VERSION 1.2 PLANNING

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

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.*

Comic. Stores each of my comics.

Task 3: Identify information to be displayed

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

The price, amount in stock, name and the image of each of the comics in stock.

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

My class containing comics called Comic

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.*

serve\_picture to supply image.

\_\_init\_\_\_ to tell what is in each comic profile.

product\_page to make the product page exist.

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 is product-page

View is product-page

Define product page

Make data a dictionary of my tickets

Pass data

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

I will make this functional by clearly showing each ticket and the data associated with it. I will make it usable by having it linked from the index page. This will allow people to access it.

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.*

Task 13: Document testing

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