OCR GCE A

COMPUTER SCIENCE PROJECT

H446-03

Name : Alexander Mills

Candidate Number : <INSERT CANDIDATE NUMBER>

<Institution Name> : <INSERT CENTRE NUMBER>

Title of Project : <INSERT PROJECT TITLE>

H446-03 – Project CONTENTS

Table of Contents

[A. Analysis 3](#_Toc88039851)

[Outline 3](#_Toc88039852)

[Stakeholders 3](#_Toc88039853)

[Game Research: Tetris 4](#_Toc88039854)

[Game research: Hue 7](#_Toc88039855)

[Survey 8](#_Toc88039856)

[Interview 8](#_Toc88039857)

[Limitations and Scope 8](#_Toc88039858)

[Why this Solution is Suited to a Computation Solution 9](#_Toc88039859)

[Abstraction 9](#_Toc88039860)

[Thinking Ahead 9](#_Toc88039861)

[Thinking Procedurally 9](#_Toc88039862)

[Thinking Logically 9](#_Toc88039863)

[Thinking Concurrently 9](#_Toc88039864)

[Proposed Feature List 9](#_Toc88039865)

[Success Criteria 9](#_Toc88039866)

[B. Design 9](#_Toc88039867)

[Systems diagram 9](#_Toc88039868)

[C. Developing the coded solution (“The development story”) 10](#_Toc88039869)

[D. Evaluation 10](#_Toc88039870)

[Project Appendixes 11](#_Toc88039871)

# A. Analysis

## Outline

There is great academic pressure on students to perform to the best of their ability. To achieve this, students must study for longer, increasing stress levels and generating concern about whether time is being used effectively. There is a subsequent reduction in time spent on activities that don’t tangibly benefit academic performance like gaming and other recreation. This has an adverse effect on mental health as it sets up a poor work life-balance and means there is no opportunity to de-stress, creating an unstainable feedback loop which will hinder long term attainment.

To rectify this, I shall develop a game which 2d top-down tile game that heavily focusses on puzzle solving and systematic thinking. This will allow students to practice their problem solving and logical reasoning skills in a relaxed, enjoyable, and interactive game environment. This allows them to decompress, improving work-life balance due to a more sustainable method of practicing cognitive skills than studying. To successfully develop this solution, I will draw inspiration from other puzzle solving games such as Retro classics like Tetris(1984) and more modern examples like Portal(2007) and Hue(2016). This will allow me to evaluate existing solutions within this genre and which features are needed to ensure the game holds up to the stakeholders’ expectations and meets their needs.

## Stakeholders

The target demographic of the game will be students in the age range of 15 to 18 who enjoy regular problem solving and logical thinking. This demographic covers a wide range of abilities; therefore, the game must have an array of tiered difficulty levels to ease beginners into the game while allowing advanced players to still enjoy it.

It is designed to be played after a study session to unwind, so the user will likely have a computer available, on which they play the game. This means the game doesn’t need to be portable, so will be controlled by mouse and keyboard. As the game will be used to unwind and relax, it will have a simple, easy to understand control scheme; this will make it easier to learn and less taxing to use. To ensure that it is accessible to as many as possible, there will be very minimal text, having a symbol focused UI to overcome language barriers. The colour pallet of the game will use colours which are not too bright and have minimal blue; this will ensure it is pleasant on the eyes and not alarming, allowing the user to relax.

I have selected Benjamin Dodwell and Mate Fehevari to represent the target demographic. They are both 17 year old students who play videogames regularly. Their experience with similar games will allow them to give clear and well-judged feedback on my game, and how it compares to similar ones in the industry, allowing me to ensure my game meets the target demographics’ needs effectively. They are also close contacts, so I will be able to regularly receive incremental feedback throughout the development process.

## Game Research: Tetris

Tetris is a 2d puzzle game where the player stacks blocks on a 10x20 grid. The square blocks come in groups of 4 called “tetrominos”, which can have many different shapes. They fall to the bottom of the board, and then stop falling, landing on top of any blocks that had previously fell. Should a full row be completed when the falling blocks are placed, this row is cleared, scoring the player some points. This makes for an engaging game where the player must organise a random stream of shapes into a compact pile at the bottom of the board, figuring out which shapes fit where to keep the board organised.

The game starts slowly, with the blocks falling slower. This allows inexperienced players to get used to the game mechanics . As more rows are cleared and more points are scored, the pieces fall faster, allowing the player less time to decide where to place the piece. This makes the game much more stressful and difficult for all but the most experienced players as even a small error can cause big problems, causing the blocks to pile up towards the top of the board, at which point the game is over.

To incentivise more advanced strategies, the game rewards clearing multiple lines at once, rewarding the user with more points. If they clear 4 lines in one go (the maximum possible), they score 8 times as many points as a single line. This leads to players risking building up larger piles so that they can clear more rows at once, earning more points more quickly.

Main menu: Graphical user interface

Description automatically generated

The game’s main menu is the first thing that a potential player sees, therefore it is designed to introduce the players to the game, setting the colour scheme, theme, and branding. To help new players learn the game, there is a question mark button, which shows the controls, how to play the game and the language used to describe gameplay. My menu should contain all these features to make it usable an engaging.

The same UI “windows” are used in both the menu and the actual game. Hence the start menu has features that are blanked out, such as the “NEXT” and “HOLD” queues, which could be distracting or confusing for a new user. It also makes the UI over-crowed, so I will in my game I won’t be re-using UI elements to reduce clutter.

Gameplay:

A picture containing text, parking

Description automatically generated

The main game screen reuses the elements of the menu, so is familiar, though now all the elements are used. The bright colours on a dark background makes the game easier to look at, as well as distinguishing the individual sprites in the game and drawing the user’s attention to the important features. The indicator of where the blocks will fall makes it easier for the user to see what the game will do next(where the block will land), reducing the chance of the user placing a block in the wrong place – this makes the game less annoying and therefore more enjoyable for the user; my game must also focus on this to meet the user’s needs.

Pause Menu:

Graphical user interface

Description automatically generated

The pause menu allows the user to stop the game and return to it later. This makes the game more convenient to play as the user can pick it up and put it down as they want. This will be less important in my game as each level will be played all in one session, though it will still be needed. The menu also offers a tutorial section for teaching inexperienced users and an options menu to allow the user to configure the game to their play style. My game should also have ample configurability to allow the user to have a comfortable gaming experience.

## Game research: Hue

Hue is a puzzle-based side scrolling adventure game with the goal of exploring the map and progressing the story line. The core game mechanic is that the player can change the colour of the background, making game objects of the same colour disappear, allowing the player to pass through them. With multiple colours, the puzzles become very intricate, requiring the player to carefully develop a strategy to deal with each new level, skilfully timing the switch between colours to avoid coloured hazards, move game objects around each other and traverse the coloured platforms to the exit. This mechanic makes for a more enjoyable and rewarding experience for the user as they must reason through how to make every move, and therefore I will implement a similar system for my game.

The game also makes strong use of a storyline developed by both narration and dialogue boxes from NPCs. The narration is triggered by the player finding letters, which are placed in longer, labyrinth style levels which are less challenging, allowing the player to absorb the story. The storyline adds depth and reason to the game, giving the player a reason to progress to the next area to further understand the situation. This makes for a more immersive and engaging gaming experience, though a good story takes time to be written and will need narration, meaning this is out of the scope of my game.

Typical level:

Graphical user interface, diagram, schematic

Description automatically generated

The colour scheme of the game is very focused around the 8 colours of the colour wheel, so they are a repeating theme throughout the whole game. The key game objects are in bright colours, which is both for the functionality and to highlight them to the player. The monochrome background complements the colours and is easy on the eyes, making it easier for the player to look at as it makes no use of bright or startling colours. I will make use of a similar colour scheme for my game, as it will make my game more relaxing to play, while still having visual interest.

The level design makes use of hazards, which the player must avoid by making use of the colour changing mechanic. These force the player to carefully time their inputs, making the game more challenging. The level also has multiple objectives: the player must acquire a key first before passing through the exit This again facilitates more advanced puzzles. To make my puzzle game equally fun, I should incorporate all these level design queues. Each level has been manually designed, making them detailed, though I don’t have time to design levels to this degree, so mine will have to be procedurally generated.

Pause Menu:

A picture containing timeline

Description automatically generated

The pause menu allows the user to pause the game, allowing them to return to it later. It also provides some configuration menus for the user to tailor their experience to their needs. This includes a controls menu, where the user can learn the controls or configure them, a video menu where the user can configure the display resolution and full screen. It also has a colour-blind accessibility option, which is important as being able to distinguish colours is critical to the game, ensuring the game can be played by all potential stakeholders. The audio menu allows the user to control the volumes of different aspects of the game to their liking. These are all quality-of-life features, which enhance the rest of the user experience, and therefore will need to be a part of my game if it is to be enjoyable to play.

## Game Research: World’s Hardest Game

World’s hardest game is a puzzle game where the player must navigate through mazes to the exit, collecting objectives before exiting. The mazes are 2d and are viewed from top down, so the player can immediately see all parts of the maze. This means that the player can heavily strategise how they are going to proceed through the level, but there is nothing to explore.

The core mechanic that makes the game much harder is the hazards moving about the maze. If the player touches one, they instantly die and return to the nearest checkpoint. They all follow pre-defined paths around the level but most move very quickly. The levels are designed such that all places in the maze baring a few have hazards moving over them, meaning the player must keep moving to stay alive, and as they are so close together, the player must perfectly time their inputs to move between them without hitting them, making the game very difficult. While this makes the game fun, it is also very stressful, something I want to avoid, so in my game there will be vastly fewer hazards and if they move, they will be much slower.

Typical level:

Diagram, schematic

Description automatically generated

The levels are all manually designed and have a standard structure: the checkpoints are green areas, the hazards are blue circles, objectives are yellow circles, and the player is a red square. This means the player knows exactly what they are doing each level, making the game intuitive to play. The maze has a checkerboard floor which clearly shows the game is tile based, allowing the player to judge the position and motion of the hazards. Manual layout makes for some clever and challenging level designs, though time must be invested to compose all the levels. As my game will need many levels, it will have to be procedural, but this will work well as it can generate a standardised colour scheme.

## Survey

To gauge the needs of a larger group of potential stakeholders, I will use a survey to collect their opinion on how features of the gamer will be designed. This will allow me to make informed decisions about how the game should look and feel to play.

|  |  |  |
| --- | --- | --- |
| Question | Input type | Function |
|  |  |  |
| How important are graphics to make a puzzle game enjoyable? | Slider: 1 to 10  Comments box | Gauges how much work must be put into graphics to meet user needs |
| How much control over graphics is needed in the settings? | Multi choice:   * No options * basic options: resolutions, vsync, fullscreen * advanced: frame rate, rendering settings, toggleable visuals * extensive: full colour scheme configurability, all rendering settings | Allows me to develop a suitable graphics menu to make the game accessible for all users |
| How important are visual effects and animations to make a puzzle game enjoyable? | Slider 1 to 10  Comments box | Gauges how much work needs to be put into visual effects and animations |
| How important are Sound effects to make a puzzle game enjoyable? | Slider 1 to 10  Comments box | Gauges how much work needs to be put into the game’s sound design |
| How much control over sound is needed in the settings? | Multi choice:   * no options * a slider for game volume, and a slider for music volume * all game sounds have individual siders | Allows me to design suitable sound settings that will allow users to configure their game to their interests |
| How important is Background music to make a puzzle game enjoyable? | Slider 1 to 10  Comments box | Gauges how important background music is for the users to enjoy the game |
| How long would you want to spend per level when playing a puzzle game? | Numerical input in minutes  Comments box | Allows me to tune the level length so the game can be challenging for users but not enduring |
| How many times would you want to die on a level before completing it? | Numerical input  Comments box | Allows me to adjust how many hazards there are in a level |
| Should the levels contain checkpoints? | Boolean  Comments box | Determines if users want checkpoints or not, and thus determines if I will implement them |
| How should the game be titled? | Multi choice:   * based on visual theme * based on the style of puzzles * based on a story | Ensures that the title of the game conveys the theme and style of game to potential players well |
| Are there any other features which you would like to see in a puzzle game? | Comments box | Allows any other responses from the users, so they can input any other features they would like to see in the game |

## Interview

### Limitations and Scope

## Why this Solution is Suited to a Computation Solution

### Abstraction

### Thinking Ahead

### Thinking Procedurally

### Thinking Logically

### Thinking Concurrently

## Proposed Feature List

|  |  |
| --- | --- |
| Feature | Function |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Success Criteria

# B. Design

<See H446-03 Project Advice Booklet for help and guidance of what must go here.>

## Systems diagram

# C. Developing the coded solution (“The development story”)

<See H446-03 Project Advice Booklet for help and guidance of what must go here.>

# D. Evaluation

<See H446-03 Project Advice Booklet for help and guidance of what must go here.>

# Project Appendixes

Insert as many project appendixes as you need for your project.

These might include, but are not limited to:

* Complete Code Listing (ESSENTIAL)
* Interview Transcripts
* Meeting notes
* Observation notes or questionnaires