**Test Plan (Open Cart..)**

[**Test Plan (Open Cart..)**](#_5cjd4yit9qf4) **0**

[Objectives:](#_ib6peqlsr70a) 1

[Scope:](#_omfxvgklvjqp) 2

[Inclusion](#_6e5tpqwe6u3a) 2

[Login:](#_ormtwuh9bio4) 3

[Registration:](#_1sab89wxiut5) 3

[Add to Cart:](#_ewgt1trqtbfb) 4

[Check out/Payment Process:](#_at0g6qr12hjd) 5

[Exclusion:](#_6l585jsf3kdk) 7

[Testing Environment:](#_xw3i4a2oqxc6) 7

[Test Strategy](#_4d34og8) 7

[Test Schedule](#_2s8eyo1) 8

[Test Deliverables.](#_17dp8vu) 9

[Entry and Exit Criteria](#_3rdcrjn) 9

[Entry Criteria:](#_26in1rg) 9

[Exit Criteria:](#_lnxbz9) 10

[Test Execution](#_35nkun2) 10

[Entry Criteria:](#_1ksv4uv) 10

[Exit Criteria:](#_44sinio) 10

[Test Closure](#_2jxsxqh) 10

[Entry Criteria:](#_z337ya) 10

[Exit Criteria:](#_3j2qqm3) 10

[Tools](#_1y810tw) 10

[Risks and Mitigations](#_4i7ojhp) 10

[Approvals](#_2xcytpi) 11

## 

## 

## **Objectives:**

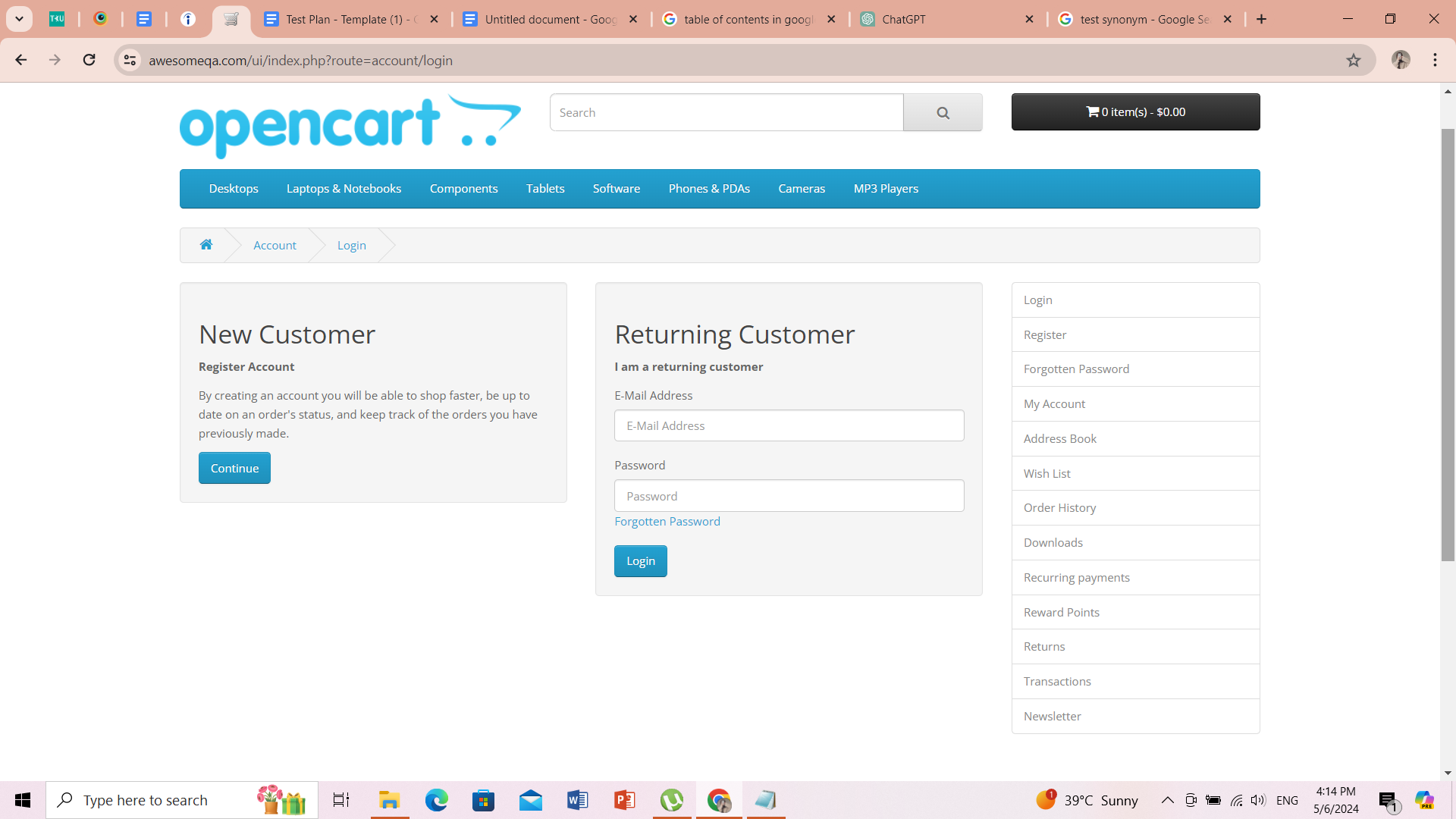
The objective to test an e-commerce open cart application is to check its main modules to ensure the usability, flexibility and security of the website. To comprehensively test the login, registration, payment, and add-to-cart modules of the e-commerce website to ensure seamless functionality, usability, and security for users throughout their shopping journey. The testing will focus on verifying that users can successfully create accounts, log in securely, add items to the cart, proceed through the payment process without errors, and complete transactions smoothly. Success criteria include successful registration, login with accurate username and password, accurate add to cart process and validating the accuracy and reliability of user data storage, encryption of sensitive information during transactions, and the smooth transition of data between modules.

## **Scope:**

The main scope is to cover four main and important modules of this website

1. Login
2. Registration
3. Add to cart
4. Make payment

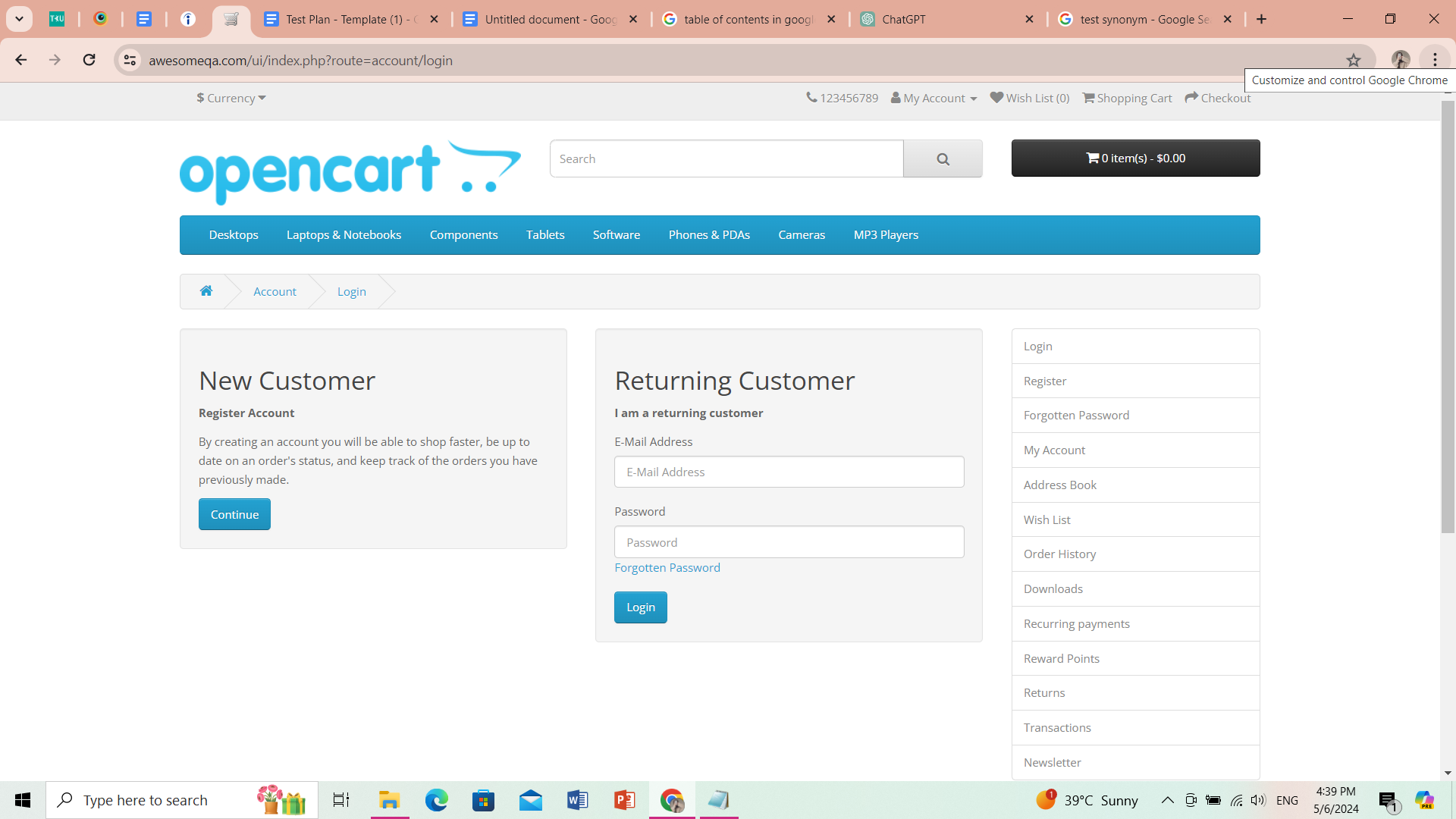
The types and testing that will be performed such as manual testing, Gui testing, Functional testing, Non-Functional Testing. The environment in which testing will be conducted such as different browsers operating system and device types. The criteria that will be used to evaluate the success of the testing, such as the number of defects found, the time taken to complete the testing, and user satisfaction ratings.The schedule and milestones for the testing, including the start and end dates, and the planned testing activities. The tools and equipment that will be used for testing, such as testing software, hardware, and documentation templates.



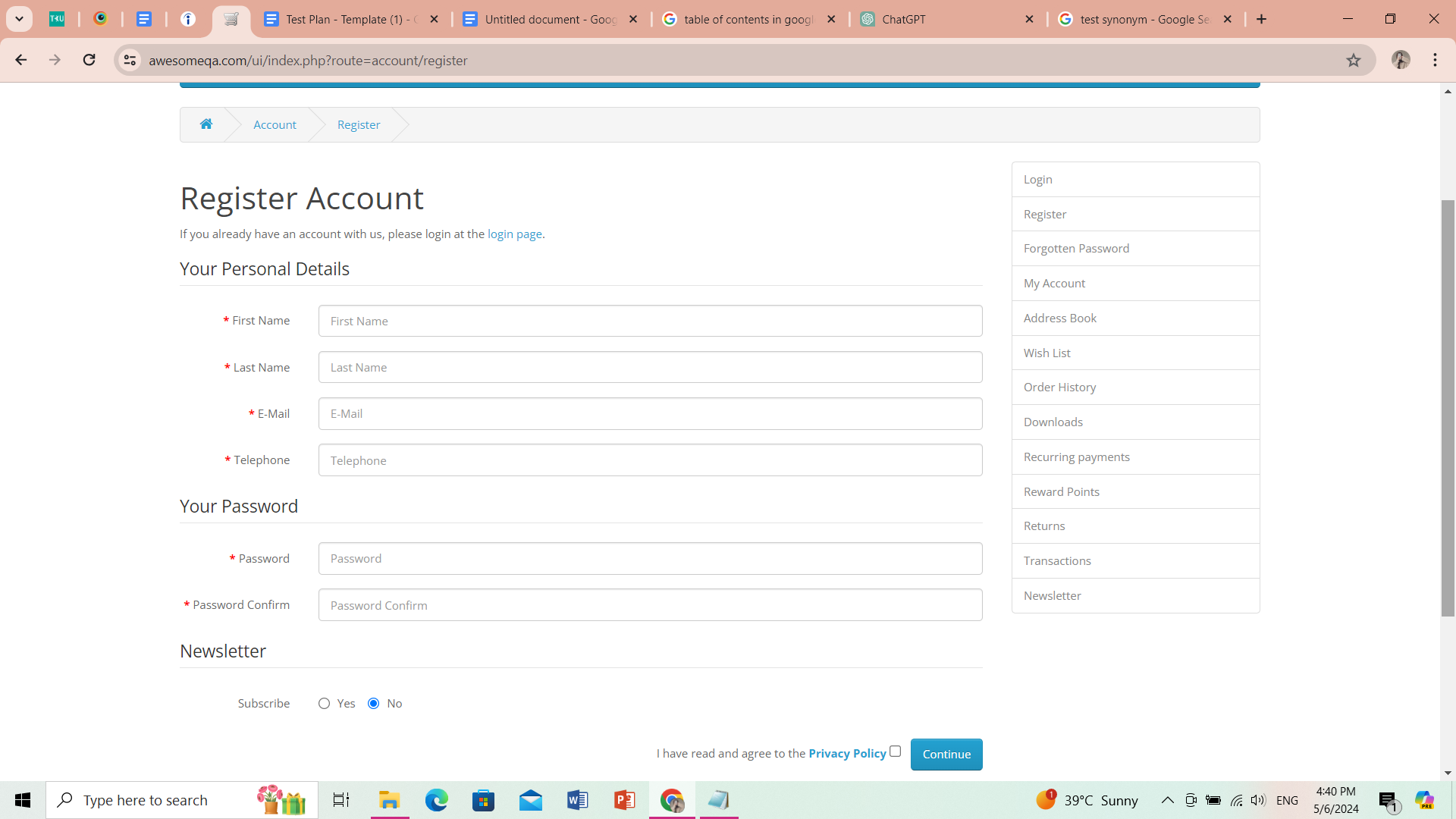
### **Inclusion**

This part will mention all the modules which we will be going to test for this website.

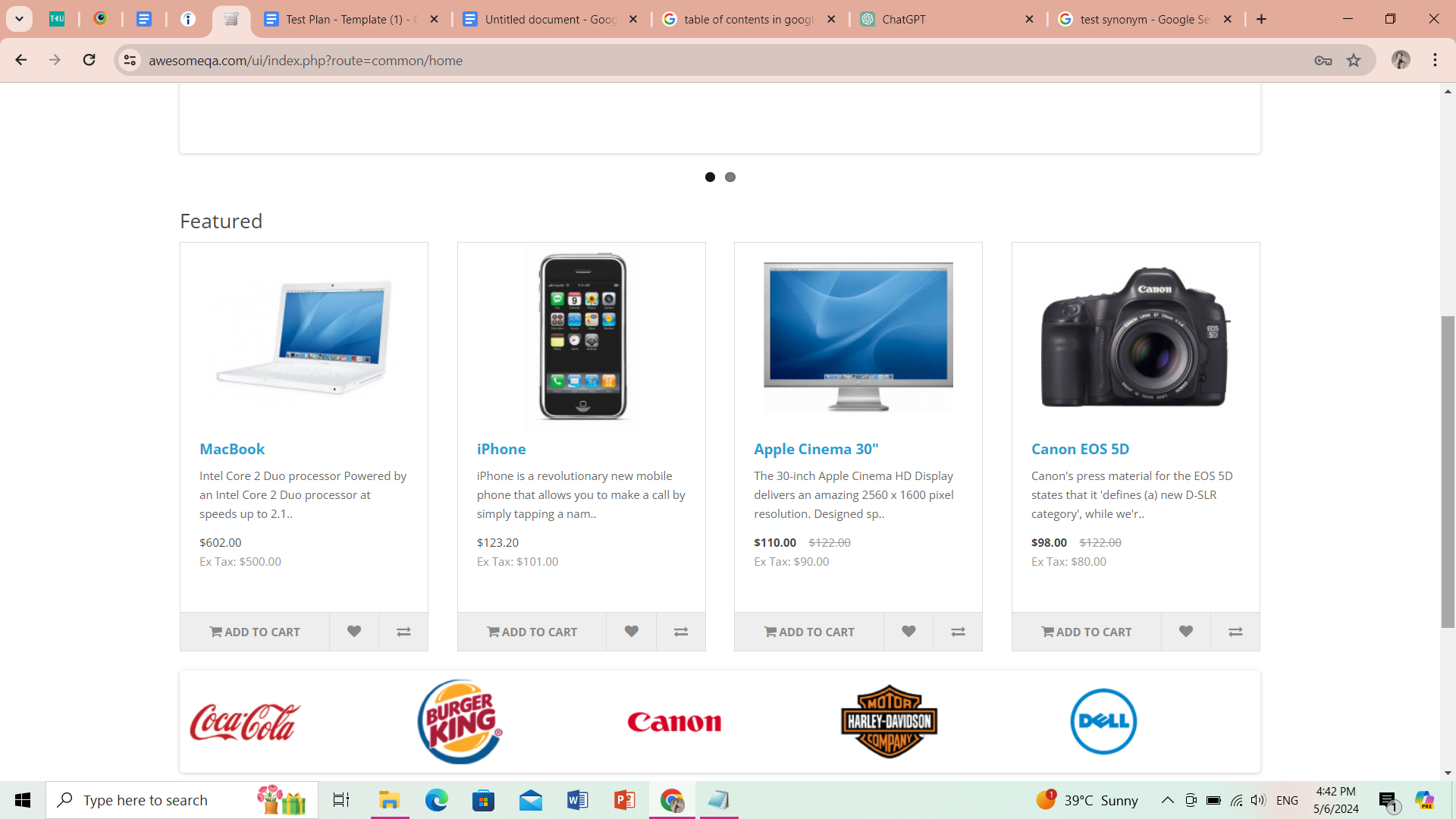
### **Login**:

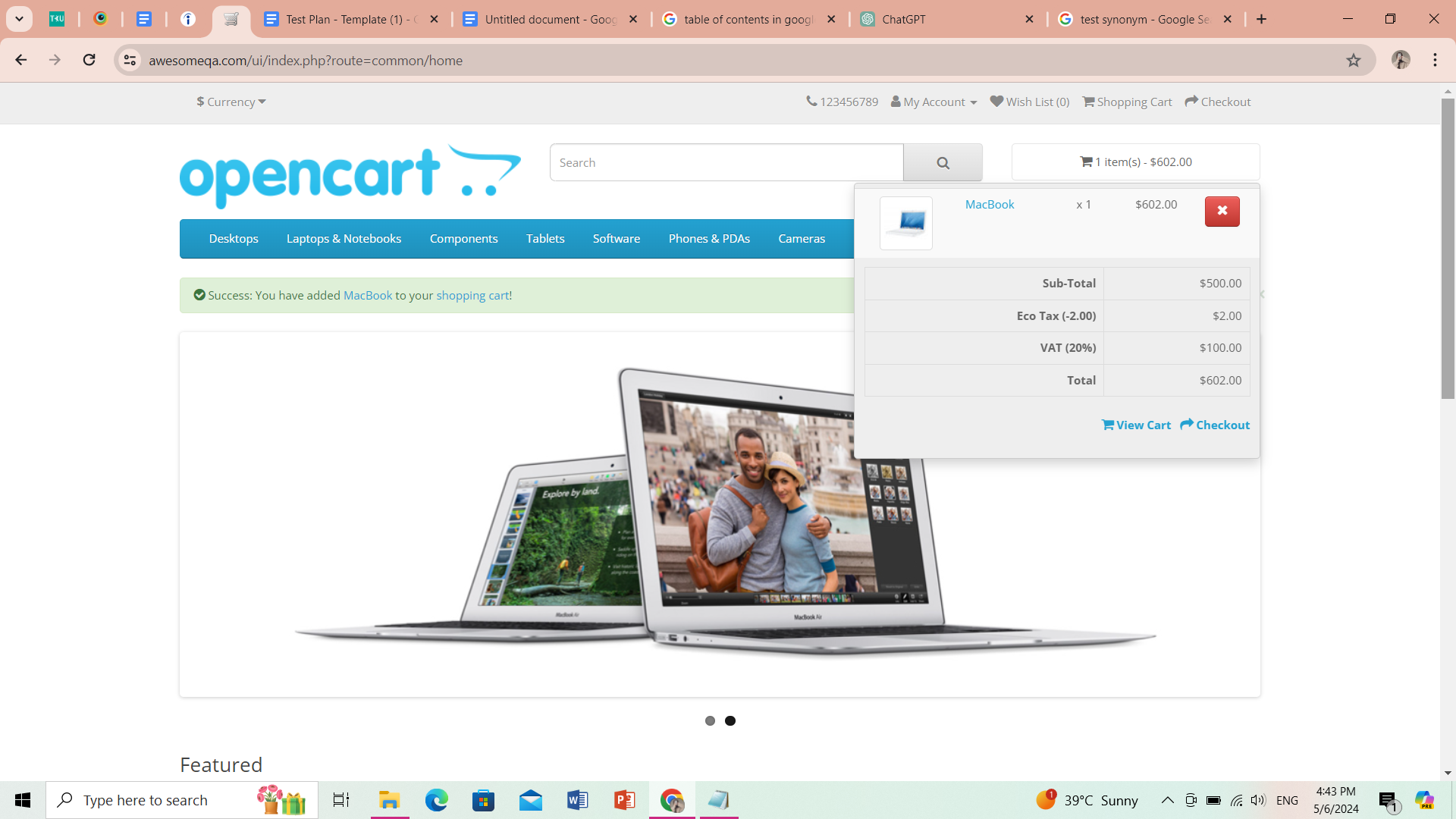
****

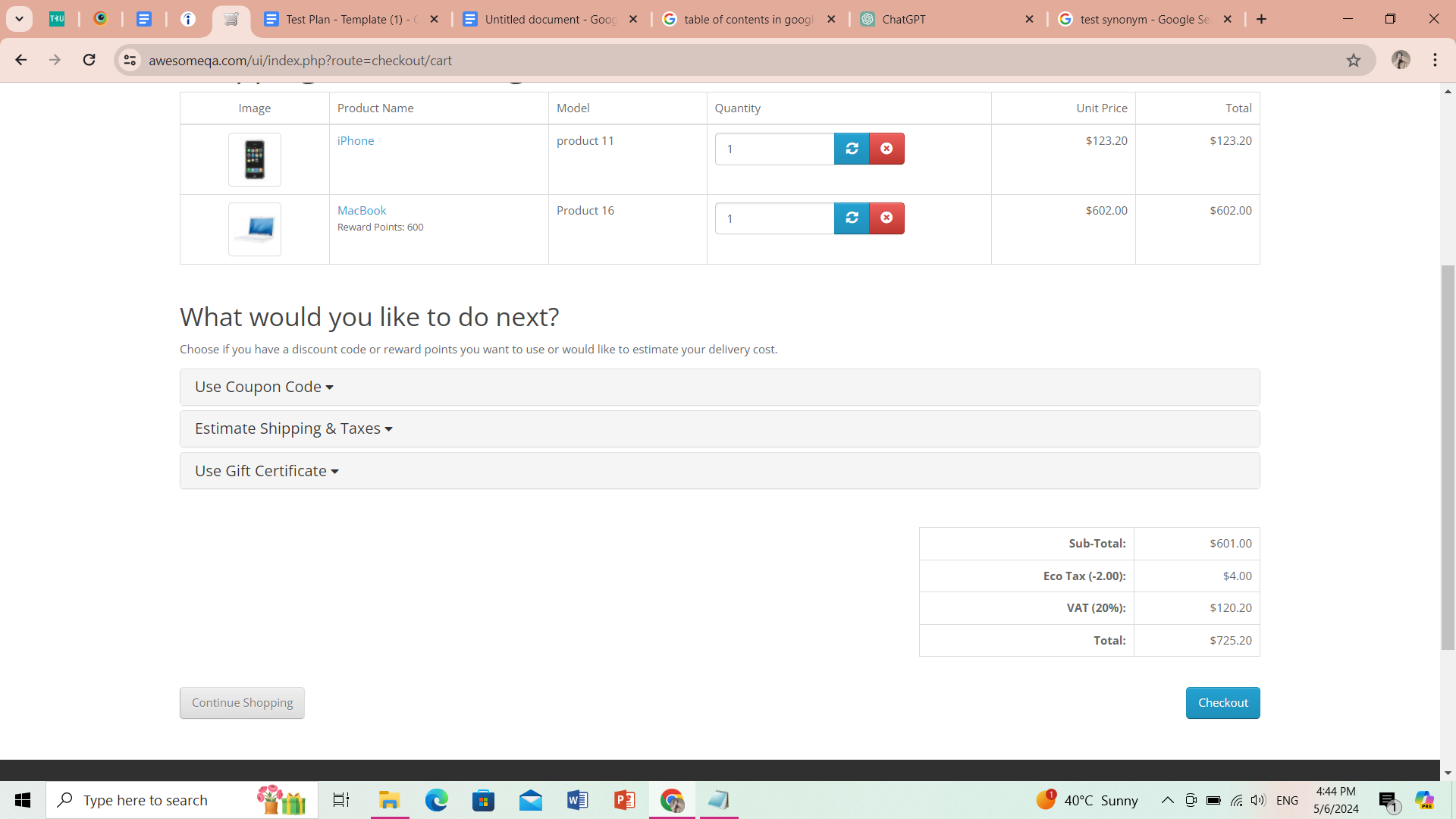
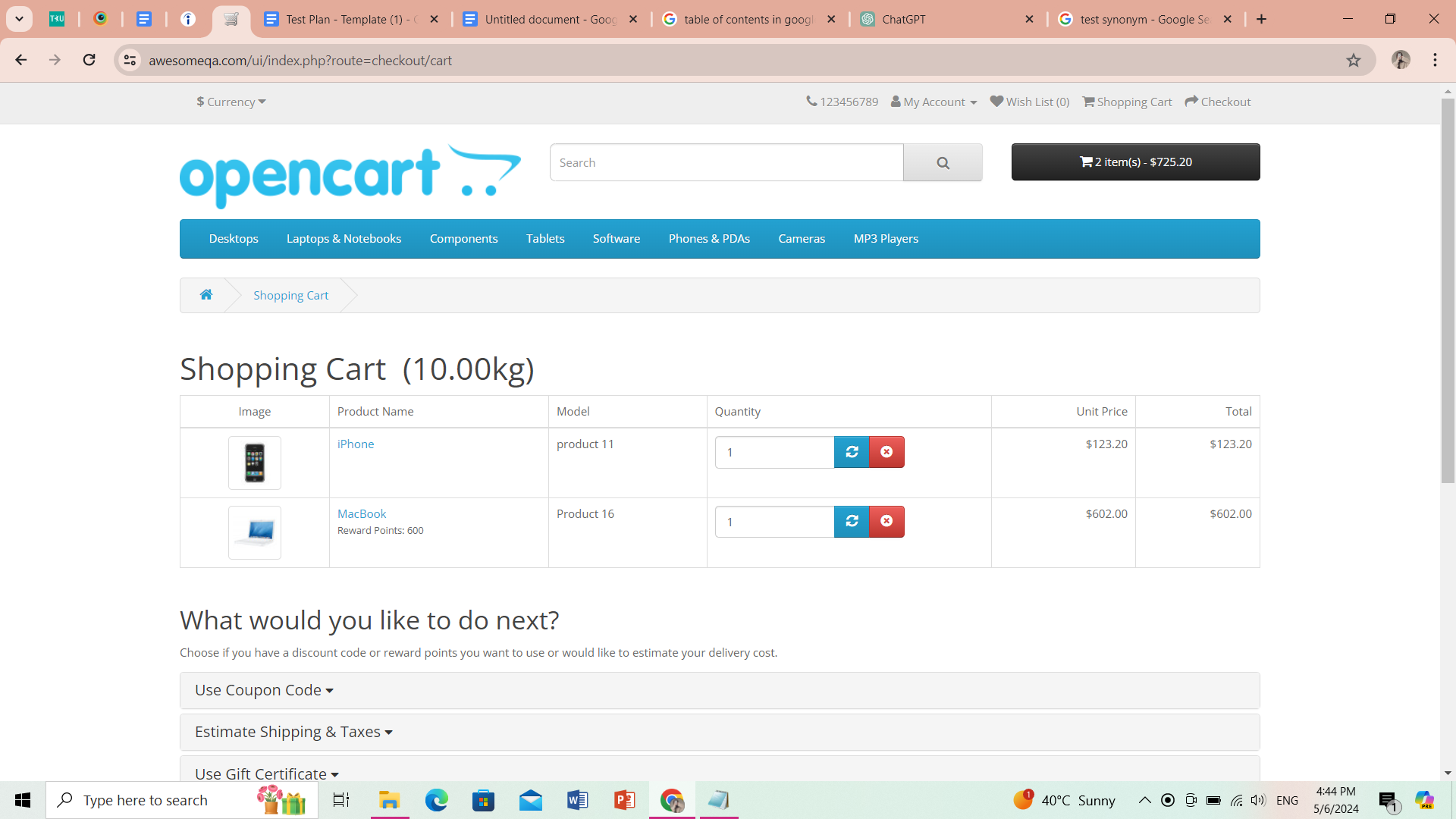
### **Registration**:

****

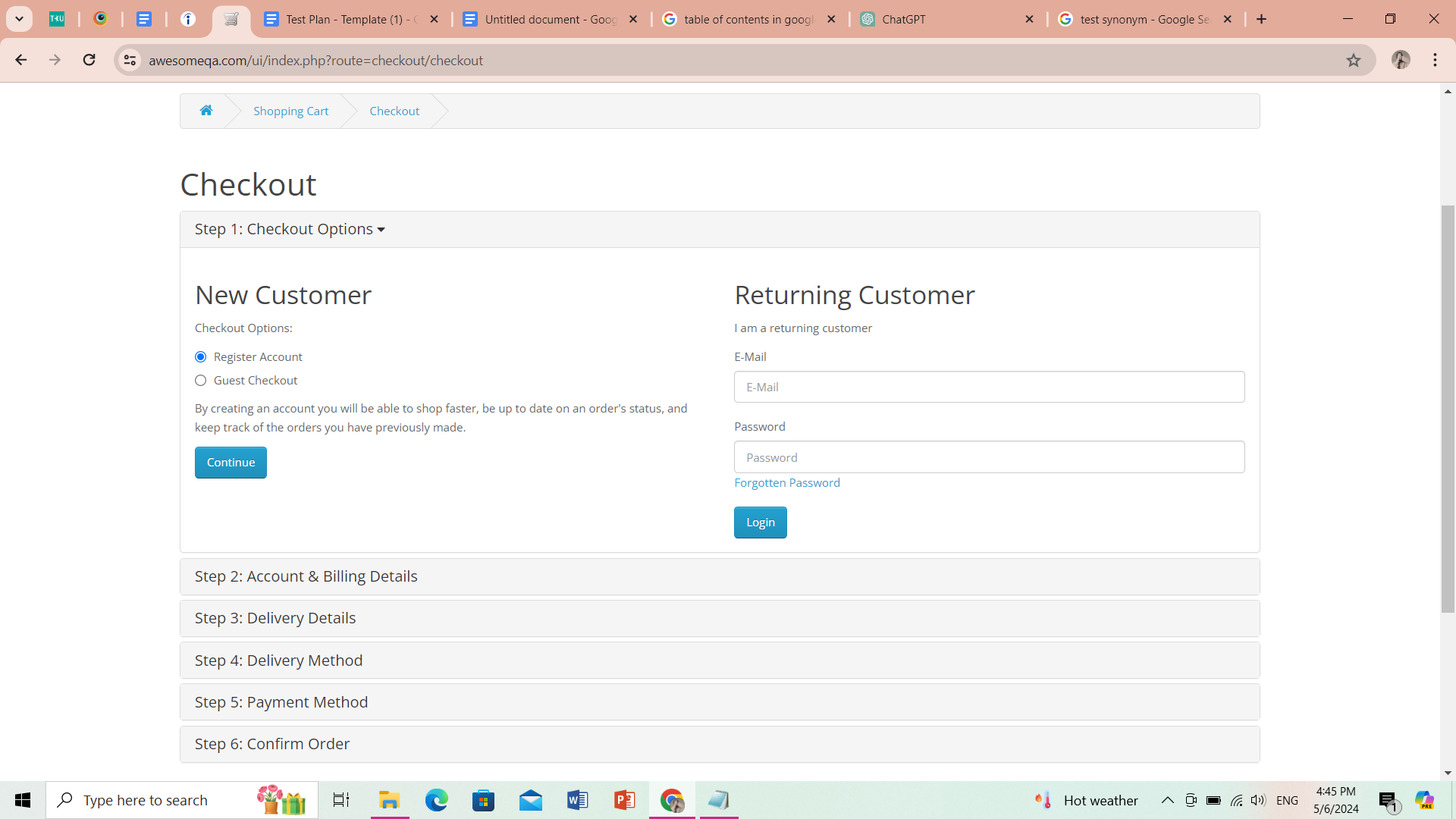
### **Add to Cart:**

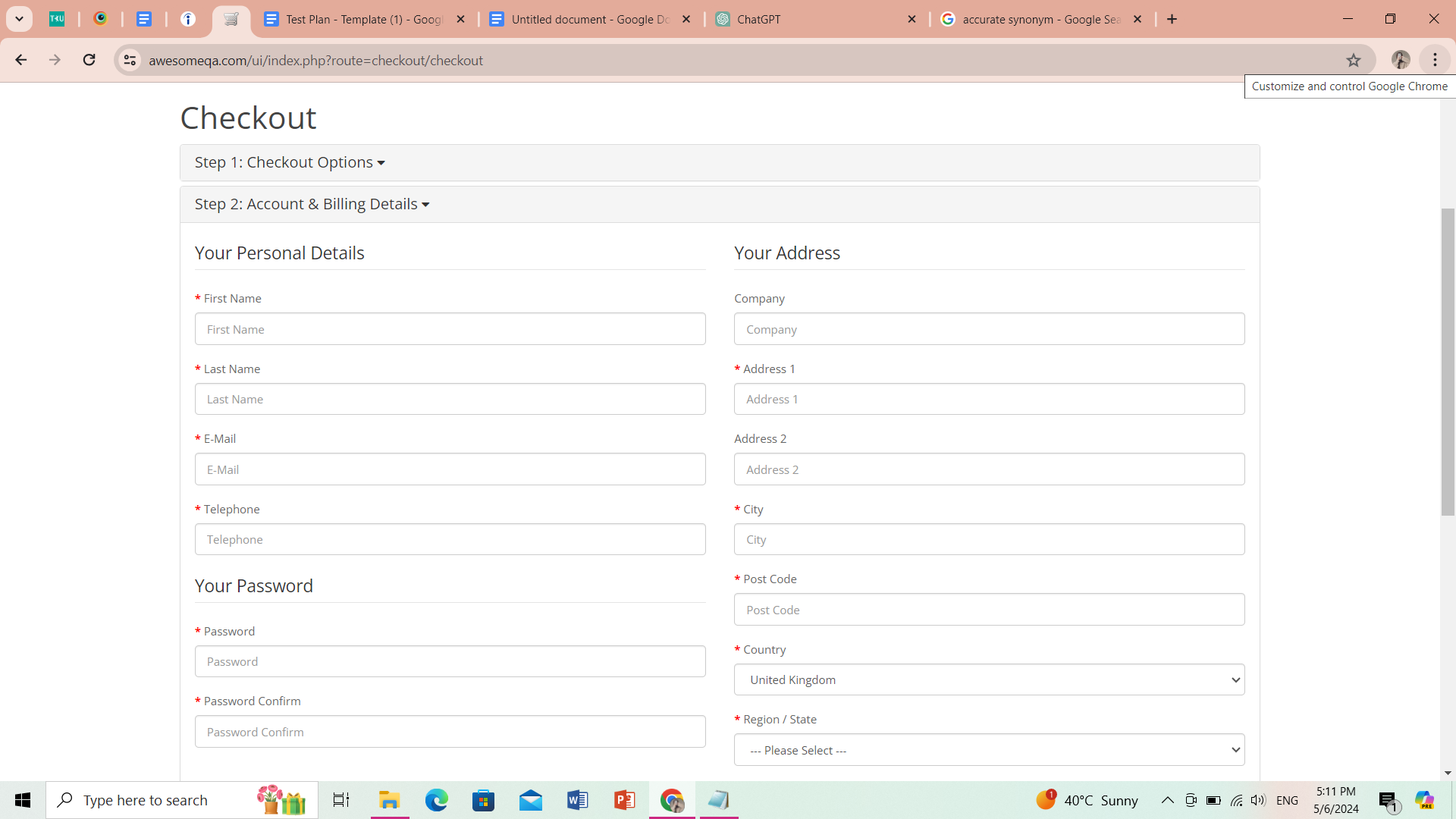
****

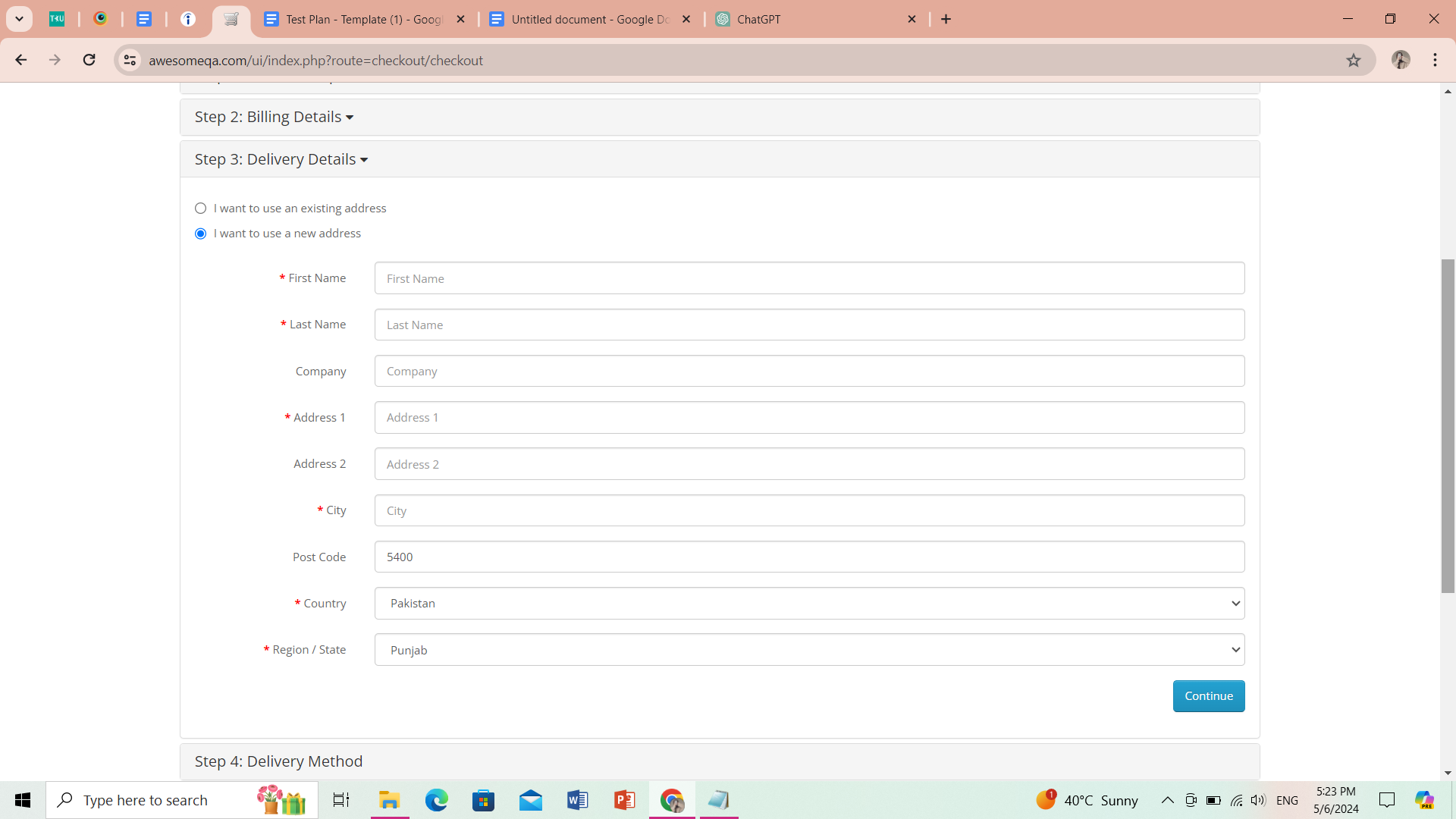
****

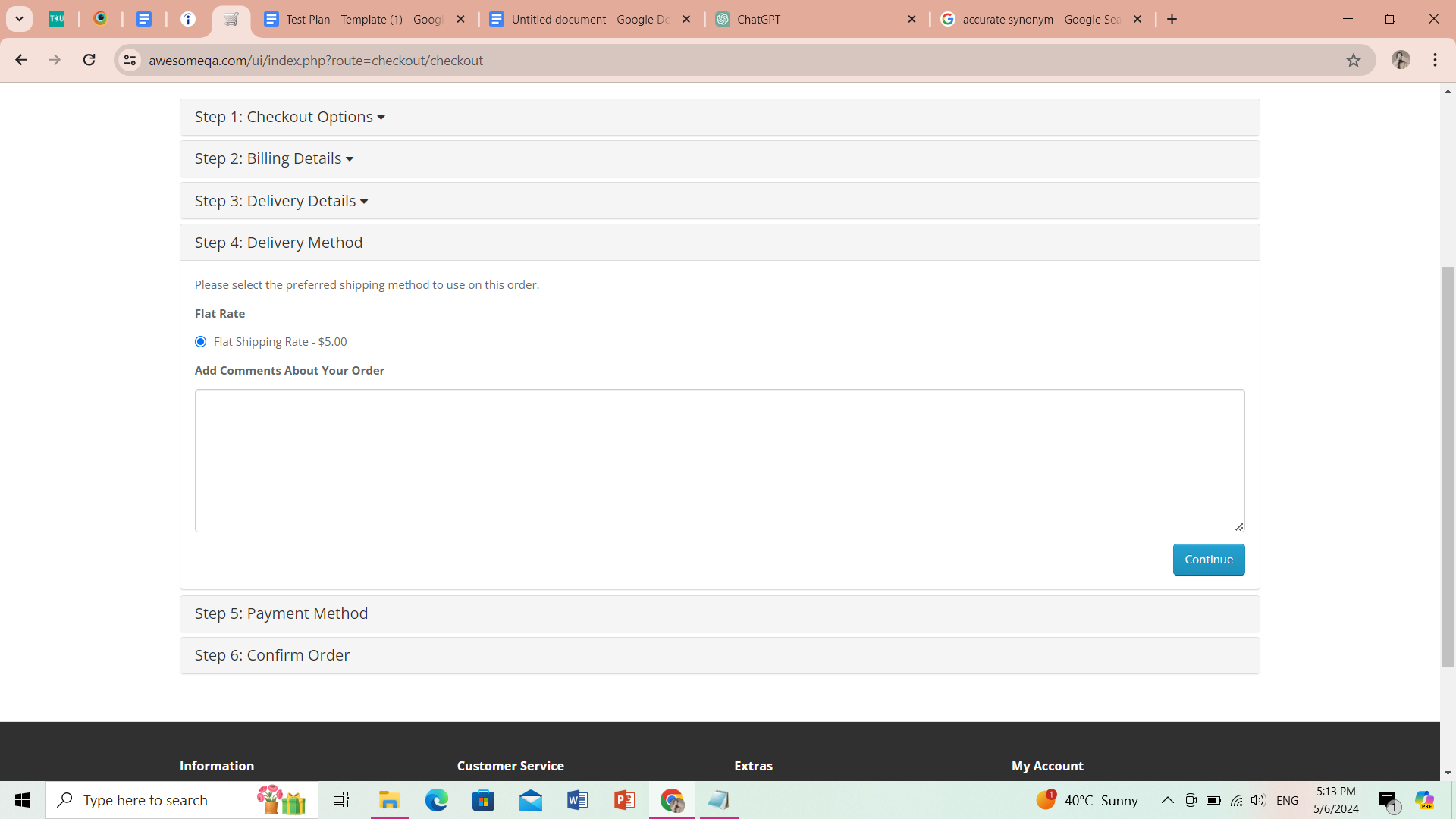
****

### **Check out/Payment Process:**

****







### **Exclusion**:

1. All options on the navbar
2. My account
3. Wishlist
4. Order history
5. Returns
6. Newsletter

### **Testing Environment:**

* The **operating systems** and versions that will be used for testing, such as Windows 10.
* The browsers and versions that will be tested, such as Google Chrome
* The device types and screen sizes that will be used for testing, such as desktop computers, laptops, tablets, and smartphones.
* The network connectivity and bandwidth that will be available for testing, such as Wi-Fi, cellular, or wired connections.
* The hardware and software requirements for running the test cases, such as a specific processor, memory, or storage capacity.
* The security protocols and authentication methods that will be used to access the test environment, such as passwords.

### **Test Strategy**

The **first step** is to create test scenarios and test cases for the features in

Scope.

While developing test cases, we'll use a number of test design techniques.

* Equivalence Class Partition
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing
* Use Case Testing

We also use our expertise in creating Test Cases by applying the below:

* Error Guessing
* Exploratory Testing

• We prioritize the Test Cases

**Step 2**: Our testing procedure when we receive a request for testing:

* First, we'll conduct smoke testing to see if the various and

important functionalities of the application are working.

* We reject the build, if the Smoke Testing fails and will wait for the stable

build before performing in depth testing of the application functionalities.

* Once we receive a stable build, which passes Smoke Testing, we perform

in depth testing using the Test Cases created.

* Multiple Test Resources will be testing the same Application on Multiple

Supported Environments simultaneously.

We then report the bugs in bug tracking tool and send dev. management

the defect found on that day in a status end of the day email.

As part of the Testing, we will perform the below types of Testing:

o Smoke Testing and Sanity Testing

o Regression Testing and Retesting

o Usability Testing, Functionality & UI Testing

• We repeat Test Cycles until we get the quality product.

**Step3** – We will follow the below best practices to make our Testing better:

• **Context Driven Testing** – We will be performing Testing as per the context

of the given application.

• **Shift Left Testing** – We will start testing from the beginning stages of the

development itself, instead of waiting for the stable build.

• **Exploratory Testing** – Using our expertise we will perform Exploratory

Testing, apart from the normal execution of the Test cases.

• **End to End Flow Testing** – We will test the end-to-end scenario which

involve multiple functionalities to simulate the end user flows.

### **Test Schedule**

Following is the test schedule planned for the project –

Task Time Duration

| **Task** | **Dates** |
| --- | --- |
| ▪ Creating Test Plan |  |
| ▪ Test Case Creation |  |
| ▪ Test Case Execution |  |
| ▪ Summary Reports Submission Date |  |

**2 Sprints to Test the Application**

### **Test Deliverables.**



### **Entry and Exit Criteria**

The below are the entry and exit criteria for every phase of Software Testing Life

Cycle:

Requirement Analysis

#### **Entry Criteria:**

• Once the testing team receives the Requirements Documents or details

about the Project

#### **Exit Criteria:**

• List of Requirements are explored and understood by the Testing team

• Doubts are cleared

### **Test Execution**

#### **Entry Criteria:**

• Test Scenarios and Test Cases Documents are signed-off by the Client

• Application is ready for Testing

#### **Exit Criteria:**

• Test Case Reports, Defect Reports are ready

### **Test Closure**

#### **Entry Criteria:**

• Test Case Reports, Defect Reports are ready

#### **Exit Criteria:**

• Test Summary Reports

#### **Tools**

The following are the list of Tools we will be using in this Project:

• JIRA Bug Tracking Tool

• Snagit Screenshot Tool

• Word and Excel documents

#### **Risks and Mitigations**

The following are the list of risks possible and the ways to mitigate them:

Risk: Non-Availability of a Resource

Mitigation: Backup Resource Planning

Risk: Build URL is not working

Mitigation: Resources will work on other tasks

Risk: Less time for Testing

Mitigation: Ramp up the resources based on the Client needs dynamically

#### **Approvals**

Team will send different types of documents for Client Approval like below:

• Test Plan

• Test Scenarios

• Test Cases

• Reports

Testing will only continue to the next steps once these approvals are done