Smart Traffic Monitoring System

Ali Saleh Baker - 2140160

TM471: Final Year Project, December 2018

# Abstract

This project is all about a system that helps universities, government and business owner, etc. to get the benefit of their CCTV cameras. This system is considered as a tool to allow the corporation to have a statistical data about the traffic in the cameras, in addition, they can make triggers when a specific object come through the camera, the system allows people to analysis recorded CCTV videos that are stored for weeks or months.

Also, the system has the ability to send an email or push notification in real time when an object detected by the system, and that allows the admins to make instant actions.

This system gives you the ability to build your own module for alerting or for control on IOT devices as triggers, or even to integrate with third-party API such as IFTTT (if this, then that) to get unlimited opportunities of using the system.

# Acknowledgements

I would like to express my gratitude to my advisor, Dr. Khalid Al Tahat, for his support, patience, and encouragement throughout my graduate project in the second-semester, It is rarely that one finds an advisor and colleague that always finds the time for listening to little problems and roadblocks that unavoidably crop up in the course of performing project.

# 

# 1. Table of contents:

**[1. Chapter 1: Introduction](#_1t3h5sf) 7**

[1.1. Introduction:](#_4d34og8) [7](#_4d34og8)

[1.2. Problem and Suggested Solutions:](#_3rdcrjn) [9](#_3rdcrjn)

[1.2.1. The problem](#_26in1rg) [9](#_26in1rg)

[1.2.2. Suggested solutions:](#_lnxbz9) [10](#_lnxbz9)

[1.3. Project Scope:](#_35nkun2) [11](#_35nkun2)

[1.4. Project Aims:](#_44sinio) [11](#_44sinio)

[1.5. Project Objective:](#_3j2qqm3) [12](#_3j2qqm3)

[1.6. Project Goal:](#_1y810tw) [12](#_1y810tw)

[1.7. Project Constraints:](#_4i7ojhp) [13](#_4i7ojhp)

**[2. Chapter 2: Literature Review](#_3whwml4) 14**

[2.1. Introduction](#_2bn6wsx) [14](#_2bn6wsx)

[2.2. Existing Projects](#_qsh70q) [14](#_qsh70q)

[2.3. Real-life existing software like this project:](#_3as4poj) [15](#_3as4poj)

[2.3.1. Roborealm:](#_1pxezwc) [15](#_1pxezwc)

[2.3.1.1. Main Features:](#_147n2zr) [16](#_147n2zr)

[2.3.1.2. Example of the software](#_3o7alnk) [16](#_3o7alnk)

[2.3.2. Objectrecognitionsoftware](#_32hioqz) [18](#_32hioqz)

[2.3.2.1. Main Features:](#_2grqrue) [18](#_2grqrue)

[2.3.2.2. Examples:](#_vx1227) [19](#_vx1227)

**[3. Chapter 3: Requirements and analysis](#_4f1mdlm) 20**

[3.1. Use Case of the project’s needs:](#_2u6wntf) [20](#_2u6wntf)

[3.2. Software development life cycle:](#_46r0co2) [21](#_46r0co2)

[3.2.1. Advantages of Agile model](#_3l18frh) [21](#_3l18frh)

[3.3. Project Schedule and activities.](#_206ipza) [22](#_206ipza)

[3.4. Project Gantt chart](#_1egqt2p) [23](#_1egqt2p)

[3.5. Software requirements for the Project](#_sqyw64) [24](#_sqyw64)

[3.6. Used open source technology in the project](#_3cqmetx) [24](#_3cqmetx)

[3.7. Details of each open source technology](#_2r0uhxc) [25](#_2r0uhxc)

[3.7.1. Operating system:](#_1664s55) [25](#_1664s55)

[3.7.2. Database server](#_3q5sasy) [25](#_3q5sasy)

[3.7.3. Backend](#_25b2l0r) [26](#_25b2l0r)

[3.7.4. Frontend:](#_kgcv8k) [26](#_kgcv8k)

[3.7.5. Web Server:](#_34g0dwd) [26](#_34g0dwd)

[3.7.6. Code repository](#_1jlao46) [27](#_1jlao46)

[3.7.7. MetaBase:](#_43ky6rz) [27](#_43ky6rz)

[3.8. Hardware Requirements:](#_2iq8gzs) [27](#_2iq8gzs)

[3.9. Project Analysis and scenario:](#_1x0gk37) [28](#_1x0gk37)

[3.9.0.1. Example](#_4h042r0) [29](#_4h042r0)

[3.9.0.2. Example](#_2w5ecyt) [29](#_2w5ecyt)

[3.9.0.3. Example](#_1baon6m) [30](#_1baon6m)

[3.9.0.4. Example](#_pkwqa1) [31](#_pkwqa1)

[3.10. Future improvement of the system](#_39kk8xu) [31](#_39kk8xu)

**[4. Chapter 4: Design, Implementation and testing.](#_1gf8i83) 34**

[4.0.1. Data Source:](#_upglbi) [35](#_upglbi)

[4.0.2. IOT Device:](#_3ep43zb) [35](#_3ep43zb)

[4.1. Use case of the system:](#_1tuee74) [36](#_1tuee74)

[4.2. Figure 14: The system use case diagram](#_1mrcu09) [37](#_1mrcu09)

[4.3. Activity Diagram:](#_4du1wux) [38](#_4du1wux)

[4.4. Database Diagram:](#_3s49zyc) [39](#_3s49zyc)

[4.4.1. Tables Description:](#_36ei31r) [39](#_36ei31r)

[4.5. Sequence Diagram:](#_1ljsd9k) [40](#_1ljsd9k)

[4.6. Evaluation method:](#_2y3w247) [43](#_2y3w247)

[4.6.1. On device analysis](#_1d96cc0) [43](#_1d96cc0)

[4.6.2. Database analysis.](#_3x8tuzt) [43](#_3x8tuzt)

[4.6.3. Alert trigger.](#_2ce457m) [43](#_2ce457m)

[4.6.4. Automation action trigger.](#_rjefff) [44](#_rjefff)

[4.6.5. Testing](#_3bj1y38) [44](#_3bj1y38)

[4.6.6. Test individual components:](#_2pta16n) [45](#_2pta16n)

[4.6.7. Test the final system](#_14ykbeg) [45](#_14ykbeg)

[4.7. Testing simulator](#_3oy7u29) [46](#_3oy7u29)

[4.7.1. Screenshot of testing result of the system.](#_243i4a2) [47](#_243i4a2)

**[5. Chapter 5: Results and discussion.](#_47hxl2r) 52**

[5.1. Findings](#_2mn7vak) [52](#_2mn7vak)

[5.2. Goals achieved](#_3ls5o66) [53](#_3ls5o66)

[5.3. Further work](#_302dr9l) [54](#_302dr9l)

**[6. Chapter 6 Conclusions](#_3z7bk57) 55**

**[7. References:](#_2eclud0) 56**

**[8. Appendix A:](#_3dhjn8m) 58**

**[9. Metabase](#_2rrrqc1) 61**

# List of Tables

|  |
| --- |
| [Table 1 Project Schedule and activities. 22](#_2zbgiuw)  [Table 2 open source technology 24](#_4bvk7pj)  [Table 3 hardware specification 28](#_3hv69ve)  [Table 4 Use case of the system 36](#_28h4qwu)  [Table 5 Final Test table 51](#_1nia2ey)  [Table 6 Achievement Goals 54](#_4kx3h1s) |

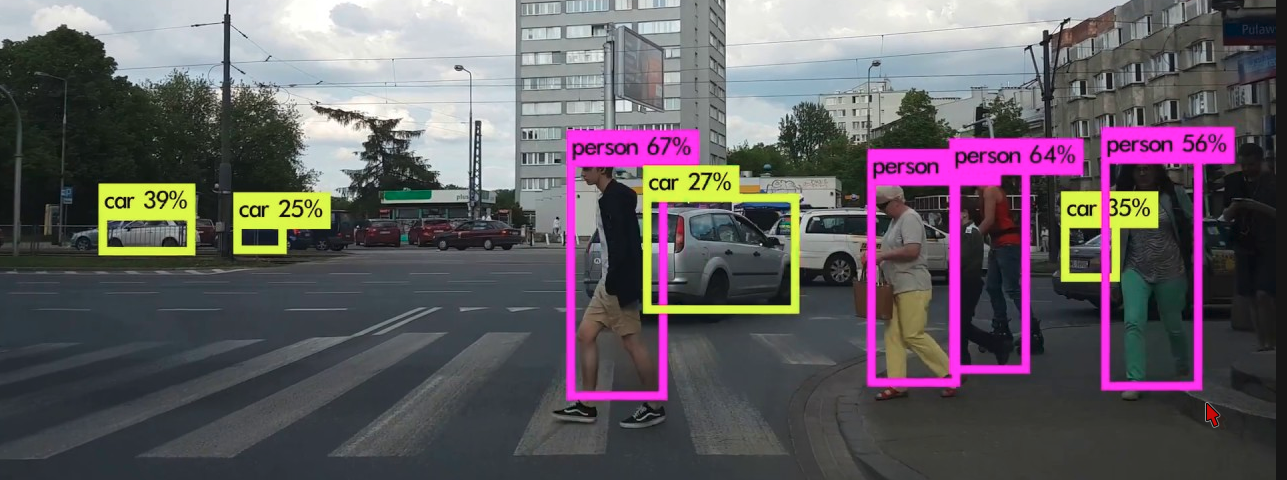
# 

# List of Figures

|  |
| --- |
| [Figure 1 Example of the object that system can take 7](#_2s8eyo1)  [Figure 2 Detected Object sample 13](#_2xcytpi)  [Figure 3 Project Logo of roborealm 15](#_49x2ik5)  [Figure 4 Project Logo of roborealm 16](#_23ckvvd)  [Figure 5 Project Logo of roborealm 17](#_ihv636)  [Figure 6 Objectrecognitionsoftware logo 18](#_1hmsyys)  [Figure 7 Objectrecognitionsoftware product exmaple 19](#_3fwokq0)  [Figure 8 Objectrecognitionsoftware product exmaple 19](#_1v1yuxt)  [Figure 9 Agile model 21](#_2lwamvv)  [Figure 10 Project Gantt chart 23](#_3ygebqi)  [Figure 11 example for the system for the Mall 30](#_3vac5uf)  [Figure 12 IFTTT Logo 31](#_1opuj5n)  [Figure 13 WeatherBug logo 32](#_2nusc19)  [Figure 14 Notification platform 32](#_3mzq4wv)  [Figure 15 report example 33](#_haapch)  [Figure 16 Infrastructure diagram 34](#_40ew0vw)  [Figure 17 use case diagram 37](#_nmf14n)  [Figure 18 Activity Diagram 38](#_2szc72q)  [Figure 19 Database Diagram 39](#_279ka65)  [Figure 20 internal components sequence diagram 41](#_45jfvxd)  [Figure 21 sequence diagram describe how system send alert/push messages 41](#_zu0gcz)  [Figure 22 system save data into CSV 42](#_1yyy98l)  [Figure 23 System testing diagram 44](#_1qoc8b1)  [Figure 24 runtime output of the system 47](#_j8sehv)  [Figure 25 video feedback output when processing stream. 48](#_1idq7dh)  [Figure 26 the real time dashboard output 48](#_2hio093)  [Figure 27 IOT Trigger action 49](#_3gnlt4p)  [Figure 28 the push notification that send to the phone/Browser 49](#_4fsjm0b)  [Figure 29 the email example 50](#_1a346fx)  [Figure 30 data saved in the database 51](#_2981zbj)  [Figure 31 Question Answering software 52](#_11si5id)  [Figure 32 Tinker board 60](#_1smtxgf)  [Figure 33 metabase example 61](#_16x20ju)  [Figure 34 metabase example 62](#_3qwpj7n)  [Figure 35 metabase example 62](#_261ztfg)  [Figure 36 metabase example 62](#_l7a3n9) |

# 1. Chapter 1: Introduction

## 1.1. Introduction:



**Figure 1 Example of the object that system can take**

a leak of statistical data in different live aspect leads the corporation/government to miss understand and lead also for unwise decision making because of leaking of the right information.

The decision maker should have the ability to look into the data to a better decision and predicted regarding the future of the business or another thing. That can be done with help of the statistics and statistical concepts.

Nowadays every decision was made needs to be based on a good bunch of statistical data. Check below why the statistical data is important for the decision maker.

* Big Picture: statistics help businesses in getting the big picture with the statically analyzed data.
* Support of judgment: it helps to create an accurate decision. That decision is based on the data.
* Insights hidden data: by use tools that can be getting the relation between data and extract hidden relation from the data.
* Ensures Quality: Statistics makes business to produce goods with limited variations and wastage; it also helps increase the productivity of the workers. Thus, saving money and ensuring the best quality with optimum utilization of the resource.

In the next paragraph, you will find how important the statically data that used by Jordan government.

Amman with a population of 4 million or 42% of the population lives in Amman in the year 2015 and over the year the number of refugees is increasing too. With this scenario, it is expected that the total number of vehicles uses the Highway roads in increase and road user safety decreasing.

This indicates a potential demand for investment in transport infrastructure. Proper utilization of such huge investments necessitates systematic planning for need-based development. Such need-based developments include determination of the required capacity expansion, provision of additional road infrastructure, improvement of existing roads, prioritization of different development phases and forecasting of which is possible upon collection of traffic data. This is done in order to eliminate bottlenecks in both international and local inter-urban road transport towards providing an efficient and effective road transport system.

So of using the statistical system that will monitor the traffic for categories it and store it into the database so the decision maker later can look into the data of the traffic to get better understand and appropriate planning and management of the traffic points.

By using an intelligent technology combined together to produce a well and meaningful data metrics, and also use the automatic data insights with any business intelligence tool that can give us a relation and hidden data that we cannot know about it.

In the next chapters you can find how what the problem the system will solve and how the system will work in details and also you will find the used technologies that are the heart of the system.

You can check Figure 1. For one of examples of the reports.

## 1.2. Problem and Suggested Solutions:

Mainly too many business or even government had a common problem which in decreasing the number of accidents and an increasing number of safety in different field and also increase the profit after all.

TO achieve that goal they need to apply efficient and reliable and less complex hardware components and of course cost effective solution to solve their pain.

### 1.2.1. The problem

* Risks Attributable To Incompatibility Between Different Types Of Vehicles And Groups Of Road Users
* Differences In Risk Between Different Types Of Traffic Environment And Speeding.
* Taxonomy Of Road Safety Problems Is Developed In Order To Identify The Characteristics Of Problems That Can Make Them Difficult To Solve.
* It’s Hard to Detect Accurate AVG/max Speed of the Road Traffic.
* Hard to Detect the Road User For Example (humans, Cars, Buses. etc.).
* Governments Do Not Have Enough Information About The Traffic In The Borders.
* The Corporation Needs To Know Extra Knowledge Of The Traffic In The Garage For Example How Many Cars Now Inside The Mall Or After A Close Hour Does Anyone Still Exists In The Area.

### 1.2.2. Suggested solutions:

This project will provide simple and cost-efficient system, will work in any type of camera already installed on the road or in any camera pointing to the traffic point, so mainly the system will able to do the following parts

* Being able to detect various objects from the video stream.
* Being able to store statistical data into database.
* Being able to take actions upon object detected.
* Analytical software able to get Insights analyzing hidden data from the database.
* Connect the system in the external part to do some action.
* The system will be in the cloud-based solutions.
* Secure and efficient system.
* The system is flexible; you can use it in many fields.

## 1.3. Project Scope:

The project will implement as three parts the major changes in the backend and the other part is the frontend that shows metrics and graphs and the important information that user need for the needed points.

Backend server: that take camera inputstream to make the needed analysis and save result into main database which is based in the cloud and performs specific operations and actions.

Frontend server: will work in parallel with the backend server and communicated with the database, the output will be a metrics and graph and needed information for the user/admin.

Data Storage:

Database will contain all data that already proceed, and all related information based for each point (camera) and also predefined action the system can do.

## 1.4. Project Aims:

1. Insure ability to detect different types of object that each end is interesting in.
2. Get real time metrics for what happening in the each point.
3. Get historical data to make a better decision for the decision maker.
4. A centralized data store for all collected data.

## 1.5. Project Objective:

1. Provide Real Time alerting for new predefined settings.
2. Statistical data summary will be generated from the data store for decision maker’s review.
3. Ability to handle and use any type of IP cameras (no specific hardware required)
4. Software that will running at the device does not require a high expensive hardware.

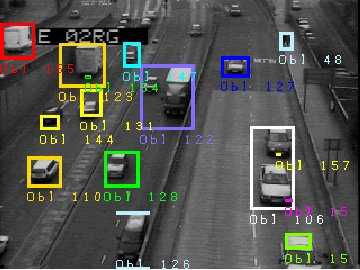
## 1.6. Project Goal:

1. System can analysis CCTV camera video input for any time of range.
2. Detect vehicle speed statistical data such as AVG/Max speed.
3. Provide historical data about the traffic example: counter of car or buses.
4. Categories object types of traffic.
5. Send real-time alert when detecting special object time. For example send push notified when detecting human after 12:00 PM
6. Do some action when detect special object time, for example, turn on the light., close the door.

An extra feature in the future:

1. Send an alert if X person detects by the system.
2. Make the system integrated with IFTTT service.

## 1.7. Project Constraints:



**Figure 2 Detected Object sample**

1. Not all types of object will be cover at the current project implementations.
2. Only IP cameras can connect to the system, not camera with coaxial cable
3. Not able to detect an accurate speed of the object because system does not tracking object just works on frames by frames as shown on Figure 2.

You can find in the next chapters more details about project and software architecture and project plan.

# 2. Chapter 2: Literature Review

## 2.1. Introduction

Usually there is a gap between a theory and practical application, When I decide to start building this project I had a good base of knowledge but later on when I start to implement this project I found I not, SO that gives me a good vision to start the new project that based on technology and good skills in development STMS, So Unfortunately there was a gap between my understanding and the complete understanding of the problem that I want to solve.

To fill that gap I had to make a deep search and reading e-book YouTube tutorials and see a lot of examples to able to start implementing the project.

## 2.2. Existing Projects

Below you can find two major examples for project that similar with my project.

## 2.3. Real-life existing software like this project:

### 2.3.1. Roborealm:



**Figure 3 Project Logo of roborealm**

http://www.roborealm.com/index.php

rob realm simplifies vision programming and allows you to easily prototype with many advanced modules to achieve the desired result.

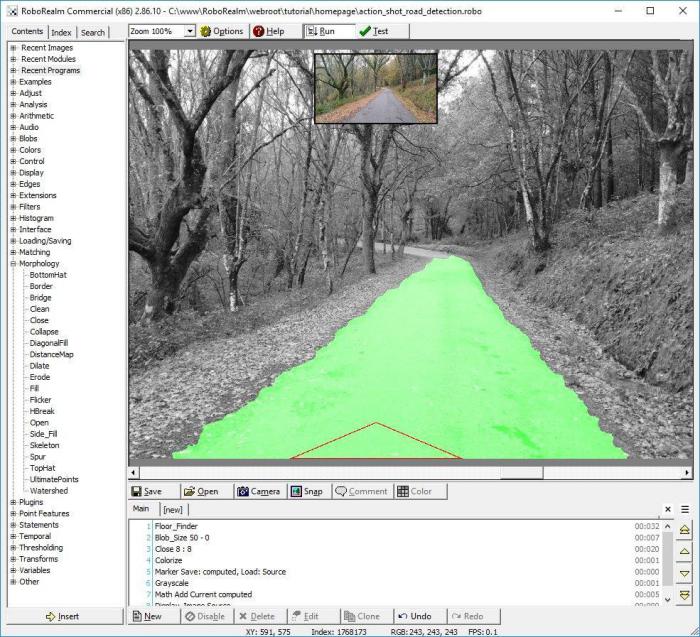
How that system work:

* One of the most basic operations in #machine vision is to segment objects from the background in order to run specific tests against only those areas of the image that are of interest.
* Comparing a sample object to a known object is an effective way of determining if a particular object conforms to a known standard.
* The Surface Texture module is used to highlight surface irregularities that define texture. Similar to edge detection.

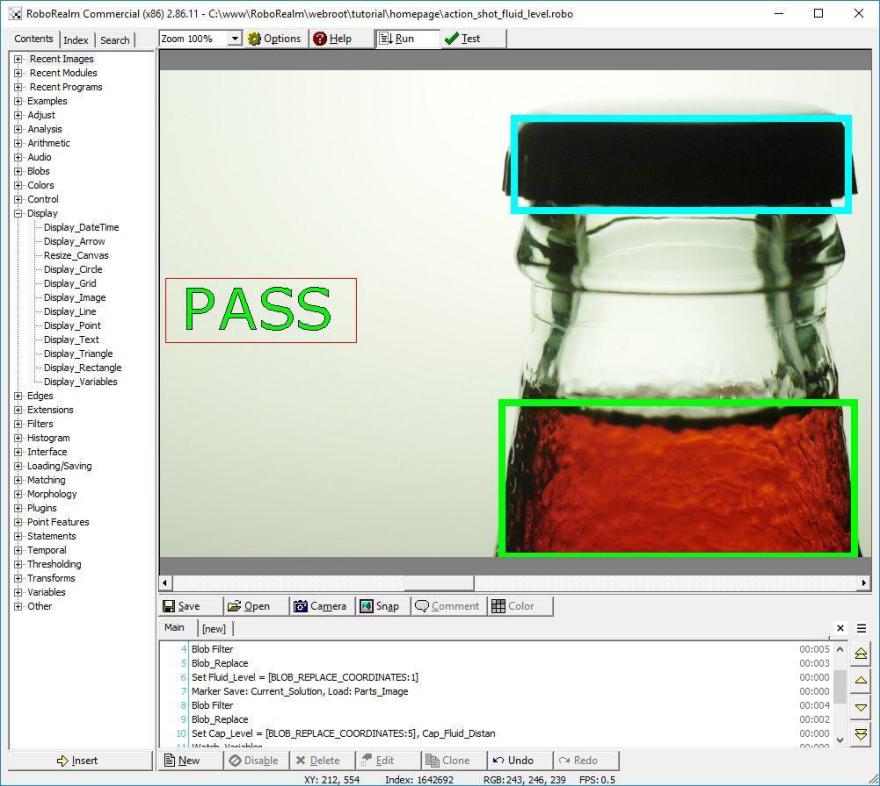
#### 2.3.1.1. Main Features:

* + Easy to Use GUI Interface
  + Hundreds of Image Processing Modules
  + Camera Agnostic
  + Real-time Parameter Changes
  + Fully Supported Server API
  + Multiple Image Sources
  + Multiple Output Interfaces (File, Web, FTP, Email, etc)
  + Plugin Framework for Custom Modules
  + Denver, Colorado, USA based company

#### 2.3.1.2. Example of the software



**Figure 4 Project Logo of roborealm**

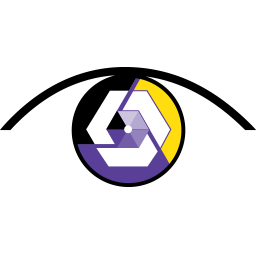


**Figure 5 Project Logo of roborealm**

For More info:

* http://www.roborealm.com/index.php

### 2.3.2. Objectrecognitionsoftware



**Figure 6 Objectrecognitionsoftware logo**

*Figure 5.*

https://objectrecognitionsoftware.com/

Object Recognition software is tailored to meet the needs of your unique use-case. The reason for this is because generic off-the-shelf software is unable to accommodate the vast differences encountered from one project to the next.

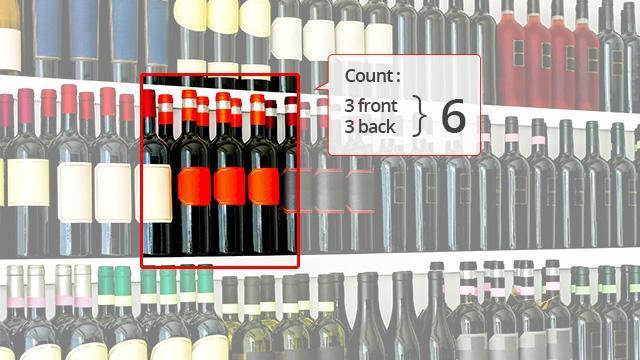
#### 2.3.2.1. Main Features:

* Recognize and analyze items in photos and video.
* Count and measure any type of object in videos and images, even when they're in motion.
* Visual search. Computer vision enables us to develop the specific application you need to search and locate any type of object.
* Analyze motions. Measure body parts and the relationships between them.

#### 2.3.2.2. Examples:



**Figure 7 Objectrecognitionsoftware product exmaple**



**Figure 8 Objectrecognitionsoftware product exmaple**

More info:

* https://objectrecognitionsoftware.com/

# 3. Chapter 3: Requirements and analysis

Sourcing the right software development for my university project is incredibly important, the main advantage of agile development model is that it is simple and easy to use and understand and provide a fast delivery, and since I change project in project 2 that approach give me the chance to and the flexibility to delivery my project in the deadline.

The project requirement will be analyzed and presented using an appropriate method which is UML (Unified Modeling Language). And also you will find the ERD (entity relationship diagram) for the database design.

## 3.1. Use Case of the project’s needs:

* System must able to analysis IP camera video stream.
* System must be able to analysis and old video data for wild time range analysis.
* System must be able to run a specific trigger if one of the object found.
* System must be able to alert admin for any X object detected in X areas.

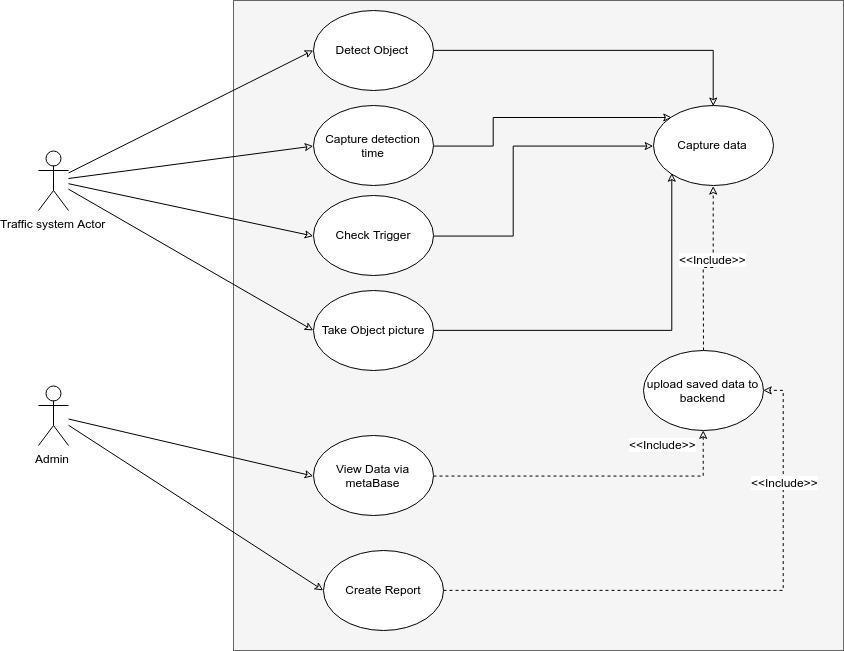
## 4.1. Use case of the system:

|  |  |
| --- | --- |
| Actor | Use Case |
| Traffic system | Detect Object |
| Capture detection time |
| Check Trigger |
| Take Object picture |
| Save generated data CSV |
| upload saved data to backend |
| Insert Data into DB |
| Admin | View Data via metaBase |
| Create Report |

**Table 4 Use case of the system**

## 4.2. Figure 14: The system use case diagram

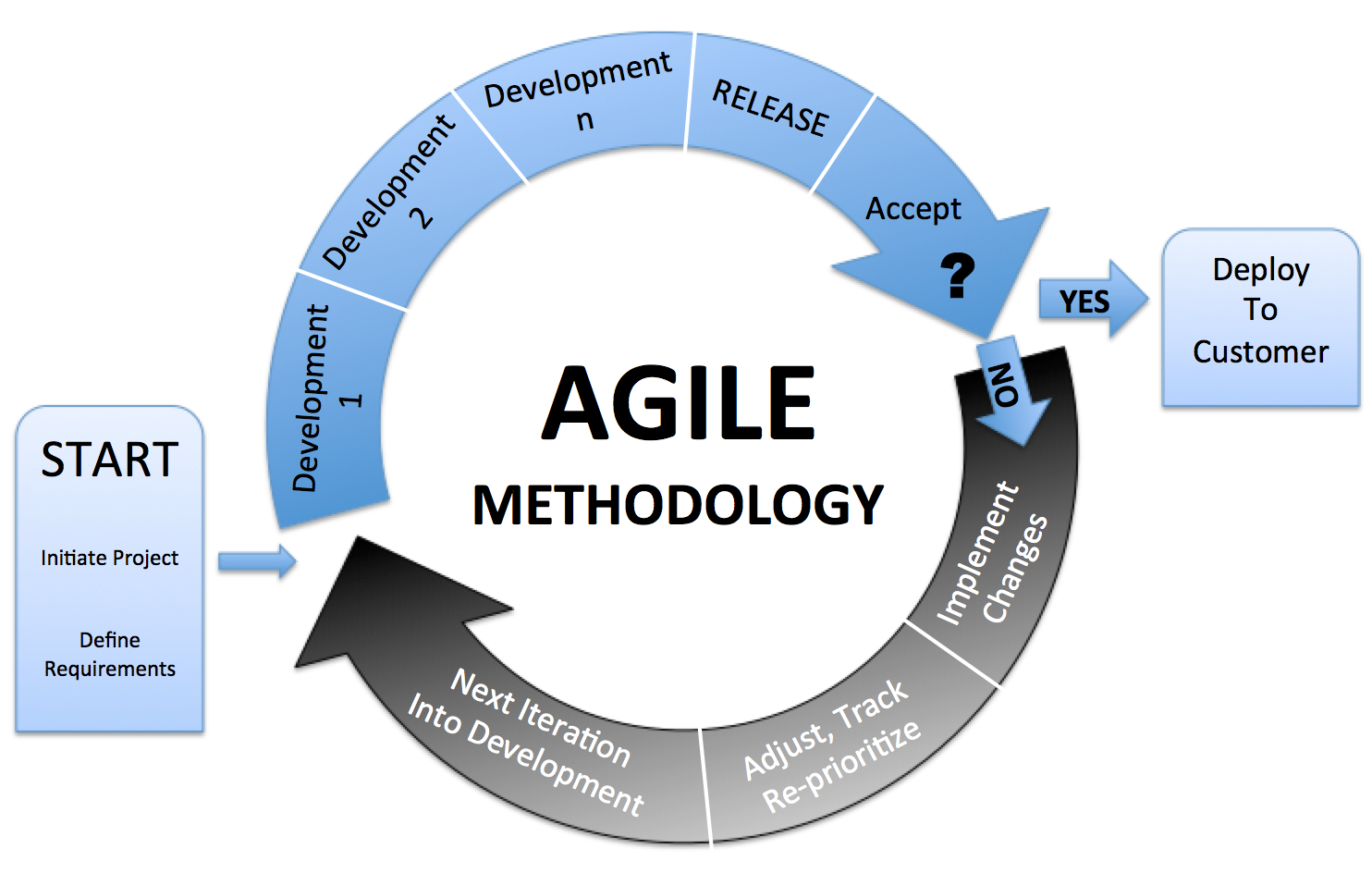
The use cases in Table 4 above will be illustrated graphically in Figure 14 below to allow for more insight into the relationships existing in the system.



**Figure 17 use case diagram**

## 3.2. Software development life cycle:

Because we looking for flexible approach to development this project, agile approach will be the fit approach to follow.



**Figure 9 Agile model**

### 3.2.1. Advantages of Agile model

1. Continuous delivery of useful software.
2. People and interactions are emphasized rather than process and tools.
3. Working software is delivered frequently (weeks rather than months).
4. Face-to-face conversation is the best form of communication.
5. Close daily cooperation between business people and developers.
6. Continuous attention to technical excellence and good design.
7. Regular adaptation to changing circumstances.
8. Even late changes in requirements are welcomed.

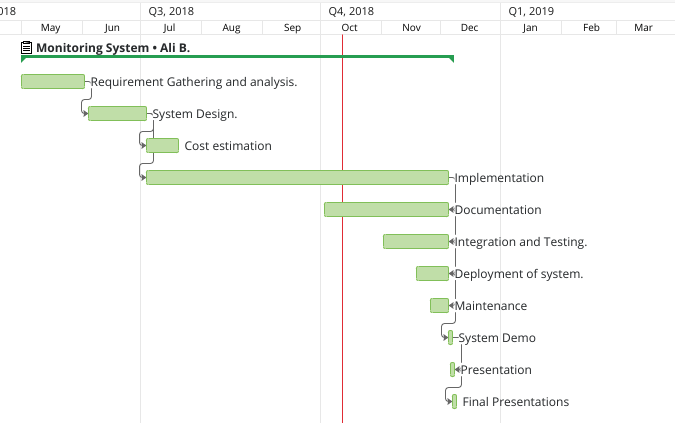
## 3.3. Project Schedule and activities.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Title | Start Date | End Date | Depends On |
| 11 | Requirement Gathering and analysis. | 5/1/2018 | 6/1/2018 |  |
| 10 | System Design. | 6/4/2018 | 7/3/2018 | 11FS |
| 4 | Cost estimation | 7/4/2018 | 7/19/2018 | 10FS |
| 9 | Implementation | 7/4/2018 | 12/4/2018 | 10FS |
| 8 | Documentation | 10/3/2018 | 12/4/2018 | 9FF |
| 7 | Integration and Testing. | 11/2/2018 | 12/4/2018 | 8FF |
| 6 | Deployment of system. | 11/19/2018 | 12/4/2018 | 7FF |
| 5 | Maintenance | 11/26/2018 | 12/4/2018 | 6FF |
| 3 | System Demo | 12/17/2018 | 12/6/2018 | 6FS |
| 2 | Presentation | 12/17/2018 | 12/6/2018 | 3FF |
| 1 | Final Presentations | 12/17/2018 | 12/7/2018 | 2FS |

**Table 1 Project Schedule and activities.**

## 3.4. Project Gantt chart

Gantt chart is a visual view of tasks scheduled over time.



**Figure 10 Project Gantt chart**

## 3.5. Software requirements for the Project

This project divided into multi-part, one part will receive the stream from camera/video and make the analysis process, and part to display the output of analysis in real time, and part to store collected data into the database, and also part uses software to do the visualization data into a dashboard.

IOT device will connect to the IP camera and do the local analysis in field analysis and push the data to the cloud system.

Cloud system will store data into database.

Below explanation of which parts were used and why.

## 3.6. Used open source technology in the project

This project totally based on open source software that combined together to produce a great software such this one.

|  |  |  |
| --- | --- | --- |
| **Software** | **Free** | **Usages** |
| Ubuntu | Open source | OS that run the project. |
| Python | Open source | Python is the programming language project that implemented to on |
| Wordpress | Open source | Wordpress is used here for marking site of the project |
| Tinker Board | Open source | Thinker board used here to run IOT triggers |
| OpenCV | Open source | OpenCV used here read stream and forward frames to the tensorflow |
| tensorflow | Open source | We used object detect API in tensorflow and labeling data. |
| Mysql | Open source | Mysql used for data storage of the system. |
| Nginx | Open source | Nginx use to serve marketing site worpdress. |
| GIT | Open source | Get used to manage code repository |
| MetaBase | Open source | Metabase is the easy, open source way for everyone in your company to ask questions and learn from data. |

**Table 2 open source technology**

*Table 2.*

## 3.7. Details of each open source technology

### 3.7.1. Operating system:

* 1. Ubuntu 18.04 LTS
     1. Ubuntu is free
     2. It’s secure. Say no to anti-virus.
     3. Supportive Ubuntu community
     4. Low system requirements
     5. It’s open source

### 3.7.2. Database server

* 1. MySQL:
     1. On-Demand Scalability
     2. High Availability
     3. High Performance
     4. Open Source
     5. Supportive MySQL community

### 3.7.3. Backend

* 1. Python
     1. Easy to Learn and Use
     2. Free and Open Source
     3. Object-Oriented Language
     4. Integrated
     5. Large Standard Library
     6. Cross-platform Language

### 3.7.4. Frontend:

* 1. Word press:
     1. Simplicity
     2. Flexibility
     3. User Management
     4. Media Management
     5. Extend with Plugins
     6. Own Your Data

### 3.7.5. Web Server:

* 1. Nginx:
     1. Reverse proxy with caching
     2. Handling of static files, index files, and auto-indexing
     3. Load balancing
     4. Support multiple backend apps
     5. GZIP compression
     6. Zero downtime
     7. Simpler security requirements
     8. Mitigate security and DDoS attacks
     9. Scalability and fault tolerance

### 3.7.6. Code repository

* 1. Git Server
     1. Feature Branch Workflow
     2. Distributed Development
     3. Pull Requests
     4. Faster Release Cycle

### 3.7.7. MetaBase:

* Metabase is the easy, open source way for everyone in your company to ask questions and learn from data.

## 3.8. Hardware Requirements:

Minimum Hardware Requirements for the central server.

This section describes the minimum hardware requirements.

* 1 CPU Cores
* 2 GB RAM
* Disk I/O subsystem applicable to a write-intensive database

6.6.2. Suggested hardware specification for the on field device:

|  |  |
| --- | --- |
| CPU | Rockchip Quad-Core RK3288 processor |
| Memory | 2GB Dual Channel DDR3 |
| Graphic | ARM® Mali™-T764 GPU\*1 |
| Storage | Micro SD(TF) card slot |
| LAN | RTL GB LAN |
| Wireless Data Network | 802.11 b/g/n |

**Table 3 hardware specification**

* Specification for Asus Tinker Board that will be used in the project.

## 3.9. Project Analysis and scenario:

When connecting the camera to the IOT device system will be start analysis the video stream and detect the object the system will store the information into the CSV file so the later the system can send the file info the backend server that will handle the file in parse it then stores it into the database.

Need from creating CSV is to make there is no data loss when the connection lost.

Workflow in details:

After turn on the IOT device, it's will connect to the cloud system so it's can download its own setting such as camera IP setting and the predefined action if required and save the setting into the device in case restart occur happened or and reason of losing setting from the memory.

After the first connection, the IOT device will connect the camera and start analysis the video input, after detecting an object from the video stream system will store each object in a new row in the CSV file. and the next step checking the predefined action so it's can do some action if X object detects for example:

#### 3.9.0.1. Example

If the system detects Human, Car in the university campus after 11:00 PM the system will send a push notification to the responsible person to alert him about unexpected motion in the area.

In the normal mode, the system will be keeping store object into the database for the statistical purposes.

#### 3.9.0.2. Example

Example of using statistical data that provided by the system:

If the camera is set up into the street system will able to detect the following:

* Human
* Vehicle
* Speed of the vehicles.

All the above information can help the decision maker to do the following:

Get better road safety as below:

If the road has too many vehicles and the avg speed is above the limit then they need to do the following:

1. Create traffic light.
2. Build a road air bomb.

If the human element in high by the time in the street, then they need to build a people bridge to get a better safety to the people in that area.

#### 3.9.0.3. Example



**Figure 11 example for the system for the Mall**

Another example for the system for the Mall owner:

* Let say the owner needs to know how many cars already exists inside the mall right now.

Answer:

* The system will be able to count the vehicle number in both directions in/ out.

So that information will be helpful for the owner to know the following:

Is the mall garage able to handle the number of cars?

AVG car number enters to the mall per hour/day/month. Etc.

It’s possible to know the number of people that enter to the mall per hour/day.. Etc.

Check if there are people inside the mall after closing hour.

all the above information can help the owner to provide a better service quality to the customers and enhance the service to get more profit.. Etc.

#### 3.9.0.4. Example

Another Use case:

If the government need to analysis CCTV recording for the previous year to any duty, system will be able to analysis the all CCTV video recording to save a huge time for human analysis..

## 3.10. Future improvement of the system

In the future I plan to improve that system to be in the cloud, and able to analysis multiple video stream at same time and begin able to trigger an action and alerts in suitable way.

Also adding new modules to help the customer to get full power of the system, below you will find the top modules that I paling to do:

* Module that integrate with IFTTT.



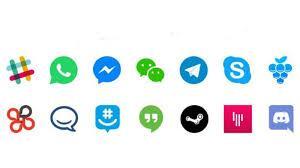
**Figure 12 IFTTT Logo**

* + If This Then That, also known as IFTTT, is a free web-based service to create chains of simple conditional statements, called applets. An applet is triggered by changes that occur within other web services such as Gmail, Facebook, Telegram, Instagram, or Pinterest
* Modules that get the camera environment information.



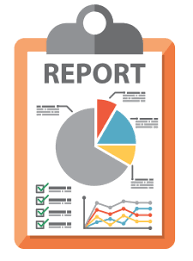
**Figure 13 WeatherBug logo**

* + Those modules will help to fill the database with helpful information that may help to getting hidden relation from the data.
* Modules for notification.



**Figure 14 Notification platform**

* + That module will be integrated with telegram, whatsapp...etc.
* Modules for sending report in daily bases for the owners.



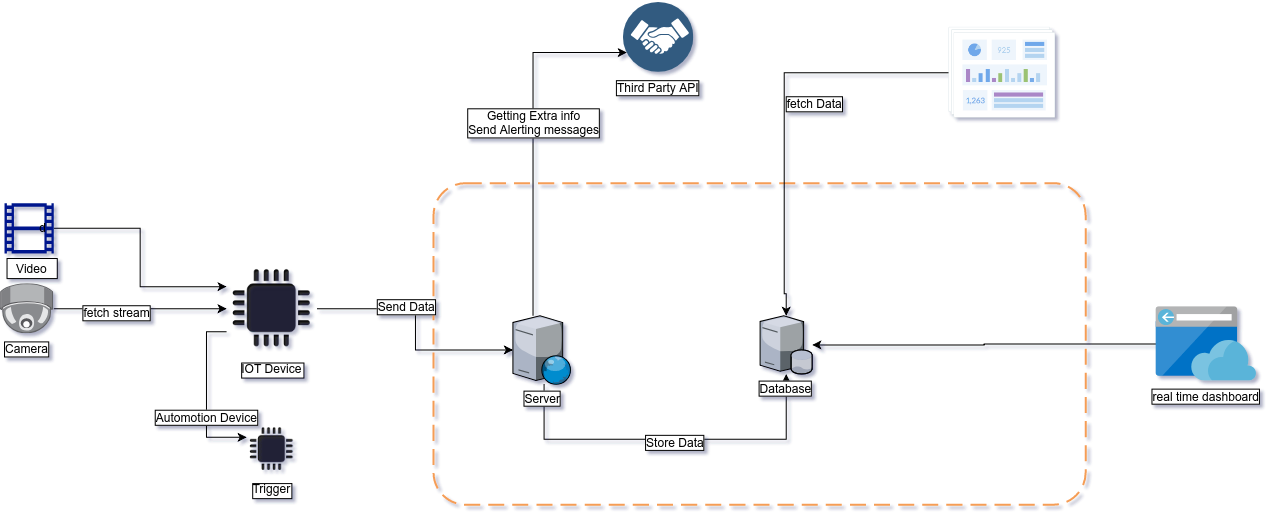
**Figure 15 report example**

* + Those modules will help business owner to get in touch for what happening during day/week in the system.

# 4. Chapter 4: Design, Implementation and testing.

Mainly the project will based in the cloud, below you can find the system infrastructure

Infrastructure



**Figure 16 Infrastructure diagram**

From the above diagram system will be as following:

### 4.0.1. Data Source:

Data stream input for the system is Camera or video that need to analysis.

1. Camera.
2. Video

### 4.0.2. IOT Device:

This part will be responsible to make the needed analysis in the video stream and detected different type of object such as Car, truck, Human.. Etc.

And also this part of the system will be responsible to make the triggers.

Trigger: Is the action system can do based in the predefined action, for example if the system detected a person it will close the door and turn on the alarm.. Etc.

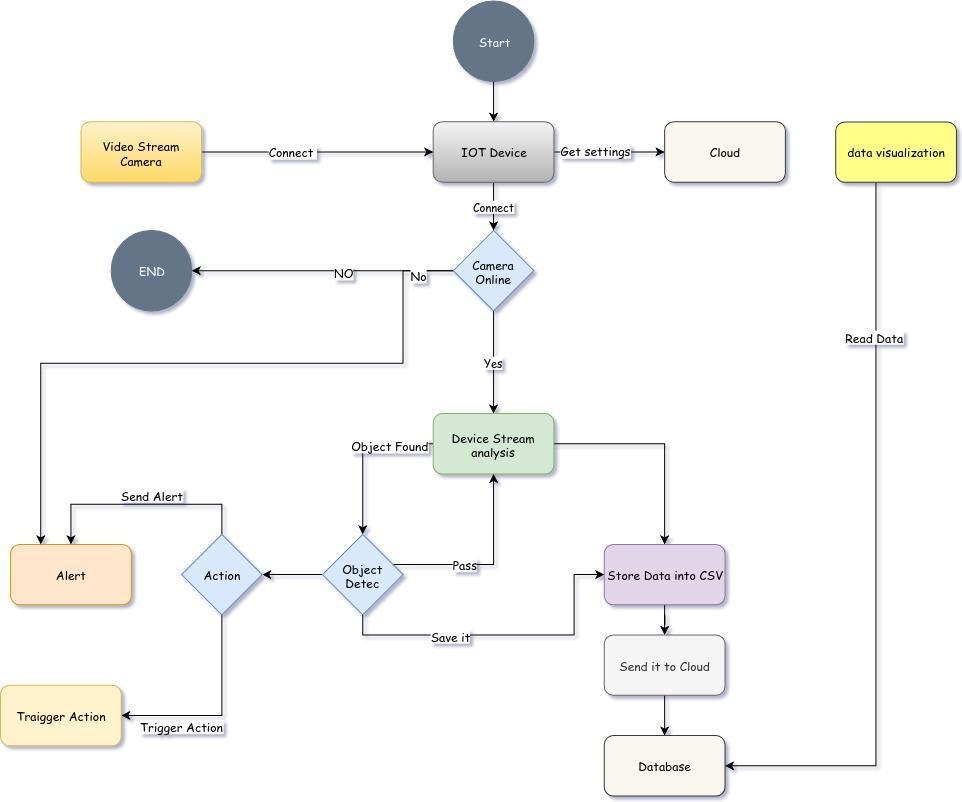
After system detected the object will save it into CSV file and every minute the system will check the file if it contains any data, if yes system will send it to the backend server to store it into the main database.

After backend stored the data into the database, the user can ask questions and learn from data.

The system will use third-party API for alerting and integration with another system.

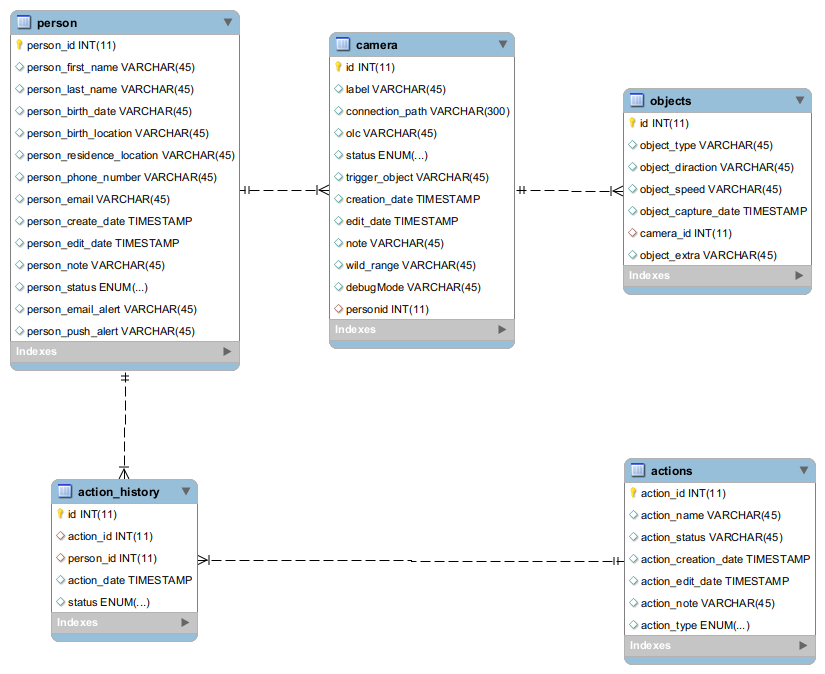
For example of the third party API: if the system detected an object it will send a push notification to the admin.

## 4.3. Activity Diagram:



**Figure 18 Activity Diagram**

## 4.4. Database Diagram:



**Figure 19 Database Diagram**

### 4.4.1. Tables Description:

* Action\_history:
  + This table store all system action that happened during the analysis, for example it’s it will save action email alert if system sent an alert, or if system run a trigger will store it into the database also.
* Actions:
  + This table contains all pre-defined triggers and alert for the system.
* Camera
  + This table contains all camera and video source that system will deal with. For example if you need system use a camera you can set camera IP into the config, or if you have a video need to analysis you can also define it there.
* Objects
  + This table contains all objects that system found during analysis process, system will insert this data after it finish from analysis.
* person
  + This table contains all admin user info such as email or push address that need to alert admin when trigger happened

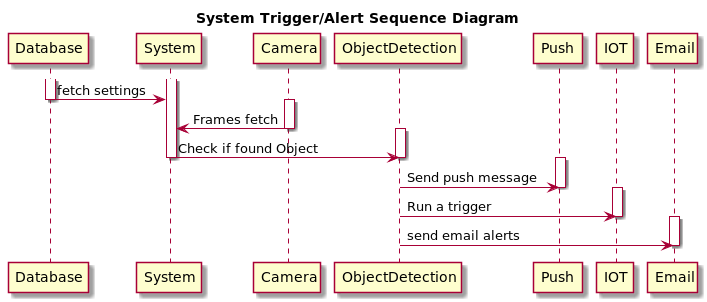
## 4.5. Sequence Diagram:

The below Diagram describe sequence for how the internal components work together:



**Figure 20 internal components sequence diagram**

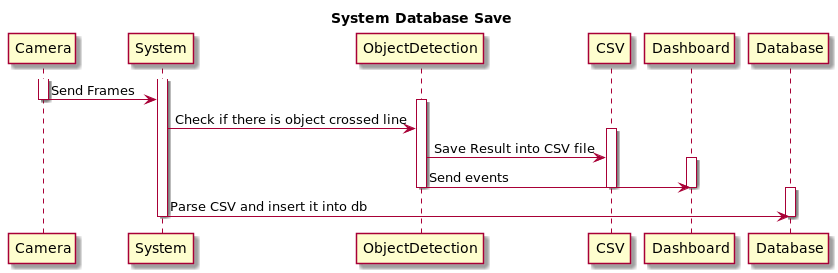
The Below sequence diagram describe how system send alert/push messages to the admin user.



**Figure 21 sequence diagram describe how system send alert/push messages**

The below diagram describe how system save data into CSV,

\*after finish from processing stream the system will save result into the CSV file.



**Figure 22 system save data into CSV**

.

## 4.6. Evaluation method:

Evaluation the system in several ways as below:

### 4.6.1. On device analysis

In this part object detection working as expects, for the evaluation system must detect several types of object mainly the below:

1. Cars.
2. Buses.
3. Human.
4. Bicycle.

So in the CSV file, we must find the detected object stored on it.

### 4.6.2. Database analysis.

In this part, the use must able to view the collected data from the database in the power tool.

These data will be presented as metrics and graphs.

Use must be able to change the matrices based in the filter criteria.

### 4.6.3. Alert trigger.

Testing alert trigger must be in the predefined alert rule that saved into the database by the admin.

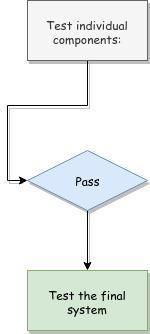
For example, if the system detects a human in the restricted area in the factory it will send an alert to the security people.

### 4.6.4. Automation action trigger.

For example, if the system detects a human in the restricted area in the factory it will run an alerting sound and light to alert the human in the restricted area for the dingers is high in his location.

### 4.6.5. Testing

System will be testing the below diagram:



**Figure 23 System testing diagram**

### 4.6.6. Test individual components:

* Analysis part
  + System will able to detect an object.
* Alert Part
  + System will be able to send an alert after detected X object.
* Trigger Part
  + System will be able to run a specific triger after deteced a X object
* CSV Part
  + System will able to save result into CSV file.
* Data storing
  + Backend will able to receive CSV and parse it and store it into the database
* Data analysis
  + Metabase software will be able to retrieve data from the database.

### 4.6.7. Test the final system

To test the final system, all components must works together during the test.

Test will be as the following:

Put a video stream to IOT device,(Video stream contain all test scenario that we need for the test)

During video analysis system will detect a human after detecting a human system will send a push notification and email to the admin.

And after that system will send a trigger to open the door.

During this time system will send CSV file to the backend server to store it into the database.

After period of time we will be able to view and analysis the data using metabase software.

## 4.7. Testing simulator

I have wrote a module that will help me to test, metabase software and it’s integration with my software.

Mainly that tool will do the following:

Get a random value of the following:

|  |
| --- |
| items = ['bicycle','bus','car','motorcycle','person','truck']  camera = [1,2,3]  object\_diraction = ['up','down']  randomDate("2018-01-01 01:00:00", "2018-12-30 23:30:00", random.random()) |

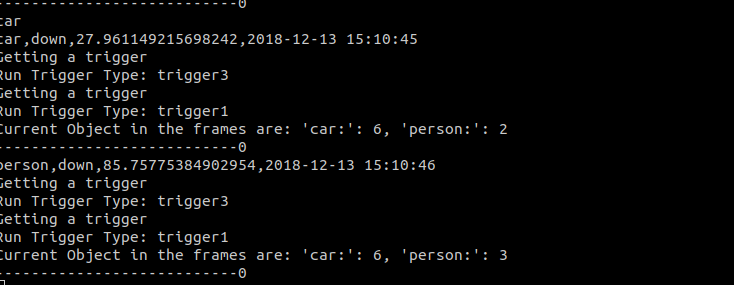
Then insert them into the database:

As below:

|  |
| --- |
| def databaseInsert(host="127.0.0.1", user="admin", passwd="pass", db="aou\_stat"):  mydb = MySQLdb.connect(host=host, user=user, passwd=passwd, db=db)  cursor = mydb.cursor()  csv\_data = range(3000000)  for row in csv\_data:  date= randomDate("2018-01-01 01:00:00", "2018-12-30 23:30:00", random.random())  cID= random.choice(camera)  objects=random.choice(items)  object\_diractions=random.choice(object\_diraction)  speed=random.uniform(10, 150)  # print("Running Insert: {0}".format(row))  queriy="""INSERT INTO objects(object\_type,object\_diraction, object\_speed,object\_capture\_date,camera\_id ) VALUES('%s', '%s', '%s','%s','%s')""" % (objects,object\_diractions,speed,date,cID)  # print(queriy)  cursor.execute(queriy)  mydb.commit()  cursor.close()  databaseInsert() |

### 4.7.1. Screenshot of testing result of the system.

The Below figure show the runtime output of the system.



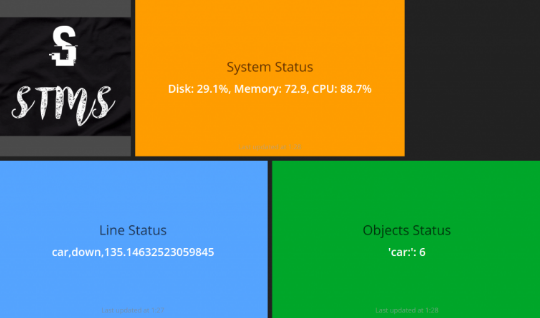
**Figure 24 runtime output of the system**

The below figure show the video feedback output when processing stream.



**Figure 25 video feedback output when processing stream.**

The below figure show the real time dashboard output,



**Figure 26 the real time dashboard output**

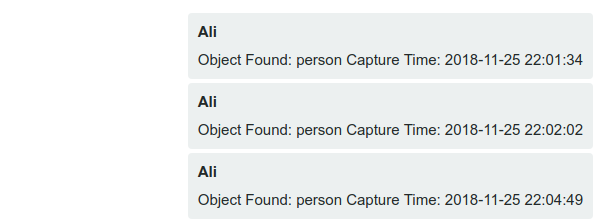
*Figure 23.*

The below figure show the IOT Trigger action, there is multi led each led represent one of the object or line cross.



**Figure 27 IOT Trigger action**

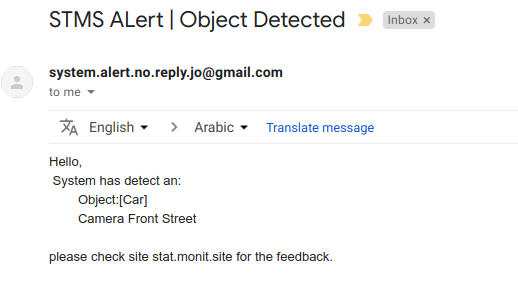
The below figure show the push notification that send to the phone/Browser.



**Figure 28 the push notification that send to the phone/Browser**

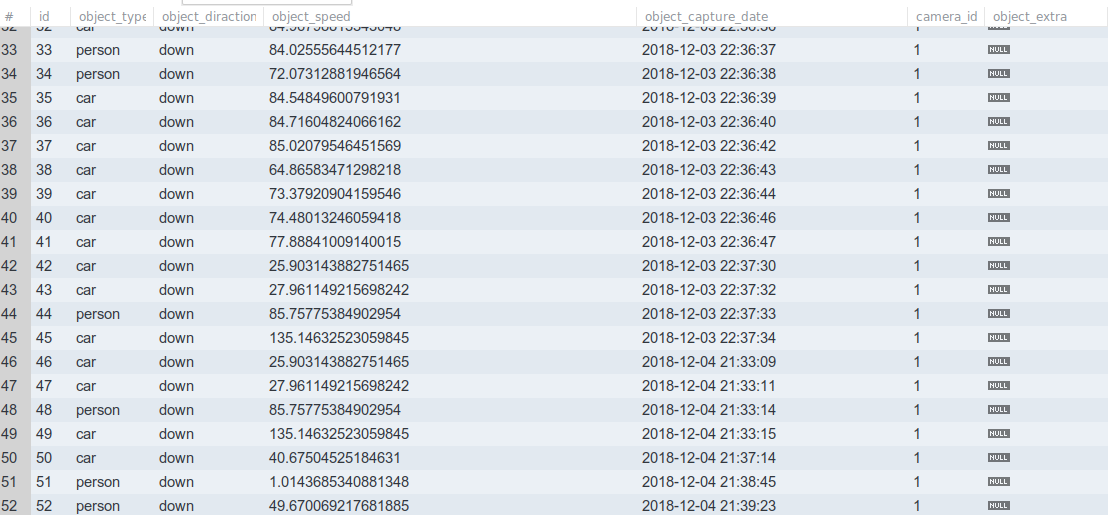
*Figure 25.*

The below figure show the email example when the system detected a specific object which is car in the below example.



**Figure 29 the email example**

The last figure here is the database output when the system finish the processing and insert the data into the database.



**Figure 30 data saved in the database**

Final Test table

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test | Result | Note |
| 1 | Add Video | Success | Use demo video |
| 2 | Add Camera | Success | Use IP camera |
| 3 | Video feedback | Success | Check video processing output |
| 4 | Email alert | Success | Alert email when object found |
| 5 | Realtime dashboard | Success | Realtime dashboard while system is working |
| 6 | IOT Trigger | Success | Led blinking when object found |
| 7 | Push Notification | Success | pushbullet |

**Table 5 Final Test table**

# 5. Chapter 5: Results and discussion.

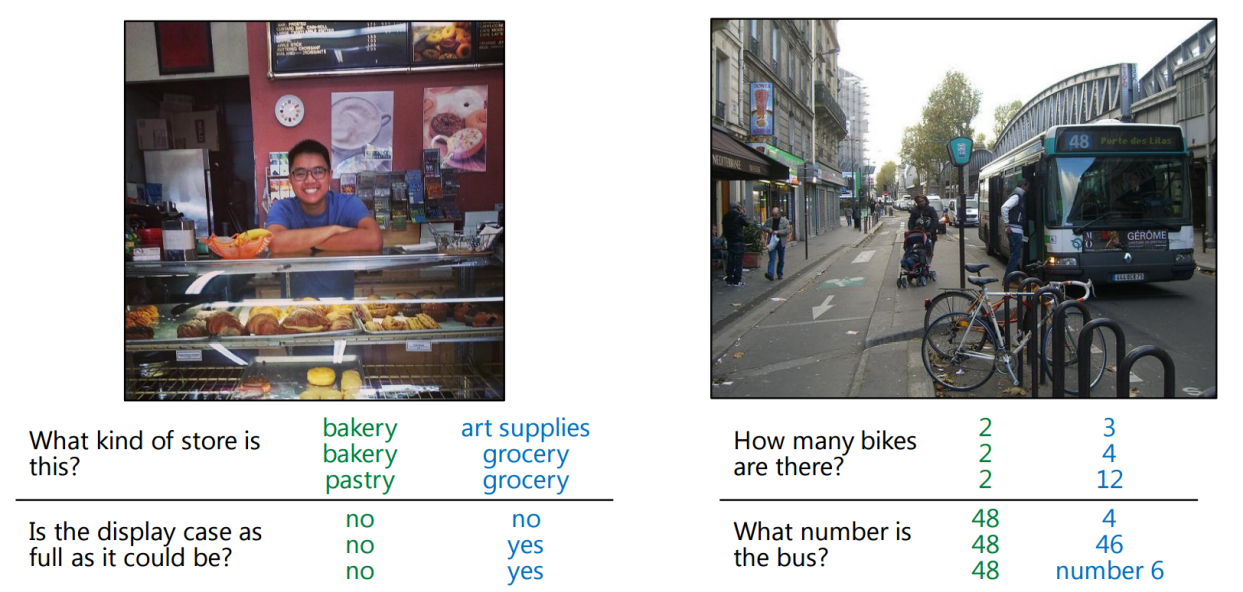
## 5.1. Findings

During working in this project, I found a lot of thing that are interesting of using the machine learning libraries to do a fantastic application in different life aspects.

Below you will find very interesting open source software that use computer vision, and that inspired me:

Question Answering software:

In summary that software can provide answer for your question on the image:



**Figure 31 Question Answering software**

And also something that I found it very hard to implement during working on the project 1 and project 2 in to extract Car plate number, that use case was very hard to implement because it’s need a very good experience on computer vision and also you need a powerful machine to process that case.

Also I found a lot of software alternative of using windows, because I Linux user I find at the beginning hard to use windows software in the Linux machine, but after search I found a lot of tools that can give you the same result of using windows software’s.

## 5.2. Goals achieved

From the introduction above i have success to achieve the below goals during my working in the project.

Some part did not success because of the bad analysis of the problem but it’s not a critical or stopper points that can stop go further in the project.

And later on, I will invest time to make project a real product.

|  |  |
| --- | --- |
| **Goal** | **Achievement status** |
| Provide Real Time alerting for new predefined settings. | Success |
| Statistical data summary will be generated from the data store for decision makers review. | Success |
| Ability to handle and use anytype of IP cameras (no specific hardware required) | Success |
| Software that will running at the device does not require a high expensive hardware. | Failed |
| System can analysis CCTV camera video input for any time of range. | Success |
| Detect vehicle speed statistical data such as AVG/Max speed. | Success |
| Provide historical data about the traffic example: counter of car or buses. | Success |
| Categories object types of traffic. | Success |
| Send real-time alert when detecting special object time. | Success |
| Do some action when detect special object time, for example, turn on the light., close the door. | Success |

**Table 6 Achievement Goals**

## 5.3. Further work

I’m looking forward with this project to make it a fantastic product that help the community in different aspect.

So below you will find the goal that I’m looking for:

1. Make this software an open source software on github to give the information technology community to contribute easily in this project to make it better product.
2. Add different modules so the project can serve many people for their need of using this project, for example integrated with IFTTT as third party API service.
3. and modules for raspberry, arduino so the user can make its own automation system based in this project.
4. Add automated report system based on the database records to summary the data and do the needed cleansing in the report.
5. Make system able to create users for each account with their own permission.
6. Add enhancement in the performance and efficiency of the system.
7. Add object tracking feature for more accurate result.
8. Make the system able to do a very smart decision based in the given data.

So far I can see that system will able to do a lot of stuff for many people and business to serve their needs and their passion of using the IT.

# 6. Chapter 6 Conclusions

This project is very useful for a different aspect of usages, as mentioned before you can use it in any field for example security people or for enhancement the business understanding of the clients.

Also, this project builds in the way to give you the ability of customizing it for personal use also such as home automation system... etc.

In the next time, I will make this project open source software to let the other people contribute to this project to make better and more efficient, below you can find the Strength and the Weakness

As my experience in my work place that help me a lot to support my project such as Linux system, databases...etc. and that is my Strength point in the project.

But one of the big weakness that I did not have the good knowledge of the project management as process and differences.

But so far I had learnt a lot of thing and skills that will help me to go further in my life and career.

At the end I would like to Thank Dr Khalid Al Tahat for his wise recommendation during building this project.

Also I had made a marketing site for the project:

URL: monit.site

# 7. References:

* Psd.gov.jo. (2018). Traffic conditions in Jordan Procedures for managing traffic in the field of traffic safety. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/010.pdf [Accessed 9 Mar. 2018].
* Central Traffic Department. (2015). The role of central traffic management in the prevention of road accidents. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/020.pdf [Accessed 9 May 2018].
* Central Traffic Department. (2016). Problems and solutions of the transport sector in Jordan. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/030.pdf [Accessed 9 May 2018].
* General department of traffic. (2018). What is Saher?. [online] Available at: https://www.moi.gov.sa/wps/portal/Home/sectors/publicsecurity/traffic/contents/!ut/p/z0/04\_Sj9CPykssy0xPLMnMz0vMAfIjo8ziDTxNTDwMTYy83V0CTQ0cA71d\_T1djI0MXA30g1Pz9L30o\_ArApqSmVVYGOWoH5Wcn1eSWlGiH1FSlJiWlpmsagBlKCQWqRrkJmbmqRoUJ2akFukXZLuHAwCkY5qs/ [Accessed 9 May 2018].
* Husseini, R. (2018). New point system for traffic violations will help curb accidents — experts. [online] The Jordan Times. Available at: http://www.jordantimes.com/news/local/new-point-system-traffic-violations-will-help-curb-accidents-%E2%80%94-experts [Accessed 9 May 2018].
* logipix.com. (2018). LOGIPIX TRAFFIC VIOLATION MANAGEMENT SOLUTION. [online] Available at: http://www.logipix.com/index.php/traffic-video-surveillance-system [Accessed 9 May 2018].
* moi.gov.sa. (2018). Saher. [online] Available at: https://www.moi.gov.sa/wps/portal/Home/sectors/publicsecurity/traffic/contents/!ut/p/z0/04\_Sj9CPykssy0xPLMnMz0vMAfIjo8ziDTxNTDwMTYy83V0CTQ0cA71d\_T1djI0MXA30vfSj8CsAmpCZVVgY5agflZyfV5JaUaIfUVKUmJaWmaxqAGfkJmbmqRqkJRYWgwTT4kGs-OLEjNSi-NQ8\_YLsqEgApzCz9w!! [Accessed 9 May 2018].
* psd.gov.jo. (2012). Automated control over driver behavior. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/050.pdf [Accessed 9 May 2018].
* psd.gov.jo. (2016). Annual Report of Traffic Accidents in Jordan. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/TRaff2016.pdf [Accessed 9 May 2018].
* psd.gov.jo. (2017). Summary of studies and work papers completed by the Jordan Traffic Institute. [online] Available at: https://www.psd.gov.jo/images/jti/docs/20171.pdf [Accessed 9 May 2018].
* Psd.gov.jo. (2018). Traffic conditions in Jordan Procedures for managing traffic in the field of traffic safety. [online] Available at: https://www.psd.gov.jo/images/traffic/docs/010.pdf [Accessed 9 Mar. 2018].
* News for Public - All News Which You want to Read. 2018. Essential Clients' Guide to Agile Development Methodology. [ONLINE] Available at: <https://www.newsforpublic.com/agile-development-methodology/>. [Accessed 13 December 2018].
* The Official 360logica Blog. 2018. Agile Development – Advantages, Disadvantages and when to use it? - The Official 360logica Blog. [ONLINE] Available at: <https://www.360logica.com/blog/agile-development-advantages-disadvantages-and-when-to-use-it/>. [Accessed 13 December 2018].
* Importance Of Statistics In Business Organization |EssayCorp. 2018. Importance Of Statistics In Business Organization |EssayCorp. [ONLINE] Available at: <https://blog.essaycorp.com/importance-of-statistics-in-business-organization-essaycorp/>. [Accessed 13 December 2018].
* Government Statistics | Encyclopedia.com. 2018. Government Statistics | Encyclopedia.com. [ONLINE] Available at: https://www.encyclopedia.com/social-sciences/applied-and-social-sciences-magazines/government-statistics. [Accessed 13 December 2018].
* www.ncbi.nlm.nih.gov. 2018. Innovations in Federal Statistics: Combining Data Sources While Protecting Privacy. [ONLINE] Available at: https://www.ncbi.nlm.nih.gov/books/NBK425873/. [Accessed 13 December 2018].

# 8. Appendix A:

11.1. Why Python:

* Python is a general purpose programming language created in the late 1980s, and named after Monty Python, that's used by thousands of people to do things from testing microchips at Intel, to powering Instagram, to building video games with the PyGame library, also powering image processing such in our project.

11.2. Why linux in on field device

* TinkerOS has been carefully designed to be extremely lightweight and responsive. Running on top of the base Debian 9 is a the LXDE desktop environment. This GUI is optimized specifically for SBC boards. It also features plug & play NTFS support allowing for easy access to Windows based flash drives and external hard drives. The included web browser has also been carefully selected and optimized. It based on Chromium allowing for speed and stability along with a number of extensions. The ASUS team has help to enable hardware acceleration of the browser allowing for improved web rendering and video playback including HD resolutions in YouTube.

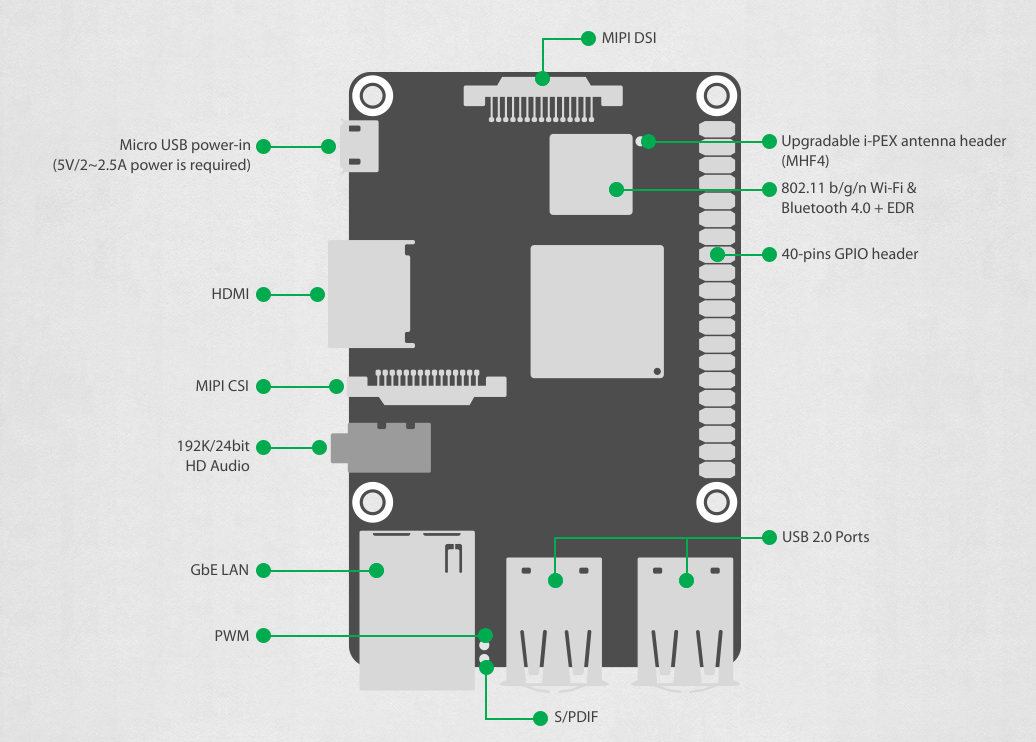
11.3. Distinctive Features Of SQLite

SQLite does not need to be "installed" before it is used. There is no "setup" procedure. There is no server process that needs to be started, stopped, or configured. There is no need for an administrator to create a new database instance or assign access permissions to users. SQLite uses no configuration files. Nothing needs to be done to tell the system that SQLite is running. No actions are required to recover after a system crash or power failure. There is nothing to troubleshoot.

11.4. Why OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

11.5. Why Tinker board



**Figure 32 Tinker board**

*Figure 28.*

Tinker Board is a Single Board Computer (SBC) in an ultra-small form factor that offers class-leading performance while leveraging outstanding mechanical compatibility. The Tinker Board offers makers, IoT enthusiasts, hobbyists, PC DIY enthusiasts and others a reliable and extremely capable platform for building and tinkering their ideas into reality.

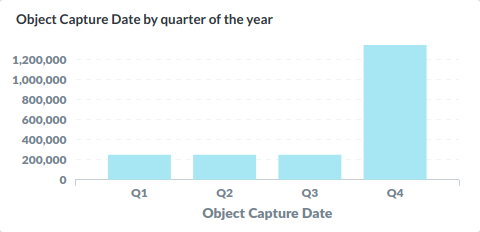
# 9. Metabase

Metabase is the easy, open source way for everyone in your company to ask questions and learn from data.

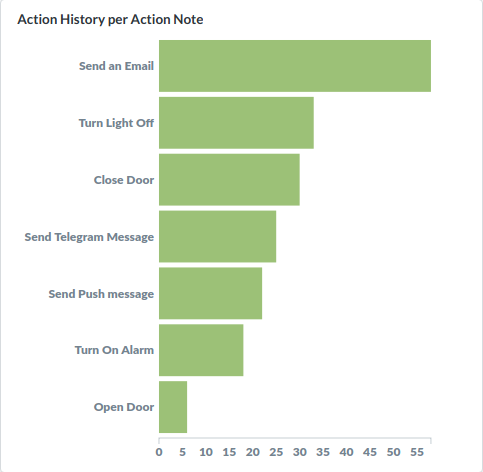
Metabase let you to do the following action in the dataset:

* Explore on your own Easily filter and group your data to find just what you’re looking for, all without ever writing a line of sql or having to wait on a co-worker.
* See connections It just takes a click to see individual records and explore connections between your data, so you can move from who, to what effortlessly.
* Visualize results Move from your data to beautiful graphs and charts with just a few clicks.
* SQL when you need it When you need to dig into the complicated stuff, Metabase provides an elegant SQL interface for people who need a little more power.
* Easily and securely add analytics to your application

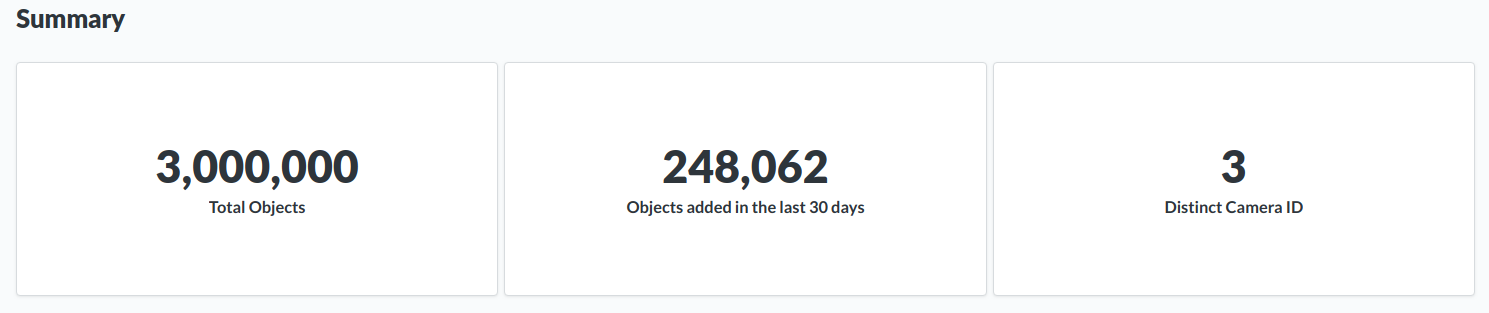
Screenshot if using metabase.



**Figure 33 metabase example**



**Figure 34 metabase example**



**Figure 35 metabase example**



**Figure 36 metabase example**