Open-Source Report (TCP Connection)

Proof of knowing your stuff in CSE312

Guidelines

Provided below is a template you must use to write your reports for your project.

Here are some things to note when working on your report, specifically about the **General Information & Licensing** section for each technology.

- **Code Repository**: Please link the code and not the documentation. If you'd like to refer to the documentation in the **Magic** section, you're more than welcome to, but we need to see the code you're referring to as well.
- **License Type**: Three letter acronym is fine.
- **License Description**: No need for the entire license here, just what separates it from the rest.
- **License Restrictions**: What can you *not* do as a result of using this technology in your project? Some licenses prevent you from using the project for commercial use, for example.

Also, feel free to extend the cell of any section if you feel you need more room.

If there's anything we can clarify, please don't hesitate to reach out! You can reach us using the methods outlined on the course website or see us during our office hours.

Flask

General Information & Licensing

Code Repository	https://github.com/pallets/flask/
License Type	BSD 3-Clause
License Description	• Redistributions are permitted with or without modification • Including Commercial uses • Any redistribution must maintain the same license
License Restrictions	• Contributors are not liable for any damages caused by the software • Names of contributors shall not be used to promote derived software w/o permission

Magic

Dispel the magic of this technology. Replace this text with some that answers the following questions for the above tech:

- How does this technology do what it does? Please explain this in detail, starting from after the TCP socket is created
    - Flask creates an instance of a web application and establishes a TCP socket connection. Once the connection is established it listens to a port on the TCP socket for HTTP requests, receives the request, and parses the data so that it is available to the user in a convenient manner.

- Where is the specific code that does what you use the tech for? You ***must*** provide a link to the specific file in the repository for your tech with a line number or number range.
    - The app starts by creating an instance of the Flask app by using the python class Flask(name) (line 98, app.py) where 'name' is what gives flask an idea of what resources the app itself will use (such as templates, instance folders, and many more). Many functions are defined in this class such as 'add_url_rule' (line 1309, app.py) which allows users to create static routes. This is the function that is called when a user uses “@app.route()” in their code. Once the app is run, the TCP connection is established.

Flask (imports request und response from Werkzeug wrapper) -> Werkzeug Request wrapper which uses WGSIEnvironment to grab data from request and pass to parent’s init -> SansIO Request wrapper whose __init__function grabs that data and stores it in class variables