A'yaan Abdul-Mughis

Dulwich College | 2025

Trains

OCR A Level Computer Science Component 3

Table of Contents

[Analysis 3](#_Toc170124100)

[Problem Identification 3](#_Toc170124101)

[Stakeholders 3](#_Toc170124102)

[S Interview 3](#_Toc170124103)

[Form 1 4](#_Toc170124104)

[Justification 5](#_Toc170124105)

[Requirements 5](#_Toc170124106)

[Must 5](#_Toc170124107)

[Should 6](#_Toc170124108)

[Could 6](#_Toc170124109)

[Won’t 6](#_Toc170124110)

[Research 6](#_Toc170124111)

[OpenTTD 7](#_Toc170124112)

[YouTube Videos 8](#_Toc170124113)

[Python Games 9](#_Toc170124114)

[Themes 10](#_Toc170124115)

[Solution 10](#_Toc170124116)

[Potential Design Ideas 10](#_Toc170124117)

[Design 11](#_Toc170124118)

[Interface 11](#_Toc170124119)

[Data Structures 11](#_Toc170124120)

[Algorithms 11](#_Toc170124121)

[Decomposition 11](#_Toc170124122)

[Solution 11](#_Toc170124123)

[Testing Approach 11](#_Toc170124124)

[Implementation 12](#_Toc170124125)

[Iterative Process & Comments 12](#_Toc170124126)

[Iteration 1 12](#_Toc170124127)

[Error 1 12](#_Toc170124128)

[Iteration 2 12](#_Toc170124129)

[Iteration 3 12](#_Toc170124130)

[Testing 13](#_Toc170124131)

[Executed Test Plan 13](#_Toc170124132)

[Evaluation 14](#_Toc170124133)

[Testing 14](#_Toc170124134)

[Solution 14](#_Toc170124135)

[Product 14](#_Toc170124136)

[Maintenance & Development 14](#_Toc170124137)

Analysis

# Problem Identification

There is an innumerable selection of games available for teenagers to play.

I know from playing some of the most popular of these that they are usually very visually stimulating and addictive. Though I do not necessarily enjoy playing the most popular games, (notably first person shooters or action-adventure games) I do enjoy the puzzle and problem solving aspect of other games such as sudoku. It would make sense, therefore, that I develop a game which suits this preference, and find suitable stakeholders who agree too—motivation is a huge factor to consider when developing this project.

It’s highly unlikely that I am the only one who shares these views towards games

Without a doubt, the most obvious project that suits the brief would be a game.

# Stakeholders

Finding a stakeholder for my game was relatively simple. One of my peers (S) plays many many strategy and puzzler games and they have agreed to be the stakeholder for my project. He is 18 years of age and considers himself to be an experienced gamer.

This game will be developed for casual gamers, like S, who enjoy problem solving.

## S Interview

To begin with, I asked S a series of questions to gauge his interest and develop an initial idea of the requirements.

**What, in your opinion, makes a game interesting?**

A unique twist—a game has to have something to make it truly stand out. So many game concepts have already been made a good game needs more than just to be fun, but also to give the audience an experience they would find nowhere else.

**What is your favourite problem-solving-oriented game, and why?**

Personally it's Chants of Sennaar, it is a puzzle game structured around the way written language is formed. It gets a little boring towards the end as it tries to oversimplify more complex grammatical techniques, but it is overall an ingenious way of forcing the player to think outside the box and use knowledge of language to solve traditional puzzles.

**What makes a game stand out for you?**

The two things that make a game stand out to me the most at a first glance are the art style and the primary mechanic. If both look interesting the game is bound to be interesting.

**What does gaming mean to you? Why do you game?**

I usually only game in my spare time to relax, meaning I am not usually as interested in competitive games and rather a more casual playstyle. I am also a big fan of games that make the player feel powerful while still being challenging. These are a great way to let loose while still having to prove my skill.

**What’s the main emotion you feel after you have finished gaming—is it mainly enjoyment or sense of achievement?**

What I usually look for after I have finished gaming is a sense of accomplishment, especially when they get recorded in the form of achievements or progression bars.

As well as the initial interview, S and I will have regular weekly meetings to update him on the progress of the project. His feedback will inform successive stages of development, and this ensures that all work carried out it in line with the requirements.

## Form 1

In early 2024, I sent out a form to gauge the interests of some other potential stakeholders to help inform my gaming decisions and narrow down my choices.

I asked the following questions:

1. What type of game would be most appealing to you?
2. What games do you play?
   * If a stakeholder answered with games I hadn’t played before, I then disregarded their other answers
3. How do you like games to make you feel?
4. What is your preferred gaming platform?

The results were inconclusive, but I did find that the majority of people answered that the game should be:

* Exciting and mentally challenging (Figure 1)
* Get progressively harder (Figure 2)
* Based on strategy (Figure 2)
* Should be on desktop (Figure 3)

Forms response chart. Question title: How would you like my game to make you feel?
Select as many options as you wish.
. Number of responses: 6 responses.

Figure 1

Forms response chart. Question title: What type of game would be most appealing to you?
Select as many options as you wish. Feel free to write anything else in the Other area.
. Number of responses: 6 responses.

Figure 2

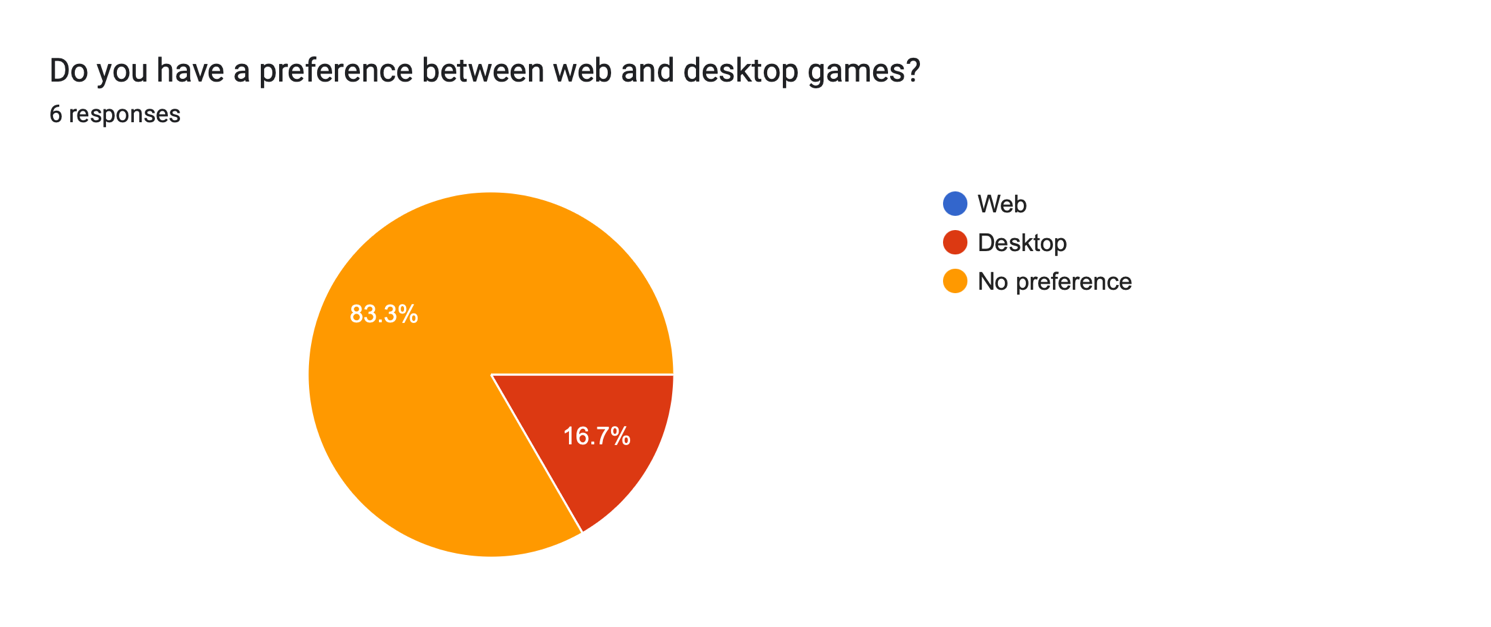


Figure 3

## Justification

As shown above, the data shows S, as well as the other potential stakeholders, would prefer a game of strategy, which aligns well with something I would enjoy designing and implementing.

It is important that I am motivated to complete the project, and the process is enjoyable.

## Requirements

### Must

1. Function offline
2. Use object-oriented programming concepts
3. Be an arcade game
4. Have an original and recognisable name
5. Run on any computer with a Python interpreter installed
6. Be controllable from a keyboard and mouse
7. Have a graphical user interface

### Should

1. Be written in Python
2. Use the Pygame library for the interface and game loop
3. Use comments for the ease of myself and others who may wish to view the code
4. Be easy to understand and to play
5. Have a 800 lines of code, including comments
6. Have a main menu
7. Have multiple levels with increasing difficulty
8. Be humorous with intelligent puns
9. Have a scoring mechanism

### Could

1. Have sound effects
2. Use text files to save and load game progress
3. Have paid and professional game art
4. Have a parallax effect
5. Have a parental controls feature that limits time on the game
6. Have an elaborate colour scheme which matches the game story
7. Have a start screen the first time the game is opened to explain the game
8. Have a dedicated download website with updates, a help page and more information
9. Support multiple languages (probably French)

### Won’t

1. Be in 3D, or try to simulate it
2. Have any major bugs that impact the enjoyment of the game
3. Have unintentional loopholes that bypass game logic
4. Be violent, graphic, or destructive
5. Not include any complex AI elements
6. Connect to the Internet for software updates

# Research

As mentioned above, I am not familiar with gaming and needed to research a wide range of games that I could potentially build. The main type of achievable game genres I found were platformer and puzzler. Most other game types like first-person shooter (e.g.: Fortnite) and sandbox (e.g.: Minecraft) require complicated 3D graphics outside of the scope of this project.

During my research, I was pleased to be proven wrong that not all games are designed to be addictive. In fact, I found a large community of games much more low-key and cleverly designed, and so wished to make a game in this genre.

The most basic template for a game is an arcade game, which is what I have decided to stick with. I already have experience in coding Space Invaders, and a basic attempt at PAC-MAN.

## OpenTTD

I found one particular open source game called OpenTTD.Une image contenant carte, capture d’écran, texte, Logiciel multimédia

Description générée automatiquement

Figure 4: OpenTTD Home Screen

Une image contenant carte, capture d’écran

Description générée automatiquement

Figure 5: OpenTTD Game

Une image contenant carte, capture d’écran

Description générée automatiquement

Figure 6: OpenTTD Gameplay

Une image contenant texte, capture d’écran, logiciel, Site web

Description générée automatiquement

Figure 7: OpenTTD Wiki

At first, the game was difficult to play, but I grew to enjoy the complexity as it meant there were so many possibilities, and that the game is designed for exploration more than anything else. The game is clearly very technical and I am aware that coding something of this nature would be a struggle for me. The design of the game felt nostalgic, which I appreciated.

## YouTube Videos

**You Are Given Strange Objects, And Must Learn Their Rules** <https://youtu.be/RDp33e3ttTE>

Figure 8

**How Have I Never Played These Iconic Games Before?** <https://youtu.be/Ti02T7X9oPo>

Figure 9

I researched some different ways I might implement the game. A popular choice was Unity, but I am most comfortable coding in Python, and am aware of the Pygame module used by other Python developers to make games.

I also found YouTube to be a good source of game reviews and gameplay videos. The Figure 8 & 9 helped me understand how puzzle games are built and what makes a successful puzzle game. I had initially started implementing an escape room game for this project, but this research revealed it would not have been successful.

## Python Games

All other Python games that I had the opportunity to play were simple.

Une image contenant capture d’écran, texte, Logiciel multimédia, affichage

Description générée automatiquement

Figure 10: <https://fusevolt.itch.io/dungeons-and-caves>

I didn’t enjoy this game, as it was very basic and difficult to understand, but this is the kind of graphics level I will be aiming for.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Figure 11: <https://ohsqueezy.itch.io>

I thoroughly enjoyed playing this game and was a little disappointed that the source code was so complex. This is the style of game I wish to replicate: I liked that it was easy to get started, and there was an element of puzzle in it that makes the user think, though the second puzzle in Figure 11 was significantly more challenging than the first! There is a clear objective.

## Themes

There were a couple of personal interests that I could feasibly apply to the game, computing and trains. Trains, however, is the much more unique option and I saw an opportunity to add humour by having cheesy National-Rail-related puns throughout the game that would appeal to my audience.

[find an example of a train game online with such puns, if possible]

# Solution

After the above research, it is very clear that my stakeholder would enjoy playing a simple train puzzler, developed in Pygame.

## Potential Design Ideas

Design

# Interface

* Draw, label, explain and **justify** choices made

# Data Structures

* Classes with attributes and their methods, **justify**

# Algorithms

* All algorithms made or used need their own pseudocode

# Decomposition

# Solution

# Testing Approach

* Any information here about how you plan to test the project and who you plan to test it with

Implementation

(mention previous game design **only if relevant**)

# Iterative Process & Comments

## Iteration 1

* Go back through Github log and take screenshots of code development and show the story of the code
* Screens, objects, classes and game loop changes all require a **justification**

### Error 1

* Show error
* Explain error
* Fix error
* Show error fix

## Iteration 2

* Movements, game logic etc.

### Iteration 3

* “AI” and how enemies react

Testing

# Executed Test Plan

* Use Computer Science textbook and approach testing formulaically

Evaluation

# Testing

# Solution

# Product

# Maintenance & Development

* How the product can be improved and maintained
* How effective was the development technique