# Parsing HTTP Headers Report

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 / Python

### General Information & Licensing

| | |
|---|---|
| Code Repository | https://github.com/emilydesantis/cse312-because-the-internet |
| License Type | MIT |
| License Description | • Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.<br>• M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty. |
| License Restrictions | • Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.<br>• 2. Redistributions in binary form must reproduce the |

| | |
|---|---|
| | above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.<br>● 3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.<br>● THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. |

## 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
- 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.
    - If there is more than one step in the chain of calls *(hint: there will be)*, you must provide links for the entire chain of calls from your code, to the library code that actually accomplishes the task for you.
    - Example: If you use an object of type HttpRequest in your code which contains the headers of the request, you must show exactly how that object parsed the original headers from the TCP socket. This will often involve tracing through multiple libraries and you must show the entire trace through all these libraries with links to all the involved code.

*This section will likely grow beyond the page

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Since Flask automatically parses incoming HTTP headers the information is stored in the request object.

In Line 10 a new instance of the Flask class is created to the app variable.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L110

In line 13 the Flask application is configuring an SQL database with Flask-SQLAlchemy.
https://github.com/sqlalchemy/sqlalchemy

In line 45 the app.route decorator in Flask is used to link the root page from the (“/”) request to the function index() to load the index.html page.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1917

In line 50 app.route decorator in Flask is used to link the (“/page2”) request to the function page2() to render the page2.html page.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1917

In line 56 app.route decorator in Flask is used to link the (“/page3/<room_name>”) request to the function page3() to render the page3.html page.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1917

In line 63 app.route decorator in Flask is used to link the (“/page4”) request to the function page3(room_name) to render the page4.html page.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1917

In line 68 **app.route** decorator in Flask is used to link the (“/lobby/<room_name>”) request to the function lobby(room_name) to render the lobby.html page.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1917

In line 78 login once the request /login is received the app.route decorator in Flask is used to call the login() function checking to see if the user is valid, the redirect library is used to go to page2.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L2041
If not valid then a failure to login prompt arrives.

In line 96 sign up once the request /signup the app.route decorator in Flask is used to call the function signup() registering a user into the database, if the user isn’t already in the database the redirect library is used to go to page2.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L2041
If it is a duplicate a failure to sign up prompt appears.

In line 124 ErrorHandling occurs when a client requests a URL that doesn’t exist it will trigger the 404 error. The function associated with the 404 request is not_found_error(404)
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1553

In 301 line the run() method is called to run the application on the local development server.
https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L1067