Student Code Online Review and Evaluation 2.0

TEAM: SHAMIK BERA, DOROTHY AMMONS, PATRICK KELLY, RAK ALSHARIF
ADVISOR/CLIENT: RAGHUVEER MOHAN

Table of Contents

Milestone 5

- Complete AI detection model
- Release SCORE 2.0 into classrooms
- Complete export grade functionality
- Address reliability issues (multiple users at once, large data, security breaches)
- Conduct evaluation and analyze results
- Create poster for Senior Design Showcase
- Complete COPS integration

Milestone 5 – Completion Matrix

Task	Dorothy	Patrick	Shamik	Rak	To Do
1. Complete Google Cloud Run Hosting	100%	0%	0%	0%	This was completed however, once some updates were pushed to the project, the MiB of ram allowed was exceeded, we need to discuss other options
2. Work with our advisor to demo a release into classrooms	0%	0%	0%	0%	Resolve hosting issues
3. Test and correct bugs and security concerns	80%	0%	0%	0%	Correct delete class, show points per test case, allow PDF file uploads
4. Complete C.O.P.S	0%	100%	0%	0%	
5. Complete AI detection	0%	0%	0%	90%	
6. Add export grades functionality/finish import	90%	0%	10%	0%	
7. Create Senior Design poster	20%	0%	80%	0%	

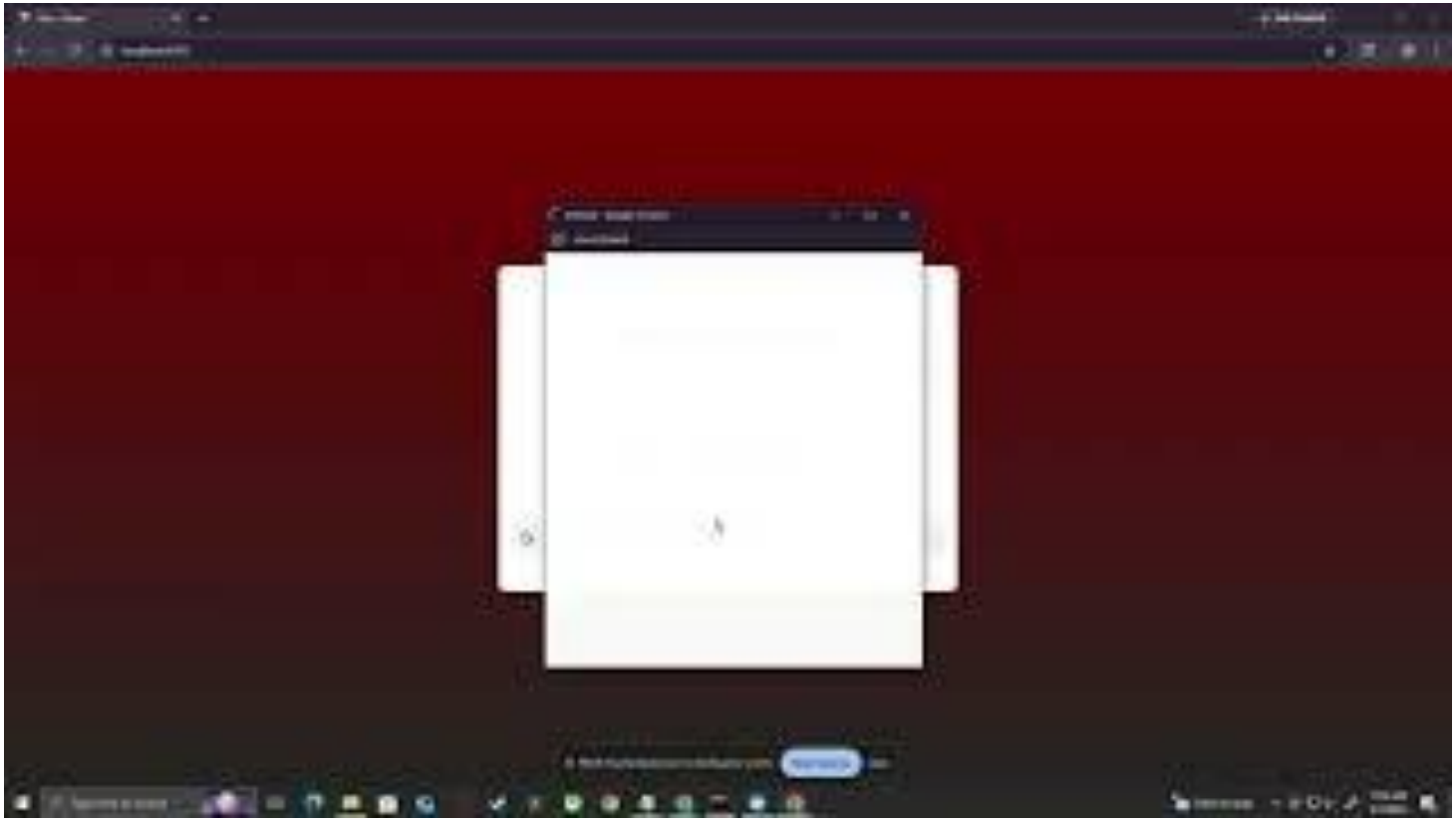
Hosting + Test run

- Containerized the project
- Pushed to the Google Cloud Run client, RAM limit exceeded
- Alternative: Create a container users can run locally
- Cloud based database allows for users to use synchronously
- Sent the container with commands and a walkthrough to Dr. Mohan to run a demo
- Sent a survey to gather responses from students

Bug Fixes

- FIXED late penalty from rubric (it deducts the rubric points PER DAY)
- ADDED download grades button (exports Canvas style CSV)
- FIXED grading portal page
- FIXED brief description + moved full description to be a downloadable file
- ADDED rubric to student view
- FIXED add students import
- ADDED roster view

Demo



C.O.P.S

- Developed backend system to compare student submissions
- Generated similarity matrix between all student files
- Used algorithm to calculate similarity scores (0–100%)
- Structured data for frontend visualization
- Helps identify potential plagiarism and collusion
- Designed for future integration into real classroom use

AI Detection

- Completed the remaining AI detection functionality
- Restructured the backend detection pipeline
- Integrated a pretrained GraphCodeBERT-based model with a GPT-based detector
- Standardized outputs into a consistent JSON structure
- Added logging, fallback handling, and safer error responses
- Tested on multiple human-written and AI-generated code samples

Milestone 6 – Task Matrix

Task	Dorothy	Patrick	Shamik	Rak
Complete any remaining bug fixes	50%	25%	0%	25%
Release project	100%	0%	0%	0%
Create Developer Manual	25%	25%	25%	25%
Create User Manual	25%	25%	25%	25%
Create video tutorials	25%	25%	25%	25%

Questions?