S.C.O.R.E

Milestone 3

FLORIDA TECH

FLORIDA'S STEM UNIVERSITY

Team

- Charlie Collins
- Tommy Gingerelli
- Logan Klaproth
- Michael Komar

Faculty Advisor/Client

- Dr. Mohan

Milestone 3

- Implement the client server application
- Implement file transfer
- Implement auto testing
- Implement feedback system

Milestone 2 - Completion Matrix

Task	Completion	Charlie	Logan	Michael	Tommy	To Do
Implement Client Server Interaction	90%	40%	0%	60%	0%	Implement a man page and more robust responses
Implement File Transfer	75%	60%	0%	40%	0%	Implement an sftp server to properly handle the file transfers
Implement Auto Testing	80%	0%	50%	0%	50%	Auto Test caller, output comparisons, locating files
Implement Feedback System	50%	0%	50%	0%	50%	Implement professor-provided feedback

Implement Client Server Interaction

Client-Server - Completed

- Previously the shell was a single application
  - Now is split into a client and server
  - Communicate through TCP

- Server is asynchronous and uses threads to handle multiple clients

Client-Server - Completed

Client

- Run locally on the users machine
    - Ideally on code01
- Any SCORE command will be sent to the server
    - If the arguments are not given, they are prompted for
- Any non SCORE command is executed locally on bash

Server

- Creates a thread for each client connection
- When a command is received, calls the associated module
- Sends a response back to the client when done

Client-Server - TODO

- More verbose responses
  - Currently just responds with a success or failure message
  - For failures, we want it to provide reasoning as to why
- Create a man page
  - Make SCORE easier to use

Implement File Transfer

FLORIDA TECH
FLORIDA'S **STEM** UNIVERSITY

File Transfer - Completed

- Upon calling “submit”, the client transfer all files
  - SFTP

- These files are transferred to a temporary directory on the server
  - The submit module on the server will then handle the files

File Transfer - TODO

- Currently the files are transferred directly to a directory on the server

- Need a way to manage the incoming files
  - Ensure that no files are overwritten by other files

Implement Auto Testing

Auto Testing - Completed

- Startup and removal of testing environment
  - Dockerfile is modified according to assignment
  - Docker image is built from a dockerfile
  - Container is spun up for test and torn down after
  - Image is removed after testing concludes

- Run file on testing environment
  - Input files copied onto docker image
  - Submission is run for every input file
  - File location is currently static

- Output log taken from docker

FLORIDA TECH
FLORIDA'S STEM UNIVERSITY

Auto Testing - TODO

- Update file paths on dockerfile
  - Take in arguments for different submissions
- Comparing output to test cases
- Scheduling the testing script
- Implement “Verification” Program

Feedback System

FLORIDA TECH
FLORIDA'S STEM UNIVERSITY

Feedback System - Completed

- Based on auto-testing results of an assignment, the feedback that is generated is a printout of the number of test cases passed vs failed.

Feedback System - TODO

- Add potential wait times for submission feedback.
- Add professor guided feedback based upon auto testing results.

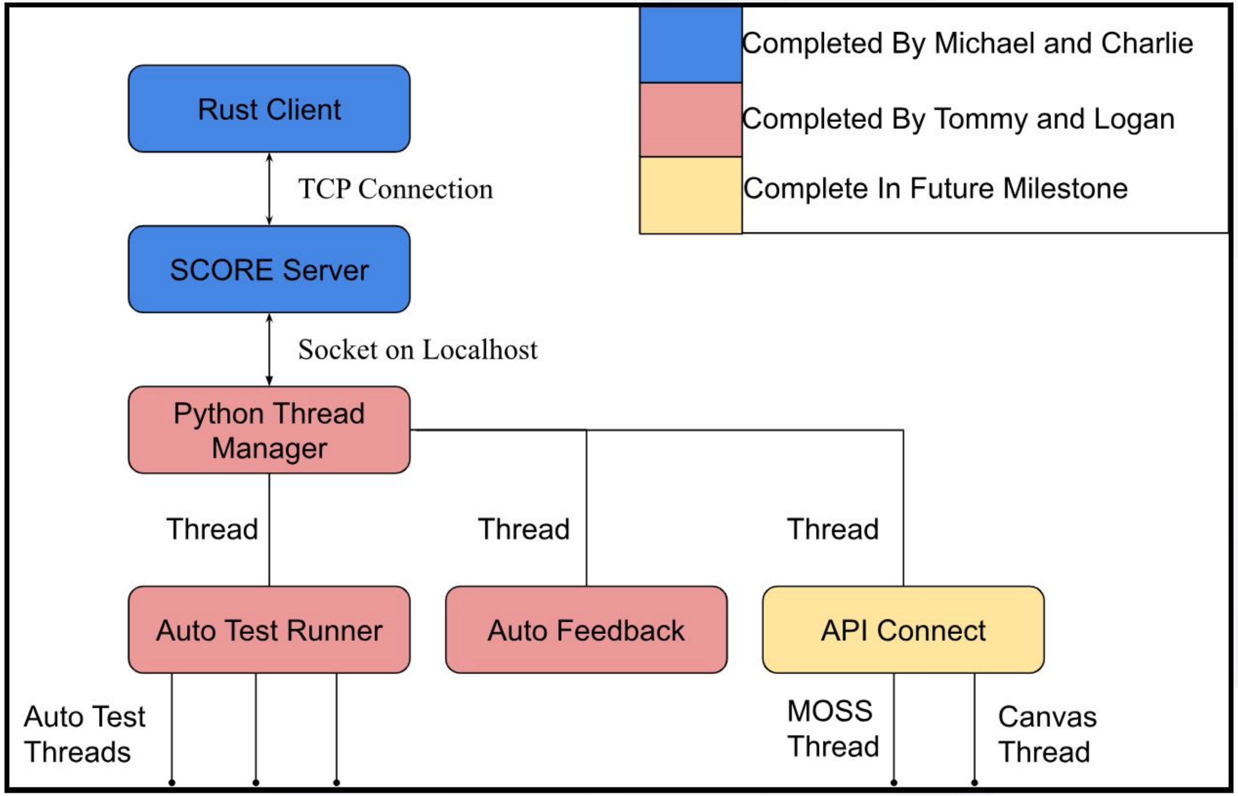
System Architecture

System Architecture - Completed

- Python “Thread Manager”
  - Manages 3 sub processes programs.
  - Wakes sub processes when not needed.
  - Handles all communications between front end and back end.
- Auto Test Runner
  - Receives submissions to test from Thread Manager.
  - Creates a thread of the autotest.py
    - Runs X tests against the submission.
    - Returns test results.
  - Returns test results to thread manager.
  - Kills completed child threads
- Auto Feedback
  - Receive submission with test results.
  - Currently returns test results.

FLORIDA TECH
FLORIDA'S STEM UNIVERSITY

System Architecture - Diagram



System Architecture - TODO

- Python “Thread Manager”
  - Management of API Connect thread
- API Connect
  - Create file
  - Allow for connection between external API and system.
- Auto Feedback
  - Improve user feedback with targeted professor feedback.
  - Report errors as different than just test case failures.
  - Report error types and implement common reasons for certain error types.
- Web UI
  - Connect with Socket.

FLORIDA TECH
FLORIDA'S STEM UNIVERSITY

Milestone 4

Task	Charlie	Logan	Michael	Tommy
Finish auto testing and feedback	0%	50%	0%	50%
Develop front end of web app	15%	35%	35%	15%
Implement user authentication	20%	20%	30%	40%
Integrate server components	50%	0%	50%	0%