CSE312 Group project Library report

Shijie Liu, Roshan Pradeep, Ryan Older, Eliza Koster

[Flask Library]

General Information & Licensing

Code Repository	https://github.com/pallets/flask
License Type	BSD
License Description	• The BSD-3-Clause license applies to all files in the Flask repository and source distribution. This includes Flask's source code, the examples, and tests, as well as the documentation.
License Restrictions	• Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. • 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. • 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.
Who worked with this?	Armin Ronacher

[Flask Application Object/Class]

Purpose

Replace this text with some that answers the following questions for the above tech:

- What does this tech do for you in your project?
- Where specifically is this tech used in your project? Give us some details like file location and line number, if applicable. If too cumbersome, a general description of where it's used for a given purpose is fine as well.

What it does for us:

- Flask allows us to render html templates for our website and handle http requests from our users.

Specifically:

- @app.route()
    - Adds a path to a specific html page that we would like to render and render it using flask.render_template.
- url_for()
    - Eliminates the need to hardcode paths, making code neater and more condensed. It accepts t

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 for you in the **Purpose** section of this report? Please explain this in detail, starting from after the TCP socket is created. Remember, to be allowed to use a technology in your project, you must be able to know how it works.

  Usually, we create a flask instance in our main module and the object acts as the central object. We pass the name of the module or package of the application. This name will be used to resolve resources from inside the package or the folder the module is contained in.

  The first parameter is used to give Flask an idea about what belongs in our application. This name is used to find resources on the file system and can be used by extensions to improve debugging information and more.

  This creates and configures the app at the name provided in the first parameter.

- flask/src/flask/app.py, starting at lines 98 to line 2091

[Flask render_template]

Purpose

Replace this text with some that answers the following questions for the above tech:

- What does this tech do for you in your project?
- Where specifically is this tech used in your project? Give us some details like file location and line number, if applicable. If too cumbersome, a general description of where it's used for a given purpose is fine as well.

What it does for us:

- This will allow us to manage templates so that it is easier to produce dynamic pages for our users. The template for Flask consists of variables and expressions that will get replaced during the rendering of the template. It also consists of tags that will handle the logic behind the rendering of the template. This will also make it easier to produce pages that inherit from other pages so that the organization of our code will be improved and more maintainable.

Specifically:

- Templates will use handlebar syntax inspired by django and python. For most of our html pages, we will need to use templates in order to handle the rendering of unique pages for our clients. It will be used in all of our static html pages except for our login page, as after a user logs in, the pages will all be dynamic. For example, we may have a line in excersiseSchedule.html that displays the username, and in order to display the username, we might write <h1> {{username}} </h1> where username is the name we would like to access. In our news feed, we would need to use the inheritance functioning of templates, in-order to dynamically generate the posts that other users have made that may be of relevance to the logged-in user.

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 for you in the **Purpose** section of this report? Please explain this in detail, starting from after the TCP socket is created. Remember, to be allowed to use a technology in your project, you must be able to know how it works.
  This imports a library Jinga2 which is used inside of the flask documentation inorder to create the templates. Jinga2 is a template library that is written in python. Because it is imported within the flask library inorder to facilitate flask templating, it's documentation will be provided below, however it will not receive its own separate library.

  Jinga handles the templates for flask. In it's SRC folder from github, there is a compiler file, which handles the compilation of templates in html using python code.

  This creates and configures the app at the name provided in the first parameter.

- src/flask/templating.py starting at lines 98 to line 2091
- https://flask.palletsprojects.com/en/2.0.x/tutorial/templates/
- https://github.com/pallets/jinja
- https://github.com/pallets/jinja/blob/main/src/jinja2/compiler.py

*This section may grow beyond the page for many features.

[Flask-Login Library]

General Information & Licensing

Code Repository	https://github.com/maxcountryman/flask-login
License Type	MIT License
License Description	● The MIT License allows for the following: ○ Commercial Use, Modification, Distribution, Sublicensing, and Private use..
License Restrictions	● Author cannot be held Liable since the work is provided “as is” ● You must include the copyright notice in all copies or substantial uses of the work. ● You must include the license notice in all copies or substantial uses of the work.
Who worked with this?	Matthew Frazier, Alan Hamlett

[flask_login.login_user()]

Purpose

- This method allows our app to log a user into our server to access the features of our application.
- This method will be used when our server receives a log-in request from the login page requesting to be logged in.

What it does for us:

- It helps us correctly log-in a user of our app instead of us handling the entirety of the log-in system for our server/application.

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 for you in the **Purpose** section of this report? Please explain this in detail, starting from after the TCP socket is created. Remember, to be allowed to use a technology in your project, you must be able to know how it works.

  After a TCP socket is created the user would log into our application to access the features of our app. Our server will handle this log-in request by utilizing the flask_login.login_user() method logging in the user into our application. This login method checks for the information attached to the user object the login_user method takes as a parameter to decide if it should log in the user or not. This method also allows for specific durations of a logged in user until their remember cookie expires which is helpful for us. The method will return true if the user has successfully logged in and false otherwise.

- flask-login/flask_login/utils.py/ from lines 144 to 191
  - Utilizes the _update_request_context_with_user in flask-login/flask_login/login_manager.py in order to store the user that logged as ctx.user.

[flask_login.login_required]

Purpose

Replace this text with some that answers the following questions for the above tech:

- What does this tech do for you in your project?
- Where specifically is this tech used in your project? Give us some details like file location and line number, if applicable. If too cumbersome, a general description of where it's used for a given purpose is fine as well.

What it does for us:

- This tech makes sure a user is logged in and authenticated before the user can view the page that is being requested.

Specifically:

- This tech is specifically used in our project to view pages that require logging in, for example, home page, direct messaging page, to log out of the account logged in.

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 for you in the **Purpose** section of this report? Please explain this in detail, starting from after the TCP socket is created. Remember, to be allowed to use a technology in your project, you must be able to know how it works.

  After creating the TCP Socket, when a user tries to access a page that requires them to be logged in, we use the login_required function whenever we wish a page to be accessed only after login/registration. If a user logs in successfully, they will be able to access all pages that require login_required. If we decorate a view like this, it will ensure that the current user is logged in and authenticated before calling the actual view.

  If the user is not logged in, it calls the LoginManager.unauthorized callback and doesn't display the page that the user is requesting.

- flask-login/flask-login/utils.py lines 233-278

[flask_login.logout_user()]

Purpose

- This method allows us to logout a user from our server.
- This method is used whenever a logged in user requests to logout from our website.

What it does for us:

- It helps us correctly log-out a user of our app instead of us handling the entirety of the log-out system for our server/application.

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 for you in the **Purpose** section of this report? Please explain this in detail, starting from after the TCP socket is created. Remember, to be allowed to use a technology in your project, you must be able to know how it works.

  Our server uses the logout_user function in flask_login to logout a user after they had requested it through our webpage. Logout_user first gets the user that is requesting to log out and removes them from the web socket session and also removes their remember me cookie.

- flask-login/flask_login/utils.py, from line 194 to line 220