

ITIS 6010/8010: Software Engineering for AI-Enabled Systems

(Spring 2025)

Course Syllabus

Faculty Information

Instructor:	Rrezarta Krasniqi, Ph.D., Assistant Professor
Email:	rrezarta.krasniqi@charlotte.edu (preferred way of contact)
Office:	Woodward 333G
Office Hours:	Monday: 11:00–1:00 p.m.
Course Hours:	Monday: 2:30–5:15 p.m.
Instructional Method:	Face-to-Face
Campus:	Main Campus (Mebane Hall, #169)

Course Information

Credit Hours:	3 credit hours
Required Textbook:	None but the recommended ones are: (1) Building Intelligent Systems by Geoff Hulten (2) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron (3) Designing Data-Intensive Applications by Martin Kleppmann
Course Material:	Lecture notes and additional supplemental materials will be referenced and provided via Canvas Course Management System

Course Description

This course bridges software engineering and AI development, focusing on key aspects such as understanding AI model requirements and constraints, strategically planning AI component integration within AI-enabled systems, and ensuring responsible and reliable development using machine and deep learning techniques. It emphasizes the role of software architecture and quality assurance in maintaining robust models beyond accuracy across various development stages while mitigating the effects of data drift. The course also highlights the importance of explainability and interpretability, moving beyond the traditional "black box" view of AI models. Additionally, students will gain hands-on experience with real-world scenarios, including debugging, design considerations, and thorough system-level testing.

Topics Covered in this Course

The topics covered in this course include:

- Software requirements for AI
- Software Development with ML and DL perspective
- Software Architecture for AI
- Software Quality Assurance for AI
- Explainability and interpretability
- Software Debugging for AI
- Software Testing for AI

Expected Learning Outcomes

Upon the completion of this course, among others, students should be able to:

- Understand system quality goals and the lack of requirements for AI-components
- Understand and construct risk-mitigating plans on how to develop robust AI components
- Identify and measure quality attributes beyond model accuracy
- Analyze tradeoffs for designing software systems with AI-components
- Learn how to detect poor data quality, poor model quality, and capture data drift phenomena
- Build robust AI-components through version control, provenance, and test automation

Course Structure and Teaching Methods

The course is designed with a weekly progression that integrates lectures, small group activities emphasizing experiential learning and regular in-class paper reading discussions. The course schedule outlines the expected weekly progress. Students are strongly advised to follow the outlined schedule on Canvas to stay updated with coursework and any tentative syllabus changes. Please note that all deadlines for course deliverables are posted on Canvas. The instructor will **not** make recordings of lectures available unless she is required to attend/present in a Research Conference.

Course Communications

Regular announcements and important reminders will be posted on Canvas. Please log in regularly to stay updated on new announcements, reminders, and course-related information. Throughout the semester, feel free to contact your instructor via email or during office hours. Every effort will be made to address your questions or concerns promptly, typically within 24 hours or sooner whenever possible.

Course Credit Workload

This 3-credit course involves **three** hours of online virtual classroom sessions led by faculty each week, along with an additional six hours of weekly out-of-class student work over a **15-week period**. Out-of-class activities may include readings research papers, programming tasks, and active group projects participations.

Prerequisites

Because this is a graduate upper-level course, it does not have formal prerequisites. But a foundational understanding of machine learning and deep learning, including **intermediate programming** skills are essential for successful project completion which covers majority of course grade. Prior software engineering experience is not required.

Machine Learning/Deep Learning Experience

Basic familiarity with the process of feature selection, feature extraction, ML/DL model building, and evaluation model techniques is recommended. Prior experience with Python and/or platforms such as Jupyter Notebooks is helpful.

Programming Experience

This is an upper-level graduate course with a strong emphasis on programming. To succeed, you should be proficient in at least one programming language. While the course does not require a specific language, Python is commonly used by student teams. You should be able to install and use necessary libraries (e.g., scikit-learn) and have a solid understanding of version control systems such as Git. Additionally, familiarity with Unix/Linux environments and basic command-line operations is essential. If you lack these skills, you may face significant challenges in completing the course team project and may need to dedicate extra time to develop them independently.

Paper Reading Assignments and Team Project

Paper reading assignments will be assigned on a weekly basis, focusing on course topics. Expect around eight paper reading assignments along with a semester project comprising several subtasks. Throughout the semester, module-based class activities will promote active participation. The course will have **neither a midterm nor a final exam**. Instead, the focus will be on the realization of a multi-milestone team project. The project will be set in teams of 2-4, which will give you the opportunity to conduct a project with broader impact. The project will be finalized with a high-quality written report. These course projects aim for you to develop a new tool using ML or DL techniques, tweak existing ML or DL with the SE domain, conduct an extensive re-evaluation or empirical study of an existing tool, re-implement an algorithm within the SE domain or create a benchmark suite. If you are unsure if a project is in scope, feel free to schedule a meeting or attend office hours with me prior to submitting the project proposal.

Canvas Course Management System

This course uses the UNC Charlotte Canvas course management system. It is mandatory to use Canvas for this class, and regular daily checks are strongly recommended. Canvas will serve as the platform for assignments, projects, exams, content distribution, and crucial announcements. Access the UNC Charlotte Canvas at: <https://canvas.charlotte.edu>.

Grading Criteria

Course grading criteria will be based on the following activities.

Activities	Percentage
Paper Reading Assignments (8)	30%
Team Project (several deliverables)	60%
Attendance (mandatory)	10%
Total	100%

Grading Scale

The following grade scales will be used to award the final grades:

Grade Scale	Letter Grade
90% – 100%	A
80% – 89.9%	B
70% – 79.9%	C
60% – 69.9%	D
<60%	F

Course and Grading Policies

- Paper reading assignments and project deliverables are due by the deadlines specified in the course schedule or on **Canvas** content pages.
- Late Submissions:** Reading Assignments and project task completions submitted late (as indicated by the Canvas timestamp and “late” flag) **will receive a grade of 0**. Project team members should adequately plan to complete group projects throughout the semester. In the event of an emergency that significantly impedes a group's ability to meet a deadline, permission and documentation must be obtained **BEFORE** the due date at least a week in advance, along with a plan for submission after the due date.
- Attendance:** Attendance in class is **mandatory** and requires being present and attentive for the entire duration of each session. Attendance will be recorded at every class. Missing more than **four classes will result in a**

University of North Carolina at Charlotte
College of Computing and Informatics
Department of Software and Information Systems

(U) grade for this course. Regardless of absences, students are responsible for all materials and assignments covered. Absences must be explained via email **sent to the instructor** before the start of the class.

- **Grade Disputes:** Any concerns regarding grading must be raised within **one week** of the grade being posted on Canvas. If you believe an assignment was graded incorrectly and it was in fact graded correctly, we aim to resolve any confusion promptly to ensure comprehension of the material. If an error in grading occurred, we will promptly acknowledge the mistake and adjust the score accordingly.
- **Grade Requests:** Requests for incomplete grades **(I)** and withdrawals **(W)** must adhere to University policies, which can be found [here](#) for incomplete grades and [here](#) for withdrawal procedures.

Statement of Copyrighted Materials

All materials provided by the instructor and other course materials, whether distributed in class or online, are original works and represent the intellectual property of the instructor or the author (excluding published reference materials such as course textbooks). These materials, including readings, lecture notes, tutorials, and/or handouts are intended solely for the individual use of students. You are not permitted to distribute or reproduce these materials for commercial purposes without the explicit written consent of the instructor. Students who sell or distribute these materials for any purpose other than their own use are violating the University's Copyright Policy available [here](#). Breaching the instructor's copyright may lead to course sanctions and violate the Code of Academic Integrity.

Medical, Personal or Family Emergencies

In case of emergencies that significantly impact your ability to complete coursework, please immediately contact the Office of the Dean of Students available [here](#). The Dean of Students can provide support and advocate for you to request appropriate accommodations across your classes. To ensure fairness for all students, I require everyone facing emergencies to use the Dean of Students resources. This ensures a consistent and documented process for handling such situations.

University Academic Policies and Procedures

Academic Policies and Procedures

As a student at UNC Charlotte, it is your responsibility to understand and adhere to the university's policies and procedures. The complete document can be accessible [here](#).

Code of Student Responsibility

The UNC Charlotte Code of Student Responsibility outlines specific rights and responsibilities related to student discipline. It defines these responsibilities and ensures your rights are protected, safeguarding you against unfair imposition of disciplinary actions. It is important to familiarize yourself with the procedures outlined in the Code. The complete document can be accessible [here](#).

Academic Integrity

This course maintains a strict policy against violations of academic integrity. It is essential that all individual work, such as **paper reading assignments**, be completed independently. Cheating or discussing solutions, code sharing with others constitutes a serious offense and will result in automatic failure of the course. Collaboration that does not meet acceptable standards will be treated as a breach of the Code of Student Conduct and will also lead to a failing grade. Furthermore, the **use of all pre-built web hosting platforms (e.g., Squarespace, Wix, Weebly)**, and all multi-cloud hosting services **(e.g., AWS, GCP)**, is **strictly prohibited for completing or facilitating your team projects**. All your projects need to be implemented from scratch, and they will be monitored by the Instructor in the GitHub repository. Any violation of this policy will be considered a breach of the UNC Charlotte Code of Academic Integrity and Academic Dishonestly and will be subject to the relevant disciplinary actions such as failing a class. Any violations of the Code of Student Academic Integrity, including instances of plagiarism, will result in disciplinary measures outlined in the Code. Definitions and examples of plagiarism and other violations can be found in the Code, accessible through the Dean of Students Office [here](#).

University of North Carolina at Charlotte
College of Computing and Informatics
Department of Software and Information Systems

Academic Dishonesty

Refers to any action or attempted action that unfairly advantages or disadvantages a student or any member of the academic community. Such actions are subject to sanctions under the Code of Academic Integrity, which may include written warnings, grade reductions, disciplinary probation, loss of credit for assignments, course failure, suspension, and/or expulsion.

Academic Misconduct

Refers to any behaviors that do not align with established standards or rules within the academic community. Such misconduct is subject to sanctions under the Code of Conduct, including restricted access to university facilities, administrative holds, warnings, probation, suspension, and/or expulsion. Examples of academic misconduct include disruptive behavior, threats, theft, or damage to university property, among others. For further details and examples, please visit UNC Charlotte's Academic Misconduct Policies page [here](#). The UNC Charlotte Writing Resources Center offers valuable tips on avoiding plagiarism, accessible [here](#). Selling course materials to other students or third parties without the instructor's explicit consent is strictly prohibited and constitutes a violation of the Code of Academic Integrity. Violations, including misuse of Canvas or NinerNet email for such purposes, may also constitute copyright infringement and will be subject to disciplinary action under the Code of Conduct.

Classroom Etiquette Policy

To create a positive learning environment, both students and the instructor share responsibility. We aim for a safe, welcoming, and inclusive environment where everyone feels comfortable and encouraged to succeed. The UNC Charlotte's Canvas Netiquette Guidelines accessible [here](#) offers communication guidelines for composing, sending, and receiving emails and other means of communication. Therefore, our focus is on the tasks at hand and following to guidelines that ensure everyone can participate and collaborate productively. Here are some additional guidelines:

- Avoid engaging in activities: **making phone calls/texting, loud chatting, raising voice, web surfing**, etc.
- During face-to-face interactions, it is essential to select your words thoughtfully, articulate your sentences clearly, and remain focused on the subject matter.
- Exercise caution when using humor or sarcasm in emails or discussion posts, as digital communication may make it challenging to accurately interpret tone.
- Share personal experiences using “I” statements to convey thoughts and feelings. Refrain from speaking on behalf of groups or individuals' experiences.
- Apply critical thinking skills to challenge others' ideas rather than targeting individuals personally.
- Students are encouraged to respectfully question research findings or classmates' opinions, as this is a natural part of academic discourse. However, it is important to refrain from disparaging others' views. Please be aware that this course adheres to university policies regarding disruptive behavior. For more information, please refer to UNC Charlotte's Policy [here](#).

Prohibition of Discrimination

UNC Charlotte firmly prohibits discrimination and harassment based on race, color, national origin, religion, sex, sexual orientation, gender identity, gender expression, age, disability, genetic information, veteran status, or any other characteristic protected under federal or state law. This policy extends across all application and admission processes, educational programs and activities, employment policies and procedures, and university facilities. The University actively takes measures to prevent such conduct, investigating and taking appropriate remedial actions when necessary. For more information, please visit UNC Charlotte Policy [here](#).

Sexual Assault Prevention

All students are required to comply with the UNC Charlotte Sexual Harassment Policy that is found [here](#) and the policy regarding Responsible Use of University Computing and Electronic Communication Resources that can be found [here](#). Sexual harassment, as defined in the UNC Charlotte Sexual Harassment Policy, is strictly prohibited, including instances occurring through computers or other electronic communication systems, such as course-based chat rooms or message boards.

University of North Carolina at Charlotte
College of Computing and Informatics
Department of Software and Information Systems

Special Need Accommodation

If you have a documented disability and require accommodation in this course, contact Disability Services at Fretwell 230, email questions to disability@charlotte.edu or call 704-687-0040 (tty/v) during the first week of the semester. Information about available services can be found [here](#). The Disability Services office will coordinate accommodations for your learning needs and communicate them to your instructor.

Religious Accommodation

Students are responsible for informing faculty in advance about their planned absences due to religious observances. This should be done by submitting a Request for Religious Accommodation Form to their instructor before the census date of each semester. You can find the census date, usually the tenth day of instruction, listed in UNC Charlotte's Academic Calendar [here](#). For more details, refer to the policy [here](#).

Student Assistance and Support Services

Any student who has difficulty affording groceries or accessing sufficient food to eat every day, or who lacks a safe and stable place to live or needs to report an emergency situation and believes this may affect their performance in the course, is urged to contact Student Assistance and Support Services (SASS) (<http://sass.charlotte.edu>) under the Dean of Students Office for support. Furthermore, please notify the instructor if you are comfortable in doing so. This will enable her to provide any resources that she may possess.

Retention of Student Records

Student records for this course are securely maintained by the instructor. This includes exams, answer sheets (with keys), and written papers submitted during the course, which are kept for at least one year after the course ends. Additionally, coursework completed through Canvas, including grades and comments, is stored electronically in a secure environment for one year. Students have the right to access their individual records, but information about a student's records will not be disclosed to others without proper written consent. Students are encouraged to review the Public Information Policy, the Family Educational Rights and Privacy Act (FERPA), and UNC Charlotte Policy 605.3 on Retention, Disposition, and Security of University Records for more information.

University Emergency Notification and Procedures

UNC Charlotte uses NinerAlert to promptly inform students about critical emergencies such as severe weather, campus closures, and public safety threats like chemical spills, fires, or violence. For more information, visit [NinerAlerts](#). If the university closes, please check Canvas for contingency plans to cover course materials.