SOFTWARE PROJECT MANAGEMENT PLAN (SPMP)

ANTHROCLOUD

Software Modernization for Cloud-based Pediatric Anthropometry

December 4, 2019

Version 2

COT 6931 Dusty Wright

Change History

Version	Date	Author	Changes
0.1	July 7, 2019	Dusty Wright	Initial Document
1.0	July 31, 2019	Dusty Wright	Baseline
1.1	October 20, 2019	Dusty Wright	Comma added. Space removed. Abbreviate WBS. Sentence spacing. Minor corrections. Minimum Project Document added to deliverables. Risk Management Table 3 Risk Management updated.
2.0	December 4, 2019	Dusty Wright	Budget, schedule, & timeline updates.

TABLE OF CONTENTS

1 Introduction 3
1.1 Project Overview 3
1.2 Project Deliverables 3
1.3 Evolution of This Document 4
1.4 References 4
1.5 Definitions, Acronyms, and Abbreviations 4
1.6 Overview 4
2 Project Organization 5
2.1 Process Model 5
2.2 Organizational Structure 5
2.3 Organizational Boundaries and Interfaces 5
2.4 Project Responsibilities 5
3 Managerial Process 6
3.1 Management Objectives and Priorities 6
3.2 Assumptions, Dependencies, and Constraints 7
3.2.1 Assumptions 7
3.2.2 Dependencies 7
3.2.3 Constraints 7
3.3 Risk Management 7
3.4 Monitoring and Controlling Mechanisms 8
4 Technical Process 9
4.1 Methods, Tools, and Techniques 9
4.1.1 Methods 9
4.1.2 Tools 10
4.1.3 Techniques 10
4.2 Software Documentation 11
4.3 Project Support Functions 11
5 Work Elements, Schedule, and Budget 12
5.1 Work Elements 12
5.2 Schedule & Timeline 13
5.3 Budget 15
5.4 Semester I & II 16

1 INTRODUCTION

This is a Software Project Management Plan (SPMP) document. An overview of this document can be found in the subsections below. This document lays out the plan for building the AnthroCloud software solution.

1.1 PROJECT OVERVIEW

An anthropometric calculator performs standard data analysis from human measurements to produce anthropometric indices. There are many WIC MIS systems in use today that use these calculators. Consequently, many anthropometric calculators exist in production environments. Despite electronic government initiatives to make available and replicate systems, common functionality is duplicated. This project offers a single anthropometric calculator solution to serve multiple funded projects.

The purpose of this project is to create a modern cloud-based WHO Child Growth Standards compliant anthropometric calculator designed to provide a more flexible, maintainable, and portable solution to meet the changing needs of users.

The scope of this project is limited to WHO Child Growth Standards, z-score calculations, percentile calculations, growth curves, and plotted scores. The intended audience for this document is both stakeholders and developers. The intended use is as cloud-based HTTP service.

The objectives of the project are as follows:

1. Choose an application platform to improve the ability of the calculator to change to new specifications or operating environments.
2. Design software that decouples the application into components to reduce the effort necessary to change the calculator.
3. Provide a common interface to better exchange data between calculator components.
4. Build a test suite to reduce the effort necessary to verify calculator changes.

1.2 PROJECT DELIVERABLES

The following deliverables will be produced in accordance with project objectives:

1. Source Code
   1.1. IDE Solution
   1.2. Database Script
   1.3. Web Service Project
   1.4. Web Application Project
   1.5. Test Project
2. Documentation
   2.1. Data Model
   2.2. Architecture Model
   2.3. User Interface Mockup
   2.4. Developer Manual
   2.5. Minimum Project Documents

1.3 Evolution of This Document

This is a living document and will be updated throughout the project. Feedback, review, and refinement of the management plan will be updated. Change history will record team member, revision number, and change comment.

The document version will be displayed on the cover page and footer by the major version number, a decimal separator, and a minor version number. The major version represents submitted work. The minor version shall record edits during final review prior to submission.

1.4 References

Azure for Students-Azure FAQ | Microsoft Azure. (2019). Retrieved June 12, 2019, from https://azure.microsoft.com/en-us/free/free-account-students-faq/

Baley, K., & Belcham, D. (2010). *Brownfield application development in .NET.* Greenwich, CT: Manning.

Dice. How Much Are Your Skills Worth? (n.d.). Retrieved July 7, 2019, from https://www.dice.com/salary-calculator

Hopkins, R., & Jenkins, K. (2008). *Eating the IT Elephant.* Upper Saddle River, NJ: IBM Press.

ISO/IEC-25010, (2011). Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.

ISO 25000 Portal. (n.d.). Retrieved July 03, 2019, from https://iso25000.com/index.php/en/

Pricing Calculator | Microsoft Azure. (2019). Retrieved June 12, 2019, from https://azure.microsoft.com/en-us/pricing/calculator/

Software Project Management Plan Template. (n.d.). Retrieved July 25, 2019, from https://ase.in.tum.de/lehrstuhl_1/component/content/article/43-books/240-oose-template-projectmanagementplan

1.5 Definitions, Acronyms, and Abbreviations

MIS	Management Information System	A computerized information system used for decision-making in the WIC program.
HTTP	Hypertext Transfer Protocol	HTTP is a protocol for transferring data over the Web.
WHO	World Health Organization	A United Nations agency concerned with international public health.
WIC	Women, Infants, & Children	A federally funded nutrition program for women, infants, and children.
WBS	Work Breakdown Structure	A project management document that organizes work into deliverables.

1.6 Overview

The following Software Project Management Plan (SPMP) describes the plan undertaken to produce the AnthroCloud software solution. The Software Modernization for Cloud-based Pediatric Anthropometry project delivers the AnthroCloud software. The SPMP document summarizes the project management

plan including project organization, managerial process, technical process, and work elements for delivery of the work product.

2 PROJECT ORGANIZATION

The project process, organization, and responsibilities are defined below.

2.1 PROCESS MODEL

An iterative and incremental model that combined a repeated cycle with work units allowing the software to be built and delivered in components and layers towards a completed project was desired. An agile process model will be used to plan the work and improve with subsequent iterations. The weekly schedule, software design, software components, benefit the time-boxed approach towards creating working usable software.

2.2 ORGANIZATIONAL STRUCTURE

The organizational structure is comprised of one team and one member.

2.3 ORGANIZATIONAL BOUNDARIES AND INTERFACES

This project will use course communication as the interface to the project client. An elaborate team structure or communication strategy will not be necessary for this project given team size. Work units are organized sequentially. Each deliverable must be completed before moving onto the next one.

2.4 PROJECT RESPONSIBILITIES

This schedule could support running tasks currently for the following roles: Project Manager, Software Architect, Business Analyst, Development Operations (DevOps) Developer, Database Administrator, Database Developer, User Interface (UX) Developer, Front-End Developer, Back-End Developer, Quality Assurance (QA) Developer, and Technical Writer. The longest running deliverable duration is 20 days. The roles and responsibilities are assigned to the author.

3 MANAGERIAL PROCESS

This section provides a context for issues impacting the project status and how to address those issues.

3.1 MANAGEMENT OBJECTIVES AND PRIORITIES

The project solution features are prioritized to exhibit the following desired characteristics: flexible, maintainable, and portable. The management objectives derive from ISO 9126 standard successor ISO/IEC 25010 product quality characteristics and sub-characteristics, specifically maintainability and portability.

A flexible solution can change, move, and adapt outside of current requirements. A maintainable solution is not tightly coupled. The components can be changed and used again. The components can be tested and measured. A portable solution can change environments or even be replaced.

Table 1. ISO/IEC 25010 quality attributes for this software.

Characteristics	Sub-characteristics	Definitions
Maintainability	Modularity	Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.
	Reusability	Degree to which an asset can be used in more than one system, or in building other assets.
	Analyzability	Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.
	Modifiability	Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.
	Testability	Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.
Portability	Adaptability	Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.
	Installability	Degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.
	Replaceability	Degree to which a product can replace another specified software product for the same purpose in the same environment.

Note: Table 2 compiles data from ISO/IEC 25010 (2011) and ISO 25000 Portal. (n.d.) for product quality characteristics and sub-characteristics.

3.2 Assumptions, Dependencies, and Constraints

The assumptions, dependencies, and constraints impacting the AnthroCloud solution are detailed.

3.2.1 Assumptions

1. A single resource is available from 8/26 to 12/13.
2. The technology stack will not impact timeline.
3. An accurate anthropometric calculator can be created.

3.2.2 Dependencies

1. The technology stack is available, installed, stable, and integrated.
2. Cloud resources can be created and persist.
3. Third-party software draws charts and calculates scores.

3.2.3 Constraints

1. A cloud-based platform must be used.
2. WHO growth standard is used.
3. The calculator draws and calculates z-scores and percentiles.

3.3 Risk Management

A risk management plan exists for the AnthroCloud Software Solution. See Software Risk Management Plan. The plan details a strategy for identifying, analyzing, and mitigating risks. The following risk mitigation strategies have been recommended, assessed, and prioritized for this project in Table 3:

Table 3 Risk Management

Risk	Severity	Probability	Mitigation
1. Discrepancies with WHO Anthro	Medium. This impacts verification of calculations.	Unlikely. The behavior can be replicated. The risk is reduced due to better understanding of calculation & user interface configuration.	AnthroCloud assumes clinician inputs are compliant with standard measurement procedure (recumbent and standing), whereas WHO Anthro adjusts inputs. Two-years and under shall be measured recumbently using Length-for-Age; otherwise, Height-for-Age and standing will be used. Unadjusted inputs are a more accurate. Measurement user interface element shall be removed or read-only.
2. Few Anthropometric Calculator Libraries	Medium. Without an accurate calculator, development time increases for coding, research, verification, and validation. Also,	Unlikely. Reduced after testing accuracy.	Erik Knudsen's anthstat-statistics (GitHub) computes WHO Child Growth Standards compliant z-scores and percentiles for children and adolescents. The library was tested and is accurate for the measurement indicators reducing likelihood of risk.

	development time decreases for other tasks.		
3. Limited Free Line Plot Controls	Medium. This impacts the display of child growth charts.	Highly Unlikely. Library found.	Use a user interface control library with a line plot chart control like Google Charts.
4. Limited Sample Data	Minor. This impacts verification testing.	Unlikely.	WHO Documentation contains some test data. Anthstat-statistics solution contains automated tests. WHO Training Course contains test data. CDC Growth Chart Training contains test data. Some State agencies provide online training.
5. Azure for Students Service limitations	Minor. This could impact architecture.	Highly Unlikely. Remaining balance is positive.	Review Azure for Students FAQ product restrictions (“Azure for Students”, 2019). Review pricing calculator for subscription prices (“Pricing Calculator”, 2019).

3.4 Monitoring and Controlling Mechanisms

Unplanned activities resulting from impediments and issues shall follow the Risk Management Plan. The impediment work type may be used to track risks when encountering such activities. The issue or impediment arising from risk may be documented outside the risk register to record blocked tasks. The following tools assist monitor and control:

Excel is used to maintain the Work Breakdown Structure and Budget.

Microsoft Project is used to maintain the Project Plan and Gantt Chart.

GitHub is the host for development version control.

Azure Boards is used to manage backlog, manage sprint, schedule tasks, create work items, integrate repository, manage pipeline, store test plans and project artifacts.

4 TECHNICAL PROCESS

The technical process outline below will be followed.

4.1 METHODS, TOOLS, AND TECHNIQUES

A planning tool will be utilized to run tasks in Sprints. Tools are necessary to execute the project plan are outlined below. The engineering principles and approach is documented.

4.1.1 Methods

This project will use an Agile Process Template available in Azure Boards. The work breakdown structure (WBS) created for this project represents a project plan used to create a backlog of user stories. The backlog represents the work to develop and ship. Work item tracking will be used. Agile work item states shall reflect New, Active, Resolved, Closed, or Removed. Five sprint iterations produce five epics.

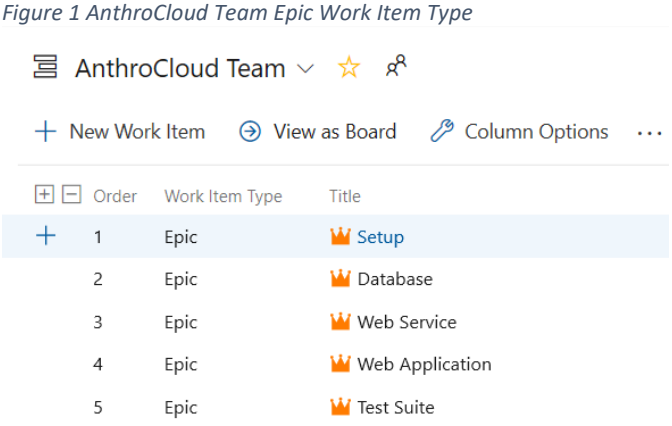
Work item type mapping for WBS tasks is as follows. The agile theme creates a cloud-based anthropometric calculator. The agile epic prioritizes setup, database, web service, web application, and test suite deliverables. The agile features represent collections of related stories that, for example, create data access components, business components, user interface components and unit tests. The agile user story level includes the creation of items such as a calculator sub-component. An agile task example includes adding the Visual Studio solution to source control. Table 4 displays mapped concepts between Agile work items and WBS tasks.

Table 4. Agile WBS Work Item Type Map

Agile Work Type Hierarchy	**Work Breakdown Structure**
Theme	1. AnthroCloud
Epic	1.1 Setup, 1.2 Database, 1.3 Web Service, 1.4 Web App 1.5 Test Suite
Feature	1.1.1 IDE Solution, 1.1.2 Cloud Resources
User Story	1.1.1.1 Source Control, 1.1.1.2 Container Image
Tasks	1.1.1.1.1 Create GitHub Account 1.1.1.1.2 Integrate with Visual Studio

The WBS and Gantt chart for this project was created in Microsoft Project and published to an Azure Board as a backlog. The individual task row in Microsoft Project has a column for work item task type. This drop-down automatically populates with options from the project's selected process template when integrated with an Azure Board. For example, tasks marked Epic in Microsoft Project are subsequently published to an Azure Board appearing in AnthroCloud Team backlog in Figure 1.

Figure 1 AnthroCloud Team Epic Work Item Type



4.1.2 Tools

The following tools will be used in this project:

Azure Boards is used to manage backlog, manage sprint, schedule tasks, create work items, integrate repository, manage pipeline, store test plans and project artifacts.

Azure for Students allows subscription access to Azure services at no cost, commitment, or time limit.

dotPeek is a free .NET decompiler and assembly browser.

JustDecompile is a free .NET decompiler and assembly browser.

Excel is used to maintain the WBS and Budget.

GitHub is the host for development version control.

Pencil is an open-source GUI prototyping tool used to create low-fidelity user interface mockup.

Postman tests the web service by constructing complex REST API quickly.

Project is used to maintain the Project Plan and Gantt Chart.

UMLet is an open-source UML tool for structural and behavioral diagrams.

Visual Studio 2019 is a full-feature integrated development environment.

[WHO] Anthro software assesses child growth using standards.

4.1.3 Techniques

This project is a forward engineering effort building software from high-level view of existing software. AnthroCloud is not just a modern equivalent of an older program, but a new program that extends beyond the original program's capabilities. Reverse engineering tools are used for interface discovery. The purpose of using these tools is to understand the design not reverse engineer it. Interfaces are created to separate layers one at a time. A bottom-up horizontal strategy is used create a newly designed, coded, and tested program. Figure 2 below illustrates product assembly from larger to smaller composition.

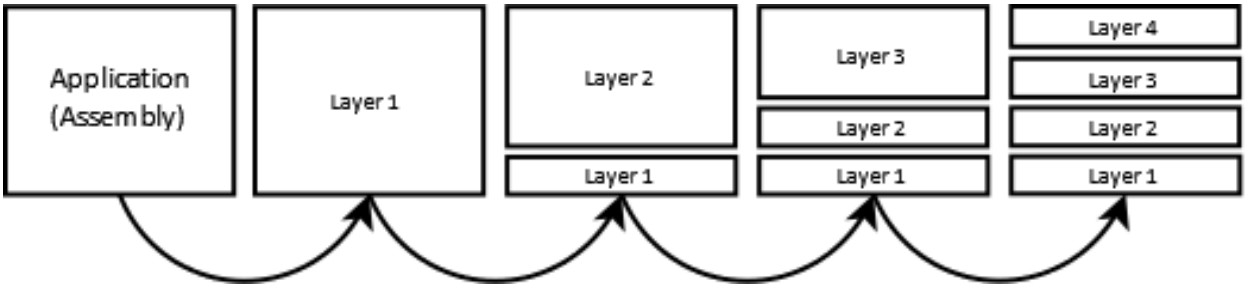


Figure 2 Layer Assembly

4.2 Software Documentation

Software documentation from section two deliverables is as follows:

2. Documentation
   2.1. Data Model
   2.2. Architecture Model
   2.3. User Interface Mockup
   2.4. Developer Manual
   2.5. Minimum Project Documents

A data model will detail various anthropometric indices and parameters needed to generate percentiles and scores by gender. Architectural layer diagram will be created. A user interface mockup of the screen will be created. A developer manual will detail how to consume the web service.

4.3 Project Support Functions

The following support functions, in Table 5, create value. While no formal structure or organizational unit supports specific functional service areas for this project, the following support areas keep the progress focused on project deliverables organizing and categorizing administrative support functions.

Table 5 Support Functions

Support Functions	
Type	**Function**
Project	Schedule updating and reporting
Project	Time sheet recording and maintenance
Project	Report progress and status
Project	Manage product backlog, user stories, & tracking
Technical	Create Source Control Repository
Technical	Integrate IDE and Source Control Repository
Technical	Commit and Version working code
Technical	Document Test plan
Technical	Create and maintain developer Manual
Technical	Create and maintain models and mockups

5 WORK ELEMENTS, SCHEDULE, AND BUDGET

This section details how the work is to be carried out. All work is carried out by the author.

5.1 WORK ELEMENTS

The following deliverables are scheduled:

1. Source Code
   1.1. IDE Solution
   1.2. Database Script
   1.3. Web Service Project
   1.4. Web Application Project
   1.5. Test Project
2. Documentation
   2.1. Data Model
   2.2. Architecture Model
   2.3. User Interface Mockup
   2.4. Developer Manual
   2.5. Minimum Project Documents

The tasked deliverables in Table 6 appear as scheduled work elements below:

Table 6 WBS Deliverable Map

WBS Code	Task	Duration	Start	Finish
1	Source Code/ IDE Solution	80 days	Mon 8/26/19	Fri 12/13/19
1.2.2	Database Script	40 hrs	Mon 9/16/19	Fri 9/20/19
1.3	Web Service	20 days	Mon 9/30/19	Fri 10/25/19
1.4	Web Application	15 days	Mon 10/28/19	Fri 11/15/19
1.5	Test Suite	20 days	Mon 11/18/19	Fri 12/13/19
1.2.1	Schema (Data Model)	40 hrs	Mon 9/9/19	Fri 9/13/19
1.2.1	Schema (Architecture Model)	40 hrs	Mon 9/9/19	Fri 9/13/19
1.2.1	Schema (User Interface Mockup)	40 hrs	Mon 9/9/19	Fri 9/13/19
1.4.1.4	Developer Manual	20 hrs	Wed 11/13/19	Fri 11/15/19

The AnthroCloud software solution is subdivided tasks into Setup, Database, Web Service, Web Application, and Test Suite. All Tasks will fall under these level 2 subdivided components. These elements can be seen below in Figure 3 with WBS Codes 1.1 through 1.5.

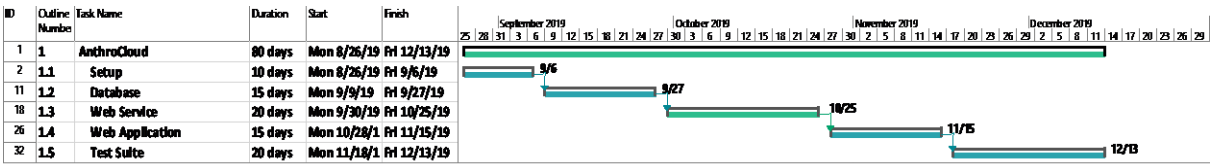


Figure 3 WBS Deliverables

5.2 SCHEDULE & TIMELINE

The work is scheduled to begin Monday, August 26, 2019, through Friday, December 13, 2019. The image below is a Microsoft Project plan including WBS outline and Gantt chart to visualize tasks. See Figure 4. Sections of work contain tasks and sub-tasks necessary to complete deliverables for this project. Completion end dates of specific milestones are highlighted in bold on the Gantt chart to show when significant work is concluded. The final week is open.

ID	Outline Number	Task Name	Duration	Start	Finish
1	**1**	**AnthroCloud**	**80 days**	**Mon 8/26/19**	**Fri 12/13/19**
2	**1.1**	**Setup**	**10 days**	**Mon 8/26/19**	**Fri 9/6/19**
3	**1.1.1**	**IDE Solution**	**40 hrs**	**Mon 8/26/19**	**Fri 8/30/19**
4	1.1.1.1	Source Control	20 hrs	Mon 8/26/19	Wed 8/28/19
5	1.1.1.2	Container Image	20 hrs	Wed 8/28/19	Fri 8/30/19
6	**1.1.2**	**Cloud Resources**	**40 hrs**	**Mon 9/2/19**	**Fri 9/6/19**
7	1.1.2.1	Database Resource	10 hrs	Mon 9/2/19	Tue 9/3/19
8	1.1.2.2	Service Resource	10 hrs	Tue 9/3/19	Wed 9/4/19
9	1.1.2.3	Application Resource	10 hrs	Wed 9/4/19	Thu 9/5/19
10	1.1.2.4	Container Resource	10 hrs	Thu 9/5/19	Fri 9/6/19
11	**1.2**	**Database**	**15 days**	**Mon 9/9/19**	**Fri 9/27/19**
12	1.2.1	Schema (Models & Mockup)	40 hrs	Mon 9/9/19	Fri 9/13/19
13	1.2.2	Seed Data & Script	40 hrs	Mon 9/16/19	Fri 9/20/19
14	**1.2.3**	**Data Access Component**	**40 hrs**	**Mon 9/23/19**	**Fri 9/27/19**
15	1.2.3.1	Data Access Interface	10 hrs	Mon 9/23/19	Tue 9/24/19
16	1.2.3.2	Data Access Layer	15 hrs	Tue 9/24/19	Thu 9/26/19
17	1.2.3.3	Data Access Logic	15 hrs	Thu 9/26/19	Fri 9/27/19
18	**1.3**	**Web Service**	**20 days**	**Mon 9/30/19**	**Fri 10/25/19**
19	**1.3.1**	**Business Component**	**120 hrs**	**Mon 9/30/19**	**Fri 10/18/19**
20	1.3.1.1	Business Interface	10 hrs	Mon 9/30/19	Tue 10/1/19
21	1.3.1.2	Business Logic Layer	30 hrs	Tue 10/1/19	Fri 10/4/19
22	1.3.1.3	Calculator	80 hrs	Mon 10/7/19	Fri 10/18/19
23	**1.3.2**	**Service Component**	**5 days**	**Mon 10/21/19**	**Fri 10/25/19**
24	1.3.2.1	Service Interface	10 hrs	Mon 10/21/19	Tue 10/22/19
25	1.3.2.2	Service Layer	30 hrs	Tue 10/22/19	Fri 10/25/19
26	**1.4**	**Web Application**	**15 days**	**Mon 10/28/19**	**Fri 11/15/19**
27	**1.4.1**	**User Interface Component**	**15 days**	**Mon 10/28/19**	**Fri 11/15/19**
28	1.4.1.1	Data Entry Form	40 hrs	Mon 10/28/19	Fri 11/1/19
29	1.4.1.2	Chart Control	40 hrs	Mon 11/4/19	Fri 11/8/19
30	1.4.1.3	Template	20 hrs	Mon 11/11/19	Wed 11/13/19
31	1.4.1.4	Developer Manual	20 hrs	Wed 11/13/19	Fri 11/15/19
32	**1.5**	**Test Suite**	**20 days**	**Mon 11/18/19**	**Fri 12/13/19**
33	1.5.1	Unit Tests	40 hrs	Mon 11/18/19	Fri 11/22/19
34	1.5.2	Integration Tests	40 hrs	Mon 11/25/19	Fri 11/29/19
35	1.5.3	End-to-End Test	40 hrs	Mon 12/2/19	Fri 12/6/19
36	1.5.4	FINAL GRADING	40 hrs	Mon 12/9/19	Fri 12/13/19

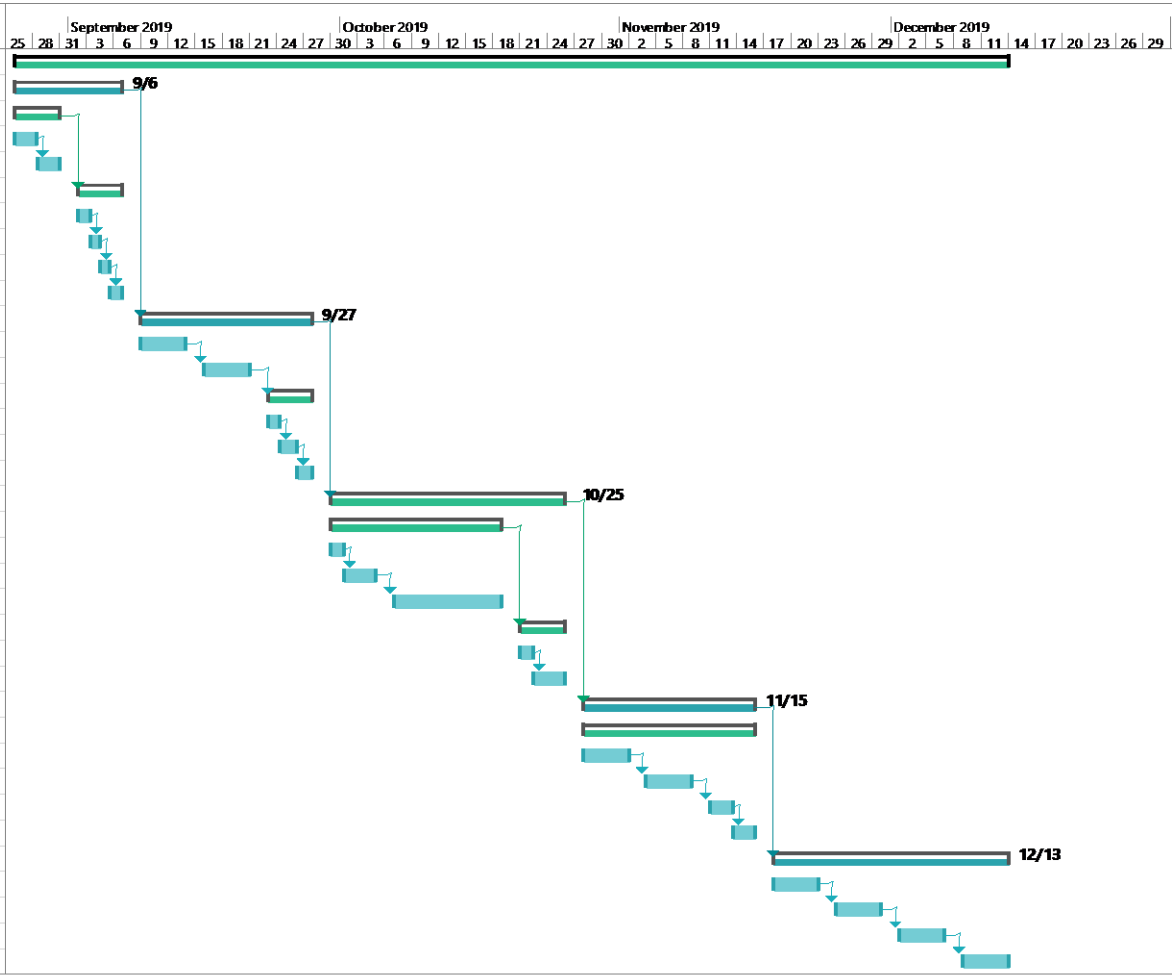


Figure 4 WBS

The timeline depicts a high-level view of project deliverables. This view summarizes the effort spent on each deliverable in the project.

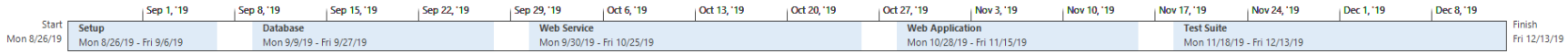


Figure 5 Timeline

5.3 BUDGET

The budget for this project, Figure 6, is calculated using a flat rate of $47 dollars. This number is derived from DICE Salary Calculator which bases salary prediction on market value, location, skill, experience, and other factors using numerous data points and professional profiles (DICE, 2019).

Task Name	Estimated	Actual	Rate	Budget	
Setup	80	60	$47	$3,760	$940
Database/Script	80	90	$47	$3,760	-$470
Data Model	13.33333333	13.33333333	$47	$627	$0
Architecture Model	13.33333333	13.33333333	$47	$627	$0
User Interface Mockup	13.33333333	13.33333333	$47	$627	$0
Web Service Project	160	160	$47	$7,520	$0
Web Application Project	100	110	$47	$4,700	-$470
Developer manual	20	20	$47	$940	$0
Test Project	120	120	$47	$5,640	$0
Project Definition	224	224	$47	$10,528	$0
Progress Report	40	40	$47	$1,880	$0
Software Project Management Plan (SPMP)	40	40	$47	$1,880	$0
Software Requirements Specifications (SRS)	40	40	$47	$1,880	$0
Software Design Document (SDD)	40	40	$47	$1,880	$0
Risk Management Plan	40	40	$47	$1,880	$0
PowerPoint Presentation	10	10	$47	$470	$0
YouTube Presentation	10	10	$47	$470	$0
Final Report	10	10	$47	$470	$0
Acceptance Letter	10	10	$47	$470	$0
Total	**1064**	**1064**		**$50,008**	$50,008

Figure 6 Budget

5.4 SEMESTER I & II

Week	WBS Code	Task Name	Estimated	Actual	Variance
5/13		Module 1: Course Introduction	24	24	0
5/20		Module 2 - Project Definition Section 1	40	32	8
5/27		Module 3 - Project Definition Section 2	40	44	-4
6/3		Module 4 - Project Definition Section 3, Project Status 1	40	48	-8
6/10		Module 5 - Project Definition Section 4, Project Status 2	40	36	4
6/17		Project Definition Section 5 (WBS), Project Status 3	40	38	2
6/24		Progress Report 1 due in its dropbox.	40	37	3
7/1		Module 8 - Project Status Update 4 due in its dropbox.	40	40	0
7/8		Module 9 - Project Status Update 5 due in its dropbox.	40	40	0
7/15		Module 10 - Progress Report 2 due in its dropbox.	40	40	0
7/22		Module 11 - Project Status Update 6 due in its dropbox.	40	40	0
7/29		Module 12 - Final Report, YouTube, PowerPoint presentation, Acceptanc	40	40	0
8/5		FINAL GRADING	0		0
8/26	**1**	**Setup**	**80**	**60**	20
8/26	1.1	IDE Solution	40	32	8
8/26	1.1.1	Source Control	20	12	8
8/26	1.1.2	Container Image	20	20	0
9/2	1.2	Cloud Resources	40	28	12
9/2	1.2.1	Database Resource	10	6	4
9/2	1.2.2	Service Resource	10	8	2
9/2	1.2.3	Application Resource	10	4	6
9/2	1.2.4	Container Resource	10	10	0
9/9	**2**	**Database**	**120**	**130**	-10
9/9	2.1	Schema (Models & Mockup)	40	45	-5
9/16	2.2	Seed Data (Script)	40	45	-5
9/23	2.3	Data Access Component	40	40	0
9/23	2.3.1	Data Access Interface	13 1/3	13 1/3	0
9/23	2.3.2	Data Access Layer	13 1/3	13 1/3	0
9/23	2.3.3	Data Access Logic	13 1/3	13 1/3	0
9/30	**3**	**Web Service**	**160**	**160**	0
9/30	3.1	Business Component	120	120	0
9/30	3.1.1	Business Interface	10	15	-5
9/30	3.1.2	Business Logic Layer	30	25	5
10/7	3.1.3	Calculator	80	80	0
10/14	3.2	Service Component	40	40	0
10/14	3.2.1	Service Interface	10	10	0
10/14	3.2.2	Service Layer	30	30	0
10/28	**4**	**Web Application**	**120**	**130**	-10
10/28	4.1	User Interface Component	120	130	-10
10/28	4.1.1	Data Entry Form	40	40	0
11/4	4.1.2	Chart Control	40	50	-10
11/11	4.1.3	Template	20	20	0
11/11	4.1.4	Developer Manual	20	20	0
11/18	**5**	**Test Suite**	**120**	**120**	**0**
11/18	5.2	Unit Tests	40	40	**0**
11/25	5.3	Integration Tests	40	40	**0**
12/2	5.4	End-to-End Test	40	40	**0**
12/9		FINAL GRADING	0		