

Journal 3:

- This week was very challenging. I struggled a lot trying to fix problems.
- Wednesday, I tried to work with different themes with Lektor, and also do scripting.
- So initially, I didn't know the themes that have to be changed are themes from the Lektor documentation. I found that out on Friday.
- So Wednesday, I wasted my time trying to figure out a bash script to make things work and changing the css file.
- Then, on Friday during the lab, I figured out how to make the theme change one by one. First you need to find the themes, which are here → <https://github.com/lektor/lektor-themes> , then clone it in your lektor project folder under the themes folder.
- Once you do that, you have to edit the .lektorproject file and add:
[project]
themes = lektor-theme-name
- Once you do that, you have to do lektor build, in the terminal and lektor server, to see on your localhost the changes of the webpage.
- I figured that out in the Friday lab. On Saturday, I was trying to figure out the scripting, but wasn't able to figure it out. I even asked chat gpt to ask for a script to change the themes every minute, but it didn't work, I was getting errors with running the server..
- Then, my teammate recommended that I use a python script to do it, since he managed to make it work.

- So, I did a python script that looked like this:

```

1 import os
2 from flask import Flask, render_template
3 import subprocess
4
5 app = Flask(__name__)
6
7 # Path to the Lektor project directory
8 PROJECT_DIR = '/home/parth/project2'
9
10 # Path to the Lektor project file
11 LKT_PROJECT_FILE = os.path.join(PROJECT_DIR, 'project2.lektorproject')
12
13 # List of themes to rotate
14 THEMES = ['nix', 'terminal', 'simple-strap', 'resume']
15
16 # Function to update the theme in the Lektor project file
17 def switch_theme():
18     # Initialize next_index in case the theme line isn't found
19     next_index = 0
20
21     # Read the current theme from the Lektor project file
22     with open(LKT_PROJECT_FILE, 'r') as file:
23         lines = file.readlines()
24
25     # Find the current theme line and update it
26     for i, line in enumerate(lines):
27         if line.startswith('theme = '):
28             current_theme = line.strip().split('= ')[1]
29             current_index = THEMES.index(current_theme)
30             next_index = (current_index + 1) % len(THEMES)
31             lines[i] = f'theme = {THEMES[next_index]}\n'
32             break
33
34     # Write the updated lines back to the file
35     with open(LKT_PROJECT_FILE, 'w') as file:
36         file.writelines(lines)
37
38     # Ensure next_index is correctly printed even if the theme line is not found
39     if next_index != 0:
40         print(f"Switched to theme: {THEMES[next_index]}")
41     else:
42         print("Theme line not found, no changes made.")
43
44 # Route for the home page that switches the theme and displays the content
45 @app.route('/')
46 def home():
47     switch_theme()
48
49     # Restart the Lektor server
50     subprocess.run(["pkill", "-f", "lektor server"], check=True)
51     subprocess.run(["nohup", "lektor", "server", "&"], check=True)
52
53     # Render a response or template
54     return "Theme switched and server restarted. Visit the Lektor site to see changes."
55
56 if __name__ == "__main__":
57     app.run(debug=True, host='0.0.0.0', port=5000)
58

```

- I never coded in python so I did ask chatgpt for some help.
- This, from what I saw in the terminal, was the closest thing to success, since I saw a message saying the themes were switching. The only problem was it wasn't switching on the web. After 1 minute, when the "theme switches", I see this on the webpage:



Unable to connect

Firefox can't establish a connection to the server at 127.0.0.1:5000.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the web.

Try Again

- I tried to fix this, asking the AI again, which did not help me at all. Because it said I had to add a time interval to let the server restart, but that did not work.

According to AI:

Wait a bit to allow the server to start

time.sleep(5) → (solution)

Adjust this wait time as needed # Check if the server is running

- So this week I still wasn't able to figure out scripting, but my teammate has been able to figure out a solution. The only problem with that solution, is that it is not every minute, it is every 12 hours.
- The guide/manual of this project is also in progress, as well as the powerpoint for Tuesday.
- On the project, our goals for the week changed a lot, we ran into a lot of problems like setting up the VPS server. We ran into problems we hadn't thought of before. We even questioned if Lektor was the right decision. The plan was to be done with scripting by now for me, but that wasn't possible. The plan was also to be completely done by week 3 which isn't the case.

Git Repository: <https://github.com/ParthPatelA/UnixProject.git>