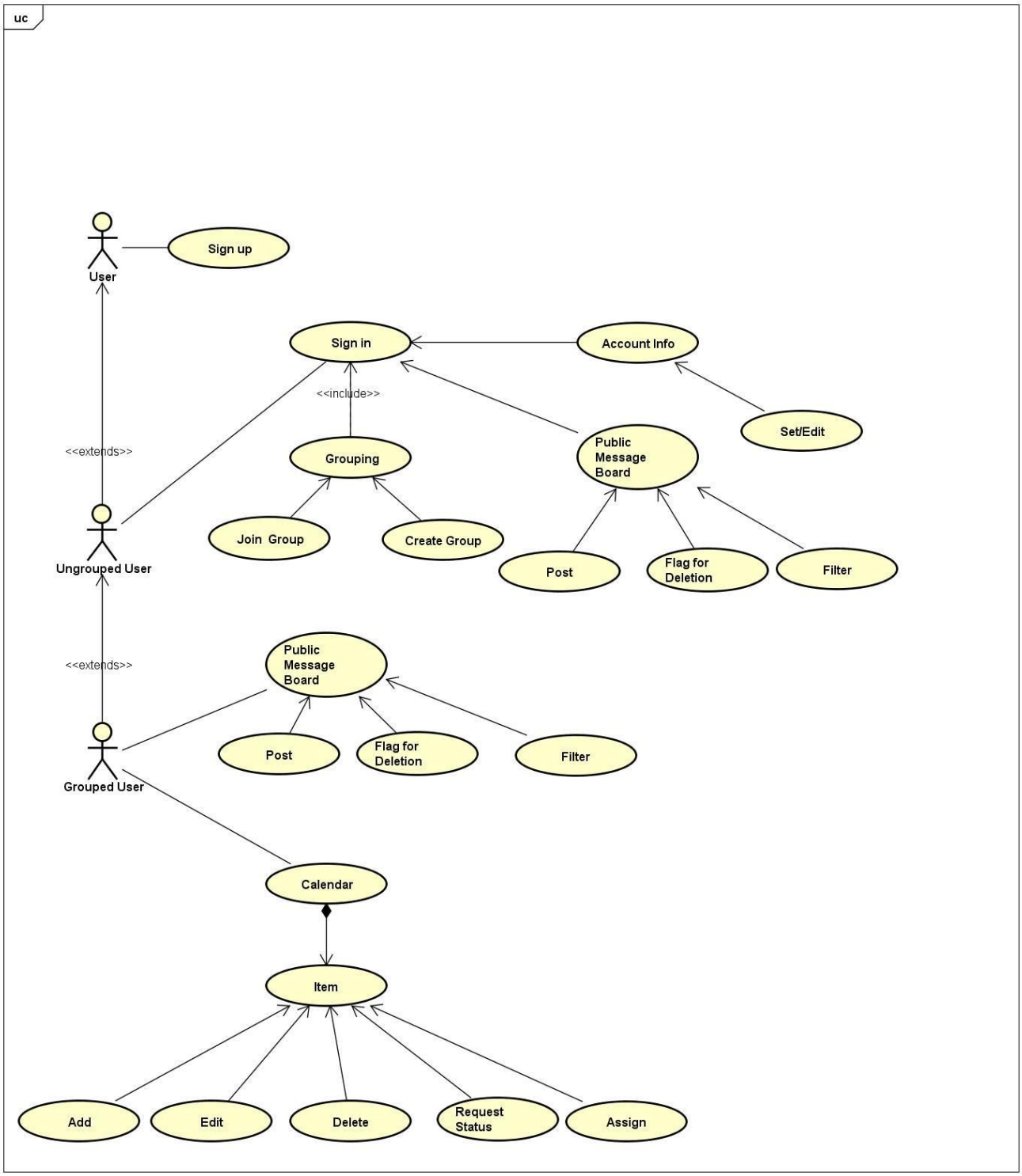
It can be tested by first starting with an apartment building, with individuals who have roommates, then expanding outwards to see what will happen and if there are any problems encountered.

Usability

The application is easy to use and navigate through in the way expected without any delays. It also reacts accordingly to each state and traverses through them as expected.

Appendix: Analysis Models

A.1 Use Case Diagram:



A.2 UML Class Diagram: