SOFTWARE DEVELOPMENT PROJECT

Alva Fandrey
1dv600

2019-02-21

| Contents

1 | Revision History

Date	Version	Description	Author
5/2-19	V1.0	Assignment 1	Alva Fandrey
20/2-19	V2.0	Assignment 2	Alva Fandrey

2 | General Information

Project Summary	
Project Name	**Project ID**
Hangman	af222ug_1dv600
Project Manager	**Main Client**
Alva Fandrey	End-user
Key Stakeholders	
Manager Developer/Software engineer Tester End-user Teachers/tutors (in course 1dv600)	
Executive Summary	
A JavaScript console application of the game Hangman.	

3 | Vision

The goal with this project is to create a JavaScript console application of the game "Hangman". The basic idea of Hangman is to let the player guess a word one letter at a time.

When the application is started the user will login by entering a nickname. The player should be greeted with a menu where they can choose to start the game, watch the high score list or quit the application. The player should be able to quit the application at all times. When the user chooses to start the game a word from a predefined list will be randomly picked and the number of letters will be displayed with equally many underscore signs. The player will then guess the word by suggesting letter after letter. For every wrong guess the player has the game will build a part of a man getting hanged. The player can have ten faulty guesses before the game is over. The high score list will present how many guesses a player has had and also how long time it took for the player to finish the game. If the user logging in is an admin, that user will be able to manage the list of words by view all available words, adding new words and removing words.

Hangman is a very popular game to play and there are many different versions of the game available to play on the internet. To make this game differentiate from other Hangman games found on the internet our version will contain a high score function based on number of guesses and time spent guessing.

3.1 Reflection

I found it a little bit hard on what to include in the vision since it was described different in the template as to what was in the lecture. According to the template, and what was said in slack from the tutors, the vision was supposed to describe the system being created so this is what I have done. In the template from lecture 3 you were supposed to include "Problem statement" and "Stakeholders". However, I chose to not include these parts because there will be more information about stakeholders in the project plan and the "Problem statement" was hard to describe since this application won't really solve a problem it's just something that has to be done as a part of this course? I did however include some bits from the "Product position statement" when I explained what would differentiate this application from other similar applications.

4 | Project Plan

4.1 Introduction

The purpose of this project plan is to gather all necessary information to plan and control the project being developed during this course. The project plan will describe the approach to develop the Hangman game. This project plan will be used by the project manager for planning and by the project team members to understand what they need to do. This project plan will describe the different stakeholders and what resources will be available during this project development. There will also be an overall project schedule provided. This is a living document and it will be updated throughout the course.

4.2 Justification

This application will be created as a project for the course 1dv600.

4.3 Stakeholders

Manager – is responsible for project planning, breaking down the work into parts and assign them to team members, anticipate what problems might arise, and prepare tentative solutions.

Developer/Software engineer – is responsible for developing the planned project.

Tester – is responsible for testing the functionality of the project with manual and unit tests.

End-user – main client who will approve the finished project.

Teachers/tutors (in course 1dv600) – will grade the assignments as well as the finished project.

4.4 Resources

The time budget for this project is 9 weeks, for more detailed information about deliverables check out *4.6 Overall Project Schedule.*

IDE – Visual Studio Code

Programming Language – JavaScript

Personnel available during the development of this project are the manager, one developer/software engineer and one tester.

4.6 Overall Project Schedule

Start Date	**End Date**	**Deliverables**
22/1-19	8/2-19	Assignment/Iteration 1
8/2-19	21/2-19	Assignment/Iteration 2
21/2-19	8/3-19	Assignment/Iteration 3
8/3-19	w.12	Iteration 4 (Finished project)

4.7 Scope, Constraints and Assumptions

4.7.1 Scope

The general idea of the project is to create an ordinary hangman game where the player will be provided with a number of underscore signs, which are equal to the number of letters in a randomly picked word. The player will then guess this word by suggesting letter after letter. The player can have ten faulty guesses at most and for every faulty guess the game will build a part of a man getting hanged. To make this game different from other Hangman games we have chosen to add in some extra functionality in form of a high score list.

4.7.2 Constraints

This should not be a very time consuming project, however the time could be a limitation anyways if the time is not well planned. Since we do not have all knowledge about the project from the start, it will make it harder to plan the time.

4.7.3 Assumptions

For this project it is assumed that the end-users know how to use the console to play this application.

4.8 Reflection

I thought some parts of creating the project plan was a bit difficult since there was not that much of a description in the project plan and I did not find that there was that much of information in the lecture either. I did look through the book and found some information that combined with the information in the template made it a little bit easier. I skipped the "Hardware & software requirements" section because I did not know what to write and it felt like it was almost the same information that I put down under "Resources". There is maybe not that much information under resources either, but I was unsure what was supposed to be written here and maybe we get more knowledge on what resources will be needed in the future assignments. Also it seems unnecessary to include "costs" under resources because I do not believe there will be any costs. I also found it hard with what was supposed to be included under "Scope, Constraints and Assumptions", but after some tutoring I think I understood what was supposed to be included it was just hard to come up with constraints and assumptions anyway.

5 | Iterations

5.1 Iteration 1

Task	Estimated Time	Responsible
Create GitHub repo	15min	Developer
Create project documentation	4h	Manager
Implementation – idea and skeleton code	3h	Developer

5.2 Iteration 2

Task	Estimated Time	Responsible
Plan for assignment 2	0.5h	Manager
Update Use Case Diagram	0.5h	Manager
Write UC2 – Play Game	1h	Manager
Write UC for extra features	2h	Manager
Create state machine for Play Game	2h	Manager
Implement the modelled behaviour	4h	Developer
Create class diagram from implementation	2h	Manager
Update Project Plan	1h	Manager

5.3 Iteration 3

Task	Estimated Time	Responsible
Create Test Plan		Manager
Manual Test Cases (using the client application)		Tester
Unit Tests		Tester
Implementation – additional features		Developer

5.4 Iteration 4

Task	Estimated Time	Responsible
Document new features		Manager
Test new features		Tester
Implementation – complete the project		Developer

6 | Risk Analysis

6.1 List of risks

Risk	Probability	Effects
The time required to develop the software is underestimated	High	Serious
Team members are ill and unavailable at critical times	Moderate	Serious
Facts needed to complete the system architecture are not known, or known imprecisely	Moderate	Catastrophic
Changes to requirements that require major design rework are proposed	Low	Serious
Lack of definition in the system requirements	High	Catastrophic

6.2 Strategies

Risk	Strategy
Underestimated development time	Exaggerate when doing time estimation and plan with “slack”.
Staff illness	Make sure team members have understanding about each other’s job.
Lack of knowledge	Investigate what facts will be needed to complete the system at the beginning of the project.
Requirements changes	Measure the impact in changing the requirements and explain the consequences.
Lack of definition	Try to get as much information about the requirements at the start of the project.

6.3 Reflection

There were many examples in the book about different risks, but many of these risks I excluded since I believe this is not a huge project with no real budget (except for time) and it’s only one person working all roles as well as I’m being my own “customer” also. Therefore risks like “impossible to recruit staff” and “organization is restructured” is not necessary to include here, but I wrote down the risks I thought was relevant for this project. I also used some of the risk examples from the lecture. When it came to “Strategies” I tried to come up with my own strategies and not use exactly what was said in the book.

7 | Time log

7.1 Assignment 1

Date	Task	Estimated Time	Actual Time
24/1-19	Create GitHub repo	15min	15min
5/2-19	Create project documentation	4h	5.5h
7/2-19	Start implementation	3h	1h 15min

7.1.1 Reflection

When it comes to creating the project documentation I did not include the time it took to watch the lectures and read the pages in the book, however sometime I had to go back and watch some parts of the lecture or read about something in the book again when I was unsure on what to write and that time is included and I think that is why there is a time difference because I was often unsure on what to write because some parts in the template was not that well describe according to me. I did add extra time to implementation because when I estimated the time I was unsure on how much code we were supposed to implement at this stage and therefore there is a big time difference.

7.2 Assignment 2

Date	Task	Estimated Time	Actual Time
	Plan for assignment 2	0.5h	45min
	Update Use Case Diagram	0.5h	0.5h
	Write UC2 – Play Game	1h	50min
	Write UC for extra features	2h	2.5h
	Create state machine for Play Game	2h	1h 45min
	Implement the modelled behaviour	4h	6h
	Create class diagram from implementation	2h	0.5h
	Update Project Plan	1h	45min

7.2.1 Reflection

I started with reading the chapters in the book and watching pre-recorded lectures before I watched the new lecture where it was said that we should include time for doing these things. I did not want to just make something up so therefore I chose to not include these parts in my time log. I did estimate some time for future iterations in the previous assignment because that was how I understood that task, but since the first task in this assignment was to plan I had to re-estimate this iteration. I also chose to update my vision a little bit when creating the use cases. The implementation took a bit longer than estimated because I had trouble deciding on how I wanted the application to function and I ended up with only one file that does almost everything except for actually starting the game (which might not be the best, but it was the easiest). This made the class diagram almost too easy to create. The only thing that took a little bit of time is that I was unsure if the app.js was supposed to be included in the class diagram. I did some research and could not really find any information so eventually I just decided to put it in the class diagram also since it does use the Hangman.js.

7.3 Assignment 3

Date	Task	Estimated Time	Actual Time
	Create Test Plan		
	Manual Test Cases (using the client application)		
	Unit Tests		
	Implementation – additional features		

7.4 Assignment 4

Date	Task	Estimated Time	Actual Time
	Document new features		
	Test new features		
	Implementation – complete the project		