# HIGH WAY DRIVING

## 1.0 Objectives

The objectives of this project is to design a path planner using C++ that is able to create smooth and safe path for the simulated car to follow along a three lane high way with traffic. The path planner will be able to keep the car inside its lane, avoid hitting other cars, and pass slower moving traffic by using localization, sensor fusion and map data.

## 2.0 Background

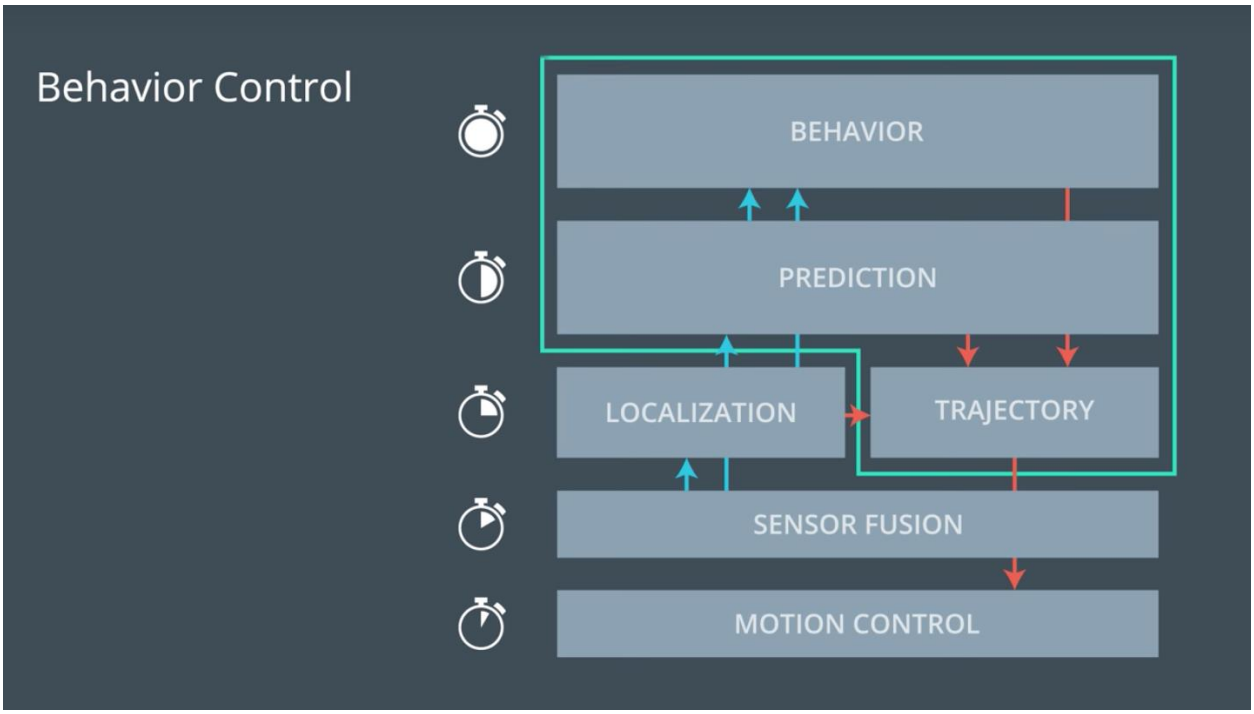
Path planning and navigation giving a map and a goal location, path planning is the calculation of the path that get robot where its need to go. The robot may encounter unmapped obstacle along the way that flick with the planned path. The robot use real time sensor reading to avoid obstacles and modify the original path.

## 3.0 Introduction

In this project, Udacity provided us with simulator that supply our C++ program with localization and sensor fusion data. There is also a sparse map list of waypoints around the highway. Our job is to write path planner in C++ that enable our car to safely navigate around a virtual highway with other traffic that is driving +-10 MPH of the 50 MPH speed limit. Also the car should not experience total acceleration over 10 m/s^2 and jerk that is greater than 10 m/s^3.

## 4.0 Path planning pipeline

There are three core components of path planning




1. **Prediction**: this involve predicting what other vehicles on the road will do in the next few seconds. Based on sensor fusion data we can locate the closest cars in front or behind our car and predict their behaviors.
2. **Behavior planning**: this involve deciding on a maneuver to execute in response to our car's goals and to our predictions about other vehicles. In this project, these decisions involve identifying possible lane change maneuvers such as change lane from the center lane to side lane or from side lane to the center lane.
3. **Trajectory generations:** this has to do with building a trajectories to execute the maneuver decided on by behavioral planner and then send these smooth trajectories to the motion controller.