Project Management Plan for Enhancing Itslearning
Assignment 2

Hossein Nourozi Shad
780401-T596

Srikar Reddy Nadikattu
930923-1554

Akshay Kumar Magapu
940829-4495

Sharen Polavarapu
940521-2573

Avinash Kiran Ivvala
940803-0196

I. INTRODUCTION

The technical system chosen for the project is the current Its Learning application - a learning management system (LMS) widely used by universities and colleges as an effective support tool for teaching and learning. The web based system was designed to have a common platform for students, teachers, and parents. The system consists of study materials, submission portals, grades, course updates, online messaging, e-portfolio, and communities. The student and teachers have their respective login credentials and these are used for signing in. The interfaces for students and teachers are different as the options like add assignments are available for teachers and submission of assignments is available for students.

The main objective of this project is to develop two new features. The first feature is facilitating students in arranging an appointment with the respective lecturers in order to clear doubts or for supervision. The request is confirmed after clearing an approval from the lecturer and both parties are sent a mail for confirmation. The second feature involves implementing Single Sign On (SSO) feature which only requires the user to sign in once for every session, without having to do so for every feature. This feature will be integrated for current and existing features, thus eliminating the need for multiple logins. Credentials for all services involved with university. The new features are developed using an agile software development methodology. Each feature is developed in one sprint and each sprint lasts 4 weeks. At the end of the first two sprints, we have the first release in the form of a potentially shippable product.

Scope and Objectives:

Scope can refer to two contexts in a software development project: products scope and project scope. Scope management involve "defining and controlling" what needs to be included and excluded from the project and analysing the product features. Scope of the project involves defining and specifying various aspects of the project based on the product scope [1].But in order to define the scope, the project team must be aware of the user

requirements and objectives to be met during and after the course of the project.

It is difficult for the project manager to plan and estimate project attributes, since a software product is intangible. Thus, requirements must be organised and mapped with the objectives. The objectives must be recorded in a detailed manner. Instead of having a generalised statement of the objectives, the project manager should explore the scope from various aspects. Scope management is a necessary predecessor to specification and documentation of features [2]. Some of the objectives and their scopes of the product are given below.

Scope and Objectives of the Release Product:

The release consists of two features developed over the course of two sprints. They have been further described in the sprint backlog section. Some of the objectives and their scope are:

Objectives (In addition to the existing features):

1. To build a fully functional and responsive lecturer consultation booking module for requesting an appointment with a lecturer. This involves two modules integrated into one feature: One where the vacant hours of the chosen lecturer are displayed and another where we can make a reservation for an appointment, pending approval from the teacher.

2. To implement Single Sign On (SSO) feature to create a single instance signing system for accessing all the features of the website. This feature is to be developed and integrated for the existing features as well as the current features.

Customer Description: The customers of Its Learning include a wide range of universities spread over a large geographical area, specifically in Western Europe. The universities however do not depend on our LMS for all course management tasks. Some of them, for instance BTH, have a student portal for performing other functions like examination registration.

User Demographic Scope: The most common users of this product include students and teachers. Other secondary

users are parents, IT support team at campuses and college administration. The users mostly fall under the age range of young to middle-aged people.

Technical Scope: Currently, ItsLearning has servers in different geographic locations to cater to the needs of a huge student-teacher base. Further, each major region has its regional office with dedicated teams for deployment and maintenance of the system.

Functional Scope: Currently, ItsLearning system involves a few features. However, this limits the choices that a student/teacher must have to deal with daily academic tasks. The scope of our project extends far beyond the existing scope as we seek to integrate 10 new features. This includes functions like an integrated system with examination signup system, course appraisal, campus event viewer, inbuilt access to electronic student databases, complaint management module and online leave/deferment system.

Scope of the Project:

In order to perform scope management, the project manager must be clear on the objectives. Modern day Agile software projects practitioners have identified seven major criteria for identifying and managing scope of the project, as concluded by a recent study [3]. Based on this we included the following scope factors:

Documentation and Specification of Objectives: The scope of the project can only be truly understood if the objectives are known. The documentation of objectives is done by converting the user stories into use cases and functions to develop UML diagrams. In this project, we will use these documents to create mapping models between use cases and project requirements.

Functionalities described by the customer:
This factor emphasizes on the extent of involvement of the customer in creation of backlogs and obtaining feedback for developed functionalities. In case of our project, we will follow an agile approach and have a high level of user involvement collecting feedback twice or thrice in between the sprints. The product owner will also be heavily involved in backlog development.

Frequency of Stand up and Review Meetings: These meetings are held to gather knowledge about the project progress and improve team communication. The stand up and review meetings will be arranged frequently by the scrum master, daily scrum meetings will be held discussing what has been done, what needs to be done and any barriers encountered while development.

Frequency of Releases: Having frequent releases helps in project planning. It reduces the "risk of errors and having poorly implemented features." For this project, we intend to have our first potentially shippable product at the end of the first two sprints. From then on, we will have a release after every one or two sprints [3].

Assigning Roles to Team Members:
Roles are assigned based on the skills and experience of the personnel. Despite having roles, each team member is involved in the planning and execution process. One of the team members is additionally assigned the role of Scrum master. In our case, despite assigning the roles, we have a self-managing and cohesive team working in close collaboration.

Flexibility of Multifunctional Teams: A multifunctional team involves team members who take up more than one distinct roles pertaining to different phases. In our project, each member is involved in multiple tasks and helps in problem solving and error rectification throughout the entire project.

The roles assigned have been described in the assumptions.

Extent of Pair Programming: Our software project will imbibe the practices of pair programming to a great extent. Pair programming facilitates knowledge sharing. This can help overcome any complexities and help in problem solving and decision making [3].

Need for implementing Learning Management Systems:
1. Using LMS compels students to submit assignments and access course material, also making it easier for the lecturers organise classes and assess individual performance of students.

2. Learning management systems are particularly useful for distance learners and off-campus study. Students can access course material uploaded by teachers, while teachers can have access to student assignments and records, on the go [4].

3. Though LMS cannot replicate the intricacies involved in face-to-face learning and is not as effective as classroom learning, it is a great supplement to conventional learning and is a great way to propagate course directed material to students [4].

4. Learning Management Systems like its learning are considered to be more formal and systematic in their approach to education, when compared to the next best alternative: social networking sites like Facebook. Furthermore, a study conducted in 2014, concluded that 70% students felt using an LMS had a positive impact on their course performance [5].

Motivation behind selecting this project:
We found that many essential features were missing from the system that would make academic learning and teaching much easier. Many university related tasks such as registration for a course/examination, viewing lecture schedule, booking a slot for appointments, viewing campus events and complaint appraisal were missing. Further, these features may or may not be present on student portals of the respective universities. If present, they were scattered in the form of small third-party developed modules and required the user to login separately for using each of

these features. The whole process was cumbersome and inefficient. Thus, we also felt the need to introduce Single Sign On and integrate it with all the old and new features. On the other hand, certain features currently present on ItsLearning, such as the "forum", can be considered as obsolete as they have fallen out of favour among the students. A study performed on students noted that students were most likely to use SNS (Social Networking Sites) to create forum pages, follow discussion threads, and post queries. SNS being feature-rich create a sense of community and foster collective knowledge sharing among students and teachers in a flexible way. On the other hand, the forum feature on LMS was seen as limiting and restraining with its naive features [5].

Assumptions:

A1: All databases may not have the same file format.

A2: All features are independent of other features, i.e. the development of a feature does not require one or more other features to be finished prior to its commencement.

A3: All members of the development team have considerable experience and skills for working on the roles assigned to them.

A4: Since, our team is an agile project team, each team member occupies multiple roles. In this case, the designer can perform requirements analysis, 1 developer can design system architecture, another developer can perform testing tasks, and the tester can also assist in development activities. Further, the analyst is given the designated role of the Scrum master.

A5: The start date of the project is the next day after submission date of the project plan - 2015-11-02.

A6: Budgeting and cost estimations are available for feature product development.

A7: Each week has 5 working days. Thus, the effective working time in each sprint is 20 days.

A8: The new features will be developed as modules so the Itslearning clients can choose whether they want these features or not.

A9: The new features are formed based on present BTH - Its Learning features and not considering another university so we assume these are new features.

A10: A day's work in the project is equivalent to 8 hours of work.

A11: After completion of Sprint 2, there will be a handoff and deployment of the latest release of the product.

II. STAKEHOLDERS

A stakeholder in a software project can be described as a person who has some stake or interest in the project who has a certain varying degree of involvement and influence on the project, that might be positive or negative. A stakeholder affects the project in some way and is also affected by it in the process. Stakeholder analysis is a vital component of requirements elicitation[6]. In order, to truly understand a product, we must be aware of all stakeholders involved, their importance and the influences that they might have over the project. In this assignment, we have analysed the stakeholder from two aspects: interest and influence.

We have chosen the 'stakeholder identification and analysis method for interorganisational environments' proposed by Ballejos and Montagna, as our preferred approach in this project [7]. Our project involves interaction between two organisations: product developers and universities, and the above method is well suited to our cause. This method involves identifying the stakeholder types and roles, selecting the stakeholders, assigning roles to the stakeholders and then analysing each stakeholder's influences. Each of the chosen stakeholders has been identified below, and their types, roles and influences have been analysed. Based on the interest (stake) and influence, we have four hierarchical levels of stakeholders:

S1: High interest and high influence.

S2: High Interest and low influence.

S3: Low Interest and High influence.

S4: Low Interest and Low Influence.

The stakeholders are divided into two primary categories: Internal and External.

1. External stakeholders:

- **Direct**

University Administration: The university administration is one of the key stakeholders. They include the deans of the universities and various departments, program directors, treasurers, student coordinators. These people are in direct communication with the project manager and development team. The representative of the university administration sits down with the team at sprint reviews and requirements elicitation phases and provides the user stories, and feedback about the functionalities from time to time. Thus, this group can be classified as S1. The reason being that they are the ones funding the project for developing their university credentials. Further, every user story is reviewed by them before development can commence.

Positive Influences:

The administration is the product owner. Thus, they are an integral part of the development meetings. They provide timely feedback over system functions and help in requirements specification. This boosts the morale of the team and acts as a driving force in everyday tasks.

Negative Influences:

They might be ambiguous in their expression of user stories. They might still be undecided on what functions need to be included. Further, their constant intervention in development activities and naive views about software development might lead to communication challenges.

University IT/Technical Support Team: This is a team located within the university premises that is responsible for integrating learning management system in university. They conduct the back-end tasks such as enrolling a student to the college database, adding/editing courses, scheduling lecture and examination timetables and so on. These stakeholders also interact with the software development team at different phases such as analysis, deployment and maintenance. They can be considered under state: S4. They have low interest and influence as they are not direct beneficiaries of the product and have little say on the features to be included.

Positive Influences:
They help in deployment and maintenance of the application at the target destinations. The technical support team facilitates a smooth transition between testing and deployment. They voice concerns about the product and report any problems encountered during daily operations.

Negative Influences:
Some of them might find it difficult to comprehend the functions due to lack of skill or cognitive ability, which can lead to bad reviews.

- **Indirect**

Students and Teachers (Lecturers): Students and teachers constitute the bulk of customers using this application. They are the ones that have driven the need for such a product. These are the people who provide the necessary functions to be included in the system. They can be classified as S1/S2, since they are the end users and provide their opinion in the analysis phase, but they do not exercise the final authority over chosen functionalities. Further, parents can also be considered as indirect stakeholders as they too have access to the system through student accounts. However, they fall under S2. Their needs include having error free system, having more new features for interaction, and ease of use.

Positive Influences: Requirements collected from these customers during analysis phase will lead to having a list of good new features as per their expectations. Further, they provide feedback regarding new releases and report any bugs in the system.

Negative Influences: Sometimes unnecessary requirements might be included as requested by them due to misconceptions about the system. Changes in requirements result in delayed development time.

Direct Competitors: Direct competitors have a great impact on the planning and development of a product. The competing product sets a benchmark standard for the development of our product. The better features from these competitors will affect the company because the customer may shift to better learning system so having up to date information about changes in education system may lead to better development. They can be classified as S4, since they have zero stakes in the project. Further, they explicitly influence our product through their product value and features.

Positive Influences: Healthy competition between companies has a positive effect and increases enthusiasm to develop better features in team. Features of other similar products act as a benchmark for standards to be maintained in our project.

Negative Influences: Sometimes competition might take an ugly turn leading to failed projects and financial losses. Company feature and success of another company feature and leads to losses for company.

2. Internal stakeholders:

Team Members: The software project team members are the analysts, designers, developers and testers. These stakeholders have a high interest in the project. These members define the success or failure of the project. All team members are involved in planning and implementing the entire project from start to finish. These will develop the required feature for the system. The needs for the team members are sufficient resources, challenging work and recognition.

Positive Influences: With sufficient resources the team can develop a quality product within time. The team interacts with the customers and the project manager and is involved in every decision made regarding the software project management**.**

Negative Influences: The skills and experience of the development team in solving problems, making decisions and managing crises influences the overall outcome of the project. Further, staff turnover may lead to delay in development and loss of intellectual property.

Project Manager: The project manager is one of the key stakeholders in the software project. The task is to ensure that the team implements the features requested by the product owner without exceeding the time and budget estimations. The project manager monitors whether there are sufficient resources and supports and motivates the team to ensure that there are no delays. He/she resolves conflicts, and looks at the problems and needs of the team. The project manager is regarded as S1. This is due to the fact that the manager has high stakes in the project and must complete the project successfully. Further, he/she also adapts and amends the project according to the situation and demands of the owner.

Positive Influences: By monitoring the status of the project they can help to meet the deadline by allocating required resources. The project manager provides technical and behavioural support to all team members and ensures coordination and communication between different roles and tasks. The manager is the single most influential internal stakeholder.

Negative Influences: The incompetence's of the project manager in performing tasks such as cost estimation and scheduling may spell disaster for the project. Further, the

manager might lose control over the software processes. All these have a negative impact on the project.

Although, there are many other stakeholders such as the platform provider, the marketing staff, the HR and legal department, the legislative bodies, customer relations manager and standardisation bodies, we have chosen the ones that are relevant and closely associated to the Agile software development practices of our project.

III. PRODUCT BACKLOG

In this section, the new features which are assumed for the selected system are discussed.
The list of new features formed by the team based on requirements and needs are discussed below:

1. Lecturer Timetable and Consultation Hours: Using this feature, students can view the lecture timetables and book a slot or appointment when there is availability. This has two modules: One is the timetable viewer, the other is the booking manager. Thus, this eliminates the need for mailing or manually requesting teachers to secure an appointment. A confirmation mail is sent to both parties (student and teacher) so that teacher is aware.
2. Single Sign-On: By this feature, the systems or services which are independent of Its Learning but are provided by university can be accessed through single sign-on option. This eliminates the need for signing in each time a service is required by student or teacher.
3. Online Project Assessment and Guidance: Using this feature, students and teachers can access the previous thesis submissions and other literary works published by university members. Filters are provided to search by author, department and topic. It can be integrated with the university database and is of great use for students.
4. Examination Scheduling System: Students can view the examination schedule prior to and after registration of course and plan ahead. Students who opt for a course are registered to the examination by default. The grading and evaluation is done by the teachers who have a modified interface.
5. Course Appraisal System: This feature enables the students to provide the feedback to the course, lecturer and other persons involved during the course. Each student obtains the final grade in a course only after submission of feedback, and then the feedback is aggregated for the administration.
6. Complaint management system: By this feature, students and teachers can lodge complaints and grievances if any problems are encountered. So the complaint is forwarded to respective department and based on the complaint necessary action is taken by the management.
7. Events Display Feature: This function displays all upcoming events to be held in the campus. Students, teachers and parents can convey their interest and discuss on the event forum.
8. Access to Student Database System: Students and teachers can access the electronic scientific databases through Its Learning instead of having to login to another third party webpage, also by integrating it with single sign-on feature there is no need of signing in required for each database.
9. Online Deferment / Leave System: By this feature, students and teachers can apply for a leave or freeze a term. So that there is no need of informing teachers or university administration.

IV. SPRINT BACKLOG

As there are two sprints in this project, we have prioritized the product backlog based on two criteria: planning poker - a popular Agile effort estimation technique and Tee shirt sizing - for estimating customer value. Planning poker is done by using a web resource and customer value is done by using tee shirt sizing method.

Planning poker is done as follows: each member participated in poker rounds and for each user story team members selected a card using a web tool: https://www.planningpoker.com/ . This was repeated until each member chose the same estimate after reaching a consensus through discussion. Based on this we got effort estimations for the features. Two features that were best suited to fit the entire length of the sprint were chosen. This was done to ensure that the there was no time and budget overhead.

Customer value is selected as we need to know how important a user story was, from the customer/owner's perspective. The motivation for selecting this criterion was that the customer satisfaction and involvement is key to the success of any software project. To evaluate the customer value, we have applied Tee- Shirt Sizing method. In this method, each member (including the product owner) picks a shirt size like S, M, L, XL, and XXL. Each of the sizes corresponds to a number on the scale from 1-10. The greater the size, higher is the value. The sizes chosen are averaged and the features which got the highest aggregated size vote are prioritized.
The two criteria are used to filter the product backlog for features to be included in the sprint backlog.

Thus, the sprint backlog is selected based on the above two criteria. The prioritization is done considering two criteria scores given by members. The features selected for Sprint backlog are lecturer consultation hours, and single sign-on. Each feature is developed for 1 sprint, which lasts for four weeks. After prioritizing the product backlog, we were left with the following features for the sprint backlog:
Feature-1: Lecturer Consultation Hours
Sprints utilized: Sprint 1
Effective duration of each phases: Analysis (3 activities) - 5 days, Design (4 activities) - 6days, Development (7 activities) - 10days, Testing (3 activities) - 2 days.
Duration: 20 days.
Feature-2: Single Sign On
Sprints utilized: Sprint 2.
Effective duration of each phases: Analysis (3 activities) - 3days, Design (5 activities)-5 days, Development (7 activities) - 9days, Testing (4 activities) - 3 days.
Actual duration: 20days.

Since, we are using an agile software development method - Scrum for software development, certain phases of development and certain tasks within each phase that have no dependencies overlap with each other. This is due to the

ID	Name	Actual Duration	Start	Finish	Predecessors	Resource Names
1	**High level project planning activities**	**8 days**	**Mon 19-10-15**	**Wed 28-10-15**		
2	Identifying Stakeholders	1 day	Mon 19-10-15	Mon 19-10-15		Brain stroming
3	Identifying Product Backlog	1 day	Tue 20-10-15	Tue 20-10-15	2	User stories,Brai
4	Scope Prioritizing Product Backlog	1 day	Wed 21-10-15	Wed 21-10-15	3	T-shirt sizing
5	Effort Estimation	1 day	Thu 22-10-15	Thu 22-10-15	4	Planning Pocker
6	Project Scheduling	1 day	Fri 23-10-15	Fri 23-10-15	4	Planning Pocker
7	Project Plan Documentation	2 days	Mon 26-10-15	Tue 27-10-15	6	Gantt Chart/Burr
8	Project Kickoff	1 day	Wed 28-10-15	Wed 28-10-15	7	
9	**Sprint #1 (Lecture Consultation Hours)**	**4 wks**	**Mon 02-11-15**	**Fri 27-11-15**	8	
10	**Analysis Phase**	**5 days**	**Mon 02-11-15**	**Fri 06-11-15**		**2 Analysts**
11	Get student/ parents requirements	2 days	Mon 02-11-15	Tue 03-11-15	9SS	Interviews,surve
12	Get lecturer requirements	2 days	Mon 02-11-15	Tue 03-11-15		Interviews
13	Feasibility Analysis (Development Platform/ Technology)	3 days	Wed 04-11-15	Fri 06-11-15		Internet,Researc
14	**Design Phase**	**6 days**	**Wed 04-11-15**	**Wed 11-11-15**		**2 Designers**
15	Object Oriented Programming analysis and design	3 days	Thu 05-11-15	Mon 09-11-15	11,12	UML and Design
16	Design timetable / Calender interface	2 days	Wed 04-11-15	Thu 05-11-15	11,12	Pencil Project
17	Design reservation user interface	1 day	Tue 10-11-15	Tue 10-11-15	11,12	Pencil Project
18	Database Design (DDL, ERD)	1 day	Wed 11-11-15	Wed 11-11-15	11,12	MS Visio / MS DI
19	**Development Phase**	**10 days**	**Thu 12-11-15**	**Wed 25-11-15**		**3 Developers**
20	Develop Calander/ Timetables	3 days	Thu 12-11-15	Mon 16-11-15	15,16	Eclipse IDE + JDK
21	Student side bookings and cancellation functions	3 days	Thu 12-11-15	Mon 16-11-15	15,17	Eclipse IDE + JDK
22	SMTP Integration	1 day	Tue 17-11-15	Tue 17-11-15	21	SMTP Server
23	Database modelling	2 days	Tue 17-11-15	Wed 18-11-15	18	MS SQL
24	Entity Beans Development	2 days	Thu 19-11-15	Fri 20-11-15	20,21,22	Eclipse IDE + JDK
25	Servlets Development	2 days	Mon 23-11-15	Tue 24-11-15	24	Eclipse IDE + JDK
26	Core Functionality Development	3 days	Mon 23-11-15	Wed 25-11-15	23,24	Eclipse IDE + JDK
27	**Testing Phase**	**2 days**	**Thu 26-11-15**	**Fri 27-11-15**		**2 Testers**
28	Unit Testing	1 day	Thu 26-11-15	Thu 26-11-15		Quickcheck
29	Integration Testing	1 day	Fri 27-11-15	Fri 27-11-15	23,24,26	Junit
30	Stress Testing	1 day	Fri 27-11-15	Fri 27-11-15	28	Fitnesse
31	**Sprint #2 (Single Sign On)**	**4 wks**	**Mon 30-11-15**	**Fri 25-12-15**		
32	**Analysis Phase**	**3 days**	**Mon 30-11-15**	**Wed 02-12-15**		**2 Analysts**
33	Define modules to be integrated with SSO	1 day	Mon 30-11-15	Mon 30-11-15	31SS	Brain stroming
34	Study the modules that need to be integrated with SSO/ rese	1 day	Tue 01-12-15	Tue 01-12-15	33	Research,Interne
35	Feasibility study (Technology and platform)	1 day	Wed 02-12-15	Wed 02-12-15		Internet
36	**Design phase**	**5 days**	**Thu 03-12-15**	**Wed 09-12-15**		**2 Designers**
37	Object Oriented analysis & design	2 days	Thu 03-12-15	Fri 04-12-15	33	UML and Design
38	Design Databases (DDL, ERD)	2 days	Thu 03-12-15	Fri 04-12-15	33	MS Visio / MS DI
39	Design signin forms, forgot passwords user interface	1 day	Mon 07-12-15	Mon 07-12-15	37	Pencil Project
40	Design Admin Interface	2 days	Tue 08-12-15	Wed 09-12-15	37	Pencil Project
41	Design for mobile Interfaces	2 days	Tue 08-12-15	Wed 09-12-15	37	Pencil Project
42	**Development Phase**	**9 days**	**Thu 10-12-15**	**Tue 22-12-15**		**3 Developers**
43	Develop databases	2 days	Thu 10-12-15	Fri 11-12-15	38	MS SQL
44	Develop user interfaces	2 days	Thu 10-12-15	Fri 11-12-15	39,40	Eclipse IDE + JDK
45	Develop web services	2 days	Mon 14-12-15	Tue 15-12-15	43	SMTP Server
46	Develop session manager	2 days	Mon 14-12-15	Tue 15-12-15	43	Eclipse IDE + JDK
47	Entity Beans and Servlet Development	3 days	Wed 16-12-15	Fri 18-12-15	37,43,45	Eclipse IDE + JDK
48	Develop forget password functionality	2 days	Mon 21-12-15	Tue 22-12-15	39	Eclipse IDE + JDK
49	Develop data encryption	2 days	Mon 21-12-15	Tue 22-12-15	47	Eclipse IDE + JDK
50	**Testing Phase**	**3 days**	**Wed 23-12-15**	**Fri 25-12-15**		**2 Testers**
51	Unit Testing	1 day	Wed 23-12-15	Wed 23-12-15	44,46,48,49	Quickcheck
52	Integration Testing	1 day	Wed 23-12-15	Wed 23-12-15	44,46,48,49	Junit
53	Stress Testing	1 day	Thu 24-12-15	Thu 24-12-15	44,46,48,49	Fitnesse
54	Security Testing	1 day	Fri 25-12-15	Fri 25-12-15	46,48	Grabber
55	Project Handoff	1 day	Mon 28-12-15	Mon 28-12-15	31	
56	Project Deployment	1 day	Tue 29-12-15	Tue 29-12-15	55	

Project: Agile project managem
Date: Sun 01-11-15

Task	Summary	Inactive Milestone	Duration-only
Split	Project Summary	Inactive Summary	Manual Summary Rollup
Milestone	Inactive Task	Manual Task	Manual Summary

Start-only	External Milestone	Manual Progress
Finish-only	Deadline	
External Tasks	Progress	

fact that the team is flexible and everyone has multiple roles. This is done in order to boost productivity and complete the project within stipulated time.

Note: The tasks of each sprint, their dependencies, their resources and the duration of each task have been depicted in detail in the Gantt chart and resources table
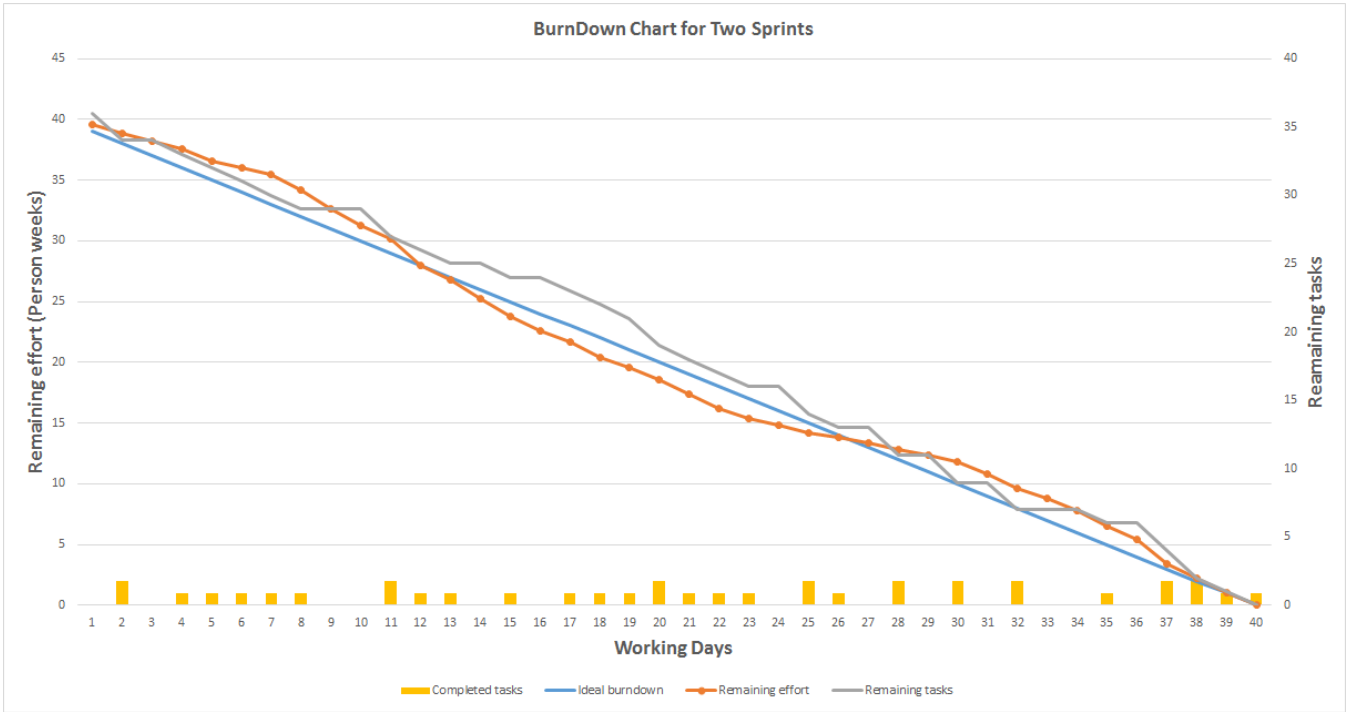
VI. BURN DOWN CHARTS
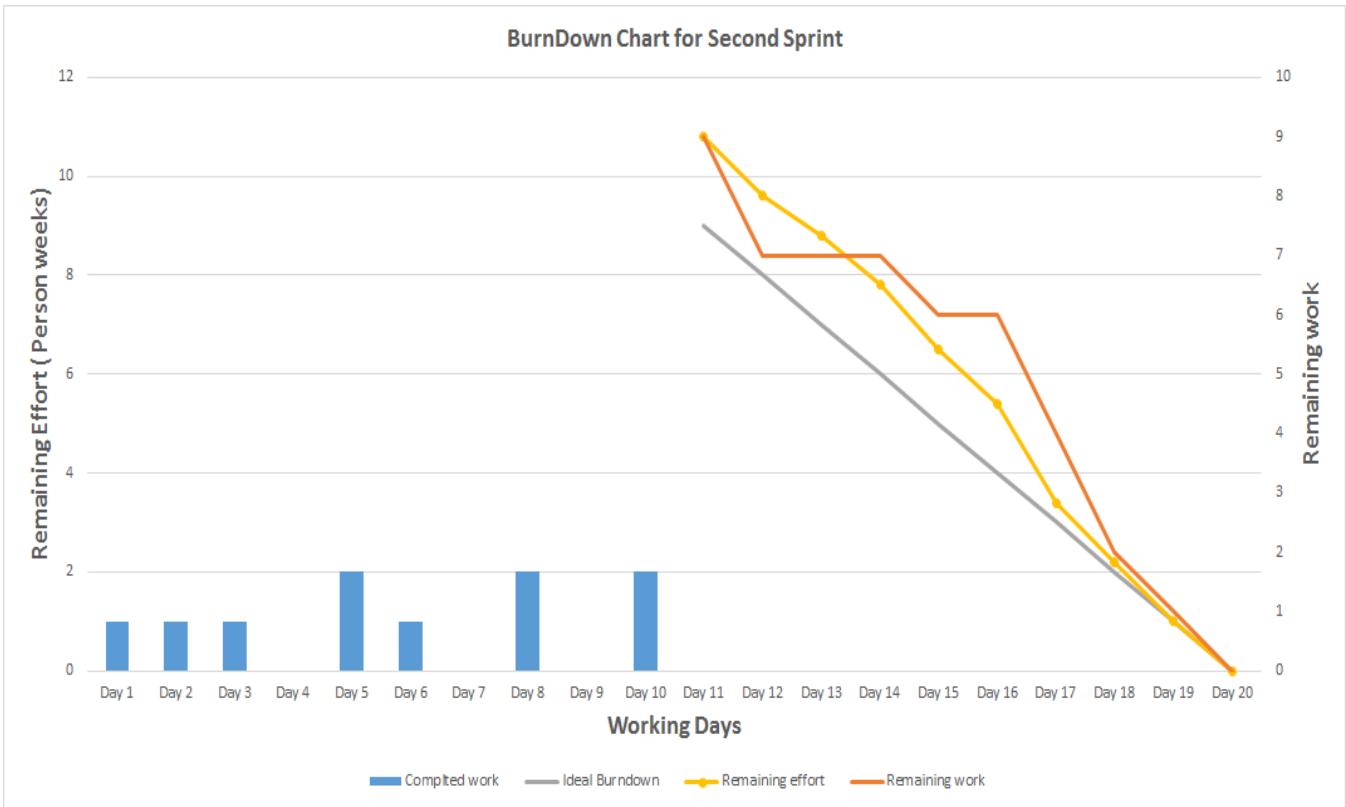

Figure 1. Burn down chart for two sprints

Figure 2. Burn down chart for second sprint with half completed sprint

VII. RESOURCES

The tables consist of resources used for **Sprint 1**.

Table 1. Resources assigned for activities in sprint 1.

Phase	Activity	Tools/Resources
Analysis Effective Human Resources:2	Get student/ parents requirements	Interviews, surveys
	Get lecturer requirements	Interviews
	Feasibility Analysis (Development Platform/ Technology)	Internet, Research,
Design Effective Human Resources:2	Object Oriented Programming analysis and design	UML and Design Patterns
	Design timetable / Calendar interface	Pencil Project
	Design reservation user interface	Pencil Project
	Database Design (DDL, ERD)	MS Visio / MS DB Diagram
Development Effective Human Resources:3	Develop Calendar/ Timetables	Eclipse IDE + JDK
	Student side bookings and cancellation functions	Eclipse IDE + JDK
	SMTP Integration	SMTP Server
	Database modelling	MS SQL
	Entity Beans Development	Eclipse IDE + JDK
	Servlets Development	Eclipse IDE + JDK
	Core Functionality Development	Eclipse IDE + JDK
Testing Effective Human Resources:2	Unit Testing	Quickcheck
	Integration Testing	Junit
	Stress Testing	Fitnesse

The following table consists of resources used in **Sprint 2**.

Table 2. Resources assigned for activities in sprint 1.

Phase	Activity	Tools/Resources
Analysis Effective Human Resources:2	Define modules to be integrated with SSO	Brainstorming
	Study the modules that need to be integrated with SSO/ research	Research, Internet
	Feasibility study (Technology and platform)	Internet
Design Effective Human Resources:2	Object Oriented analysis & design	UML and Design Patterns
	Design Databases (DDL, ERD)	MS Visio / MS DB Diagram
	Design signin forms, forgot passwords user interface	Pencil Project
	Design Admin Interface	Pencil Project
	Design for mobile Interfaces	Pencil Project
Development Effective Human Resources:3	Develop databases	MS SQL
	Develop user interfaces	Eclipse IDE + JDK
	Develop web services	SMTP Server
	Develop session manager	Eclipse IDE + JDK
	Entity Beans and Servlet Development	Eclipse IDE + JDK
	Develop forget password functionality	Eclipse IDE + JDK
	Develop data encryption	Eclipse IDE + JDK
Testing Effective Human Resources:2	Unit Testing	Quickcheck
	Integration Testing	Junit
	Stress Testing	Fitnesse
	Security Testing	Grabber

VII. QUALITY CRITERIA

The deliverables of this project are:

1. Lecturer Timetable and Consultation Hours

This feature is developed in sprint 1 and deployed after the end of sprint 2. In this we need to integrate this module to the existing system. So we need to consider the quality criteria for the developed feature to ensure that it satisfies the user requirements. The quality criteria for this product are described below.

2. Single Sign-On

In this feature there are several modules we need to be integrated with it to get access, so quality should be assessed whether every service is working for this module and if any errors exist between the services.

To test the project quality, we need to follow one of the software quality standards to ensure that the final software product meets all requirements and quality standards. The appropriate Product Quality standard for this project is **ISO/IEC 25010**. This standard provides following quality characteristics [10]:

Functional Suitability: This is a formal type of testing performed by testers. Functional suitability focuses on testing software against design document, Use cases and requirements document. We are using *Unit testing* technique to fulfil this type of quality standard.

Reliability: To ensure of software reliability we need to check the functioning of the software. We are going to use *Stress testing* and *Unit testing* techniques to ensure of software and product reliability.

Performance Efficiency: Performance testing is done to check non-functional requirements so we need to consider different factors such as platforms that the software will be setup and also hardware. Since there are a large range of hardware and platforms out there and we are not able to test the software on all different and combination of hardware and platforms, we are going to test it on one of the most common environment and provide a recommendation to use this software. In order to ensure of software performance efficiency, we are going to use *Stress testing* technique.

Usability: Usability talks about User interface readiness and how user friendly it is to ensure software usability, we are going to use *Interface Testing* under *Unit testing* techniques.

Maintainability: Maintainability describes that the software should be designed and programmed in a way that it is easy to comprehend and change. It includes how easy it is to understand, change and extend the software. During the project analysis and design, especially during database design and OOP analysis and design, the developers have to put the maximum effort to consider maintainability criteria's. Modularity, reusability, analysability, modifiability and testability are some of the maintainability criteria's which have to be considered.

Security: Objective of secure quality testing is to secure the software is to external or internal threats from humans and malicious programs. Security testing basically checks, how good is software's authorization mechanism, how strong is authentication, how software maintains confidentiality of the data, how does the software maintain integrity of the data, what is the availability of the software in the event of an attack on the software by hackers and malicious programs. In order to ensure that our software is secure, we are going to perform *security testing* using tools such as Grabber.

Compatibility: *Compatibility testing* is one of the test types performed by testing team. Compatibility testing checks if the software can be run on different hardware, operating system, bandwidth, databases, web servers, application servers, hardware peripherals, emulators, different configuration, processor, different browsers and different versions of the browsers etc.,

Compatibility testing can be done under unit testing and Integration testing.

Portability: Portability in software quality control talks about how much the software product is platform independent in case if the product has to move to other platforms in future. it express that how ties the product to one platform. This matter has been considered during project feasibility and analysis, so this software is going to be developed with Java technology and MySQL database to address the portability matter of software product quality control. Although the chosen development technologies and tools are platform independent but we are going to test our final product on different platforms such as different flavour of Linux and different version of Microsoft Windows Servers during the *Unit testing and Integration testing*.

VIII. RISK PLAN

Based on reviewing relevant literature [8] [9], we found the following risks that could be encountered during the course of the project. Their impact and mitigation strategies have also been discussed:

Risk	Risk severity	Risk category	Impact	Mitigation
Staff turnover	Negligible	Actors	Low	Loyalty Bonus, Increasing salary, Giving challenging work to the team members based on their talent, Plan for attrition on beforehand.
Insufficient skills for team members	Critical	Actors	Low	As per assumption A3, the selected team members are efficient in their roles.
				Scrum master takes care of each

Lack of teamwork	Marginal	Actors	High	individual to improve their readiness and willingness through motivation.
Customer interaction	Critical	Actors	Moderate	High level of user involvement.
Complexity of the tasks may be more than they are assumed to be	Critical	Tasks	Moderate	Using knowledge sharing strategies like pair programming.
Underestimated effort	Marginal	Tasks	Moderate	Already we have used planning poker in estimating effort required for our project by having consensus between team members. By regularly observing burn down charts, we could recalculate effort if required.
Indistinct task prioritization	Negligible	Tasks	Low	To mitigate this risk we have chosen two criteria to prioritize the product backlog.
Unclear project scope	Negligible	Tasks	Low	Scope is defined by following agile intended requirements management processes.
Unclear project planning	Critical	Structure	High	Planning is done by following agile project management process.
Duration of tasks not determined clearly	Marginal	Structure	Moderate	By regularly observing burn down charts, we could reschedule if required.
Lack of knowledge in agile practices	Negligible	Structure	Low	As per assumption A4, the team members are knowledgeable in agile practices.
Data loss	Negligible	Technology	Low	Data will be backup on to offline databases will be maintained frequently to avoid loss.

Availability of resources	Marginal	Technology	Moderate	All the resources required are planned in advance before the start of project.

REFERENCES

[1] Project Management Institute, *A Guide to the Project Management Body of Knowledge - PMBOK Guide*. 2013.

[2] F. McGovern, "Managing software projects with business-based requirements," *IT Prof.*, vol. 4, no. 5, pp. 18–23, 2002.

[3] M. D. A. Santos, P. D. S. Bermejo, and M. Silva, "Improving the Management of Cost and Scope in Software Projects Using Agile Practices," *Airccse.Org*, vol. 5, no. 1, pp. 47–64, 2013.

[4] A. Heirdsfield, S. Walker, M. Tambyah, and D. Beutel, "Blackboard as an Online Learning Environment: What do Teacher Education Students and Staff Think?," *Aust. J. Teach. Educ.*, vol. 36, no. 7, pp. 1–16, 2011.

[5] E. Dogoriti, J. Pange, and G. S. Anderson, "The use of social networking and learning management systems in English language teaching in higher education," *Campus-Wide Inf. Syst.*, vol. 31, no. 4, pp. 254–263, 2014.

[6] H. Sharp, A. Finkelstein, and G. Galal, "Stakeholder identification in the requirements engineering process," *Proceedings. Tenth Int. Work. Database Expert Syst. Appl. DEXA 99*, pp. 1–5, 1999.

[7] L. C. Ballejos and J. M. Montagna, "Method for stakeholder identification in interorganizational environments," *Requir. Eng.*, vol. 13, no. 4, pp. 281–297, 2008.

[8] T. Chow and D.-B. Cao, "A survey study of critical success factors in agile software projects," *J. Syst. Softw.*, vol. 81, no. 6, pp. 961–971, 2008.

[9] L. Siddique and B. A. Hussein, "Practical insight about risk management process in agile software projects in Norway," in *2014 IEEE International Technology Management Conference*, 2014, pp. 1–4.

[10] S. Wagner, *Software Product Quality Control*, Springer Berlin Heidelberg.2013.

APPENDIX

Note: Here, you can find evidence regarding the planning poker exercise involving all team members. The screenshots can be found in the link:

https://drive.google.com/folderview?id=0Bz_7Gx1hn6L9ODQtTlZKNnhxZkk&usp=sharing