

Team Project:

Your team project should include the following:

- Microservice
 - Build a microservice that interfaces with a data pipeline. You can choose Python or Rust for development. The microservice should include logging and be containerized using the Distroless Docker image. A Dockerfile must be included in your repository.
- Load Test
 - The microservice must be capable of handling 10,000 requests per second. A load test verifying this performance should be included.
- Data Engineering
 - Your project should involve the use of a library specializing in data engineering such as Spark, Pandas, SQL, a vector database, or any other relevant library.
- Infrastructure as Code (IaC)
 - Your project must utilize an IaC solution for infrastructure setup and management. You can choose among AWS CloudFormation, AWS SAM, AWS CDK, or the Serverless Framework.
- Continuous Integration and Continuous Delivery (CI/CD)
 - Implement a CI/CD pipeline for your project. It could be through Gitlab Actions or AWS Cloud Build or any other relevant tool.
- README.md
 - A comprehensive README file that clearly explains what the project does, its dependencies, how to run the program, its limitations, potential areas for improvement, and how AI Pair Programming tools (Github Copilot and one more tool of your choice) were used in your development process.
- Architectural Diagram
 - A clear diagram representing the architecture of your application should be included in your project documentation.
- Github/Gitlab Configurations
 - Your Github/Gitlab repository must include Github/Gitlab Actions and a .devcontainer configuration for Github/Gitlab Codespaces. This should

make the local version of your project completely reproducible. The repository should also include Github/Gitlab Action build badges for install, lint, test, and format actions.

- Teamwork Reflection
 - Each team member should submit a separate 1-2 page management report reflecting on the team's functioning according to the principles discussed in your teamwork book. This report should not be part of the Github/Gitlab README but rather a separate document. It should include a peer evaluation in which each team member is graded on their performance, stating three positive attributes and three areas for improvement as the basis for the grade. Note that each student will share the teamwork reflection with their team and discuss it in a session before turning in the report. The outcome of this feedback session must be included in the report for full credit.
- Quantitative Assessment
 - The project must include a quantitative assessment of its reliability and stability. You must use data science fundamentals to describe system performance, e.g., average latency per request at different levels of requests per second (100, 1000, etc.). Think of the software system as a data science problem that needs to be described using data science principles.
- Demo Video
 - A YouTube link in README.md showing a clear, concise walkthrough and demonstration of your application, including the load test and system performance assessment.
- Team Size and Makeup
 - The team should consist of 3-4 people, ideally composed of 1-2 strong programmers and 1-2 quantitative storytellers.