

Script for our project presentation:

### **POWERPUFF GIRLS SLIDE**

#### **Slide 2 - Group Members (Natalie) 14 seconds.**

Good evening, everyone. We are Software 3 Group 1.

We are really excited to introduce you to “Guess The Song”, an enjoyable and interactive musical quiz game that we’ve designed to challenge players’ musical knowledge and quick-thinking skills while guessing popular songs.

### **NEXT SLIDE 3 Flash “Agenda” onto the screen.**

### **NEXT SLIDE INTRODUCTION**

#### **Slide 4 - Introduction (Natalie) 21 seconds.**

Our playlist today is linked to Spotify’s UK Top 40 playlist which will fetch the up-to-date songs.

### **NEXT SLIDE GAME DESIGN**

#### **Slide 5 - Game Specification and Design Pari (45 seconds)**

So once the game begins Kivy creates the game app & instantiates the gameplay screen with the question index and score set to 0

Python checks the question index to determine if we've reached the maximum number of questions, which is 5 in our game.

If the check is successful Python connects to the Spotify API to generate a song and its details. A 10-second clip of the song is played and its title and the four artist options, are displayed to the player

If the player selects an option within the 10s ,and its correct the player earns 10 points and index is increased by 1. If its incorrect or the 10s elapses index is increased by 1 and the player proceeds to the next question.

This continues for 5 rounds. Then the check fails, and we proceed to finalize the game. Here Python logs the player's score and Kivy instantiates the end screen, displaying the player's score and 3 options. 'View High Scores, which displays the top 10 scores. 'Play Again,' which takes you back to the start screen and End game which ends the program

**(Get ready to play the sample of the game to demonstrate the “How to play”)**

**POST LINK IN SLACK RESOURCES**

## NEXT SLIDE

### Slide 4 - Project Implementation and Execution: Development Approach (45 seconds)

Our team adopted the Agile methodology;

- We held weekly meetings where we shared progress, discussed challenges, and assigned tasks for the following week.
- Each team member took responsibility for a specific aspect of the project. These roles were well-defined but also flexible, allowing support across tasks.
- A different scrum master was appointed weekly for KANBAN task organisation, clear product vision and conducting the weekly meeting.

The project timeline was split into five weeks:

- **Week 1:** research and learning.
- **Week 2:** lay the foundations of each member's responsibilities.
- **Week 3:** Build on foundations, and commitment to Kivy over HTML and CSS.
- **Week 4:** System integration, including unit testing and initial documentation.
- **Week 5:** Project completion and submission.

## NEXT SLIDE

### Slide 5 - Testing (45 seconds)

It was our first time coding with a Graphical User Interface, and that posed testing challenges. The Kivy code and game logic were closely intertwined, making traditional unit testing tricky. Still, we separated out some python logic and managed unit tests for song data, API connections, tracking answers, and the 10-second timer. Limited user input (through multiple-choice buttons) in the software design helped reduce edge cases.

Integration testing followed a step-by-step approach to ensure harmony among new and existing components. After each code change, we did manual regression testing. For example, when we made the shift from free user text input to multiple-choice, it made the game speed up too much; we quickly fixed that.

\*CHANGE SLIDE\*

User Acceptance Testing (UAT) involved sharing a demo version with 14 users through Google Forms, covering four gameplay scenarios. Feedback was largely positive, prompting tweaks like adding start screen instructions and made us aware of audio sync issues on some devices.

**NEXT SLIDE**

**Slide 6 - Challenges & Achievements (22 seconds)**

Next to consider our achievements and our challenges.

Our main achievement was bringing the core game logic to life and managing to integrate this with Spotify API for fresh song data.

Onto challenges, • Navigating the Spotify API proved intricate, necessitating multiple trials. Learning Kivy was another challenge.

**NEXT SLIDE**

**Slide 7 - Future Direction (22 seconds)**

Now let's think toward the future and the aims we have to grow our project. Some examples are:

- Expanding song variety Introduce multiplayer mode
- and Implementing different difficulty

In conclusion, while we faced many challenges, and have more we wanted to develop, our achievements stand as a testament to our team's dedication and teamwork as it is through collaboration that we were able to succeed.

**NEXT SLIDE**

**Last slides - Honourable Mentions (Natalie): 14 seconds.**

And finally, it is essential for us to acknowledge the dedicated efforts we put into aspects that ultimately did not make the cut because members of the team worked tirelessly on them and they deserve recognition! Eliza's beautiful HTML & CSS work and Natalie and Erin's constant development of the database.

Thank you for listening! We open the floor for any questions.