HSFC

CO3 – Programing Project.

‘Fireboy and Watergirl’

Rose Addison

Table of Contents

[Analysis of the problem 3](#_Toc9242524)

[Problem Identification-Description 3](#_Toc9242525)

[Problem Identification – Computational methods 4](#_Toc9242526)

[Stakeholders- identification and descripition 5](#_Toc9242527)

[Research – Identifying similar problems 6](#_Toc9242528)

[Web based solution 6](#_Toc9242529)

[Visual basic solution 6](#_Toc9242530)

[Research and stakeholder preferences 6](#_Toc9242531)

# 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.

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

## 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 arounds the red ‘cat warrior’ uses AWD( to go left, jump/up, and right retrospectively). And the blue character uses ←↑→ with the same operatins as the red character. Warrior cats is similar to fireboy and watergirl because of its genre

-The Prince and Princess Elope

-Get My Pill

-Snow Bros. (1990 arcade game)

Snow Bros. is a 1990 platform arcade game. The game supports up to 2 players, each taking control of one of the two snowmen. Each player can throw snow at the enemies until they are comply covered and turns into snowballs. An enemy partially covered in snow cannot move until it shakes it off.

Meh

### Visual basic solution

## Research and stakeholder preferences