SDDEC18 – 22 Autonomous Lawn Mower

Members: Andi Li, Bryton Hayes, Grant Duncan, Joel Seaser, Sam Tinklenberg
Advisor: Dr. Phillip Jones
Client: Ryan Marrion - Micron Technology

Introduction

Problem

The problem we intend to solve concerns the time and financial commitment required to upkeep a well-groomed lawn. There is a long list of reasons a certain individual may not be able to mow their lawn, ranging from lack of time to physical incapabilities. Someone who falls into this category does not have many options to get the job done, without hiring expensive, third-party help.

Solution

We intend on solving this problem by creating a safe, inexpensive, and reliable autonomous lawn mower which can mow someone's yard on a set schedule and require little to no to no upkeep.

Intended Users/Uses

Our intended user base is residential home owners who are either unable to mow their lawns due to physical limitations or user's who do not have the time to mow their lawn. Since the mower is hands free after initial setup, anybody with a internet connected device could schedule mowing times and see the status of the mower.

Requirements

Functional Requirements

- Algorithm to efficiently mow entirety of area given a mapped perimeter
  - Intelligent routes will ensure straight lines
- System to map perimeter of the lawn
  - Easy installation
  - Low power
- Object detection and avoidance
  - Use various sensors to accurately detect and avoid objects
  - Navigate around hazardous areas in a lawn
- Mobility through standard lawns
  - Drive up 30% grade
- Algorithm to detect and avoid safety concerns
  - Easy shutoff consistent with sensor stimuli
- Power efficiency
  - Mowing time to falls between 60-90 minutes
- Streamlined interfacing
  - Android app for communication and diagnostics
- GPS module for directional guidance and mapping

Non – Functional Requirements

- Mower reacts to input in a reasonable time.
- Mower must be cheaper than other commercial mowers.
- Ability to record server or communication errors.
- Data does not get lost when mower loses connection to network.

Operating Environment

- Residential
- Dry Lawn
- Water and Dust Resistant
- Slight hills

Technical Details

Software

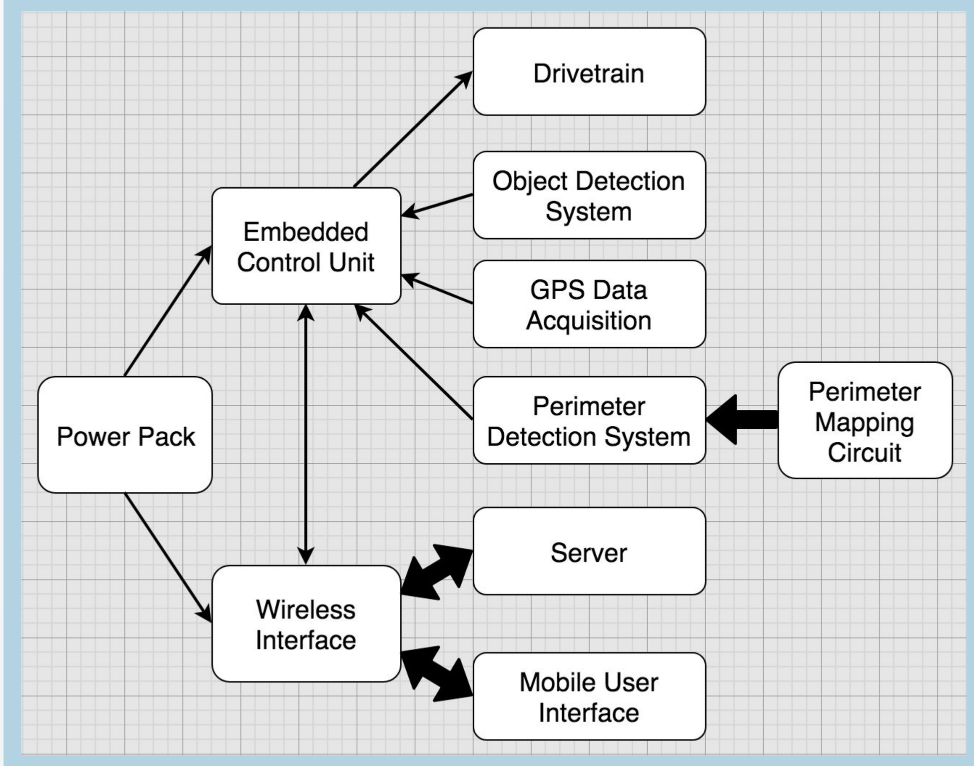
- C++ Arduino libraries
- 10Hz GPS WAAS data acquisition
- Serial data transfer between Arduino and Raspberry Pi
- REST Api on Raspberry Pi
- Python and Flask web backend
- MariaDB databases on Raspberry Pi
- SQLite DB on phone to store temporary information
- Android minimum API 26 (Oreo)
- Kotlin/Java used in Android app
- Google Maps API

Hardware

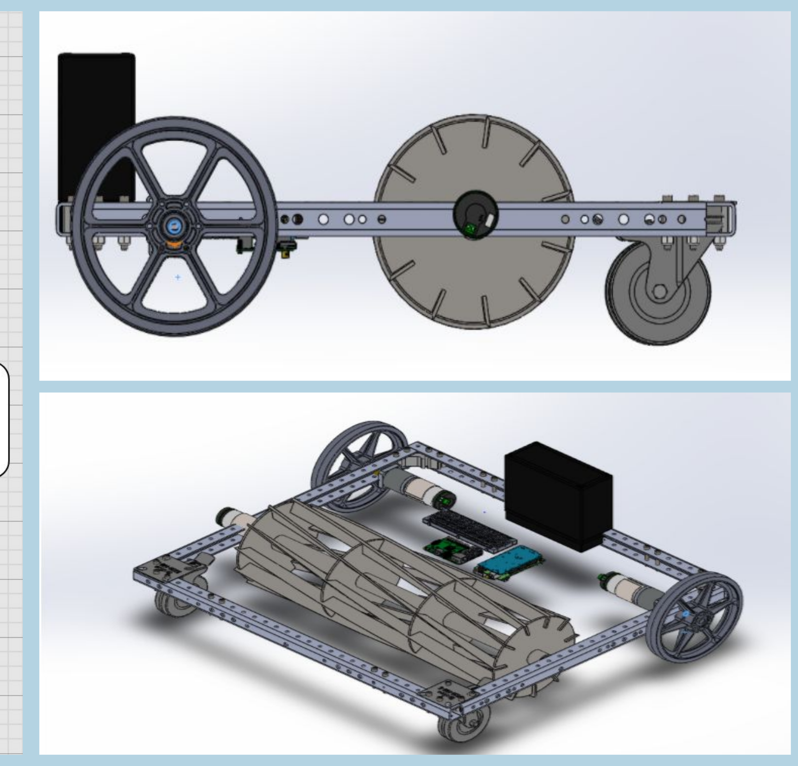
- 50W motors with 16-bit PWM control
- 16.8 Ah battery
- Max Velocity of 6ft/s
- Max weight of 40 lbs.
- 60 minute runtime
- Reel lawn mower blade
- Rear-wheel drive
- Hall-effect motor encoders

Design Approach

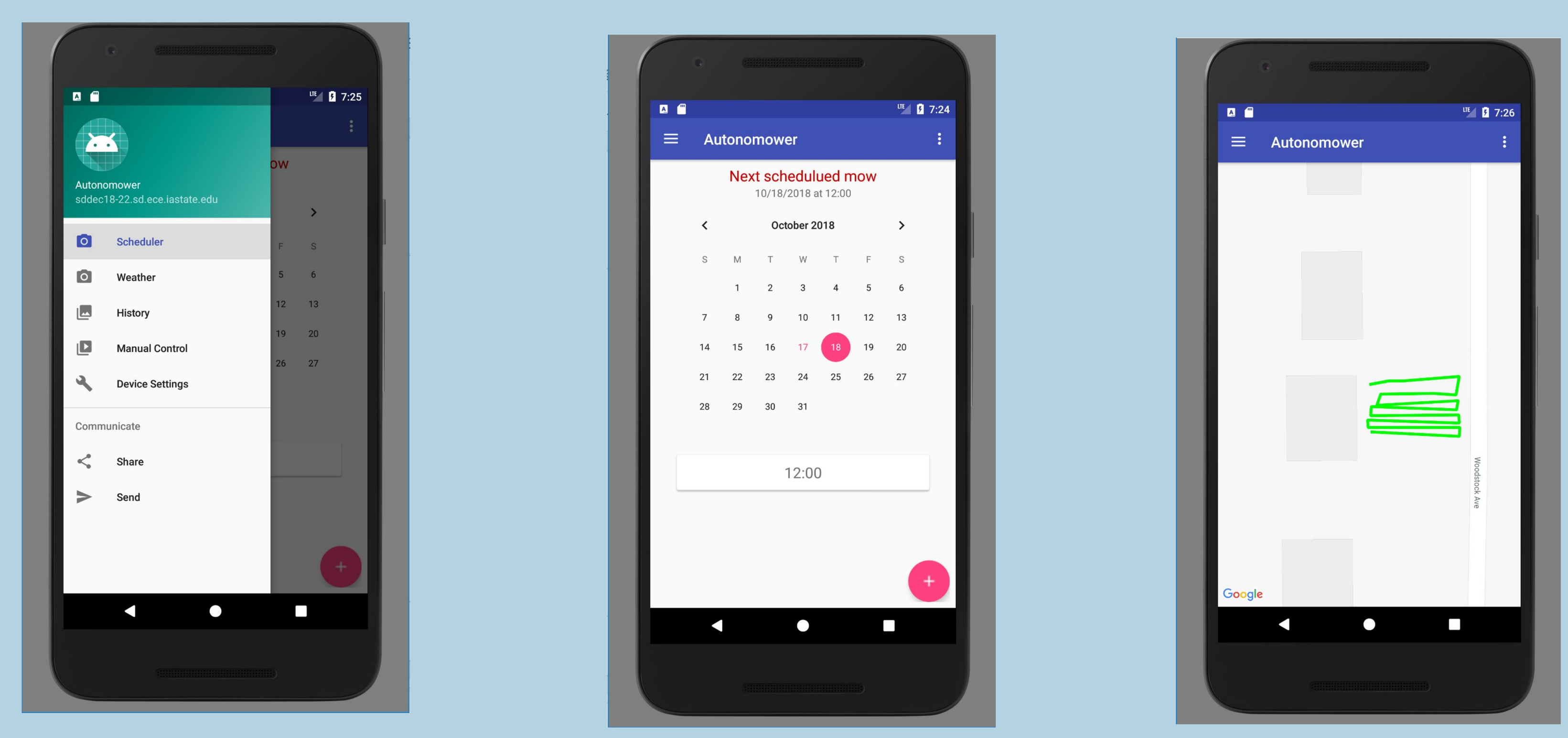
Conceptual Diagram



Concept Sketch



User Interface



Testing

Embedded Component Testing

- Motor Control
  - Run two drive motors and one blade motor using 16-bit PWM waveforms
  - Align speeds with with desired movement and cutting rpm
  - Obtain effective turning radius
- Perimeter Wire
  - Utilize data to obtain device distance and approach angle
  - Tune output frequency for optimal operation

Mobile Testing

- Databases
  - User selects new mow schedule, updates SQLite database
  - User syncs with mower, sends new data to mower database and updates entries history
- User Interface
  - Device rotation does not crash app or lose data
  - Data reflects current database values

Budget/Project Resources

Resource Allocation

- Drivetrain: $701.74
- Microcontrollers: $142.46
- Perimeter wiring: $44.87
- Sensors: $25.68

- Total: $915.15