

# Project Plan

# General Information

1. Project Name: IGVC - Intelligent Ground Vehicle Competition
2. Team members:
  - ▶ Viet Dung Nguyen (CSE) (dnguyen2016@my.fit.edu)
  - ▶ Jinwen Zhang (jzhang2015@my.fit.edu)
  - ▶ John Light (jlight2016@my.fit.edu)
  - ▶ Sijin Yang (yangs2017@my.fit.edu)
  - ▶ Peter Tarsoly (ptarsoly2016@my.fit.edu)
3. Faculty Sponsor: Dr. Marius Silaghi
4. Client: Dr. Ken Gibbs
5. Meetings with the Client:
  - ▶ 06/30/2019

# Goal and Motivation

The goal of the project is to build an unmanned autonomous robot to compete in IGVC in Michigan. The objective of the competition is to have the robot complete an outdoor obstacle course under a prescribe time within the speed limit of 1-5 mph while remaining on the lane and avoiding obstacles.

# Approach:

The key features of the software components are:

- ▶ Lane detection
- ▶ Obstacle detection and avoidance
- ▶ Mapping
- ▶ Motion planning

# Lane detection

Lane detection is necessary for the robot to follow lane specified in the competition.

# Obstacle detection

Obstacle detection and avoidance allow the robot to detect and move around obstacles in the path.

# Mapping

Mapping let the robot remember the positions of the obstacles and where it has been. This feature is necessary to ensure that the robot won't backtrack. With mapping, the robot can also navigating through previous obstacles without detection.

# Motion planning

Motion planning is a crucial feature of the robot. With the starting configuration, the robot must find the shortest path to the goal from data collected from lane detection and obstacles detection.



## Technical Challenges:

- ▶ Legacy code is not well documented.
- ▶ Some features are not implemented currently.
- ▶ Team members have no experience with Arduino.

# Milestone 1:

- ▶ Examine the options available for each features.
- ▶ Establish best practices for code.
- ▶ Examine the legacy code in order to reuse.
- ▶ Create requirement document
- ▶ Create design document
- ▶ Create test plan

## Milestone 2:

- ▶ Implement automated testing.
- ▶ Refactor the legacy code.
- ▶ Test for regression.

## Milestone 3:

- ▶ Implement and test motion planning.