Software Design Document

Student Code Online Review and Evaluation (2.0)

S.C.O.R.E. (2.0) aims to enhance and extend the original S.C.O.R.E. application. Adding functionality in server connections, grading with rubrics, and cheat detections are all intentions to deploy the system into CSE classrooms.

Team Members

Dorothy Ammons - dammons2022@my.fit.edu
Patrick Kelly - pkelly2022@my.fit.edu
Shamik Bera - sbera2022@my.fit.edu
Rak Alsharif - ralsharif2021@my.fit.edu

Faculty Advisor and Client

Raghuveer Mohan

November 23, 2025

Table of Contents

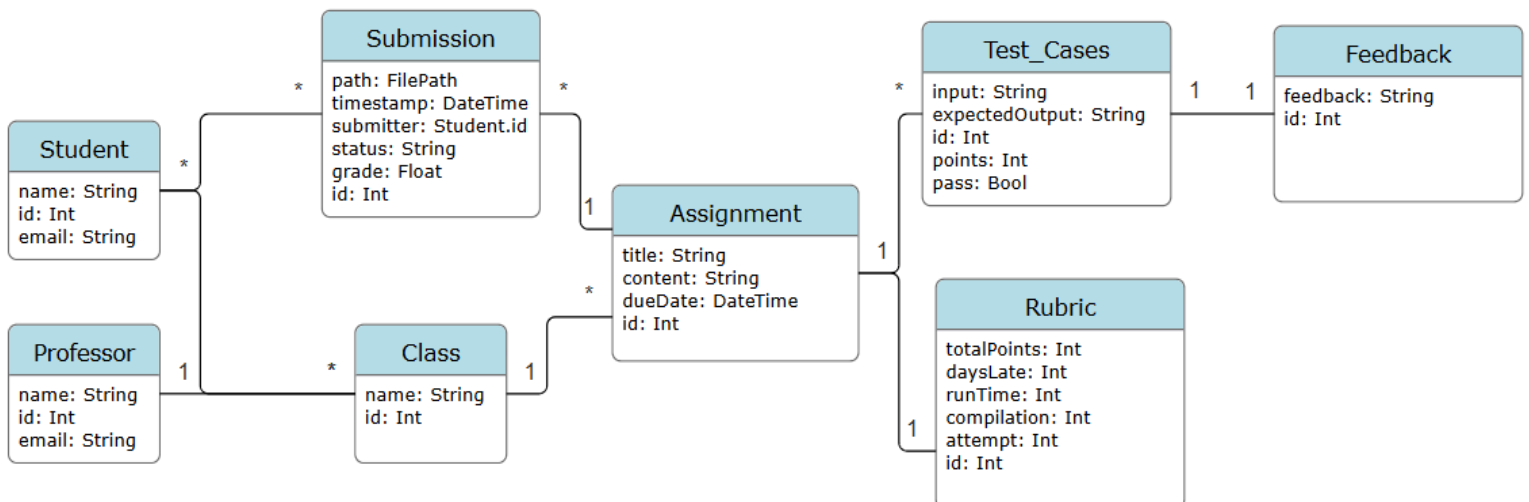# 1. Introduction

This document shows how the SCORE 2.0 additions will be designed. It contains the UML diagrams provided by the system architecture. The web application interface mockups will show the additions to the current pages, as well as entirely new pages, that showcase the introduced functionality.
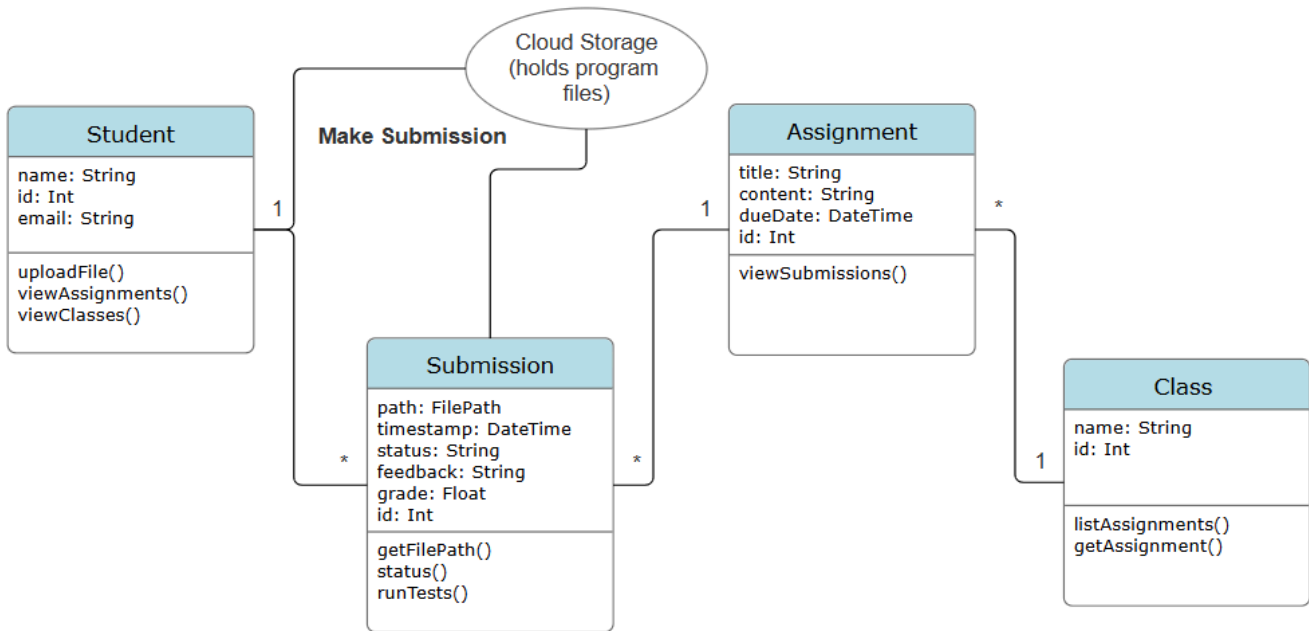
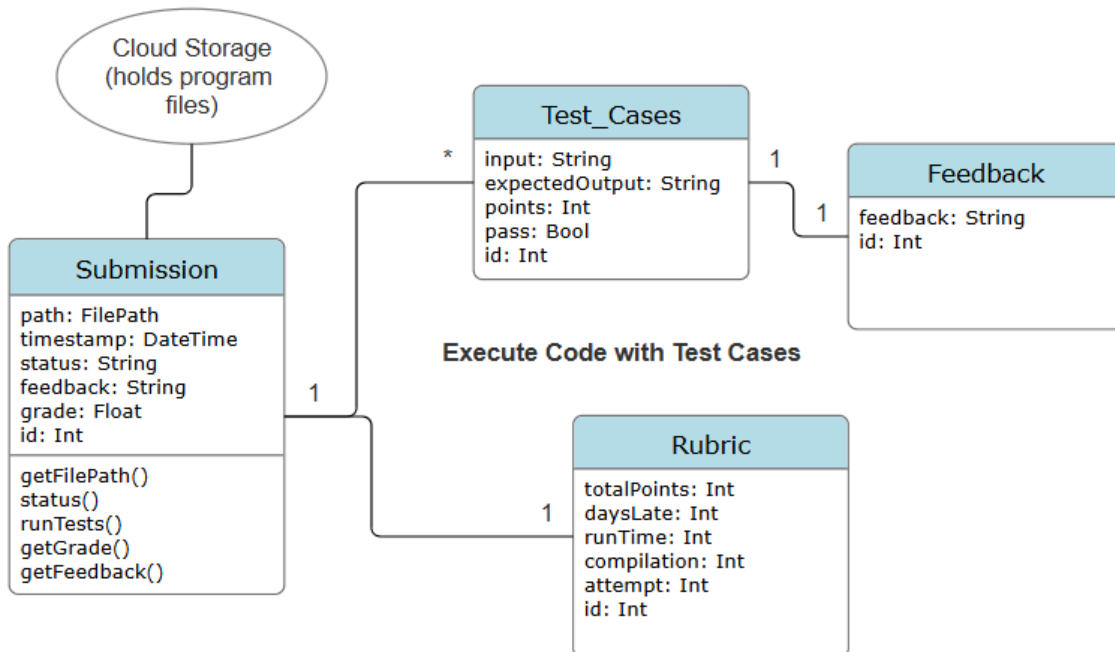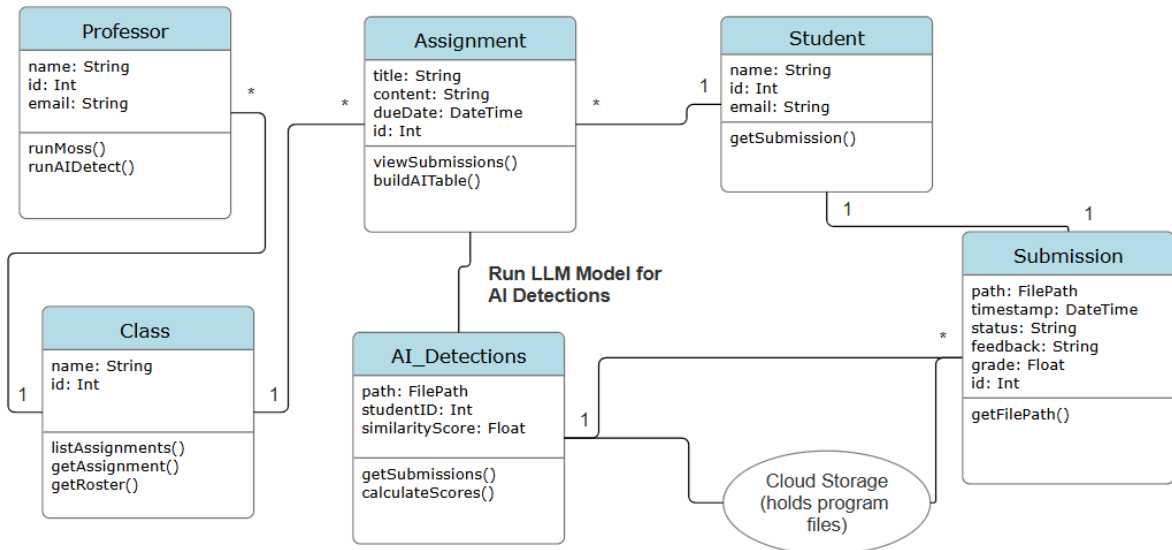# 2. System Architecture

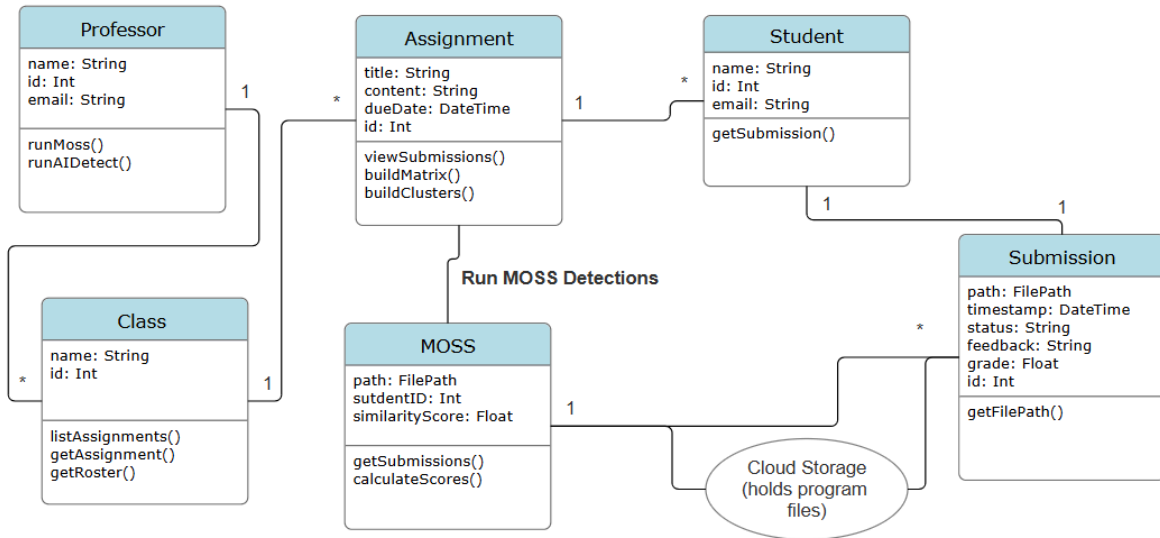## 2.a UML Diagrams
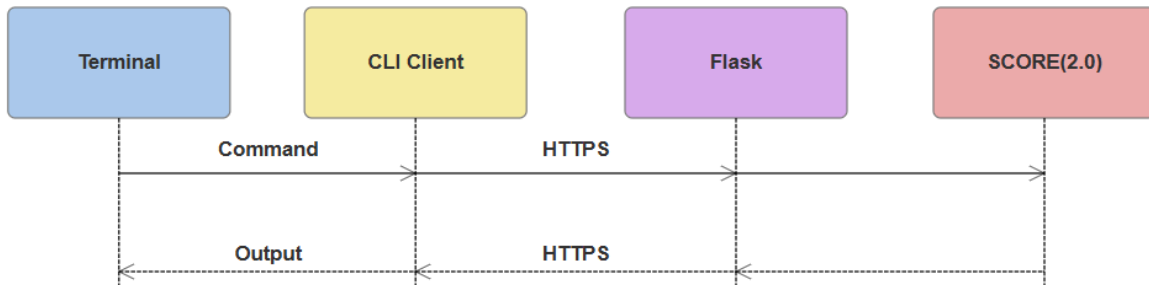
2.a.1 Generalized Database

2.a.2 Submissions

Student

name: String
id: Int
email: String

uploadFile()
viewAssignments()
viewClasses()

Make Submission

Cloud Storage
(holds program files)

1

Assignment

title: String
content: String
dueDate: DateTime
id: Int

viewSubmissions()

1

*

*

Submission

path: FilePath
timestamp: DateTime
status: String
feedback: String
grade: Float
id: Int

getFilePath()
status()
runTests()

*

Class

name: String
id: Int

listAssignments()
getAssignment()

1

2.a.3 Grading with Feedback

Cloud Storage
(holds program files)

Test_Cases

input: String
expectedOutput: String
points: Int
pass: Bool
id: Int

*

1

Feedback

feedback: String
id: Int

1

Execute Code with Test Cases

Submission

path: FilePath
timestamp: DateTime
status: String
feedback: String
grade: Float
id: Int

getFilePath()
status()
runTests()
getGrade()
getFeedback()

1

Rubric

totalPoints: Int
daysLate: Int
runTime: Int
compilation: Int
attempt: Int
id: Int

1

2.a.4 Cheat Detections

Professor

name: String
id: Int
email: String

runMoss()
runAIDetect()

1

*

Assignment

title: String
content: String
dueDate: DateTime
id: Int

viewSubmissions()
buildMatrix()
buildClusters()

1

*

Student

name: String
id: Int
email: String

getSubmission()

1

1

Class

name: String
id: Int

*

1

listAssignments()
getAssignment()
getRoster()

Run MOSS Detections

MOSS

path: FilePath
sutdentID: Int
similarityScore: Float

getSubmissions()
calculateScores()

1

*

Submission

path: FilePath
timestamp: DateTime
status: String
feedback: String
grade: Float
id: Int

getFilePath()

Cloud Storage
(holds program
files)

Professor

name: String
id: Int
email: String

runMoss()
runAIDetect()

*

*

Assignment

title: String
content: String
dueDate: DateTime
id: Int

viewSubmissions()
buildAITable()

*

1

Student

name: String
id: Int
email: String

getSubmission()

1

1

Class

name: String
id: Int

1

1

listAssignments()
getAssignment()
getRoster()

**Run LLM Model for
AI Detections**

AI_Detections

path: FilePath
studentID: Int
similarityScore: Float

getSubmissions()
calculateScores()

1

*

Submission

path: FilePath
timestamp: DateTime
status: String
feedback: String
grade: Float
id: Int

getFilePath()

Cloud Storage
(holds program
files)

2.a.5 Command Line Interface Connection

Terminal
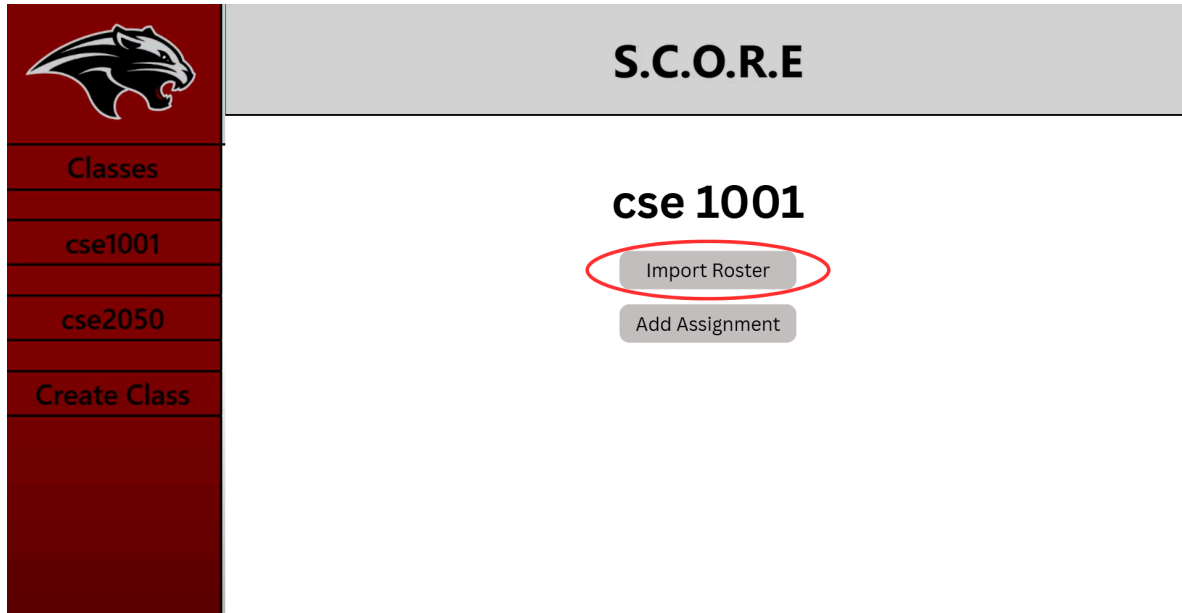
CLI Client

Flask

SCORE(2.0)

Command

HTTPS

Output

HTTPS

3. Web Application Interface Mockups

All mockups are edited screenshots of the previous projects web interface. Additions/changes are circled red.
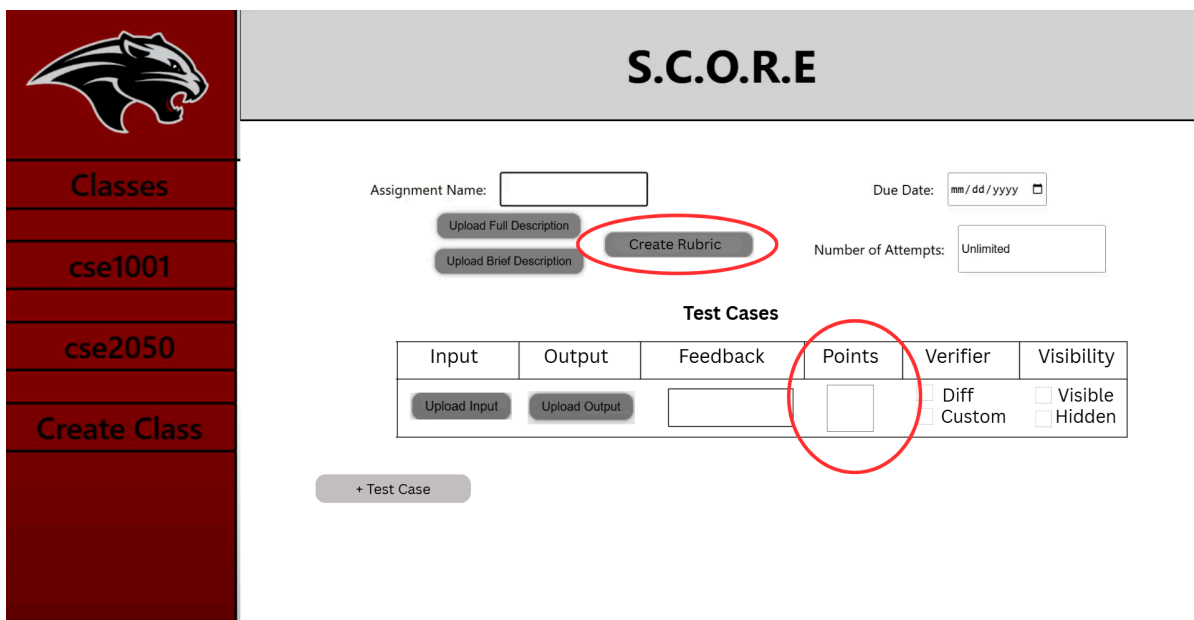
3.a Import Rosters Functionality

3.b Rubric Functionality

3.b.1 Add rubric to an assignment / give an assignment's test cases points

3.b.2 Rubric Page

S.C.O.R.E

Classes
cse1001
cse2050
Create Class

Rubric for Assignment <Title>

Factors	Points
Total:	
Compilation:	
Attempt:	
Under __ seconds:	
After __ days, remove:	

Submit Rubric

3.c Detect AI, Detect Collusion, Export Grades Features

S.C.O.R.E

Classes
cse1001
cse2050
Create Class

Demo Submissions

Student Name	Student Email	Numeric Grade
Firstname Lastname	exampleemail@gmail.com	2/2

Run AI Checker   Run Similarity Scores   Export Grades

3.c.1 Click "Run AI Checker"

3.c.2 Click "Run Similarity Scores"