UDACITY
FOR ENTERPRISE

THE SCHOOL OF AUTONOMOUS SYSTEMS

Self Driving Car Engineer

NANODEGREE SYLLABUS

Overview

This Nanodegree Program is built in partnership with

Mercedes-Benz NVIDIA UBER ATG DiDi BMW McLaren

You'll first apply computer vision and deep learning to automotive problems, including detecting lane lines, predicting steering angles and more. Next, you'll learn sensor fusion, which you'll use to filter data from an array of sensors in order to perceive the environment. Then you'll have the opportunity to run your code on a virtual simulation of Udacity's real self-driving car, Carla.

Program Information

TIME
6 months
Study 10 hours/week

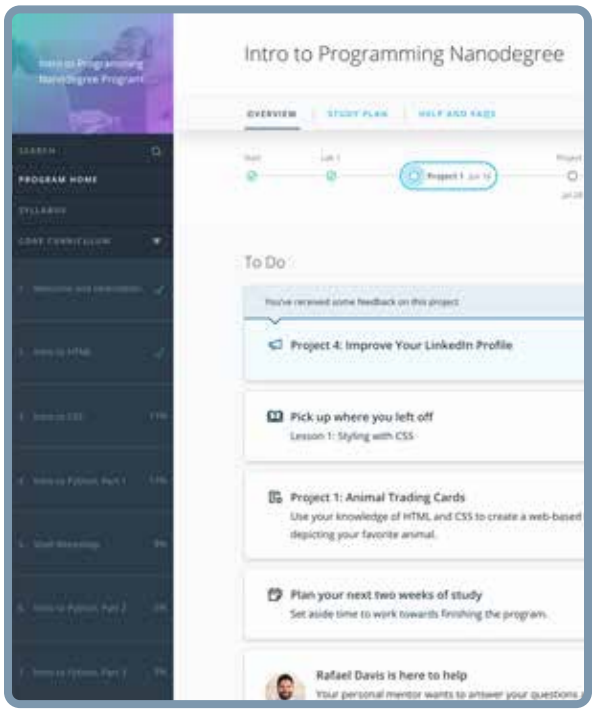
LEVEL
Specialist

PREREQUISITES
Intermediate Python (Classes, Data structures), Intermediate C++ (Classes, Memory management, Linking), Basic Linear Algebra (Matrices, Vectors, Matrix multiplication), Basic Calculus (Derivatives, Integrals), Basic Statistics (Mean, Standard deviation, Gaussian distribution) and Basic Physics (Forces).

HARDWARE/SOFTWARE REQUIRED
PC: Windows 7 or higher with the latest updates installed (note: Internet Explorer is not supported).
Mac: OS X 10.11 or higher with latest updates installed.
Linux: Any recent distribution that has the supported browsers installed.
Ubuntu: 17.10+ or 14.04 LTS+.

LEARN MORE ABOUT THIS NANODEGREE
Contact us at enterpriseNDs@udacity.com.

Our Classroom Experience



REAL-WORLD PROJECTS

Learners build new skills through industry-relevant projects and receive personalized feedback from our network of 900+ project reviewers. Our simple user interface makes it easy to submit projects as often as needed and receive unlimited feedback.

KNOWLEDGE

Answers to most questions can be found with Knowledge, our proprietary wiki. Learners can search questions asked by others and discover in real-time how to solve challenges.

LEARNER HUB

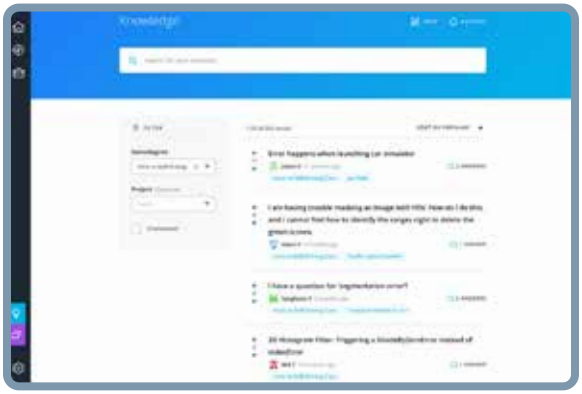
Learners leverage the power of community through a simple, yet powerful chat interface built within the classroom. Learner Hub connects learners with their technical mentor and fellow learners.

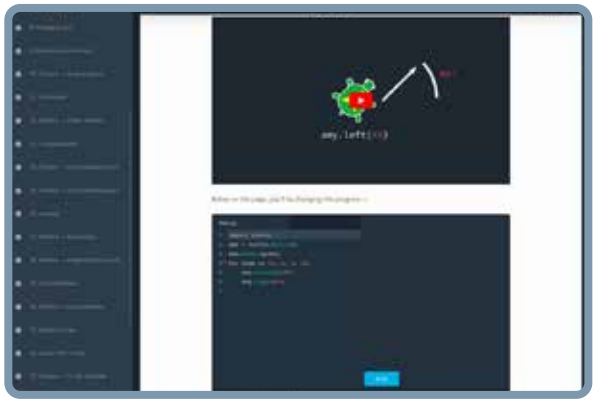
WORKSPACES

Learners can check the output and quality of their code by testing it on interactive workspaces that are integrated into the classroom.

QUIZZES

Understanding concepts learned during lessons is made simple with auto-graded quizzes. Learners can easily go back and brush up on concepts at anytime during the course.





CUSTOM STUDY PLANS

Mentors create a custom study plan tailored to learners' needs. This plan keeps track of progress toward learner goals.

PROGRESS TRACKER

Personalized milestone reminders help learners stay on track and focused as they work to complete their Nanodegree program.

Learn with the Best

Sebastian Thrun

UDACITY PRESIDENT

Scientist, educator, inventor, and entrepreneur, Sebastian led the self-driving car project at Google X and founded Udacity, whose mission is to democratize education by providing lifelong, on-demand learning to millions of students around the world.

David Silver

CURRICULUM LEAD

David Silver leads the Udacity Curriculum Team. Before Udacity, David was a research engineer on the autonomous vehicle team at Ford. He has an MBA from Stanford, and a BSE in Computer Science from Princeton.

Ryan Keenan

COURSE DEVELOPER

Ryan has a PhD in Astrophysics and a passion for teaching and learning. He is also a lead instructor in the Robotics Nanodegree program. When he's not building Udacious learning content you'll find him up in the mountains or out in the surf.

Cezanne Camacho

COURSE DEVELOPER

Cezanne is an expert in computer vision with an M.S. in Electrical Engineering from Stanford University. Inspired by anyone with the drive and imagination to learn something new, she aims to create more inclusive and effective STEM education.

Learn with the Best

Mercedes-Benz

MERCEDES-BENZ TEAM

Mercedes-Benz R&D North America develops the world's most advanced automotive technology and vehicle design with luxury and style. The team from Mercedes built our Sensor Fusion, Localization, and Path Planning content.

NVIDIA

NVIDIA TEAM

NVIDIA is a company built upon great minds and groundbreaking research. GPU deep learning has ignited modern AI - the next era of computing - with the GPU acting as the brain of computers, robots, and self-driving cars that can perceive and understand the world.

Uber ATG

UBER ATG TEAM

The Advanced Technologies Group is comprised of Uber's self-driving engineering team dedicated to self-driving technologies, mapping, and vehicle safety.

Nanodegree Program Overview

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	LEARNING OUTCOME
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 6-month journey.
WORKSPACES	• Many projects and some quizzes will be accessed via Workspaces. These workspaces streamline environment setup, simplify project submission, and can be enabled with GPU support. All workspaces are Linux-based and can be interfaced via a shell (BASH). Some workspace interfaces are direct from the shell, others run a JUPYTER Notebook server and interaction is mainly through the JUPYTER notebook interface.

Course 2: 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.

Project: Finding Lane Lines on the 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.

Project: Advanced Lane Finding

Building on the previous project, your goal now is to write a software pipeline to identify the lane boundaries in a video from a front-facing camera on a car.

LESSON TITLE	LEARNING OUTCOME
COMPUTER VISION FUNDAMENTALS	• 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.
ADVANCED COMPUTER VISION	• Discover more advanced computer vision techniques, like distortion correction and gradient thresholding, to improve upon your lane lines algorithm.

Course 3: 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.

Project: 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.

Project: Behavioral Cloning

Put your deep learning skills to the test with this project! Train a deep neural network to drive a car like you.

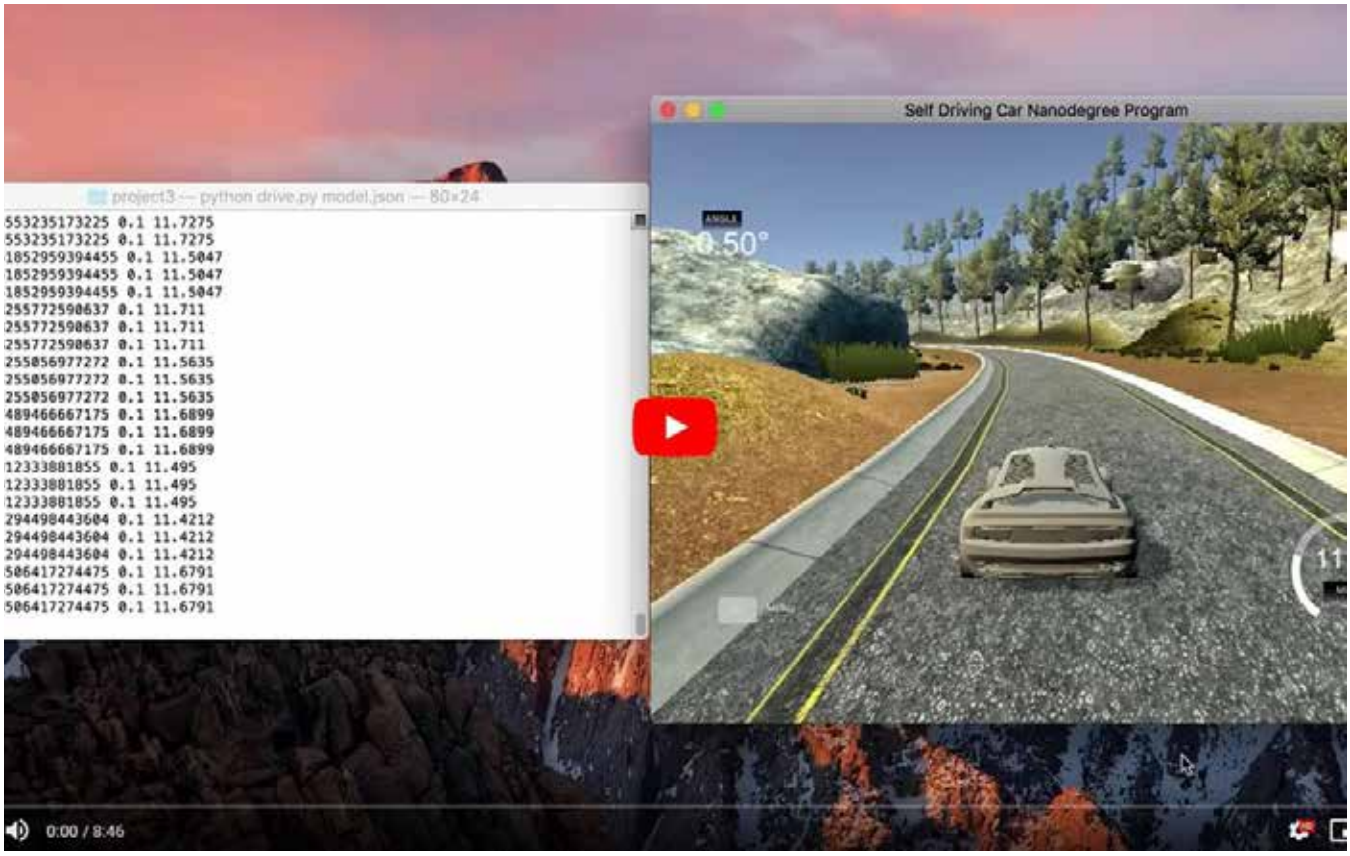
LESSON TITLE	LEARNING OUTCOME
NEURAL NETWORKS	• Learn to build and train neural networks, starting with the foundations in linear and logistic regression, and culminating in backpropagation and multilayer perceptron networks.
TENSORFLOW	• Vincent Vanhoucke, Principal Scientist at Google Brain, introduces you to deep learning and Tensorflow, Google's deep learning framework.
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.
CONVOLUTIONAL NEURAL NETWORK	• Vincent explains the theory behind Convolutional Neural Networks and how they help us dramatically improve performance in image classification.

Nanodegree Program Overview

LESSON TITLE	LEARNING OUTCOME
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.
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.

Project Example

Behavioral Cloning*



**Actual learner-submitted course project solution.*

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 an advanced technique: the extended Kalman filter.

Project: 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	LEARNING OUTCOME
SENSORS	• Meet the team at Mercedes who will help you track objects in real-time with Sensor Fusion.
KALMAN FILTERS	• Learn from the best! Sebastian Thrun will walk you through the usage and concepts of a Kalman Filter using Python.
C++ CHECKPOINT	• Are you ready to build Kalman Filters with C++? Take these quizzes to find out.
EXTENDED 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.

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.

Project: Kidnapped Vehicle

In this project, you'll build a particle filter and combine it with a real map to localize a vehicle.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO LOCALIZATION	• Meet the team that will guide you through the localization lessons.
MARKOV LOCALIZATION	• In this lesson, you'll learn the math behind localization as well as how to implement Markov localization in C++.
MOTION MODELS	• Here you'll learn about vehicle movement and motion models to predict where your car will be at a future time.
PARTICLE FILTERS	• Sebastian will teach you what a particle filter is as well as the theory and math behind the filter.
IMPLEMENTATION OF A PARTICLE FILTER	• Now that you understand how a particle filter works, you'll learn how to code a particle filter.

Course 6: 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.

Project: Highway Driving

Drive a car down a highway with other cars using your own path planner.

LESSON TITLE	LEARNING OUTCOME
SEARCH	• In this lesson you will learn about discrete path planning and algorithms for solving the path planning problem.
PREDICTION	• In this lesson you'll learn how to use data from sensor fusion to generate predictions about the likely behavior of moving objects.
BEHAVIOR PLANNING	• In this lesson you'll learn how to think about high level behavior planning in a self-driving car.
TRAJECTORY GENERATION	• In this lesson, you'll use C++ and the Eigen linear algebra library to build candidate trajectories for the vehicle to follow.

Course 7: 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.

Project: 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	LEARNING OUTCOME
PID CONTROL	• In this lesson you'll learn about and how to use PID controllers with Sebastian.

Project Example: PID Control*

**Actual learner-submitted course project solution.*

Course 8: System Integration

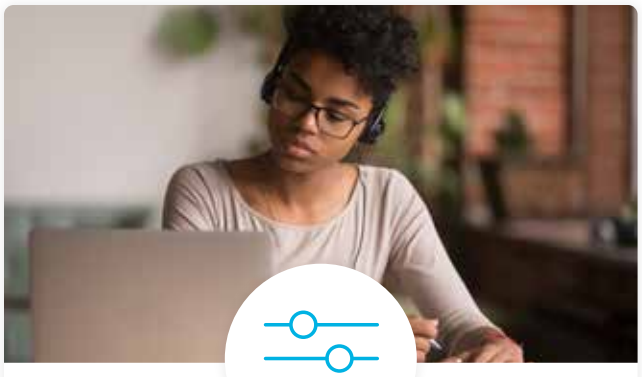
This is capstone of the entire Self-Driving Car Engineer Nanodegree Program! We'll introduce Carla, the Udacity self-driving car, and the Robot Operating System that controls her. You'll work to combine what you've learned over the course of the entire Nanodegree Program to run your code on a virtual version of Carla, a real self-driving car!

Project: System Integration

In this project, you will run your code on a virtual version of Carla, Udacity's self-driving car.

LESSON TITLE	LEARNING OUTCOME
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.
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.
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.

Our Nanodegree Programs Include:



Pre-Assessments

Our in-depth workforce assessments identify your team's current level of knowledge in key areas. Results are used to generate custom learning paths designed to equip your workforce with the most applicable skill sets.

Dashboard & Progress Reports

Our interactive dashboard (enterprise management console) allows administrators to manage employee onboarding, track course progress, perform bulk enrollments and more.

Industry Validation & Reviews

Learners' progress and subject knowledge is tested and validated by industry experts and leaders from our advisory board. These in-depth reviews ensure your teams have achieved competency.

Real World Hands-on Projects

Through a series of rigorous, real-world projects, your employees learn and apply new techniques, analyze results, and produce actionable insights. Project portfolios demonstrate learners' growing proficiency and subject mastery.

Our Review Process

Real-life Reviewers for Real-life Projects

Real-world projects are at the core of our Nanodegree programs because hands-on learning is the best way to master a new skill. Receiving relevant feedback from an industry expert is a critical part of that learning process, and infinitely more useful than that from peers or automated grading systems. Udacity has a network of over 900 experienced project reviewers who provide personalized and timely feedback to help all learners succeed.

Vaibhav
UDACITY LEARNER

"I never felt overwhelmed while pursuing the Nanodegree program due to the valuable support of the reviewers, and now I am more confident in converting my ideas to reality."

now at

CODING VISIONS INFOTECH

All Learners Benefit From:

- Line-by-line feedback for coding projects
- Industry tips and best practices
- Advice on additional resources to research
- Unlimited submissions and feedback loops

How it Works

Real-world projects are integrated within the classroom experience, making for a seamless review process flow.

- Go through the lessons and work on the projects that follow
- Get help from your technical mentor, if needed
- Submit your project work
- Receive personalized feedback from the reviewer
- If the submission is not satisfactory, resubmit your project
- Continue submitting and receiving feedback from the reviewer until you successfully complete your project

About our Project Reviewers

Our expert project reviewers are evaluated against the highest standards and graded based on learners' progress. **Here's how they measure up to ensure your success.**

900+

Expert Project Reviewers

Are hand-picked to provide detailed feedback on your project submissions.

1.8M

Projects Reviewed

Our reviewers have extensive experience in guiding learners through their course projects.

3

Hours Average Turnaround

You can resubmit your project on the same day for additional feedback.

4.85 /5

Average Reviewer Rating

Our learners love the quality of the feedback they receive from our experienced reviewers.

UDACITY
FOR ENTERPRISE

2440 W El Camino Real, #101
Mountain View, CA 94040, USA - HQ

For more information visit: **www.udacity.com/enterprise**