

Spring 2018 Syllabus Applied Software Technology - CET 4962

Course Layout Designer & Instructor: Prof. Danielle Safonte (dsafonte@citytech.cuny.edu)

Class Meeting Times

Lecture via BlackBoard Collaborate / ZOOM Sessions PROMPTLY Wednesdays 8:30 AM-11:00 AM (You can login to the session up to a half hour prior to class. You are expected to be logged in and ready by class time as if we were in a physical classroom.

Lab V0613 Fridays 8:30 AM-11:00 AM

The Virtual Schedule:

ALL Students are expected to attend all lectures on Wednesdays from 8:30-11:00am using the class link for that date on Blackboard.

Course Description:

This is a lecture/lab that provides a foundation in using Software to create programs and applications for modern day technology. This course focuses on a Design – Program – Build ideology.

❖ Design

- Creator: Design basic 2D and 3D assets
- Builder: Combine 2D and 3D assets in the assembly of a project
- Developer: Use multiple designed assets in completed products and models
- Maker: Design multiple and integrating assets for use in complex finished projects and models

❖ Program

- Creator: Use basic programming constructs to create simple programs
- Builder: Combine programming constructs to solve a problem
- Developer: Apply abstraction and decomposition to solve more complex problems
- Maker: Apply higher-order programming techniques to solve real-world problems

❖ Build

- Creator: Prototypes
- Builder: Build and Test Circuits
- Developer: Create multiple programs using Scratch and C++
- Maker: Build prototypes with a breadboard and Raspberry Pi computer with self-created applications

Textbook (physical or digital copy is ok):

Cook, Mike; Robinson, Andrew. Raspberry Pi Projects. 1st Edition. ISBN: 978-1118555439

Required Equipment / Software / Emulator:

Raspberry Pi 3 kit. Sense HAT ([physical](#) or [emulator](#)) and SD card.

**Physical Sense HAT setup Instructions can be found here: <https://codeclubprojects.org/en-GB/resources/physical-sense-hat/>

Python 3.2 Emulator: <https://codeclubprojects.org/en-GB/resources/python-working-offline/>

Grading:

Homework	20%
Labs	30%
Examinations	20%
Final Project	30%

All work must be submitted on time, via Blackboard.

Assignments that are late will be deducted 5 points per week!

Attendance:

Any work missed as a result of an absence is the responsibility of the student. In the case of an examination, a makeup will be allowed only in the case where the student has made arrangements **PRIOR** to the absence, or has a doctor's note proving illness or police report.

If any labs/experiments were missed and not made up within two classes, they will be graded zero. Missed laboratory experiments may be completed in the laboratory after completing the scheduled experiment. As per College Policy, attendance will be taken in class. College attendance policies are in effect.

Technology Prerequisites:

1. You must **register for the CUNY Portal**, <http://www.cuny.edu>
2. You must enter your **current/correct email address in Blackboard**
3. You must have access to and be able to use the Netscape and/or Internet Explorer browsers versions 4.0 or above.
4. You will need an email account and must be comfortable using it.
5. You need access to a computer that can operate Blackboard Collaborate. Instructions will be provided in the Course Content
6. You must send an email to me, dsafonte@citytech.cuny.edu, so I am able contact you, and form a distribution list for the class

Online Behavior:

Each student has the right to study and learn in a comfortable environment without fear, without humiliation, without intimidation, without offensive or suggestive language, and without the destruction of self-esteem.

****Posts that violate this policy will be removed by the instructor. ****

****I monitor all class activity, this includes your computer activity during class. Behave as we are in a physical classroom, and there will not be any issues. ****

Classroom Behavior:

The instructor requires a professional atmosphere in the classroom. No cursing, making noise, whistling, playing music, or inappropriate touching other people or their stuff will be allowed. Mute cell phones during lectures.

During Labs, if you have to take an urgent call, first leave the classroom so as not to disturb the other students.

Talking or otherwise communicating (i.e. texting) with other students during tests will result in a grade of 0. All tests will be open book and open notes. WILL BE STRICTLY ENFORCED.

Homework Rubrics:

Homework is to be submitted via blackboard by the due date

Grading criteria for programming assignments: Neatness and format	20 points
Readme describing how to run program, and intended results	20 points
Commenting in program describing what each part does	20 points
Program Logic	20 points
Working software	20 points

Lab Reports:

Students are expected to prepare for the laboratory before coming to class. Handouts will be provided in person or via Blackboard.

Lab reports must include a cover page, introduction, summary of procedure, design, copy of working software/script and conclusions. Students are to submit laboratory report by the due date either in person or via Blackboard. Clarity plays a major role in the grading of lab write-ups. The instructor will initial work completed in the lab on the handout, which will be included in hard copies of the lab report submitted in class. For reports submitted via blackboard, the initialed work has to be provided to the instructor on request. (It would also be acceptable to scan in the initialed page, and submit it with the electronic copy via blackboard, but it is not required.)

Grading Rubric for Lab Reports are as follows:

Neatness and format (including cover page), initialed work	10 points
Objective & Conclusion(s) stated and clearly tied to procedure	20 points
Procedure described	20 points
Design	15 points
Program	20 points
Build	15 points

Final Project:

The final project is an individual programming and custom build project by each student, chosen and developed by the student. We will discuss this later in the course.

The proposal is a plan that includes your idea/problem and your plan to solve it (suggestions for projects will be supplied). The instructor reserves the right to change or disapprove a proposal. The student must have an approved proposal by the posted deadline.

The paper describes the idea/problem, approach, solution, references and pseudocode is to be included in the proposal. The paper should be at least five pages, including figures such as screen captures of the computer program in action and pictures of robot, if appropriate. There should be at least two references. The code appendix should not be included in the page count of the paper (i.e. five pages plus the code).

The presentation will be allocated 15 minutes, to include a presentation and a demonstration of the software, with 10 additional for minutes for Q&A afterwards.

You are required to use design in your final project. The goal of this project is to develop a working software solution to a problem, and to present it clearly to the class. However, for more challenging projects, generous partial credit will be given for partial solutions that are well described and presented, as well as a plan well followed, but not leading to the solution as anticipated, for 5 reasons uncovered during the pursuit of the solution. In these cases, explaining clearly what you did, what worked, and what didn't, will be an important part of the grade.

Plagiarism Policy

Any of the following acts constitutes an offense of plagiarism:

- Using a phrase, sentence, or passage from another person's work without quotation marks and attribution of the source.
- Paraphrasing words or ideas from another person's work without attribution.
- Reporting as your own research or knowledge any data or facts gathered or reported by another person.
- Submitting in your own name papers, tests, examinations, or reports completed by another person.
- Submitting creative works, including images or reproduction of the creative works, of another person without proper attribution.
- Submitting oral or recorded reports of another without proper attribution.
- Downloading documents in whole or part from the Internet and presenting them as one's own.

Other offenses against academic integrity include the following:

- Collaborating with other students on assignments without the express permission of the instructor.
- Giving one's work to another student who then submits it as his or her own.
- Sharing or copying answers from other students
- Copying material from any digital resource/website during examinations (unless expressly authorized).
- Using notes or other sources to answer exam questions without the instructor's permission.
- Secreting or destroying library or reference materials.
- Submitting as one's own work a paper or results of research purchased or acquired from a commercial firm or another person.

Spring 2018 Semester Schedule

Date 8:30 AM – 11:00 AM	Location	CET 4962 Objective	Lab / Homework
Friday – 02/02	V613	Course Introduction	LAB 1: Lost in Space
Wednesday – 02/07	Online	Some Programming Basics – Scratch Python	
Friday – 02/09	V613	Working with Scratch	LAB 2a: Ghostbusters LAB 2b: ChatBox
Wednesday – 02/14	Online	Programming Logic Basics: <ul style="list-style-type: none"> ▪ Write code in a readable way and/or comments where necessary Create subroutines/ procedures/ functions in their programs ▪ Programmatically read and manipulate a data structure ▪ Apply basic programming constructs to solve a problem ▪ Loops and Conditions 	Homework: Olympic Hurdler
Friday - 02/16	V613		LAB 3a: Dodgeball LAB 3b: Create Your Own World
Wednesday – 02/21	Online	Raspberry Pi 3 Basics <ul style="list-style-type: none"> ▪ NOOBS 	QUIZ # 1 – Programming & Scratch Basics **MUST HAVE RASPBERRY PI AND

		<ul style="list-style-type: none"> ▪ Raspbian <p>Python Basics</p> <p>Sense HAT</p>	SENSE HAT BY THIS CLASS!!**
Friday – 02/23	V613	Raspberry Pi 3 Basics	LAB # 4: Raspberry Pi 3 Setup NOOBS Raspbian
Wednesday – 02/28	Online	Python Basics Continued	
Friday - 03/02	V613	Programming your Raspberry Pi with Python SENSE HAT Emulator	LAB #5: Robot Antenna
Wednesday – 03/07	Online	<ul style="list-style-type: none"> ▪ Use standard communication protocols and APIs to transfer data between computers and applications <p>Apply abstraction and decomposition to solve more complex problems</p>	QUIZ # 2 – Raspberry Pi Basics Python Basics
Friday – 03/09	V613		EXTRA CREDIT LAB: Puzzle Box
Wednesday – 03/14	Online	<ul style="list-style-type: none"> ▪ Decompose a large problem into parts and design algorithms to solve them ▪ Applying programs and working with materials to create a project prototype ▪ Recognize similar problems, and apply generic solutions and abstractions ▪ Effectively combine functionality from multiple libraries or APIs and refer to documentation ▪ Working with Algorithms 	
Friday - 03/16	V613	<ul style="list-style-type: none"> ▪ How to make a paper plate button to trigger an event 	Lab # 6: Whoopi Cushion

		<ul style="list-style-type: none"> ▪ How to program the GPIO pins using loops, lists, and variables 	Or (Choose 1) Burping Jelly Baby
Wednesday – 03/21	Online	<p>FINAL PROJECT DISCUSSIONS</p> <p>Sorting Algorithms</p> <ul style="list-style-type: none"> ▪ Understand the mechanics and performance of the selection sort algorithm ▪ Program a selection sort algorithm in Python ▪ Describe the basic mechanics of the selection sort algorithm ▪ Implement a selection sort in Python ▪ Describe the mechanics of the selection sort algorithm in detail <p>Critically analyse the performance of the algorithm and identify its weaknesses</p>	
Friday – 03/23	V613	<ul style="list-style-type: none"> ▪ To understand how to use a simple list comprehension to create a random list of numbers ▪ To be able to produce simple graphs using matplotlib 	Lab # 7: Graphing with Matplotlib
Wednesday – 03/28	Online	MIDTERM EXAM CUMULATIVE	
Friday - 03/30	V613		Lab # 8: Laser Tripwire continued
Wednesday – 0/04	NO CLASSES		NO CLASSES
Friday – 04/06	NO CLASSES	NO CLASSES	
Wednesday – 04/11	Online	<ul style="list-style-type: none"> ▪ Combine programming constructs to solve a problem ▪ Combine inputs and/or outputs to create projects or solve a problem ▪ Use basic materials and tools to create project prototypes 	<p>QUIZ # 3 – Algorithms</p> <p>FINAL PROJECT PROPOSALS DUE</p>
Friday - 04/13	V613		Lab # 9: Music Box

Wednesday – 04/18	Online	<ul style="list-style-type: none"> ▪ Be a Maker ▪ Create automated systems to solve complex real-world problems ▪ Learners can analyze a problem and design and create automated systems that monitor, react to, or influence an environment, in order to solve a real-world problem. 	
Friday – 04/20	V613		<p>Lab # 10: Create one of the following using Python to program:</p> <ul style="list-style-type: none"> ▪ Create a system that monitors and controls the environment of living things such as fish or plants ▪ Build a self-driving robot buggy that uses an ultrasonic sensor to avoid objects ▪ Produce a mechanical arm that can be monitored and controlled with a program
Wednesday – 04/25	Online	Process multiple data and/or input sources, and use them to control multiple and interconnected output devices	QUIZ # 4
Friday - 04/27	V613		Lab # 10 continued
Wednesday – 05/02	Online	Final Project Focus	
Friday – 05/04	V613	Final Project Focus	
Wednesday – 05/09	Online	<p>FINALS REVIEW</p> <p>Finalizing Your Final Project</p>	
Friday - 05/11	V613	Final Project Test Runs	Final Project Test Runs
Wednesday – 05/16	Online	FINAL EXAM	
Friday - 05/18	V613	FINAL PROJECT PRESENTATIONS ROUND 2	