Software Requirements Specification
For
Team 3

February 15, 2017
Version 1.0

Prepared by:
Lucy Kull, Caleb Hillhouse, David Jones, Candice Johnson

Table of Contents

Revision History

Version	Date	Name	Description
1.0	2/15/17	Caleb,David, Lucy, Candice	Created the document
1.1	4/24/17	Caleb	● Added use cases for creating, modifying, and removing budgets ● Added new feature: Notifications ● Added use cases for Notifications: budget notifications and goal notifications

1 Introduction

1.1 Overview

This document defines the requirement for the financial web application that is being developed for Commerce Bank. The purpose of this document is to represent the system requirements in a readable way so that clients and stakeholders can understand them and verify them for correctness but with enough detail that developers can design and implement a software system from them.

This document doesn't address *project* issues such as schedule, cost, development methods, development phases, deliverables, prototypes, user guides, user interfaces, and testing procedures. Those are addressed in separate project documents and quality assurance test plans.

The financial web application is a tool for Commerce Bank account holders to use in planning their financial goals. Through the application, they will be able to track their goals, see how close they are to achieving them, and get a motivational boost to help them succeed. Gamification elements are paired with user-entered goals to provide encouragement. Also, visual elements of the application make it easier to see at a glance things like goal completion and other useful information.

1.2 Goals and Objectives

The three main goals of the product are:

1. Provide an easy way to set goals and track progress towards those goals.
2. Provide an easily understandable view of the user's financial information to empower their future financial decisions.
3. Improve the experience of financial planning with gamification elements to make finances fun.

1.3 Scope

The budgeting application will access spending habits saved in the application's database. User goal progress will also be displayed to the user. No user information will displayed to anyone outside of internal Commerce Bank access without the user's permission. The reason for this is the security and safety of the user's information.

1.4 Definitions

User – a Commerce Bank customer using the system

Budgeting Web Application - the project currently being worked on
Application - shorthand for Budgeting Web Application

Use case – describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

Scenario – one path through a use case

Product – what is being described here; the software system specified in this document.

Project – activities that will lead to the production of the product described here. Project issues are described in a separate project plan.

Shall – adverb used to indicate importance; indicates the requirement is mandatory. "Must" and "will" are synonyms for "shall".

Should – adverb used to indicate importance; indicates the requirement is desired but not mandatory.

May – adverb used to indicate an option. For example, "The system may be taken offline for up to one hour every evening for maintenance." Not used to express a requirement, but rather to specifically allow an option.

Controls – the individual elements of a user interface such as buttons and checkboxes.

1.5 Document Conventions

Portions of this document are incomplete. Areas with little or no detail will be expanded upon when more information is known.

1.6 Assumptions

Based on requirements elicited during the first meeting, it is assumed that databases at Commerce Bank will be able to conform to the databases we design, or conform our code to fit their current databases, although the way they will do this is unknown.

It is also assumed that no one session will require pulling data from multiple users, unless said users are using a shared account.

2 General Design Constraints

2.1 Product Environment

The budgeting web application will be a component of Commerce Bank's already existing user interface. Commerce Bank's user interface already includes a log-in for separate users. The budgeting web application will only become available after the user has logged into their account. The account information as well as other information needed by the budgeting web application will be held in a SQL server database.

2.2 User Characteristics

Experienced Users - Know how to fully navigate the application as well as Commerce Bank's website. These users will know how to create and set goals and budgets. These users will not be prioritized as the application will be simple to use.

New Users - First time users who have never used a budgeting application before (or who have never used this specific budgeting application). These users will not know how to create or set budgets and goals. These users will learn how to do these through the Help section of the application. These users will not necessarily be prioritized because the application will be created in such a way that is able to cater to both New and Experienced Users.

PC Users - Users who are accessing the application through a desktop or laptop computer. These users can also be considered either a New User or an Experienced User but cannot also be a Mobile User. In terms of how the application works, all users will have the same priority. However, if the resource management gamification idea goes through PC Users will have priority over Mobile Users in terms of gamification.

Mobile Users - Users who are accessing the application from their mobile devices or smaller screens. These users can also be considered either a New ser or an Experienced User but cannot also be considered a PC User. These users will have the same priority as all other users in terms of how the application works. As stated above, if the resource management gamification idea is liked by Commerce Bank, then PC Users will be given priority over Mobile Users in terms of gamification.

2.3 Mandated Constraints

1. Must be a web application built in .NET Framework.
2. Client side framework/libraries must be included (no external resources).
3. DB must be in SQL server 2012.
4. Responsive on mobile devices/smaller screens
5. Tool should allow for multiple goals to be entered and possible goals created.
6. Daily screen should be easy to read and show goals created as well as how customer is doing so far.
7. Show example of what happens when user reaches goal and what happens when they don't reach their goal.
8. Gamification features when goals are met (like FitBit and Apple Watches).
9. Notifications when appropriate.

2.4 Potential System Evolution

The current version of gamification (a resource management style game) is currently being developed into a functional prototype to show to Commerce Bank for verification on what they are looking for in terms of gamification. The user interface for the

application is also being worked on to gauge what Commerce Bank is looking for in a user interface. For right now, the only foreseeable changes that might occur are either cosmetic or how gamification is implemented.

3 Nonfunctional Requirements

3.1 Usability Requirements

Users should find the application easy to use. That is, very limited use of a User Guide or help section should be necessary for 90% of users to experience 90% of the functionality of the application. Sections of the application should be intuitive and cohesive. 80% of users should feel confident that their ability to use the app is conducive to the success of their goals.

Setting a goal should be an easy and efficient task. 90% of users should be able to set a new task accurately in under a minute.

Removing a goal should be fast and error resistant. 99% of all goal deletions should be intended. Removing of the goal should be instantaneous after user confirmation to remove it.

Modifying a goal should be as easy as goal creation. 90% of users should be able to modify a goal correctly on the first attempt.

Transaction information screens should be easy to understand by most users. 95% of users should be able to obtain their desired understanding in under twenty second.

3.2 Operational Requirements

The user's environment may be noisy, so the application should not depend on the user hearing audible output.

The user is expected to have access to a screen of some sort (phone, computer monitor, etc.) so visual elements can be relied on for data presentation and user control.

The user must have a connection to Commerce Bank's servers throughout the application's operation. Connectivity requirements will mirror requirements of Commerce Bank's online service.

3.3 Performance Requirements

The application startup should take less than 3 seconds. Actions within the application should take less than 2 seconds. General performance of the application should not hinder the user from experiencing features.

3.4 Security Requirements

In this version of the financial web application, security is not a priority. In later versions, details of security will be addressed, since only the later versions will be accessed by the public.

3.5 Safety Requirements

Safety requirements of the financial web application are not a major concern. Because the system is solely a software system, and self-contained, there is minimal risk of danger to be concerned with. Compartmentalization of information views (e.g. only seeing sensitive account information when needed) may be considered.

3.6 Legal Requirements

A user's data will only be accessible to the user, Commerce Bank, and any parties addressed in the Terms of Service provided by Commerce Bank. If sharing features are implemented, they will be carried out only at the user's discretion.

3.7 Other Quality Attributes

The financial web application will be designed in a way so that most operations for most people will be handled in a short amount of time. This is valuable due to how some users will frequently open the application.

The application will be portable between different screen sizes by altering the elements on screen, changing the format, etc. We want the user to be able to access the system on the go, so this is important.

The application will have an attractive user interface. Outdated or visually unappealing user interfaces can sometimes deter users from using an application. Ensuring that the user interface is visually appealing will prevent users from having the impression that a visually unappealing application is not valuable to them.

3.8 Documentation and Training

A help section may be accessed within the system to explain how the system works or to address frequently asked questions. This help section is comprehensive and ordered into categories to allow the user to locate relevant information quickly.

3.9 External Interface

The application will require a user interface and access to a database.

3.9.1 User Interface

The user interface will have a simple professional design with a modern style, in line with what most internet users experience on most professional web applications. The user interface will include two main sections. Navigation will allow the user to select from separate sections of the application. The content section will change according to what

the user has selected in the navigation section.

Elements of the user interface will change dynamically so that the number of pages is limited. As such, media such as images will be limited, either in quantity or quality, in order to optimize performance. Most design elements of the user interface will be produced through the use of program languages rather than external media. Specifically, the user interface will be implemented using HTML5, CSS3, and JavaScript.

The user interface will make use of visual aids such as charts or graphs in order to present data to the user in a friendly environment.

A section of the user interface will include an entertaining game in which the content of the application is conceptually applied. The game is not required to experience or access the remaining functionality of the application.

The user interface should be intuitive. The user should make very little use of the User Guide or help section. This implies that the user interface should be well organized and cohesive. 90% of users should be able to experience at least 90% of the features of the application without consulting the User Guide or help section.

3.9.2 Database Interface

The application requires the use of an SQL Server 2012 database system in order to maintain accurate and up-to-date data associated with each user.

4 System Features

4.1 Feature: Financial Goal Planning

4.1.1 Description and Priority

Users of this application will be able to set financial goals and budgets. Users will also be able to observe their progress towards their goals and budgets. There are various types of goals, such as setting aside a particular amount for a purchase, spending a certain amount less at a coffee shop, etc. Users should be able to set such goals and budgets and track their progress using the application and the Commerce Bank databases.

Cost: medium
Risk: low
Value: high

4.1.2 Use Case: Set a monetary goal to be reached

A user can assign a specific monetary value they hope to reach for a particular thing. That value will be recorded and a sum towards it will be allocated. A savings account, or a percentage of a savings account, will be allocated towards the goal.

4.1.3 Use Case: View Progress Towards a Goal

A user can check any of their current goals and quickly see how close they are to completion. This information is displayed differently by goal type in order to make understanding progress easy and quick.

4.1.4 Use Case: Remove a Goal

The user should be able to easily remove a goal. A user removing a goal will trigger the back end to alter the database automatically.

4.1.5 Use Case: Modify a Goal

The user should be able to change goal details easily. For example, goal amounts, time specified, etc. The system should then update the database to reflect the new goal.

4.1.6 Use Case: Set a Spending Budget for a Category

A user can define a spending limit for a category of their choice as well as the duration of the spending limit. A notification will be sent to the user once a certain threshold within the budget is reached.

4.1.7 Use Case: View Progress Towards Budget Limit

A user can check the budgets they have set to see how close they are to reaching their budget limit. The display of this information will be similar to how goal progress is displayed.

4.1.8 Use Case: Remove a Budget

The user will be able to remove a budget. This will trigger the back end to alter the database automatically.

4.1.9 Use Case: Modify a Budget

The user should be able to change budget details such as spending limit and the duration of the budget. The system should then update the database to reflect the new budget.

4.1.10 Additional Requirements

Depending on the goal implementation, which is not fully finalized at the time of writing this document, requirements will be based on savings accounts the user has access to at Commerce Bank.

4.2 Feature: Notifications

4.2.1 Description and Priority

The web application will send either email or text notifications to the user when certain thresholds are reached during goal and budget progression. The user will be able to go into their profile and select whether they want to receive notifications via email or text.

The user will also be able to select whether or not they wish to receive notifications at all.

Cost: low
Risk: low
Value: medium

4.2.2 Use Case: Budget Notifications

The web application will send a notification via email or text to the user when a budget reaches a certain percentage.

4.2.3 Use Case: Goal Notifications

The web application will send a notification via email or text to the user when a goal reaches a certain percentage of completion.

4.3 Feature: View Pertinent Financial Information

4.3.1 Description and Priority

Users will be able to use the financial web application to view information regarding accounts that are used for the goal setting and tracking. Things like transaction history and balance are viewable without requiring the user to go elsewhere.

Cost: low
Risk: low
Value: medium

4.3.2 Use Case: View Accounts

A user should be able to view balances of accounts tied to the financial web application.

4.3.3 Use Case: View Transaction information

A user should be able to see transaction information that influences the financial web application.

4.3.4 Additional Requirements

Requirements will be based on savings accounts the user has access to at Commerce Bank.