Background

Established in the 1980s, Te Wānanga Māori is one of the first faculties of Whitireia Community Polytechnic and is made up of three departments: Māori Art, Nursing, and Te Reo Māori.  The client for this project is the Reo Māori school.

In order to achieve proficiency in both formal and informal situations, the school takes the approach of teaching students traditional Māori.  Once students have become proficient in using more traditional language they are then in a position to learn more colloquial expressions and judge when it is appropriate to use them.

One consistent problem tutors have noticed, is that students have a problem differentiating between the use of *ehara* and *kahore* in negative sentences.  Both words are used in traditional Māori to indicate the negative, however they are not interchangeable. Further contributing to the confusion between the two words is the fact that *ehara* can also be used as colloquial expression meaning in more informal Māori.

Students have several resources available to assist them with independent learning In addition to classroom textbooks and handouts. These include CDs, an online dictionary for looking up individual words and the hard copy Dictionary of Māori Language (H. W. Williams) which provides more detailed information around vocabulary and its use. These resources are all valuable in their own right, however they lack interactivity and customisability.

Scope

The purpose of the project was to come up with a working prototype of an Android application that would enable students to practise differentiating between *ehara* and *kahore.* While the focus for the first prototype focuses exclusively on sentences containing *ehara* and *kahore*, the application has been designed so that other sentence structures can be added to extend the application in future.

1. The project used RAD development methodology and was split five two‑week iterations that progressively built on the previous iterations.
2. Each iteration was be made up of:
   1. The development of an interface that is useable and intuitive to use. It should be noted that the current interface is a placeholder, as the goal for this project was to produce a prototype that can be further developed, rather than one that fully enhances the user experience.
   2. The development of a database in SQLite that stores the sentence structures.
   3. The development of the backend programming in Android Studio which ties together the database and interface and provide the application’s functionality.
   4. Black box testing of the components that were developed in each iteration.
   5. At the end of each iteration the Client was presented with the product for feedback. In the initial iterations this was a simple interface and as the project went through each iteration further functionality was added.
3. The application was designed to work on an LG Nexus 5 running Android 6.0.1 phone.
4. A user manual and system documentation created to support the application by providing the client and future developers with background and reference material about the project.

Objective

The objective of our Android application was to provide a means for users to improve their learning of Te Reo sentence structures, specifically negative sentences using ehara and kahore.

This aim of the application is to provide students with a portable resource that can be easily accessed and that complements existing classroom resources.

Methodology

The methodology used for this project was RAD (Rapid Application Development) with Kanban. RAD was suitable for this project as it emphasises places more emphasis on rapid prototyping and iterative delivery rather than extensive planning. The prototyping was done in multiple iterations with new functionalities and features added at each iteration.

The life cycle of RAD has four stages:

***1.    Initial Requirements***

The project team met with the client and came to a rough agreement on the business needs, application requirements, and project scope before prototyping began.

***2.     User Design/Prototyping***

This stage allowed the creation of prototypes that satisfied the requirements given by the client in the first stage. A prototype was then presented to the client to collect user feedback. Any suggested changes that emerged from this feedback were considered, and if necessary, added to the requirements.

***3.     Construction***

Coding, testing, and integration took place in this stage. This, along with the user design stage, were repeated as often as necessary as alterations and refinements were added based on the client’s feedback.

***4.     Implementation***

When all the system requirements were met and the user feedback is entirely positive, project team will finalise and launch the product. This phase also includes testing and user training.

To help with work transparency and task planning, the Kanban methodology was used alongside RAD. Kanban’s Agile process framework fitted in well with RAD’s flexibility to changes and continuous delivery approach.  This helped give structure to the methodology used throughout the development of the project.

Artefact

Technology

The project was developed in Java using Android Studio as the development environment. Java was the natural choice for language for the project as all team members had previous experience using it for course work.

Android Studio's platform is fully supported by Google, which brings in Android tools such as debuggers, emulators, and wide documentation. Android Studio offers several other advantages, such as the ability to drag-and-drop user interface components, preview layouts on multiple screen configurations, and more. Android Studio is an official tool of Android, so any updates to development features were automatically made available.

SQLite, an open-source, lightweight, and standalone database was used as the backend database for the application. SQLite supports embedded relational database features Android Studio has a built-in SQLite database implementation.

Because SQLite is embedded within the code, this presented an issue of not being able to view and manage the data. To work around this issue, we downloaded a DB browser.  
In order to access the data through a DB browser, we need to save the database SQL file in the application as a DB file. Then drag that file to the DB browser so we can view the data. This process needed to be run each time we made changes to the database.

For repository and backup purposes, Whitireia's GitLab repository manager was used. This enabled us to work collaboratively and access the files of the project through a single source. And allowed for easier documentation, testing, and kept track of the application's versioning. In order to upload files to GitLab, we needed to manually copy the files to a GitHub account that was connected to the project on GitLab.

The hardware we used to develop the application were HP laptops provided by Whitireia which were already installed with Android Studio. We were also provided with two LG Nexus 5 smartphone devices which were used to test our application.

Google drive was used to store the proposal, system documentation, presentation, poster, and application files. This allowed us to easily edit documents and keep the secure.

Slack was used to send code between us. It was helpful for when multiple team members were working on code and one of us needed to see a specific section of code that the other was working on.

Conclusion

Recommendations