Math Tutor: An Interactive Android-Based Numeracy Application for Primary Education

Zainab Masood **Rashina Hoda**
SEPTA Research, Electrical and Computer Engineering
The University of Auckland
Private Bag 92019, Auckland, New Zealand

zmas690@aucklanduni.ac.nz r.hoda@auckland.ac.nz

Abstract

With growing exposure of children to handheld and mobile devices, there is an increasing interest in exploring the use of mobile technology for educational purposes. In particular, touch-based devices seem to promise great potential in this domain. In this paper, we present Math Tutor – an Android-based application designed to help children learn and practice early numeracy addition and subtraction (take away) as well as help teachers monitor and review children's progress, with support for English and Māori languages. We describe the design and development process, features of the application, and the results of a usability evaluation. This project takes a step towards creating interactive platforms required for educating the upcoming generation of digital natives.

Keywords: Interactive software applications, primary education, numeracy, child-computer interaction, Android, Māori

1 Introduction

In the last few years, mobile technology has changed faster than ever in ways that affect our lives in every aspect. Children, in particular, who are born or growing up in this 'brave NUI world' (Wigdor and Wixon 2011) are considered digital natives (Palfrey et. al 2008). It's not unusual to find children manipulating mobile devices even before they've mastered their alphabets and numbers. Touch-based interactive devices being easy to use by children as well as the teachers are considered important and handy tools for teaching core concepts such as numeracy in early and primary years learning.

There have been variety of effective traditional ways in which children are taught numeracy in classrooms, but research suggests that children are more engaged and grasp concepts quickly when they enjoy the subject matter being taught. At the same time, children are more attracted to colourful and interactive applications – aspects weakly supported by traditional white-board based teaching approaches.

The aim of our project was to conduct research and development into this upcoming area of inquiry.

Our Android-based application - *Math Tutor* - is targeted for classroom teaching and been designed specifically to be used both by students and teachers. A part of our application is designed to help students learn and practice single digit addition and subtraction using numbers and images. The application's GUI and colours have been carefully chosen as to attract children and make learning easy and fun. The children are encouraged to learn by using the simple technique of earning medals when they complete a particular level. The second part of the application has been designed for the teachers and includes functionalities such as language selection, exercise mode selection, etc. (described later in detail). The functionality for teachers to login and perform administrative activities can be easily found in web-based applications but is currently hard to find in tablet-based applications. There have been other touch-based applications and games designed to teach children early single digit addition and take away, but unfortunately we find very few applications that associate numbers with (equal number of) objects visually and have multi-lingual options. Our society being a multicultural society consists of people belonging to different parts of the world who speak more than one language. As such, many children take some time learn to speak and understand English. So the teachers may find it difficult to teach addition and take away with applications that just have English as a language option. For this reason the second language chosen for this application is Māori, which is one of the three official languages for the people living in New Zealand and is a core part of early and primary school education. This prototype can be expanded to include other languages with relative ease.

The intended users of this application are children between five and six years of age and the teachers who will be assisting and using the application to manage students. A student can practice addition and take away for up to two levels i.e. for numbers from 1 to 5 and between 1 to 10. The application uses two techniques used for addition and taking away, first using simple single digits e.g. 3+2=5 and second using the visual concept of adding/taking away things with counting equal number of objects like fruits. Children can use the objects (e.g. fruit images) to count and find the answers. Upon successful completion, children will earn different coloured star medals as rewards that will encourage them to practice more and improve their skills.

The application tracks student responses. Results, including number of correct and incorrect responses with percentage and medals are available for teachers for further analysis and evaluation.

2 Related Works

Unique capabilities of computer applications for providing learning practice include: the combination of visual displays, animated graphics and speech; the ability to provide feedback and keep a variety of records; and the opportunity to explore a situation interactively (Clements 2002). A study by Starkey et. al (2004) found that children can develop mathematical competence with developmentally appropriate mathematics learning software. Another longitudinal study examining the role that technology plays in mathematical skill achievement for kindergartners and 1st graders shows that students' access to and use of technology influences their future academic achievement in school (Espinosa et. al 2006).

Year one (at school) marks a transition to a more academically-oriented approach to learning as children may now be in a full program after half-day long kindergarten. In New Zealand, numeracy instruction, along with other subjects, becomes more structured in school as opposed to a more flexible framework offered in kindergarten by the *Te Whā riki* (curriculum) (2007). Lessons are more structured, and there are new facts to master. After acquiring the basic skills of number recognition, counting, addition and subtraction are naturally taught next, as students combine sets of objects and count the results.

It has been argued that carefully designed sequences of worked-out examples (without lectures or other direct instruction) can prove to be more effective in teaching several mathematical skills (Price et. al 2013). The role of drills has been emphasized in achieving the fluency to memorize basic mathematical facts and improve these skills. The use of this technique not only seen to help in achieving fluency in mathematical knowledge but also expected to aid in higher mathematics skills as per research (Zhu et. al 1987). However, children can find these math drills as repetitive, monotonous and boring exercises.

Trends suggest that pre-school and elementary-age children may soon be using smart devices seamlessly first at home and then perhaps in the classroom of 2015 as a normal part of growing up in a digital age (Chiong and Shuler 2010). Therefore, we need to focus on designing useful and productive applications which not only educate children but are also fun to use. Norris et. al compared the use of tablets with netbooks and laptops and concluded that features like portability, availability, instant-on devices, and longer battery time makes tablets a preferred medium over others for learning (Norris et. al 2011).

One of the most popular smart phone technologies today is Android. Android has witnessed a growth of over 250% with 300 million android devices activated, adding 850,000 new device activations every day (Rubin 2012). All those new Android users also have access to 450,000 applications on the Android market, an increase of 300% from last year (Rubin 2012). And this growth rate continues to grow along with the android applications available for users of all age. Android users have access to numerous applications on the Android market through Google Play (https://play.google.com). These include numerous learning applications for primary years.

Some such educational android applications which are available on Google Play to help children improve numeracy skills include: Kids Math, Math Mole, Math flash cards, Kids Math Book, First Grade Math Challenge and 1st Grade Math.

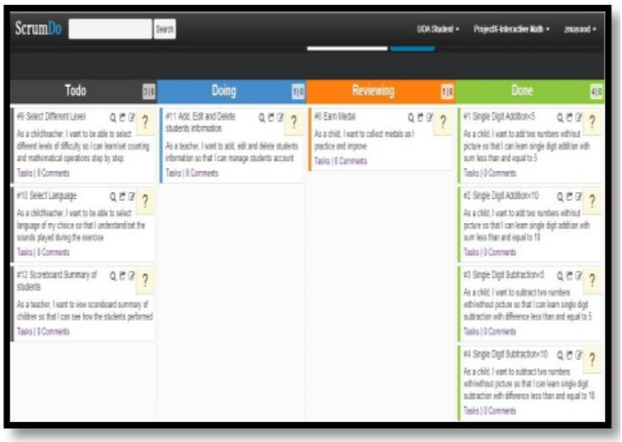

Figure 1. ScrumDo Project Management Board

While possessing different functionalities in parts, most of these applications are focused on individual users (parents and children at home) and do not offer comprehensive solutions for classroom-based teaching, such as teacher access and administration.

3 Project Design

3.1 Software Development Method

We used Agile software development as our software development methodology for this project (Cockburn and Highsmith 2001). Agile methods are based on an iterative and incremental style of development where each iteration involves set of tasks to be accomplished during the iteration. Agile software development methods are known to be flexible to change and are based on changing needs and preferences. In particular, we employed Scrum – one of the most popular Agile methods in the world today (Schwaber and Beedle 2002).

For this project, a roadmap was created to plan the activities and the timelines based on the user stories created using ScrumDo as the tool (http://www.scrumdo.com). Functionalities represented as user stories were created to help visualise the system and streamline the requirements of the application. Then a set of objectives were defined for the user stories during weekly scrum/stand-up meetings scheduled with the project supervisor (second author) to share progress and plan activities of the week. The meeting focused on what was done during last week, what was the plan for the coming week, and what was getting in way i.e. the impediments. The scrum board was updated accordingly with status of user stories using ScrumDo and iterations were updated as well.

Based on this, tasks for next iteration were assigned and were expected to be delivered before the next meeting. For any queries during the week, the coordination was done through emails. It helped to ensure that the project was on the right track with status of final deliverables.

3.2 Software Tools

3.2.1 User Interface Design Tools

For designing the user interface of the application, a custom set of images were created and some free images were chosen and modified including icons for the application, buttons and background. Adobe Photoshop and Adobe Illustrator were used as image customisation and processing tools.

3.2.2 The Android Framework

Figure 2. Samsung Galaxy Tablet (front/back) and Emulator (left)

The Android framework was chosen as the platform for the development of the application primarily due to its popularity and relative affordability of tablet devices supporting Android (as compared to iPad for example.) Eclipse was used as the Integrated Development Environment (IDE) with Java as the programming language. Android 14 was used as the version of Software Development Kit (SDK) for compatibility with the android device selected for this project: Samsung Galaxy 7 inches tablet and Android OS 4.1 (Jelly Bean).

3.2.3 Testing and Validation Tools

Unit tests were implemented to verify the correctness of the methods that were developed using the LogCat and the associated Log class provided by Android. Also, a usability evaluation was conducted to verify the usability of the android application with users, which is described later along with results and refinements resulting from the evaluations. Due to time limitations of this project, evaluations were conducted using university students rather than ideal users – children and teachers – which required lengthy ethics approval processes. This aspect is further discussed in the limitations section later.

3.3 Software Design Decisions

Certain critical software design decisions were made to enhance the application's usability based relevant research. Research was conducted to select the best design strategies to apply during the design and development of the application for young children. As proved by research, generally the more a software interface adapts to children's cognitive levels, the more it will be accepted and adored by children (Xiaodong 2010). Similarly, using bright, vivid colours that stimulate the senses, creating a happy, playful mood using cartoon characters (Lazaris, 2012), making icons look "clickable" by using three-dimensional imagery, adding navigational elements that are large and easy to find (Hafit et. al 2012) are the best practices which should be used for designing applications for children.

Also interface with much word information results in the attention detraction for children and turbulence of vision procedure, so the principal interface i.e. the exercises screens were designed to be simple. However, other screens are designed to include colours and imagery in such a way that the children enjoy and love using the application.

For children, it is the physical design (in this case, the user interfaces) that appeals to them; and this factor is perhaps as important as the actual functionality provided by the application. Children tend to choose the application that features characters they can relate to and those that appeal to them aesthetically. Also, while designing the prototypes the strategies from the research conducted were incorporated. The designed interfaces catering to students and teachers are presented in Fig.3 and Fig.4 respectively.

Figure 3. Design Prototypes for Student Functionalities

The following sections describe the application key features, the application design, and application implementation with the details of the iterations followed from the beginning towards the end of project.

4 Math Tutor: Key Features

The actual implementation of the application focused on the major features: Child Learner Exercises (Addition and Take Away), Reward System, Teacher Administration, Language Setting, and Exercise Mode Settings.

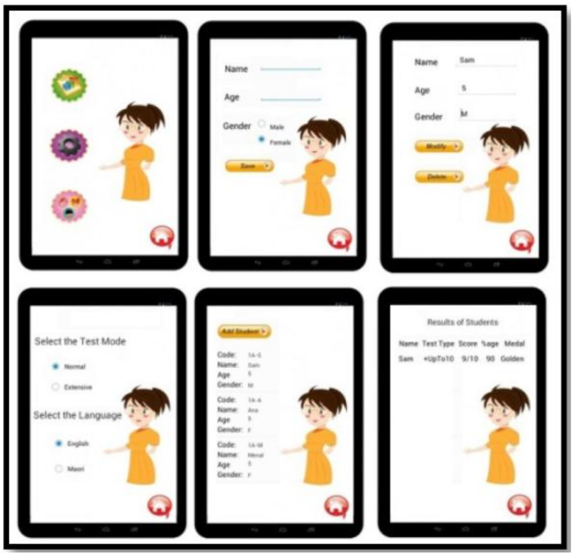

Figure 4. Design Prototypes of Teacher Functionalities

4.1.1 Child Learner Exercises (Add/Takeaway)

When the child enters the student menu screen (Fig. 3) and selects the exercise type as *Add up to 5 or 10* or *Take Away up to 5 or 10*, then a set of questions are dynamically generated randomly with the fruit images and corresponding numbers. The child has to answer each question by clicking one of the answer options displayed. If the answer is correct, a sound is played to help the child associate the answer with the sound as correct; and another sound for incorrect response. Once all the questions are displayed the child is prompted with either moving to the scoreboard or to retry the exercise.

4.1.2 Reward System

When the child selects to move to the scoreboard the results of the exercise can be viewed with the earned medal based on the percentage achieved.

Four different medals are rewarded to students as shown in Fig.5. If the percentage achieved is greater than 80% a golden medal is earned by the student, if the percentage is greater than 65% a silver medal is earned by the student. Similarly if the percentage is greater than 50% then silver medal is earned by the student and red medal if the percentage is less than 50%.

4.1.3 Teacher Administration

The functionality of this part includes choosing the language for the audio outputs (English or Māori) and selecting the exercise mode from normal to exhaustive. The functionality also includes adding, deleting and modifying student's details. Teachers can view every student's scores during different practice exercises and compare them with the previous results.

4.1.4 Language Settings

A teacher can use this screen to select between the two languages English and Māori. English is used as the default language. However if the user wants to change the language, radio button can be used for switching the mode of the language.

Figure 5. Scorecard (top) and Reward medals (bottom)

4.1.5 Exercise Mode Settings

A teacher can use this screen to select the questions mode as normal or exhaustive. This is set as 10 for the normal mode, i.e. the number of questions for exercise is 10.

5 Application Design: LoFi Prototyping

To start the project first the user stories were created. The application was designed keeping in mind the age of the end users which are children from age's 5-6. The design was kept fairly simple and easy to understand by the children. The design was also made colourful and attractive so that the children find it enjoyable. The navigation was purposely kept simple and not over-crowded with too many options as children can get confused if they are given a lot of options to move forward, backward and to the main menu. The colours for the overall design were carefully chosen and kept consistent throughout the application.

We created low-fidelity paper prototypes to get an understanding of the application flow and design (Kangas and Kinnunen 2005, Virzi et. al 1996). These prototypes were transformed to digital black and white mock-ups using *Lucidchart* (https://www.lucidchart.com) as a wireframe tool (Fig. 6)

Figure 6. Paper (left) and Wireframe (right) Prototypes

We focused on improving the application prototypes by adding colours and pictures. Since the end-users of the application are children so care was taken to design with children's interest in mind.

6 Application Development

Following Scrum's iterative and incremental, the development was executed in a series of iterations. With each of these iterations, the design and development were assessed and refined, while moving closer to completion. In the following sub-sections, we describe the activities and key design aspects achieved in the iteration providing a sense of the evolving application.

6.1 Iteration 1

The development of Android applications involves some installation pre-requisites like the installation Eclipse SDK, Android SDK and Android Development Tools (ADT) to get a running environment. For developers, Android SDK provides a rich set of tools, including debugger, libraries, handset emulator, documentation, sample code, and tutorials. Android applications can be easily developed using Eclipse (Android's official development platform) with the help of a plug-in called Android Development Tools (ADT).

The development phase consists majorly of designing and developing activities. A concise definition of activity as stated in the official help for eclipse states "An activity is a logical grouping of functionality that is centred around a certain kind of task." Hence an activity is a group of sub-classes designed and developed to create the interface and add functionality to them.

The first development iteration involved implementing the user stories of the students. The addition of numbers up to 5 was programmed initially by generating different questions randomly using the built in random number function and against every questions four options were randomly generated correspondingly. For this a java class was created for random generation of numbers and method of the class was called in the addition activity.

This activity generated questions with two random numbers such that there sum is less than or equal to five. Then four options were generated and displayed on buttons with one as the sum of the two numbers and the other three options as random numbers other than the result and less than and equal to 5. This was called recursively till the number of questions is equal to mode selected. To ensure that the options are not displayed always on same position, the position is randomly relocated by generating different sequence of locations (Fig. 7)

6.2 Iteration 2

The second iteration involved addition of numbers up to 10. For this the addition class was modified to handle the random numbers generation dynamically based on the selected exercise type. Also the take away class was implemented following the same pattern. This iteration also involved the modification of `AddActivity` and `SubtractActivity` to make it more generalized for all exercises. For this purpose the referring activity passes the exercise type and `Add /SubtractActivity` then first extracts the information from the Bundle class to get the intent variables and check the maximum value of `maxnumber`. The `maxnumber` decides which type of exercise has to be started. For example if `max=5` the 5 digit addition is started.

```
Step1:Generate RandomSequence for displaying options randomly on buttons
    int optionSeq=ran.generateRandomOption()
Step2:Convert random sequence into string for conditional comparison
    StringselectedoptionSeqStr =Integer.toString(optionSeq)
Step3:Display options randomly on buttons
    if selectedoptionSeqStr equals firstsequence{
            firstbuttontext= resultStr
            secondbuttontext= option1Str
            thirdbuttontext = option3Str}// Display in this sequence
        else if selectedoptionSeqStr equals secondsequence
            Display text on buttons in different sequence
        else if selectedoptionSeqStr equals thirdsequence
            Display text on buttons in different sequence
```

Figure 7 Pseudo-code for Options Relocation

6.3 Iteration 3

In the third iteration, the objects (e.g. fruits) were displayed alongside the numbers to help student to calculate the correct answer. The results were calculated and the scores were created and displayed on the scorecard with the medal earned. During this iteration the orientation was fixed so that while using the tablet if the orientation is changed then the application is not affected. For Android, this is neither an automatic functional and needs to be implemented. Sounds (audio files) were added against the numbers in the English language.

6.4 Iteration 4

In this iteration, the user stories pertaining to the teacher-user were implemented. User Management classes were created to add and delete the students. Then the settings class was added as a separate activity to configure the language and the exercise mode and Māori audio files were added for supporting two languages to the application.

This iteration also consisted of designing the database and the schema with the students and score table in the database. The Score table is used to save results after each exercise. The iteration used SQLite database as the base class for working with a SQLite database in Android which provided methods to open, query, update and close the database. In addition it provides the execSQL() method, which allows to execute an SQL statement directly. The object ContentValues allows to define key/values. The "key" represents the table column identifier and the "value" represents the content for the table record in this column. ContentValues was used for inserts and updates of database entries. This was implemented as a dataAdapter class. Also a java class was added to pass the values from the business logic to the database.

The major functions of the dbAdapter included creating the student, fetching the students in the student section, updating and deleting the selected students, saving the scores of the student and fetching the scores against the student. This iteration also consisted of using the List View layout to display list of all student in the student's layout.

6.5 Iteration 5

In the last iteration both the modules were integrated with implementation of all the user stories and the scores were stored in the database against the student. In the end the results class was added so that teacher could view the results of the students saved in the database using the dataAdaptor class and displaying using ListView layout in the teacher section.

Every iteration was followed by the functional testing of the features added using the positive and negative test data to ensure the correctness of the application and in the end after the integration of the modules, the overall application was verified and the discrepancies were fixed and retested accordingly.

The upcoming section will provide a complete analysis of the evaluation results performed after the completion of this application.

7 Usability Evaluation and Results

An evaluation was conducted to see how effective and useful the different features are and how suitable the user interface design is to the users. The evaluations took place in The University of Auckland, Engineering Building, Masters Room.

7.1 Participants

The participants for this evaluation were seven university students with ages between 20-29 years. The application was installed on the Android emulator device and all the participants were asked to run and test the application on the same settings of emulator as of Samsung Galaxy 7 inches Tablet.

7.2 Evaluation Method

The evaluation of the application was conducted using the Android emulator configured on personal laptop. Each user was first given written instructions to follow after providing the demographic details. This document listed a small description of the tasks to be performed with the steps of the execution procedure both for student and teacher's role.

The main tasks as a student included completing the addition and takeaway exercises for all the four exercises with normal mode setting, two exercises with language as English and two with Māori as language. The tasks to be completed while evaluating the teacher module were to add/update and delete student information and to view the score of the students, doing the configurations like setting the language for the exercise.

The participants were first asked to fill in the demographic section of the questionnaire. Then after the execution of the application the usability and usefulness sections were filled by the evaluator. This questionnaire focused on the effectiveness and usability of the application. The multiple choices consisted of statements in a Likert scale having values between 1 and 5, where strongly disagree equals 1 and strongly agree equals 5.

7.3 Results

We now present the complete analysis and results obtained after performing the evaluation.

7.3.1 Demographics

The results indicate that all participants had intermediate to expert level experience in using touch based devices. A majority (85.5%) of participants have observed children using touch-based devices. Nearly all the participants agreed that these applications can add fun to learning mathematics. All of the participants agreed that they have used numerical practice sheets in their primary years learning.

7.3.2 Usefulness of Application

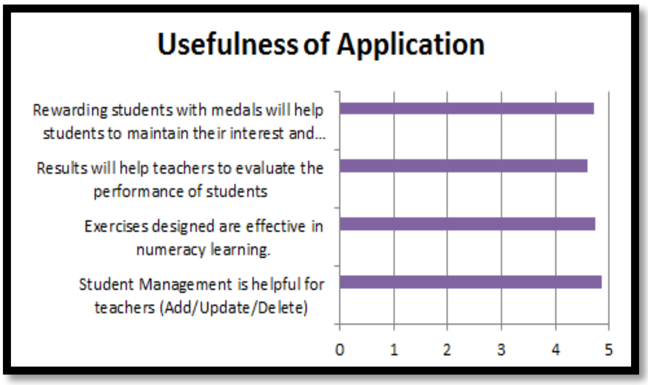

Figure 8 Mean values for Usefulness Questions

The results obtained during the evaluation procedure were used to calculate likert scale values. A median for each option was calculated using the weighted mean formula for likert scale. The results are plotted in the graphs below.

The likert scale value of 4.86 for the student update/delete value shows that almost all the participants believed that the student management module for teacher will be helpful for maintaining students. The value 4.6 in the graph below indicates that participants agree that displayed results will help teacher to monitor student's progress in a classroom. Participants agreed that the exercises designed are effective in numeracy learning with value of 4.75 and rewarding students with medals gets a 4.71 Likert scale. The results show that the participants agree that the application can actually prove to be a handy and effective tool in classroom teaching.

7.3.3 Usability of Application

The results of the usability evaluation are presented in the graph below which indicates that changing language was easy for users (4.86). All other areas including consistency in flow of application, ease in application usage and clarity of sound reached values approximately between 4.7 and 4.8 which supports that these aspects of application were approved by the evaluators.

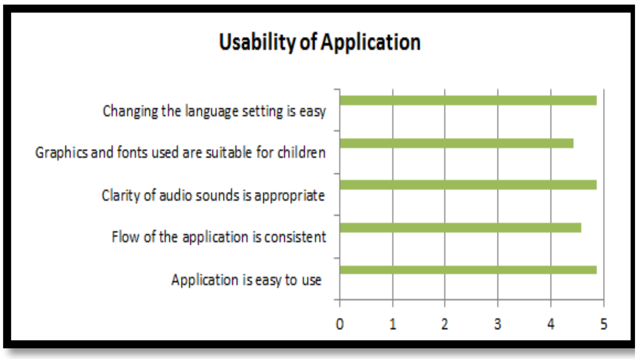

Figure 9 Mean values for Usability Questions

7.4 Suggested Refinements

The results and comments after the evaluation suggest some improvements in the application in the future.

- In the teachers section if the student names were displayed in blocks and detail of each student was displayed separately would make it easier for the teacher to view each student's details.
- In teachers sections some charts and graphs would help teacher to analyse student progress more effectively.
- If the teachers have the option to email/share the progress of a student with parents or guardians will be a helpful feature.
- More languages options such as Mandarin, Spanish, German or French can be introduced for the users belonging to different regions of world.
- At the end of each correct answer different and more interesting sounds like clapping or music could be added to make children happy and excited.

The next section will provide a general discussion and concluding remarks for the project.

8 Discussion

The application was designed and developed to help children between 5-6 years of age learn basic addition/takeaway using touch-tablet devices and help teachers manage students and monitor student progress. The results of a preliminary usability evaluation indicate that the application can be potentially effective in teaching children about basic numeracy concepts (addition and takeaway up to 5 and 10). The teacher section also promises to be used a useful classroom tool that can be used to add/update and delete students in their class list. They can view and compare the results from previous exercises and decide which students need help and more practice.

The application could not be evaluated with the actual users i.e. children aging between 5 to 6 and teachers practically in a classroom setting due to limitations of time as the university's ethics policy application process takes some months. All the participants involved in the evaluation of the application agreed that the application with some minor improvements could be a very good source for teaching a 5-6 year old children addition and take away. The evaluations if conducted using the ideal user-groups i.e. teachers and children, could be used to draw firmer conclusions about the role of this application in primary numeracy learning.

9 Conclusion

The changing trends in technology and classroom teaching suggest that touch-based mobile devices and attractive/engaging applications can be essential parts of future classrooms. The use of such devices amongst children is increasing by leaps and bounds. However, research and development of educational software that is both engaging and appropriate for classroom usage is still in its infancy. Our Math Tutor application can serve to be a good foundation for more complicated and sophisticated learning tools in future. Extension to two and three digits addition and taking away can be added in this application to be used by children ages 6 and above. The teacher section can include graphical analysis to monitor student performance with option to share student's progress report with parent's guardians as suggested by some participants. The application can also be extended to support more languages. Overall, our project is a step towards creating interactive educational software for use by digital natives in classrooms of the near future.

10 References

Ministry of Education. The New Zealand curriculum. Wellington, Learning Media, (2007) http://www.educate.ece.govt.nz/learning/curriculumAndLearning/TeWhariki.aspx

Wigdor, D. and Wixon, D. (2011): *Brave NUI World: designing natural user interfaces for touch and gesture*. Elsevier.

Palfrey, J. G., and Urs Gasser. (2008): *Born digital: Understanding the first generation of digital natives*. Basic Books.

Clements, D. H. (2002): Computers in Early Childhood Mathematics. *In Contemporary Issues in Early Childhood*, **3**(2): 160-181.

Starkey, P., Klein, A., and Wakeley, P. (2004): Enhancing young children's mathematical knowledge through a pre-kindergarten mathematics intervention. *In Early Childhood Research Quarterly*, **19**: 99–120.

Espinosa, L. M., Laffey, J. M., Whittaker, T., and Shcng, Y. (2006): Technology in the home and the achievement of young children: in Findings from the early childhood longitudinal study. *In Early Education & Development*, **17**: 421-441.

Price, G. R., Mazzocco, M. and Ansari, D. (2013): Why Mental Arithmetic Counts: Brain Activation during Single Digit Arithmetic Predicts High School Math Scores, *Journal of Neuroscience*, **33**(1):156-163.

Zhu, X. and Simon, Herbert A. (1987): Learning Mathematics from Examples and by Doing. *In Cognition and Instruction*, **4**(3): 137-166.

Chiong, C. and Shuler, C. (2010): Learning: Is there an app for that? URL: http://www.joanganzcooneycenter.org/wp-content/uploads/2010/10/learningapps_final_110410.pdf

Norris, C., Hossain, A. and Soloway, E. (2011): Using Smartphones as Essential Tools for Learning. *In Educational Technology*, **51**(3): 18-25.

Rubin, A. (2012): Android@Mobile World Congress: It's all about the ecosystem. [Online]. Available: http://googlemobile.blogspot.co.nz/2012/02/androidmobile-world-congress-its-all.html

Cockburn, A., Highsmith. J. (2001): Agile software development: the people factor. *Computer* **34**(11):131–133

Schwaber, K. and Beedle, M. (2002): *Agile Software Development with SCRUM*. Prentice-Hall.

Xiaodong P. (2010): Research of iPhone Application VI Design Based on Children Cognition Feature. *In Computer-Aided Industrial Design & Conceptual Design (CAIDCD), IEEE 11th International Conference*.

Hafit, H., Abdul Razak, F. (2012): Drawing with Fun: What We Can Learn From the Children [Online]. Lazaris, L. (2012): Designing Websites for Kids: Trends and Best Practices. http://www.smashingmagazine.com/2009/11/27/designing-websites-for-kids-trends-and-best-practices. Last accessed on 23rd August, 2013.

Available: http://www.designingforchildren.net/papers/h-hafit-designingforchildren.pdf

Kangas, E. and Kinnunen, T. (2005): Applying User-Centered Design to Mobile Application Development. *Communications of the ACM*, **48**(7), ACM.

Virzi, R. A. Sokolov, J. L., Karis, D. (1996): Usability Problem Identification Using Both Low and High-Fidelity Prototypes. *In Proceedings of Human Factors in Computing Systems Conference CHI'96*, Vancouver, BC, Canada.