e

American International University-Bangladesh (AIUB)  
Department of Computer Science  
Faculty of Science &Technology (FST)  
Spring 21 22

Section: C  
Software Quality Assurance and Testing

**TICKET ISSUEING BOOTH FOR DSS (Dhaka Metro Rail)**

A Report submitted By

|  |  |
| --- | --- |
| Name | Id |
| EZAZUL ISLAM | 18-38362-2 |

Under the supervision of

Abhijit Bhowmik

Software Test Plan

for

<TICKET ISSUEING BOOTH FOR DSS>

Version 1.0 approved

Prepared by

EZAZUL ISLAM

<American International University Bangladesh>

<25/04/2022>

Table of Contents

[Revision History 3](#_Toc101749673)

[1. TEST PLAN IDENTIFIER: 4](#_Toc101749674)

[2. REFERENCES: 4](#_Toc101749675)

[3. INTRODUCTION 4](#_Toc101749676)

[Background to the Problem 4](#_Toc101749677)

[Solution to the Problem 4](#_Toc101749678)

[4. REQUEIREMNT SPECIFICATION 4](#_Toc101749679)

[4.1 System Features 4](#_Toc101749680)

[4.2 System Quality Attributes: 5](#_Toc101749681)

[4.3 System Interface: 8](#_Toc101749682)

[4.4 Project Requirements 11](#_Toc101749683)

[5. FEATURES NOT TO BE TESTED AND TO BE TESTED 13](#_Toc101749684)

[6. TESTING APPROACH 14](#_Toc101749685)

[6.1 Testing Levels 14](#_Toc101749686)

[6.2 Test Tools 14](#_Toc101749687)

[6.3 Meetings 17](#_Toc101749688)

[7. TEST CASES/TEST ITEMS 17](#_Toc101749689)

[8. ITEM PASS/FAIL CRITERIA 20](#_Toc101749690)

[9. TEST DELIVERABLES 22](#_Toc101749691)

[10. STAFFING AND TRAINING NEEDS 22](#_Toc101749692)

[11. RESPONSIBILITIES 22](#_Toc101749693)

[12. TESTING SCHEDULE 23](#_Toc101749694)

[13. PLANNING RISKS AND CONTINGENCIES 24](#_Toc101749695)

[14. APROVALS 25](#_Toc101749696)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Updated by | Update Comments |
| 0.1 | 18.04.2022 | Islam, Ezazul | First Draft |
| 0.2 | 19.04.2022 | Islam, Ezazul | Second Draft |
| 0.3 | 20.04.2022 | Islam, Ezazul | Third Draft |
| 0.4 | 21.04.2022 | Islam, Ezazul | Fourth Draft |

# TEST PLAN IDENTIFIER:

TP\_Ticket Issuing Booth For DSS\_1.0

# REFERENCES:

* Kelly, J. C., Sherif, J. S., & Hops, J. (1992). An analysis of defect densities found during software inspections. Journal of Systems and Software, 17(2), 111-117.​
* R.S. Pressman & Associates, Inc. (2010). Software Engineering: A Practitioner’s Approach.

# INTRODUCTION

## Background to the Problem

This project is about the testing approach and overall framework of the Ticket Issuing Booth for Bangladesh Railway. Unit testing will be done at the beginning. Integration testing will be next. After that, the System Testing will be done. Finally, acceptance testing will finish the job. Online ticket issuing system is gaining popularity day by day. Even after people buy ticket in online, they had to go in the ticket counter for the hard copy. As we all know how crowded the station is and for the traffic jam of Dhaka city It’s really hard for the people to issue the ticket.

## Solution to the Problem

After researching a lot, we came up with a great idea. If we continue ticket issuing booth system in Dhaka city it will solve the problem for the regular people. This booth can print hard copy even if you buy a ticket online. All you have to do is put the unique number of the ticket and it will print a hard copy. Anyone can buy a hard copy from the booth and pay using internet banking or Bkash. But you must have a valid NID and phone number. No registration is needed to buy a ticket.

# REQUEIREMNT SPECIFICATION

## System Features

* The software will support interface to touch screen monitors as well as keyboard interface.
* The software will support display of the list of incoming trains, their destinations and arrival and departure times, fare.
* The software will support multiple ticket purchase simultaneously.
* The software will support limiting the number of tickets purchased at the same time.
* This privilege control will be done by the administrator access only.
* The software will support ticket cancellation before final confirmation of the purchase.
* The software will support purchased ticket cancellation support by the administrator.
* The software will support credit transaction and validation.
* The software will support next and previous navigation during ticket purchase process.
* The software will support ticket availability information.
* The software will support information display via web.
* The software will use Oracle database server.

## System Quality Attributes:

**Functional Requirements:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement** | **Description** | **Priority** | **Condition** |
| **Software Login:** | System shall allow user to login to the system with valid NID, phone number with OTP and Face Scan. | High | Must have a valid NID and Phone number |
| **Theme:** | System will let user choose which theme to choose.  After choosing the theme user can change it also. | Low | Must have a valid NID and Phone number |
| **Language:** | System will automatically use English language.  User can change language.  User don’t need to choose language every time, it will be automatically saved by system after first time. | Low | Must have a valid NID and Phone number |
| **Search:** | User can search the destination he wants to travel. | High | Must have a valid NID and Phone number |
| **Advertisement:** | User will find advertisement for of government services. | Medium | Must have a valid NID and Phone number |
| **Money Transection:** | User can pay by banking method.  User can pay by net banking. | : High | Must have a valid NID and Phone number |
| **Discount Offer:** | User can enjoy discount for advertisement in some specific occasion. | Low | Must have a valid NID and Phone number |
| **Account profile:** | User’s Name and information will be in their own profile | High | Must have a valid NID and Phone number |
| **Contact:** | User’s contact information will be in their profile  User can hide it if they want  User can block another user.  Emergency contact information will be added if contact is hiden. | High | Must have a valid NID and Phone number |
| **Account security:** | Booth get alert about unrecognized logins.  User can use two factor authenticators. | High | Must have a valid NID and Phone number |
| **Notification:** | Ticket purchasing information will be sent to user’s phone number. | Medium | Must have a valid NID and Phone number |
| **Update:** | Booth will regularly get updates in the background. | Medium | Must have a valid NID and Phone number |
| **FAQ:** | Here answers to some common questions about how to use this software. | Low | N/A |

**Non-Functional Requirements:**

**1. Availability:**

* The system shall be at least 99% available all over the week.
* Users can post job anytime throughout the week.
* The website should be able to handle at least 0.1 million people at a time.
* The failure rate in the ratio of 1:10000 is acceptable.

**Priority Level:** High

**2. Performance:**

* The software will be loaded in maximum 6 seconds

**Priority Level:** Medium

**3. Efficiency:**

* The booth should have enough processor and ram to work without any interruption.

**Priority Level:** Medium

**4.Flexibility:**

* A maintenance programmer who has at least 1 year of experience supporting this product shall be able to make a new copy output available product or adding new features include code modifications and testing, with no more than one hour of labor.

**Priority Level:** Medium

**5. Integrity:**

* Only the person who don’t have any account will be able to open a new account.
* Only the administration will able to see the personal details like phone numbers, email etc.
* The will able to see the transaction history but can’t change it.

**Priority Level:** High

**6.Interoperability:**

* The system shall be able to import any required data from Google and Facebook.
* The system shall be able to import and check data from NID servers.
* The system shall be able to import data from mobile banking and internet banking services.

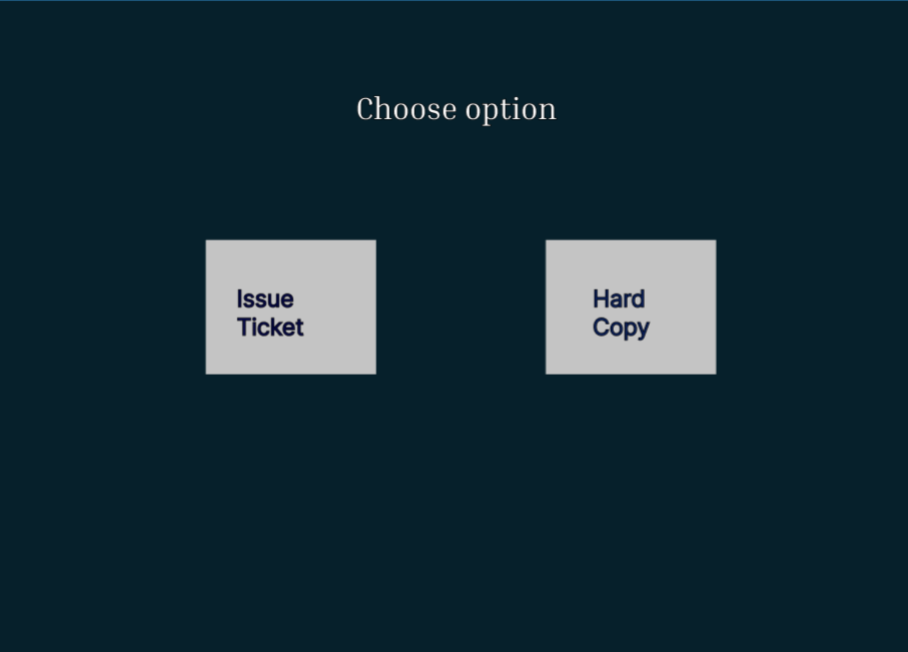
**Priority Level:** High

**7.Portability:**

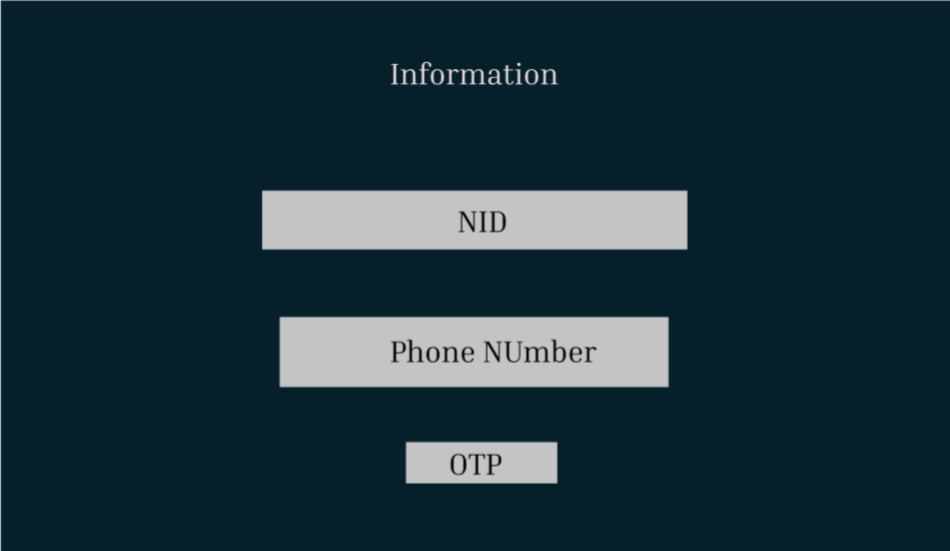
* There will be a fixed operating system for every booth

**Priority Level:** Low

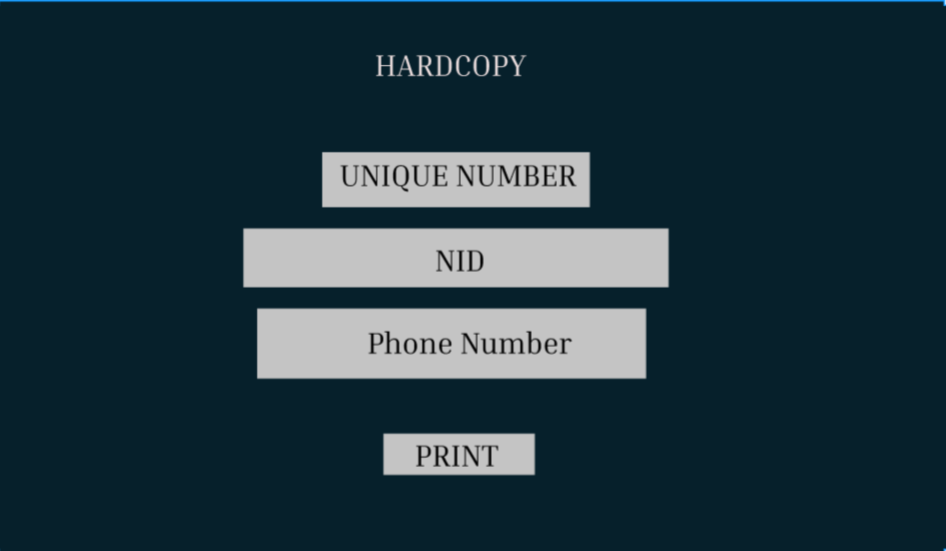
## System Interface:

****

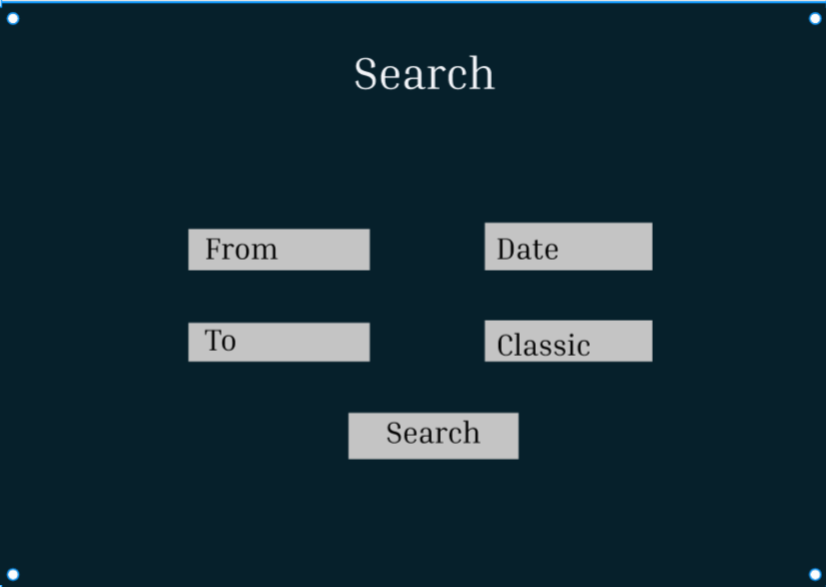
**FIGURE 1**

****

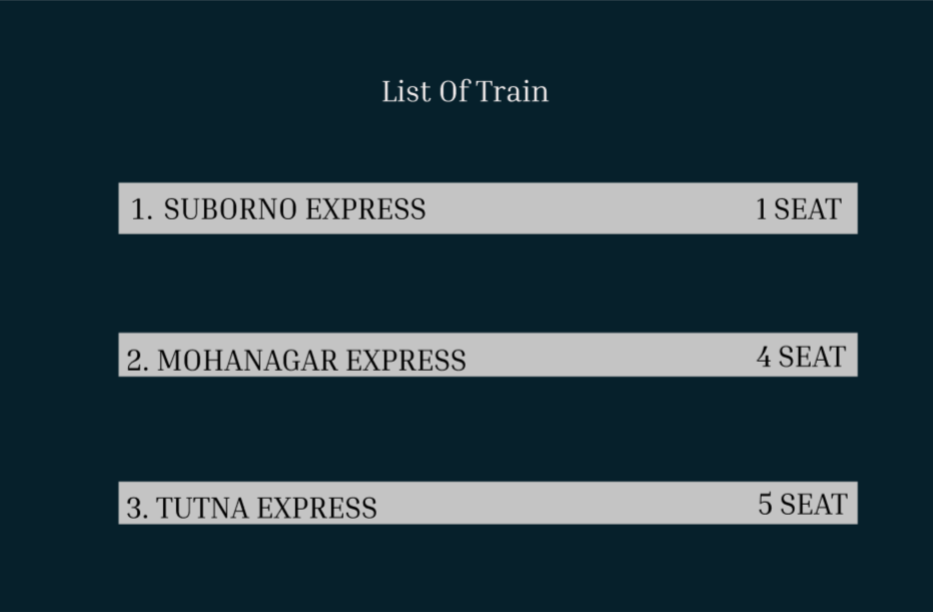
**FIGURE 2**

****

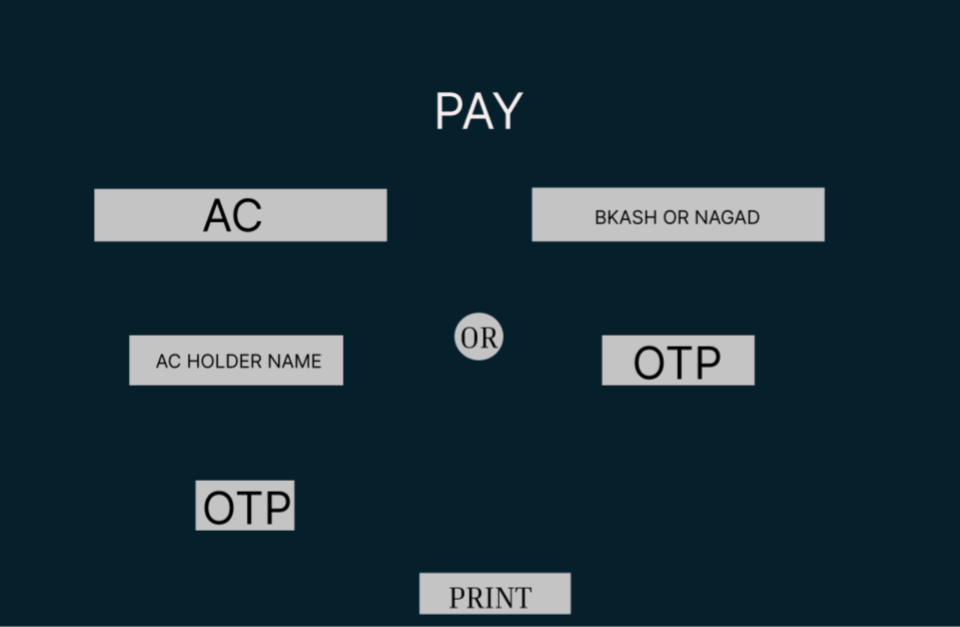
**FIGURE 3**

****

**FIGURE 4**

****

**FIGURE 5**

****

**FIGURE 6**

## Project Requirements

Based on the team size our project fell into organic model

So, here-

Source lines of code, SLOC = 14000

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

=3.0\*(14)^1.12 [14000SLOC/1000=14K SLOC]

=47.04 person-months

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

= 2.50\*(47.04)^0.35 [PM=39]

= 41.16 months

Required number of people= ST = PM/DM

= 47.04/41.16 [PM = 47.04, DM = 41.16]

= 1.14 members

Working hour per day for single person = 8hour

In 10 months, number working days = 22\*10 (Form 10\*30 = 300 days)

= 220

= 225 workings days

Total Hours = 225\*8

= 1800 hours

Per person salary in a month = 100000 BDT

Per hour salary for a person = 100000/180 [In a month working hours= 22\*8 = 180 Hours]

= 555 BDT

So, charge for the project = 1800\*555

= 9,99,000

= 10,00,000 BDT

Requirement analysis = 15 days\*8 hours

= 120 hours

Charge for requirement analysis=120\*300=36000 BDT

Travel expense=15000 BDT

Office rent expense=1,20,000 BDT

Electricity & Gas bill=15000 BDT

Training and hardware cost=100000 BDT

Maintenance for 6 months=6\*10=60 hours

Maintenance cost=60 hours\*1200

=72000 BDT

Total cost=10,00,000+36000+15000+1,20,000+15000+100000+72000

=13,54,000 BDT

=13,60,000 BDT

Bill=Total cost\* (20% profit)

= 13,60,000 + 2,72,000

= 16,35,000 BDT

# FEATURES NOT TO BE TESTED AND TO BE TESTED

5.1 The features which will not be tested are as follows:

* **Privilege accessibility:** Who has the privilege is not a concern of users. It is matter that they are getting the service or not.
* **Oracle database server support:** Database will not show to the users. So, whether it supports or not does not matter because there can be alternative server.
* **Menu selection:** Main features that will available for the process should be enough. So, other small features will not bother the users.

5.2 The feature which will be tested are given below:

* Touch Screen and keyboard interface usability.
* The accuracy of displaying arrival and departure time and fare as well as information of trains.
* Multiple ticket purchasing is properly working or not.
* Number of tickets purchasing limitation.
* Cancellation of ticket purchase before the final confirmation of the purchase.
* Purchased ticket cancellation support by the administrator.
* Support credit card transaction and validation.
* Support navigation during the ticket purchasing process.
* Tickets availability information.
* Support display of information via Web.

# TESTING APPROACH

## Testing Levels

The approach of the testing process is a follow:

* At first, test cases need to be prepared and arranged according to the specification.
* To get the proper operation of each unit the system requirements and test cases shall be reviewed.
* To execute the test cases, testing tools such as Selenium, Katelin Studio, Test Complete will be used.
* Workshops will be arranged in case of any training needed.
* A documentation of the results for every test case should be kept for further verification.
* The tests process shall be performed manually and after that automated testing will be conducted.
* Before the system is eligible for integration testing or system testing, unit testing should be done.
* If there is any new defects or bugs arrive, the development team should be informed immediately.
* Regression testing shall be used for verifying the modifications.
* Immediate action needed if any modification or update is required.

## Test Tools

We have used selenium to test our project.

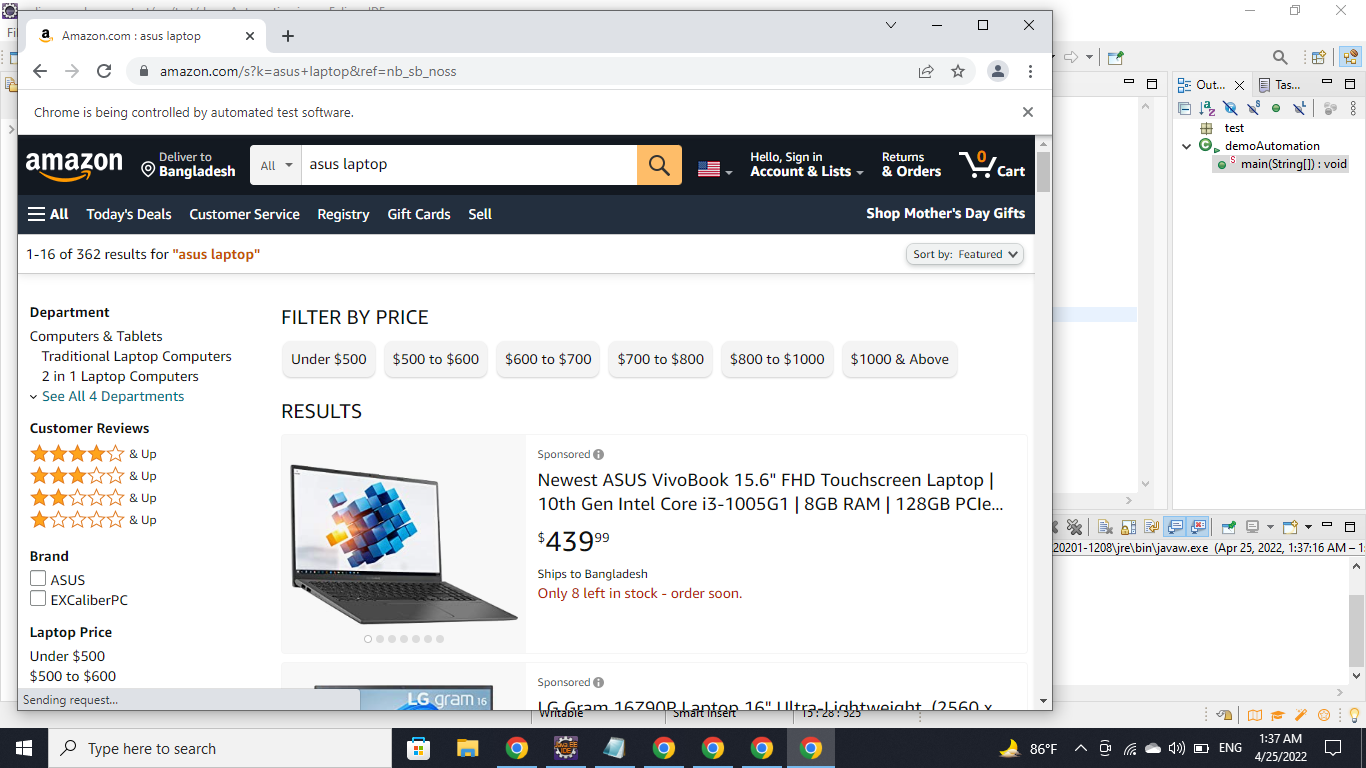


Figure:1

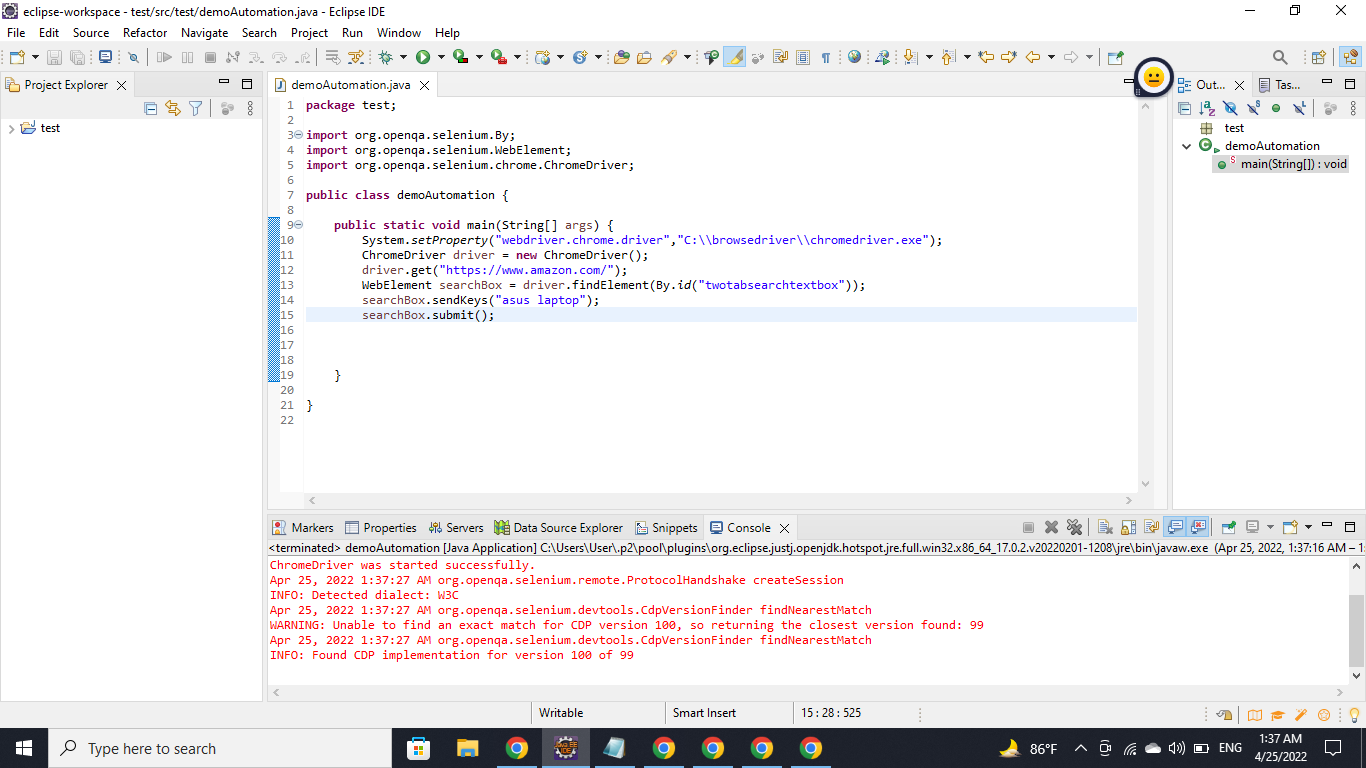


Figure: 2

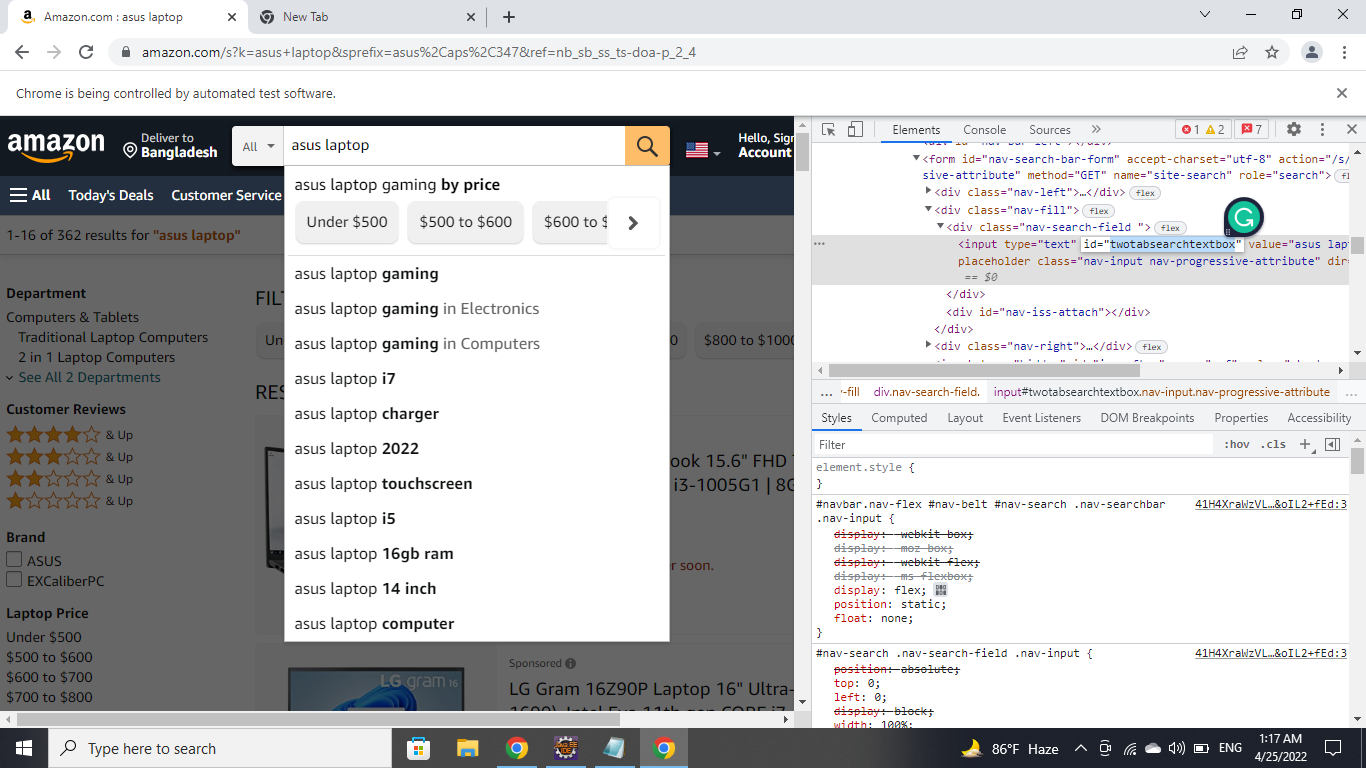


Figure:3

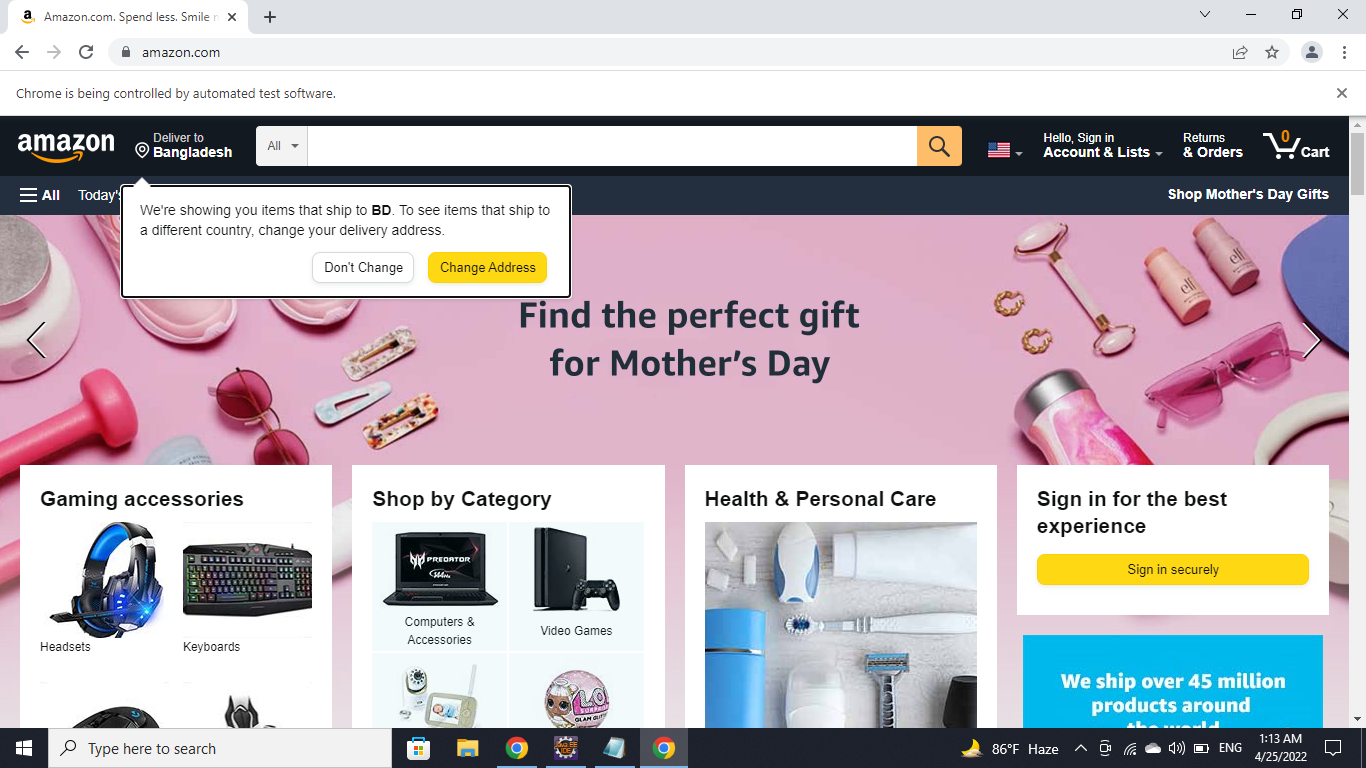


Figure:4

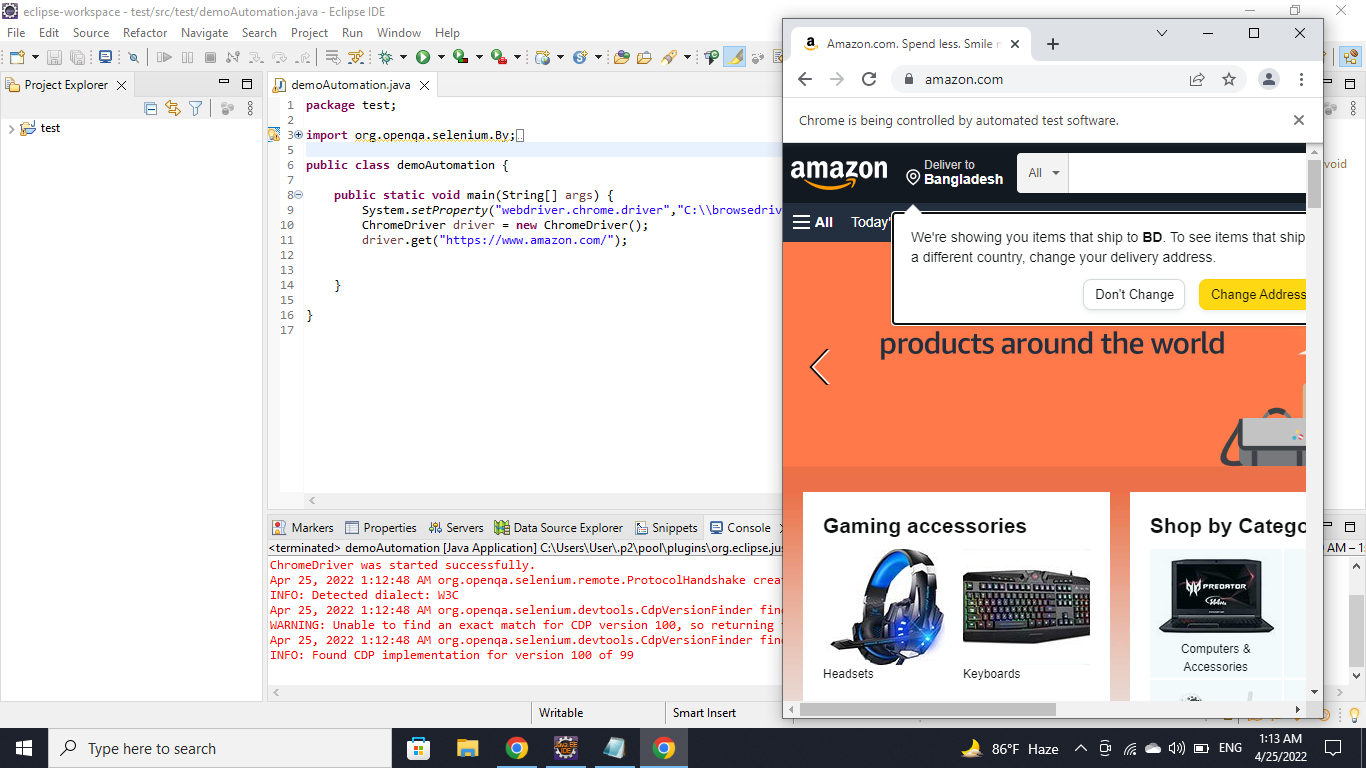


Figure:5

## Meetings

The test team will meet once every two weeks to evaluate progress to date and to identify error trends and problems as early as possible. The test team leader will meet with development and the project manager once every two weeks as well. These two meetings will be scheduled on different weeks. Additional meetings can be called as required for emergency situations.

# TEST CASES/TEST ITEMS

Test case 01:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project name: Ticket Issuing Booth for DSS. | | | | |
| Test case ID: TC DSS 01 | | | | |
| Test priority: High | | | | |
| Module name: Ticket showing system | | | | |
| Test title: Verification of proper ticket schedule showing | | | | |
| Description: When users enter into the system, they can view the list of upcoming trains, their destinations, arrival times and de artier times and the fare. | | | | |
| Pre-Condition: N/A | | | | |
| Steps | Test Data | Expected Result | Actual Result | Status |
| 1.Enter into the system  2.Look for a train | Mohakhali to  Uttara train details | User should see the train information with fare and times. | As expected, | Passed |
| Post condition: N/A | | | | |

Test case 02:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project name: Ticket Issuing Booth for DSS. | | | | |
| Test case ID: TC DSS 02 | | | | |
| Test priority: High | | | | |
| Module name: Find train. | | | | |
| Test title: Searching for train with user's entered data. | | | | |
| Description: User can enter form and to, and then they can find the upcoming trains for that. | | | | |
| Pre-Condition: N/A | | | | |
| Steps | Test Data | Expected Result | Actual Result | Status |
| 1. Enter into the system   1. Enter from. 2. Enter to.   4.Click on Find. | Find for a train Mohakhali to  Uttara | User should see some trains based on the search. | As expected, | Passed |
| Post condition: N/A | | | | |

Test case 03:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID: TC DSS 03 | | | |  |
| Test priority: High | | | |  |
| Module name: Book Ticket | | | |  |
| Test title: Book ticket. | | | |  |
| Description: User can select a train, and book for seats for the train. | | | |  |
| Pre-Condition: N/A | | | |  |
| Steps | Test Data | Expected Result | Actual Result | Status |
| 1.Select a train  2.Select number of tickets   1. Click on confirm 2. Pay ticket price though their card. 3. Done | Book a ticket from Mohakhali to  Uttara. | User should see a booking confirmation message. | As expected, | Passed |
| Post condition: N/A | | | |  |

Test case 04:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project name: Ticket Issuing Booth for DSS. | | | |  |
| Test case ID: TC DSS 04 | | | |  |
| Test priority: High | | | |  |
| Module name: Cancel Ticket | | | |  |
| Test title: Cancel a ticket by administrator. | | | |  |
| Description: An administrator can cancel user's booked ticket. | | | |  |
| Pre-Condition: Have to be logged into the system as an administrator. | | | |  |
| Steps | Test Data | Expected Result | Actual  Result | Status |
| 1. Login as administrator 2. Search for the ticket to cancel. 3. Select the ticket and cancel. | Cancel one user's ticker | Administrator should see the ticket cancelling confirmation. | As expected, | Passed |
| Post condition: N/A | | | |  |

Test case 05:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project name: Ticket Issuing Booth for DSS. | | | | |
| Test case ID: TC DSS 05 | | | | |
| Test priority: High | | | |  |
| Module name: Payment | | | |  |
| Test title: Payment system testing. | | | |  |
| Description: User can pay for his/her booked ticket. | | | |  |
| Pre-Condition: Have to select a train and number of tickets first. | | | |  |
| Steps | Test Data | Expected Result | Actual  Result | Status |
| 1. Select train and number of tickets. 2. Confirm booking. 3. Pay the amount though card. | Pay for a randomly selected ticket. | User should see the payment successful message. | As expected, | Passed |
| Post condition: N/A | | | |  |

# ITEM PASS/FAIL CRITERIA

At the unit testing:

* If the source code of the small units gives expected result, then the item will be passed.
* The developer will fix the constraints and signatures of the design specification if the test case fails.

At the integration testing:

* If the outcomes of the merging units are accurate, then the test case will be passed.
* In case of failure, the developers will fix the problems.

At the system testing:

* If the entire set of test cases passes, then the system testing will pass.
* If system testing is fails, then the entire process will start from the beginning.

At the acceptance testing:

* If the client did not accept the software according to their requirements, then the test will be failed.
* In that circumstances the software will be rejected.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TCI | Module name | Expected Input | Expected Result | Pass Rate | Fail Rate |
| TC DSS O  1 | Ticket showing system | Mohakhali to Uttara train details | User should see the train information with fare and times. | 100% |  |
| TC DSS O 2 | Finding Train | Find for a train  Mohakhali to  Uttara | User should see some trains based on the search. | 85% | 15% |
| TC DSS O  3 | Book Ticket | Book a ticket from Mohakhali to Uttara. | User should see a booking confirmation message. | 95% | 5% |
| TC DSS O 4 | Cancel Ticket | Cancel one user's ticker | Administrator should see the ticket cancelling confirmation. | 100% | 00/0 |
| TC DSS O  5 | Payment | Pay for a randomly selected ticket. | User should see the payment successful message. | 92% | 8% |

# TEST DELIVERABLES

Projects create deliverables, which are simply the results of the project or the processes in the project. That means a deliverable can be something as big as the objective of the project itself or the reporting that is part of the larger project.

* Test Strategy
* Test Plan Document
* Test Cases
* Test Summary Report
* Test Evaluation Report

# STAFFING AND TRAINING NEEDS

To build software a certain number of staffs is essential. The staffs will be categorized according to their expertise. Such as, developer, project manager, quality assurance, requirement engineer and so on.

It is not necessary that every staff will understand every tool which will use to build the software. That is why there should be training sessions which will help them to understand the tools such as, Selenium, Katelin Studio and so on.

# RESPONSIBILITIES

**Test Lead:**

* + Building up and leading the Testing Team to the success of project.
  + Defining the scope of testing within the context of each delivery.
  + Deploying and managing resources for testing.
  + Applying the appropriate test measurements and metrics in the product and the Testing Team.
  + Planning, deploying and managing the testing effort for any given engagement.

**Project Manager:**

* Plan and implement projects.
* Help define project scope, goals and deliverables.
* Define tasks and required resources.
* Collect and manage project team.
* Manage budget.
* Allocate project resources.
* Create schedule and project timeline.
* Track deliverables.

**Quality Assurance Tester:**

* Determining, negotiating and agreeing on in-house quality procedures, standards and specifications.
* Assessing customer requirements and ensuring that these are met.
* Setting customer service standards.
* Determining training needs.
* Writing management and technical reports.

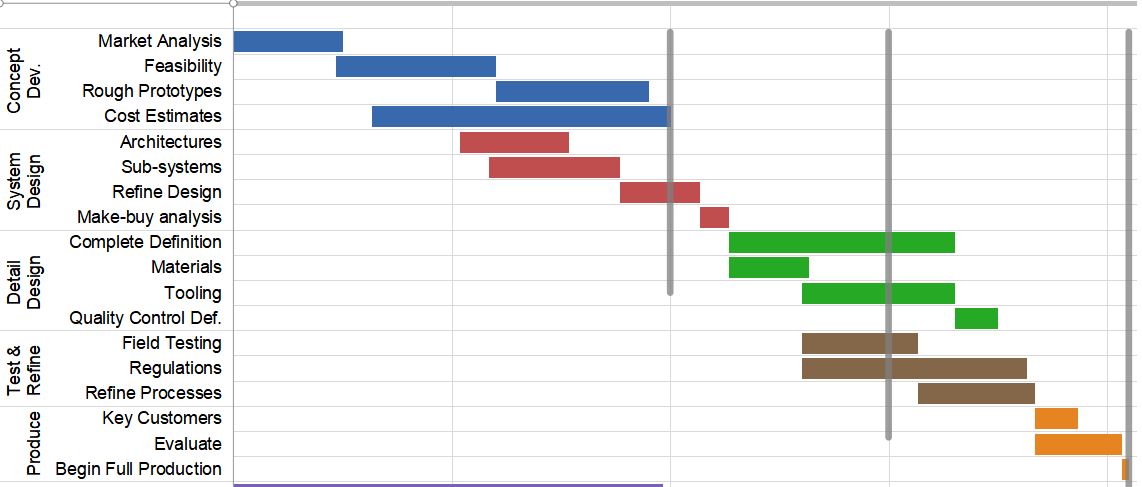
**Developers:**

* Writing and implementing efficient code.
* Determining operational practicality.
* Deploying software tools, processes and metrics.
* Maintaining and upgrading existing systems.

Identifying areas for modification in existing programs and subsequently developing these modifications

# TESTING SCHEDULE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Serial | Tasks | Start | Duration | Responsibility |
| 1 | Unit testing | 3-03-22 | 6 days | Developer/Tester |
| 2 | Integration testing | 9-03-22 | 6 days | Tester/Developer |
| 3 | System review | 15-03-22 | 4 days | Testing team lead |
| 4 | Design review | 19-03-22 | 4 days | Testing team lead |
| 5 | Changing control | 21-03-22 | 2 days | Developer/Tester |
| 6 | Regression testing | 25-03-22 | 4 days | Tester |
| 7 | System testing documentation | 27-03-22 | 2 days | Tester/Team lead |
| 8 | System testing | 1-04-22 | 4 days | Tester |
| 9 | Changing control | 5-04-22 | 2 days | Developer/Tester |
| 10 | Regression testing | 7-04-22 | 4 days | Tester |
| 11 | System testing documentation | 11-04-22 | 2 days | Tester/Team lead |
| 12 | System testing | 13-04-22 | 4 days | Tester |
| 13 | Acceptance testing | 17-04-22 | 4 days | End user/Third party testing team |
| 14 | Reviewing test case | 21-04-22 | 4 days | Testing team lead |



# PLANNING RISKS AND CONTINGENCIES

There can be two types of risk.

* **Risk during planning:** The most common failure while planning is budget underestimation. Sometimes, more feature adds to the previous requirement list. As a result, previous budget fails to complete the project. Another is deadline miss. For adding more feature, it takes more time to complete the project. That is why software cannot be delivered in time. As well as, it causes budget failure.
* **Risk during using the software:** Though it has a feature of using personal information, especially while transaction. It can affect the user by breaching the privacy of the user. Such as, leaking their private information, hacking account for money, follow their destination for bad reason and so on.

As a backup for planning risk, time and budget should be estimated to the maximum level. In terms of users’ risk, there can be biometric safety system which will confirm the user after inserting their transaction password. The required number of test employee may not be obtained on time, which can be a danger for the working schedule. The schedule must be updated in time to account if needed. At the time of working some software may need to use which are not available at that moment. For this the person who will realize it firstly he/she should inform the lead and lead has to manage these resources if they are must. Similarly, if any contingencies will appear at the time of using available resource, then the person who will face should knock the lead immediately. If the possibility of finishing project in the deadline will decrease day by day manager or main project lead should take some necessary steps to fix with ay parameter even the schedule may need to change too. If any small problem of different team will appear then tester may knock them directly.

# APROVALS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Designation** | **Institute** | **signature** | **Date** |
| Abhijit Bhowmik | Project Supervisor | AIUB IT Farm | Abhijit | 22-02-2022 |
| Islam, Ezazul | Project Manager | AIUB IT Farm | ezaz | 22-02-2022 |
| Himi, Humayra | Test Lead | AIUB IT Farm | Himi | 22-02-2022 |
| Chowdhury, Suriya | Development  Management | AIUB IT Farm | Suriya | 22-02-2022 |
| Murad, Md Asif Dewan | Quality Assurance | AIUB IT Farm | Murad | 22-02-2022 |
| Sakib, Md Nazmus | Client | Sakib Enterprise | Sakib | 22-02-2022 |