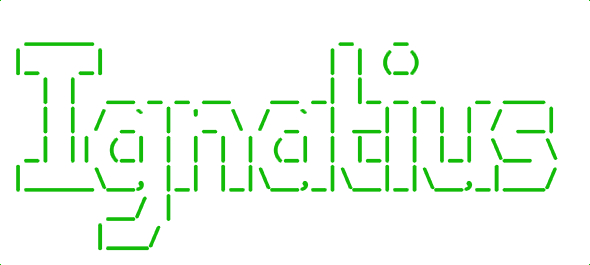
Android Mobile Applcation



Patrick Ware

2021001676

Contents

[1. Introduction 2](#_Toc136949390)

[2. Application features 2](#_Toc136949391)

[2.1 Multi-language support 2](#_Toc136949392)

[2.2 Device Support 3](#_Toc136949393)

[3. Application Design 3](#_Toc136949394)

[3.1 Design Inspiration (Original Console Application) 4](#_Toc136949395)

[3.2 Wireframe Application Sketches 5](#_Toc136949396)

# Introduction

Ignatius Will be created as an android application based on the C# Console program of the same name that was created during the summer break at the end of last year.

The purpose of Ignatius will be used as a study aid for end of term examinations which commonly employ a combination of short answer and multiple-choice questions.

After creating the original program, the next logical step was to upgrade to a platform that offers GUI rather just a terminal and creating the application for mobile means that I can use it anywhere.

# Application features

As Ignatius is based on the console of the same name it is intended that the following features will be available:

* The options to add and remove questions in the quiz bank.
* The option to be quizzed on all questions on your choice of subject.
* The option to be quizzed on all questions in the quiz bank regardless of subject.
* A Custom quiz mode that will users choose an amount of random questions on their choice of subject from the list. To be quizzed on.

## 2.1 Multi-language support

The application will have two languages available for users which will be English and German, the multi-language translation will include the application interface only, the quiz questions and topics will not be translated and will be limited to the language in which the user added them.

## 2.2 Device Support

Ignatius will need to function on a wide range of android devices including phones and tablets with varying screen size

The application will Target API 26 (Android 8.0 Oreo) As this will mean that Ignatius will be supported by almost 91 percent of devices running the android Operating System.   
During the development and testing Ignatius the following virtual and physical devices will be utilized:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Device Model | Device Type | Screen Size | Android Version | Physical OR Virtual |
| Samsung Galaxy A13 | Phone | 6.6” | Android 12  (Snow Cone API 31) | Physical |
| Nexus 10 | Tablet | 10.05” | Android 12 (Snow Cone API 31) | Virtual |
| Pixel 6 | Phone | 6.4” | Android 8 (Oreo API 26) | Virtual |

# Application Design

In regards the design of the applications user interface the intention is for it to laid out in a similar way to the original console application with a banner/app bar image and simple navigation menu to direct users to the desired activity.

## 3.1 Design Inspiration (Original Console Application)

The original console application requires users to enter the number of the option they want. Screenshots of the original applications menus can be found below.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated with medium confidence

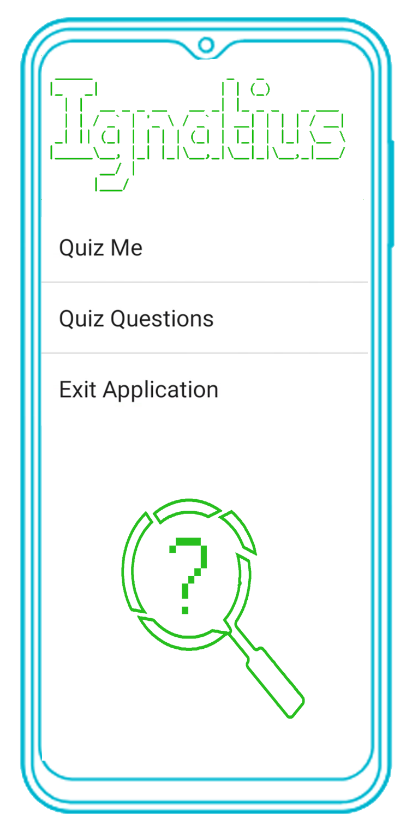
A screenshot of a computer

Description automatically generated

## Wireframe Application Sketches

The basic Wireframe design for how the application will look on a Phone can be seen below

### Home Screen



### Edit Quiz Menu

Upon opening this activity users will be shown a prompt on how to edit the quiz content.

This message will inform them on how to both add new questions to quiz bank and remove old ones

### Quiz topic selection menu

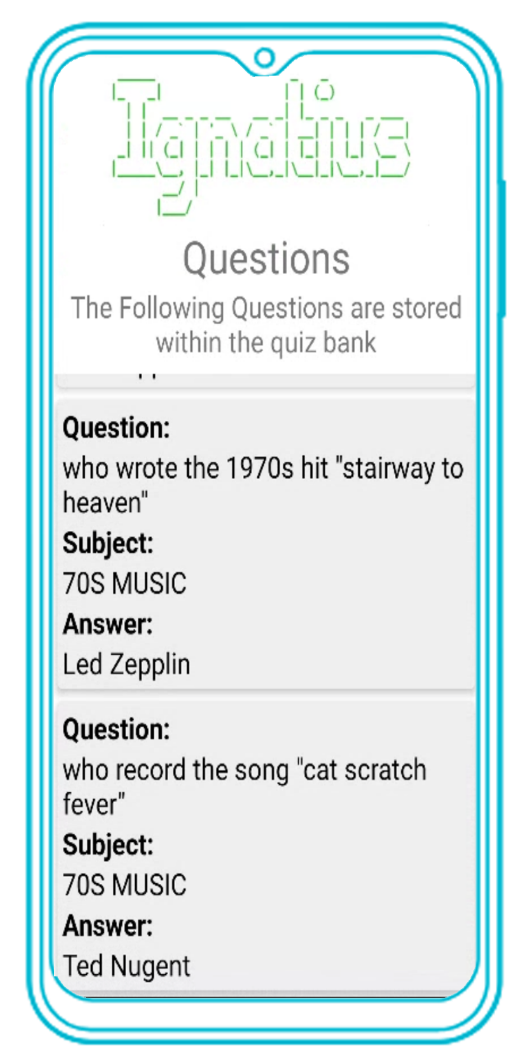


### Quiz Activity

the quiz application will be set to show a question and three possible answers to the user and then inform them whether or not they got the right answer by using a snack bar message for a correct answer and an alert for incorrect answers in which they will be shown their answer and what the correct answer should be.

At the end of the quiz users will be shown screen showing their final score and a button that will return them to the main menu

### 3.2.5 Show All Questions View

all of the questions currently stored in the quiz bank will be shown n recycler to the user when this option is selected from the edit quiz menu.  
  
To avoid showing too much information only the question and the correct answer will be shown to the user.

The use of questions about music in 70s are for demonstration purposes only.

# Database Implementation

The SQLite Database used within the application uses a single table to store the following information and display it as required:

* Question
* Subject
* Option1
* Option2
* Option3
* Correct Answer

Currently the question field in the database is used as the primary key to prevent duplicate questions From being Stored within the quiz bank in order to ensure it is easy for users to delete questions the method that handles data insertion is configured with both the “.toLowerCase()” And “.trim()” method to ensure that all questions are lower case and have no leading or trialing spaces that could errors when interacting with the edit quiz menu.

To ensure that the subject is entered the same way every the the subject field uses the “.toUppercase()” and “.trim()” methods

The option1,option2,option3,correct answer questions only have the “.trim()” method in effect to allow users to enter possible answers that are case sensitive such as programming questions for example, however this means that users have to double check data before submission