2016

Amy Knowles

Steve Kundert

Yashwanth Reddy Muddireddy

Vaisali Namburi

2016 February 19

v. 1.2



Project Plan

Table of Contents

[1 Introduction 3](#_Toc444727474)

[1.1 Purpose of Plan 3](#_Toc444727475)

[1.2 Project Scope and Objectives 3](#_Toc444727476)

[1.2.1 Statement of Scope 3](#_Toc444727477)

[1.2.2 Major Functions 3](#_Toc444727478)

[1.3 Overview of Document 3](#_Toc444727479)

[2 Project Organization 4](#_Toc444727480)

[2.1 Process Model 4](#_Toc444727481)

[2.2 Team Structure 4](#_Toc444727482)

[2.3 Communication 5](#_Toc444727483)

[2.4 Reporting 5](#_Toc444727484)

[3 Project Resources 6](#_Toc444727485)

[3.1 Hardware and Software 6](#_Toc444727486)

[4 Risk Management Plan 7](#_Toc444727487)

[4.1 Risks 7](#_Toc444727488)

[5 Testing 8](#_Toc444727489)

[6 Schedule 8](#_Toc444727490)

[6.1 Project Breakdown 8](#_Toc444727491)

[6.2 Time Line 8](#_Toc444727492)

[6.3 Task List 9](#_Toc444727493)

[7 Tracking and Control Mechanics 10](#_Toc444727494)

[8 Consultants 10](#_Toc444727495)

[9 References 11](#_Toc444727496)

[10 Glossary 11](#_Toc444727497)

[11 Appendix 12](#_Toc444727498)

Table of Figures

Figure 1. Agile Scrum process [1] 4

Table 1. Potential Risks 7

Figure 2. Project Timeline 9

Table 2. Task List 10

Figure 3. Task Network Diagram 12

Figure 4. Project Gantt Chart 13

Figure 5. Objective Grading Sheet 14

# Introduction

## Purpose of Plan

The purpose of this plan is to evaluate the available resources, which are needed for completing the project successfully, and plan accordingly. This document elaborately discusses the important tasks, duties to perform and their management. Additionally, foreseen risk analysis and methods for risk mitigation are reviewed. Moreover, the timeline, milestones, responsibilities and roles of the team members as well as methods of quality assurance, reporting and backup plans are also specified.

## Project Scope and Objectives

### Statement of Scope

The main objective of this project is to produce a prototype that the customer can use for advising during the fall of 2016. The prototype will provide a user-friendly, graphical user interface that works on top of a mini-database generated by reading in a text file that was converted from an available PDF of the course catalog and should demonstrate enough features that the customer can successfully utilize the GUI to access open core and elective courses during the advisement period. The development phase will take place over a twelve-week period during the spring 2016 semester.

### Major Functions

The functionalities of the software are described as follows:

* Open classes search: option to return ONLY open classes or all classes, independent of availability, based on type of course selected.
* Fundamental Core Classes search: option to return classes under the fundamental core umbrella.
* Fundamental Elective Classes search: option to return classes under the fundamental electives umbrella.
* Major Core Classes search: option to return classes required by the computer science department for all undergraduate computer science students.
* Major Elective Classes search: option to return computer science courses that can be selected on a per student basis that would count toward the computer science undergraduate degree requirement of 18+ hours in advanced topics related to the computer science field.

## Overview of Document

This document is arranged as follows, section one describes the purpose and the objective of the plan document. Section two specifies the process model and team structure. Hardware and software requirements, and risk management are included in the third and fourth sections respectively of the document. The schedule of the project, tracking and control mechanics, as well as the glossary are discussed in the final sections.

# Project Organization

## Process Model

The process model that will be used in this project is the Agile Scrum model, which is a practice-based methodology for effective modeling and documentation of software products as shown in Figure 1. In this process model, the tasks are divided into small time frames to deliver specific features for a release. As the project involves designing the system that opens up the courses that are available for a particular semester based on the search specifications by the customer, the requirements of the project may change over time. For example, this project is only for B.A and B.S degree plans but new degree plans could be added at a later date in order to aid faculty and staff advising undergraduates in any major. The Agile process model may be used for large, complex projects, which is well suited for the Open Classes Project.

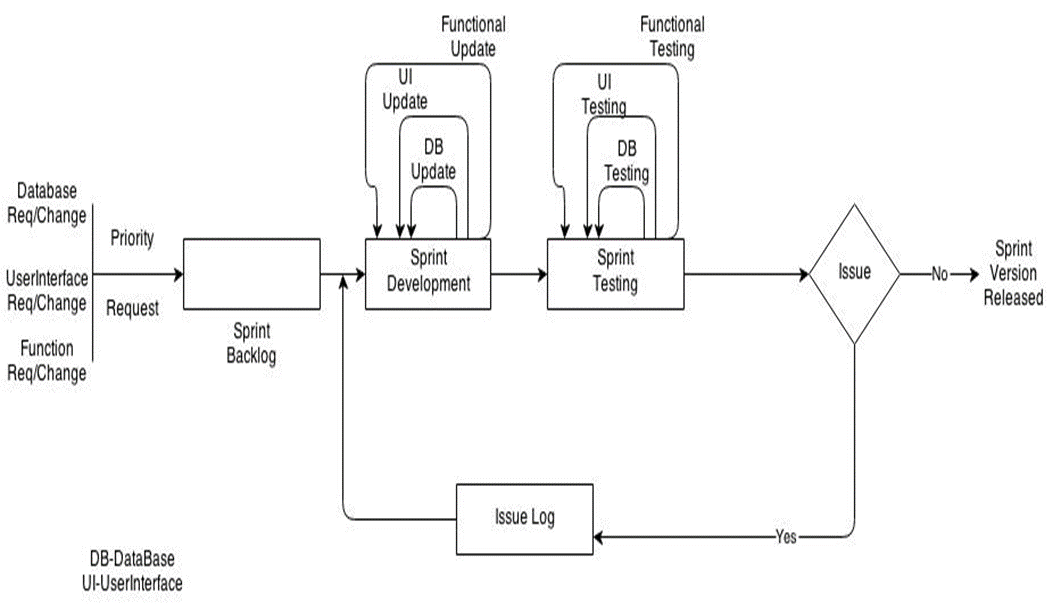


Figure 1. Agile Scrum process [1]

## Team Structure

Team members of our project are listed below.

* Amy Knowles
* Steve Kundert
* Yashwanth Reddy Muddireddy
* Vaisali Namburi

According to the skills and expertise required, the role played by each member of the team may change during the process of development. Dr. Catherine Stringfellow will fulfill the role of the client and will be monitoring the overall success of the project. Roles played by the team members are listed below.

* Team Leader (Amy Knowles)
* Documents Reviewer (Everyone)
* DB Developer (Vaisali Namburi)
* DB Peer Reviewer (Everyone)
* DB Tester (Everyone)
* UI Developer (Yashwanth Reddy Muddireddy)
* UI Peer Reviewer (Steven Kundert)
* UI Tester (Everyone)
* Core Developer (Everyone)
* Core peer Reviewer (Everyone)
* Core Tester (Everyone)

In this project we are following a controlled decentralized structure where the team leader will make final decisions based on the input of team members. The roles fulfilled by the team members will be decided by the team leader based on the team member’s skills and expertise and will set the expectations of that role for that situation and will estimate the outcome for that role.

## Communication

The communication between the team members will be through e-mails and a social network application called WhatsApp as well as through regular meetings. During team meetings, the status of the project will be discussed, individual goals will be set and analysis will be completed on previously set goals. The problems and their possible solutions will be explored. If the situation demands, micro teams will be formed within the team. Checkpoints such as milestones, timeline of the project etc., will be checked against the project schedule. Individual and team log tables will be maintained and will be available to all team members. The source code and other documents such as the requirements and specifications document and the project plan will be shared via GitHub.

## Reporting

The team leader is responsible for keeping the team on schedule and in her absence a supporting leader, Yashwanth, will take charge as the team leader. She will arbitrate any conflicts between team members. Issues will be logged and discussed during an emergency team meeting, should the need arise. If team members are unable to resolve an issue during the emergency meeting, the course instructor, Dr. Stringfellow, will have the final word in resolving the issue.

# Project Resources

## Hardware and Software

The project development environment will be as follows:

**Hardware** will include a personal (desktop or notebook) computer with the following specifications:

* an Intel i5 microprocessor running at 2.0 GHz or higher
* a minimum of 128 megabytes of system RAM
* a CD-RW drive
* a v.92-compliant 56K modem
* a hard drive with at least 2 gigabytes of free space
* an XGA screen display 14” or larger
* a graphics card with 16 megabytes or more of RAM

**Software** will include:

* Windows 7/ 8 / 10 or OS X Yosemite 10.10.5
* Eclipse IDE
* PDF to TXT converter
* Microsoft Project
* MySQL Workbench

The prototype developed will utilize the following:

**Hardware** – a personal (desktop or notebook) computer with the following specifications:

* an Intel i5 microprocessor running at 2.0 GHz or higher
* a minimum of 128 megabytes of system RAM
* a CD-RW drive
* a v.92 compliant 56K modem
* a hard drive with at least 2 gigabytes of free space
* an XGA screen display 14” or larger
* a graphics card with 16 megabytes or more of RAM

**Software:**

* Windows 7 / 8 / 10
* OS X Yosemite 10.10.5
* Executable file developed the project team
* PDF to TXT software
* MySQL Workbench

# Risk Management Plan

## Risks

Table 1 lists potential risks, the likelihood of that risk occurring, the impact the occurrence would have on the team, and a strategy for overcoming the challenge proposed by the risk.

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Chance | Impact | Strategy |
| Illness of Team Members | Low | Tolerable | Plan project timeline to include a buffer. Redistribute work if necessary. |
| Procrastination | Low | Serious | Have frequent meetings to monitor progress. |
| Changes in Requirements | High | Serious | Additional effort of team members will be required for solving this problem. |
| Compatibility Constraints | Low | Serious | Additional research. |
| Learning Curve for Use of Applications | High | Serious | Find and use as much reference material as possible. |

Table 1. Potential Risks

Risk management is important for any project as poor planning in regards to project risks leads to many serious issues. Therefore, every member of the team should recognize and acknowledge possible risks. Because this team is being introduced to the Java programming language, there is a risk that team members will be unprepared to develop software in this language. However, all team members have knowledge of Object-Oriented programming techniques and class-based coding languages. In order to detect and resolve bugs as well as to add new features, extra effort should be taken by the team in order to overcome the possibility of a slow learning curve. The risks involved in the development of this system also include a lack of communication. Students in this development team are involved in other research groups and regular meeting times could create scheduling conflicts. An effort by team members to keep other lines of communication open, such as utilizing phone applications such as WhatsApp and e-mail, will reduce these risks.

# Testing

Every version release of the project will be complete with code review, technical review, function review, unit testing and integration testing. This will help the team release a bug free and risk free environment.

Test cases for each sprint will be created keeping in mind all of the roles involved in the usage of the software. How users and administrators will login and perform the task in each scrum will be unit tested and integration tested. It will be finalized only after the completion of risk analysis. All the details of the test plan will be included in the test plan document.

# Schedule

The project timeline, and task list can be found in the subsections to follow. The objective grading sheet and the task network diagram can be found in the appendix. Tasks appearing in bold font on the objective grading sheet have been completed.

## Project Breakdown

This Schedule will give the outline for the tasks to be performed in the project.

The project goes as follows:

1. Gathering requirements from the client.
2. Clarifying requirements and specifications with the client.
3. Creating the sprint outline
4. Building a Sprint
5. Designing sprint
6. Developing sprint
7. Execution and Testing of sprint.
8. Updating the changes in the sprint
9. Submitting the project to the client

Steps 3 through 8 will be repeated as necessary during the project until completion of the software.

## Time Line

The timeline divides the project into different tasks and gives a detailed description of the project schedule by allocating specific time slots to each task as shown in Figure 2. Tasks will be scheduled using Microsoft Project.



Figure 2. Project Timeline

## Task List

Table 2 shows the project tasks and their dependencies.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Task Name | Duration (days) | Predecessor ID |
| 0 | **Software Development** | **68** |  |
| 1 | **Requirements** | **17** |  |
| 2 | Use Case | 5 |  |
| 3 | HW & SW Requirements | 6 |  |
| 4 | Draft Documents | 6 |  |
| 5 | Other Diagrams | 10 |  |
| 6 | Final Req. Document | 5 | 4 |
| 7 | Reviews | 5 | 4 |
| 8 | Presentation | 1 | 7 |
| 9 | Requirements Complete | 0 |  |
| 10 | **Plan** | **29** |  |
| 11 | Plan Resources | 12 |  |
| 12 | Schedule 1 (MS Project) | 2 |  |
| 13 | Task Network Diagram | 2 |  |
| 14 | Risks | 4 |  |
| 15 | Schedule 2 (MS Project) | 9 |  |
| 16 | Reviews | 14 | 12 |
| 17 | Plan Complete | 0 | 15 |
| 18 | **Prototype / Design Imp** | **34** |  |
| 19 | UID Draft | 12 | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Task Name | Duration (days) | Predecessor ID |
| 20 | First Prototype / HLD | 5 | 9 |
| 21 | Reviews | 4 |  |
| 22 | Final UID | 4 |  |
| 23 | Final Code | 20 | 21 |
| 24 | Prototype Complete | 0 | 23 |
| 25 | **Testing** | **42** |  |
| 26 | Schedule | 2 | 12 |
| 27 | Test Cases / Criteria | 10 |  |
| 28 | Reviews | 8 |  |
| 29 | Test Plan | 5 |  |
| 30 | Testing | 10 | 24 |
| 31 | Testing Complete | 0 | 30 |
| 32 | **Manual** | **36** |  |
| 33 | Outline | 5 |  |
| 34 | Screen Shots | 3 | 22 |
| 35 | Preliminary User Manual | 8 |  |
| 36 | Reviews | 8 | 35 |
| 37 | Manual Complete | 0 |  |
| 38 | **Presentation** | 62 |  |
| 39 | Deliverables | 58 |  |
| 40 | Presentation | 14 |  |

Table 2. Task List

# Tracking and Control Mechanics

The project will be monitored during the development process to catch any errors that may arise. The team will develop a plan to test the software at various stages of the development. The team will also review the requirements and specifications plan to ensure there have been no oversights.

The testing documentation will outline the process that the testers will follow to catch any possible error in the system. Testing will be documented and will state what the errors are and what was done to fix them. Such a plan, carefully executed, would ensure the system is deployed with optimal accuracy.

# Consultants

1. Stringfellow, Dr. Catherine, Customer, Open Classes Project, MSU Wichita Falls, Fall 2015.
2. Kundert, Carley, Possible Future Customer, Open Classes Project, MSU Wichita Falls, Fall 2015.

# References

1. Sommerville I., Software Engineering 9th edition, Delhi, Pearson Education, Ltd, 2011, 84-88.
2. Uyttewaal E., “Microsoft Project: Plan Better with Microsoft Project”, New York, TechNet Magazine, February 2012.

# Glossary

**B.A. Degree Plan** – a degree plan within the computer science department that allows students to take less rigorous mathematics courses in exchange for a number of credit hours in humanities/literature courses and fourteen semester hours of a foreign language.

**B.S. Degree Plan** – a degree plan within the computer science department that requires students to take more rigorous mathematics courses.

**Course** – an available college course during a given semester; can be either traditional or on-line.

**Degree Plan** – a roadmap of the required courses for a particular major within a collegiate institution.

**Fundamental Core / Electives** – courses required of all university students independent of major.

**GUI** – the graphical user interface that allows a person to interact with the software easily by utilizing a mouse and keyboard.

**Major** – a student’s desired field of study within the university.

**Major Core / Electives** – courses required by a specific department for students graduating with a degree from that department.

**On-line Course** – college class in which students access course information through an on-line portal, such as D2L, and are not required to present to a physical location.

# Appendix

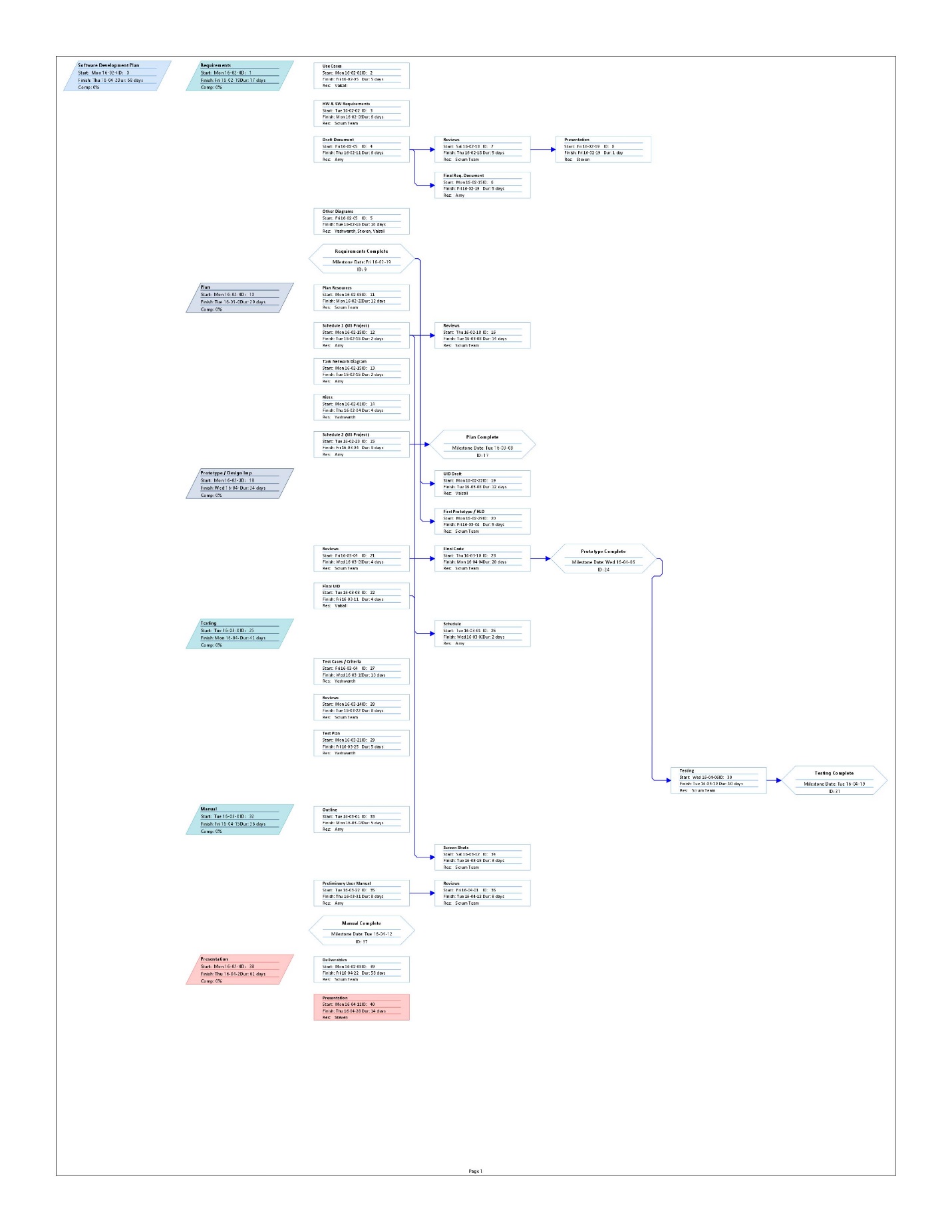


Figure 3. Task Network Diagram

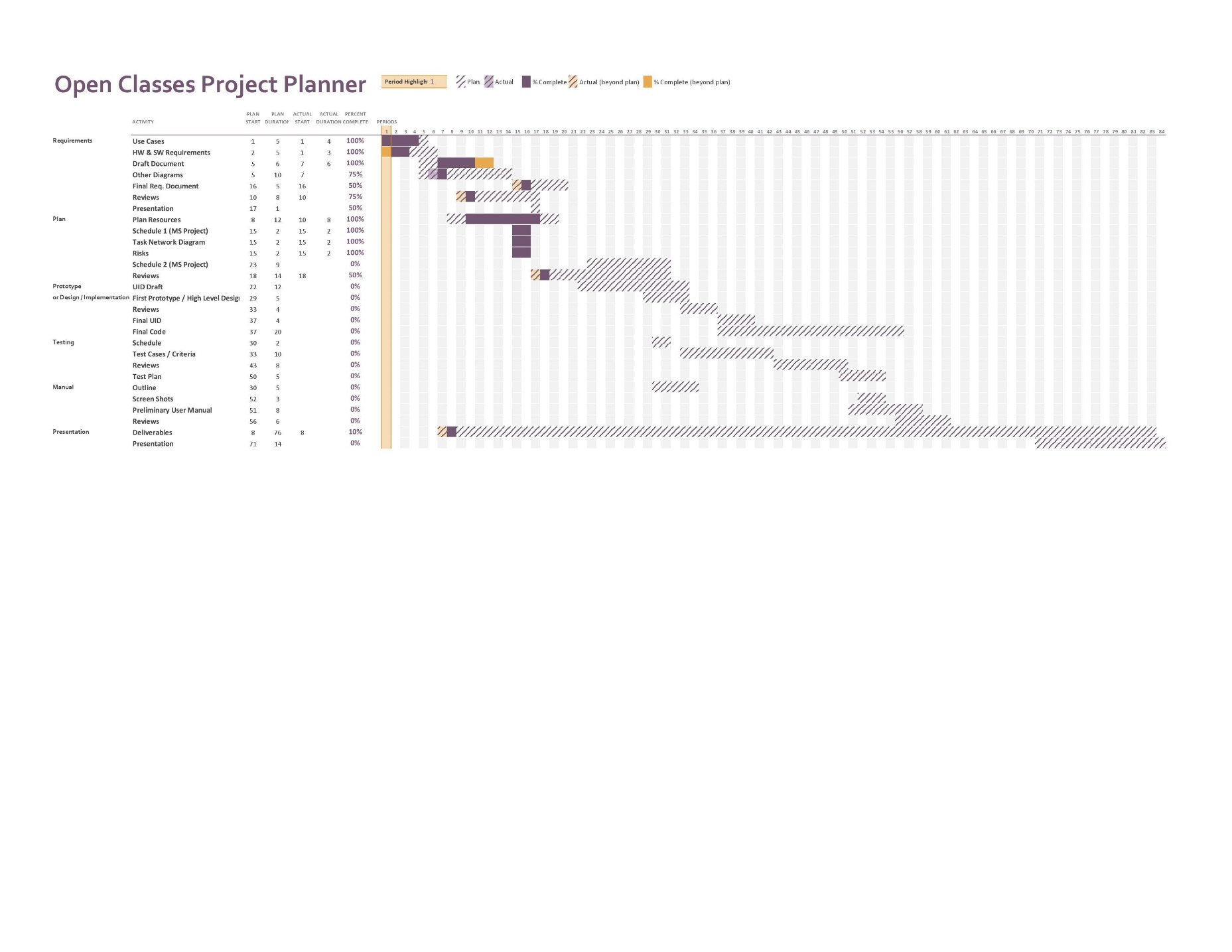


Figure 4. Project Gantt Chart

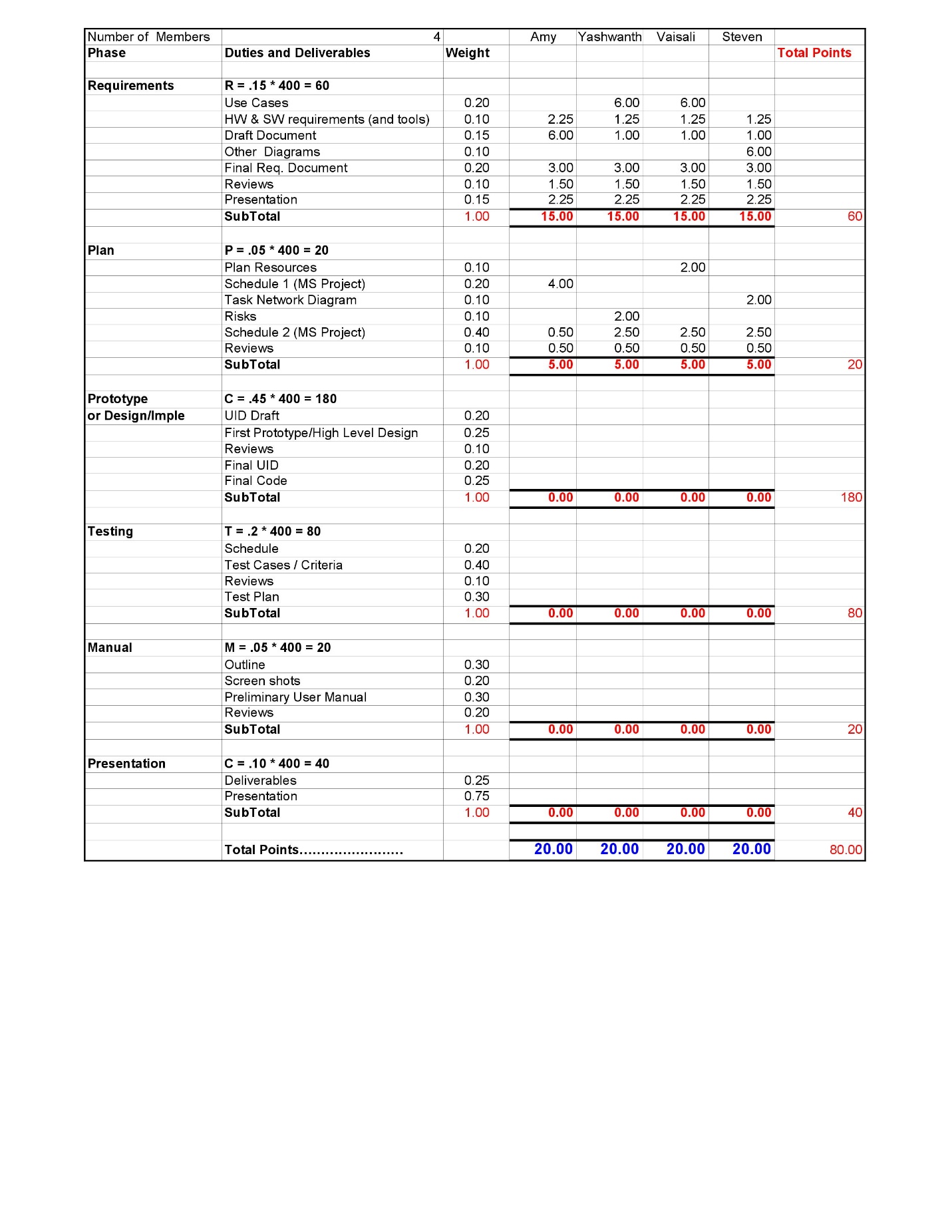


Figure 5. Objective Grading Sheet