**Appendix 1: Planning Guide**

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

Create the index page, the cart page and the header of the 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.*

Header, Nav Bar

Task 3: Identify information to be displayed

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

A table of stock of the comics

Task 4: Identify user inputs

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

Just in the header, the user will need to click to different pages.

Task 5: Identify any constants or existing data if required

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.*

IMPORT run, route, view, get, post, request FROM BOTTLE

IMPORT count FROM INTERTOOLS

PROGRAM\_IDS\_recievesself.name, image, amount

Self.ids is set to next self\_ids

Self.comic\_name is set to name

Self.comic\_image is set to image

Self.comic\_amount is set to amount

END

SET LIST test\_comics to

Set comic\_book to super dude image 8

Set comic\_book to lizard man image 12

Set comic\_book to water woman image 3

run(host='0.0.0.0', port=8080, reloader = True, debug = True)

END

@route(“/”)

@view():

PASS

@route(“Kid”)

@view(Kid):

|  |
| --- |
|  |
| Def kid():  PASS  RUN(host=’0.0.0.0’, port=8080, reloader = TRUE, DEBUG = TRUE) |

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.*

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.*

There is now an interface with the index page and a second page

Task 12: Refine the plan

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

N/A

Task 13: Document testing

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