

# American International University-Bangladesh (AIUB)

Department of Computer Science Faculty of Science &Technology (FST) Spring 21-22

Section: B

Software Quality Assurance and Testing

Electronic Devices Exchange, Specification & Quality Checking System (QUALEX)

## A Report submitted By

|  |  |  |
| --- | --- | --- |
| SN | Student Name | Student ID |
| 1 | Barua, Naresh | 19-40332-1 |
| 2 | Rahman, Md. Saqib | 19-40399-1 |
| 3 | Kanta, Maria Akter | 19-40343-1 |
| 4 | Saha, Swarnab | 19-40398-1 |

Under the supervision of

Abhijit Bhowmik

Associate Professor & Special Assistant [OSA], Computer Science

Software Test Plan

for

QUALEX

## Version 1.0 approved

Prepared by Swarnab, Kanta, Saqib, Naresh

American International University-Bangladesh

<date created>

Table of Contents

[Revision History 3](#_bookmark0)

1. [TEST PLAN IDENTIFIER: RS-MTP01.3 4](#_bookmark1)
2. [REFERENCES 4](#_bookmark2)
3. [INTRODUCTION 4](#_bookmark3)

[Background to the Problem 4](#_bookmark4)

[Solution to the Problem 5](#_bookmark5)

1. [REQUEIREMNT SPECIFICATION 5](#_bookmark6)
   1. [System Features 5](#_bookmark7)
   2. [System Quality Attributes 6](#_bookmark8)
   3. [System Interface 7](#_bookmark9)
   4. [Project Requirements 12](#_bookmark10)
2. [FEATURES NOT TO BE TESTED 13](#_bookmark11)
3. [TESTING APPROACH 13](#_bookmark12)
   1. [Testing Levels 13](#_bookmark13)
   2. [Test Tools 14](#_bookmark14)
   3. [Meetings 21](#_bookmark15)
4. [TEST CASES/TEST ITEMS 21](#_bookmark16)
5. [ITEM PASS/FAIL CRITERIA 29](#_bookmark17)
6. [TEST DELIVERABLES 29](#_bookmark18)
7. [STAFFING AND TRAINING NEEDS 29](#_bookmark19)
8. [RESPONSIBILITIES 30](#_bookmark20)
9. [TESTING SCHEDULE 31](#_bookmark21)
10. [PLANNING RISKS AND CONTINGENCIES 31](#_bookmark22)
11. [APROVALS 32](#_bookmark23)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Updated by | Update Comments |
| 0.1 | 2022.04.19 | Md. Saqib Rahman | First Draft |
| 0.2 | 2022.04.21 | Maria Akter Kanta | Second Draft |
| 0.3 | 2022.04.23 | Naresh Barua | Third Draft |
| 0.4 | 2022.04.24 | Swarnab Saha | Final Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## TEST PLAN IDENTIFIER:RS-MTP01.3

## REFERENCES

* + Software Requirement Specification (SRS document): <https://github.com/SakibRahman1177/Software-Requirement-Specification>
  + Testing Tool - selenium: <https://www.selenium.dev/>
  + Jira tools for test scheduling: <https://www.atlassian.com/software/jira>
  + Web browser used for testing Module: Chrome
  + Similar Software for Test plan: <https://swap.com.bd/>
  + IDE used for Testing Module – eclipse IDE: <https://www.eclipse.org/downloads/>
  + Template used for Test Plan:

‘Software Quality and Testing’ – Project Template

## INTRODUCTION Background to the Problem

Many developers are currently implementing many global or local electronic device quality checking and exchange software all over the world. But there is a very inefficient software in Bangladesh to solve the problem for any confusion of budget related issue of a customer and also ensures product quality which a customer can evaluate themselves and know the price of each product. Besides, there can be lots of issues in globally developed similar software like gsmarena.com, mobiledokan.com and some social platforms. In such software, there can be a lack of updating price issue or product quality issue and also, they were work on global customer demand, not a specific country. So, we have decided to make proper effective and integrated software by which we can serve a country a better-quality checking and exchange system and make a test plan for it.

Our system is about online electronic devices selling, exchanging, specifications, & quality checking software. We can use this software through an application or a website. In our system, a user can perform some specific tasks like "viewing the content of our system portal, can search for any product using the search bar, can get any notifications over the application, can sell their devices in the SELL section, also can purchase any product from BUY section, can complete their transactions or billing method in our payment section & lastly they can give ratings or feedback in our Reviews area.

## Solution to the Problem

Overview the 'Online Electronic devices Exchange, Specifications & Quality checking system (QUALEX)' services strive to provide solutions to develop and transfer easy and efficient way in the digital age and to help reduces the human pressure and time. QUALEX is a website or application written for all operating systems, designed to help users maintain and organize shop virtually. This software is easy to use for both beginners and advanced users. It features a familiar and well thought out, an attractive user interface, combined with strong searching Insertion and reporting capabilities.

The main purpose of building this software are:

* To provide a unique user (customer) experience,
* To develop business relations,
* To reduce management costs,
* For boosting the efficiency of services,
* To provide quality online customer service.

Therefore, there is a similar software solution related to this problem area such as [SWAP.](https://swap.com.bd/) It gives some specific solutions like,

* Arranges personalized orders for customers and delivers at their doorstep.
* Offers a seller to sell unneeded items at any convenient location and time, and accepts various payment methods either digitally or by cash.
* Helps organizations to get rid of old products' selling procedure by saving time for organization's mundane works.
* organizes Activation Points at work places which help corporate employees to sell or buy directly and avail exchange offers, gift cards, and so on

## REQUEIREMNT SPECIFICATION

## System Features

* + 1. System Login
       1. The software shall allow users to log in with their given username and password
       2. If the username and password have been inserted wrong more than three times, the system will generate a random verification code to retry the login.
       3. If the number of login attempts exceed its limit (5 times), the system shall block the user account login for one hour.

Priority Level: Medium

Precondition: user has valid user id and password

* + 1. System Registration
       1. The software will allow users to register with a valid email or phone number and password.
       2. If the email/number is used to do the registration, an OTP will be sent to the user to verify the process.
       3. If the email/number has been used earlier, the system won't allow users to register with it.

Priority Level: High

Precondition: The user must use a valid and new email or phone number

* + 1. Searching
       1. The user can browse the system by using the search bar
       2. The users can also find their desired buy sell swap options on our home page.

Priority Level: High

Precondition: user must input a valid product name or id.

* + 1. Buy, sell and exchange
       1. Users can select a product that they want to buy, sell or exchange from the search
       2. Users can see details about the product in the description and ensure the quality
       3. User can add the product to the cart
       4. Users can see the reviews from previous buyers and ask questions also.

Priority Level: Medium

Precondition: Users have to find out the product they want to buy, sell, or exchange.

* + 1. Payment method
       1. Users can make the payment after finalizing their product
       2. User can select their flexible payment option for payment
       3. Users can also pay through EMI
       4. Users can enter coupon codes to get discount

Priority Level: High

Precondition: Users must select the product, and payment requirement data must be verified.

## System Quality Attributes

* **Usability**: A user shall be able to buy a mobile selected from our website catalog in an average of two and a maximum of four minutes.
* **Efficiency**: the system shall use the proper amount of ram and processor and deliver excellent performance.
* **Security**: Users will supply their banking details if they make any transition through their bank account. So, if any hacker does any attack, then the details of the users will be compromised.
* **Reliability**: The system ensures that the software is dependable, that it fulfils the purpose for which it was created for a certain period in a particular environment and that it can provide fault-free operation. So, this type of attribute should work correctly at that moment under the system condition.
* **Correctness**: The system must be bug free. So, the users don’t face any problem during use of the system. We must make sure our modules are working fine and supplying the expected outcome.
* **Flexibility**: The system should be flexible enough so that in future we can do integration with other third-party systems.
* **Scalability**: the system will be made as for future so that in need we can scale our system as its need.

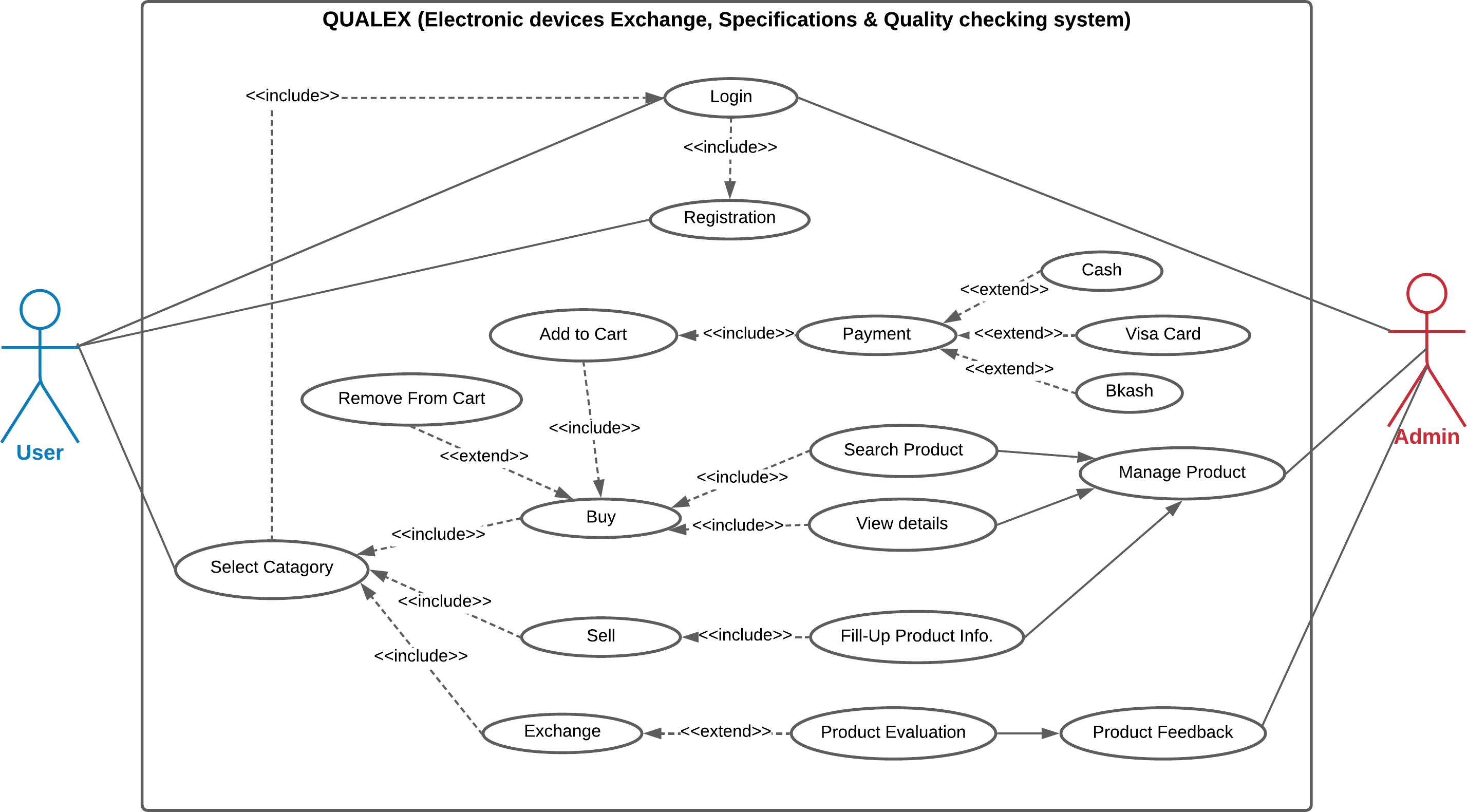
## System Interface

### MockUp for system design

|  |  |
| --- | --- |
| *Figure 1: System Login interface* | *Figure 2: System Dashboard interface* |

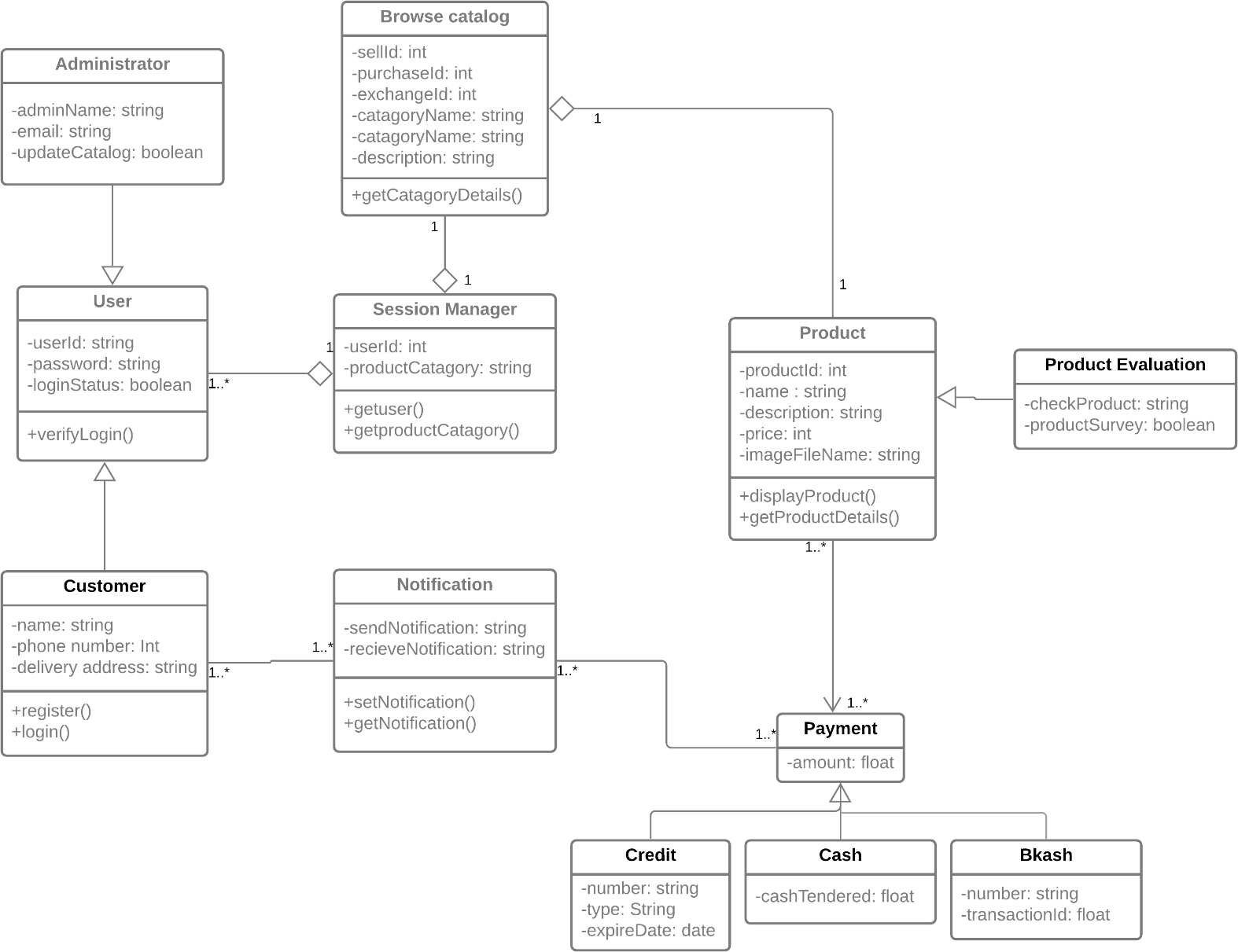
|  |  |
| --- | --- |
| *Figure 3: SELL Module* | *Figure 4: BUY Module* |

**Use Case Diagram**



*Figure 5: Use Case diagram of Electronic devices Exchange, Specifications & Quality checking system*

### Class Diagram:



*Figure 6: Class diagram of electronic devices Exchange, Specifications & Quality checking system*

## Project Requirements

Our Project model is an Organic model.

|  |  |  |  |
| --- | --- | --- | --- |
| **Software Project type** | **Coefficient**  **<Effort factor>** | **P**  **(Project Complexity)** | **T**  **(SLOC-dependent coefficient)** |
| Organic | 2.4 | 1.05 | 0.38 |

Here,

SLOC (Source lines of code) = 9000 SLOC/1000

= 9k SLOC

Now,

Effort =PM= Coefficient <Effort Factor> \*(SLOC/1000) ^P

=2.4\*(9) ^1.05

=24 person

Development time = DM = 2.50\* (PM) ^T

= 2.50\* (24) ^0.38

= 8 months

Required number of people = ST = PM/DM

= 24/8

= 3 members

For a person,

Working hours per day = 8 hours Salary per month = 60,000 BDT

In a month working days = 22 days

So, working hours in a month = 22\*8 = 176 hours

Now, per hour salary in a month = 60000/176 = 340 BDT In 8 months working hours= 176\*8 = 1408 hours

For the project = 1408\*340 = 4,78,720 = 4,80,000 BDT (approx.)

Requirement analysis = 22 days \* 8 hours = 176 hours Expense for requirement analysis = 176\*300 = 52800 BDT Expense for travel = 10000 BDT

Expense for office rent = 50,000 BDT Expense for electricity & gas bill = 25000 BDT

Expense for training and hardware = 1,00,000 BDT

Total hours for 6-month maintenance= 6\*18 = 108 hours Expense for Maintenance = 108\*1000 = 1,08,000 BDT

Total cost = 480000+52800+10000+50000+25000+100000+108000

=8,25,000

Bill = Total cost + 20%

= 825000 + 165000

= 9,90,000 BDT

=10,00,000 BDT(approx.)

## FEATURES NOT TO BE TESTED

This is a list of things that should not be tested, both from the user's perspective and from the perspective of configuration management and version control. This is a user's perspective on the functions, not a technical description of the software.

So, the following is a list of the areas that will not be specifically addressed. All testing in these areas will be indirect because of other testing efforts. For example:

* + Microsoft Excel / Spreadsheet analysis on bought product,
  + Out of scope features (Networks & security),
  + Negligible risk features (Wired Connection, Online bank payment),
  + Future functionality (retail ecommerce sales tracking).

## TESTING APPROACH

## Testing Levels

For the software testing there will consist of Unit testing, System/Integration testing (Combined) and Acceptance testing.

**UNIT TESTING**: Unit testing is the process of testing each unit or component of a software application separately. It's the first step in the functional testing process. The goal of unit testing is to ensure that unit components work as expected. A unit is a single testable item that is tested during the application software development process. The goal of unit testing is to ensure that isolated code is exact. Unit testing is often done by developers using the white box testing technique. When the program is complete and sent to the Test Engineer, he or she will begin unit testing by inspecting each component of the module one by one.

* + - It helps testers and developers in understanding the code's foundation.
    - It helps in the documentation process.
    - Unit testing identifies and corrects issues early in the development process.
    - By relocating code and test cases, it aids with code reuse.

There are also some techniques used by the tester. Data flow testing, Control Flow testing, Branch Coverage testing, Statement Coverage testing and Decision Coverage testing.

**SYSTEM/INTEGRATION TESTING**: After unit testing, integration testing is the next step in the software testing process. Units or individual components of the software are tested in a group during this testing. The integration testing level focuses on exposing faults that occur during the interaction of integrated components or units. After integration, testing system testing is done.

System testing is putting a fully integrated software system to the test. The program is created in modules, which are then combined with other software and hardware to form a full computer system. System testing is a collection of several types of tests designed to exercise and assess the overall functionality of an integrated software computer system in comparison to its requirements.

**ACCEPTANCE TESTING:** Acceptance testing is a technique for deciding whether a software system meets its requirements specifications. The major goal of this test is to determine whether the system complies with business needs and if it meets the requirements for distribution to end users. Acceptance testing can take many different forms. User acceptance testing (UAT), Business acceptance testing (BAT), Alpha testing and beta testing.

## Test Tools

In this project, we have used selenium web driver tool for testing any software. Basically we used this tool for just testing any random website/application, by which we can practice the tool and shows any outcome. So, before submitting the result of our practice testing, we would like to describe about the steps we’ve done for -

### Step 1:

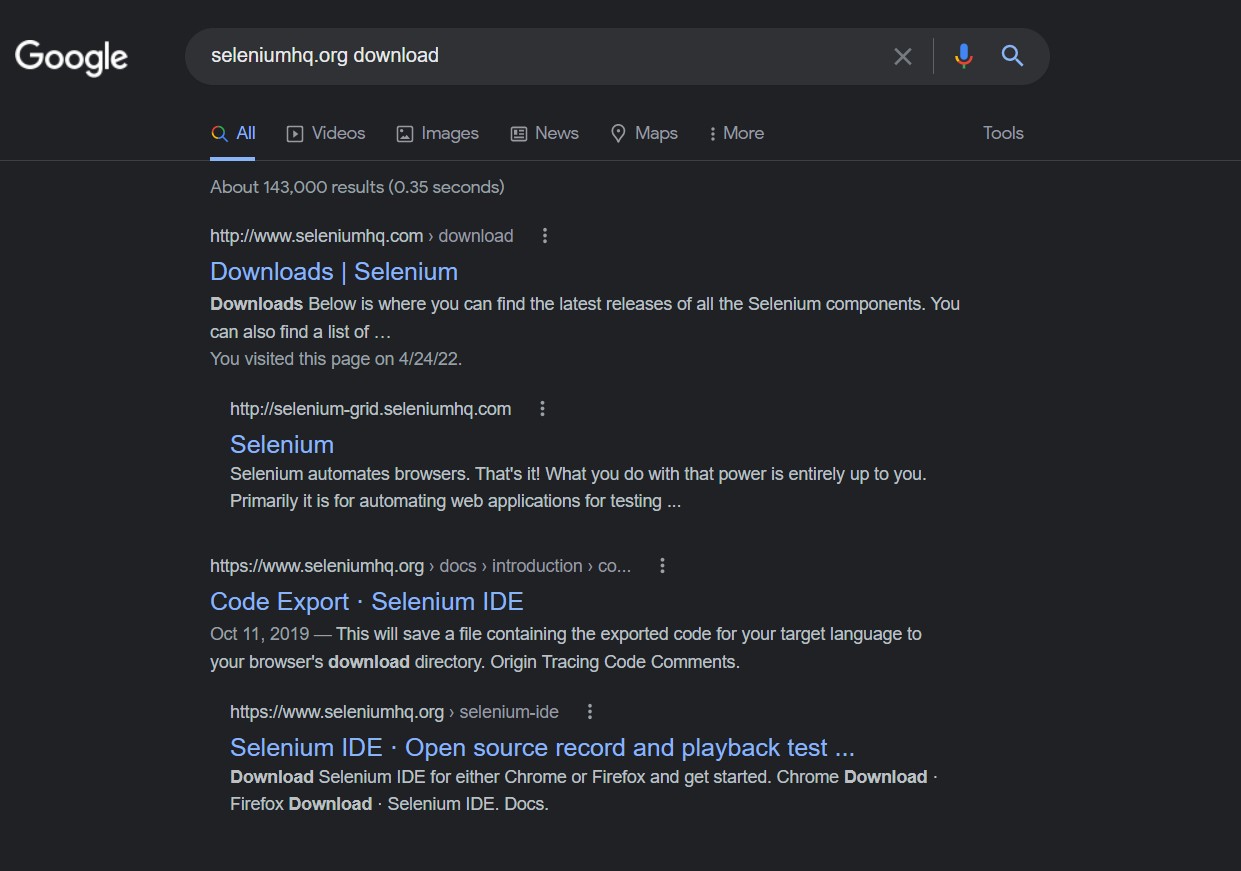
* + - Installing Java on our computer
    - Download and install the Java Software Development Kit (JDK)

### Step 2:

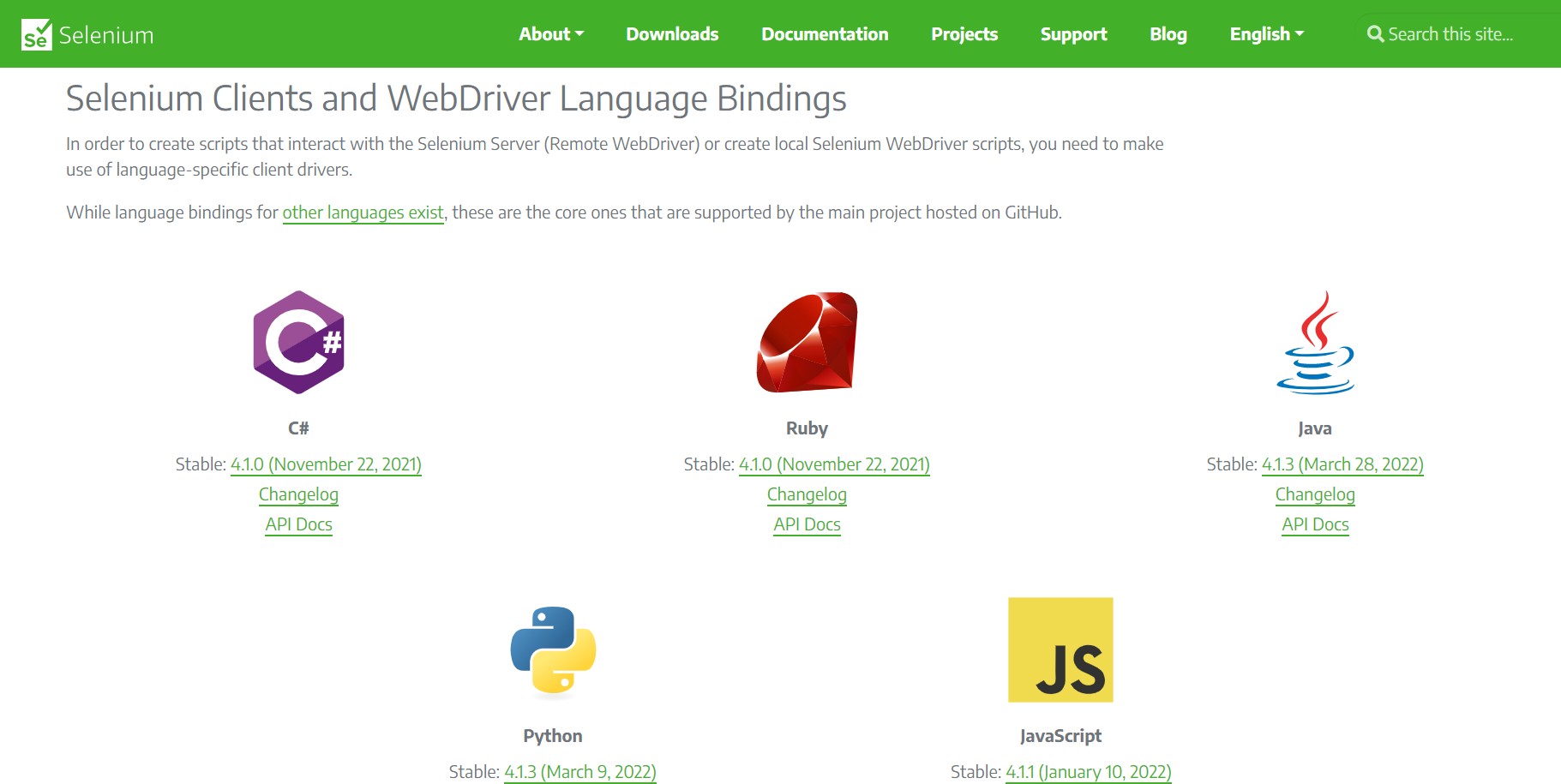
* + - Install Eclipse IDE
    - Download latest version of “Eclipse IDE for Java Developers” . Be sure to choose correctly between Windows 32 Bit and 64 Bit versions.

### Step 3:

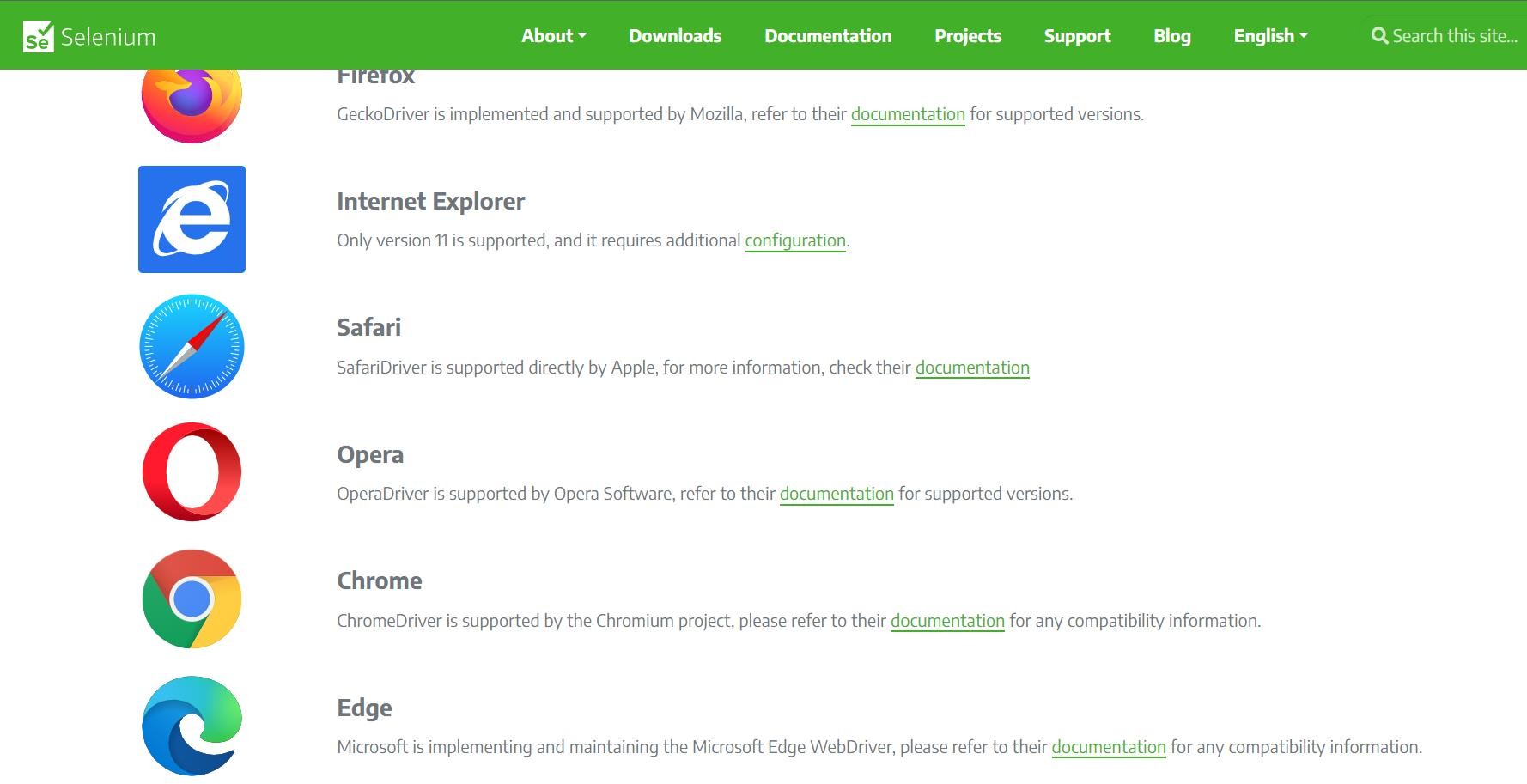
* + - Download the Selenium Java Client Driver
    - You can download Selenium Webdriver for Java Client Driver. You will find client drivers for other languages there, but only choose the one for Java.



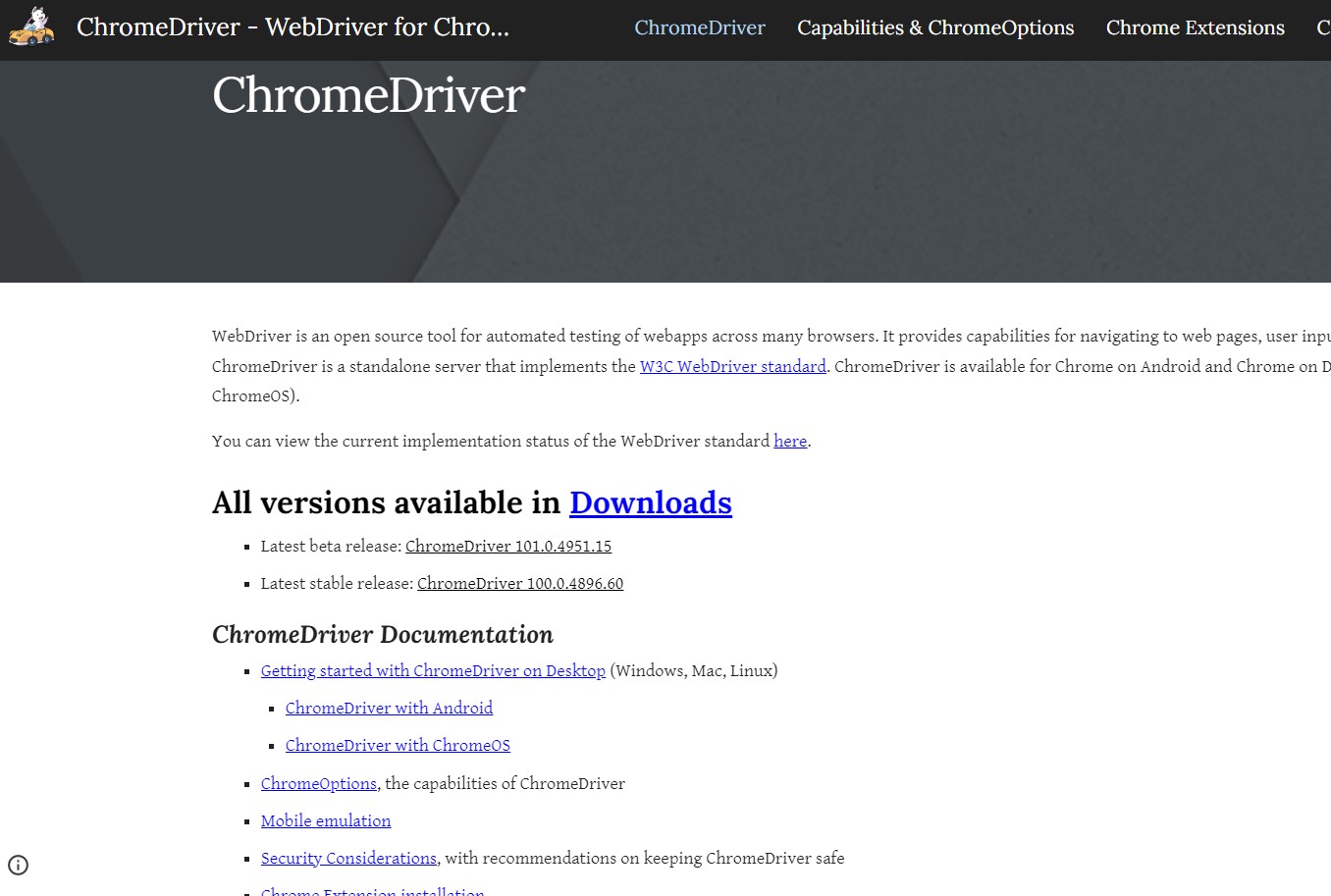
*Figure 7: seleniumhq.org download*



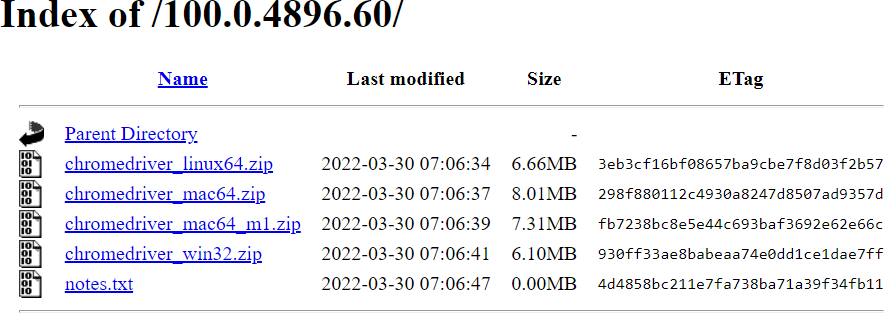
*Figure 8: Selenium clients and webdriver language server - JAVA*



*Figure 9: selecting Chrome browser*



*Figure 10: Latest sable release version of ChromeDriver*



### Step 4:

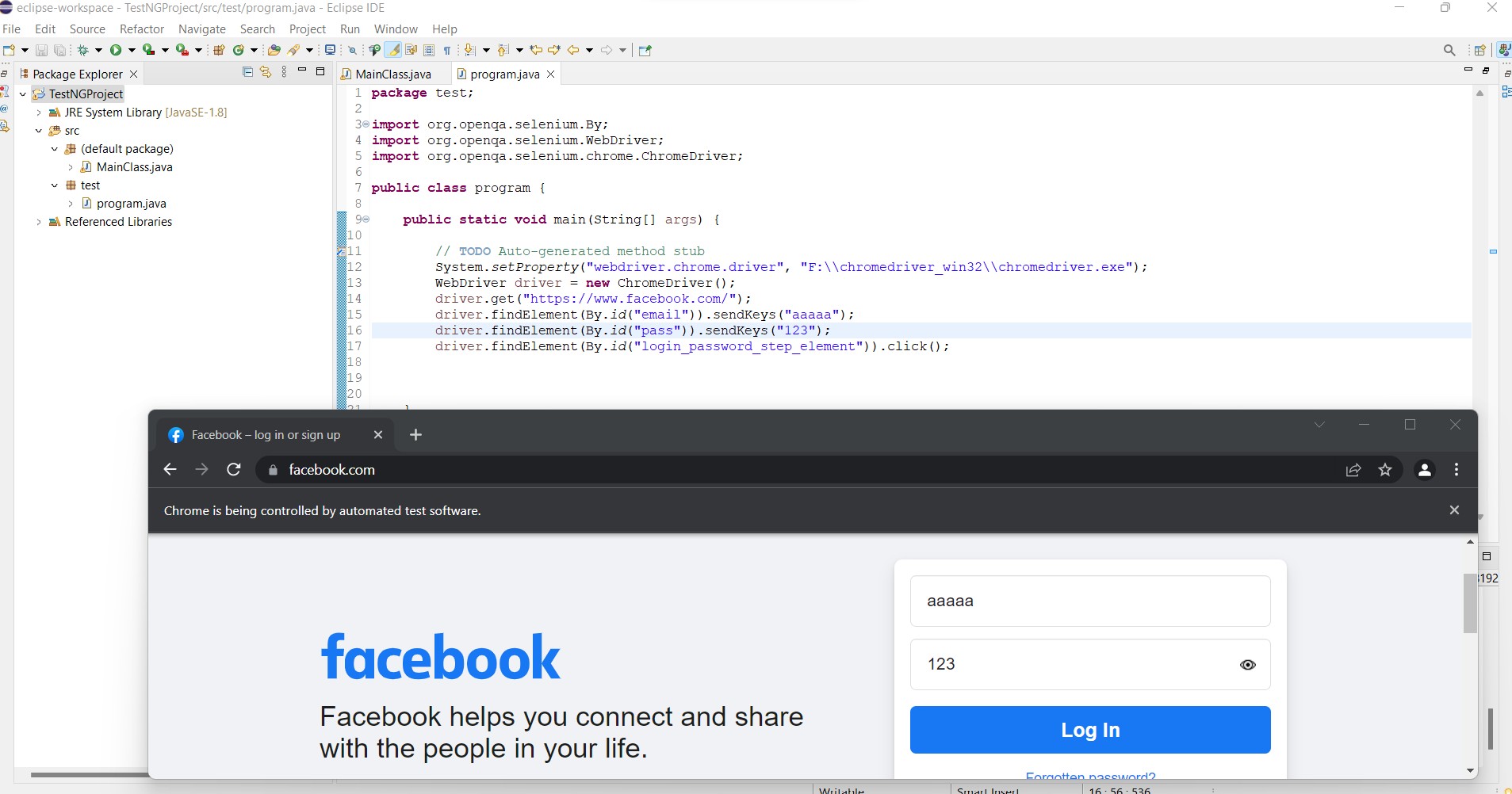
*Figure 11: Chromedriver\_win32.zip*

* + - Configure Eclipse IDE with WebDriverNow selenium WebDriver’s into Java Build PathIn this step,Right-click on “newproject” and select Properties.
    - On the Properties dialog, click on “Java Build Path”.
    - Click on the Libraries tab, and then
    - Click on “Add External JARs..”
    - After selecting jar files, click on OK button.Select all files inside the lib folder.
    - Select files outside lib folder
    - Once done, click “Apply and Close” button

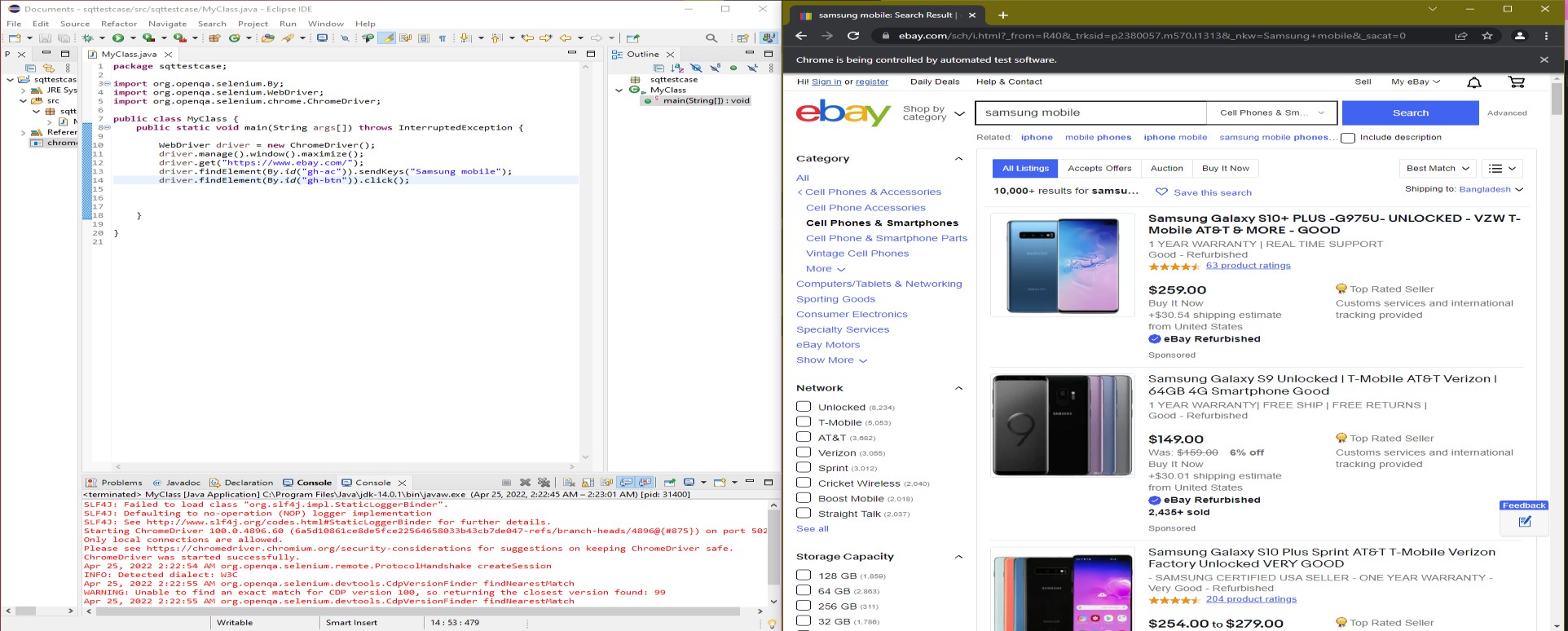
Finally, click OK and we are done importing Selenium libraries into our project.

Selenium is an open source umbrella project for a range of tools and libraries aimed at supporting browser automation. It provides a playback tool for authoring functional tests without the need to learn a test scripting language.

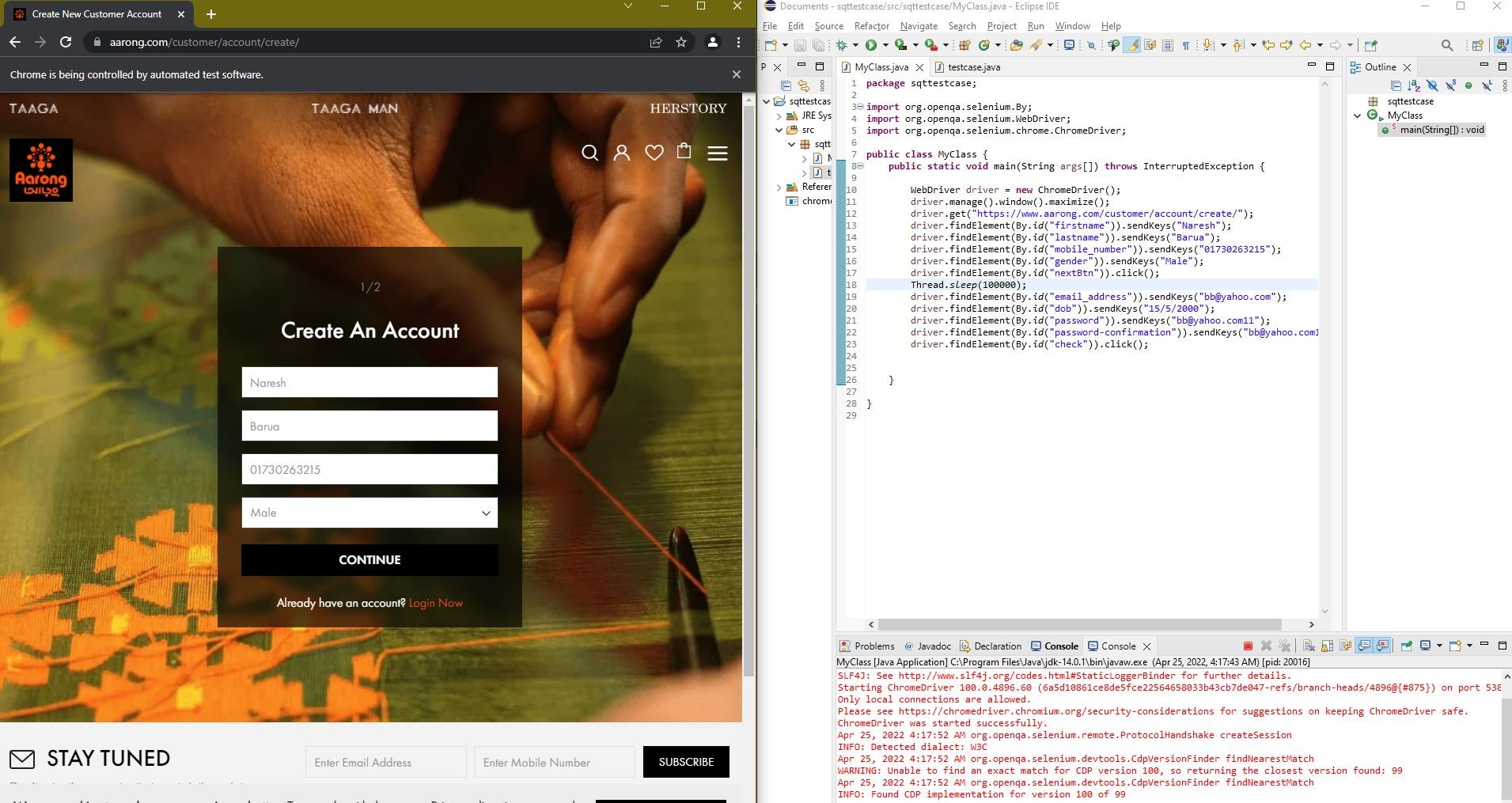
### Sample Screen shots from our practice selenium testing:



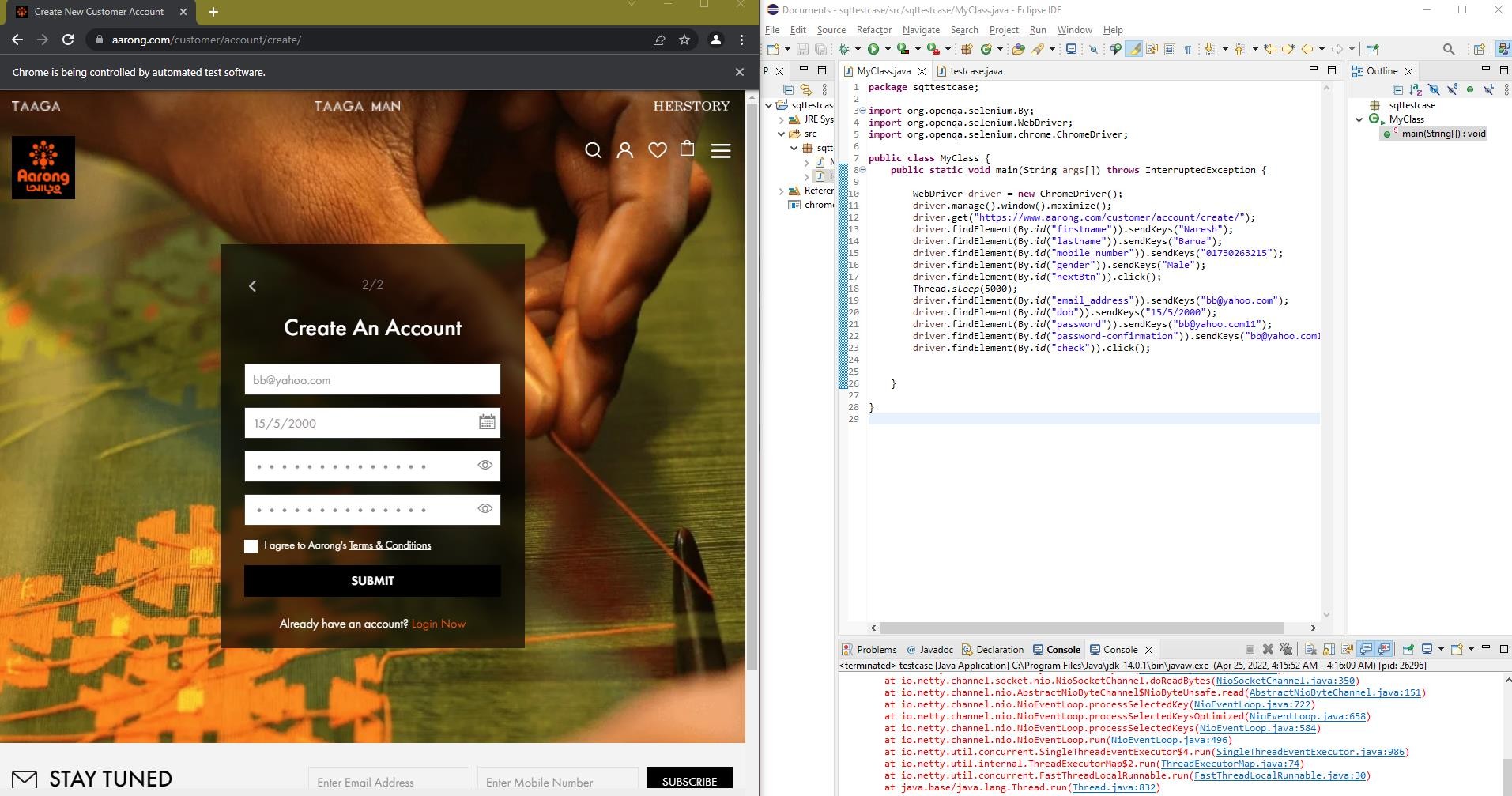
*Figure 12: Login interface test using selenium*



*Figure 13: Search module testing using selenium*



*Figure 14: Registration interface test using selenium too part-1*



*Figure 15: Registration interface test using selenium too part-2*

## Meetings

Once every two weeks, the test team will meet to analyze progress and identify mistake trends and concerns as soon as feasible. Once every two weeks, the test team leader will meet with development and the project manager. These two sessions will take place in separate weeks. In case of an emergency, more meetings might be convened.

## TEST CASES/TEST ITEMS

### Log In Test Case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Kanta | | |
| Test Case ID: FR\_1 | | | Test Designed date: 21.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Kanta | | |
| Module Name: Login Session | | | Test Execution date: 21.04.2022 | | |
| Test Title: Verify login with valid username and password | | |  | | |
| Description: Test website login page | | |  | | |
| Precondition (If any): User must have valid username and password | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Enter username 3. Enter password 4. Click submit | Username: kanta Password: 321123 | User should login into the application | | As expected, | Pass |
| Post Condition: User is confirmed with database and successfully login to account. The account session details are logged into the database. | | | | | |

1. **Registration Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Swarnab | | |
| Test Case ID: FR\_2 | | | Test Designed date: 21.04.2022 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Swarnab | | |
| Module Name: Registration Session. | | | Test Execution date: 21.04.2022 | | |
| Test Title: Verify registration with valid email and password | | |  | | |
| Description: Test website registration page. | | |  | | |
| Precondition (If any): User must do registration with valid email and password. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website. 2. Enter valid email address. 3. Enter password. | Username: Swarnab Password: 321321 | User successfully registered to the website. | | As expected, | Pass |
| Post Condition: User successfully created an account into the system and saved into the database. | | | | | |

1. **Search Bar Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Naresh | | |
| Test Case ID: FR\_3 | | | Test Designed date: 21.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Naresh | | |
| Module Name: Searching Session | | | Test Execution date: 21.04.2022 | | |
| Test Title: Verify the search is available based on the product name, brand, size etc. | | |  | | |
| Description: Test website search bar. | | |  | | |
| Precondition (If any): User must have to search with valid product name, brand. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Login to the website 3. Enter the valid product name in the search bar. 4. Click search | Product name:  Samsung s22 ultra 5G  Price: 1,23,000/- | User should see the product what they searched for. | | As expected, | Pass |
| Post Condition: User has to validated with database and successfully login to account. The search must come from database. | | | | | |

1. **Product model details Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Saqib | | |
| Test Case ID: FR\_4 | | | Test Designed date: 20.04.2022 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Saqib | | |
| Module Name: Product Model | | | Test Execution date: 20.04.2022 | | |
| Test Title: Verify if the specifications and price of the product is shown properly. | | |  | | |
| Description: Test website product details page | | |  | | |
| Precondition (If any): User must search for any product with valid product name or ID.. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Select SELL option 3. Select any categories 4. Choose a Model 5. Click Continue | Select Categories: Phone/smart- watch/tablet/laptop  Choose Model: iPhone 11 pro Max | User should visit the product information page. | | As expected, | Pass |
| Post Condition: User must search for a valid product name. The product details must be updated into the database. | | | | | |

1. **Payment Gateway Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Saqib | | |
| Test Case ID: FR\_5 | | | Test Designed date: 20.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Saqib | | |
| Module Name: Payment Gateway | | | Test Execution date: 20.04.2022 | | |
| Test Title: Verify that each of the payment options is selectable. | | |  | | |
| Description: Test website payment gateway page | | |  | | |
| Precondition (If any): The gateway must offer Address Verification System Protection. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Select product categories (Buy, Sell) 3. Add to cart 4. Select payment gateway 5. Click Continue | Select Categories: Phone/smart- watch/tablet/laptop  Choose Model: iPhone 11 pro Max | User should access all the payment gateway options. | | As expected, | Pass |
| Post Condition: User must have a valid gateway account. The purchased amount details are updated in the database. | | | | | |

1. **Product Purchase Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Kanta | | |
| Test Case ID: FR\_7 | | | Test Designed date: 20.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Kanta | | |
| Module Name: Product purchase. | | | Test Execution date: 20.04.2022 | | |
| Test Title: Verify that the user can purchase their selected products. | | |  | | |
| Description: Test website purchase page | | |  | | |
| Precondition (If any): The user must select the products they want to buy. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Login 3. Select products from the categories 4. Select buy now 5. Select payment method 6. Click Continue | Select Categories: Phone/smart- watch/tablet/laptop  Choose Model: iPhone 11 pro Max | User should be able to purchase the product. | | As expected, | Pass |
| Post Condition: User must have selected their product and have valid payment method. The purchased amount details will be updated in the database and show the delivery details. | | | | | |

1. **Product Sell Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Swarnab | | |
| Test Case ID: FR\_8 | | | Test Designed date: 20.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Swarnab | | |
| Module Name: Product sell. | | | Test Execution date: 20.04.2022 | | |
| Test Title: Verify that the user can sell their products. | | |  | | |
| Description: Test website sell page | | |  | | |
| Precondition (If any): The user must enlist their products in the website they want to sell. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Select sell option 3. Add information of their product 4. Select get value 5. Click Continue | Select Categories: laptop  Choose Model: Acer Aspire Nitro VN7-592G | User should be able to see how much value they will get if they sell their enlisted items. | | As expected, | Pass |
| Post Condition: User must have valid documents of their products. Their enlisted products details will be updated in the database. | | | | | |

1. **Product Exchange Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Qualex.com | | | Test Designed by: Naresh | | |
| Test Case ID: FR\_9 | | | Test Designed date: 20.04.2022 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Naresh | | |
| Module Name: Product Exchange | | | Test Execution date: 20.04.2022 | | |
| Test Title: Verify that the user can exchange their products | | |  | | |
| Description: Test website exchange products page | | |  | | |
| Precondition (If any): The user must enlist their products in the website they want to exchange. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Select exchange offers 3. Select categories 4. Select the product they want to exchange with. 5. Add information of their product. 6. Click Continue | Select Categories: laptop  Choose Model: (Buy) Acer Aspire Nitro VN7-592G  Choose Model: (Exchange) MacBook Pro Core i7 3.1 13" Early 2015 | User should access all the details of the exchange offer. | | As expected, | Pass |
| Post Condition: User must have valid documents of their products. Their enlisted products details will be updated in the database. | | | | | |

## ITEM PASS/FAIL CRITERIA

The pass or fail criteria for a test item is dependent on the type of testing being performed on that item. All functionality in the system must meet the requirements specified in the Vision Document. If the feature does not comply with the documented requirements, then the test case will fail. Each test case will include an expected outcome which will help the tester determine if the test case passed or failed. Acceptance testing will be considered complete once all requirements with a “High” or “Medium” priority in our Requirements

Document have been marked as “accepted”.

## TEST DELIVERABLES

* + Acceptance test plan
  + System/Integration test plan
  + Unit test plans/turnover documentation
  + Screen prototypes
  + Report mock-ups
  + Defect/Incident reports and summaries
  + Test logs and turnover reports

## STAFFING AND TRAINING NEEDS

 In this section we specify the names of the testing team, those who involve in this project. **Training on the application/system:** Need to provide training if the Testers are not aware of the domain

**Training for any test tools to be used:** Need to provide schedule for training to the Testers on the required tool of the project.

 This section also identifies all personnel and the hierarchies relevant to the test plan.

 This includes all areas of the plan such as setting risks, selecting testing and non-testing features, scheduling and most importantly critical go/no go decisions.

Example:

* Staff will require training on new equipment.

## RESPONSIBILITIES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | TM | PM | Dev Team | Test Team | Client |
| Acceptance test Documentation & Execution | X | X | X | X | X |
| System/Integration test Documentation & Exec. | X |  | X | X |  |
| Unit test documentation & execution | X |  | X | X |  |
| System Design Reviews |  |  | X | X | X |
| Detail Design Reviews |  |  | X | X | X |
| Test procedures and rules | X | X | X | X |  |
| Screen & Report prototype reviews |  |  | X | X | X |
| Change Control and regression testing | X | X | X | X |  |
|  |  |  |  |  |  |

### Team Manager:

* Managing team and project budget
* Planning and setting goals for the team
* Conducting performance evaluations of employees
* Supporting employees with training and development activities
* Monitoring team performance to ensure objectives are met

### Project Manager:

* Develop a project plan
* Develop functional specifications
* Plan, estimate and manage resources and the project budget
* Keep track of the project
* Identify necessary adjustments to the plan
* Provide regular updates to the senior management

### Developers:

* Develop the features laid out in the Sprint.
* Update the status of the software project to the Project Manager or Tech Lead.
* Estimate the amount of time needed to deliver a given task.

### Testers:

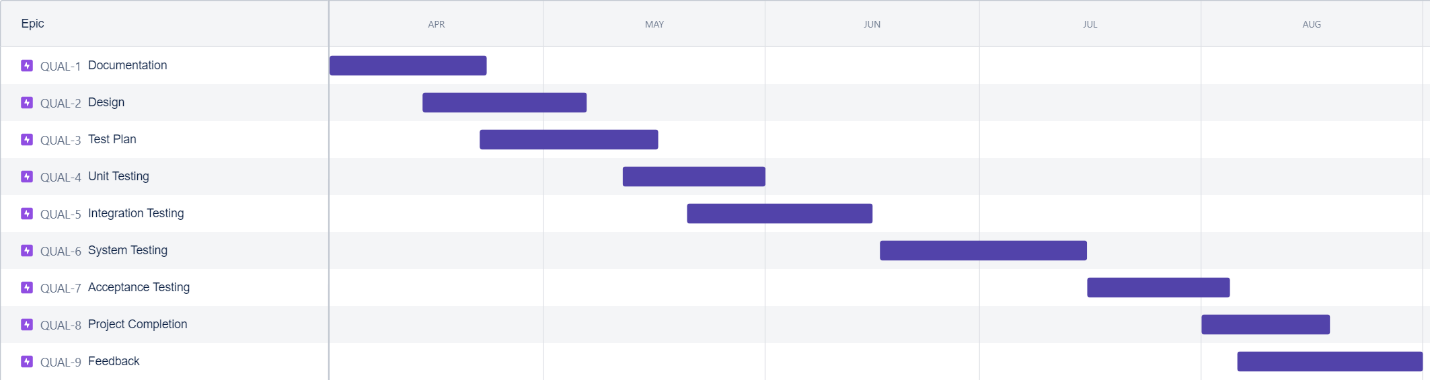
* Understand what the system’s requirement are in order to create and review adequate test cases.
* Alert functional consultants about inconsistencies in the specifications.
* Create and execute test cases to detect bugs and report them in the tracking tool.
* Use automation tools to facilitate regression tasks.

### Client:

* Set requirements and communicate with the developers.
* Maintain and update the product backlog.

## TESTING SCHEDULE

Time has been allocated within the project plan for the following testing activities. The specific dates and times for each activity are defined in the project plan timeline. The persons required for each process are detailed in the project timeline and plan as well. Coordination of the personnel required for each task, test team, development team, management and customer will be handled by the project manager in conjunction with the development and test team leaders.



*Figure 16: Project plan test scheduling*

## PLANNING RISKS AND CONTINGENCIES

* Establish a Contingency Team. Choose a few cross-functional supervisors and anyone else who can help. Assign someone to be the leader in order to keep things going and make decisions as necessary.
* Limited Reassigned Sales staff. The Reassigned Sales administration staff currently has two positions unfilled. As a result of this staff shortage there may be delays in getting staff to review appropriate documents and to participate in the Acceptance test process. Should client staff become a problem, the

appropriate dates for reviews and acceptance testing will slip accordingly. No attempt will be made to bypass any part of the review and testing processes.

* Identify the most serious potential threats to the company, then thoroughly research and analyze them. Consider what would happen if we had to cut, modify, or eliminate important services or functions with the team. All of the issues that have been identified, as well as their impact on the business, should be documented.

## APROVALS

|  |  |
| --- | --- |
| Project Sponsor | Swarnab Saha |
| Development Management | Naresh Barua |
| EDI Project Manager | Maria Akter Kanta |
| RS Test Manager | Saqib Rahman |
| RS Development Team Manager | Swarnab Saha |
| Reassigned Sales | Saqib Rahman |
| Order Entry EDI Team Manager | Naresh Barua |