



STREAMLIT:

The fastest way to build Data apps

Steven Kolawole

Problem Statement

01

What is Streamlit?

02

**Install Streamlit &
Play around**

03

TABLE OF CONTENTS

04

Building a Data App

05

**Sneak Peek & Use
Cases**

06

**Maybe Streamlit is
not that awesome...**

ABOUT ME

Steven Kolawole

- A Machine Learning Engineer **looking for his next gig**
- Computer Science Undergrad @FUNAAB
- Somewhere in the intersection between Software Engineering & Data Science
- Liverpool FC fan since kid
- Python Freak
- Foodie





01

Problem Statement

Building a Data-driven Web App Process



Data processing and analysis in notebooks



Copy-paste in a .py script



Create an app via Flask, and at least, HTML, CSS, JavaScript and JSON



Outlining requirements and UI/UX designing



Create a state-of-the-art app



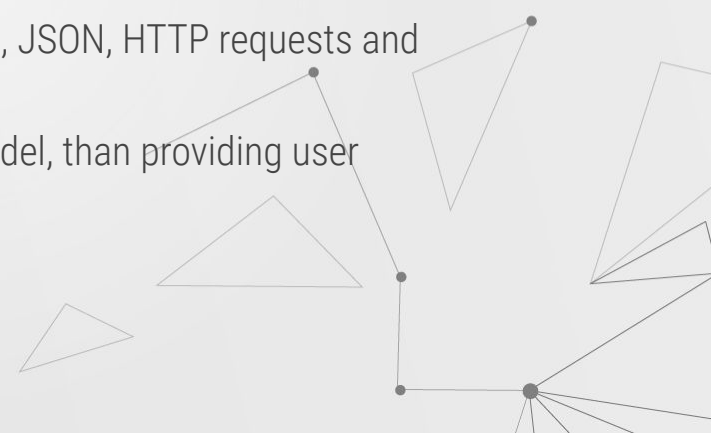
Updates once in 3 months or so




The Pain Points

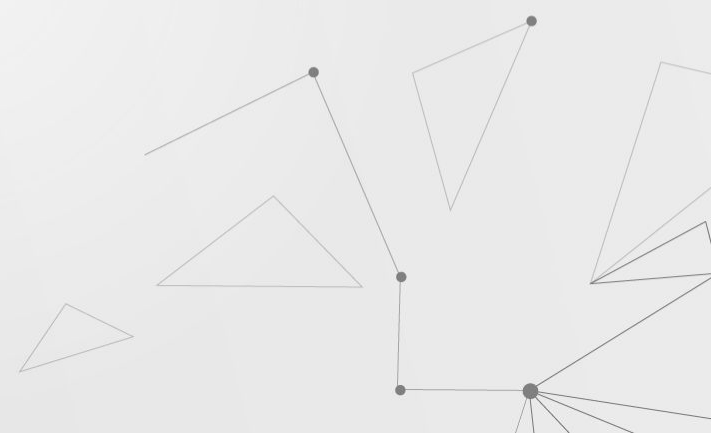
- Depending on the software engineering team takes up a bunch of our time
- We can't create new features and updates because the software dev team have lots on its plate

We'd create those apps ourselves but;

- We are not particularly artsy, and we suck at UI designs
 - Creating web apps consume a humongous amount of time
 - Aside from Flask, we have to bother with JS, HTML, CSS, JSON, HTTP requests and others
 - We prefer to use code to generate insights or build a model, than providing user with aesthetic experience.
- 



What if we could
build apps as easily
as writing Python
Scripts???





02

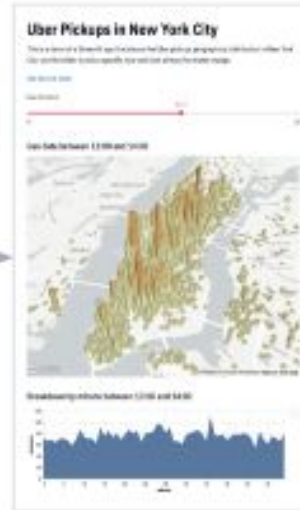
What is Streamlit?

- Building apps as easily as writing Python scripts

The Streamlit workflow



Step 1:
Sprinkle in a few API
calls into your existing
Python script.



Step 2:
Show off your
beautiful, performant
tool 🍷.

- It covers most UIs used in data apps

```
def add_boxes(image, boxes):
    image = image.astype(np.float64)
    for _, (xmin, ymin, xmax, ymax, label) in boxes.iterrows():
        image[ymin:ymax, xmin:xmax, :] += LABEL_COLORS[label]
        image[ymin:ymax, xmin:xmax, :] /= 2
    return image.astype(np.uint8)

def main():
    st.write("# Let's Download Some Data")

    metadata = load_metadata(LABELS_FILENAME)

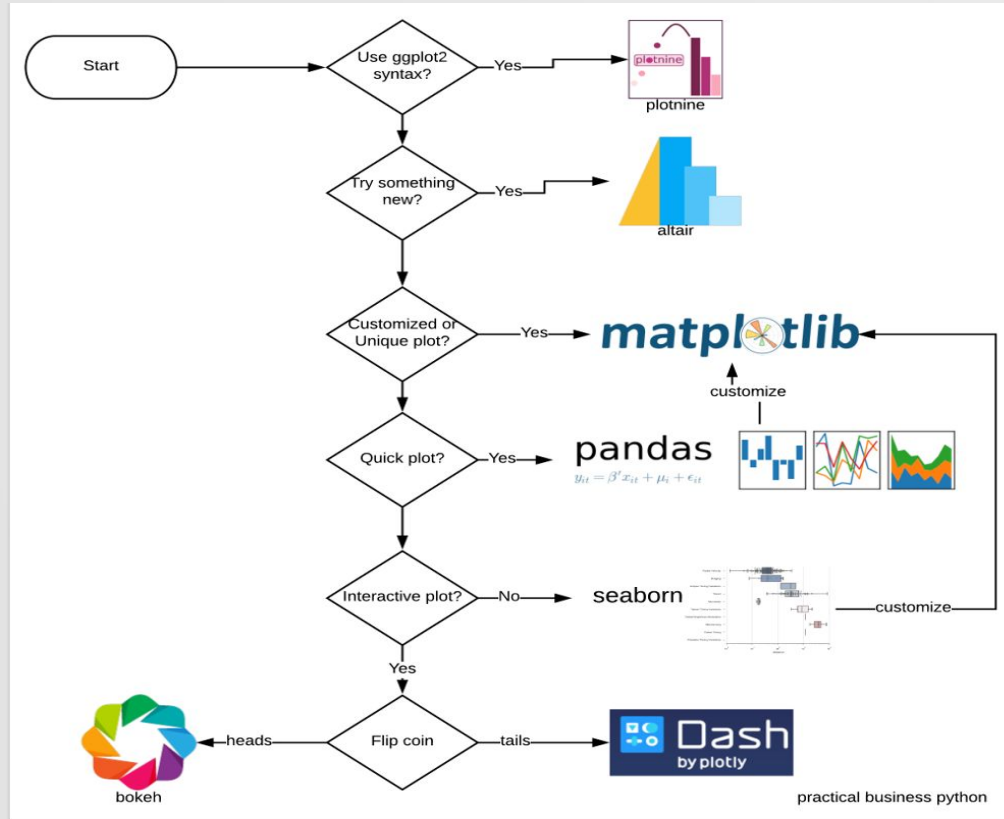
    # st.image(image, use_column_width=True)

if __name__ == '__main__':
    main()
```

Streamlit

Let's Download Some Data

- Supports multiple interactive visualization libraries like plotly, altair, bokeh, pydeck, etc.





03

Install Streamlit & Play

\$ pip install streamlit

\$ streamlit hello

hello - Streamlit x +

localhost:8501

Choose a demo

Animation Demo

Show code

Level of detail

1 100

Separation

0.79 0.70 2.00

Frame 67/100

Animation Demo

This app shows how you can use Streamlit to build cool animations. It displays an animated fractal based on the the Julia Set. Use the slider to tune the level of detail.

Re-run

04

Building a data app

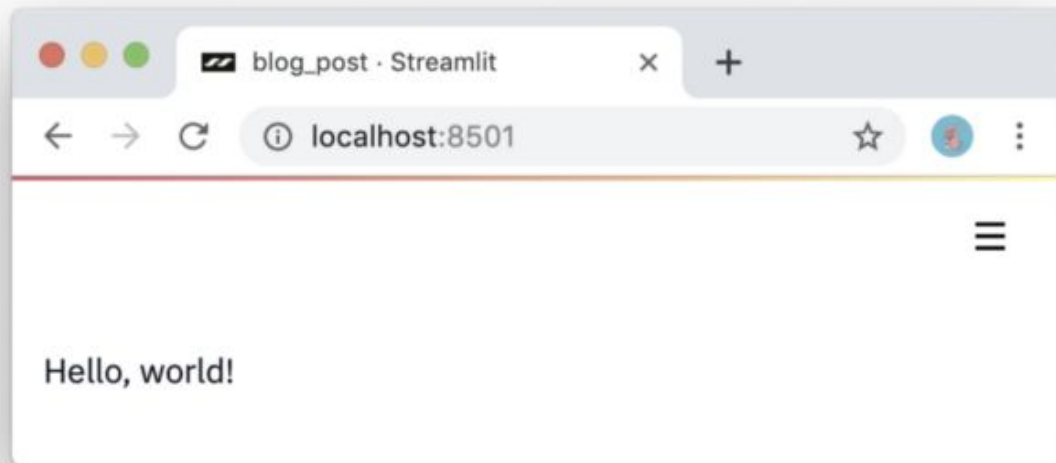
Streamlit process' walkthrough



```
import streamlit as st
```

```
st.write('hello world')
```

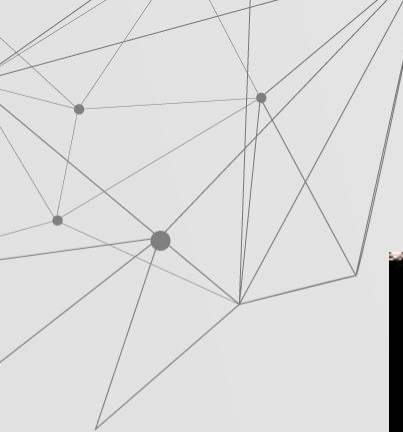
```
$ streamlit run hello.py
```





05

Sneak Peek & Use Cases



This Streamlit demo lets you perform semantic search across the **entire Udacity self-driving car photo dataset**, visualize human-annotated ground truth labels, and run a **complete neural net (YOLO)** in real time from within the app, in **less than 300 lines of code**.






06

**Maybe Streamlit is not
that awesome...**

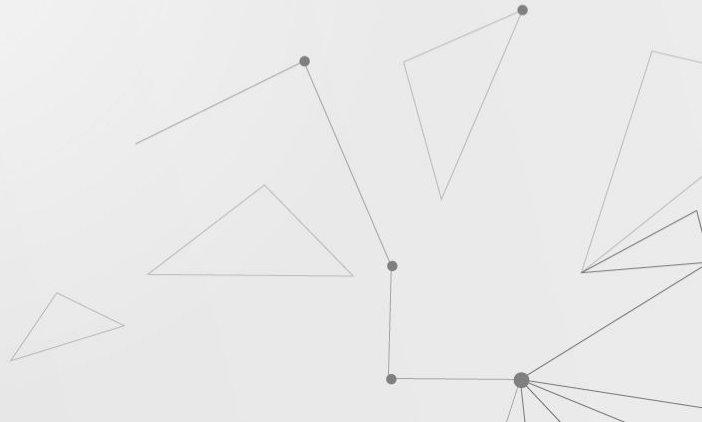


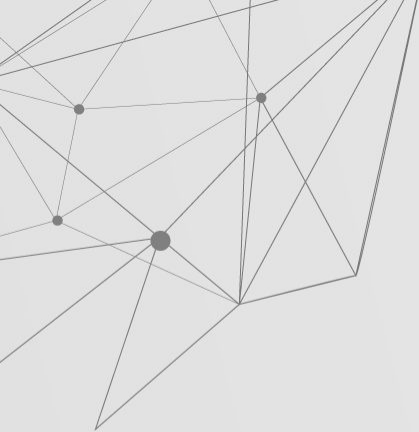
The Cons of Streamlit

- The Convenience-Flexibility Tradeoff
 - Size of data input (50mb max for data upload)
 - Limited support for video/animation
 - You can't build an app with functionalities like
 - User Authentication
 - Newsletter subscription
 - User-to-user interaction
- 



Even still...

- It's simple enough to use for every Python programmer.
 - It's superb if you want an interactive data app quickly.
- 

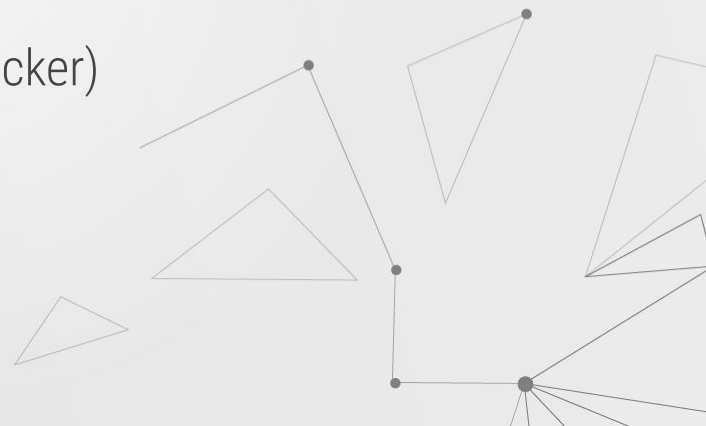


Fullstack Flask developer tools:

Flask and at least, JS, HTML, CSS & JSON

Fullstack Streamlit developer tools:

Streamlit (and maybe Docker)





Thank You!

@stevedev

<https://www.linkedin.com/in/steven-kolawole-80/>

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

Please keep this slide for attribution.