Abstract

Acknowledgement

Index

[Abstract 1](#_Toc482825670)

[Index 3](#_Toc482825671)

[Chapter 1. Introduction 5](#_Toc482825672)

[1.1 Introduction 5](#_Toc482825673)

[1.2 Problem Definition 5](#_Toc482825674)

[1.3 Scope 5](#_Toc482825675)

[Chapter 2. Literature Review 6](#_Toc482825676)

[Chapter 3. Project Management Plan 7](#_Toc482825677)

[3.1 Feasibility Analysis 7](#_Toc482825678)

[3.2 Lifecycle Model 7](#_Toc482825679)

[3.3 Project Cost and Time Estimation 7](#_Toc482825680)

[3.4 Resource Plan 7](#_Toc482825681)

[3.5 Task and Responsibility Assignment Matrix 7](#_Toc482825682)

[3.6 Project Timeline Chart 7](#_Toc482825683)

[Chapter 4. Project Analysis and Design 8](#_Toc482825684)

[4.1 Software Architecture diagram 8](#_Toc482825685)

[4.2 Architectural style and justification 8](#_Toc482825686)

[4.3 Software Requirements Specification Document 9](#_Toc482825687)

[1.2 Software Design Document 14](#_Toc482825688)

[Chapter 5. Project Implementation 15](#_Toc482825689)

[1.3 Approach/System Architecture / Main Algorithm / Methodology 15](#_Toc482825690)

[1.4 Programming Language used for Implementation 15](#_Toc482825691)

[1.5 Tools used 15](#_Toc482825692)

[1.6 Deployment diagram 15](#_Toc482825693)

[Chapter 6. Integration and Testing 16](#_Toc482825694)

[1.7 Testing Approach 16](#_Toc482825695)

[1.8 Testing Plan 16](#_Toc482825696)

[1.9 Unit Test Cases 16](#_Toc482825697)

[1.10 Integrated System Test Cases 16](#_Toc482825698)

[Chapter 7. Conclusion and Future Work 17](#_Toc482825699)

[Appendix 19](#_Toc482825700)

[I. Minimum System Requirement 19](#_Toc482825701)

[II. User’s Manual 19](#_Toc482825702)

[III. Technical Reference Manual 19](#_Toc482825703)

[IV. Data Sheets of chips used (for hardware projects only) 19](#_Toc482825704)

# Introduction

## Introduction

## Problem Definition

## Scope

# Literature Review

# Project Management Plan

## Feasibility Analysis

## Lifecycle Model

## Project Cost and Time Estimation

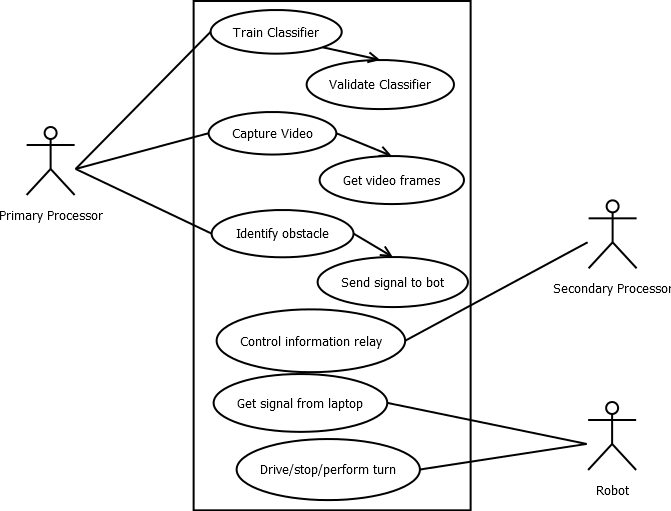
## Resource Plan

## Task and Responsibility Assignment Matrix

## Project Timeline Chart

# Project Analysis and Design

## Software Architecture diagram



## Architectural style and justification

This system uses a **Component based architecture**, where the hardware module and the software module act as logically separated components that exchange information via a well-defined communication interface. Component based architecture was chosen as the flow of data happens to and fro between the software and hardware modules depending on the presence/absence of an obstacle in front of the bot.

## Software Requirements Specification Document

### Introduction:

### Purpose

The intended product is a prototype of an autonomous vehicle that is able to navigate and drive on its own with minimal human intervention. The bot (vehicle) can avoid any obstacles in its path in order to arrive at its destination safely.

### Document Conventions

The priorities mentioned in the functional requirements are in bold for emphasis. Similarly, bold characters are used for emphasis on specific topics.

### Intended Audience and Reading Suggestions

The intended audience is the project review committee and the project mentor.

* **Project Review Committee:** The main focus for the review committee would be the validity and feasibility of the scope of the said project. The review committee may brief through the overview and focus on the system features and other non-functional requirements.
* **Project mentor:** The project mentor may go through the overview of the SRS and may focus more on the external interface requirements, system features and the non-functional requirements.

### Product Scope

The autonomous bot will be mounted with a camera at front which will record a video of the surroundings straight ahead of the bot. It shall then splice the video into frames where each frame will be analyzed to detect the presence of obstacles. Depending on the situation, the car will decide its movement. Situations include – Obstacles, road signs and symbols.

### References

SRS Template:

<https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwilqMLTmrPQAhUIKo8KHVYhBs4QFggZMAA&url=http%3A%2F%2Fwww.csc.villanova.edu%2F~tway%2Fcourses%2Fcsc4181%2Fs2010%2Fsrs_template-1.doc&usg=AFQjCNH5RCUtifVhnKm9Kfwnx6_Z2T5LzA&sig2=6M7DZfJjgaWsXStrx3MwWQ&bvm=bv.139250283,d.c2I>

### Overall Description

### Product Perspective

This product is a prototype of the already in-progress development of self driving vehicles. One of the popular examples of such a vehicle would be that of Google’s self driving car. The proposed product attempts to achieve autonomous behavior similar to Google’s self driving cars on a smaller and feasible scale.

### Product Functions

This product will be able to perform the following functions:

* Detect and avoid the obstacles in its path.
* Identify the type of obstacle (whether it can be overcome or not, e.g. a small bump in its path)
* Identify any traffic signals in front of it and move/stop accordingly.

### User Classes and Characteristics

This product being a prototype of a much larger product does not cater to the needs of any user class. Although, a complete version of the product (real life size model) shall cater to the needs of the general masses in the form of a regular automobile that has the ability to drive to its destination on its own with minimal human intervention. Such a product may me widely used by cab companies to replace their human driven cabs with the new autonomous alternatives. Once completely tested, any user may access the services of the real life product without any special technical/educational expertise.

### Operating Environment

This product shall work on **Windows operating system** and shall use Matlab as the major operating environment. Apart from this, it shall incorporate the services of hardware components such as Arduino microcontroller, motors, batteries, etc. The image processing shall be performed using Matlab which will be used in sync with Arduino IDE which in turn is responsible for controlling the hardware components of the product. The product shall operate in open surroundings with suitable assumptions and controls.

### Design and Implementation Constraints

Following design and implementation constraints are endured:

* Being a prototype, the entire computation cannot be done on the device itself. Thus the bot shall be connected to a laptop via the USB port in order to perform the complex image processing mechanisms.
* Using small scale motors and batteries, the bot shall use the laptop as the main source of energy.
* Being at a rudimentary stage, the bot shall not simultaneously drive and process its surroundings; the bot shall first process its surroundings and then accordingly move/stay.
* Sudden obstacles cannot be detected.

### Assumptions and Dependencies

Following assumptions and dependencies exist for the product to function:

* Since this product uses many hardware components, the product assumes that each hardware component performs without any errors.
* The video captured by the camera on the bot does not have unwanted ambient lighting (reflections on floor) which may compromise on the product’s ability to process the video frames efficiently.

### External Interface Requirements

### Hardware Interfaces

The software part of the product shall be performed by Matlab which relays relevant information to the Arduino IDE, which in turn controls the microcontroller that controls the motor of the vehicle (bot). This is done by using the in-built libraries that allow communication between Matlab code and Arduino code by the creation of an Arduino object in Matlab.

### Software Interfaces

The major software used is Matlab that runs on Windows operating system. Another software interface that plays an important role is the Arduino IDE which is a development environment for developing Arduino codes. This IDE works in collaboration with the Matlab. Matlab is responsible for processing the video captured by the camera. After processing, the outputs are then checked against appropriate conditions. Accordingly, the relevant output is relayed to the Arduino chip which runs/stops the motor that controls the movement of the bot.

### Communication interfaces

The product mainly implements the Matlab-Arduino serial port interface for transfer of image processing output data which will be used for manipulating the movement of the bot.

### System features

### Obstacle Detection

### Description and Priority

This feature deals with the ability of the bot to detect any obstacle that may be in front of it. Since avoiding collisions with is the prime concern of any vehicle, this functional requirement is **high priority**.

The risk involved with this feature is also a **high priority** risk as it pertains to the purpose of driverless vehicles- safety.

### Stimulus/Response Sequences

A short video is captured by the camera placed on top of the vehicle. This video is then immediately processed using appropriate image processing algorithms to detect the presence of any object that may hinder the safe movement of the bot.

### Functional Requirements

1. The camera must capture proper videos of appropriate duration.
2. The bot must be stationary while capturing the video.

### Image Identification

### Description and Priority

This feature deals with identification of the surrounding that is captured by the camera. This includes identification of traffic signal lights and signs. Since traffic rules are of utmost importance for road safety, this feature is of **high priority.**

The risk involved with this feature is also a **high priority** risk as it pertains to the purpose of driverless vehicles- safety.

### Stimulus/Response Sequences

The image is tested upon by a pre-trained classifier which classifies the obstacle (if any) into pre-defined categories. Depending on the category, appropriate signal is sent to the robot which navigates accordingly.

### Functional Requirements

1. The color of the traffic light should be correctly analyzed.

### Other non-functional Requirements

### Performance Requirements

The sequential execution of various processes is paramount for safe navigation. The video must be captured and processed in minimal time which in turn reduces the waiting time of the bot. the obstacle (if any) must be detected by the image processing algorithm for the bot to decide whether it should move or not.

### Software Quality Attributes

### Appendix A: Glossary

* Bot: Robotic vehicle

### Appendix C: To-Be-Determined List

## Software Design Document

### Introduction

### Design overview

For the selected project, sequential design principle has been used. This is because the working of the autonomous bot heavily depends on the correct functioning of the a series of activities namely,

* Object identification
* Information relay
* Hardware operations

### Software Architecture Design

### Chosen System Architecture

### System Interface Design

The following system interfaces take part in the working of the bot:

* Matlab-Arduino interface: This interface is responsible for exchange of data regarding the movement of the bot. After the video is processed by Matlab, an appropriate output is generated which signals the Arduino chip to relay corresponding movement functions to the Hardware interface.
* Hardware interface: The Arduino controls the functioning of the motor that runs the bot. This is done by the Arduino libraries that allow it to communicate with the mechanical hardware components.

### Detailed Description of Components

### Software: Object Identification

### Responsibilities-

This component is responsible for capturing the video feed from the camera, splicing it into multiple frames and applying obstacle detection algorithms to perform obstacle detection and identification on the generated series of frames.

### Interactions-

This component interacts with the camera feed, processes the video and relays germane information to the Arduino component for hardware operations to take place.

### Constraints-

The video is not processed in run time due to computational constraints.

### Information Relay

### Responsibilities-

This component is responsible for accepting the output from the image processing module (Matlab) and notifies the hardware component of the same.

### Interactions-

This component interacts with the Matlab code for accepting the output of the processed video. It also interacts with the motors that are responsible for the movement of the bot.

### Hardware Operations

### Responsibilities-

This component is responsible for driving the motor thereby causing the actual movement of the bot.

### Interactions-

It interacts with Arduino chip receiving control signals that are needed for the movement of the bot.

# Project Implementation

## Approach/System Architecture / Main Algorithm / Methodology

## Programming Language used for Implementation

## Tools used

## Deployment diagram

# Integration and Testing

## Testing Approach

## Testing Plan

## Unit Test Cases

## Integrated System Test Cases

# Conclusion and Future Work

References

Appendix

# Minimum System Requirement

# User’s Manual

# Technical Reference Manual

# Data Sheets of chips used (for hardware projects only)