**C868 – Software Capstone Project Summary**

**Task 2 – Section A**



|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | Simple Recipe Locator |
| **Student Name:** | Amit Kohli |

**Table of Contents**

Contents

[**Business Problem** 3](#_Toc10981156)

[**The Customer** 3](#_Toc10981157)

[**Business Case** 3](#_Toc10981158)

[**Fulfillment** 3](#_Toc10981159)

[**Existing Gaps** 3](#_Toc10981160)

[**SDLC Methodology** 3](#_Toc10981161)

[**Deliverables** 4](#_Toc10981162)

[**Implementation** 4](#_Toc10981163)

[**Validation and Verification** 4](#_Toc10981164)

[**Environments and Costs** 5](#_Toc10981165)

[**Programming Environment** 5](#_Toc10981166)

[**Environment Costs** 5](#_Toc10981167)

[**Human Resource Requirements** 5](#_Toc10981168)

[**Project Timeline** 6](#_Toc10981169)

[**Sources** 8](#_Toc10981170)

# **Business Problem**

**The Customer**

The customer is Simple Food Solutions. Simple Food Solutions is a provider of food products and services. They are a medium-sized company located in Charlotte, NC. Their mission is to provide simple ways to access food products and services. The company is projected to grow substantially since they are looking to add many new services while minimizing the increase of expenditures.

## **Business Case**

Simple Food Solutions is looking to offer more food services. In a survey conducted by their public relations department, Simple Food Solutions was able to draw the following conclusions: there are many people who like to cook, but are unable to find their old cookbooks or they might be too “lazy” to open their cookbooks or they could be traveling and do not have access to their cookbooks. As a solution, they proposed to create a service that provides recipe information via a simple means (as an alternative to paper cookbooks) to the general public in order for Simple Food Solutions to improve public relations and public awareness. After brainstorming, Simple Food Solutions proposed the creation of a web application that would provide recipe information.

## **Fulfillment**

A web application that provides the capability of locating recipes is a great solution for Simple Food Solutions. The web application will contain a responsive, user-friendly web interface. The user will be able to perform the following basic functions: searching for recipes, registering for an account, logging into the account, customizing user settings, and generating reports. The recipe information will be provided through a web service that Simple Food Solutions will create. Any user specific data such as account information will be persisted in a MySQL database.

# **Existing Gaps**

No existing system or application is being replaced. This application and web service is the first project for Simple Food Solutions.

# **SDLC Methodology**

The selected methodology for this project will be Agile. Development for the web application will follow an iterative approach with new features added incrementally in new releases. The phases used will be requirements gathering, designing, developing, and testing, and release. An example of an iteration, will be to gather requirements for the web user interface, design a web user interface, develop the user interface, test the user interface for functionality, and then release the user interface for integration with future releases. Others iterations will be concerned with implementing the recipe web service, recipe search, user accounts, and utility functionality.

# **Deliverables**

There are six deliverables associated with the Agile methodology (Muslihat).

1. **Product vision statement (**Schuurman**): In this document, the purpose of our web application is defined. In our scenario, the purpose of the application is to provide a recipe information on a website through a web service.**
2. **Product roadmap: In this document, the “hows” of satisfying the product vision are defined? All the requirements and the ways to achieve them are mentioned. In our scenario, the fulfillment section in this document describes the ways we will satisfy the business requirements.**
3. **Product Backlog: In this document, we will mention all the individual tasks that need to be completed in order to produce the application. For our scenario, a separate document will be created that contains the individual tasks that are required.**
4. **Release Plan:** In this document, the timetable of the project will be defined. In our scenario, the Project Timeline section will contain the release plan for this project.
5. **Sprint Backlog: In this document, the tasks and goals for a particular sprint are defined. In our scenario, separate documents will keep track of each sprint/iteration.**
6. **Increment:** The result of the sprint is defined in this deliverable. In our scenario, after each sprint, a working component for the application will be produced and integrated with the results of other sprints.

# **Implementation**

The implementation of this project will be done as a cooperative effort with Simple Food Solutions and their third-party hosting provider. After the software is completed, validation and verification will take place and then the resulting enterprise archive will be deployed to the third-party hosting servers using a FTP client. The project managers, web developers, software developers, and database developers, and third-party hosting administrators will oversee the process to ensure that everything goes smoothly. The project manager will recheck the documentation to ensure requirements are met. The different developers will provide guidance for any errors encountered due to development. The administrators will make sure that the required production systems are operational. The estimated time to deploy the application will be 5 hours.

# **Validation and Verification**

Since the agile methodology is being followed, emphasis is given on involving the customer throughout the SDLC. Constant validation will be used with the customer in order to ensure that the right product is being built. In order to perform verification, the product and sprint backlog will be compared to the product roadmap in order to ensure that the product being built conforms to the specifications. Unit, usability, and user acceptance testing will be the primary methods of testing for this application. Simple Food Solutions developers and selected customers will partake in the testing process.

# **Environments and Costs**

## **Programming Environment** (VPSCheap Team)

**For web application:**

* Software: CentOS 7 running TomcatEE
* Hardware: VPS host with 4GB RAM, 4 core processor, and 70GB SSD storage. The specific hardware details such as the models of the hardware components are not known.

**For web service:**

* Software: CentOS 7 running TomcatEE
* Hardware: VPS host with 4GB RAM, 4 core processor, and 70GB SSD storage. The specific hardware details such as the models of the hardware components are not known.

## **Environment Costs** (VPSCheap Team)

The environmental costs include web application hosting costs which are about $150 dollars a year. If Simple Food Solutions determines that more resources are needed, they have the option to upgrade to an enterprise plan which would cost about $340 dollars a year. Other add-ons are also available as needed such as daily backups ($60 year), premium support ($300 year), software control panel-CPanel ($168 year).

## **Human Resource Requirements**

The time to complete this application is about eight weeks. The human resources required for this project include developers, designers, and a project manager. The cost for these human resources will be their yearly salaries. Designers will be present during all design activities and will consume approximately 20% of the total time. Developers will be present during the development and testing activities and will consume about 60% of the total time. The project manager will be present during the requirement activities as well as other appropriate activities in order to ensure the project is on track and will consume about 20% of the time.

# **Project Timeline**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase/Sprint | Milestone/Task | Deliverable | Description | Dates |
| Pre-development | Task 1 | Product Vision Statement | Meeting with customer and understand business problem and determine solutions and goals. | 4/5 – 4/9 |
|  | Task 2 | Product Roadmap |  | 4/10-4/12 |
|  | Task 3 | Product Backlog |  | 4/13-4/14 |
|  | Task 4 | Release Plan |  | 4/15-4/17 |
|  | Task 5 | Spring Backlog |  | 4/18-4/21 |
| Sprint 1 – Recipe Web Service | Requirements | Review Sprint Backlog | Determine what must be done for this sprint (create the web service) | 4/22 |
|  | Design | UML Design Diagrams | Design necessary classes such as resource classes and Data Access Objects for web service | 4/23 – 4/24 |
|  | Develop | RESTful Web Service | Develop the web service | 4/25 -4/28 |
|  | Test | Functional Testing | Test the web service with Postman API Tool | 4/29 |
|  | Release | Web service | Host the web service as a web application archive | 4/30 |
| Sprint 2 – Web User Interface | Requirements | Review Sprint Backlog | Determine what must be done for this sprint (create the user interface) | 5/1 |
|  | Design | Wireframe | Design necessary pages such as search, results, login, and user pages | 5/2 – 5/3 |
|  | Develop | Website prototype | Create a prototype using XHTML, CSS, and Javascript | 5/4 – 5/7 |
|  | Test | Usability testing | Let users try out the prototype and determine if changes are needed | 5/8 – 5/9 |
|  | Release | Website | Add website to codebase | 5/10 |
| Sprint 3 – Recipe Search Functionality | Requirements | Review Sprint Backlog | Determine what must be done for this sprint (create functionality for recipe search) | 5/11 |
|  | Design | UML Diagrams | Create appropriate class, use case, and activity diagrams | 5/12 – 5/13 |
|  | Develop | Recipe search module | Functionality that allows users to search for recipes provided via the web service | 5/14 – 5/16 |
|  | Test | Functional testing | Test the functionality of searching for recipes | 5/17 - 5/18 |
|  | Release | Recipe search module | Integrate recipe search module with web user interface | 5/19 |
| Sprint 4 – User Accounts functionality | Requirements | Review Sprint Backlog. | Determine what must be done for this sprint (create functionality for user accounts). | 5/20 |
|  | Design | UML Diagrams,  ERD Diagram | Create appropriate class, use case, and activity diagrams. Create ERD diagram for database tables | 5/21 – 5/22 |
|  | Develop | User Account module | Functionality that allows users to create and log into accounts | 5/23 -5/28 |
|  | Test | Functional testing | Testing the functionality to create and log into accounts | 5/29 -5/30 |
|  | Release | User Account Module | Integrate user account module with existing components in the codebase | 5/31 |
| Sprint 5 –  Reporting Functionality | Requirements | Review Sprint Backlog. | Determine what must be done for this sprint (create functionality for reports). | 6/1 |
|  | Design | UML Diagrams,  ERD Diagram | Create appropriate class, use case, and activity diagrams. Create ERD diagram for database tables | 6/2 – 6/3 |
|  | Develop | Report module | Functionality that allows users to generate reports | 6/4 -6/7 |
|  | Test | Functional Testing | Test the functionality to generate reports | 6/8 -6/9 |
|  | Release | Cookbook module | Integrate report module with existing components in codebase. | 6/10 |
| Post Development | Deploy | Web Archive (war) | Deploy web archive to server | 6/ 11 |

# **Sources**

Muslihat, Dinnie. “Agile Methodology: An Overview.” *Zenkit*, 2 Mar. 2018, <zenkit.com/en/blog/agile-methodology-an-overview/>.

Schuurman, Robbin. “10 Tips for Product Owners on the Product Vision.” *Scrum.org*, 29 Nov. 2017, [www.scrum.org/resources/blog/10-tips-product-owners-product-vision](http://www.scrum.org/resources/blog/10-tips-product-owners-product-vision).

VPSCheap Team. *Unmetered SSD and Budget VPS Hosting | VPSCheap.NET*, <www.vpscheap.net/pricing.aspx>.