VALET:
High Level Design Document

William Chestnut, Alden Kane, Jonathan Peterson, Perfect Mfashijwenimana,
Sean Scannell

Table of Contents

1 Introduction

Wouldn't it be nice if there was some sort of ordering system that could deliver you an item in a matter of minutes? Summon a package, pizza, or parcel with the click of a button. Well, this dream can become a reality with the VALET.

VALET (VAriable Location Electronic Transport) is as small electric vehicle that can autonomously pickup and dropoff items all through the power of GPS, optical sensing, LIDAR, and radio frequencies. The goal of this project is for VALET to successfully deliver it to a provided location autonomously, while avoiding obstacles.

2 Problem Statement and Proposed Solution

Leaps and bounds have been made in parcel delivery. The pizza man has evolved into UberEats, Postmates, and other ordering solutions. Yet, these solutions still require manual labor — and only are pertinent where well-paved roads exist.

Further innovations in the delivery space will come with autonomous drone delivery, whether it is land-based, road-based, or aerial. These all present opportunities for doorstep delivery, or venture where there are not paved roads for vehicles (e.g. college campuses, large buildings).

These delivery methods need to have the ability to navigate from A to B, while avoiding obstacles and ultimately managing a handoff between the robot and customer.

We propose VALET, an autonomous drone-delivery solution for parcels that operates best in small environments, such as college campuses or urban areas. VALET will make use of a sensor suite with optical, LIDAR, and GPS sensors. It will be capable of navigating between set GPS waypoints, while avoiding local obstructions (walls, trees, humans, trash cans, etc.). Upon arrival at its destination, the customer will be able to unlock VALET to retrieve the parcel using a QR code.

3 System Requirements

In order for VALET to be successful, some system requirements have to be realized. The system's success is dependent upon its ability to receive and to navigate to specific GPS coordinates, while avoiding obstacles, communicating with several sensors, controlling motor inputs, and supplying required power to the devices.

Dispatching VALET and communicating with a smart device: the car has to be directed to a predetermined location for delivery. This is demonstrated through sending a signal to the device with GPS instructions from an external transmitter. The relayed information has to contain a GPS address to start directing VALET. Since VALET requires image processing capabilities, a Raspberry Pi will be used.

VALET system is required to execute GPS path following and object detection and avoidance: VALET adds value to the delivery industry through introducing a smart system that adds an extra layer of precision by employing GPS and a sensor suite for accurate location delivery. GPS has to communicate with the microcontroller through UART and I2C interfaces that are available on dsPIC33.

To enable path following and object avoidance, VALET will employ an accurate technology for distance measurement using laser light. The system has to work hand in hand with Raspberry Pi Camera module that is employed in object detection. The system is required to complete the delivery by scanning a QR code at the arrival to the delivery location.

Regarding the vehicle and drivetrain of VALET, the motor controller is required to accept control signals from the microcontroller and output correct current and voltage ratings to the motors. The motor sensing system uses the magnetometer and accelerometer for orientation towards waypoint and maintaining directional heading.

For user interface, we will use Python based application to send coordinates and start signal to the VALET through WiFi. The system is required to receive a stop signal for safety purposes either from the same Python application or an embedded RF receiver and relay combo.

4 System Block Diagram

4.1 Overall System:

The overall system consists of small autonomous, 4-wheel-drive rover carrying a suite of electronics. When supplied with a specific GPS location, the vehicle will be able to autonomously navigate to the location while avoiding obstacles along the way. Upon arrival, it will validate it has reached its destination, via QR code recognition from a customer, and complete delivery of the package. The system can be divided into several subsystems with dedicated functions. These include the sensor suite, logic controllers, battery pack, mechanical rover, and motor control system. They main controllers will sit at the center of the overall system and provide the computing power for the entire system.

The purposes of the subsystems are as follows: The input side contains all of the different sensors that we will use, including LIDAR, Pi-camera, accelerometer and a GPS chip. The movement subsystem will include the electric motors and control systems associated with them. The Power system will be built from a battery. The main controller will be the Microchip that we will integrate on a custom PCB with other sensors and the Raspberry-Pi, which we will use for interacting with the Pi-camera and for wireless communication.

Refer to Appendix A for a system block diagram.

4.2 Subsystem and Interface Requirements:

Sensor Suite

Our system will require a variety of sensors to have enough information to achieve our goals. We envision that this suite will include a LiDAR sensor, a Pi-Camera, accelerometer, and a GPS chip.

LiDAR

The LiDAR Sensor: this system will primarily be for object avoidance, giving us the ability to judge distance from objects and determine if there is a potential issue with forward movement. Our initial selection for the LiDAR sensor will be the LiDAR-Lite v4 LED Rangefinder from Garmin. This sensor communicates using an I2C connection. The link to the item is here:

Garmin LIDAR-Lite v4 LED | Distance Measurement Sensor

Vision Subsystem

The Pi + Py-Cam combo will be a secondary method of determining object determination. The Pi-Camera would also provide the ability to read a QR-code to validate the VALET has reached its delivery location. In real life, this QR code could be posted on the recipients door, or displayed on the phone and help up for the VALET to see if delivery is to the correct person. Our Raspberry Pi and Py-Cam have already been purchased and a vision based QR code reader and object detection program running on the Pi. We plan to relay this information to our microcontroller via some yet to be determined serial interface.

GPS

The GPS chip will be used to do large-scale navigation between locations. This provides the guide on general direction that we would want to travel. The GPS chip that we plan on using is the TESEO-LIV3F-Tiny GPS Module from STMicroelectronics. The link to this item is:

Teseo-LIV3F - Tiny GNSS module

However there are many options for purchase. This sensor can communicate using either I2C or UART. It includes a built in FLASH memory, in order to store GPS waypoints and traveled paths. This will allow us to potentially implement some sort of learning and path optimization for repeated deliveries. It will also allow the vehicle to more efficiently return to its home following a delivery. Additionally, the GPS module supports geofencing, enabling us to implement "safe areas of operation" and "no go zones."

Accelerometer

The Accelerometer/Magnetometer will provide the system with an understanding of its own speed and orientation. This is essential when navigating via GPS, and for maintaining safe operating speeds/limits. This system will prevent VALET form accelerating endlessly downhill, which could have some catastrophic results, alert the dispatcher if crashes or flips over, and allow it to determine the correct direction of travel to get to a GPS waypoint. The sensor that we will use for this function is the ISM303DACTR from STMicroelectronics. This sensor communicates using either I2C or SPI. The link to this product is:

https://www.digikey.com/product-detail/en/stmicroelectronics/ISM303DACTR/497-17727-1-ND/7927709

Motion System

The Motion system is going to be comprised of the motors and wheels for movement of the VALET system. We plan on purchasing brushed DC motor controller to handle low level logic and motor control signal generation. For the purpose of this project, we did not want to get stuck working on highly technical motor control (PWM, current sensing, EMF measurement) if we could possibly avoid it. Nor, as electrical engineers did we think it was appropriate to dedicate time (and our limited knowledge) to designing the mechanical side of the vehicle (wheels, chassis, gearing). Fortunately, there are several robot kits available on the market that fulfill our requirements. Our selection for motors and chassis are combined in the 4 Wheels Scout Platform Robot Kit. The link to this item is:

4 Wheels Scout Platform Robot Kit

For the motor controller, our selection is the Sabertooth Dual Regenerative Motor Driver from Robotshop. This motor driver communicates using I2C. The link to this product is:

https://www.robotshop.com/en/sabertooth-dual-regenerative-motor-driver.html

Refer to Appendix B for block diagram of the motion system.

Power

For our battery subsystem, we need to be able to supply power to all of our different subsystems as well as have enough power to keep the motors running for long enough to deliver the package a distance we specify. We believe that a good choice would be the 3 of the 6V, 2800mAh Battery Packs from Robotpack. Two batteries in series will be used to power the motors (12V Brushed DC), while the remaining battery will power all auxiliary devices. The link to this product is:

https://www.robotshop.com/en/6v-2800mah-nimh-battery.html

Control System:

For our control system, we need adequate computing power to control all of the different subsystems. Therefore we will use a PIC32 microcontroller integrated on custom built PCB (designed with Eagle). This will allow us to specify the number of I2C, SPI, or GPIO ports that we want and integrate other components such as the GPS and accelerometer modules. This will be the highest priority controller on the board.

The wireless communication between user and the control system will be handled by the Raspberry Pi, which will then feed the information to the PIC32. The Raspberry Pi will also interface with the Pi-Camera.

Software System

The software system that we need to implement will have to be able to take in all of the data from the different sensors, make a decision and send a signal to the motors to move. This will include an ability to read from several different sensors in quick succession, some having different communication types. Different sensor input includes orientation, object detection, and vision input. With this data, the computer must then plan its next path forward and follow that path, which will be sent to the motor controllers.

4.3 Future Enhancement Requirements

Options for future enhancements include a method of secure storage, where the items inside would only be released if the QR code associated with it was displayed. This would involve some form of lid with a motor that would open up only the necessary sections. An additional option for small packages would be multiple containers, each with an individual lid. This would secure each customers package from being tampered with by other customers. Other container improvements could include package specific containment units, like heated containers for coffee, accommodations for delicate items, or securing for unusually shaped objects.

Other improvements to consider are different battery packs that could offer better performance in cold temperatures or have longer lifetimes, all for higher costs. Additionally, it would be interesting to try different types of motors for different circumstances. This would be especially helpful in an environment like Notre Dame, where the seasons have such drastic changes.

For a long-term implemented business plan, we would create a fully realized version of the customer interface using either a mobile application and or website. This would send the drop-off coordinates of the GPS signal and the QR-code so the customer could access their package.

5 High Level Design Decisions

Requirement	Part name	Justification
GPS	TESEO-LIV3F - Tiny GPS Module.	Capable of interface with PIC32 through UART and I2C. The module possesses built in flash memory to remember previous paths.
LiDAR	LIDAR-Lite v4 LED Rangefinder	Operates in a range of 10 m and is accurate in laser light measurement. Unit cost was a major factor.
Motor and Driver	4 Wheels Scout Platform Robot Kit H-Bridge Motor Driver	Ability to talk to the microcontroller and source correct voltage and current using simple packetized serial commands.
Image processing	Raspberry Pi	Supports a Camera for image

		detection and has WiFi and Bluetooth modules for pinging VALET.
Power Supply	2800mAh Battery Packs from Robotpack	Supplies constant 6V (12V 2 with 2 in series) and meets 4.9A stall current requirement of motors. At average 2A current draw, enough power for 30 minutes of run time.
Controller	PIC32	Supports multiple communication interfaces, sufficient processing, and team familiarity with the software and interface

6 Open Questions

1. Implementing The GPS

- ➢ Our group has not worked with a GPS device before. We are going to have to figure out how to implement a GPS device and also make it connect to a communication device, like a PIC32. Connecting the GPS to a communication device so it can move to a certain location is going to be a challenge that we have to solve. A WiFi or a bluetooth signal might need to be used to send the VALET the coordinates of the desired travel location.
- ➢ Also, we have to decide whether or not there is going to be a single coordinate that the VALET has to travel to, or a multitude of coordinates that creates a path for the VALET to follow.
- ➢ The GPS has memory so it can store the coordinates of a particular path. Also, if the VALET cannot reach a coordinate because an obstacle is in the way, we need to create logic that can move past that obstacle and travel to the next coordinate point.

2. LiDAR

- ➢ LiDAR obstacle detection is going to be a must if we want our vehicle to transport an object successfully. We are most likely going to have to connect the LiDAR to the pic32 and use an I2C interface.

3. Stop or Reset Functions

- ➢ When the VALET needs to be powered off (or if something goes wrong), there needs to be a function that stops the VALET from moving. Our group still needs to figure out how we want to implement this function. The VALET could receive a wifi, bluetooth, or RF signal once it reaches the desired location, and this signal could stop the VALET from moving
- ➢ Our preliminary idea (for the greatest safety) is an RF relay. The eMylo 12V DC RF Relay Switch is a viable option

4. Implementing Control Logic

- ➢ We still need to figure out where all of the control logic for our VALET project is going to be programmed on. We think that most of the control logic is going to be programmed onto the Raspberry Pi, but if the LiDAR uses the PIC32 and I2C, then our group is going to have to decide whether to do control logic on microcontrollers, the Raspberry Pi, or both devices.
- ➢ With the camera and other devices being connected to the Pi, we are hoping that we can do most of the control logic with the PIC.

7 Major Component Costs

Part Description	Manufacturer	Supplier	Part #	Qty	Unit Cost	Total
Rover Chassis/Motors: Scout Robot Kit	ServoCity	SevoCity	N/A	1	169.99	169.99(1)
Motor Controller: Sabertooth Dual Regenerative Motor Driver	Dimension Engineering	Robotshop	RB-DIM-42	1	77.95	77.95 (2)
Lidar-Lite v4 LED	Garmin	Garmin or SparkFun	010-02022-00	1	59.99	59.99

Raspberry Pi 3	Raspberry Pi Foundation	Stock Part	Model B	1	$0.00	$0.00
Pi Cam	Raspberry Pi Foundation	Stock Part	Module v2	1	$0.00	$0.00
GPS	STMicroelectro nics	DigiKey	TESEO-LIV3 F	1	$9.00	$9.00
Magnetometer/ Accelerometer	STMicroelectro nics	DigiKey	ISM303DAC TR	1	$2.05	$2.05
PIC32MX	Microchip	Microchip	TBD	1	$0.00	$0.00
Battery: 6V, 2800mAh, NiMH Battery	Lynxmotion	Robotshop	RB-Sta-08	3	$26.95	80.85
Charger: 6 - 12V NiMH / NiCd Smart Charger	Lynxmotion	Lynxmotio n	USC-02	1	$24.99	$24.99
Total						$424.77

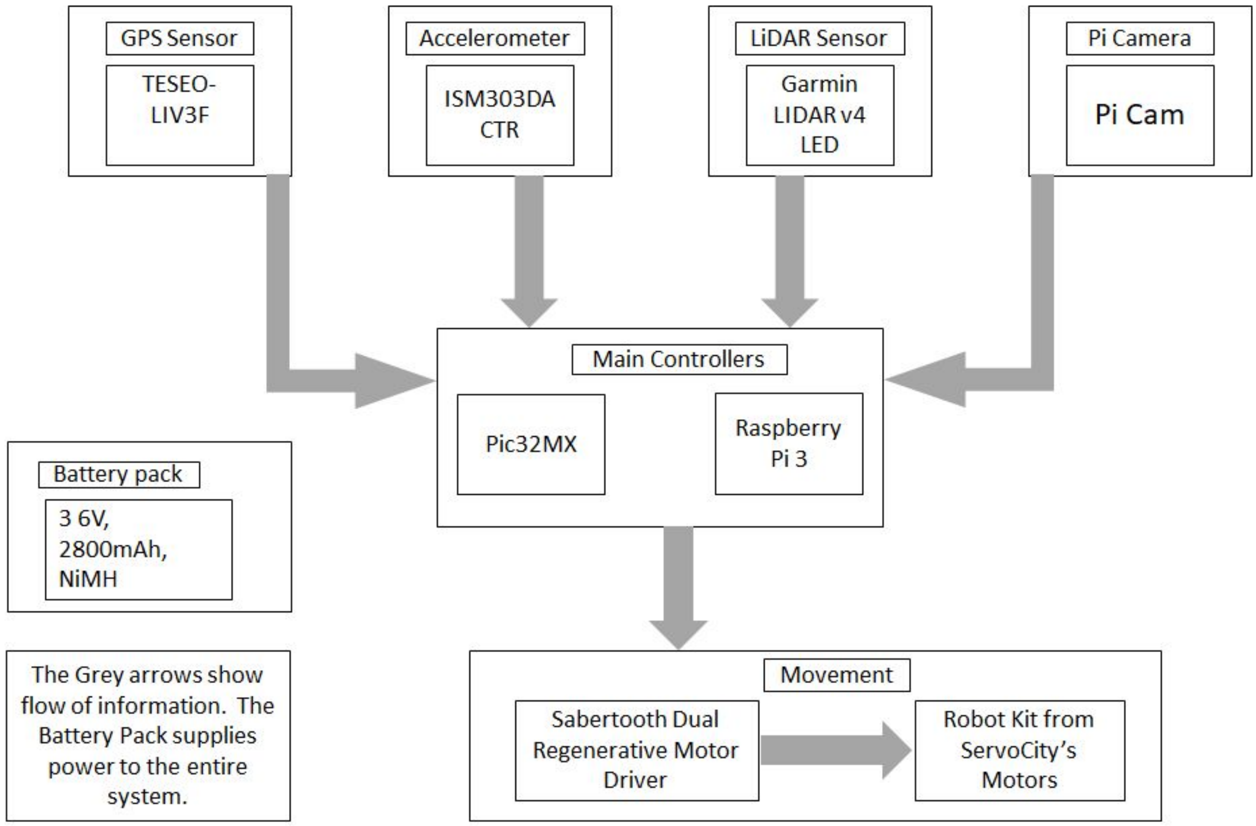
1. Servo City has a program for which the accept project proposals and upon approval, customers are given 15% off all necessary components.
2. RoboClaw 2x15A Motor Controller (V5E) is an option if feedback is desired. Chassis would have to be ordered with motors that include encoders. Incremental Cost: $36.00

NOTE: The cost of the controller PCB (~$50) and supporting circuitry was not included in this analysis.

8 Conclusions

VALET hopes to ameliorate some of the problems regarding delivery systems, like long wait times and the necessity of human labor. Our group will be using a motor kit that was found online, so most of the work for this project will be focused on how the vehicle can send and receive data, travel to a certain location, avoid obstacles, and deliver the desired item. We hope to have a functioning VALET system that can receive the coordinates of a desired location, travel to the location while avoiding obstacles, and deliver the items it is carrying successfully.

Appendix A



Appendix B: