# SCORE Milestone 6 Project Evaluation

**Team Members:**
Charlie Collins, ccollins2021@my.fit.edu
Michael Komar, mkomar2021@my.fit.edu
Logan Klaproth, lklaproth2021@my.fit.edu
Tommy Gingerelli, tgingerelli2021@my.fit.edu

**Faculty advisor/client:**

- Dr. Mohan - rmohan@fit.edu

**Milestone 6 Progress**

| Task | Completion | Charlie | Logan | Michael | Tommy | To Do |
|---|---|---|---|---|---|---|
| Finish remaining tasks | 95% | 40% | 40% | 10% | 0% | Handle multi-file submissions |
| System deployment | 85% | 30% | 0% | 70% | 0% | Deploy to permanent location |
| Conduct test run evaluation | 90% | 25% | 25% | 25% | 25% | Full class test |
| Create user manual | 100% | 10% | 30% | 0% | 60% | N/A |
| Create demo video | 100% | 25% | 25% | 25% | 25% | N/A |

**Discussion of accomplished tasks:**

Task 1 (Finish Remaining Tasks): At the conclusion of the previous milestone, we were advised by Dr. Mohan to drop the Canvas and Moss integrations in order to clean up any loose ends, so for this milestone we reserved a task dedicated to doing just that. Some of the things that were covered in this task included adding authentication to the web app, creating a first time login page, cleaning up testing and feedback, and handling file submissions from the web app. In a previous milestone, we set up authentication for our Rust client, so in this milestone we added the api keys to the web app in order to also be able to sign in from that interface as well. This led to the need for a first time login page to allow the user to create their account the first time they login. For auto testing and feedback, we switched over to storing all test cases and results and json files, so that they could be more easily viewed on the web app. Then we added any additional functionalities we needed in the front end of the web app.

Task 2 (System Deployment): Once we wrapped up task 1, development on the SCORE application had concluded, so we looked to deploy the application so that it could be demoed. The first part of this was to "bundle" the system so that it could be easily deployed onto a machine. Then we were able to temporarily deploy the application on Michael's personal server, but we would like to find a more permanent place for it before we leave. The hope would be that we could get it onto the CS server, so that it can function much like the previous submit server.

Task 3 (Conduct Test Run Evaluation): After getting our application deployed, we were now able to demo the application in a real world use case. Since the whole point of SCORE was to be used in a classroom setting, we worked with Dr. Mohan to set up a demo with ten students from his class. This trial run started with Dr. Mohan creating a class, adding some students to the class, then creating an assignment. We then gave access to some of Dr. Mohan's students where they created their account, viewed the assignment, submitted code for the assignment, then viewed feedback. As a part of this, we created a google form so that we could get feedback from the users, with a focus on student feedback. This survey asked each trial user to rate the features from 1-10 to give some quantitative data on what the strongest and weakest features are. We also provided a place to leave comments for features, so that we could receive helpful feedback.

Task 4 (Create User Manual): Since Dr. Mohan expressed interest in using this application in future classes, we felt it was vital to create a user manual detailing the use of each feature so there would be as little confusion as possible. To do this, we actually created three manuals. The first was for students, and detailed all features from account creation, to assignment submission, and submission feedback. The second manual was for professors. This included the overlapping features from students, like creating an account, but also the professor only features like creating assignments and viewing the grading portal. The last manual we made was for developers. This was important to us as Dr. Mohan mentioned he might try to have students expand on the application in the future. In this developer manual we detailed exactly how the

application is run, as well as a breakdown of the structures and design choices, to make future development easier.

Task 5 (Create Demo Video): Now that we concluded the development of our application, our last task for the project was to create a demo video that shows all of the features. In order to do this, we made two separate videos, one from a student account, and one from a professor account. This allowed us to show the full use case from a professor's view, from creating a course all the way through to viewing the grades, and from the student's view, from creating their account to viewing feedback.

**Discussion of member contribution:**

Charlie: I started this milestone working on cleaning up the remaining tasks. This included creating the first time login page, working on file submission, and displaying feedback in the frontend. I also added a type to the user context for our web app so that we could distinguish between professors and students. I then had to make some changes through the app so that a user could only access features that they were allowed to access. I then worked with Michael to get the application deployed onto his personal server. Then I worked with the team to create the user manuals, and demo videos.

Michael: My first big contribution to this milestone was with reworking the auto feedback object to work with json files. This means reading both the expected and actual output from a json file, as well as writing the results of the test to a json. Then I played a big role in getting the application deployed. This included setting up the domain, creating the environment it would be deployed onto, and making a script that loads the entire application. Furthermore, I established a reverse proxy system using Nginx to deploy the web app with HTTPS on a public domain. Then I finished the milestone by helping to create our demo videos. Charlie and I proceeded to conduct a miniature test of the deployed web app to verify the Google Oauth2 API was working as intended with the rest of the system.

Tommy: When we started this milestone, the group decided to transition to storing test cases as json files rather than text files for ease of displaying, and on this front, I reworked the assignment creation command in the backend so that it would create our test case json file. After this, I did some work planning how we could deploy our application in a more permanent setting. To do this, I met with Dr. Silaghi twice to talk about the CS server, and was able to obtain permission to deploy our application by the end of the semester. I then spearheaded the user manuals by creating the basic structure and layout of each.

Logan: My work this milestone primarily centered around cleaning up the front end. I spent a lot of time fixing css and html bugs so that the application looked as expected. I also made the create course page, and styled the edit course window. Besides front end development, I also did

some work to our auto testing object so that it could read test cases from a json file and then create and fill the outputs.json file with the results from the test. I rounded out the milestone by doing some work on user manuals and creating the demo videos.

**Lessons Learned**

Importance of planning: This was probably our biggest lesson learned. In the beginning of the first semester, we were given a whole milestone for planning and documentation. While we did use this time for planning, we feel we focused too much on detail, and not enough planning on the system as a whole. For instance, we focused on how we might create a Rust CLI, and web app, but never planned how the two would interact. This lack of integration planning hurt us as we progressed into the project, as we learned that our components were not in a state where they could easily be integrated. This led to significant development time being spent reworking and refactoring existing components, so that they could be better integrated.

The need for a focused scope: When we started this project, we were very eager and bit off more than we could handle. We had intended to do two separate interfaces, Canvas integration, and MOSS integration. We ultimately had to end up cutting Canvas and MOSS, due to time constraints, but we still spent time planning and researching those features, which could have been spent elsewhere. Additionally, while we created two separate interfaces, if we had just chosen one interface, we feel it would have been a better polished and more usable product.

Co development of interfaces: The decision to independently develop two separate interfaces for our application, we feel, caused a lot of pain in development. We only worked on the CLI the first semester, and then only worked on the web app the second semester. However, in doing this, we made certain development decisions in the first semester with only the CLI in mind. When we got to making the web app in the second semester, some of these decisions were not ideal for use in a web application. For instance, storing the test cases as text files, while easier to use in a shell application, was far less ideal for a web app. If these interfaces were developed at the same time, then certain areas of development would have gone smoother.

Better communication: As a team, communication was definitely our weakest trait. We all had very busy schedules, and it was often difficult to plan meetings that everyone could attend. This led to smaller meetings that didn't include everyone, and the information discussed in those meetings was not always relayed to those that couldn't attend. This led to some pain points in development as sometimes not everyone was on the same page.

Time management: The second biggest issue we had as a team was proper time management. While we had the intention of using a development tool like Jira, to keep our development focused and on track, we ultimately failed to use a tool after the first milestone. This made time management tricky as it was hard to see the scope of what still needed to be done, and led to

being more relaxed at certain times during development.  As we worked in the previous two milestones, it became clear that we still had a lot of work to do, and if we had used a time management tool, we would have been able to see this sooner, and work harder earlier in development.

**Dates of meetings with the client/advisor:**
4/9/2025 at 4 pm
4/16/2025 at 1pm

**Client/Advisor feedback**
Task 1 (Finish Remaining Tasks):

- I like the finished application
    - The application is in a good state for another group to take over in the future

Task 2 (System Deployment):

- I'm glad that you were able to get the system deployed, at least temporarily
    - I would appreciate if you can get it in some permanent state before the semester ends
- Consider creating some script that can install and start the server with as little input as possible so that I could run it from a home machine if I want

Task 3 (Conduct Test Run Evaluation):

- Great test run

Task 4 (Create User Manual):

- I like the manuals
- Consider less words and more pictures
- Maybe add some of the diagrams you made to the developer manual

Task 5 (Create Demo Video):

- No comments

**Faculty Advisor Signature: _____**

**Date: _____**
**Evaluation by Faculty Advisor**

- **Faculty Advisor: detach and return this page to Dr. Chan (HC 209) or email the scores to pkc@cs.fit.edu**
- **Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)**

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Charlie Collins | 0 | 1 | 2 | 3 | 4 | 5 | 5.5 | 6 | 6.6 | 7 | 7.5 | 8 | 8.5 | 9 | 9.5 | 10 |
| Tommy Gingerelli | 0 | 1 | 2 | 3 | 4 | 5 | 5.5 | 6 | 6.6 | 7 | 7.5 | 8 | 8.5 | 9 | 9.5 | 10 |
| Michael Komar | 0 | 1 | 2 | 3 | 4 | 5 | 5.5 | 6 | 6.6 | 7 | 7.5 | 8 | 8.5 | 9 | 9.5 | 10 |
| Logan Klaproth | 0 | 1 | 2 | 3 | 4 | 5 | 5.5 | 6 | 6.6 | 7 | 7.5 | 8 | 8.5 | 9 | 9.5 | 10 |

**Faculty Advisor Signature: _____**
**Date: _____**