SOFTWARE DEVELOPMENT PROJECT 7,5 hp

Lone Nilsson

Linnæus University

2019-02-21

Contents

1. Reversion History

Date	Version	Description	Author
2019-02-08	v.0.1.0	First iteration, planning and structuring work	Lone Nilsson
2019-02-21	v.0.2.0	Second iteration, models and easy program code implemented	Lone Nilsson

2. General Information

Project Summary	
Project Name	**Project ID**
Hangman	ln222tu_1dv600, hangman-game (https://github.com/onlylonely1986/ln222tu_1dv600/tree/master/hangman-game)
Project Manager	**Main Client**
Lone Nilsson	**Tobias Andersson Gidlund,** **Tobias Ohlsson,** **Daniel Toll**
Key Stakeholders	

- **Teachers**
- **Me**
- **My friends**

Executive Summary

This document is about developing a software-program of the classical game “hangman” in javascript. It is not only about building up the code, it is more focus on how to plan, and build a project, learn how to estimate risks and estimate a time log. It also covers a deeper knowledge about testing.

3. Vision

The vision about this project is to make an application of the game hangman played in the console. The user should be able to choose from a menu what to do from a list of numbers. The picture of a hanged man should be presented with ACII-signs. The game is about to guess a word from empty placeholders, try to guess letter for letter. If the letter is correct it will be put in the right place, and if wrong the picture of a hanged man should be shown step by step. The words are taken from a document of words (nouns) and will randomly be picked every time a new game is started. The user should be able to write in a username and save results in a high score list. The game should be built in JavaScript by separate modules for each functionality.

Reflections about writing a vision

It is difficult to know *how* to write it because it is a new thing to think about and write. It might be fun to write visions and dreams. It is of course important that every person in a development-team is on the same goal to reach. In this case it is only me developing this simple game in console, and maybe it is not that fashionable and fancy, but I try to make it as good as I can.

4. Project Plan

4.1 Introduction

A hangman game should be made.

4.2 Justification

It is an assignment to be made in a course at university and at the same time I learn more about how to plan and work with a software project in a professional way.

4.3 Stakeholders

Project manager is the person responsible for running the project from start to end.

Programmer/software developer is the person who writes the software-program.

Tester is the person that writes tests and try out if the program is safe and run well, without any bugs.

Client is the company that ordered the assignment, in this case it will be the teachers.

End users is the people who can use the program in the end. In this case maybe myself or my friends.

4.4 Resources

Resources is money, time, software, hardware and staff who work with the project.

In this case there is no money involved because it is a university course, the time is set by the length of the course. The hardware to work with is my own Lenovo-computer.

The software in this project is Visual Studio Code, Node.js, gitbash and github.com. another resource is the time we have to complete the work, it should all be finished in week 12. The only employee in this project is me. I run every role in it.

4.5 Hard- and Software Requirements

The hardware to work with is my Lenovo-computer. It should also run the game in a terminal, on my computer or in another one.

The software in this project is Visual Studio Code, Node.js, gitbash and github.com. The requirements by these programs is

4.6 Overall Project Schedule

w. 6 hand in iteration 1 (8/2)

w. 8 hand in iteration 2 (21/2)

w. 11 finish the program and testing, iteration 3 (10/3)

w. 12 hand in rapport, iteration 4 (21/3)

4.7 Scope, Constraints and Assumptions

This project is about developing a console-based game, hangman. It is also about learning more around the professional process when developing software projects. The scope reaches in this case not so much about design, frameworks, and good-looking stuff, but it is not either unimportant. It is more about learning analysis, *what* to develop - w*hat* kind of application it should be and how to plan my work and follow my plan in a really good way. In the scope of this project I'm going to build in the opportunity to choose the difficulty level in three steps. It is common to have opportunity to start new game, quit game and a button for reading "how to play".

Some constraints to take in account is my own lack of experience and of course, even if I make a good plan and risk analysis and stuff the time limit could always be a problem.

I have one assumption about users in this case. I think that they are familiar with using consoles or terminals. But I also have in mind to build it very clearly so that every person with some knowledge about computers and computer-games have ability to run the game.

5. Iterations

5.1 Iteration 1

w.4

- Introduction in course (2 h)
- Reading trough material (2 h)
- Reading the book: Software Engineering, Ian Sommerville, (5 h)
- Starting to prepare the rapport in word (2 h)
- Work with questions to prepare before exam 1 (8 h)
- Listen to rerecorded lecture (1,5 h)

  (Summa: 20,5 h)

w.5

- Reading the book: Software Engineering, Ian Sommerville, (10 h)
- Prepare for online-exam (8 h)
- Make online-exam 1, (0,5 h)
- Listen to rerecorded lecture (1,5 h)
- Listen to lecture (1,5 h)
- Working with rapport in word (4 h)

  (Summa: 25,5 h)

w.6

- Reading the book: Software Engineering, Ian Sommerville, (5 h)
- Searching more information about the project (2 h)
- Working with rapport (10 h)
- Building skeleton-code and push it to github.com (1 h)

  (Summa: 18 h)

5.2 Iteration 2

w.7 - 8

- Reading the book: Software Engineering, Ian Sommerville, (15 h)
- Listen to lectures, (8 h)
- Make online-exam 2, (8 + 0,5 h)
- Writing on rapport and searching for more knowledge (10 h)
  - build use case diagram
  - write use case text
  - create UMLs
- Building up code (10 h)
  - start up with skeleton-code, using UMLs and use case
  - build a functional game
  - build in a database with words

-

5.3 Iteration 3

w.9 - 10

- Reading the book: Software Engineering, Ian Sommerville, (15 h)
- Listen to lectures, (8 h)
- Make online-exam 3, (8 + 0,5 h)
- Writing on rapport and searching for more knowledge (10 h)
- Building up code
- Testing

5.4 Iteration 4

w.11-12

- Finishing the rapport and hand it in
- Finish project, clean and polishing the code
- Hand in and make a release at github

6. Risk Analysis

Project managers have to assess the risks that may affect a project, monitor these risks, and take action when problems arise. In this project it is some sort of risks to deal with. Specially because I am the developer, and my experience is not so great yet. If you are more experienced, you have learnt from your earlier mistakes. It is more difficult to plan if you have not developed much before. It is difficult to deal with how much time it will take, and how to create the plan and vision. Because I don't really know everything about what you *can* do, and what is my highest level to reach at this moment?

6.1 List of risks

Project risks:

Risks that is about the project itself. I can be about the developer being sick or other unexpected things coming up during the project. About the hardware or software or about the employees. It is important that the project manager put right person in right place. Risk I identify at this moment:

- time-plan (high risk)
- knowledges/experience (moderate risk)
- failures/difficulties (high risk)

Product risks:

A product risk influences the quality of the software being developed or the hardware itself.

Risk I identify at this moment:

- bad solutions that slows the system down (moderate risk)

Business risks:

Business risks affect the organization developing or procuring the software. For example, a competitor introducing a new product is a business risk. The introduction of a competitive product may mean that the assumptions made about sales of existing software products may be unduly optimistic.

Risk I identify at this moment:

I should not say that there are any business risks in this project because it is a course at university. Of course, there are lots of competing hangman games out there, so it will probably not sell very well. (But I now it isn't the greatest sense here.)

6.1 Strategies

I must manage risks and take control over them by avoiding it or by minimizing its effects on the project. If I make a good plan and thinking about every step to get through, it should be easier to not fail the time-plan. I must be one step ahead because I am not only student, also a mother and I can't plan if my kids or me get sick any day in the middle of project. But with hard work and good planning I minimize the risk about failing on this one.

The other point about my own knowledges, is to not set too high goals, because even small developing projects could be difficult to finish when you are not that experienced yet.

The third thing about failures that causes me as developer problems, both with time plan and maybe also affect the product, so the computer gets slow caused by bad solutions. I don't really know how to manage or minimize these effects, because I think it is a part of programming to struggle. Problem solving is what it is. And therefore, it also is a part of time planning, to set reasonable goals and put in lots of extra time in the project to get rid of troubleshooting process.

7. Time log

7.1 Assignment 1

When:	What to do:	Time it will take:	Time it took:	Reflection:
w.4 22/2	Reading trough material		2 h	
23/2	Introduction in course		2 h	
	Reading the book: Software Engineering		5 h	
	Starting to prepare the report in word		2 h	
	Work with questions to prepare before exam 1		8 h	
	Listen to rerecorded lecture		1,5 h	
	Sum:	20 h	20,5 h	
w.5 28/2	Reading the book: Software Engineering		10 h	
	Prepare for online-exam		8 h	
	Make online-exam 1		0,5 h	
	Listen to rerecorded lecture		1,5 h	
	Listen to lecture		1,5 h	
	Working with report in word		4 h	
	Sum:	20 h	25,5 h	
w.6 4/2	Reading the book: Software Engineering		5 h	
4/2	Searching more information about the project		2 h	
5/2	Working with report		1 h	
6/2	Building skeleton-code and push it to github.com		1 h	

6/2	Write on report	2 h	4 h	It took longer time than I thought, to understand and think about what to write and how to describe it.
7/2	Write on report	3h	2 h	
8/2	Write on report, complete the first iteration	2h	2, 5 h	It took longer time than I thought, to understand and think about what to write and how to describe it.
	Sum:	20 h	17,5 h	
	Sum Iteration 1 (w.4-6)		**63 h**	I have put in more time then the course requirement says, it is a 50 %-course, so it is fair to put in approximately 20 hours a week. And it is about what I did.

In this first part I have done estimates afterwards (first 2 weeks in my time log) because I did not pay attention at this in a early time. So next time, I should be more attentive and read through all parts before I start my work.

7.2 Assignment 2

When:	**What to do:**	**Time it will take:**	**Time it took:**	**Reflection:**
w.7 11/2	Reading trough material on course site	1 h	1 h	
11/ 2	See tutorial from last week	2 h	1 h	Split it up I two parts, and continue later.
11/2	See tutorial from last week	1 h	1 h	
11/2	Reading chapter 4 in book	1,5 h	1,25 h	
12/2	Reading chapter 5 in book	0,5 h	0,5h	
13/2	Reading chapter 5 in book	1 h	1 h	
13/2	Build a use case diagram	0,5 h	0,5 h	
13/2	Start coding something	1 h	1 h	
13/2	Search more knowledge about UML	1 h	1 h	
13/2	Listen to rerecorded lecture 4	1,5 h	1, 5 h	
14/2	Reading the book chap 5	1, 5 h	1 h	
14/2	Listen to rerecorded lecture 5	1, 5 h	1, 5 h	
14/2	Building UMLs	4 h	0 h	Did not reach my goal, tomorrow new take.

14/2	Reading material, preparing UMLs etc	1 h	1,5 h	
17/2	Task 2 Write Use Cases	1 h	0,75h	It went faster than I thought now because I have read and thought about it quite a lot.
17/2	Task 3 Modeling Behavior del 1	8 h	3,75h	
	Sum:	20 h	18,25h	
18/2	Task 3 Implementing code	4 h	2 h	
	Lecture 6	1, 5 h	1 h	Jesper is good in 1,5 speed on youtube ;)
18/2	Reading the book: Software Engineering	3 h	1,5h	Chapter 6, chapter 7
18/2	Lecture Hobbe	1, 5 h	1 h	
19/2	Lecture 7	1, 5 h	1,25h	
19/2	Working with questions/study material	1,5 h	1,5h	
19/2	Reading the book: Software Engineering	3 h	1 h	Chapter 15
19/2	Task 3 Implementing code del 2	2 h	1 h	
20/2	Task 4 Modeling Structure, Class diagram	2,5 h	0 h	Did not feel very well, wasn't able to finish what I supposed to.
20/2	Lecture 8	1,5 h	0 h	Did not feel very well, wasn't able to finish what I supposed to.
20/2	Lecture Hobbe	1,5h	2 h	
20/2	Implementing code	2 h	2 h	
21/2	Reading the book: Software Engineering. Chapter 20, study before onlinetest	3h	3h	
21/2	Task 4 Modeling Structure, Class diagram	2 h	2 h	

21/2	Write on report, complete the second iteration	2h	2 h	
21/2	Make online-exam 1	0,5 h	0,5 h	Did not pass the test, it was several questions with alternatives where I missed one thing or boxed one to much ☹
	Sum:	20 h	18,25h	
	Sum Iteration 2 (w.4-6)	**40 h**	**36,5h**	

Reflections

I think I am quite good at planning and using the time I need to do what I should. I study hard and not slacking, and I always do my best. In this course it is often information about what to do and what to learn is a bit all over the place, so it takes a lot of time to structure material and feel comfortable about it.

In my planning I did put colours to each part of the tasks to easier discuss and see what I put time in. In my planning I did some assumptions, and I can now afterwards see that I overall. I can see that I used reading in book a lot less than I first planned and thought, so I can learn to next iteration to plan less time in that moment. I did a miscalculation about how many lectures I should listen to, but I also noticed that it was a good thing to speed lectures at youtube a bit, because I still had the opportunity to understand and get the substance out of it. It was a way to save some time in that. Working on code and studying and preparing models was probably quite good in my assumptions about time to put in it. It is improving doing these estimates and reflections about it. It also helps me to focus more on one task at a time, and not mix and match. Multitasking is not very effective, it is more effective to make a plan, and then follow it as good as possible.

7.3 Assignment 3

8. Handing in