Computer Science Large Practical

Part 2

Submitted by:	**s1504942**
Submitted on:	Thu Nov-09-2017 at 23:23:21
Coursework deadline:	Fri Nov-10-2017 at 16:00:00
This document produced:	Fri Dec-01-2017 at 11:18:12

1 Introduction

This document contains a copy of your Part 2 submission for the Computer Science Large Practical.

— ◇ —

Feedback and the mark awarded are presented in Section 2 at the end of this document.

Songle – Discover Songs Like You Have Never Before

An android based song guessing game.

Overview

Songle is an android game where players try to guess songs. Each song is a puzzle with the puzzle-pieces being the song lyrics. Players walk around in the real world following a map, to collect words. They use the collected words to guess the song-puzzle. More puzzles are added dynamically to keep the user interested. For more information please look at the following document: http://www.inf.ed.ac.uk/teaching/courses/cslp/coursework/songle/coursework-cslp.pdf [1].

Features

1. **Main Controls** – The user may decide to **Play**, change **Settings** or look at S**tats**
2. **Playing Progress (Bonus)** – User can see all available puzzles, BUT only play the first three in the beginning. As user progresses, and completes these puzzles, more puzzles become available to play. This gives the user a sense of progress and achievement each time more puzzles become available. The puzzle selection screen also shows your progress in each puzzle and acts as a review screen.
3. **Puzzle Attainment Level (Bonus)** – When player selects a puzzle, he is shown a map with words that he can collect. As specified in [1], maps with varying level of information is available. More informative maps are shown based on progress. Once you collect all words on Map1, you are shown Map2 and so on. Since songs have different number of words, each song will give a different experience.
4. **Guessing –** The map always shows a guess button which takes user to the guessing screen for the particular song. The Guess screen contains all the words collected yet and gives the user ability to enter the guess.
5. **Progress bar (Bonus)** in map shows how many words are collected and how many are available to collect.
6. In the guess screen, **Sentence Builder (Bonus)** is provided. The users can use the collected words to try and build a sentence which may help them in identifying the song.
7. If the user guesses correctly, he is taken to the Correct Answer screen, otherwise, he stays on the same screen and is **made aware that his guess is incorrect**.
8. The Correct Answer screen congratulates the user and shows information relating to the song. The user is prompted to go back to the song puzzle-selection screen.
9. **Stats (Bonus) –** The user can select to see his stats at the home screen. Following stats are shown: Distance covered/steps, Number of songs guessed correctly and tentative calories burned.
10. **Settings –** User can reset his progress. More options can be made available to the user in Settings in future.

Features and typical game-play are described in more detail in sections below. The sections also mention some ideas which may be implemented but **should not be expected in the first submission** (Part 3 of coursework) of the app. These ideas provide an insight into how future features may be added or improvements be made. They are **clearly mentioned** to be speculations and are not commitments. They are added so that if a feature seems very exciting, it can be discussed.

Design

Entry Point

The entry point of the game is shown in figure 1 below. From here, the player can choose to start playing, look at stats or change settings. Since the game depends on an active network connection to download puzzles, the game will insist user connect to the internet. This is shown in figure 2. The user can click the message to make it disappear and try again. If the download still fails, clicking the Play button won't proceed to next screen and will display a message as shown in figure 3.

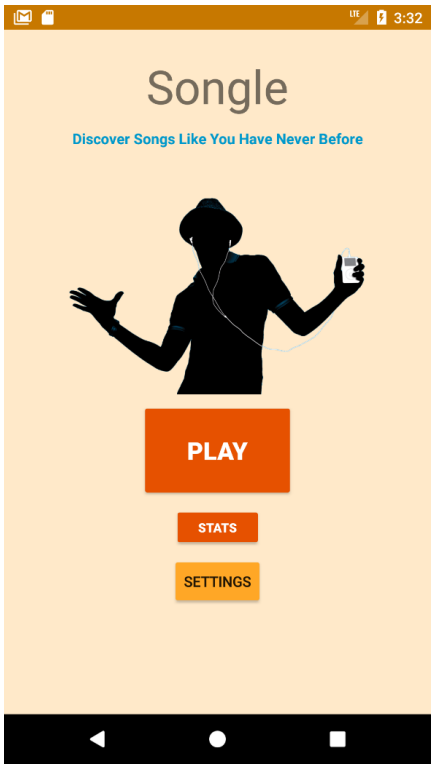

Figure 1: Entry point of the app

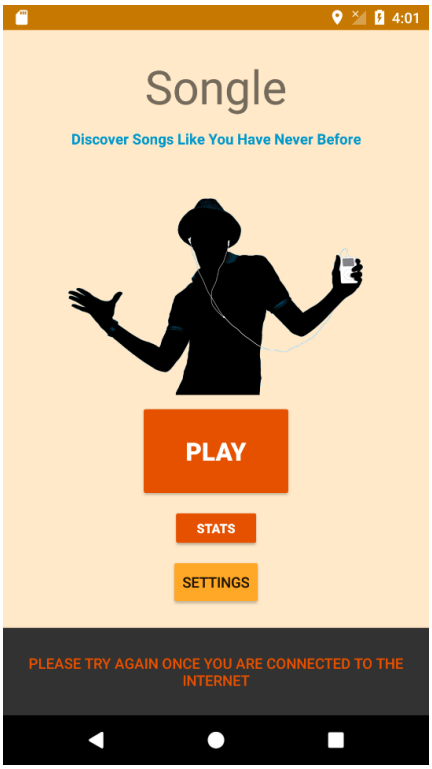

Figure 2: No internet Connection

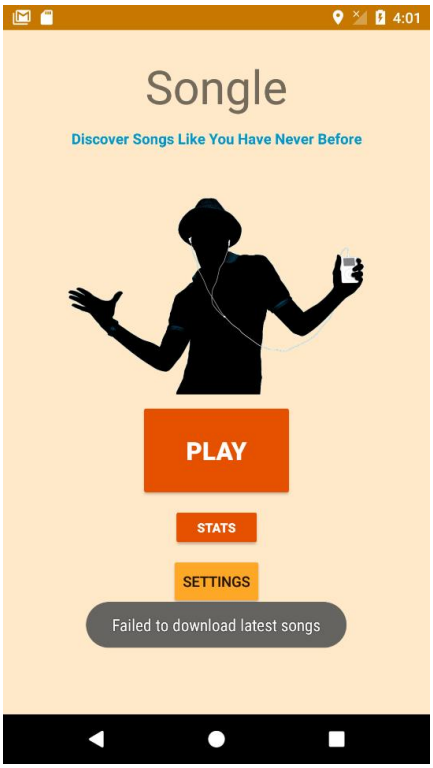

Figure 3: User clicks play but no network connection

Offline play is not allowed since the game is played with a map and the map requires an active internet connection. In the case where internet connection is lost after this screen – the metadata is already available and the app will continue to work, but when the user tries to play a map, he will be notified again to get an internet connection.

As a final note regarding this screen, this screen lays out the colour and design theme for the whole app. All other layouts will follow similar colour schemes to provide a continuousness for all app components.

Song-puzzle Selection

As described in features above, the user will be given a chance to play a song-puzzle with a predefined scheme. Consider figure 4 below.

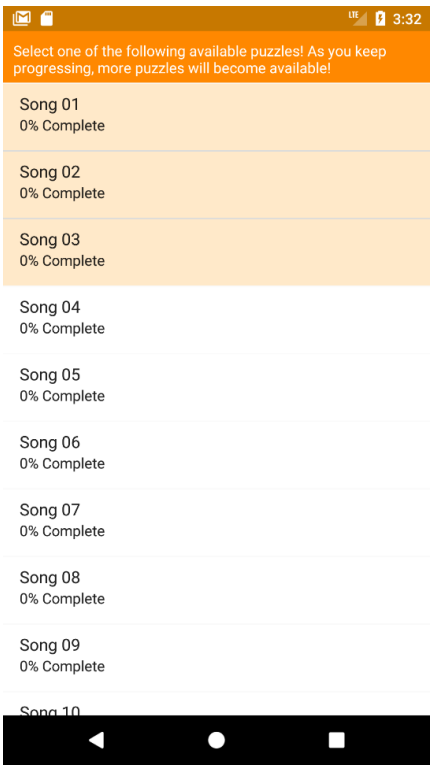

Figure 4: User may select any from the first three available puzzles.

In the figure, all songs available are displayed but only three are highlighted to indicate that they can be selected. Inactive puzzles can't be clicked but can be seen by scrolling down till the end of the list. For each puzzle, the list also shows progress. Progress becomes 100% as soon as the song is guessed regardless of the number of words collected. Progress on incomplete puzzles is determined by words collected divided by total words. This screen is where the user can review all the puzzles.

Progression scheme – Once the user has completed any one of the three, next puzzle becomes playable. This scheme continues till all puzzles become available. This incentivises the user to stick to puzzles and not give up early and hence making the task fruitful for the user.

In future, the scheme may be changed. The scheme may be changed to (for example) – opening three puzzles for each solved instead of just one. And so on. However, **this is not to be implemented in the first submission**. This merely suggests that the scheme can be altered without losing its essence (of unlocking puzzles based on achievement).

Puzzle Map

Once the user selects a song-puzzle, he is taken to a map where he can start playing immediately. Consider figure 5 and figure 6 below.

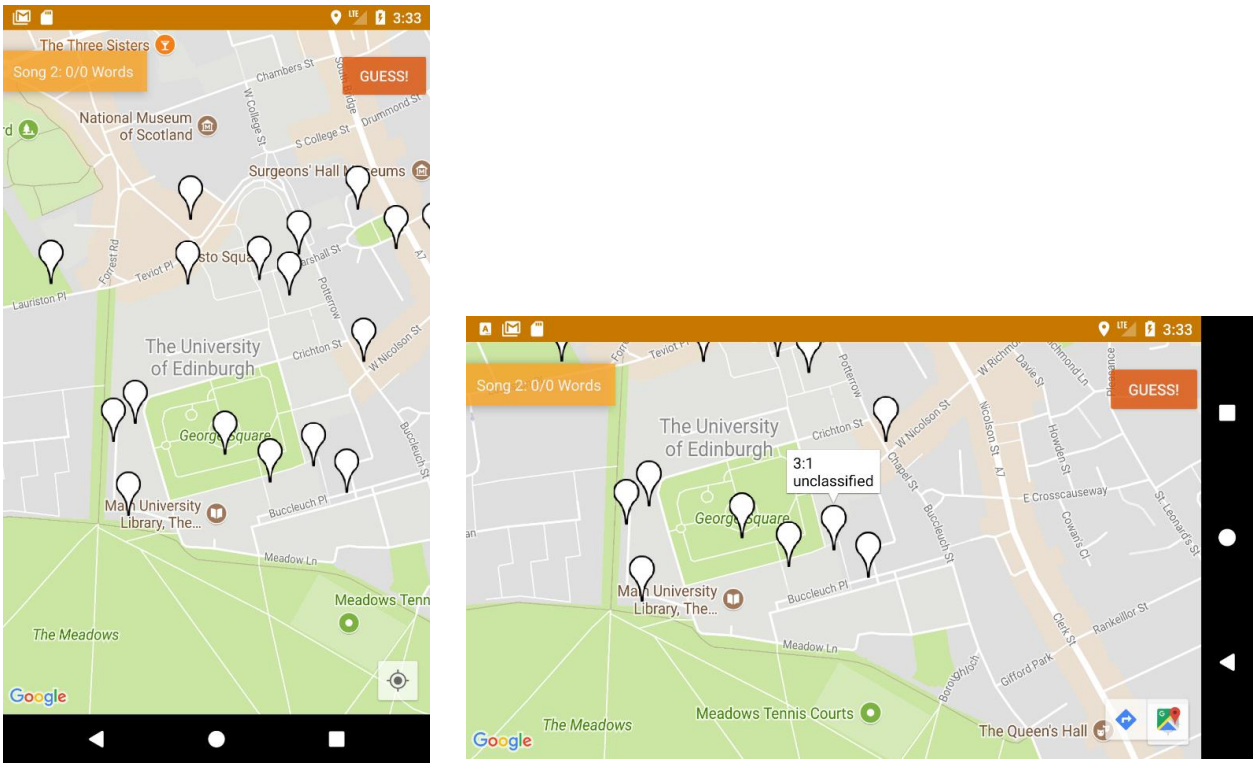

Figure 5: Shows the map screen in portrait mode and figure 6 in landscape mode. Both have a status bar on the left side, and a Guess button on the right which takes you to the guessing screen. Shown in figure 6, is what happens when a user clicks a placemark.

On the top left corner and top right corner, there are progress indicators and guess button respectively. The progress indicator will display puzzle number, words collected and map level (not shown above). When a user is sufficiently near a placemark, user is indicated (using a Toast with text – "Collecting word…" and "Word Collected. Review in Guess mode.") that a word has been collected and he may review the word in Guess screen. Once the user has collected a word, the placemark disappears. This is because as the game progressively shows more and more word placemarks, the map starts becoming cluttered. On clicking a placemark, user gets an option to see that placemark in google maps and get directions to the placemark (right-hand side bottom; shown in landscape mode).

Map scheme – Once the user has collected all words in a particular map (see [1]), next map is shown minus the words already shown in the previous map. This keeps happening till the user is on the 4th map. The 5th map's very-interesting words are given as hints for free when the user asks for a hint.

Guess Mode

When the user clicks the guess button on the Map, Guess screen is shown. See figure 6, Text field is in focus by default where the user may enter his guess and check by clicking the 'GUESS!' Button. User may press the back key to remove the keyboard and see the list of words collected yet. Figure 8 shows how the screen typically looks. If the user enters an incorrect guess, he is notified as shown in figure 9. On a correct guess, the user is taken to a different congratulating screen.

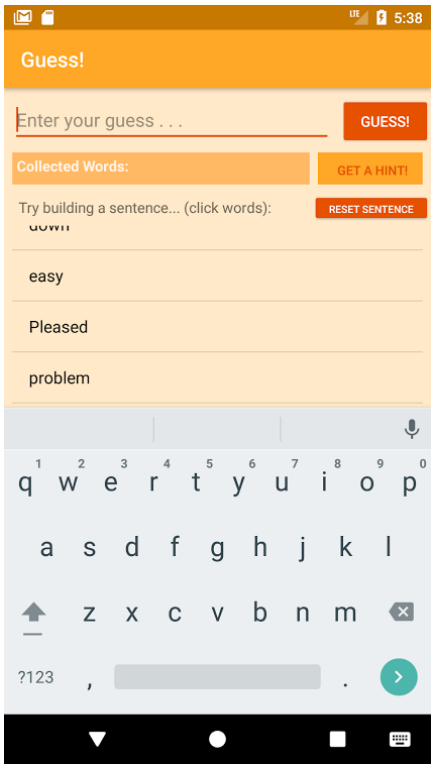

Figure 7: Text filed in focus for the user to guess the song

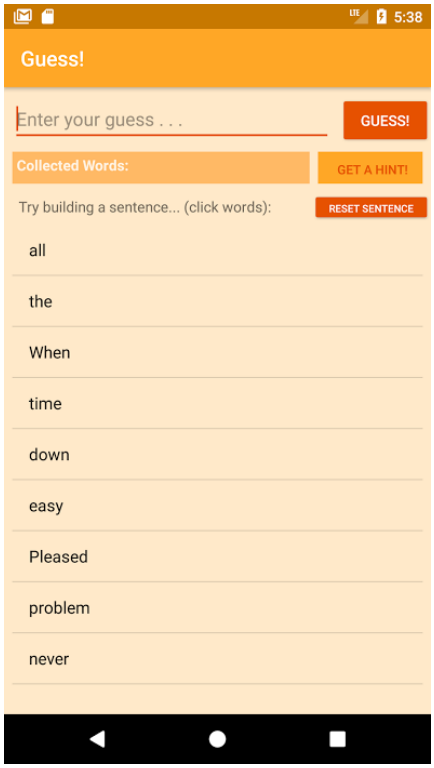

Figure 8: Typical look of the screen

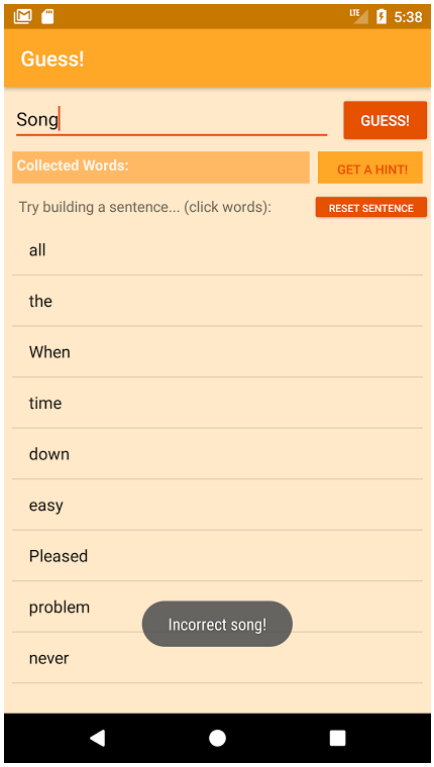

Figure 9: An incorrect guess by user

The 'Get a Hint!' button may be pressed at any time and the user will be given a 'very interesting' word for free (the user is asked to confirm with an alert box). The free word is added at top of the list, hints are given until all very interesting words are over. If user still fails to guess the song, he may not get an answer at all. This decision was made as games are made to be won! If a user could simply get the correct answer he may be tempted to choose that option every time.

Just above the word list, a sentence-builder is provided which is explained in the next section.

Sentence Builder

Users are given an option to try and build a sentence from the collected words. Building sentences may help user to solve the song-puzzle. The user may simply click the words in the list and they will be concatenated together to form a sentence. User may click the Reset sentence button at any point to reset the sentence. For example, in figure 10, the user builds 'all the time . . .'

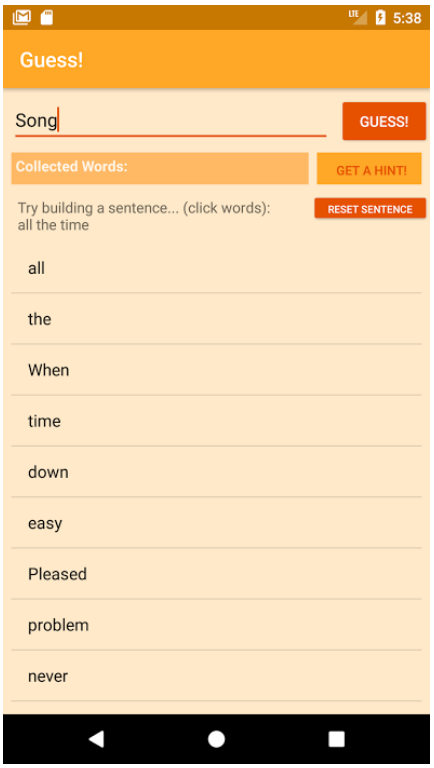

Figure 10: User attempting to build a sentence

If the sentence is longer than a line, the list shifts down making space for more lines. However, after a certain word limit, no more words can be added.

Reason for this feature – The simplest strategy to guess the song would be to try and figure out whole sentences from the collected words. This feature provides a handy way of doing this without leaving the app at all.

Congratulation Screen

If the user is successful in guessing the song, the 'GUESS!' button should take the user to a page like the one in figure 11. Once successful, appropriate changes in the puzzle review page will be made automatically. Like more puzzles will become available to play and progress for particular song will change to 100%. The user is presented with information such as song name, artist, distance covered trying to guess the song and a link to YouTube video of the song. Clicking the link will open the YouTube app and start playing the song video. User is prompted to go back to puzzle selection screen with a big button – "BACK TO SONG LIST".

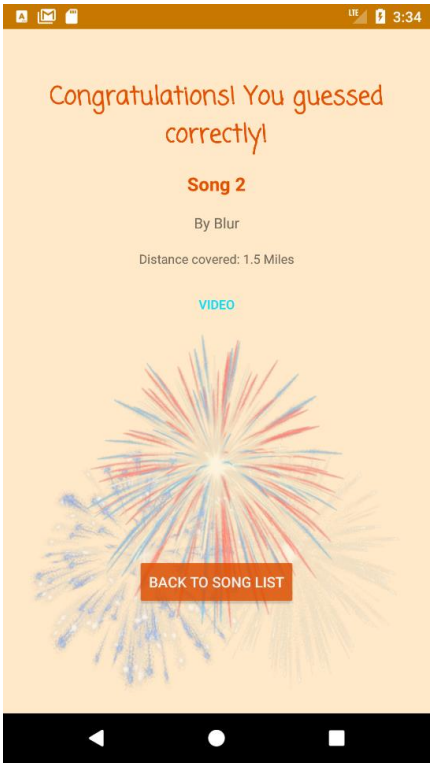

Figure 11: Congratulating user for successfully guessing Song 2 by Blur

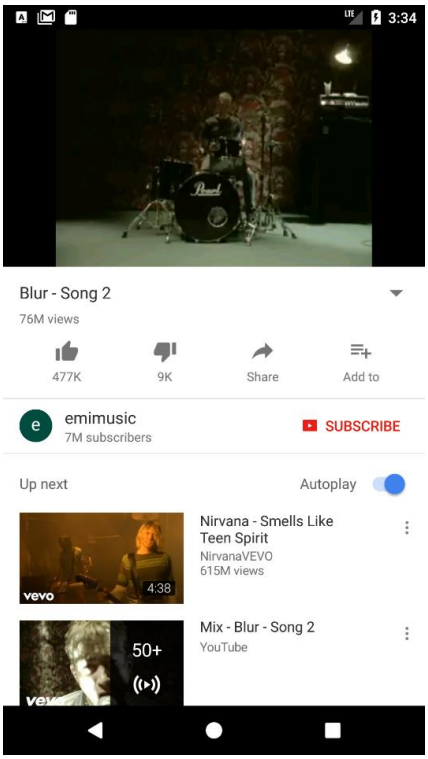

Figure 12: YouTube app opens to play the video.

Even after guessing the song, user may still browse the same puzzle-song and keep collecting words. Even though the progress shows 100%, the user has every right to forget and try guessing the same song again!

Being successful should give a sense of achievement. This is done partially by displaying the distance covered by the user. But more so by making more puzzles available. The price/award are the new puzzles!

Stats Screen

On the main screen, the user may choose to see his stats of game-play. This is provided as a bonus feature and introduces a fitness/health element in the game. The calories-burned count should incentivise the user to keep playing and burn more calories.

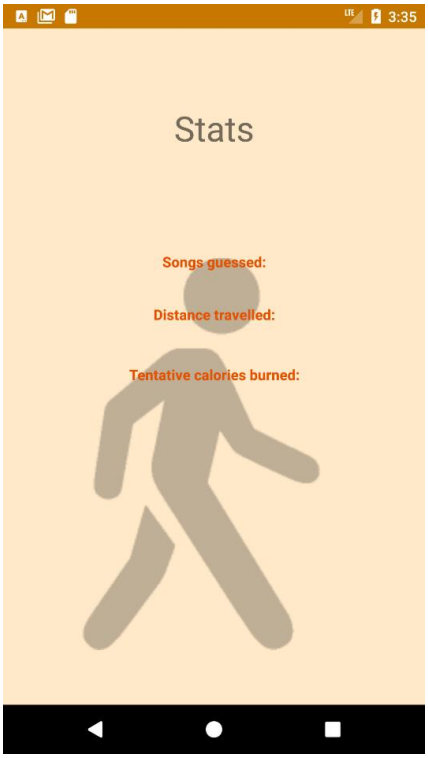

Figure 13: Stats screen. The blank values will be replaced with real game-play data.

The stats shown will be number of songs guessed, total distance walked while playing the game, and a tentative calorie burn count.

Settings

The user may change setting by selecting the settings button on main screen. In settings, user may reset the game data. In future (not for the first submission),user may change background music setting and set any user preference that the game may provide as the development process finishes.

Comment on the App Design

Notice that the game play is designed to be seamless. The user does not choose map level, he progresses and gets better maps. The user does not randomly select a song-puzzle, but he progresses in the game and more puzzles become available. The user can guess the song on the same screen where he can review his collected words. User is encouraged to try and guess the song every time he reviews his words by providing a sentence builder.

All design decisions were made keeping in mind that the purpose of the design is to keep user interested in the game by providing him a sense of achievement with every bit of progress. As more words are collected, a more clear and harmonious screen is presented by removing the collected word placemarks. Then, a more colourful map is viewed once first map words are all collected. And it feels like the puzzle has reached a new level.

Since everything just happens by itself as the user progresses, I believe the user-interface feels user friendly. The user just needs to concentrate on guessing a song-puzzle! And then more song-puzzles!

2 Feedback on your submission

2.1 Your design document overall

Your design document should be a record of your design decisions for your app and should include a definitive list of the bonus features which are offered by your app.

— ◇ —

Overall, this is an excellent submission and the document has a helpful structure which makes clear a standard play of the game. The design is clear and consistent and the bonus features provide an enhancement on the gameplay.

2.2 Design decisions

You were given a list of typical questions for the design document to answer.

- How is the song that the player has to identify chosen?

  ▷ You have answered this question.

- What does a player have to do to collect a word? Does anything happen to the placemark when the word is collected?

  ▷ You have answered this question.

- After a player has collected several words can they review them? If so, how do they do that? What does the "review screen" look like?

  ▷ You have answered this question.

- When a player thinks that they can guess the song how do they enter their guess?

  – What happens if their guess is correct?
  – What happens if their guess is incorrect?

  ▷ You have answered this question.

- When a player thinks that they *can't* guess the song how do they indicate that they give up? What happens then?

  ▷ You have not discussed this.

- What determines which of the five versions of the map is shown?

  ▷ You have answered this question.

- Can the player set the "level of difficulty" of the game? If so, how?

  ▷ You have answered this question.

- After a player has identified a song, could that song be chosen again as the puzzle to solve?

  ▷ You have answered this question.

- After a player has identified several songs, can they review their list of solved puzzles? What does that "review screen" look like?

  ▷ You have answered this question.

- What does the game do if a data connection (4G) is not available? Can it be played at all?

  ▷ You have answered this question.

2.3 Bonus features

You were asked to design some *bonus features*, which set your app apart from others and to include a definitive list of your bonus features in your design document. Bonus features are enhancements which are intended to make the game more interesting to play, or more rewarding, causing the user to play more frequently, or for longer sessions. You were asked to add two additional bonus features of your own invention, although games with more than two additional bonus features are also welcome.

— ◇ —

You listed several bonus features in your design document.

- Playing progress
- Puzzle attainment level
- Progress bar
- Sentence builder
- Statistics

Each of these is a useful feature and there are some very extensive bonus features which add a lot of value to the Songle game. Together they certainly satisfy the requirement to provide bonus features for your app.

2.4 Design of views

You were asked to provide screenshots to give the clearest picture of how your app will work in practice. You were asked to aim to show a typical play of the game including at least the user viewing a map, guessing which song this is, and being informed whether or not they are correct.

▷ You have done this, and it is easy to follow the progress from one activity to another.

2.5 Your mark for this submission

Taking into account the above assessment of your submission, your mark for Part 2 of the Software Engineering Large Practical is **98%**