SOFTWARE DEVELOPMENT PROJECT TEMPLATE

**Donald Appiah**

2019-02-06

page1image24045504

Optional logo

Logo

**Contents**

**Revision History………………………………………………………………………… 2**

**General Information………………………………………………………………………… 3**

**Vision ………………………………………………………………………….4**

**Project Plan 5**

4.1 Introduction ........................... 5

4.2 Justification ........................... 5

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

**5 Iterations 6**

5.1 Iteration1 ............................ 6

5.2 Iteration2 ............................ 6

5.3 Iteration3 ............................ 6

5.4 Iteration4 ............................ 6

**6 Risk Analysis 7**

6.1 List of risks ............................ 7

6.2 Strategies............................. 7

**7 Time log**

**8 Handing in**

**8 9**

Project Name – Version – Author – Da

**2 General Information**

|  |  |
| --- | --- |
| Project Summary | |
| Project Name: Legend Hangman | Project ID: L5111\_v1.0 |
|  |  |
| Project Manager: Donald Appiah | Main Client: lmd |
|  |  |
| Key Stakeholders | |
|  | |
| Executive Summary | |
|  | |

Project Name – Version – Author – Date

**3 Vision**

The game Hangman will be a word guessing game which will consist mostly familiar words and easy to guess. Players try to figure out an unknown word by guessing letters.

The one playing guess an alphabet in a word correctly the player will continue till the word is guess correctly and the player wins, otherwise if an alphabet is guessed wrongly a man will be hanged as the name predicts, the player will have a maximum of certain chances to be hanged by adding diagram each time an alphabet is guessed wrong.

Project Name – Version – Author – Date

**4 Project Plan**

The project plan of the application will first start with planning on what will be needed for the project, like resources, hardware and software requirements how the project should be like, the risk that might pop up while the project is in progress. Next stage will be about how I will model my application how it is going to look like and function and also designing the application with codes and algorithm. Lastly will be the application will be the testing of the application to see if everything is functioning as it should be else must be fixed.

**4.1 Introduction**

My application process and planning.

* The development is a simple and traditional hangman game. Implemented in a text form with java code language.

**4.2 Justification**

The application should be made to meet the requirements of the users.

**4.3 Stakeholders**

A project manager – Donald Appiah.

Users – this includes the individuals going to make use of the project’s product or outcome,

**4.4 Resources**

Resources for the application will be my computer, books and IDE software to run my application

**4.5 Hard- and Software Requirements**

My requirement will be a functioning computer and eclipse to run my codes.

**4.6 Overall Project Schedule**

|  |  |
| --- | --- |
| **Activity** |  |
| Process and planning | February 2019 |
| Modelling | February 2019 |
| Design | February 2019 |
| Testing | March 2019 |

**4.7 Scope, Constraints and Assumptions**

The project will be a game application containing words to be guessed and the goal of the application is to improve brain activities how people think and how easily and difficult it is to guess a word.

Project Name – Version – Author – Date

page7image23705472

**5 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**

The first iteration is this project plan along with some degree of implementation. Complete the documentation first so that the implementation goals are met in code. You need to implement an idea and some skeleton code for your project to work with. This is assignment one.

**5.2 Iteration 2**

In this iteration you need to add some features to the game *but* after you have first modelled them using UML. All diagrams need to be included in the project documentation and should be implemented in the way modelled.

**5.3 Iteration 3**

You may include additional features to the game in this iteration, but the main focus is on *testing*. Plan, perform and document your tests in this iteration.

**5.4 Iteration 4**

The outcome of this iteration is *the complete* game. Reiterate the steps in iteration 1 – 3 for a set of new features but also remember to see the project as a whole, not only its parts.

Project Name – Version – Author – Date

page8image23661568

**6 Risk Analysis**

All projects face risks that make it important to prepare for what might hap- pen. 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.

|  |  |  |
| --- | --- | --- |
| **Phase** | **Risk Description** | **Scale** |
| Iteration #1 | Unable to connect to GitHub | moderate |
| Iteration #1 | Unable to complete iteration #1 | moderate |
| Iteration #2 |  |  |
| Iteration #2 |  |  |
| Iteration #2 |  |  |
| Iteration #3 |  |  |
| Iteration #3 |  |  |
| Iteration #3 |  |  |
| Iteration #4 |  |  |

**6.2 Strategies**

* Unable to connect to GitHub   
  *Solution: Check online YouTube and online websites for tutorials.*

Unable to finish coding.  
*Solution: Check on Stack Overflow and YouTube for ideas & implementation.*

Project Name – Version – Author – Date

page9image23720512

**7 Time log**

Each assignment must be accompanied with a time log. This time log should contain the date, time and task to be performed. The reason for doing this is for you to get some experience in estimating your own time – creating a time log is one of the best ways of doing this. Take into account the time for learning and understanding of the problem when you plan the time. Make your planning with 15 minutes as the minimum unit. In the time log you start by *planning* the amount of time you believe a task will take and after it is done you mark *the actual time*. If every entry that has a difference in planned and actual time spend, analyse the time difference.

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Time** | **Version** |
| 7-Feb-19 | Assignment 1 | 4hours 45mins | 1.0 |
| 22-Feb-19 | Assignment 2 | 3hours 30mins. | 1.0 |
|  | Assignment 3 |  |  |
|  | Assignment 4 |  |  |

Project Name – Version – Author – Date

page10image23758144

**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.

Project Name – Version – Author – Date