**Team D- Phoenix**

**PROJECT MANAGEMENT PLAN**

**Project Name:** Project Management collaboration tool

**Team Name:** Phoenix **Prepared date: 26/06/2017**

**Project Overview:**

Collaboration tool is an integration of variety of software’s and online services used by group of people to work on together. There are several components like messaging, file sharing, shared calendar, task management etc. which allows users to work simultaneously regardless of their physical location.

**Project Deliverables:**

We need to deliver a working standalone application meeting the requirements specifications and functionality of the client.

List of items to be delivered:

1. Standalone Application
2. Requirements Document
3. User Manual
4. Installation Guide
5. Use cases
6. Test Suite
7. Test Plan
8. Other miscellaneous items

**Client Name:** Dr. Michael Oudshoorn

**Project Organization:**

**Organizational Structure:** Functional structure is being used. There is no hierarchy in this project and all are at similar level. In daily stand-up meetings, we report to one another.

**Organizational boundaries and interfaces:** Each member of the team was assigned a task supervision as their responsibility to keep them active and contribute towards the completion of the project.

**Project responsibilities:**

* Deliverables of the project should be developed by managing the triple constraints like time, scope, and cost constraints along with the standards and procedures of the organization.
* All the members in the team should always be supportive, engaging and work with the team by providing the required resources in the correct time.
* All the members should make team effort to finish the project on time and within budget. Attend meetings and be interactive with all the team members.

**Managerial process:**

* It is vital to think about the management objectives like objective of the top administration, needs of the project and think about the assumptions.
* The project should be monitored and keep track of all the changes, the project should be reviewed at regular intervals and the deliverables should be discussed with the client and formally accepted by the sponsor.

**Technical Processes:**

* The tools and techniques required for the project comes under technical processes. We should carefully choose the tools and techniques for the project as they vary from project to project and organization to organization. Determining them early could be helpful because we can work on it as project progresses forward.

**Budget Allocation:**

The most crucial resource for the project here is time, it is the ultimate budget we have. Our estimation for this project is approximately 7,000 lines of code. As of now, upon discussing with team we are estimating this figure.

Avg. person writes 1000 lines of code overall for the completion of this project. Which means typically a person writes 75 lines of code per week.

Each person will spend about 5 hours a day in weekdays and 7 hours in weekends. we will have team meeting of duration 2 hours per week and a client meeting for an hour per week which adds up to the total and makes 42 hours.

For each person

Weekdays: 5 X 5 = 25

Weekend: 2 X 7 = 14

Team meetings: 1 X 2 = 2

Client meetings: 1 X 1 = 1

**Total: 42 hrs. per week**

Therefore, we are spending **42 hours per person** **per week** which makes a total of 42X7 = 294 hours per week per team. So, the total budget of the project as of now is 294X6 = 1, 764 (6 weeks in GDP 1)

**Time Estimation for GDP 1:**

Per week per person: 42 X 1 = 42

Per week per team: 42 X 7 = 294

Total hours per team: 294 X 6 = **1,764 hours.**

Here, 6 is the total number of weeks the project is going to take in GDP 1.

In GDP 2, we are going to spend 15 hours per person per week i.e. 15X14 = 105 hours per week per team. Therefore, the total budget in GDP 2 is 105X12 = 1,260 (12 weeks in GDP 2)

**Time Estimation for GDP 2:**

Per week per person: 15 X 1 = 15

Per week per team: 15 X 7 = 105

Total hours per team: 105 X 14 = **1,470 hours.**

14 number of weeks for completing the project in GDP 2.

**Schedule:**

The project started on 06/12/17 and it will be completed by 12/15/17.

**Roles and Responsibilities of each member in the team:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Team Member | Role | Email | Contact No. | Responsibilities |
| Sanjay Bedudoori | Primary Contact | [S528106@nwmissouri.edu](mailto:S528106@nwmissouri.edu) | +1 660-528-0163 | Represents the entire team and primary contact to the client. Communicates project progress, issues, and changes with the client. |
| Venkata Bhardwaj Avasarala | Client Management | [S528103@nwmissouri.edu](mailto:S528103@nwmissouri.edu) | +1 660-528-0954 | Responsible for meeting with a client, give updates to the client and get requirements from client. |
| Sudharshan Reddy Kankara | Requirements management | [S528138@nwmissouri.edu](mailto:S528138@nwmissouri.edu) | +1 660-528-0849 | Responsible for requirements gathering, maintaining and updating them as per client. Also, makes sure that these requirements are met. |
| Shravani Alampalli | Issues management | [S528100@nwmissouri.edu](mailto:S528100@nwmissouri.edu) | +1 484-644-8469 | Responsible for tracking all the issues and ensures that they are resolved on time. |
| Hemanth Sai Kishore Nersu | Data Management | [S528158@nwmissouri.edu](mailto:S528158@nwmissouri.edu) | +1 660-528-0191 | Responsible for collecting, storing, managing the data in the database and connect data with the system. Manages all the data of the project including the database. |
| Anudeep Reddy Mallidi | Quality and testing management | [S528146@nwmissouri.edu](mailto:S528146@nwmissouri.edu) | +1 660-541-3937 | Responsible for handling quality and testing throughout the SDLC process. Manages test cases and provides appropriate deliverables for the test modules. |
| Vineeth Gajula | Communications and documentation management | [S528124@nwmissouri.edu](mailto:S528124@nwmissouri.edu) | +1 660-528-0509 | Manages all the documentation and communication for the project. Tracks all the changes and maintain them. |

|  |  |  |
| --- | --- | --- |
| List of proposed requirements | Assigned to | Percentage Completed |
| Shared document | ore Hemath Sai Kishore Nersu | 102 100 % |
| Shared Calendar | Vineeth Gajula | 100 % |
| Chat tool | Sudharshan Reddy Kankara | 100 % |
| Task creation | Sanjay Bedudoori | 100 % |
| Task Assigning | Sravani Alampalli | 100 % |
| Roles to team members | Anudeep Reddy Mallidi | 100 % |
| Google drive integration | Venkata Bhardwaj Avasarala | 100 % |
| System overview | Sanjay Bedudoori | 100 % |

**Scope Management Plan:**

1. **Collecting Requirements:** In the Scope Management Plan, the first step is to collect requirements. We have Acquired the requirements from the client by meeting with the client in weekly client meetings and having discussions with the client. We maintained a record log which contains all the notes taken from client in the client meetings and recorded the meeting session using voice recorder. We have documented each client meeting andthese documentation serves as input for our next client meeting.
2. **Define Scope:** The scope statement is developed in detail and all the project description, deliverables, problem statement etc. are identified.
3. **Create WBS:** The work breakdown structure for GDP 1 was developed which break downs all the project deliverables in a hierarchical manner.

**WBS for GDP 1:**



1. **Verify Scope:** Scope is verified where the final acceptance is done by the project sponsor by taking all the deliverables into the consideration.
2. **Control Scope:** Changes in the scope is monitored and controlled in this process. To eliminate scope creep, the changes are done carefully and properly integrated.

**Change Management Plan**

Changes made in this project can be done using a template in the following format. A formal change request form is filled and submitted so that the information regarding a change can be collected and tracked later.

Change Request Form

|  |  |
| --- | --- |
|  | |
| **Change Request ID** | ... |
| **Requested Change** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | Software | [ ] |  | Major | [ ] | | Hardware | [ ] |  | Minor | [ ] | |
| **Is this change an addition to a change or addition made previously ?** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | No | [ ] |  | Yes | [ ] | |
| **Problem Statement** Provide a brief description of the requested change |  |
| **Requester Details** | |  |  |  | | --- | --- | --- | | **Name** | **Position** | **Contact Information** | |  |  |  | |
| **Where & When Changed Requested** | |  |  |  | | --- | --- | --- | | **Date** | **Location** | **Time** | |  |  |  | |
| **Date Needed** (dd-mm-yyyy) |  |
| **Hyperlinks** |  |
| **Description of Problem**  Provide e detailed description of the problem, circumtances leading to the requested change |  |
| **Supporting Information**  Provide screenshots of an error or printout of an error in a document/report/software |  |
| **Reasons and Justification**  Describe the reason why the change has been requested and the justification for the request |  |
| **Affected Areas**  According to the Perception of the Requester | |  |  |  | | --- | --- | --- | | System Affected | Subsystem Affected | Documentation Affected | |  |  |  | |
| **Impact of Not Implementing Proposed Change** |  |
| **Alternate Actions**  Describe alternatives to the change according to the Perception of The Requester |  |
| **Priority to Implement**  Describe priority assigned by the requester – may be on Five Point Scale |  |

Customer OpenThink Labs

------------ ------------------

**Quality Management Plan**

Verification and validation plan:

* In this plan, all the components are carefully observed to ensure that they work as intended and properly meeting all the client requirements.
* It ensures that every module in the component is test and debugged to make it more efficient and feasible.
* It ensures that proper validations are done and ensures a very minimal number of errors in the code.

Quality Assurance plan:

* To ensure the accuracy of the source code, server-side interactions and hardware are properly handled.
* To ensure quality is achieved by minimizing the errors.
* Ensures that the software is bug-free and functioning as intended.
* Ensures that the system that runs this software doesn’t overload the other components and work smoothly.

Reviews and audits:

* Auditing is useful for getting the status of different project phases.
* Proper Assessment can be made by reviewing techniques.
* Standardization and abiding to business standards are made to ensure the consistency of the software.

Problem Resolution plan:

* To ensure that any errors or defects present in the development phase is handled in a proper manner.
* Ensures that every defect is documented and proper description is prepared to resolve the defect.
* Ensures that defects are resolved without causing any additional problems for the rest of the system components.

**Risk Management Plan**

In this plan, the identification and managing risks for this project is done. There are various types of risks identified for this project.

1. Risk which is associated with time and budget.
2. Risk associated with the expertise from the team members of the project.
3. Risk of losing the key personnel of the team due to medical issues which is also known as Risk of Human Resources.
4. Risk associated with the security of the user credentials.

**Risk Mitigation strategies**

We have few risk mitigation strategies to control and minimize the amount of impact associated with it.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Risk Category** | **Risk Mitigation Strategy** |
| 1 | Changes in the requirements | Client should be signing every requirement he/she is requesting for so that the changes can be monitored with the current status of the system during development phases. |
| 2 | Project Integration Risk | All the team members should properly aware of the integration of several project components to avoid this risk. Unit Testing can be done to eliminate this risk. |
| 4 | Scope Risk | Scope management plan has to be prepared well enough to overcome the scope risk. |
| 5 | Stakeholder Risk | Involvement of stakeholders in every aspect of the project is must so that they can understand the project in various levels. |
| 6 | Communication Risk | Communications should be standardized so that every team member can communicate with each other. Each member in team should be involved in the project activities. |
| 7 | Time Risk | Milestones and deadlines should be carefully monitored. Scheduling the tasks can help manage the time throughout the development life cycle. |
|  |  |  |
|  |  |  |

Updated Work Break Down structure with Roles and Responsibilities in GDP 2:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Task Mode | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
| **Manually Scheduled** | **Design** | **7 days** | **Mon 8/28/17** | **Tue 9/5/17** |  |  |
| Auto Scheduled | Reviewing the prototypes | 2 days | Mon 8/28/17 | Tue 8/29/17 |  | Sanjay,Shravani |
| Auto Scheduled | Changes to the protoypes | 1 day | Wed 8/30/17 | Wed 8/30/17 | 2 | Sudharshan,Bhardwaj |
| Auto Scheduled | Review the Software Architecture | 2 days | Thu 8/31/17 | Fri 9/1/17 | 3 | Vineeth,Sanjay |
| Auto Scheduled | Design the final Software Architecture | 2 days | Mon 9/4/17 | Tue 9/5/17 | 4 | Hemanth |
| **Manually Scheduled** | **Development** | **47 days** | **Wed 9/6/17** | **Thu 11/9/17** |  |  |
| Auto Scheduled | Review the System functional specifications | 4 days | Wed 9/6/17 | Mon 9/11/17 | 5 | Anudeep,Sudharshan |
| Auto Scheduled | Review use cases | 2 days | Wed 9/6/17 | Thu 9/7/17 | 5 | Vineeth |
| **Manually Scheduled** | **Coding for UI development** | **1 day** | **Fri 9/8/17** | **Fri 9/8/17** | **8** |  |
| Auto Scheduled | Login/Sign Up | 1 day | Fri 9/8/17 | Fri 9/8/17 |  | Anudeep |
| Auto Scheduled | Chat | 1 day | Fri 9/8/17 | Fri 9/8/17 |  | Sudharshan |
| Auto Scheduled | Shared Calendar | 1 day | Fri 9/8/17 | Fri 9/8/17 |  | Vineeth |
| Auto Scheduled | Task Management | 1 day | Fri 9/8/17 | Fri 9/8/17 |  | Sanjay |
| Auto Scheduled | Coding for server -side components | 5 wks | Fri 9/8/17 | Thu 10/12/17 | 8 | Everyone |
| Auto Scheduled | UI testing | 2 wks | Fri 10/13/17 | Thu 10/26/17 | 14 | Everyone |
| Auto Scheduled | Backend testing | 2 wks | Fri 10/27/17 | Thu 11/9/17 | 15 | Everyone |
| **Auto Scheduled** | **Documentation** | **2 days** | **Fri 11/10/17** | **Mon 11/13/17** |  |  |
| Auto Scheduled | Documenting the UI | 2 days | Fri 11/10/17 | Mon 11/13/17 | 16 | Sanjay,Vineeth,Sudharshan |
| Auto Scheduled | User Manual | 1 day | Fri 11/10/17 | Fri 11/10/17 | 16 | Shravani |
| Auto Scheduled | Installation Guide | 1 day | Mon 11/13/17 | Mon 11/13/17 | 19 |  |
| **Auto Scheduled** | **Testing** | **13 days** | **Tue 11/14/17** | **Thu 11/30/17** |  |  |
| Auto Scheduled | Unit Testing | 5 days | Tue 11/14/17 | Mon 11/20/17 | 20 | Everyone |
| Auto Scheduled | BlackBox Testing | 6 days | Tue 11/21/17 | Tue 11/28/17 | 22 | Everyone |
| Auto Scheduled | Changes after testing | 2 days | Wed 11/29/17 | Thu 11/30/17 | 23 | Sanjay,Vineeth |
| **Auto Scheduled** | **Deployment** | **4.25 days** | **Fri 12/1/17** | **Thu 12/7/17** |  |  |
| Auto Scheduled | Review the deliverables | 2 days | Fri 12/1/17 | Mon 12/4/17 | 24 | Shravani |
| Auto Scheduled | Deploy the deliverables | 1 day | Tue 12/5/17 | Tue 12/5/17 | 26 | Sudharshan |
| Auto Scheduled | Testing in real-time environment | 1 day | Wed 12/6/17 | Wed 12/6/17 | 27 | Bhardwaj |
| Auto Scheduled | Final Presentation to deliver the product | 2 hrs | Thu 12/7/17 | Thu 12/7/17 | 28 | Everyone |
| **Auto Scheduled** | **Closing** | **4 days** | **Thu 12/7/17** | **Wed 12/13/17** |  |  |
| Auto Scheduled | Lessons Learnt Report | 1 day | Thu 12/7/17 | Fri 12/8/17 | 29 | Hemanth |
| Auto Scheduled | Final changes in the project | 2 days | Thu 12/7/17 | Mon 12/11/17 | 29 | Anudeep |
| Auto Scheduled | Documenting the changes and distributing the software | 2 days | Mon 12/11/17 | Wed 12/13/17 | 32 | Sanjay |