Software Project Management Plan
for
Snake Project

31.05.2017
Version 1.0

Prepared by:
Alexander Orel

Table of Contents

Revision History

Version	Date	Name	Description
1.0	31.05.2017	Alexander Orel	Initial document.

1. Overview

1.1 Purpose and Scope

The purpose of Snake Project is to increase an expertise level in game development.

The Snake application will provide users the ability to play in easy and fun game. Besides, users will have an opportunity to see a rating board with nicknames and best scores.

The application will be created for use on a PC with Windows and Linux platforms. The user interface will be intuitive and simple to navigate, with 95% of new users being able to use the application without referencing the user manual. A user guide and documentation will be provided.

1.2 Goals and Objectives

The overall objective of this project is to create a Snake game application for educational purpose.
Project Goals:

- Create an application that functions as expected and looks great.
- Learn about game development and creation of cross-platform application.

Project Objectives:

- Create game application that functions in a simple and intuitive manner.
- Familiarize with full process of software development.
- Learn new techniques, libraries and technologies.

1.3 Project Deliverables

Date	**Deliverable**
29.05.2017	Requirements Specification

1.4 Assumptions and Constraints

1.4.1 Assumptions

- Necessary third-party libraries support required platforms.

1.4.2 Constraints

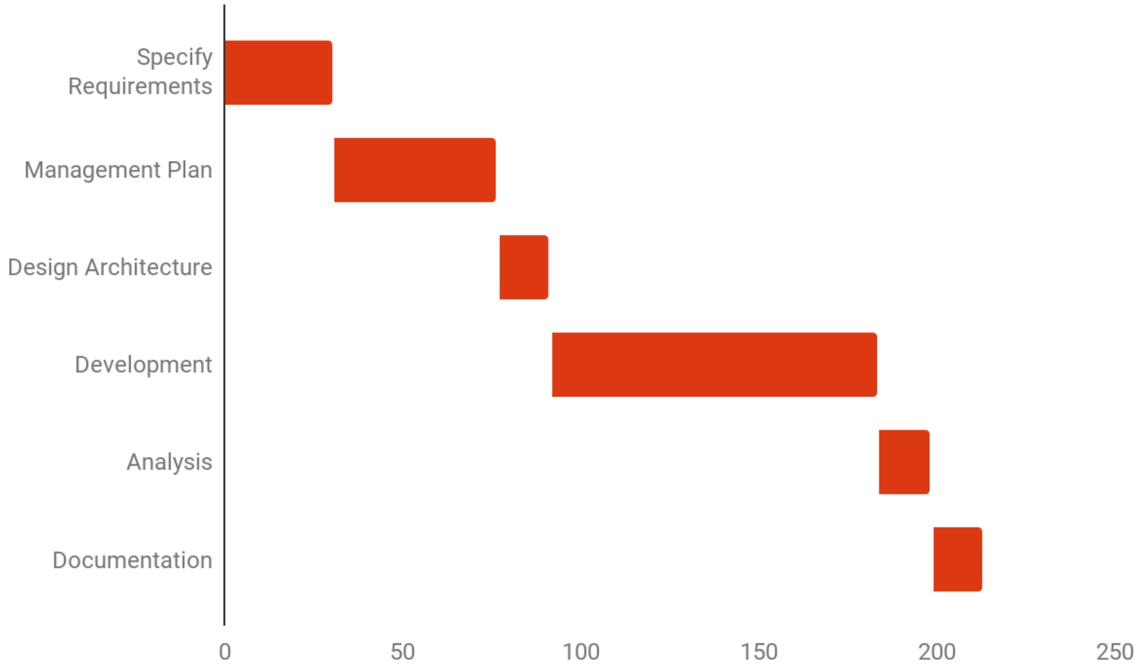
- Application will run only on Windows and Linux platforms.
- Application will support only text system terminal as the output device.
- All third-party libraries must be open-source.

1.5 Schedule and Budget Summary

1.5.1 Cost Estimate

One software engineer at 4 hours per week for 24 weeks: 96 hours * $20/hr = $1920.

1.5.2 Schedule Summary



1.6 Success Criteria

A working prototype, which is easy to use, that allows users to play game.

1.7 Definitions

Snake Application – the product that is being described here. The software system specified in this document.
Project – activities that will lead to the production of the Snake application.
User – the person or persons who will actually interact with the Snake application.
Use case – describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.
Scenario – one path through use case.
Actor – user or other software system that receives value from a user case.

2. Startup Plan

2.1 Team Organization

Role	Actor	Responsibility
Project Manager	Alexander Orel	Coordinate communications within group, coordinate communications outside group, break out tasks and assign them to teammates.
Developer	Alexander Orel	Develop software based on requirements and architect specifications.

2.2 Project Communications

Event	Information	Audience	Format	Frequency
Stand Up Meeting	Task status: completed since last meeting and planned for next.	All team members	Informal meetings	As needed

2.3 Technical Process

An iterative and incremental agile software development process is planned. Feedback will be used from each iteration to improve the next. The first iteration will focus on basic functionality of the application. Subsequent iterations will build upon that and incorporate more features as time allows.

2.4 Tools

- Programming Languages – C++11
- Operating Systems – Windows, Linux
- Version Control – git
- Development Tools – CodeLite
- Frameworks – SQLite

3. Work Plan

3.1 Activities and Tasks

Name	*Description*	*Planned start date*	*Actual start date*	*Planned end date*	*Actual end date*
Cross-platform window	Create cross-platform game window which works in system terminal				
Main menu	Create interactive game menu with selector moving to activated item				
Screen for each menu item	Add screen for each menu item which contain a poor content for beginning				
Player’s nickname	Add a form for choosing exist or entering new one player's nickname				
Game’s main screen	Create bounded empty game main screen which appears when the Play item of main menu is chosen				
Main character	Add main character to the main game screen which can move and die when the bound of game field is reached				
Level’s goodies	Add goodies for main character which makes its longer when reached by “head” of the main character				
Level’s walls	Add walls to the main game field which serve as an obstacle				
Levels generation	Create a system of levels generation				
High scores database	Create the database of high scores and nicknames of players				

3.2 Release Plan

Iteration #1
Summary: Demonstrate fundamental architecture and have only cross-platform game window.

Features / Deliverables	Estimated Effort	Actual Effort
Architecture / Framework design		
Cross-platform window		

Iteration #2
Summary: Create the game main screen which demonstrate all gameplay features.

Features / Deliverables	Estimated Effort	Actual Effort
The game main screen		

Iteration #3
Summary: Create high scores database and game screen for displaying this data.

Features / Deliverables	Estimated Effort	Actual Effort
High scores database		

3.3 Iteration Plans

A detailed iteration plan will be provided for Iteration #1. Further task details are available in the schedule.

3.3.1 First Iteration

Cross-platform main window functions for both Windows and Linux operating system.

3.3.2 Second Iteration

Game main screen which gives user an ability to use all gameplay features.

3.3.3 Final Product

Complete cross-platform game application with ability to save high score and nickname of the player.

4. Control Plan

4.1 Monitoring and Control

The following list of dates includes formal reviews. Milestones are included to reference where the project is scheduled to stand as these reviews occur:

Date	Review / Milestone
31.07.2017	Milestone: Technical Prototype Complete
31.07.2017	5-Minutes Status Report
15.08.2017	Milestone: Iteration #1 Complete
01.10.2017	Milestone: Iteration #2 Complete
15.10.2017	Milestone: Iteration #3 Complete
30.11.2017	Final Presentation

5. Supporting Process Plans

5.1 Risk Management Plan

Rank	Risk	Probability of Loss	Size of Loss	Risk Exposure	Response
1	Schedule / time line delivery	Likely	Major	High	Mitigate: Stick to the schedule.
2	Learning curve for new tools and technologies longer than expected	Likely	Moderate	Moderate	Begin working on a basic prototype early to test out fundamental programming concepts.
3	Operating systems not support some needed functionality	Unlikely	Minor	Low	Use libraries and frameworks to emulate needed functionality

5.2 Test Plan

The test plan defines the items that will be tested, methods for testing, and a schedule detailing the tasks, owners, and timeline. The test plan will be available in a separate document in the version control system.

5.3 Product Acceptance Plan

At the conclusion of each iteration, the prototype created will tested to ensure it meets the requirements of that iteration. For the final iteration, product acceptance testing will ensure that the prototype functions as expected with real users behavior.