

COM814: Project 2013-14

Dissertation

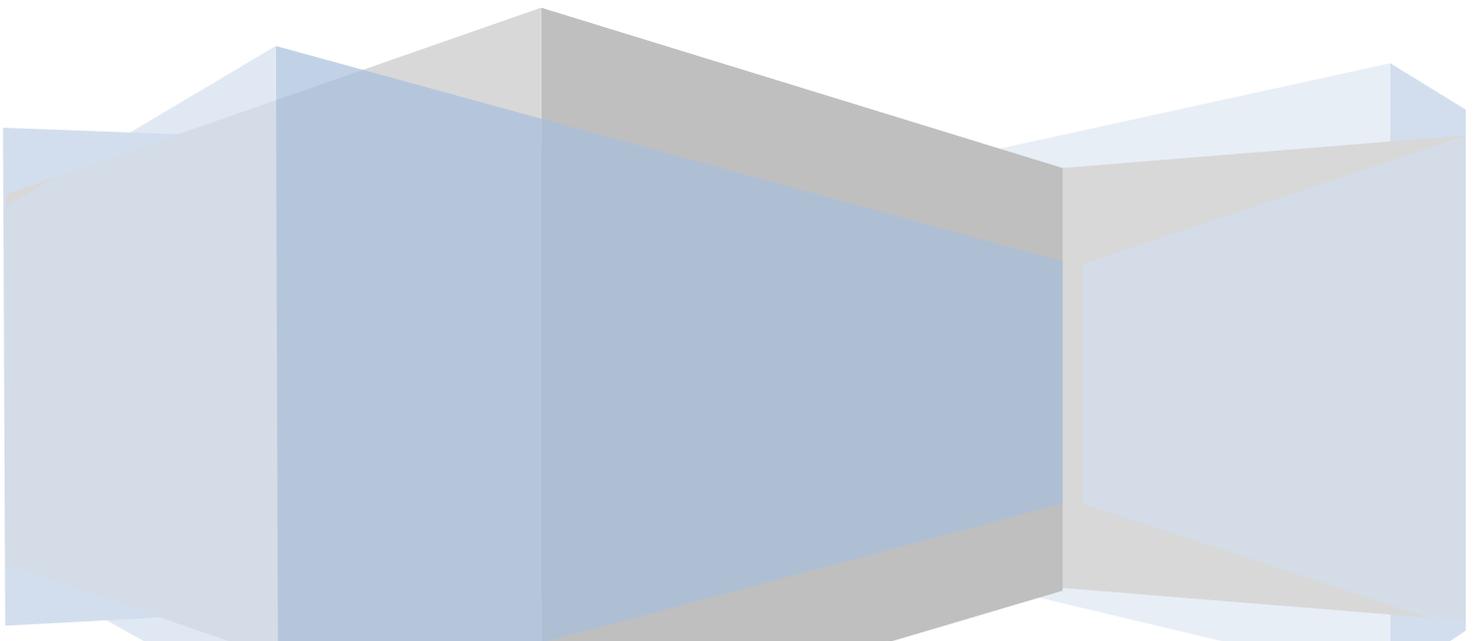
School of Computing & Information Engineering

Binary Number App

Supervisor: Janet Alison

Second Marker: Adrian Moore

29/08/16



Plagiarism Statement

“When submitting your work you are agreeing with the following statement: I declare that this is all my own work and that any material I have referred to has been accurately referenced. I have read the University’s policy on plagiarism and understand the definition of plagiarism. If it is shown that material has been plagiarised, or I have otherwise attempted to obtain an unfair advantage for myself or others, I understand that I may face sanctions in accordance with the policies and procedures of the University. A mark of zero may be awarded and the reason for that mark will be recorded on my file.”

Signed – Mark Conaghan

Acknowledgements

In particular I would like to thank Janet Alison for her much welcomed support and guidance throughout the project and for keeping me on a strict deadline to ensure the project finished ahead of schedule. I would like to thank Chris Stretch for his help in getting me started with the project. I would also like to thank my family and my girlfriend Gemma who regularly became an impromptu test subject for the Binary101 application.

Contents

Plagiarism Statement	
Acknowledgements	
Abstract	1
1. Summary	2
Introduction	2
Problem Statement	3
Aim	3
Project Objectives	3
Design Platform	4
2: The Problem	4
The problem	4
Current Learning Model	6
Competition summary.....	8
3. The Solution	9
Effective learning.....	9
The Binary 101 Solution	9
4. Professional Issues	10
Business Case	10
Project Risks	10
Physical Risk Table.....	11
Management Risk Table	11
Ethical Considerations	12
5. System requirements	12
Functional requirements	12
Non Functional Requirements.....	13
Effort and Overall Achievability.....	13
6. User Interface Design	14

7. Architecture.....	15
8. Implementation.....	16
Core Files.....	16
AndroidManifest.xml.....	16
MainActivity.java.....	17
MenuAcivitiy.java.....	18
Theory.....	18
Tutorials.....	18
Videos.....	18
Test Section.....	19
BinaryQuiz.java.....	20
PerformanceActivity.java.....	22
NIM.class.....	23
ScoreActivity.java.....	25
BinaryCalculator.java.....	25
NumberConvertorActivity.java.....	26
Common code Elements.....	26
Artificial intelligence.....	27
9. Testing and evaluation.....	27
Introduction.....	27
Developer Testing.....	28
Focus Group Testing.....	30
Summary.....	33
10. Conclusions.....	33
Evaluations of objectives.....	33
Project Overview.....	35
Recommendations.....	36
Summary.....	37

9. References.....	38
Appendices.....	40
Appendix 1: Rich Picture Solution	40
Appendix 2: Questionnaire.....	41
Appendix 3: Questionnaire results	44
Appendix 4: Initial mock ups	45
Appendix 5: App Navigation Tree.....	46
Appendix 6: Navigation and GUI diagram	47
Appendix 7: Button placement diagram	48
Appendix 8: Focus Group Testing Questionnaire Results	49
Appendix 9: Java code of Key Elements	51
9.1 ComputerMove method.....	51
9.2 Database helper code.....	52
9.3 bubble sorting algorithm.....	54
9.4 Insert Score Method.....	54
9.5 Shared Preferences Storing and Retrieving Method	54

Abstract

This project sets about to determine the need for an interactive learning tool. The learning tool is to assist students studying computer science or similar courses. The aim of the tool is to provide a convenient and novel way of learning the fundamentals of binary. Computers are so common place almost everyone in developed societies has some sort of computer device the most common being the smart phone. A mobile app was deemed the most appropriate method of delivering the interactive learning tool. It would be easy to use and be able to be accessed anywhere through there smart phone and would not have the physical restrictions of a desktop app. It is proposed to take this project through the design; implementation stages and have a working prototype witch will be eventually developed into a final product. The project will investigate the need for such an app, its benefits and the stakeholders. The final product will have addressed the problem statement and all the needs of the stakeholders, as well as providing an engineered solution to any design concepts or needs in the market as addressed in this report.

Keywords: interactive, learning, tool, app, mobile.

1. Summary

Introduction

This project will investigate effective methods of learning in today's modern society and aim to create an app that provides a one solution fits all approach to learning binary. A review of current apps purporting to provide this will be carried out. The short comings of these apps will be highlighted along with useful aspects. From this information it will be determined what the current apps in this area are missing and aim to provide the solution.

The way people are learning today is changing with the consumer trends as people have more access to electronic material through iPads and smart phones it is often cheaper and easier to research and learn this way rather than reaching for the traditional text book it has been observed that more than ever people want to learn when they want and how they want. (Rosen 2010) describes it as, *"anytime, anywhere, anyway... my way"*. As students want learning to be flexible then a different way of delivering content, providing tuition and testing must be developed.

The usefulness of a well written textbook or the delivery of knowledge via a well-structured lecture as a learning tool cannot be disregarded. However, e-learning is becoming ever more popular. One of the main drawbacks in textbook learning is that they are not very interactive. It has been shown that interactive learning yields a far greater retention of information than that of non-interactive forms.

It has been shown that although a certain level of knowledge may be gained through whatever method of learning, if that knowledge is never accessed then the knowledge is likely to be forgotten. The brain will literally get rid of it this is known as pruning as described by (Draganski et al ,2004)

"people who learned how to juggle increased the amount of grey matter in their occipital lobes, the area of the brain is associated with visual memory. When these individuals stopped practicing their new skill, this grey matter vanished."

From these observations it is suggested that there are particular techniques and practices which result in the absorption and retention of information. It should be obvious then that the

inclusion of a combination of these would provide a most effective learning tool. A learning tool for some can be utilized as a teaching tool for others so this also increases to the potential markets for such an application.

Problem Statement

Current apps today provide a results only based solution to binary manipulations. There is a need to provide an interactive learning platform to allow students or others to check their binary number calculations and manipulation whilst also providing them with the background theory and tuition to understand binary and its uses. Users' ability should be able to be tested and tracked to ensure adequate progress is being made.

Aim

It is the aim to develop an effective multi approach teaching and learning tool via the means of a mobile app. The app will include a multitude of learning and practice platforms to ensure better retention of knowledge and skills in the area of binary number manipulation. It is also proposed to have a testing and performance tracking system to allow for users to monitor their progression.

Project Objectives

- To carry out in depth research into similar apps available whilst analysing the drawbacks of these apps and propose additional features to combat these short comings and to apply these in the development of the app to which this report lends itself to.
- To utilize a multi-layered learning approach backed by psychological research to ensure maximum learning and retention of knowledge and skills. This approach will also allow the app to be utilized and be more effective to a broader demographic as a learning tool.
- A focus group survey will be carried out to determine the most popular aspects of the app from aesthetics and functionality to uniqueness of features.
- The app should have multiple features including the use of a forum based question and answer section, an interactive exercise feature to test and practice skills and a theory based research section to broaden the users' knowledge. The app is to also make use of SQL database features.
- To develop an aesthetically pleasing app that engages the user to minimize the likelihood of them losing interest and becoming bored.

A prototype is to be developed at a fairly early stage to test the functionality and user-friendliness of the app. The prototype will also be used to highlight any issues with the app and can be tested on the focus groups who can provide feedback on what they feel the app lacks. This can then be implemented before the final testing stage or proposed for additional features in future versions. This would ensure maximum profitability of the product if the free app route is disregarded.

Design Platform

The development of the app is to be carried out using the IDE Android Studio which is based on the IntelliJ IDEA. This is the proposed platform due to the familiarity with the software it also provides the necessary framework for the features that are to be included in the app.

2: The Problem

The problem

Galileo once said *“Mathematics is the language in which God wrote the universe”* (www.quotationspage.com, 2016). Everything in the Universe can be understood through some form of mathematics. Biology is complicated chemistry. Chemistry is complicated physics and at the root of it all physics is complicated mathematics. Every human being on the planet is affected by mathematics on a daily basis, many without even being aware of it. Today is the age of computers and the internet *“The internet was accessed every day, or almost every day, by 78% of adults (39.3 million) in Great Britain in 2015, compared with 35% (16.2 million) in 2006”* (Internet Access - Households and Individuals, 2015). Many people like to have some knowledge of how something works if they use it every day even if it is just out of curiosity; others have an obligation to understand it. Computers at a most basic level work using binary numbers. Binary numbers as a stand-alone cannot convey very much, at very best 1, 0, yes, no, on or off. It is the manipulation and mathematical operations with binary numbers which makes them useful and functional.

In the future the growth of Information Communication Technology (ICT) based jobs is set to increase more than any other sector and that there is likely to be a great skills gap as described in a report by Hollingworth and Harvey-Price (2013) *“The most common skills thought to be lacking are job specific, and advanced IT or software skills.”* It is therefore in the governments’

interests to tailor their education system to combat this skill shortage and provide students with more employment opportunities. In a press release from the Department of Education the Education Secretary Michael Gove introduced £20000 teaching bursaries for graduates to train to be teachers in areas such as ICT and Computer Science (department of education, 2012). With the enticing new bursaries and push for computer scientists it would be beneficial to have new learning and teaching platforms developed.

Binary is at the very core of Computer Science. It is a subject area that many students have difficulty in gaining any real understanding of it or how it's used. "There are 10 types of people who understand binary, those who do and those who don't". This play on words is a reference to the binary representation of the number 2.

There are lots of binary converter apps on the market but these just tell you an answer so you don't really learn anything. There are also apps which do binary operations. Again they only provide an answer. So unless you already know how to do these calculations these apps are no use as learning platforms. The full review of these apps is provided in this report under the Competition Summary section.

The Textbook can be a cumbersome learning tool. They can be useful in providing theory or instruction via diagrams, but it can't always be at hand if you need to check a calculations or do some practice. As it stands there is no 'one size fits all' solution to learning binary here in lies a need which is to be satisfied.

Current Learning Model

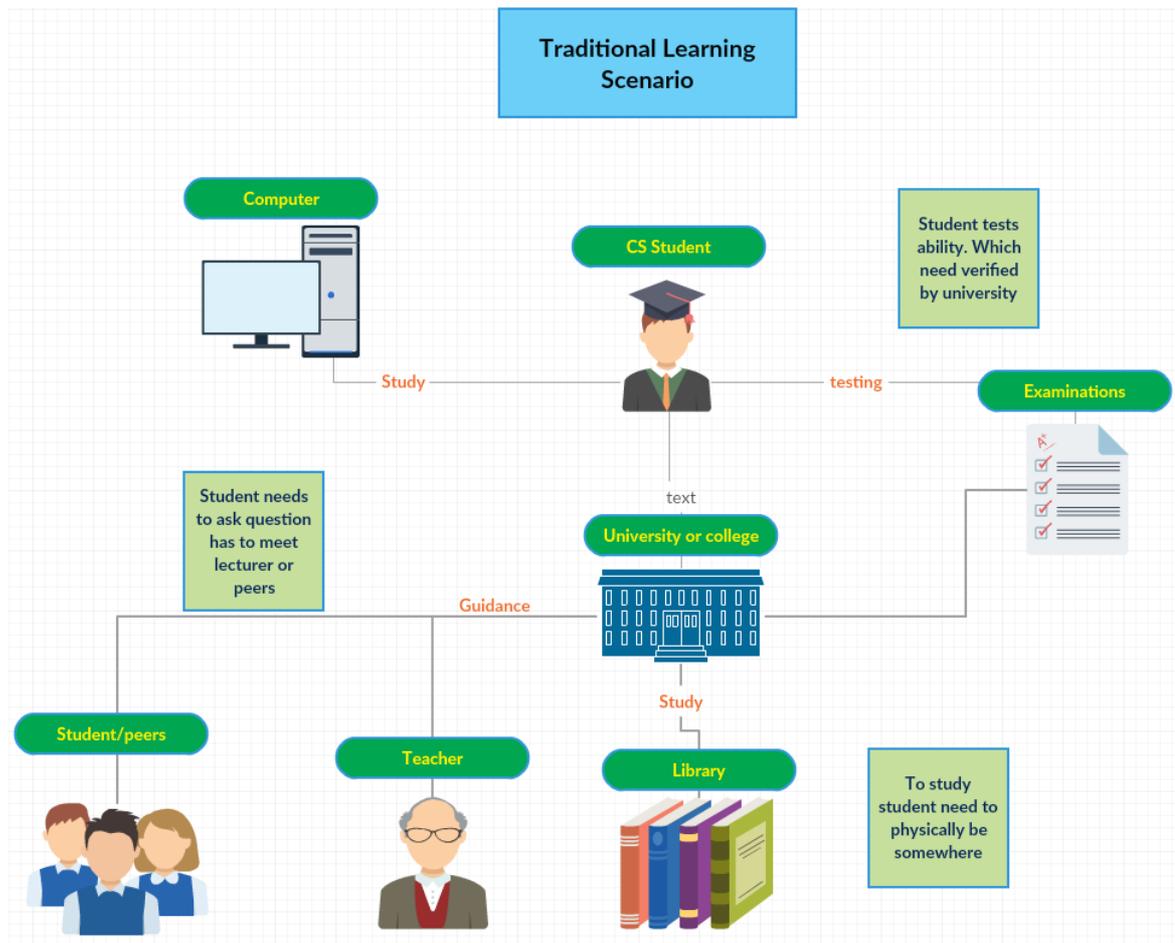


Figure 1 Rich Picture diagram

Figure 1 shows the current model of learning for computer science students. All of the resources that the student would like to utilise are either are difficult to access physically through distance or time consuming to access. These aspects make the learning model inefficient. If resources could be collected and pooled together via a useful tool travel time would be reduced leading to increased study time.

**Binary Practice App**

This app allows you to practice binary conversions at 3 levels of difficulty and time how fast you complete a list of questions to keep score and track progress

Limitations

- Only tested on binary to decimal conversions no option for hex to octal etc.
- No tuition provided to how to do conversions

**Binary Challenge App**

This app allows you to test your conversions to decimal in binary, quaternary and hex. Adjustable difficulty feature, it provides explanations of conversion steps.

Limitations

- Doesn't provide theory or uses for binary.
- Limited as a learning tool. Better for practice if you already know how to manipulate binary

**Advanced Binary Converter App**

This allows for conversions of many bit numbers to quite a few numbering formats

Limitations

- Only performs conversions with no testing or learning features
- Not very interactive or engaging with the user

**Bit Fiddler app**

This useful app allows the user to calculate the results of binary operations in a variety of binary formats. Useful for Software engineers and computer science students alike

Limitations

As a learning tool it provides no instruction how to perform calculations. User ability can't be tested. Acts only a digital tool for checking calculations. It doesn't develop user's ability to perform hand calculations or conversions. Difficult to use.

**Learn Binary app**

This tool claims to teach binary but provides little to no tuition. Can convert numbers to binary and vice versa showing the value of each bit.

Limitations

No direction given on how to use app. Very un-engaging visually. Slight suggestion of how binary works but more of a guessing game for the user.

Competition summary

The five apps above were reviewed and their pros and cons determined. Some had quite useful aspects as standalone feature like the Binary Practice apps ability to keep score and time it took to complete exercises. This means that performance is measurable a useful tool to determine if the user is progressing with their binary skills. Binary Challenge provided some step by step instruction on the conversions, useful if the user is a complete novice as it gives some explanation on the process. The steps however were not very in depth. Advanced binary convertor allowed conversion in a wide variety of numbering formats but lacked the ability to perform operations and provided no tuition. Bit Fiddler was a great tool for performing various binary operations and even chaining several operations together. Bit Fiddler was great for checking your answers from say a textbook but provided no way of challenging the user within in the app. The Learn binary app was very limited in its features and its usefulness. The name tends to mislead the user giving them some inclination that they will be provided with some

guidance how binary works, the process of conversion or carrying out operations. None of which is provided. It is a binary to decimal converted which does slightly demonstrate the significance of the position of a 1 within the binary number by indicating its value. Most of the similar apps had some degree of useful functionality but lacked a holistic approach to learning. To be an effective learning tool all the features mentioned plus some additional features would need to be brought together in cohesive manner.

3. The Solution

Effective learning

When learning any new subject, the process is often a difficult and sometimes tedious, it is therefore logical to take the most effective approach that is suited to the subject. There have been many books wrote and numerous studies done on effective learning techniques. There are a few techniques to help the brain store information more effectively. A few of these techniques include

- Learn in a variety to ways - According to Judy Willis (2006) an authority on brain research regarding learning. Different ways of learning mean the information will be stored in different parts of the brain and thus more likely to be retained.
- Keep Score – Testing yourself regularly and tracking progress is shown to boost motivation resulting in a better overall learning outcome. Make it fun – game based learning tool have been shown to increase the users’ skills in specific tasks
- Make it relatable. Judy Willis (2010) explains you cannot teach maths in isolation. Seeing how something is applicable in a real life scenario makes it more interesting and easier to understand

The Binary 101 Solution

Binary 101 is the solution to students and computer programmers struggling to find a definitive tool for honing their binary skills. It employs the use psychology based techniques backed by studies to ensure the most comprehensive learning experience. It provides video tutorials, step by step instruction to learning binary manipulation and gives users an in depth understanding of the processes involved. Once users have gained sufficient knowledge they can test their skill level and solidify what they have learned through built in practice tests. The results of these tests can be tracked along with the time in which the test took to complete. This can be used to

mark their proficiency. The in app game also allows users to use the skills they acquired to beat the computer plus providing a fun element to keep students engaged. Students doing some of their own binary calculations can use Binary 101 to check answers with its convenient binary converter and binary calculator. All the features included serve to reduce the time it takes to master binary whilst also reducing the chances of students getting bored or disheartened through making the learning fun, relevant and progress evident. The idealised rich picture solution can be seen in Appendix 1

4. Professional Issues

Business Case

Binary 101 is a mobile based app. The app's aim is to make the life easier of budding computer scientists. It does this by providing all the necessary resources to become proficient in the use of binary all in the one place. It proposes to include a number of binary based logic games such as NIM. This is a simple game where players take turns to remove objects from stacks and the player to remove the last object wins or loses depending on which convention is used. Logic games such as Sudoku have grown hugely in popularity since 2005 with one free Sudoku app being installed in between 5-10 million devices (Google Play, 2016).

- Reduces the need for text books
- Can be used to check binary operations needed for school or work
- Fun game, possible stress relief
- Possible scope to allow advertising to generate income
- Or in app purchases to allow conversions in more format types.

Project Risks

A thorough and systematic Risk Assessment was carried out to highlight and manage risks. This project has been deemed to be very low level of risk. The app is mainly research and office based. The likely risks are covered in Table 1 and Table 2.

Physical Risk Table

Risk	Probability	loss	threat	Mitigation
Slips, trips & falls	3	3	9	Take care in labs and maintain tidy work space
RSI	2	3	6	Use correct typing techniques
Stress	5	2	10	Manage work time, regular breaks. Follow work plan to remain on schedule
Postural problems	2	3	6	Follow postural diagram provided in figure 2.

Table 1 Physical Risk Assessment

Management Risk Table

Risk	Probability	loss	threat	Mitigation
Missing submissions	2	4	8	Keep to a planned work schedule
Sickness	2	4	8	Allow sufficient time in work schedule to allow for sickness
Data loss	2	5	10	Use several methods of storage eg Dropbox flash disks etc.
Hardware /software issues	1	3	3	Ensure all relevant resources are available to complete work.

Table 2 Management Risk Assessment



Figure 2 Correct Posture diagram

Ethical Considerations

The ethical considerations of this project have been reviewed and an application has been submitted for category Z as no invasive techniques were employed and the completion of the project does not involve the use of minors or vulnerable adults.

5. System requirements

To get an understanding of what functionalities a possible user would like to see, a questionnaire was put to a focus group. The focus group consisted of 20 Postgraduate students in a Software Development based course and some adults with no background in Computer Science etc. The Questionnaire and the results can be viewed in appendices 2 and 3

Functional requirements

From the results of the survey carried out system requirements were determined and expressed in terms of user stories. These User stories are a representation of the functional requirements of the app and are essential to the success of the app.

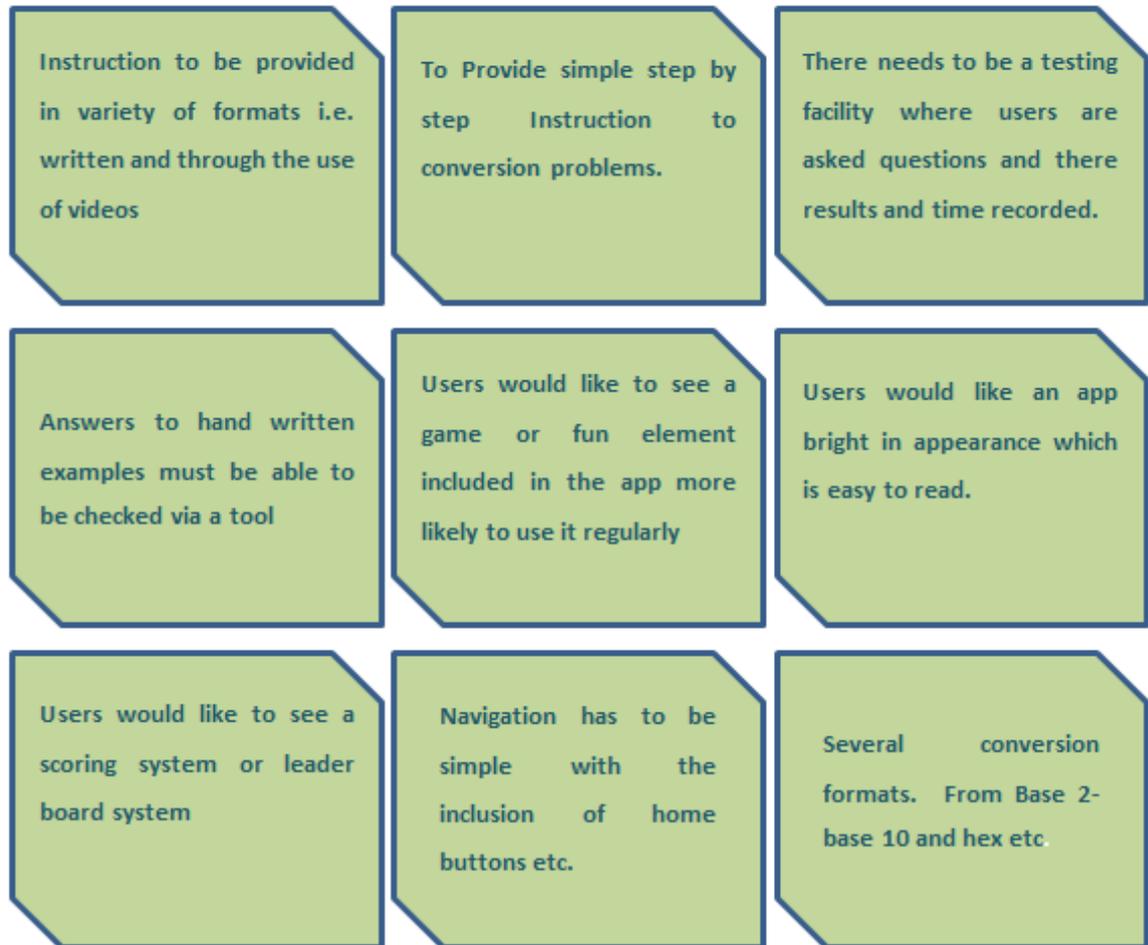


Figure 3 User Stories

Non Functional Requirements

Some careful consideration into the non-functional requirements has been given. These are requirements that are not essential to the working of the app but do contribute to the success of the app in some way. Some of these requirements include: -

- Professional and pleasing aesthetic
- User-friendly navigation system
- A user guide or instruction on how to use features
- Well written and tested code to ensure no crashing.

Effort and Overall Achievability

The total effort expressed in hours of work for this project is 600 hours. Given the current level of app building and technical experience this will be a huge undertaking however knowledge

and technical expertise will continue to grow throughout this project and the remainder of the course. It is believed that overall level of achievability to be quite high due to careful planning and well devised work schedule which is accounts for possible unanticipated delays to the projects schedule.

6. User Interface Design

The UI (user interface) design is a highly critical aspect of mobile app design even more so in comparison to traditional desktop applications. Mobile app users tend to be less tolerant of badly designed user interfaces (UIs) and demand a more streamlined experience.

*“While 79 percent of consumers would retry a mobile app only once or twice if it failed to work the first time, only 16 percent would give it more than two attempts. Poor mobile app experience is likely to discourage users from using an app again.”
(Compuware.com)*

Binary 101’s user interface design adhered to a strict design ideology throughout. The user is to be presented with a simple uncluttered interface that is simple with the navigation to be self-explanatory. The initial mock ups can be seen in appendix 4. From the focus group questionnaire, it was decided that the blue and orange contrasting theme would be adopted throughout the app as this received an 85% for being most appealing. The simple self-explanatory design was achieved using sectionalised interface with four main areas branching off from the Main Menu activity. Each of the four main areas is broken into subsections, it is these subsections are used to provide services and deliver the content. The user navigation pathway is illustrated diagrammatically in the Appendix 5

This navigation layout allows any users to reach their desired page within 2 levels of the navigation tree meaning that it is quick, simple and no pages are time consuming to reach.

“On a mobile device, you want to keep things accessible. If the user has to tap through 4 levels of menus to find something, chances are high they’ll leave after the 2nd tap. Avoid multi-level menus on mobile sites as much as possible” (Richa Jane, 2015)

Appendix 6 demonstrates one pathway through Binary 101 with the inclusion of the GUI (Graphical User Interface). The GUI has been carefully considered to maximize user experience which ultimately ensures the continued use of the app by its users. Since this will be run on touch screen enabled devices the app employs a widget based GUI compared to the WIMP (windows, icon, menus, and pointer) of desktop applications. On adopting this approach there are specific design considerations that must be addressed? All buttons are large and easily recognisable. They have also been placed in strategic places to ensure they are easily reachable. The top left hand corner is the most cumbersome to reach so it was decided to place the app logo in this position so that no buttons can be placed here whilst also maximizing screen usage. Appendix 7 illustrates ideal button placement.

The idea of screen usage has been considered throughout and all pages aim to have well spread out widgets and text to ensure no clustering to maximize the screen usage. A dull or drab display can give a negative impression to users whilst also making it difficult to identify components of the UI. Binary 101 addresses this issue by adopting a bright contrasting colour scheme of blue, orange and green to a lesser extent. This scheme can be seen in Appendix 6 and gives a bright and lively experience whilst also being functional. The menu page has also been designed in the shape of a 1 to tie in with the binary theme -not a functional design feature.

7. Architecture

“Software application architecture is the process of defining a structured solution that meets all of the technical and operational requirements, while optimizing common quality attributes such as performance, security, and manageability. It involves a series of decisions based on a wide

range of factors, and each of these decisions can have considerable impact on the quality, performance, maintainability, and overall success of the application.” (Microsoft, 2009)

It is important to the developer to have a good knowledge of the architecture that is to be used when designing software. In the case of Binary 101 it will be android Architecture. When dealing with android architecture it is important to choose a suitable API (application program interface). If an app with the most recent up-to-date features is to be made, then the highest level API must be selected in the design. This will however limit the number of devices that can run the app and narrow your potential markets. For this app it was decided that the app will be created using API 19 or the android 4.4 KitKat version. This API was chosen as it is the most widely used android version as of 2016 (android.com, 2016) and also provides all the necessary features in the built-in libraries for the app to be created. Use of the SQLite database feature may be utilized in the creating a question database for the testing feature while media framework was used for audio and animations used in the app and game feature.

8. Implementation

In this section the implementation is discussed and how development evolved from the design and analysis stage. The features which make binary 101 unique will be examined in detail and how the core functionality aspects were achieved. The java classes will be dissected and key programming elements will be studied.

Core Files

AndroidManifest.xml

The Manifest file is an extremely important resource when it comes to android application development. It outlines all the files that are needed at runtime. If these files are not included in the manifest then the application will crash when the application tries to access the relevant file. Managing resources within an application is an extremely important aspect of app development. The android SDK assists in this process and reduces development time for the developer by automatically generating some base resource files. This saves the developer

having to create them as they go. The developer may need to create some of their own files to store additional resources. Important directories generated by the SKD are drawables, layout and values folders. The drawables folder stores any images or graphics files to be used within the app. The layout folder is a highly important folder in terms of the compiling of the app. The layout folder houses all the xml files that the java classes with call in their onCreate(); methods. These files define the layout, widgets and general appearance of the app as a whole. The values directory folder contains a few folders one of which is the strings.xml files. Strings.xml is a highly effective way of storing strings or text values as storing in this way allows java classes to reference from this file thus improving the apps ability to be translates to other languages efficiently. These files discussed provide an important supportive role within the app dynamic however the bulk of the work is carried out by the java classes these are discussed in detail in the next section. The 18 .Java classes listed in the Binary101 manifest include.

BinaryCalculator	NumberTheory1
BinaryQuiz	NumberTheory2
BinarySignificance	PerformanceActivity
ConversionTutorial	PlayMenuActivity
DbHelper	Question
LearnMenuActivity	ScoreActivity
MainActivity	TestMenuAcivity
NIM	ToolsMenuAcivity
NumberConverterActivity	VideoTutorialArithmetic

The java classes will be addressed in the logical order in which they are presented within Binary101 in a sectionalised manner. This allows for greater understanding of the java classes and their role within the app.

[MainActivity.java](#)

This is the homepage of the app through which all other sections can be accessed. It implements onClickListeners. These are types of event handlers which listen for a button click event and subsequently carry out the associated code.

```
//button to navigate to the other menus
Button goToLearnButton = (Button) findViewById(R.id.goToLearnButton);
goToLearnButton.setOnClickListener((v) -> {
    buttonClick.start();
    Intent intent = new Intent(getApplicationContext(), learnMenuActivity.class);
    startActivity(intent);
});
```

Figure 4 OnClickListener

In this case the buttons start a new activity by creating a new intent and calling the `startActivity();` Method and passing the intent in as a parameter.

MenuActivity.java

These follow the similar principles of the MainActivity.java and are used as a means of navigating through Binary101 in a logical and self-explanatory manner. The menu activities also serve to reduce the navigation levels to reach any of the services within Binary101.

Theory

NumberTheory1.java, NumberTheory2.java, BinarySignificance.java

These java files are simple java files which are used to display information how decimal and binary numbering is derived. They use the `.setText();` method to make a TextView display the relevant information by passing in a string resource saved in the Strings.xml resource file. This method is a much more elegant solution compared to hard coding the text. This method ensure that the app can be easily translated to many languages and thus increasing its market potential.

Tutorials

NumberConversionTutorial.java

This file sets up the predefined layout from the `oncreate();` method to display an example and step by step instruction on how to convert to binary. It also employs the use of a button to quickly invert the conversion process through an IF statement which checks a Boolean status then decides which string resource to pass to the `setText();` method.

Videos

VideoTutorialArithmetic.java

This class creates a VideoView and passes the associated mp4 file which is saved in the resource section of the project structure. A MediaController is then created to link to it.

```
// Sets Video view
final VideoView videoView = (VideoView)findViewById(R.id.video_tut_vv);
videoView.setVideoPath("android.resource://" + getPackageName() + "/" + R.raw.binary_arithmetic);

//sets video menu controls
MediaController mediaController = new MediaController(this);
mediaController.setAnchorView(videoView);
```

Figure 5 Setting up VideoView

Test Section

Here the functionality is provided to allow the user to test their binary ability through a multiple choice binary quiz. User performance is then tracked in the performance pages which ranks their best performances.

Question.java

This class unlike most of the other classes in the app is not an activity. Question.java acts as an object class which when called creates a question object through the use of a constructor.

```
public Question(String qQUESTION, String oPTA, String oPTB, String oPTC,
                String aANSWER) {

    QUESTION = qQUESTION;
    OPTA = oPTA;
    OPTB = oPTB;
    OPTC = oPTC;
    ANSWER = aANSWER;
}
```

Figure 6 Question Constructor

The constructor allows the question and all the relevant information to be added to the SQLite database. When called the question answer and 3 options are stored to the database. Set and get methods within this class allow other classes to access and set the variables stored within the object.

DbHelper.java

This class extends the SQLiteOpenHelper java library built into the Android Studio IDE. This java file is a crucial element in testing section as it allows a database of question objects to be created, updated and accessed. This code was adapted from (developerfeed.com 2013) the full code can be referenced in appendix 9

```
public void onCreate(SQLiteDatabase db) {
    dbase=db;
    String sql = "CREATE TABLE IF NOT EXISTS " + TABLE_QUEST + " ( "
        + KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " + KEY_QUES
        + " TEXT, " + KEY_ANSWER+ " TEXT, "+KEY_OPTA +" TEXT, "
        +KEY_OPTB +" TEXT, "+KEY_OPTC+" TEXT)";
    db.execSQL(sql);
    addQuestions();
}
```

Figure 7 Database OnCreate Method

The onCreate method used in this class serves a special purpose. It makes use of SQL DDL (database Definition Language) this is used to define table Attributes conceptualised as the columns of a table. This is the logical structure not the physical. Physically the data within the database can be stored in multiple locations. There are multiple methods contained within this class that play a vital role on how the database is maintained. The AddQuestions(); method initialises question objects into the database. These are hardcoded into the app. OnUpgrade(); plays a crucial role in maintaining the database. This ensures that if a Table is updated or altered that the out of date table is deleted or 'dropped'. It does this by keeping track of the versions.

```
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldV, int newV) {
        // Drop older table if existed
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_QUEST);
        // Create tables again
        onCreate(db);
    }
```

Figure 8 Database OnUpdate Method

The database is passed in as a parameter along with the old and new version values. If a table exists with that name and the new version is off the higher value than the old, the table is dropped and a new one is created from the onCreate();. The addQuestion(); distinct from addQuestions(); provides the functionality for the user of the app to manually add additional questions to the database. Currently this functionality is not available in the release version of binary101 but may be made available for future or patched version. getAllQuestions(); returns a list of question objects see Appendix 9 for full details.

BinaryQuiz.java

This class is a runnable activity class which uses the two previous classes. It sets up the database by making an instance of the DbHelper class then saves the questions as a List of objects by calling .getAllQuestions(); on the database instance. On creation of the quiz activity a chronometer is implemented and begins to track the users' time to complete the quiz. The setQuestionView(); method is to set up the relevant views and radio buttons with the question data from the list of question objects. Users advance through the quiz by using the submit button once they have selected what they believe the answer to be. The onClickListener(); responds to the button click by extracting the text from the selected radio button and comparing it to the answer saved within the question object. If the comparison is a match the score variable is incremented along with the question ID variable (qid). If it is a non-match players are

displayed with a toast informing them so. The correct answer is displayed and a negative tone sounds. This improves the feedback the user experience.

The quiz is broken up into two parts the questions from the database which provide mainly theory based questions then the quiz proceeds to randomly generated questions to test binary arithmetic. The random questions are generated through the randomQuestion(); method. This method uses the randomInteger(); to generate a random operands between (1-100). It then uses a switch statement to randomly decide what operation to perform on the operands.

```
switch (randomNo) {
    case 1:
        operation = '+';
        randomAns = randomOperand1 +randomOperand2;
        break;
    case 2:
        operation = '-';
        if (randomOperand1 > randomOperand2)
        {
            randomAns = randomOperand1 - randomOperand2;
        }//if
        else
        {
            randomAns = randomOperand2 - randomOperand1;
        }
        break;
}
```

Figure 9 Random Operator Switch Statement

The quiz needs to generate alternate wrong answers. To keep answers with a reasonable range of the correct answer a random deviator is generated and the wrong answers are adjusted from the correct answer. The program utilizes random numbers and if statements to ensure that the answer options are placed randomly within the radio group. This ensures the users cannot just memorize the correct sequence.

Once the quiz is complete, an algorithm works out a performance score then passes the score value as an Extra to the new activity Performance.java. It does this through the built in method .putExtra(); to the intent just created. The process can be seen below

```
long_time = SystemClock.elapsedRealtime() - timer.getBase();
int seconds = (int) (long_time / 1000 % 60);
performanceScore = (int)((score * 1000) / (Math.sqrt(seconds*10)));

Intent performance = new Intent(BinaryQuiz.this, PerformanceActivity.class);
performance.putExtra(PERFORMANCE_TAG, performanceScore); //Your score
startActivity(performance);
// finish();
```

Figure 10 Performance Algorithm

The performanceScore variable is a function of the time and the score received within the quiz. The time uses a negative power to ensure that time is least critical when determining best performance otherwise a user who completed the quiz with one correct answer very fast may out perform a user with 100% but a large completion time. This would not be a very intelligent scoring algorithm.

PerformanceActivity.java

One of the key features of Binary101 is its performance tracking capabilities this is achieved through a scoreboard system. The score is received and stored in a bundle from the quiz activity if the performance page is open via the quiz page otherwise this step is bypassed. If a score is received a dialog box is utilised to prompt the user to enter a name to pair with the score received. The dialog displays an editText to allow user to input their name. If ok is selected then the name is extracted and saved to a variable Player_from_quiz. The performance_from_quiz is passed as a parameter to the method insetScore(); which uses a linear search technique to determine where in the array the score should be placed. The same index position is used to place the name in the string array used to save the player names this ensures that players pair up with their respective scores. If the score is not larger than the smallest value already stored then the score is discarded. Binary101 only allows for the top 6 scores to be recorded as these are the most significant in there of progression.

With new scores being added to the array regularly a sorting algorithm must be utilised to ensure that the scores are displayed in the correct order. A bubble sort technique was decided to perform this operation. See appendix 9 for full code.

With activities and applications being killed after use the scores page will automatically revert to an original empty state unless you programmatically ensure that the scores persist after termination. This is achieved through the use of shared preferences. A for loop is utilised to store the arrays of values with their identifier strings. In a similar fashion the values are extracted in the same order using the same identifier strings. This is demonstrated below in figures 11 and 12

```
public void storeIntArray(int[] array) {
    quizPrefs = getSharedPreferences("QUIZPREFS", MODE_PRIVATE);
    SharedPreferences.Editor edit = quizPrefs.edit();
    edit.putInt("Count", array.length);
    int count = 0;
    for (int i : array) {
        edit.putInt("IntValue_" + count++, i);
    }
    edit.commit();
}
```

Figure 11 StoreIntArray Method

```
public int[] getFromPrefs() {
    int[] ret;
    SharedPreferences prefs = getSharedPreferences("QUIZPREFS", Context.MODE_PRIVATE);
    int count = prefs.getInt("Count", 6);
    ret = new int[count];
    for (int i = 0; i < count; i++) {
        ret[i] = prefs.getInt("IntValue_" + i, 0);
    }
    return ret;
}
```

Figure 12 getFromPrefs Method

The Strings of the player names are stored in a similar fashion that can be viewed in appendix 9.

NIM.class

This activity has been developed to introduce a fun way of forcing you to think about binary and to get familiar using it. The game is NIM and there is a strategy that can be adopted to ensure that the user wins the game. Firstly, the rules of the game:

“The traditional game of Nim is played with a number of coins arranged in heaps: the number of coins and heaps is up to you. There are two players. When it's a player's move he or she can take any number of coins from a single heap. They have to take at least one coin, though, and they can't take coins from more than one heap. The winner is the player who makes the last move, so there are no coins left after that move. (Some people play the game the other way around, with the last person to make a move losing the game)” (Marianne Freiberger 2014)

Binary 101 adopts the rule that whoever take the last brick loses. The games layout is built up using collections of buttons to represent the stacks of coins.



Figure 13 NIM GUI

Each button or coin generally performs a similar operation it sets all the coins above it to an unclickable state. It also calls the `setVisibility(INVISIBLE);` method to make it appear to the user that the bricks have been removed. Each button calls the `binaryAddition();` method.

```
public String binaryAddition(int a, int b, int c) {
    String resultString;
    int result;

    result = (a ^ b ^ c);
    resultString = Integer.toBinaryString(result);
    if (resultString.length() == 1) {
        resultString = "000" + resultString;
    }
    if (resultString.length() == 2) {
        resultString = "00" + resultString;
    }
    if (resultString.length() == 3) {
        resultString = "0" + resultString;
    }
    return resultString;
}
```

Figure 14 Binary Addition Method

This method provided the hints button has been pressed sets the textViews which contain the stack totals in binary plus the binary total disregarding carry textView to their current state. The binary total disregarding carry is often called the “NIM sum” as referred to by Gillman and Housman. It is also known as exclusive or and written as $x \oplus y$. The game can be played in two modes, one or two player. Two player mode users take turns to make a selection and pass the handheld device back and forth. One player mode on each click if it the computers turn the method `comptuerMove();` is called. This method utilises Binary101’s Artificial Intelligence (AI) element which will be examined in greater detail later in the report. Once the last brick of a stack is removed the a method is called `checkIfWon();` this method checks if the others stacks are empty and if so notifies the user that there has been a winner. The `showWinningMessage();` method carries out this task and an presents the User with three options if they are playing 1

player mode. The three options include 'Save score?', 'Play again?' and No. Save score option passes the users' time in which they took to beat the computer to the ScoreActivity.java this activity performs in much the same ways as performanceActivity.java discussed above. The 'Play again?' Option sets up the game for another round while keeping track of that session scores between the player and the computer. Other methods used within this class are the switchPlayer(); this class is used to set up the graphics to indicate to the user whose turn it currently is. It achieves this by adjusting the alpha or opacity of the

```
public void switchPlayer() {  
    if (player1 == true) {  
        player2_icon.setImageAlpha(255);  
        player1_icon.setImageAlpha(75);  
    }  
    if (player1 == false) {  
        player1_icon.setImageAlpha(255);  
        player2_icon.setImageAlpha(75);  
    }  
    player1 = !player1;  
}
```

Figure 15 Switch player Method

The player Icon who is not in play, their icon appears transparent. Along with informing the user it also helps binary101 keep track of which players' turn it is through the use of a Boolean. The Buttons of each stack are stored as an array of buttons this allows the stacks to be passed as parameters to the computerMove(); method so that that the computers AI can make a decision on which button to select.

```
tile3_1 = (Button) findViewById(R.id.tile3_1);  
tile3_2 = (Button) findViewById(R.id.tile3_2);  
tile3_3 = (Button) findViewById(R.id.tile3_3);  
tile3_4 = (Button) findViewById(R.id.tile3_4);  
tile3_5 = (Button) findViewById(R.id.tile3_5);  
final Button[] rightStack = {tile3_1, tile3_2, tile3_3, tile3_4, tile3_5};
```

Figure 16 Button Array

This allows the computer to make a selection based on the index position within the array.

ScoreActivity.java

This class works in exactly the same principles as the PerformanceAcvitiy.java only difference is that the NIM.java passes a time not a score value.

BinaryCalculator.java

This class serves a tool or utility to assist the users when doing binary calculations. The class utilizes a simple system by which two editText fields accepts two binary operands. There are four ImageButtons which when clicked perform the 4 mathematical operations add, subtract,

multiply and divide. When a button is clicked the text is extracted from the editTexts using `.getText()`; converted to Integers using `Integer.parseInt()`; then the mathematical operation is carried out. Once an answer is determined the answer is converted to a binary string using the built in method `Integer.convertToBinaryString()`; and the answer textView is set to this value. Several error handling techniques are used to ensure the app never crashes. The class makes use of 'Try catch' to try the code first to see if it produces an error. If an error is detected the user is presented with a toast to inform them that there is an issue with a suggestion on how to rectify the error. Possible errors that are avoided include: producing a negative answer, a number not being in binary format and a null pointer error due to an empty field.

NumberConverterActivity.java

This class provides binary101 with the functionality to convert numbers from different bases. Firstly, two instances of spinners are initialised these provide the dropdown menu functionality to allow the input and output formats to be changed easily. The 3 options in the spinners are decimal, binary and hexadecimal. These options are stored in the string.xml file as a string array and are referenced from the layout.xml file. An `onItemSelectedListener()` method is used and depending on what is selected it sets an int variable `inRadix` and `outRadix` to that value. The variable will have a value of two for binary selected, 10 for decimal and 16 for hex, these radices are then used in the conversion algorithm and the corresponding views are set up.

```
findViewById(R.id.convertButton).setOnClickListener((v) -> {  
    buttonClick.start();  
  
    try {  
        String inputString = input.getText().toString();  
        int inputInt = Integer.parseInt(inputString, inRadix);  
        String answer = Integer.toString(inputInt, outRadix);  
        output.setText(answer);  
    }  
    catch (NumberFormatException e) {Toast.makeText(NumberConverterActivity.this, "Please ensure number is base" + inRadix, Toast.LENGTH_LONG).show();}
```

Figure 17 Convert Button Method

Common code Elements

All activities had functionality to return to the home screen. This was implemented via the use of the menu bar available from the Java libraries. This ensures that Binary101 is easily navigable and eliminates laborious pressing of the back button.

```
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();
    Intent intent;

    //noinspection SimplifiableIfStatement
    switch (id) {
        case R.id.goToLearnButton:
            intent = new Intent(this, MainActivity.class);
            startActivity(intent);

            Toast.makeText(getApplicationContext(), "Binary 101", Toast.LENGTH_SHORT).show();

            break;
    }
    return super.onOptionsItemSelected(item);
}
```

Figure 18 Menu Bar Method

Artificial intelligence

"The term "artificial intelligence" is applied when a machine mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving"."(R, Stuart J,P Norvigr 2009)

The Binary101 NIM.class file implements an Artificial Intelligence (AI) element. When the 1 player mode is selected each time the user removes a tile the computerMove(); method. A while loop is used to ensure that the computer always takes a brick. A random number generator is used to randomly select a stack then an if statement is used to determine that if the other stacks in the game are empty and the computer can take the second to last brick it will if not it will take the last. This means the computer makes a conscious decision not to beat itself by its own move thus showing decision making and artificial intelligence. The full method can be seen in Appendix 9

9. Testing and evaluation

Introduction

This section covers the testing stage of Binary101 as a whole and discusses whether the main functional elements which were proposed in Chapter 1 have been delivered. Are the user stories that were developed from the questionnaire satisfied? If so binary101 will be evaluated for its intuitiveness and effectiveness. If any user stories where not satisfied, then the reason why will be examined. As with any software development project the project will undergo different stages of testing, Binary101 adopted this strategy. Throughout the project continuous testing

was carried out as well as a prototype being put to a focus group as a final testing and analysis to determine overall success with the app.

Developer Testing

Testing throughout the project took place on a Samsung Galaxy S6, Android 6.0.1 Marshmallow operating system and 5.10-inch touchscreen. Throughout the development project an AGILE methodology was adhered to and thus the testing approach followed similar principles. After each iteration the code was tested to ensure a smooth progression in development and avoiding any lengthy rebuilds to code.

The Theory section of the app function as expected. All pages can be accessed and the final page provides a button back to the learn menu page useful if user is progressing through the learning section linearly. If the user was just on the theory section to clarify some information, then they can exit to home screen via the menu bar.

The tutorial section displays all relevant information on how to convert from decimal to binary whilst also providing a worked example followed by the procedural steps. All Views display correctly with no problems with extents or problems rendering. The invert button works as expected and easily toggles between conversion directions.

The video page functions fully with the video playing on pressing of the required button. The media player has all functionality need with the pause/resume button and the search buttons working fine allowing the user to step through the video to a desire section if needed.

Play NIM section works fine. All buttons/tiles are clickable and carryout the tasks as expected. However, on two player mode there is the issue that if a player takes their go and proceeds to take another turn before the computer has its turn then the game gets out of sync and the computer tries to catch up on turns resulting in the game to be invalid. This seems to be a concurrency issue that may be resolved with the use of threads and mutex locks. All other functionality provided within the game works well. The 'choose game type prompt' sets the game into one or two player with no issues. The hints button displays the toast explaining the binary strategy to win. If the Player ends their turn each time with the NIM sum = 0001 then the opposing player cannot win. It also displays the stack totals with the NIM sum total to aid the player in their binary conversion and additions until they reach a sufficient level where the hint button is no longer needed. There was an issue in the final stage of the game development by where the game continued to make moves on fresh game board. This was addressed by the use

of a conditional statement to ensure that the game only takes a move if these conditions are met one being the game is not won.

```
checkIfWon();  
  
if ((!player1) && (computerVs) && (!won)) {  
    comuterMove(leftStack, middleStack, rightStack);  
}
```

Figure 19 Check If Won Method

Session Score tracking working as expected and clears once a game session is ended.

Score activity achieves its purpose and satisfies the user story where they requested a scoring or leader board system to track progress. All scores are recorded and stored and persist after the user session is terminated which is essential if users of binary101 need to monitor performance over a larger period of time. One minor issues uncovered through testing is that the score times recorded in the score activity are between one and two second larger than the time displayed when the game of NIM is won. It is the developers' belief that the score records a time greater than shown by one second and a certain number of milliseconds but this time consistent. The ambiguity between the one and two second differences is likely due to the rounding carried out by the JVM. It is felt that this is only a minor issue as this will never result in a longer game time being ranked higher than a shorter time. An issue with the reset score button was highlighted. The button presents the user with a prompt to ensure the button wasn't pressed in error, if you select yes then the reset button must be pressed again to clear the scores. This issue only arose after the addition of the prompt message. The developer feels that it is better to have this minor issue within the prototype rather than the user being able to wipe the score board with a slip of the hand.

BinaryQuiz works to a high level that the User is able to test their knowledge with the theory based question issued the first half of the quiz then their binary conversion and arithmetic is put to the test in the second half through the randomly generated question. This provides a varied approach in terms of testing knowledge and testing skill. The performance score passes to the performance score activity with no issues and the algorithm seems to give a good indication of how well the users has performed taking into account speed and accuracy. The Performance page has same issue with reset as score page but this is deemed acceptable at this stage given the alternative.

The tools provided within the work very well and were even used by the developer throughout the development of other sections of Binary101. Through the development stage there were many issues that caused the applications to crash on these pages however the error handling has worked very well to ensure a seamless user experience. Only issue is that if a user tries to convert overly large numbers then there is not enough room to display the answer. User will be informed that there is an issue through a toast.

Focus Group Testing

A focus group was used to test the Binary101 app. The focus group was split into two categories one category being that of students studying a computer science or software related subject, the other being adults age 16+ with no specialist knowledge of the subject area. The testing procedure consisted that each individual was allotted 15 minutes to use and get familiar with the app then they would be asked to complete a questionnaire rating various aspects of the app on a scale of 1 to 5, 5 being excellent down to 1 being poor. The questionnaire had a final section to allow the test subject to write a short answer whether there was anything they particularly liked or enjoyed in the app or if anything they felt needed a lot of work or could be omitted. The Table 3 shows the questionnaire question and the average score out of 5 for each question. The full breakdown of the Questionnaire results can be found in appendix 9

Questions	Average mark	
	Com students	Non students
Is the Navigation easy to follow with each page easy to reach?	4.7	4.7
Is the theory section well-presented and easy to read?	3.3	2.7
Are the worked examples easy to follow?	4.8	3.5
Is the appearance (colour scheme/layout) appealing?	3.2	4.5
Where you able to follow game instruction easily?	4.3	3.7
Where you able to apply the strategy to beat the computer?	4.3	2.5
Were the Score keeping / Performance pages useful?	4.2	4.2
Did you encounter many bugs or issues? *1 = a lot, 5 = none	3.7	3.7
Was the game enjoyable and fun?	4.0	4.3
Did you find the tools easy and intuitive to use?	4.3	4.3
Was the quiz challenging enough?	2.8	4.2
Would you use the application in the future?	Yes -4 Maybe - 2 No - 0	Yes -5 Maybe - 1 No - 0

Table 3 Focus Group Testing Questionnaire

Analysing and evaluating the metrics from the focus group it can be seen that the application was a success as a whole with some areas out performing others. Also it is apparent that the two categories of participants in the focus had different trends when it came to what they felt where the best aspects and what needed improving. The computer Science group felt that the

theory and examples were easy to follow and well-presented but it is obvious that non CS adults had more difficulty following.

This being said the worked examples performed well in both groups, the delivery of the theory could be improved. The lowest scoring metric was the “Where you able to apply the strategy to beat the computer?” but only with the non-student group. Upon asking a non-student test subject about this they said “I understand the strategy it’s just more difficult in practice, maybe with more time”. In the questionnaire there was a short answer question that provided the test subjects a platform to give feedback. While only a few gave feedback in this section it was very useful to determine what could be improved for future versions of binary101. The responses are listed below.

- “Theory pages could be presented across more pages to give a less clutter look. May be ok on a larger screen”
- “Game really well developed I like the Computer and AI element even if it is basic AI the computer does make informed choice as can be seen when playing the game a few time. Computer very occasionally does not take a turn. Maybe an issue here.”
- “Quiz could be better developed randomise the order of the first 5.”
- “Really liked the how to convert to binary example had no idea how to do it before but now it makes sense”

From The feedback statements a lot was learned about how the public would receive the app if it were to be released to the market in its current state. Testing provides an invaluable platform to iron out issues before it enters the market especially issues that may be over looked by the developer.

Areas that performed well with the majority of the test subject were the navigation. Almost all felt the app was simple and easy to navigate through. The decision in the development stage to minimise the navigation levels seems to have been effective. The tools page proved very useful and easy to use. The computer science students likely found them useful were the non-students were probably more inclined to find them simple and intuitive. Both focus groups felt the app was fun and enjoyable. The performance/ score page received good praise this was one functionality that was deemed necessary from research done in chapter 1 User must be able to monitor the progression and improvement.

Summary

Through the testing process many pitfalls were avoided in the development through the continuous iterative testing. This helped to keep the project on track and within the time frame that was set out. Many deadlines on aspects of the app development shifted but this is the nature of AGILE development. If continuous testing was not implemented I can say with an almost certainty the project would have ran over time. The final stage of testing highlighted any issues missed by the developer especially the non-technical issues that can be over looked. By adopting the binary strategy to beat the game NIM under performed with the non-computer science students but this is more likely an issue with the testing format as subjects only had 15 minutes to get their head around the app. The binary conversion and arithmetic are skills that are developed over time, skills that one focus group possessed and the other didn't. With time Binary101 would get the non-skilled group up to the level of the other faster compared to say a group who didn't have access to binary101.

10. Conclusions

This chapter draws to on the objective outlined in Chapter 1 and aims to examine to what extent these were addressed and developed. The Development process to which was adhered to throughout will be critiqued including testing and recommendations for future development will be put forward.

Evaluations of objectives

It is highly important to review your objectives from the beginning of the project to assess whether the project was a success however it is noted that some objectives tend to evolved with the project and these evolutions will be discussed.

- **To carry out an in depth research into similar apps available whilst analysing the drawbacks of these apps and propose additional features to combat these short comings and to apply these in the development of the app to which this report lends itself to.**

From the extensive market research, it was deduced that many of the apps out their currently were falling short in some aspect of teaching or testing whether that was not having the functionality to test performance or being too dull to engage with their users

to ensure repeated use. From data collected from focus groups it has been determined that the majority of users would use the app again and that they felt that the app was fun and engaging.

- **To utilize a multi-layered learning approach backed by psychological research to ensure maximum learning and retention of knowledge and skills. This approach will also allow the app to be utilized and be more effective to a broader demographic as a learning tool.**

Throughout development binary101 was able to incorporate a variety of learning and teaching techniques which research shows leads to better retention of knowledge. Binary101 provides a video based tutorial along with simple step by step worked examples which was met with great response by the focus group. This indicates that the information was easy to follow and improved understanding of the subject. The game NIM the allowed users to solidify this knowledge in a game based scenario. Computer based student were able to implement their knowledge easily while non student required a little more time to develop their skill.

- **A focus group survey will be carried out to determine most popular aspects of the app from aesthetics and functionality to uniqueness of features.**

Two Focus groups were utilized at two different stages in development. One at the beginning used to generate user stories to determine what potential users would want and expect from the app. Then in the final testing stage to determine how well these user stories were addressed and how well the app prototype functions in the real world. This object has been achieved

- **The app should have multiple features including the use of a forum based question and answer section, an interactive exercise feature to test and practice skills and a theory based research section to broaden the user knowledge. The app is to also make use of SQL database features.**

The forum based question and answer section was dropped for several reasons:

- Lack of interest from the focus group.
- It required the use of an online web server hosting facility.
- It did not really add anything two the app as issues that may come up would never really be that complex.

The other elements in this objective were achieved. Binary Quiz allows users to test their knowledge and skill while making use of the SQLite database features. NIM helps

users develop the speed at which they can convert to binary and how to apply the 'exclusive or' addition.

- **To develop an aesthetically pleasing app that engages the user to minimize the likelihood of them losing interest and becoming bored.**

The 4.5/5 and 3.2/5 were the average scores from the focus group regarding the ascetics of the app this suggests that the overall the apps look was a success however it is noted that the computer science group who scored it 3.2 were not as convinced. They may have felt that the apps appearance may have come across as juvenile this is something that would need to be discussed for future versions.

From this evaluation it can be seen that for the most part the objectives laid out in chapter one have been achieved.

Project Overview

In my own view Binary101 is a sophisticated approach to bridging the shortfalls of the learning binary apps out there. The metrics of the focus group support this view. Along with drawing all the crucial elements for effective learning together binary101 provides some key functionality that sets it apart from other apps out there. Key standout elements include the one player functionality of the NIM game which implements a degree of artificial intelligence (AI) see appendix 9 for full AI code. This makes the game a much more useful tool as it allows the users to practice and hone their skills anytime any place. The computer science students really grasped the level of complexity this aspect of the app entailed, the complexity was lost on non-students who just seen it as a fun element in the game. Either way the NIM game really added a level of complexity increased the usefulness of the app immensely. The Quiz implements SQLite - a powerful feature to allow a vast database of questions to be added to the app if it were brought up to full functionality. The performance ranking system which made use of searching and sorting algorithms along with shared preferences to store the arrays used to hold names and scores. This serves as an important aspect to binary101 to give users confirmation that they are improving with their binary knowledge and skills.

The project from the beginning has encouraged the developer to advance their skills in numerous software packages and computer languages. Packages as such as Android Studio and object orientated languages (OOP) Java along with other packages Photoshop, SQL, MS Excel

and others. From this the invaluable experience gained is evident and goes to show how important and beneficial the individual projects are to budding software developers. The success of the project comes down to several aspects, setting clear defined goals and objectives in the early stages of development sets out a clear path and direction to work towards. An in-depth and intense research phase from the outset helped outline the problem statement and highlighted key areas in development needed to address the problem. A well-structured software development life cycle (SDLC) which followed the agile methodology which helped the developer keep on top of the work load and ahead of the schedule. Regular meeting with their project supervisor to discuss issues and ensure project was progressing at a steady rate. The developer received invaluable guidance from their supervisors on which direction to take and where to focus their efforts. These aspects and many more culminated in the success of the project but as with any project there is scope for improvement which only comes to light through participating in a project of this nature and through the evaluation of its merits. Recommendations will be put forward on how the project could be advanced to a higher level in the next section.

Recommendations

Although the specified objectives outlined previously have been met, the focus groups feedback and experience gained by the developer highlighted areas that could be developed further should the time and resources become available.

- The theory section would be developed with a tabbed interface to allow for less text and information to be on the screen at one time. It would function like swiping between pages in a book. This would make it easier to read and follow on smaller devices.
- Randomly chose questions from the database in the quiz section would provide a better testing experience and discourage memorisation of the questions.
- Develop the 2 player NIM game to work via Bluetooth to allow it to be played on separated devices.
- Develop a difficulty selection to be applied to the game NIM i.e. easy, medium, hard. This would be achieved using a random number generator to generate the probability of the computer making the best possible move.
- Binary calculator would be developed implementing two complement to allow it to deal with negative and decimal numbers.

- A more sophisticated artificial intelligence would be developed.
- Develop the app to work on a large variety of devices i.e. tablets, touch screen notebooks etc.

Summary

The project was developed to deliver a holistic learning tool which provided the all in one solution to learning from delivery of content, testing content retention and provide a platform to apply the knowledge gained. This was achieved through a fun and lively app that all test subjects would use again.

9. References

Android. (2016). *Dashboards*. Available: <http://developer.android.com/intl/zh-cn/about/dashboards/index.html>. Last accessed 20/01/16.

Compuware. (2012). *Mobile App Survey*. Available: https://info.dynatrace.com/rs/compuware/images/Mobile_App_Survey_Report.pdf. Last accessed 26/04/2016.

Department of Education. (2012). *Teaching and school leadership*. Available: <https://www.gov.uk/government/news/new-industry-backed-plans-to-boost-computer-science-teaching-and-help-britain-compete-in-the-world>. Last accessed 20/01/16.

Draganski, B., Gaser, C., Busch, V., & Schuierer, G (2004). *Neuroplasticity: Changes in grey matter induced by training*. *Nature*. . . p.311-312.

Judy Willis (2010). *Learning to love Math*. London: Association for Supervision & Curriculum Development. Chapter 1.

Liz Hollingworth and Andrew Harvey-Price . (2013). *Technology and Skills in the Digital Industries*. Available: <http://webarchive.nationalarchives.gov.uk/20140108090250/http://www.ukces.org.uk/assets/ukces/docs/publications/evidence-report-73-technology-skills-digital-industries.pdf>. Last accessed 20/01/16.

Marianne Freiberger. (2014). *Play to win with nim*. Available: <https://plus.maths.org/content/play-win-nim>. Last accessed 25/07/16.

Microsoft (2009). *Microsoft Application Architecture Guide, 2nd Edition*. 2nd ed. n/a: Microsoft Press. Software & Architecture: Chapter 1 p.1.

Office for National Statistics. (2015). *Internet Access – Households and Individuals, 2015*. Available: <http://www.ons.gov.uk/ons/rel/rdit2/internet-access---households-and-individuals/2015/index.html>. Last accessed 17th Jan 2016.

Richa Jane. (0/04/15). *7 Best Practices for Designing a Mobile User Experience*. Available: <https://www.sitepoint.com/7-best-practices-designing-mobile-user-experience/>. Last accessed 15/06/16.

Rick Gillman, David Housman. (2009). Deterministic Games. In: *Models of Conflict and Cooperation*. USA: AMS. 44.

Rosen, L.D. (2010). *Rewired: Understanding the iGeneration and the way they learn*. U.S.A.; Palgrave MacMillan. p. 205

Russell, Stuart J.; Norvig, Peter . (2009). AI applications widely used behind the scenes. In: n/a , *Artificial Intelligence: A Modern Approach*. New Jersey: n/a. p.2.

Other websites used

http://www.techotopia.com/images/1/1b/Android_architecture2.png

<http://www.sagewoodwellness.com/wp-content/uploads/2015/03/Why-Sitting-Is-Bad-For-Your-Health-Sitting-Posture-Sagewood-Wellness-Center.jpg>

<http://psychology.about.com/od/educationalpsychology/tp/effective-learning.htm>

<http://www.quotationspage.com/quote/40028.html>

https://www.developerfeed.com/simple-quiz-game-andriod/#disqus_thread

<http://www.freesfx.co.uk/sfx/whistle>

<http://mp3cut.net/>

http://www.electronics-tutorials.ws/binary/bin_2.html

Appendices

Appendix 1: Rich Picture Solution

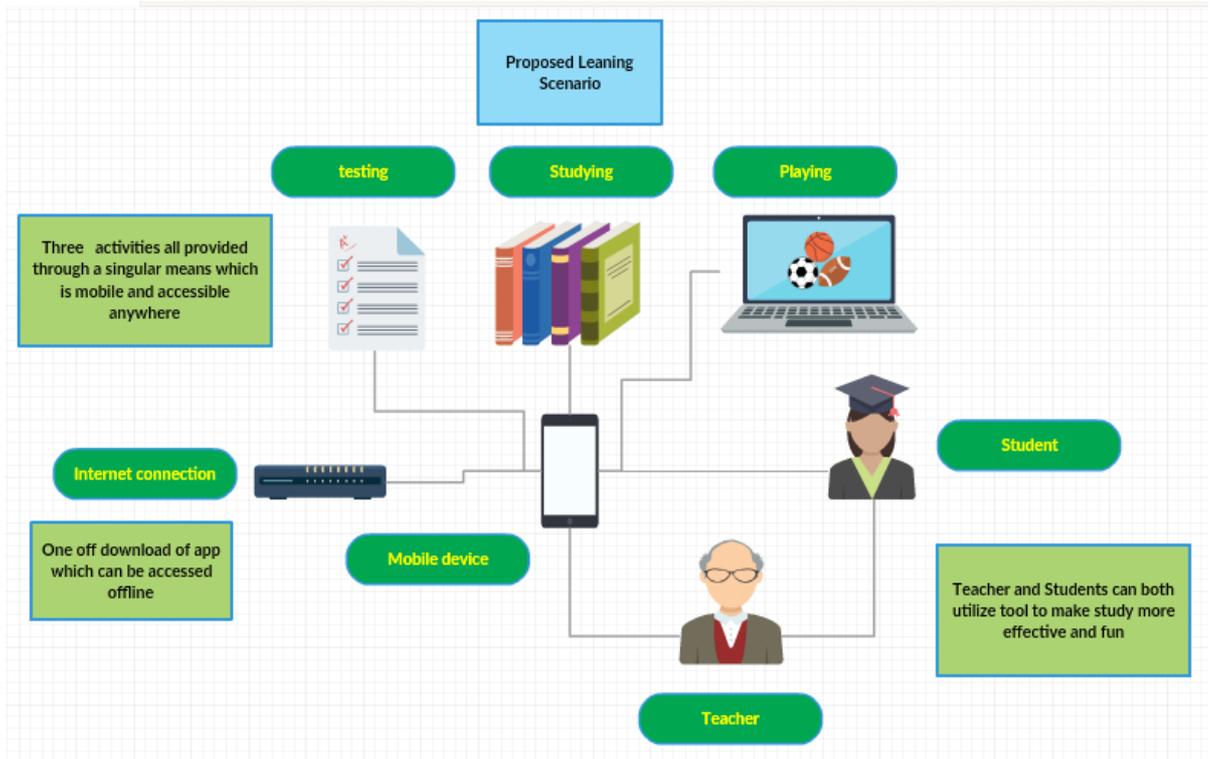


Figure 20: Rich picture diagram showing proposed solution

Appendix 2: Questionnaire

QUESTIONS RESPONSES 20

Section 1 of 2

Binary101 Expectations

Please take the time to fill out the following short questionnaire. The aim is determine what the public feel a binary based mobile learning application should provide. Appearance and navigation elements will also be covered.

What functional elements are important in a learning app?

Please rate each possible functional elements that could be included in Binary101. 5 most important 1 being least important.

Text based theory (similar to textbook learning)

1 2 3 4 5

Worked examples

1 2 3 4 5

Video instruction/tutorials

1 2 3 4 5

Testing facility (quiz/examination)

1 2 3 4 5

A way to monitor progression/performance

1 2 3 4 5

A peer to peer forum where questions could be put forward

1 2 3 4 5

...

Fun or novelty element

1 2 3 4 5

Profile/sign in account system to save your information

1 2 3 4 5

○ ○ ○ ○ ○

After section 1 Continue to next section

Section 2 of 2

App Appearance and User Interface

Description (optional)

What are the maximum levels of navigation you would navigate through to reach a page

1-3

3-5

5+

Best way to navigate to previous page or home screen?

Android back button

On Screen back buttons

Menu bar return to homescreen button accessible everywhere

⋮

Which color scheme below is most appealing

Option 1

Option 2

Option 3

⋮

Option 1



⋮

Option 2





Option 3



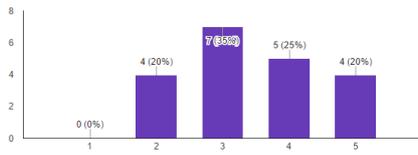
Which type of interface is most appealing

- Tiled
- Tabbed
- Distinct pages

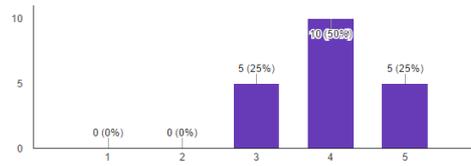
Appendix 3: Questionnaire results

What functional elements are important in a learning app?

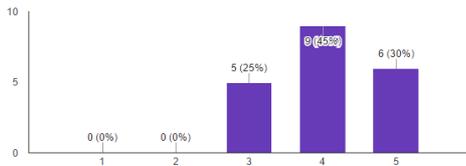
Text based theory (similar to textbook learning) (20 responses)



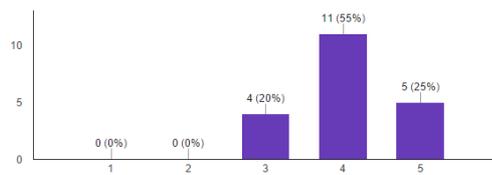
Testing facility (quiz/examination) (20 responses)



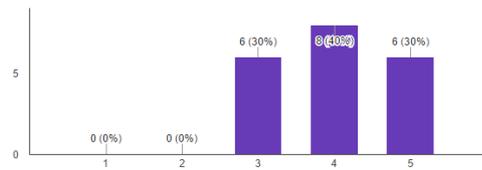
Worked examples (20 responses)



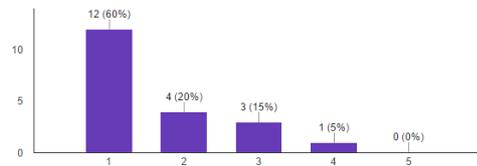
A way to monitor progression/performance (20 responses)



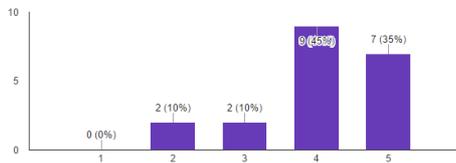
Video instruction/tutorials (20 responses)



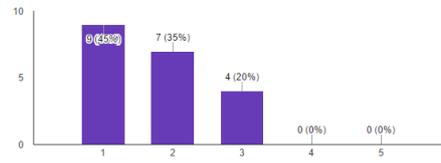
A peer to peer forum where questions could be put forward (20 responses)



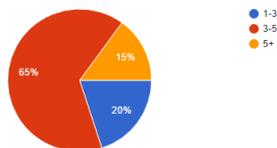
Fun or novelty element (20 responses)



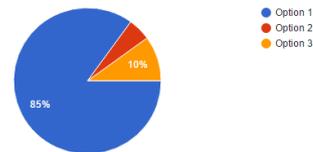
Profile/sign in account system to save your information (20 responses)



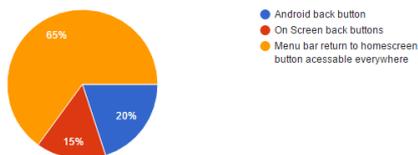
What are the maximum levels of navigation you would navigate through to reach a page (20 responses)



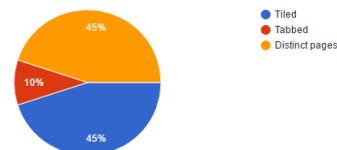
Which color scheme below is most appealing (20 responses)



Best way to navigate to previous page or home screen? (20 responses)

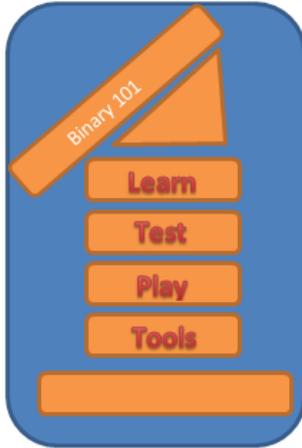


Which type of interface is most appealing (20 responses)

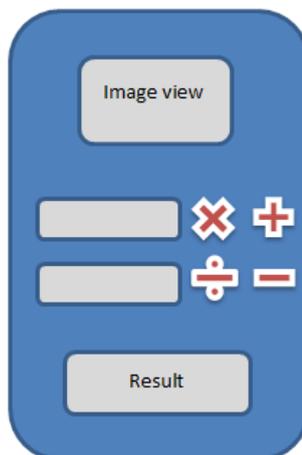


Appendix 4: Initial mock ups

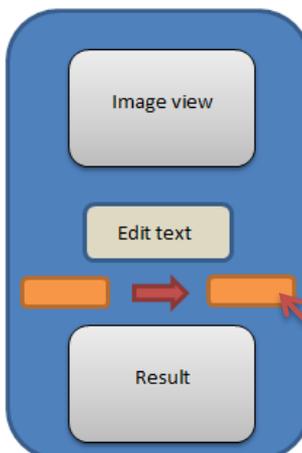
Below are some initial layout designs mock ups.



Here is one possible design template for the menu screen. I used the layout in the shape of the digit 1 to tie in with the binary theme and used contrasting colours to make a bright and engaging interface that is easy to see.



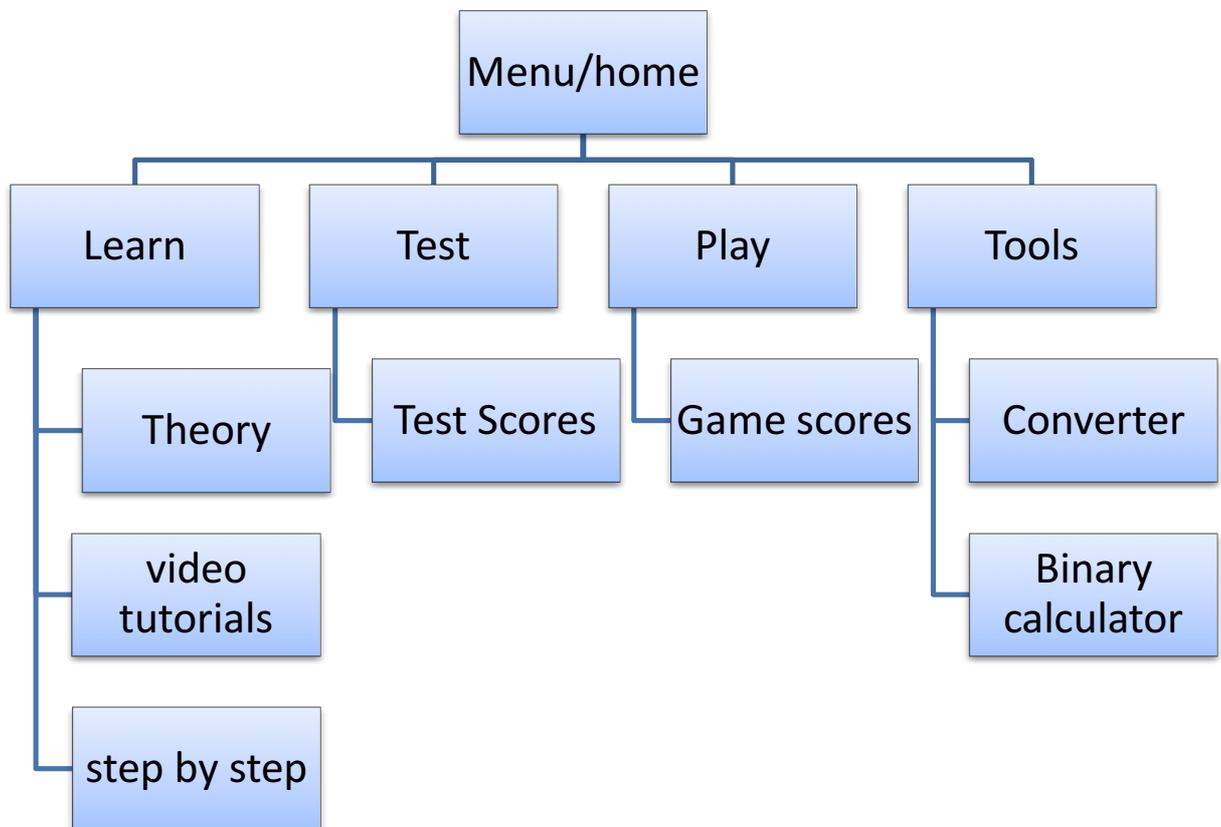
This is the proposed template for the binary operation calculator. A number is inserted in each of the blank boxes and an operator is chosen with the output displayed below.



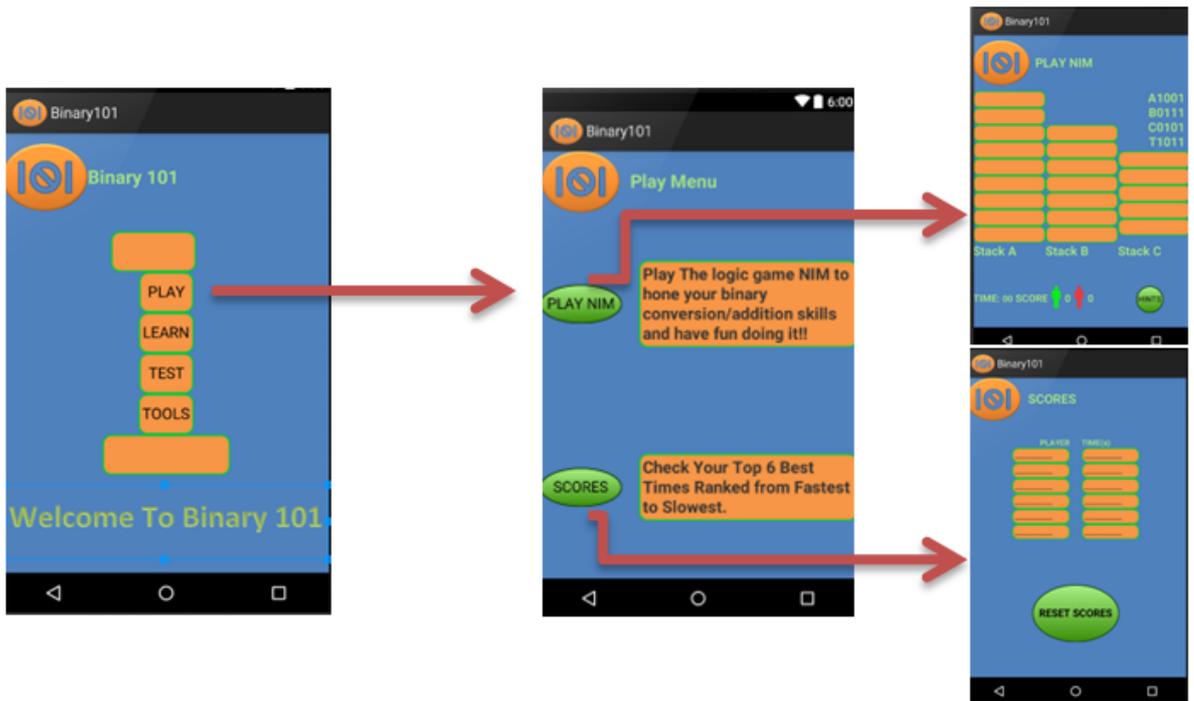
This is the template for the Number converter. A number is inserted in the edit text box and a format for conversion is chosen from the dropdown menus. The result is displayed below.

Dropdown menu for choosing numbering format

Appendix 5: App Navigation Tree

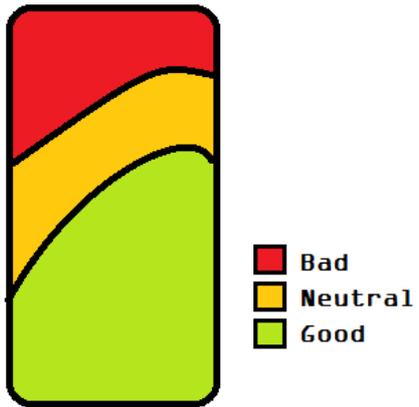


Appendix 6: Navigation and GUI diagram



 Navigation Path

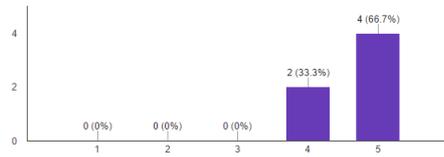
Appendix 7: Button placement diagram



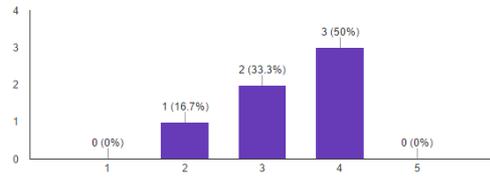
Appendix 8: Focus Group Testing Questionnaire Results

Computer Science Students

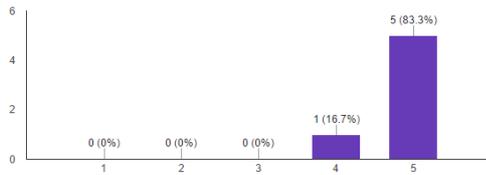
Was the navigation easy to follow with each page easy to reach? (6 responses)



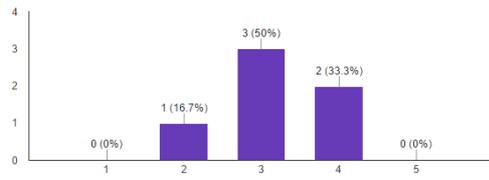
Is the theory section well-presented and easy to read? (6 responses)



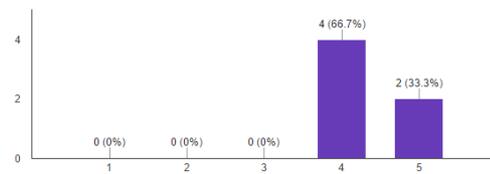
Are the worked examples easy to follow? (6 responses)



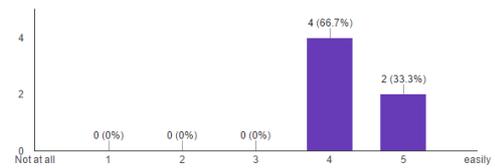
Is the appearance (colour scheme/layout) appealing? (6 responses)



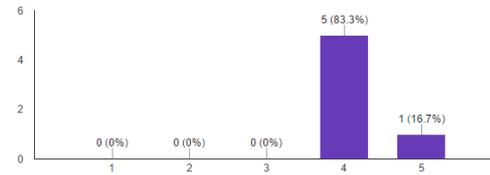
Where you able to follow game instruction easily? (6 responses)



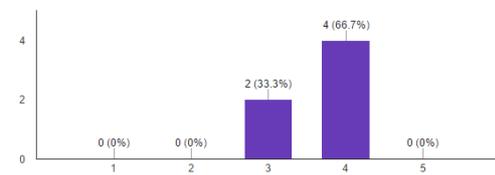
Where you able to apply the strategy to beat the computer? (6 responses)



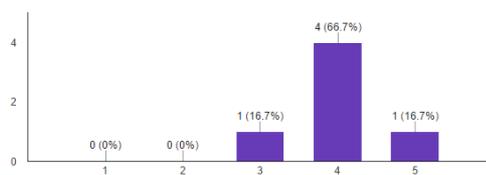
Were the Score keeping / Performance pages useful? (6 responses)



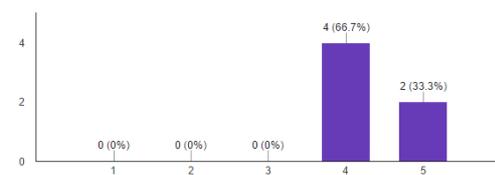
Did you encounter many bugs or issues? *1 = a lot 5 = none (6 responses)



Was the game enjoyable and fun? (6 responses)



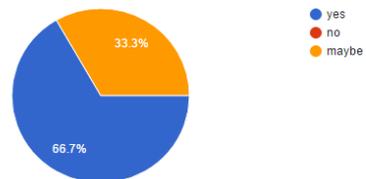
Did you find the tools easy and intuitive to use? (6 responses)



Was the quiz challenging enough? (6 responses)

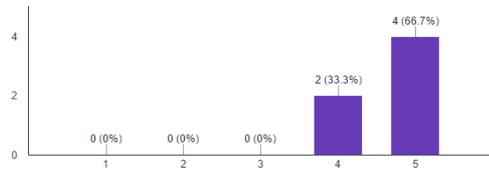


Would you use the application in the future? (6 responses)

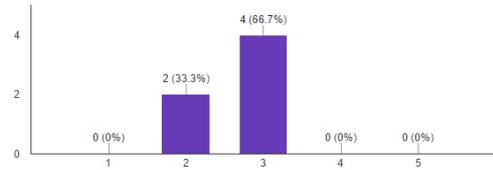


Non computer science adults

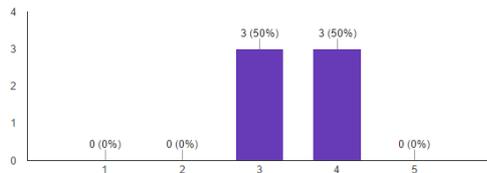
Was the navigation easy to follow with each page easy to reach? (6 resp)



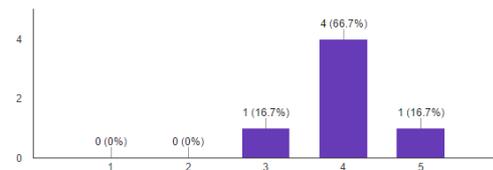
Is the theory section well-presented and easy to read? (6 responses)



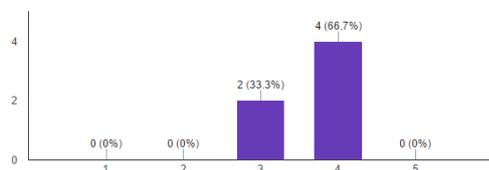
Are the worked examples easy to follow? (6 responses)



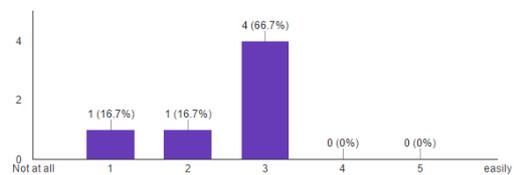
Is the appearance (colour scheme/layout) appealing? (6 responses)



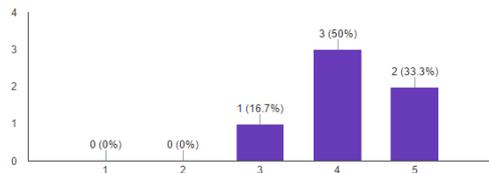
Where you able to follow game instruction easily? (6 responses)



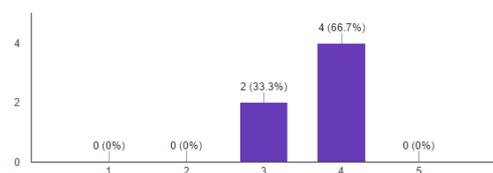
Where you able to apply the strategy to beat the computer? (6 responses)



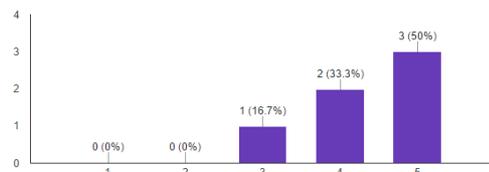
Were the Score keeping / Performance pages useful? (6 responses)



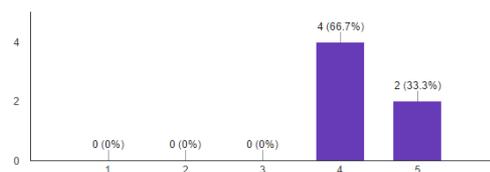
Did you encounter many bugs or issues? *1 = a lot 5 = none (6 responses)



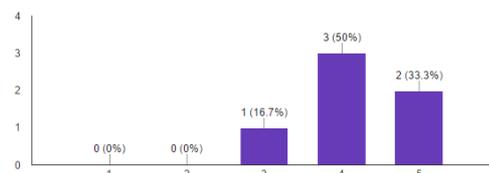
Was the game enjoyable and fun? (6 responses)



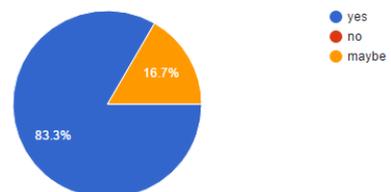
Did you find the tools easy and intuitive to use? (6 responses)



Was the quiz challenging enough? (6 responses)



Would you use the application in the future? (6 responses)



Appendix 9: Java code of Key Elements

9.1 ComputerMove method

```

public void comuterMove(final Button[] a, final Button[] b, final Button[] c)
{
    Handler handler = new Handler();
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
            moveMade = false;

            while (moveMade == false) {

                int randomNo = randomInteger(1, 3);

                if ((randomNo == 1) && (!isEmptyA))
                {
                    if ((isEmptyB) && (isEmptyC)) {
                        if (a[7].isClickable()) {
                            a[7].performClick();
                            moveMade = true;
                        }
                        else {a[8].performClick();
                            moveMade = true;}
                    }
                    //if
                else
                {int selection = randomInteger(0, 8);

                    if (a[selection].isClickable() == true) {
                        a[selection].performClick();
                        moveMade = true;
                    }

                    //else
                }

                if ((randomNo == 2) && (!isEmptyB))
                {
                    if ((isEmptyA) && (isEmptyC)) {
                        if (b[5].isClickable()) {
                            b[5].performClick();
                            moveMade = true;}
                        else
                        {
                            b[6].performClick();
                            moveMade = true;
                        }
                    }
                    //if
                else
                {int selection = randomInteger(0, 6);
                    if (b[selection].isClickable() == true) {
                        b[selection].performClick();
                        moveMade = true;
                    }
                    //else
                }

                //if

                if ((randomNo == 3) && (!isEmptyC))
                {
                    if ((isEmptyA) && (isEmptyB)) {
                        if (c[3].isClickable()) {
                            c[3].performClick();

```

```

        moveMade = true;}
    else
    {
        c[4].performClick();
        moveMade = true;
    }

} //if
else
{
    int selection = randomInteger(0, 4);
    if (c[selection].isClickable() == true) {
        c[selection].performClick();
        moveMade = true;
    } //if

} //else
} //if

} //while

}}, 2000);

}

```

9.2 Database helper code

Sourced from (https://www.developerfeed.com/simple-quiz-game-andriod/#disqus_thread)

```

public class DbHelper extends SQLiteOpenHelper {
    private static final int DATABASE_VERSION = 1;
    // Database Name
    private static final String DATABASE_NAME = "triviaQuiz";
    // tasks table name
    private static final String TABLE_QUEST = "quest";
    // tasks Table Columns names
    private static final String KEY_ID = "id";
    private static final String KEY_QUES = "question";
    private static final String KEY_ANSWER = "answer"; //correct option
    private static final String KEY_OPTA = "opta"; //option a
    private static final String KEY_OPTB = "optb"; //option b
    private static final String KEY_OPTC = "optc"; //option c
    private SQLiteDatabase dbase;
    public DbHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        dbase=db;
        String sql = "CREATE TABLE IF NOT EXISTS " + TABLE_QUEST + " ( "
            + KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " + KEY_QUES
            + " TEXT, " + KEY_ANSWER+ " TEXT, "+KEY_OPTA+" TEXT, "
            +KEY_OPTB+" TEXT, "+KEY_OPTC+" TEXT)";
        db.execSQL(sql);
        addQuestions();
    } //db.close();
}
private void addQuestions()
{
    Question q1=new Question("Binary is also known " +

```

```

        " by what name?", "Base10", "Base2", "Base16", "Base2");
    this.addQuestion(q1);
    Question q2=new Question("What is the answer to 5 + 6 " +
        "in binary format?", "1011", "1111", "10001", "1011");
    this.addQuestion(q2);
    Question q3=new Question("Five digits to the left of " +
        "the right most bit represents units of which value?", "16",
"8", "32", "32");
    this.addQuestion(q3);
    Question q4=new Question("The decimal value 16 is represented as" +
        " what value in hexadecimal?", "b", "10", "13", "10");
    this.addQuestion(q4);
    Question q5=new Question("The binary value 1001010 " +
        "is represented in decimal as", "70", "73", "74", "74");
    this.addQuestion(q5);
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldV, int newV) {
// Drop older table if existed
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_QUEST);
// Create tables again
    onCreate(db);
}
// Adding new question
public void addQuestion(Question quest) {
//SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(KEY_QUEST, quest.getQUESTION());
    values.put(KEY_ANSWER, quest.getANSWER());
    values.put(KEY_OPTA, quest.getOPTA());
    values.put(KEY_OPTB, quest.getOPTB());
    values.put(KEY_OPTC, quest.getOPTC());
// Inserting Row
    dbase.insert(TABLE_QUEST, null, values);
}
public List<Question> getAllQuestions() {
    List<Question> quesList = new ArrayList<Question>();
// Select All Query
    String selectQuery = "SELECT * FROM " + TABLE_QUEST;
    dbase=this.getReadableDatabase();
    Cursor cursor = dbase.rawQuery(selectQuery, null);
// looping through all rows and adding to list
    if (cursor.moveToFirst()) {
        do {
            Question quest = new Question();
            quest.setID(cursor.getInt(0));
            quest.setQUESTION(cursor.getString(1));
            quest.setANSWER(cursor.getString(2));
            quest.setOPTA(cursor.getString(3));
            quest.setOPTB(cursor.getString(4));
            quest.setOPTC(cursor.getString(5));
            quesList.add(quest);
        } while (cursor.moveToNext());
    }
// return quest list
    return quesList;
}
public int rowcount()
{
    int row=0;
    String selectQuery = "SELECT * FROM " + TABLE_QUEST;
    SQLiteDatabase db = this.getWritableDatabase();
    Cursor cursor = db.rawQuery(selectQuery, null);
    row=cursor.getCount();
    return row;
}
}

```

9.3 bubble sorting algorithm

```

public void sortScores(int[] scoreArray, String[] names) {
    int temp, pass = 0, last = scoreArray.length - 1;
    boolean exchangeMade;
    String nameTemp;

    do {
        exchangeMade = false;
        pass = pass + 1;

        // Compare adjacent elements up to index position last
        for (int index = 0; index < last; index++) {
            // Exchange adjacent elements if not in order
            if (scoreArray[index] < scoreArray[index + 1]) {
                temp = scoreArray[index];
                scoreArray[index] = scoreArray[index + 1];
                scoreArray[index + 1] = temp;
                nameTemp = names[index];
                names[index] = names[index + 1];
                names[index + 1] = nameTemp;

                exchangeMade = true;
            } //if
        } //for

        last = last - 1;

    } while ((exchangeMade) && (last > 0));
}

```

9.4 Insert Score Method

```

public void insertScore(int[] scores, int mscore, String[] names) {
    //find insert point
    int i = 0;
    while (i < scores.length && scores[i] > mscore) {
        i++;
    }
    if (i < scores.length) {
        //you found a place to insert the score
        for (int j = scores.length - 1; j > i; j--) {
            scores[j] = scores[j - 1];
            names[j] = names[j - 1];
        }
        scores[i] = mscore;
        names[i] = player_from_quiz;
    }
}

```

9.5 Shared Preferences Storing and Retrieving Method

```

public void storeStringArray(String[] array) {
    quizPrefs = getSharedPreferences("QUIZPREFS", MODE_PRIVATE);
    SharedPreferences.Editor names = quizPrefs.edit();
}

```

```
names.putInt("array_size", array.length);
for (int i = 0; i < array.length; i++)
    names.putString("array_" + i, array[i]);
names.commit();

}

public String[] getFromNamePrefs() {
    quizPrefs = getSharedPreferences("QUIZPREFS", MODE_PRIVATE);
    String[] tempArray;
    int size = quizPrefs.getInt("array_size", 6);
    tempArray = new String[size];
    for (int i = 0; i < size; i++) {
        tempArray[i] = quizPrefs.getString("array_" + i, null);
    }
    return tempArray;
}
```