DnDCity

Lightweight Project Plan

Project Overview

This project deals with the popular tabletop RPG Dungeons and Dragons, or DnD. Anyone who has ever played DnD knows how frustrating it can get to manage your characters, campaigns, even just get together and just play. DnDCity aims to fix that. DnDCity is a website that uses multiple pages and forms where people can come and organize DnD campaigns, create and manage characters, items, or even campaigns.

The users of the website can be divided into three distinct categories: the Campaign Creator, the Game Master (or DM as he is sometimes referred to), and the Player. The Campaign Creator is the person who initially organizes the group of players to form campaign parties. The Game Master is the person who directs the game play, and determines the outcomes of players actions. finally, the Player is the person who determines the actions of one or more characters in the game itself.

Team Organization

Anarchic Team structure based on the fact that we trust each other.

Software Development Process

The development will precede using very simple process model that uses elements of Boehm's spiral process [Boehm-1988], iterative process models, and agile software development. The following table shows the entrance and exit criteria for these phases.

Phase	Iteration	Tentative Exit Criteria
1.	Phase 1 - Project Planning	Project selected, plan completed, team organized, configuration management in place
2.	Phase 2 - Requirement Capture and Analysis	A common understanding of requirements and system analysis captured in meaning documentation A prototype complete that helped improve understanding of requirements
3	Phase 3 - Architectural, UI, and DB Design	A system design that satisfies the requirements A prototype that explores a design alternative
4	Phase 4 - Detailed Design, Implementation, and	Initial implementation (60% - 80% of the functionality) with some unit test cases)

	Unit Testing	
5	Phase 5 - Implementation and Testing	Final implementation and Testing

For analysis and high-level design activities, we will use a conceptual-model language, called *Unified Modeling Language* (UML). The UML is expressive and semi-formal. Its expressiveness allows it to describe a wide range of concepts in both the problem and solution domains. Its formalism is rooted in a meta-model that describes its syntax and some of its semantics.

Communication policies and procedures

Email, github, meeting Mon, Wed, Fri between 11:30 and 1:00, as well as talking in/after class.

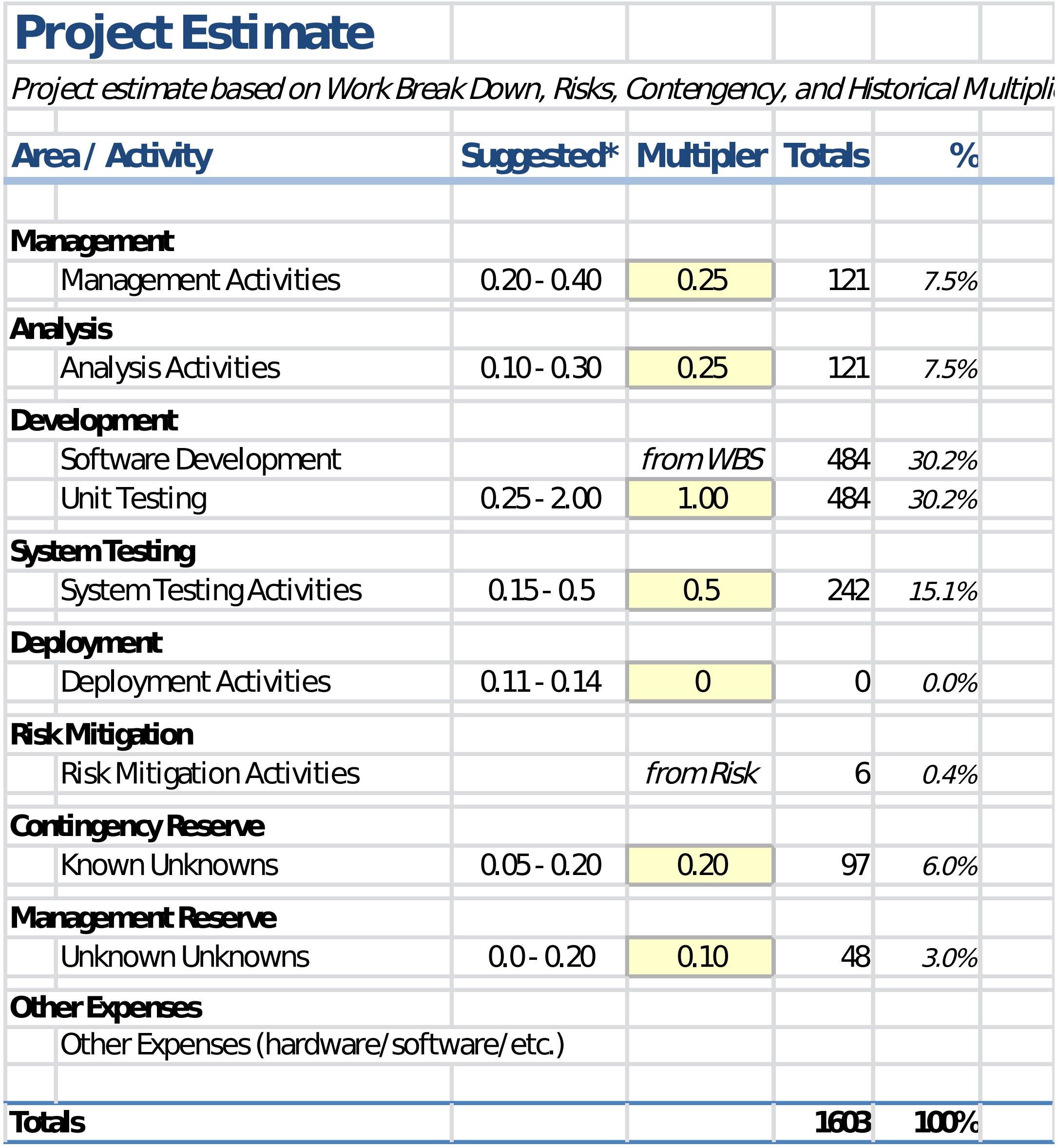
Initial work breakdown schedule

Category / Task	Estimate	
Web Framework		113.8333
	User Interface (HTML Pages)	15
	enable email verification	1.5
	enable email invites	1.5
	Integration	95.83333
Features Implementation		370
	User	50
	Character	
	Game Master	50
	Campaign Owner	
	Integration	270
Project 95% Confidence		483.8333

Risk Analysis

Identify your risks by area and then estimate their probability and impact

Risk / Mitigation Activity		Probability	Impact	Estimate (additional hours if risks occurs)
R1	**Personal issues**	**Medium/High**	**Medium/High**	**2**
1.1	Someone drops the class/implament fewer features	Low	Low/Medium	0
1.2	Illness/implament less features	Low	Low/Medium	2
R2	**Technical issues**	**Medium/High**	**Medium/High**	**2**
2.1	Web hosting server crashes/deploy to a different service	Low	Low	1
2.2	Data loss/ redundancy	Low	Low	1
R3	**Bad Time Estimates**	**Medium/High**	**Medium/High**	**2**
3.1	unable to get work done on time/ implament less features	High	Low	2
Total				**6**



Project Estimate

Project estimate based on Work Break Down, Risks, Contengency, and Historical Multipli

Area / Activity	Suggested*	Multipler	Totals	%
Management				
Management Activities	0.20 - 0.40	0.25	121	*7.5%*
Analysis				
Analysis Activities	0.10 - 0.30	0.25	121	*7.5%*
Development				
Software Development		*from WBS*	484	*30.2%*
Unit Testing	0.25 - 2.00	1.00	484	*30.2%*
System Testing				
System Testing Activities	0.15 - 0.5	0.5	242	*15.1%*
Deployment				
Deployment Activities	0.11 - 0.14	0	0	*0.0%*
Risk Mitigation				
Risk Mitigation Activities		*from Risk*	6	*0.4%*
Contingency Reserve				
Known Unknowns	0.05 - 0.20	0.20	97	*6.0%*
Management Reserve				
Unknown Unknowns	0.0 - 0.20	0.10	48	*3.0%*
Other Expenses				
Other Expenses (hardware/software/etc.)				
Totals			**1603**	**100%**

Hardware and Software resources

We will be deploying our application to the Heroku web deployment service.
Heroku provides the Web Server infrastructure, including a Postgresql Database.
Our web framework will be Ruby on Rails, using Haml and Sass for our html and css template shorthand, RSpec and Cucumber with Capybara for our unit and acceptance testing.
Javascript with JQuery may/will be used to enhance the user experience in the browser.

Configuration Management

We will be using Github for our source code repository, Task/Issue management, and documentation wiki.

Change Log

Date	Description of Change	Review by	Review on
Jan 17	Initial Draft	Jan 17	Jan 17