Project Plan - Kanban Board

Pablo Arellano ⧫ Oliver Barry ⧫ Jennifer Ryan ⧫ Kevin Seldomridge
CMIS 495 - Current Trends and Projects in Computer Science
Professor David Castillo

TABLE OF CONTENTS

1. Description and objectives of the project

1.1 Description of the project.

Our project is to make an interactive Kanban Board program. In its most basic structure, a Kanban board will have three categories: To Do, Doing, and Done. In more complex implementations, a board may have many more categories with each one reflecting a different nuanced step along the work-flow process. The initial development of our program will consist of the basic category structure along with a user-friendly GUI for adding, moving, and editing tasks. As the project timeline is relatively short, we will keep the program design as simple as we can while still maintaining the functionality needed for a Kanban Board structure. Later, with time permitting, we may add more attributes to our program that would allow for greater user-customization (such as choosing the number of categories, scheduling reminders for tasks, and

other features that would increase the usefulness of the program to a wider customer base). Additionally, later versions of the program could include higher quality graphics rendering so as to improve the visual experience of the user.

1.2 What problem is the project attempting to solve?

Well structured work-flow methodologies are crucial for any successful project. This is especially true in Software Development, where team members need to have a clear understanding of what is required of them, and when it is needed by. A simple hold-up in the early stages of the development process can easily cause ripples that lead to missed deadlines, unhappy customers, or even total project failure. Having a sound methodology, and (just as importantly) a way to easily and successfully implement it, is critical to the success of any project. Our team project is an attempt to design an easy to use, intuitive resource that individuals and companies alike can use to ensure that their projects are following a sound management methodology, and that it is being correctly and successfully implemented.

2. Project Management

2.1 What project management methodology will you be using to manage your project (e.g.Scrum Agile, Kanban, Waterfall, etc.)

For this project we will use the Kanban methodology. This methodology fits well with the small number of team members and programmers we have. Using the Kanban methodology we will produce several incremental versions of our product until we see that no more changes need to be made for final delivery. Our team members will pull work from a queue of work that must be done that week and will be programmed to fit with our architecture. In this way each week we will be able to have a final product with the ability to improve on it for the next week by adding features.

2.2 Will your requirements take the form of user stories, scenarios, or functional requirements?

In order to complete this project, our requirements will have to be both functional and scenario requirements. We must create not only a functional program that fulfills the needs of the requirements set forth for the class, but also that fulfills the requirements of our users. Scenario requirements will entail the successful testing of theoretical situations that an average user would present.

2.3 What tools will you use to document and track weekly goals and deliverables (Jira, Confluence, Excel, etc.)?

We will be using Microsoft Project to document, track weekly goals and deliverables. We will also be using Google Drive and Google Docs in order to collaborate as a group. The drive has sections for each deliverable of the project and are sorted by the due dates as required by the syllabus.

2.4 What tools will you use to communicate with your team (LEO, Google, etc.)?

In order to communicate with our team we will primarily be communicating through email. Our team is separated by multiple time zones and virtual communication is essential to our success. We will also use LEO if needed to communicate.

2.5 What tools will you use to capture Analysis and Design artifacts (e.g. UML tools, Visio, etc.)?

In order to capture Analysis and Design artifacts we will be utilizing both Visio and draw.io. We will incorporate this into our submitted documents and use them to also track progress throughout the project.

2.6 How often will you communicate with your team?

Regular communication every week will be paramount to this project. As previously stated, our team is dispersed over great distance and in order to successfully collaborate a group project with all members involved, all members will be expected to communicate multiple times weekly.

3. Team Members

3.1 Who comprises your team?

Our team is comprised of four University of Maryland University College students currently enrolled in the CMIS 495 class. The team members are Pablo Arellano, Oliver Barry, Jennifer Ryan, and team lead Kevin Seldomridge. Each member has differentiating backgrounds and will therefore contribute in unique ways.

3.2 Name each person on your team and identify the primary role they will play and why?

Pablo Arellano works as an IT specialist with a great knowledge base of troubleshooting, diagrams, and documentation. His work history will enable Pablo to be a fantastic asset, especially in our testing phase.

Oliver Barry also works in IT and has a strong programming base in the languages of C++, Perl, and Java. His experience in repairing and troubleshooting software will be beneficial as our project develops and is fine tuned to ideal specifications.

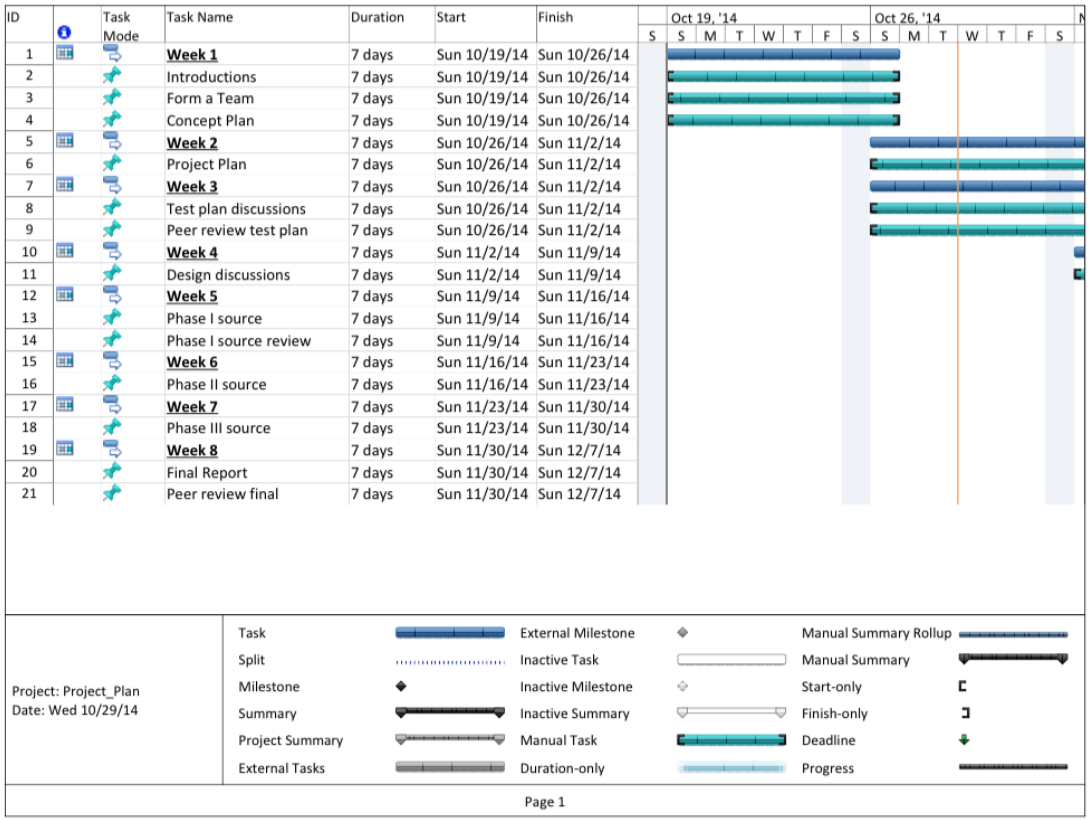
Jennifer Ryan works as an Operations Specialist and has extensive experience in documentation and diagramming as she continues to develop her programming skills. Her role in the team will be aiding in the organization of the project as well as the presentation/documentation of the process.

Kevin Seldomridge works as a software engineer and has the most experience of our team in software development. He has a high level of understanding concerning project management, design patterns, and will be serving as our team lead. Kevin originated the idea of a kanban board application to help users organize their tasks and to increase their efficiency.

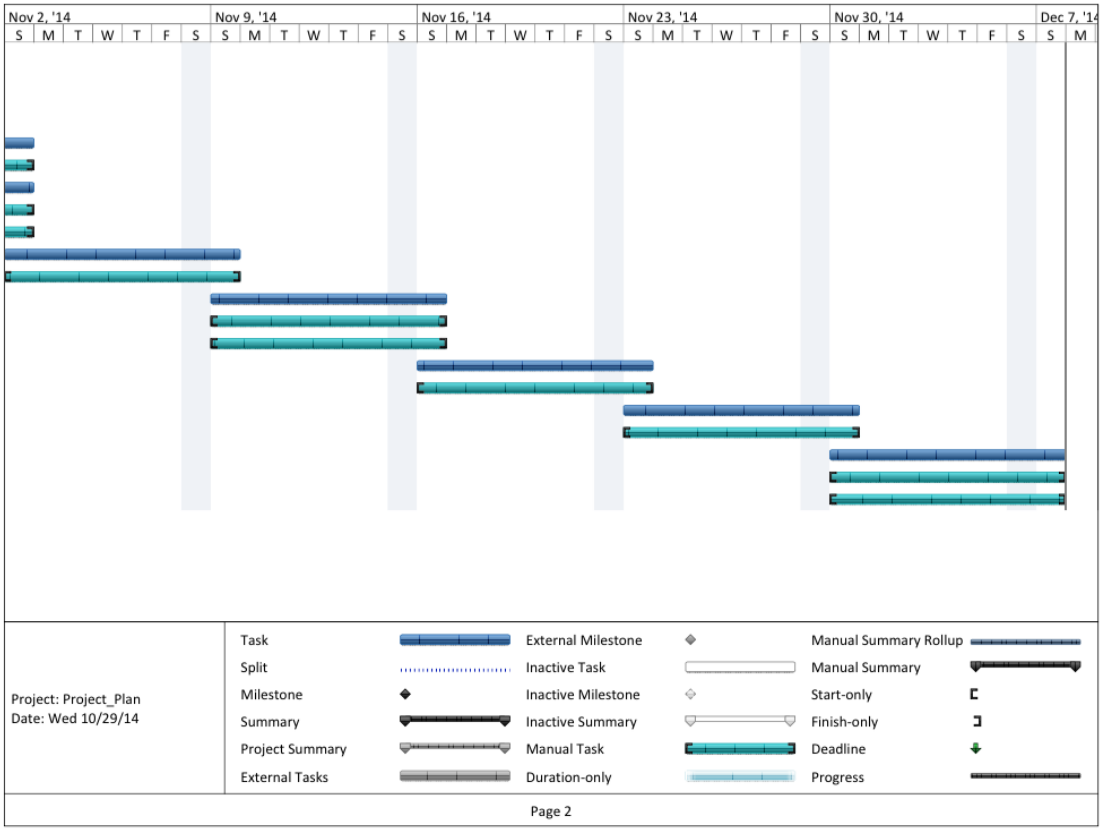
4. Project Timeline

Our project timeline was created in accordance to the syllabus presented for CMIS 495 in the Fall 2014 Syllabus. The Project plan will be created and submitted November 2 2014, the

test plan on November 9 2014, project design on November 16, 2014, peer review testing throughout the project, and multiple phase sources. We have utilized Microsoft Project to create the following outline based on said syllabus and pictured below.



ID	Task Mode	Task Name	Duration	Start	Finish
1		**Week 1**	7 days	Sun 10/19/14	Sun 10/26/14
2		Introductions	7 days	Sun 10/19/14	Sun 10/26/14
3		Form a Team	7 days	Sun 10/19/14	Sun 10/26/14
4		Concept Plan	7 days	Sun 10/19/14	Sun 10/26/14
5		**Week 2**	7 days	Sun 10/26/14	Sun 11/2/14
6		Project Plan	7 days	Sun 10/26/14	Sun 11/2/14
7		**Week 3**	7 days	Sun 10/26/14	Sun 11/2/14
8		Test plan discussions	7 days	Sun 10/26/14	Sun 11/2/14
9		Peer review test plan	7 days	Sun 10/26/14	Sun 11/2/14
10		**Week 4**	7 days	Sun 11/2/14	Sun 11/9/14
11		Design discussions	7 days	Sun 11/2/14	Sun 11/9/14
12		**Week 5**	7 days	Sun 11/9/14	Sun 11/16/14
13		Phase I source	7 days	Sun 11/9/14	Sun 11/16/14
14		Phase I source review	7 days	Sun 11/9/14	Sun 11/16/14
15		**Week 6**	7 days	Sun 11/16/14	Sun 11/23/14
16		Phase II source	7 days	Sun 11/16/14	Sun 11/23/14
17		**Week 7**	7 days	Sun 11/23/14	Sun 11/30/14
18		Phase III source	7 days	Sun 11/23/14	Sun 11/30/14
19		**Week 8**	7 days	Sun 11/30/14	Sun 12/7/14
20		Final Report	7 days	Sun 11/30/14	Sun 12/7/14
21		Peer review final	7 days	Sun 11/30/14	Sun 12/7/14



5. Hardware and Software tools required.

5.1 What hardware is required to develop your project?

The hardware needed to develop the Kanban board application is dependent upon the development task that is being performed. To program the code for Kanban board, a standard computer (PC or MAC) can be used as long as it has the capability of running a Java Integrated Development Environment (IDE).

5.2 What hardware is required to run your project?

To run the Kanban board application, a computer with the Java Virtual Machine (JVM) installed is required. This computer must have at least 128 MB of RAM and 124 MB of disk space to support the installation and running of the Java JVM.

5.3 What hardware is required to test your project?

To test the Kanban board application, a computer with the Java Virtual Machine (JVM) is required. This computer must also have a Java debugger installed such as those that come with most Java IDEs.

5.4 Name the development tools you intend to use (e.g. Java JDK, NetBeans or Eclipse, MySql, Swing for UI, etc.)

Project Plan - Kanban Board

The development tools being used for the development of Kanban board application is Eclipse and NetBeans with the Java JDK installed. The UI will be developed with Java Swing components.