**1. Introduction**

**1.1. Purpose of Plan**

The purpose of this plan is to evaluate the available resources, which are needed in 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 mitigations are reviewed. Moreover, the time line, milestones, responsibilities and roles of the team members and methods for quality assurance, reporting and backup plans are also specified.

**1.2. Project Scope and Objectives**

**1.2.1. Statement of Scope**

**1.2.2. Major Functions**

**1.3. Overview of Document**

The following 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, risk management are also included in the third and fourth section of the document. The schedule of the project, tracking and control mechanics followed by glossary are discussed in the further sections.

**2. Project Organization**

**2.1. Process Model**

**2.2. Team Structure**

Team members of our project are listed below.

* Nagendra Babu Pasupuleti
* Avinash
* Prakriti
* Gayatri Bhimani

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 be monitoring the project. Roles played by the team members are listed below.

* Team Leader (Nagendra)
* Documents Reviewer (Everyone)
* DB Developer (Avinash)
* DB Peer Reviewer (Gayatri)
* DB Tester (Prakrithi)
* UI Developer (Prakrithi, Nagendra)
* UI Peer Reviewer (Avinash)
* UI Tester (Gayatri)
* Core Developer (Nagendra, Avinash)
* Core peer Reviewer (Prakrithi, Gayatri)
* Core Tester (Prakrithi, Gayatri)

In this project we are following a controlled decentralized structure where the team leader will make decisions with the opinion of team members. The roles to be played by the team members will be decided by the team leader based on the skills and expertise and will set the expectation of that role for that situation and will estimate the outcome for that role.

**2.3. Communication**

The communication between the team members will be through e-mails and regular meetings. During team meetings, the status of the project is discussed, individual goals are set and analysis is done on the previously set goals. The problems and their possible solutions are explored. If the situation demands, micro teams are formed within the team. Checkpoints like milestones, timeline of the project etc., are checked. Individual and team log tables are maintained and they will available to all the team members. The source code is shared by using GitHub.

**2.4. Reporting**

The team leader is responsible for everything and in his/her absence next supporting leader will take the charge as a team leader. She/he arbitrates the conflicts between the team members. Issues are logged and were discussed in immediate team meeting. If things are get out of control, the course instructor, Dr.Stringfellow, will have the final word in resolving the issues.

**3.1. Hardware and Software**

The project development environment will be as follows:

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

* a Pentium 4 microprocessor running at 2.0 GHz or above
* a minimum of 128 megabytes of system RAM
* 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
* Microsoft Visual Studio 2015
* SQL Server
* Microsoft Project

The customer’s environment will be as follows:

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

* a Pentium 4 microprocessor running at 2.0 GHz or above
* a minimum of 128 megabytes of system RAM
* 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** utilized will include:

* Windows 2000 or above
* a web browser.

**4. Risk Management Plan**

**4.1 Likely risks:**

Table 1 shows a few of the listed possible potential risk.

Table 1. Risk Table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Chance** | **Impact** | **Strategy** |
| Illness of Team Members | Low | Tolerable | Allow a few days buffer. Redistribute work |
| Procrastination | Low | Serious | Have frequent meetings to monitor progress |
| Changes in Requirements | High | Tolerable | Additional effort of the team 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. |

Risk management is important for any project as it leads to many serious issues, so every member of the team should have the knowledge of risks. For this team is being introduced to C#, .net for the first time, but the team has good knowledge on OOPs concepts using C++ and on the user controls present in Visual Studio. In order to fix bugs and to add new features extra effort should be taken by the team.

**5. 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 environment and risk free environment.

Test cases for each Scrum will be created keeping in mind of all 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 these details of the test plan will be included in the test plan document.

**6.** **Schedule**

**6.1. Project Breakdown**

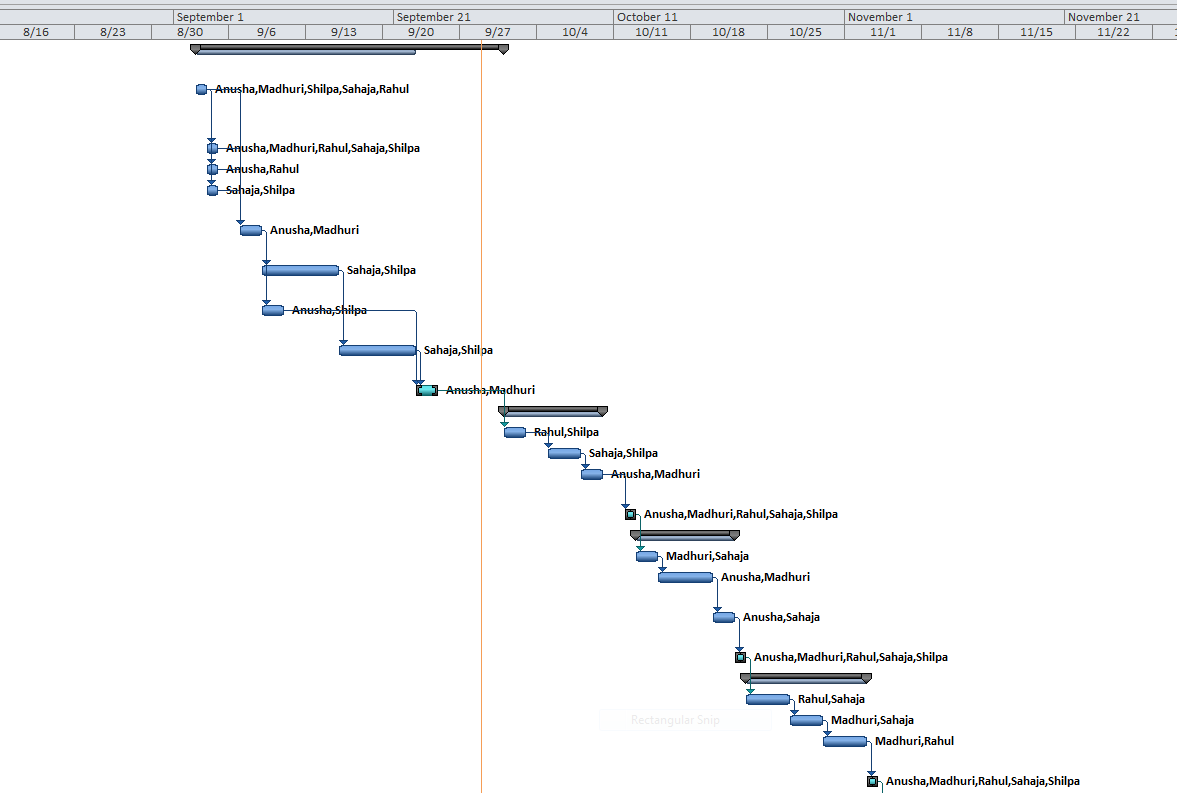
This Schedule will give the outline for the tasks to be performed in the project.

The project goes as follows:

* Gathering all the details form the client.
* Clarifying the doubts with the client.
* Creating the sprint outline
* Building a Sprint
  1. Designing sprint
  2. Developing sprint
  3. Execution and Testing sprint.
  4. Updating the changes in the sprint.
* Step 5 is performed for the rest of the sprints.
* Submitting the project to the client

**6.2. Time Line**

The time line divides the project into different tasks and gives detailed description of the project schedule by allocating specific time slots to each task as shown in Figure5. Team member is responsible for scheduling the task using Microsoft project. The project schedule is attached along with the network task diagram and the objective grading sheet.



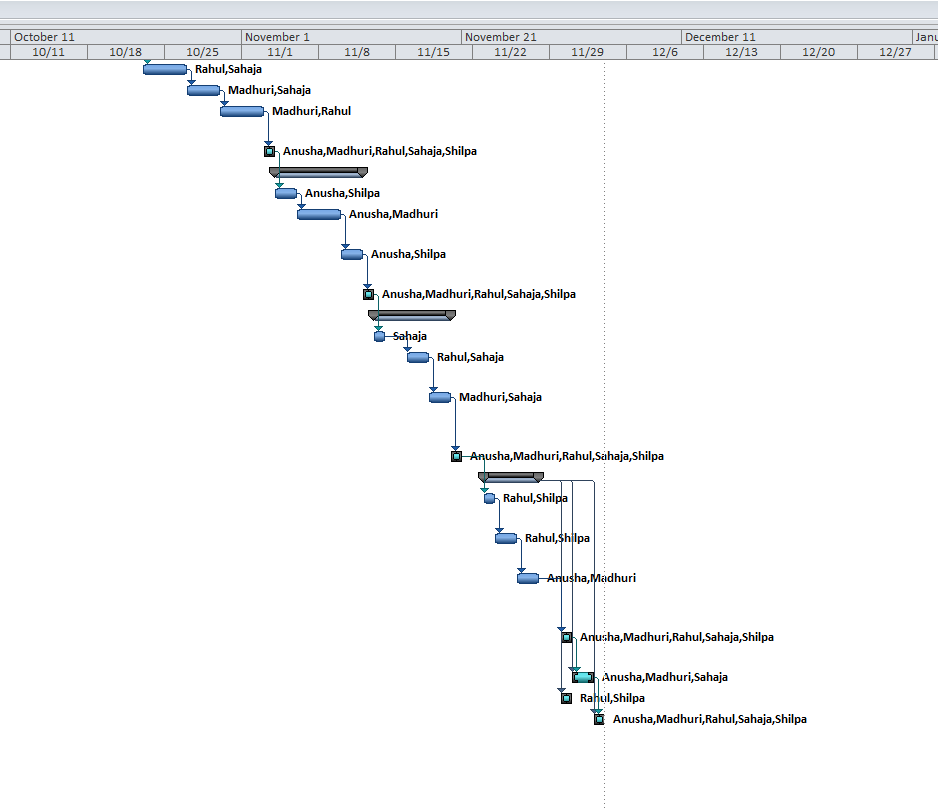
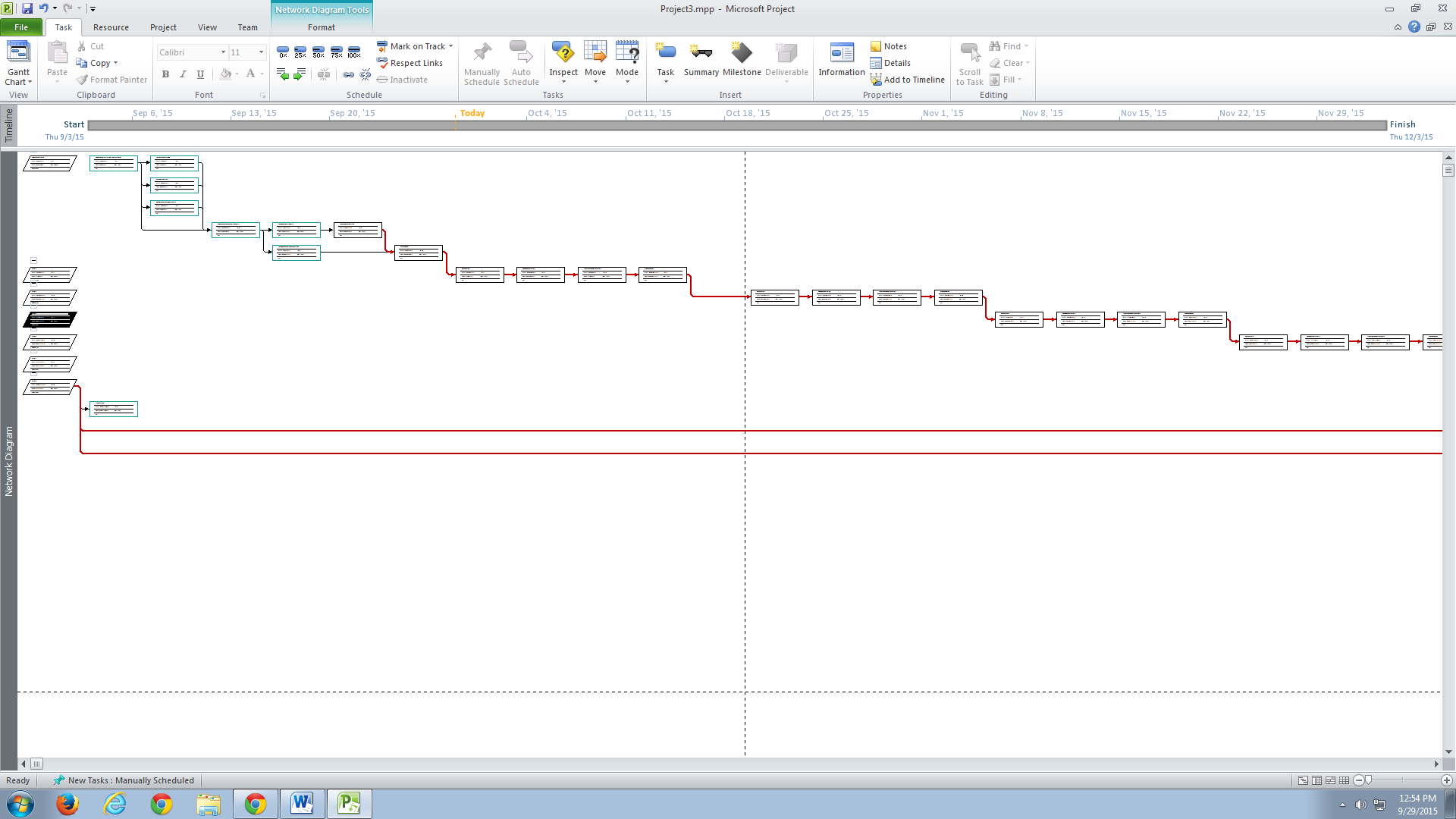


Figure 2. Time Line.

**6.3. Task Network Diagram**

The Figure 3 given below shows the tasks and their dependencies



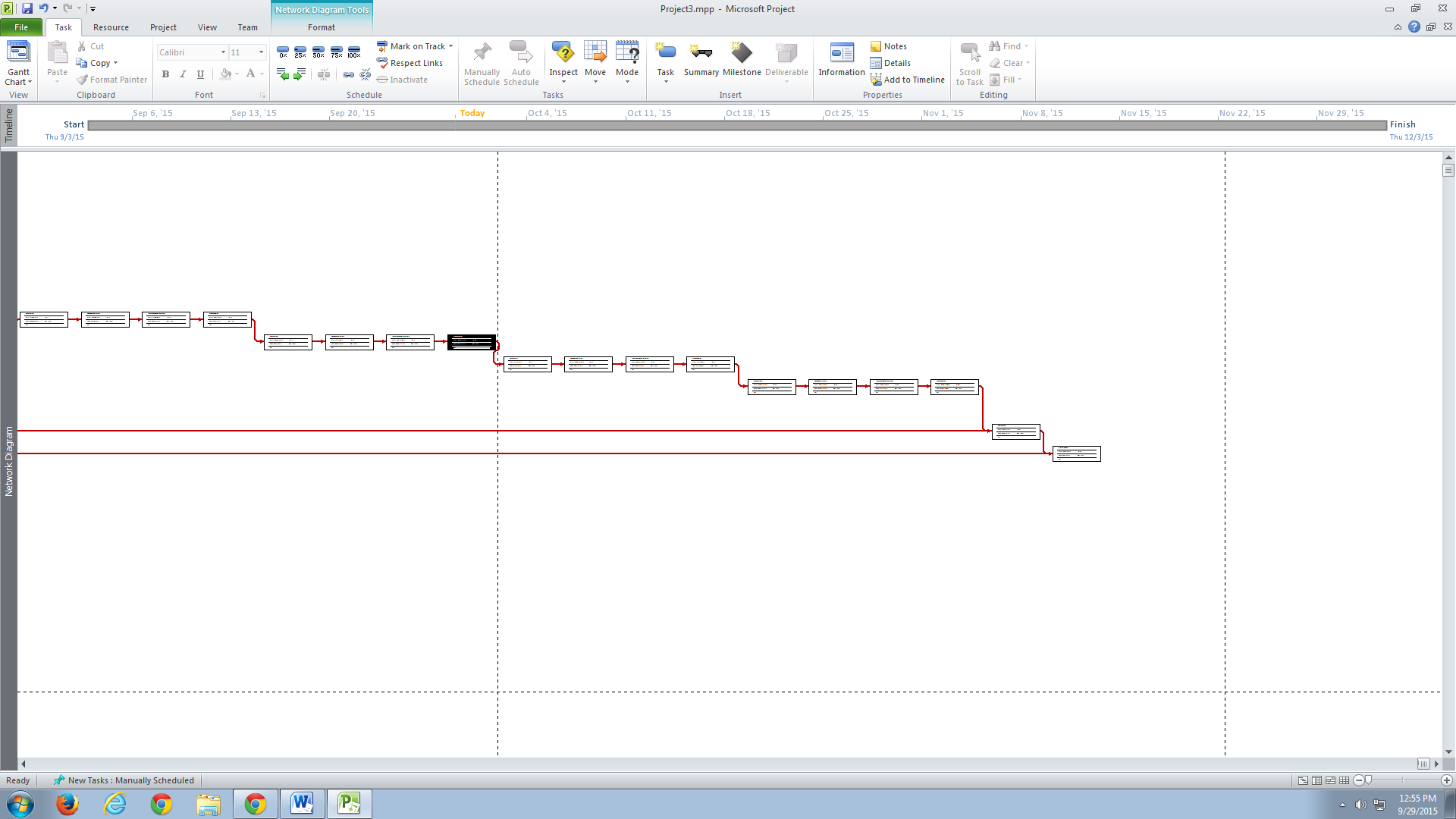


Figure 3. Task network diagram.

**6.4. Objective Grading Sheet**

Figure 4 represents grading sheet of team members and the bold cells are represents completed tasks.

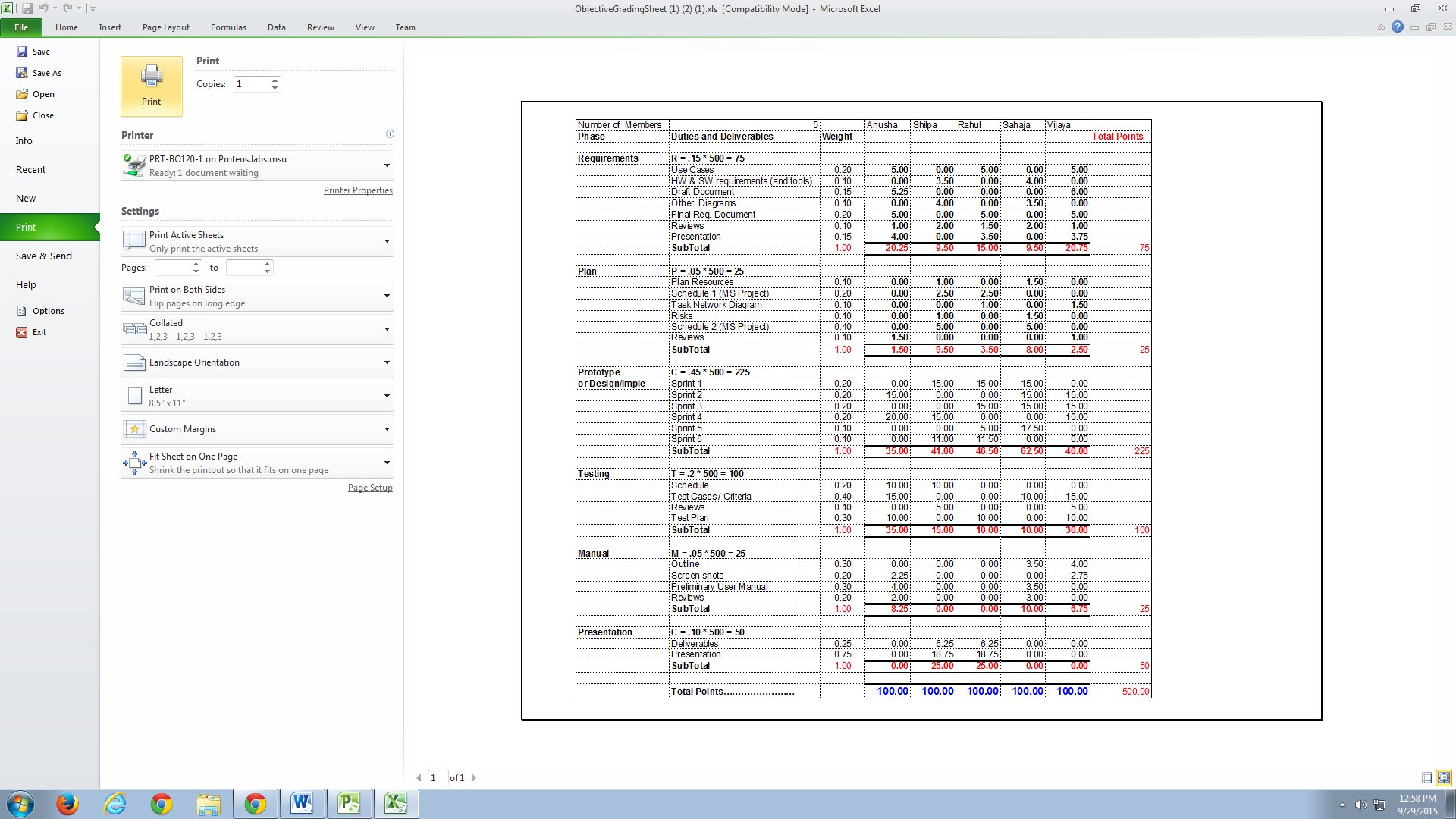


Figure 4. Objective Grading Sheet

**7. Tracking and Control Mechanics**

The project will be monitored during the development process to catch any error that may arise. The group will develop a plan to test the software at various stages of the development. The group will also review the requirements and specifications plan to ensure 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 were and if there were not errors. Such a plan, carefully executed, would ensure the system is deployed with optimal accuracy.

**8. References**

[1]. Wei, Dr.Bingyan, Automatic Greenhouse Monitor and Control System, MSU Wichita Falls, Fall2015.

[2]. Stringfellow, Dr.Catherine, Automatic Greenhouse Monitor and Control System, MSU Wichita Falls, Fall2015.

[3]. Sommerville I., Software Engineering 9th edition, Delhi, Pearson Education, Ltd, 2011.

[4]. Pressman Roger S., Software Engineering – A Practitioner’s Approach 7th edition, New York, McGraw-Hill, 2010.

[5]. Uyttewaal E., “Microsoft Project: Plan Better with Microsoft Project”, New York, TechNet Magazine, February 2012.

**9. Glossary**

**Humidity:** A quantity representing the amount of water vapor in the atmosphere or a gas.

**Soil acidity:** soil that has a pH (indication of the acidity) of less than 7.0 (neutral)

**Fertilization:** The action or process of fertilizing an egg, female animal, or plant, involving the fusion of male and female gametes to form a zygote.

**Growing plan:** A growing plan is responsible for keeping track of all interesting actions associated with growing a plant, correlated with the times at which those actions should take place. For example, on day 15 in the lifetime of a certain plant, our growing plan might be to maintain a temperature of 78°F for 16 hours, turn on the lights for 14 of these hours, and then drop the temperature to 65°F for the rest of the day. We might also want to add certain extra nutrients in the middle of the day, while still maintaining a slightly acidic pH.

**Simulation:** Simulation is the imitation of the operation of a real world process or system over time. The model represents the system itself, whereas the simulation represents the operation of the system over time.