

Jayen Lare  
Brian Poblete  
COP 4521- Fall 2022  
Dr. Piyush Kumar

### 1) What we built and how we built the project

- For this project, we built a web application using Python as our programming language and Flask as the framework. The application is being served using Gunicorn as our WSGI with Nginx for reverse proxy to our domain name. Our domain name was purchased through Namecheap. The application is running on a virtual machine with 15GB of storage and 1GB of RAM running Ubuntu 22.04 LTS. The server machine is hosted by DigitalOcean. In order for both of us to contribute to the source code of the program, we set up virtual machines on our personal computers that have identical environments to that of our hosting machine on DigitalOcean. We used GitLab to host our source code and made use of GitLab's Issues feature to keep track of our requirements and Pipeline for the CI/CD of our app. To make updates to the source code of the app, we connect to our hosting machine using SSH and pull the main branch of the GitLab repo onto the machine.

### 2) Links

- Gitlab repo: <https://gitlab.com/Brian-Pob/cop4521>
- Zoom meetings: <https://docs.google.com/spreadsheets/d/1LKSHWKToTxY8kFq1PCNFfmWyuN-6nIDIDaK5-yJctK0/edit?usp=sharing>
- Tutorial video: [Demo and Tutorial Video](#)

### 3) The architecture of the implementation

- We use Flask for our application backend and frontend. First, the application is served by Gunicorn running as a systemd service. Nginx is then used to serve the application to our machine's IP address.
- In the backend, the application makes requests to the Hackernews API to get the current top stories on Hackernews. That data is then cached and keywords are extracted from the title of each story. We then pass that data to the frontend through Flask and display it using HTML files with Jinja 2.0 code. We allow users to create accounts in order to like and dislike the posts that are displayed. Account authentication is handled by Auth0. We store created accounts in our SQLite3 database. We also store the likes and dislikes of each user in our SQLite3 database. Certain users are designated as admins and they have the ability to remove likes and dislikes from posts. Admin access is also stored in our database.

- Our frontend is rendered by Flask with static files being served by Nginx. Styling is done with CSS files.

### 3) A summary of tools utilized for this project

- Python, Flask, Gunicorn, Nginx - serving the website
- Auth0 - User authentication
- SQLite3, SQLAlchemy, FlaskDB - Database management
- Pipenv - managing virtual environments and Python packages
- GitLab - source code hosting and VCS
- Spacy - keyword extraction
- Flask-caching - data caching
- Mozilla Observatory, GitLab Pipelines, pytest - tests
- Fail2Ban, UFW, SSH, Ubuntu unattended-upgrades - security and updates
- DigitalOcean - hosting and snapshots
- Namecheap - domain name provider

### 4) Instructions for using this web application

- You can access our web application by going to *news.brainpob.me*. This will take you to the landing page of our web app. Here you will be welcomed and shown a brief overview of how to use our application. At the top of the webpage there will be a navbar with 4 tabs: FlaskApp, Home, About, and Login. The FlaskApp tab will always take you back to the landing page. The Home tab will take you to view the top news stories. Since you are not logged in at this point, you will be greeted as a guest and only be allowed to view the stories. The About tab will take you to the About Us page, where you can learn more about who we are and how we developed the web app. The Login Tab will allow you to login into an existing account or create a new account. Once logged in, you will have the ability to like and dislike posts on the Home page. Also, you will gain access to the Profile page. Here you will be able to see all the posts you have either liked or disliked. If you are logged in as an admin you will have access to the Admin page as well. This will allow you to see all posts liked or disliked by any user, as well as remove likes or dislikes from any posts. Finally, you can click the Logout tab to log out of your account.