AWARE_ness;

Not everything is as it seems...

Project Plan

Johns Hopkins University
Foundations Of
Software Engineering
(EN.605.401.81.SU13)

Version 1.0

Date: 06/22/2013

PROJECT PLAN VERSION CONTROL

Version	Date	Author	Message
1.00	06/07/2013	Shan Sabri	Document and template initialization
1.01	06/12/2013	Shan Sabri	Creation of Risk Assessment Plan v.1
1.02	06/13/2013	Hartanto Thio	Creation of Quality Plan v.1
1.03	06/17/2013	Shan Sabri	Creation of Work-Breakdown Structure/Schedule v.1
1.04	06/21/2013	Davis Gigogne	Testing and Configuration implementation add-on
1.05	06/22/2013	Shan Sabri	Project Timeline and Introduction add-on

INTRODUCTION

This document's purpose is to server as the initial project plan for Team Aware_ness' implementation of Project Clue-less. The project plan will address project milestones, scheduling, architecture, risk assessment quality assurance, and configuration management.

TABLE OF CONTENTS

1	Version Control	1
2	Table of Contents	2
3	Work-breakdown Structure	3 – 4
4	Risk Assessment Plan	5 – 6
5	Quality Plan	7 – 8
6	Document Acceptance	9

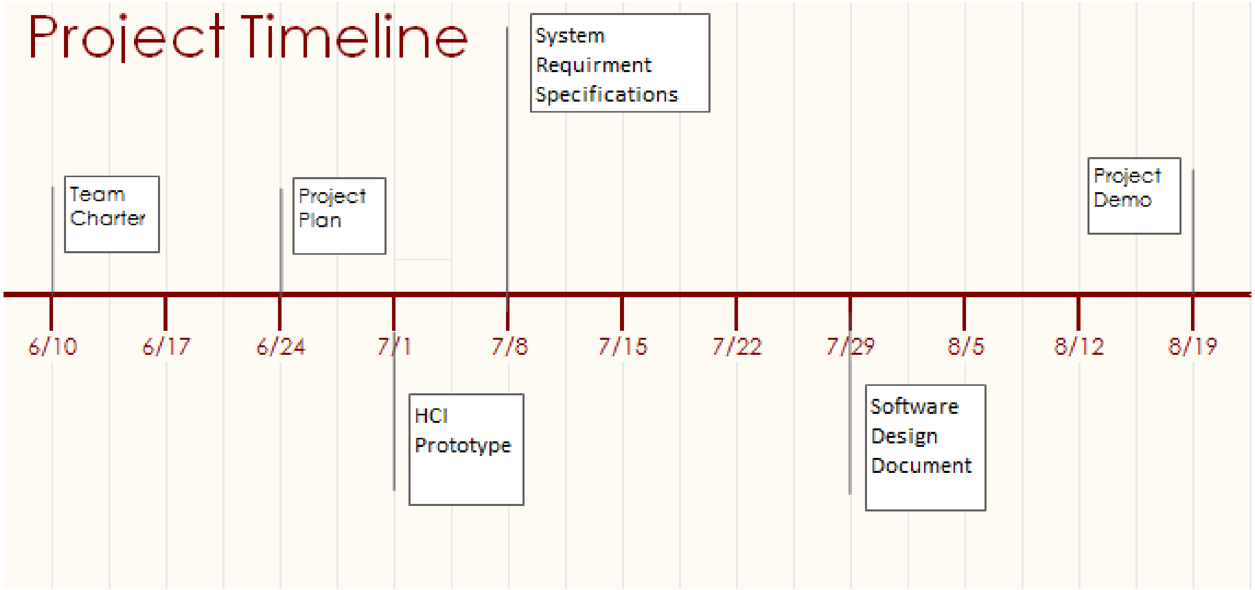
WORK-BREAKDOWN STRUCTURE/SCHEDULE

*Subject to change accordingly to accommodate for project risks and deadlines

ID	WBS	Deliverable/Milestone	Estimated Completion	Actual Completion
1	1	**Team Charter**	06/10/2013	06/06/2013
2	1.2	Manage Risk Assessment Plan	06/13/2013	06/12/2013
3	1.2.1	Develop Risk Assessment Plan		
4	1.2.1.1	Address Risk Identification		
5	1.2.1.2	Address Risk Analysis		
6	1.2.1.3	Address Risk Response		
7	1.2.1.4	Address Risk Control		
8	1.3	Manage Quality Assurance Plan	06/19/2013	06/21/2013
9	1.3.1	Develop Quality Assurance Plan		
10	1.3.1.1	Address Quality Assurance		
11	1.3.1.2	Address Testing		
12	1.3.1.3	Address Configuration Management		
13	1.4	Manage Work-breakdown Structure	06/20/2013	06/17/2013
14	1.4.1	Develop Work-breakdown Structure		
15	1.4.1.1	Address Task list		
16	1.4.1.2	Address Estimates		
17	2	**Project Plan**	06/24/2013	06/22/2013
18	2.1	Manage Human-Computer Interface prototype	06/27/2013	
19	2.1.1	Develop HCI prototype		
20	2.1.1.1	Address Requirements		
21	2.1.1.2	Address Design		
22	2.2	Manage Paper prototype	06/29/2013	
23	2.1.1	Develop PowerPoint slides		
24	3	**HCI Prototype**	07/01/2013	
25	3.1	Manage Software architecture	07/05/2013	
26	3.1.1	Develop Software Architecture		
27	3.1.1.1	Address Subsystems		
28	3.1.1.1.1	Describe Information domain		
29	3.1.1.1.2	Describe Functional domain		
30	3.1.1.1.3	Describe Interface		
31	3.1.1.1.4	Describe Implementation		
32	4	**Software Requirement Specifications (SRS)**	07/08/2013	
33	4.1	Validate Software architecture		
34	4.1.1	Develop Stubs and Drivers for testing		
35	4.1.2	Develop game rules		
36	4.1.4	Develop game objects and entities		
37	4.1.5	Develop system/subsystem interface		
38	4.1.6	Develop subsystem/subsystem interface		
39	5	**Skeletal System (internal)**	07/12/2013	
40	5.1	Prove Software architecture		
41	5.1.1	Develop functional user interface		
42	5.1.2	Develop support for 2 players and a winner		
43	5.2	Revise Project Plan		
44	5.3	Revise Software Requirements Specifications (SRS)		

Continued on next page.

45	6	**Minimal System (internal)**	07/19/2013
46	6.1	Manage Class Diagram	07/22/2013
47	6.1.1	Develop Class Diagram	
48	6.1.1.1	Address Classes	
49	6.1.1.1.1	Describe Associations	
50	6.1.1.1.2	Describe Aggregations	
51	6.1.1.1.3	Describe Specializations	
52	6.1.1.1.4	Describe Attributes	
53	6.1.1.2	Address Object Interactions	
54	7	**Software Design**	07/29/2013
55	7.1	Manage Working Software	
56	7.1.1	Develop graphical user interface	
57	7.1.2	Prove two player usage	
58	7.2	Demonstrate Software	
59	8	**Target System**	08/06/2013
60	8.1	Anticipate Future Add-ons	
61	8.1.1	Develop Multiplayer Usage (WAN/LAN)	
62	8.1.1.1	Address ability to randomly match users	
63	8.1.1.2	Address ability to invite new users	
64	8.1.1.3	Address ability to share user data	
65	8.1.1.4	Address ability to implement 'house rules'	
66	8.1.1.5	Address ability to implement global scoring	
67	8.1.1.6	Address ability to support different platforms	
68	9	**Dream System**	
69	9.1	Final Software Testing	08/12/2013
70	9.2	Final software Validation	08/15/2013
71	10	**Project Demo**	08/19/2013



Risk Assessment Plan

1 Purpose

The purpose of a Risk Assessment Plan is to aid in identifying events that may pose as a risk to a project. A risk can properly be defined as the effect, positive or negative, of an action/event that may conflict with obtaining a goal. Team AWARE_ness' risk assessment plan is focused on identifying and suppressing vulnerabilities within Project Clue-less.

This plan will specifically address:

- Risk Identification
- Risk Analysis
- Risk Response
- Risk Control.

2 Risk Identification

Risk Identification is the determination and documentation of which risks may affect a project. The goal of risk identification is to ensure that every project member is aware of what constitutes a risk and its potential impact to the project.

The project manager will be required to identify and document any known risks within the project's Risk Log (shown below). Though it is the project manager's responsibility to update this log, team members must assist by identifying new risk factors and communicate with the group as a whole.

3 Risk Analysis

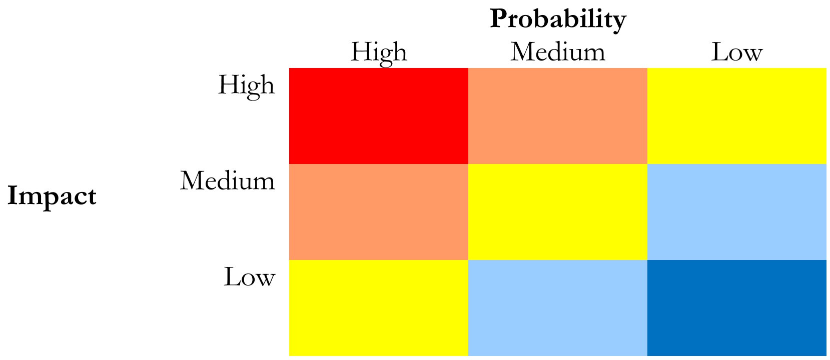
Risk Analysis is the process of determining the impact a risk can potentially have on the project. Risks will be prioritized according to their level of significance.

Risk Levels:

- High → $p(risk) > 75\%$
- Medium → $25\% \leq p(risk) \leq 75\%$
- Low → $p(risk) < 25\%$

Risk Impact:

- High → risk poses a heavy threat to the project
- Medium → risk poses a medium threat to the project
- Low → risk poses a light threat to the project



4 Risk Response

Each identified risk must be dealt with accordingly. It is the responsibility of the project team as a whole to determine the response to a risk, given its potential impact on the project.

Responses:

- Avoid – to eliminate and/or not become involved with
- Mitigate – to reduce severity of loss
- Accept – to accept the loss and adapt the project
- Postpone – to delay the project until further resources are available

5 Risk Control

In order to reduce the event of a risk occurring, a log will be generated to monitor and control the risks which have already been identified. As a team, the Risk Log will be analyzed at the conclusion of each milestone and modified as needed. Though, risk prioritization and consequent planning will be based on the risk perception at the time which risk analysis is performed.

A template Risk Log is shown below.

ID	Risk	Category	Impact	Probability	Response
0	Life Events	All	Medium	Low	Accept
1	Scheduling	Planning	High	Medium	Avoid
2	Initial Design	Planning	Medium	High	Accept
3	HCL Prototype	Coding	Medium	Medium	Avoid
4	Data Breach	Coding	High	Low	Mitigate

QUALITY PLAN

1 Purpose

The purpose of a Quality Plan is to ensure quality standards are met throughout a project and determines how they can be satisfied. Team AWARE_ness' quality plan is focused on the documentation and control of Project Clue-less.

This plan will specifically address:

- Quality Assurance
- Testing
- Configuration Management

2 Quality Assurance

Quality Assurance is the process in which we ensure the end product is developed according to the set of requirements and is working appropriately. In addition, this team is also responsible in identifying whether the quality (features, user interface) is good.

Accountability evaluations for quality control will be assessed as a team on a bi-weekly basis.

3 Testing

Testing is the process of ensuring that the features implemented are working as it should be. The features implemented are based on the things defined in the requirements. This will look at the aspect of performance, bugs, etc.

Team Aware_ness will mainly be implementing unit testing on a frequent basis throughout project Clue-less, though functional testing may be applied where necessary. Unit tests will be run in each case that a new function is created in order to eliminate the risk of a "cascade effect". Each unit test will consist of a report detailing the test name, pass/fail conditions, and corresponding results.

It is the Software Tester's job to evaluate for functional deficiencies and clear the function for merging. Though, each team member is expected to run every instance of a test to ensure that its functionality is proper prior to merging with the master branch. Should merging issues arise, the test creator will be responsible for correcting the issues.

Additionally, usability testing will be applied once the project is in working condition. Users will be invited to work through the project and give feedback. The user's behavior will also be monitored and this feedback will be analyzed to improve the program.

All tests are expected to pass by the deadline given in the WBS/schedule.

4 Configuration Management

Configuration Management is the process which establishes consistency of a project's performance and functionality. Team Aware_ness will be employing GitHub via Git version control and source code management (SCM) systems to control the distribution and development on code.

Quality assurance for configuration management will ensure that code is properly pushed via Git upon completion.

Client: Modern Browsers (e.g., Mozilla Firefox v.5+, Google Chrome build 27+, and Internet Explorer v.10+). Team Aware_ness will also implement basic functionality that allows older browsers to run the program. Though, older browsers may not have the same functionalities (or features) as the modern browsers.

Server: Implementation is expected to be based in Apache/Tomcat and Java. Though, a MySQL database may be employed if needed.

Bug Tracking: Should software bugs arise throughout project development, the team will utilize Github's Issues feature to document the bug. Documentation will explain how to reproduce the issues and any attempts at a solution. The commit comment will also document the solution that closes the bug.

Ref: [https://github.com/features/projects]

Document Acceptance

Member	Signature
Shan Sabri	
Hartanto Thio	
Davis Gigogne	