DELL Technologies Envision the Future Competition 2019-20
Interim Progress Report Structure

Section 1: Refined Project Description:

1.1 The problem addressed

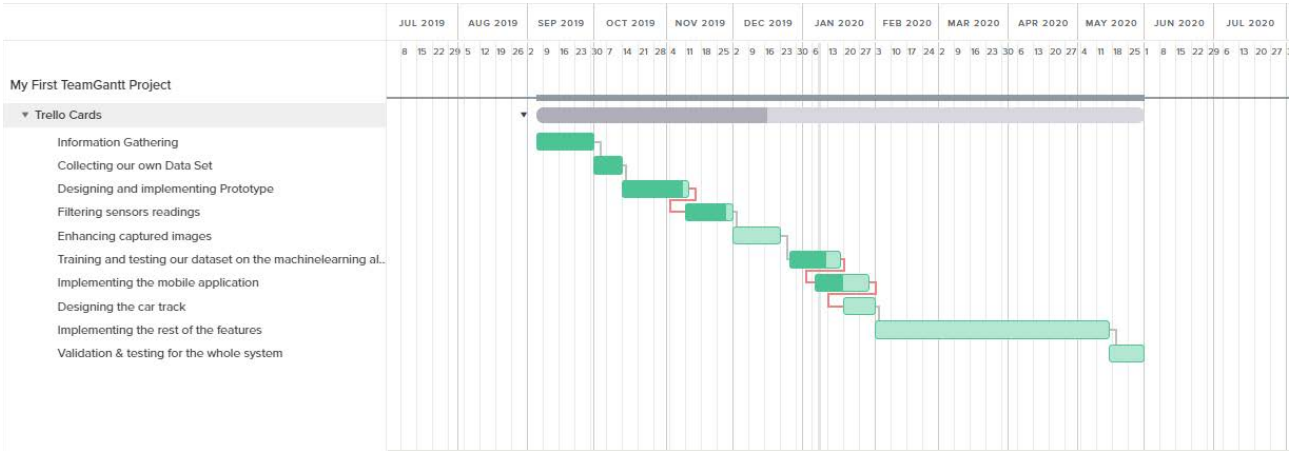
Autonomous cars are not legal in many countries nowadays because they are not safe enough. Although most road accidents are caused by a human error but according to our project, we are improving new methods in implementing a Self-Driving car which is more accurate and purely analytical to decide the right decision. A self-driving car is not only about maintaining safety for humans, but it is also beneficial in maintaining the car safety.

1.2 Project scope and expected outcome

Different algorithms and mechanisms are used to help the self-driving car to make decisions safely including Stereo-Vision depth map calculations, accurate anomalies detection algorithm using SVM to classify the readings from the sensors then use CNN to classify the images of the lanes. The system is responsible for collecting the data, applying algorithms then deciding the action to be applied. The system is also developed to be a mobile application that receives the live updates of the actions and alert the driver before hitting any road anomalies detected.

Section 2: Refined Project Plan:

2.1. Detailed schedule and milestones (clearly show current status, progress, and issues)



2.2. Team structure and detailed roles/responsibilities of each member

•Team Member 1: Classification process of the road lanes and anomalies.
•Team Member 2: Mobile Application development & Mobile user interface.
•Team Member 3: Mobile Application development & image enhancement process.
•Team Member 4: Car hardware connections and development and Image enhancements process.

2.3.1 Risk mitigation plan

- As we are working on a small car prototype, using GPS might not be as accurate as we want to save the anomalies locations.
- Batteries that are responsible for powering up the car system (Raspberry Pi and Arduino) might get empty.

2.3.2 Contingency

- We will use accelerometer instead of GPS to know the current location of the car in this prototype.
- We will use rechargeable batteries as it won't cost as much as the normal batteries do.

Section 3: System Requirements:

3.1. Requirements Elicitation Process

A) Requirements elicitation and analysis process:

We have gathered our system requirements from different sources. We have read lots of research papers ,reports and topics that helped us to come up with the new system idea.

B) Stakeholders: Misr International University**.**

Users: Car Driver.

Clients: Valeo

C) Challenges:

- Stereo-vision is not common in our region nowadays as it was a challenge for us to deal with a hardware project because we are not familiar with hardware connections.
- Currently, we are creating our own datasets because we didn't find similar data sets that fit our project requirements.

Lessons Learned:

We are computer science students, we only deal with software and code. So, we were able to learn hardware connections during the development of our project.

3.2 System Requirements List

A)Functional Requirements

-Road Lane Classification Function
-Anomalies Classification Function
-Start Self Driving Function
-Move to Right Lane Function
-Measure Depth Map Function
-Stop Car Function
-Filter Sensors Reading Function
-Move Car Forward Function
-Android Application Retrieve Distance
-Android Application View Distance
-Region of Interest To Draw Lines
-Road Lane Image Enhancement Function
-Get Readings from Sensors
-Get Video Frame Function
-Get Car Current Location from Sensors
-Slow Car Down Function
-Set Destination Function
-Upload Current car Location Coordinates Function
-Create & Delete Anomalies to Upload Query
-Alert Driver Function
-Move to LeftLane Function
-Draw lines on lane Function

B) Non-Functional Requirements

-Performance and speed: high speed in processing video frames and receiving the readings.

-Reliability: Any driver can rely on our autonomous car as the sensors reading will be very clear and accurate.

-Safety: Make the driving as safe as possible and it is useful for disabled people who can't drive for themselves so our system will help them go wherever they want.

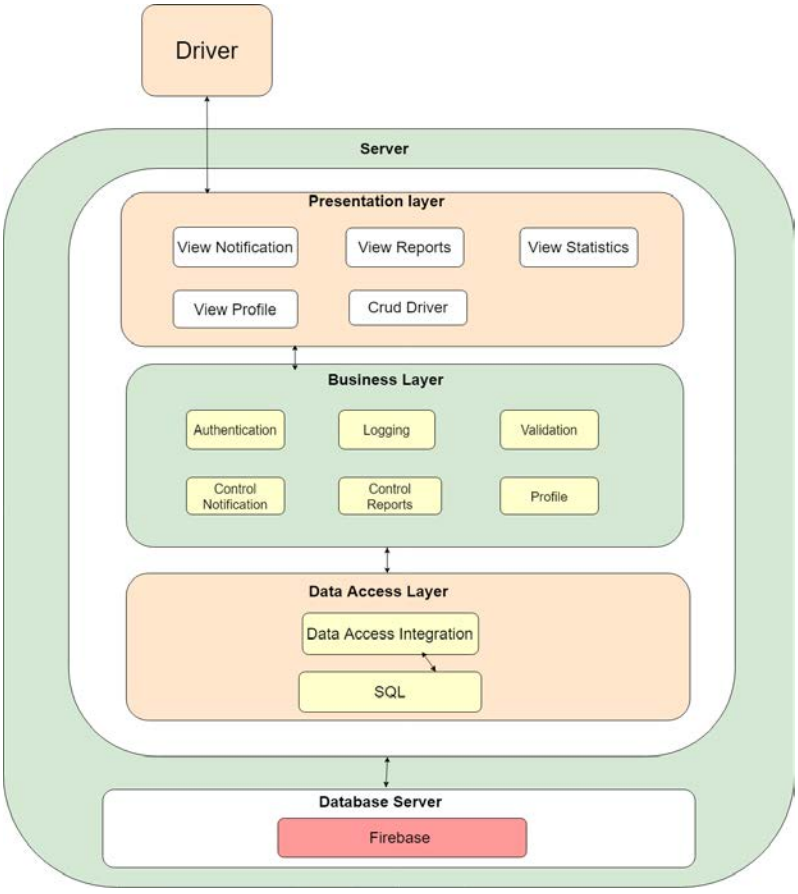
-Re-usability: As in our car there are pre-processing for images, so anyone who is interested in embedded systems can re-use our pre-processing component.

-Extensibility: Our car will be integrated with many web services like google maps api to update the positions of any anomalies.

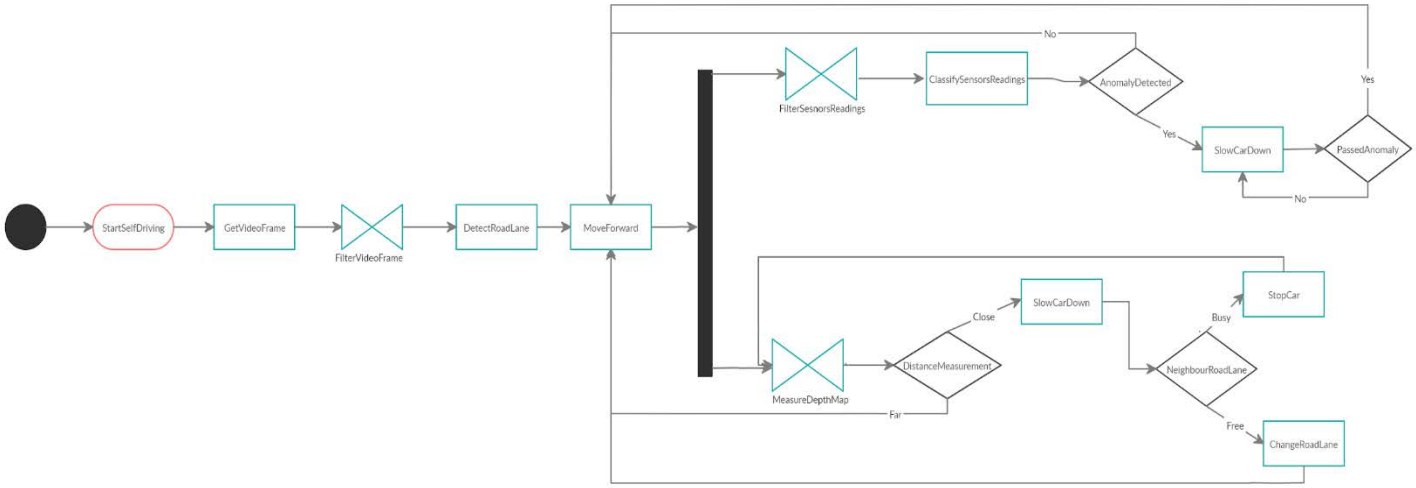
Section 4: System Design

4.1 High level system architecture, data flows, etc

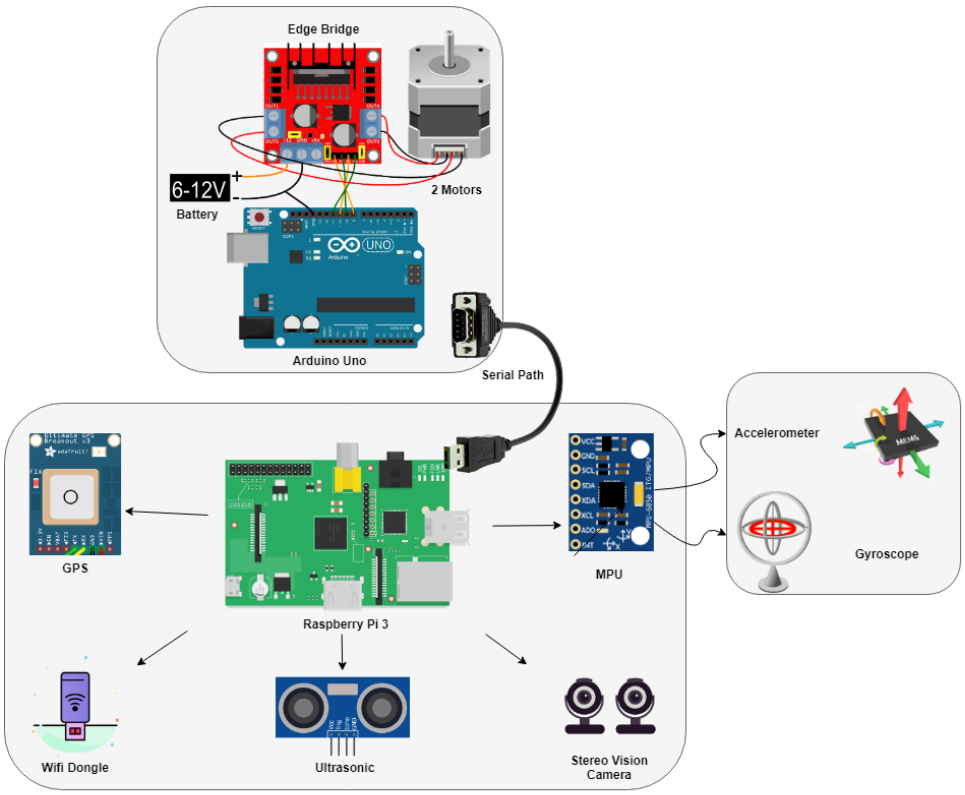
Software Architecture Diagram:



Activity Diagram:

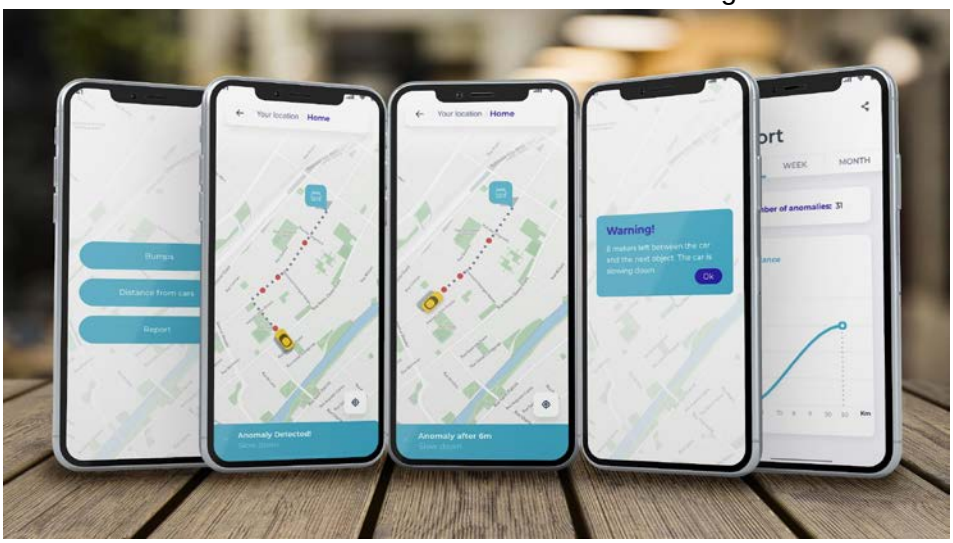


Hardware Architecture:



4.2. User interfaces

Our system user interface is considered to be an android application which is very usable and clear to any user who can use smartphones. The system retrieves the anomalies state from the car, then alerts the user when detecting any bump or holes, also the system alerts the user before hitting any saved road anomaly and finally notify the user before slowing down or any sudden action so that the user will be relaxed. Furthermore, there is a report screen per day, week and month to show the number of anomalies which have been detected and a graph for the distance. The detailed user interface shown in the figure below.



4.3. Algorithmic components

CNN: In our self-driving car CNN algorithm is used to classify the images of the road Lanes.

SVM: SVM is used to classify the road anomalies.

Disparity map: Stereo-vision is used to help the car to measure the disparity map to calculate distance between the vehicles and obstacles in front of it.

4.4. Innovative aspects of the design

What's new and innovative in our self-driving car is that it's an integration between self-driving car and anomalies detection, furthermore, this car is different from the rest of the self-driving cars as it's supplied with stereo vision that will help the car to know the depth map between the vehicles in front of it and this feature will be a good addition in the field of autonomous cars and also will help autonomous vehicles to achieve more safety. These features will make the car take faster and accurate decisions.

Section 5: System Implementation:

5.1 hardware and software platforms.

5.1.1 Hardware platforms:

- Raspberry pi 3.
- Arduino UNO.
- Android Mobile.

5.1.2 Software platforms:

- PyCharm
- Android Studio
- Arduino IDE

5.2 Hardware and software development tools and languages

- Android Development tool:
  The process of developing new applications on android mobiles. Usually developed Java programming language using Android.Software Development Kit. We used it to develop the android application that the driver will use to receive notifications about the car's current state and statistics of the drive.
- Python & Java:
  We are using Arduino IDE that supports Java programming language mainly to help us develop the Arduino which controls the car motors and car directions based on the received signals from the raspberry pi. Also, we are using python programming language to implement car classification algorithms like CNN and SVM, also to implement the disparity map algorithm on the raspberry pi.

5.3 Modules/components acquired from external sources:

We have bought a stereovision based raspberry PI so we can use it for the disparity map and for the road lane classification.

5.4 Innovative aspects of the implementation:

What's new and innovative in the implementation process is that we are implementing the disparity map method using stereo vision cameras to calculate the distance of the obstacles Infront of the car which will help the car achieve higher safety taking most of the decisions. Also, we have combined disparity map, anomalies classification with road lanes classification in a single self-driving car in our program in order to cover most of the situations that might be a danger to the car. Also, we our implementation is mostly based on Object Oriented programming and software design patterns which will make the implementation process easier.