Day 4: Evaluation - Test Case Runner

**1. Objective**

Implementing a Test Case Runner that can run a set of test cases against a student's solution (abstracted as a CandidateSolution), collect test results, calculate a score, and produce a report. Provide unit tests for the runner.

## **2. UML (PlantUML)**

### **(Search for ClassDiagram2 in intellij)**

@startuml

package "org.example" {

class TestCase {

- id: String

- input: String

- expectedOutput: String

+ TestCase(id: String, input: String, expectedOutput: String)

+ getId(): String

+ getInput(): String

+ getExpectedOutput(): String

}

class TestResult {

- testCaseId: String

- passed: boolean

- actualOutput: String

+ TestResult(testCaseId: String, passed: boolean, actualOutput: String)

+ isPassed(): boolean

}

interface CandidateSolution {

+ run(input: String): String

}

class MockSolution {

+ run(input: String): String

}

class TestCaseRunner {

+ runTests(solution: CandidateSolution, tests: List<TestCase>): List<TestResult>

}

class EvaluationEngine {

+ runTestCases(assignment: Assignment): int

+ calculateScore(testResults: List<TestResult>): int

+ generateReport(student: Student, assignment: Assignment, results: List<TestResult>, score: int): String

}

Student "1" --> "\*" Assignment : submits

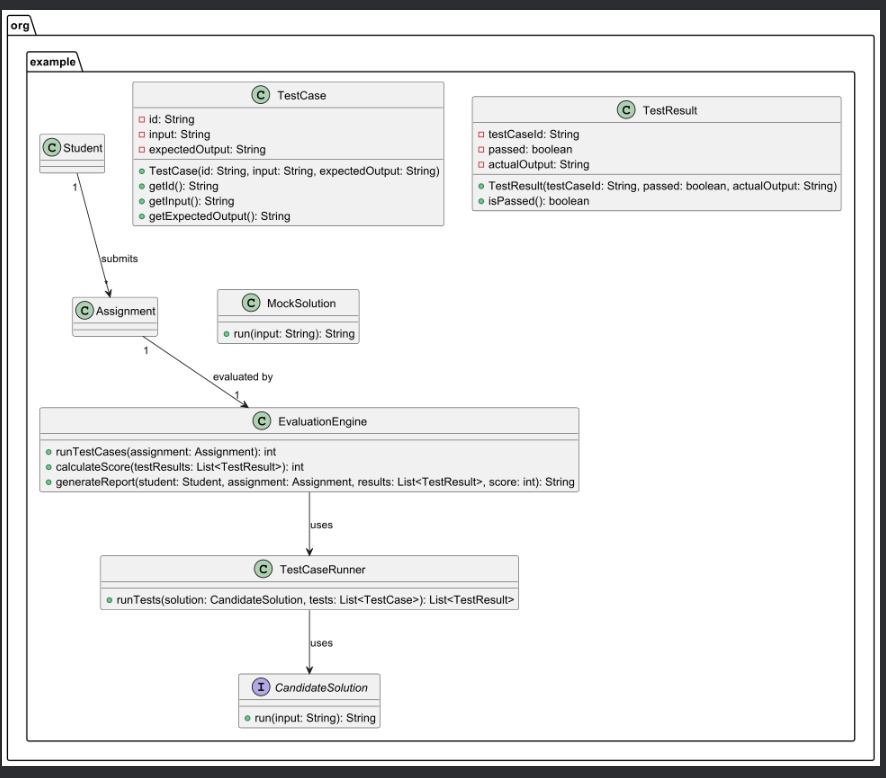
Assignment "1" --> "1" EvaluationEngine : evaluated by

TestCaseRunner --> CandidateSolution : uses

EvaluationEngine --> TestCaseRunner : uses

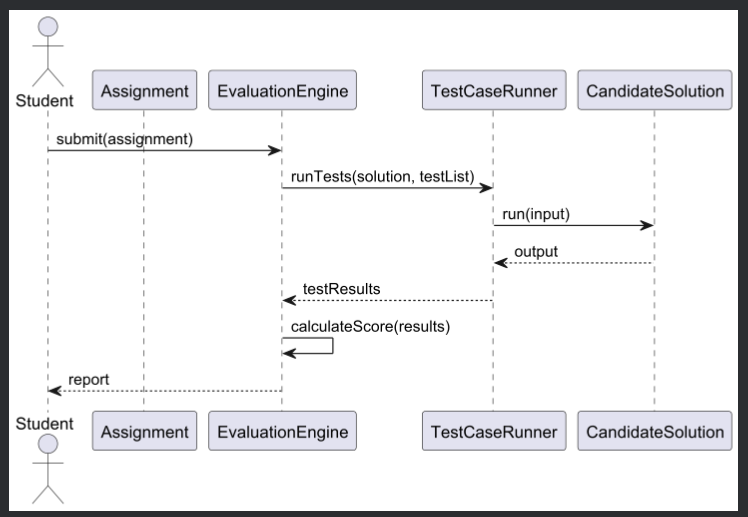
}

@enduml



**Sequence Diagram**

**(Search for SequenceDiagram3.puml)**

****

**3. Java code — files to create / replace**

* 3.1 TestCase.java

### 3.2 TestResult.java

### 3.3 CandidateSolution.java (interface)

### 3.4 MockSolution.java (demo implementation)

### 3.5 TestCaseRunner.java

### 3.6 Updated EvaluationEngine.java

### 3.7 Update AtlasApplication.java to demo Day 4

**4. Unit test for TestCaseRunner**

**Created -TestCaseRunnerTest.java**

**(src/test/java/org/example/)**

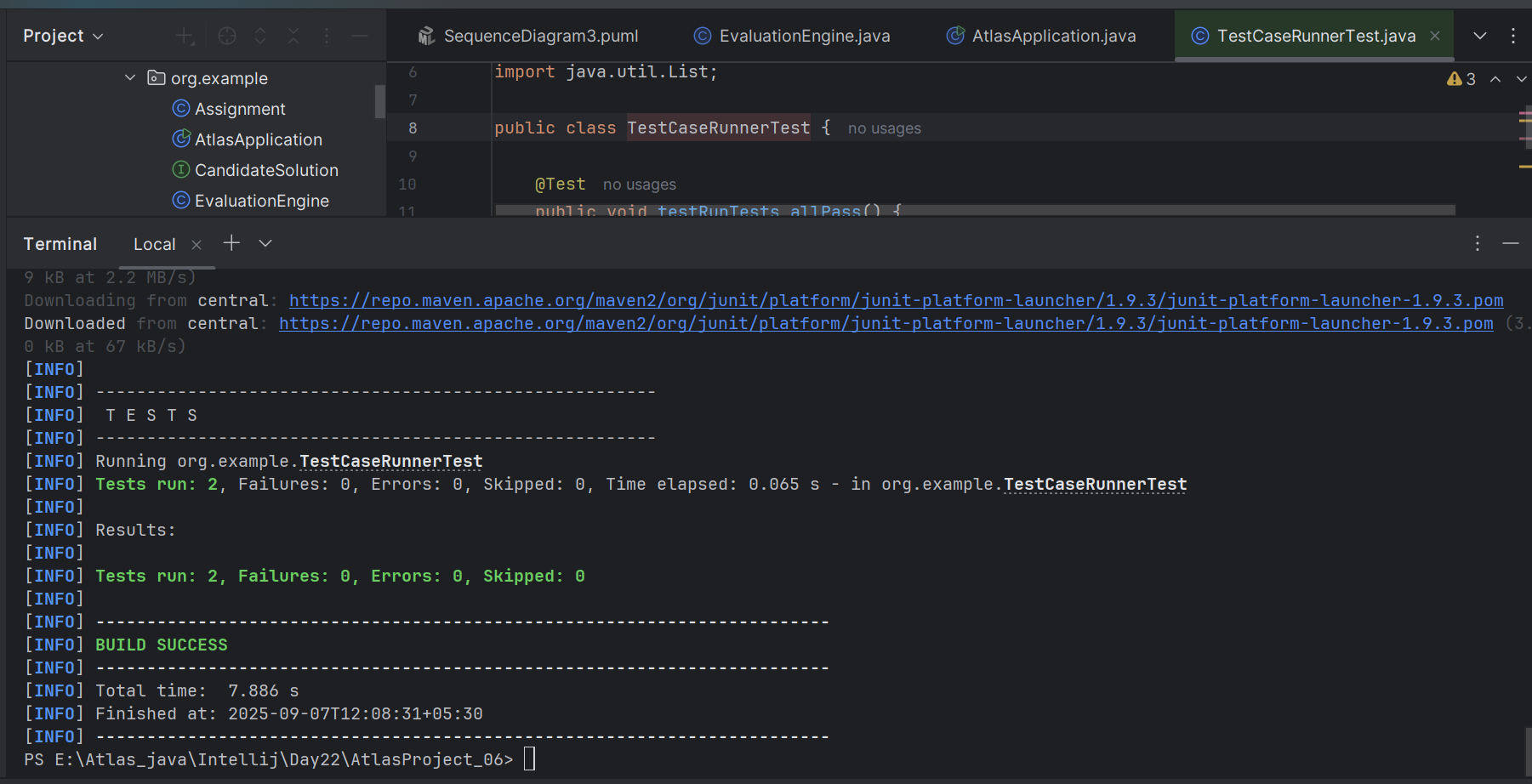
## **5. How to run in IntelliJ / Maven**

### **In IntelliJ:**

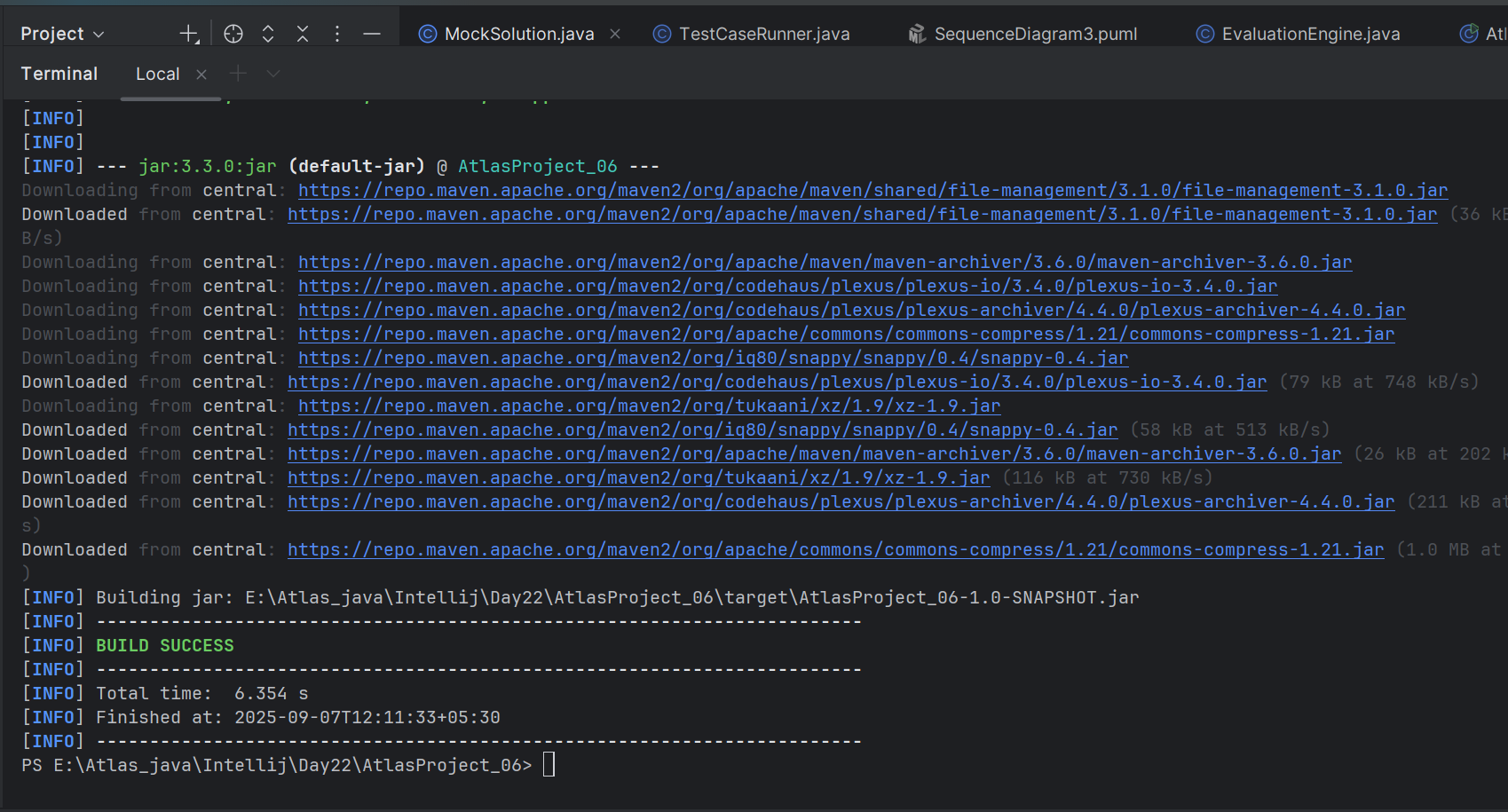
* Put the Java files under src/main/java/org/example/
* Put the test under src/test/java/org/example/
* Right-click AtlasApplication → Run to start Spring Boot (it will print the demo report).
* Right-click TestCaseRunnerTest → Run to execute unit tests, or run mvn test.

**Using Maven:**

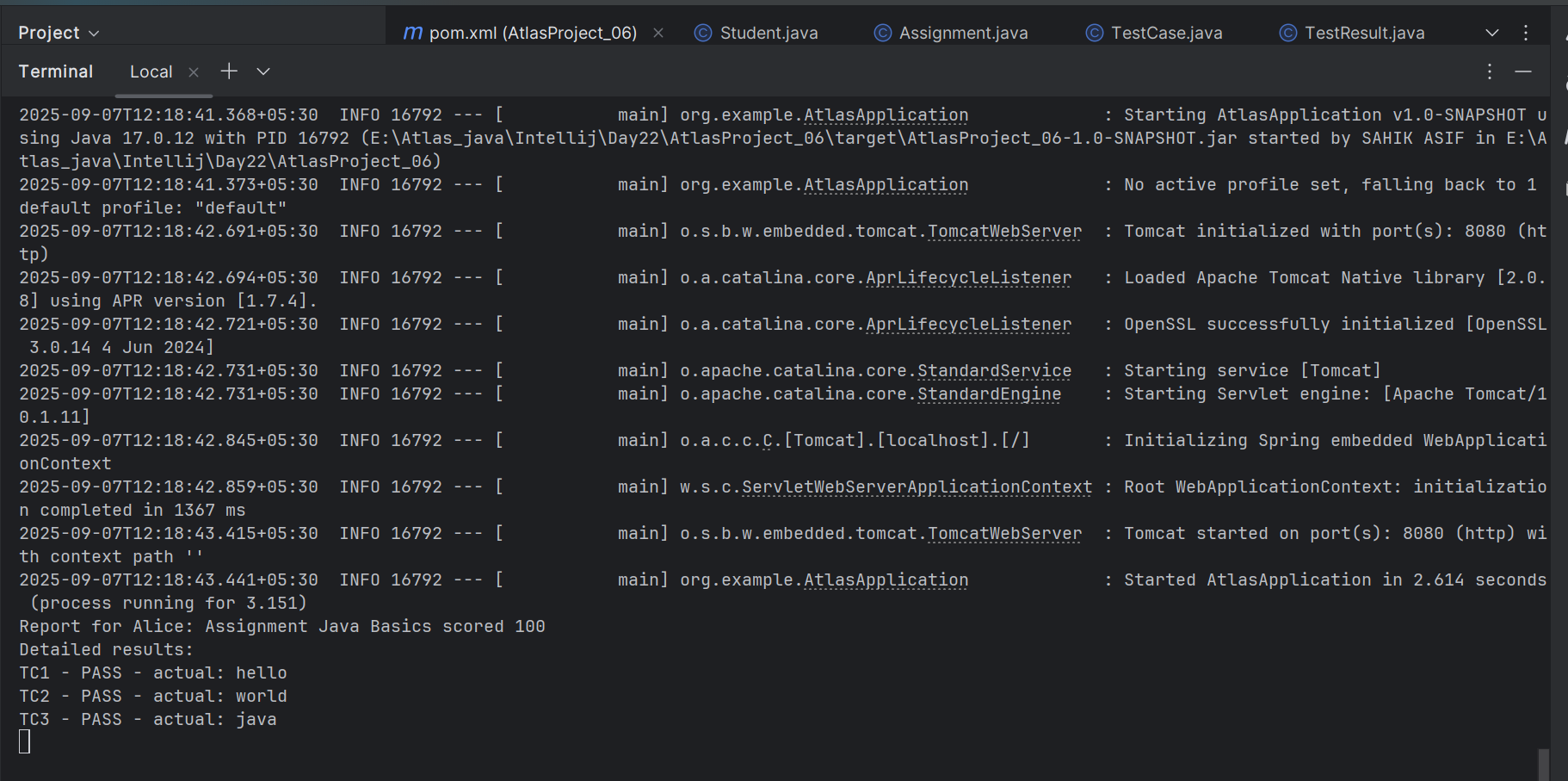
**mvn clean test:**

****

**mvn clean package:**

****

**java -jar target/AtlasProject\_06-1.0-SNAPSHOT.jar**

****

**Day 4 – Conclusion:**

* Implemented a Test Case Runner to execute test cases against an abstract CandidateSolution.
* Created TestCase, TestResult, TestCaseRunner, and a MockSolution to demonstrate the evaluation workflow.
* Updated EvaluationEngine to calculate score from test results and generate a detailed report.
* Wrote JUnit tests to validate the runner behavior (all pass / some fail scenarios).
* This prepares us to integrate real student code execution (with sandboxing) and persistent storage in the upcoming days.