Project Proposal

Mobile Flashcard Application with Learning Tracking

COMP 495
Computer & Info Systems Project

Jason Bishop
3042012

Introduction

With the modern trend toward globalization seemingly growing every year, the need to communicate with individuals around the world is also growing. These individuals come from different backgrounds and cultures, and importantly, also use different languages. While many have stated that the international language of business is English (Neeley, 2012), and certainly most of the internet is in English, it is often necessary and/or convenient to be able to communicate in other languages. And for others whose first language is not English, learning English as a second language is necessary to access these resources.

To illustrate with an example we can examine my current travels through Central and South America. While there may be sufficient use of English to get buy in heavily touristed areas, there are often many situations where only Spanish, or non-verbal forms of communication can be used. This can be from something as simple as buying a soda in a store, or trying to strike up a conversation with an interesting local. During this journey, I have also met many individuals who wish to study abroad in places like the U.S.A., Canada, the U.K., and so on. So for travel, business, education, and many other pursuits, learning another language is a need that many people have today. Furthermore, I have spent 6 years as an English teacher in both South Korea and Taiwan. I have experienced first-hand the struggles, achievements, and needs of language learners.

To this end, I would like to help language learners become better and more fluent, while using a tool that makes learning as fun, quick, and easy as possible. What I am proposing is the creation of a mobile application that will allow users to study vocabulary word pairings in a flashcard style format. I want the program to be able to track which words are strongest or weakest, and encourage the user to study on a consistent basis with feedback and fun activities. The user interface should help guide and aid users with recall, and as the program tracks the weakest/strongest words for the user, it can provide feedback as to which items are most necessary to study. In fact, this application need not be limited to language learning at all, but could be applicable to learning any sort of item pairings for a broad range of subjects. For example, a medical student learning the names of various anatomical parts of the human body. Therefore the potential for this type of application to be of use to many people seems to be quite high.

Background Material and Literature Review

To build this application, some background information is obviously required. The primary function is to aid user learning, so a focus on learning theory is necessary. Several studies have looked in to the best ways to retain information. Others have looked at the best way to present information for access and retention with learners. Both of these aspects are explored below in the 'Theory and Research' section.

Additionally, to help solidify the idea into a concrete project, I have examined several language learning materials and flashcard applications on my own mobile devices: an Android Nexus 9 tablet, and the Android LG Prada smartphone (unfortunately, I have no way to access the iPhone store at this time). There are several applications in each category, so I had to limit my evaluations to only the most popular in the Android Play Store. I did this by searching for the terms 'language learning' and 'flashcards' and looked at the top 5-10 applications, downloading 2-3 from each search for personal use and testing on my devices. These applications and findings are explained under the appropriate headings.

Theory and Research

In terms of memory retention, it is widely accepted that spaced repetition is one of the most useful methods for retention of information, as explained by Eich in *The Cognitive Science of Learning Enhancement: Optimizing Long-Term Retention*. He goes further by explaining that the longer the space between study sessions, the longer the item will be retained. In effect, the research shows that the spacing time equates to 10%-20% of the retention time (Eich, n.d.). Therefore, to help users of this app, the study interval should continually increase so that the retention effect will also increase.

Eich also mentions study with flashcards specifically, stating that current research shows that all items should be studied consistently because items that are learned can benefit from testing recall, and items that are unlearned benefit from practice and the spacing that is provided by studying other items (Eich, n.d.). Another aspect of flashcard studies showed that when learners are left to their own devices to study flashcards in the order and amount of their choosing, they often do a poor job of selecting the best way to retain information (Eich, n.d.). Additionally, when giving feedback to learners, it has been shown that it is better to delay feedback somewhat for better retention (Eich, n.d.). Thus, the proposed application should help users continuously study all items, and when quizzing, should delay feedback until the end of the session at the very least.

Work by other researchers also has direct implications on the proposed application. Research by Palvik and Anderson (2008) confirms that a period of time between learning sessions is required for each individual item of study. However, if the period is too long, the information will likely be forgotten. Likewise, a period that is too short and the information is easily remembered resulting in time wasted that could be spent learning other items of information. Therefore, to achieve the maximum benefit of spaced repetition, a balance must be achieved between these two extremes. (Palvik & Anderson, 2008).

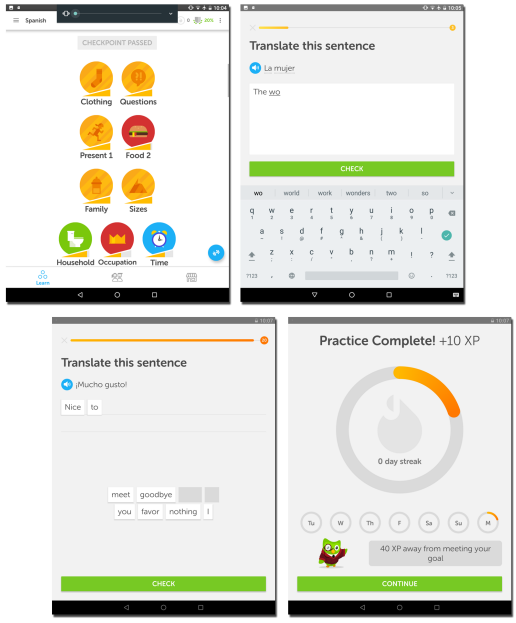
However, in the same study, Palvik also finds that time is not the only factor influencing the retention rate of an information item because some items are easier to remember than others. He

concludes that for each item, a formula must be used to calculate the optimal practice schedule to balance these competing factors. (Palvik & Anderson, 2008). This model can be adapted to the proposed flashcard application. This will allow users to track their progress with each item, and concentrate review on the areas it is most needed.

Now that the underlying learning theory has been solidified, we need to consider the user interface. Up to this point, it has been assumed that flashcards are the interface model of choice. This is confirmed by Eich, but more fully explored in recent research by Pham, Chen, Nguyen, and Hwang *(2016)*. What they show is that a flashcard interface that adapts to individual needs and is appealing to users will increase consumption and retention (Pham et al., 2016). It is also concluded that "card-based design has a great potential to be applied widely in the future for mobile learning application[s] (Pham et al., 2016)." Hence, it would seem that the initial assumption of using a flashcard design paradigm will be of benefit to potential users.

Language Learning Applications

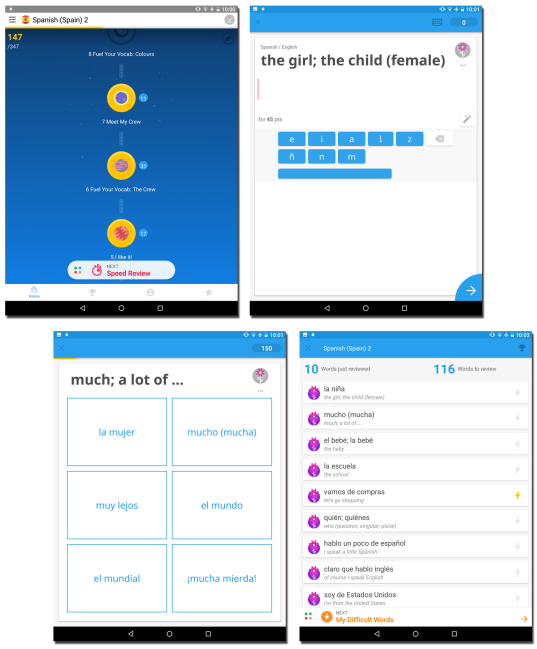
In the category of language learning, I examined 3 applications: DuoLingo (Duolingo, 2016), Memrise (Memrise Ltd., 2016), and Rosetta Stone (Rosetta Stone Ltd., 2016). Each of these had a different focus and approach to language learning. DuoLingo focuses exclusively on vocabulary. Groups of words are categorized by subject and type. For example, there is a category for "Household Objects" and another for "Adverbs". These words are then used to build sentences as the learning continues. The application has lessons in each category, and it keeps track of the "strength" of learning in each category, but it is not clear exactly how this is tracked. Duolingo's aim is to make language education free, personalized, and fun, without simply playing games (Duolingo, n.d.).



images from DuoLingo app

Each Duolingo session of use can involve tasks like multiple choice selection, translation of sentences either written or heard, and pronunciation of words (with speech recognition by the application). There is also some choice that the user can exert in what the wish to learn. There are levels in the application, and the user can choose a category at each level. The user can also review words learned at a previous level at any time. However, because sentences are built on what the user has already learned, the application cannot let the user choose any category that is beyond their current level. Additionally, while the amount of words to learn forms a very comprehensive list, there is no way to study words that are not included and prescribed by the program.

The Memrise application also concentrates on vocabulary, but will also help the user to remember common and/or useful phrases. The application's methods are based on several scientific studies centered around the concepts of elaborate encoding, choreographed testing, and scheduled reminders (Memrise, n.d.). The vocabulary is divided into categories, for example "Creatures" or "Colors" This application keeps track of user progress with each item and prompts the user to review "weak" words. Again there are different types of activities that the use can do to aid in learning. These included multiple choice selection of a written or spoken (via video) word, multiple choice selection of a spoken work (via audio) to match a definition. There is also a speed recall game in which the user is time limited in selecting an answer. Finally, users can also mark words as "difficult" which puts them in a special category for easy access to further study and review.



images from Memrise app

Like DuoLingo, the content in Memrise is divided into category, and each category comprises a level. Users can review and items that have been previously learned, but can not skip ahead to a new category without completing the previous categories. In addition, some languages also have multiple levels. For example, the Spanish language currently contains 8 different levels. However, the tracking and review activities only focus on words within each level. Users can download and use multiple levels at the same time on the same device, but mixing or catch-all review is impossible.

Finally, the Rosetta Stone application is the most comprehensive learning tool in that it covers all aspects of language learning including vocabulary, grammar, reading, listening, writing, and speaking. The program starts at a very basic level, dividing content into levels, and then each level into a series of four lessons. Within each lesson, there are several activities that either review or focus on one of the language learning areas mentioned above. Unfortunately, the writing and level review options in the application were unavailable for use on either one of my devices (The reasons for this limitation are currently unknown, but are limitations of the program itself, not due to any action or inaction by the user). Users have the freedom to attempt any activity in any lesson at any level at any time. However, each lesson builds on the previous ones, so users may have difficulty if they skip too far ahead.

One of the most unique aspects of the Rosetta Stone program, is that it is totally immersive. There is no written translation of any learning information within the program itself. All activities involve the use of information displayed pictorially. The user can select pictures to match sentences and words, or vice versa. There are also spelling and pronunciation exercises that operate in a similar manner. For speaking exercises, the program uses speech recognition software to match what the user is saying, again to match pictorial prompts. These types of activity occur at all levels and within any lesson.

images from Rosetta Stone app

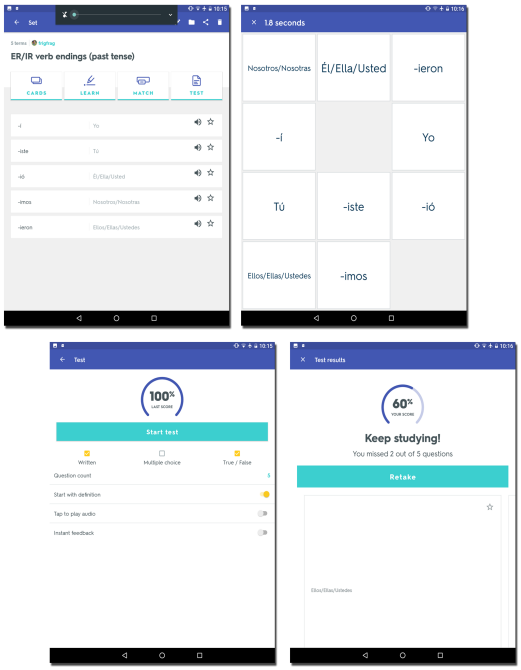
Rosetta Stone is operating under the theory that this immersion is better for total language learning. Their theory is that when we learn our first language, it is through total immersion. Words and meaning are learned through association, and they are trying to achieve the same effect with their program (Rosetta Stone, 2016). However, this can be limiting at times, because as children we are often free to ask about words and meanings to our parents and teachers, and they can explain these things to us. However with this application, the only context is the picture used. If the user does not infer the meaning correctly, or the picture offers ambiguous and/or multiple meanings, the user will have to turn to other resources. Within the program, there is no real way of tracking what specifically the user has learned, and there is little feedback to gauge any type of progress, other than observing the current level/lesson being learned.

The strength of these applications is that they provide a way to track user progress, and they have a variety of testing and recall functions to keep the pace varied and more interesting. What these applications fail to incorporate is the aspect of user choice. The program chooses the words and other learning items for the user to study. There is very little room to study topics outside the narrow scope of the application's presentation. This can be demotivating because some of the subjects or groups my not be fun. Furthermore, a learner may have a specific area of focus for their language learning needs, such as a medical student needing to learn the parts of the body or other anatomical items. These individuals will have to wait until the program lets them study these areas, not addressing the needs of these users.

Flashcard Applications

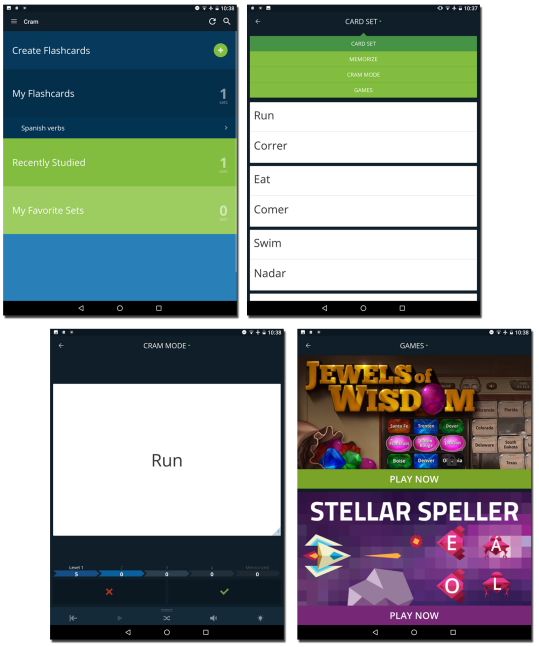
Flashcard applications allow users to enter their own vocabulary (or other learning type) pairings. This gives the user complete freedom to choose which words they feel are most needed or important. The flashcard applications examined were more similar in their approach than the learning applications, probably due to the narrowing of focus. The two applications I tried were Quizlet Flashcards & Learning (Quizlet LLC, 2016) and Cram.com Flashcards (Cram LLC, 2016). At the most

basic level, each offers a way for the user to create a flashcard, by having the user input text for the terms on both sides of the virtual card. Each application also allows users to categorize several cards into various sets, and each has a function that allows users to study the flashcards by presenting them in a series and allowing the user to view the matching pair in some way. There were also several applications that used only predetermined cards and sets, but these were eliminated from consideration because of their more narrow focus. Finally, each tested application also allows the user to quiz themselves on a series of cards, usually in a multiple choice manner, using definitions from other cards in the set as alternate answers.



images from Quizlet app

However, there are also several unique aspects to each program that add more functionality or fun to the learning process. For instance, the Quizlet application offers several fun ways for the user to learn. There is a matching game, simple review, and a quiz with different types of questions. Additionally, there is functionality that lets users share sets of cards with each other. The user can search among these sets and download them to their own device. The program also allows input in multiple languages, and has functionality to recognize and play back audio of individual items. Another unique aspect is that individual sets of cards can be added to folders, which allows a further hierarchical grouping of card sets. The user interface for all these functions is also very clean and minimal, allowing the user to focus in on the learning materials. Finally, one very unique aspect is that the review quizzes are highly configurable, allowing selection of different question types, and even a selection to offer instant or delayed feedback from the quiz.



images from Cram.com app

The Cram.com program has the basic flashcard features, but is not as sophisticated or clean as the Quizlet application. However, like Quizlet, users can easily create, share and search for sets of flashcards. What makes this application unique are two features in particular. First, flashcards are more customizable, they can be tagged with several subject keywords, and the card can have a third "side" that functions as a hint. Multiple languages can be selected for the front, back and hint sides as well. The second unique feature are the two games the app offers. The games are much more graphically appealing and have more sophisticated gameplay than those offered in other applications. The first game called "Jewels of Wisdom" is a matching game that functions similarly to popular "match-3" games such as Candy Crush or Bejeweled. The second game, "Stellar Speller," is a space shooter type game where the user shoots space ships containing letters that spell out the answer. The game is like a horizontal version of Space Invaders, but with power-ups. These games can make the

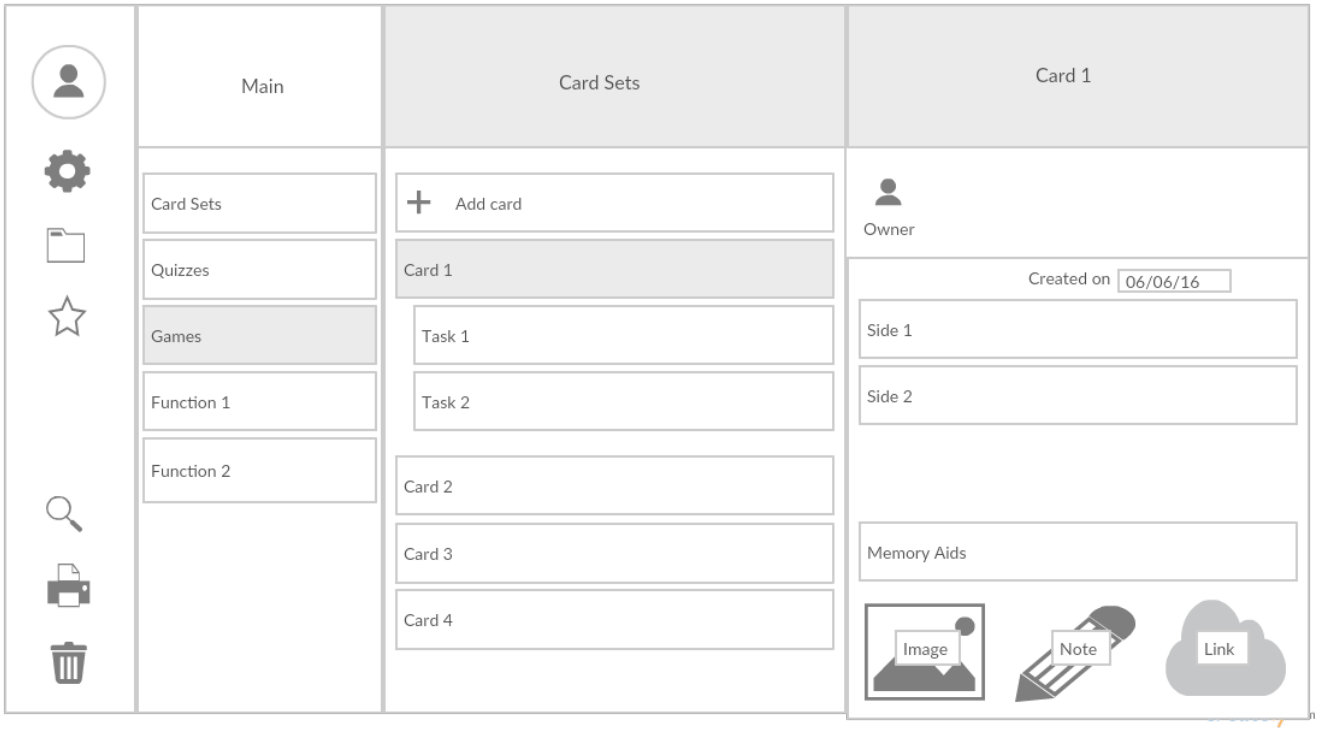
learning aspect more fun, but are not necessarily optimized for learning. Additionally, the search function to locate other users cards, is clumsy and awkward to use, and other parts of the user interface could be better presented.

While these applications give users the freedom to study virtually any material they choose, they also have certain limitations. The most prominent feature missing from these applications is a comprehensive way of tracking what has been learned, and what items are in most need of review/study. Users are given complete freedom to study what and when they want, but as is shown in the literature, users are not likely to choose the optimal study method. I also found that any difficulty in using the interface of the program led to reduced willingness to use and explore the application, as would be expected of almost any interface. However, I did find that statistics, games, and feedback increased interest in practice and use of the apps.

Project Description

As mentioned above, the planned application is for a flashcard based application to aid students in learning individual language vocabulary pairs. Based on the background research, the application should aid users in retention by provided a method to track the learning progress of study items. Users should be encouraged to study items at the optimal retention time, and be provided feedback with some delay. This will require the development of an algorithm to apply to each learning item. I am hoping to modify the aforementioned algorithm used by Palvik. It is very important that users study and recall items at the optimal rate, to achieve optimal retention. This type of feedback was not seen in the flashcard applications studied, making this feature a relative novelty among these types of applications.

The research from Pham along with my own informal research, shows the need for an adaptive and interactive user interface that will sufficiently pique user interest and help with retention. The plan is to use a clean, flat interface, incorporating the general concepts of interface design I have learned through my coursework at Athabasca University. The main constraint in this area will be the screen size available for the application, since it is intended for mobile use. Screens on mobile devices seem to be trending towards larger dimensions and higher resolutions, but screen real estate is still a significant limiting factor. Yet, as again shown by Pham, the flashcard interface is highly adaptable and works well for mobile devices. I hope to adopt several aspects of the PACARD system used by Pham, as well as lessons learned from the other applications studied. Below are some wire-frame designs to serve as the base model for the interface:



These wire-frames are only intended as outlines to begin the project. Design of the interface may need to be changed in several ways to accommodate certain features, and to enhance retention and learning.

Materials

Based on the outline above and the general needs of other software development projects, it is clear that few materials are needed to complete the project. Here is a list of required materials, all of which are currently in my possession:

Item	Need	Option
Modeling software	for planning	Visual Paradigm
Android SDK	for development	n/a
IDE for Android SDK	for development	Android Studio
Image software	for UI development	Photoshop
Productivity software	for additional materials development	LibreOffice suite
Computer	for running planning/dev software	Asus K401L Laptop
Android tablet/phone	for testing app	HTC Nexus 9/LG Prada

Additional materials may be required as the planning stages progress or additional features and content are added to the application.

Methods

Since this project is being developed in its entirety by one individual, the method of development can be relatively simple and planning doesn't not need to be overly thorough. There is no team which requires dissemination of information, and no disparate code from separate teams that will need to be integrated into the overall project. To this end, a modified version of the standard waterfall software development model shall be used to work through the various stages of development. However, adherence to a strict waterfall method does not allow for much innovation or adaption in the middle of development. Therefore, a modified waterfall method will be used, allowing for more flexibility in development, allowing movement between the different stages to adapt to change. Yet overall, the waterfall structure will still guide the flow of development.

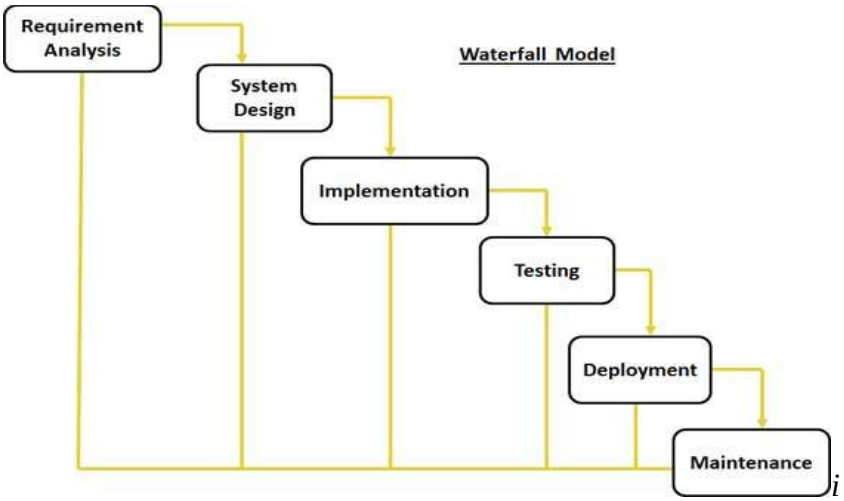


image courtesy of tutorialspoint.com

This document itself will act as the basis for the requirements in the model. The system will be designed and modeled using various UML diagrams, such as user interaction diagrams, etc. The final design steps can be solidified once the final draft of the document has been submitted and accepted.

After design and modeling, implementation will be carried out using the Android SDK and one of the aforementioned IDE's. Here is where strict adherence to the waterfall model will begin to break down. Implementation will probably be done in a series of parts. However, this is only an assumption as the requirements and design are still in very early stages. Individual parts will be tested and as they are created, allowing further modes/features to be added to the application. Some of these additions may require re-visiting the earlier design or even requirements stages to properly integrate these aspects into the final application.

After implementation and initial testing is complete, the application will be partially deployed for further testing. The app will be tested with a few users, mostly friends in the area, with questionnaires developed to elicit feedback and criticism. The exact process for this testing phase will be developed in the design section. Any changes necessary after this will require revisiting of the previous development steps to result in further implementation and testing. Finally, the application will be considered "deployed" when it has been submitted for approval and grading for the course, along with other relevant materials. At this time, there will be no planned maintenance for the application as it will be unnecessary for completion of the course.

Schedule & Deliverables

Here is the rough proposed development schedule for the project, expecting to take place over a period of 17 weeks. This process will inevitably extend beyond the course end date of 30/06/2017. However, the course has been extended for the standard two month period, and the contracted end date is now 31/08/2017. (Note: milestones highlighted in red text):

Week	Date	Tasks to Complete
0	11/27/2016 to 05/07/2017	1. Background scholarly research 2. Existing market research 3. Develop requirements 4. Develop completion schedule 5. Complete proposal document 6. Familiarization with IDE and SDK software
1	05/14/2017	1. Submit proposal document 2. Create bi-weekly reports for future submission 3. Begin design phase 4. Begin development of UML diagrams
2	05/21/2017	1. Update and resubmit proposal document based of feedback (if necessary) 2. Continue work on UML diagrams 3. Re-familiarize with IDE and SDK 4. Begin work to develop visual assets for use in UI
3	05/28/2017	1. Complete UML and design documents 2. Begin implementation phase 3. Begin coding of basic layout and elements based on UML diagrams 4. Continue development of UI assets
4	06/04/2017	1. Continue development of UI assets 2. Continue coding of basic elements

Week	Date	Tasks to Complete
5	06/11/2017	1. Continue coding of basic elements
6	06/18/2017	1. Finish coding of basic elements/features 2. Update UML diagrams to include additional features 3. Begin coding of additional features/elements
7	06/25/2017	1. Continue coding of additional elements
8	07/02/2017	1. Finish coding of additional elements 2. Integrate UI assets into application
9	07/09/2017	1. Finish coding and integration of all features and UI
10	07/16/2017	1. Begin testing phase 2. Develop test plan
11	07/23/2017	1. Execution of test plan
12	07/30/2017	1. Finish first test phase 2. Make necessary changes based on test results 3. Revise test plan as necessary with new details
13	08/06/2017	1. Re-test application with final changes 2. Finish initial testing phase and write up test results 3. Develop evaluation requirements for user testing
14	08/13/2017	1. Conduct user evaluations
15	08/20/2017	1. Finish evaluations and write up evaluation results 2. Make necessary changes based on user evaluations
16	08/27/2017	1. Begin deployment phase 2. Polish and add final touches to application 3. Complete final report 4. Submit all deliverables for course grading

Development is expected to take place through several South American countries as I travel. Due to limitations of internet access in certain countries, as well as delays and other impacts from the nature of travel, the minimum work per week can only be expected at about 10 hours per week. More work hours can be added to the minimum when time and situation permit. Additionally, as the project nears completion, my traveling is also expected to decrease, in turn allowing the amount of weekly hours to increase as well.

Naturally, the schedule will also be subject to change as development progresses. Changes will be reflected in the bi-weekly reports submitted during development. The first bi-weekly report will be handed in at the end of week 2, as the previous weeks consist only of background and preparation before the project begins. Bi-weekly reports will include hours worked and tasks worked on and/or completed. Updates will be made to the following 2 week schedule to reflect the progress made.

At the end of development, and when the final project has been submitted for marking, the

following items will be delivered as a part of the final package:

1. Proposal document
2. Set of completed bi-weekly reports
3. UML diagrams and other design documents
4. Completed application, runnable on an Android system
5. Complete source code for completed application
6. Testing and user evaluation results
7. Project final report

Once all deliverables are submitted, the project will be considered complete. Marking and feedback will be given by the course instructor.

Conclusion

The development of the application is aimed at creating a flashcard application that is simple and appealing to users. Following the research, it will be designed to aid users in retention of items through spaced repetition, and increasing time between tests. During these tests, feedback will be delayed until the end. The research discussed clearly shows the benefits of these three factors. Additionally, the app will be designed to maximize the best features of the other applications on the market, combining them into a very useful and approachable user interface and user experience. Using the methods and schedule outlined, the project should turn out to be a success.

References

Cram LLC (2016). Cram.com Flashcards (Version 1.6.1) [Software].
Available from https://play.google.com/store/apps/details?id=com.studymode.cram.

Duolingo (2016). DuoLingo (Version 3.36.0) [Software].
Available from https://play.google.com/store/apps/details?id=com.duolingo.

Duolingo (n.d.). *About us*. Retrieved from https://www.duolingo.com/info.

Eich, Eric (n.d.). *The Cognitive Science of Learning Enhancement: Optimizing Long-Term Retention*.
Retrieved from http://ctlt.ubc.ca/resources/isotl/resources-archives/the-cognitive-science-of-learning-enhancement-optimizing-long-term-retention/.

Memrise Ltd (2016). Memrise (Version 2.9_3857_memrise) [Software].
Available from https://play.google.com/store/apps/details?id=com.memrise.android.memrisecompanion.

Memrise Ltd. (n.d.). *Science*. Retrieved from http://www.memrise.com/science/.

Neeley, Tsedal (2012, May). Global Business Speaks English. *Harvard Business Review*.
Retrieved from https://hbr.org/2012/05/global-business-speaks-english.

Palvik, P. I., & Anderson, J. R. (2008). Using a model to compute the optimal schedule of practice.
Journal Of Experimental Psychology: Applied, 14(2), 101-117. doi:10.1037/1076-898X.14.2.101

Pham, X., Chen, G., Nguyen, T., & Hwang, W. (2016). Card-based design combined with spaced repetition: A new interface for displaying learning elements and improving active recall.
Computers & Education, 98142-156. doi:10.1016/j.compedu.2016.03.014

Quizlet LLC (2016). Quizlet Flashcards & Learning (Version 2.5.2) [Software].
Available from https://play.google.com/store/apps/details?id=com.quizlet.quizletandroid.

Rosetta Stone Ltd. (2016). Learn Languages: Rosetta Stone (Version 2.4.1) [Software].
Available from https://play.google.com/store/apps/details?id=air.com.rosettastone.mobile.CoursePlayer.