---

EE/CprE/SE 491 Weekly Report #4
10/4/2024 – 10/10/2024
Group sdmay25-23
Project title: Code Critiquer System for the C Language and Embedded C
Client &/Advisor: Dr. Diane Rover

Team Members/Role:
James Joseph - Secure System Design, CPR E 288 Liaison
Samuel Lickteig - Backend System Design
Alix Noble - Testing
Andrew Sand - Team Organization, CPR E 288 Liaison
Owen Sauser - Client Interaction, Frontend System Design, CPR E 288 Liaison

---

- **Weekly Summary**

  Primary, this week was all about continuing to get acquainted with the work of the prior team, sdmay24-34, and further refining the project definition. The members spent most of their time downloading and attempting to get the current codebase running locally to understand better what is complete, what works, and what needs improvement. Unfortunately, there were a few issues with getting the project running locally, which will hopefully be resolved shortly. Additionally, in the group's most recent advisor meeting, the team worked on better defining what to do with the current position the codebase is in and how it can be improved to get to a place where the project can best serve the CPR E 288 class in future semesters or potentially have further applications.

- **Past Week Accomplishments**
  - James Joseph:
    - Got old program up and running
    - Took notes on possible security issues
      - CSRF possible with static key
    - Determined that dynamic analysis will run on a separate program
    - Decided to stick with the flask app
    - Started discussion on front-end templating over static HTML
    - Supported team members in getting app running and understanding the infrastructure
  - Samuel Lickteig:
    - Worked on getting previous team’s application running
    - Read through and documented previous teams API

        - Examined sdmay24-34 app deployment and looked into gunicorn server they used
            - First time seeing project-level secure files being used; only file being secured is .env file to protect flask secret key
                - https://docs.gitlab.com/ee/ci/secure_files/
        - Examined SQLite and tried to understand how it worked
    - Alix Noble:
        - Worked on getting previous team's application running
        - Mostly successful, but ran into problems with uploading files
    - Andrew Sand:
        - Read through some of the frontend code on the sdmay24-34 GitLab
            - The team is investigating the possibility of using a frontend framework, so information on what is there currently is required
        - Looked through the sdmay24-34 project documentation
            - Specifically focused on their Final Report, as that had information about their project mentality
            - Looked for why the prior team made choices such as using Python or Flask
    - Owen Sauser:
        - Worked to get the previous team's code running locally
        - Browsed through the CprE 288 discord server to find any issues encountered by the students.

- **Pending Issues** (If applicable: Were there any unexpected complications? Please elaborate.)
    - James Joseph:
        - Waiting for to create a ticket with ETG based on further information on their role with Code Composer
    - Samuel Lickteig:
        - Confused by the apps database, might just be because I'm not used to python and SQLite
            - The sqlite file that seems to be holding the db data is in .gitignore so does each instance of the app currently have its own saved data?
    - Alix Noble:
        - Ran into issues with uploading files in the previous team's application. Unsure if it is due to a lack of knowledge on my part or an issue with the app.

    - Andrew Sand:
        - Just needing to confirm a few things about the Iowa State ETG and their role with Code Composer Studio
    - Owen Sauser:
        - Finish getting the previous teams code running locally with file upload
        - Re-evaluate if monitoring the 288 discord is useful
            - Most of the issues in the server are related to the data-sheet and Cybot rather than the C programming language

- **Individual Contributions**

| NAME | Individual Contributions | Hours this Week | Hours Cumulative |
|---|---|---|---|
| James Joseph | Got the previous team's code running. Helped members understand how to run and work with the code (need to do more next week). Took notes on next steps for infrastructure examination | 5 | 15 |
| Samuel Lickteig | Worked to get previous code running. Documented previous teams API. Examined previous teams code deployment. Looked through gunicorn documentation (http server). Looked through sqlite documentation and how it interacted with the previous teams application. | 6 | 14 |
| Alix Noble | Worked on getting previous team's code up and running | 2 | 9 |
| Andrew Sand | Continued getting acquainted with prior team's work | 2 | 12 |
| Owen Sauser | Worked with the team to try to get previous code running. Looked over the 288 discord to find any instances of bad code to use. | 3 | 8 |

- **Comments and Extended Discussion** (Optional)
  Feel free to discuss non-technical issues related to your project.

- **<u>Plans for the Upcoming Week</u>**
    - James Joseph:
        - Create a new endpoint for dynamic analysis
        - Need to move previous code to new repo
        - Need to update the readme to help with questions and problems we encountered
        - Create diagrams for current infrastructure
    - Samuel Lickteig:
        - Resolve local errors in previous team's code
        - Evaluate previous team's backend to decide if/what changes should be made to it
    - Alix Noble:
        - Continue work getting prior team's app running locally
        - Once errors are resolved, take stock of what is good and bad with the current frontend design
        - Look into frontend frameworks
    - Andrew Sand:
        - Reach out to the Iowa State ETG to see what their strategy for the upcoming Code Composer Studio overhauls
            - Need to confirm with Dr. Rover about contact information first and the ETG's role in CPR E 288
        - Try to get the prior team's work running locally
    - Owen Sauser:
        - Continue to monitor the 288 discord server
        - Try to get previous team's code up and running
            - If having issues, talk to previous members of the group.

- **<u>Summary of Weekly Advisor Meeting</u>** (If applicable/optional)

- Began by discussing project definition
    - Need to hone in on what exactly we want to be doing with the project
    - Project Routes
        - Need more anti-patterns
        - How do you enter an anti-pattern into the system?
        - Current front end is not very user friendly or missing desired features
        - Look into runtime antipatterns
        - Consider using the emulated environment that CPR E 288 used during the pandemic

- Should we switch backend frameworks?
  - Currently Flask
  - Should probably create a pro/con list to help determine
- Should we use a front end framework?
- Should we create another application that runs separately for analysis?
- Ask prior team why they went with Python
- Should we create compiler translations to more understandable English explanations?
- Focus on making the tool more user friendly on both the student and instructor sides
- Join the CPR E 288 Discord and begin conversations with TAs