SOFTWARE DEVELOPMENT PROJECT TEMPLATE

IPEK UYANIK

18/04/2019

Hangman GAMES

Contents

1 | Revision History

Date	Version	Description	Author
07/02/2019	1.0	Iteration 1	Ipek Uyanik
20/02/2019	1.1	Iteration2 and updated time-log	Ipek Uyanik
04/04/2019	1.2	Iteration3 and updated time-log	Ipek Uyanik
10/04/2019	1.3	Updated iteration2	Ipek Uyanik
18/04/2019	1.4	Finished project	Ipek Uyanik

2 | General Information

Project Summary	
Project Name	Project ID
Hangman	iu222au_1dv600
Project Manager	Main Client
Ipek Uyanik	Primary school students who learn English
Key Stakeholders	
*Development team *Main client *Tester *End user	
Executive Summary	
The project is implementing Hangman game. The player tries to find the word by guessing the letters in a given time and with a certain life change. The player loses if he / she does not have the change to life or has expired. The aim of the application is entertaining to primary school age children and expanding their vocabulary while having fun.	

3 | Vision

The project is implementing a game called Hangman which is designed with GUI.

The main purpose of the project is to teach the primary school students the English words while playing. In addition, many words in the game allows students to learn different words every time they play. The player tries to find the given word by guessing the letters in a certain period of time and with certain estimation right. The game builds a part of the man getting hanged for every wrong guess. With register option, also the guest user can start the game without the registration. Players can see both their own score table and high score table. The game will have three difficulty levels. Short words are the difficult, and long words form the easy level. In all of these features, the game becomes more interesting and the player is more encouraged to play the game and learn vocabulary.

Users will have lots of fun while playing that game and having a hangman game that can be easily used by users.

REFLECTION

While i writing the vision for the hangman game, i could be more flexible or more creative. But i preferred being more concrete. Even those are can be a little bit much for me. Because this project will be my first game project and i am not sure that i am be able to do that much features for a a game. Nevertheless, even in thought, I did not refrain from mentioning the basic features that should be present in a game. I am strongly hoping that i will be able to implement that features to the game.

4 | Project Plan

4.1 Introduction

The project is implementing a game called Hangman which is designed with GUI. The game is that a player trying to guess a word using a strategy based solely on letter frequency. As the player continues, a part of the stick figure on the noose is added. Once a full body is drawn, the game is over, and the player loose.

4.2 Justification

As today's children are very interested in technology, it is the dream of all families and schools that they can learn while they have fun. This game provides students to get to know letters better. It also expands their vocabulary knowledge. And of course the greatest purpose is fun.

4.3 Stakeholders

Manager: Wants to see that the project is completed successfully and in a given time

Developer: Wants to well-structured code for better maintainability

Tester: Wants to test the written code correctly and also expected features is exist.

End-users: (Students-Children) want to an application that they can play with fun

4.4 Resources

Money: There is no money involved this project because it is a course project.

Time: 9 weeks, 20 hour/week

Software: Java-Eclipse-Github

Hardware: Macbook-air, 1,8 GHz Intel Core i5

Personal: Manager,developer,tester is the one and only employee in this project

4.5 Hard- and Software Requirements

Software: Eclipse 4.4 or above for IDE and Java 1.8 or above and JDK

Hardware: There is no spesific requirement for hardware except a pc.

4.6 Overall Project Schedule

Phase I: 08/02/2019

-Project plan, skeleton code, time log

Phase II: 22/02/2019

-Use case diagram, state machine , usa cases, class

diagram, source code, time log

Phase III: 08/03/2019

-Test plan, manual test cases, unit tests, time log

Phase IV: 22/03/2019

-Updated diagrams, finished game, final project documentation and time log

4.7 Scope, Constraints and Assumptions

Scope:

The project will introduce a game which is a basic Hangman game. The graphical user interface will be used in the game for the visual facility. The player will be able to login to game as a registered user (which is registration feature) or as a guest user without a registration need. The words will be in English and the length of words will change for every level. There will be three levels in the game; easy, medium, hard. The player will be able to choose a level before started to game. The game will play at a certain time. The player can continue to play the game with new words if the player guesses the word correctly until the time is up. There will be also a score feature. The player will be able to see his/her own score table or high score table which contains every user's score.

Out of Scope:

The game will not has either a multiplayer feature or ability to play on online. Also, there will not be that features; adding words or deleting words.

Constraints:

For the project, the main constraints are limited time; limited in 20 hours a week, lack of person; only one student works on the project and lack of experience; the student does not have so much experience in such like that kind of a game.

For the application, the main constraint is that users need a java IDE to run the programme.

Assumptions:

It’s assumed that the user has JDK his/her own computer and a Java IDE; Eclipse. The user has to download all files about the game from GitHub and open them as a new project and run the main class.

REFLECTION

Planning the project was little bit complicated. In the beginning, I did not understand exactly what i am suppose to do. It is not stil clear as i want but i think it because of content of the project. In my opinion this is a student project which includes implementing game and if we compare the other big projects we do not have the exact and extensive knowledge about the content. Despite of this i tried to rely on the content of the template and complete the documentation according to the request as much as i could.

5 | Iterations

5.1 Iteration 1

Due to-08/02/2019

In the first iteration, general information will be written, the vision will be created, the project will be planned under those subtitles; introduction, justification, stakeholders, resources, hard- and software requirements, overall project schedule, scope, constraints and assumptions. All iterations will be planned and determined, risks and strategies will be determined and time log will be written for each tasks and iterations. Also a skeloton code will be written.

Task	Estimated Time(hrs)
Write general information	0.5
Write vision	1
Write project plan	1
Determine the iterations	1
Write the risks and strategies	1
Create the time log	0.5
Write skeleton code	0.5

5.2 Iteration 2

Due to-21/02/2019

In the second iteration, that diagrams will be created for implementing the basic code: use case, fully dressed, state machine diagram and class diagram. According to those diagrams, some features will be added to the code. The documentation and the time log will be updated.

Task	Estimated Time(hrs)
Create use-case diagram	1
Create fully-dressed use-case	1
Create state machine diagram	1
Create class diagram	1
Add feature to the game	10
Update the time log	0.5
Update the documentation	0.5

5.3 Iteration 3

Due to-08/03/2019

In the third iteration, will be focused on testing. The test will be planned with those contents: objectives, what to test and how, time plan. Unit tests and manual test cases will be written for chosen methods. Tests will be performed and the test result will be written as a report. Als,o new features will be added to the game.

Task	Estimated Time(hrs)
Plan the test	0.5
Write the unit test	2
Write manual test cases	2
Perform the tests	20min
Write report for test results	1
Add new features to the game	10
Update the time log	0.5
Update the documentation	0.5

5.4 Iteration 4

Due to-08/03/2019

In the fourth iteration, 1-3 iterations will be reiterated, almost every step will be updated and improved. The game will be finished and the project will seen as a whole.

Task	Estimated Time(hrs)
Reiterate the iteration1	2
Reiterate the iteration2	2
Reiterate the iteration3	2
Finish the game	6
Complete the documentation	2
Update the time log	0.5
Prepare everything for hand-in	2

6 | Risk Analysis

6.1 List of risks

Risk	Probability	Impact
Software size is underestimated	High	Tolerable
Misunderstanding the requirements	Moderate	Tolerable
Testing time is underestimated	High	Serious
The product does not convenient for requirements when the deadline is passed.	Moderate	Catastrophic
Loosing the the whole project file.	Low	Catastrophic
Work environment or IDE brokes.	Low	Tolerable
Hardware breakdown	Low	Catastrophic
Lack of experience/knowlage	Low	Serious

6.2 Strategies

Risk	Strategy
Software size is underestimated	Plan the project carefully. Consider estimated time and actual
Misunderstanding the requirements	Good communication with costumer
Testing time is underestimated	Estimate time before started testing and give extra for estimated time.
The product does not convenient for requirements when the deadline is passed.	Make sure an enough estimation for each tasks in the iterations and keep in touch with customer.
Loosing the the whole project file.	Make sure store every file in the cloud or github
Work environment or IDE brokes.	Update the tool regularly to avoid to any crush
Hardware breakdown	Regular maintanence and backups for hardware
Lack of experience/knowlage	Before took the responsibility, should study and make sure learned every need.

REFLECTION

In the beginning, analysing the risks was little bir tricky because predicting the future is always hard. However, when I started to break down the whole system into components to analyze the risks it got a bit more predictable. Yet still, it was a hard task due to lack of experience. It was my first software project that I had to think about the risks and strategies. I have tried to come up as many risks I can and find strategies to prevent them.

7 | Time log

Task	Estimated Hrs.	Actual Hrs.
Write general information	0.5	1
Write vision	1	1
Write project plan	1	2
Determine the iterations	1	2
Write the risks and strategies	1	2
Create the time log for iter.1	0.5	15min
Write skeleton code	2	4
Create use-case diagram	1	2
Create fully-dressed use-case	1	**3**
Create state machine diagram	1	2
Create class diagram	1	0.5
Add features to the game for iter.2	10	**15**
Update the time log for iter.2	0.5	15min
Update the documentation for iter.2	0.5	15min
Plan the test	0.5	1
Write the unit test	2	3
Write manual test cases	2	3
Perform the tests	20min	0.5
Write report for test results	1	2
Add new features to the game	10	11
Update the time log for iter.3	0.5	15min
Update the documentation for iter.3	0.5	0.5
Reiterate the iteration1 -Update every step	2	3
Reiterate the iteration2 -Update every step	2	3
Reiterate the iteration3 -Update every step	2	3
Finish the game	6	**1**
Complete the documentation	2	3

Update the time log for iter.4	0.5	0.5
Prepare everything for hand-in	2	2

REFLECTION

Time estimation was an easy task to do when we compare with the other tasks. But at some points, i was wrong. Like implementing new features took more time in every step. Because it is hard to predict a time for a feature which is added later. I did not add the "understanding the task" time in the time log hence every task's time took 1 hour longer in general.

8 | Handing in

*** https://github.com/ipekuyanik/iu222au_1dv600

REFLECTION

Because I have already finished my code in iteration3, I have not added any new functionality to the source code. I have revisited and revised all my fully-dressed use cases and whole documentation which contains general information, vision, project plan, description of iterations, risks and strategies and time-log since I misunderstood them. I have improved my diagrams. So overall, involving in this kind of a project was new for me but I feel that I learned a lots of valuable information about how a project is documented and created from scratch. I am hoping that this experience of a real project will help me in my future projects both as student and as a developer.