FoodieRover:
The Food Delivery Robot

Group 29 Members

Mauricio Ferrari - Computer Engineering

Alexis Gyselinck - Electrical Engineering

Antonia Soto - Electrical Engineering

Chidoziri Maghiro - Electrical Engineering

Committee Members

Aman Behal - Ph.D.

Narrative Description

The objective behind this project is to cut down on time wasted waiting to receive our orders from the busy restaurants on campus (e.g. Chick-fil-A). As full-time students, we experience a restrictive time window for lunch due to our tight schedules so, when we spend 20 out of the 30 minutes we have loafing around in await for our meal, it takes away from what is supposed to be a midday break where we can recharge and relax as we enjoy what we're consuming. Rather, it becomes a stressful contest of how fast can you finish an entire combo and still have time to walk to your next class. Additionally, it would offer students and faculty alike the chance to be more productive as they won't have to remove themselves from their workspace to acquire sustenance.

FoodieRover is a small self-navigating robot that doubles as a food convoy. Our grandest goal for this project is to design FoodieRover to be able to detect objects. Obstacle detection is no small feat, especially when dealing with active traffic (moving objects). Examples of active traffic on our campus could range from human feet, squirrels, bikes, scooters, skateboards, and even golf carts. The issue here is finding the right sensitivity for our obstacle detection program. We certainly don't want FoodieRover to come to a lasting halt each and every time a foot cuts in front of it. Ideally, FoodieRover should only stop if the object is close enough for impact and not moving out of the way anytime soon. Static objects could include large rocks, parked vehicles, or even conversing humans. Our plan is to attach an array of ultrasonic sensors onto the chassis of FoodieRover and when their feedback is high enough and consistent enough then we can determine that we should come to a stop.

Figuring out when a detected object actually presents a notable obstacle is only one piece of the puzzle though. We want FoodieRover to not only be able to pick up any obstructions in its path but also be capable enough to route around the said obstruction. This is mainly a software level challenge and we plan to address this by developing an algorithm in conjunction with API calls to Google's Maps services. Google Maps allows users to send coordinates of their starting and ending locations for a trip in order to return an acceptable route.

This service coupled with the GPS dedicated hardware (e.g. magnetometer) will enable FoodieRover to come up with new routes for itself on the fly. Which leads us into our next goal of this project, which is to showcase our ability to integrate varying electrical components together. For the processes already described above we would require a microcontroller with wifi capabilities (possibly a microprocessor instead if the code is too heavy). Other components we have to consider are the battery pack, motors for the tracks, locking mechanism, and camera.

The locking mechanism is a feature we want to include within the food storage container so that the food won't be stolen. To couple this lock and key pad, we'd also like to create an interface for cellular devices that can communicate with FoodieRover (i.e. provide tracking information, notification when FoodieRover has arrived) for both the customer and restaurant to utilize. As for the camera, we're interested in implementing computer vision to enhance FoodieRover's obstacle detection capabilities but also plan on using it for surveillance purposes.

Autonomous delivery robots are already present in the market, although they do not dominate the space. They are a slow, but growing, sphere of technology with one of the industry leaders being Kiwibot. Kiwibot also operates on college campuses and similarly provides interfaces in which users can interact with their product. Kiwibot features cameras, 3 frontal and one rear wide, along with LTE and GPS capabilities.

Kiwibots are significant as they exhibit level 4 autonomy, which is achieved when the vehicle has full self-driving capabilities. However this is within limited ODDs (operational design domains) so outside of these constraints it is acceptable for the vehicle to rely on human intervention. Another aspect of level 4 vehicles is that they do not need to be continuously monitored. They should be competent enough where the driver can have trust that it will arrive at its location safely without hiccups. Likewise, my group aspires to develop FoodieRover to have level 4 autonomy.

Specifications

As mentioned above, FoodieRover aspires to have obstacle detection capabilities. In order to achieve this, proper sensory information about its environment is required. Therefore we are looking to provide FoodieRover with 8 ultrasonic sensors: 1 for each side (front/back/left/right) and then one for each shoulder. The idea here is that when the sensors are picking up strong feedback (indicating an obstacle), we can check the other ones to decide what the best route around the obstacle would be. In addition to this, we are also looking into implementing cameras and using computer vision to aid in obstacle detection and routing. Ideally we would have 4 wide cameras (one for each direction).

Other software specifications are the necessity of an application interface for users to interact with FoodieRover. The application will support two types of users: consumer and producer. The consumer interface needs to have options to put in orders, enter drop-off locations, and unlock the container. The producer side needs to be able to support commands to deliver food, return to base, and post food items.

After the application, the last key component of the software is its API capabilities. This would require FoodieRover to have a wifi-chip so that it can make the API calls. As of now, the two servers we’re looking to contact is Google Maps and the eventual server of the application.

In terms of performance, FoodieRover will be expected to travel distances over 200 meters per trip at a speed of 1.5 to 2 miles per hour. It will carry around 3 pounds of food kept fresh by insulating material. While delivering food the power consumption will be kept under 400 Watts. When not moving, the bot ideally will reduce its power consumption to under 50 Watts. The battery capacity will correspond to 1 to 2 hours of runtime with a battery that can be replaced and recharged. FoodieRover will adapt to the different conditions it will encounter while delivering food. Its hardware design will accommodate for the non-ideal scenarios it may encounter.

Furthermore, its dimensions will exceed or be equal to a cubic foot to provide enough space for a meal with a drink. The interior design of the container will prevent spillage. The electrical components will be kept safe and in place with custom frames.

Marketing and Engineering Requirements:

Marketing Requirement	Engineering requirement	Justification
7	The robot should have dimensions of >= 1 cubic foot	The size of the robot should be big enough to fit every component and extra weight.
2,9,3	The robot should be able to move at speeds of 1.5 to 2 miles per hour	The speed should be fast enough to where it can get to the consumer at a reasonable time but slow enough to read any interference
6,2,3	The robot should be able to travel 200 meters per trip	Distance should be reasonable to make any close routes and long enough in case routes change
4,7	The robot should be able to carry 3 pounds of food	Weight should be enough for food for 2 people
5,7,4	The robot should have insulating material that should keep food fresh	Food needs to feel and taste fresh
6,7,4	The robot should have a battery life that will correspond to 1-2 hours of runtime with a battery that could be replaced and/or recharged	Battery life long enough to get products from place to consumer
9,5,2,7,4	The robot should have space for electrical components to be kept safe	The electrical needs to be kept safely
9,7,4	The robot should have interior components that will prevent spillage	There should be cup holders
8,7,4	The robot should have 8 ultrasonic sensors and 4 wide cameras to be able to handle interference made from the outside world	The sensors help with navigation around obstacles
5,9,7,4	The robot should have a covering for food	Food needs to be safe from outside
9,7,4	The robot should have a keypad	So theft can't happen
9,7,4	The robot should have a lock	So theft can't happen
1,8,7,4	The robot should have a wifi-chip that can make API calls	To Navigate to where it needs to go

Marketing requirements:

1. **This Robot should move on its own**
2. **The robot should be able to move at fast enough speed**
3. **The robot should be able to travel good distances**
4. **The robot should be able to carry the weight**
5. **The robot should have something to keep food fresh**
6. **The robot should have good battery life**
7. **The robot can't be too small**
8. **The robot should be able handle stuff getting in its way**
9. **The robot should be able to keep the food safe**

Project Block Diagrams

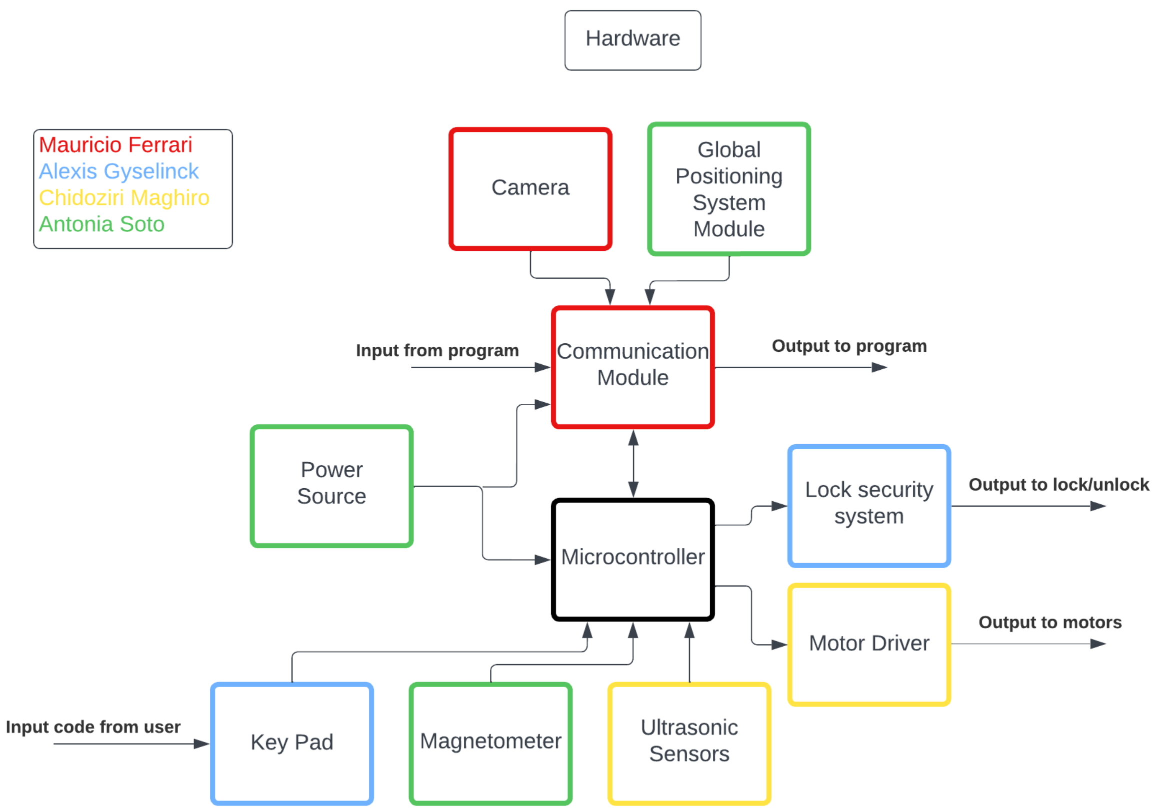
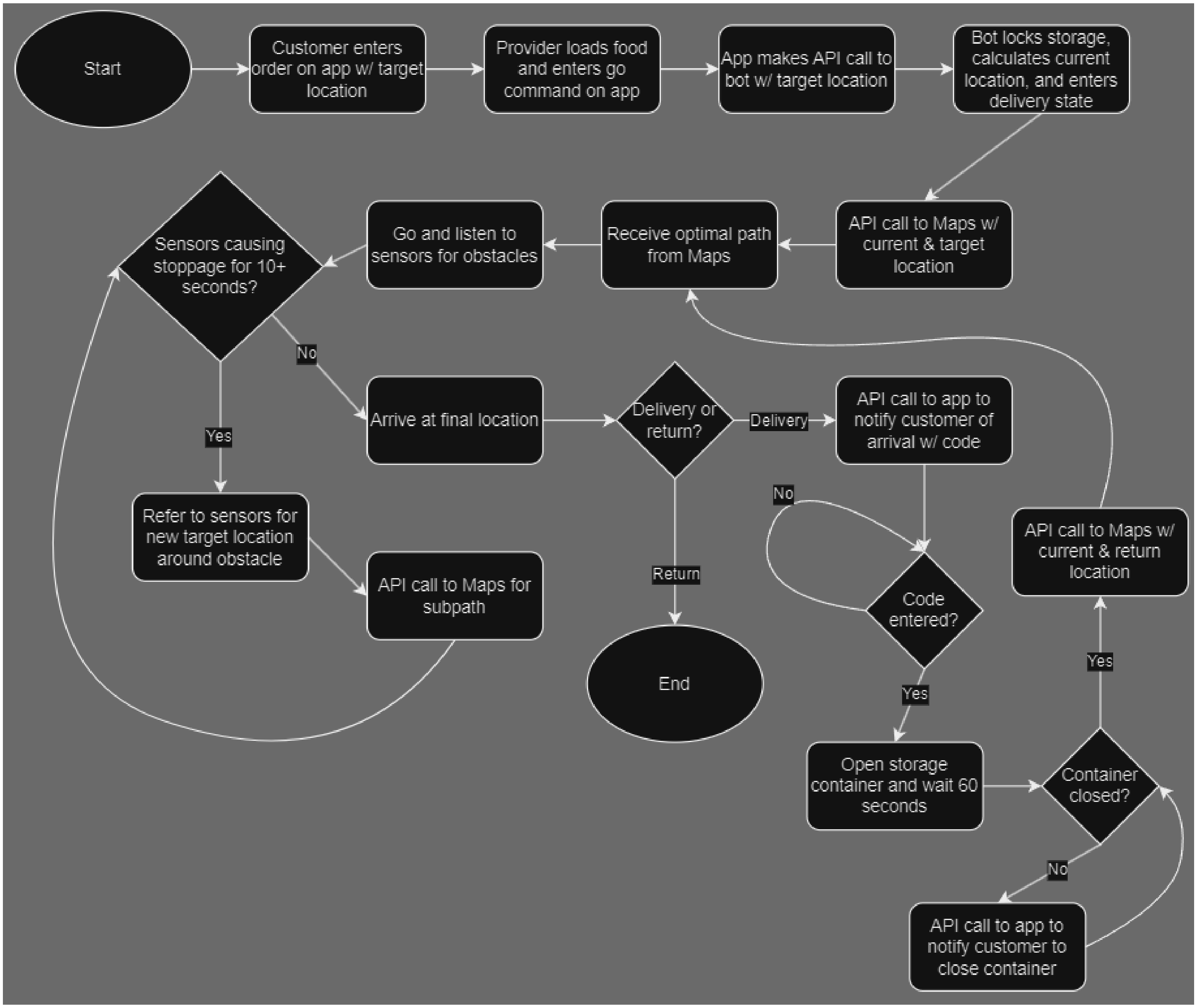


Diagram Legend

BLOCK NAME	BLOCK DESCRIPTION	BLOCK STATUS 9/15/23
Camera	Records visual images.	Research
Communication Module	Transmission module from and to program.	Research
Global Positioning System Module	Satellite-based location receiver.	To be acquired
Key Pad	Small set of keys.	Acquired
Lock Security System	Controls access authorization and provides security.	Research
Magnetometer	Instrument that measures magnetic forces; works as a compass.	To be acquired
Microcontroller	Compact integrated circuit designed to control operations in embedded systems.	Research
Motor Driver	Current amplifiers that bridge the controller to the motors.	Research
Power Source	Provides electrical energy to the circuit.	Research
Ultrasonic Sensors	Sensor that measures distance using ultrasonic sound waves.	Acquired

Software Flowchart:



Budget and Funding

ITEM	QUANTITY	PRICE ESTIMATE	TOTAL COST
Camera	4	$14 for 6	$14
Magnetometer	1	$10	$10
GPS module	1	$25	$25
Motors*	2	$25	$50
Robot chassis	1	$100	$100
Microcontroller	2	$30	$60
Motor driver	2	$15	$30
Ultrasonic sensor	8	$9 for 5	$18
Key pad	1	Acquired	$0
Power bank	1	$40	$40
Magnetic lock	1	$20	$20
Cooler	1	$30	$30
API	1	$30	$30
		Total cost:	$427

* Many chassis' come pre-equipped with motors.

We expect much of the funding for this project to come from our dedicated Committee Members. Self-funding from the team members will also be supplemented if needed.

Project Milestone

Task	START DATE	DUE DATE	%COMPLETE	NOTES
Group Discussions	9/5/23	5/1/24	5%	Group meets every week to talk about project
Parts for Prototype Ordered	11/5/23	1/20/24	0%	Parts for Prototype will be ordered
Parts for Prototype Received	1/20/24	1/20/24	0%	When the Parts are received
Assembly of Parts	1/21/24	2/2/24	0%	Time for which the Robot will be built
First Tests	2/3/24	2/9/24	0%	First wave of tests for robot
Changes to Prototypee Made(1)	2/10/24	2/13/24	0%	Learned from mistakes from first tests
Second Tests	2/14/24	2/18/24	0%	Second wave to text changes made from first mistake
Changes to Protoype Made (2)	2/18/24	2/22/24	0%	Learned from mistakes from second test
Third Tests	2/26/24	3/11/24	0%	Third wave to test changes made
Final Changes Made	3/12/24	3/20/24	0%	Last minor changes made for finished product
Last Tests	3/21/24	3/31/24	0%	Final test for prototype
Final Demonstration	4/1/24	Senior Design demonstratio n day	0%	Demonstration of robot