SCHOOL OF SCIENCE AND ENGINEERING

Capstone Project Final Report

Houda SALHI

Supervised by:

Dr. Hanaa Talei

30th, April 2019

Spring 2019
# TABLE OF CONTENT

Acknowledgements........................................................................................................5  
Table of figures..................................................................................................................6  
Abstract.............................................................................................................................7  
1. Introduction ...................................................................................................................8  
2. Steeple Analysis.............................................................................................................9  
   2.1 Socio-Cultural...........................................................................................................10  
   2.2 Technological..........................................................................................................10  
   2.3 Economic................................................................................................................10  
   2.4 Environmental.......................................................................................................10  
   2.5 Political..................................................................................................................10  
   2.6 Legal.......................................................................................................................11  
   2.7 Ethical....................................................................................................................11  
3. Methodology and Development Process....................................................................11  
   3.1 Feasibility Study.....................................................................................................11  
   3.2 Requirement Specification.....................................................................................13  
      3.2.1 Functional Requirements.............................................................................14  
      3.2.2 Non-Functional Requirements.................................................................15  
      3.2.3 ER Diagram.................................................................................................17  
   3.3 Project Design and Development.........................................................................18  
      3.3.1 Technology Enablers...................................................................................18
Capstone Report

Student Statement:
I Houda Salhi affirm that I have applied ethic rules in the design and the development of WE CAN2 android application. Additionally, I confirm holding my application user’s safety and wellbeing in the design of my project as a primary concern.

___________________________
Houda Salhi

Approved by the Supervisor

___________________________
Dr. Hanaa Talei
AKNOWLEDGEMENTS

Firstly, I would love to take this chance to thank Dr. Hanaa Talei for granting me the opportunity to be her supervisee for this project. For reminding me to keep working hard and giving me advice and guidance in the right direction. I would like to thank her for her time, and research effort, and constant level of seriousness and support.

Secondly, I would like to thank my parents and my siblings who have been a great support throughout this project. For answering my questions that were not always necessarily easy questions about the implications of attention deficit hyperactivity disorder, and for testing my prototypes and giving me good constructive feedback through every step.

Finally, I would like to thank my university Al akhawayn. For providing me with the right learning environment and the right tools to be able to flourish as a developer.
TABLE OF FIGURES

Figure 1 Android Logo.................................................................12
Figure 2 SQLite vs Firebase..........................................................12
Figure 3 Project Time table..........................................................13
Figure 4 WE CAN 2 use case Diagram........................................15
Figure 5 WE CAN 2 ER Diagram..................................................17
Figure 6 Android studio Logo......................................................18
Figure 7 Java Logo.................................................................18
Figure 8 Photoshop Logo..........................................................19
Figure 9 Subtraction Hint Design...............................................19
Figure 10 Addition Hint Design..................................................19
Figure 11 Firebase Synchronization...........................................20
Figure 12 Authenticating a New User...........................................21
Figure 13 We Can 2 Three Tier Architecture................................21
ABSTRACT

The objective of this capstone project is to design an android application that will aid children with deficit of attention disorder, in learning Mathematics as they go through their primary school curriculum.

Mathematics is one of the hardest topics to grasp for these children in a class environment, which causes a lot of child and parental frustration. This application aims to mend the relationship between these students, their families, and school.

The first phase of the project consists of a lengthy research about the psychological state of these children. The aim of the research is to find out how their brain processes information, and what kind of data form stimulates their learning abilities and enhances them. The second phase of the project is concerned with the adequate implementation for the project. This report shows the step by step progress of the project towards building We can 2. An android application voice and visually enhanced to aid ADHD children in learning mathematics.
1. INTRODUCTION

WE CAN 2 is an android application that aims to aid the learning process of mathematics to students that suffer from a deficit of attention disorder. Although diagnosable, recognizable, and representing a major issue in the academic path of students, people suffering from ADHD are still not accounted for in schools today [1]. This condition represents a major drawback in many subjects that essentially to be understood require from the instructors the application of special care and learning tricks. Presented with a full classroom of at least 20 students, an instructor usually finds it hard to help these students flourish in a classroom environment. Which in turn leaves parents in a lot of frustration vis a vis their children, especially certain subjects like mathematics.

For this subject, I have specifically selected mathematics as a subject of focus, although the project is expandable into other subjects later on. Learning mathematics and performing mathematical computations can often be a challenge for students with ADHD. As they usually suffer from impairments in working memory, weak attention, impulsiveness, non-organization, and slower processing speed than average [3]. Unfortunately, their nature comes as a major setback in a class environment where focus and the ability to sit still is imperative to the learning experience.

The overall layout of the application should be one that takes into account the issues that these children suffer from. Features in the application should strive to replicate a one to one teaching experience that a parent or trained instructor tries to create when teaching ADHD suffering students.

Following a lengthy research about these children, their brain functioning, as well as the way to bring their attention I was able to come up with the following conditions the application should imperatively include:

- Skill set of primary school’s main learning points: Numbers, Counting, Addition,
Subtraction, Multiplication, and Division.

- Inviting game mimicking layout, that is still minimalistic in nature
- A variety of challenge levels for different practicing experiences to accommodate different age groups and levels
- Reward system: points that are given out with every cleared challenging exercise.

With every skill the main goal is to have:
1. An Audio-visual experience that draws attention. Since according to research material users need visual and audio enhancement of text material as they find it difficult to focus on text elements.
2. Endless combinations of exercises that are different every time, exercises stacked from easy to hard dependent on choice
3. Tips given if stuck on operations, with the main goal to teach a calculation method
4. Voice effect, when student gives wrong or right answer

2. STEEPLE ANALYSIS

WE CAN 2 is a debut version of a software that can be upgraded to cover more issues as future work. For one, the application can expand to aiding with the learning process of more subjects. For two, it can also become a platform that stretches to different levels and types of disabilities. An application that strives to truly allow a mixture between technology and learning.

Before diving deeper into designing and developing the application, it is necessary to learn about the macro environmental external factors surrounding the piece of software. That is where a STEEPLE analysis comes to play, as it studies multiple fields and their impact on the software.
2.1 SOCIO-CULTURAL IMPACT

The main target for this project is children with ADHD. As this specific segment of society is firstly children, and secondly suffering from a mental disability, working on this issue requires real proven techniques and skills. Working with basic assumptions could be hazardous to the children’s wellbeing. Thus, my solution is to be crafted with care and inspired by scientifically proven research, scholarly articles, and with the aid of psychologists and parents who deal with these children on a daily basis. The application should bring value to the target users, and not take away from their wellbeing.

2.2 TECHNOLOGICAL IMPACT

WE CAN 2 will take advantage of the most popular mobile operating system; Android. This will allow the application to look and feel familiar and easy to use especially to children.

With the help of Android Studio’s features, the application will be available on various android versions of mobile phones as well as tablets that run on the system as well. The application will also be easily updatable and does not take up a lot of space on the device.

2.3 ECONOMIC IMPACT

The application will be free to download, what is more the application aims to help parents spend less on private tutoring sessions for their children.

2.4 ENVIRONMENTAL IMPACT

The application works on taking learning into a new level that does not depend on the use of paper, plastic pens, and polluting substances to the environment. As it aims to make the learning and practicing experience fully digital.
2.5 POLITICAL IMPACT

WE CAN 2 does not have any political implication or impact. It is a purely academic learning platform that does not aim to sway or share opinions.

2.6 LEGAL IMPACT

No law or regulation limiting this project was found.

2.7 ETHICAL IMPACT

This project is an original project, the idea is carefully crafted so that it does not violate any ethical rule. The design is crafted using Android studio and Photoshop and is made by me personally, therefore it does not violate any plagiarism rule as well.

3. METHODOLOGY AND DEVELOPMENT PROCESS

3.1 FEASIBILITY STUDY

A feasibility study is an analysis, that studies the conditions that precede the creation of a software project. The main goal is to know if a project can be designed in the real world. It is the very first step of the software engineering process, throughout this step it is important to be discuss the technical, economic, legal, and scheduling feasibility of the project WE CAN 2.
Technical

In evaluating the technical enablers of the project, we need to evaluate the different platforms, languages, and tools that will enable this project to come to life.

Firstly, the platform I chose for the project is Android, as it constitutes of the larger portion of users in the world. Additionally, Android has a wider variety of open source material and support in the coder community which will be of great help during the development process.

![Android logo](image1)

Secondly, thinking of the choice of database, the choices on my hands are a Google Firebase cloud database or an SQLITE local database. For this particular choice, I at first selected SQLITE for its ease of use and speed, but upon development I decided to switch to a firebase database for the multiple advantages it offers such as online synching of the database.

![SQLite vs Firebase](image2)

Moreover, for IDE choice, Android Studio would be the most suitable choice for this project. The last choice that I had to make was regarding creating design elements and the source for the voice effects in the applications, for this specific task I chose Photoshop and Freesound.org as they are tools I am familiar with, and are guaranteed to produce good quality content.

Economic

Economically, most of the development platforms I will be using are free, the paying ones such as photoshop are already available on my laptop which means that an investment is not necessary. In order to have the application available to all users over the world for free on Google Play platform, it is necessary to pay a fixed fee of twenty-five dollars.
Legal
Legally, it is important to make sure all my voice samples are mine or provided by a copyright free tool which is Freesound.org. No copyrighted material would be used in the design process of the application, as I will be personally be using photoshop to design every bit of the look and feel of the application.

Schedule
On a scheduling level, it is important to stay on a schedule that limits and sets deadlines. That is extremely important in order to stay safe from bad time management. I prefer for this project to work with a timetable that will allow me to time myself and not stay behind. My tentative time management plan is the following:

<table>
<thead>
<tr>
<th>Week</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>• Selection of project Idea</td>
</tr>
<tr>
<td>2</td>
<td>• Initial Specification</td>
</tr>
<tr>
<td>3</td>
<td>• A Feasibility Study</td>
</tr>
<tr>
<td>4-7</td>
<td>• Requirement Specification</td>
</tr>
<tr>
<td></td>
<td>• Interim report</td>
</tr>
<tr>
<td>8-12</td>
<td>• Design and implementation</td>
</tr>
<tr>
<td>12-13</td>
<td>• Implementation and testing</td>
</tr>
<tr>
<td>14</td>
<td>• Final report Submission and Project Defense.</td>
</tr>
<tr>
<td>15</td>
<td>• Updated Final Report submission</td>
</tr>
</tbody>
</table>

Figure 3: Project Timetable

3.2 REQUIREMENT SPECIFICATION
Since the project idea for WE CAN 2 is my personal software idea, I did not have to work with a real client that would usually provide needs to turn into requirements. Thus, I chose to work together with a psychology graduate friend as well as parents that deal with an ADHD suffering child in order to create a list of requirements that will represent the application’s goals. The results were the following:
3.2.1 Functional Requirements

- **Sign up, Create account, register**
  User chooses to create an account, if user already has an account, user is redirected to login activity

- **Login to the application**
  User logs into the application using account information, if user does not have an account, user is redirected to signup activity to create account

- **Select game type**
  For operations user selects the type of game amongst: Challenge, Casual, and Practice

- **Select level of ease of exercises**
  For operations user selects level of ease of the operations from: Easy, Intermediate, Challenging, Advanced, and Extreme

- **Select type of exercise**
  User selects the type of exercises they would like to solve from: addition, subtraction, multiplication, division, numbers, counting, and conditions

- **Select answer**
  User selects an answer amongst 4 choices

- **Check statistics**
  User select display personal statistics in the application

- **Receive points**
  User receives points after successful exercises

- **Logout**
  User logs out of the application
My application also has to also satisfy a set of non-functional requirements.

3.2.2 Non-Functional Requirements

- Performance
  Smoothly running fast software
  Good responsiveness level
- Usability
  WE CAN 2’s look and feel should reflect the young users main target by displaying a clean minimalistic interface that is also fun and easy to navigate for children. Numbers should be big, and script should be easy to read.
  What is more, the application should imperatively have voice effects, and exercises should be spoken to user in order to attract the attention and grasp it.
- Availability
  The application should be on and available for the user to use at all times, crashing is out of question for this piece of software as children would not necessarily have the patience to reboot the application. Aside from the lack of patience, users are also expected to have a
short span of attention, crashing could also mean the complete loss of focus.

- **Maintenance & Extensibility**

  WECAN2 should be easy to upgrade, new features should not disturb the usual layout of the application. The application should keep a level of stability in order for the children to develop a relationship of routine with the application. Which means that in order for the application to be maintainable and upgradable it is primordial for the initial layout to allow for seamless growth.

- **Reliability**

  The software should be reliable by making sure that the content displayed in the software is 100% reliable and true content. Risking children to learn a wrong information is a big problem that should be avoided at all cost.

- **Security and safety**

  The application should not harm or affect the user’s device in any way shape or form.
3.2.1 ER Diagram

The ER diagram for this project is a simple 2 class diagram. Relating by a 1.1 relationship. For every user, a statistics page is set up.

The users have an Email as a primary key, and an ID as secondary key, and a password. The statistics on the other hand have an ID as primary key, and the email as secondary key. Additionally, as attributes, the high score, the total score, the correct answers, and the played games.

Figure 5: WE CAN 2 ER Diagram
3.3 PROJECT DESIGN AND DEVELOPMENT

3.3.1 Technology enablers

WE CAN 2 relies on a few technology enablers for the design, and development process. The coming section will be diving into detailed elements of the development of each part of the software.

- Front end

  As stated in the feasibility study, the main goal was to create a software running on wide variety of Android powered devices. Android studio is a tool specifically designed for the development of android software. Additionally, among the options that I have been evaluating, I was able to find the most documentation on android studio development which made me feel more at ease developing using it as IDE choice as I personally did not have a lot of experience with android development previously.

  When it came to the choice of development language, Java was my personal choice. Java is a language which I have used for a large number of projects before, therefore, I felt most comfortable using it for the development process of my android application project. On top of that, java support is also widely available in the development community and a lot of open source material as well as tutorials are accessible for free for learning developers, which makes it perfect for my case.
The last step of the interface was to design the personalized elements to the application. For this task I chose to use photoshop CS6, as it is a tool I am personally familiar with and have a good deal of experience in using.

Figure 8: Photoshop Logo

Some of the designed features include the hints displayed to the users:

Figure 9: Subtraction hint design

Figure 10: Addition hint design
Back end

When it comes to back end development, I firstly worked with SQLITE database for its ease of use and speed, however, after discussing the disadvantages of SQLITE databases over cloud base databases with my supervisor, I decided to work with a Firebase Cloud based database. I at first experimented with synching my SQLITE database to firebase cloud. This would have allowed me to take advantages from both of these enablers. However, after many technical difficulties in the synching process, I decided to work solely on a firebase database system.

Firebase is a cloud-hosted database, where data is stored as JSON and synchronized in real time to every connected client [2]. The main advantage we gain by using Firebase is that your statistics and your account will be available over different devices containing the application, opposite to SQLITE which is a local based database that would only allow you to have your statistics and account on your usual device.

![Firebase synchronization diagram](image)

Figure 11: Firebase synchronization
Another good asset Firebase brings to the application is the ability to synchronize users data to further platforms other than android. For instance, if I chose to extend the application to IOS platform. The database I am currently using for the android version, would easily be encapsulated into the IOS version of the application without further development modifications needed to be done on the backend level [2].

System Architecture: Three Tier Architecture
3.3.2 The key features

- Random Number generation

WE CAN 2 offers a variety of math exercises for users. One of the important features for mathematic exercises is the random generation of numbers during runtime that will enable every exercise to be unique, and essentially no redundancy in exercises. This featured was possible thanks to the java.lang.Math and java.util.Random classes. In this fashion, for addition for instance we are able to generate 2 numbers randomly, add them up and put the result in one of the slots for answers, and generate 3 random numbers to put in the answer slots as well. This method not only enabled me to avoid redundancy in exercises, but it also gave me the ability to have different levels of difficulty in the exercises. As it allows the possibility of picking a range that the numbers will be in.

```java
Random randint = new Random();
int index = mathTypes.getRandomInt(MathAmount);
ansLoc = randint.nextInt(4);
if (index == 0) {
    //////////////////////////////////////////
    /////////////////////////////////////////////////// Additions

    int tempRand = 0;

    switch (difficulty) {
    case 1:
        tempRand = 30;
        break;
    case 2:
        tempRand = 60;
        break;
```
```java
    case 3:
        tempRand = 100;
        break;
    case 4:
        tempRand = 200;
        break;
    case 5:
        tempRand = 300;
        break;
    }

definitionText = "ADDITION";
Random randAdd = new Random();
int first = randAdd.nextInt(tempRand) + 1;
int second = randAdd.nextInt(tempRand) + 1;
int tempSolution = first + second;
solution = Integer.toString(tempSolution);
expressionText = first + " + " + second;
speech(first + "+" + second);
tempScore = additionScore;

getSimpleFakes(randAdd, tempSolution, tempRand);
```

- Text to speech

Sound is also a major functionality of the application, in order to grab attention of the user, it is important for the user to hear the numbers being spoken. At first I was looking at the possibility of hiring a voice actor to record every number for me, however this solution quickly seemed impossible as I started implementing for two reasons: firstly, the would mean that the application would only have a set of numbers; as many as I could afford to record professionally, secondly, that also meant that the solution with random number generation during runtime for exercises would
not be possible in this case as I needed to know the numbers in order to select the adequate recording. The solution then was to use a text to speech feature library as follows:

This function is responsible for initialization of text to speech instance

```java
public void initTextToSpeech() {
    progressDialog = new
    ProgressDialog(ConditionsActivity.this);
    progressDialog.setCancelable(false);
    progressDialog.show();

    textToSpeech = new
    TextToSpeech(ConditionsActivity.this, new
    TextToSpeech.OnInitListener() {
        @Override
        public void onInit(int status) {
            if (status == TextToSpeech.SUCCESS) {
                int ttsLang =
                textToSpeech.setLanguage(Locale.US);

                if (ttsLang == TextToSpeech.LANG_MISSING_DATA
                    || ttsLang ==
                    TextToSpeech.LANG_NOT_SUPPORTED) {
                    Log.e("TTS", "The Language is not supported!");
                } else {
                    Log.i("TTS", "Language Supported.");
                }

                progressDialog.dismiss();
                getQuestion(); // starts the game
                Log.i("TTS", "Initialization success.");
            } else {
```
Toast.makeText(ConditionsActivity.this, "TTS Initialization failed!", Toast.LENGTH_SHORT).show();

After that we pass any string to this function and voice will generate

    public void speech(String text) {

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP)
        {

            Set<String> a = new HashSet<>();
            a.add("female"); //here we can select gender of speaker
            Voice v = new Voice("en-us-x-sfg#female_2-local", new Locale("en", "US"), 400, 200, false, a);
            //here we can select language and accent
            textToSpeech.setVoice(v);

        }

        int speechStatus = textToSpeech.speak(text, TextToSpeech.QUEUE_FLUSH, null);

        if (speechStatus == TextToSpeech.ERROR) {
            Log.e("TTS", "Error in converting Text to Speech!");
        }

    }

    • Sound enabling

        Sound effects are also important to the functioning of the application. In order to
include mp3 sound files into the application I followed these steps:

Fetch and download adequate copyright free audio from Freesound.org
Put MP3 file in res -> raw folder
Use MediPlayer object to play mp3 file as follows:

```java
MediaPlayer mp = MediaPlayer.create(this, R.raw.clap);
mp.start();
```

Timmer functionality

In order to add pressure to solving operation exercises and accommodate a wider variety of users. I opted to use a timer for different levels of challenges. This feature enables the application to have a game like stress factor that challenges the user and pushes him to play more. This feature was possible thanks to the java.util.concurrent.TimeUnit class
3.3.3 Results

SIGNUP → LOGIN → Main page
The statistics activity

Operations exercises
(1 or more operation types)

Numbers exercises
(Only one)

Exercise Examples
With timer

Without timer
4. CONCLUSION AND FUTURE WORK

WE CAN 2 aims to aid children with learning disabilities in learning mathematical skills that they essentially cannot grasp in a normal school environment. The selection of mathematics was due to the fact that it is one of the subjects that these children struggle with the most. Throughout this semester, I was hoping to get into the head of a child with a learning disability, in order to design the best learning aid for them. Therefore, I have worked on an excessive research that tried to understand these children further and tried to analyze their learning capabilities in order to be able to reproduce them in a software environment. Furthermore, I have worked on a feasibility study for the application that aided me in making decisions regarding proper platforms, languages, and time management of the project. The next natural step was working on requirement specification before moving on to the design and implementation phase for both front end and backend features. This phase brought new challenges and a lot of work and dedication. It also forced me to get to know a lot more tools and software used for development, which is something that can only be learned with hands on experience and rarely in a classroom.

Although having faced some difficulties, this project taught me that it is possible to get started from an idea and passion for a topic, and to develop something beautiful that can change someone’s day or life to the better.

When it comes to future work, I believe in the near future the application could expand in two possible ways:

1. Including more mathematical types of exercises such as geometry exercises and problems
2. Including audio visual tutorials
5. REFERENCES


2. APPENDIX

Initializing text to speech

```java
public void initTextToSpeech() {
    progressDialog = new ProgressDialog(getActivity()
            .setCanceledOnTouchOutside(false);
    progressDialog.show();
    textToSpeech = new TextToSpeech(getActivity()
            .setOnVoiceChangeListener((status) -> {
                if (status == TextToSpeech.SUCCESS) {
                    int ttsLang = textToSpeech.setLanguage(Locale.US);
                    if (ttsLang == TextToSpeech.LANG_MISSING_DATA)
                        Log.e(tag, "TTS", "The Language is not supported!");
                    else {
                        Log.i(tag, "TTS", "Language Supported.");
                        progressDialog.dismiss();
                        getQuestion();
                    }
                    Log.i(tag, "TTS", "Initialization success.");
                } else {
                    Toast.makeText(getActivity(), "TTS Initialization failed!", Toast.LENGTH_SHORT).show();
                }
            });

public void speak(String text) {
```

Numbers activity: This activity allows user to move through numbers while being able to view them in both number format, letter format, as well as hear them through the text to speech feature.
Random number generation for conditions activity

```java
public void getQuestion() {
    int randomNo = new Random().nextInt(100);
    int randomNo2 = new Random().nextInt(88);
    op1 = randomNo;
    op2 = randomNo2;

    expression.setText(randomNo + " ? " + randomNo2);

    if (randomNo > randomNo2) {
        ansLoc = randomNo;
    } else {
        ansLoc = randomNo2;
    }

    option1.setText("<");
    option2.setText(">");
    option3.setText("=");
}
```
Setting difficulty for operations

```java
mathAmount = mathTypes.size();
difficulty = pref.getInt("difficulty", 1);
type = pref.getInt("type", 5);

if (type == 0) {
    timeLimit = 5;
    multiplier = 2;
} else if (type == 1) {
    timeLimit = 10;
    multiplier = 1;
} else {
    timeBar.setVisibility(View.INVISIBLE);
    multiplier = 0;
}
System.out.println("TYPE:" + timeLimit);

additionScore = 2 * difficulty;
subtractionScore = 4 * difficulty;
multiplicationScore = 6 * difficulty;
divisionScore = 8 * difficulty;
```
Shuffle array to make sure no numbers are repeated, with the use of `java.util.random`

```java
package com.wecan2;

import java.util.Random;

public class ShuffleArray {
    public static void shuffleArray(int[] a) {
        int n = a.length;
        Random random = new Random();
        random.nextInt();
        for (int i = 0; i < n; i++) {
            int change = i + random.nextInt(bound: n - i);
            swap(a, i, change);
        }
    }

    private static void swap(int[] a, int i, int change) {
        int helper = a[i];
        a[i] = a[change];
        a[change] = helper;
    }

    public static void main(String[] args) {
        int[] a = new int[] { 1, 2, 3, 4, 5, 6, 7 };  
        shuffleArray(a);
        for (int i : a) {
            System.out.println(i);
        }
    }
}
```