Software Project Management Plan

Dwebble

a Classroom Scheduling Application

2020.03.08

Group 12
Team Members

Ibrahim AbuAnz
Elias Garcia
Aiden Cornstubble

Document Control

Change History

Revision	Change Date	Description of changes
V1.0	2020.03.05	Initial release for review
V1.1	2020.03.07	Final review and edits before release submission

Table of Contents

1 Overview

1.1 Purpose and Scope

Purpose

The purpose of the project is to reduce the complexity and human-effort required to prepare classroom schedule for an educational institution. There are several constraints, preferences, and rules to follow while making such schedules. For a human user, it becomes a time-consuming and highly complex task to prepare accurate schedules based on those constraints and rules. Hence, the Classroom Schedule web application, named ***Dwebble***, would be useful to make the schedule preparation faster, and prone to significantly less errors. Thus it will help in saving time, effort, and cost.

Scope

The scope of the project is limited to development of the proposed classroom scheduling solution for only a single *department* of the university, the Department of Computer Science and Electrical Engineering. The project will be focused on exploring different aspects of a classroom scheduling process and to prepare weighed schedules possible under a given scenario so that an administration can select a suitable schedule from a list of schedules prepared by the system. However, personalization or customization of the system is not under the scope of the project and no other aspect of a university process is under the scope of this project.

The exclusions include, but are not limited to, the scheduling needs of:

- Individual students
- The specific sequencing requirements of classes themselves
- Inter-departmental scheduling conflicts and coordination

A best attempt effort will be made to leave the delivered system with a design and documentation such that Dwebble’s functionality could be extending to include such scheduling needs as a *separate* project in *the future*.

1.2 Goals and Objectives

Goals

1. Developing a generic Classroom Schedule application accessible through web and mobile platforms.
2. Allowing administrators and staffs to set rules, constraints, and preferences of professors to create multiple weighted schedules for an upcoming semester that account for the aforementioned rules, constraints, and preferences.

Objectives

1. Creating a backend database containing all rules, preferences, constraints, and schedules
2. Create a backend schedule resolver that can take in known rules, preferences, and constraints to generate potential schedules or, when impossible, a list of known conflicts for administrators to resolve.
3. Creating schedules dynamically based on changing rules and preferences. New data gets reflected in the next generated set of schedules.
4. Creating user-friendly interfaces for different types of end-users so that they can access the system platform-independently.
5. Create a front-end application that allows different sets of users to access the application with different levels of permissions.
6. Create an application that allows browser access from different sets of devices, i.e. smartphones, tablets, and desktop browsers.

1.3 Project Deliverables

The project deliverables of the Classroom Scheduling application development project are to be delivered <u>on or before</u> **2020.05.04**:

1. Source code for the entire application, including:
   i. Front-end
   ii. Application Server
   iii. Database and database migrations
   iv. Infrastructure
   v. Deployment
2. Users Guide
3. Detailed test plan with test outcomes
4. Detailed system test cases
5. Detailed results of regression testing
6. Execution of detailed data migration from paper based system to the information system
7. System administration manual

1.4 Assumptions and Constraints

The assumptions for the system are,

1. The test environment containing software and hardware for the proposed information system will contain all required support and data to develop the system
2. The hardware containing the RAID servers and Linux servers will host the database server and application servers on a cloud platform
3. The client will provide an office space with required amenities to support the system development process
4. All members of the project will be available from the beginning to the completion of the project based on their roles and skills.

The constraints for the project are,

1. The system will be access from web and mobile platforms
2. The database will be developed using open source solutions
3. Cloud computing platform will be used for the project and the subscription cost will be handled by the client who will use the solution.
4. The software should be developed and available for the client to use by 05/7/20

1.5 Schedule and Budget Summary

All scheduling and budgetary estimates are provided in further detail in later sections below, along with a supplementary Gantt chart to visualize the timeline of deliverables.

Schedule

The project begins, in earnest, on Monday, 2020.02.17 and proceeds in approximately 2 week intervals for each iteration. By Monday, 2020.03.16, a project prototype will be delivered and presented to stakeholders to demonstrate progress and receive feedback on development. By Friday, 2020.04.03, the architecture and design of the project will largely be finalized with the delivery of the Architecture Document. The final product will be delivered and released by the end of iteration 5 on Monday, 2020.05.04. Feedback on the final deliverable will be received by Saturday, 2020.05.09. The total project duration is approximately 92 days, slightly short of the average UMKC spring semester.

Budget

The project budget estimate comes in at approximately $72,000, with the primary cost input being system development at approximately

$48,000.00. While iterations vary slightly in duration, each will incur a cost of approximate $9,000 on average over all five.

1.6 Success Criteria

The success criteria of the project have two purposes. It will help the developers and other project members to focus on the project scope and it will also help in the verification and validation process during the project closure.

The project success criteria to consider in this project are,

1. The total cost of the project should be within the project budget.
2. The milestones of the project must be met as presented in the project schedule
3. There will be multiple iterations of the project. The main functions must be completed within the 4 iterations of the project.
4. The system should be up and running before the closure of the project.

1.7 Definitions

Project Sponsor: Responsible to take decisions about the project on behalf of a client. A client and a project sponsor may be the same or different persons or organizations. A project sponsor is responsible for the success or failure of a project and has the highest authority after the client.

Client: A client is an organization or individual who is bearing the project cost and has asked to develop the system.

1.8 Evolution of the Project Plan

The first iteration of the project will start from the initial requirements specified in the initial system requirements specification document. At the end of the first iteration, there will be a review phase and that may have changes in the initial requirements within the scope of the proposed system. So, in the second iteration, the requirement specification document will be updated based on the review and those will be handled in the second iteration. At the end of each iteration, there will be a conclusion step that will summarize the effort and cost spent in that iteration.

The risk management process will start from the beginning of the project. The evaluation of the risk mitigation plan will start from the beginning of each iteration. If any risk evolves during the project, then that risk will be analyzed as soon as possible with proper risk mitigation strategies.

2 Startup Plan

2.1 Team Organization

The project team organization details in terms of responsibilities, and authorities assigned to different project team members have been summarized in the following table.

Team Members/ Roles	**Explanations**
Project Manager	The project manager is responsible to carry out the project successfully. Some key responsibilities of the project manager are, planning for the project schedule, budget, risk management, communication, control and overall project team management. The project manager must report to the project sponsor and client. The project manager is given some authority to manage minor change requests to the project, evaluation of risks.
System Analyst	The system analyst is responsible to collect system requirements details and to prepare and update system requirement specification details in each iteration.
Developers	The developers are responsible for the coding and unit testing for the project components and modules. They will be responsible to provide unit testing results to the testers also.
Testers	Testers are responsible to prepare test cases, carrying out regression testing, integration testing, and system testing, and to document test results. Testers are also responsible to verify and validate whether the system meets all the requirements or not.
Designer	A designer is responsible to prepare interface designs for the system and deliver the same to the developer so that developers can integrate front end designs with the back end system.
Technical Writer	A technical writer is responsible to prepare system manual and other system related documentation for the end users.
IT Staffs	General IT staffs are responsible to set-up the development environment and to help in general troubleshooting with the network and computing environment used in the project.

2.2 Project Communications

The project communication plan is,

Communication name	Frequency	Method(s)	Targeted audience	Owner
Project team meeting	Weekly	Meeting	Project team	Project manager
Stakeholder update	Daily basis	Slack	All project team members	Project manager
Project sponsor meeting	Bi-monthly	Meeting	Project sponsor	Project manager
Development instructions	As and when required	Email	Designer, developers, testers	Project manager
Project updates	Weekly	Emails, reports	Project manager	Designer, developers, testers

2.3 Technical Process

The detailed information about the system development process for the proposed Classroom Scheduling application following the agile methodology have been summarized in the following table.

Phase	Entry Criteria	Exit Criteria	Major Milestones	Workflow	Other aspects
Project Initiation	The project sponsor and project manager have been assigned	The project Initiation document is signed by a project sponsor	Creation of PID	Project manager carries out feasibility study, prepares initial planning and project initiation document. Project sponsor reviews,	NA

				approves, and signs the document to start the project.	
Requirements Gathering	PID is approved	Requirements specification document is approved by project sponsor	Creation of SRS, Formation of the project team, Setting up the development environment	System analyst creates the SRS, same is approved by the project sponsor, external vendors are contracted to set-up the development environment for the project.	NA
System Design	SRS is available	System design document	Functional, architectural, and user-interface designs are prepared	Designers prepare interface designs, system analyst prepares functional and architectural designs of the proposed system.	NA
System Development	System design document is	Source code and unit testing	Source code and unit test results	Developers write the code and unit test	NA

	available	results are available	available through multiple iterations	each module. All results of the unit testing are documented.	
Testing	Source code and unit test results available	System verification and validation details are available	System test details are available	Testers follow the system test plans and prepares the test results.	NA
Project Closure	System development and testing is completed successfully	Project report is submitted	Client accepts the project report or project sponsors signs the completion of the project	Project manager submits the project report and accepted by the client	NA

2.4 Tools

- Programming Languages
  - Primarily Rust, with supplementing JS, HTML, and CSS for front-end.
  - PostgreSQL for database management.
- Version control: Git with hosting on GitHub
- Defect and feature tracking: GitHub’s issue and KanBan integrations.
- Build tools:
  - Rust’s built-in Cargo, a package manager and project management tool for Rust based projects.
- Automated testing:
  - Rust’s built-in and first class support for Unit-Testing, which can be managed and ran by Cargo without third-party tooling.
  - GitHub Actions for CI/CD.

  - ◦ Docker for isolating development environments and deployment of code.
- Hosting and Deployment
  - ◦ DigitalOcean for server hosting.
  - ◦ Nginx for reverse-proxy and load balancing.
  - ◦ GNU/Linux, specifically *Ubuntu*, for server OS.

3 Work Plan

3.1 Activities and Tasks

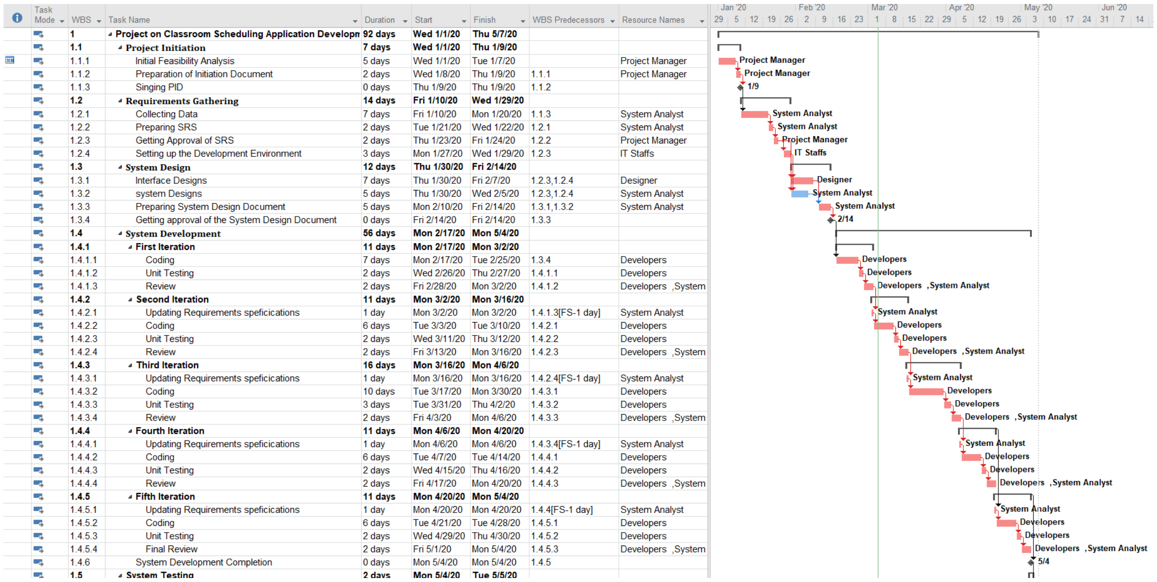
The work breakdown structure or WBS for the project is given below.

WBS	Task Name	Duration	Start	Finish	WBS Predecessors	Resource Names
1	**Project on Classroom Scheduling Application Development**	**92 days**	**Wed 1/1/20**	**Thu 5/7/20**		
1.1	**Project Initiation**	**7 days**	**Wed 1/1/20**	**Thu 1/9/20**		
1.1.1	Initial Feasibility Analysis	5 days	Wed 1/1/20	Tue 1/7/20		Project Manager
1.1.2	Preparation of Initiation Document	2 days	Wed 1/8/20	Thu 1/9/20	1.1.1	Project Manager
1.1.3	Singing PID	0 days	Thu 1/9/20	Thu 1/9/20	1.1.2	
1.2	**Requirements Gathering**	**14 days**	**Fri 1/10/20**	**Wed 1/29/20**		
1.2.1	Collecting Data	7 days	Fri 1/10/20	Mon 1/20/20	1.1.3	System Analyst
1.2.2	Preparing SRS	2 days	Tue 1/21/20	Wed 1/22/20	1.2.1	System Analyst
1.2.3	Getting Approval of SRS	2 days	Thu 1/23/20	Fri 1/24/20	1.2.2	Project Manager
1.2.4	Setting up the Development Environment	3 days	Mon 1/27/20	Wed 1/29/20	1.2.3	IT Staffs
1.3	**System Design**	**12 days**	**Thu 1/30/20**	**Fri 2/14/20**		

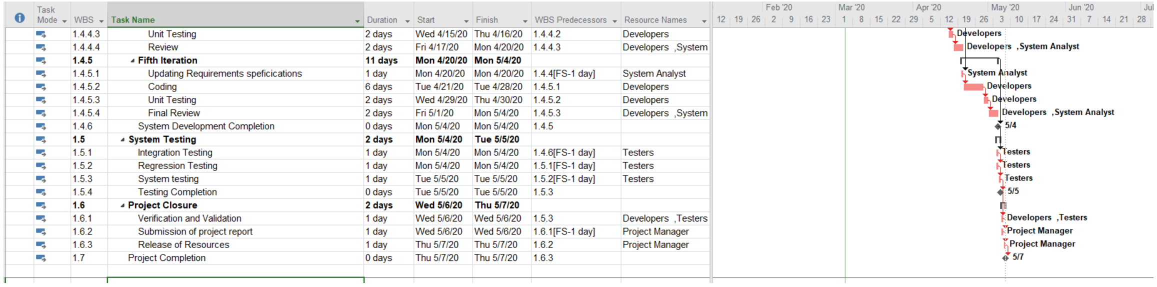
1.3.1	Interface Designs	7 days	Thu 1/30/20	Fri 2/7/20	1.2.3,1.2.4	Designer
1.3.2	system Designs	5 days	Thu 1/30/20	Wed 2/5/20	1.2.3,1.2.4	System Analyst
1.3.3	Preparing System Design Document	5 days	Mon 2/10/20	Fri 2/14/20	1.3.1,1.3.2	System Analyst
1.3.4	Getting approval of the System Design Document	0 days	Fri 2/14/20	Fri 2/14/20	1.3.3	
1.4	**System Development**	**56 days**	**Mon 2/17/20**	**Mon 5/4/20**		
1.4.1	**First Iteration**	**11 days**	**Mon 2/17/20**	**Mon 3/2/20**		
1.4.1.1	Coding	7 days	Mon 2/17/20	Tue 2/25/20	1.3.4	Developers
1.4.1.2	Unit Testing	2 days	Wed 2/26/20	Thu 2/27/20	1.4.1.1	Developers
1.4.1.3	Review	2 days	Fri 2/28/20	Mon 3/2/20	1.4.1.2	Developers ,System Analyst
1.4.2	**Second Iteration**	**11 days**	**Mon 3/2/20**	**Mon 3/16/20**		
1.4.2.1	Updating Requirements speficications	1 day	Mon 3/2/20	Mon 3/2/20	1.4.1.3[FS-1 day]	System Analyst
1.4.2.2	Coding	6 days	Tue 3/3/20	Tue 3/10/20	1.4.2.1	Developers
1.4.2.3	Unit Testing	2 days	Wed 3/11/20	Thu 3/12/20	1.4.2.2	Developers
1.4.2.4	Review	2 days	Fri 3/13/20	Mon 3/16/20	1.4.2.3	Developers ,System Analyst
1.4.3	**Third Iteration**	**16 days**	**Mon 3/16/20**	**Mon 4/6/20**		
1.4.3.1	Updating Requirements speficications	1 day	Mon 3/16/20	Mon 3/16/20	1.4.2.4[FS-1 day]	System Analyst

1.4.3.2	Coding	10 days	Tue 3/17/20	Mon 3/30/20	1.4.3.1	Developers
1.4.3.3	Unit Testing	3 days	Tue 3/31/20	Thu 4/2/20	1.4.3.2	Developers
1.4.3.4	Review	2 days	Fri 4/3/20	Mon 4/6/20	1.4.3.3	Developers ,System Analyst
1.4.4	**Fourth Iteration**	**11 days**	**Mon 4/6/20**	**Mon 4/20/20**		
1.4.4.1	Updating Requirements speficications	1 day	Mon 4/6/20	Mon 4/6/20	1.4.3.4[FS-1 day]	System Analyst
1.4.4.2	Coding	6 days	Tue 4/7/20	Tue 4/14/20	1.4.4.1	Developers
1.4.4.3	Unit Testing	2 days	Wed 4/15/20	Thu 4/16/20	1.4.4.2	Developers
1.4.4.4	Review	2 days	Fri 4/17/20	Mon 4/20/20	1.4.4.3	Developers ,System Analyst
1.4.5	**Fifth Iteration**	**11 days**	**Mon 4/20/20**	**Mon 5/4/20**		
1.4.5.1	Updating Requirements speficications	1 day	Mon 4/20/20	Mon 4/20/20	1.4.4[FS-1 day]	System Analyst
1.4.5.2	Coding	6 days	Tue 4/21/20	Tue 4/28/20	1.4.5.1	Developers
1.4.5.3	Unit Testing	2 days	Wed 4/29/20	Thu 4/30/20	1.4.5.2	Developers
1.4.5.4	Final Review	2 days	Fri 5/1/20	Mon 5/4/20	1.4.5.3	Developers ,System Analyst
1.4.6	System Development Completion	0 days	Mon 5/4/20	Mon 5/4/20	1.4.5	
1.5	**System Testing**	**2 days**	**Mon 5/4/20**	**Tue 5/5/20**		
1.5.1	Integration Testing	1 day	Mon 5/4/20	Mon 5/4/20	1.4.6[FS-1 day]	Testers
1.5.2	Regression Testing	1 day	Mon 5/4/20	Mon 5/4/20	1.5.1[FS-1 day]	Testers
1.5.3	System	1 day	Tue	Tue	1.5.2[FS-1	Testers

	testing		5/5/20	5/5/20	day]	
1.5.4	Testing Completion	0 days	Tue 5/5/20	Tue 5/5/20	1.5.3	
1.6	**Project Closure**	**2 days**	**Wed 5/6/20**	**Thu 5/7/20**		
1.6.1	Verification and Validation	1 day	Wed 5/6/20	Wed 5/6/20	1.5.3	Developers ,Testers
1.6.2	Submission of project report	1 day	Wed 5/6/20	Wed 5/6/20	1.6.1[FS-1 day]	Project Manager
1.6.3	Release of Resources	1 day	Thu 5/7/20	Thu 5/7/20	1.6.2	Project Manager
1.7	Project Completion	0 days	Thu 5/7/20	Thu 5/7/20	1.6.3	



WBS	Task Name	Duration	Start	Finish	WBS Predecessors	Resource Names
1	**Project on Classroom Scheduling Application Developm**	**92 days**	**Wed 1/1/20**	**Thu 5/7/20**		
1.1	**Project Initiation**	**7 days**	**Wed 1/1/20**	**Thu 1/9/20**		
1.1.1	Initial Feasibility Analysis	5 days	Wed 1/1/20	Tue 1/7/20		Project Manager
1.1.2	Preparation of Initiation Document	2 days	Wed 1/8/20	Thu 1/9/20	1.1.1	Project Manager
1.1.3	Singing PID	0 days	Thu 1/9/20	Thu 1/9/20	1.1.2	
1.2	**Requirements Gathering**	**14 days**	**Fri 1/10/20**	**Wed 1/29/20**		
1.2.1	Collecting Data	7 days	Fri 1/10/20	Mon 1/20/20	1.1.3	System Analyst
1.2.2	Preparing SRS	2 days	Tue 1/21/20	Wed 1/22/20	1.2.1	System Analyst
1.2.3	Getting Approval of SRS	2 days	Thu 1/23/20	Fri 1/24/20	1.2.2	Project Manager
1.2.4	Setting up the Development Environment	3 days	Mon 1/27/20	Wed 1/29/20	1.2.3	IT Staffs
1.3	**System Design**	**12 days**	**Thu 1/30/20**	**Fri 2/14/20**		
1.3.1	Interface Designs	7 days	Thu 1/30/20	Fri 2/7/20	1.2.3,1.2.4	Designer
1.3.2	system Designs	5 days	Thu 1/30/20	Wed 2/5/20	1.2.3,1.2.4	System Analyst
1.3.3	Preparing System Design Document	5 days	Mon 2/10/20	Fri 2/14/20	1.3.1,1.3.2	System Analyst
1.3.4	Getting approval of the System Design Document	0 days	Fri 2/14/20	Fri 2/14/20	1.3.3	
1.4	**System Development**	**56 days**	**Mon 2/17/20**	**Mon 5/4/20**		
1.4.1	**First Iteration**	**11 days**	**Mon 2/17/20**	**Mon 3/2/20**		
1.4.1.1	Coding	7 days	Mon 2/17/20	Tue 2/25/20	1.3.4	Developers
1.4.1.2	Unit Testing	2 days	Wed 2/26/20	Thu 2/27/20	1.4.1.1	Developers
1.4.1.3	Review	2 days	Fri 2/28/20	Mon 3/2/20	1.4.1.2	Developers ,System
1.4.2	**Second Iteration**	**11 days**	**Mon 3/2/20**	**Mon 3/16/20**		
1.4.2.1	Updating Requirements speficications	1 day	Mon 3/2/20	Mon 3/2/20	1.4.1.3[FS-1 day]	System Analyst
1.4.2.2	Coding	6 days	Tue 3/3/20	Tue 3/10/20	1.4.2.1	Developers
1.4.2.3	Unit Testing	2 days	Wed 3/11/20	Thu 3/12/20	1.4.2.2	Developers
1.4.2.4	Review	2 days	Fri 3/13/20	Mon 3/16/20	1.4.2.3	Developers ,System
1.4.3	**Third Iteration**	**16 days**	**Mon 3/16/20**	**Mon 4/6/20**		
1.4.3.1	Updating Requirements speficications	1 day	Mon 3/16/20	Mon 3/16/20	1.4.2.4[FS-1 day]	System Analyst
1.4.3.2	Coding	10 days	Tue 3/17/20	Mon 3/30/20	1.4.3.1	Developers
1.4.3.3	Unit Testing	3 days	Tue 3/31/20	Thu 4/2/20	1.4.3.2	Developers
1.4.3.4	Review	2 days	Fri 4/3/20	Mon 4/6/20	1.4.3.3	Developers ,System
1.4.4	**Fourth Iteration**	**11 days**	**Mon 4/6/20**	**Mon 4/20/20**		
1.4.4.1	Updating Requirements speficications	1 day	Mon 4/6/20	Mon 4/6/20	1.4.3.4[FS-1 day]	System Analyst
1.4.4.2	Coding	6 days	Tue 4/7/20	Tue 4/14/20	1.4.4.1	Developers
1.4.4.3	Unit Testing	2 days	Wed 4/15/20	Thu 4/16/20	1.4.4.2	Developers
1.4.4.4	Review	2 days	Fri 4/17/20	Mon 4/20/20	1.4.4.3	Developers ,System
1.4.5	**Fifth Iteration**	**11 days**	**Mon 4/20/20**	**Mon 5/4/20**		
1.4.5.1	Updating Requirements speficications	1 day	Mon 4/20/20	Mon 4/20/20	1.4.4[FS-1 day]	System Analyst
1.4.5.2	Coding	6 days	Tue 4/21/20	Tue 4/28/20	1.4.5.1	Developers
1.4.5.3	Unit Testing	2 days	Wed 4/29/20	Thu 4/30/20	1.4.5.2	Developers
1.4.5.4	Final Review	2 days	Fri 5/1/20	Mon 5/4/20	1.4.5.3	Developers ,System
1.4.6	System Development Completion	0 days	Mon 5/4/20	Mon 5/4/20	1.4.5	
1.5	**System Testing**	**2 days**	**Mon 5/4/20**	**Tue 5/5/20**		



WBS	Task Name	Duration	Start	Finish	WBS Predecessors	Resource Names
1.4.4.3	Unit Testing	2 days	Wed 4/15/20	Thu 4/16/20	1.4.4.2	Developers
1.4.4.4	Review	2 days	Fri 4/17/20	Mon 4/20/20	1.4.4.3	Developers ,System
1.4.5	**Fifth Iteration**	**11 days**	**Mon 4/20/20**	**Mon 5/4/20**		
1.4.5.1	Updating Requirements speficications	1 day	Mon 4/20/20	Mon 4/20/20	1.4.4[FS-1 day]	System Analyst
1.4.5.2	Coding	6 days	Tue 4/21/20	Tue 4/28/20	1.4.5.1	Developers
1.4.5.3	Unit Testing	2 days	Wed 4/29/20	Thu 4/30/20	1.4.5.2	Developers
1.4.5.4	Final Review	2 days	Fri 5/1/20	Mon 5/4/20	1.4.5.3	Developers ,System
1.4.6	System Development Completion	0 days	Mon 5/4/20	Mon 5/4/20	1.4.5	
1.5	**System Testing**	**2 days**	**Mon 5/4/20**	**Tue 5/5/20**		
1.5.1	Integration Testing	1 day	Mon 5/4/20	Mon 5/4/20	1.4.6[FS-1 day]	Testers
1.5.2	Regression Testing	1 day	Mon 5/4/20	Mon 5/4/20	1.5.1[FS-1 day]	Testers
1.5.3	System testing	1 day	Tue 5/5/20	Tue 5/5/20	1.5.2[FS-1 day]	Testers
1.5.4	Testing Completion	0 days	Tue 5/5/20	Tue 5/5/20	1.5.3	
1.6	**Project Closure**	**2 days**	**Wed 5/6/20**	**Thu 5/7/20**		
1.6.1	Verification and Validation	1 day	Wed 5/6/20	Wed 5/6/20	1.5.3	Developers ,Testers
1.6.2	Submission of project report	1 day	Wed 5/6/20	Wed 5/6/20	1.6.1[FS-1 day]	Project Manager
1.6.3	Release of Resources	1 day	Thu 5/7/20	Thu 5/7/20	1.6.2	Project Manager
1.7	Project Completion	0 days	Thu 5/7/20	Thu 5/7/20	1.6.3	

3.2 *Release Plan*

The release plan for the project is given below.

Milestone	Estimated Delivery Date	Key Deliverable	Actual Delivery Date
Project Charter	Thu 1/9/20	PID	2020.02.23
Requirements baseline	Sun 3/1/20	Initial Requirements documentation	2020.02.29
Project Prototype	Mon 3/16/20	First implementation and MVP of product	-
Getting approval of the System Design Document	Fri 04/03/20	SRS, and System design document	-
User and System Documentation	Sun 04/27/20	Documentation for end-users and system maintenance	-
System Development Completion	Mon 5/4/20	Source code, unit test results, test plans	-
Testing Completion	Tue 5/5/20	Detailed test plans and results	-
Project Completion	Thu 5/7/20	-Fully developed Classroom scheduling application	

3.3 Iteration Plans

The iteration plans are given in the following table.

WBS	Task Name	Duratio n	Start	Finish
1	**First Iteration**	**11 days**	**Mon 2/17/20**	**Mon 3/2/20**
1.1	Coding	11 days	Mon 2/17/20	Tue 2/25/20
1.1.1	System setup	1 day	-	-
1.1.2	Project skeleton	1 day	-	-
1.1.3	Basic UI and templating	1 day	-	-
1.1.4	Basic REST endpoints	3 days	-	-

1.2	Unit Testing	2 days	Wed 2/26/20	Thu 2/27/20
1.3	Review	2 days	Fri 2/28/20	Mon 3/2/20
2	**Second Iteration**	**11 days**	**Mon 3/2/20**	**Mon 3/16/20**
2.1	Updating Requirements speficications	1 day	Mon 3/2/20	Mon 3/2/20
2.2	Coding	11 days	Tue 3/3/20	Tue 3/10/20
2.2.1	Research scheduling logic	3 days	-	-
2.2.2	Implement scheduling logic MVP	3 days	-	-
2.2.3	UI refinements and templates	1 day	-	-
2.2.4	User login/logout logic	2 days	-	-
2.2.5	Data base models	2 days	-	-
2.3	Unit Testing	2 days	Wed 3/11/20	Thu 3/12/20
2.4	Review	2 days	Fri 3/13/20	Mon 3/16/20
3	**Third Iteration**	**16 days**	**Mon 3/16/20**	**Mon 4/6/20**
3.1	Updating Requirements speficications	1 day	Mon 3/16/20	Mon 3/16/20
3.2	Coding	16 days	Tue 3/17/20	Mon 3/30/20
3.2.1	Refine scheduling MVP	4 days	-	-
3.2.2	Integrate front-end and back-end	3 days	-	-
3.2.3	Basic dynamic content on UI	3 days	-	-
3.2.4	Refine database models	3 days	-	-
3.3	Unit Testing	3 days	Tue 3/31/20	Thu 4/2/20
3.4	Review	2 days	Fri 4/3/20	Mon 4/6/20
4	**Fourth Iteration**	**11 days**	**Mon 4/6/20**	**Mon 4/20/20**
4.1	Updating Requirements speficications	1 day	Mon 4/6/20	Mon 4/6/20
4.2	Coding	11 days	Tue 4/7/20	Tue 4/14/20
4.2.1	Further refinements against scheduling logic	3 days	-	-
4.2.2	Implement admin and staff	2 days	-	-

	account logic			
4.2.3	Implement admin and staff UI views	2 days	-	-
4.2.4	Implement canned DB transactions	3 days	-	-
4.3	Unit Testing	2 days	Wed 4/15/20	Thu 4/16/20
4.4	Review	2 days	Fri 4/17/20	Mon 4/20/20
5	**Fifth Iteration**	**11 days**	**Mon 4/20/20**	**Mon 5/4/20**
5.1	Updating Requirements speficications	1 day	Mon 4/20/20	Mon 4/20/20
5.2	Coding	11 days	Tue 4/21/20	Tue 4/28/20
5.2.1	Finalize scheduling models and logic resolver	3 days	-	-
5.2.2	Setup project hosting and configuration	4 days	-	-
5.2.3	Finalize UI views and templating	3 days	-	-
5.2.4	Finalize database models and transactions	2 days	-	-
5.3	Unit Testing	2 days	Wed 4/29/20	Thu 4/30/20
5.4	Final Review	2 days	Fri 5/1/20	Mon 5/4/20

3.4 Budget

The project budget details are given below.

WBS	Task Name	Duration	Resource Names	Cost
1	**Project on Classroom Scheduling Application Development**	**92 days**		**$78,280.00**
1.1	**Project Initiation**	**7 days**		**$5,600.00**
1.1.1	Initial Feasibility Analysis	5 days	Project Manager	$4,000.00
1.1.2	Preparation of Initiation Document	2 days	Project Manager	$1,600.00
1.1.3	Singing PID	0 days		$0.00

1.2	**Requirements Gathering**	**14 days**		**$8,800.00**
1.2.1	Collecting Data	7 days	System Analyst	$5,040.00
1.2.2	Preparing SRS	2 days	System Analyst	$1,440.00
1.2.3	Getting Approval of SRS	2 days	Project Manager	$1,600.00
1.2.4	Setting up the Development Environment	3 days	IT Staffs	$720.00
1.3	**System Design**	**12 days**		**$10,560.00**
1.3.1	Interface Designs	7 days	Designer	$3,360.00
1.3.2	system Designs	5 days	System Analyst	$3,600.00
1.3.3	Preparing System Design Document	5 days	System Analyst	$3,600.00
1.3.4	Getting approval of the System Design Document	0 days		$0.00
1.4	**System Development**	**56 days**		**$48,160.00**
1.4.1	**First Iteration**	**11 days**		**$8,920.00**
1.4.1.1	Coding	7 days	Developers	$4,760.00
1.4.1.2	Unit Testing	2 days	Developers	$1,360.00
1.4.1.3	Review	2 days	Developers ,System Analyst	$2,800.00
1.4.2	**Second Iteration**	**11 days**		**$8,960.00**
1.4.2.1	Updating Requirements speficications	1 day	System Analyst	$720.00
1.4.2.2	Coding	6 days	Developers	$4,080.00
1.4.2.3	Unit Testing	2 days	Developers	$1,360.00
1.4.2.4	Review	2 days	Developers ,System Analyst	$2,800.00
1.4.3	**Third Iteration**	**16 days**		**$12,360.00**
1.4.3.1	Updating Requirements speficications	1 day	System Analyst	$720.00

1.4.3.2	Coding	10 days	Developers	$6,800.00
1.4.3.3	Unit Testing	3 days	Developers	$2,040.00
1.4.3.4	Review	2 days	Developers ,System Analyst	$2,800.00
1.4.4	**Fourth Iteration**	**11 days**		**$8,960.00**
1.4.4.1	Updating Requirements speficications	1 day	System Analyst	$720.00
1.4.4.2	Coding	6 days	Developers	$4,080.00
1.4.4.3	Unit Testing	2 days	Developers	$1,360.00
1.4.4.4	Review	2 days	Developers ,System Analyst	$2,800.00
1.4.5	**Fifth Iteration**	**11 days**		**$8,960.00**
1.4.5.1	Updating Requirements speficications	1 day	System Analyst	$720.00
1.4.5.2	Coding	6 days	Developers	$4,080.00
1.4.5.3	Unit Testing	2 days	Developers	$1,360.00
1.4.5.4	Final Review	2 days	Developers ,System Analyst	$2,800.00
1.4.6	System Development Completion	0 days		$0.00
1.5	**System Testing**	**2 days**		**$2,160.00**
1.5.1	Integration Testing	1 day	Testers	$720.00
1.5.2	Regression Testing	1 day	Testers	$720.00
1.5.3	System testing	1 day	Testers	$720.00
1.5.4	Testing Completion	0 days		$0.00
1.6	**Project Closure**	**2 days**		**$3,000.00**
1.6.1	Verification and Validation	1 day	Developers ,Testers	$1,400.00
1.6.2	Submission of project report	1 day	Project Manager	$800.00

1.6.3	Release of Resources	1 day	Project Manager	$800.00
1.7	Project Completion	0 days		$0.00

4 Control Plan

4.1 Monitoring and Control

The milestones will be used to track and monitoring the project progress. In addition to the milestones several meetings will be there. The details have been given below.

Milestone	**Estimated Delivery Date**
Singing PID	Thu 1/9/20
Getting approval of the System Design Document	Fri 2/14/20
System Development Completion	Mon 5/4/20
Testing Completion	Tue 5/5/20
Project Completion	Thu 5/7/20

4.2 Project Measurements

Phase	**Measurement**	**Source**
Project Initiation	Effort, time, and cost consumed by the phase	Project manager
Requirements Gathering	Effort, time, and cost consumed by the phase	Project manager
System Design	Effort, time, and cost consumed by the phase Update actual dates of tasks, milestones etc.	Project manager
System Development	Effort, time, and cost consumed by the phase	Project manager, developer
Testing	Update actual dates of tasks, milestones etc.	Project manager, tester

Project Closure	Evaluate the total cost, time, and effort for the project, Check the verification and validation Check the performance of the newly developed system	Project sponsor, project manager

5 Supporting Process Plans

5.1 Risk Management Plan

The details of the risk management plan have been given in the following tables.

Risk ID	Risk Type	Risk Description	Likelihood of Occurrence	Possible Impact
1	Managerial	The project may miss deadlines and there may be schedule slippage	High	High
2	Managerial	There may be budget overrun	High	High
3	Managerial	The resources may not available during the project	Medium	High
4	Managerial	The skills of the project team members may not be sufficient for the project	Low	High
5	Technical	The technologies used in the project may become outdated soon after the completion of the project or during the project.	Low	Medium

The risk mitigation and contingency strategies are given in the following table.

Risk ID	Risk Mitigation/ Contingency Strategy
1	• The project schedule must be flexible enough during the planning

	• There should be monitoring and controlling all the time so any issue can be addressed fast to avoid such risks
2	• There should be budgeting consider all costs related to the project • There must be contingency amount of 20% of the total budget of the project • All external costs must be considered during the project.
3	• There should be suitable policies and contracts to make sure all resources are available to the project based on those contracts.
4	• Team members should have cross-functional skills so a single member can take different roles.
5	• New technologies should be used in the project where adequate support is available • If it requires to embrace new technologies, then proper resources must be hired to use the technology in the project. Current industry standards must be followed.

5.2 Configuration Management Plan

The configuration management plan has the following activities,

- Cloud computing platform will be used to develop the system. It will be a hybrid cloud that will have the development environment set-up locally to keep all data and the application. The webserver and proxy server will be online on a cloud platform to handle load based on demands.
- Document naming convention will be universally followed in the project
- All documents, code, test results, test data, code dumps would be available in a central repository available on the local private cloud platform.
- There will be an integrated version management system
- Coding must be done based on the previously decided coding standard agreed by the project team.
- All review details would be documented
- The change control process has two parts. For a minor change the project manager can take a decision to go for the change or not. For a major change, a project manager must send the request to the project sponsor for a decision. Anyone who wants a change may submit a duly filled change request form to the project manager.

5.3 Verification and Validation Plan

All verification and validation details would be documented on a separate document called verification_validation_details.pdf.

5.4 Product Acceptance Plan

If the product or the newly developed Class Scheduling application meets 90% of the criteria in the verification and validation process, then the product will be accepted by the client.