IE 482/582: Special Topics in Robotics

Course Syllabus – Spring 2025

Instructor:	Chase Murray, Ph.D.
Email:	cmurray3@buffalo.edu (please include "IE 482:" or "IE 582:", as applicable, in the subject)
Office:	309 Bell Hall
Office hours:	TBD, or by appointment.
Teaching Assistant:	Angelo Soto Vergel angeloso@buffalo.edu See GitHub for office hours

Course Meeting Days, Times, and Locations

The course is officially scheduled as follows:

Tues/Thurs, 11:00am – 12:20pm, 427 Bell Hall

Course Description

This course introduces Industrial Engineering students to robots and robotic systems, including the design of robot controllers, coordination of multiple robots, simulation of robotic systems, and optimization of robot task scheduling.

Prerequisite(s): This course is computer programming intensive. Students should have programming experience, preferably in Python.

Corequisite(s): None

Textbook and/or Other Required Materials

- [optional] Programming Robots with ROS, by Quigley, Gerkey, and Smart http://www.amazon.com/Programming-Robots-ROS-Practical-Introduction/dp/1449323898
- [required] Each student must have a laptop that meets UB's requirements: https://www.buffalo.edu/ubit/service-guides/hardware/getting-started-with-hardware/purchasing-or-using-an-existing-computer.html

Course Websites

- GitHub will be used for disseminating course materials (e.g., code) https://github.com/IE-482-582/spring2025
- Brightspace will be used only for posting course grades

Course Learning Outcomes

The following table lists learning outcomes for this course. The statements generally complete the sentence, "Upon completing this course, students will be able to..."

	Course Learning Outcome	Program Outcomes*	Assessment Methods
1	Understand the basics of the Robot Operating System (ROS).	7	Quizzes, Midterm Exam, Homework
2	Apply concepts of Systems Engineering to both simulated and real robotic systems.	1, 6	Course Project
3	Understand, interpret, and modify Python code for the purposes of controlling robotic systems.	1, 7	Homework, Course Project
4	Demonstrate and improve upon project management skills. Students will be developing a complex system with numerous deadlines. To be successful, each student team must establish (and hit) their milestones.	5	Course Project
5	Appreciate the importance of developing thorough documentation of a system. Students will be responsible for writing README files that document important elements of the system, including instructions for running the code, citations for reference materials, opportunities for future enhancement, and known issues.	3	Homework, Course Project
6	Acquire and apply new knowledge related to computer programming. To successfully complete the course project in particular, students will need to learn how to find solutions to challenges. This will involve looking for code examples online and reading journal articles.	7	Homework, Course Project

* The Student Outcomes from the Engineering Accreditation Commission of ABET have been adopted, see http://www.abet.org/

Program Outcome Support

0: No Coverage 1: Introduced 2: Practiced/Reinforced 3: Mastered

1	**2**	**3**	**4**	**5**	**6**	**7**
1	0	3	0	3	2	3

(1) an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics;

(3) an ability to communicate effectively with a range of audiences;

(5) an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives;

(6) an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions

(7) an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

Course Requirements

Requirement	**Quantity**	**Material Covered**	**Approx. Date**
Attendance	~28	N/A	Each class period
Homework/Coding Assignments	~6	Variable; related to material covered within lecture, see schedule	~Weeks 2-10
Project Proposal	1	Topic selected by student(s) in consultation with instructor	Week 6
Project Status Report	1	See below	Week 11
Project Documentation	1	See below	Week 14
Project Demonstration	1	See below	Week 15

Grading Policy
Students will be evaluated on:

Attendance [15%]: Attendance will be taken at 11:00 (late = absent). You get 3 free passes (both excused and unexcused) for attendance. There will be a 10% grade reduction per absence beyond 3 absences.

Homework/Coding Assignments [30%]: You should expect to have approximately six programming assignments. Some of these assignments will be individually assigned, some will be group assignments. Details and requirements will be provided for each assignment. The purpose of these assignments is to help students gain a deeper understanding of the material covered in each topic. Homework/coding assignments will be concentrated in the first 10 weeks of the semester.

Course Project [55%]: The last 1/2 (approximate) of the semester students will primarily focus on activities related to the course project. Students will work in teams of no more than two (2) students. The exact topic of the project will be up to each team to decide (with approval from the instructor). The project should involve expanding the students' basic robotics knowledge (as obtained during the first 1/2 of the semester) to develop a real or simulated robotic system. Examples and suggestions will be provided during the first 6 weeks of the semester.

There are four key project deliverables:

1) A project proposal [10%] – This will be in the form of a 1-page markdown document, due at the beginning of Week 6 (unless directed otherwise). Each project team will give a brief (<5-minute) presentation (probably during office hours, but possible in class).

2) A project status report [20%] – Each team will be required to submit a markdown document outlining the status of the project. This document should clearly list all milestones, and should identify those activities that have been completed, those that are in progress, and those that are in danger of being missed. In the latter case, a plan to address these potential misses should be provided. Each team will present their status reports (probably during office hours, but possibly in class).

3) Project documentation [50%] – By the last week of class, each team must submit detailed documentation of the project. This should include instructions for running the code, example outputs, and citations to reference materials. The documentation should also identify potential areas for future improvement. Each team must have their documentation reviewed by at least one other team (and must also review the documentation of at least one other team). Grades will also be based on how thoroughly your team evaluates another team.

4) Project demonstration [20%] – During the last week of class, each team will present their projects in class (or will record a YouTube video of their presentation).

Final course grades will be determined based on the following scale:

Final Avg ≥ 94.0%	A
Final Avg ≥ 90.0%	A-
Final Avg ≥ 87.0%	B+
Final Avg ≥ 83.0%	B
Final Avg ≥ 80.0%	B-
Final Avg ≥ 77.0%	C+
Final Avg ≥ 73.0%	C
Final Avg ≥ 70.0%	C-
Final Avg ≥ 67.0%	D+
Final Avg ≥ 60.0%	D
Final Avg < 60.0%	F

Do not expect your grade to be “rounded up.”

Incomplete Grades

A grade of incomplete (“I”) indicates that additional course work is required to fulfill the requirements of a given course. Students may only be given an “I” grade if they have a passing average in coursework that has been completed and have well-defined parameters to complete the course requirements that could result in a grade better than the default grade. An “I” grade may not be assigned to a student who did not attend the course.

Prior to the end of the semester, students must initiate the request for an “I” grade and receive the instructor’s approval. Assignment of an “I” grade is at the discretion of the instructor.

Grade Disputes

If you disagree with the manner in which an assignment was graded, you may request a re-evaluation of your assignment within two (2) weeks of the due date of that assignment. A re-evaluation request should consist of two (2) components:

- Page 1: A photocopy of the graded assignment.
- Page 2: A detailed explanation, not exceeding one-half page in length, describing why you believe your answer was correct.

The instructor will consider each case at the end of the term, but only if it appears that it may change your final grade. Obvious arithmetic errors will be corrected immediately.

Professionalism

UB SEAS aims to enhance the education of the students in various aspects of professionalism, and to elevate the standards of behavior that are expected from students. The goals are two-fold: (1) to improve the working and learning environment within SEAS, and (2) to best equip students for employment after graduation.

- Students are expected to use professional style in all communications, including email, with course faculty and teaching assistants. This includes the use of salutations and closings (including clear identification of the author) and correct grammar.
- Students are expected to log in prior to the start of class, and to remain for the duration of the class.

Accessibility Resources

If you have a disability and may require some type of instructional and/or examination accommodation, please inform me early in the semester so that we can coordinate the accommodations you may need. If you have not already done so, please contact the Office of Accessibility Resources; 60 Capen Hall; email: stu-accessibility@buffalo.edu Phone: 716-645-2608 (voice); 716-645-2616 (TTY); and on the web at https://www.buffalo.edu/studentlife/who-we-are/departments/accessibility.html. All information and documentation is confidential.

Academic Integrity

This course will operate with a zero-tolerance policy regarding cheating and other forms of academic dishonesty. Any act of academic dishonesty will subject the student to penalty, including the high probability of failure of the course (i.e., assignment of a grade of ‘F’). It is expected that you will behave in an honorable and respectful way as you learn and share ideas. Therefore, *recycled papers, work submitted to other courses, and major assistance in preparation of assignments without identifying and acknowledging such assistance* are not acceptable. All work for this course must be original for this course. Please be familiar with the University at Buffalo Academic Integrity Policy and Procedure outlined at https://catalog.buffalo.edu/policies/integrity.html .

Tentative Course Schedule

Week	Topic	Reading/Assignment
Week 1	Introduction	Review syllabus, check out ROS videos.
Week 2	Python/JavaScript bridge	Homework (documentation)
Week 3	Computer Vision	Homework (find/test modules, documentation)
Week 4	Dynamixel demo	Homework (documentation)
Week 5	Discuss projects OpenDroneMap	Homework (ODM demo)
Week 6	Audio processing Project proposals due	Homework (coding, documentation) Proposal presentations (office)
Week 7	Creating LLM chatbot	Homework (coding, documentation)
Week 8	GPS	Homework (coding, documentation)
Week 9	Sensors	Homework (coding, documentation)
Week 10	Intro to ROS (installation, topics/services)	
Week 11	Class Project status reports due Intro to ROS (Gazebo)	Each project team must report on their progress (office)
Week 12	Intro to ROS (SLAM)	Small coding exercises
Week 13	Controlling other robots (drones, differential drive rover, quadruped)	
Week 14	Project Documentation due	Finalize the README document for your class project.
Week 15	Class project demonstrations during Finals Week	Documentation due. Demonstrations.