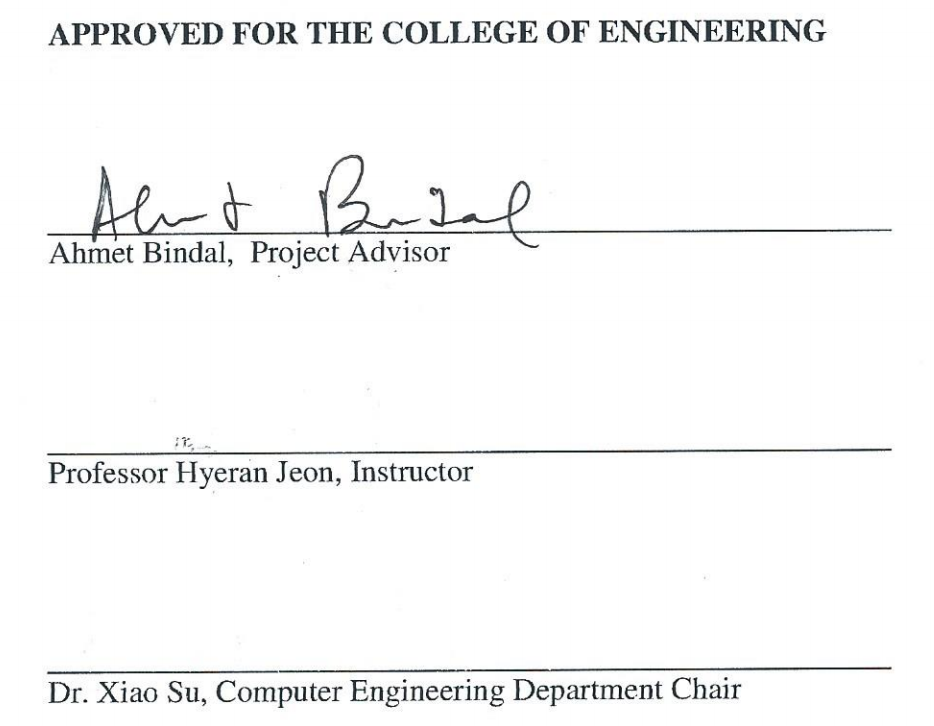
**Autonomous Secure Delivery**

|  |
| --- |
| A Project Report Presented to  The Faculty of the Computer Engineering Department |
| San Jose State University In Partial Fulfillment Of the Requirements for the Degree Bachelor of Science in Computer Engineering |

|  |
| --- |
| By |
| Mohit Bhasin  Brendan Donahue  Nadim Sarras |
| 05/2017 |

|  |
| --- |
| Copyright © 2017 |
| [Mohit Bhasin, Brendan Donahue, and Nadim Sarras] |
| ALL RIGHTS RESERVED |



|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**ABSTRACT**

Trading and delivering goods is a large part of our economy. Public and private corporations are both involved in the delivery process, which has not seen much progress in recent decades. Companies are currently experimenting with new and innovative ways to deliver packages. Companies are trying to lower the cost and complications that come with the current outdated process. The result from these experiments show the next logical step is automation.

Although large delivery companies have been delivering packages and mail for decades the process is far from perfect. Delivery services are not available outside of business hours, are subject to human error, and are not cost efficient. Package delivery is dependent on human truck drivers manually delivering mail door to door. Resulting in the package not taking the most cost effective route towards the destination and may even be missed due to human error.

An autonomous secure delivery system designed to deliver packages with minimal human intervention. The vehicle is designed to transport a user’s package with the help of an image recognition camera and the SJSUOne development board.  The Pixy CMU camera will be taught to recognize a predetermined object. The use of our algorithm and other peripherals will be used to guide the robot to its destination. The chassis of the robot will include a secure compartment to store the user payload from start to destination.  Automating the delivery process minimizes cost and complications, while increasing efficiency and reliability.

**Table of Contents**

[**List of Figures**](#_d5yue9yg3q2u) **8**

[**List of Tables**](#_5dhitckq8o2a) **9**

[**Chapter 1 Introduction**](#_72v50nofp38m) **10**

[Project Goals and Objectives (Mohit)](#_1zr0idpfsuxy) 10

[1.2 Problem and Motivation (Brendan)](#_lkq7acuoegb4) 10

[1.3 Project Application and Impact (Mohit)](#_9odtjaxjry0b) 10

[1.4 Project Results and Deliverables (Brendan)](#_y5l8dp8wjd7) 11

[1.5 Market Research (Mohit)](#_hcknt4to9t49) 12

[1.6 Project Report Structure](#_hxad3mjzqvpz) 12

[**Chapter 2 Background and Related Work**](#_30j0zll) **13**

[Background and Technologies (Mohit)](#_4rp33qd8yea2) 13

[State-of-the-art (Mohit)](#_ge45fl19fklv) 14

[Literature Survey (Mohit)](#_w2wwoqrs6tvf) 15

[**Chapter 3 Project Requirements**](#_mf4vi6mkwdc5) **19**

[3.1 Domain and Business Requirements (Nadim)](#_itctwjqbikp) 19

[3.2 System (or Component) Functional Requirements (Brendan)](#_3at7jjlzblov) 19

[Functional Requirements](#_lq0hm6rrcqv5) 20

[3.3 Non-functional Requirements (Nadim)](#_w73e54kmucuj) 20

[Non Functional Requirements](#_d7nxoj64ejoi) 20

[Context and Interface Requirements (Brendan)](#_j35u5gkatzs) 21

[Technology and Resource Requirements (Mohit and Brendan)](#_q1dlfc72ko0q) 22

[**Chapter 4 System Design**](#_8miphz6axf3d) **25**

[4.1 Architecture Design (Nadim)](#_3cudxt3frogu) 25

[Interface and Component Design (Nadim)](#_q5e0hchyrju5) 26

[Structure and Logic Design (Nadim)](#_i2fvwcbs5608) 27

[4.4 Design Problems and Challenges (Nadim)](#_y176ejhq9yol) 29

[4.4.3 Design Solutions and Trade-offs (Mohit)](#_7tcgntq14tx4) 30

[**Chapter 5 System Implementation**](#_c7yxfid6uuhu) **31**

[5.1 Implementation Overview (Brendan)](#_kqyk16cd2zrn) 31

[5.2 Implementation of Developed Solutions (Brendan)](#_v9yhw22igfxx) 31

[5.3 Implementation Problems, Challenges, and Lessons Learned (Brendan)](#_m80t5bkh4neg) 32

[**Chapter 6   Tools and Standards**](#_i8cn057lrdbl) **33**

[6.1.   Tools Used (Nadim)](#_l1sx5bknp2sm) 33

[6.2.   Standards (Nadim)](#_adxe2vg8qg3v) 34

[**Chapter 7   Testing and Experiment**](#_hkybpfhg23rk) **34**

[7.1    Testing and Experiment Scope (Mohit)](#_agb54k41s0u0) 34

[7.2    Testing and Experiment Approach (Mohit / Brendan)](#_1ye8ypmpti6p) 36

[7.3 Testing and Experiment Results and Analysis (Brendan)](#_k5ro707qxs2f) 37

[**Chapter 8 Conclusion and Future Work**](#_y6tpowktcx6a) **38**

[8.1 Conclusion (Mohit)](#_7x775x4qt3zs) 38

[8.1 Future Work (Brendan)](#_gpbgwoee36ki) 39

[**References**](#_1fob9te) **40**

# List of Figures

Figure 1. Market Capitalization of Robotics for future years 3

Figure 2. Process Summary UML 2 Activity Diagram 15

Figure 3. ASD bot architecture 18

Figure 4. Component Diagram 20

Figure 5. Logic Design Flowchart 21

# List of Tables

Table 1. Hardware and Software Components Table 15

Table 2. Hardware Parts Table 16

Table 3. Software Module Table 16

# Chapter 1 Introduction

## Project Goals and Objectives (Mohit)

The goals and objectives of this project is to build an embedded system making an autonomous delivery robot, which has the ability to deliver the contents of the robot’s on board storage to a unique location. By the end of this project, we would like to have a fully functional autonomous robot that can guide itself. We plan to have a storage compartment on the robot. Extra peripherals will also be added to increase object avoidance.

## 1.2 Problem and Motivation (Brendan)

Current delivery methods are outdated and inefficient, according to a study by HorizonTech around 5% of USPS mail does not get delivered.  The motivation for this project comes from the need to optimize the package delivery system.  The current delivery methods rely heavily on humans to deliver packages, Thus causing the system to be limited by the humans limitation of precision, energy, and reliability. This robot answer is to limit the amount of human interaction of package delivery.

## 1.3 Project Application and Impact (Mohit)

The best solution is an autonomous secure delivery system designed to deliver packages without human intervention. The vehicle is designed to transport a user’s package with the help of an image recognition camera and the SJSUOne development board. With the use of our algorithm and other peripherals will be used to guide the robot to its destination. The chassis of the robot will include a secure compartment to store the user payload from start to destination.  Automating the delivery process minimizes cost and complications while increasing efficiency and reliability.  This is an efficient way for companies to increase productivity and revenue is to limit reliability on humans. When applied on a large scale this solution can have measurable positive impacts on the delivery company.

Our project has an impact on how packages can be delivered and a significant impact in a workplace environment.  The main problems our project will solve are saving time and money when moving or delivering goods. This implementation of machine learning and object recognition will allow for the vehicle to make an impact on the delivery system like never before. Overall, the autonomous delivery robot can be used in a wide variety of settings and be helpful in accomplishing daily tasks.

## 1.4 Project Results and Deliverables (Brendan)

The main deliverable of this project will be an autonomous robot capable of delivering packages to discoverable locations.  The code for image recognition, motor controls, and interrupt controls will also be included.  Additionally the custom driver to CPU to camera communication will be included as this is the center of the autonomous technology.  A report documenting the development process and research will be submitted as well.

Besides the actual robot, we will submit machine learning algorithms and realtime operating system structures that made the vehicle possible. Our project deliverable includes all of our embedded code, modules, and application code in a .zip file. There will also be all system diagrams, which include both hardware diagrams and software diagrams. The final results of the projects testing and verification will be included.

## **1.5** **Market Research (Mohit)**

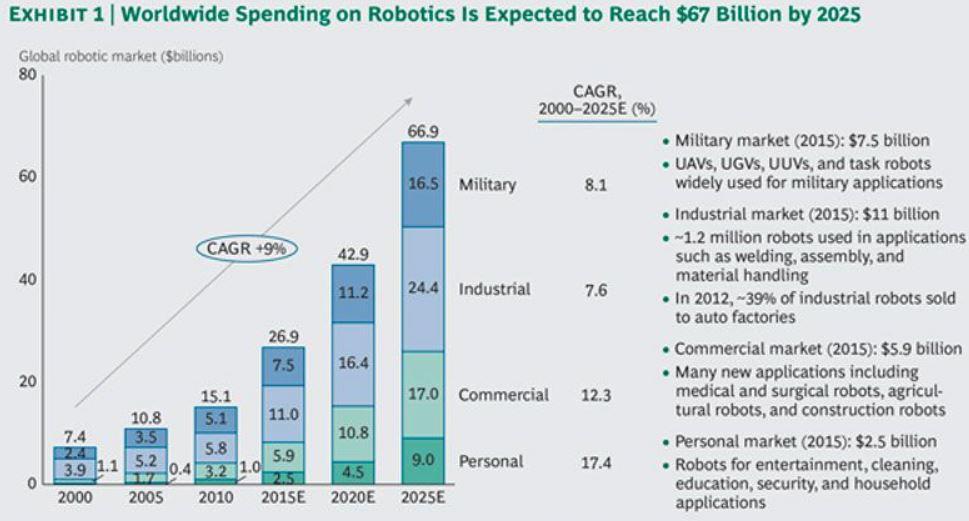
The Mobile robots market is increasing exponentially and should approach tens of millions dollars by the year 2020. Robots can be used for ground, marine and aerial purposes for a wide variety of applications. Different companies are coming up with different innovations for advancements in their field. Some large players in market are  iRobot Corporation , Amazon.Com, , Google, Bluefin Robotics Corporation , Adept Technology, Gecko Systems , and Northrop Grumman Corporation. These companies have strongly committed to researching and attempting to implement mobile robots. 

Figure 1: Market Capitalization of Robotics for future years

## 1.6 Project Report Structure

The next chapter will outline the remaining information need to complete this project.  The project requirements will be listed and explained to display what is necessary for this project to be successfully.  The design of the system and how the pieces will be integrated will be shown in detail as well as the tools used to

# Chapter 2 Background and Related Work

## Background and Technologies (Mohit)

The project consists of implementing an autonomous secure delivery system on a mobile platform. The pixy camera, a machine learning camera will be used and programmed to recognize a docking station with given dimensions and certain features. With autonomous secure delivery, human error can be limited and significantly less packages will be misplaced. The cameras data interpretation and motor control will be achieved with a development board. The processor relays signals between the camera and the development board.

Since this is an embedded systems project it will consist of both hardware and software. The objective is to have the pixy camera communicate with the processor to find the image. When the pixy is able to effectively communicate data to the processor and back, the pixie camera will be able to recognize unique objects that the user wants to search for. IR sensors will be implemented around the robot to avoid objects and be the eyes for the robot. The IR sensors will relay information back and forth to the microcontroller about the distances of surrounding objects. The robot will have a chassis and servos to allow it to move at a reasonable speed towards its destination.

The tools being used include eclipse for embedded c, Pixymon, Github, and hercules. The technologies being implemented are infrared sensors, PWM servo motors. pixy camera, and the SJSUOne development board.

The San Jose State courses that were relevant in the building of this project are CMPE 146: Embedded Systems Design, CMPE 130: Advanced Algorithms and Data Structures, CMPE 110: Electrical Engineering and Hardware Design, CMPE 124: Digital Design, CMPE 127: Embedded Systems Coding, and CMPE 140: Digital Design II.

## State-of-the-art (Mohit)

The state of the art for the ASD robot includes recognizing objects, maneuvering around objects and delivering packages autonomously. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds is an IEEE article that has been peer reviewed. This article discusses the different benefits of autonomous vehicles and how they are controlled. The articles also discusses that there is more advancement available in the field of autonomous vehicles. Autonomous vehicles are able to connect to each other on a grid as well as connect to the internet and each other. This capability makes each vehicle more knowledgeable and easier to manage.

Essentially our robot will be an autonomous vehicle that will have the ability to search, until it has recognized the object it needs to deliver to. An image recognition camera can be taught to identify different objects. Since the camera can be taught different objects, we can use this to our advantage to have the robot deliver items to different locations.

We are also making use of the latest sensors for object avoidance. This allows the robot to maneuver around objects in its path as well as avoid objects. Using sensors to allow the robot to think for itself in real time situations makes operations autonomous. This allows for less human error, since the robot will make precise decisions. The main goal our project it to implement a vehicle that will not need any human supervision and will be able to deliver items without any help.

## Literature Survey (Mohit)

Autonomous vehicles have changed the way people think about traveling and is being implemented in everything from household appliances like vacuums to large automobiles. Advances in embedded systems, communications with electronics and controlling vehicles has led to innovation of autonomous cars. Technology available to cars is rapidly increasingly. Different types of sensors and cameras are being added cars to make them more aware of their surroundings as well have more functionality within the car itself. Meanwhile implementing autonomous vehicles in warehouses can increase productivity and can be readily available at any time.

The use of autonomous vehicles is very practical since it gets rid of human error and vehicle is able to operate smoothly without the human controlling it. The use of embedded systems is the key for all these components to work together in harmony and communicate with each other because a single process can control multiple peripherals. With embedded system design different types of sensors and cameras have the ability to communicate with each other in real time. When the vehicle is able to recognize objects, patterns, road conditions, routes and the path to take, it will be able decide its own best path to the destination.

The next technological advancement in autonomous cars is connected vehicles and IOT devices available for the car to use. When the vehicle is able to communicate with other internet enabled devices, the whole system becomes grows incorporating new inputs and outputs. If the vehicle has the capability to connect to the internet, then it will be able to communicate with other vehicles over a network. These vehicles can be connected to other autonomous vehicles with the same capabilities. This can include advancements in autonomous driving and advance how vehicles move on the road.

The main benefit of autonomous secure delivery is it takes out human error as well the service is readily available at all times. The fact there will be less human error involving these autonomous vehicles almost eliminates the chances of accidents and other incidents. The autonomous vehicle will also not be restricted to working certain day or hours. These vehicles increase productivity and lower the cost to operate.

Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses by:

Peter R. Wurman, Raffaello D'Andrea, Mick Mountz is an article about autonomous robots in warehouses increasing productivity, accountability and flexibility. This article pertains to our topic since it discusses how autonomous robots were used to delivery and organize goods in a large warehouse. The vehicles are able to transport items securely from one place to another and organize products by themselves.

Trajectory Tracking for Autonomous Vehicles: An Integrated Approach to Guidance and Control is an article about guidance and control system for autonomous vehicles. This research will lead to efficient design controllers for unmanned air vehicles, but we will use the approach for design controller for our unmanned autonomous vehicle. This article goes over how to find the correct path and how to guide vehicle through the correct path to reach its destination.

Factories in California are progressing their technological advances and working towards the future, by creating autonomous robots. Autonomous robots that are able to operate by themselves, and accomplish tasks without supervision are the future of manufacturing products. Autonomous robots can also increase efficiency and productivity, which leads to increasing the bottom line of any company by reducing human cost.

For example, a large company like target is using autonomous robots to move and transport items of shelves. These robots are completing simple tasks and replacing humans in the warehouses and large factories. Since the robots are autonomous, they just have to be monitored every now and then. They also increase efficiency, by moving items at a faster rate and are available at all times of the day.

Autonomous robots can navigate through their environment and discover information about their surroundings. For example, a vacuum robot can function without any human interaction, then clean its surroundings and learn more about its surroundings. The use of autonomous robots is very useful, since they do not require human help to function properly. These robots have the capability to view their surroundings. Technologies like cameras, infrared sensors, ultrasound sensors and navigation software to guide themselves through obstacles.

The applications for autonomous robots are endless. The military and department of defense is using these technological advances to have an upper hand on their enemies. Since, autonomous robots are in disposable, they can be used for disaster relief, performing complex tasks, surveillance, and be used in dangerous situations. As technology, increases and use of technology on a daily basis increases, autonomous robots can be viewed as crucial to human advancement. A relationship with robots and humans has to evolve, while improving advancements for the elevation of humankind.

# Chapter 3 Project Requirements

## 3.1 Domain and Business Requirements (Nadim)

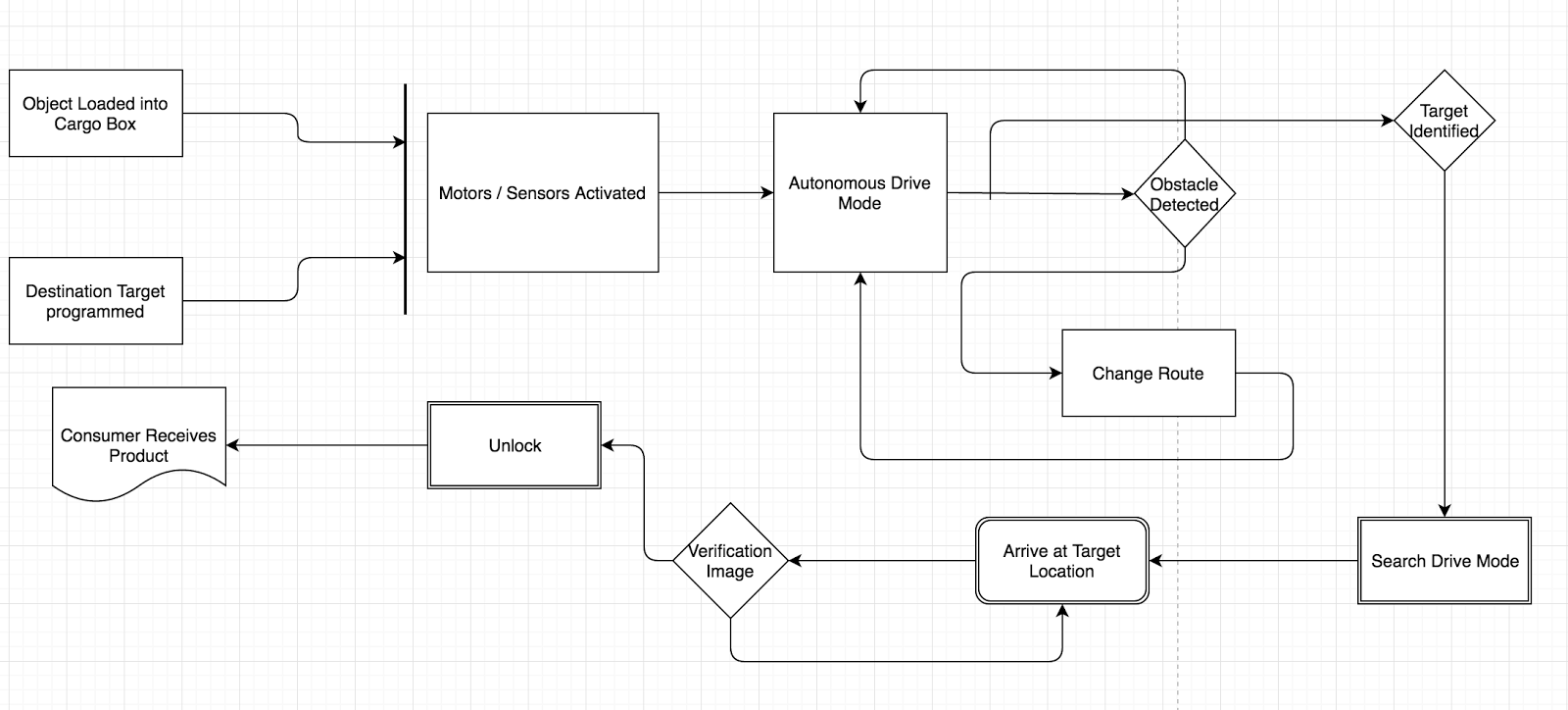


Figure 2: Process Summary UML 2 Activity Diagram

As shown in the UML 2 Activity Diagram the process begins with the bot receiving the package until finally delivering the package to the target location. For this robot to fit in the current delivery space it must be able to securely deliver a package to a destination much like current delivery works today.

## 

## 3.2 System (or Component) Functional Requirements (Brendan)

The functional requirements for this vehicle are split up between optional and functional.  The functional requirements are those that are necessary to complete the main objective of the project.  The optional requirements are those that are optimal and would improve an already working project.

### **Functional Requirements**

* Object Recognition
* Object Tracking
* Motor control
* Object Avoidance
* Real time data analysis
* Chassis with storage compartment
* On board power source
* Programmable micro controller

## 

## 3.3 Non-functional Requirements (Nadim)

The non functional requirements on this project include the robustness, speed, and other components that are not necessary for the ASD robot to complete its desired task.

### Non Functional Requirements

* Speed
* Hold varying package size
* All terrain drive train
* Lockable compartment
* Multiple package capability
* varying Docking stations
* Battery consumption

## Context and Interface Requirements (Brendan)

1. Enabling Pixie Cam to recognize images from mobile phone. This is one of the main keys to the implementation of the ASD robot will look for the object and guide itself to that object. The locking mechanism is a compartment that locks and unlocks dependant on whether the Pixie Camera is able to recognize the image.
2. Pixie Cam must be able to identify unique object located as part of docking system.
3. Integrate Pixie Cam communication to SJSU One Board, without communication between these devices the microcontroller will not know which way to direct the motors of the bot.
4. Configuring IR sensors with motor direction, is for autonomous driving to avoid objects.

The entire development process utilizes a variety of tools and environments. All of the software written for the product is developed in the C language, through the use of the Eclipse IDE.

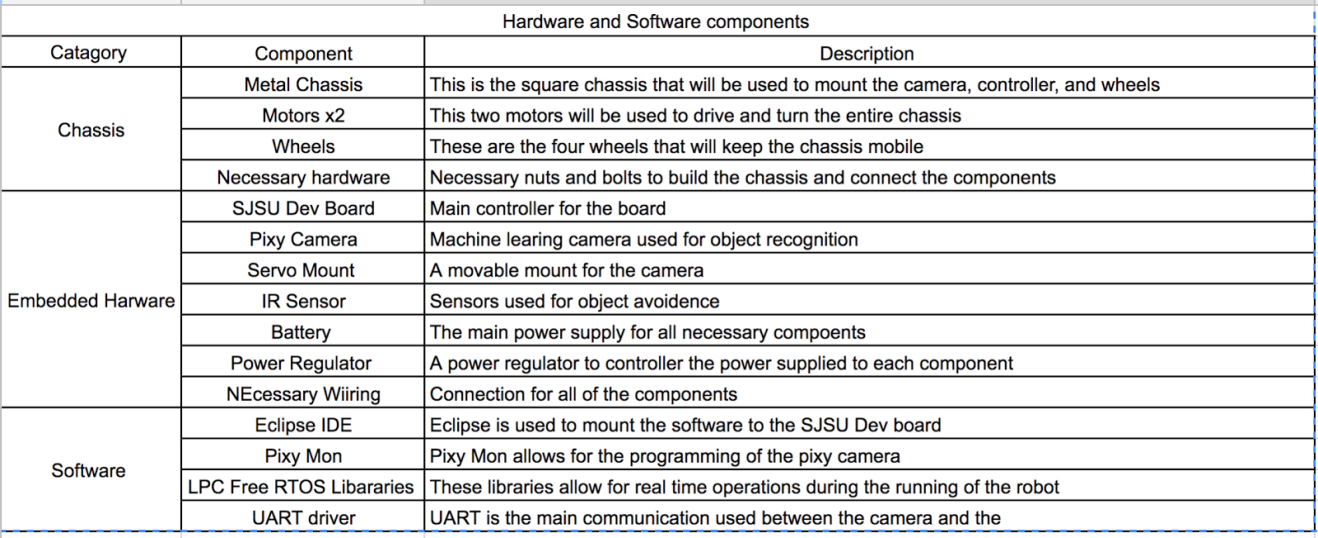


Table 1: Hardware and Software Components Table

## Technology and Resource Requirements (Mohit and Brendan)

|  |
| --- |
| **Hardware Components** |
| PWM Motors |
| SJSU One Board |
| IR Sensors |
| Robot Chassis |
| Chassis Compartment |
| PCB Boards |
| Circuit Boards |
| Power Supplies |
| Wires |
| Pixy Camera |
| Pixy Camera's Motors |
| 4 Wheels |
| Servo Attachments |

Table 2: Hardware Parts Table

|  |
| --- |
| **Software** |
| Embedded Eclipse Compiler |
| Embedded C |
| Pixy Cam Software 2.06 |
| IR Sensor Module |
| PWM Motors Module |
| UART Module |
| Real Time Operating System (RTOS) |
| Scripts to Test Software |
| Power Supply Software |

Table 3: Software Module Table

# Chapter 4 System Design

## 4.1 Architecture Design (Nadim)

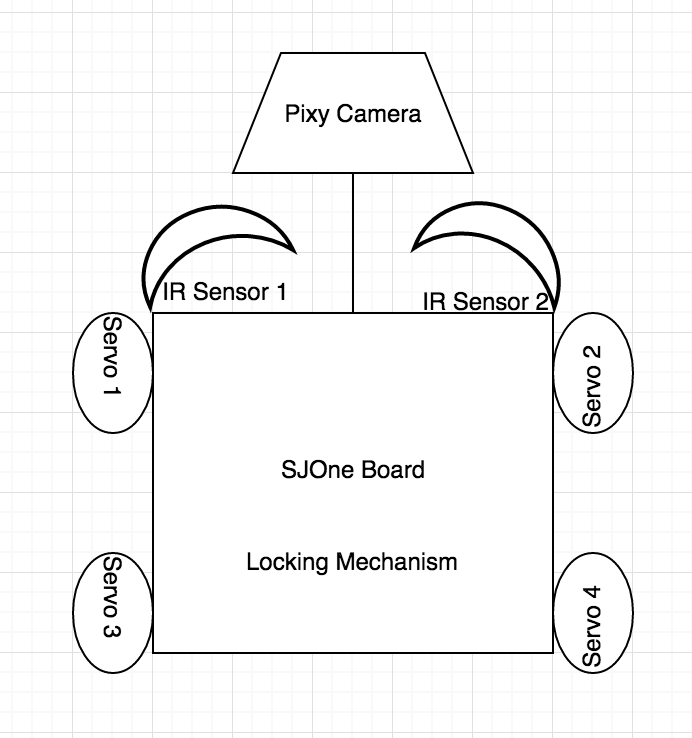


Figure 3: ASD bot architecture

The core of the ASD bot relies on the SJSU One Development Board This microcontroller is where all components are integrated and controlled on the ASD bot. After the ASD bot receives the package and needs to find the Docking Station, it will begin autonomously searching until the Pixie Camera recognizes the symbol on top of the docking station that was programmed into the camera. Once the camera has marked the object it will begin outputting the information of where the object is in relation to the camera. The microcontroller will then direct the motors to park the ASD bot at the docking station. The docking station serves as the central location designed to drop off the package for the user.

## Interface and Component Design (Nadim)

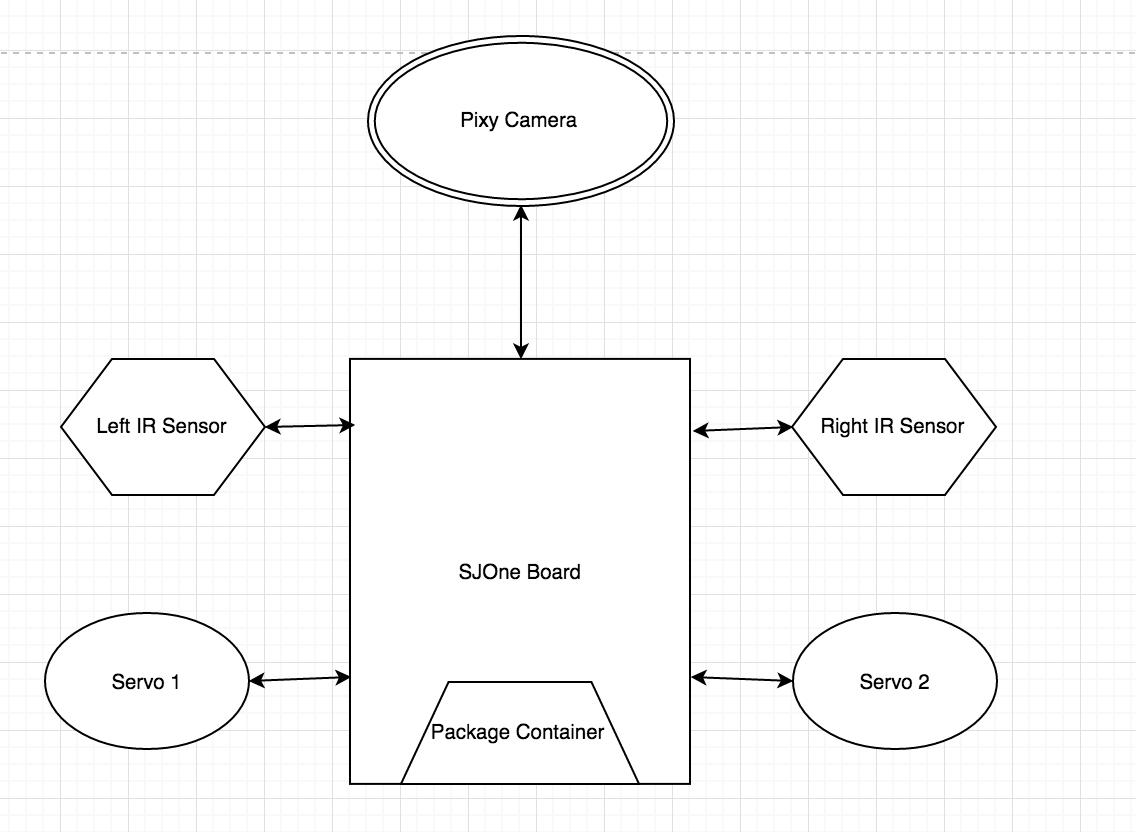


Figure 4: Component Diagram

All parts of the Autonomous Secure Delivery bot interface with the main microcontroller, the SJOne board. The SJOne board receives inputs from the IR sensors and the Pixy Camera. Depending on these inputs the SJOne board will control the current flow to the servos controlling the package container and the wheels of the vehicle.

## Structure and Logic Design (Nadim)

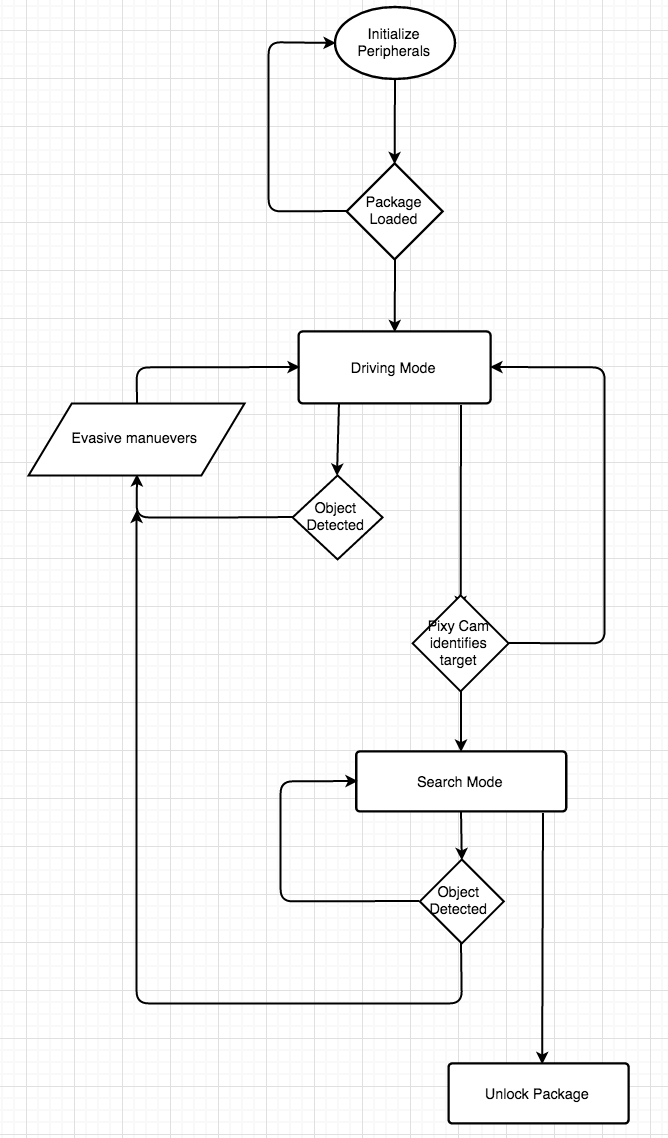


Figure 5: Logic Design Flowchart

The entire logic design of the product consists of many input dependent operations.  The bot first checks to see if a package has been loaded in the compartment and then initializes all of the peripherals. Once this is done the bot begins to drive in autonomous mode until the Pixy Camera sends the micro controller a sync byte signifying that the target location has been identified. If an object is detected along the way by the IR sensors the bot will then evade the object and resume the current drive mode it was in. Once the sync byte is received the bot will enter search mode where it will make its way towards the target location, taking into account obstacles along the way.

## 4.4 Design Problems and Challenges (Nadim)

Incorporating a container to transfer packages within the Autonomous Secure Delivery bot resulted in many design challenges. The team needed to design a compartment that could hold most size objects that are being delivered through the mail. The robot also needed a power supply capable of powering the embedded components as well as the motors to ensure packages of different sizes and weight could be delivered. The package concept of the design required the entire team to take inventory of the weight of every component and calculate how much power was required to drive the vehicle toward its destination. As for economics, the team had to carefully budget for components required for the design. Certain brand named components would have not allowed the final product to be cost efficient. Once the correct components were purchased and integrated the final main challenge was programming the development board to work in real time. Reading in multiple sources of data that all required changes within the motor was challenging. This required the team to calculate which inputs had the most priority over motor control.

## 4.4.3 Design Solutions and Trade-offs (Mohit)

Our group discussed the pros and cons of ultrasonic sensors vs IR sensors. We tested both sensors out and chose the IR sensors. The IR sensors would not fail as much as the ultrasonic sensors and would consistently output data that the development board could understand.

While some members of the group wanted to use a large chassis others wanted to use a small chassis. We came to a compromise and used a chassis that was in between the large and small size. This design solution was made to optimize movement and object avoidance.

Our advisor also helped us choose a development board. We discussed the pros and cons of using another development board. We decided to move forward with the SJSUOne Development board since we had more knowledge on how to operate it. Although our advisor did not like this decision, we had to make this trade off to make our project more successful. At the end when we showed him our end product and demonstration he then appreciated the switch we made.

# Chapter 5 System Implementation

## 5.1 Implementation Overview (Brendan)

The implementation of this project relies on a seamless interaction between the necessary software and the hardware.  The controller used for this project is the SJSU Dev board.  The programming of this controller is done in C.  For this communication to work a few dependencies are necessary.

The main communication between the controller and the pixy camera required a custom made UART library.  This dependency allows for communication between the board and the camera run in the background while deciphering the camera output.

Controlling the motors is done using the controllers PWM (Pulse With Modulation) controls.  Once the correct dependency is added the SJSU Dev board is able to communicate perfectly with multiple motors.

The onboard IR sensors used for object avoidance are connected through the onboard analog to digital convertor pins. This allows the sensor to detect objects and allow the program to know how the object is in its path. Having extra peripheral devices allows the ADS Robot to be well rounded and operate smoothly in a real time.

## 5.2 Implementation of Developed Solutions (Brendan)

The center of this project was developing the interface between the pixy camera and controller over UART communication.  For this communication a real time operating task was created.   Running this task first initializes the communication between the two by setting the correct ports for the communication.  Next the correct communication speed was set for UART.  Once initialized correctly the controller is able to correctly read and decipher what the camera is seeing.

Once the communication between the camera and controller was established the controller needed to generate the correct outputs based off what the camera was seeing.  This required coding motor control functions that would follow the object the camera was tracking.  To interrupt the motor control functions IR sensors were installed to detect objects outside of the field of the camera

## 5.3 Implementation Problems, Challenges, and Lessons Learned (Brendan)

The key component of this project is the communication between the SJSU Dev board and the pixy camera.  This piece was the most difficult to code and stabilize.  The initial problem was trying to get constant reliable communication between the two devices.  Initially the team was trying to establish SPI communication with a different controller but this brought up many unforeseen issues that were taking time to figure out.  Eventually the team decided to switch to the SJSU Dev and UART communication. More time should have been put into this communication at an earlier time since many of the other functions within the robot are based off this communication.  The team should have also decided earlier to switch to a controller that was more comfortable to everyone on the team.  Since this communication was put off the rest of the functions that were based off of this communication were delayed and delayed the timeline of the project.

While this was the main hurdle of the project other smaller hurtles taught the team many lessons.  The second hurdle the team faced was in the decision to use IR or Ultrasonic sensors.  While Ultrasonic sensors did not work as expected after some experimentation the team did decide on IR sensors for object avoidance.  Both of these issues taught the team that we need to make decisions early and definitely.  Earlier decisions would have allowed for the team to go down the correct paths earlier.

# Chapter 6   Tools and Standards

## 6.1.   Tools Used (Nadim)

*Software Tools*

All software development occurred in the Eclipse Mars IDE. Eclipse was selected for the development environment due to the custom applications that allow source code to be directly flashed onto the SJOne microcontroller. The team decided to use GitHub as a versioning tool while developing source code. GitHub is proficient in managing multiple development branches and provides tools to keep all developers on the same track. This feature was extremely useful because every group member would focus on developing code for different modules. GitHub was then used to merge all of the source code into a single source package.

*Hardware Tools*

The hardware development team utilized a multitude of breadboards to prototype the hardware components before integrating them together. Breadboards were chosen to prototype the different components of the ASD (Autonomous Secure Delivery) Robot to minimize the risk of damage to parts. In addition, the usage of breadboards allowed the hardware development team to work a lot more efficiently and to waste less time. In order to thoroughly test components, the team had used wire wrapping tools instead of soldering. This allowed the team to put modules together for testing and to separate them easily for future integration and testing purposes. Multi-meters and oscilloscopes were also used to analyze current outputs of certain terminals to ensure every component was properly powered. The multimeter was also used to measure voltage sources through voltage regulators to ensure correct voltage input into devices.

## 6.2.   Standards (Nadim)

Autonomous Secure Delivery utilized Agile methodology to develop different modules and their functionality. Frequent code reviews were enacted, as well as constant collaboration with other group members while developing code. The team had also enforced a strict documentation standard. Strict documentation standards were critical in order to allow the entire team to be on the same page when members worked on individual modules. Development of hardware incorporated a strict standard to prototype all testing of modules on a breadboard first. This protocol was implemented to ensure that no components were short circuited and all components are functionally before being integrated together.

# Chapter 7   Testing and Experiment

## 7.1    Testing and Experiment Scope (Mohit)

Our test objective is to make sure we have a functional autonomous secure vehicle that will be able to deliver products. Our focus was making sure that every part functions properly during real time application. We unit tested every component of the vehicle individually. We also test every function of the code individually. This made it a lot easier to combine the components together and integrate all the functions with one another. Since this project is an embedded project and we have to verify both the software and hardware are working, we tested it different from how software is usually tested. We had to run a lot of subsystem tests to make sure all the subsystems are working correctly. Also, we ran system integration tests to make sure the system we integrated was functioning together. We used a combination of black box and white box testing. The black box testing was to make sure anyone who was to operate this vehicle could do so with ease. The white box testing was to make sure all the functions written for each peripheral was functioning properly.

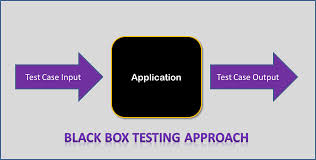


Figure 7: Black Box Testing

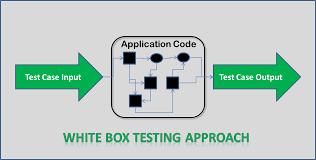


Figure 8: White Box Testing

## 7.2    Testing and Experiment Approach (Mohit / Brendan)

The tests for the Autonomous Secure Delivery system had different criteria for each used case. The testing strategies were implemented so the team could run unit and integration test cases seamlessly.  All components needed to be tested during real time to be sure that they work in real environments and can respond to their surroundings. We also did full regression test cycles to ensure that previous requirements were not affected after integrating.  When testing through each component periodically the team makes sure that the requirement is met while also studying possible flaws or improvements.

**Unit Testing Included:**

IR Sensors

H-Bridge

Power Supply 1

Power Supply 2

PWM Motors (1-4)

Pixy Camera

**SJSUOne Development Board:**

Integration Testing Included

H-Bridge with Motors

Pixy Camera with Dev Board

H-bridge with Dev Board

IR sensors with Dev Board

**System Test:**

Autonomous Delivery Vehicle

## 7.3 Testing and Experiment Results and Analysis (Brendan)

The unit testing of each component was extensive since the team was not familiar with most of the components.  The IR sensors while easy to use were hard to calibrate and decipher their outputs.  This meant that this unit test took more time than expected.  The motors were simple to test and only required extra testing for different speeds.  The pixy camera was already given a complete GUI interface so unit testing was simple.

Integration testing was much more rigorous as expected.  Integrating the motors with the Board and H-Bridge completely changed how the team planned on controlling the movement of the vehicle.  The Pixy Camera was taken out for more unit testing to increase reliability. Once we received consistent outputs and wrote application code for it, we incorporated it back into the rest of the design.

The final systems test while tedious was much easier than expected due to the extensive testing prior to this point.  Once the system was put together the vehicle was put through different use cases. Multiple use case test were ran to test the robot’s ability to complete required tasks.  Although this require some calibration of the motors and sensors the changes were simple and did not edit the schedule of the project.

# Chapter 8 Conclusion and Future Work

## 8.1 Conclusion (Mohit)

In conclusion, our group was able to code and implement all the peripherals for our autonomous secure delivery vehicle. Our team was successfully able to build each component for the autonomous car as well as code everything ourselves in an embedded environment. A lot of hard work and effort was put into this project throughout the semester and the result was well worth our efforts. All though there are difficulties with every project, our team had our own conflicts and resolutions. Some group members were not contributing as much as other team members.

Our goal was to create a robot that autonomous delivers a package from point A to point B seamlessly. There are quite a few real world applications for our project, which including delivering goods in a workplace setting to delivering products for various purposes. The delivery robot that we have established is not big, but a miniature size to show that it possible to create a robot for this purpose. After the creation of the miniature size robot, this proves this can be implemented on a larger scale for real life applications.

Our autonomous secure delivery vehicle is designed to deliver packages without human intervention. The vehicle is designed to transport a user’s package with the help of an image recognition camera and the SJSUOne development board.  The Pixy CMU camera can be taught to recognize predetermined objects. With the use of our algorithm and other peripherals, it is used to guide the robot to its destination. The chassis of the robot will include a secure compartment to store the user payload from start to destination.  Automating the delivery process minimizes cost and complications, while increasing efficiency and reliability.

## 8.1 Future Work (Brendan)

The future work for this project could be endless to improvise the robotic movements, camera integration, and object avoidance although the main limitation of the current project is scalability. For future work, the main goal would be to take the working proof of concept vehicle and make the camera and processor modular. The goal of this would be to have a camera and processor module that could connect to multiple different vehicles. This would limit the product cost of the product and would allow customers to build this capability into different applications. Purchasing just the module at a lower cost would allow customers to integrate the product into different robot chassis for different purposes. Since the main functionality of the product is the object recognition to deliver packages the chassis the camera and processor are connected to each other. The functional requirements can be easily met since both units will be modular and easy to move.

# References

1.     M. Gerla, E. K. Lee, G. Pau and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," Internet of Things (WF-IoT), 2014 IEEE World Forum on, Seoul, 2014, pp. 241-246.  
doi: 10.1109/WF-IoT.2014.6803166

<http://ieeexplore.ieee.org/document/6803166/?arnumber=6803166>

2. Isaac Kaminer, Antonio Pascoal, Eric Hallberg, and Carlos Silvestre.  "Trajectory Tracking for Autonomous Vehicles: An Integrated Approach to Guidance and Control", Journal of Guidance, Control, and Dynamics, Vol. 21, No. 1 (1998), pp. 29-38.

<http://dx.doi.org/10.2514/2.4229>

3. Jaggars, Rob. "What Are Autonomous Robots?" Industrial Robots for Sale at RobotWorx. New and Used Robots: Motoman, Fanuc, Kuka, ABB, and Others. Robot Wrox, n.d. Web. 10 Mar. 2017.

<https://www.robots.com/articles/viewing/what-are-autonomous-robots>

4. Whelan, Robbie. "Fully Autonomous Robots: The Warehouse Workers of the Near Future." *The Wall Street Journal*. Dow Jones & Company, 20 Sept. 2016. Web. 09 Mar. 2017.

<https://www.wsj.com/articles/fully-autonomous-robots-the-warehouse-workers-of-the-near-future-1474383024>

5. Wurman, P. R., D'andrea, R., & Mountz, M. (2008, March 22). Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses. AI Magazine,29(1), 9-20.

DOI: <http://dx.doi.org/10.1609/aimag.v29i1.2082>