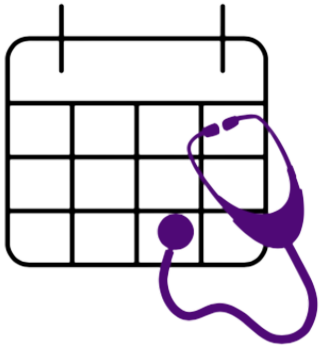
TCU & UNTHSC Volunteer Scheduler



Software Development Plan

Version 1.1

Revision History

Date	Version	Description	Author
04/18/2021	1.0	Created document	Maria Amoros
05/04/2021	1.1	Added section headers based on last year's Mercy Clinic project, filled in the sections	Lydia Pape

Table of Contents

Introduction **4**

1.1 Purpose 4

1.2 Scope 4

Project Overview **4**

2.1 Purpose, Scope, Objectives 4

2.2 Assumptions and Constraints 4

2.3 Project Deliverables 5

2.4 Evolution of the Software Development Plan 5

Project Organization **5**

3.1 Organizational Structure 5

3.2 External Interfaces 5

3.3 Roles and Responsibilities 5

Management Process **6**

4.1 Project Estimates 6

4.2 Project Plan 7

4.2.1 Phase Plan 7

4.2.2 Project Resourcing 8

4.3 Project Monitoring and Control 8

4.3.1 Requirements Management 8

4.3.2 Schedule and Budget Control 9

4.3.3 Quality Control 9

4.3.4 Reporting and Measurement 9

4.3.5 Risk Management 9

4.3.6 Configuration Management 10

1. Introduction

1.1 Purpose

The purpose of the *Software Development Plan* is to gather all information necessary for the management of the TCU & UNTHSC Volunteer Scheduler. It should be used by the dev team to collaborate on task management.

1.2 Scope

This document describes the overall plan of development and deployment of the SoM Volunteer System.

2. Project Overview

2.1 Purpose, Scope, Objectives

The project's purpose is to provide an online tool which the SoM volunteers and their administrators can use to easily navigate the process of signing up for volunteering opportunities and having volunteer hours recorded as necessary for academic purposes. This will be accomplished using a web-based system that can be accessed through the browser.

2.2 Assumptions and Constraints

2.2.1 We assume that any costs associated with the deployment, maintenance, and compatibility of our system with 3rd-party components (such as Firebase) will be approved and covered by the TCU & UNTHSC SoM.

2.2.2 We assume that any further maintenance on the system beyond the release at the end of the spring 2021 semester will be handled by staff in the TCU & UNTHSC SoM.

2.2.3 We assume that all users of the system will have access to the necessary equipment/hardware (computers and/or smart phones with browsers and internet access).

2.2.4 We assume that administrators of the system will be responsible for all information stored in the system's database.

2.3 Project Deliverables

All deliverables will be completed and produced by the end of the spring 2021 semester.

2.4 Evolution of the Software Development Plan

This is a general plan for the phases of the development of the system, subject to change throughout the software development process.

Inception	September 1st - September 30th
Elaboration	October 1st - October 14th
Construction Iteration 1	October 15th - December 20th
Construction Iteration 2	January 1st - March 14th
Construction Iteration 3	March 15th - April 14th
Deployment & Testing	April 15th - April 30th

3. Project Organization

3.1 Organizational Structure

This team comprises 6 software developers, each with more specific areas of expertise. One of these is also given the responsibility of Team Lead, meaning they are responsible for overseeing the other team members' work, collaboration, and organization on the project. There is also a professor who assumes the role of Project Manager, helping the team stay on schedule and accomplish tasks to acceptable standards of quality.

3.2 External Interfaces

3.2.1 Deployment of the software and database are handled automatically by Heroku.
3.2.2 System authentication of users is handled by Firebase, which interfaces with both the front end and back end as appropriate for this requirement.
3.2.3 Further maintenance of the system is to be handled by staff at the TCU & UNTHSC SoM as necessary.

3.3 Roles and Responsibilities

Person	**Roles and Responsibilities**

Dr. Binyang Wei, Project Manager	Project Manager Reviewer for project architecture Reviewer for project requirements Reviewer for all documentation Tester Consultant on various technical questions
Maria Amoros, Team Lead & Front End Developer	Manager for project structure and plans Caller of meetings Keeper of deadlines Front End Developer Tester Front End Bug Fixer Speaker at SRS Presentation
Jeshua Suarez-Lugo, Website Manager & Front End Developer	Manager of the project's Rio Grande page Keeper of documentation available in the Rio Grande Page Front End Developer Tester Manager of project deployment
Peyton Freeman, Full Stack Developer	Full Stack Developer Tester Bug Fixer
Riley Durbin, Technical Lead & Back End Developer	Leader on technical issues with the project structure and deployment Aid in project deployment Back End Developer Tester Full Stack Bug Fixer
Lydia Pape, Back End Developer	Back End Developer Tester Full Stack Bug Fixer Note taker at meetings with the client
Emery Wolf, Back End Developer	Back End Developer Tester Back End Bug Fixer

4. Management Process

4.1 Project Estimates

The cost of the domain name tcusomservice.com for the system is $19.99.

The cost of deployment with Heroku, including the database, is $7/month.

These costs are expected to remain the same until and unless further upgrades/changes are needed for the continued maintenance and functionality of the system toward its usefulness to the TCU & UNTHSC SoM.

4.2 Project Plan

4.2.1 Phase Plan

Phase	Deliverable	Description	Objectives and Releases	To Be Completed By
Inception	Pre-iteration planning	The team met over the course of a few weeks to develop early plans for the structure and form of the project and divide early tasks between members.		
Elaboration	Use Cases	The team settled on a general project structure and documented use cases.	Have a plan and be ready to move forward with software development.	10/01/20
Development	Prototype, Iteration 1	Development of a prototype, which will also serve as an early version of the project's front end. Begin work on the back end and database structure.	The prototype and early progress to be presented to Advisory Board members at the end of the fall semester.	11/15/20
	Iteration 2	Finish development of features	Beta release will have as many	02/15/21

		corresponding to functional requirements.	functional requirements fulfilled as possible.	
Testing	Iteration 3	Bug Fixes; connect the system with Firebase; deploy the project to Heroku, making it available to the client for acceptance testing.	The system will be deployed and ready for early use by the client and assigned SoM student testers.	03/25/21
Evaluation	Final Product	Bug fixes; final improvements as requested by the client.	The software will be finished and ready to be used by the client long-term.	04/20/21

4.2.2 Project Resourcing

Team members will remain the same throughout the course of the above plan. Every team member must make an effort to learn the frameworks and languages for the front end and back end so that as many people can work on as many different parts of the software as possible.

4.3 Project Monitoring and Control

4.3.1 Requirements Management

The team demos working software and prototypes to the client as often as possible, and asks requirements questions of the client as they come up. Changes to the software and/or documentation are made throughout the development process in accordance with the client's requests for changes and responses to questions.

4.3.2 Schedule and Budget Control

The client will approve and handle any required expenses on behalf of the SoM. Costs are not expected to be at all high; if expenses for the project start to become too high, then the team and client will reevaluate as necessary.

4.3.3 Quality Control

The client and Project Manager will assess the quality of the progress on the software at each demonstration. The dev team will note their suggested or required changes and work on them in the following iteration, or later in the current iteration, as necessary.

The project manager will also review all elements of the software and other deliverables, suggesting improvements along the way so the team can finally turn in their best work.

4.3.4 Reporting and Measurement

After each iteration, the team will reevaluate what needs to happen in the next iteration to ensure that all deliverables are completed by the corresponding deadlines.

4.3.5 Risk Management

Risks are to be identified in the inception phase and reevaluated during each iteration as necessary.

Severity	**Likelihood**	**Description**	**Mitigation Strategies**
High	Medium	Scheduling flaws (inability to finish deliverables in a timely manner).	Plan ahead and start early in case some additional bugs or features need to be fixed or developed.
Low	Low	One or more members for some reason leave TCU and other members will have to take on their tasks.	Starting early and trying to develop as many functionalities as possible, so testing can take over.
Medium	High	Need to deploy in the cloud or any other platform that may have a cost associated	Talk with our client in advance to check that the cost of deployment (or

		with it.	any other technology that may be necessary) is affordable for the SoM.
Medium	High	As we are still students, we may not be experienced developers or project managers.	Ask for help from people with more experience. We count on Dr. Wei who is an expert on software development, and if we have specific technical difficulties, Dr. Wei might help us or refer us to an adequate faculty member or person in the field.
High	Low	Some natural disaster or any other event may occur that stops the development of the project.	Always plan ahead.

4.3.6 Configuration Management

Github repositories will be used for version control and collaboration across multiple team members' computers.

The beta system will be developed and tested using a local, temporary H2 database.

All documentation will be updated as the development process progresses.

Team members will review each other's code as time allows, and team members will meet to collaborate on hard problems as necessary.

Team members will communicate over Slack messaging for scheduling meetings and minor questions or requests for changes to the software that might be subject to opinion (i.e. everything other than bug fixes or obvious improvements).