**Functional Requirements**

* **Assignment upload:**Students must be able to upload assignments (files/code) through an API. Uploaded data should be stored and linked to the student.
* **Auto Evaluation:**  
  An evaluation engine will run uploaded assignments against a predefined set of test cases. Scores will be generated automatically.
* **Score Management:**  
  Scores should be stored in MySQL in a way that links them to specific students and assignments.
* **Logs Management:**  
  Unstructured outputs, execution traces, and error logs should be stored in DynamoDB for analysis/debugging.
* **HashMap for Scores:**A Java-based in-memory HashMap will be used to hold scores temporarily before persisting them.
* **BDD Testing:**  
  The auto-evaluation should include Behavior-Driven Development (BDD)-style scenarios to validate end-to-end flows.
* **CI/CD Pipeline:**  
  Automated build and deployment with test execution on every update.
* **Demo Deployment:**  
  Final deployment to a demo environment for demonstration purposes.

**Non-Functional Requirements**

* **Scalability:** Must handle multiple student submissions simultaneously.
* **Reliability:** Evaluation results should be consistent and re-runnable.
* **Data Integrity**: Scores in MySQL must align with evaluation engine results.
* **Performance:** Evaluations and test executions should complete within acceptable time limits.
* **Maintainability:** Code and components should follow modular, object-oriented design.