SELF-DRIVING CAR ENGINEER NANODEGREE SYLLABUS

Course 1: Introduction

In this course, you will learn about how self-driving cars work, and you'll take a crack at your very first autonomous vehicle project - finding lane lines on the road! We'll also introduce the Nanodegree Program and the services we provide over the course of the journey.

Lesson	Title	Description
1	**Welcome**	Take your first steps towards becoming a Self-Driving Car Engineer! In this lesson, we'll introduce you to the program, help you discover the services we provide, and show you all the incredible projects you'll build. Get ready for an incredible 9 month journey!
2	**Finding Lane Lines**	In this first lesson, you'll get taste of some basic computer vision techniques to find lane markings on the road. We will be diving much deeper into computer vision in later lessons, so just relax and have some fun in this first week!

Project 0 : Finding Lane Lines on a Road
In this project, you will be writing code to identify lane lines on the road, first in an image, and later in a video stream (really just a series of images). To complete this project you will use the tools you learned about in the lesson, and build upon them.

Course 2: Deep Learning

Deep learning has become the most important frontier in both machine learning and autonomous vehicle development. Experts from NVIDIA and Uber ATG will teach you to build deep neural networks and train them with data from the real world and from the Udacity simulator. By the end of this course, you'll be able to train convolutional neural networks to classify traffic signs, and to drive a vehicle in the simulator the same way you drive it yourself!

Lesson	Title	Description
1	**Introduction to Neural Networks**	Learn to build and train neural networks, starting with the foundations in linear and logistic regression, and culminating in backpropagation and multilayer perceptron networks.

2	**MiniFlow**	In this assignment, you will build your own neural network library from scratch! Your library, MiniFlow, will behave much like TensorFlow, Google's deep learning library.
3	**Introduction to TensorFlow**	Vincent Vanhoucke, Principal Scientist at Google Brain, introduces you to deep learning and Tensorflow, Google's deep learning framework.
4	**Deep Neural Networks**	Vincent walks you through how to go from a simple neural network to a deep neural network. You'll learn about why additional layers can help and how to prevent overfitting.
5	**Convolutional Neural Networks**	Vincent explains the theory behind Convolutional Neural Networks and how they help us dramatically improve performance in image classification.
6	**Traffic Sign Classification**	You just finished getting your feet wet with deep learning. Now put your skills to the test by using deep learning to classify different traffic signs!

Project 1: Traffic Sign Classifier
You just finished getting your feet wet with deep learning. Now put your skills to the test by using deep learning to classify different traffic signs! In this project, you will use what you've learned about deep neural networks and convolutional neural networks to classify traffic signs.

Lesson	**Title**	**Description**
7	**Keras**	Take on the neural network framework, Keras. You'll be amazed how few lines of code you'll need to build and train deep neural networks!
8	**Transfer Learning**	Learn about some of the most famous neural network architectures and how you can use them. By the end of this lesson, you'll know how to create new models by leveraging existing canonical networks.
9	**Behavioral Cloning**	Put your deep learning skills to the test with this project! Train a deep neural network to drive a car like you!

Project 2: Behavioral Cloning
Put your deep learning skills to the test with this project! Train a deep neural network to drive a car like you!

Course 3: Computer Vision

You'll use a combination of cameras, software, and machine learning to find lane lines on difficult roads and to track vehicles. You'll start with calibrating cameras and manipulating images, and end by applying support vector machines and decision trees to extract information from video.

Lesson	Title	Description
1	**Advanced Lane Finding**	Train segmentation networks, which paint each pixel of the image a different color, based on its class. Use segmented images to find free space on the road.

Project 3 : Advanced Lane Finding
In this project, your goal is to write a software pipeline to identify the lane boundaries in a video from a front-facing camera on a car.

Lesson	Title	Description
2	**Machine Learning and Stanley**	Learn about classification, training and testing, and run a naive Bayes classifier using Scikit Learn.
3	**Support Vector Machines**	See how support vector machines work, and code one using Scikit Learn.
4	**Decision Trees**	Learn about decision trees, including entropy and information gain.
5	**Vehicle Detection and Tracking**	In this lesson you'll learn how to detect and track vehicles using color and gradient features and a support vector machine classifier.

Project 4 : Vehicle Detection and Tracking
In this project, your goal is to write a software pipeline to identify vehicles in a video from a front-facing camera on a car.

Course 4: Sensor Fusion

Tracking objects over time is a major challenge for understanding the environment surrounding a vehicle. Sensor fusion engineers from Mercedes-Benz will show you how to program fundamental mathematical tools called Kalman filters. These filters predict and determine with certainty the location of other vehicles on the road. You'll even learn to do this with difficult-to-follow objects, by using two advanced techniques: the extended Kalman filter and the unscented Kalman filter.

Lesson	Title	Description
1	**Introduction and Sensors**	Meet the team at Mercedes who will help you track objects in real-time with Sensor Fusion.
2	**Kalman Filters**	Learn from the best! Sebastian Thrun will walk you through the usage and concepts of a Kalman Filter using Python.
3	**C++ Checkpoint**	Are you ready to build Kalman Filters with C++? Take these quizzes to find out.
4	**Lidar and Radar Fusion with Kalman Filters**	In this lesson, you'll build a Kalman Filter in C++ that's capable of handling data from multiple sources. Why C++? Its performance enables the application of object tracking with a Kalman Filter in real-time.

Project 5: Extended Kalman Filters
In this project, you'll apply everything you've learned so far about Sensor Fusion by implementing an Extended Kalman Filter in C++!

Lesson	Title	Description
5	**Unscented Kalman Filters**	While Extended Kalman Filters work great for linear motion, real objects rarely move linearly. With Unscented Kalman Filters, you'll be able to accurately track non-linear motion!

Project 6: Unscented Kalman Filters
Put your skills to the test! Use C++ to code an Unscented Kalman Filter capable of tracking non-linear motion.

Course 5: Localization

Localization is how we determine where our vehicle is in the world. GPS is great, but it's only accurate to within a few meters. We need single-digit centimeter-level accuracy! To achieve this, Mercedes-Benz engineers will demonstrate the principles of Markov localization to program a particle filter, which uses data and a map to determine the precise location of a vehicle.

Lesson	Title	Description
1	Introduction to Localization	Meet the team that will guide you through the localization lessons!
2	Localization Overview	Sebastian Thrun will give you an overview of the theory behind localization!
3	Markov Localization	In this lesson, you'll learn the math behind localization as well as how to implement Markov localization in C++.
4	Motion Models	Here you'll learn about vehicle movement and motion models to predict where your car will be at a future time.
5	Particle Filters	Now, Sebastian will teach you what a particle filter is and will teach the theory and math behind the particle filter.
6	Implementation of a Particle Filter	Now that you understand how a particle filter works, you'll learn how to code a particle filter.

Project 7 : Kidnapped Vehicle
In this project, you'll build a particle filter and combine it with a real map to localize a vehicle!

Course 6: Control

Ultimately, a self-driving car is still a car, and we need to send steering, throttle, and brake commands to move the car through the world. Uber ATG will walk you through building both proportional-integral-derivative (PID) controllers and model predictive controllers. Between these control algorithms, you'll become familiar with both basic and advanced techniques for actuating a vehicle.

Lesson	Title	Description
1	PID Control	In this lesson you'll learn about and how to use PID

		controllers with Sebastian!

Project 8: PID Controller
In this project you'll revisit the lake race track from the Behavioral Cloning Project. This time, however, you'll implement a PID controller in C++ to maneuver the vehicle around the track!

Lesson	Title	Description
2	Vehicle Models	In this lesson, you'll learn about kinematic and dynamic vehicle models. We'll use these later with Model Predictive Control.
3	Model Predictive Control	In this lesson, you'll learn how to frame the control problem as an optimization problem over time horizons. This is Model Predictive Control!

Project 9: Model Predictive Control
In this project, you'll implement Model Predictive Control to drive the vehicle around the track even with additional latency between commands!

Course 7: Path Planning

Path planning routes a vehicle from one point to another, and it handles how to react when emergencies arise. The Mercedes-Benz Vehicle Intelligence team will take you through the three stages of path planning. First, you'll apply model-driven and data-driven approaches to predict how other vehicles on the road will behave. Then you'll construct a finite state machine to decide which of several maneuvers your own vehicle should undertake. Finally, you'll generate a safe and comfortable trajectory to execute that maneuver.

Lesson	Title	Description
1	Search	In this lesson you will learn about discrete path planning and algorithms for solving the path planning problem.
2	Prediction	In this lesson you'll learn how to use data from sensor fusion to generate predictions about the likely behavior of moving objects.
3	Behavior Planning	In this lesson you'll learn how to think about high level behavior planning in a self driving car.
4	Trajectory Generation	In this lesson, you'll use C++ and the Eigen linear algebra library to build candidate trajectories for the vehicle to

		follow.

Project 10: Path Planning
Drive a car down a highway with other cars using your own path planner.

Course 8: Elective: Advanced Deep Learning

Students in this elective, built with the NVIDIA Deep Learning Institute, will learn about semantic segmentation, and inference optimization, active areas of deep learning research. This course is an elective. Students choose between completing either Advanced Deep Learning or Functional Safety for graduation.

Lesson	**Title**	**Description**
1	**Fully Convolutional Networks**	In this lesson you'll learn the motivation for Fully Convolutional Networks and how they are structured.
2	**Scene Understanding**	In this lesson you'll be introduced to the problem of Scene Understanding and the role FCNs play.
3	**Inference Performance**	In this lesson you'll become familiar with various optimizations in an effort to squeeze every last bit of performance at inference.

Project 11: Semantic Segmentation
Build a semantic segmentation network to identify free space on the road. *This project is part of an elective course. Students choose between completing either Advanced Deep Learning or Functional Safety for graduation.*

Course 9: Elective: Functional Safety

Students who select the Functional Safety specialization, built with Elektrobit, learn functional safety frameworks to ensure that vehicles are safe, both at the system and component levels. This course is an elective. Students choose between completing either Advanced Deep Learning or Functional Safety for graduation.

Lesson	**Title**	**Description**
1	**Introduction to Functional Safety**	You will learn to make safer vehicles using risk evaluation and systems engineering.

2	**Safety Plan**	A functional safety plan is critical to any functional safety project. Here you will learn what goes into a safety plan so that you can document your own.
3	**Hazard Analysis and Risk Assessment**	In a hazard analysis and risk assessment, you will identify vehicular malfunctions and evaluate their risk levels. You can then derive safety goals defining how your vehicle will remain safe.
4	**Functional Safety Concept**	You will derive functional safety requirements from the safety goals and then add extra functionality to the system diagram. Finally you document your work, a part of functional safety.
5	**Technical Safety Concept**	Once you have derived functional safety requirements, you drill down into more detail. In the technical safety concept, you refine your requirements into technical safety requirements.
6	**Functional Safety at the Software and Hardware Levels**	The last step in the vehicle safety design phase is to derive hardware and software safety requirements. In this lesson, you will derive these requirements and refine a software system architecture.

Project 11: Functional Safety of a Lane Assistance System
You'll use the guidance from your lessons to construct part of a safety case for a lane departure warning feature. You'll begin with the hazard analysis and risk assessment, and create further documentation for functional and technical safety concepts, and finally software and hardware requirements. Analyzing and documenting system safety is critical for autonomous vehicle development. These are skills that often only experienced automotive engineers possess! *This project is part of an elective course. Students choose between completing either Advanced Deep Learning or Functional Safety for graduation.*

Course 10: System Integration

This is capstone of the entire Self-Driving Car Engineer Nanodegree Program! We'll introduce Carla, the Udacity self-driving car, and the Robot Operating System that controls her. You'll work with a team of othe Nanodegree students to combine what you've learned over the course of the entire Nanodegree Program to drive Carla, a real self-driving car, around the Udacity test track!

Lesson	Title	Description
1	**Autonomous Vehicle Architecture**	Learn about the system architecture for Carla, Udacity's autonomous vehicle.

2	**Introduction to ROS**	Obtain an architectural overview of the Robot Operating System Framework and setup your own ROS environment on your computer.
3	**Packages and Catkin Workspaces**	Learn about ROS workspace structure, essential command line utilities, and how to manage software packages within a project. Harnessing these will be key to building shippable software using ROS.
4	**Writing ROS Nodes**	ROS Nodes are a key abstraction that allows a robot system to be built modularly. In this lesson, you'll learn how to write them using Python.

Project 12: Programming a Real Self-Driving Car
Run your code on Carla, Udacity's autonomous vehicle!