Software Project Management Plan

Commerce Bank

10/11/2020

Team Members

Blake Brown
Josiah Conely
Jacob Gleinser
Jimmy Blundell

Document Control

Change History

Revision	Change Date	Description of changes
V1.0	10/11/2020	Initial release

Document Storage

This document is stored in the project's SVN repository at:
https://github.com/UMKC-CS451R-FS2020/semester-group-project-group-2

Document Owner

Jacob Gleinser is responsible for developing and maintaining this document.

Table of Contents

1.1 Purpose and Scope

The project is meant to fulfill the requirements given by Commerce Bank and the professor of 451R – Software Engineering Capstone.

The purpose of this project is to provide bank users with a user interface in which they can easily create and modify alerts for bank transactions.

The boundaries of the scope of this project are that the data will be "made-up" to a degree that in the real world the database containing users and accounts will be far more complex. The extent of the back end in which the database and the means to query that database only exist to facilitate the alert features of the website

1.2 Goals and Objectives

Project goals:

1. Establish a web-page in which users can make alerts for transactions.
2. The web-site is well designed, being both visually appealing and intuitive to users.

Project objectives:

1. Create a database of users and transactions.
2. Create an interface between the Database and the application to send POST/GET requests.
3. Create an interface where users can sign up, create accounts.
4. The website should be held to modern standards in terms of responsiveness and usability on a variety of screen resolutions including cell phones.

1.3 Project Deliverables

This section lists the outputs of the project that are delivered to the customer.

The following items will be delivered to the customer on or before 12/9/2020:

1. Source code for both the client and server portions of the system.
2. User's Guide
3. System Administrators Manual
4. Test Plan
5. Total coverage of code by unit tests
6. Any changes made from the product of unit tests.

1.4 Assumptions and Constraints

Assumptions:

1. The API for POST/GET requests and querying the DB works on the test hardware.
2. Browser to test website and functionality will be either Chrome, Firefox, or IE and be on a version released in the last year.

Constraints:

1. The website must run on Chrome, Firefox, and IE in a similar capacity.
2. The software must be ready by 12/9/2020.

1.5 *Schedule and Budget Summary*

Due Date:
10/9/2020 - Database built with the model for testing functionality.````
10/16/2020 - Database connects to the website, able to make simple queries.
10 /21/2020 - Login/Register page - Add users to Database. Users can login with appropriate business logic set up.
11/05/2020 - Retrieve a summary of transactions made.
11/11/2020 - Basic home-page/alt page with the above requirement functionality.
11/21/2020 - Website CSS/HTML is complete
12/05/2020 - Unit testing / Unit tests are built for appropriate business logic concerns. Testing complete.

There is no current monetary budget for this project, however in terms of time, our current estimate

1.6 *Success Criteria*

1.6.1 All high priority user stories and functionality can be accomplished by the users.
1.6.2 Website conforms to standard conventions for UI elements and design.
1.6.3 Website is considered appealing by 90% of users polled.
1.6.4 Website is able to be used by 99% of consumers without a manual.

1.7 *Evolution of the Project Plan*

We use the “iteration tracker” spreadsheet to document what each participant is working during each iteration with deliverables. We document how long we expect each item to take, and then we record the actual time.

2 Startup Plan

2.1 Team Organization

Project Manager:	The project manager is responsible for creating the project plan (with input from those doing the work), managing risks, running the weekly team meeting and providing monthly status reports to senior management.
Programmers (3):	Programmers are primarily responsible for coding and unit testing modules. They are also expected to take part in architecture planning and review meetings.
UI Developer:	The UI developer is responsible for creating a site map of the website, selecting styles and implements the visual framework.
Build Coordinator:	The build coordinator is responsible for setting up, running and distributing the results of the nightly build.
Database Coordinator:	Responsible for choosing, creating and maintaining the database that will be used inside of the project.

2.2 Project Communications

The project team will meet once a week in a virtual meeting. Communication throughout the week will be conducted in chat on Slack/Zoom. When a story is finished, team members should notify other team members and it should be reviewed and if acceptable committed on Github.

2.3 Technical Process

This group will follow the agile methodology with stories being posted each iteration and those stories being selected by each member - with certain members being assigned roles to facilitate and manage the creation and completion of stories.

2.4 Tools

This section specifies the development tools the team will be using to perform their work.

- Programming Language –Angular, HTML, CSS, TypeScript, MySQL
- Version Control – source code and written artifacts will be stored in a Github repository.
- Defect tracking – defects and issues will be tracked using Github
- Automated testing – unit tests will be implemented with the JUnit testing framework.
- AWS for hosting.
- Visual frameworks – Angular/Bootstrap

3 Work Plan

3.1 Activities and Tasks

A work breakdown structure is an excellent tool for identifying a complete list of tasks.

Depending on the needs of the project, some or all of the following attributes will be recorded for each task:

- Task name
- Task Description
- Owner
- Effort estimate
- Actual effort
- Planned start and stop dates
- Actual start and stop dates
- Dependencies among other tasks

3.2 Release Plan

Due Date:
10/9/2020 - Database built with the model for testing functionality.
10/16/2020 - Database connects to the website, able to make simple queries.
10 /21/2020 - Login/Register page - Add users to Database. Users can login with appropriate business logic set up.
11/05/2020 - Retrieve a summary of transactions made.
11/11/2020 - Basic home-page/alt page with the above requirement functionality.
11/21/2020 - Website CSS/HTML is complete.
12/05/2020 - Unit testing / Unit tests are built for appropriate business logic concerns. Testing complete.

3.3 Iteration Plans

First iteration:

Database built with sample data.
Website sitemap/wireframe developed

Second iteration:
Website built at low depth - no major functionality but resembles the wireframe/sitemap.
Stored procedures written for DB

Third Iteration:
API calls to DB, post/get queries
Styles applied to website, the UI of the website will be complete.

4 Control Plan

4.1 Monitoring and Control

Weekly – Team meeting. Project participants report status, progress and potential problems.

Daily - Keep in contact through the use of Slack.

11/1/2020 – Critical Design Review. Formal inspection of product architecture.

12/09/2020 – Executive Review. The project manager presents current project status to project sponsor and senior executives.

4.2 Project Measurements

Product and process measures support project management and estimation by

Phase	Measurement	Source
Iteration Planning	Record effort estimates for scheduled tasks Update effort estimates for product features Update estimated dates in release plan	Mgr/Pgr/QA
Iteration Closeout	Record actual effort for scheduled tasks Record actual effort for product features	Mgr/Pgr/QA
System Test	Record the rate at which errors are found.	QA
Project Closeout	Archive project performance data on project charter in Github.	QA

5 Supporting Process Plans

5.1 Risk Management Plan

- Team members unfamiliar with current technology – frameworks/libraries/tools might change as the project progresses.
- Weekly scrum meetings to make sure we can stay on the same page throughout the progress of our project.
- Outside of the meetings we will communicate mostly through Slack if we have additional information, questions, or concerns about the project.

5.2 Configuration Management Plan

Configuration management plans for this document and other baselined work products including review procedures and change management procedures.

1. All products will be hosted on a Github repository that each member has access to.
2. All project (work products) items (documents, source code, test cases, program data, test data, etc) will be stored in the Github repository but not all will be under change control.
3. Items that are subject to change control will be considered baselined after a group review at the end of the life cycle phase during which they are created. Baselined here means that the product has undergone a formal review and can only be changed through the prescribed change control procedures.
4. The change control procedure once a product is baselined is: (1) anyone wanting to make a change to a baselined item notifies the rest of the group on Slack describing the change, reason for the change, expected impact, and timeline for integrating the change. (2) if no one responds to the group within 2 days with a reason for why the change request shouldn't be permitted, it will be considered accepted and the person proposing the change may proceed with the change. If anyone does object to the change, the reason for objecting will be discussed at a meeting where everyone is invited to attend and voice their opinion.

5.3 Product Acceptance Plan

- *The website should be responsive.*
- *The website is functional while being hosted on a server that all team members and product owners can use.*
- *Queries to the database follow standard security protocols, i.e. stored procedures to avoid SQL injection.*
- *Functional error messages, for example, if the user inputs a string of letters instead of an amount, not enough characters in password, etc…*
- *Meets previously documented functional and technical requirements*
- *Meets the 10% Unit Testing criteria.*