**PROJECT MANAGEMENT PLAN**

**FOR THE**

**BookIt**

**January 24, 2019**

***Place project logo here***

Team Name: Group #2

Project Manager: Zach Brockway

Team Member 1: Vina Ta

Team Member 2: Jacob Forte

Team Member 3: Marcus Johnson

**PREFACE**

This Project Management Plan (PMP) is intended to provide guidance on the management of the BookIt.

The template conforms to the Institute of Electrical and Electronics Engineers (IEEE) Standard for Software Project Management Plans, IEEE Std 1058-1998, for format and content. The template and its standard were selected as they are flexible enough to be applied to any type of project. The management, technical, and supporting processes comply with the guidance provided by Standard for Information Technology - Software Life Cycle Processes, IEEE/Electronic Industries Association (EIA) 12207 Series; Systems Engineering – System Life Cycle Processes, International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 15288; or the Processes for Engineering a System, Electronic Industries Alliance (EIA) Standard 632.

**DOCUMENT CONVENTIONS**

The outline of this Project Management Plan (PMP) has been tailored from the Institute of Electrical and Electronics Engineers (IEEE) Standard for Software Project Management Plans, IEEE Std 1058-1998.

Standard conventions are used within this document to direct the reader to specific sections of the text. These sections provide instructions and explanations and require users to insert their own project-specific information. The conventions used in this document are shown below.

[[text]] Global changes. Items that appear in regular text and are surrounded by double brackets represent changes that can be made globally throughout the document.

*Italics* Instructions and explanations. Each section of the template has been annotated with a guidance box, derived from the IEEE 1058-1998 standard, to assist the reader in drafting the content. For example:

***IEEE Std 1058-1998 Guidance***

*The guidance box provides instructions and explanations from the IEEE 1058-1998 Standard, in italics, as required to assist the user in drafting their own information.*

Guidance boxes should be deleted from the final PMP.

Clear Sections that are applicable to our project.

Yellow Sections that might be applicable to our project. We do not have enough information or knowledge to fill out these sections.

Red Sections that are not applicable to our project.

Blue Last things to do.RECORD OF CHANGES

\*A - ADDED M - MODIFIED D – DELETED

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| VERSION NUMBER | DATE | SECTION NUMBER CHANGED | **A\* M D** | TITLE OR BRIEF DESCRIPTION | CHANGE ID |
| 1 | 1/24/2019 |  | A | Filled in basic information |  |
| 1.0.1 | 1/28/2019 | General | A | Filled in more basic information | VT |
| 1.0.2 | 1/30/2019 | 5.1.3 | A | Add initial information to 5.1.3 | Z.B. |
| 1.0.3 | 1/30/2019 | 5.1.1 | A | Added basic information | M.J.J |
| 1.0.4  1.0.5 | 1/30/2019  1/31/2019 | 1 & 4 | A | Filled in more info in section 1 and completed section 4. | J.F. |
| 1.0.6 | 1/31/2019 |  | M | Edited completed sections | Z.B. |
| 1.0.6 | 2/11/2019 | 5.2.2 & 5.3.5 | A | Filled in information and added charts for some sections | M.J.J |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**TABLE OF CONTENTS**

**Section Page**

[SECTION 1. OVERVIEW 1](#_3ep43zb)

[1.1](#_2et92p0) Project Summary 1

[1.1.1](#_tyjcwt) Purpose, Scope, and Objectives 1

[1.1.2](#_3dy6vkm) Assumptions and Constraints 1

[1.1.3](#_1t3h5sf) Project Deliverables 1

[1.1.4](#_4d34og8) Master Schedule and Budget Summary 1

[1.2](#_2s8eyo1) Evolution of the Plan 2

[1.3](#_1tuee74) Document Structure 2

[SECTION 2. REFERENCES 4](#_3rdcrjn)

[2.1](#_26in1rg) Standards and Documents 4

[SECTION 3. DEFINITIONS 5](#_lnxbz9)

[SECTION 4. PROJECT ORGANIZATION 6](#_35nkun2)

[4.1](#_1ksv4uv) External Interfaces 6

[4.2](#_2jxsxqh) Internal Structure 6

[4.2.1](#_4du1wux) The Project Manager 6

[4.3](#_3whwml4) Project Roles and Responsibilities 6

[SECTION 5. MANAGEMENT PROCESS 7](#_qsh70q)

[5.1](#_3as4poj) Start-up 7

[5.1.1](#_1pxezwc) Estimation 7

[5.1.2](#_49x2ik5) Staffing 7

[5.1.3](#_2p2csry) Resource Acquisition 7

[5.1.4](#_147n2zr) Staff Training 8

[5.2](#_3o7alnk) Work Planning 8

[5.2.1](#_ihv636) Work Activities 8

[5.2.2](#_32hioqz) Schedule Allocation 8

[5.2.3](#_1hmsyys) Resource Allocation 9

[5.2.4](#_41mghml) Budget Allocation 9

[5.3](#_2grqrue) Project Controls 9

[5.3.1](#_3fwokq0) Requirements Control 9

[5.3.2](#_1v1yuxt) Schedule Control 10

[5.3.3](#_1mrcu09) Budget Control 10

[5.3.4](#_2zbgiuw) Quality Control 11

[5.3.5](#_1egqt2p) Project Reporting and Communication 11

[5.3.6](#_2szc72q) Metrics Collection 11

[5.4](#_2r0uhxc) Risk Management 12

[5.5](#_1664s55) Project Closeout 12

[SECTION 6. TECHNICAL PROCESS 13](#_3q5sasy)

[6.1](#_25b2l0r) Process Model 13

[6.2](#_kgcv8k) Methods, Tools and Techniques 13

[6.3](#_34g0dwd) Project Infrastructure 13

[6.4](#_43ky6rz) Product Acceptance 13

[SECTION 7. SUPPORTING PROCESSES 15](#_2iq8gzs)

[7.1](#_xvir7l) Configuration Management 15

[7.2](#_3hv69ve) Independent Verification and Validation 15

[7.3](#_1x0gk37) Documentation 15

[7.4](#_2w5ecyt) Quality Assurance 16

[7.5](#_1baon6m) Reviews and Audits 16

[7.6](#_3vac5uf) Problem Resolution 16

[7.7](#_184mhaj) Contractor Management 16

[7.7.1](#_3s49zyc) Contracting Process 17

[7.7.2](#_279ka65) Contractor Performance Monitoring 17

[7.8](#_1opuj5n) Process Improvement 17

[7.8.1](#_48pi1tg) Systems/Software Process Improvement Lead 17

[7.8.2](#_2nusc19) Systems Engineering Process Group 17

[SECTION 8. ADDITIONAL PLANS 18](#_1302m92)

[APPENDIX A. BookItMASTER SCHEDULE (MICROSOFT PROJECT) 2](#_2250f4o)

[APPENDIX B. BookItFACILITIES PLAN 3](#_haapch)

[APPENDIX C. BookItPROJECT TRAINING PLAN 4](#_319y80a)

[APPENDIX D. BookIt MEASUREMENT PLAN 5](#_1gf8i83)

[APPENDIX E. BookIt PRODUCT ENGINEERING AND QUALIFICATION PROCESS 6](#_40ew0vw)

[APPENDIX F. BookItQUALITY ASSURANCE PLAN 7](#_2fk6b3p)

[APPENDIX G. BookItCONFIGURATION MANAGEMENT PLAN 8](#_upglbi)

**LIST OF FIGURES**

**Figure Page**

**LIST OF TABLES**

**Table Page**

[Table 7-1. BookIt Documentation 16](#_meukdy)

# **SECTION 1. OVERVIEW**

## **1.1 PROJECT SUMMARY**

The purpose of this project is to create a website that allows users to view a list of class availability, while providing a web user interface that allows for smooth transactions for students to sell and buy books for their courses.

### **1.1.1 Purpose, Scope, and Objectives**

Mobile website that allows users to track courses and post/purchase books  
User account that allows tracking courses, finding available books for upcoming course and posting books for sale   
User selected option for course notifications based on predefined criteria   
Course rating system. Allows user to post reviews and rate their class experience

### **1.1.2 Assumptions and Constraints**

The only constraint for this project is that it must be finished before the end of the semester. Additionally, we only have four members on our team and will have no opportunity to expand.

### **1.1.3 Project Deliverables**

The planned delivery date will be TBA at the Kent State Stark Campus in Room 306. The team will present the completed website to the potential buyer in exchange for a grade, preferably an A.

### **1.1.4 Master Schedule and Budget Summary**

This project will be worked on until the end of the Spring 2019 school semester. There will be no monetary budget for this project. Since this project will be worked on via an agile process, there are no scheduled milestones throughout our allotted time period.

## **1.2 EVOLUTION OF THE PLAN**

For now, we will be reviewing and updating this document on Tuesdays. All updates will be recorded in the changelog located underneath the document conventions section. Unscheduled updates will likely only occur when our client requests a change in our project requirements.

## **1.3 DOCUMENT STRUCTURE**

This plan is organized as follows:

1. Section 1, Project Overview. This section provides an overview of the scope and objectives of the project, the project’s assumptions and constraints, reference to the project deliverables, schedule and budget, and a description of the evolution of the plan.
2. Section 2, References. This section provides a list of all documents, policies, templates, processes, and other sources of information referenced in the plan.
3. Section 3, Definitions. This section contains the abbreviations and acronyms required to properly understand this planning document.
4. Section 4, Project Organization. This section identifies interfaces to organizational entities external to the project, the project’s internal organizational structure, and defines roles and responsibilities for the project.
5. Section 5, Management Process. This section describes the planning, measurement, tracking, reporting, risk control mechanisms needed to provide management control over the technical processes and product quality, and appropriate project initiation and closeout procedures.
6. Section 6, Technical Process. This section describes the technical solution in terms of a process model and implementation methods, tools, and techniques to be used to develop the various work products, plans for establishing and maintaining the project infrastructure, and the product acceptance.
7. Section 7, Supporting Processes. This section describes processes that are employed to facilitate and control the technical processes and the state of the product. These include, but are not limited to, configuration management, verification and validation, documentation, quality assurance, reviews and audits, problem resolution, and contractor management, and methods to ensure continuous process improvement.

# S**ECTION 2. REFERENCES**

## **2.1 STANDARDS AND DOCUMENTS**

The standards and documents listed below are referenced in this document:

Software Requirement Specification - <https://docs.google.com/document/d/1HJnQZGqYFhS92jpcrqzaSeM8DXlDUUSdGhQ4s30FC8M/edit#heading=h.3whwml4>

# S**ECTION 3. DEFINITIONS**

***IEEE Std 1058-1998 Guidance***

***(Clause 3) Definitions***

*This clause shall define, or provide references to, documents containing the definition of all terms and acronyms required to properly understand this planning document.*

# **SECTION 4. PROJECT ORGANIZATION**

## **4.**1 **INTERNAL STRUCTURE**

Our development team has four members, one of which is the manager. The manager makes all final decisions with regards to the project. All interactions with the client will mostly be handled by the manager. There will be no defined QA or verification and validation.

## **4.**2 **PROJECT ROLES AND RESPONSIBILITIES**

There is only one defined role in our project, the project manager. The project manager performs most interactions with the client. All other roles share equal responsibilities for the project.

# **SECTION 5. MANAGEMENT PROCESS**

## **5.1 START-UP**

## 5.1.1 **Estimation in Real life**

The estimated time for completing this project is 4 months with an initial cost of $25,000-$40,000. This information is based off of research conducted over the course of 2 days that looked after several aspects of estimating the cost of a project. The information that was researched was the cost of setting up and maintaining a server for a small business with averaged $500-$1,500. Then the estimated cost of employing a small team of web and database developers to build the site and database which came in at $35,000. This number was base on the average salary of a entry level programer multiplied by 4 (size of team) and then again multiplied by 4 for the length of months they are employed. Keep in mind this number can vary as the lowest percentile for entry level programmers is $25,000 annually and the number estimated for this project was slightly above that.   
   
As for tools that are recommended for the project. For scheduling a simple microsoft excel spreadsheet would suffice along with a free or open source take management tool such as trello. Microsoft Azure would provide a easy to use interface for the database however this would not be necessary. For a better understanding of how the planning and management of the project will go it is recommend that “Managing and leading Software Projects” by Richard E. (Dick) Fairley chapters 4 - 5 be read. More information will be added based on first hand experiences as the project progresses.

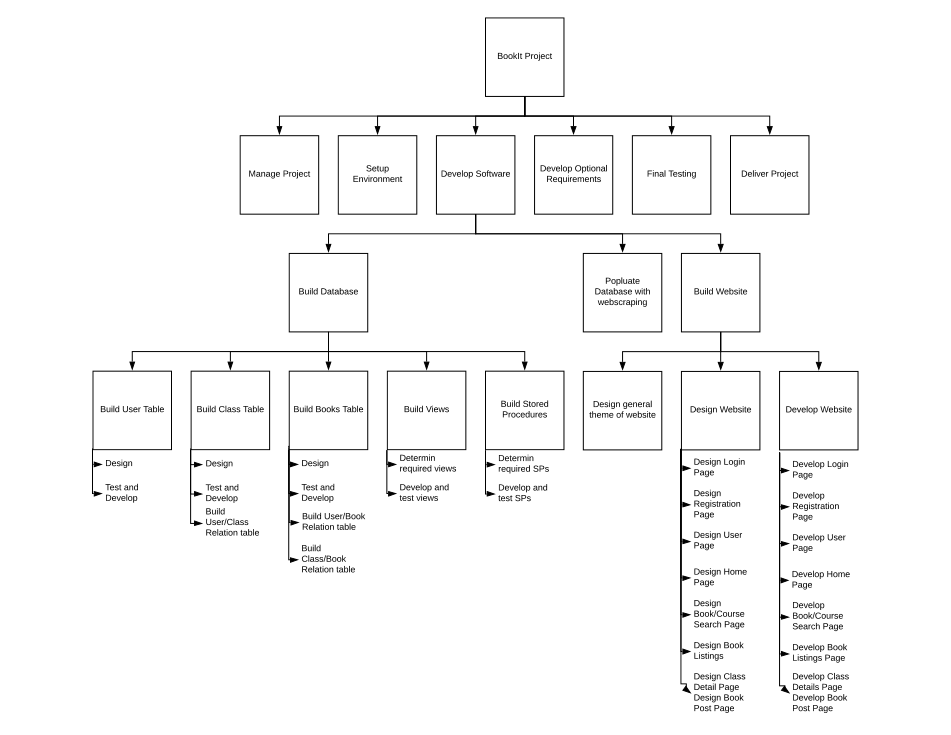
### **5.1.**2 **Resource Acquisition**

Two of the main resources needed to complete the project will include local and production environment. These environments include a variety of both software and hardware. Each team member will be responsible for setting up their own local environment (IDE, web server, PHP, and database), ideally by utilizing the web stack package XAMPP (for Windows machines). The production environment will consist of a virtual machine, rented from a cloud provider, to serve the final project through a web server. This production environment will be as close to the local environments as possible and will be the project manager’s responsibility to set up.

## **5.2 WORK PLANNING**

The following paragraphs provide a working management plan for the acquisition of the BookIt.

### **5.2.1 Work Activities**

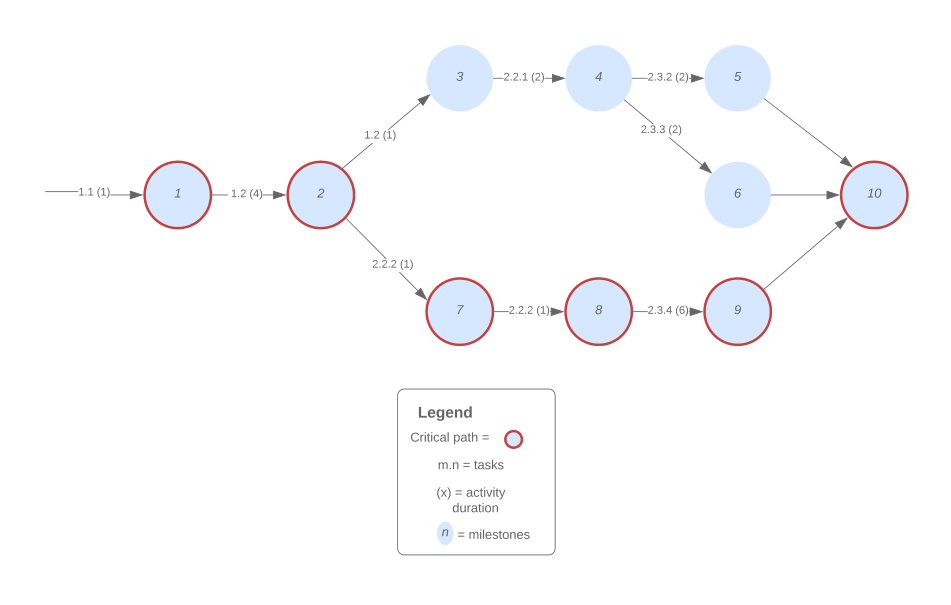


*Figure 5.2.1.1 - A work breakdown structure*

**Table 5.2.1.1 A task list**

|  |  |  |  |
| --- | --- | --- | --- |
| Activity Member | Description | Predecessors | Duration  (Weeks) |
| 1.1 | Initiate Project | --- | 1 |
| 1.2 | Analyse Requirements | 1.1 | 1 |
| 1.3 | Manage Project | 1.1 | 4 |
| 2.2 | Design |  |  |
| 2.2.1 | Design Database | 1.2 | 1 |
| 2.2.2 | Design Website | 1.2 | 2 |
| 2.3 | Implement code |  |  |
| 2.3.1 | Implement Database Tables | 2.2.1 | 2 |
| 2.3.2 | Implement Stored Procedures and Views | 2.3.1 | 2 |
| 2.3.3 | Implement Web Scraping | 2.3.1 | 2 |
| 2.3.4 | Implement Website | 2.2.2 | 6 |
| 2.4 | Implement Optional Requirements | 2.3 | 2 |

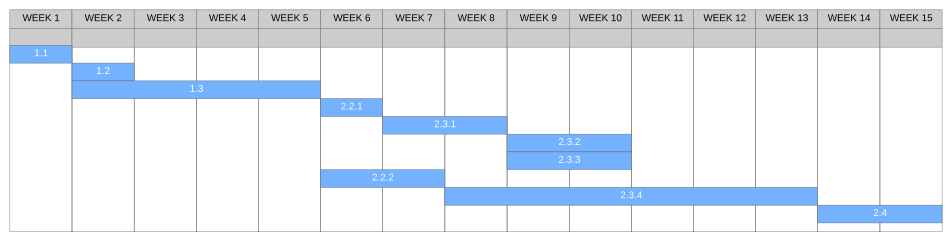
### **5.2.2 Schedule Allocation**



*Figure 5.2.2.1 - Critical-path activity network*

**Table 5.2.2.1 - Milestones for figure 5.2.2.1**

|  |  |
| --- | --- |
| Event | Description |
| 1 | Project initiation |
| 2 | Requirements analysis completed |
| 3 | Database Design Complete |
| 4 | Database Tables Complete |
| 5 | Database Views and Stored Procedures Complete |
| 6 | Web Scraping Complete |
| 7 | General Theme of Website Designed |
| 8 | Website Designed |
| 9 | Website Implemented |
| 10 | Project Delivered |



*Figure 5.2.2.2 - Gantt Chart for Figure 5.2.2.1*

### **5.2.3 Resource Allocation**

Our team really only has two resources, time, which is handled in the previous section, and people. Since the planning and development of the front end website is estimated to take far longer than the planning and development of the database, three fourths of our team will be dedicated to the frontend and one fourth will be dedicated to the database.

## **5.3 PROJECT CONTROLS**

### **5.3.1 Requirements Control**

New project requirements will be evaluated by the entire project team to determine the feasibility based on the limited time and skills of the project team. The entirety of the project must be completed in 16 weeks (a semester), any new requirements will only be accepted if they can be completed within this timeframe. To determine the amount of time needed to complete any new requirements, the skill of the team will be a major factor. Due to the team being composed of students, extra time may be required to complete advanced requirements.

### **5.3.2 Schedule Control**

The control mechanisms to be used to measure the progress of work completed at the major and minor project milestone will be an excel sheet labeled Project Tracking which will allow each of us to update how far along we are on our separate section.

### **5.3.4 Quality Control**

Quality control will not be specifically enforced due to a lack of resources. Every developer will be expected to sufficiently test their own code changes and perform regression tests before committing their changes.

### **5.3.5 Project Reporting and Communication**

In order to make reports this project will use weekly planning documents along with task management programs/site such as trello. the use of flowcharts were also used to provide a visual aid in how the flow of the project should go. for meetings the group will use agile based meetings due to the size of the team. Github with continuous deploy will be used as a release tool. Lucid charts and google drive were used to house documentation and charts that allow all members of the team to make updates to the document while saving data on a cloud service.

## **5.4 RISK MANAGEMENT**

Each group member will be given exactly a week to to complete their assigned tasks. Risks to the completion of these tasks could be due to a team member getting sick, running out of time, or not possessing the required skills to complete the task. To mitigate these risks, team members should report any difficulties related to the completion of tasks to the project manager at-least two days before the task due date. The project manager will then work to distribute the work among the remaining project members.

## **5.5 PROJECT CLOSEOUT**

The project will be concluded at the end of the semester, regardless of how complete the project is. The project will be presented to our employer and our peers at the end of the semester. After the end of this project, all members will be leaving the group and moving on to bigger and better projects with actual companies. All documents related to this project will be uploaded to the GitHub used for version control, including resources found on the server regarding how to setup the project for future use.

# **SECTION 6. TECHNICAL PROCESS**

## **6.1 PROCESS MODEL**

## 

*Figure 6.1.1: Project Model: Basic overview draft*

## **6.2 METHODS, TOOLS AND TECHNIQUES**

The product will be developed using an Agile development methodology. Reviews of previous week's work will occur during weekly meetings and will be conducted by each team member. PHP, SQL, and HTML are the primary languages used to build the product. To aid the creation of the product, industry standard templates (such as IEEE 1058 AKA Software Requirements Specifications) will be written by the entirety of the team. The product will be tracked and versioned by Git, and automatically deployed using C.I. (continuous integration).

## **6.3 PROJECT INFRASTRUCTURE**

Each team member is responsible for maintaining a functioning development environment compatible with the production environment. During the beginning of product development, the team will meet to ensure all development environments are working correctly. Development on the product will have no set location. Each team member will need access to the internet to commit and fetch any code changes, as well as a basic text editor and web server to write and test code.

## **6.4 PRODUCT ACCEPTANCE**

Each week, team members will take turns presenting progress made on the product to the customer in a face to face meeting. Customer defined standards (rubric) will be used to determine acceptability of the deliverables presented in face to face meetings. Deliverables will be uploaded online to Blackboard on a weekly basis, and kept available on a Google Drive account. All deliverables will be delivered by the project manager on behalf of the whole team.

# **SECTION 7. SUPPORTING PROCESSES**

## **7.1 CONFIGURATION MANAGEMENT**

All PHP source code files for the product will be placed in version control. Change requests will be handled during weekly group meetings, or alternatively through Github's built in system. Any large changes to the overall project will be approved by the team and presented during weekly updates.

## **7.2 INDEPENDENT VERIFICATION AND VALIDATION**

Automated testing and unit tests are not a requirement, but are highly desirable. Writing and performing tests, if implemented, would be completed using PHPUnit.

## **7.3 DOCUMENTATION**

All team members will be responsible for contributing to and reviewing the documents listed below.

TABLE 7-1. BookIt DOCUMENTATION

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Type** | **Format Standard** | **Estimated Page Count** | **Peer Review Type** |
| User Manual |  | 5 |  |
| Software Requirements Specification | IEEE 1058 | 15 |  |
| Source code | PHP, SQL |  |  |
| Presentation, written report |  |  |  |

## **7.4 QUALITY ASSURANCE**

Quality assurance will be performed at the end of each sprint by each team member. Unit tests are desired for the code, but are currently not a requirement. Regular updates will be given weekly on the status of the project and its completion.

## **7.6 PROBLEM RESOLUTION**

Problem reporting, tracking, and resolutions will occur either on Github or in weekly group meetings. Github's built-in issue tracker allows for the assignment of issues to specific team members, opening of a new issues, discussion, and log of activities, making it easy to coordinate as a team. Team members will discuss together and agree on a level of severity for each problem.

## **7.8 PROCESS IMPROVEMENT**

The project will be assessed every 2-3 weeks during group’s weekly meeting. Reoccurring problems will be analyzed by an assigned team member and tasked with presenting possible solutions during the following week’s meeting. All team members will be responsible for engaging in discussing possible solutions and, if finding a reasonable solution, working on implementing the solution.