Software Development Plan

Ver 2.0

Sheepdog Defense Group

Revision History

Date	Version	Description	Author
09/21/18	*1.0*	*Initial Draft*	*Hayden, Phong, Muoi*
03/04/19	*1.1*	*Update Risk Management and Project Plan*	*Phong*
05/04/19	*2.0*	*Reworked various layout issues, sizing and space inconsistencies, and phrasing throughout entire document.*	*John Hodnett*

Sheepdog Defense Group

Table of Contents

Software Development Plan

1.   Introduction

1.1   Purpose

The purpose of the *Software Development Plan* is to gather all information necessary to control the project. It describes the approach to the development of the software and is the top-level plan generated and used by project managers to direct the development effort.

The following people use the *Software Development Plan*:

- The **project manager** uses it to plan the project schedule and resource needs, and to track progress against the schedule.

- **Project team members** use it to understand what they need to do, when they need to do it, and what other activities they are dependent upon.

1.2   Scope

This *Software Development Plan* describes the overall plan to be used by the SheepDog project, including deployment of the product. The details of the individual iterations will be described in the Iteration Plans document.
The plans as outlined in this document are based upon the product requirements as defined in the *Vision Document* and Use Cases.

1.3   Definitions, Acronyms, and Abbreviations

See the Project Glossary.

1.4   References

- [Vision](#) [1]

- [SRS](#) [2]

- [Glossary](#)[3]

1.5   Overview

This *Software Development Plan* contains the following information:

Project Overview — provides a description of the project's purpose, scope, and objectives. It also defines the deliverables that the project is expected to deliver.

Project Organization — describes the organizational structure of the project team.

Management Process — explains the estimated cost and schedule, defines the major phases and milestones for the project, and describes how the project will be monitored.

Applicable Plans and Guidelines — provides an overview of the software development process, including methods, tools and techniques to be followed.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

The objective of SheepDog is to provide a website to simplify the registration and purchase process for customers and class organization for the Administrator. It provides necessary document and login page for the Guards as well as holds their personal information. The project will provide a web interface for Administrator, Guards, and customers to enter data, retrieve documents and information.

2.2 Assumptions and Constraints

Customer has no experience in web development. Therefore, SheepDog will find system maintenance challenging after the development team graduates from TCU.Also, the system stores confidential information so it is difficult for the team to encrypt the database.

2.3 Project Deliverables

The following deliverables will be produced during the development of this project:

Software Development Plan

Vision Document

Requirements Document

Use Cases

Glossary

Iteration Plans

User Manual

Development Guide

2.4 Evolution of the Software Development Plan

The *Software Development Plan* will be revised prior to the start of each Iteration phase. The target dates for

the end of each phase and iteration are shown below in section 4.2.4

3. Project Organization

3.1 Organizational Structure

Team: Hayden Southworth (Team Lead), Muoi Pham, John Hodnett, Tek Ghimire, Phong Nguyen.

Advisor: Dr. Bingyang Wei.

SDG Owner: David Riggall.

3.2 External Interfaces

Name	Role	Contact Information
Dr. Donnell Payne	Project Supervisor	d.payne@tcu.edu

| Dr. Bingyang Wei | Advisor | b.wei@tcu.edu |
| David Riggall | Client | david@sheepdogdefensegroup.com |

3.3   Roles and Responsibilities

Member	Role	Role Description
Hayden Southworth	Project Lead	Supporting and reviewing project development.
Phong Nguyen	Back-end Developer	Building and maintaining application server and database.
Muoi Pham	Full-stack Developer	Handling application design and deployment.
Tek Ghimire	Testing Lead	Testing the limits of software through development.
John Hodnett	Front-End Developer and Quality Assurance Engineer	Handling the front-end UI design as well as testing to make sure everything is functioning properly and bug-free.

4.   Management Process

4.1   Project Estimates

First Iteration - December 2018.

4.2   Project Plan

4.2.1   Phase Plan

4.2.2   Iteration Objectives

Iteration I:

Customer Sign up for class

Admin Create class

Research Google Calendar API

Research ActiveAdmin for user account

Create responsive homepage, Super Admin login page, Admin and Guard Edit Profile page, Users signup page, Class create page.

Integrate UI to existing website

Integrate Google Calendar functions to the website

Create image slideshow with Javascript

User signup function for testing

Function to edit user role for Super Admin

Input field validation

Implement the Users table, admin table in the database

Iteration II:

Implement the payment method with Stripe

Switch to FullCalendar. Google Calendar is no longer used.

Admin create guard login

Guard modify personal ID photo

Customer's transaction email

Iteration III:

Admin update merchandise

Customer buy merchandise

Admin verify gun license

Iteration IV:

Customer request consult

Customer provide feedback

UI and compatibility improvement

Deployment

4.2.3 Releases

The website was deployed on March 18th, 2019.

4.2.4 Project Schedule

Completion Date	Task
04/Nov/18	*Finalizing Project Requirement*
25/Nov/18	*Iteration I*

10/Feb/19	*Iteration II*
10/Mar/19	*Iteration III*
7/Apr/19	*Iteration IV*
27/Feb/19	*SRS Registration Deadline*
29/Mar/19	*SRS Poster Deadline*
12/Apr/19	*Day of SRS*
02/May/19	*Final Presentation*

4.3   Project Monitoring and Control

4.3.1   Requirements Management

The requirements for this system are captured in the Vision document. Requested changes to requirements documented by the team lead and sent to client for approval or clarification.

4.3.2   Schedule and Budget Control

The team lead keeps track of the weekly schedule for each team member. This schedule include what each team member is planning on finishing during that time and any open issues that they have run into. Each team member reports their progress on their respective tasks at a weekly team meeting. Changes to the overall scope of the iteration and the weekly schedules will be based upon the progress seen during these meetings and how to best optimize the progress to complete the project by the complete deadline.

There are no expenses expected until deployment. Once deployed, Heroku will cost an estimated $7/month. Amazon AWS S3 bucket is expected to stay within the free tier limit. In the even of going over the limit, the charges will be minimal (<$10/month). Our client has approved all of the above listed expenses.

4.3.4   Quality Control

During the development process, we will be conducting regular test to ensure the user features are working was intended. Once, development has reached the end of iteration 3, we will deploy the site on Heroku under a obscure domain and provide it to the client to have them perform client acceptance test and inform us of any issues they might encounter.

4.3.5   Reporting and Measurement

Updated cost and schedule estimates, and metrics summary reports, will be generated at the end of each iteration.

The Minimal Set of Metrics, as described in the RUP Guidelines: Metrics, will be gathered on a weekly basis. These include:

Earned value for completed tasks. This is used to re-estimate the schedule and budget for the remainder of the project, and/or to identify need for scope changes.

Total defects open and closed – shown as a trend graph. This is used to help estimate the effort remaining to correct defects.

Acceptance test cases passing – shown as a trend graph. This is used to demonstrate progress to stakeholders.

In addition, overall costs will be monitored against the project budget.

4.3.6   Risk Management

Risk Ranking	Risk Description and Impact	Mitigation Strategy and/or Contingency Plan
Medium	Google Calendar API is not very developer friendly	FullCalendar is implemented and class info is kept in database.
Low	Email via Sendgrid	Email might go to spam box instead of inbox. If the recipient adds sender email to their address list or mark the first email as not spam then it does not happen anymore.
Medium	Paypal API is not very developer friendly	Stripe payment is implemented instead of Paypal.
High	Heroku automatic server restart	Either pay for the higher tier plan or change the heroku error page so visitor knows to refresh the page.
Medium	Domain redirect with different url.	"www" must be in the url to access the website. Godaddy doesn't support Heroku's domain redirect so we might need to rely on a third party to manage DNS.

4.3.7   Configuration Management

We will use Bitbucket to provide a database of Change Requests and a controlled versioned repository of project artifacts.

All code pushed to the repository will be pull down by every member of the team and reviewed for any bugs or issues that arise during development.

When all known bugs and issues are resolved, we will push to code to the live site on Heroku and conduct client acceptance tests. Heroku then performs twice daily server restarts.

5.   Annexes

The project will follow the RUP for Small Projects process, as tailored by the project Development Case. Other applicable process plans are listed in the references section, including Programming Guidelines.