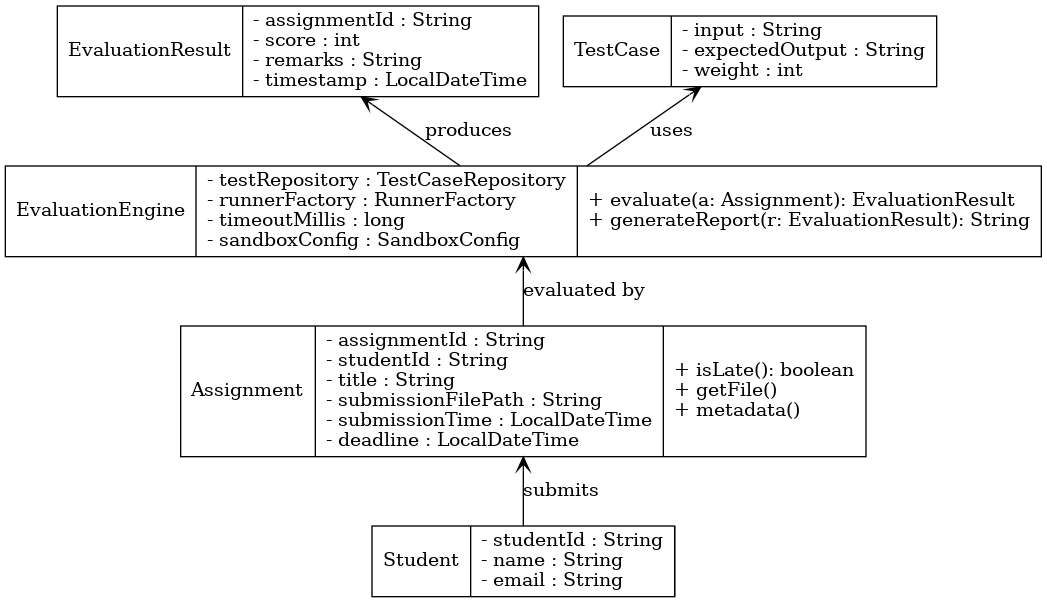
# UML Class Diagram

This diagram shows the relationship between Student, Assignment, EvaluationEngine, EvaluationResult, and TestCase classes.



**Classes in the Diagram:**

**1. Student**

* Represents a learner using the platform.
* **Attributes:** studentId, name, email.
* **Responsibilities:**
  + Submits assignments.
  + Views evaluation results (grades).

**2. Assignment**

* Encapsulates details of a student’s submission.
* **Attributes:** assignmentId, studentId, title, submissionFilePath, submissionTime, deadline.
* **Responsibilities:**
  + Tracks metadata of submission.
  + Provides utility methods like isLate() to check deadlines.
  + Supplies file references to the evaluation engine.

**3. EvaluationEngine**

* The core component responsible for **automatic evaluation**.
* **Attributes:** testRepository, runnerFactory, timeoutMillis, sandboxConfig.
* **Responsibilities:**
  + Runs test cases on an assignment.
  + Produces evaluation results with pass/fail status and feedback.
  + Generates reports for user-friendly display.

**4. EvaluationResult**

* Represents the outcome of evaluating an assignment.
* **Attributes:** assignmentId, score, remarks, timestamp.
* **Responsibilities:**
  + Stores score and feedback.
  + Used by students to view their grades.

**5. TestCase**

* Represents a single test input-output pair used during evaluation.
* **Attributes:** input, expectedOutput, weight.
* **Responsibilities:**
  + Provides criteria for evaluation.
  + Weighted test cases allow partial scoring.

**Relationships in the Diagram**

* **Student → Assignment**: A student submits assignments.
* **Assignment → EvaluationEngine**: Assignments are processed and evaluated by the engine.
* **EvaluationEngine → EvaluationResult**: The engine produces results after evaluation.
* **EvaluationEngine → TestCase**: The engine uses test cases to validate assignment outputs.