

**Computer Science and Engineering**

**ItWorks**

**Project Management Plan**

**Version 2.0**

Document Number: SPMP-001

Project Team Number: B28

Project Team Members:

|  |  |
| --- | --- |
| Rong Fang | rsf288 |
| Tian Lin | tcl344 |
| Lisa Frankel | lhf238 |

**REVIEW AND APPROVALS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Printed Name and Title** | **Function (Author, Reviewer, Approval)** | **Date** | **Signature** |
| Tian Lin | Author | February 15, 2018 |  |
| Rong Fang | Author | February 15, 2018 |  |
| Lisa Frankel | Author | February 15, 2018 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**REVISION LEVEL**

|  |  |  |
| --- | --- | --- |
| **Date** | **Revision Number** | **Purpose** |
| February 15, 2018 | Version 2.0 | Revised Release |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

1. **OVERVIEW……………………………………………………………………………………….……..6**
   1. PROJECT SUMMARY……………………………………………………………………….…..6
   2. PURPOSE, SCOPE, AND OBJECTIVES……………..…………………….…………………6
   3. ASSUMPTIONS AND CONSTRAINTS …………………….……………………………........7
   4. PROJECT DELIVERABLES.…………………………………………….…………...………....7
   5. SCHEDULE AND BUDGET SUMMARY.………………………………………………..…….7
2. EVOLUTION OF THE PLAN.……………………………..……………………………..….…….…..8
3. **REFERENCES…………………………………………………………………………………………..8**
4. **DEFINITIONS……………………………………………………………………………………….…...8**
5. **PROJECT ORGANIZATION……………………………………………………………………...…...8**
   1. EXTERNAL INTERFACES.……………………………………………………………………...8
   2. INTERNAL STRUCTURE.…………………………………………………………….…..…....9
   3. ROLES AND RESPONSIBILITIES ………………………………………………..….……….9
6. **MANAGEMENT PROCESSES………………………………………………………………………..9**
   1. START-UP PLAN…………………………………………………………..……………………..9
      1. *Estimation Plan*…………………………………………………………..……………..9
      2. *Staffing Plan*…………………………………………………………..………….….10
      3. *Resource Acquisition Plan*…………………………………………………………..10
      4. *Training Plan ………………………………………………...……….…………..…..10*
   2. WORK PLAN …………………………………………………………………………….……...10
      1. *Work Activities ……………………………………………………………...………...10*
      2. *Schedule Allocation ………………………………………………...………………..11*
      3. *Resource Allocation ……………………………………………………………..….11*
      4. *Budget Allocation*
   3. CONTROL PLAN……………………………..…………………..………………………..…..11
      1. *Requirement Control and Traceability …………………………………………......12*
      2. *Schedule Tacking and Adjustment …………………………..……...………....….12*
      3. *Budget Tracking and Adjustment…………………………..………...………....….12*
      4. *Quality Control ………………………………………….…………………………….12*
      5. *Reporting Mechanisms…………………………….…………..……...………....….12*
      6. *Metrics Collection Plan…………………………………………….…………………12*
   4. RISK MANAGEMENT PLAN……………………………………….…………………….…....13
   5. POST IMPLEMENTATION PLAN*…………………………..……...………....……………....14*
7. **TECHNICAL PROCESSES…………………………………………………………………………..14**
   1. PROCESS MODEL …………………………………………………………….……………...14
   2. METHODS, TOOLS, AND TECHNIQUES ………………………….……………………....14
   3. INFRASTRUCTURE PLAN ……………………………………………………………….…...15
   4. PRODUCTION ACCEPTANCE AND MIGRATION PLAN…………………………….…...15
8. **SUPPORTING PROCESSES PLANS ……………………………………………………………...15**
   1. CONFIGURATION MANAGEMENT PLAN………………………………………………...15
   2. QUALIFICATION (VERIFICATION AND VALIDATION) PLAN ………………………….15
   3. DOCUMENTATION (LIBRARY) PLAN ………………………………………..…………...16
   4. QUALITY ASSURANCE PLAN ……………………………………………………………..16
   5. REVIEWS AND AUDITS…………………………………………………………………..…...16
   6. PROBLEM RESOLUTION PLANS …………………………………………………….……..17
   7. ENVIRONMENT MANAGEMENT PLANS…………………………….…............................17
   8. PROCESS IMPROVEMENT PLAN

**13. APPENDICES……………….…………………………………………………..……………………..17**

13.1 SCHEDULE TRACKING…………………………………………………………….……….17

13.2 DEFECT TRACKING ……………………………………………………………..……...….19

13.3 GANTT CHART/MICROSOFT PROJECT SCHEDULE……………………………...…..22

1. **OVERVIEW**
   1. **Project Summary**

ItWorks is a mobile/web based application that values user input. When conventional solutions don’t work, there is no good way to seek alternative solutions. ItWorks will not only enable its users to look for answers, but also will have a feedback system that will let users know if the answers they find were effective. We will create a crowd sourcing platform that allows users to access tried-and-tested, yet unconventional, methods of solving problems that come about in day-to-day life.

The purpose of this SPMP is to provide a detailed description of the project’s organization, as well as the processes that will be put in place in order to develop this product. This SPMP will explain the external and internal organization as well as the roles and responsibilities for the project. This SPMP will also explain the management process, technical and supporting processes plans.

The intended audience of this SPMP is the developers of the system, corporate executives, project management, and the SQA team.

* 1. **Purpose, Scope, and Objectives**

***Purpose***

The purpose of this project is to develop a community based website where users can crowd source answers to their problems.

***Scope***

ItWorks is a mobile/web based application that values user input. When conventional solutions don’t work, there is no good way to seek alternative solutions. Users can post advice and search for answers. ItWorks will not only enable its users to look for answers, but also use a feedback system that lets users know if the answers they find are effective. Additionally, there will be a reputation system, in which other users vote on whether a piece of advice is useful or not. This makes the application more engaging.

It is a crowd sourcing platform that allows users to access tried-and-tested, yet unconventional methods of solving problems that come about in day-to-day life.

**Objectives**

Project Priority: Quality and on schedule

Type of delivery: Evolutionary

Initial deliverable:

1. Project Proposal – 9/21/17

2. Initial SRS – 10/17/17

3. Final SRS – 10/24/17

4. Software Project Management Plan (SPMP) - 11/7/17

5. Software Analysis Specification (SAS) - 11/25/17

6. Software Design Document (SDD) - 3/7/18

Final deliverables:

* + 1. Project Proposal 2.0 – 2/7/18
    2. Requirements and Analysis Documentation (RAS) – 2/7/18
    3. Project Management Plan (SPMP) 2.0 – 2/14/18
    4. Software Design Document Final (SDD 2.0 and code) – 4/25/18
  1. **Assumptions and Constraints**

It’s assumed that users will be using an IOS or Android phone and are able to connect to the internet. Our project has to be developed by our team of four.

* 1. **Project Deliverables**
     + Project Proposal - 9/21/17
     + Initial SRS - 10/17/17
     + Final SRS - 10/28/17
     + Software Project Management Plan (SPMP) - 11/7/17
     + Software Analysis Specification (SAS) - 11/25/17
     + Project Proposal 2.0 – 2/7/18
     + Requirements and Analysis Documentation (RAS) – 2/7/18
     + Project Management Plan (SPMP) 2.0 – 2/14/18
     + Software Design Document (SDD) – 3/7/18
     + Software Design Document (SDD) 2.0 - 4/25/18
  2. **Schedule and Budget Summary**

Our schedule follows the iterative waterfall processes, breaking our project into six distinct phases. The timeline is represented in the schedule table. This table represents the first occurrence of each phase. Requirements analysis, design, implementation, verification/testing, and maintenance will be cycled through every 2-3 months.

*Schedule Table*

|  |  |
| --- | --- |
| **Phases** | **Dates** |
| Finish planning | 9/21/17 |
| First requirement analysis | 10/28/17 |
| Complete initial design | 12/15/17 |
| Begin implementation | 1/22/18 |
| Verification / Testing | 2/22/18 |
| Maintenance | 3/22/18 |

Budget Summary

|  |  |
| --- | --- |
| Anticipated Spending | Current Spending |
| $0 | $0 |

1. **EVOLUTION OF THE PLAN**

Scheduled and unscheduled changes will be determined by the completeness of the project at each milestone. Defect metrics will be used to quantify the amount of change required and plan accordingly. The planned changes will require approval from the project management and then it will be inspected, redistributed and the version number updated

1. **REFERENCES**

Team B28, ItWorks Project Proposal, V 1.0, 9/21/2017

Team B28, ItWorks Initial SRS, V 001, 10/17/2017

Team B28, ItWorks Final SRS, V 002, 10/28/2017

Team B28, ItWorks Software Project Management Plan (SPMP) - 11/7/17

Team B28, ItWorks Software Analysis Specification (SAS) - 11/25/17

Team B28, ItWorks Software Design Document (SDD) - 3/7/18

Team B28, ItWorks Project Proposal 2.0 – 2/7/18

Team B28, ItWorks Requirements and Analysis Documentation (RAS) – 2/7/18

1. **DEFINITIONS**

**Iterative Waterfall Process** - Repetition of the basic waterfall process: requirement, design, implementation, testing, maintenance.

**Reputation** - a metric which defines a user’s effectiveness as solving problems

**Post** - a body of text which a user shares with the ItWorks community

**Voting** - a system which allows users to vote for or against the usefulness of a solution

**CI** = Configuration Item

**QA** = Quality Assurance

**SAS** = Software Analysis Specification

**SDD** = Software Design Document

**RAS** = Requirements Analysis Specification

**SPMP** = Software Project Management Plan

1. **PROJECT ORGANIZATION** 
   1. **External Interfaces**

The external interfaces are SQA, Project manager, client, etc.

* 1. **Internal Structure**

The internal structure of the project will be organized as a team of developers. The organization structure is democratic.Each developer will have a specific role within the team. These roles are team leader, implementation leader, product manager, and design leader. There is also an advisory board, consisting of two advisors.

* 1. **Roles and Responsibilities**

Each of the documents will be kept up to date throughout their life cycles by the personnel in charge of each of the roles. The person responsible for a document will also ensure that the document is created and complete on schedule.

Table 5.3.1 - Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| Roles | Responsibility | Primary Leader | Secondary Leader |
| Team leader | SPMP | Rong Fang | Tian Lin |
| Implementation leader | SAS | Tian Lin | Lisa Frankel |
| Requirements Management Leader | SRS | Lisa Frankel | Tian Lin |
| Design leader | SDD | Lisa Frankel | Rong Fang |
| Advisory board | Advise | Professor Strauss | Vaishali Pari |

1. **MANAGEMENT PROCESSES** 
   1. **Start-up Plan**

In this section, you will find oru estimation plan, staffing plan, resource acquisition plan, and training plan.

* + 1. **Estimation Plan**

What is added to doc:

We have determined which software and services we will use in order to build our software project. They are as follows: Google docs, github, Slack, Sublime text, VirtualBox, AWS, and GoDaddy. To estimate our costs, we will reach out to Amazon for a quote for the cost of data storage, servers, and web hosting. We will also determine which domain name we are going to use so that we can get a quote for the domain name. All other software and services we will use are free resources. There is a risk that the cost of the domain name can change, or the domain name can be bought before we can buy it. We could also run out of storage on Google docs, in which case we would have to purchase a data storage plan.

Notes

* + - 1. Domain name - buy from goDaddy $10-$40
      2. AWS - data storage, servers, and web hosting - waiting to hear from AWS about pricing for our needs
      3. Linux VM - VirtualBox which is free
      4. Sublime Text - free
      5. Slack - Free
      6. Google Docs - Free
      7. Github - Free
    1. **Staffing Plan**

The development of our application demands only three software engineers.  They must trained in Go, C++, HTML & CSS.  These three software engineers will be responsible for all documentation and implementation of the application.  They will work from September 2017 - May 2018.  Once the application is live, this application will require a single moderator to review flagged posts.  A moderator will be needed for the full time that the application is live, and the application may require more moderators as the application user base grows.  Please refer to the Gantt chart in section 13 to see current staffing work distribution.

* + 1. **Resource Acquisition Plan**

Hardware & equipment: three computers, which are already acquired.

Software: each computer will need to install VirtualBox and Sublime Text, which are both free and can be acquired online. Google docs, github, and slack accounts have already been acquired.

Services: AWS needs to be acquired by first getting a quote from Amazon, and then being purchased. A domain name must be purchased through the GoDaddy website.

All of the above that is not already acquired must be acquired by March 20th so that the team may begin development.

* + 1. **Training Plan**

Team members should be taking or have completed a software engineering course at NYU Tandon. All team members who lack programming knowledge in C++ or Go shall spend two week learning the language they need to code in.

* 1. **Work Plan**

This section specifies the work activities, schedule, and resources for the project. Refer to Gantt chart (section 13.3) for activities and schedule. Human resources is in the work activities chart 6.2.1)

* + 1. **Work Activities**

Software Development

1. Project management
   1. Project Proposal
      1. Write sections
      2. Inspection
      3. Rework
      4. Post
   2. Requirement
      1. Initial SRS
         1. Write sections
         2. Inspection
         3. Rework
         4. Post
      2. Final SRS
         1. Rework
         2. Post
   3. SPMP
      1. Write sections
      2. Inspection
      3. Rework
      4. Post
   4. Design
      1. SAS
         1. Write sections
         2. Inspection
         3. Rework
         4. Post
      2. SDD
         1. Write sections
         2. Inspection
         3. Rework
         4. Post
      3. **Schedule Allocation**

Refer to Gantt chart (section 13.3)

* + 1. **Resource Allocation**

Refer to Gantt chart (section 13.3)

* + 1. **Budget Allocation**

As our staff are all founders and we have not yet received funding, all staff are working for free. Therefore, there is no budget for personnel.

|  |  |
| --- | --- |
| **Software / Service** | **Cost** |
| AWS | Waiting for Amazon to reply with estimate |
| GoDaddy Domain Name | $10 - $40 |

* 1. **Control Plan**

The team will meet at the beginning of each development phase. Additionally, the team will meet each Monday from 6:00pm to 8:00pm. The team will also meet on Fridays from 12:00pm to 2:00pm (if determined necessary at the Monday meeting). The team leader will send a google calendar invite for the Friday meeting to all team members if the Friday meeting is to take place.

* + 1. **Requirement Control and Traceability**

Each requirement will be traceable forwards and backwards. In documents, there will be section numbers so that it can be traced in the future. When we code any object we will have in the header, the requirement that it supports, so that it can be traced back to the requirements.

* + 1. **Schedule Tracking and Adjustment**

The project leader will report to team members the status of the schedule during every meeting, and if there is need for adjustments.

The schedule tracking and adjustment for deliverables / artifacts can be found in section 12.1.

* + 1. **Budget Tracking and Adjustment**

The project leader will report to team members the status of the budget during every meeting, and if there is need for adjustments.

The budget tracking and can be found in section 13.

* + 1. **Quality Control**

The leader of the QA team (not currently a part of the team) will report to the team leader each Wednesday about the quality and found defects of the project. The last work session will be an inspection for defects and correction afterwards.

There will be various stages of review. There will be self-checks, peer review, walkthroughs and inspections to review this document. This document will be reviewed for quality based on the metrics in section 6.3.6

The defect tracking for deliverables / artifacts can be found in section 13.2.

* + 1. **Reporting Mechanisms**

The project team meets every Thursday at 1:30 to discuss the status of the project. Lisa is responsible for booking a room in the library. Rong informs the group of the status of milestones. Tian inform the group of schedule and budget status and adjustments. There is currently no customer to report to.

* + 1. **Metric Collection Plan**

The QA team will collect metrics about the project, and will report it to the team leader. After every work session, the number of work hours will be documented on a schedule tracking chart. For every defect found, the metric will be updated in the defect tracking chart. The reporting plan can be seen in section 6.3.4.

The following metrics will be used to determine project quality:

* Correct
* Unambiguous
* Complete
* Consistent
* Stable
* Verifiable
* Modifiable (malleable)
* Traceable
  1. **Risk Management Plan**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Title | Estimated likelihood of occurring (L:1-10 with 1 lowest likelihood | Estimated impact (I: 1-10 with 1 lowest impact) | Estimated cost of managing (M:1-10 with 1 lowest cost) | Priority number (lowest number handled first) (11-L) \*(11-I) \*M | Retirement plan | Responsible person | Target completion date |
| 1 | Database is corrupted or down | 2 | 9 | 4 | 72 | See Note 1 | Rong | 5/1/2018 |
| 2 | Team member unable to meet schedule due to emergencies | 8 | 5 | 3 | 54 | See Note 2 | Lisa | Continual |
| 3 | Unable to handle amount of users because of incorrect size estimate | 6 | 8 | 3 | 45 | See Note 3 | Tian | 5/1/2018 |
| 4 | Product deliverables produced behind schedule | 5 | 10 | 6 | 36 | See Note 4 | Lisa | Continual |
| 5 | Product does not meet customer requirement | 3 | 10 | 5 | 40 | See Note 5 | Rong | Continual |

1. Insure our data will be stored redundantly.
2. Reschedule to deliver on time.
3. Have plan in place for expanding server and bandwidth so that it's easy to quickly retire risk.
4. Track metrics on progress so that product manager may intervene before it’s too late.
5. Agile process ensures constant communication with customer to ensure product meets their requirements.
   1. **Post Implementation Plan**

All code will be archived on github. We will hold a final meeting to have a post-mortem debriefing. During this meeting, we will schedule a date to hold our Post Implementation Review.

1. **TECHNICAL PROCESSES** 
   1. **Process Model**

The project will be done using an iterative waterfall process. The first iteration will produce a working prototype and will be fully documented. The second iteration will produce version one of the ItWorks application. Subsequent iterations will be decided after evaluating product’s quality at version one.

* 1. **Methods, Tools, and Techniques**

The app will be written in object oriented style

Documentation will be done mainly with google docs, and transferred to Microsoft Word if required for deliverables.

Programming language: C++, Go

Compiler: GCC

Operating System: Ubuntu 16 LTS

Repository: GitHub

Communication: slack

Task tracking: taiga.io

Project tracking: Microsoft Project

Documentation: Google Docs, Microsoft Word

Techniques: Agile, UML

* 1. **Infrastructure Plan**

A Linux system image will be created to clone all test environments and made replicate-able in VMs. A backup server will be maintained along with the main server.

* 1. **Product Acceptance and Migration Plan**

Before launching the application, we will have 25 people representing different demographics test the application. They will create an account, and spend a week using the application. They will be instructed to test out all of the functionality of the app (e.g. posting, searching, commenting, etc.). Their feedback will be recorded, and a meeting will be held to discuss the feedback and determine how to incorporate it into the application. Once developed, we will bring this new version of the application back to those 25 people, to see if they believe the application has improved. Once our test group is satisfied, the application will be launched.

1. **SUPPORTING PROCESSES PLANS**
   1. **Configuration Management Plan**

All files and documentations will be stored on GitHub. Each change made will require editor to create a new pull request into a sub branch of the repository. The files will only be merged when a full inspection is conducted.

The app version will be control via X.Y.Z

X will be the major baseline version number

Y will be used when there is significant change in the software base on the current baseline

Z will be used for nightly builds / timely minor updates

* 1. **Qualification (Verification and Validation) Plan**

**Verification Tools:**

Trace-ability: All source codes and documents will use SRS section and subsection numbers as reference for project tracing.

Milestone reviews: conducted at the end of each milestone with a thorough inspection by the development team and the SQA team.

Walkthroughs: conducted weekly by the team leader and the developer through testing and analyzation.

Peer reviews: conducted daily by other developers to test and analyze for defects.

Prototyping: conducted when a baseline is developed. A demonstration of the software and a full inspection will be conducted on the software to create a near perfect baseline.

Simulation: the app will be run in real and simulated mobile environments to make sure the app is adaptable to all test environments.

Modeling: conducted at half of a milestone with a thorough inspection to check if the app follows the written guidelines, diagrams and charts.

* 1. **Documentation (Library) Plan**

Each team member is responsible for documenting their work. He/She will do a self-review before distributing it to other team members. Team members will do a peer review and notifying any defects they found. Before each milestone, every unapproved document will go through a team inspection, and approved as the baseline version.

Source codes will need to be commented to describe the purpose of every function. Source codes will go through the same review and approval process as documents.

* 1. **Quality Assurance Plan**

There will be various stages of review. There will be self-checks, peer review, walkthroughs and inspections to review this document. This document will be reviewed for quality based on the following metrics:

* + Correct
  + Unambiguous
  + Complete
  + Consistent
  + Stable
  + verifiable
  + Modifiable (malleable)
  + Traceable
  1. **Reviews and Audits**

Peer review: conducted daily by developer at end of workday. Reviewing subjects include codes and tasks completed.

Walkthrough: done weekly between the developer and the team leader to check if the code strictly follows the requirements and to find defects.

Inspection: conducted at each half and a full milestone by the development team and SQA team to verify project correctness of the code. Each individual will be assigned a role to conduct review on one subject matter only.

Audit: Documentations and logs will be checked daily by the team leader to stay up to date with the changes and to find any error that may exist in documents and logs.

* 1. **Problem Resolution Plans**

Each developer will have to save a version of his code after fixing defects, and at least once every work hour. The developer will save the version using the GitHub repository system. Any software problem the developer is unable to fix during his work session shall be documented and reported to the development team.

* 1. **Environment Management Plans**

The development team will work on the project in the school library and from home.  They will use their laptops in both scenarios, which will have VirtualBox and Sublime Text installed, and to access Slack, Google Docs, Github, AWS. Testing will not be done in a specific location, rather a distributed approach to asking people to try out the application.

* 1. **Process Improvement Plan**

During weekly meetings on Thursday, the team will discuss the status of the project.  In this meeting, members are encouraged to bring up any defects or bugs that they’ve encountered.  Each week, members should take an hour to look for these defects before the meeting.  We will also take time to discuss any adjustments (e.g. adding or removing features).  Before each code inspection, the members of the team will meet to discuss what is and isn’t working with the software, and ensure that the software meets the requirements.  We will also make sure that everything will be completed on time, by frequently checking if project assignments are on schedule.

**13 APPENDICES**

**13.1 Schedule Tracking**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| Initial SRS | Tian Lin | 12 | 10 | 2 less |
|  | Rong Fang | 12 | 10 | 2 less |
|  | Lisa Frankel | 12 | 9 | 3 less |
|  | Alex Huang | 12 hrs | 9 hrs | 3 less hrs |
|  | Summary for entire team | 48 | 38 | 10 less |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| Final SRS | Tian Lin | 2 | 2 | 0 |
|  | Rong Fang | 2 | 3 | 1 more |
|  | Lisa Frankel | 2 | 3 | 1 more |
|  | Alex Huang | 2 hrs | 0 hrs | 2 less hrs |
|  | Summary for entire team | 8 | 8 | 0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| SPMP | Tian Lin | 12 | 11 | 1 less |
|  | Rong Fang | 12 | 10 | 2 less |
|  | Lisa Frankel | 12 | 10 | 2 less |
|  | Alex Huang | 12 | 9 | 3 less |
|  | Summary for entire team | 48 | 40 | 8 less |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| SAS | Tian Lin | 12 hrs | 10 hrs | 2 less hrs |
|  | Rong Fang | 12 hrs | 10 hrs | 2 less hrs |
|  | Alex Huang | 12 hrs | 10 hrs | 2 less hrs |
|  | Lisa Frankel | 12 hrs | 10 hrs | 2 less hrs |
|  | Summary for entire team | 48 hrs | 40 hrs | 8 hrs less |
| Artifact or  Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| Initial RAS | Tian Lin | 4 hrs | 2 hrs | 2 less hrs |
|  | Rong Fang | 4 hrs | 2 hrs | 2 less hrs |
|  | Lisa Frankel | 4 hrs | 2 hrs | 2 less hrs |
|  | Summary for entire team | 12 hrs | 6 hrs | 6 less hrs |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| SPMP 2.0 | Tian Lin | 4 hrs | 3 hrs | 1 less hrs |
|  | Rong Fang | 4 hrs | 3 hrs | 1 less hrs |
|  | Lisa Frankel | 4 hrs | 3 hrs | 1 less hrs |
|  | Summary for entire team | 12 hrs | 9 hrs | 9 less hrs |

***Cumulative***

|  |  |  |  |
| --- | --- | --- | --- |
| Who | Estimated (hrs) | Actual (hrs) | Difference (hrs) |
| Tian | 46 hrs | 37 hrs | 9 less hrs |
| Rong | 46 hrs | 38 hrs | 8 less hrs |
| Lisa | 46 hrs | 38 hrs | 9 less hrs |
| Alex | 38 hrs | 28 hrs | 10 less hrs |

**13.2 Defect Tracking**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (# of defects found) | Actual (# of defects found) | Difference (# of defects found) |
| Initial SRS | Tian Lin | 9 | 2 | 7 |
|  | Rong Fang | 9 | 1 | 8 |
|  | Lisa Frankel | 9 | 1 | 8 |
|  | Alex Huang | 9 | 2 | 7 |
|  | Summary for entire team | 36 | 6 | 30 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (# of defects found) | Actual (# of defects found) | Difference (# of defects found) |
| Final SRS | Tian Lin | 5 | 1 | 4 less |
|  | Rong Fang | 5 | 1 | 5 less |
|  | Lisa Frankel | 5 | 0 | 5 less |
|  | Alex Huang | 5 | 0 | 5 less |
|  | Summary for entire team | 30 | 2 | 28 less |
| Artifact or  Deliverable | Who | Estimated (# of defects found) | Actual (# of defects found) | Difference (# of defects found) |
| SPMP | Tian Lin | 9 | 7 | 2 less |
|  | Rong Fang | 9 | 5 | 4 less |
|  | Lisa Frankel | 9 | 3 | 6 less |
|  | Summary for entire team | 36 | 18 | 18 less |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Artifact or***  ***Deliverable*** | ***Who*** | ***Estimated (# of defects found)*** | ***Actual (# of defects found)*** | ***Difference (# of defects found)*** |
| ***SAS*** | ***Tian Lin*** | ***9*** | ***3*** | ***6*** |
|  | ***Rong Fang*** | ***9*** | ***2*** | ***7*** |
|  | ***Alex Huang*** | ***9*** | ***1*** | ***8*** |
|  | ***Lisa Frankel*** | ***9*** | ***2*** | ***7*** |
|  | ***Summary for entire team*** | ***36*** | ***8*** | ***28*** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artifact or  Deliverable | Who | Estimated (# of defects found) | Actual (# of defects found) | Difference (# of defects found) |
| RAS | Tian Lin | 5 | 4 | 1 |
|  | Rong Fang | 5 | 3 | 2 |
|  | Lisa Frankel | 5 | 1 | 4 |
|  | Summary for entire team | 15 | 8 | 7 |

***Cumulative***

|  |  |  |  |
| --- | --- | --- | --- |
| Who | Estimated (# of defects found) | Actual (# of defects found) | Difference (# of defects found) |
| Tian Lin | 37 | 17 | 20 less |
| Rong Fang | 37 | 12 | 25 less |
| Lisa Frankel | 37 | 6 | 26 less |
| Alex Huang | 32 | 7 | 30 less |
| Team Summary | 128 | 26 | 101 less |

**13.3 Gantt Chart/Microsoft Project Schedule**



