

---

# HANGMAN PROJECT

---

**Mohammadali Rashidfarokhi**  
Linnaeus University

2020-01-31

## Contents

|  |              |
|--|--------------|
| <b>1 Revision History .....</b>              | <b>1</b>     |
| <b>2 General Information .....</b>           | <b>2</b>     |
| <b>3 Vision.....</b>                         | <b>3</b>     |
| <b>4 Project Plan .....</b>                  | <b>4</b>     |
| 4.1 Introduction .....                       | 4            |
| 4.2 Justification .....                      | 4            |
| 4.3 Stakeholders .....                       | 5            |
| 4.4 Resources .....                          | 5            |
| 4.5 Hard- and Software Requirements .....    | 5            |
| 4.6 Overall Project Schedule.....            | 5            |
| 4.7 Scope, Constraints and Assumptions ..... | 5            |
| <b>5 Iterations .....</b>                    | <b>7</b>     |
| 5.1 Iteration 1 .....                        | 7            |
| 5.2 Iteration 2 .....                        | 7            |
| 5.3 Iteration 3 .....                        | 9            |
| 5.4 Iteration 4 .....                        | 9            |
| <b>6 Risk Analysis .....</b>                 | <b>9</b>     |
| 6.1 List of risks.....                       | 9            |
| 6.2 Strategies .....                         | 10           |
| <b>7 Time log.....</b>                       | <b>11</b>    |
| <br><b>8 Handing in .....</b>                | <br><b>9</b> |

|

## 1. Revision History

| Date       | Version | Description  | Author                    |
|------------|---------|--|---------------------------|
| 2020.01.31 | 0.1     | For creating the first iteration, different tasks such as vision, project plan and skeleton of hangman project were covered. | Mohammadali Rashidfarokhi |
|            |         |  |                           |
|            |         |  |                           |
|            |         |  |                           |

## 2. General Information

| Project Summary   |                     |
|---|---------------------|
| Project Name  | Project ID          |
| Hangman project   | 0003                |
| Project Manager   | Main Client         |
| Mohammad Ali Rashidfarokhi  | Linnaeus University |
| Key Stakeholders  |                     |
| <ul style="list-style-type: none"><li>• User</li><li>• Developer</li><li>• End-user</li></ul>   |                     |
| Executive Summary   |                     |
| <p>This project aims to create a hangman game base on java programming language. The user must choose different letters to complete the missing words in several attempts. There are different modes for playing the game. If the user wins, a list of information is necessary for user to fill out.</p> |                     |

|

### **3. Vision**

Hangman is a very popular game. It can be created by different programming languages (e.g. JavaScript, C++, python etc.). Since java is one of the programming languages that enables the developers to create such games, this project will use java programming language to fulfill the purposes of the project.

The general idea behind this game is that after the developer has carried out all essential actions to create the game, the user is supposed to guess a specific word by entering different letters. The player is presented with different words. Consequently, the user must choose correct letters to win the game. The game will be based on different modes. Therefore, playing the game will become challenging and fun at the same time.

There will be an already loaded list of words in English that will be used for the game so that the game will become fun to play for users because each time playing the game, the user will face with a new challenge (new word to guess).

#### **Personal Reflection:**

The creation of this game is a way to improve the skills of programming and test the planning skills at the same time. A useful website that I used a lot and helped me along the process was the java API, which was unclear for me at the beginning, but thanks to the hangman project not completely but it became clear for me. However, there were some difficulties along the way. Due to the shortage of time, it was difficult for me to think about the whole project since I was unfamiliar with the process, but I tried to cover all the necessary features that their existing is essential for creating this game.

## **4. Project Plan**

The developer aims to create a game called hangman to challenge the player to guess some letters from specific words. Once the player has started the game, the screen will display a message and ask the user to state whether ready to begin the game or not.

Since the game is not an ordinary game, the game has different modes of playing. That is, if the user decides to begin the game (the user will be asked if he/she is ready to start the game), the game will provide the user with 2 modes of playing (hard and medium). As it is obvious winning the game in hard mode is challenging (number of attempts to choose the missing letters will be reduced to 6). For helping the player, an option that helps the player with the game will be in the menu of game.

If the user wins the game, there should be a form that asks the user to fill out some data. Then, a farewell message will be displayed. On the other hand, there will be an option in the game menu that will enable the user to restart the game.

If the time permits, there is a possibility that more exciting features take a place in the game (e.g. limiting the user with a timer to choose the missing letters or add graphical features to the game).

### **4.1 Introduction**

A game called hangman will be created to enables user to play the game by choosing the missing letters from pre-defined words in the code. The player should find the missing letters based on the chosen mode for playing.

### **4.2 Justification**

Apart from the fun experience of creating the hangman game, one of the tasks of software technology course (1DV600) is to make a hangman game in detail and learn new techniques about software development and improve the planning skills.

### 4.3 Stakeholders

**User:** Logically the player is going to experience the game and all provided features in the game by a programmer, the user can either win or lose the game.

**Developer:** The responsibility of generating, implementing all essential features and deciding how to test the provided code is developer responsibility.

**End-user:** End-user is the individual who uses the product after it has been fully developed based on the user experience.

### 4.4 Resources

Since the game will be written in Java language programming, one of the most useful platforms to use will be Java API.

<https://docs.oracle.com/javase/7/docs/api/>

### 4.5 Hard- and Software Requirements

One of the useful programming environments is IDE. Therefore, the program will be created in IntelliJ IDEA and using information provided in Java API.

### 4.6 Overall Project Schedule

First iteration: 03/02/2020

Second iteration: 24/02/2020

Third iteration: 09/03/2020

Final iteration: 20/03/2020

|

#### **4.7 Scope, Constraints and Assumptions**

The main goal for this project is to create a simple game where the player must fill out the blanks by choosing the missing letters. Consequently, the player will be faced with two messages at the beginning of the program that are “welcome to the hangman game” and “Are you ready to begin?”, that happens inside the game. In order to spice up the game, different modes for playing will be available for user to choose from.

Then hopefully, the game will continue its flow and the player can enjoy the game.

#### **Personal Reflection:**

As I have mention before, this project can become challenging for someone who has not previous knowledge about this kind of projects and to create a flawless project which requires knowledge in both programming and planning. If I had previous knowledge in programming, it would be better and easier to plan a better project (more details).



## **5. Iterations**

The course 1DV600 required to complete the project in 3 different stages. That means, the developer is required to update and add extra features to the hang man game at each stage. The final stage, the developers is required to combine all stages together and at that point the game will be ready for the user to play and enjoy the hangman game.

### **5.1 Iteration 1**

This stage is the start of the project. The developer is required to plan perfectly and decide how the game will be created.

The essential classes that are required for the hangman project will be implemented. Obviously one of the important classes that is required for the project is the scanner class which allows user to enter the inputs for playing the game. Therefore, after creating the scanner class, the developer will write a code to print a welcome message to the user and ask the user to start the game with approval (yes/no). In order to guess to missing letters, the developer is required to provide a list of pre-defined words inside the code. This goal can be accomplished by using an array that is filled with list of words in English. For choosing random words, a

random class will be created.

More features will be added in the next iteration.

| Category            | Actual time dedicated | Assumption for finishing | Date of start | Date of finish | Deadline   |
|---------------------|-----------------------|--------------------------|---------------|----------------|------------|
| Project plan        | 4 hours               | 3 hours                  | 2020-01-31    | 2020-01-31     | 2020-02-03 |
| Risk analysis       | 55 minutes            | 2 hours                  | 2020-01-31    | 2020-01-31     | 2020-02-03 |
| Iteration 1         | 1 hour                | 2 hours                  | 2020-01-01    | 2020-01-01     | 2020-02-03 |
| Vision              | 2 hours               | 3 hours                  | 2020-01-31    | 2020-01-31     | 2020-02-03 |
| Code implementation | 45 minutes            | 1 hour                   | 2020-01-02    | 2020-01-02     | 2020-02-03 |

## 5.2 Iteration 2

In the second iteration, a significant feature has been added to the game. As usual, the player will be welcomed by a welcome message. Then the user should state if he/she is ready to begin the game. If the player choice is yes, the second message will be displayed which will ask the user to choose a level between medium and hard. The difference between moderate and hard is that in the moderate the player has 8 attempts to choose the missing letters. However, only 6 attempts will be given to the user in the hard mode. Then, the user will begin the game. After finishing the game, the player will be directed to a stage where the result of the game will be displayed. Finally, the user will be directed to a stage where the game will give the user two options to choose from. Consequently, two options namely restart, and exit will emerge. If the user decides to play again, by choosing the restart option, the player will be directed to the same stage (if the user is ready to continue or not)

to repeat the same procedure. However, after winning the game, the final message will show up where it will ask the user to fill out his/her personal information along with a survey that asks for user feedback. After going through all these stages, the program will terminate.

### 5.3 Iteration 3

Creation of test cases that can help the developer to evaluate the game.

### 5.4 Iteration 4

The game will be ready to hand in to the client with all necessary implementations.

## 6. Risk Analysis

Logically, every project may face some risks. Therefore, a developer should have a backup plan or be prepared to overcome those risks. Due to the importance of this matter, there is a chapter called project management (Chapter 22), that introduce some chances to reduce and avoid the risks. In order, to minimize the risk of the project, there are some steps that project manager should consider to be ready, avoid and reduce the impact of the risk.

### 6.1 List of risks

It is very important to identify the possible threats to reduce their impacts on the project. There are some prominent threats to the project:

- **Lack of time** (since the time is limited, it can result in some errors).
- **Lack of human resource** (Only one developer is responsible for code implementation).
- **Lack of knowledge** (It can be challenging for a developer who is the first time).

- **System failure** (There may be a possibility that the system crash in one stage).
- **Delay in delivery** (The project may not be ready by the deadline because of loads of works).

## 6.2 Strategies

There are some approaches that can reduce the chance of project risk that are:

- **Using popular libraries and websites** (e.g. stack overflow is popular community of developers where a developer can consult with others and get help).
- **Using previous knowledge** which can result in acceleration of the process and planning of the project.
- **Having a backup plan** (e.g. uploading the files in google one drive) can help the developer for retrieving the files.
- **Not developing a difficult project** which eventually may increase the chance of risks.

### Personal Reflection:

Since I consider my code simple and it does not contain many features, the chance of risks will eventually reduce. However, working on complicated projects and implementing hard and extra features in the game will result in the creation of more risks.

I have decided to mention the above risks and strategies because there are very common to happen during my project especially the strategies. Also, the given risks may happen in my project due to my lack of knowledge in programming and planning.

## 7. Time log

| Assignment  | Actual time dedicated | Assumption             | Overview  | Deadline   |
|-------------|-----------------------|------------------------|---|------------|
| Iteration 1 | 3 days                | 4 days (for finishing) | A very first plan (skeleton) for the code structure and an overview of the first steps of a game was created. | 2020-02-03 |
| Iteration 2 |                       |                        |   |            |
| Iteration 3 |                       |                        |   |            |
| Iteration 4 |                       |                        |   |            |

## 8. Handing in

All assignments have a number of files to hand in. The overall advice is to *keep it simple*. Make it easy for the receiver to understand what the files are by using *descriptive* file names. Use as *few* separate documents as possible. Always provide a *context*, that is *do not* send a number of diagrams in “graphics format”, but always in a document where you provide the purpose and meaning of the diagrams. Remember that the “receiver” is in reality a customer and as such has very little knowledge of the diagrams and documents – always provide context that make anything you hand in understandable to a non-technical person.

To hand in an assignment, make a git release and hand in the link via Moodle to that release.