Developing App

Produced by Cristofer H

Table of context

[**Introduction**](#_2fg482bybbzm) **2**

[Testing Table](#_4uouwpxql2lg) 3

[Error Documentation](#_6y3i3wlmv035) 4

[Device Testing](#_sytn5nkkt1i8) 4

[**Feedback & Improvement**](#_38flnydgpvwm) **5**

[Optimisation](#_kuaag68y9c0b) 5

[**Evaluation**](#_e3akb3ahsyqk) **7**

[**Development & Optimisation**](#_6fw4bdx6jlwk) **7**

[**Comparison**](#_m8kvffesi2vo) **9**

[**Meeting the requirements**](#_sh6fvd3mx8kd) **10**

[1. The app should run a game where the user has to answer a series of simple random arithmetic questions.](#_rutkgsnujcl3) 10

[2. Subtraction and division should not produce negative answers.](#_4tpx5lc251kl) 11

[3. Each game should have 10 questions and the app should keep track of the score.](#_z5jo0bggsva8)

[Grading Hierarchy](#_s8yrsusq5ilc) 12

[4. The app should include suitable graphics and sounds](#_lxul0ueak21f) 13

[**Design Justification**](#_idhrud1rwyn3) **14**

[Prototype Design](#_2777pc5bnd7g) 14

[Actual Design](#_e4n3n0wh4152) 15

[**Future development**](#_75rkxjtcqe6j) **16**

[**Conclusion**](#_rwt5skwt41ih) **17**

[**Time management**](#_fi2tk4ocozve) **18**

[Tasks](#_siw829trgxvy) 18

# Introduction

This is the assignment on application development. I have delineated everything here in the assignment, please check the page of context to know where each section is.

Functional Testing

### Testing Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test | Method | Expected result | Actual result | Comments |
| 1 | Type an answer and click on continue. | When the user answers a question they can go to the next and in the background, the application will begin marking and assessing his input. | The question will change when the user clicks the continue button and it progresses more difficult after every 10 questions you answer. | Evidence: View the application. |
| 2 | When on the settings panel or page the user can click the reset button. | The application will go blank (fade perhaps) and then the score board and name resets. | N/A | Due to the development time the settings panel couldn't be delivered. |
| 3 | When clicked on the button inside the lower tab it will change the interface. | Should be able to change the UI and return back to the previous page. | When clicked on the start test button the application will redirect you to the gameplay interface. | Evidence: View the application. |
| 4 | Once the user clicks done on the congratulation screen the software will update their current grade to the latest one. | The current grade on the settings page will be replaced. | N/A | Due to the development time the grading couldn't be delivered. |
| 5 | When clicked on any part of the application a sound effect will play to make the application feel interactive. | When a button is clicked a sound effect will play from an mp3 library folder. | N/A | Due to the development time the sound couldn't be delivered. |
| 6 | Stability performance run. | The program should be able to run efficiently without crashing. | The application runs fast without crashing as it progresses through it. | Evidence: View the application. |
| 7 | The timer in the application works. | After the time limit is gone it skips to the next question. | N/A | Due to the development time the timer couldn't be delivered. |
| 8 | No abnormalities or glitches. | The program should do as it states in the design.. | The data displayed on the screen represents the information with no test abnormalities and fits within the ranges of the answer e.g no negative numbers. | Evidence: View the application. |
| 9 | The application will output if the answer is correct or incorrect. | If the correction is right or wrong a widget will show, informing the user their result. | If the question is right or wrong it will either state correct or incorrect. | Evidence: View the application. |
| 10 | The user can input numbers and it will display on the keyboard. | The user can input numbers and it will display on the keyboard. | When I click any number on the keyboard it will automatically display the number and if I click another it will connect the two numbers e.g. 1, 2 = 12. | Evidence: View the application. |
| 11 | The user clicks clear on the keyboard button and it clears the text. | The user clicks clear on the keyboard button and it clears the text. | When clicked the clear button the answer the user submitted will be eliminated. | Evidence: View the application. |

### 

### Error Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Recognition | Resolution | Evidence. |
| 1 | When clicking on start test the program would just crash. | The generation method maxim and minimum value was the other way around that caused the crash. | int Second\_Generation = RandomInt.nextInt(Maximum - Minimum); |
| 2 | The answer would always show up as incorrect. | We had to fetch the information from the text field by fetching it’s ID. | TextView txtanswer = findViewById(R.id.*User\_Response*); |
| 3 | The question would be “what’s 1 null 3?” | Logical error, the program didn’t register the third value as a string; I had it down as an int instead. | public String Question\_Display(int a, int b, String c) |

### 

### Device Testing

|  |  |  |  |
| --- | --- | --- | --- |
| Device name | Evidence | Operation | Function |
| NEXUS |  | It runs as normal with no prior delay or start up issues. | IThe features are normal and don't have any glitches or abnormalities of what it should have done. |
| TABLET | View tablet ↗ | The tablet itself is very slow although it didn’t stop from the program running and lunching. | The program meets its goal and there are no prior crashes when using the application. |

**Operates -** Program can lunch and run perfectly normal.

**Function -** Meets its purpose and features aren't bugged out.

# Feedback & Improvement

I have been asked for a review from a few classmates in my classroom to give external feedback to better accommodate and optimise for the users. There were two main feedback I have gotten, with this information I can add this to the development and optimisation of the program..

*"The application looks good and it's very adaptable and easy to understand, maybe make the continue button smaller perhaps as it takes up the screen." - Classmate*

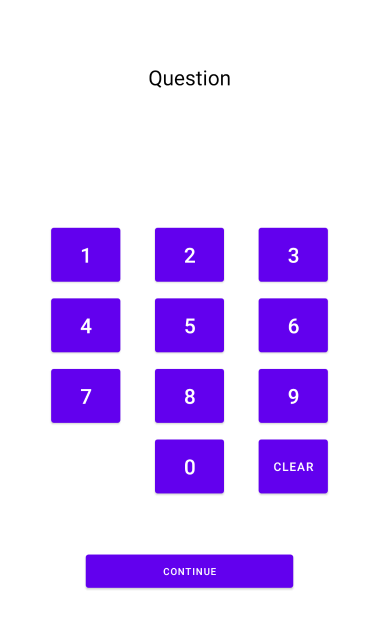
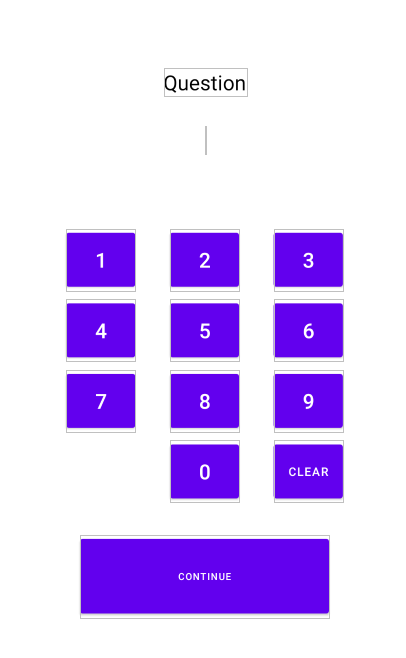
The design of the application is appropriate meaning the UI according to this user is satisfactory, although the button at the continue button was really huge before expanding over 200dp, I reduced this number instead to 100 and decreased it making it the smallest button on the interface to make it easier to read the other buttons and use.

*“I love the oversimplified minimalistic design but I don't like the text box behind the original text because it's very distracting and unneeded space, removing it would make it look cleaner and take up less space.” - Classmate*

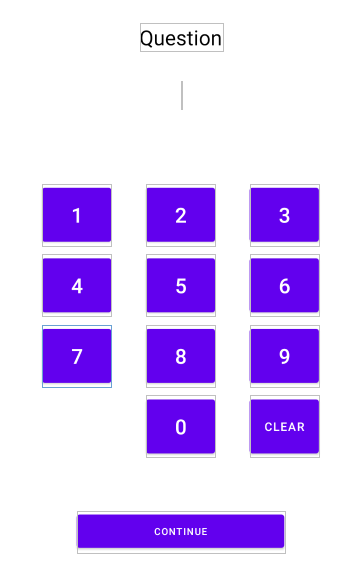
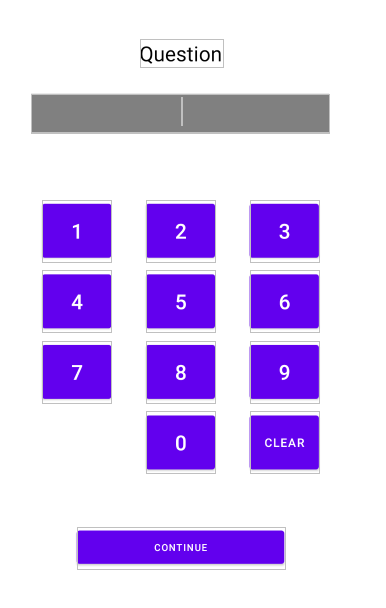
Again this further emphasises the current interface is very good although this reviewer requires the grey answer background on the gameplay interface to be removed from the application, to make it more appealing to the eye and remove the unwanted space.

### Optimisation

According to the feedback, I have applied the changes on making the continue button smaller giving it a cleaner look. At this before and after you can see how clear this implementation made the application feel.



I have also removed the design of the grey background on the application because it looked really distracting so I did and it definitely made the asthmatics look pleasing.



# 

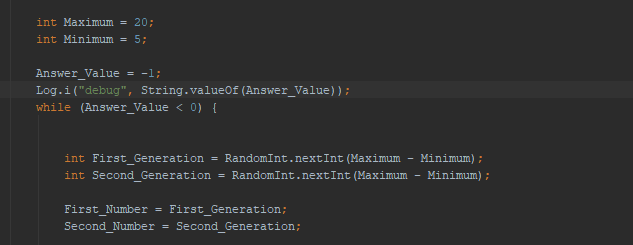
# Evaluation

# Development & Optimisation

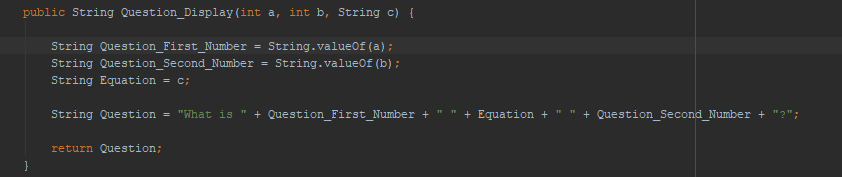
The application was originally intended to have many features that would offer the user a lot of options when working on their arithmetic skills. Although due to the time and delivery of the application it had to have many features cut due to the prior ambition and time left in order to produce a stable and feasible version of the application.

The application was only focused on the main version task which is to allow children to test their arithmetic skills to ensure that they can do basic mathematical operations. This means the home page was developed and the question page was also developed. The other features such as the settings panel and the grading system had to be removed because the development time was set back with the exam boards.

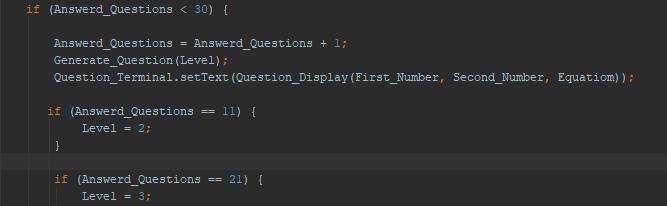
Due to this major restriction, we had to only focus on outing a basic version that follows the guidelines of the benchmark example that was delivered. The naming convention was consistent though the application was similar to its UI design as established in the design phase. The program ran efficiently without any major crashes or delay, this can be due to the classes I utilised and functions that allowed less repetition of the same code meaning fewer lines were occupied and it kept the game working in a stable form. The classes added to the program were one used to generate the question so the numbers it can use and the equations it would be allowed to use. This ensured the questions generated reinforced the requirement of not outputting negative results and the question wouldn't be as complicated from the start but gets harder towards the end.



The second function was used to display these values in the text box after the question would be generated it would then show it, for (example, what's \_\_ + \_\_?.)



The check answer function would check if the answer the user typed matched the actual answer, after outputting the answer the program then would regenerate a new question although after every junction point meaning 10 - 20 - 30 the level would increment by 1. In simple terms, the question progressively gets harder after every ten-question because the generator for the question will add larger and complex operations to the program.



Instead of making it singular and a linear repetition of code until it reaches 30 questions, I made an advanced loop that does this automatically, reducing the reuse of the code. This provides a far better way of function because the more it increments the harder the question gets meaning the level progresses. There was no asset being pulled from the program being developed such as font or music and sounds due to the cut development time.

# Comparison

# 

My solution compared to this application offers more in terms of the way of answering a question because the user can type the question, this means the indirect effect is asking the user to use their mental arithmetic instead of just showing the answer like this did. Furthermore, my solution is more designed for an older target audience preferable like 9 -10 years of age whereas this is younger, this means mine offers a more focused yet less childish more mature look appealing to older people compared to this application. My solution has a minimalistic design meaning it's easier for them to navigate whereas this overwhelms the user with too many interactive features that can be confusing.

Their solution offers more content than mine and they also made their application very initiative by adding many features such as lily pads on numbers which at least keeps their users entertained, they also have animals as mascots appealing to their age group. Taking this on board I can implement some of these features to my application as well to benefit mine further.

In my personal approach, their application is better than mine because it's more developed and advanced, mine was cut short just to output a basic functionality and that's it.

# Meeting the requirements

Client Requirements & Justification

Client requirement will show the given obligation set out from the client that will be required to be added to the program, this will be the default hard coded features in the program and will come in order to satisfy the requirement as soon as possible and deliver what's demanded.

### *1. The app should run a game where the user has to answer a series of simple random arithmetic questions.*

The app should run a game where the user has to answer a series of simple random arithmetic questions and is amended to the requirements, this means every question will have to be randomly generated by a randomisation function. This prevents the event of the children misusing the application by typing the same answer over and over by memory which in effect doesn't help their math skills.

The equations used for this arithmetic are required to use addition, subtraction, multiplication and division. This makes it easier for the children to answer; we are only working on improving the fundamental arithmetic based skills.

The requirement suggests using round numbers from 1 and 12, this means this will help these children develop their time's tables all the way up to twelve. This means the last or maximum question with large numbers can only be 12 \* 12 which is 144. This keeps it manageable and relatively easy and manageable to answer.

*"In maths, it says pupils should be introduced to the two, five and 10 times tables by year two - at the age of six and seven. Between the age of seven and eight, children should start to learn the three, four and eight times tables, the document says."*

*Sport, B., 2021. Nine-year-olds should recite times tables by heart, says Schools Minister. [online] The Telegraph. Available at: <*[*https://www.telegraph.co.uk/education/*](https://www.telegraph.co.uk/education/educationnews/11251221/Nine-year-olds-should-recite-times-tables-by-heart-says-Schools-Minister.html)*> [Accessed 23 April 2021].*

Using this data we can determine that the age of the target audience of this application would be kids around the ages of 9 - 12. This allows us to make extensive alterations to the design to better accommodate that age group.

### *2. Subtraction and division should not produce negative answers.*

Keeping to the requirements the application shouldn't provide any difficult arithmetic type questions by not only appropriating numbers from 0 to 12 for calculations but also not producing a negative output as answers. This will make the application easier to suit more of its demographic which would be children; making negative possible outcomes can make it too difficult for them, likely resulting in the users getting a piece of paper and working out the answer. Which completely defeats the usage of an application that is used to test and examine your mental math skills.

This will mean while generating the answers there will need to be quality control of the output and ensure the outcome isn't less than zero. This means as an example if an answer shown in the next question is -2 the program will be required to generate another question that will have the answer to be more than equal to zero ensuring this requirement is mandatorily met.

### *3. Each game should have 10 questions and the app should keep track of the score.*

The requirement states that each level should have 10 questions per stage or levels, per threshold. This means that the game should at least provide 10 generated questions within all levels, per threshold. The way the application will delineate the level will be the following:

The first level known as level 0 will provide 10 basic arithmetic questions with a no set limit timer as the requirement requests, this level will be an easy based level and will get the users used to the game and get a feeling for the types of questions they will be challenged to answer.

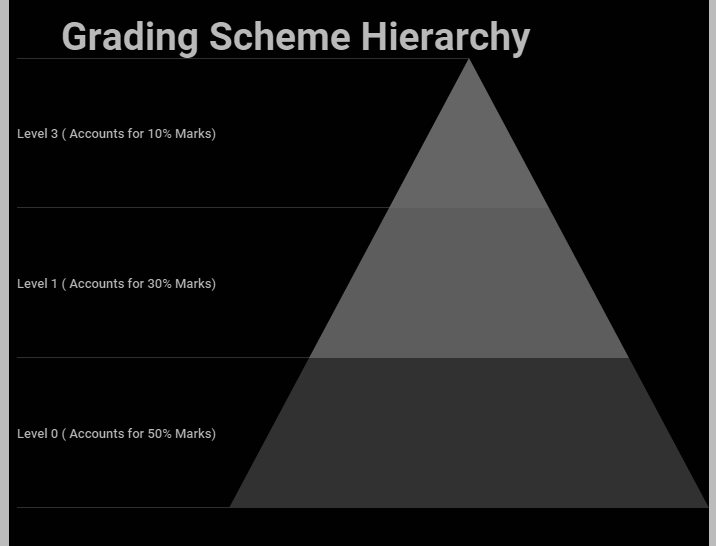
Level 1 will be the first-ever level to start to challenge the user by adding a timer. The timer will be 20 seconds per question so it will provide a plethora of time for the user so it should still be straightforward to answer although the time is now limited; the user has 20 seconds to answer the question. If the user fails to answer the question before the timer hits zero it will automatically skip the user to the next question.

Lastly, Level 2 is the final level of the examination of the application of the user's feedback. This will offer relatively more complex arithmetic based questions with an even smaller time frame to answer in. The user will have up to 10 seconds to answer the difficult questions. After the level is complete the user's information will be kept and stored in the storage of the application.

This data can then be further looked at and diagnosed with performance scans to determine what the output of the user's result was during the test or "gameplay". This will indicate to the parent or career of the child playing this game what their mathematical capabilities are. The application will factor in the levels, level 0 - 1 being fundamental. This suggests that levels 0 will be the most important feedback as it will offer the most basic fundamental math questions such as addition and subtraction.

Level 1 will consist of easy division and multiplication. This will be more advanced compared to level 0 but it will harvest the information of the user about their multiplication and division skills. Then level 2 will offer complex numbers with equations and this will show how advanced the users are dealing with multiplication, addition, division, multiplication on large numbers.

### *Grading Hierarchy*



# 

# 

# 

# 

# 

# 

# 

Fondly enough as established 50% of the grading will account for 50% as this stage is the easiest as it requires the important fundamental mathematical skill. Level 1 will begin to offer more advanced equations such as multiplication and division although again it is fairly easy and shouldn't be difficult. Level 1 shows that you know multiplication and division which is why it offers 30% of the grade. The 10% will be the most advanced question and will offer low marks because the application is only to assess the fundamentals and nothing more.

The grading or track record is designed to be fair, not who can get the "most marks". This means that the application tests you on your knowledge rather than guessing and times you on it as well. Most of the people using this application will have to at least get level 0 and partially level 1 right before getting a pass showing they understand these fundamental components. However, those people who excel at level 3 will get the very few marks required to achieve a high grade in the marking grid.

### *4. The app should include suitable graphics and sounds*

To ensure the application better suits its target audience as a requirement, the user experience would appeal to a younger audience. This would mean by providing a colourful interface that's kept minimalistic would be optimal. This means that not only will the user interface be easy to navigate and operate but it also offers extensive features to meet the requirement making the application fulfil its primary purpose. By providing a simple and minimalistic interface it reduces the chances drastically of users not knowing how to use the application.

Secondly, the colours will have to suit its target audience meaning the main bias of the colour election will have to be bright, although since the application isn't primarily being targeted to solely male or female this means that it will have to remain gender-neutral.

Since the anticipated target audience of the kids utilising the application will be around key stage 2 (7 - 11 years of age.) according to sciencing.com

*"Children tend to be attracted to the bright block colors of the color wheel rather than pastels or muted blends. Primary colors red, yellow and blue, and secondary colors green, orange and purple, are more appealing than light shades of pink and beige or neutral shades of gray and brown.."*

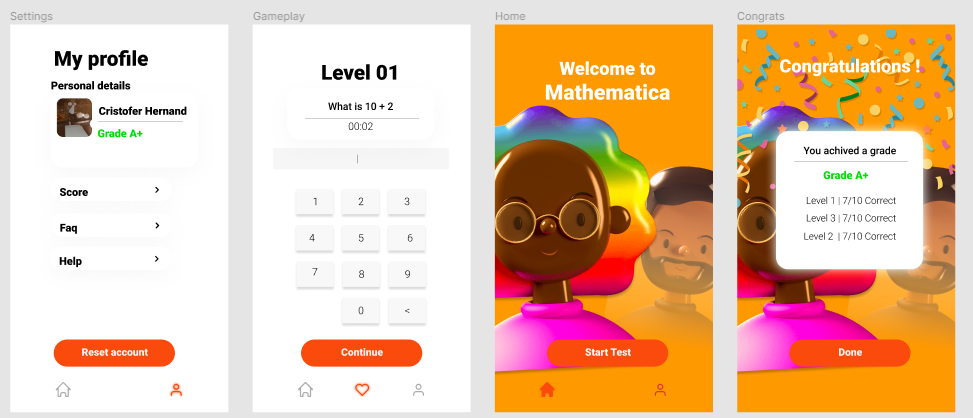
*Sciencing. 2021. How Do Bright Colors Appeal to Kids?. [online] Available at: <*[*https://sciencing.com*](https://sciencing.com/do-bright-colors-appeal-kids-5476948.htm)*l> [Accessed 28 April 2021].*

This means that the colour pallet will have to include either red, yellow or blue. Since the app will primarily have a clean white background with plenty of white space the buttons highlighted can be corrected to the colours that will appeal to the target audience. This means that the audience will be more inclined to select the button due to its colour, this can increase the engagement and performance of the clients when engaging with the software.

Areas such as answers, the start button will have these highlighted to ensure the whole process of examining the target audience can be more natural. Sound effects will also be included such as a tick once a question has been answered however this will have to be thought of heavily as it can become bothersome promptly if the sound effects are awful.

# Design Justification

### Prototype Design



*Hernandez, C., 2021. FIGMA UI Design. [image] Available at: <*[*https://www.figma.com*](https://www.figma.com/file/RcWSb5iGfuyEgQ7fqVebKe/Untitled?node-id=0%3A1)*> [Accessed 28 April 2021].*

I have used FIGMA to create my prototype for the creation of the application. The application contains a simplistic minimalistic user interface that makes the application easy to use and move around. There will only be 2 separate tabs the user can navigate to.

The user can go to their accounts and look at their grade, track record, information, settings. The other is the home tab which will only have one button appealing to the user to "start test".

**Consistency** - The buttons will always be in the same coordinates and will always be present on each page as continue, start test, reset account, done. The application will also have the same font throughout which is Roboto. Again ensuring the requirement suitable graphics is met.

**White Space** - The application will have a clean minimalistic look so it will provide the most white space, this means that the application will be easy to navigate as the buttons are highlighted clearly and there is plenty of space to ensure there is suitable graphics.

**Reversal actions** - There will be no prior reversal actions in the application although it will provide tabs that the user can navigate instead of again keep it simple.

**Colour** - The application has the same colour interface pallet throughout and has a similar layout on each page. Orange will represent a call to action while grey will represent an option that can be selected. The colour better suits the target audience for the application as explained in the requirements, ensuring the demand of suitable graphics is met.

The interfaces in the application will be the Home page, Accounts, Examination, Congrats. This interface will be similar to each other and have design crossovers.

**My profile** - This will display the account user and the grade they have achieved after taking the examination meeting the necessity of keeping track of the score.

**Congratulations** - Will show how they did and the overall achieved grade, this will show them their stats when using the application. Meeting the requirement of keeping the track score and informing the user about this.

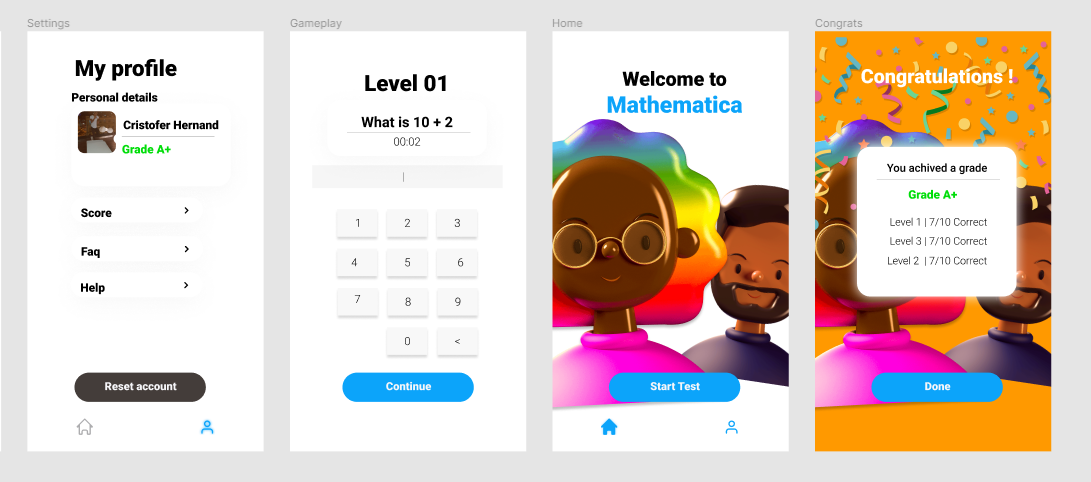
**Accounts** - will offer them to read their data and also reset their account if they wish, this means they can also keep track of their records using this part interface.

**Home** - this offers nothing but an enticing button that when clicked it starts the examination process this essentially meets the demands of making the quiz.

**Examination** - Will provide the question and the tools the user needs to get the question right. It can't be accessible, the only way to access it is through the start test button on the home page. The build of the UI is like a calculator that will show buttons laid out like one used for typing in the numbers rather than the phone standard keyboard. This keeps it clean and minimal keeping the graphics suitable. It will show the question and the timer on the top making it visible. As the timer goes to less than 5 it will turn red to alert the user.

### Actual Design

Using the data from the survey I have resigned the application to better suit the user's needs and requirements from external opinions of the application to deliver a non-bias approach.

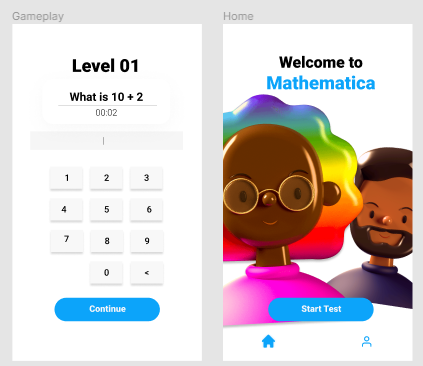


**UI Updates - Colour scheme**

Working on the first set of improvements I have worked on the user interface and designed the colour scheme to make it more appropriate. Following the studies of what colour children tend to select seen on the requirement justification part. I have used blue as the standard colour for selecting anything. This makes the button stand out and looks more pleasing to the colour scheme compared to the orange. In the profile, I have also changed the button to grey to ensure this stands out to the rest and add volume to the button; this button if clicked would reset the user score.

The background was changed to white to give it more of a cleaner outlook as well, although on the congratulations page the user interface of the orange background was kept; the reason for this is to make the message board be more visible and orange portray a sense of celebration and happiness relative to its similar colour yellow.

**UI Updates - Text enlargement**



Previous reviews of the application advocated that the question font was too small so I decided to enlarge the text to make it more visible and the buttons were also enlarged to make it more readable for the users that can't read the small text as this could disadvantage them when using the application.

# Future development

Assuming we were provided enough development time I would have added:

1. T**he grading scheme** and the settings where the user can see their score and how far they have progressed. Also, this will also include having a congratulation page after the exam is done and it will show the user their new grade and previous one.
2. **The timer** would have been amended also into the game if ensuring it skips and automatically disqualifies them from answering the question.
3. **The sound** would also be added such as in-game music such as Kahoot and correct and incorrect buttons.
4. I**n general**, also improving the functionality and user interface making it look a lot closer to the design.
5. **More interactive** features, inspired by the comparison, add more features to make the game look more alive.

Although we have provided the basic features such as making the questions simple by generating them in full and within a range meaning it can't be a decimal and it can't be more than 80 used for the questions.

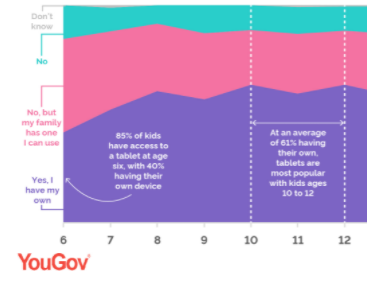
We have also ensured that the questions cannot cause any negative numbers, facilitated by the program as established before it doesn't do that and even has been tested.

# Conclusion

Overall I'm fairly satisfied that a basic output of the design has been made. Although I wish more development time was added so I can add all the full extent of the requirement and make the application as I envisioned, due to this cut, I had to cut features and ideas and focus on only making a basic application that provides its fundamental functionality.

In the test table, some features were cut meaning some tests were dismissed although every other test the program succeeded after a few configurations. Most of the default tests and additionally added tests were tested and the program managed to handle them and output perfectly the expected result. Although there were loads of bugs and issues when the program was being developed which greatly slowed the development time of the program this means that fixing these simple but yet complex bugs lead to the cut of features to get the basic output feature. The error documentation took time however Identified and delineated the cause and the resolution applied to it along with the example code of the fix. Lastly, the device testing of the program was better than expected as it ran perfectly without any prior issues on the program. No actual issues were detected, although the devices tested were limited. The program's design was only designed for tablet uses and not phones; children will mostly have tablets according to you gov "At age six, 85% of children say they have access to a tablet at home, and 40% have their own."

Yougov.co.uk. 2021. *How many children have their own tech? | YouGov*. [online] Available at: <[https://yougov.co.uk/](https://yougov.co.uk/topics/education/articles-reports/2020/03/13/what-age-do-kids-get-phones-tablet-laptops-)> [Accessed 26 May 2021].



The feedback was overwhelmingly positive compared to the improvements. Most improvements were to the UI changes and improvements to better accommodate the users. There were two classmates that provided the best form of improvements I could have used to the application. Later I then optimised the program to better accommodate them showing examples of the program differences to show the application.

Later In the development, I spoke about how the program was constructed and how ell it was done to meet its task or purpose. I also explained how the program was made more efficient but replacing lines of code made the performance run drastically faster enhancing the application. I also looked at competitors and looked at how well it compares to mine, even though mine wasn't as good it provided great ideas and concepts to add to feature expand abilities in future. I have also found the strengths of my application but mainly understood and learnt where else can I improve.

I have spoken about the requirement and how well my program meets its solution and the intended ideas I was attempting to add along with UI and design improvements compared to the old and new prototype. I also discussed why it would be appropriate and meet the requirement of the client even further which it did.

I also established the ideas in future that I would like to add and what I need to complete if further expansion of the development time was added. Overall the program was well justified and met its best. It's a requirement to the client within the current time frame it was developed.

# Time management

### Tasks

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Report | Date | Done. |
| Create the homepage as a benchmark. | This allows me to copy a consistent layout of the program. | 01/05/21 | COMPLETED. |
| Import the icons from the prototype to be used. | The icon was imported and had them named, I also imported the logo and characters. I also made and established a logo for the app. | 04/05/21 | COMPLETED |
| Create the buttons to start the test. | This allows them to be able to interact with different pages that will be later added and I also made the test page that was needed. | 10/05/21 | COMPLETED |
| Create an interactive keyboard. | The keyboard was made and values were added and implemented, I also tested it’s functionality and it worked. | 17/05/21 | COMPLETED |
| Create Functions used for the application. | I have made the main functions of the program that was needed inorder to run and operate to generate questions and check them. | 28/05/21 | COMPLETED |
| Import the work to the assignments. | I have imported all the work on my assignment to ensure I meet the deadline and get the app working in a decent state. | 29/05/21 | COMPLETED |

Cristofer Jimenez Hernandez takes responsibility for this assignment and every work I have done is produced by me. All ideas and contributions made to this assignment were done by me, besides the expandability, I took inspiration from the other competitors.

Due to the cut-down and the prior ambition of the assignment I couldn't complete it within the time frame which led to the app getting features cut although I did my best to provide it to the requirements within the time I had managed. I have behaved appropriately to provide the best solution to the client requirements.