American International University-Bangladesh (AIUB)
Department of Computer Science
Faculty of Science &Technology (FST)
Summer 22 23

Section: C
Software Quality Assurance and Testing

Online Movie Ticketing System

A Report Submitted By

SN	Student Name	Student ID
1	Mushfiqur Rahman Abir	20-42738-1
2	Tasnim Binta Hossain Shakal	20-42711-1
3	Marina Afroj	20-44121-2
4	Sharmin Zaker Zerin	18-38783-3

Under the supervision of

MD MOMAN UL HAQUE KHAN

Software Test Plan

for

Online Movie Ticketing System

Version 1.0 approved.

Mushfiqur Rahman Abir
Tasnim Binta Hossain Shakal
Marina Afroj
Sharmin Zaker Zerin

American International University-Bangladesh
26 August 2023

Checked by Industry Personnel

Name:

Designation:

Company:

Sign:

Date:

Table of Contents

Revision History

Revision	Date	Updated by	Update Comments
0.1	2023.4.15	Mushfiqur Rahman Abir	First Draft
0.2	2023.4.18	Sharmin Zaker Zerin	Second Draft
0.3	2023.4.21	Tasnim Binta Hossain Shakal	Third Draft
0.4	2023.5.3	Marina Afroj	Fourth Draft

1. TEST PLAN IDENTIFIER: RS-MTP01.3

2. REFERENCES

- Software Engineering Course Slides
- Software Quality and Testing Course Slides

3. INTRODUCTION

Background to the Problem

The online movie ticketing system faces several background problems, including scalability, security, integration, and customer support issues. These problems can negatively impact the user experience and lead to dissatisfied customers. Therefore, it is essential for online movie ticketing systems to address these issues to ensure a smooth and secure ticketing process for users.

Solution to the Problem

To address the background problems of online movie ticketing systems, the following solutions can be implemented:

Scalability: To handle the increasing load on the system, it is essential to have a scalable architecture that can drive traffic. This can be achieved using cloud-based services or distributed systems that expand or shrink according to demand. Load testing and performance monitoring should be done regularly to identify and address any bottlenecks.

Security: The online ticketing system should be designed with robust security measures to protect user data. This can include using SSL encryption for all transactions, two-factor authentication, and implementing security best practices to prevent hacking attacks. Regular security audits should also be conducted to identify and address any vulnerabilities.

Integration: The online ticketing system should be designed to integrate seamlessly with third-party platforms. This can be achieved using standard protocols and APIs, allowing easy data exchange between systems. Regular testing and quality assurance should be done to ensure proper integration.

Customer Support: A comprehensive customer support system should be in place to address any issues that users may face while using the system. This can include a 24/7 support hotline, a chatbot or AI-powered virtual assistant, and a knowledge base with frequently asked questions. The support system should be regularly monitored and updated based on user feedback.

By implementing these solutions, online movie ticketing systems can provide a secure, scalable, and user-friendly platform that meets the needs of their customers.

4. REQUEIREMNT SPECIFICATION

4.1 System Features

- **Two types of users**:1. Admin
  2. Seller
  3. Customer

4. Manager

- **Common Features for all users:**
    1. All users can Login to the system.
    2. Can register to the system.
    3. Can manipulate their profile information (edit-delete-view etc.)
    4. Also, can log out.

- **Admin User Functionality:**
    1. Can log in, log out and registration validation.
    2. Can remove manager, user, and seller.
    3. Can delete notice and can update notice.
    4. Can remove movies and update it.
    5. Can process and reset password.
    6. Can see all customer/seller/manager information.
    7.

Priority Level: High
Precondition: Admin must have valid ‘username’ and ‘password’ saved in database.

Suggestion for Admin functionality:

Admin needs to add verification system for the customer who buys tickets from his website. When any customer fills up all the requirements for buying those tickets then he/she will get a confirmation mail/text from system.

Seller Functionality:

- Can check availability of ticket.
- Can sell tickets.
- Can check seller notice.
- Can contact users/customers.
- Can cancel the ticket.

Priority Level: High

Precondition: Seller must have valid ‘username’ and ‘password’ saved in database.

Suggestion for Seller functionality:

- ✓ For View → Can check availability of ticket and check notice.

- ✓ For Login → Need to fill up with (username and password.)

Customer Functionality:

1. Can view homepage.
2. Can check the venue.
3. Can buy a ticket.
4. Can check notice.
5. Can make online payment.

Priority Level: High

Precondition: Customer must have valid ‘username’ and ‘password’ saved in database.

Suggestion for Customer functionality:

- ✓ For View → Can check homepage, venue, and notice.
- ✓ For Login → Need to fill up with (username and password.)
- ✓ For Payment → Can payment through BKash and Nagad.

Manager Functionality:

1. Can update hall.
2. Can add or remove customers.
3. Can edit customer information.

Priority Level: High

Precondition: Manager must have valid ‘username’ and ‘password’ saved in database.

Suggestion for Manager functionality:

- ✓ For View → Can update hall, add, and remove customer.
- ✓ For Login → Need to fill up with (username and password.)

4.2 System Quality Attributes

System quality attributes are characteristics or properties of a software system that determine its quality and effectiveness. Some of the critical system quality attributes for online movie ticketing system include:

Performance: The system should be able to handle many users and provide fast and responsive performance for ticket bookings, seat selection, and payment processing.

Security: The system should ensure the security and privacy of user information, including credit card details and personal information.

Reliability: The system should always be reliable and available to users, with minimum downtime and disruptions.

Usability: The system should be user-friendly and easy to navigate, with clear and intuitive interfaces for ticket booking, seat selection, and payment processing.

Scalability: The system should be scalable and able to handle increasing users, transactions, and data volume.

Maintainability: The system should be easy to maintain, update, and modify to meet changing user and market needs.

Availability: The system should always be available to users with minimum downtime and disruptions.

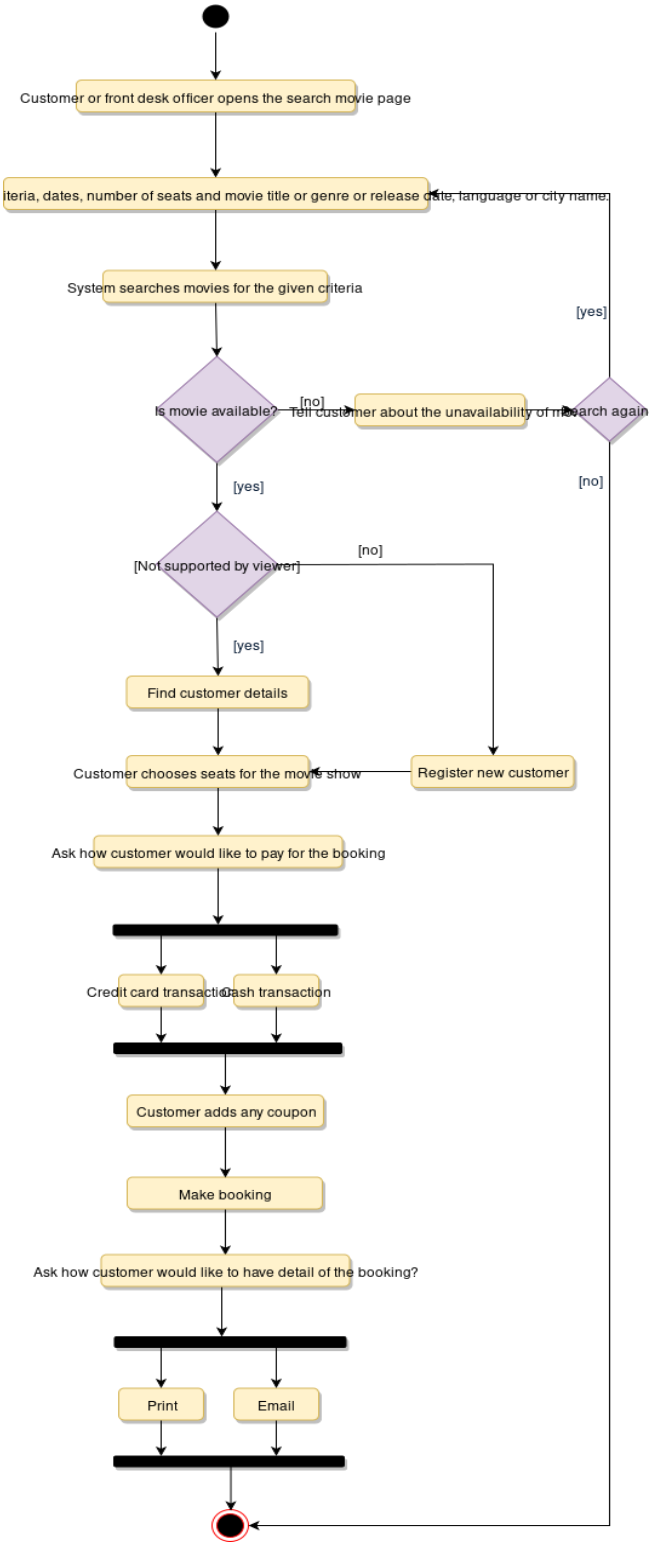
Overall, these system quality attributes play a critical role in determining the effectiveness and success of online movie ticketing system.

4.3 UML Diagrams

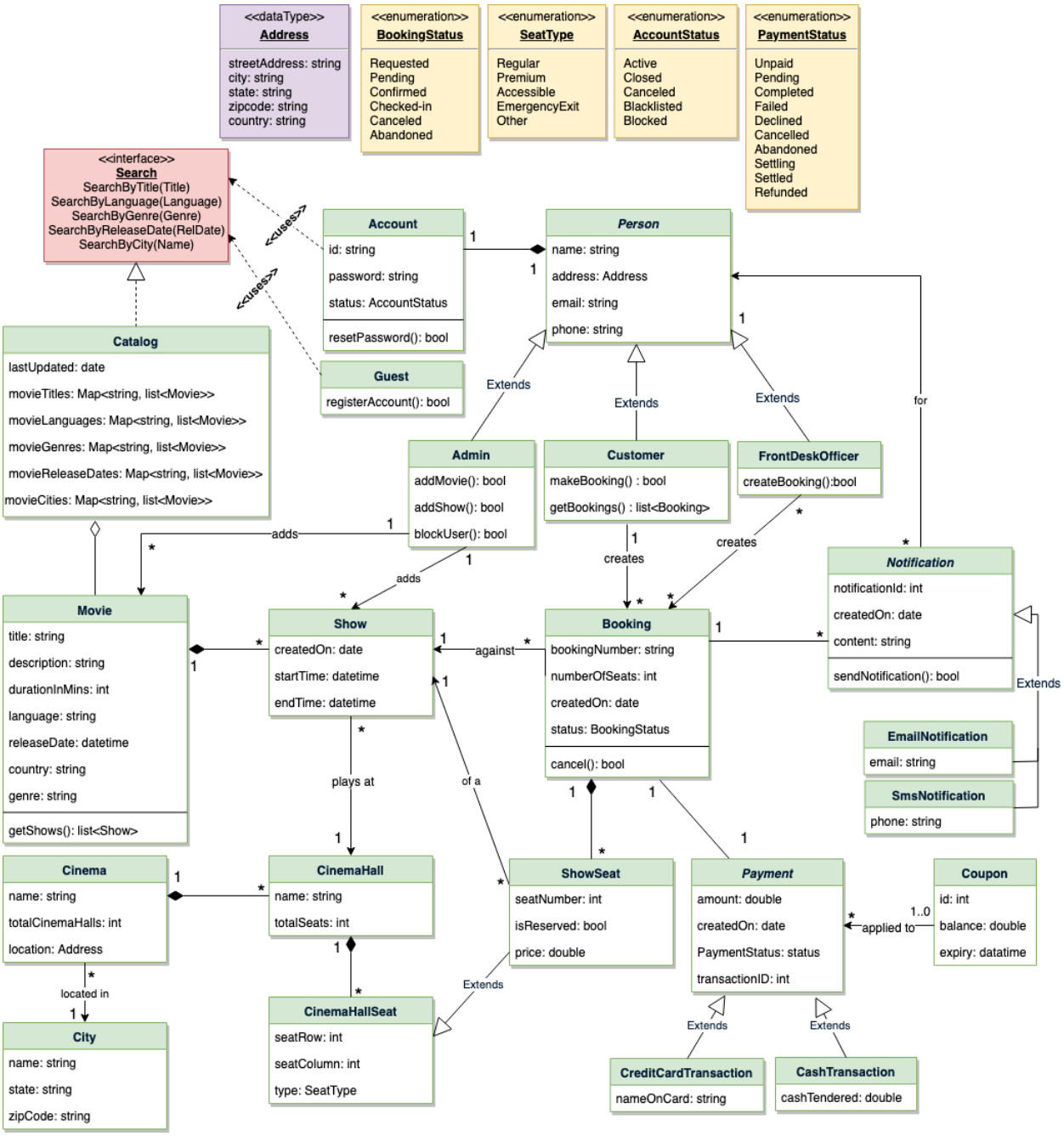
This report delves into the meticulous process of evaluating the Online Movie Ticketing Website's software quality through a comprehensive array of UML diagrams. Unified Modeling Language (UML) diagrams are a visual representation tool used in software engineering to model, visualize, and communicate the design and structure of software systems. UML diagrams provide a standardized way to illustrate various aspects of a system's architecture, behavior, and interactions.

4.3.1 Use Case Diagram

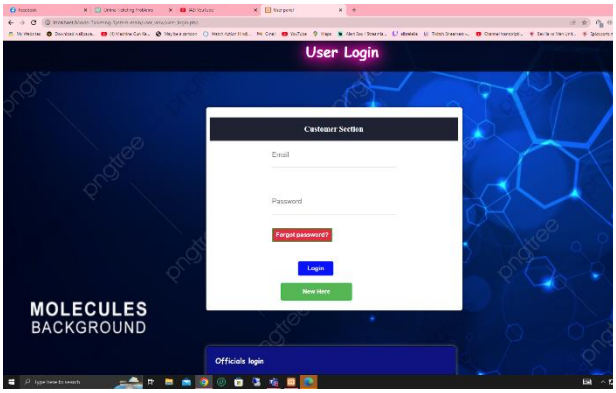
4.3.2 Activity Diagram


Customer or front desk officer opens the search movie page
iteria, dates, number of seats and movie title or genre or release date, language or city name.
System searches movies for the given criteria
[yes]
Is movie available?
[no]
Tell customer about the unavailability of m
earch again
[no]
[yes]
[Not supported by viewer]
[no]
[yes]
Find customer details
Customer chooses seats for the movie show
Register new customer
Ask how customer would like to pay for the booking
Credit card transactio
ash transaction
Customer adds any coupon
Make booking
Ask how customer would like to have detail of the booking?
Print
Email

4.3.3 Class Diagram



4.4 System Interface



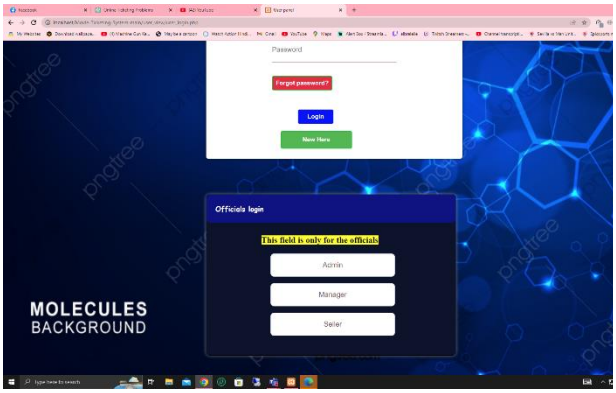


Figure: 4.3(1) Website Home Page

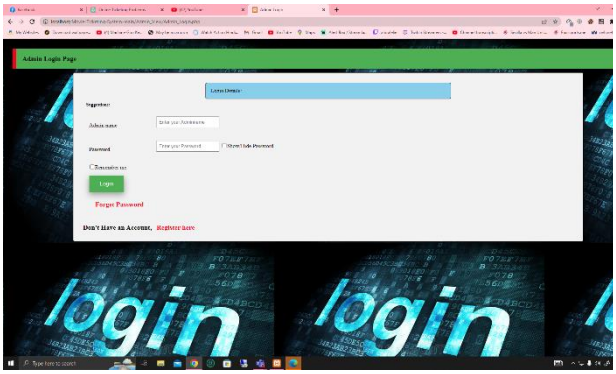


Figure: 4.3(2) Admin Login Page

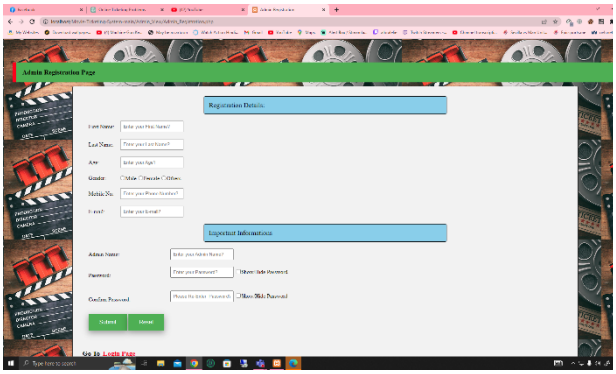
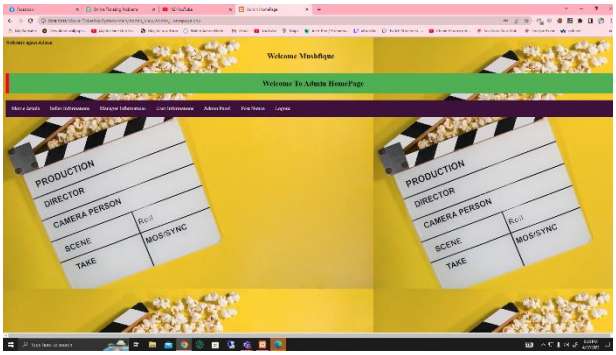


Figure: 4.3(3) Admin Registration Page



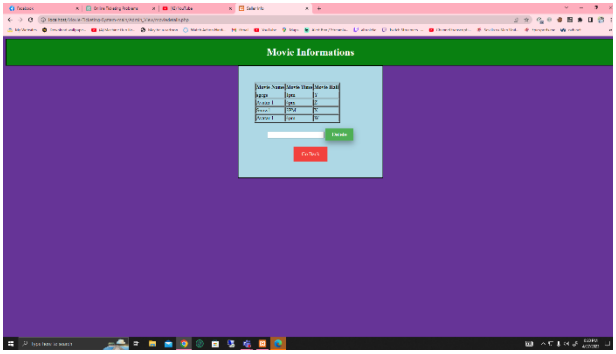
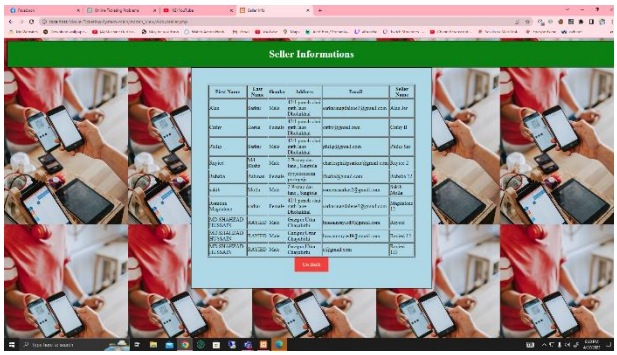


Figure: 4.3(3) Admin Home Page

Figure: 4.3(4) Admin Movie Information



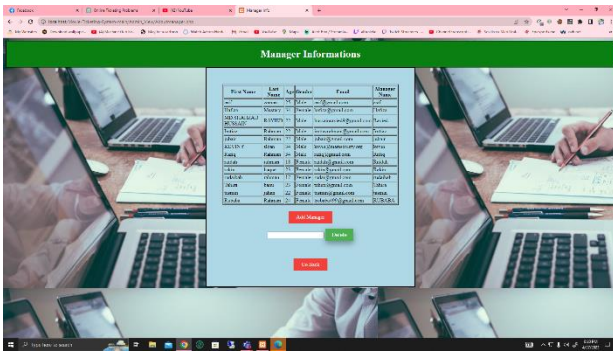
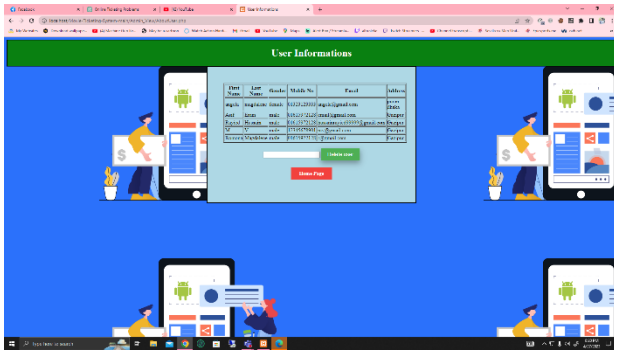


Figure: 4.3(5) Admin Seller Information

Figure: 4.3(6) Admin Manager Information



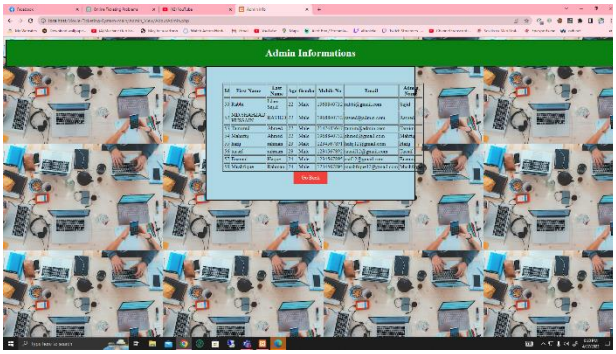


Figure: 4.3(5) Admin User Information

Figure: 4.3(6) Admin's Admin Information

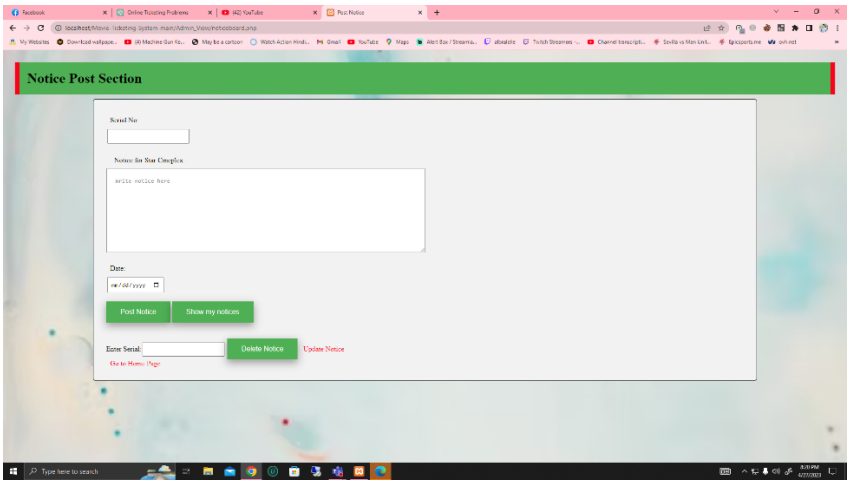


Figure: 4.3(7) Admin Post Notice

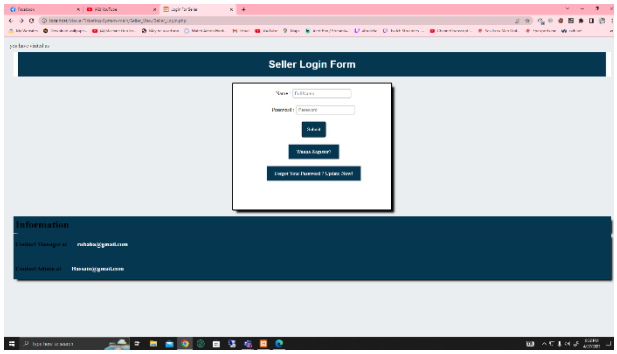


Figure: 4.3(8) Seller Login Page

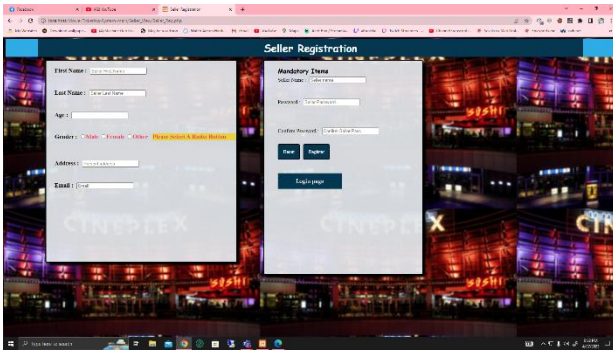


Figure: 4.3(9) Seller Registration Page

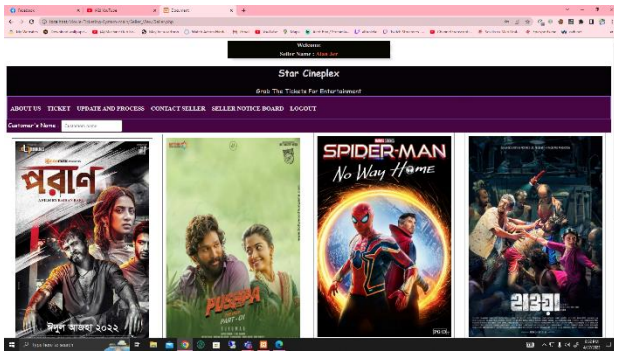


Figure: 4.3(10) Seller Home Page

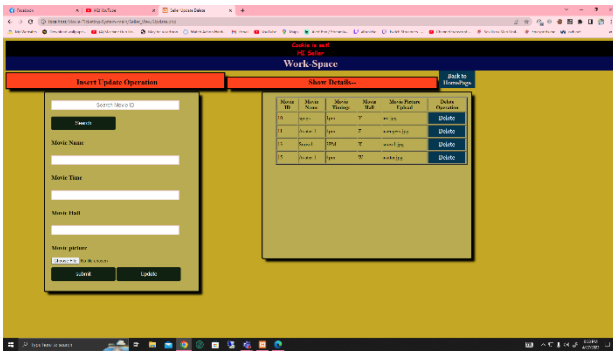


Figure: 4.3(11) Seller Workspace

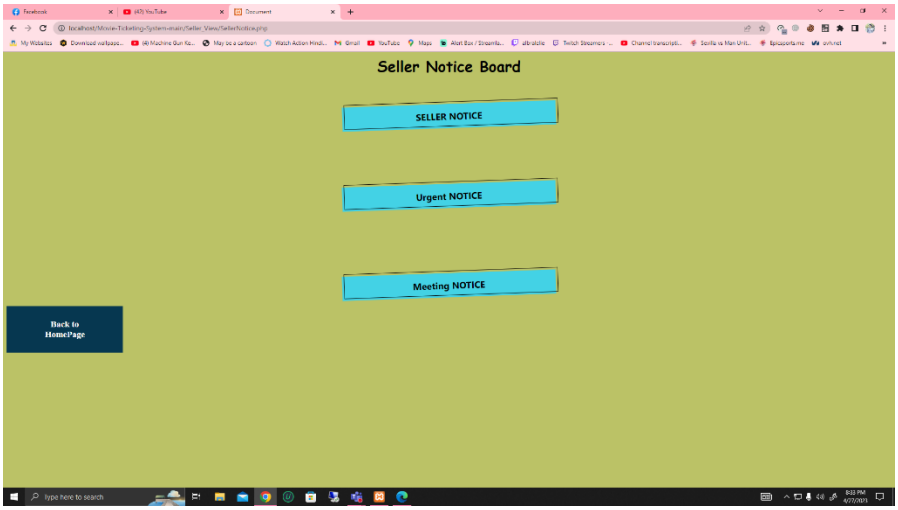


Figure: 4.3(12) Seller Notice Board

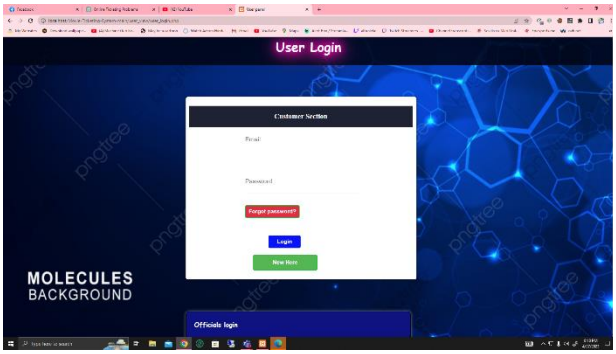


Figure: 4.3(13) User Login Page

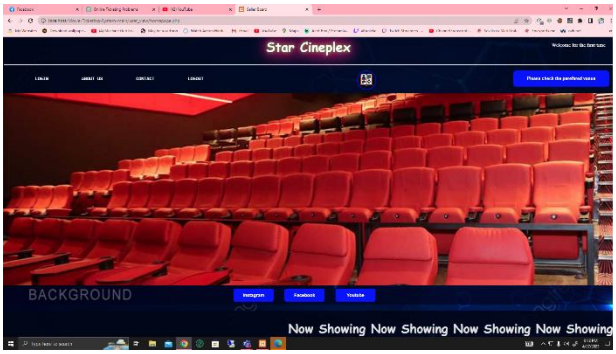


Figure: 4.3(14) User Home Page

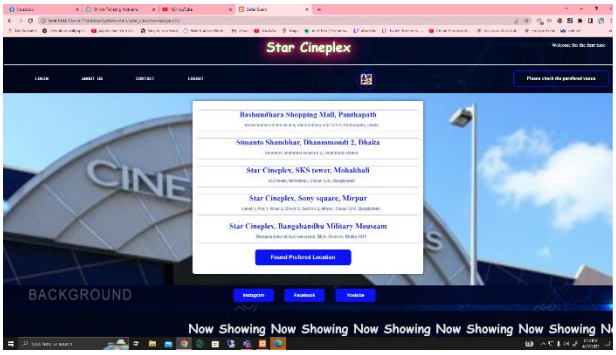


Figure: 4.3(15) User Venue Cheek

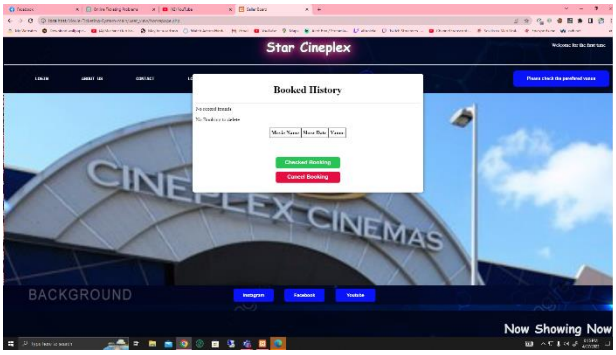


Figure: 4.3(16) User Booked History

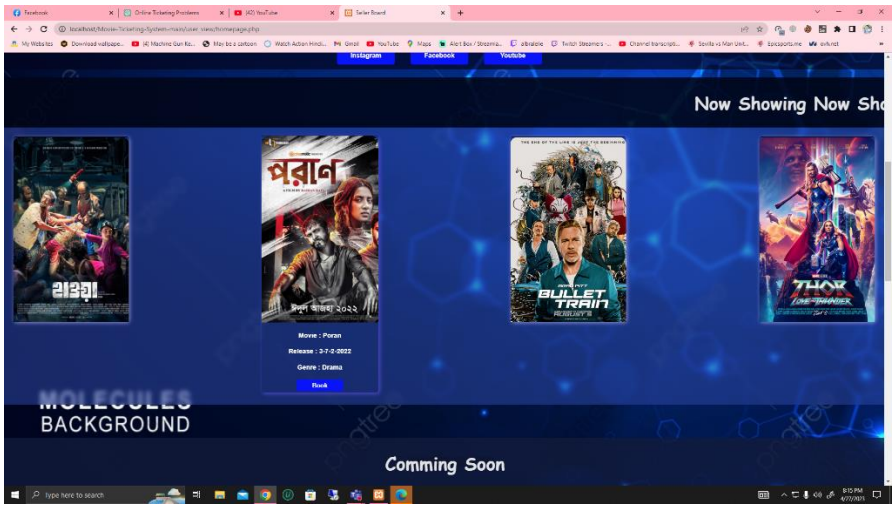


Figure: 4.3(17) User Ticket Booking Page

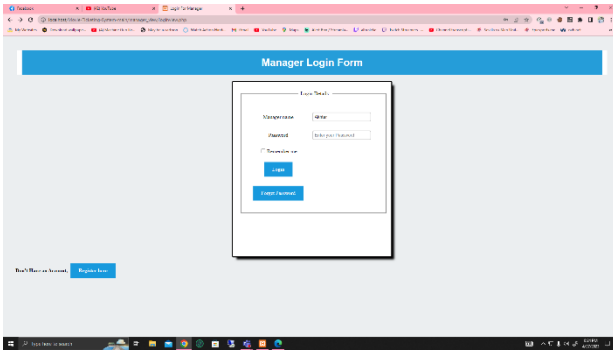


Figure: 4.3(18) Manager Login Page

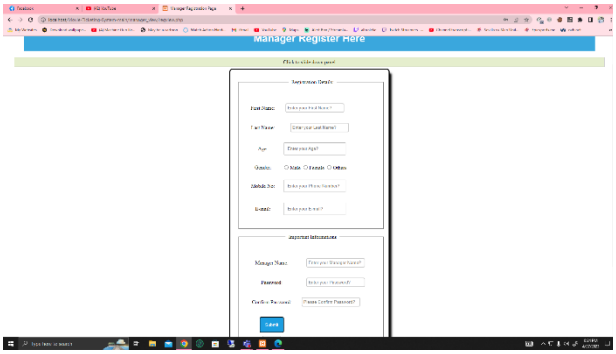


Figure: 4.3(19) Manager Registration Page

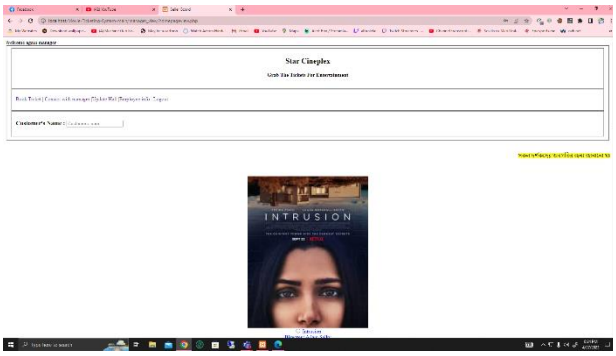


Figure: 4.3(20) Manager Home Page

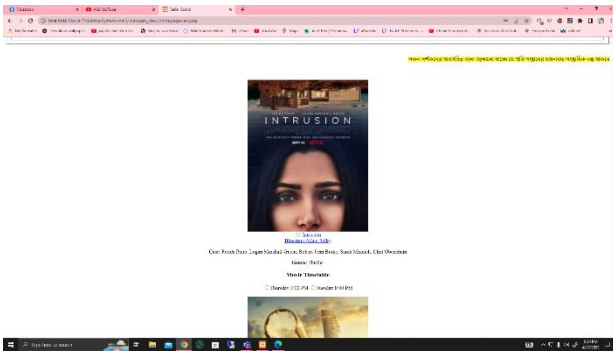


Figure: 4.3(21) Manager Movie Selection

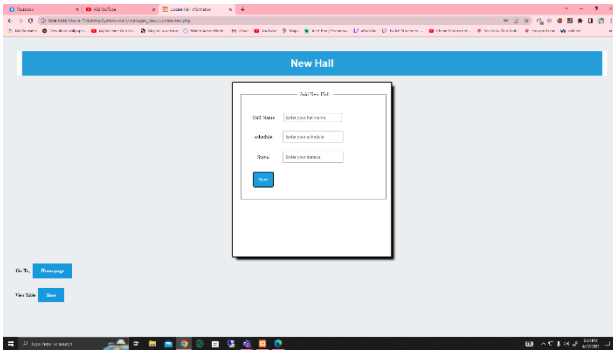


Figure: 4.3(22) Manager Update Hall

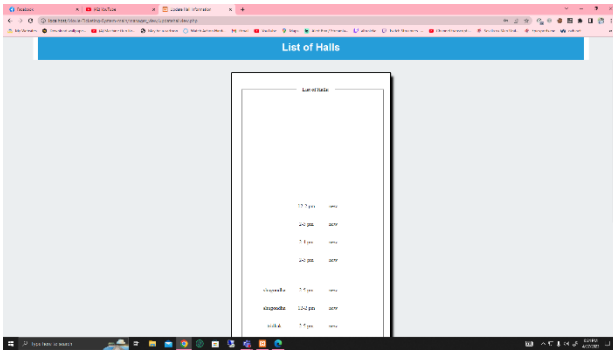


Figure: 4.3(23) Manager Update Hall List

4.5 Project Requirements

For our project development,

- The number of members needed → 4
- Time needed → 2 months
- 1 month 4 weeks. So, we need 2*4 → 8 weeks
- In 1 week, we get 5 working days
- 1 working days, we get 8 hours
- So, total working hour in per week → (5*8) → 40 hours
- Now total working hour in 8 weeks → (8*40) → 320 hours
- Developers' salary per hour → 1250 BDT
- Developer's salary in 320 hours → (1250*320) → 4,00,000 BDT

- Office rent for 3 months → (20000*3) → 60,000 BDT
- Maintenance cost in 1 week (5*1000) → 5,000 BDT
- Maintenance cost in 8 weeks (5,000*8) → 40,000 BDT
- Snacks cost for 1 per in 1 day → 200 BDT
- Snacks cost for 4 per in 1 day → 800 BDT
- Snacks cost for 4 per in 40 day → 32,000 BDT

Expanses	Total Amount (BDT)	Timeline
4 Developers Salary	4,00,000	3 Months
Office rents for 3 months	60,000	3 Months
Maintenance Cost	40,000	3 Months
Electricity Cost	4,000	3 Months
Snacks Cost	32,000	5 Month
Miscellaneous Fee	5,000	1 Month
Total Cost is	5,41,000	
For Profit 15% of total cost	81,150	
Total Budget is	6,22,150	

5. FEATURES NOT TO BE TESTED

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:

Customer Payment Method: In our web-based application one feature is the customer payment method. In testing time, we did not test this feature.

Manager Book Ticket: In our web-based application we already test customer book ticket by customer so that we did not check again with manager book ticket.

Seller Book Ticket: In our web-based application we already test admin customer information by admin so that we did not check again with seller book ticket.

6. TESTING APPROACH

6.1 Testing Levels

6.1.1 Unit Testing:

Test a small software unit at a time. In this test we will test part by part software units. We need to test every single method. It is usually performed by the developer and white box testing is used in it. Main goal of this testing is to isolate written code to test and determine if it works as intended.

1. Test Structure: You organized your tests into separate test classes that corresponded to the functions or methods you were testing.

2. Assertions: Within your test methods, you used various assertion methods provided by PHPUnit to check if the actual outputs matched the expected outputs.

3. Mocking: If your functions depended on external resources or databases, you used PHPUnit's mocking capabilities to isolate the function being tested.

4. Test Suite: You set up a test suite that ran all your unit tests together, ensuring that any changes to your code didn't introduce unexpected bugs.

6.1.1.1 Unit Tests

Function: calculatePrice($ticketType)
Input: $ticketType (string) - Type of ticket (regular, child, senior)
Expected Output: Calculated ticket price based on the type
Test Cases:
Test with regular ticket type
Test with child ticket type
Test with senior ticket type\

Function: reserveSeat($seatNumber)
Input: $seatNumber (integer) - Seat number to be reserved
Expected Output: Confirmation of seat reservation or error message
Test Cases:
Test with valid seat number
Test with invalid seat number

6.1.2 Integration Testing:

After completing the unit testing, we will start integration testing. Here we will make sure that all the modules in our software are integrated logically. Bottom- up integration is followed in this testing.

6.1.2.1 Server-Client Interaction:

This level of integration testing focuses on the interaction between the server-side (PHP) and client-side (JavaScript) components of your system.

Scenario: Reserving a seat triggers communication between the client's front-end (JavaScript) and the server's back-end (PHP) to update seat availability.

Test Steps:

Simulate a seat reservation request on the client side.
Monitor network interactions using browser developer tools or tools like Postman to observe the data exchanged between the front-end and back-end.
Ensure that the correct seat status is updated on both ends.

6.1.2.2 Database Interaction:

Integration testing for database interactions ensures that data is correctly stored and retrieved between your application and the database.

Scenario: Storing and retrieving user information, reservation details, and ticket information in the database.

Test Steps:

Create test data in the database specifically for integration testing.
Perform CRUD (Create, Read, Update, Delete) operations on the data.
Check if data is accurately stored, retrieved, and updated in the database.

6.1.2.3 Payment Gateway Integration:

If your system involves online payments, integration testing should cover interactions with the payment gateway.

Scenario: Processing a payment for a reserved ticket.

Test Steps:

Simulate a payment transaction.
Verify that the payment gateway is properly triggered.
Check if the payment status is correctly updated in the system after a successful payment.

- **System Testing:** After completing both tests we will start system testing. In this test we will test all the features in the software. Everything should be done in system testing. Black Box Testing is used to verify all the requirements meet perfectly.

- **Acceptance Testing:** It is the last step of testing. Not only developers but also users can also check this testing. If all the functions work properly the user accepts. Otherwise, the developer needs to fix this problem again. After that, software is ready to release for users.

6.2 Test Tools

We used Selenium IDE for testing cases.

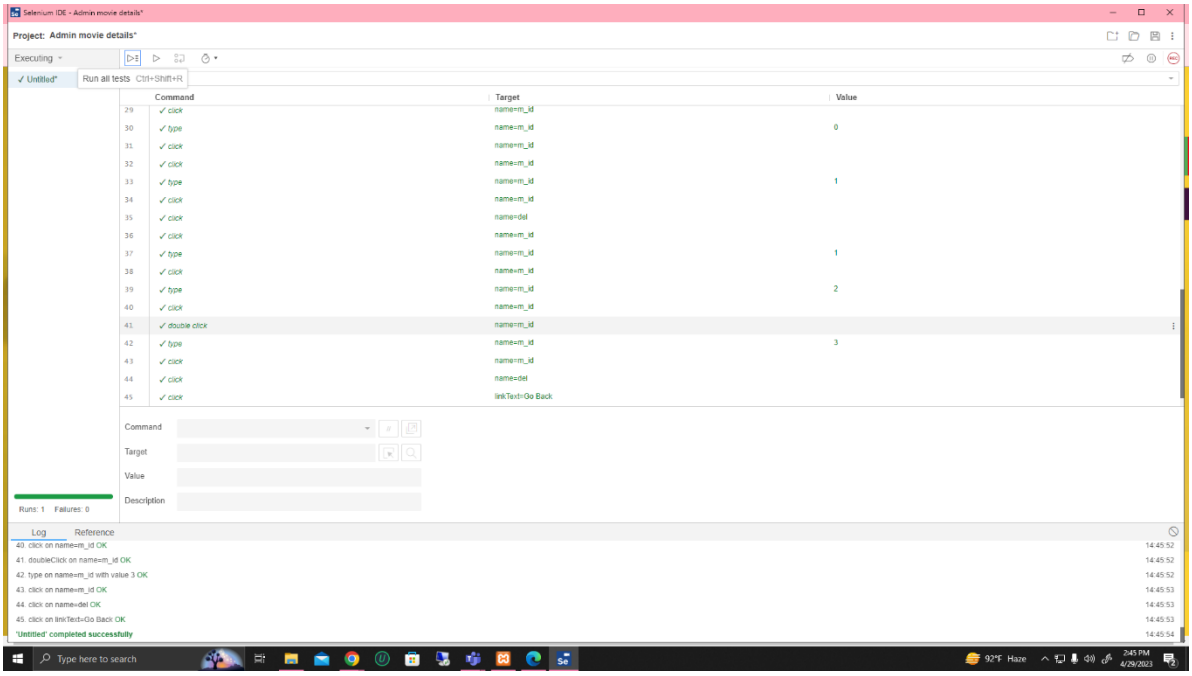


Figure: 6.2(1) admin movie details test completed successfully

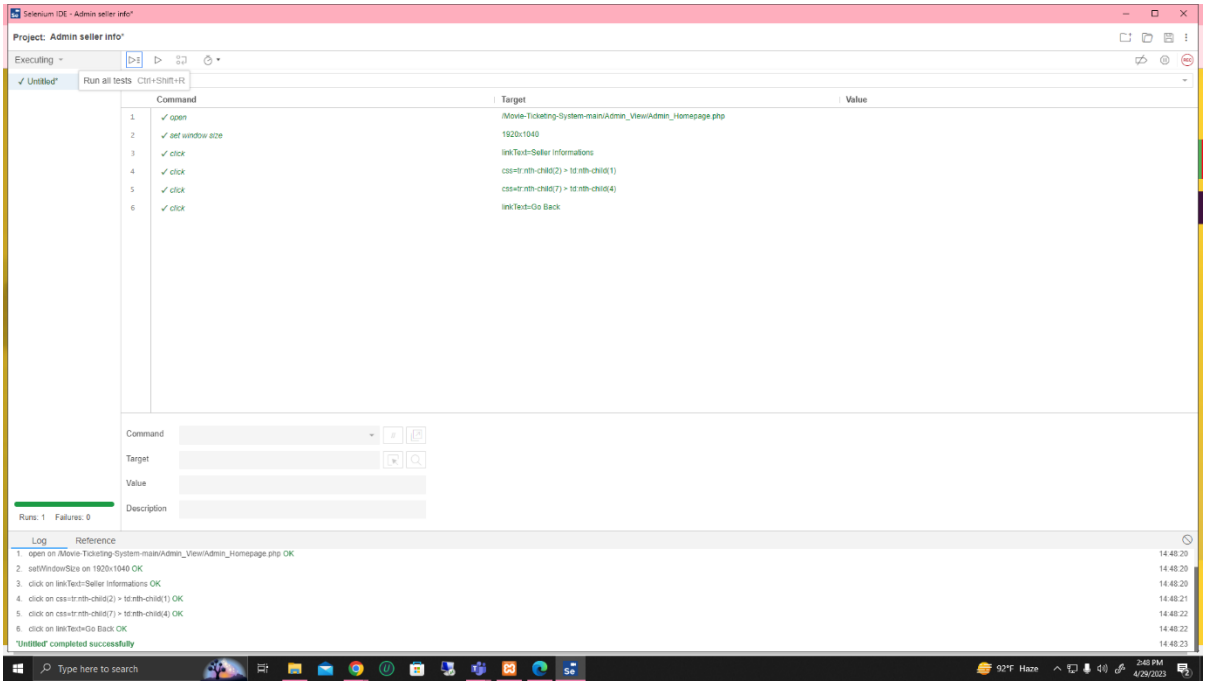


Figure: 6.2(2) admin seller info test completed successfully

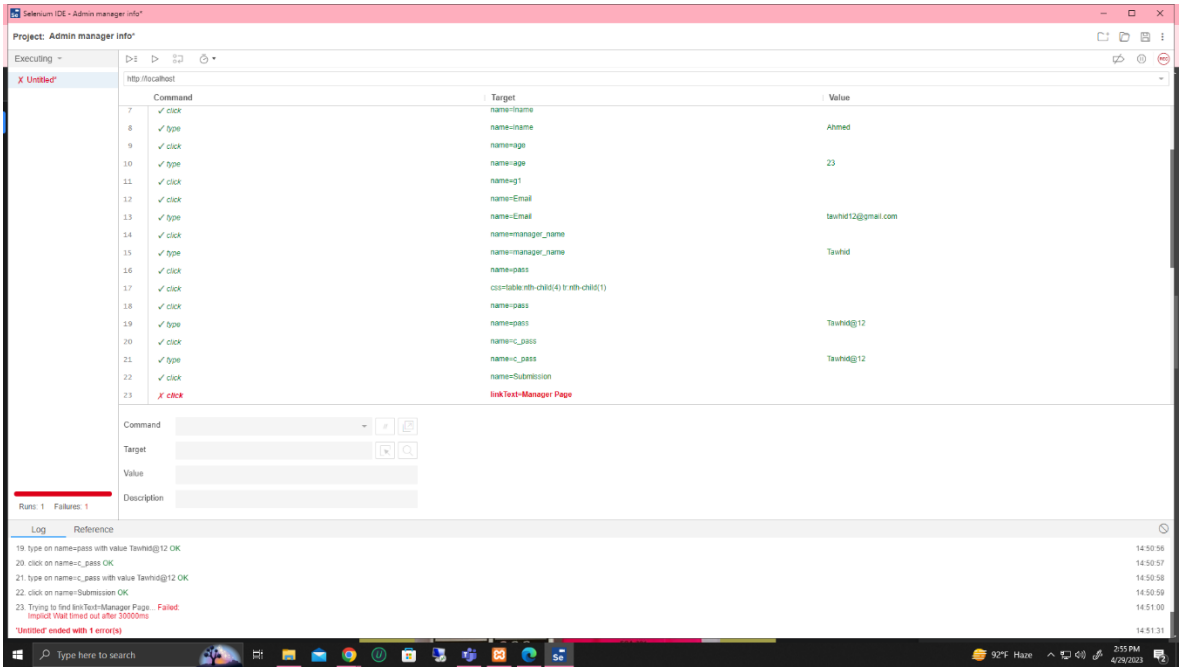


Figure: 6.2(3) admin manager info test failed

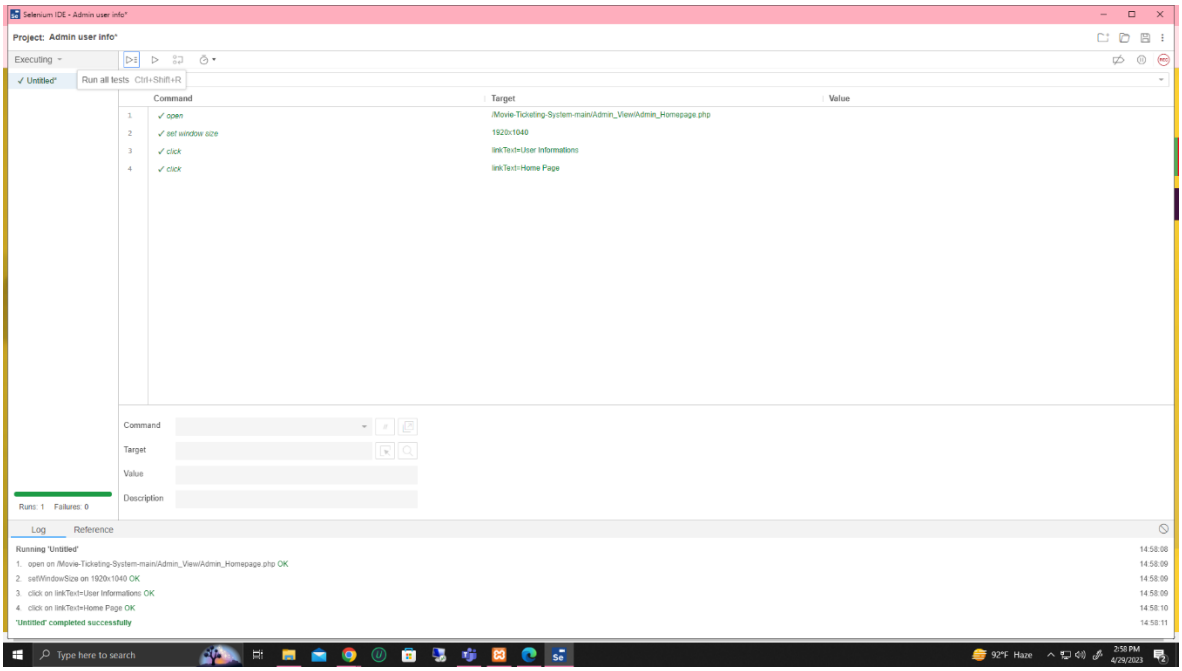


Figure: 6.2(4) admin user info test successfully completed

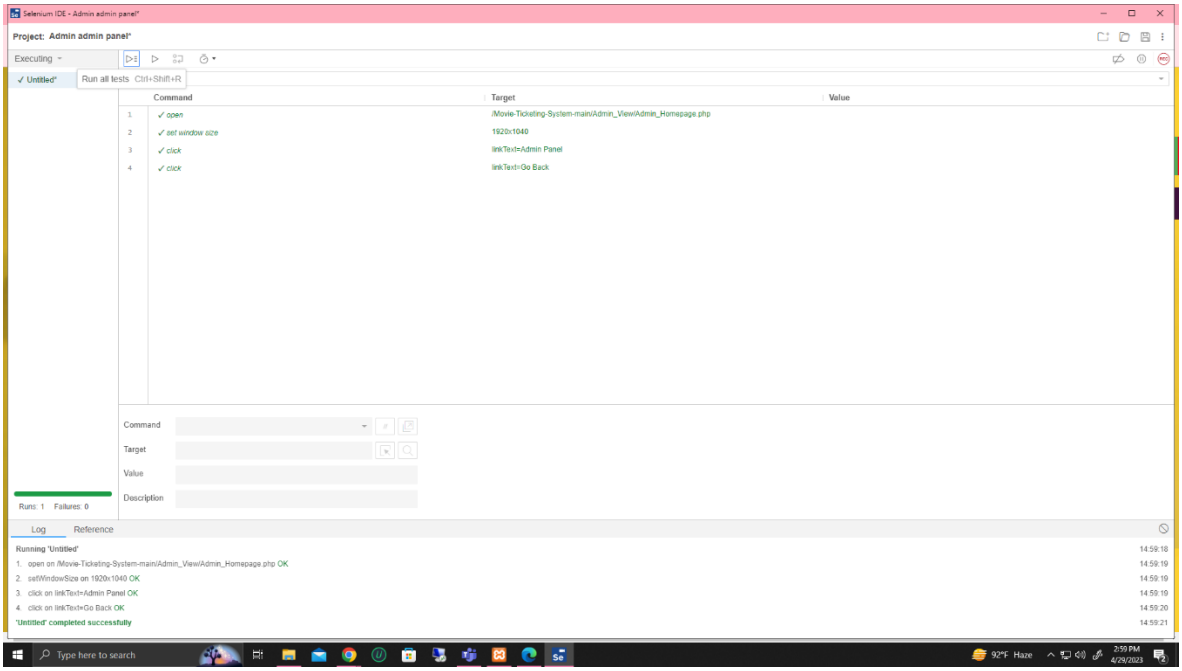


Figure: 6.2(5) admin test admit panel successfully completed

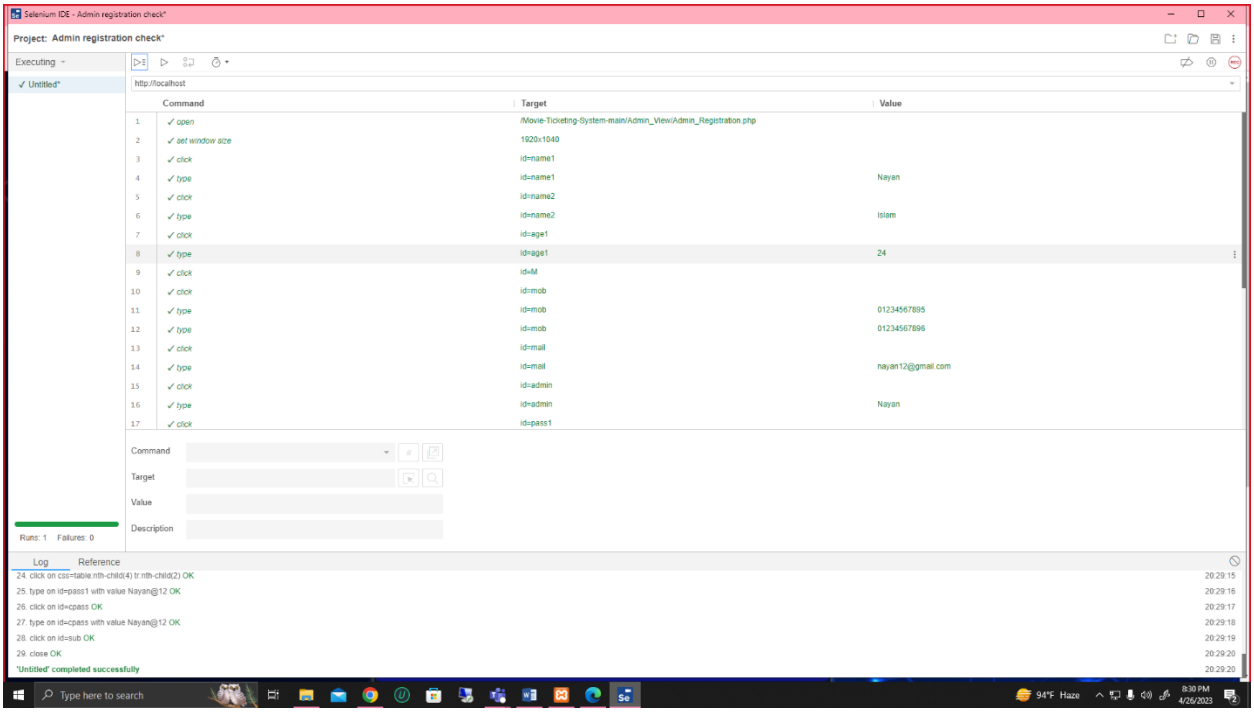


Figure: 6.2(6) admin registration test cheek successfully completed

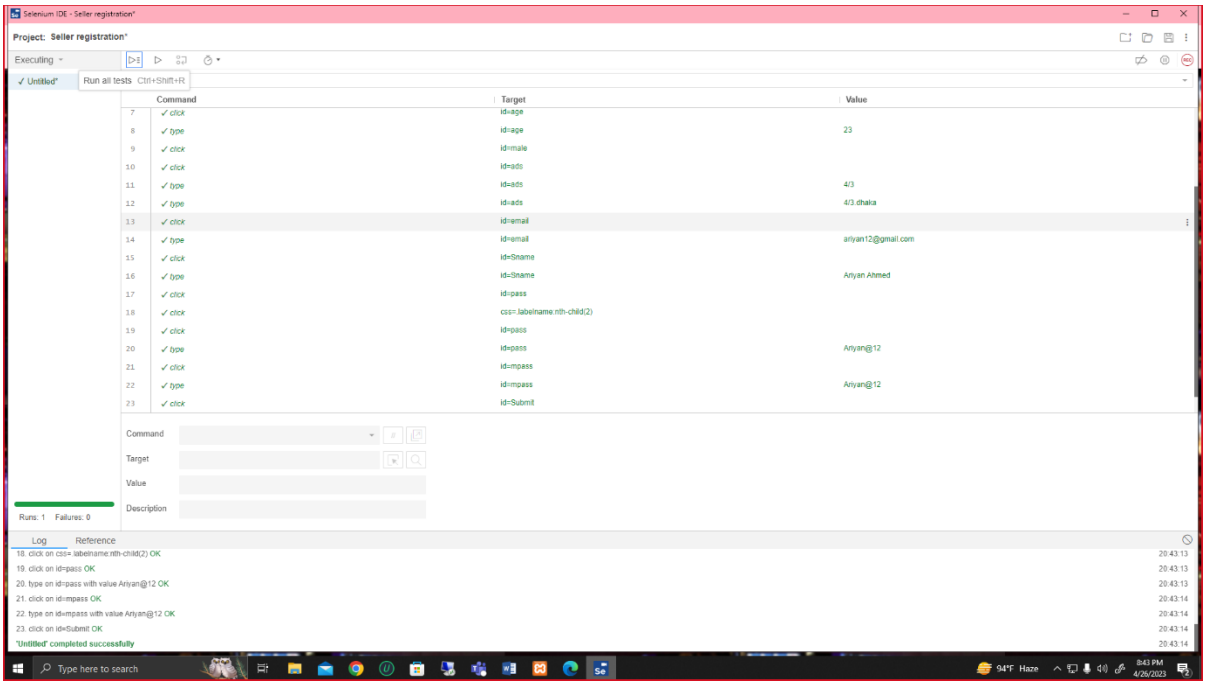


Figure: 6.2(7) seller registration test cheek successfully completed

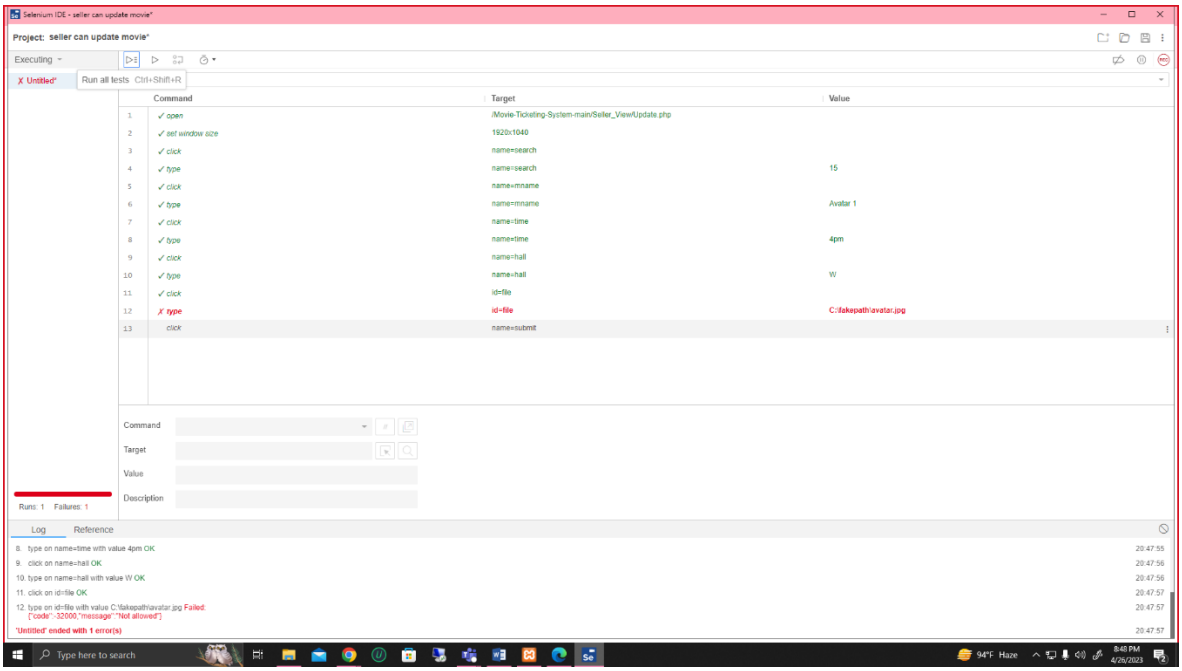


Figure: 6.2(8) seller update movie and hall name test failed

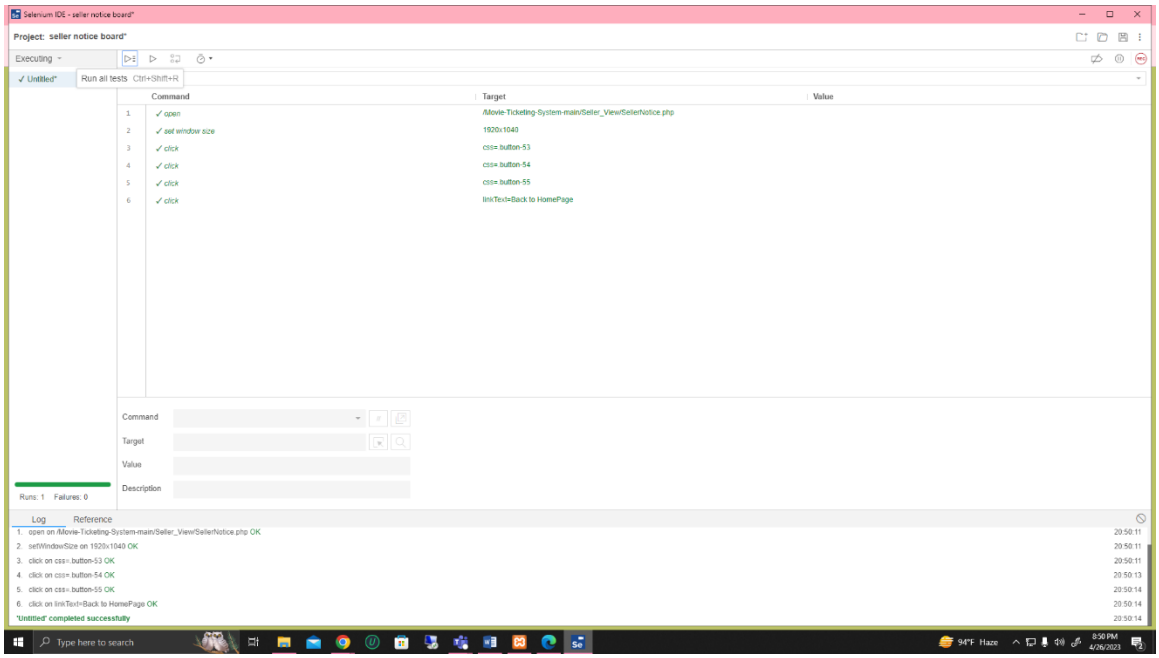


Figure: 6.2(9) seller notice board test successfully completed

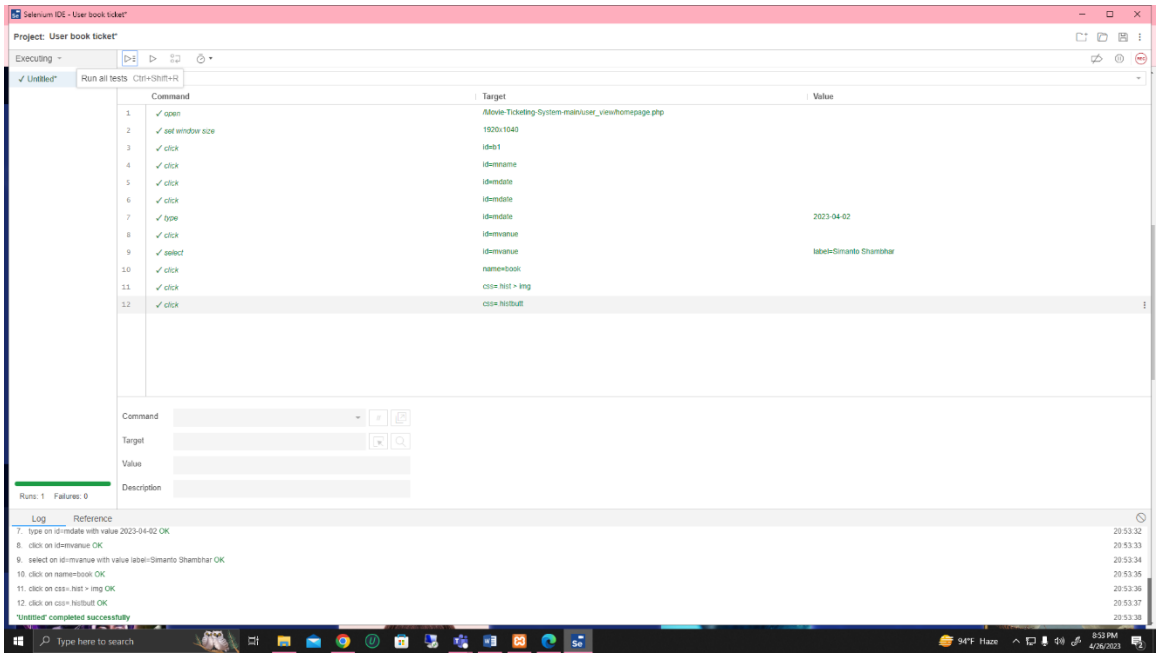


Figure: 6.2(10) user booking ticket test successfully completed

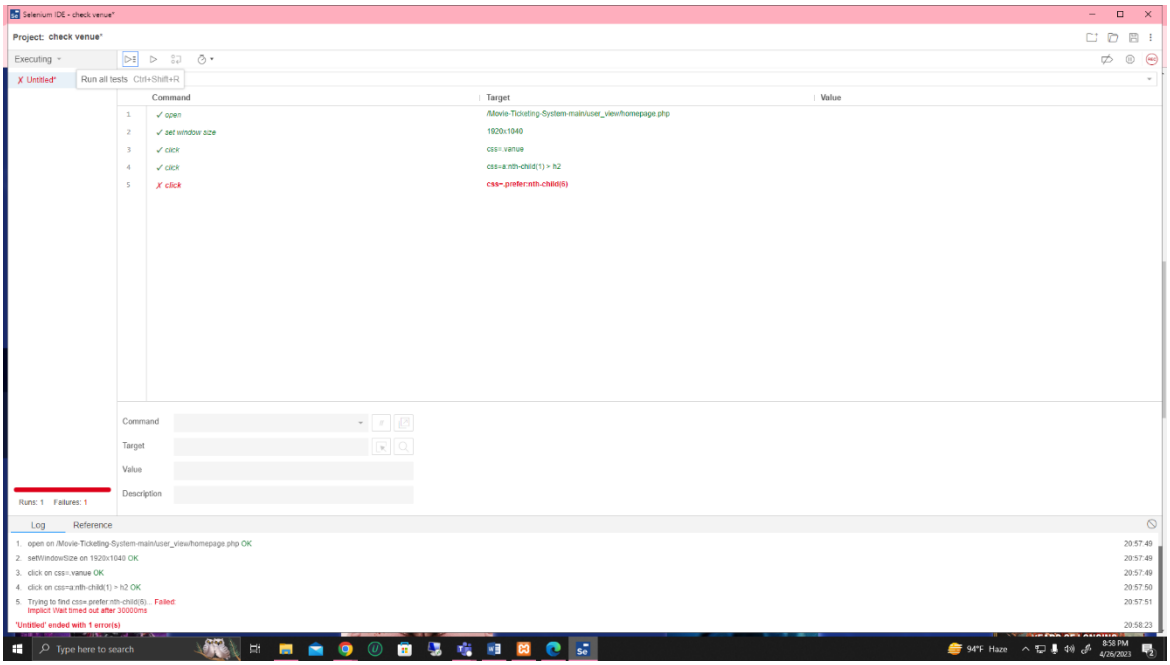


Figure: 6.2(11) user venue cheek test failed

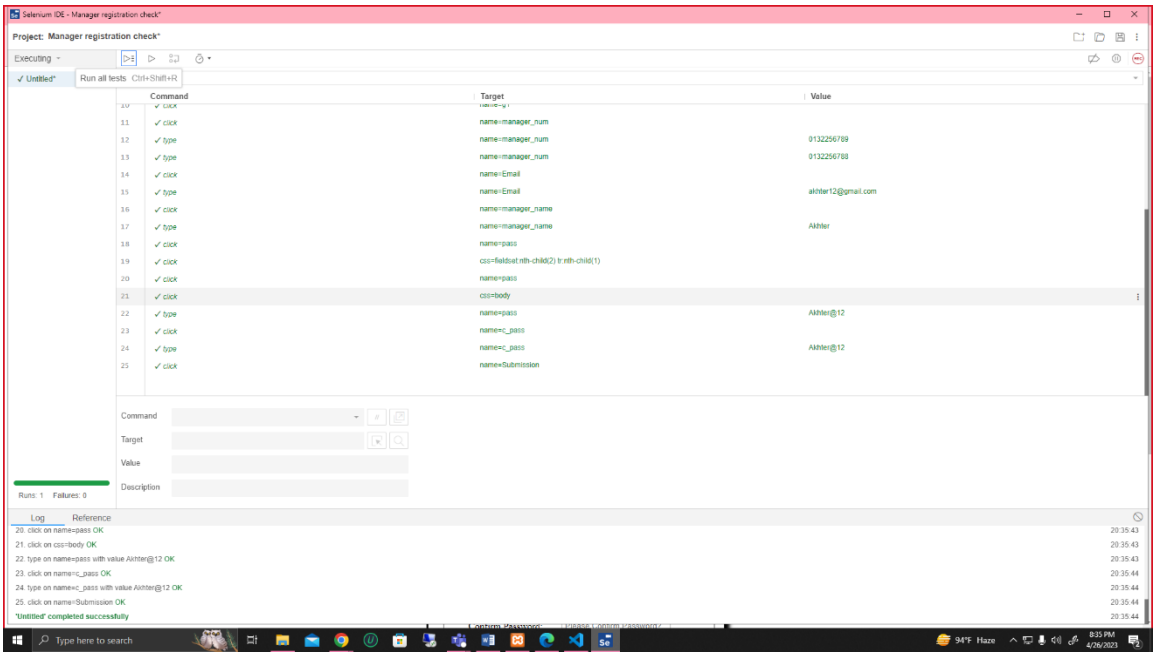


Figure: 6.2(12) manager registration cheek test successfully completed

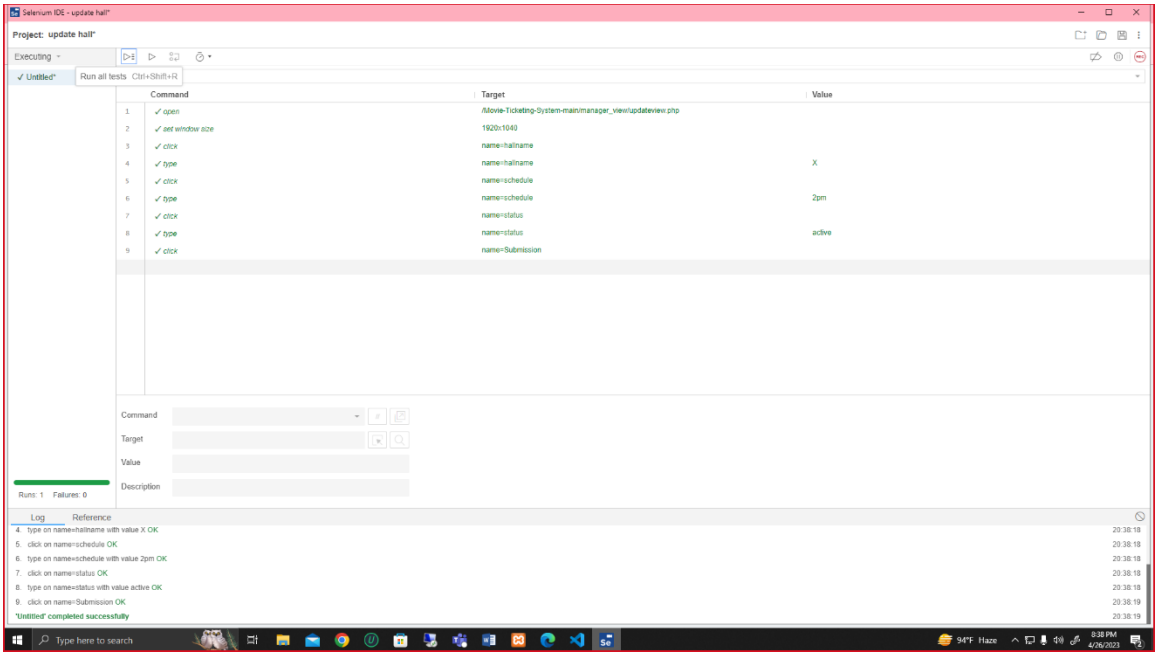


Figure: 6.2(13) manager update hall test successfully completed

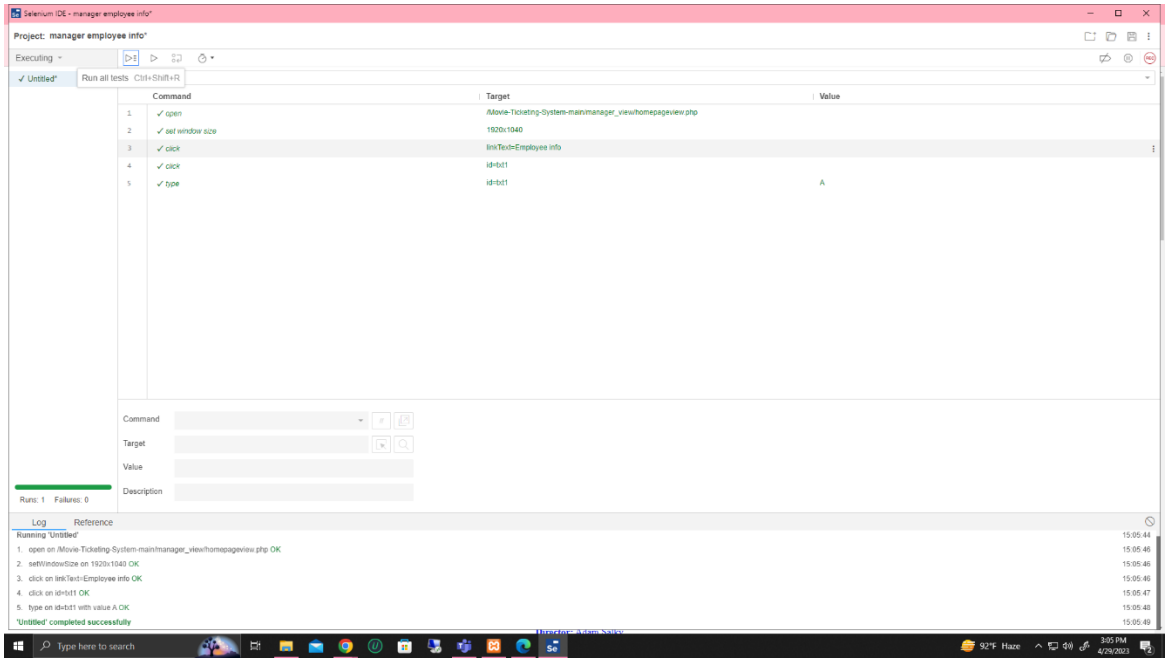


Figure: 6.2(14) manager employee info test successfully completed

6.3 Meetings

To complete a project perfectly work should be distributed to the right person/team. All the members in a team need to understand their work in their selected area. The test team will meet once every week to evaluate progress to date and to identify error trends and problems asearly as possible. The test team leader will meet with development and the project manager once every two weeks. These two meetings will be scheduled in different weeks. Additional meetings can be called as required for emergency situations.

Meeting Topic	**Date**	**Platform**	**Participants**
Initial Design	10/04/23	Google Meet	Design Team, Developers
Planning of Test Cases	20/04/23	Microsoft Teams	Test Manager, SQA Team
Progress Checking	1/05/23	Microsoft Teams	Team Lead
Overview	15/5/23	Google Meet	Developers, Team Lead

7. TEST CASES/TEST ITEMS

7.1 Admin Signup

Project Name: Online Movie Ticketing System			Test Designed by: Mushfiqur Rahman Abir	
Test Case ID: Admin Signup _01			Test Designed date: 30/04/23	
Test Priority (Low, Medium, High): High			Test Executed by: Mushfiqur Rahman Abir	
Module Name: Admin Signup Session			Test Execution date: 2/05/23	
Test Title: Signup with valid username and password				
Description: Test website admin signup page				
Precondition (If any): Admin Cannot use null value in User_Name / Password				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Enter User_Name 4. Enter Password 5.Enter Confirm Password 6. Click submit	User_Name: Tanj Password: 1234** Confirm Password: 1234**	Admin should signup into the application	As expected,	Pass
Post Condition: User is validated with database and successfully signup to account. The account session details are stored in the database.				

7.2 Admin Login

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin Login _02	Test Designed date: 30/04/23

Test Priority (Low, Medium, High): High			Test Executed by: Mushfiqur Rahman Abir	
Module Name: Admin Longin Session			Test Execution date: 2/05/23	
Test Title: Login with valid username and password				
Description: Test website admin Login page				
Precondition (If any): Admin must type valid User_Name / Password which is already stored in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name 5.Enter Password 6. Click submit	User_Name: Tanj Password: 1234**	Admin should login into the application	As expected,	Pass
Post Condition: User is validated with database and successfully login to account. The account session details are stored in the database.				

7.3 Admin Can Remove Manager

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin can Remove Manager _03	Test Designed date: 30/04/23

Test Priority (Low, Medium, High): High			Test Executed by: Mushfiqur Rahman Abir	
Module Name: Admin can remove manager session			Test Execution date: 2/05/23	
Test Title: Remove manager				
Description: Test website admin can remove manager				
Precondition (If any): Admin must have valid 'username' and 'password' saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on manager User List. 9.View all the details. 10.Click on the Delete button. 11.See manager information removed from list	User_Name: Tanj Password: 1234**	Admin can remove manager into the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid User_Name & Password. Then easily modify the data also.				

7.4 **Admin Can Remove seller**

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin can Remove Seller _04	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir

Module Name: Admin can remove seller session			Test Execution date: 2/05/23	
Test Title: Remove seller				
Description: Test website admin can remove seller				
Precondition (If any): Admin must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on seller User List. 9.View all the details. 10.Click on the Delete button. 11.See seller information removed from list	User_Name: Tanj Password: 1234**	Admin can remove seller into the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid User_Name & Password. Then easily modify the data also.				

7.5 Can delete notice and can update notice

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin can delete notice and can update notice _05	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Admin can delete notice and can update notice session	Test Execution date: 2/05/23

Test Title: delete notice and can update notice				
Description: Test website admin can delete notice and update notice				
Precondition (If any): Admin must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on notice board. 9.View all the notices. 10.select notice. 11. click the delete button. 11.See notice removed from board. 12. click notice board. 13. select notice. 14. click update button. 15.see notice updated from board.	User_Name: Tanj Password: 1234**	admin can delete notice and update notice into the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid User_Name & Password. Then easily modify the data also.				

7.6 Admin Can Remove Movie and Update Movie

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin can delete movie and can update movie _06	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Admin can delete movie and can update movie session	Test Execution date: 2/05/23

Test Title: delete movie and can update movie				
Description: Test website admin can delete movie and update movie				
Precondition (If any): Admin must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on movie list. 9.View all the movies. 10.select movie. 11. click the delete button. 11.See movie removed from list. 12. click movie list. 13. select new movie. 14. click update button. 15.see movie updated from list.	User_Name: Tanj Password: 1234**	admin can delete movie and update movie into the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid User_Name & Password. Then easily modify the data also.				

7.7 Admin Can Reset Password

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin can Reset Password _07	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Admin can reset password session	Test Execution date: 2/05/23
Test Title: Reset Password	

Description: Test website admin can reset password				
Precondition (If any): Admin must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on forget password 9.Type new password 10.Re-Type new password 11.click the confirm button. 12.Show Changed password.	User_Name: Tanj Password: 1234**	Admin can reset password to the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid User_Name & Password. Then easily modify the data also.				

7.8 Admin view all data

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Admin view all data _08	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Admin view all data session	Test Execution date: 2/05/23
Test Title: View all data	

Description: Test website admin view all data				
Precondition (If any): Admin must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter Admin home page. 8.click on manager User List. 9.View all the details. 10.Click on customer user list. 11.View all the details.	User_Name: Tanj Password: 1234**	Admin can view All information into the application	As expected,	Pass
Post Condition: Admin can view any time after login with his valid Username & Password.				

7.9 Seller can cancel the ticket

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Seller can cancel the ticket _09	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Seller can cancel the ticket session	Test Execution date: 2/05/23

Test Title: Cancel the ticket				
Description: Test website seller can cancel the ticket				
Precondition (If any): Seller must have valid ‘username’ and ‘password’ saved in database				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Seller Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter seller home page. 8.click on ticket booking List. 9.View all the booked tickets. 10.Click on ticket. 11.click cancel button. 12. canceled ticket.	User_Name: Tanj Password: 1234**	Admin can view All information into the application	As expected,	Pass
Post Condition: Seller can view any time after login with his valid User_Name & Password.				

7.10 Seller Logout

Project Name: Online Movie Ticketing System	Test Designed by: Mushfiqur Rahman Abir
Test Case ID: Seller Logout _10	Test Designed date: 30/04/23
Test Priority (Low, Medium, High): High	Test Executed by: Mushfiqur Rahman Abir
Module Name: Seller Logout Session	Test Execution date: 2/05/23

Test Title: Seller Logout				
Description: Test website Seller Logout page				
Precondition (If any): Seller must have valid ‘username’ and ‘password’ saved in database				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Click on Admin Button 3.Click on Login Button 4.Enter User_Name 5.Enter Password 6. Click submit 7. After this come to the Seller home page. 8. Click on the logout button. 9.successfully logout and come to website home page.	User_Name: Tanj Password: 1234**	Seller logout from the application	As expected,	Pass
Post Condition: Seller is validated with database and successfully login to account and then after completing his task can log out from the website.				

7.11 manager Can add or remove customer

Test Case ID: Manager can add or remove customer_11			Test Designed date: 30/04/23	
Test Priority (Low, Medium, High): High			Test Executed by: Mushfiqur Rahman Abir	
Module Name: Manager can add or remove customer session			Test Execution date: 2/05/23	
Test Title: add or remove customer				
Description: Test website Manager can add or remove customers.				
Precondition (If any): Manager must have valid ‘username’ and ‘password’ saved in database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1. Go to the website 2. Click on manager Button 3.Click on Login Button 4.Enter User_Name. 5.Enter Password 6. Click submit 7. Successfully login and enter manager home page. 8.click on customer list. 9.View all the customers. 10.select customer. 11. click remove button. 11.See customer removed from list. 12. Again Click customer list. 13. select add customer. 14. click update button. 15.see customer updated from list.	User_Name: Tanj Password: 1234**	manager can remove customer and add customer into the application	As expected,	Pass
Post Condition: Manager can view any time after login with his valid User_Name & Password. Then easily modify the customer list.				

8. ITEM PASS/FAIL CRITERIA

After testing the test cases, we can say that 100% testing is impossible. Here we have tested 14 test cases. From this we can see that 11 test cases were tested successfully and the rest of the 3 cases failed. After checking the problem, we can say that there were some database issues. At last, we can measure that 82.70% of test cases were successful and 17.30% of test cases were failed.

9. TEST DELIVERABLES

- Test plan document
- Test design standard

- Test case document
- Test procedures
- Test logs
- Test execution report
- Bug report
- Test closure report
- Release notes

10. STAFFING AND TRAINING NEEDS

For completing this project, we need a Project Manager, Programmer & Test Engineer.

Project Manager: In this project we will approve all the test plans. He is responsible for the planning, executing and completion of this project. He decided about all the budgets, working tools and other necessary staff are needed for the complete project. At least three years of experience is needed to become e project manager.

Programmer: To create a perfect project programmers need to be very sincere about their responsibilities. Programmer can design code, implement code, modify code, etc. Programmer also need at least three years of working experience in the IT industry. Programmers also need to know about different programming languages, and it can be different in different companies as their working requirements.

Test Engineer: Test engineer checks all the programs which were written by programmers. They work on a system to ensure that clients get a high-quality product. They also need to know the basic idea about programming because when they are trying to manual test, they might need to check everything but for dynamic it is different.

10.1 Training Needs:

1. **Project Management Fundamentals**: Since the Project Manager is responsible for planning, executing, and completing the project, a comprehensive understanding of project management principles, methodologies, and tools is essential.

2. **Budgeting and Financial Management:** Training in budgeting and financial management will enable the Project Manager to effectively allocate resources and manage project costs.

3. **Leadership and Team Management**: Enhancing leadership and team management skills will aid in guiding the project team towards successful completion.

4**. Communication and Stakeholder Management**: Training in effective communication and stakeholder management will help in maintaining clear lines of communication and ensuring project alignment with stakeholders' expectations.

5. **Version Control Systems**: Familiarity with version control systems like Git will enable programmers to collaborate seamlessly and track code changes.

6. **Software Design Principles**: Understanding software design principles will empower programmers to create well-structured and scalable solutions.

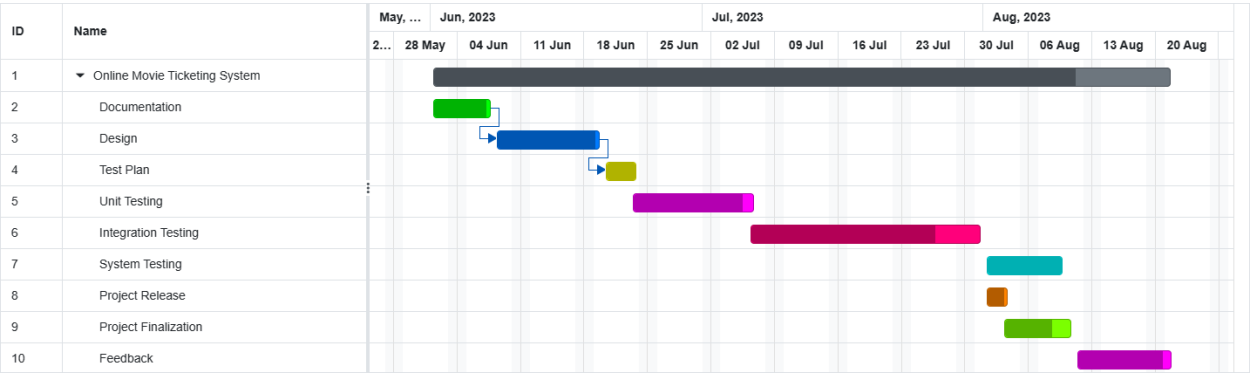
7. **Problem-Solving and Debugging**: Training in problem-solving and debugging techniques will help programmers identify and resolve issues efficiently.

11. RESPONSIBILITIES

	TM	PM	Dev Team	Test Team	Client
Acceptance test Documentation & Execution	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	X	X
Detail Design Reviews	X	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	X

12. 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. Schedule must be done using any PM tool.



13. PLANNING RISKS AND CONTINGENCIES

- 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.

14. APROVALS

Project Sponsor	Mushfiqur Rahman Abir
Development Management	Mushfiqur Rahman Abir
EDI Project Manager	Sharmin Zaker Zerin
RS Test Manager	Tasnim Binta Hossain Shakal
ES Development Team Manager	Marina Afroj