Hang The Man
Project Plan

Thimmy Stenlund
Returns Void

Date: 08/02/19

Returns
VOID

Contents

Revision History

Date	Version	Description	Author
08/02/19	1.0	Hang The Man project setup and Project Plan creation	Thimmy Stenlund

General Information

Project Summary	
Project Name	**Project ID**
Hang The Man	1073
Project Manager	**Main Client**
Thimmy Stenlund	Friends & Family
Key Stakeholders	
Thimmy Stenlund Returns Void	
Executive Summary	

Vision

Returns Void envisions a Hangman Game slightly outside the ordinary boundaries of the classic Hangman specifications. While it will support (and start as) the basic, normal game flow that everyone has come to love (and hate), the aim is also to support a secondary game mode that aims to hang the man – thus showing how the name 'Hang The Man" came to be. First, core features will be implemented, supporting the original game. As you start a new game, you will see a series of underlines showing how many letters the word contains. In turns, you will guess a letter. If the letter is found within the word, it will then display the letter instead of underline in the correct position of the word. Should the game not find your guessed letter in the word, you will see that with each failed letter – an island, gallows, nose and finally a person hang in the noose will appear in the console. When all parts are visible – you lose. If you manage to solve the word, there should be a winner announcement and in future iterations of the game – a scoreboard displaying number of guesses and time it took. When the base game is functionable, to a reasonable degree, development of the "Hang The Man" special mode will begin – where you aim to hang the man, and not save him from hanging. In this mode, there will only be words with 7 letters. If you manage to guess all 7 letters without more than 5 missed guesses, the (guilty) man will hang. It will be developed with JavaScript as its main language.

Key Needs & Features:

* Core Game – Guess letters in a word and either fail or win depending if you can guess it before losing.
* Feature – Reverse the game mode, so that the plan is to actually hang the man.
* Feature – Time it took to find out the word.
* Feature – High score list sorted after the time it took.

Reflections of the Vision Document:
I personally think its important that the Vision Document is present and available to all members of the team. As it's named, it describes the key vision of the game or application, thus making sure that everyone has a similar view on what and how the application should be working in the end. It's a type of "soft requirements", which better describes the actual mechanics than just a point-list with each requirement. It's a gathered vision of what is to come. It's quite clear, as you write the document, that it has several benefits. Not only does it help the team to get a shared vision, it also makes you think twice (or even thrice) times about the features, future and base development of the Software. I would deem it vital.

Project Plan

4.1   Introduction

Hang The Man is a classic with a twisted take on the game mode that has enthralled millions across the world, both on pen & paper but also through games. We aim to create a new variant of this classic – with a new game mode that aims to have you hang the (guilty) man before the time runs out.

4.2   Justification

Todays minigames can't hold up to the true classics. One important factor is that it's a learning game – meaning that a child/youth can learn English and English words from playing the game. The twist with having a game mode that will instead hang the man, might be borderline offensive to some – but there are a lot of other games that brings much more violence and gore to young children. This is just a fun twist.

4.3   Stakeholders

Thimmy Stenlund – The current owner and founder of ReturnsVoid.
ReturnsVoid – A programmer group that focuses on creating simple and reliable applications & games.

4.4   Resources

Resource – Time:
For this project, roughly 20 hours per week will be allocated in different spans over the weekdays and weekends.

Resource – Money:
The budget set aside for this project is currently standing on 0$. Time spent will allot for the actual cost (as time always has a monetary value in reality). Should a cost occur, that will be discussed and agreed upon before added to the resource field.

Resource – Personnel:
Project Manager – Thimmy Stenlund
Lead Developer – Thimmy Stenlund
Environment Specialist – Thimmy Stenlund
Test Engineer – Thimmy Stenlund

4.5   Hard- and Software Requirements

We are using JavaScript as our coding language to develop Hang The Man. The application will be developed on both laptop and stationary computer. Requirements to run the game, would be a PC, Phone or Tablet that can handle a web-browser and thus see the console, as the game will be played there.

4.6   Overall Project Schedule

First step (Process and Planning) is to be finished by 8/2 2019.
Second step (Modelling and Software Design) is to be finished by 21/2 2019.
Third and final step (Software Testing) is to be finished by 8/3 2019.

Slight delays can impend on the time plan and later dates can be determined, should the time plan fail. At
latest, the third and final step will be completed in August 2019. See iterations further down to see a
more in-depth view of the

4.7   Scope, Constraints and Assumptions

Scope for the project is displayed using the MoSCoW method.

Must:
- Core gameplay, the possibility to play the game in Console.

Should:
- Optional game mode that consists of actually hanging the man
- High Score list
- Time taken

Could:
- Database or Storage that store the high scores and/or the word list that is used to randomize a word.

Won't:
- Graphics
- Multiplayer
- Global Score list
- Installation executable

Reflection – Project Plan: It feels like it should be expanded and worked on further as the project goes. I
do understand its value in a project planning stage, and that it should be something that can be followed,
explained and simplistic yet detailed. It's a good overview really for the entire project. As I work in an
agile environment (and has done so in the last few employments), it's still a lot of new information to take
in. We use SCRUM or more correct – Large Scale Scrum, as our process of choice. The differences from
real world and in theory is quite large, but it's interesting to find new ways to plan and display what is
planned. Will take this into my team.

Iterations

Plan for four iterations, including this. This is a fine-grained plan on what is to be done in each iteration and with what resources. To begin with, this is a plan of what we *expect* to do, update this part with *additions* (never remove anything) when plans do not match up with reality. Also make time estimates for the different parts.

In this course the overall planning has in some ways already been decided, so use the template to provide more details on specific tasks that define *your* project. Remember that you can plan to add features to any of the phases as long as the main focus is also met.

The first assignment is to complete iteration one.

5.1   Iteration 1

What to do	How to do it	Who does it	Est. time	Deadline
Complete Project Plan	Add text and data into the current document.	Thimmy Stenlund	10 hours	8/2 2019
Create Github Repo	Setup a new github repo from the instructions on the moodle page.	Thimmy Stenlund	15 minutes	8/2 2019
Add Skeleton Code	Use Visual Studio Code to add "Skeleton" code to the Git repo.	Thimmy Stenlund	15 minutes	8/2 2019
Turn in Assignment 1	Check that documents are uploaded. Send link to git repo on moodle.	Thimmy Stenlund	15 minutes	8/2 2019

5.2   Iteration 2

What to do	How to do it	Who does it	Est. Time	Deadline
Model w/ UML	Model features in UML.	Thimmy Stenlund	4h	13/2 2019
Add Diagrams (UML) to Project Plan	Edit this document to have it display the UML for the features to be implemented.	Thimmy Stenlund	30m	13/2 2019
Start the game with a const word.	Add a const word, display _ for each letter. Guess all letters to win. (No fail available)	Thimmy Stenlund	4h	14/2 2019

Construct a counter for number of guesses to fail.	Add the counter that counts failed guesses. Make the player lose if that counter reach x.	Thimmy Stenlund	2h	14/2 2019
Arrange a way to display the hanging.	Add console.log calls that shows each step of the hanging as the player fails. "You lose" message to be displayed.	Thimmy Stenlund	2h	15/2 2019
Timer for the game.	Add a timer that count as the play2er guesses.	Thimmy Stenlund	2h	16/2 2019
Create a "menu" to start the game or optional game mode.	Menu addition will go fast. What is also required is turning around the main mechanic (new graphics). Rest can be reused and just revamped.	Thimmy Stenlund	4h	18/2 2019
Turn in Assignment 2	Add the code and documents to the git repository.	Thimmy Stenlund	30min	21/2 2019

5.3   Iteration 3

What to do	How to do it	Who does it	Est. Time	Deadline
Plan Tests	Plan the tests that needs to run through for these steps to be completed.	Thimmy Stenlund	12h	24/2
Perform	Implement and run tests on the application. Review and correct any issue on the code.	Thimmy Stenlund	16h	2/3 2019
Document the tests	Make sure that the documentation of the tests, the results and all Is collected and presented.	Thimmy Stenlund	4h	4/3 2019

5.4   Iteration 4

What to do	How to do it	Who does it	Est. Time	Deadline
Check towards the Project Plan	Compare requirements, see to it that all test run through.	Thimmy Stenlund	4h	6/3 2019
Review what features could be desired in the future.	Review and go through the code and playtest the completed product. Anything small to be added, removed or changed?	Thimmy Stenlund	2h	6/3 2019
Turn in Assignment 3	Add the code, documentation and the tests to the git repository. Build Release.	Thimmy Stenlund	6h	8/3 2019

Risk Analysis

All projects face risks that make it important to prepare for what might happen. Use the chapters in the book as well as the content of the lectures to identify the risks within this project. As always, write down your reflections on creating a risk analysis. This reflection should be about 100 words.

6.1 List of risks

List the identified risks and specify, as far as possible, the probability of them happening as well as the impact they would have on the project.

Risk	Probability	Impact on the Project
Project team member will not be in place when required	High	Severe
Risks with the hardware and software for the development	Low	Mediocre
Risk that the workstation environment of the user will experience malfunction	Low	Low
Risk to the project resulting from a mandatory completion date connected with the project	Medium	Low
Risk of exceeding the current expected deadline(s)	Medium	Mediocre

6.2   Strategies

Project team member will not be in place when required:
To minimize this risk, there will be a fluent schedule to have multiple times available to complete it (before its deadline). There should also be gaps that has not been accounted for that can be used to catch up, if the schedule would appear to not hold. It would lower the Impact from Severe to Low.

Risks with the hardware and software for the development:
Make sure to test any type of modules or solutions before using them in a release environment. Do not add new technologies without making sure they work on the end environment.

Risk that the workstation environment of the developer will experience malfunction:
There are several workstations available, all setup with VSC and would just need GitHub to connect to the git repository. This remains a low risk.

Risk to the project resulting from a mandatory completion date connected with the project:
As mentioned before – this will be minimized as the time estimation is put high, thus making sure that extra time can be used to complete the iterations, if needed. There are also soft deadlines, that means that a later turn in can be done – although it should be avoided.

Risk of exceeding the current expected deadline(s):
There is a risk that the deadlines could be breached but with fluently allocated time and pockets of extra time that can be spared, chained with the deadlines being "soft" deadlines (a later turn in can be done but should always be avoided), it's as good as it can get.

Reflection – Risk Analysis:

There is always a great plus with doing a risk analysis. You think every risk through, see if there is anything to do in advance to minimize the risk and/or its impact on the project. The creation itself is quite straightforward, I could have added more risks but decided to go with this for now. Evaluation of the risks and their impact feels like something that you always should do when planning for a project. I feel personally that I will expand on this knowledge and read up on the risk assessment part of Project Planning. I need it for my own future projects that me and some friends are about to start up.

Time Log

Date	Time	Task	Actual Time	Analysis
8/2 2019	10 hours	Complete Project Plan	~10 h	-
8/2 2019	15 minutes	Create Github Repo	15 minutes	-
8/2 2019	15 minutes	Add Skeleton Code	15 minutes	-
8/2 2019	15 minutes	Turn in Assignment 1	15 minutes	-
12/2 2019	4h	Model w/ UML		
13/2 2019	30m	Add Diagrams (UML) to Project Plan		
13/2 2019	4h	Start the game with a const word.		
14/2 2019	2h	Construct a counter for number of guesses to fail.		
14/2 2019	2h	Arrange a way to display the hanging.		
15/2 2019	2h	Timer for the game.		
16/2 2019	4h	Create a "menu" to start the game or optional game mode.		

20/2 2019	30min	Turn in Assignment 2		
24/2 2019	12h	Plan Tests		
2/3 2019	16h	Perform		
4/3 2019	4h	Document the tests		
6/3 2019	4h	Check towards the Project Plan		
6/3 2019	2h	Review what features could be desired in the future.		
8/3 2019	6h	Turn in Assignment 3		