

Test Plan

for

Students' Auditorium Management Software (SAMS)

Version 1.0 approved

Prepared by

**Rushil Venkateswar (20CS30045)
Rahul Mandal (20CS30039)
Sourabh Soumyakanta Das (20CS30051)**

Indian Institute of Technology, Kharagpur

22nd March 2022

1. Introduction.....	1
2. References.....	1
3. Test Items.....	1
4. Software Risk Issues.....	1
5. Features to be Tested.....	1
6. Features not to be Tested.....	2
7. Approach.....	2
8. Item Pass/Fail Criteria.....	2
9. Suspension Criteria and Resumption Requirements	3
10. Test Deliverables	3
11. Environmental Needs	3
12. Schedule.....	3

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

This is the Master Test Plan for the Students' Auditorium Management System. The testing strategy for the software has been described in this document, with the objective being communication of project-wide quality standards and procedures. All components of the software, including all classes and their functions, will be addressed here.

The project will have three levels of testing, **Unit, System/Integration & Acceptance**. The details are mentioned in this document on the following pages.

The estimated time line for this project is 4 weeks, including the testing phase. Any delays would result in features not getting implemented and testing getting adversely affected. Throughout the testing process, the test documentation specifications described in the IEEE Standard 829 for Software Test Documentation will be applied.

2. References

This document is a follow-up on the SRS Document submitted previously which contained the class and function definitions and implementation details. Other references include referring to internet in general.

3. Test Items

1. Salesperson :: Ticket Booking and Printing/Emailing Digital Ticket
2. Show Manager :: Create Salesperson
3. Show Manager :: Create Shows and Seats
4. Show Manager :: Display Seat Occupancy details
5. Show Manager :: Display Transaction Records for all Salespersons
6. Show Manager :: Print the Balance Sheet
7. Seat Availability
8. Salesperson :: Cancel Ticket
9. Accounts Clerk :: Add Expenses to Ledger
10. Login

4. Software Risk Issues

1. In case of a system crash, the software should backup the database file so that the current session is saved to disk.
2. Secure handling of data is a necessity, therefore, the software should allow secure access to its database to the two entities handling payment information.
3. The software should backup login ID and Passwords.

5. Features to be Tested

1. Login screen. Handling of login failure and login success to be tested.
2. Ticket booking and Printing/ Emailing Digital Ticket.
3. Display Seat Occupancy details.
4. Display Transaction Records for all Salespeople.

5. Print the Balance Sheet.
6. Seat Availability.
7. Add Expenses to Ledger of Auditorium.
8. Cancellation of Ticket.

6. Features not to be Tested

1. OTP Verification: Not tested separately as System Testing would cover it.
2. Creation of different entities: This is not tested separately, because System Testing would cover it.
3. Accessing and Backing Up Database files: This has not been tested separately as System Testing would ensure that it is tested.

7. Approach

The test plan for Students' Auditorium Management System would consist of Unit, System, Use-Case and Acceptance Testing.

Unit Testing will consist of Black Box and White Box testing. Black Box testing would consist of high-level behavioral tests to check if the software works as envisioned. The tests for Black Box testing would consist of descriptions of tests in simple English with the output being what the observation would be. White Box testing would consist of low-level tests bypassing the User Interface, and checking how the backend of the software works. It requires knowledge of code base and can be automated using certain software.

System Testing will consist of Performance Testing, Functional Validation Testing. In performance testing we will try to create an event with a large number of seats (possibly employing multiple auditoriums) to see how much time it takes to store/retrieve information about that show from the database. We will also try to book those many seats to see how fast the software can handle storing and accessing information from the database. In functional validation testing we would check if each mouse click on the front end of the website corresponds to the correct action in backend. The main motive here is to check the interaction between the backend and the frontend.

Use-Case Testing would consist of verifying the usability of the software by different classes of people namely Salesperson, Audit Clerk, Spectator and Show Manager. Their login process and all the functionality present with each class would be tested here.

Acceptance Testing would be performed for the end-user by actual people completely unfamiliar to the software.

8. Item Pass/Fail Criteria

Pass Criteria

The criteria is fulfilled if the software performs all the operations correctly and without delay.

Fail Criteria

- The software is unable to load pages properly or the pages take too much time to load.
- Buttons on the page do not work as expected.

- The software is unable to render templates as expected.

9. Suspension Criteria and Resumption Requirements

Suspension Criteria

A. Software is unable to authorize salesperson, manager or audit clerk.

The software will not be usable if it is unable to authorize the manager and salesperson. As Only a manager can perform all the internal operations. And if software cannot authorize salesperson, then no tickets can be booked for spectators.

Resumption Requirements

- A.** It could not be solved until the bug is identified and fixed.

10. Test Deliverables

- Test specification document
- Test plan documentation
- Test cases and test data

11. Environmental Needs

- Any operating system with a modern web browser of latest version.
- Access to SQL database management

12. Schedule

For all the testing activities, time has been allocated within the project plan. The time for each activity is provided in the project plan timeline.

- A. Team personnel along with other team members review requirement document, and initial creation of Inventory and classes, sub-classes and objectives
- B. Development of Master test plan by test manager and test with time allocated for at least two reviews of plan.
- C. Review of the System design document by test team personnel. This will provide the team with a clearer understanding of the application structure and will further define the Inventory classes, sub-classes and objectives.
- D. Development of system and acceptance test plans
- E. Unit test time within the development process.
- F. Time allocated for both System/Integration and Acceptance test processes.