New York University Tandon School of Engineering

Course Outline **CS-GY 9223 H**
Selected Topics in CS
Advanced Practical Software Engineering in Teams

Spring 2021

Professor Gennadiy Civil

Contact Professor:

Email:
Office hours:

Contact Course Assistant:

Email:
Office hours:

Pre-Work:

- Each student is to complete CS-GY 9223 Spring 2021 Initial Skills Survey.

Commented [1]: @jx692@nyu.edu Could you please check the survey for any issues (I may have left something over from previous year or something) and if OK, please send it out
Assigned to Jack Xu

Course Prerequisites:

1. Successful completion of CS-GY 6063 **OR** Combination of all or some of the following
a) Graduate Status. Undergraduates with appropriate experience may participate at the discretion of the professor. Please submit your resume and email explaining why you are a good fit for this class.
b) Solid ability to program in **Python**. Familiarity with **Django** is helpful. Please note: The course will **not** teach how to program in Python or how to use Django. The students are required to come up to speed on their own. Experience using version control systems (git in particular) is important.
c) Team Work Requirement: This class requires heavy teamwork participation. A student wishing to work "on their own" will not be able to satisfactorily complete the class. Students are expected to meet in their respective teams outside the lecture hours for daily or near-daily scrum meetings. The team meeting minutes have to be reported in writing and submitted to the instructor.

Course Description:

As a software engineer, one is often asked to get up to speed and start maintaining and developing an already existing codebase. In this class we will learn how to take previously developed proof of concept software projects to “production readiness”.

Course Objectives:

- Learn how to be a good teammate and work as part of the Software Development Team
- Get exposure to Code Archaeology and how to maintain and enhance a previously developed codebase
- Get exposure to Open Source Development
- Gain exposure to practical approaches for Agile development
- Gain exposure to continuous integration techniques
- Learn efficient unit testing and test-first development principals
- Learn how to write readable and maintainable code, instead of code that “just works”

Course Structure

Weekly lectures/labs plus a significant practical hands-on project.

- The project work will follow Agile methodology. The students are organized into Scrum teams. Each Scrum Team follows development Sprints with clearly identified, written down and reported Sprint goals.
- Each team will be presenting a Sprint demo at the end of each Sprint during the class time.
- There is constant communication expected between the team members within the teams. The teams are expected to follow daily or near-daily scrum meetings. The daily scrum meetings minutes are to be submitted to the instructor.

Hands-On Software project:

- Format: Working software accessible over the Internet.
- All project code deployed to GitHub
- All project code complies with established Code Style guidelines
- High percentage of the project code has to be covered by a Test Suite as measured by the test code coverage percentage

Readings

Recommended books:

Essential Scrum: A Practical Guide to the Most Popular Agile Process (Addison-Wesley Signature): A Practical Guide To The Most Popular Agile Process (Addison-Wesley Signature Series (Cohn))
Link to purchase the book

Grade Calculation

The class will be graded as follows:

- 20% Successful Delivery of the shared class project. (Note that there is one class project that will be delivered by the entire class).
- 60% - Teamwork
  - 360 Team mutual evaluation forms
  - GitHub code and code reviews statistics.
- 10% Individual Homework Assignments
- 10% Class Participation
  - Feedback Forms
  - General class participation
- Extra credit for participation in the release team in addition to working on your agile team

Course requirements

Very few widely-used products are built by an individual, so students must learn to collaborate with others both in design and construction of the codebase. Software engineering is a team effort. ThisPractical Software Engineering classes **requires** heavy teamwork participation.

Teamwork plays a large role in your grade for this course. Team members will be evaluating each other throughout the course. Poor peer reviews will have a sizable negative effect on overall class grade and may even cause the student to fail the class altogether.

Pre-Work:

- Each student is to complete CS-GY 9223 Spring 2020 Initial Skills Survey.

Class 1: Class Introduction.
Class overview, Grading overview, Emphasis on teamwork. Class Slack Workspace and explanation of communication. (Class Central document explained)
Introduction to the problem domain - demo.
Introduction to the existing codebase.

- Assignment given, **Code Archeology 1**:
    - Individual Assignment, All Students.
        - Deploy and Run the existing code to the new individual Code Repository on GitHub
        - Add minimal new functionality to the existing code and deploy to cloud hosting

Reading(Optional): Django Tutorial.
https://docs.djangoproject.com/en/2.2/intro/tutorial01/

Class 2: Practical Architecture and Design Overview

Note: Team Formation will have been announced prior to the class. The students come to class knowing which team they are on.

Team icebreaker exercise. Practical Design/Architecture overview. Design Diagrams for this class/Just enough UML

- (Lab) Team Icebreaker Exercise
- (Lecture) Initial System design for team projects. Potentially useful UML diagrams explained
- (Team Lab) Reverse engineer an existing small body of code to produce UML sequence diagram.

- Assignments given:
    - Assignment 1: Team Code Archeology 2. Produce design documentation for the existing repo reverse-engineered from the codebase
    - Assignment 2**:** Produce Team Design Proposal for the class project. Choose one of the following **and explain why and how**.
        - 1: Keep the existing structure and design
        - 2. Change it
    - Reading: UML sequence diagram explained: https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/

Note: Feb XX - last day to drop/add classes according to the NYU academic calendar.

Class 3: Design Bake-Off, class discussion

- (Lab) Class discussion - Bake Off for the Design Proposals produced by the teams in the previous Sprint. Each team has 20 minutes to present and explain their proposed design. + 10 minutes for questions.
- Live voting for the proposed design
- (Lecture) Requirements Management as related to the class project
- (Lecture) Technical Stories and how we will use them for this class.
    - API design document as a form of a technical story
    - Unit test as a form of a technical story
- Assignment given - team assignment:
    - Formalize winning design: (**need volunteers to form the release team** or Product Owners from each team get together). **Requirements Archeology.**, Review and evaluate existing stories attempting to extract project requirements

Class 4 Agile Process Refresher. Class development workflow overview.

Agile development overview. Scrum process overview and essential Scrum ceremonies. Directions for students teams. Daily Scrum meeting reporting requirements. What is the role of Scrum Master? What is the definition of Done?

- (Lab) Class discussion, go over user stories that the teams analyzed as per this class assignment. Discuss, review questions.
- (Lecture) ZenHub Refresher
- (Lecture) Sprint Planning. Agile Poker.
- (Lab) Teams specialization discussion and decision
    - Who is the overall Product Owner for this project?
- Assignment given: Sprint 0:
    - Sprint Goals, teams (each team):
        - Team Repository created, connected to Travis CI, Connected to Coveralls, Connected to ZenHub, master branch build with green Travis CI Badge
        - Document team-specific **Epics** and **Stories** for the project for each team's ZenHub
        - Sprint 0 starts. Teams proceed with Sprint Planning and execution.

Class 5: Test-Driven Development

- (Lab) Sprint Demos, all teams. Class discussion
- (Q/A) Any questions on the agile procedures that the class/teams are supposed to follow?
- (Lecture) Test-Driven development
- Assignment given: Sprint 1:
    - Sprint Goal:
        - 95+% test coverage for all team repos

- Reading: Testing in Django. https://docs.djangoproject.com/en/2.2/topics/testing/

Class 6: Better Code Review practices
(Lab) Sprint Demos, all team. Class discussion about tests.

- Assignment given: Sprint 3:
    - Sprint Goal: Stories TBD as per Sprint Planning
    - Add fuzzing to the project testing

- Reading: https://testing.googleblog.com/2019/12/testing-on-toilet-tests-too-dry-make.html

Class 7: Readability, Maintainability, Code Smells, Refactoring

- (Lab) Sprint Demos, all teams
- (Lecture) Writing code for people, not machines, and an argument for constant refactoring
- Assignment given: Sprint 2:
    - Sprint Goal: Find an example of "bad" or "smelly" code, ensure there's a test for it (or write a new one), then refactor that code to be more readable/maintainable/etc.

- Reading: https://hypothesis.readthedocs.io/en/latest/ and https://hypothesis.readthedocs.io/en/latest/django.html

Class 8: No Class, Spring Recess.

Class 9: More test-driven development. DRY, DUMP Testing, Fuzz Testing, Property-Based testing.

- (Lab) Sprint Demos, all teams
- (Lecture) Open Source Development, how to structure your open source repository. LICENSE, README, CONTRIBUTING guidelines, etc
- Assignment given: Sprint 5:
    - Sprint Goal: Stories TBD as per Sprint Planning
    - Proper Open Source license and managing files added to the class project

Class 10:

- (Lab) Sprint Demos, all teams
- (Lecture) Software Versioning. Semantic versioning: a promise and a lie.
- Assignment given: Sprint 6:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 11: Midterm

- (Lab) Sprint Demos, all teams -- **Midterm Presentations, all teams**
- Assignment given: Sprint 4:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 12:

- (Lab) Sprint Demos, all teams
- Assignment given: Sprint 7:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 13:

- (Lab) Sprint Demos, all teams
- Assignment given: Sprint 8:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 14:

- (Lab) Sprint Demos, all teams
- Assignment given: Sprint 9:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 15:

- (Lab) Sprint Demos, all teams
- Assignment given: Sprint 10:
    - Sprint Goal: Stories TBD as per Sprint Planning

Class 16, - last class, class project due.

- (Lab) Project Demos, all teams. FINAL Presentation
- All teams - make a git tag 1.0
- Release team - make a GitHub release version 1.0

Moses Center Statement of Disability

If you are a student with a disability who is requesting accommodations, please contact New York University's Moses Center for Students with Disabilities (CSD) at 212-998-4980 or mosescsd@nyu.edu. You must be registered with CSD to receive accommodations. Information about the Moses Center can be found at www.nyu.edu/csd. The Moses Center is located at 726 Broadway on the 3rd floor.

NYU School of Engineering Policies and Procedures on Academic Misconduct – complete Student Code of Conduct here

A. Introduction: The School of Engineering encourages academic excellence in an environment that promotes honesty, integrity, and fairness, and students at the School of Engineering are expected to exhibit those qualities in their academic work. It is through the process of submitting their own work and receiving honest feedback on that work that students may progress academically. Any act of academic dishonesty is seen as an attack upon the School and will not be tolerated. Furthermore, those who breach the School's rules on academic integrity will be sanctioned under this Policy. Students are responsible for familiarizing themselves with the School's Policy on Academic Misconduct.

B. Definition: Academic dishonesty may include misrepresentation, deception, dishonesty, or any act of falsification committed by a student to influence a grade or other academic evaluation. Academic dishonesty also includes intentionally damaging the academic work of others or assisting other students in acts of dishonesty. Common examples of academically dishonest behavior include, but are not limited to, the following:

    1. Cheating: intentionally using or attempting to use unauthorized notes, books, electronic media, or electronic communications in an exam; talking with fellow students or looking at another person's work during an exam; submitting work prepared in advance for an in-class examination; having someone take an exam for you or taking an exam for someone else; violating other rules governing the administration of examinations.

2. Fabrication: including but not limited to, falsifying experimental data and/or citations.
3. Plagiarism: intentionally or knowingly representing the words or ideas of another as one's own in any academic exercise; failure to attribute direct quotations, paraphrases, or borrowed facts or information.
4. Unauthorized collaboration: working together on work meant to be done individually.
5. Duplicating work: presenting for grading the same work for more than one project or in more than one class, unless express and prior permission has been received from the course instructor(s) or research adviser involved.
6. Forgery: altering any academic document, including, but not limited to, academic records, admissions materials, or medical excuses.

NYU School of Engineering Policies and Procedures on Excused Absences – complete policy here

A. Introduction: An absence can be excused if you have missed no more than 10 days of school. If an illness or special circumstance has caused you to miss more than two weeks of school, please refer to the section labeled Medical Leave of Absence.

B. Students may request special accommodations for an absence to be excused in the following cases:

1. Medical reasons
2. Death in immediate family
3. Personal qualified emergencies (documentation must be provided)
4. Religious Expression or Practice

Deanna Rayment, deanna.rayment@nyu.edu, is the Coordinator of Student Advocacy, Compliance and Student Affairs and handles excused absences. She is located in 5 MTC, LC240C and can assist you should it become necessary.

NYU School of Engineering Academic Calendar – complete list here.

If you have two final exams at the same time, report the conflict to your professors as soon as possible. Do not make any travel plans until the exam schedule is finalized.
Also, please pay attention to notable dates such as Add/Drop, Withdrawal, etc. For confirmation of dates or further information, please contact Susana: sgarcia@nyu.edu