

# Bookworm Rec

## Tech Review

Priyam Gupta, Sue Boyd,  
Lawrie Brunswick, Jacob Peterson



# Background and Use Case

**Application:** Book Recommender Tool

**Use Case:** We need to create a graphical user interface to:

- Collect user input (type of book they are searching for, filtering requirements) AND
- Share recommendations based on user input + our recommendation engine

**Package Requirements:**

- Enables dynamic, interactive UI
- Can be run as a Python module
- Easy to install and use for all team members

# Python Packages Investigated

## **Flask - Created by Armin Ronacher**

- A lightweight and flexible micro web framework for Python.
- Provides a simple and minimalistic approach to building web applications, allowing developers to have more control over the application structure.
- Highly customizable and allows developers to choose their preferred tools and libraries for various components like ORM, templating engines, and authentication.
- Well-suited for developers who prefer to have more control over the application architecture and want to build from the ground up.

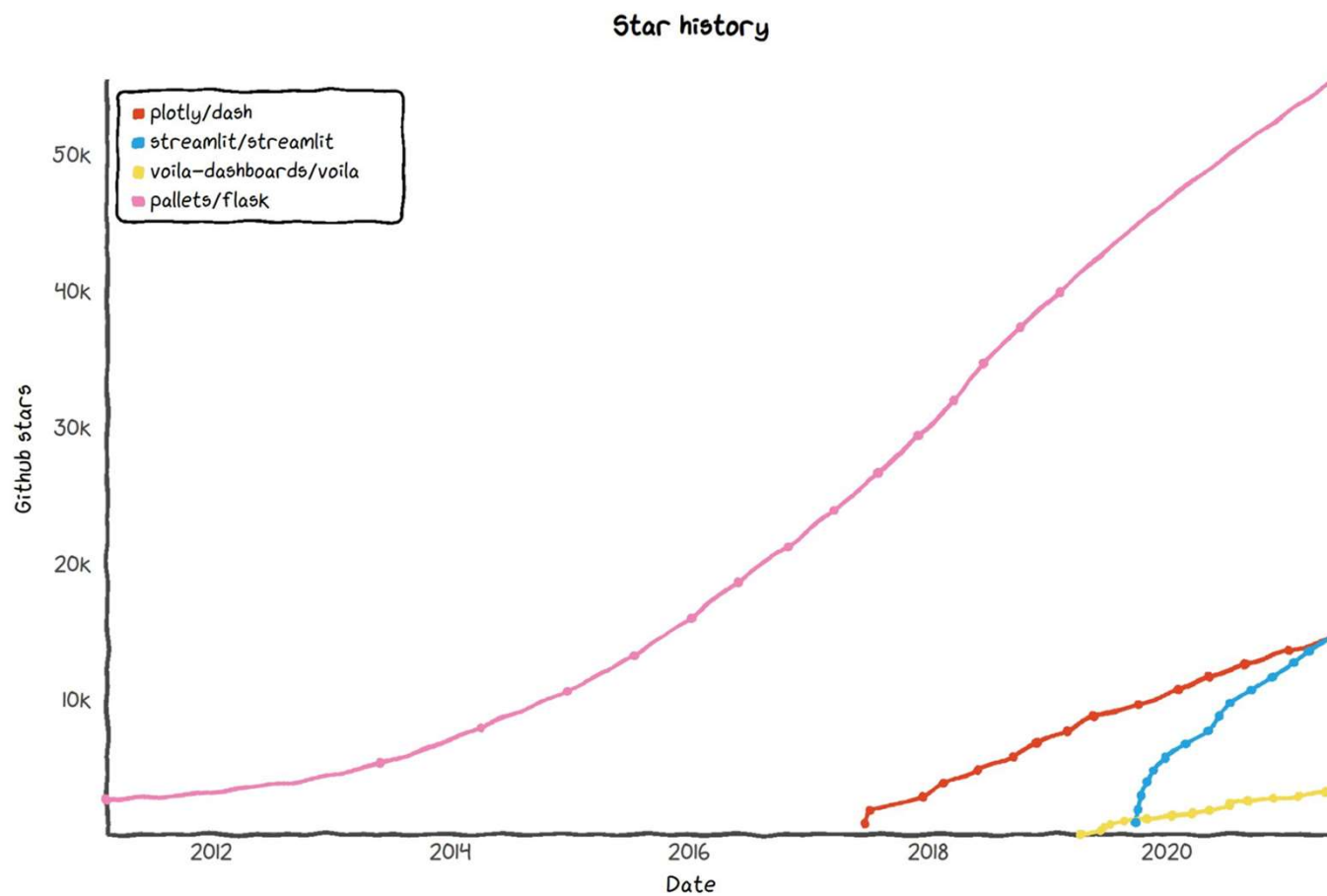
## **Dash - Created by Plotly**

- A Python framework for building analytical web applications.
- Built on top of Flask, Plotly, and React, allowing creation of interactive and complex user interfaces.
- Allows developers to create interactive dashboards and data visualization tools using Python, HTML, and CSS without requiring knowledge of JavaScript or web development.
- Provides a wide range of components and features for building sophisticated web applications, making it suitable for projects that demand high levels of customization and interactivity.

## **Streamlit - Created by Snowflake Inc.**

- A python framework for quickly developing and sharing data-driven applications without extensive programming knowledge.
- Best for simple apps (e.g single web page)
- Allows hosting of apps on Streamlit Community Cloud for easy deployment.

# Popularity Comparison: Github Stars



# Package Comparison (1 of 2)

|                    | Dash  | Flask   | Streamlit   |
|--------------------|---|---|---|
| <b>Ease of Use</b> | <b>Easy:</b> Simple to install and use, but has a slightly more complex setup.    | <b>Easy/Moderate:</b> Easy for beginners to build small scale projects. | <b>Easiest:</b> Simple to install and use. Standard UI elements (drop down, slider) precoded; no html required. |
| <b>Flexibility</b> | <b>Good.</b> Dash offers greater customization than Streamlit..                   | <b>Good.</b> Core web-dev features with flexibility for external tools  | <b>Lowest - but Good enough.</b> Less opportunity for customization, but provides what we need for our project. |
| <b>Performance</b> | <b>Good.</b> Can use Memoization and Flask Caching. Quick visualization plotly.js | <b>Good.</b> Fast and flexible, Flask-caching library                   | <b>OK.</b> Inefficient unless caching strategies deployed   |

## Package Comparison (2 of 2)

|                                    | Dash   | Flask  | Streamlit   |
|------------------------------------|--|--|---|
| <b>Documentation / Examples</b>    | <b>Good.</b> Has been around for 9 years. Plenty of support and tutorials. | <b>Good.</b> Has been around since 2010. Most popular micro framework for Python | <b>Good.</b> Newer, but extensive tutorials available |
| <b>Integration / Compatibility</b> | <b>Good:</b> Integrates well with Python, Docker                           | <b>Best:</b> Integrates well with Python, Docker, external APIs                  | <b>Good:</b> Integrates well with Python, Docker      |
| <b>Deployment</b>                  | <b>Ok:</b> Github Codespaces or Azure Credits                              | <b>OK:</b> Github Codespaces or Azure Credits                                    | <b>Best:</b> Streamlit Community Cloud                |

## **We Choose Streamlit**

- Has all of the capabilities that we anticipate needing
- Easiest to get started with and simple UI
- Customizable and moderately flexible for more complicated design
- One webpage, one request, no complexities
- Data Science focused
- Community Cloud hosting for web page

## **Drawbacks/Remaining Concerns**

- Not optimized for multiple unique url routings
  - Will mitigate by keeping design to one page
- Difficult to have intricate designs
- Potential performance issues
  - Minimize real time calculations