EMERGENCY MEDICAL SYSTEM

Submitted by

Vishnu M	**[20AD039]**
Naveen V	**[20AD020]**
Farhan Ahmed Khan	**[20AD008]**
Dhaanus K	**[20AD007]**

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

SRI ESHWAR COLLEGE OF ENGINEERING

(AN AUTONOMOUS INSTITUTION)

COIMBATORE – 641 202

JUNE 2022

BONAFIDE CERTIFICATE

Certified that this project report **"Emergency Medical System"** is the bonafide work of

Vishnu M	**[20AD039]**
Naveen V	**[20AD020]**
Farhan Ahmed Khan	**[20AD008]**
Dhaanus K	**[20AD007]**

who carried out the project work under my supervision

……………………………………

SIGNATURE

Dr. R. SUBHA M.E., Ph. D.,

Professor & Head,

Dept. of Computer Science & Engineering,

Sri Eshwar College of Engineering,

Coimbatore-641 202.

……………………………………

SIGNATURE

Mr.M.Lakshmi, M.E.,

SUPERVISOR

Assistant Professor,

Dept. of Computer Science & Engineering,

Sri Eshwar College of Engineering,

Coimbatore-641202.

Submitted for the End Semester practical examination – Mini project work viva-voce held on ………………………

………………………

(Internal Examiner)

………………………

(External Examiner)

TABLE OF CONTENTS

CHAPTER NO.	Tittle	PAGE NO.
	ABSTRACT	**1**
1	**INTRODUCTION**	**3**
2	**SYSTEM ANALYSIS AND DESIGN**	**5**
	2.1SOFTWAREDEVELOPMENTLIFE CYCLE	**5**
	2.1.1 REQUIREMENT ANALYSIS	**5**
	2.1.2 ANALYSIS PHASE	**5**
3	**PROPOSED SOLUTION**	**7**
	3.1 SOLUTION	7
	3.2 OVERVIEW	
	3.3 BLOCK DIAGRAM	7
4	**SYSTEM SPECIFICATION**	10
	4.1 HARDWARE SPECIFICATION	10
	4.2 SYSTEM REQUIREMENTS	
	4.3 SOFTWARE SPECIFICATION	10
5	**PROJECT DESCRIPTION**	12
	5.1 METHODOLOGY	12
	5.1.1 Machine Learning 5.1.2 Automation	12
	5.2 IMPLEMENTATION	18
6	**CONCLUSION & FUTURE SCOPE**	**22**

ABSTRACT

We all know that automobile industries have achieved their success in creating autonomous driving systems and we can see them exist in our daily life.

Have you ever wondered how the safety features would be? So what will happen if your health goes down to critical condition while travelling in an autonomous vehicle?

You'll be moving to the fixed destination with the critical condition you are facing. This problem can be solved by our method of approaching the system on how autonomous cars work. In the existing scenario autonomous cars are considered to be the safest way of transportation. But it has its own limitations.

Despite all the hype, fully self-driving cars are not yet commercially available to the public. In limited geographical areas, individuals can sometimes be passengers in these vehicles.

However, mastering the technology behind these cars has proven highly costly, and a wider rollout has taken much longer than expected.

CHAPTER 1

INTRODUCTION

An autonomous car is a vehicle capable of sensing its environment and operating without human involvement. A human passenger is not required to take control of the vehicle at any time, nor is a human passenger required to be present in the vehicle at all. An autonomous car can go anywhere a traditional car goes and do everything that an experienced human driver does.

The Society of Automotive Engineers (SAE) currently defines 6 level of driving automation ranging from Level 0 (fully manual) to Level 5 (fully autonomous). These levels have been adopted by the U.S. Department of Transportation.
Fully autonomous vehicles will be available on the mass market by the end of the decade – but, for now, barriers to widespread adoption remain. The potential for connected vehicles during the interim, however, is an exciting, multi-faceted and high-growth area in the IoT's development – enabling enhanced road safety, smart traffic management, advanced navigation assistance, passenger entertainment and much more. Our free new white paper – 'From here to autonomy: How to fulfil the requirements of the next generation connected car' – sets out the promise of connected vehicles, assesses the technical and commercial challenges they face, and considers the road ahead. Read on for details on where the opportunities in connected driving lie, which forms of connectivity are set to enable the next generation of automotive services, and how Quectel can support those planning a move into this space.

CHAPTER 2

SYSTEM ANALYSIS AND DESIGN

2.1 SOFTWARE DEVELOPMENT LIFECYCLE

2.1.1 REQUIREMENT ANALYSIS

In the requirement analysis phase we will be acquiring the owner's health report, any heart related diseases, preferred hospital, well connected internet access and nearby hospitals database.

2.1.2 ANALYSIS PHASE

Analysis phase involves analysis of operational feasibility, technical feasibility and economic feasibility.

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

This problem is worth solving because the future of transportation is going to be based on autonomous vehicles.

Technical feasibility is the process of figuring out how you're going to produce your product or service to determine whether it's possible for your company.

This project is practically possible and much needed for the society when we are aiming for the future.

Economic feasibility step of business development is that period during which a break-even financial model of the business venture is developed based on all costs associated with taking the product from idea to market and achieving sales sufficient to satisfy debt or investment requirements.

This project is being redefined to make sure that it is very cost efficient.

CHAPTER 3

PROPOSED SOLUTION

The vehicle will be provided with an application where the client saves his personal data, like his name, address, his past medical records etc in that given application.

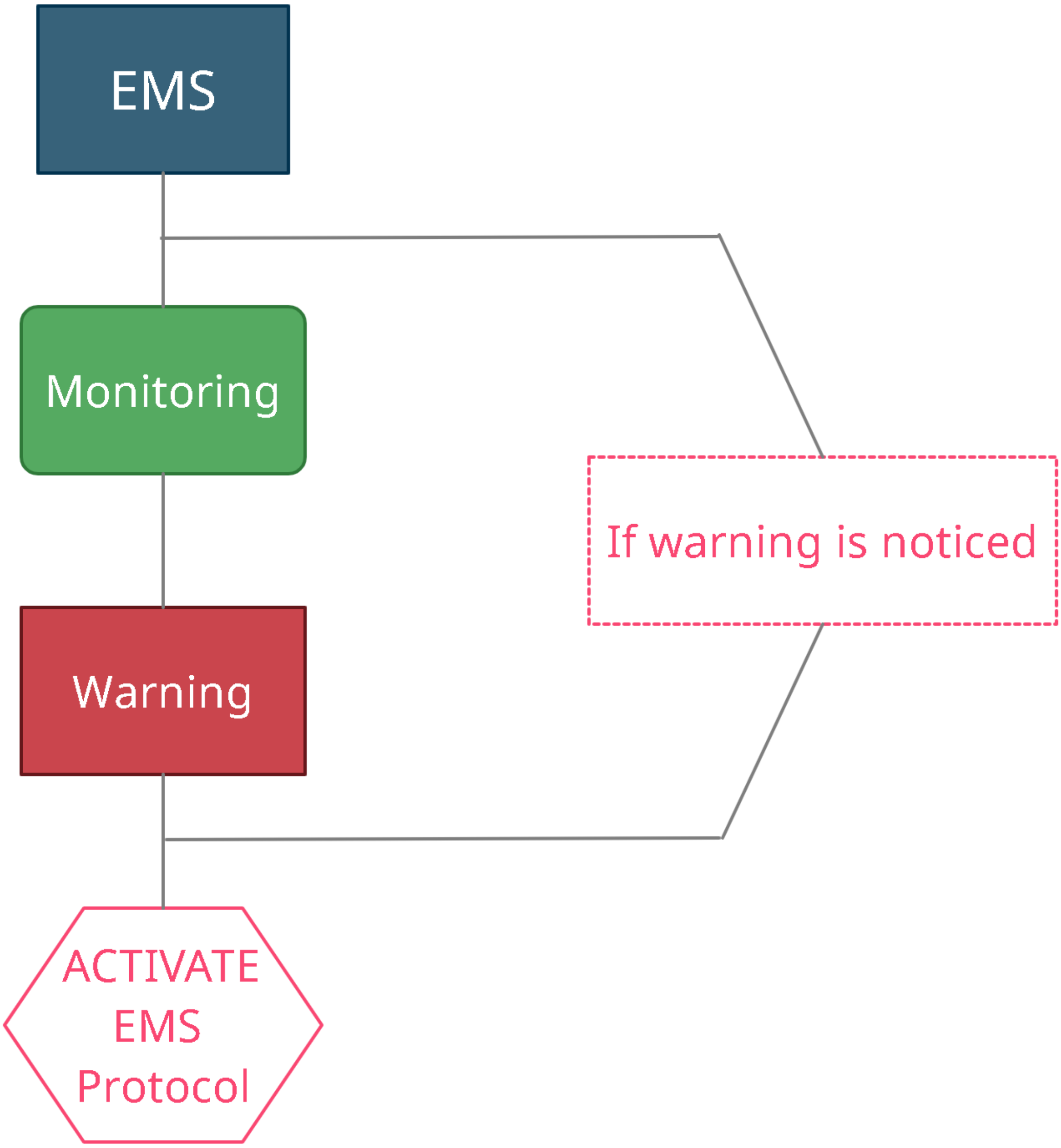
When the owner/client gets into some critical situations, the emergency mode will be turned on.

After this, the AI in the vehicle will search for the nearest hospital based on the client’s medical records & detect the problem or the regular hospital the client goes to frequently, if the regular clinic is far away, the AI will search for the best hospitals nearby and drive him to the hospital.

And side by side the AI will make a call or will send information to the respective hospital about the client and his condition and give the details of the client to the respective hospital.

By this action the hospital staff will make precautions until the client reaches the hospital.And the information will be sent to the client’s family

BLOCK DIAGRAM



CHAPTER 4

SYSTEM SPECIFICATION

4.1 Hardware Specification

This project is mainly based on the machine learning concepts and programming, therefore the hardware part is much less than the core part.

So, the hardware's is used in order to obtain certain basic human functions such as heartbeat and pulse .

The hardware's which have used are as follows:

- Pulse sensor
- Heart rate sensor
- Raspberry pi
- Arduino

4.2 System Requirements

RAM: 4 GB

PROCESSOR: Intel(R) Dual Core CPU @ 2.5GHz

4.3 Software Requirements

OPERATING SYSTEM: Windows, Linux, MacOS

PROGRAMMING LANGUAGE: Python,

TOOLS & DEPENDENCIES: Web browser(Chrome)

CHAPTER 5

PROJECT DESCRIPTION

5.1 Methodology

5.1.1 Machine learning

In this project CNN algorithm is used to analyse the user’s condition.

There are some stages where we use face recognition in this project for example there is a drowsiness detection and motion detection part.

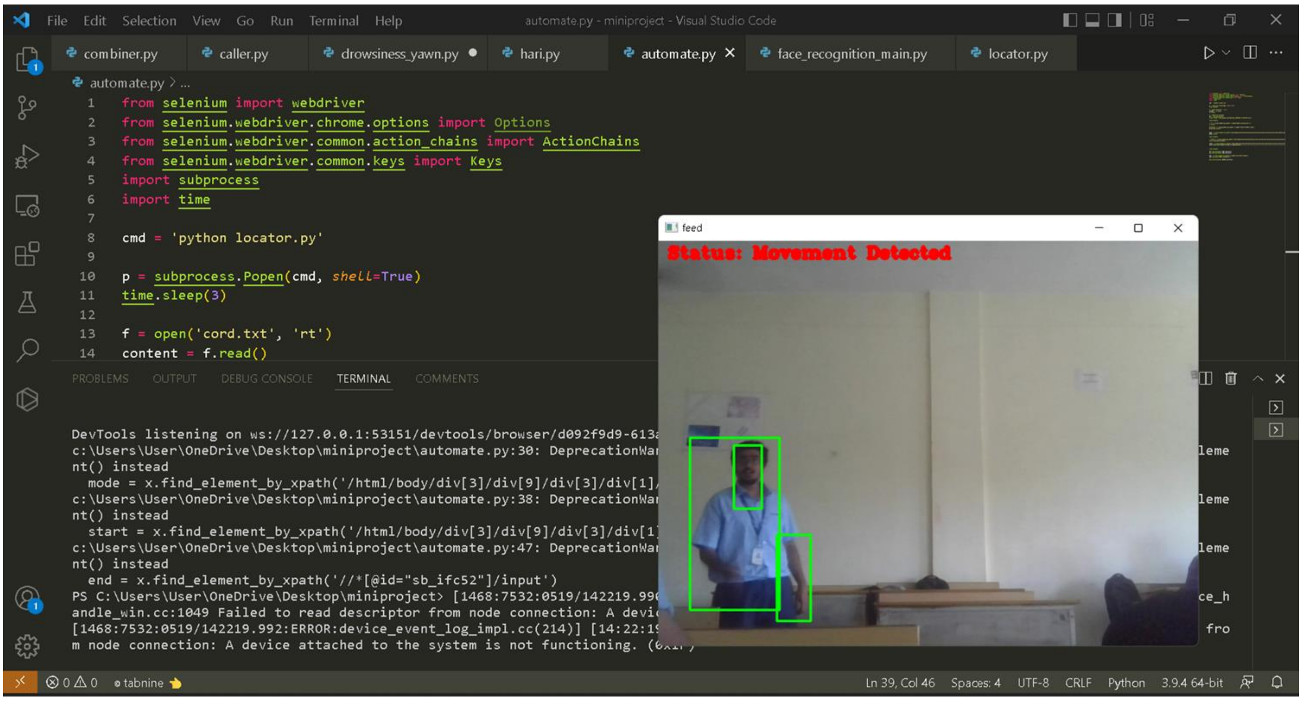
So, these are levels where the machine learning concepts are applied.

Code part for Motion Detection:

```
cap = cv2.VideoCapture(0)

frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))

frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

fourcc = cv2.VideoWriter_fourcc('X', 'V', 'I', 'D')

out = cv2.VideoWriter("output.avi", fourcc, 5.0, (1280, 720))

ret, frame1 = cap.read()

ret, frame2 = cap.read()

'''print(frame1.shape)'''

while cap.isOpened():

    diff = cv2.absdiff(frame1, frame2)

    gray = cv2.cvtColor(diff, cv2.COLOR_BGR2GRAY)

    blur = cv2.GaussianBlur(gray, (5, 5), 0)

    _, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH_BINARY)

    dilated = cv2.dilate(thresh, None, iterations=3)

    contours, _ = cv2.findContours(dilated, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
```

```
    for contour in contours:

        (x, y, w, h) = cv2.boundingRect(contour)

        if cv2.contourArea(contour) < 900:

            continue

        cv2.rectangle(frame1, (x, y), (x + w, y + h), (0, 255, 0), 2)

        cv2.putText(frame1, "Status: {}".format('Movement Detected'),
(10, 20), cv2.FONT_HERSHEY_COMPLEX_SMALL,

                    1, (0, 0, 255), 3

    image = cv2.resize(frame1, (1280, 720))

    out.write(image)

    cv2.imshow("feed", frame1)

    frame1 = frame2

    ret, frame2 = cap.read()

    if cv2.waitKey(40) == 27:

        break
```

output:

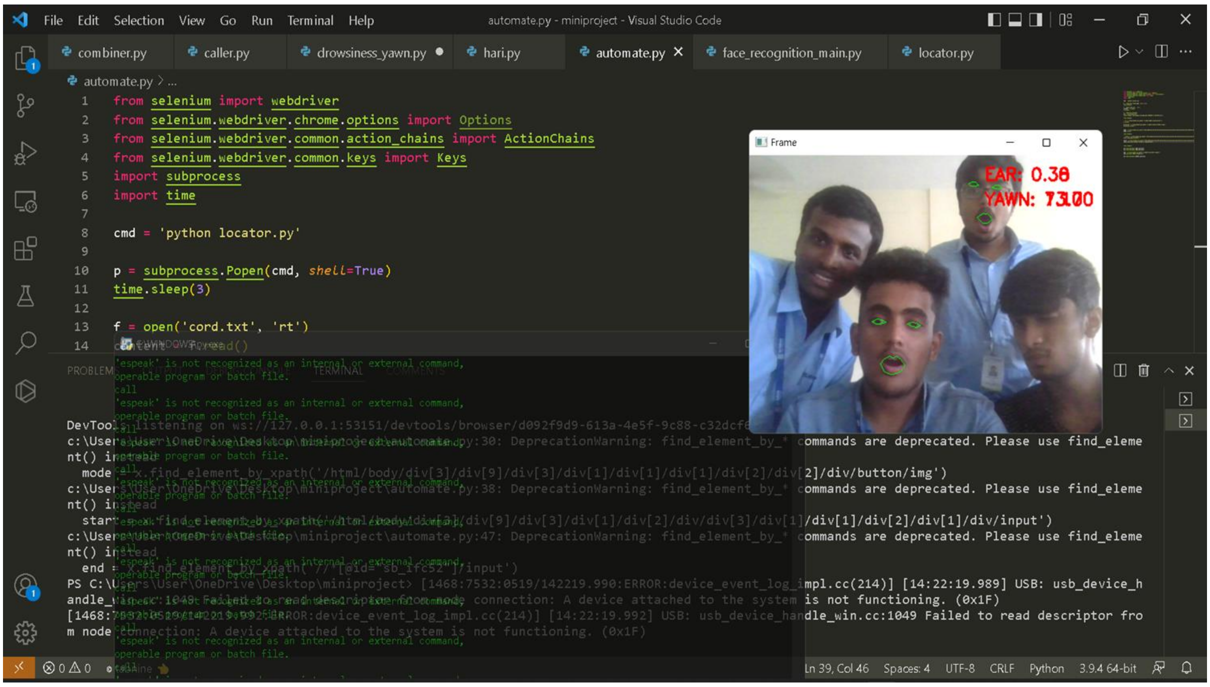


Code part for drowsiness detection:

```
while True:

    frame = vs.read()

    frame = imutils.resize(frame, width=450)

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)


    #rects = detector(gray, 0)

    rects = detector.detectMultiScale(gray, scaleFactor=1.1,

        minNeighbors=5, minSize=(30, 30),

        flags=cv2.CASCADE_SCALE_IMAGE)


    for (x, y, w, h) in rects:

        rect = dlib.rectangle(int(x), int(y), int(x + w),int(y + h))


        shape = predictor(gray, rect)

        shape = face_utils.shape_to_np(shape)


        eye = final_ear(shape)

        ear = eye[0]

        leftEye = eye [1]

        rightEye = eye[2]


        distance = lip_distance(shape)


        leftEyeHull = cv2.convexHull(leftEye)

        rightEyeHull = cv2.convexHull(rightEye)

        cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)

        cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)
```

```
        lip = shape[48:60]
        cv2.drawContours(frame, [lip], -1, (0, 255, 0), 1)
        if ear < EYE_AR_THRESH:
            COUNTER += 1
            if COUNTER >= EYE_AR_CONSEC_FRAMES:
                if alarm_status == False:
                    alarm_status = True
                    t = Thread(target=alarm, args=('wake up sir',))
                    t.deamon = True
                    t.start()
                cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),
                            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
        else:
            COUNTER = 0
            alarm_status = False
        if (distance > YAWN_THRESH):
                cv2.putText(frame, "Yawn Alert", (10, 30),
                            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
                if alarm_status2 == False and saying == False:
                    alarm_status2 = True
                    t = Thread(target=alarm, args=('take some fresh air sir',))
                    t.deamon = True
                    t.start()
        else:
            alarm_status2 = False
        cv2.putText(frame, "EYE: {:.2f}".format(ear), (300, 30),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
        cv2.putText(frame, "YAWN: {:.2f}".format(distance), (300, 60),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
    cv2.imshow("Frame", frame)
```

output:



5.1.2 Automation

In this project we have created an automation part where the program finds our current location and sets the destination to the nearest hospital or the prefered hospital.

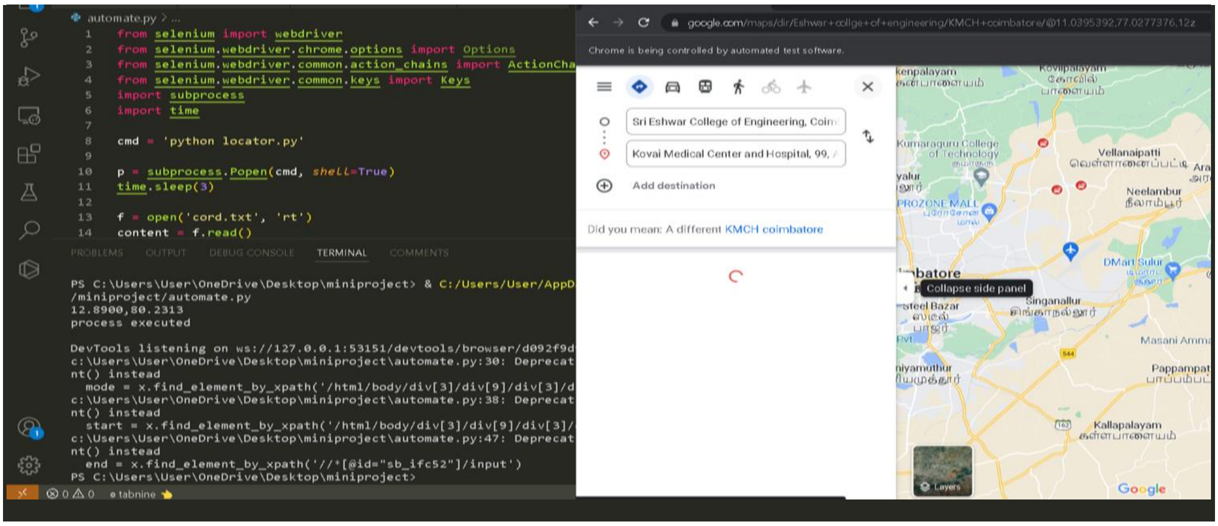
This part of the project is reached only if the EMS protocol is activate.

Code part for Destinaton Automator:

```
cmd = 'python locator.py'

p = subprocess.Popen(cmd, shell=True)

time.sleep(3)

f = open('cord.txt', 'rt')

content = f.read()

f.close()
```

```
x = webdriver.Chrome()

act = ActionChains(x)

x.get('https://www.google.com/maps/@11.0395392,77.0277376,12z')

time.sleep(3)

mode =
x.find_element_by_xpath('/html/body/div[3]/div[9]/div[3]/div[1]/div[1]/
div[1]/div[2]/div[2]/div/button/img')

mode.click()

time.sleep(4)

start =
x.find_element_by_xpath('/html/body/div[3]/div[9]/div[3]/div[1]/div[2]/
div/div[3]/div[1]/div[1]/div[2]/div[1]/div/input')

start.send_keys('Eshwar collge of engineering')

time.sleep(4)

act.send_keys(Keys.TAB).perform()

act.send_keys(Keys.TAB).perform()

end = x.find_element_by_xpath('//*[@id="sb_ifc52"]/input')

end.send_keys('KMCH coimbatore')

act.send_keys(Keys.ENTER).perform()
```

output:



5.2 Implementation

This Project is mainly focused on improving safety feature of autonomous car which is going to essential in the future .

This autonomous Car is a developing field so the result of Implementing this Project will improve the safety feature of the car and due to this the reliability of this car will we increased .

This leads the car into a bestselling product and which will become a profitable to the industry

CONCLUSION

With EMS we can make autonomous cars into smart cars which will lead to the vision of this project which is to prevent every possible accident and ensure that everybody is having a safe ride. Since autonomous cars are the future of transportation, making it safer is the top most priority and from this project the future is not so far.