UNIVERSITY PARTNER
UNIVERSITY OF WOLVERHAMPTON

HERALD COLLEGE
KATHMANDU

Final Year Project Proposal

Student Name	: Sandip Thapa
Student Number	: NP03A180140
Supervisor	: Manish Deuja
Submitted on	: 23rd September 2020
Words Count	: 2353
Word Count	: 2547

Contents

Table of Figures

1. Statement of Project Details

1.1. Project Title

Smart Calculator with OCR.

2. Academic Question

- How to extract numbers from the image?
- Which algorithm is most suitable?
- How to train the program with data sets?

3. Aim

- Develop an AI enabled application that can perform mathematical calculations by scanning the problem.

4. Objectives

- To make Numerals recognition as accurate as possible*.*
- To Implement Nepali Numerals in OCR.
- Extract data sets from "*kaggle".*
- Make a use of Mobile phone's camera to scan problems.

5. Artifact to be developed

- Optical Character Recognition model
- An android application built on top of Tesseract OCR.

6. Project Proposal

6.1. Introduction

6.1.1. Introduction to OCR

Optical Character Recognition is a technology that allows machine to recognize the text whether it is scanned or printed text images or handwritten text. The machine can do further processing on the data extracted from that text. It can be considered same as the combination of human eye and mind. An eye can see the text from some source but mind is the one that actually processes and interprets that text.

OCR system is made up of combination of both hardware and software. Hardware such as, optical scanner or some specialized circuit board is used to read or extract text. And software does the advance processing.

The most common use of OCR is to convert hard copy documents into softcopy files such as PDFs. This will make easier to edit the document. There are many other applications of OCR, such as: image text extraction, extracting texts from scanned documents, License plate recognition and answer paper checker. (Patel, et al., 2012)

The main concept of this project is to implement OCR in an android based calculator application. The objective of this application is to scan handwritten or printed numbers and perform mathematical calculations on it.

6.2. Background

6.2.1. Problem Domain

In this modern era of technology, people don’t want to waste time in less important tasks like performing mathematical calculation in traditional ways which are generally time consuming. For that they have calculators, but they still need manual input from the users. They need to follow some sets of guidelines and go through each processes to get correct answer. It would be much more time saving if user don’t have to manually enter the data for calculation. If the calculator gets the data with just one click, it will definitely be more efficient and time saving.

Besides, while checking the answer paper, teachers need to solve the question first by themselves and then only can correct the paper. Students are also in dilemma after solving a problem whether their answer was correct or not. It would be great help for them if they knew if their answer was correct.

6.2.2. This project as a solution

The main objective of this project is to develop an app that can perform mathematical calculations without even having to type to give input. It scans for a mathematical problem mentioned in any paper and gives back the answer. Since it is a mobile application, user can use their camera to scan the problem. The app then extracts the problem from paper using OCR and perform operations on it and return the result. Teachers and students can check whether their answers were right or not just by a click. Since this app knows all the mathematical rules and principles, users don't need to bother about remembering rules like BODMAS, which they had to keep in mind if they were to perform calculation manually in traditional calculators.

6.3. Details on Academic Question

The most challenging pat of the project is to extract the numbers from the image. The following approach can be used to overcome the challenge.

- Capture Image
- Binarize Image
- Segment numbers and expressions
- Correct perspective distortion
- Recognize number
- Prompt for user confirmation
- Solve the problem

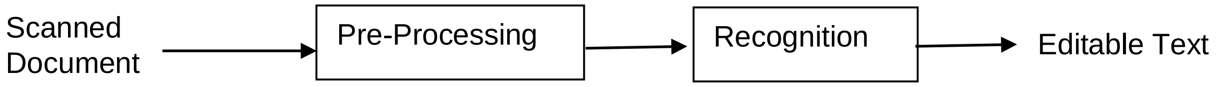
For this project Tessract OCR is the best option to choose from other character recognition systems. Following points are the main reasons to choose Tesseract OCR:

- It is platform independent
- Supports multiple languages (Google Open Source, 2020)
- High accuracy
- Open source
- Ease of access and use
- Tesseract has the font accuracy in the range of 85-90% (Sikka & Wu, 2012)

6.4. Initial Research

6.4.1. Working Principles

The objective of OCR is to extract the text and convert it into editable form. For that, a document is first scanned using an optical scanner which produces an image form of the document. Now this text image is converted into editable character code such as ASCII. The basic working principle of OCR can be show as following figure.



Tesseract OCR is used to get image input from documents. It works on step by step manner as per the diagram shown below.

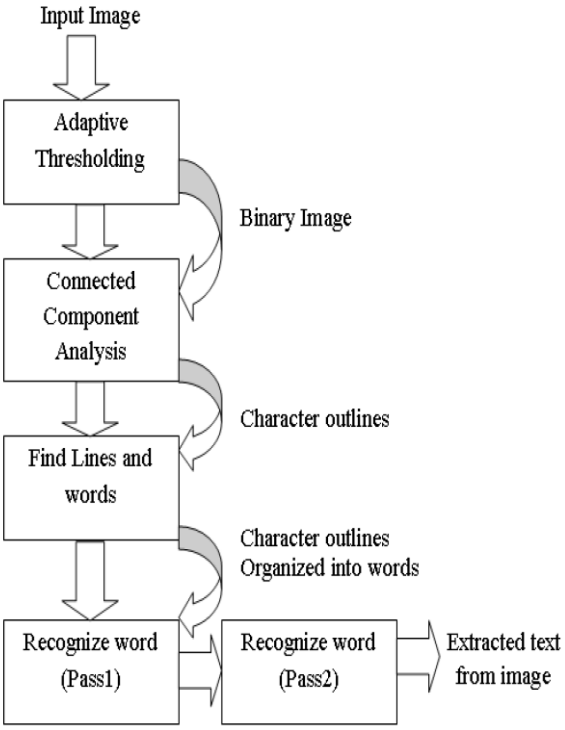


Figure 1: Architecture of Tesseract OCR (Patel & Patel, 2012)

6.4.2. Handwritten Character Recognition

Tesseract performs well for printed numbers with the detection rate more than 85% for the fonts within its database. But for the handwritten numbers it drops to about 50%. The main reason for this difference in result is due to the variation in the size of numbers that are handwritten and also due to the lack of matching fonts in its database.

To overcome this problem a machine learning algorithm based on Support Vector Machines (SVM) can be applied. This algorithm analyzes data and recognize patterns. This algorithm first converts character images into vector form. After using line segmentation region labels are used to determine the bounding box for each individual characters. A small amount of padding is added to the border, as shown in Figure 2. The segmented character is now downsampled to 32x32 pixels and then divided into 64 4x4 regions. The count in each region is the determined vector value, as shown in Figure 3. This conversion thus results in a 64 dimensional vector for each character image. (Sikka & Wu, 2012)

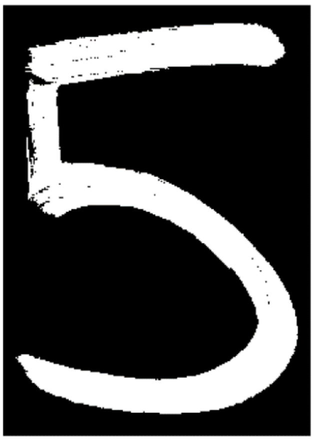


Figure 3: Segmented Character from input image

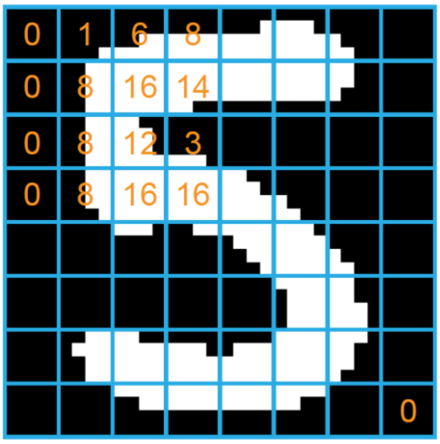


Figure 2: Downsampled Character

6.4.3. Similar Systems

6.4.3.1. Photo Math

Photomath is one of the best math problem solver application. It uses phone's camera to capture a picture of the math problems. Then the picture is scanned by the application. The application uses advanced OCR technology in order to recognize both, handwritten and printed characters. The recognized characters are then processed through Photomath's own algorithm that examines every character and determines the formula for the scanned problem. Finally, a problem solving algorithm is applied to the formula and the solution is provided with every solving steps. (Photomath, 2020)

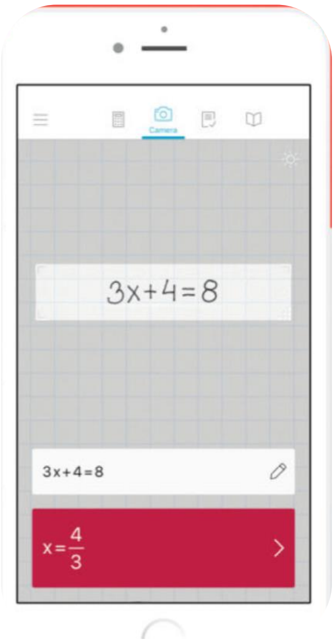


Figure 4: PhotoMath Application

6.4.3.2. Math Solver

Microsoft Math Solver can not only solve simple mathematical calculations, it can solve various math problems like quadratic equations, calculus and statistics. The application can also show graphs of the equations.

We can either type our problem query using a scientific calculator in the application or draw it on the phone's screen. But most importantly we can just use our phone's camera to scan the problem on our books or on the copy written by us. (Microsoft, 2019)

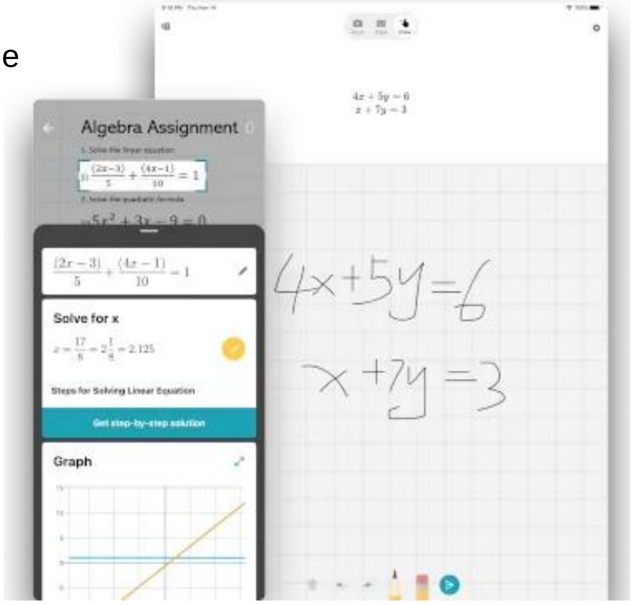

Algebra Assignment
$\frac{(2x-3)}{5}+\frac{(4x-1)}{10}=1$
$5x^2+3x-9=0$
Solve for x
$x=\frac{17}{8}=2\frac{1}{8}=2.125$
Steps for Solving Linear Equation
Get step-by-step solution
Graph
$4x+5y=6$
$x+7y=3$

Figure 5: Math Solver Application

6.4.3.3. Review of similar systems

All of the above mentioned systems have their separate math content team. Because of this there is solution to every math problem from arithmetic to calculus. These systems read and solve mathematical problems by just using the camera of mobile phones. The most astonishing feature of these systems is that they provide step-by-step solutions too. We can even choose multiple explanation methods for same problems. Moreover, they also provide animated calculation steps.

The only bad aspect or the limitation of these systems is that they support only English language. They can only perform calculations on English numerals. This is the aspect where my project is going to work on. My smart calculator will be able to perform calculations on Nepali numerals too. The system will be able to recognize the Nepali numerals on which the calculation can be carried out.

6.5. Full details of Artefact (Proposed)

6.5.1. Development Methodology

Software development methodology is a series of processes or tasks used in software development in order to improve quality of a product. It can also be known as system development life cycle. Following certain methodology allows us to document policies, procedures and processes making development process easier.

As most of the development works happens in a team, so there needs to be a good communication between developers. This is where a methodology followed comes in handy. It sets norms between a group of people working on a project about how they are going to pass information between each-others. Whether that be documentation, discussion, or diagrams on paper. (Gianpaul Rachiele, 2018)

Some of the Software Development Methodologies are:

- Agile
- Waterfall
- Incremental
- Extreme Programming
- Rapid Application Development

Among these, Incremental model is best suited for my project and is defined below in section 6.5.2.

6.5.2. Preferred Methodology

For this software development Increment methodology is best suited.

Incremental Model is one of the methodologies of software development where requirements are divided into multiple standalone modules. Each of this modules go through phases involved in this methodology.

In this model a simple working system with basic features is built and delivered to client at first. Then the other features are added in many successive iterations. Due to this reason, this model is also known as Successive Version Model. (GeeksforGeeks, 2020)

In the below figure, Software Product is incrementally developed and delivered.

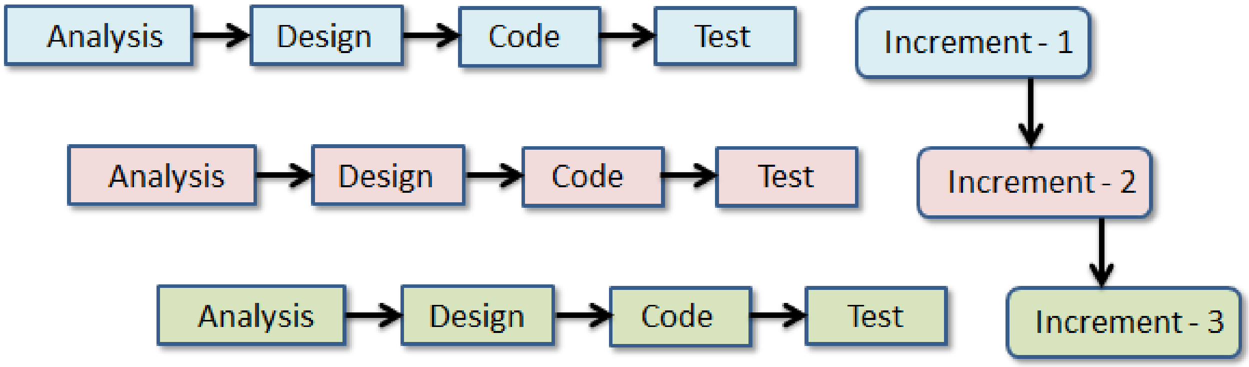


Figure 6: Incremental Model Visualization (EduCba, 2020)

Reasons to choose Incremental:

- Generates working software quickly and early during the software life cycle.
- Works on iteration, so it would be easy to develop promotional website at first and then mobile application.
- Parallel development can be done
- Supports change of requirements
- Testing and debugging is easier because of small iterations
- More flexible
- Easy to manage risks

6.5.3. Techniques and Tools

Software Development techniques includes preparing a plan of upcoming work, estimating task to perform, allocating resources and monitoring quality and deadlines. There are various techniques available for various development methodologies. Some of them are:

- Strategic Planning Technique
- Structured Programming Technique
- Object Oriented Analysis and Design Technique
- Software Testing Technique (Spring Digital, 2019)

From all the above mentioned techniques, Object Oriented Analysis and Design techniques is most suited for this project as the core programming language to be used in this project is *Python*, which is Object Oriented programming language.

When applying any techniques to the project we need to use specific tools for successfully implementing techniques. We needs tools to plan, design and develop the system. Some of the tools that come handy during software development are as follows:

- Modeling tools for designing
    - *Star UML, Adobe XD, Corel Draw*
- Word processor for report
    - *Microsoft Office*, Microsoft Excel
- Core Programming Language
    - *Python with Tesseract and OpenCV modules*
- Code Editor
    - *Visual Studio Code, Jupyter Notebook*
- Version Controlling
    - *Github*

6.5.4. Testing

As we are doing this project on Incremental fashion, each increment will give us simple working system with basic features. So, we can perform testing after every increment. And for the testing we will only conduct *Black Box Testing.*

6.6. Probable Issues During the project

- Hardware failure

One of the major issue that may occur during the project is hardware failure. No matter how careful we are, hardware like hard drive may crash at any time. Likewise problem in different peripherals like mouse and keyboard may occur.

To mitigate this problem project will be stored in cloud storage as well as eternal hardware. Extra pair of peripherals will also be available to use during the project.

- Version Controlling

As I am a beginner programmer, I have no experience to manage the flow of programming. While adding new features I might mess up the previous code and project may fail.

To overcome this issue I will use version controlling tools like “Github”. This way I can fall back to the previous running code.

- Lack of Dataset to train the system

Artificial Intelligence works on the basis of data provided to it. Similarly OCR will only recognize the data that is already fed to it. So we need huge amount of dataset to teach the system which may not be available free to use. And as a student I might be able to purchase those datasets.

To mitigate this problem we can ask for the data set that is already used by the senior students who had done similar type of projects.

- Incompatible SDLC methodology

Due to the insufficient experience in using different software development methodologies, the methodology chosen might not be compatible.

To mitigate this problem supervisor plays vital role. His guidance will help to choose the best methodology before starting the project. Moreover internship in similar project related companies will also help to build some experiences.

6.7. Plan/Schedule

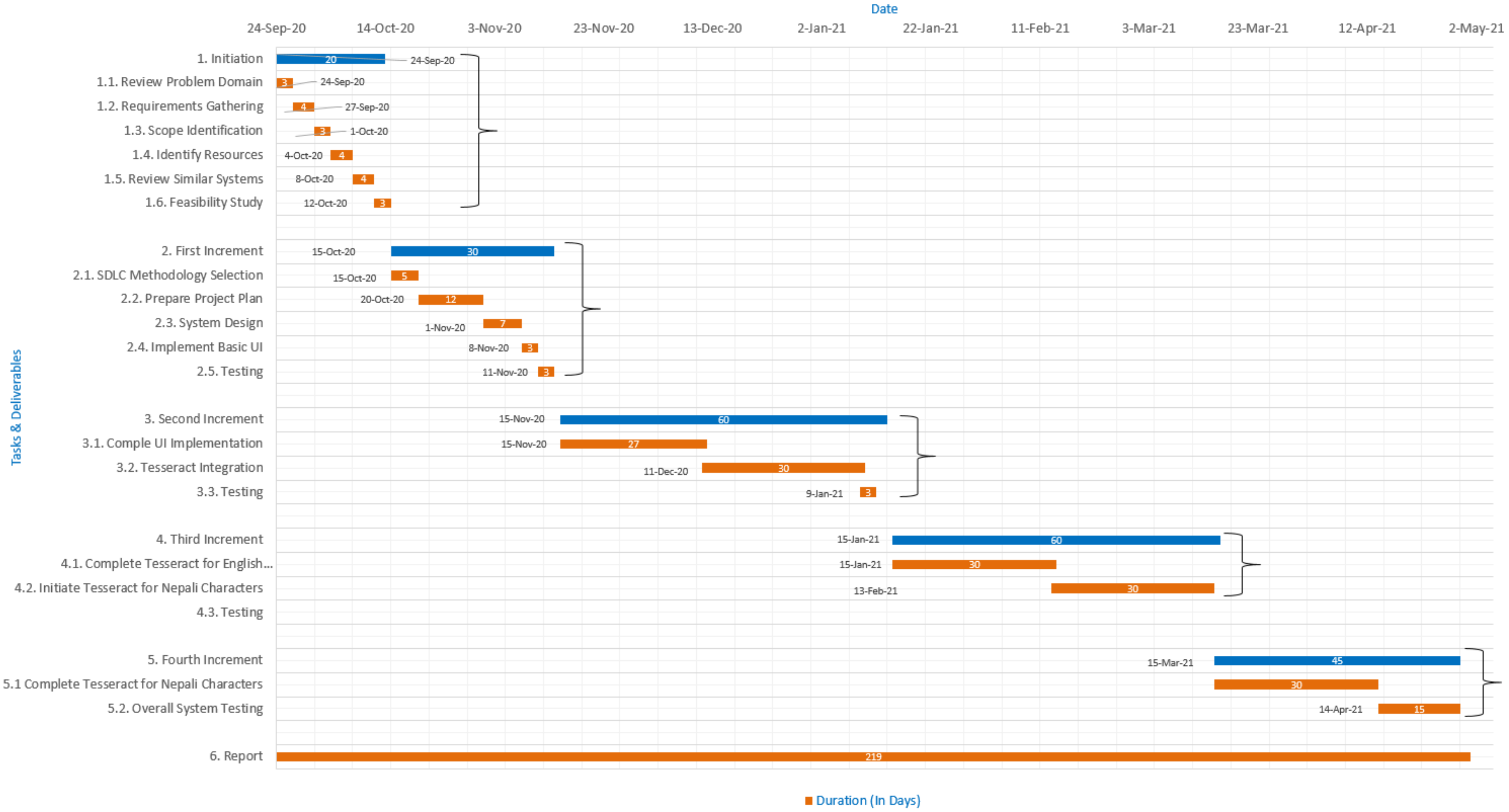


Figure 7: Gantt Chart

7. Conclusion and Recommendation

7.1. Conclusion

In conclusion, **Smart Calculator** is a mobile application that uses emerging Artificial Intelligence technology known as “OCR”. This application can do mathematical calculations just by scanning the handwritten or printed problems on paper. It can also be used as normal and traditional calculator as well. The main purpose of building this application is to make calculations a little faster and easier to teachers and students. There are already many similar applications but they are confined within English language. So this application tries to break this barrier a little bit and allows users to perform calculations even in Nepali language or numerals.

7.2. Recommendation and further works

No any system in the world is perfect. Neither is this one. This is a simple system that can do simple calculations. There are lots of rooms to improve. This system can do basic calculations like addition, subtraction, multiplication and division by scanning the problem. We can improve it to solve the algebraic equations, quadratic equations, calculus as well as geometric problems.

We can also work on the area to increase the accuracy of character recognition. As mentioned above most of the OCR systems are based on some of the widely spoken languages. One of the future task can be to conduct research on languages other than widely spoken and implement in this application.

References

EduCba, 2020. *Incremental Model.* [Online]
Available at: https://www.educba.com/incremental-model/
[Accessed 20 September 2020].
GeeksforGeeks, 2020. *GeeksfoGeeks.* [Online]
Available at: https://www.geeksforgeeks.org/software-engineering-incremental-process-model/
[Accessed 4 May 2020].
Gianpaul Rachiele, 2018. *Medium.* [Online]
Available at: https://medium.com/@gianpaul.r/software-development-methodologies-a856883a7630
[Accessed 5 May 2020].
Google Open Source, 2020. *Tesseract OCR.* [Online]
Available at: https://opensource.google/projects/tesseract
[Accessed 15 May 2022].
Microsoft, 2019. *Microsoft Math Solver.* [Online]
Available at: https://math.microsoft.com/en
[Accessed 20 April 2020].
Patel, C. I., Patel, A. & Patel, D., 2012. Optical Character Recognition by Open source OCR Tool Tesseract. *International Journal of Computer Applications,* 55(10), pp. 50-56.
Patel, C. I. & Patel, D., 2012. Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study. *International Journal of Computer Applications,* 55(10), pp. 50-56.
Photomath, 2020. *Photomath.* [Online]
Available at: https://photomath.net/en/help/how-does-photomath-work
[Accessed 20 April 2020].
Sikka, A. & Wu, B., 2012. *Camera Based Equation Solver for Android Devices,* California: Stanford University.
Spring Digital, 2019. *Spring Digital.* [Online]
Available at: https://www.springdigital.com.au/software-apps/tools-and-techniques-for-software-development/
[Accessed 16 May 2020].