Introduction

This test plan shall ensure the functionality of the entire computer builder website from front to back. The three main parts of this website are the web scraping, database, and website. Each part will be tested thoroughly to keep the website running efficiently since each part depends upon the others. For the web scraping part, the goal is to test each individual function to verify that the data being pulled is up to date, relevant, and accurate. The goal of database testing is to check that the information obtained from web scraping is stored correctly and can be pulled efficiently by the website. While testing the website, the goal is to ensure that customers can easily navigate the website and purchase a computer with no hassle.

Approach

The following testing is designed to assess the reliability and speed of the website from front to back. These tests will include function calls, button clicks, and data insertion/deletion. The test method that will be used is white box testing because the tester knows all the internal components of the product. These tests will be executed manually due to multiple moving parts.

Test Case ID	1	Test Case Description	Check database connection		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Host = localhost
2	Web-scraping Program	2	User = root
3		3	Password = ******
4		4	Database = ComputerParts

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Launch web-scraping program	You are connected	You are connected	Pass
2	Wait for program to finish execution	You are disconnected	You are disconnected	Pass
3				

Test Case ID	2	Test Case Description	Check that all records are deleted from each table in the database		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL database	1	Table full of data
2	Web-scraping program	2	
3		3	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Run web scraper to delete all records in each table	Each table should be empty and no error	No error	Pass
2	Run MySQL query to show all records in each table	Each query should return empty set	MySQL query returned “empty set”	Pass
3				

Test Case ID	3	Test Case Description	Check that records are inserted properly into the according table		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Name (varchar 200)
2	Web-scraping Program	2	Price (decimal(6,2))
3		3	table
4		4	Insert into query

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Run web scraper into query on table	Records inserted into table	Records are inserted properly into appropriate table	Pass
2	Run a select all query on table	New record should be in the table	MySQL query returned all part names and price inserted	Pass
3				

Test Case ID	4	Test Case Description	Check that main function runs on a 24hr timer		
Tester's Name		Nate	Test Date	04/01/2024 – 04/02/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	URLs
2	Web-scraping Program	2	
3	BeautifulSoup library	3	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Run main.py	You are connected	You are connected	Pass
2	Wait for disconnect message	You are disconnected	You are disconnected	Pass
3	Keep program running for 24hrs	You are connected, you are disconnected	PC shutoff before 24hr timer	Suspended

Test Case ID	5	Test Case Description	When the submit button is clicked, the website reroutes to the checkout page		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Part names
2	Web-scraping Program	2	Part prices
3	Website	3	
4		4	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Go to pcbuilder page	Pcbuilder page loads	Pcbuilder loaded properly	Pass
2	Select desired computer parts	Price is updated and names of selected parts is shown	Price update failed and selected part was shown	Fail
3	Click the purchase button	Checkout.blade.php is redirected automatically	User is redirected to checkout page	Pass

Test Case ID	6	Test Case Description	Check that getProductInfo function pulls every item on each page only if it has a listed price		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	URL
2	Web-scraping Program	2	An empty array
3	BeautifulSoup library	3	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Call getProductInfo passing a URL as a parameter	The array is full of data where every even position in the array including 0 has is a string name and odds are prices stored as double	Array is filled with part names and prices according to website they were scraped from	Pass
2	Check that every iteration of the array has a price and part name	Array has no missing price or part name	Every iteration of the array has one part name and one price	Pass
3				

Test Case ID	7	Test Case Description	Check that runPages function will run recursively until there are no more pages		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	URL
2	Web-scraping Program	2	
3	BeautifulSoup library	3	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Call runPages passing a URL as the parameter	Run pages calls itself (number of pages – 1) times	Runpages ran 5 times with 6 pages	Pass
2				

Test Case ID	8	Test Case Description	Check that getData function will return a string full of html code		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	URL
2	Web-scraping Program	2	
3	BeautifulSoup library	3	
4		4	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Call getData function passing URL as a parameter	Returns a string containing all the html code from the requested URL	Only requested HTML data was returned by getData function	Pass
2				
3				
4				

Test Case ID	9	Test Case Description	Check that the price updates as user selects different parts		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Part names
2	Web-scraping Program	2	Part prices
3	Website	3	
4		4	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Go to website	Website loads in browser	Website is visible on local machine	Pass
2	Change a part	Total price updates to reflect change in price	Price was not updated	Fail
3	Change a different part	Total price updates to reflect change in price	Price was updated	Pass
4				

Test Case ID	10	Test Case Description	Check that the website dropdowns populate with everything in the appropriate table		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Part names
2	Web-scraping Program	2	Part prices
3	Website	3	
4		4	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Run web-scraping program	Database is updated with fresh data	You are connected then you are not connected	Pass
2	Go to website	Website loads in browser	Website is visible on local machine	Pass
3	Select a part dropdown menu	Every part listed in that corresponding table is shown to the user	All parts in MySQL database populate the dropdown box	Pass
4				

Test Case ID	11	Test Case Description	New users are able to register their account as long as email is valid		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Name
2	Website	2	Email
3	Login Route	3	Password
4		4	Confirm Password

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Go to register page	Website redirects to register page	Redirected to register page	Pass
2	Fill in input boxes	Errors pop up if applicable	Bad email displayed error	Pass
3	Click register	Account is created in database and website redirects to home page	Account is created in database and user is redirected to home page	Pass
4				

Test Case ID	12	Test Case Description	Returning users are able to login if account is already created		
Tester's Name		Nate	Test Date	04/01/2024	

S #	Prerequisites	S #	Test Data
1	MySQL Database	1	Email
2	Website	2	Password
3		3	
4		4	

Step #	Step Details	Expected Results	Actual Results	Pass/ Fail / Not Executed / Suspended
1	Go to login page	Website redirects to login page	Website redirects to login page	Pass
2	Fill in input boxes	Errors show if account isn’t in database. Otherwise no errors.	No errors	Pass
3	Click Login	Account is verified and user is redirected to home page.	Account is found and website redirects to home page	Pass
4				