HSFC

CO3 – Programing Project.

‘Fireboy and Watergirl’

Rose Addison

Table of Contents

[Analysis of the problem 3](#_Toc10621059)

[Problem Identification-Description 3](#_Toc10621060)

[Problem Identification – Computational methods 4](#_Toc10621061)

[Stakeholders- identification and descripition 5](#_Toc10621062)

[Research – Identifying similar problems 6](#_Toc10621063)

[Web based solution 6](#_Toc10621064)

[Questions for stakeholders:- 7](#_Toc10621065)

[Overall view on TWC compared to FB and WG 9](#_Toc10621066)

[Stakeholders overview/conclusions in response to the questions 9](#_Toc10621067)

[Stakeholders consultation 10](#_Toc10621068)

[Success Criteria 11](#_Toc10621069)

[SC1 11](#_Toc10621070)

[SC2 11](#_Toc10621071)

[Hardware and network configuration 12](#_Toc10621072)

[Hardware requirements 12](#_Toc10621073)

[Software requirements 12](#_Toc10621074)

# Analysis of the problem

## Problem Identification-Description

The project I attend to code is ‘Fireboy and Watergirl’. It’s a 2d platform maze game where two characters, fireboy and watergirl controlled by the keyboard functions AWD and ←↑→. They have to move up the platforms on the board, collecting gems to pass into the next level. They characters have to jump over or avoid there corresponding ‘ponds’, fireboy can’t go in the water pond and watergirl can’t go in the fire pits, otherwise the level restarts and the player has to start again. The game ends when the character reaches a door which opens to take the character on to the next level.

The original game ‘Fireboy and Watergirl: The Forest Temple’ was released in 2007

My target audience for my game for people who like using logic and solutions to solve problems so in the age range of teens from 10 to 18, for both sexes. However can be played by anyone with an interest because it doesn’t have any themes that are discriminate themes, which would exclude sexes or ages. But will not be recommended for children below the ages of 7. And because the game doesn’t have any violent themes or graphic images this widens the audience for the game.

The platform that the game is on is the computer. A key requirement for the game is for the characters to be able to go up and down the platforms collecting gems and jumping over their retrospective ‘ponds’ and die if they touch them, restarting the game.

The features that will be required in my program that will need to be solved: A user controlled character who can move up the platforms up and down and jump in either direction. This is needed because the character needs to be able to move around the ‘maze’ collecting the gems to solve the level(s). Another entity that is needed is to have the ‘ponds’ which when the retrospective characters go in they die but when that character goes in their own ‘pond’ they don’t die because they are made of the same ‘stuff’ so it doesn’t kill them. There also needs to be gems to collect, gems that both characters can collect but then also gems that only one specific character can collect to move on to the next level.

In my project the

Ill discuss with my stakeholders what other features are necessities for them and what details to leave out for a more rounded game.

## Problem Identification – Computational methods

There are five computational methods that apply to this course.

1. Thinking abstractedly [and visualisation] (a) the nature of abstraction. (b) The need for abstraction. (c) The differences between an abstraction and reality. (d) Devise an abstract model for a variety of situations.

2. Thinking ahead a) Identify the inputs and outputs for a given situation. (b) Determine the preconditions for devising a solution to a problem. (c) The nature, benefits and drawbacks of caching. (d) The need for reusable program components.

3. Thinking procedurally [and ‘decomposition’] (a) Identify the components of a problem. (b) Identify the components of a solution to a problem. (c) Determine the order of the steps needed to solve a problem. (d) Identify sub-procedures necessary to solve a problem.

4. Thinking logically (a) Identify the points in a solution where a decision has to be taken. (b) Determine the logical conditions that affect the outcome of a decision. (c) Determine how decisions affect flow through a program.

5. Thinking concurrently (a) Determine the parts of a problem that can be tackled at the same time. (b) Outline the benefits and trade-offs that might result from concurrent processing in a particular situation.

The actual thing

The first computational method is thinking abstractedly and visualisation. This applies to the game because abstraction is when you remove attributes of the entity i.e. simplifying the game from reality. This works with my game because it extremely hard to mimic the game as though in real life it is unrealistic to put that much details into my project, because apart from being out of my skill range would use large quantities of memory and resources and would be unnecessary. In my game there are some elements which will be simplified. These include: simplifying the visual graphics so features like the background (might not be the best one to do) if it had too much details it would distract from the characters and other parts of the game and would look bad. Also the characters will be visually simple to stop from distracting from the game.

The second computational method is thinking ahead. This consists of considering data or inputs which are required form this program to work: the input of the arrow keys will be used for the characters to move around the 2D maze. (There will be checks to see if the characters go into the wrong ponds which will kill them and the characters will fail the level. This is shit)

The third computational method is thinking procedurally or “decomposition” this is when a problem is broken down into its component parts. This can be done in my program through the character which will be controlled through the keyboard and they will have its own characteristic s such as the jumpSpeed and force as integers. And jumping which will be a Boolean. They will also have gravity so they will fall at even intervals.

The fourth computational method is thinking logically. This is when you identify the points in a program where a decision has to be taken. And then determine the logical conditions that affect the outcome of a decision. And finally determine how the decisions affect the overall flow through the program. This applies to my program an example is when the decision when the time that the player took to finish and the amount of gems that where collected, determines whether or not that the user passes on to the next level or has to retake to re-try and unlock the next level.

Finally the fifth computational method is thinking concurrently. This determine the parts of a problem that can be tackled at the same time. Because I have to character that have to be controlled by different keyboard sets of keys, so the characters have to move individually to each other. On top of that they will have to work together to collect the gems and beat the time frame to pass to the next level. There will be some gems that only one character will be able to collect. And finally the ‘ponds’ that one of the characters can pass through and the other cant i.e. watergirl can pass through the water ‘pond’ but if the fireboy touches that ‘pond’ the game ends and has to restart from the beginning of the level.

## Stakeholders- identification and descripition

My stakeholder are Seb Roffey, a 17 year old male who has had a lot of previous experience playing computer games. And my other stakeholder is my mum Ann Addison, a 60 year old woman who has very little experience playing computer games prior to the project. My proposed project age range is aiming for ages 10+, with neutral gender. Seb fits the criteria by being interested in games which focus on solving problems and using logic to win. Ann fits the criteria because she is looking for an insight into computer games because she is a therapist for children and is looking for a relatively easy gam which doesn’t rely on quick reflexes and has simple controls and not a complex ‘storyline’. And also because the game doesn’t have any violent themes it fits her needs perfectly. My stakeholders are different genders and ages which helps me get a range of opinions of my game and needs. Seb is interested in the game because he is looking for a game which he can play with his younger brother of age 11, which they can work as a team. Ann is interested because she is looking for a interesting introductory into gaming, which she can play with her clients, and the game takes her though the steps clearly.

Seb will be helpful in helping me achieve a game which has interesting features and is fun to play and challenging enough and keeps the players interested throughout the game and making sure it targets the correct age criteria. Ann because of her limited experience of gaming can point out to me when the game isn’t making sense, where I’ve skipped over because I had thought it was obvious, she will keep the game simple and clear for people who haven’t got experience in gaming. Seb will make use of my program by playing it with his younger brother who will share it with his friends. And Ann will make use of my game through learning how to play computer games with her clients and introducing them to games which doesn’t have a violent them which instead makes the user think though the game and use their brain to win and teamwork to win. Which will help them in building relationships thought something they can relate to and enjoy.

I attend to use my stakeholder’s through each stage of the project and use their feedback to improve the design and the way the game works.

## Research – Identifying similar problems

### Web based solution

On the internet I found similar web based solutions to ‘fireboy and watergirl’ which have the same sort of layout and aims such as they both involve collecting gems to win points and are 2D maze games. ‘Twin Cat Warrior’ GUI looks as follows for the first level:-



In the game the each character uses different sets of controls to move the character around the red ‘cat warrior’ uses AWD (to go left, jump/up, and right retrospectively). And the blue character uses ←↑→ with the same operations as the red character. The aim of the game is to collect the all the gems which are recorded in the score in the top right hand corner. The characters have to work together to go up the levels and overcome the obstacles to reach the checked flag to go on to the next level.

## Questions for stakeholders:-

#### Q1 Do you like the GUI of TWC which is simplified or the GUI (above) of FireBoy and WaterGirl (below)?



Seb: I prefer the original games graphics

Ann: I like the TWC game because its clearer and less dark

#### Q2 How did you find the controls and instructions in TWC?

Seb: I didn’t know which set of controls where for which character but I could test them out to find out.

Ann: I found it difficult because I’ve never player a computer game so I don’t know what to press to get the characters to move and I had to be told what to do. Instructions would have been helpful on the screen (like the first level in fireboy and watergirl) to start of the game, so I can get a hold of it.

#### Q3 Should there be music playing the background?

Seb: Yes

Ann: There should be an option to turn of the music because it is distracting when learning the game.

#### Q4 Do you think there should be a timer in the game at the end of the level should there be a grade based on the time it took to complete AND out of (no. of gems) per amount collected OR just a ‘you passed the level if you collect the gems in time?

Seb: I think there should be a timer and the game needs to be completed in a time frame for each level and you need to collect all the gems to move on to the next level.

Ann: A timer should be used because otherwise the players can complete the game in any time also all gems should be collected to move to the next level. So I think if that all of that is done the player can move to the next level, so no grade.

#### Q5 should there be moving platforms?

Seb: yes because it makes the game more interactive and makes the users work together to win the level.

Ann: I don’t think it is a vital part of the game however it makes the game more fun to play.

#### Q6 Should there be boxes that need to be moved to help their characters boost themselves on to another platform?

Seb: Yes I think there should with the same reasoning as the moving platforms. However I found it annoying when both characters in TWC pushed in two boxes together they couldn’t move it.

Ann: Same answer as Q5

##### C:\Users\rsa8916\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\F147F748.tmp Q7 Do you think there should be a home page with all the levels on? i.e. in fireboy and watergirl where there are branches and open different paths or (2nd image) just sequentially like TWC?



Seb: the original fireboy and watergirl because if you get stuck you can go down another path.

Ann: the first image because there is more choice and I’m less likely to get stuck.

#### Q8 Should there be doors at the end of the game that each character needs to go to (like in fireboy and watergirl, first image) or just one finish spot for the characters to end (image 2 from TWC)?



Seb: I prefer the fireboy and watergirl grafics for the ending.

Ann: I prefer image one, it looks beter.

### Overall view on TWC compared to FB and WG

Seb: I love the GUI of FB and WG compared to the TWC. And there needs to interesting aspects to the game that the users need to

## Stakeholders overview/conclusions in response to the questions

For the background of my game and general GUI, because the stakeholders didn’t come to a conclusion, I’ve decided to take inspiration from the fireboy and watergirl original graphics. But however I will keep in mind the comment about the over colour scheme is dark so I’ll make it lighter and brighter so it is clearer for the users.

For the controls and instructions I’ll lay them out like the original FB and WG, so the instructions will appear in the first level and appear as the character progresses through the maze. And I’ll also be sure to make it clear which set of controls i.e. ←↑→ and AWD are clear which control which character.

With background music I’ll give the users the option weather there is background music playing. I’ll put it along with the instructions, a help option as such.

For the score my stakeholders wanted a timer for each level which would give the players a score, but they don’t need to have a certain time to go onto the next level. However the users have to collect all the gems to pass on to the next level. And if they don’t collect all the gems they can’t move on to the next level.

With the moving platforms and boxes I’ll add to make the game more interesting and complex so the users can work together to complete the level.

I’ll add a screen which will show the levels and I’ll create it as such that the levels are arranged in a spider web arrangement, like the FB and WB game.

For the ending of each level I’ll make it so that the GUI looks like FB and WG, so there will be two doors one for each character that will open and ‘lead’ the characters to the next level.

Notes// golden triangle put important info in top left hand corner.

## H:\My Documents\computer science\HSFCProjects\code\IMAGES\background sketch.pngStakeholders consultation

#### Above is a rough sketch that I’m planning of for my first level, is there improvements or recommendations that I should implement. This is just a mock-up of how the platforms will be located and positioned.

//https://docs.microsoft.com/en-us/visualstudio/productinfo/vs2017-system-requirements-vs

## Success Criteria

### SC1

Have two characters that are visually appealing which stand out on the background who have movements that are controlled by the keyboard and are moved individual to each other. They also die and win.

### SC2

Have an environment which will include obstacles platforms and gems.

## SC3

## 

## Hardware and network configuration

### Hardware requirements

I am using visual studio 2017 to write my program, so my project will need the following hardware to run my game:

* 1.6 GHz or faster processor
* 2 GB of RAM; 4GB of RAM recommended (2.5 GB minimum If running on a virtual machine)
* Hard disk space: up to 130 GB of available space, depending on features installed; typical installations require 20-50 GB of free space.
* Hard disk speed: to improve performance, install Windows and Visual Studio on a solid state drive (SSD).
* Video card that supports a minimum display resolution of 720p (1280 by 720); Visual Studio will work best at a resolution of WXGA (1366 by 768) or higher.
* Keyboard
* Mouse
* Monitor

### Software requirements

Visual studio 2017 will install and run on the 64-bit version of the following operating systems:

* Windows 10 version 1507 or higher: Home, Professional, and Enterprise
* Windows Server 2016: Standard and Datacenter
* Windows Server 2012 R2 (with [Update 2919355](https://support.microsoft.com/kb/2919355)): Essentials, Standard, Datacenter
* Windows 8.1 (with [Update 2919355](https://support.microsoft.com/kb/2919355)): Core, Professional, and Enterprise
* Windows Server 2012: Essentials, Standard, Datacenter
* Windows Server 2008 R2 SP1: Standard, Enterprise, Datacenter
* Windows 7 SP1 (with latest Windows Updates): Home Premium, Professional, Enterprise, Ultimate

The executable file will be a .Net version and the user will need to have the files that hold all the files that are needed to be in the project.