¡¡¡¡¡¡¡ HEAD ======= ¿¿¿¿¿¿¿ origin/master ¡¡¡¡¡¡¡ HEAD ======= ¿¿¿¿¿¿¿ origin/master

Software Requirements Specifications for Quarters

James Anthony (1135089)
Wenqiang Chen()
Carolyn Chong (1139105)
Kevin Ly()

October 9, 2015

Contents

List of Figures

List of Tables

Revision History

Date	Comments
October 9, 2015	Created first draft.

Template

This document makes use of the Volere Template for all of its organization.

1 Project Drivers

1.1 The Purpose of the Project

1.1.1 Project Background

Communication, the idea of exchanging information between one party to another one is an important aspect of people's everyday lives. With the introduction of Internet and cellular devices people's capacity to communicate has vastly increased, however this information hardly centralized. Landlords and their tenants commonly communicate via email or test messages, but a response is not guaranteed or the reply is hard to retrace.This lack of centralized information may introduce discrepancies between the two parties and hardships upon getting documents and other vital data.

Developing a new software platform to serve as an intermediate between landlords and tenants as a centralized hub of information will improve the quality of communication and create a more functional living environment for the occupants.

1.1.2 Project Goal

This platform is designed to handle information between landlords and their tenants, as well as between other tenants. It will efficiently and systemically handle requests from both parties and give a detailed status of the household.

1.2 The Client, the Customer, and Other Stakeholders

1.2.1 The Client

N/A

1.2.2 The Customer

This platform is targeted at tenants, and landlords as they will be the primary users of the software. With this implementation, tenants will be able to communicate efficiently with each other and their landlord. This target audience will require an easy to learn service that is robust enough to handle any cases that occur.

1.2.3 Other Stakeholders

There are an abundant of other stakeholders than can provide valuable insight towards the product.
These people/organizations include:

- Business analyst can collect the potential data, and complete data analyse on it.
- Plumbers, and electricians can also provide information directly to solving maintenance issues in the house.
- Legal experts can protect the users and developers from unlawful acts.
- Developers are required to create and test the platform.
- Marketing Experts can provide beneficial tactics on how to advertise.

1.3 Users of the Product

There are two key users that will be using this platform where these users are critical to the product, thus the requirements generated is heavily influenced by tenants and landlords.
Tenants are responsible for the following actions:

1. communicate information to other tenants when needed
2. to complete the chores assigned to them
3. report issues in the house via the ticketing system
4. be on time with payments
5. ensure the rules outlined for the house are followed

Landlord are responsible for the following actions:

1. communicate information in regards to the house to the tenants
2. actively engage in discussion questions
3. diligently completing maintenance requests
4. be adequately available when issues arise

2 Project Constraints

2.1 Mandated Constraints

Solution Constraints

Description: Landlords and tenants shall communicate through a discussion board.
Rationale: Landlords and tenants will not need to communicate via email, text message, or phone.
Fit criterion: All communications conducted via quarters discussion board will be instant and accessible by all members of the group.

Description: Property concerns will be brought to the attention of the landlord by issuing a maintenance ticket.
Rationale: Tenants will not email, text, or phone their landlord in order to inform them about property related issues.
Fit criterion: Tenants will have the option to pay rent/bills through paypal via the payment portal.

Description: Tenants will be able to make payments through PayPal
Rationale: Landlords will be able to offer a payment method other than post-dated cheques.
Fit criterion: Landlords receive rent/itility payments on time through PayPal from tenants who have agreed to use the service.

Implementation Environment of the Current System

- See off-the-shelf software and collaborative applications
- Server: Intel i3-4430 running Ubuntu Server

Partner or Collaborative Applications

- Facebook Login: Provides users with the option to sign up via Facebook
- Google Sign-In: Provides users with the option to sign up via Google

- PayPal Payments: Provides users with the options to make payments via PayPal

Off-the-Shelf Software

- MySQL
- ExpressJS
- AngularJS
- NodeJS

Anticipated Workplace Environment

- Home: Website must display properly on desktop and laptop computers
- Mobile: Website must display properly on mobile browsers

Schedule Constraints

- Proof of Concept Demonstration, November 16-27
- Revision 0 Demonstration, February 1-12
- Final Demonstration, Exam Period

Budget Constraints

N/A

Enterprise Constraints

N/A

2.2 Naming Conventions and Terminology

Glossary of All Terms

- Landlord: A person who owns or runs a boarding house, inn, or similar establishment
- Tenant: A person who occupies land or property rented from a landlord

2.3 Relevant Facts and Assumptions

Relevant Facts

- Pay-Pay is a secure method of making online payments.
- Facebook Login is a secure form of account management.
- Google Sign-In is a secure form of account management.

Assumptions

- It is assumed that both landlords and tenants will be capable of using web applications for comminication, planning, and payments.

3 Functional Requirements

3.1 The Scope of the Work

3.1.1 The Current Situation

There is currently no existing software platform that attempts to simplify and document communication between landlords and tenants. A web application is needed to serve as a centralized management solution that will benefit both types of users. The web application will include document storage, in-app payment, a calendar, instant messaging, and discussion boards.

3.1.2 The Context of the Work

See Figure 1.

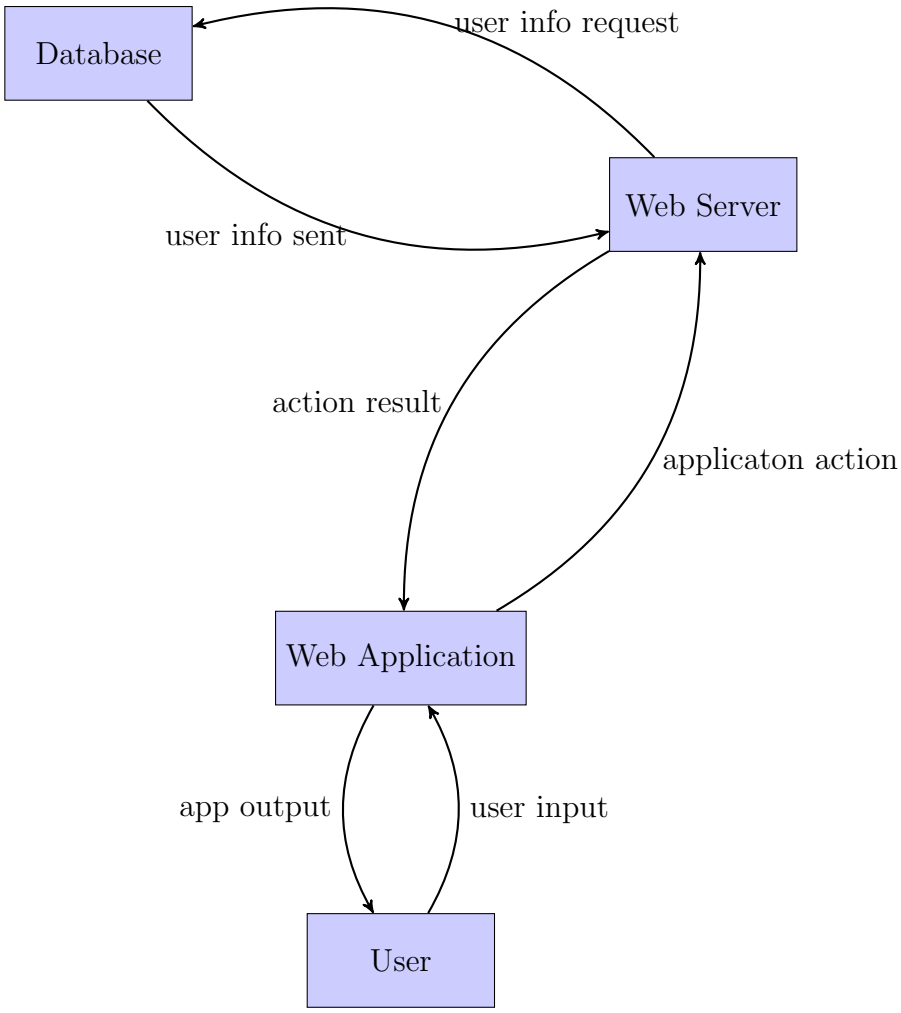

Figure 1: Work Context Diagram

3.1.3 Work Partitioning

See Table 1.

3.2 Business Data Model and Data Dictionary

N/A.

Event Name	Input and Output	Summary
1. User logs in.	User ID (IN) User password (IN)	User logs in to app.
2. User logs out.	User ID (IN)	User logs out from server.
3. User creates house.	User ID (IN) House name (IN)	Landlord inputs the house address into system. The system then saves the house in the database.
4. User joins house.	User ID (IN) House name (IN)	User selects a house to join. The system adds the user to the house.
5. User uploads file.	User ID (IN) File (IN)	User adds a file to be visible to house. The system then saves the file in the database directory.
6. User submits maintenance request.	User ID (IN) Request (IN) Request (OUT)	User that is tenant submits a maintenance request to be received by landlord.
7. User adds chore.	User ID (IN) Chore (IN) Calendar (OUT)	User submits a chore to be completed and displayed on the Calendar.
8. User initiates chat with other user.	User ID (IN) User ID (IN) Chat (OUT)	User creates chat window with other user.
9. User sends instant message.	User ID (IN) Message (IN) Chat (OUT)	User submits a chore to be completed and displayed on the Calendar.
10. User adds post.	User ID (IN) Post (IN) Discussion Board (OUT)	User adds post to discussion board.
11. User comments on post.	User ID (IN) Comment (IN) Discussion Board (OUT)	User comments on post on discussion board.
12. User transfers funds.	Record of Transaction (OUT)	User transfers funds to another user using third-party software. A record of the transaction is displayed.

Table 1: Work Partitioning

3.3 The Scope of the Product

3.3.1 Product Boundary

3.3.2 Product Use Case Table

3.4 Functional Requirements

3.4.1 Functional Requirements

Requirement #: 1 **Requirement Type:** 9 **Event/Use Case:**
Description:
Rationale:
Fit Criterion:
Customer Satisfaction: **Customer Dissatisfaction:**

4 Nonfunctional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

The interface of the web application shall be attractive and intuitive for a young adult and adult audience. A sampling of potential users shall, without prompting or enticement, create a login within one week of their first encounter with the application.

4.1.2 Style Requirements

The web application shall appear professional and secure. After their first encounter with the application, 70 percent of potential users shall agree they feel they can trust the application.

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

The web application shall be easy for young adults and adults to use. The application shall be used by users with no prior training. A casual user should be able to use the application with the same ease of a frequent user. The

application shall make the users want to use it. A test panel of current landlords and their tenants shall be able to successfully create a user account and use the application's functions without guidance within their first encounter.

4.2.2 Personalization and Internationalization Requirements

The web application shall be available in the English language (EN-US), and use Canadian currency (CAD $) and the metric system. The interface and functions of the web application shall be personalized for the type of user, either landlord or tenant.

4.2.3 Learning Requirements

The web application shall be easy for young adults and adults to learn. The web application shall be constructed so that all of its functionality is apparent upon first encountering it. A brief tour of the web application shall be presented as an option to first time visitors of the site. A test panel of current landlords and their tenants shall be able to successfully create a user account and use the application's functions productively without guidance within their first encounter.

4.2.4 Understandability and Politeness Requirements

The web application shall use symbols, icons, and words that are naturally understandable by the user community.

4.2.5 Accessibility Requirements

The web application shall rely on the web browser's accessibility features to make it available to the disabled.

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

The speed of the web application depends on the speed of the user's operating system and internet connection.

4.3.2 Safety-Critical Requirements

N/A.

4.3.3 Precision or Accuracy Requirements

The web application shall keep accurate time by working in UTC. All monetary amounts shall be accurate to two decimal places.

4.3.4 Reliability and Availability Requirements

The web application shall be available for use 24 hours per day, 365 days per year.

4.3.5 Robustness or Fault-Tolerance Requirements

The web application shall successfully display an error message to the user should an incorrect username/password combination be input, or in the event of one of its features crashing.

4.3.6 Capacity Requirements

The web application shall cater to [x number —CC] simultaneous users.

4.3.7 Scalability or Extensibility Requirements

The web application shall be capable of expanding to nearby cities within two years of its launch.

4.3.8 Longevity Requirements

The web application shall be expected to operate as long as there exists a housing rental market.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

The web application shall be used by users who may be distracted because they are simultaneously completing and managing several other tasks.

4.4.2 Requirements for Interfacing with Adjacent Systems

The web application shall work on the last three releases of the five most popular web browsers (Chrome, Firefox, Internet Explorer, Opera, Safari). The web application shall interface with PayPal to handle monetary transactions between users. The web application shall interface with Google Sign-In and Facebook Login to enable users to login with pre-existing social media accounts. The details of the communication standards/protocols will be outlined in the Design Document after implementation is completed.

4.4.3 Productization Requirements

The web application shall be accessible on the World Wide Web.

4.4.4 Release Requirements

The initial release of the web application will be in February 2016. The next release will be in April 2016. Subsequent releases will be made on an annual basis.

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

The web application shall be able to be maintained by developers who are not the original developers.

4.5.2 Supportability Requirements

N/A.

4.5.3 Adaptability Requirements

The web application is expected to run on web browsers on mobile phones, tablets and desktop computers.

4.6 Security Requirements

4.6.1 Access Requirements

Only the user has access to edit their own personal stored information and choose what information of their profile is visible to other users. Users have access to view other users' profiles. Only landlords can [add —CC]. Only tenants can [add —CC]. Only the landlords and tenants belonging to the same property can view the property's group and add content to the property's group.

4.6.2 Integrity Requirements

The web application shall prevent incorrect data from being introduced and protect itself from unwanted attacks by unauthorized users. The web application shall have a back-up of its stored data on an alternate server.

4.6.3 Privacy Requirements

The web application shall make its users aware of its information practices before collecting data from them. The web application shall use a third-party interface to store credit card information and perform secure monetary transactions between users.

4.6.4 Audit Requirements

N/A.

4.6.5 Immunity Requirements

N/A.

4.7 Cultural and Political Requirements

4.7.1 Cultural Requirements

N/A.

4.8 Legal Requirements

4.8.1 Compliance Requirements

N/A.

4.8.2 Standards Requirements

N/A.

5 Project Issues

5.1 Open Issues

- Size of the user group is uncertain, may need to upgrade hardware in the future to accommodate more user
- User interface has yet to be designed
- Browser compatibility
- Methods to store documents, database vs locally on server

5.2 Off the Shelf Solutions

5.2.1 Ready-Made Products

No ready-made products exist with the same functionalities.

5.2.2 Reusable Components

Facebook and Google accounts can be used for sign-in and connecting members in the house

5.2.3 Products That Can Be Copied

Not applicable

5.3 New Problems

5.3.1 Effects on the Current Environment

N/A

5.3.2 Effects on the Installed Systems

N/A

5.3.3 Potential User Problems

N/A

5.3.4 Limitations in the anticipated Implementation environment That May Inhibit the New Product

Old browsers are not compatible with the system

5.3.5 Follow-Up Problems

- User abusing the system
- User uploading sensitive data

5.4 Tasks

5.4.1 Project Planning

- present requirement document to supervisor for feedbacks
- development prototype for demo purpose

5.4.2 Planning of the Development Phases

- design the database
- split application into different modules for assign one module to each member

5.5 Migration to New Product

N/A

5.5.1 Requirements for Migration to the New Product

N/A

5.5.2 Data That has to be Modified or Translated for the New System

N/A

5.6 Risks

- Legal issue with online payment
- Not able to get enough user
- Certain features may not be compatible with the hardware
- Project becomes too complicated and not able to meet deadlines

5.7 Costs

- Domain cost and web server cost, if we decide to go live
- Approximately 7 months of development time

5.8 User Documentation and Training

5.8.1 User Documentation Requirements

- A help guide will be included
- FAQ section
- Tour of the website is shown for first time user

5.8.2 Training Requirements

No training is required for the user. When they visit for the first time they will be given a tour.

5.9 Waiting Room

The next release will include a separate module for housing advertisements, ability to attach picture and files in bulletin board.

5.10 Ideas for Solutions

- MySQL for relational database
- Node js and Express js for server side
- Angular Bootstrap for front end