UDACITY
FOR ENTERPRISE

THE SCHOOL OF AUTONOMOUS SYSTEMS

Robotics Software Engineer

NANODEGREE SYLLABUS

Overview

Robotics Software Engineer Nanodegree Program

This program was built in collaboration with robotics engineers to ensure you learn the skills necessary for success in the field.

Learn probabilistic robotics algorithms such as Localization, Mapping, Simultaneous Localization and Mapping (SLAM), Path Planning, and Navigation, and implement them using C++, the Robot Operating System (ROS), and Gazebo. Build five interactive projects using C++ and ROS, and use them to demonstrate your skills in probabilistic robotics algorithms.

The program is delivered in one term spread over four months. On average, learners will need to spend about 10-15 hours per week in order to complete all required coursework, including lecture and project time.

Program Information

TIME
4 months
Study 10 hours/week

LEVEL
Specialist

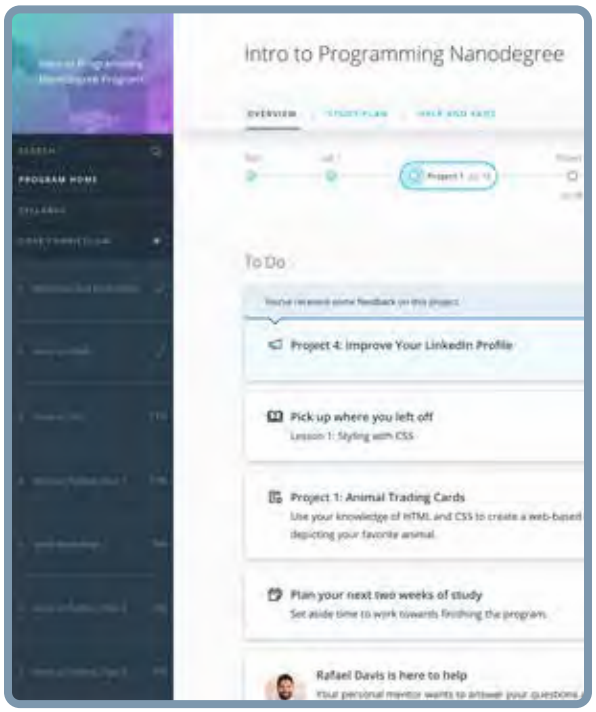
PREREQUISITES
Advanced knowledge in any object-oriented programming language, preferably C++

- Intermediate Probability
- Intermediate Calculus
- Intermediate Linear Algebra
- Basic Linux Command Lines

HARDWARE/SOFTWARE REQUIRED
Robot Operating System (ROS) and Gazebo. You will code primarily with C++. These platforms and languages are freely available. We will provide you with a GPU-enabled Linux Workspace that runs in your browser, and an internet connection is required. Optionally, you can install the Linux image on a Virtual Machine.your browser.

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

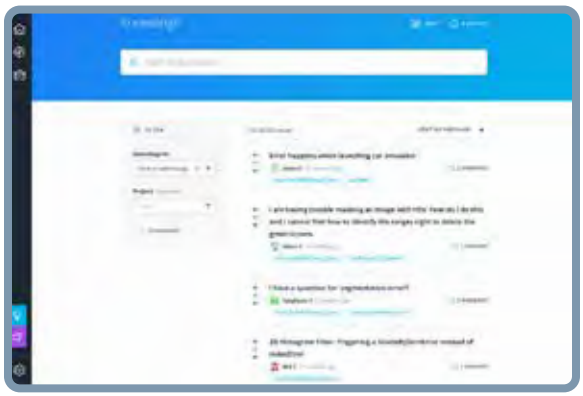
Learners leverage the power of community through a simple, yet powerful chat interface built within the classroom. Learner Hub connects learners with their technical mentor and fellow learners.

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

INSTRUCTOR

As the founder and president of Udacity, Sebastian's mission is to democratize education. He is also the founder of Google X, where he led projects including the Self-Driving Car, Google Glass, and more.

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.

Karim Chamaa

COURSE DEVELOPER

Karim started his early career as a Mechanical Engineer. He earned his M.S. in Mechatronics and Robotics Engineering from NYU. His specialties include Kinematics, Control, and Electronics.

Julia Chernushevich

COURSE DEVELOPER

Julia is an experienced educator and robotics specialist. Her previous work experiences include teaching Mechatronics Engineering at the University of Waterloo and designing electric vehicles for underground mines.

Course 1: Introduction To Robotics

Learn the essential elements of robotics, meet your instructors, and get familiar with the tools that will help you succeed in this program.

Course 2: Gazebo World

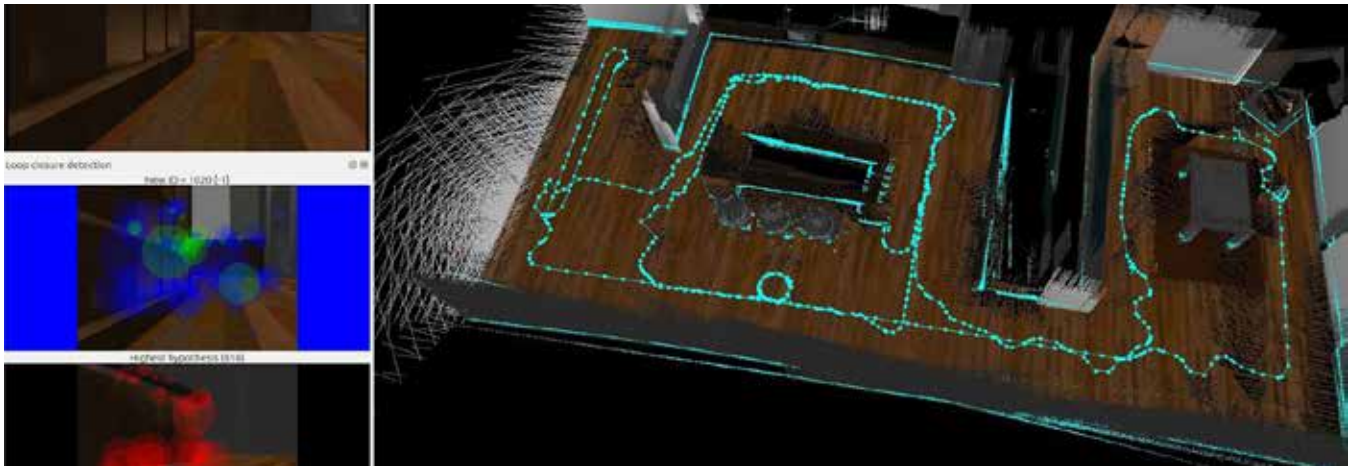
Learn how to simulate your first robotic environment with Gazebo, the most common simulation engine used by Roboticists around the world.

Project: Build My World

Use Gazebo to simulate a robotic environment comprised of a building to house your future robot. Skills you will apply: Gazebo, C++ plugins.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO GAZEBO	• Work with the Gazebo simulator to build new environments, and deploy assets.
PROJECT: BUILD MY WORLD	• Step by Step - Design and build your first Gazebo environment.

Project Example: Map My World*



Actual learner-submitted course project solution.

Course 3: ROS Essentials

Discover how ROS provides a flexible and unified software environment for developing robots in a modular and reusable manner. Learn how to manage existing ROS packages within a project, and how to write ROS Nodes of your own in C++.

Project: Go Chase It

Use the Robot Operating System (ROS) to design a mobile robot. Then, house your newly-designed robot in the robotic environment you built in Project 1. You will program your robot with C++ to chase a ball through this world. Skills you will apply: catkin workspaces, ROS packages, ROS nodes, ROS launch files, RViz integration, and C++.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO ROS	• Obtain an architectural overview of the Robot Operating System Framework.
PACKAGES & CATKIN WORKSPACES	• Learn the ROS workspace structure, essential command line utilities, and how to manage software packages within a project.
WRITE ROS NODES	• Write ROS nodes in C++.
PROJECT: GO CHASE IT!	• Step by Step - build your first robot in Gazebo. • Step by Step - build a C++ service server node in ROS. • Step by Step - build a C++ service client node in ROS.

Course 4: Localization

Learn how Gaussian filters can be used to estimate noisy sensor readings, and how to estimate a robot's position relative to a known map of the environment with Monte Carlo Localization (MCL).

Project: Where Am I?

Use the Monte Carlo Localization algorithm in ROS, in conjunction with sensor data and a map of the world, to estimate a mobile robot's position and orientation so that your robot can answer the question "Where am I?" Skills you will apply: Localization algorithms: Kalman Filter and MCL, ROS parameters, ROS packages integration, C++.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO LOCALIZATION	• Learn what it means to localize and the challenges behind it.
KALMAN FILTERS	• Learn the Kalman Filter and its importance in estimating noisy data.
LAB: KALMAN FILTERS	• Implement an Extended Kalman Filter package with ROS to estimate the position of a robot.
MONTE CARLO LOCALIZATION	• Learn the MCL (Monte Carlo Localization) algorithm to localize robots.
BUILD MCL IN C++	• Code the MCL algorithm in C++.
PROJECT: WHERE AM I?	• Set up and explore the steps for the Where Am I? Project using AMCL with ROS in C++.

Course 5: Mapping and SLAM

Learn how to create a Simultaneous Localization and Mapping (SLAM) implementation with ROS packages and C++. You'll achieve this by combining mapping algorithms with what you learned in the localization lessons.

Project: Map My World

Simultaneous Localization and Mapping (SLAM) can be implemented in a number of ways depending on the sensors used via various ROS packages. Use a ROS SLAM package and simulated sensor data to create an agent that can both map the world around it, and localize within it. Skills you will apply: Mapping and SLAM algorithms, Occupancy Grid Mapping, Grid-based FastSLAM and GraphSLAM, ROS debugging tools, C++.

LESSON TITLE	LEARNING OUTCOME
INTRODUCTION TO MAPPING AND SLAM	• Learn the Mapping and SLAM concepts, as well as the algorithms.
OCCUPANCY GRID MAPPING	• Map an environment by coding the Occupancy Grid Mapping algorithm with C++.
GRID-BASED FASTSLAM	• Simultaneously map an environment and localize a robot relative to the map with the Grid-based FastSLAM algorithm. • Interface a turtlebot with a Grid-based FastSLAM package with ROS to map an environment.
GRAPHSLAM	• Simultaneously map an environment and localize a robot relative to the map with the GraphSLAM algorithm.
PROJECT: MAP MY WORLD ROBOT	• Deploy RTAB-Map on your simulated robot to localize it and create 2D and 3D maps of your environment.

Course 6: Path Planning and Navigation

Learn different Path Planning and Navigation algorithms. Then, combine SLAM and Navigation into a home service robot that can autonomously transport objects in your home!

Project: Home Service Robot

Combine everything you've learned in this program to simulate a home service robot that can map, localize, and navigate to transport objects, moving from one room to another autonomously. Skills you will apply: Path Planning search algorithms, ROS navigation stack, C++.

LESSON TITLE	LEARNING OUTCOME
INTRO TO PATH PLANNING AND NAVIGATION	• Learn what the lessons in Path Planning and Navigation will cover.
CLASSIC PATH PLANNING	• Learn a number of classic path planning approaches that can be applied to low-dimensional robotic systems.
LAB: PATH PLANNING	• Code the BFS and A* algorithms in C++.
SAMPLE-BASED AND PROBABILISTIC PATH PLANNING	• Learn about sample-based and probabilistic path planning, and how they can improve on the classic approach.
CAPSTONE PROJECT: HOME SERVICE ROBOT	• Program a home service robot that will autonomously map an environment and navigate to pick up and deliver objects.

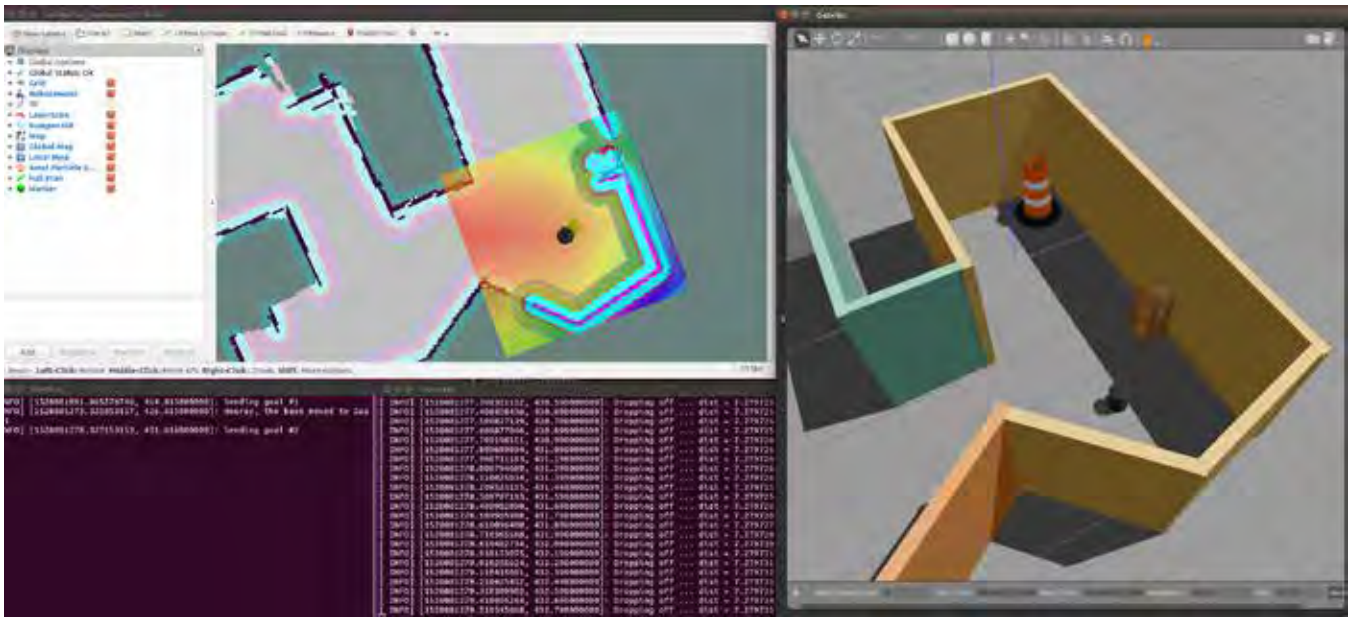
Nanodegree Program Overview

Optional Project — Home Service Robot

Students will apply what they have learned about ROS and path planning to search for a path and navigate a KUKA robot through a 2D maze.

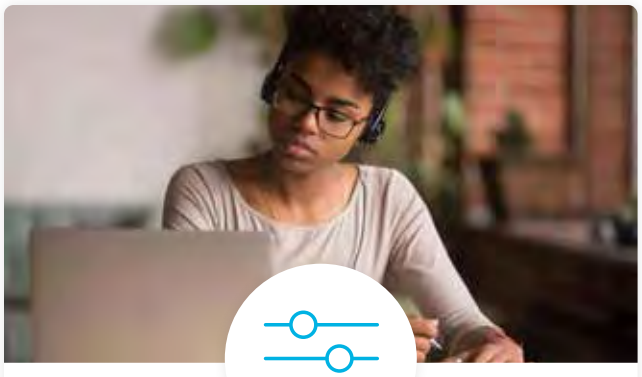
LESSON TITLE	LEARNING OUTCOME
PROJECT INTRODUCTION	• Learn the requirements of the project.
PROJECT DETAILS	• Learn the project specifications and how to get started.
PROJECT: KUKA PATH PLANNING	• Search for a path and navigate a KUKA robot through a 2D maze.

Project Example — Home Service Robot*



**Actual learner-submitted course project solution.*

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