**Appendix 1: Planning Guide**

Task 0: Explain what you are doing/ going to accomplish

In this version I am going to add pictures to my menu

Task 1: Sketch interface design

*Draft a rough design for the interface that allows the user to trigger functionality in task 1, while also annotating where the information in task 2 will be displayed. Create another sketch listing the interface widgets used to create the interface.*

Task 2: Identify any classes required

*Explain what the class will represent, plus listing what information will be stored in the class and any functions the class will have.*

I will need to add picture names to the existing class

Task 3: Identify information to be displayed

*What information will the interface need to display to the user?*

Picture will be displayed to the user

Task 4: Identify user inputs

*What program functions can the user trigger through the interface?*

N/A

Task 5: Identify any constants or existing data if required

N/A

Task 6: Identify indexed data structures

Task 7: Determine what calculations are necessary

*Write out the calculations the program will have to compute.*

Task 8: Develop a modular structure for your program

*Describe any functions that the computer program will have, identifying any sub-functions where required.*

I will need a function for my photo converter in python so that it is a static image

Task 9: Define the functions identified

*Describe the functions for both the main program and any classes in terms of input and/or output where required. You may choose to do this with flow charts or pseudo-code (not Python code!). Add in additional steps or explanations using sequential, conditional, iterative statements where required. Identify global and/or local variables.*

|  |
| --- |
|  |
|  |  |

Route will be ‘/picture/<filename>’

PROGRAM serve\_picture (filename)

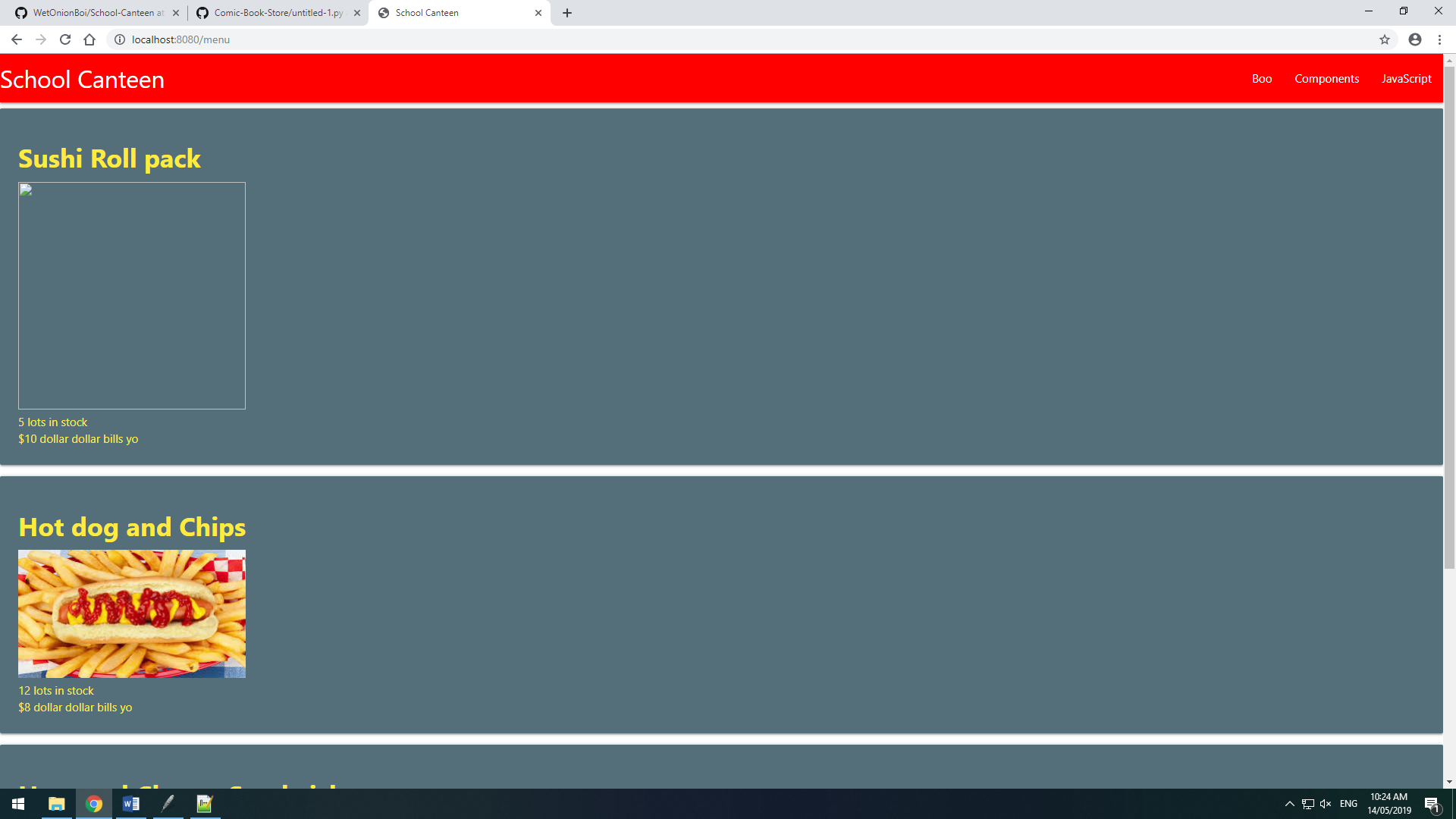
Return static\_file with filename, root = './Images')

Task 10: Address any relevant implications such as usability, functionality, legal/ethical requirements.

Task 11: Document test cases for testing the program

*Document any testing that can be used to test your program. If any input is inputted using the keyboard, describe the expected input, plus any exceptional, boundary or invalid cases.*

I am going to run my website and see whether the pictures appear like they are supposed to



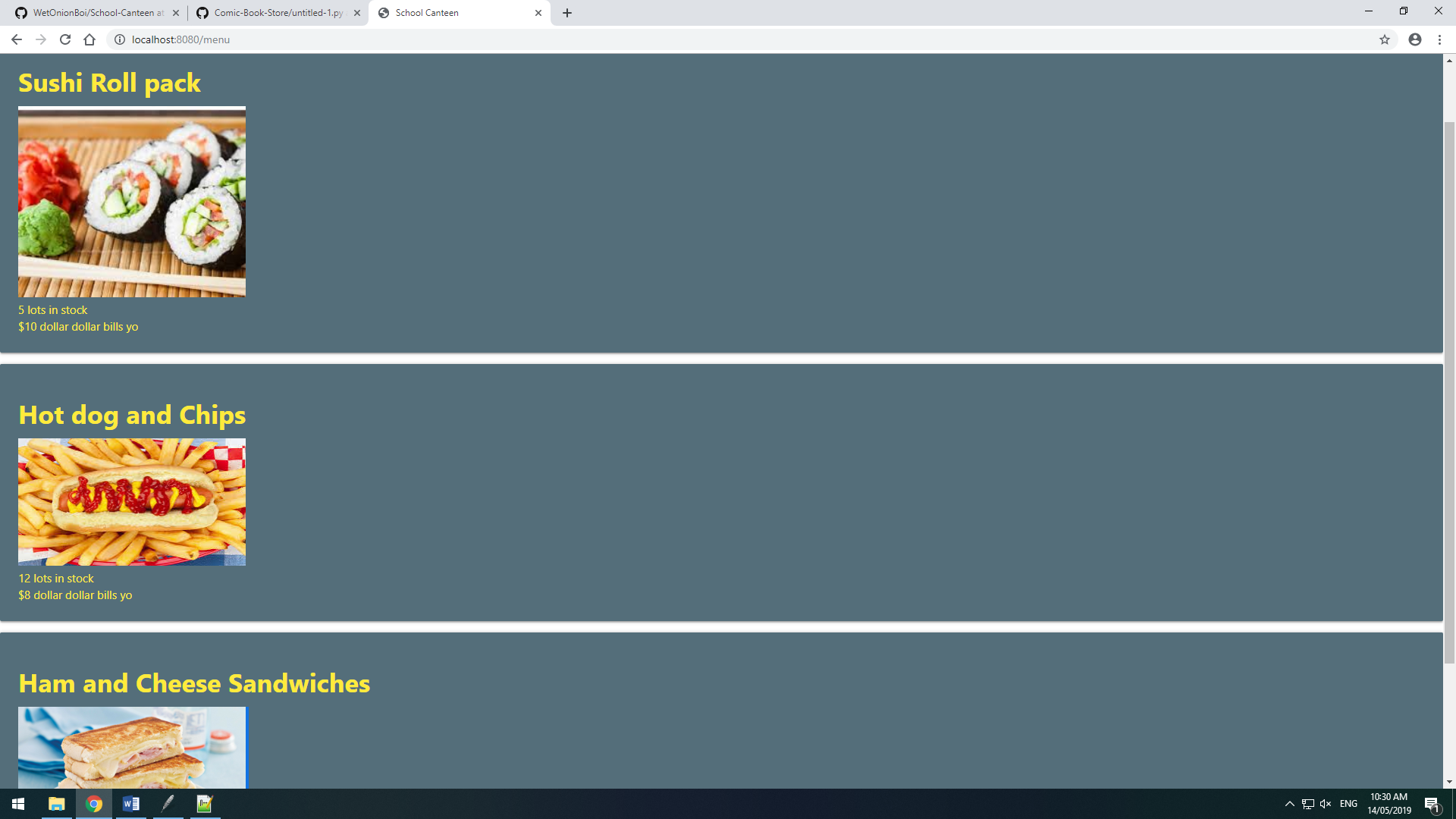
Task 12: Refine the plan

*Note any modifications here when iterating through the development cycles.* \

When I ran the code the first time and checked my website pictures appeared for my .jpg files but not for my .jpeg. I am going to try find a different picture that isn’t a .jpeg file.

Task 13: Document testing

*Show screenshots of your program working with descriptions of each image. These images should test the tests cases listed above.*



After converting the file to jpg it worked well

Task 14 : Evaluation

*How did your version turn out*

This version worked as I planned it to after I changed it