****

**The Arab American University**

Faculty Of Engineering and Information Technology

Computer Systems Engineering Department

**Senior Project I**

**Camera Motion Detection and Alarm System**

***Prepared By:***

|  |
| --- |
| **Amal Irshaid- CSE- 21511169** |
| **Fayyad Salahat - CSE – 21510887**  **Shatha Abu AlRob – CS - 21511773** |

***Supervisor***

**Prof. Dr. Adwan Yasin**

**Submitted in Partial Fulfilment of the Requirement for the Degree of Bachelor of Computer Systems Engineering**

**January 2020 © Arab American University – Palestine**

**Jenin-2019.**

**All Rights Reserved**

**We hereby approve the report entitled:**

**Project Title: (Camera Motion Detection and Alarm System)**

**Prepared by:**

**Amal Irshaid- CSE- 21511169**

**Fayyad Salahat - CSE - 21510887**

**Shatha Abu Alrob - CS - 21511773**

**Advisors: Prof. Dr. Adwan Yasin**

**Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Abstract**

Our project is to create a security system that helps people to feel safer and more comfortable leaving their houses and assets instead of feeling insecure while leaving their houses with their kids inside or any other reasons.

This system is designed to detect motion in the zone where is placed in to monitor. The camera will monitor all time when a motion occurs, the camera start to detect the motion if it human motion or not. If the motion is human motion the camera capture the motion and save it in firebase database then send a notification with a photo to the mobile application.

That’s how our system will be providing the needed monitoring and security requirements to the user in the easiest way with the least cost possible. Attacking people’s houses anonymously and steal people’s own properties will be detected and reported to the user. Houses and personal assets are safer with our system.

**Table of Contents**

List of Tables …………………………………………………………….……………………………. 6

List of Figures ………………………………………..…….…………………………...……………... 7

Chapter One Introduction .…………………………………..………………………………………..... 8

1.1    Background and Motivation …………………………………………………….……………....8

1.2    Aims and Objectives …………………...…………………………………………………........ 9

1.3    Problem Statement …………………………..……………………..................………………...9

1.4    Contributions ..……………………………………..…………………………..……………... 10

1.5 Structure of the Report ………..……………………………………………...……………….10

Chapter Two. Literature Review .……………………….....………………………………..…….…...11

2.1    Overview …………………………………..…………………...……………………………...11

2.2    Existing Systems ……..………………………………..………………………..……………..11

2.2.1 Home surveillance and motion detection .……...............................................................11

2.2.2 Room Guard………………………………………………………………………..……12

2.3    Summary ……………………………………………………………………….……..………....13

Chapter Three. System Design and Solution..……………………………………………………........14

3.1 Introduction…………………………………..…………………………………………………….14

3.2 Hardware………..……………………..…….……………………………………………………..14

3.2.1 Hardware Components…………………….……………………………………………..14

3.2.2 Hardware Description……………………….………………………………………...…15

3.3 System Architecture and Algorithms ………..………………….………………………………...17

3.3.1 Main Algorithm ………………………….…………………..…………………….…….18

3.3.1.1 Send Notification Algorithm ………………..…………………………………19

3.4 Use Case Diagram………………..…………………………………………………………….…..19

3.5 Class Diagram…..…….…………..………………………………………………………………..20

3.6 Project description............................................................................................................................21

3.6.1 Hardware description.....................................................................................................................21

3.6.2 Algorithm description....................................................................................................................22

3.6.3 Software description.......................................................................................................................23

Chapter Four. Conclusion and Future Work..……...…………………………………….………….....29

4.1 Conclusion………….……..………………………………..……………………………………....29

4.2 Future Work……………………………………………..………………………………………....29

Acknowledgments………………………………………..…………………….……………………....30

References……………………………………………..………………………….……………………31

**List of Tables**

 Table 1-1 List of main indicators of victim survey……………………………………………………8

 Table 2-2 Comparisons between our system & other systems……….………………………………13

**List of Figures**

Figure 2.1 Pi cam motion detection.......................................................................................................11

Figure 2.2 Room Guard System………………………………………………………………………12

Figure 3.1 Raspberry pi 3 module B+………………………………….………………………….….15

Figure 3.2 Pi camera. …………………………………………………...…………………………….16

Figure 3.3 Architectural Diagram……………………………………..………………………………17

Figure 3.4 Main System Algorithm Design…………………………..………………………….……18

Figure 3.5 Send Notification Algorithm……………………………..………………………………..19

Figure 3.6 Use Case Diagram………………………………………..………………………………..20

Figure 3.7 Class Diagram…………………………………………..………………………………....20

Figure 3.8 a. Result Number 1………..................................................................................................21

Figure 3.8 b. Result Number 2………………………………………………………………………..22

Figure 3.9 Add firebase to your Android app.......................................................................................24

Figure 3.10 Firebase setting..................................................................................................................24

Figure 3.11 Connect Android studio with Firebase..............................................................................25

Figure 3.12 Firebase authentication......................................................................................................25

Figure 3.13 User Sign-Up Interface…………………………………………………………………..26

Figure 3.14 User Sign-Up Information is added to Firebase…………………………………………26

Figure 3.15 Sign In interface………………………………………………………………………….27

Figure 3.16 Notification Display Interface………………………………..…………………………..27

Figure 3.17 Display Photo interface………………………………………………………………,,,...28

**Chapter One**

**Introduction**

# **1.1 Background and Motivation**

We were looking for problems that people generally suffer from, which needed solutions. In 2016, based on a survey that was made in Palestine by the Palestinian Central Bureau of Statistics, as shown in Table 1.1 that lists the main indicators of victimization survey for the past years from 1996-2016, the robbery crimes were very high in 1996 until 2004, the percentage kept decreasing every year. Then suddenly, it started increasing at a very high rate. 62% of the people living in Palestine were attacked and exposed to theft\robbery attempts. 15% of the Palestinian people were threatened with robbery crimes, 9% of them were physically harmed. Palestinian people’s houses and personal properties, assets or holdings are in danger, and sometimes, in the worst cases, they are in danger too! So we decided as a team to create a system that provides the appropriate help for these people and to provide suitable solutions for them so they can easily watch their own properties, assets or holdings from any place, at any time, within any condition using our “Camera Motion Detection and Alarm System” so that they can make the necessary countermeasures before it's too late.

**Table 1.1   Main Indicators of Victimization Survey 1996-2016:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Indicator** | **Years** | | | | | |
| **1996** | **1999** | **2004** | **2008** | **2012** | **2016** |
| **Victims at the Individual Level by Last Criminal Offense\*** |  |  |  |  |  |  |
| **Percentage of persons exposed to theft\robbery attempt** | 54.2 | 55.2 | 19.5 | 33.9 | 41.9 | 61.8 |
| **Percentage of persons exposed to threat\ assault** | 18.8 | 18.0 | 13.1 | 18.4 | 10.6 | 14.9 |
| **Percentage of persons exposed to property damage** | 16.1 | 4.4 | 8.2 | 18.3 | 3.0 | 0.4 |
| **Physical Harm and Tangible Losses of Last Crime\*** | | | | | | |
| **Percentage of persons exposed to criminal offense and caused physical harm** | 22.6 | 16.5 | 10.5 | 12.3 | 5.5 | 8.6 |

[1]

# **1.2 Aims and Objectives**

Our system aims to achieve several goals to provide services that help people protect their personal properties, assets, or houses by using our “Camera Motion Detection and Alarm System” to monitor their properties using cameras that are connected to our system to detect any undesirable human motion and inspect any motion near their own properties or assets by simply using the mobile application that is connected to our system. It can help the user to receive and easily monitor anything they desire or demand.

This system will provide several goals:

1. Delve deeper into the principles of data and information security.
2. Learn how to develop motion detection algorithm (OpenCV) inside our security system using Raspberry pi and pi cam to detect human motion.
3. Increase our knowledge while using Ubuntu 18.04.4 LTS (Linux distribution).
4. Obtain more experience about how to build a mobile application using Java programming language by Android Studio.
5. Examine (trying to use) firebase to use it as a replacement for SQL database.

# **1.3 Problem Statement**

A lot of people prefer easy money which leads them to steal other people’s houses, money or assets. Sometimes things get too dangerous! Such as hurting others for some personal psychological illness, revenge or any other personal reasons. People with bad intentions are ready to do whatever it takes to hurt other people to satisfy their own needs. Sometimes, they do it for no reason but for the enjoyment of hurting others, these people generally have mental issues.

We need to feel safe and make ourselves safe by keeping an eye on our houses, streets, gardens, containers, workplaces, schools, governmental authorities, etc. To solve this problem we will design the “Camera Motion Detection and Alarm System” so that when any undesired human motion occurs, the camera must capture this motion as a picture, and send it to the user. Finally, the user can view the aimed protected zone in real-time.

To achieve this goal, we will use PIR (Passive Infrared Motion Sensor), which is a sensor used for motion detector, it will detect any motion occurrence. Then the pi camera will detect the motion in real-time, it will make object classification, object locating, object detection to determine if the motion is human motion or not. If the motion that was detected is a human motion, the system will send a notification containing a photo for the motion that was detected to the user mobile application.

When the notification received to the user, the user can view the motion in real-time.

The user can use more than one camera to monitoring what he/she want to monitor , the user can access to multiple camera by her/his mobile application.

There are a lot of security systems that are similar to our system. But all other systems detect any kind of motion whether it was a cat or a dog, and will notify the user with a lot of fake alarms. This is non-professional work that wastes a lot of system resources such as network bandwidth and local memory, and it will waste the user’s time too to look after wrong alarms. But in our system, this will never happen. When a motion happen the camera detect the motion if it is human or not in real-time.

# **1.4 Contributions**

This is a security system that helps people to make them feel safer, by making it easier for them to monitor their own properties to detect any undesirable human motion, then announce them by a notification that contains photos for the detected motion, to make it a well-designed system.

We are so proud to decrease the level of assault on personal properties and to raise the feeling of safety from external threats that could happen at any time for any unknown and unexpected reasons. Either this attack was a robbery for stealing money or if it was for any psychotic or personal intentions. Our system will keep people safe by monitoring their houses and will take care of their assets to not be stolen by any thief.

**1.5**           **Structure of the Report**

The organization of the document for the system as follows: In chapter 2, we will present literature about systems that are like our system and close to our system. In chapter 3, we will talk about system analysis like system requirements, functional requirements, non-functional requirements, system architecture, describe the hardware components needed for the system. In chapter 4, we will design a class diagram, describe algorithms for the system and interface design.

**Chapter Two**

**Literature Review**

# **2.1 Overview**

In this chapter, we will talk about the existing systems that are partially similar to our system and the hardware and software used in our designed system. However, there are many systems that achieve the same objectives of our system in different regions in our country or other outside countries to help people to save and monitor their own assets from any unwanted anonymous attacks.

# **2.2 Existing Systems**

**2.2.1   Home surveillance and motion detection**

This project is related to our system since it has the same functionality as our project.

This project is designed for monitoring the **refrigerator.**

**The hardware used in the project are:**

1. **Raspberry pi**
2. Pi camera

To access Raspberry pi video stream, should installed OpenCV on Raspberry pi.

OpenCV is a library of programming functions mainly aimed at real-time computer vision, in simple language it is library used for Image Processing. It is mainly used to do all the operation related to Images.

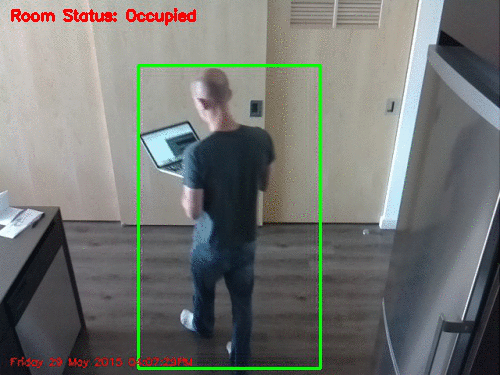
[2]

Fig 2.1 Pi cam motion detection

**The differences between the above system and our project:**

The above system is designed for monitoring the **refrigerator, our system designed for home monitoring and send alarm to user phone with home photo. The photo in our system is upload to firebase then send to the mobile application.**

# **2.2.2** **Room Guard**

This system detect motion using PIR sensor, camera model and buzzer with Raspberry pi, when pir sensor detect motion the camera start to captor photo for the motion and the buzzer start get alarming, then send notification for user phone.

The hardware used:

1. Raspberry pi
2. Camera model
3. PIR sensor
4. Buzzer

The programming language used is python.

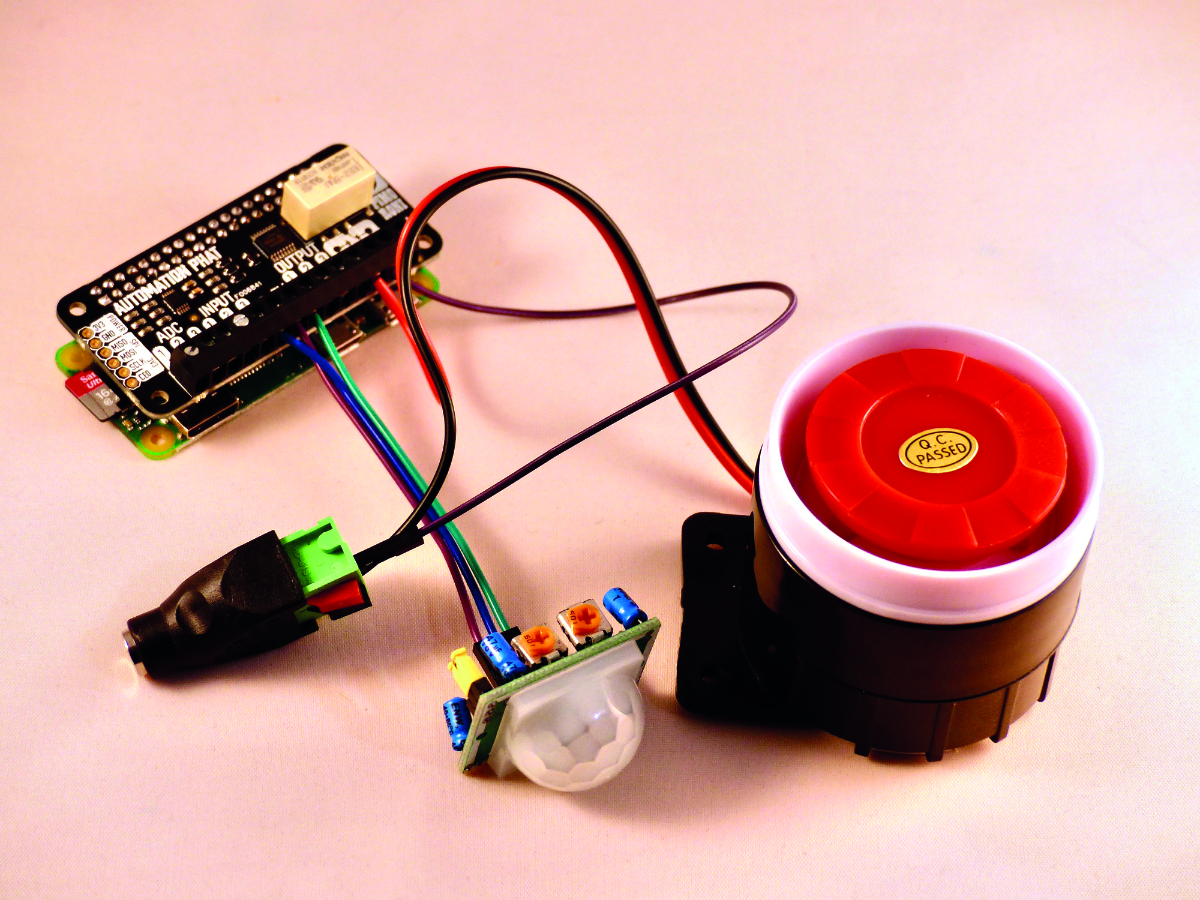
[3]

Fig 2.2 Room Guard system

**The differences between the above system and our project:**

The above system is designed when the PIR sensor detect motion to get alarm by buzzer and send notification to the user phone. Our system designed to detect human motion only then capture photo for the motion and send it to the user phone.

**2-1 Comparisons between our system & other systems:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Home Surveillance and Motion Detection** | **Room Guard** | **Motion Detection System with Alarm System** |
| **Use Sensor** | **No** | **Yes** | **No** |
| **Connected to an application** | **No** | **Yes** | **Yes** |
| **Programming Language** | **Python** | **Python** | **Python** |
| **Accuracy** | **Medium** | **Strong** | **Medium** |
| **Alarm** | **None** | **Buzzer + Send Notification on the Mobile App** | **Send Notification on the Mobile App** |
| **System Component** | **Raspberry pi**  **Pi Camera** | **Raspberry pi**  **Camera model**  **PIR sensor**  **Buzzer**  **Android App** | **Raspberry pi**  **Camera model**  **Android App** |

**2.3 Summary**

In this chapter, we covered one existing system that shares similar characteristics and functions as our own system. It was applied in other countries and it was sold on the famous online shopping website “Amazon.com”. Both systems have their own way of controlling the system using special hardware/software.

In our system, we will be using the Raspberry Pi to control hardware in our system. We will use a camera with a motion detector that will track motion, then if any human motion is detected, then the camera will capture the motion of that object. Then finally it will be sent to the application that’s installed previously on the user mobile. Differently, in our project we used the firebase to upload home security photos. Which makes it faster and more secure for the images to be sent or to be kept/saved.

**Chapter Three**

**System Design and Solution**

# **3.1 Introduction**

This world is so much dangerous and our own assets aren’t safe anymore. So we decided to solve one of the world’s problems and defeat people’s fears and concerns. To help them feel more secure, so they can focus on other daily life business. So we designed this system to help them achieve high safety levels.

In brief, our system will provide you a full 24 hours supervision for your own precious assets either it was a place (home, school, etc.) or personal objects such as (containers, cars, safe, etc.).

All that you must do is to compose our camera in front of the object or the zone that you want to monitor and install our simple usable application. The camera will be automatically verified on that application. You can add several cameras watching different places/zones through your application.

# **3.2 Hardware**

Here we describe the hardware components needed in our project to build our system. We will connect them easily any the Raspberry pi will be programmed using [raspberry pi (UNC viewer)] and will be programmed in python.

## **Hardware Components**

1. Raspberry pi 3 module B+
2. The pi camera module 2

**3.2.2 Hardware Description**

1. Raspberry pi 3 module B+



Fig3.1 Raspberry pi 3 module B+ [4]

 For the microcontroller (Raspberry pi) based security system using a motion detector to be used for effective monitoring and warning, the system must provide these functions: detecting human motion, alerting mechanism, and see the current action in real-time.  

[5]The Raspberry pi specification are:

The Raspberry Pi 3 Model B+ is the final revision in the Raspberry Pi 3 range.

* Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
* 1GB LPDDR2 SDRAM
* 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
* Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
* Extended 40-pin GPIO header
* Full-size HDMI
* 4 USB 2.0 ports
* Raspberry Pi standard 40 pin GPIO header (fully backward compatible with previous boards).
* 2 × micro-HDMI ports (up to 4kp60 supported).
* 2-lane MIPI DSI display port.
* 2-lane MIPI CSI camera port.
* 4-pole stereo audio and composite video port.
* H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode).
* OpenGL ES 3.0 graphics.
* Micro-SD card slot for loading operating system and data storage
* 5V DC via USB-C connector (minimum 3A\*).
* 5V DC via GPIO header (minimum 3A\*).
* Power over Ethernet (PoE) enabled (requires separate PoE HAT).
* Operating temperature: 0 – 50 degrees C ambient.

B-The pi camera module 2



Fig 3.2 Pi camera [5]

We use the camera to capture images and video, it is the only camera made specifically for the Raspberry Pi device. This camera can manually switch between day and night modes. It has n IR LED (infrared light-emitting diode) is a solid-state lighting (SSL) device that emits light in the infrared range of the electromagnetic radiation spectrum. IR LEDs allow for cheap, efficient production of infrared light, which is electromagnetic radiation in the 700 nm to 1mm range.

The Features:

* Fully Compatible with Both the Model A and Model B Raspberry Pi
* 5MP Omni vision 5647 Camera Module
* Raspberry Pi Camera, supports all revisions of the Pi
* Using IR-CUT filter, it can eliminate color distortion in the daylight
* Comes with infrared LED, supports night vision
* Can Attach IR LEDs if Night Vision mode is required
* 15-pin MIPI Camera Serial Interface - Plugs Directly into the Raspberry Pi Board
* Still picture resolution: 2592 x 1944
* Max video resolution: 1080p
* Camera Size: 20 x 25 x 9mm
* Camera Weight 3g
* Fully Compatible with many Raspberry Pi cases

**3.3 System Architecture and Algorithms**

The main goal of our system is to give people safety and security over their homes and properties from thieves and attacks. By using mobile application the user can surveillance his/her home and their own properties.

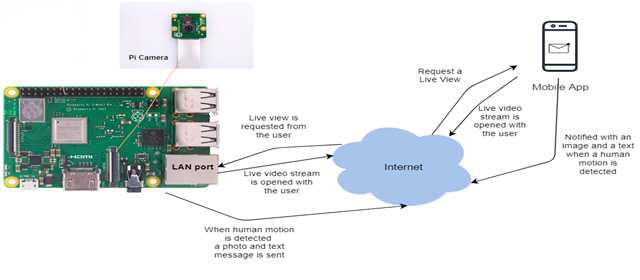
****

Figure 3.3Architectural Diagram

**3.3.1 Main Algorithm**

# We developed an algorithm represented as a “flowchart” explaining how the system should work, represented as several steps defining our main system functions. First of all, we have the main algorithm which will describe the whole system. Then, each main function in the main algorithm will be described as sub algorithms in details as shown in fig 3.4.

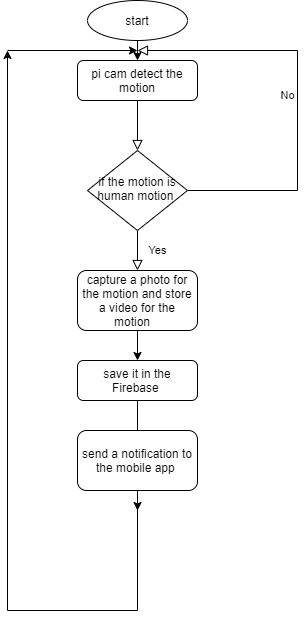


Fig 3.4: Main System Algorithm Design

**3.3.1.1 Send Notification Algorithm**

Here will discussing how the notification send to the user with the photo of the human motion.

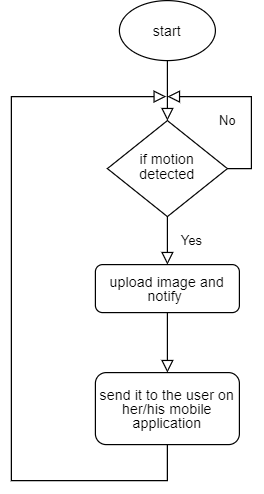


Fig 3.5 Send notification algorithm

**3.4 Use Case Diagram**

Here are the use cases for each part which using the system, as shown in fig 3.6.

For the user, any user can register to the application, login to the application and logout from application. The user will receive notification when human motion happened containing with photo to the human motion. The camera will monitoring the environment when motion occur it detect if it human or not.

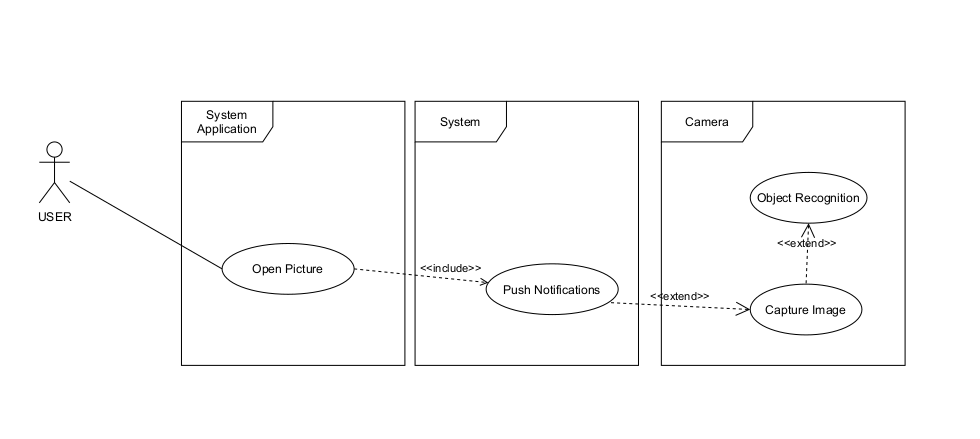


Fig 3.6 Use Case Diagram

# **3.5 Class Diagram**

Here are the classes for each part of the system, as shown in fig 3.9. Here we have the objects in our system we use the inheritance to make the code dynamic, so it can be easily changeable. We have 1 kind of application user, which are the client. The client is person. We have 2 hardware application, pi camera and Raspberry pi.

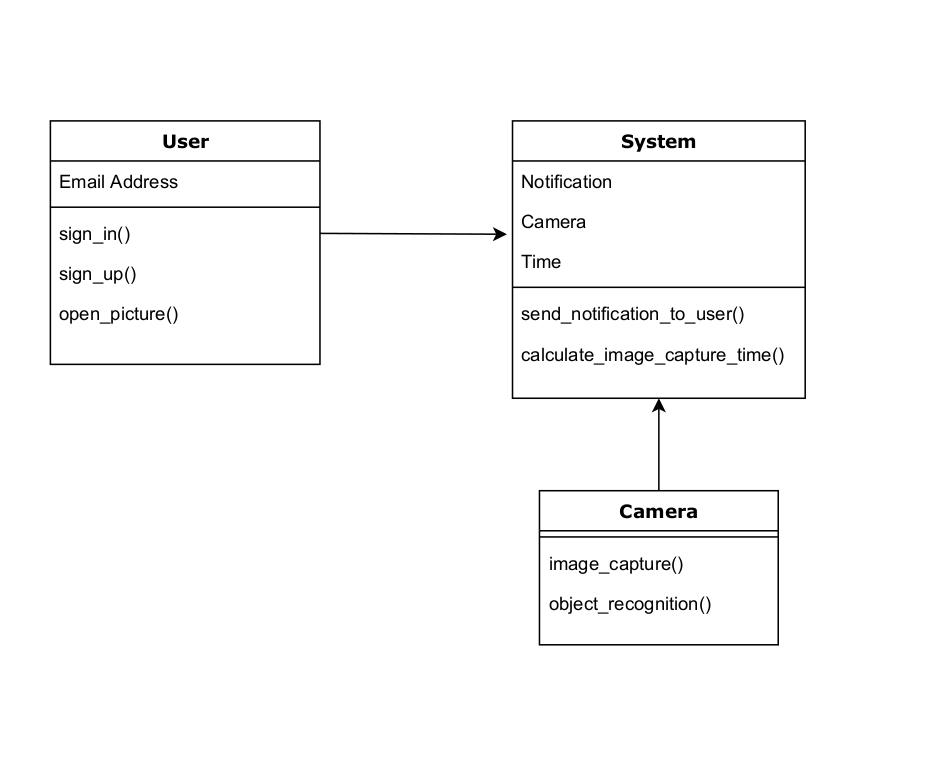


Fig 3.7 Class Diagram

# **3.6 Project description**

# **3.6.1 Hardware description**

The hardware components that are used were previously listed in this document. After collecting all components together and connecting them with each other, the next step was to choose the best suitable OS to be installed. We chose “Raspbian” as our OS. Why Raspbian? Well, first of all, raspberry pi 3 module B+ is a 32-bit system with needs a light OS that works with 32-bit. Next, developers describe Raspbian as "A free operating system based on Debian". It is optimized for the Raspberry Pi hardware. It provides more than a pure OS: it comes with more than 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi. It is free, completely customizable, more secure, runs without installing, better suited for development, it can be updated without restarting, and finally, of course, it is an open-source system.

The following step was to install the needed packages to develop the motion detection system using the “Python Programming Language”. Such as <DateTime> which determines the date that this video or picture was taken within it, and finally, CV2 which is used to import our main algorithm methods and objects. This works with both pre-recorded videos and live streams from your webcam. This system will capture any detected motion and upload it to your personal Dropbox. Here’s a GIF picture that describes how the algorithm really works.



Figure 3.8 a Result Number 1

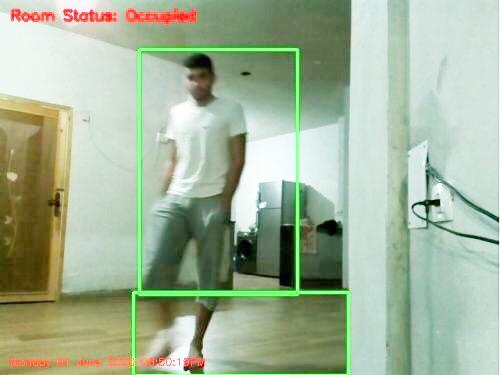


Figure 3.8 b Result Number 2

There are 2 types of algorithms for this purpose which both are based on two contradictory reasons:

1. Based on a static background. Such as in the above GIF picture.
2. Based on a dynamic background. Such as wearing the camera while moving around, the background changes every millisecond, which makes it harder to detect objects in a fast mode and harder to distinguish them from other unwanted objects in the moving background too.

But our system uses the first type. Because obviously we have a static background with only moving objects in the foreground. So we installed the OpenCV’s version that’s more suitable to our system’s objective. Obviously, in the real-world many virtual assumptions can easily fail. Due to lighting conditions, reflections, shadowing, and any other possible change in the environment. And if the background appears to be different, it can throw our algorithms off.

**3.6.2 Algorithm Description**

There are several steps that are done in our algorithm. These steps will be listed and briefly explained. These steps are:

1. Preparing:
   1. Install a python compiler on Ubuntu to be able to run the python code written using any text application.
   2. Install all needed packages to run this algorithm. Basically, OpenCV.
   3. Reading the video file either as .mp4 file or as a live video stream on webcam using the web browser.
   4. Cutting the video into several frames to be image processed separately. The first frame will contain no motion so that’s how the background will be noticed and distinguished.
   5. Loop over the video frames. If frame can’t be found then it’s the end of the video.
2. Ready to perform motion detection:
   1. Compute the absolute difference between the basic frame and the current frame.
   2. Dilate the threshold image to fill in holes, then find contours on threshold image. Then loop over these contours.
   3. If the contour is too small, ignore it. Compute the bounding box (Green Box) for the contour, draw it on the frame, and update the text (Occupied or Not Occupied).

**3.6.3 Software description**

We used the “Android Studio” to develop our mobile application for motion detection and we decided to use the firebase as a server to store the captured photo for several reasons:

1. Unleashing the power of Google Analytics
2. Crash reporting to fix Bugs
3. Monitoring errors: It is capable of monitoring fatal errors for iOS apps and both fatal and non-fatal errors for Android apps. Generally, reports are initiated as per the impact caused by such errors on the user experience.
4. Required data collection to fix errors: The reports also enlist all the details concerning the device in use, performance shortfalls, and user scenarios concerning the erroneous events. According to the contributing factors and other similarities, the issues are grouped in different categories.
5. Email alerts: It also allows sending email alerts as and when such issues or problems are detected.
6. The configuration of error reporting: The error reporting can also be configured remotely to control who can access the reports and list of events that occurred before an event.
7. It is free: Crash and bug reporting is free with Firebase. You don’t need to pay a penny to access this feature.
8. Synchronizing data with real-time database
9. Real-time: Unlike the so-called HTTP requests that work to update the data across interfaces, the Real-time Database of firebase syncs data with every change thus helping to reflect the change in real-time across any device in use.
10. Offline: As Firebase Real-time Database SDK helps save your data in local disk, you can always access the data offline. As and when connectivity is back, the changes are synced with the present state of the server.
11. Access from multiple devices: The Firebase Real-time Database allows accessing application data from multiple devices and interfaces including mobile devices and web.
12. Splitting and scaling your data: Thanks to Firebase Real-time Database, you can split your data across multiple databases within the same project and set rules for each database instance.

When the “cam” camera takes a photo for objects in motion, it sends it immediately to the firebase to save it. And when a new photo is added to the firebase, a notification with the new photo will be sent to the mobile application, so that the user can check who came near his own home or asset. The firebase is a mobile and web application development platform developed by the “Firebase” company.

By using firebase the dealing with the API and transactions will be much easier. Plus, connecting the android studio with the firebase is done with few steps only, no need to write any code. And for API to make a connection between the app and the server (firebase server), we need to write a few lines of code.

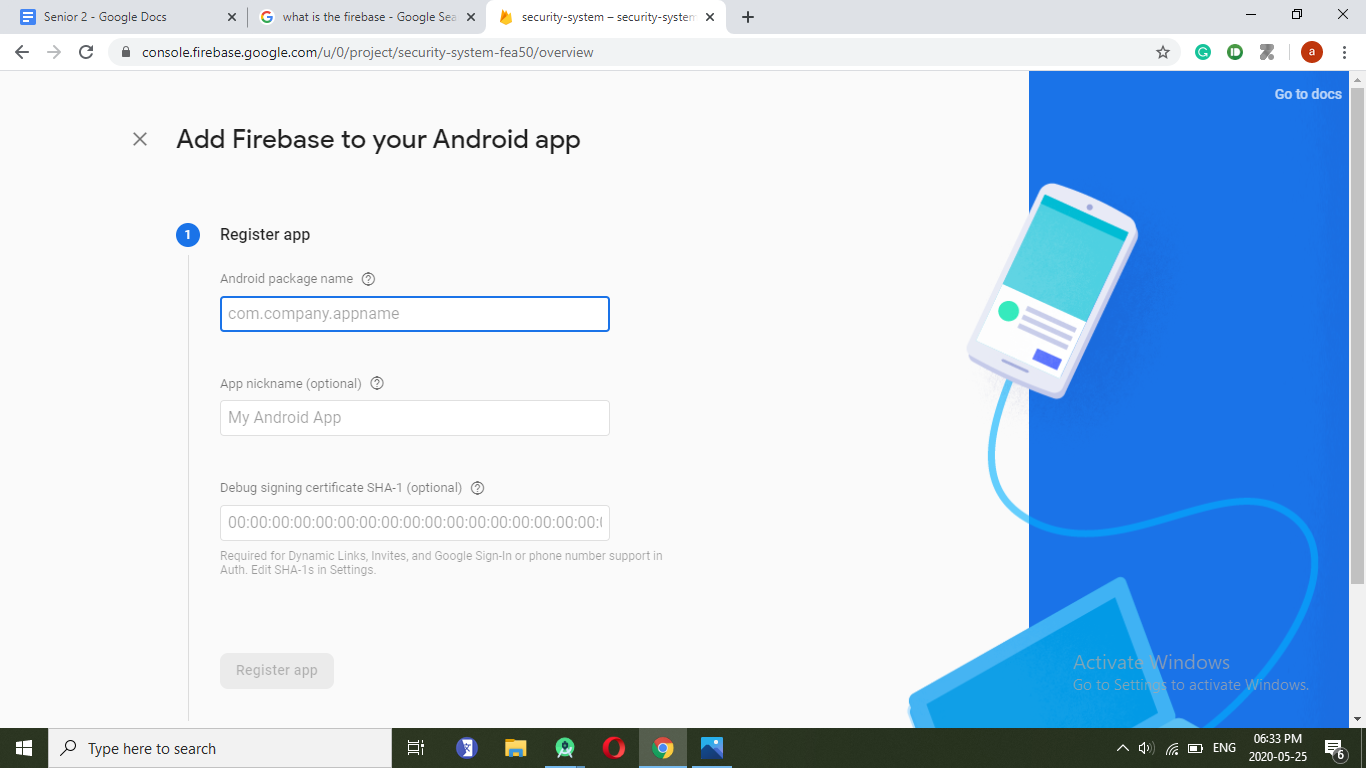


Fig 3.9 Add firebase to your Android app

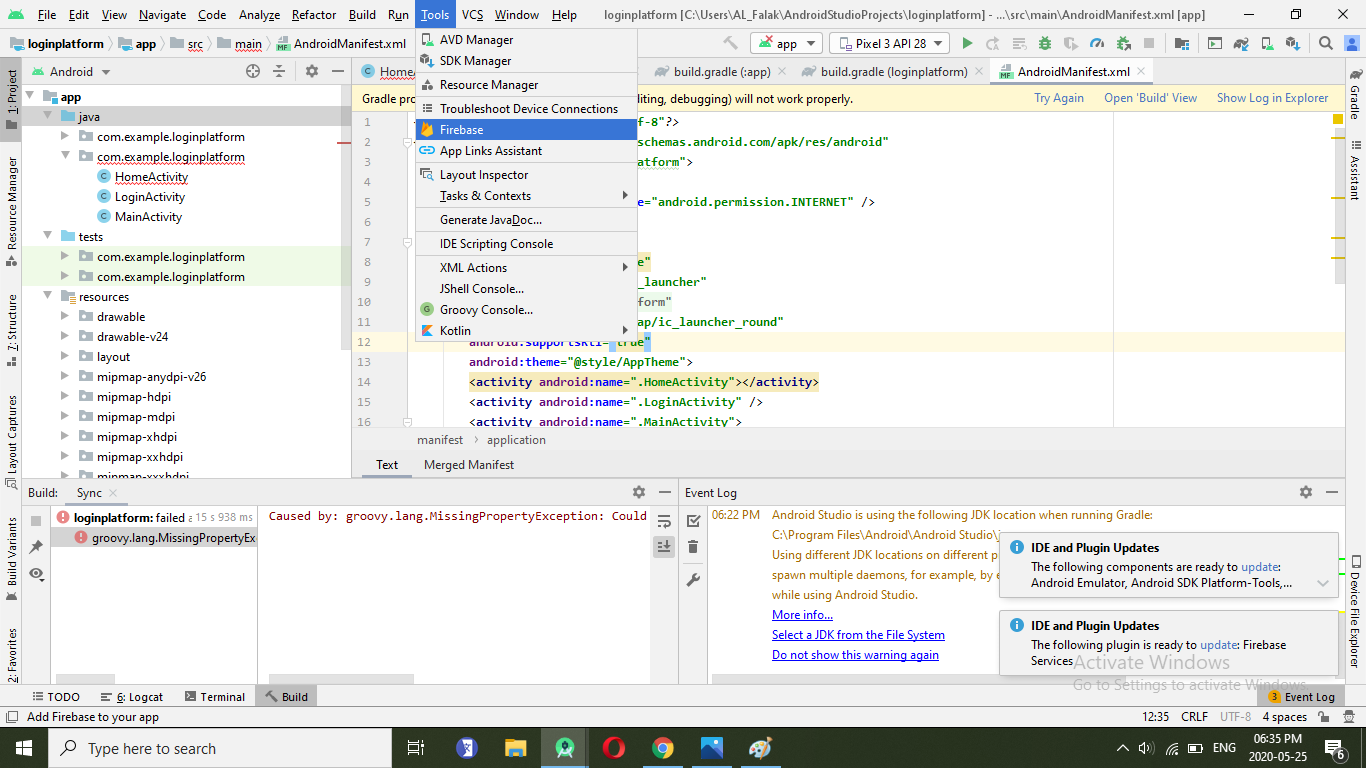


Fig 3.10 Firebase Setting

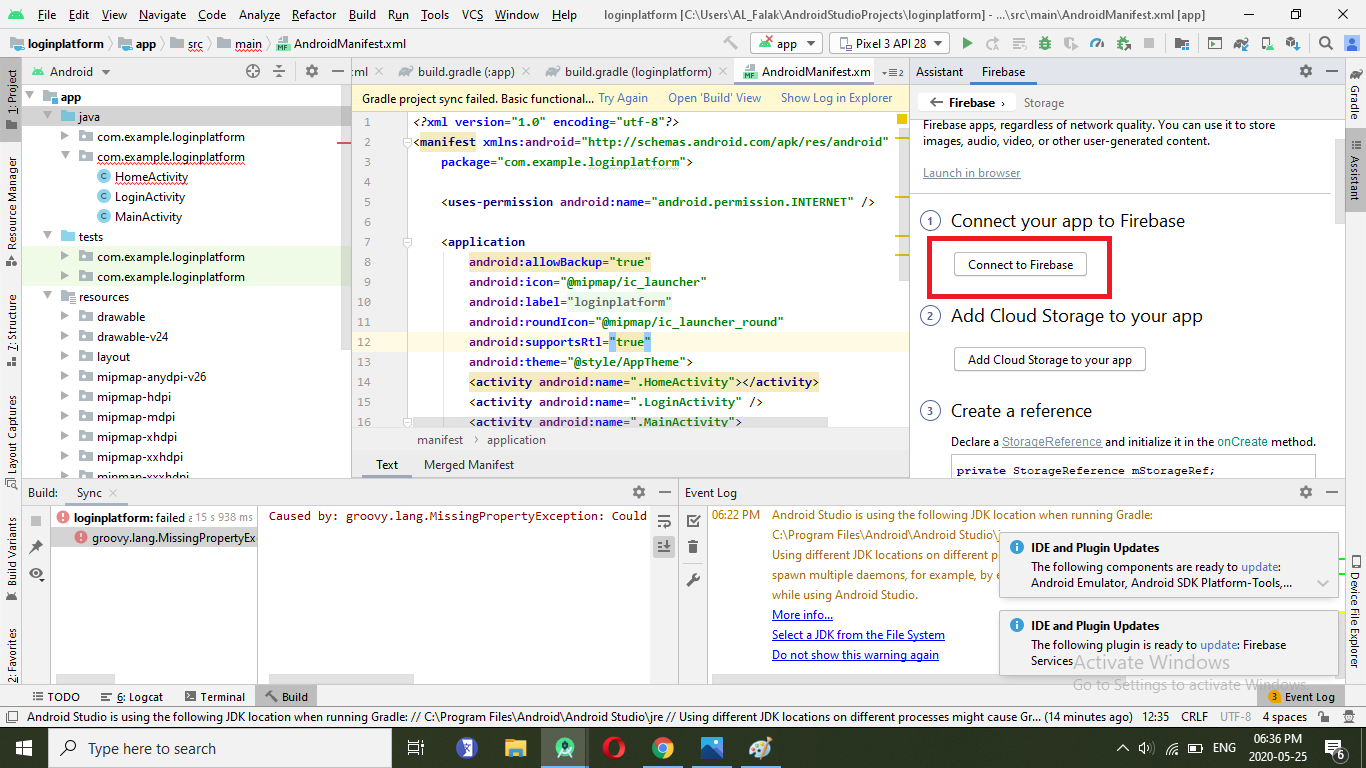


Fig 3.11 Connect Android studio with firebase

The android studio provides a virtual machine (or a virtual mobile) that provides the ability to run the developed java code, so you can see the logical view of the out coming result.

Our mobile application contains three interfaces, the first interface is for login, and the second interface is used for signing up to the system, the third Interface in the home page where the user can see the photos that were sent to him.

The sign in and signup functions will be done by using the user’s Gmail, it was done by authenticating the user’s info with the firebase to save the user’s info in it, simply by using their personal Gmail.

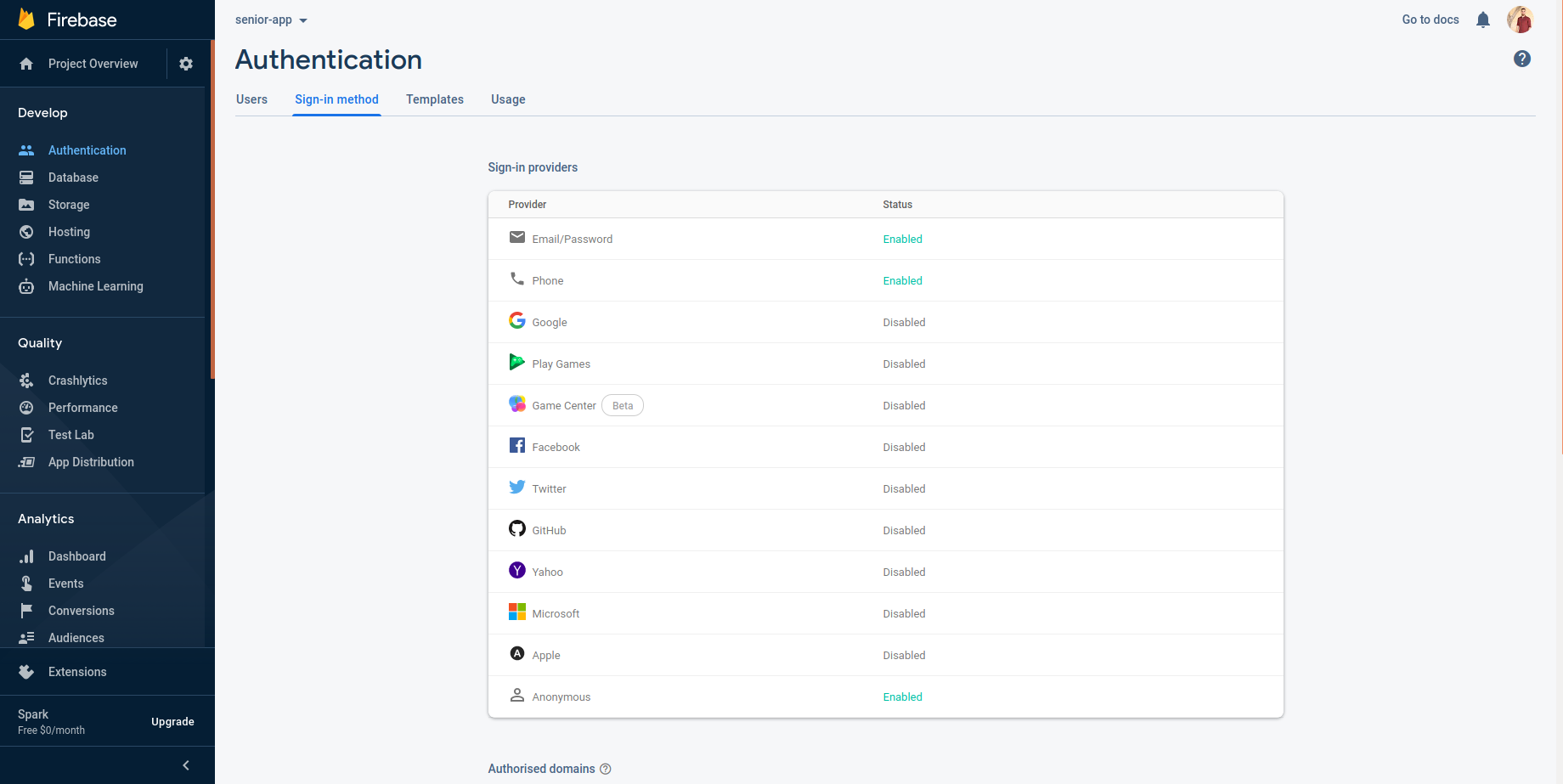


Fig 3.12 Firebase authentication

# This application has several functions:

1. **Sign Up**

In the first time use for this application, the user shall enter his/her own personal Gmail account and his/her private password to access the system to be added to the system’s own Firebase. Then it can be used later to access his/her own information (e.g. Photo)

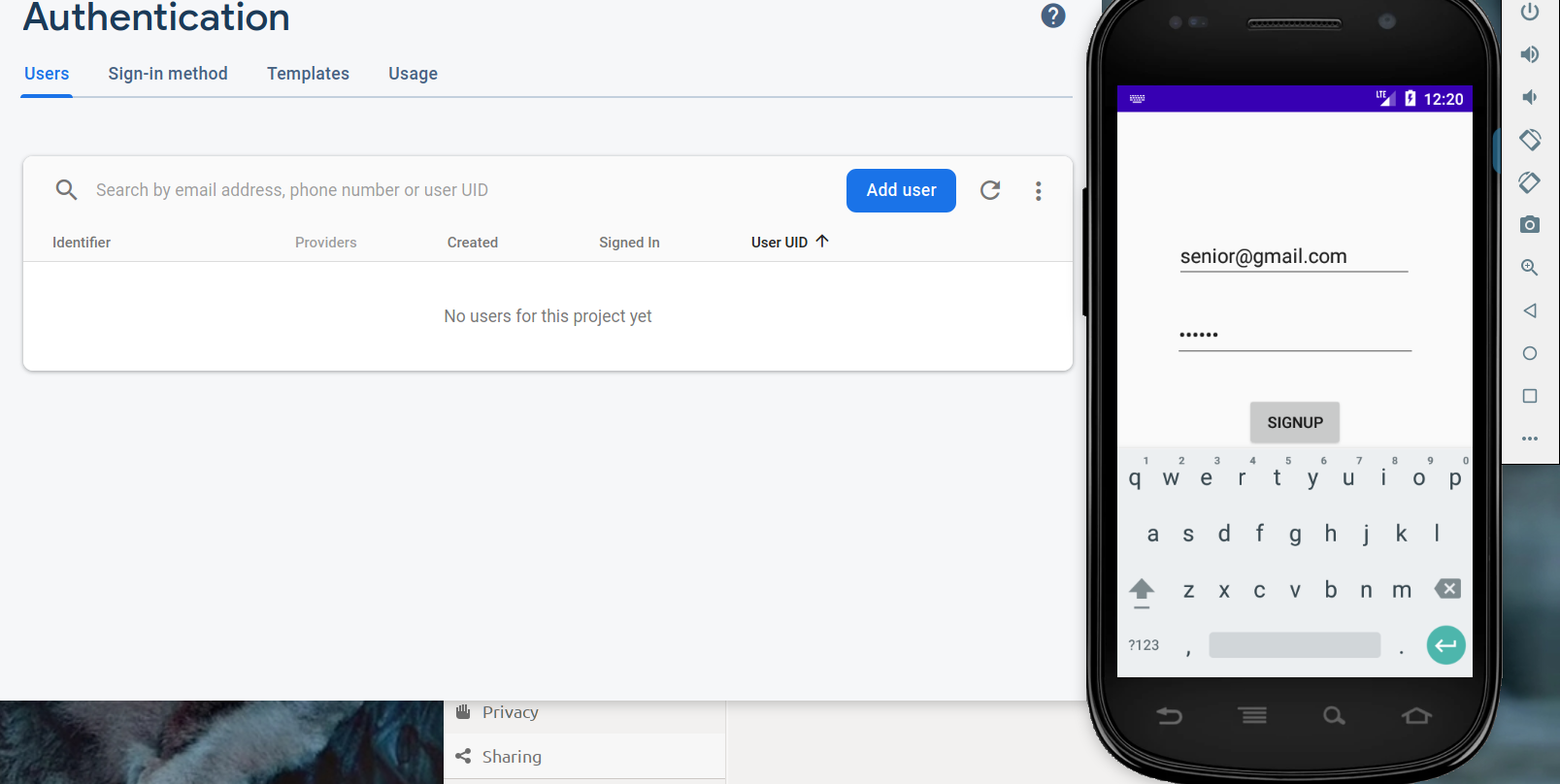


Fig 3.13 User Sign-Up Interface

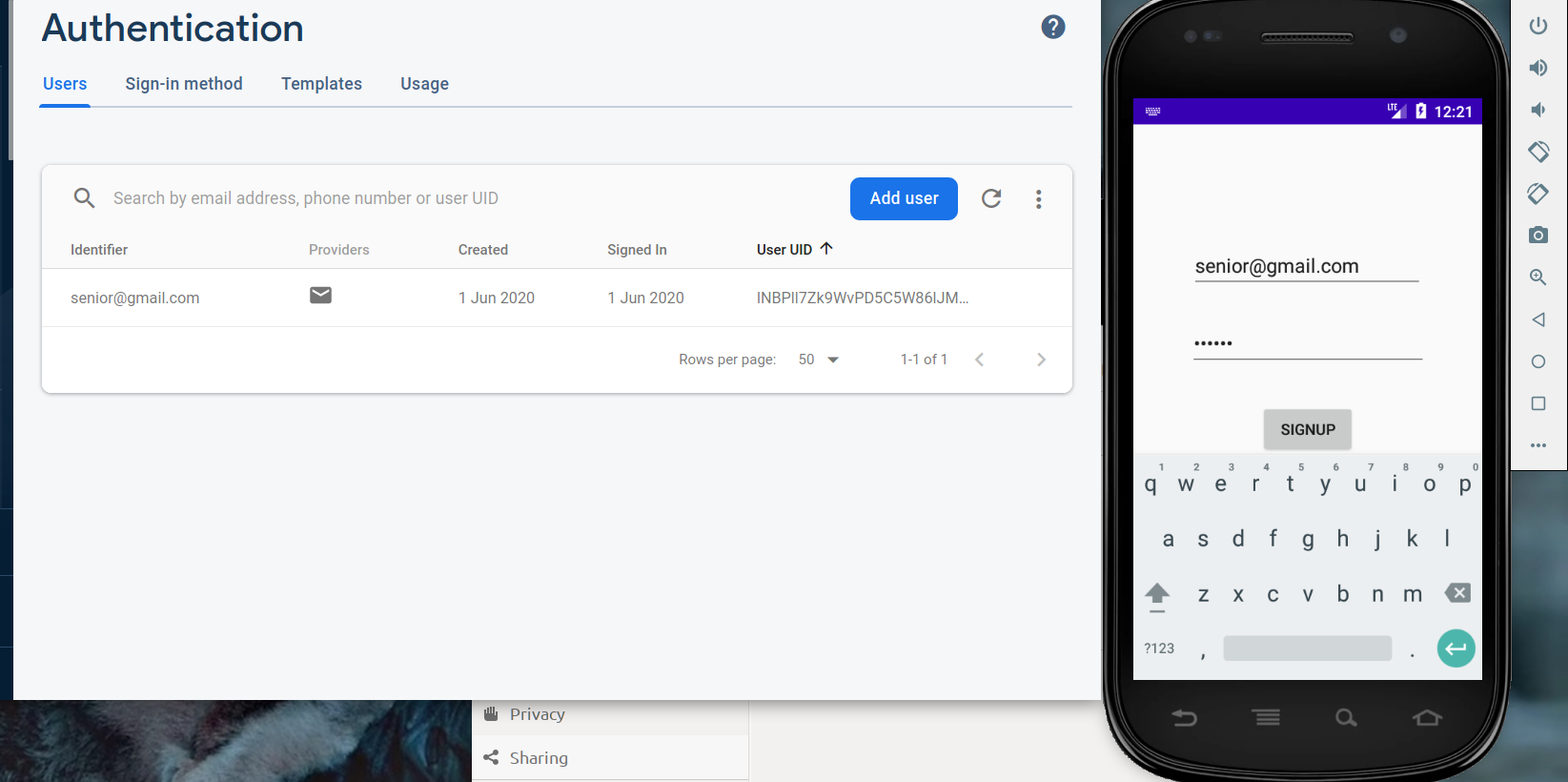


Fig 3.14 User Sign-Up Information is added to Firebase

1. **Sign In**

When the user signup by his/her Gmail, he/she will saved in firebase. Then the user will get authentication to use the application and he/she can sign in to the application by the Gmail.

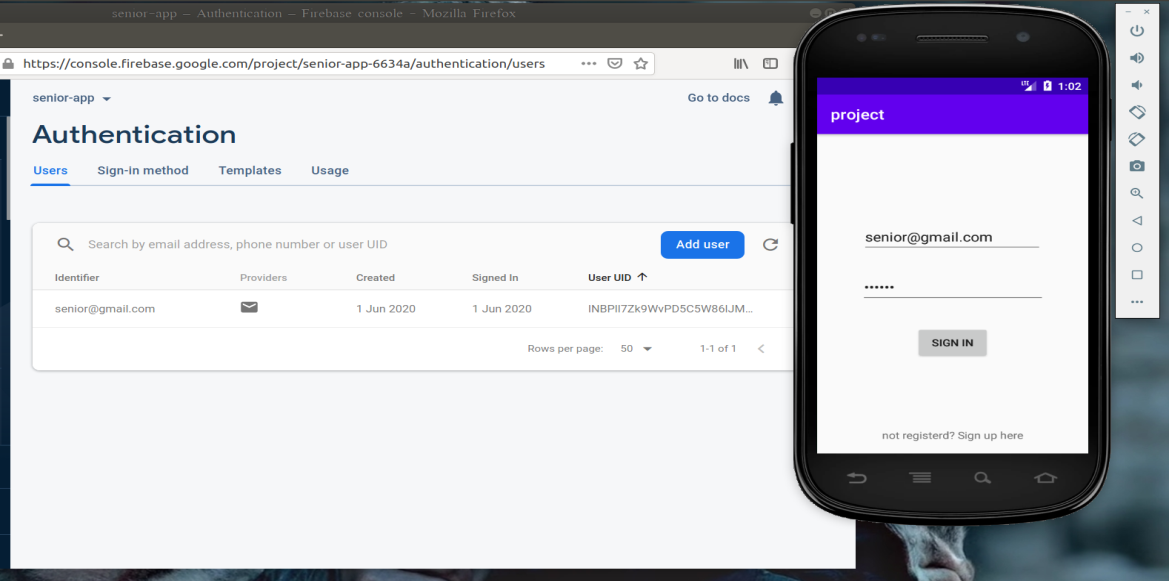


Fig 3.15 Sign In interface

1. **Notification Display**

When the pi cam detects any human motion, it send a picture for that motion to firebase. When a new photo added to the firebase, the mobile application receive this photo and display it to the user as a notification from the system’s android application.

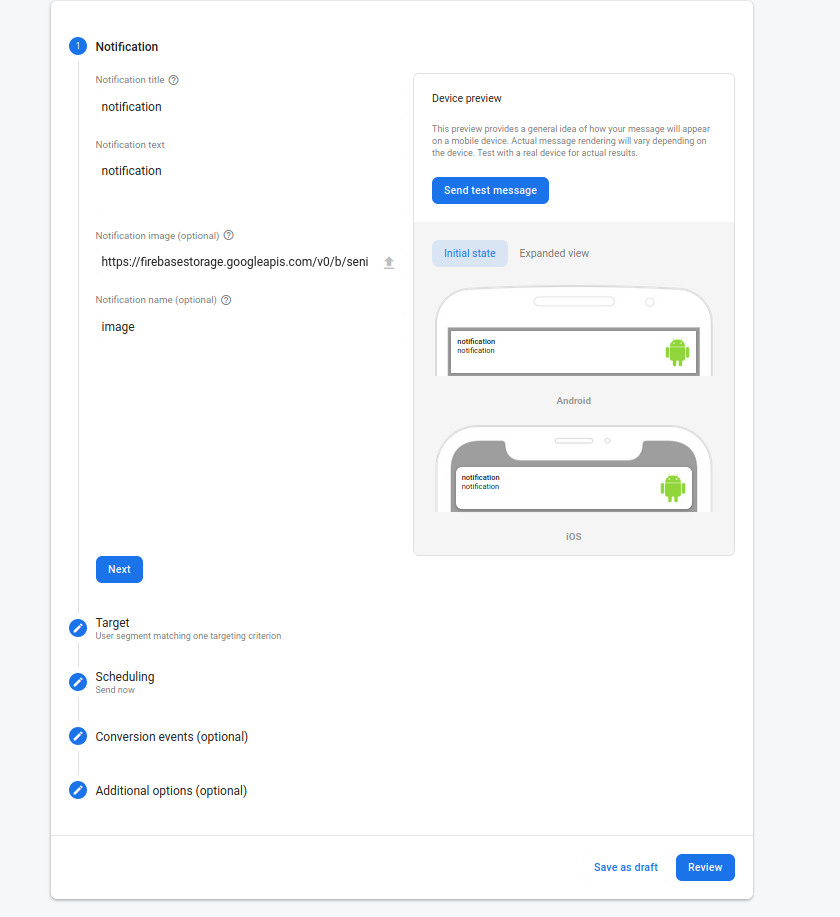


Fig 3.16 Notification Display Interface

1. **Photo Display**

When the user receives a notification with the captured photo, he/she can open the application to see the photo.

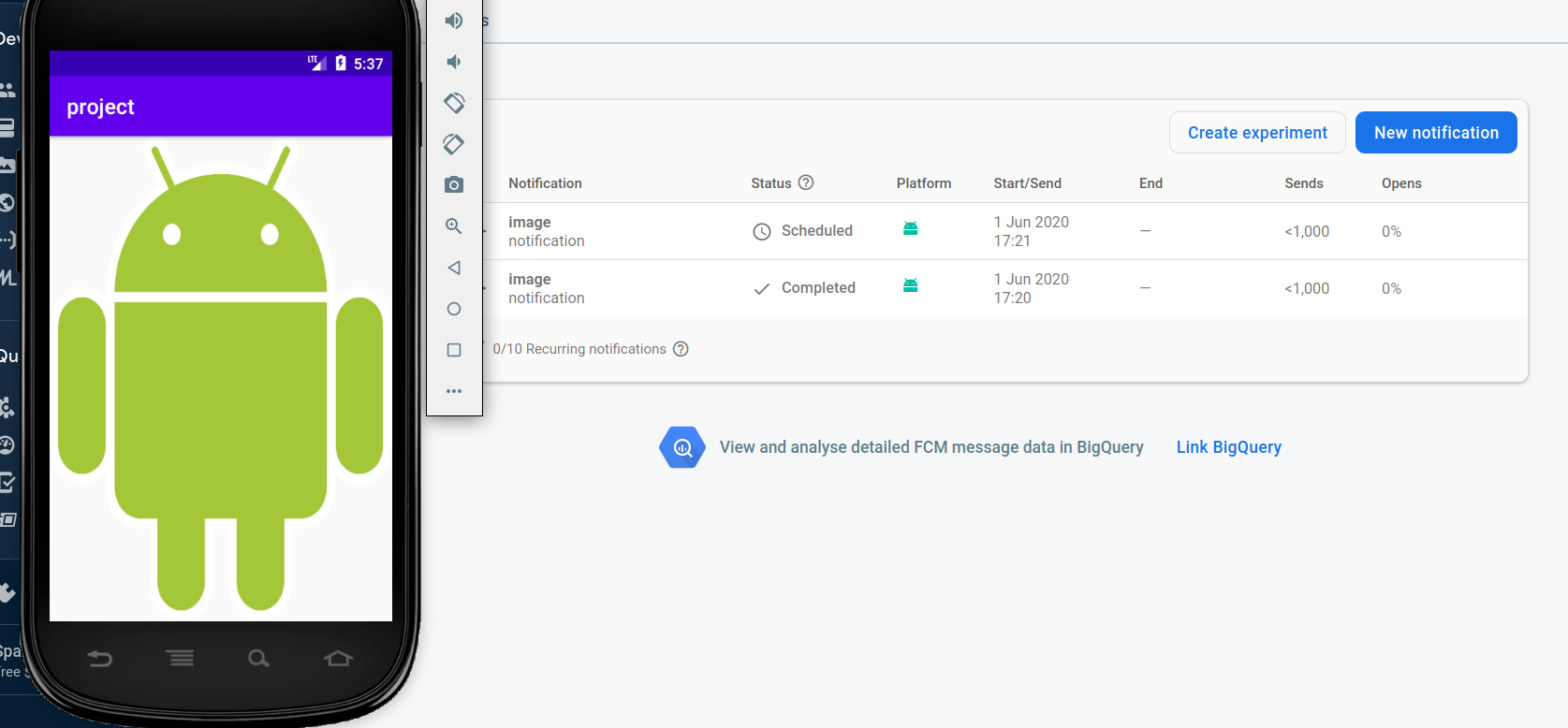


Fig 3.17 Display Photo interface

**Chapter Four**

**Conclusion and Future Work**

# 

# **4.1 Conclusion**

We have introduced Camera motion detection and alarm system which will help people to surveillance their properties and homes by using mobile application.

The system has many benefits:

* Giving safety to people over their things and possessions and homes.
* Notifying the user when a human motion happen in the area that the user put it under camera surveillance.

# **4.2 Future Work**

Add a new features such as:

1. Open live video stream, this gives the user of the android application, the ability to watch a what’s happening in front of the camera in real-time
2. Connect the system to a Firebase rather than Dropbox.

**Acknowledgments**

We would like to extend our sincerely thank our supervisor Prof. Dr. Adwan Yasin, who has been a great credited for our support and assistance in completing this project, we thank the Faculty of Engineering and Information Technology for its Dean and its lecturers, and we do not forget our Families, who are sacrificing their most precious possessions for us, in order to become good engineers and able to assume the responsibilities and support our beloved country Palestine

**References**

1. <http://www.pcbs.gov.ps/post.aspx?lang=en&ItemID=1872> - 15/3/2020
2. https://www.pyimagesearch.com/2015/06/01/home-surveillance-and-motion-detection-with-the-raspberry-pi-python-and-opencv/ - 28/4/2020
3. https://magpi.raspberrypi.org/articles/room-guard-build-a-raspberry-pi-motion-sensor-alarm - 31/6/2020
4. <https://www.google.com/search?safe=strict&rlz=1C1EJFA_enPS779PS779&biw=1242&bih=553&tbm=isch&sxsrf=ACYBGNTGZsIN8MD2SDVc7rq4jFzshBRLUg%3A1579037411529&sa=1&ei=4zIeXor4H9aFhbIPg4uP-AY&q=raspberry+pi+3+model+b%2B+&oq=raspberry+pi+3+model+b%2B+&gs_l=img.3..0l10.2896.13898..14225...0.0..0.163.2230.0j15......0....1..gws-wiz-img.......35i39j0i24j0i30.Oh9qCY2DcHQ&ved=0ahUKEwjK9YDghITnAhXWQkEAHYPFA28Q4dUDCAc&uact=5#imgrc=uo5gxuPCCgkygM:> - 31/3/2020
5. https://www.raspberrypi.org/products/camera-module-v2/ - 31/3/2020