

ENGM 4620 – Python for Engineers

Group Project #3: The Development of a Practical Python Package – Part #2

For Project #3, you are presented with multiple paths to choose from. Regardless of the option you select, ensure your final submission is comprehensive and follows the submission instructions closely. This project is final, and submissions are not returnable.

Option #1: Enhance Project #1 (No Web Interface Required)

- Objective: Build upon Project #1 by introducing significant enhancements and new features.
- Requirements:
 - Your enhancements should aim for at least a 150% expansion in scope or functionality compared to the original project.
 - Documentation or an explanatory video must clearly detail the newly added features.
 - Incorporation of data analytics (using tools like NumPy, Pandas, Matplotlib) can be an option where applicable (only if it makes sense to your project).
- Evaluation: Projects will be assessed based on the scale and impact of the enhancements.

Option #2: Add a Web Interface to Project #1 Using Django

- Objective: Develop a web interface for Project #1, facilitating interaction through a browser.
- Requirements:
 - Utilize Django to create the web interface.
 - The web application must include at least two models (to create database tables) and three views (web pages), excluding any pages that confirm successful form submissions.
 - A local deployment of the web application is sufficient.
 - Include screenshots of the web application in your report and demonstrate it in your video presentation.
- Evaluation: Projects will be judged based on functionality, design, and adherence to the specified requirements.

Option #3: Create a New Project from Scratch

- Objective: Start anew by developing a completely original project, mirroring the opportunity provided by project #1 but with a fresh concept.
- Requirements: The new project should be as comprehensive and detailed as the original assignment, showcasing a new idea and execution.
- Evaluation: The novelty and execution of the project idea will be key criteria in its assessment.

Submission: For the final project, there is no need to fill out a common Excel sheet nor upload on a public URL. Just submit your files via Brightspace. You can work with the same partner (if applicable).

Team Size: Maximum of 2 students per group (can be a single student as well)

Topic Selection: Students should come up with their own idea. Please check the project examples for clarity.

Evaluation Criteria: Like project #1, this project is evaluated subjectively against four criteria: Innovation, Functionality and Reliability, Code Quality and Personal Contributions, and Documentation and Video Quality. Below are the details of each criterion. Please note that the evaluation will take into consideration all received peer-review reports from other students.

1. Innovation and Originality (5 points):

- Creativity in the project concept and the introduction of new features.
- Extent of modification and enhancement from any existing code used (if applicable).
- Documentation and comparison with some existing solutions which solves the same problem.

Note: For this project, **"Innovation and Originality"** means **applying your own unique approach to the solution, not necessarily inventing something completely new**. This includes introducing your own simplified version of other existing solutions.

2. Functionality and Reliability (10 points):

- **Effective Utilization of Python Features:** Review the proper use of Python programming features, including but not limited to Object-Oriented Programming (OOP) principles, functions, and list comprehensions, where applicable

- **Implementation Precision:** Assess the accuracy and precision in implementing project features, emphasizing how well they align with the intended documented functionality.
- **Error Handling:** Examine the project's robustness in handling various error scenarios.
- Any other aspects related to the functionality and reliability of the code

Deployment: The code should run error upon deployment by the peer-review evaluator. This means defining the package requirements using requirements.txt appropriately (if applicable) and ensuring you have a clear **documentation** on how to use the code.

3. Code Quality and Personal Contribution (5 points):

- Clarity, readability, code comments, and organization of the code.
- Assessment of personal contribution and effort in coding, beyond basic modifications.
- Documentation of each group member's role and the extent of their contribution.

4. Documentation and Video Quality (5 points):

- Quality of the project report, reflecting the development journey and decision-making process.
- Effectiveness and clarity in communicating the project's purpose, features, and during the video.
- Organization, visual appeal, and engagement level of the video.
- Clarity of other project documentation aspects, including requirements.txt, README files, and test-use cases.

Important Note: (utilizing an existing online repository):

You have the option to enhance (or redevelop in a simplified fashion) an existing online repository for your project instead of starting from scratch. However, please note that merely reusing a repository with minimal alterations will have a substantial impact on the assessment for categories 1 to 3. Your evaluation will primarily focus on the value-added contributions you introduce.

If you choose to utilize an existing repository, it is essential to transparently cite it in your project report by providing the repository link and offering comprehensive details about the modifications and enhancements you have implemented.

Video Presentation Requirements

1. **Duration and Format:** The video must be 5-7 minutes long, featuring clear audio and visual quality. It should be well-edited for smooth flow and coherence.
2. **Content Overview:** Include an introduction to the project, a high-level summary of its purpose, key functionalities and features, technical aspects, personal contributions/teamwork, and a brief conclusion with potential future developments.
3. **Presentation Style:** Ensure the presentation is engaging, with effective communication of the project's significance and features. Visual aids or demonstrations should be used to enhance understanding.

Evaluation Metric:

Each aspect of your project will be evaluated based on the following criteria: functionality and reliability, originality and creativity, code quality and organization, documentation quality, and teamwork and collaboration. Each criterion will be rated on a scale of 1 to 5, with the following guidelines.

Note: the scale will be from 1 to 10 for 10 points questions.

- **No Answer (0):** No attempt made or no relevant response provided.
- **Limited (1):** The aspect demonstrates poor quality, largely incomplete or inaccurate. Shows minimal understanding or effort.
- **Basic (2):** Below-average quality. The aspect is partially complete but lacks detail or precision. Indicates some understanding but with notable gaps.
- **Adequate (3):** Satisfactory quality. The aspect meets basic requirements and demonstrates a fair understanding with room for improvement.
- **Proficient (4):** Good quality. The aspect is well-executed, detailed, and mostly precise. Shows a strong understanding and capability.
- **Excellent (5):** Exceptional quality. The aspect is thorough, detailed, precise, and demonstrates an exceptional understanding and high-level proficiency.

Submission Guidelines: Two submissions are required: one using Brightspace for assessment purposes and another for peer-review sharing since students cannot access each other's files on Brightspace.

First: Students should submit the following to Brightspace:

1. **PDF Report:** Prepare a comprehensive project report in PDF format. The report should document the project's concept, development journey, decision-making process, and any relevant details. Ensure that the report is well-organized and includes clear explanations of project functionalities. **If opted in for option #1, your report should clearly indicate the new features introduced.**

2. **Video Presentation:** Create a 5-7 minute video presentation that effectively communicates the project's purpose, features, and uniqueness. Please refer to the video guidelines for specific requirements. **If opted in for option #1, your report should clearly indicate the new features introduced.**
3. **Source Code Upload:** Upload a copy of the project's source code to Brightspace. The code can be in Jupyter Notebook format or as .py files, depending on the project structure. Ensure that the code is well-commented and organized for readability.

Submission Deadline: The submission deadline for the project is **April 8th** (11:59pm), with a two-day grace period allowed without the need for permission. Please note that this submission is nonreturnable.

Late Submission Policy: Unapproved late submissions beyond the two-day grace period will result in a penalty of 2.5 marks loss per day (equivalent to a 10% deduction). Additionally, continued late submissions may result in the exclusion of the project from the peer-review process.