

Where Shiny Shines and Dash Disappoints: How to get started with reactive web GUIs in python

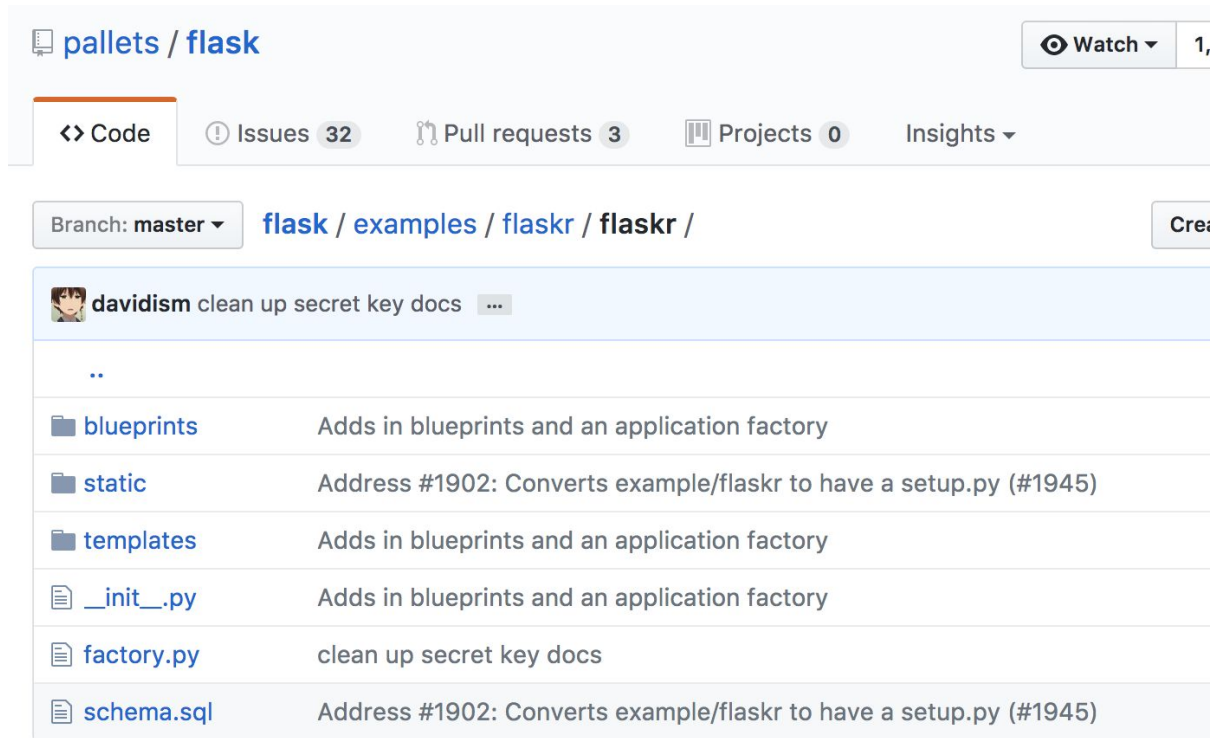
Kevin Mader/ CTO 4Quant
<https://github.com/4Quant/DashIntro>

Why?

You build cool tools other people need to use them

Web GUIs

1. Server (backend)
 - a. Python
 - b. Flask
 - c. Django
 - d. Tornado
 - e. SimpleHTTPServer
2. Client (front-end)
 - a. Javascript
 - b. ReactJS
 - c. VueJS
 - d. New one every week



The screenshot shows the GitHub interface for the `pallets/flask` repository. At the top, there's a navigation bar with tabs for `Code`, `Issues 32`, `Pull requests 3`, `Projects 0`, and `Insights`. Below this, the branch is set to `master`. The file list shows the following structure:

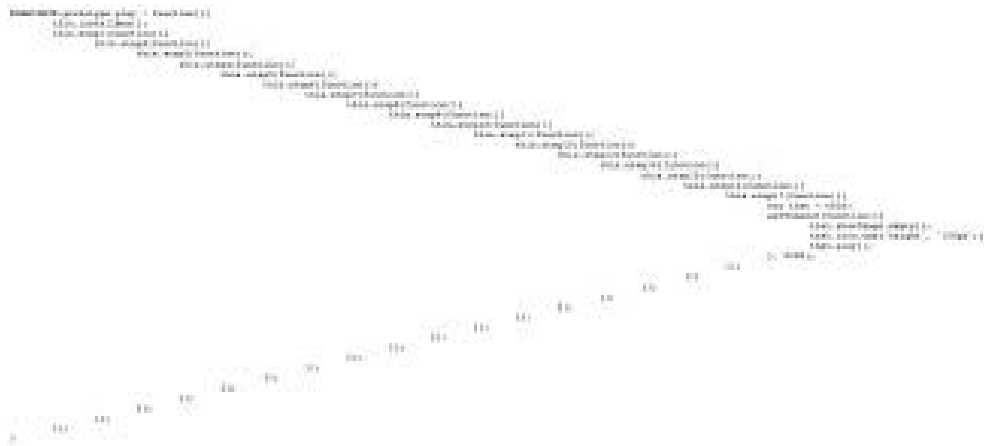
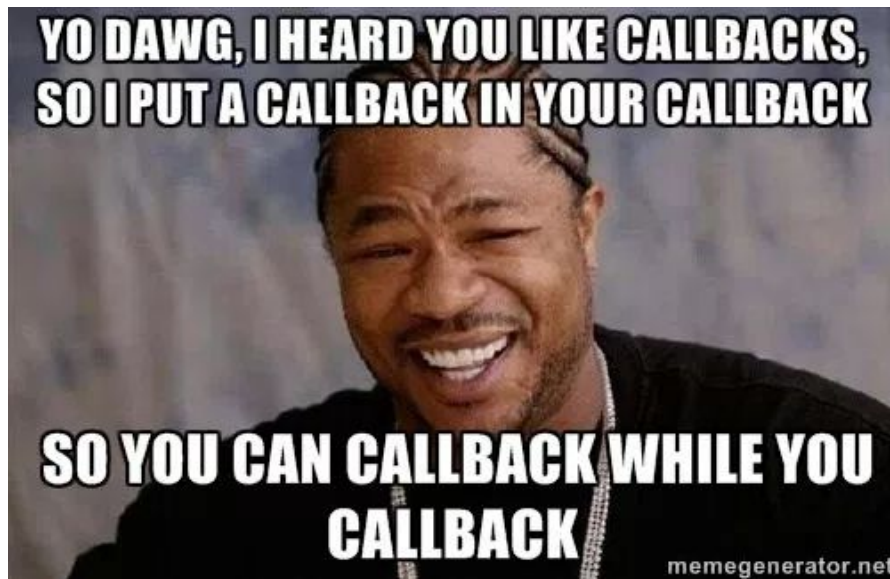
- `..`
- `blueprints`: Adds in blueprints and an application factory
- `static`: Address #1902: Converts example/flaskr to have a setup.py (#1945)
- `templates`: Adds in blueprints and an application factory
- `__init__.py`: Adds in blueprints and an application factory
- `factory.py`: clean up secret key docs
- `schema.sql`: Address #1902: Converts example/flaskr to have a setup.py (#1945)

A recent commit by `dauidism` is shown at the top of the file list, titled "clean up secret key docs".

Web GUIs

1. Callback hell
2. Add new button to website
3. Add new hook to backend
4. Add new template to code
5. Pull hair out making sure all names match **input-var != input_var**

6.



Enter Shiny

<https://www.showmeshiny.com/>

Plotly in Shiny



Just use a DAG

<https://shiny.rstudio.com/gallery/single-file-shiny-app.html>

```
# Global variables can go here
```

```
n <- 200
```

```
# Define the UI
```

```
ui <- bootstrapPage(  
  numericInput('n', 'Number of obs', n),  
  plotOutput('plot')  
)
```

```
# Define the server code
```

```
server <- function(input, output) {  
  output$plot <- renderPlot({  
    hist(runif(input$n))  
  })  
}
```

```
# Return a Shiny app object
```

```
shinyApp(ui = ui, server = server)
```

Dash

<https://www.showmeshiny.com/>

