MICHIGAN STATE
UNIVERSITY

Project Plan Presentation

Hardware in the Loop (HIL) Vehicle Simulator

The Capstone Experience

Team Bosch

Justin Armstrong
Luke Monroe
Aditya Raj
Alan Wagner
Christian Zawisza

Department of Computer Science and Engineering
Michigan State University

Fall 2021

From Students…
…to Professionals

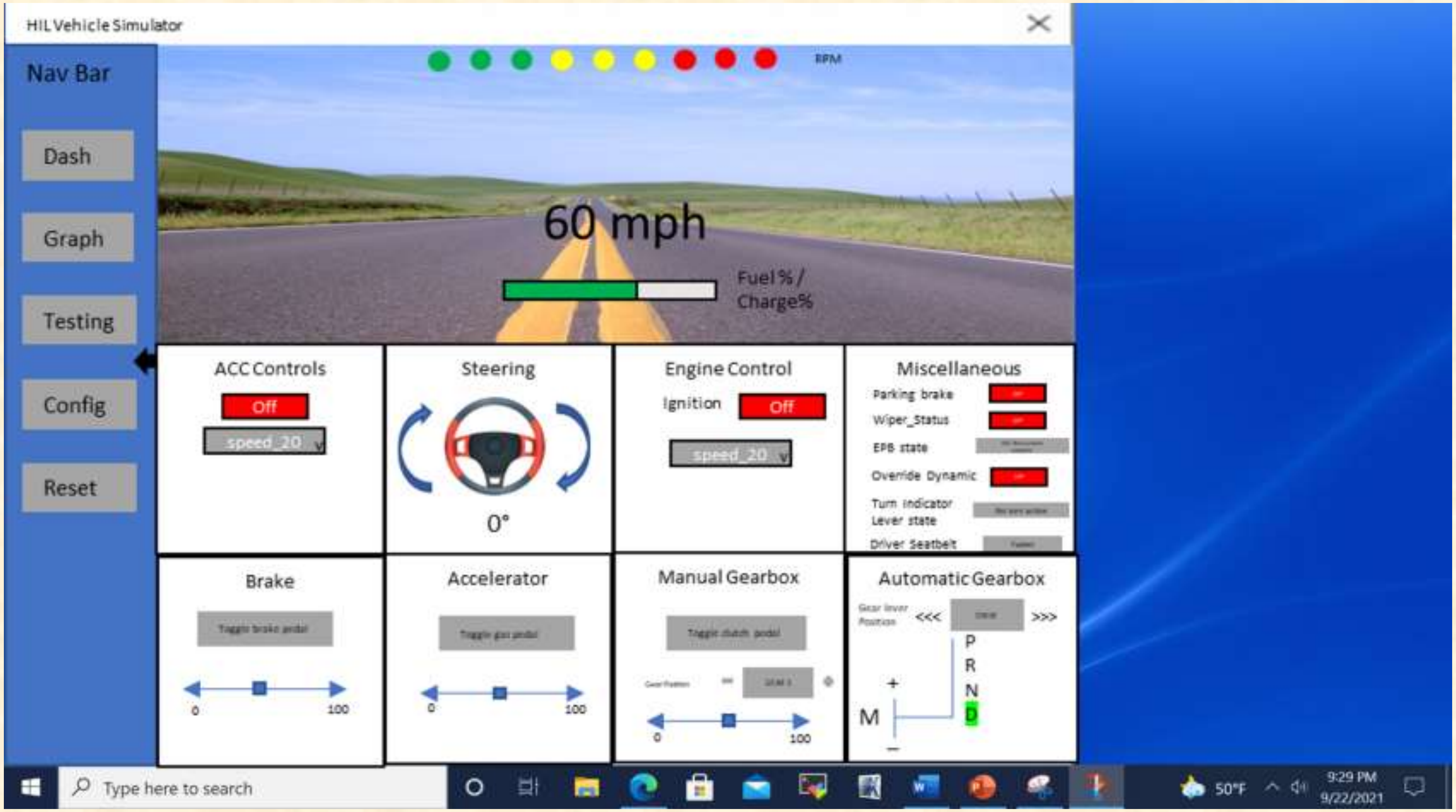
Functional Specifications

- Windows 10 application that will simulate a vehicles CAN Bus using cost-effective hardware
- Current hardware is too expensive and not available to all of Bosch's engineers at once
- Allows vehicle function such as acceleration, steering, braking, ACC and more to be simulated on cheaper hardware
- Ability to simulate different variations of vehicles configurable by the user

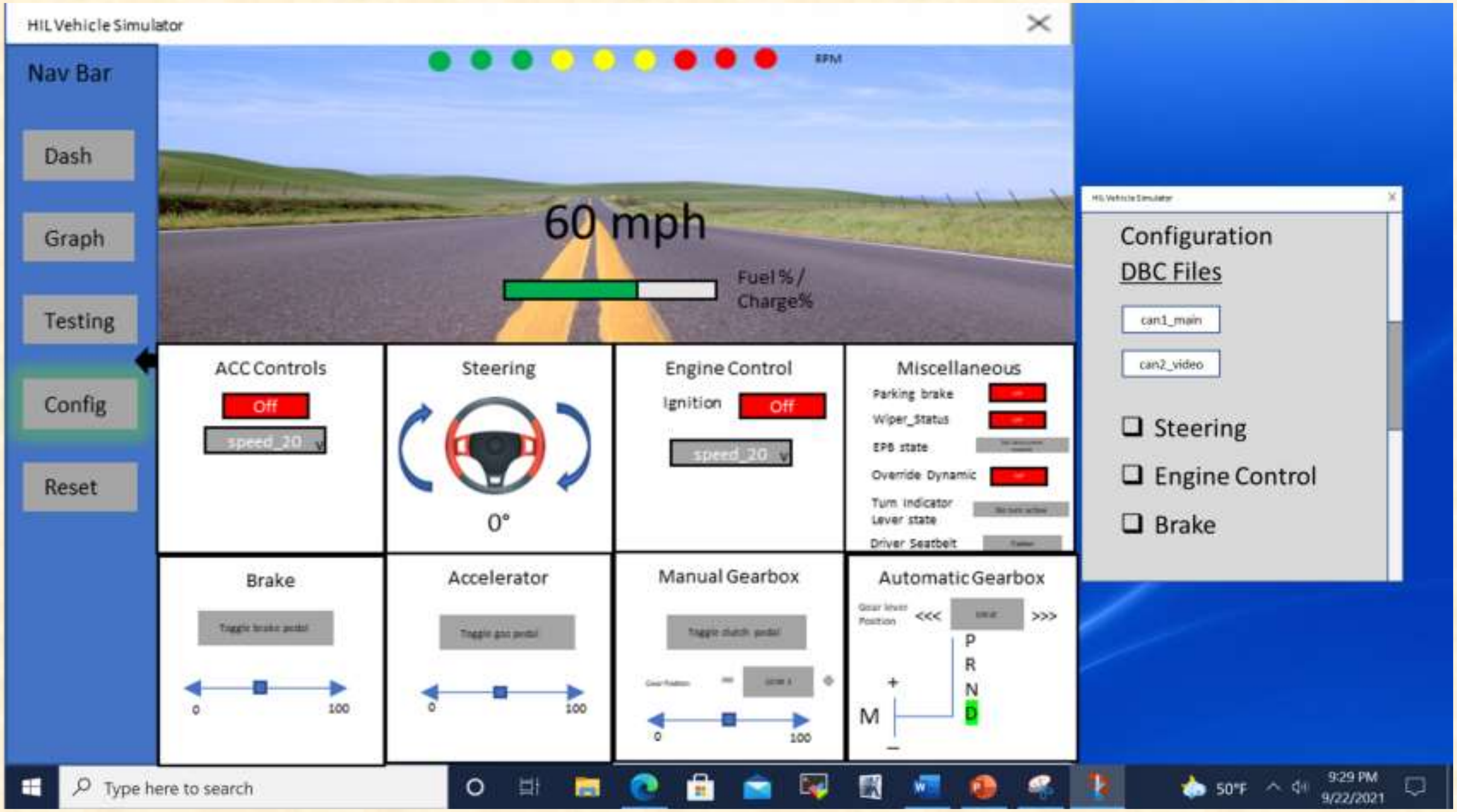
Design Specifications

- Easy to use GUI to allow user to control vehicle simulator
- GUI will allow user to configure vehicle’s base parameters to accurately simulate vehicle they wish to test
- GUI will show user a live graph of the data being sent and received to the vehicle’s CAN Bus
- User will also be able to create automated tests that will run a series of vehicle operations that the user specifies

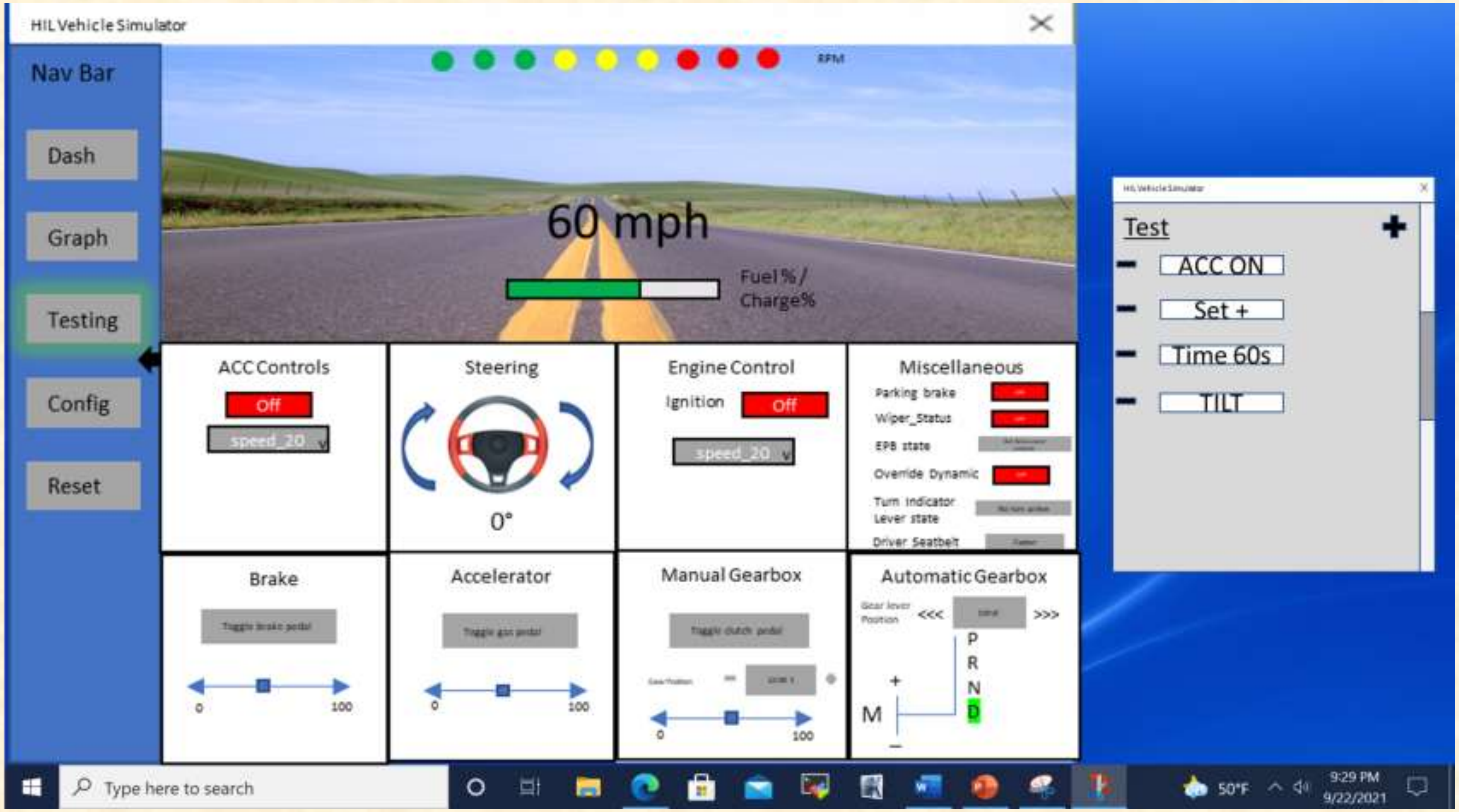
Screen Mockup: Main Dash



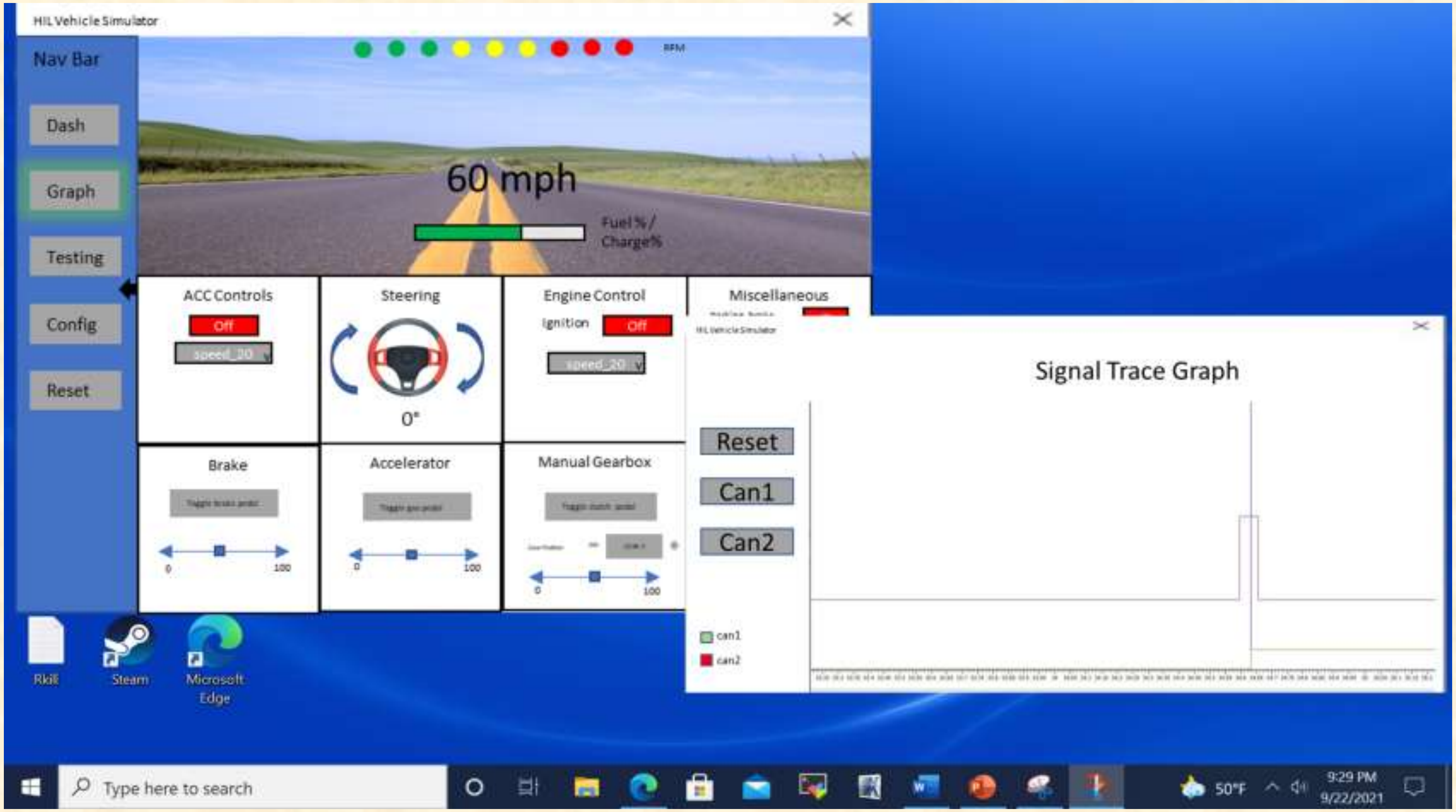
Screen Mockup: Configuration



Screen Mockup: Automatic Testing



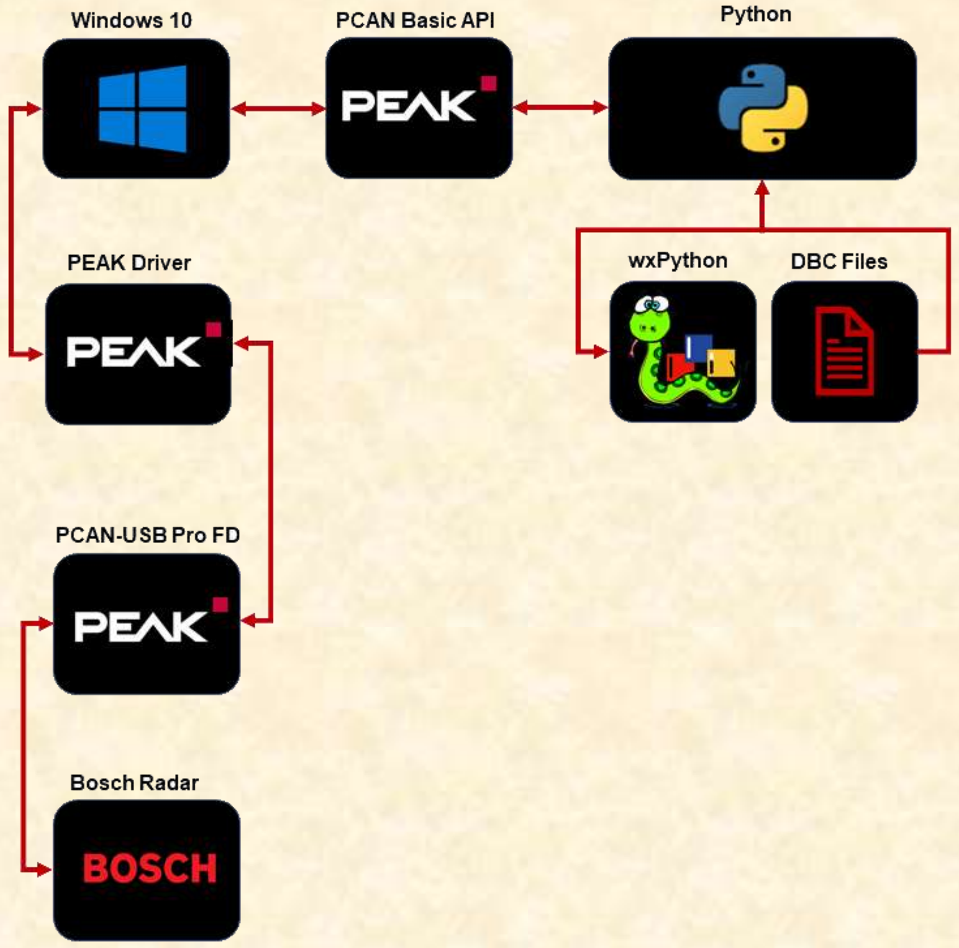
Screen Mockup: Signal Trace Graph



Technical Specifications

- Hardware Components
  - Bosch Radar: An ECU used for controlling NCC and ACC. Focal point of this simulation.
  - PCAN-USB Pro FD: Adapter that enables the connection of CAN networks to computer via USB
- Software Components
  - Python 3.9: an interpreted high-level general-purpose programming language.
  - wxPython toolkit 4.1: a python-based, cross platform GUI toolkit
  - PCAN-Basic API 3.3: a python-based API developed by Peak Systems

System Architecture



System Components

- Hardware Platforms
  - PEAK PCAN USB Pro FD
  - Bosch Radar
- Software Platforms / Technologies
  - Python
  - wxPython
  - PCAN Basic API

Risks

- Risk 1
  - Communicating with hardware through the PCAN Basic API and PEAK drivers
  - Familiarize ourselves with the hardware API and documentation to ensure smooth communication
- Risk 2
  - Creating a simple, easy to use GUI that will include all required functionality
  - Show prototypes to client as soon as possible to get feedback and find flaws through conducting real world tests that Bosch engineers would conduct
- Risk 3
  - Hardware could break physically, or we could brick the Bosch radar through software
  - Ensure all our communication with the hardware is correct and understand the signals we are sending to it. Handle the hardware with care
- Risk 4
  - Create a DBC parser that can parse any DBC file the user may submit and handle any errors
  - Ensure our parser works with the example DBC files given to use by our sponsor, test our parser against different variations of those files

Questions?

? ? ? ?

? ?

? ? ?