Meeting #3

9-23-21

Michael Lee

Deliverables

- Create simple website
  - Multi-tabs
  - Includes graphs/videos
- Familiarize with individual components of dash

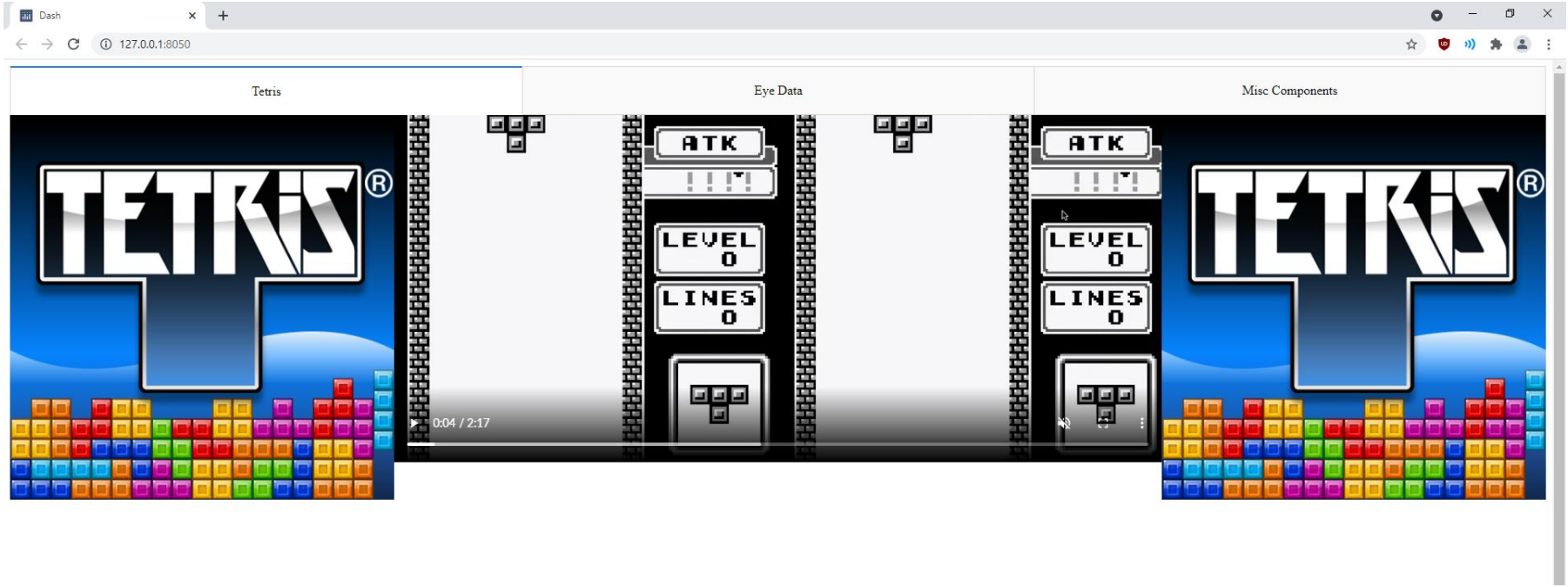
Methodology and Learnings

- Example code on plotly dash website
- Youtube tutorials
- Python formatting
- Callbacks

Results

```
import dash
from dash import dcc
from dash import html
import pandas as pd

app = dash.Dash(__name__)


def sep23_demo(appin, filepathin):
    df = pd.read_csv(filepathin)

    appin.layout = html.Div([
        dcc.Tabs([

            # create first tab
            dcc.Tab(label="Tetris", children=[
                # plays video content
                html.Video(
                    controls=True,
                    id='movie_player',
                    # video location - for static content, needs to be in a directory called static
                    # in same location as python file
                    src='./static/tetris.mp4',
                    autoPlay=True,
                    muted=True,
                    # format video size, location, etc.
                    style={'width': '50%', 'padding-left': '25%', 'padding-right': '25%'}
                ),

                html.Img(
                    id='left_tetris_img',
                    # image location - for static content, needs to be in a directory called static
                    # in same location as python file
                    src='./static/tetris.png',
                    # format image size, location, etc.
                    style={'width': '25%', "position": "relative", "top": "-430px"},
                    title="Tetris logo"

                ),
```

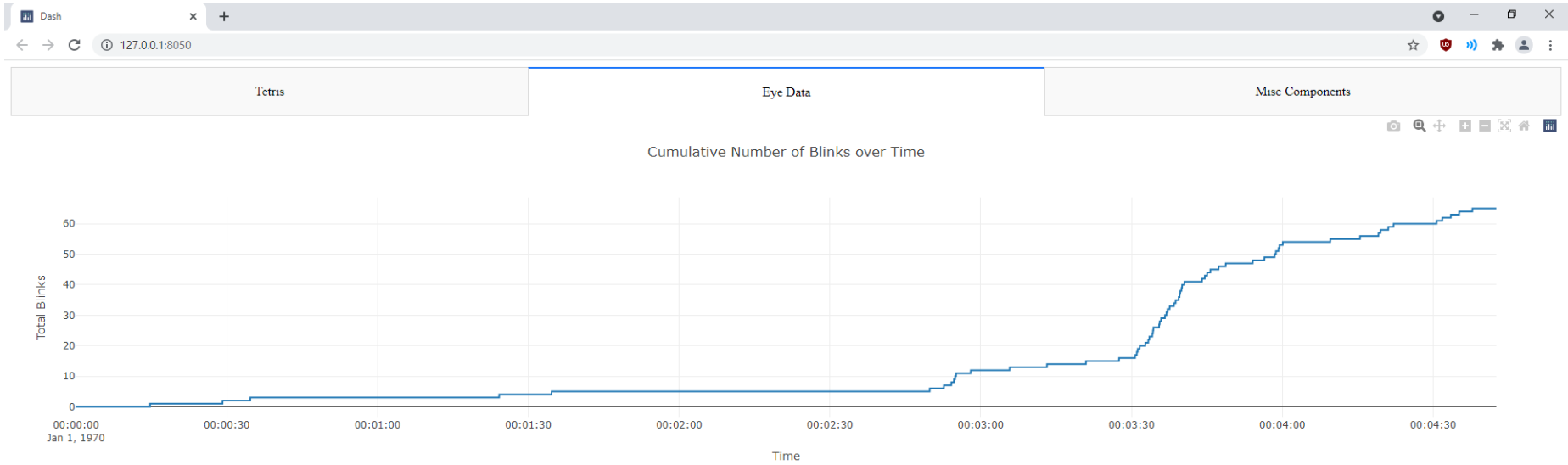
Results


Dash
127.0.0.1:8050
Tetris
Eye Data
Misc Components
ATK
LEVEL
0
LINES
0
0:04 / 2:17

Results

```
        # create second tab
        dcc.Tab(label="Eye Data", children=[
            dcc.Graph(
                figure={
                    # input and type of graph
                    'data': [
                        {'x': df["TimeStamp"], 'y': df["Cumulative_NumBlinks"],
                         'type': 'line'}
                    ],
                    # labels
                    'layout': {
                        'title': 'Cumulative Number of Blinks over Time',
                        'yaxis': {
                            'title': 'Total Blinks'
                        },
                        'xaxis': {
                            'title': 'Time'
                        }
                    }
                }
            )
        ]),
```

Results

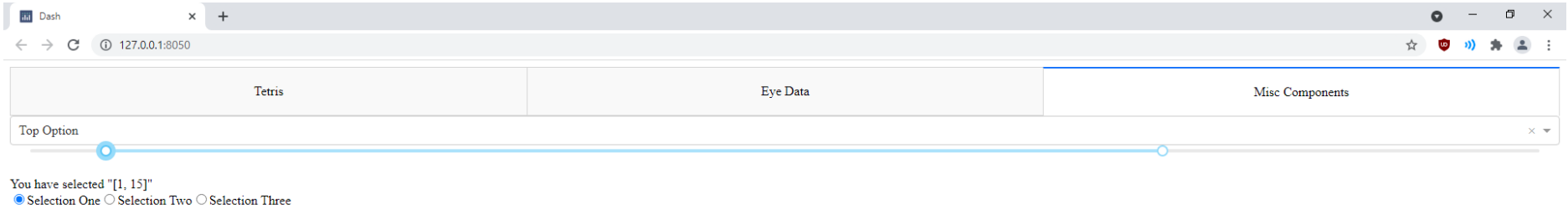

Dash
127.0.0.1:8050
Tetris
Eye Data
Misc Components
Cumulative Number of Blinks over Time
Total Blinks
60
50
40
30
20
10
0
00:00:00
Jan 1, 1970
00:00:30
00:01:00
00:01:30
00:02:00
00:02:30
00:03:00
00:03:30
00:04:00
00:04:30
Time

Results

```
        # create third tab
        dcc.Tab(label="Misc Components", children=[
            html.Div([
                # Example dropdown menu
                dcc.Dropdown(
                    id='mydropdown',
                    options=[
                        {'label': 'Top Option', 'value': 'TO'},
                        {'label': 'Bottom Option', 'value': 'BO'}
                    ],
                    placeholder=' '
                ),

                # Example slider with a range, uses a callback
                dcc.RangeSlider(
                    id='my-range-slider',
                    min=0,
                    max=20,
                    step=1,
                    value=[5, 15]
                ),

                html.Div(id='output-container-range-slider'),

                # Example Radio items
                dcc.RadioItems(
                    options=[
                        {'label': 'Selection One', 'value': "S1"},
                        {'label': 'Selection Two', 'value': "S2"},
                        {'label': 'Selection Three', 'value': "S3"}
                    ],
                    value='S1'
                )
            ])

        ])
    ])
])
```

```
# Updates text when the slider is changed
@app.callback(
    dash.dependencies.Output('output-container-range-slider', 'children'),
    [dash.dependencies.Input('my-range-slider', 'value')]
)
def update_output(value):
    return 'You have selected "{}"'.format(value)


if __name__ == '__main__':
    # filepath of data - data should be a csv
    filepath = 'D:/Users/Michael/Documents/College - Computer Science C++/University of Texas at Dallas/Fall 2021\CS ' \
               '4485-001 Computer Science Project/sep16demo/static/eyesstream_blink_data.csv'
    sep23_demo(app, filepath)
    app.run_server(debug=True)
```

Results


Dash
127.0.0.1:8050
Tetris
Eye Data
Misc Components
Top Option
You have selected "[1, 15]"
Selection One
Selection Two
Selection Three