206: Assignment 03 Report

Name: Buster Major
GitHub: Buster-Darragh-Major
UPI: bmaj406
Partner: Nathan Cairns

Tatai is an application designed to aid with the learning of Maori numbers. In its current form, the application prompts the user with an integer on screen, to which the user can record and submit their own pronunciation. The program then gives immediate visual feedback on their pronounciation, and at the end of answering questions the user can see their results in an intuitive, easy to read format.

The current Tatai application prototype serves as a foundation for the most basic features found in the eventual finished product. At this point in time, there are two main functions; gameplay and statistics viewing. Within the gameplay function there are several key features to point out. Firstly, there are two game modes currently implemented, level 1, which prompts the user with ten randomly generated questions with integers ranging from 1-9, and level 2, which prompts the user with ten randomly generated questions with integers ranging from 1-99. Each question receives its own screen which can be navigated through by answering each successive question through voice recognition methods. A single question screen consists of the integer itself with a randomly generated colour scheme for text and background. A further feature of the gameplay environment is the visual feedback given to the user when an answer is submitted. For a correct answer, the colour scheme changes to reflect a green background with correct answer in text shown on the screen. An incorrect answer changes the background colour to red, and for a first attempt allows the user to try again. On a second incorrect attempt the correct answer is displayed in text and a button leading to the next question appears. At the end of a game the user is shown a table of questions they answered with corresponding correct pronunciations and feedback on whether the user answered correctly. The second main function of the application shows the user an easy to navigate view of their all-time game statistics. Information shown includes total questions answered, total correct, incorrect and an all-time average.

Throughout the assignment Nathan and I were able to work in a relatively consistent and concurrent manner. It was our collective responsibility for the construction the front end and backend, the former of which we began building first. Implementing the HTK resource was the last step in construction. Much of the code was worked on concurrently, we would delegate the implementation of features to one another rather than separating work flow by front end/back end. Furthermore, we found it beneficial to work in the same space which allowed face to face communication for many issues and design decisions. This often meant meeting at university in our own time and working on the project together.

Initially, it was my responsibility to layout the different window/menu views, and Nathan would apply styling and set out resizing properties. For the backend, it was my responsibility to write classes for question and colour scheme generation, integer to Maori string translation and game logic. Nathans responsibilities consisted of writing statistics storage classes, the entirety of the statistics view, game logic, refactoring and improving code readability and reusability. It is notable that the game logic code had to communicate with both question classes and statistics classes, so much of the code found in this functionality was written by Nathan and I concurrently, which is an obvious example of the necessity of version control.

The form of version control utilized in the project was that of GitHub. The standard methods for implementing Git repositories, commits and branches were applied. A branch was created for every major feature or potentially code breaking implementation, including but not exclusive to statistics implementation, GUI modification, game logic and major refactoring. There were very few merge conflicts that arose, to which I attribute our effective communication and delegation of tasks. Conflicts that did arise were often trivial, and frequent face to face communication meant they were resolved instantly.

The working relationship forged during the project was successful for a variety of reasons. Significantly, Nathan and I are at a similar skill level in code writing and design. This meant there was little misunderstanding of shared code, and the workload was shared evenly throughout, as we each had to put in the same effort into learning and mastering new skills. Because code writing is a relatively new skill for both of us we chose a relatively simple and straight forward design for the back end architecture. Communication was a further factor in the success of the project. Frequent meetings to discuss project progress and direction, and coding together were helpful in ensuring we both had a similar vision for the demo program.

An aspect in our processes we felt could have been refined was that of long term project planning. On reflection Nathan and I decided we were somewhat hasty with our starting of code work. Design decisions like storing cross session statistics in a .csv file are now being reconsidered, and with more research and further planning a more robust implementation could have been written earlier. A further slight planning issue was the inclusion of the HTK hmm files needed to run the voice recognition software. With better planning the implementation of these files could have been more seamlessly integrated with the .jar file submitted at the prototype. This issue was one of several minor issues we had when working on packaging up the demo into the required .jar format and testing on the virtual machine. Once again, this could have been reduced or prevented with better planning.

There are several key design decisions in the Tatai demo that were made at an early stage. The button layout and aesthetics are intended to reflect the clean and lightweight vision Nathan and I had for the project. An effort was made to remove clutter and give the user a minimalistic interface. This allows for much expansion and enhanced functionality later in the project without adding clutter to the user interface. The in-game colour aesthetics were chosen to be easy to digest and playful. Lighter pastel tones make up the range of colours chosen for the background and darker, bolder colours make up the range of possibilities for the text itself, resulting in a high contrast, easy to read arrangement. The menu colour scheme is a dulled palette derived from the colours of the Maori flag, which ties the application to its cultural influence.

The Tatai demo is an indication of the direction we have chosen to develop this project. There are many features yet to be implemented, and the bare functionalities are the only ones shown.