IS 601 – Python for Web API Development

Course Information

Item	Details
Section / CRN	101 / 93875
Semester	Fall 2025
Credits	3
Delivery Mode	Face-to-Face
Days / Times	Monday, 6:00 – 8:50 PM
Location	FMH 207
Status	Open (35/33 enrolled)
Special Note	Students must bring their own device for this section

---

Instructor Information

Item	Details
Instructor	Keith Williams
Office	GITC 3410
Email	kwilliam@njit.edu *(Discord preferred)*
Office Hours	Monday 1:00 – 5:30 PM (before class), Wednesday 2:30 – 3:30 PM (after IS 373)

Course Description

This course introduces Python programming through the lens of web data transfer and API development. Students will gain experience with Python syntax, data structures, testing, and object-oriented design, while building applications that interact with REST APIs, databases, and containerized environments. By the conclusion of the course, students will have created a complete backend web service that supports creating, reading, updating, and deleting objects stored in a database.

Prerequisites: None

Course Technologies

- Python 3.x
- FastAPI
- Docker
- Git & GitHub
- VS Code (recommended IDE)
- SQLite/PostgreSQL
- Pydantic & pytest

Learning Outcomes

By the end of this course, students will be able to:

1. Utilize Git for version control and collaborative development.
2. Navigate and execute basic commands in a Linux environment.
3. Create Python applications with automated testing.
4. Set up GitHub Actions for CI/CD pipelines with testing and Docker builds.
5. Develop a command-line application using the REPL pattern.
6. Implement object-oriented programming principles in Python.
7. Apply professional terminology and concepts related to web systems development.
8. Create and manipulate CSV files using Python.
9. Containerize applications using Docker.
10. Create, consume, and test REST APIs in Python.
11. Integrate Python programs with SQL databases.
12. Serialize, deserialize, and validate JSON using Pydantic.
13. Apply secure authentication, authorization, and encryption techniques.

Grading Policy

Component	Weight
Assignments & Reflections	20%
Quizzes	10%
In-Class Participation	10%
Midterm Project	20%
Final Project	20%
Midterm Exam	10%
Final Exam	10%

Grading Scale

- A: 90–100
- B+: 85–89
- B: 80–84
- C+: 75–79
- C: 70–74
- F: 0–64

Late Policy

- Assignments lose 10% per day late.
- No work accepted after 4 days without documentation.

Tentative Weekly Outline

Week	Topic	Deliverables
1	Setting up environment: Git & Linux basics	Assignment 1
2	Python syntax, testing, VS Code setup	Assignment 2

Week	Topic	Deliverables
3	OOP and advanced testing in Python	Quiz 1, Assignment 3
4	Advanced OOP & design patterns	Assignment 4
5	Data management with pandas	Quiz 2, Assignment 5
6	Midterm Exam & Project Work	Midterm Exam
7	Docker and containerization	Midterm Project
8	Web applications & FastAPI basics	Assignment 6
9	Databases & Python integration	Quiz 3, Assignment 7
10	Security, Pydantic, CI/CD	Assignment 8
11	Modeling & validation with Pydantic	Quiz 4, Assignment 9
12	Implementing routes & testing APIs	Assignment 10
13	JWT authentication, Playwright testing	Assignment 11
14	Completing CRUD for final project	Final Project
15	Final Exam & Wrap-up	Final Exam

Generative AI Policy

Students are encouraged to use tools like ChatGPT to **ask questions, debug, and brainstorm**.

- Do **not** copy-paste code you don't understand.

- You are responsible for every line of code you submit.
- Proper use of AI will make you faster and more effective; misuse will leave you unprepared for exams.

Academic Integrity

The NJIT Honor Code will be strictly enforced. All work must be your own. Collaboration is limited to peer discussions; submitted code must be written and understood by you. Plagiarism or code-copying will be referred to the Dean of Students.

Instructional Philosophy

This course emphasizes **hands-on coding** and **practical application**.

- Learn by doing, not memorizing.
- Ask clear, specific questions.
- Expect to fail fast and fix quickly.
- Treat GitHub as your portfolio: your work here will be useful to show employers.