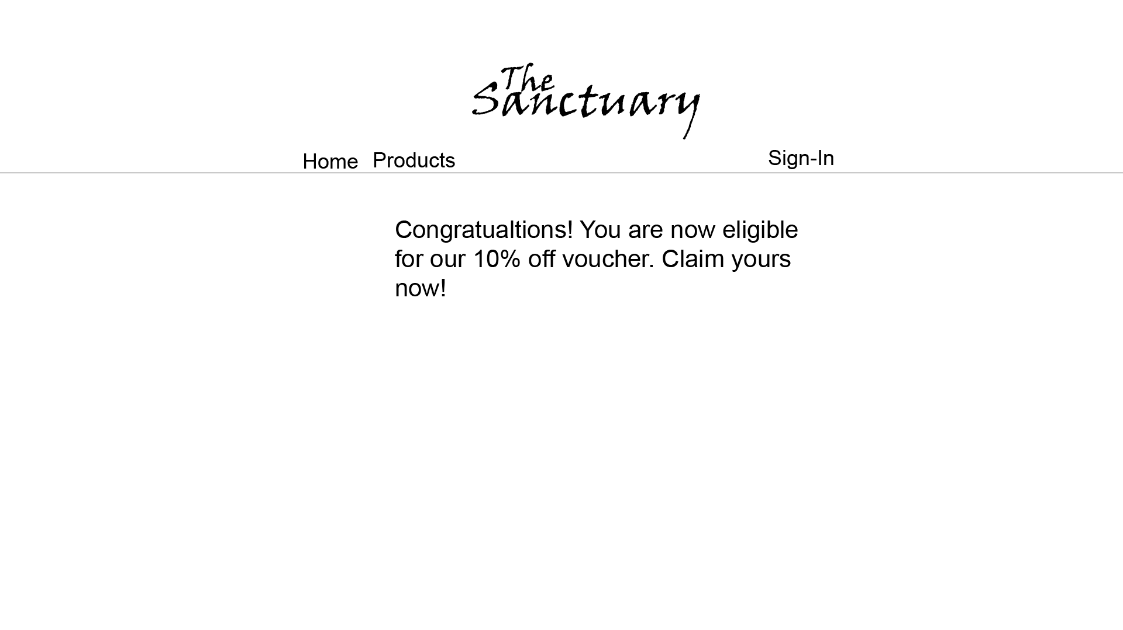
Task 0 : Explain what you are doing/ going to accomplish

I am going to create a success page that is the continuation of the book-list page.

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.*

None

Task 3: Identify information to be displayed

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

Display book information of the book that they are purchasing

Task 4: Identify user inputs

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

User input from button which will be comic id

Task 5: Identify any constants or existing data if required

None

Task 6: Identify indexed data structures

None

Task 7: Determine what calculations are necessary

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

None

Task 8: Develop a modular structure for your program

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

PRINT COMIC BOOK NAME  
PRINT COMIC BOOK IMAGE  
PRINT COMIC BOOK PRICE

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.*

None

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

Same -

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.*

Task 12: Refine the plan

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

I decided that I should separate the book-list-page into two and make a buy-book-success page as this would make it more like buying an actual product from a store. You first check the book out, then decide whether to buy it or not.

I also realised that the stock could go negative, this should not be possible so I will add code that “greys out” the button if there is 0 stock so that it cannot be negative.

PSEUDO CODE

if comic.stock is greater than 0:

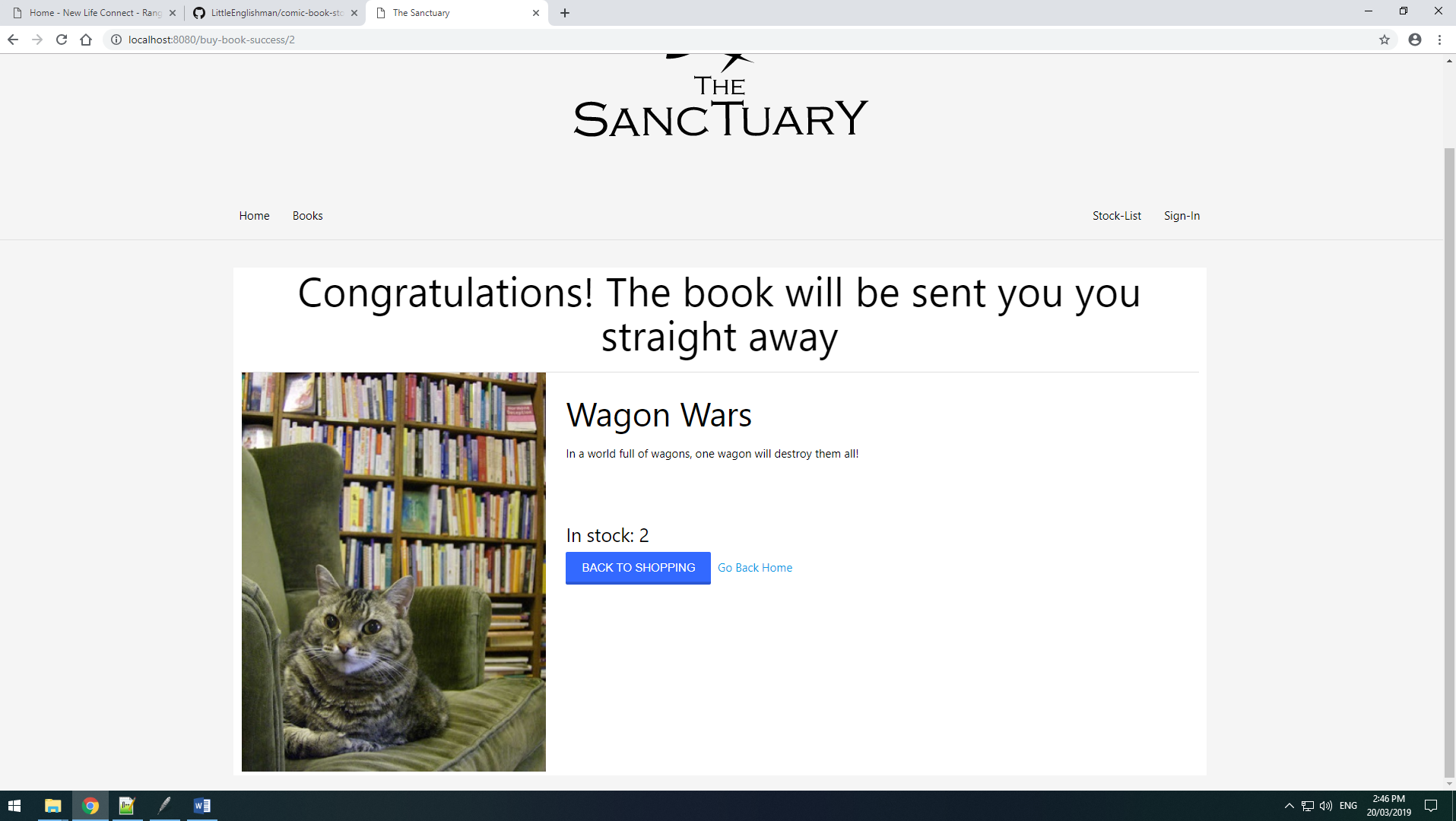
Have a normal button

Else do this

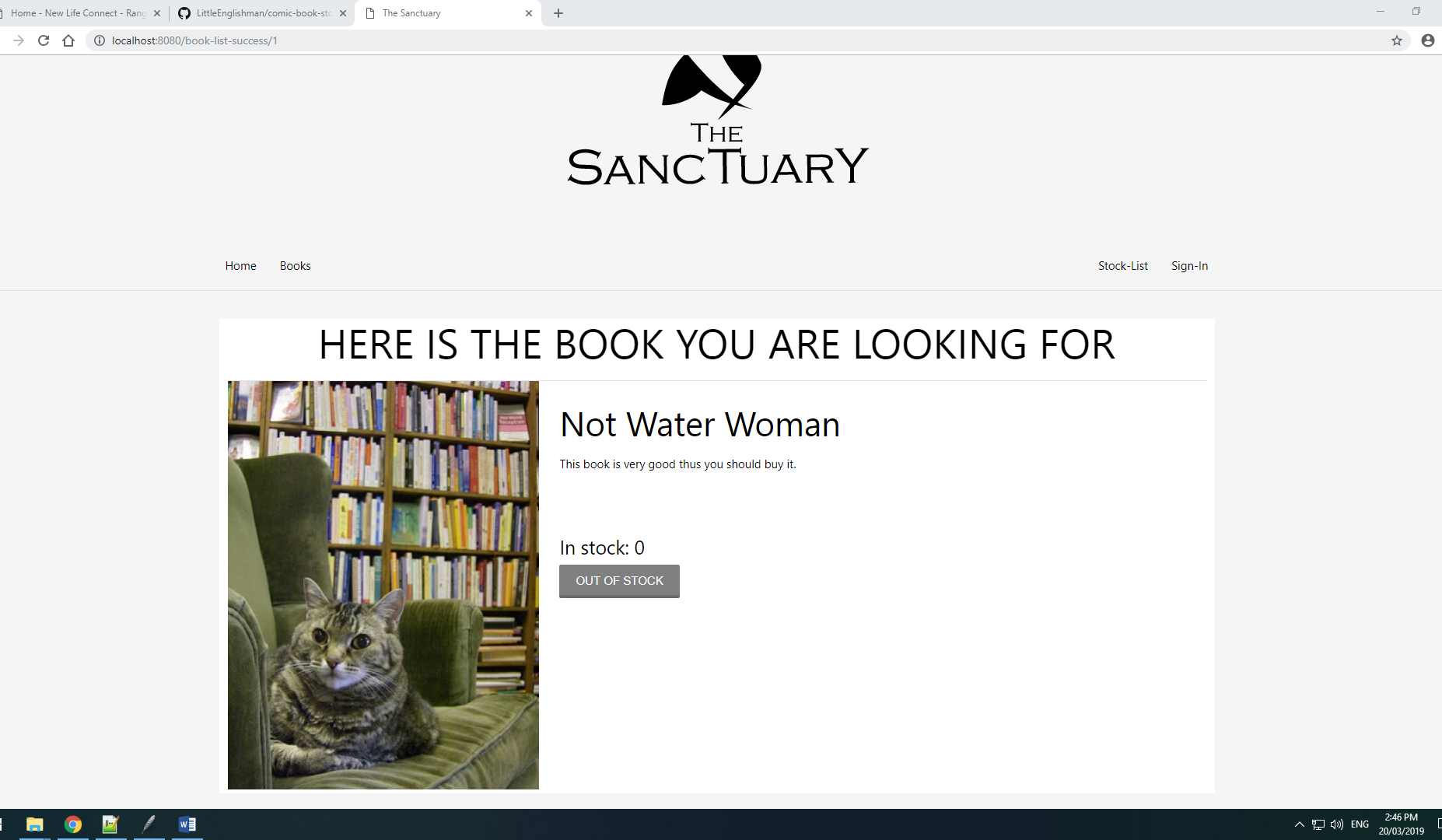
Show an unusable button with out of stock written on it.

Task 13: Document testing

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



Here the shop page is working as it display all the correct things as well as after being bought giving the option of going back to home or back shopping.



Here the stock is 0 so the buy button has been changed to an out of stock button to show the user it is unavailable at the moment.