Software Requirements Specification

For

Student Auditorium Management System (SAMS)

Version 1.0 approved

Prepared by

Aditi Mishra (19CS10015)

Yash Agarwal (19CS10066)

Monal Prasad (19CS30030)

Submitted On

20 March 2021

1. Introduction **3**
1.1 Purpose 3
1.2 Intended Audience and Reading Suggestions 3
1.3 Product Scope 3
1.4 References 3

2. Overall Description **4**
2.1 Product Perspective 4
2.2 Product Functions 4
2.3 User Classes and Characteristics 5
2.4 Operating Environment 5
2.5 Design and Implementation Constraints 5
2.6 Assumptions and Dependencies 6

3. External Interface Requirements **6**
3.1 User Interfaces 6
3.2 Hardware Interfaces 6
3.3 Software Interfaces 6

4. System Features **6**
4.1 Show Manager - Create Logins for Salespersons 6
4.2 Show Manager - Query for Seat Occupancy details 7
4.3 Show Manager - Initialize new shows and seats 7
4.4 Show Manager - Query for auditorium availability 8
4.5 Show Manager - View Salesperson Stats 8
4.6 Show Manager - Print Balance Sheet 9
4.7 Spectator or Salesperson - View seat availability for a particular show 9
4.8 Salesperson - Book and print tickets 10
4.9 Salesperson - Cancel tickets 10
4.10 Audit Clerk - Add expenses to ledger 11
4.11 Login 11

5. Other Nonfunctional Requirements **12**
5.1 Performance Requirements 12
5.2 Security Requirements 12
5.3 Software Quality Attributes 12

6. Other Requirements **13**
Appendix A: Glossary 13
Appendix B: Analysis Models 13

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Auditorium Management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the users and the developers of the system.

1.2 Intended Audience and Reading Suggestions

This document is intended for future developers, users and administrators of this system. For users, we recommend you jump to section 4 and find the section that's relevant to you. If you're setting up this system at an auditorium, or want to know the technical requirements, check out sections 3, 2.4, 2.5 and 2.7 (not necessarily in that order). For future developers, we suggest you go through the entire document if you can, or at the very least, section 4 and the use-case and class figures.

1.3 Product Scope

This software provides a systemized portal for booking seats at the Students' Auditorium. The software allows students to quickly find out the available shows, look at the available seats, and book whichever seat they want to sit at. This makes it more systemized, because students can now choose particular seats at the time of buying tickets, instead of buying a ticket and finding space for themselves in the auditorium. This makes the experience smoother for students physically and mentally, and prevents crowding in the alleys before a show starts.

On the back end, this allows smooth scheduling of shows, and instant deployability of new show seats, which means the manager has to spend less time manually scheduling new shows, and conveying this information to the sales people, which results in increased productivity.

This system also minimizes the need for manual accounting and book-keeping, giving more accurate results while cutting down on manpower expenditures.

1.4 References

https://stackoverflow.com/

https://pandas.pydata.org

https://www.python-excel.org

https://www.python.org

2. Overall Description

2.1 Product Perspective

The Auditorium Management System allows booking seats in the auditorium on the frontend, and organizing and scheduling shows on the backend. It also keeps track of transactions and expenses, and can furnish balance sheets as required. It has 4 types of users which interact with it - show managers who run the show, sales persons who make ticket sales, audit clerks who handle auditorium accounts and spectators who can choose seats for the show they want to watch.

2.2 Product Functions

The user logs in as:

- *Show Manager - can schedule shows, manage sales persons, and audit accounts for the auditorium.*
- *Spectator - can view seat/show availability.*
- *Sales Person - can make sales of tickets; receives commission calculated by the system.*
- *Audit Clerk - handles inputting expenses into the system.*

In addition to this, the system creates balance sheets and calculates commissions of sales persons.

2.3 User Classes and Characteristics

The different user classes, along with the functions they carry out are summarized in section 2.2, and also in the use case diagram given below.

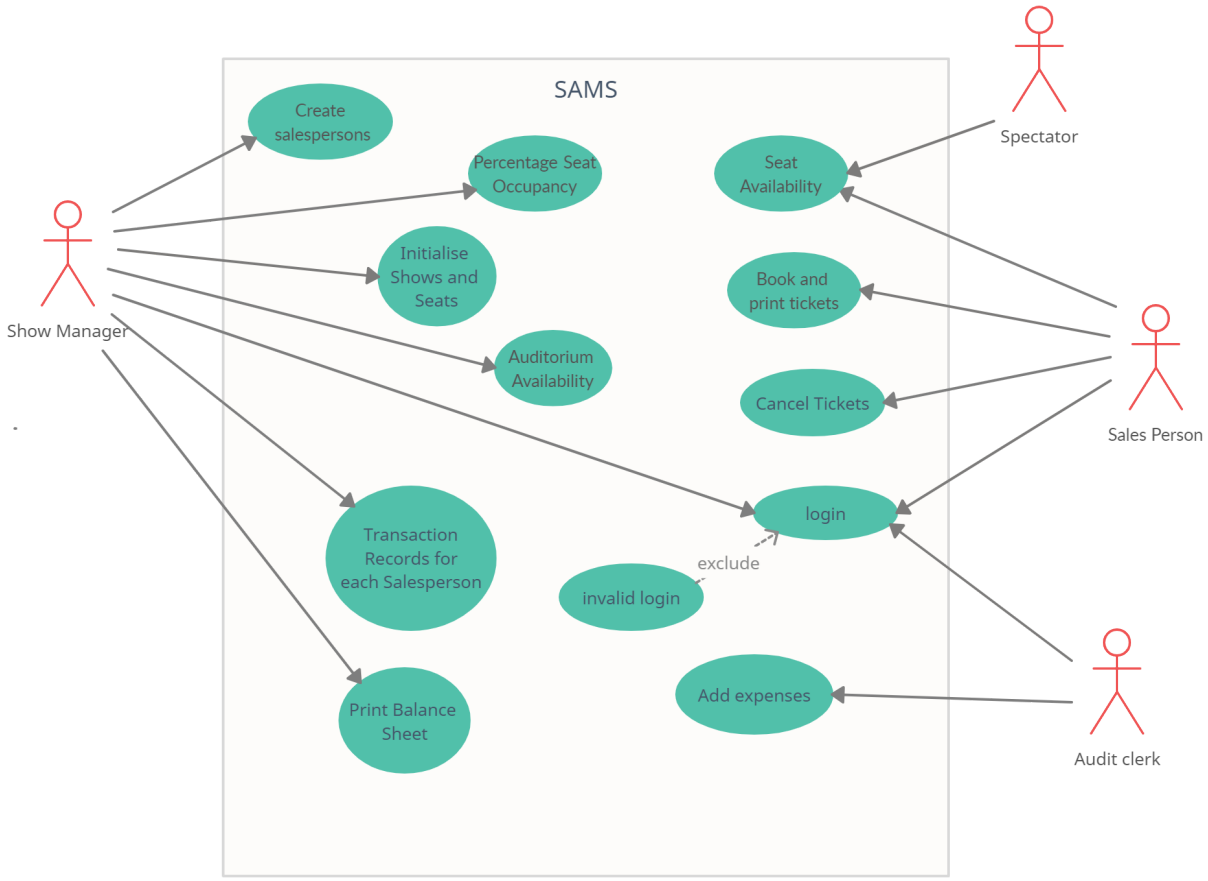

Figure 1: USE CASE DIAGRAM

2.4 Operating Environment

Requires Python3, with Pandas installed. Should run on most modern personal computing hardware. Been tested on a 2018 MBP, running MacOS BigSur, but should work on Windows or other Unix systems with Python3.

2.5 Design and Implementation Constraints

Practically no design or implementation constraints. The pilot version hasn't been designed for actual deployment over the internet, so the absence of a web server might be a constraint for future developers, but this can be easily overcome. Assuming that a minimum of 6000 data points in the xls/csv can be imported and handled efficiently enough, the system can handle data for around 2 months.

2.6 Assumptions and Dependencies

Python3 with Pandas is required for programs in this system to run. The csv/xls files that this program will create must be available to it in later runs, or else the system will need to restart from scratch every time it is run.

3. External Interface Requirements

3.1 User Interfaces

This is a command line program, thus the user interface is text based using inputs from the keyboard. All interaction between all types of users and the system is done through on the console, using text based inputs and outputs. The system also outputs excel files that a user can view after the program is done executing.

3.2 Hardware Interfaces

Since the system is to be run offline on a personal computing device, its only hardware interface is with the device it is being run on (i.e. it's keyboard etc.)

3.3 Software Interfaces

The system uses csv/xls files for maintaining system data across different runs of the program. It also uses the Pandas library for importing these files, and then doing operations on them.

4. System Features

4.1 Show Manager - Create Logins for Salespersons

4.1.1 Description and Priority

Enables the show manager to create login IDs for new salespersons. This is a low priority feature, as this event does not take place too frequently, and can be handled externally by editing the corresponding files.

4.1.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*
- *The system prompts the show manager for User ID*
- *The show manager enters the User ID*

- *The system accepts the User ID if it does not clash with a pre-existing user ID, and prompts the show manager for password*
- *The show manager enters the password*
- *The system accepts and saves these details, ready to be used for a salesperson login.*

4.1.3 Functional Requirements

Unique User ID: The entered user ID must not clash with a pre-existing user ID. If it does, the system returns to the show manager menu, and they would have to select the feature option again to try again.

4.2 Show Manager - Query for Seat Occupancy details

4.2.1 Description and Priority

Enables the show manager to find out the current seat occupancy details for any type of seat for a given show. This is a low priority feature because it is not essential to the auditorium's functioning.

4.2.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*
- *The system prompts the show manager for show name*
- *The show manager enters the show name*
- *The system displays all shows matching this show name, and gives the show manager an option to select one from this list*
- *The show manager selects a particular screening of a show*
- *The system prints the percentage occupancy for each class of seats for this show, along with a few other details about the selected show*

4.2.3 Functional Requirements

Validity of show name: The show name entered must be valid, or else the system prints an empty list to choose from.

4.3 Show Manager - Initialize new shows and seats

4.3.1 Description and Priority

Enables the show manager to schedule new shows. The system automatically checks if this scheduling is possible, given the other pre-existing shows that might clash with it. It also initializes the seating capacities and the seat prices for this show. This is a high priority feature as it is essential for continuous functioning of the auditorium.

4.3.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*

- *The system asks the show manager for the show name, the running time of the show, the seating capacities of each seat category, and the price for each seating category*
- *The show manager enters these details sequentially*
- *The system accepts and returns to the Show Manager Menu.*

4.3.3 Functional Requirements

DateTime Validity - The date time input must be in the format specified, and the starting time must be before the ending time for the show.

Auditorium Availability - The time slot for the new show must not clash with some other show that is already taking place in the auditorium.

4.4 Show Manager - Query for auditorium availability

4.4.1 Description and Priority

Enables the show manager to find out the availability of the auditorium between two datetimes. The system provides its schedule of shows that occur between 2 datetimes. This is a high priority feature as this is essential for quick scheduling of new shows. In the absence of this feature, the show manager has no systematic way to search for shows based on time.

4.4.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*
- *The system asks the show manager for the start and end DateTimes for the search window*
- *The show manager enters these details sequentially*
- *The system finds all shows that are scheduled to take place between these 2 times and prints their details in brief, before returning to the main menu.*

4.4.3 Functional Requirements

DateTime Validity - The date time input must be in the format specified, and the starting time must be before the ending time for the search window.

4.5 Show Manager - View Salesperson Stats

4.5.1 Description and Priority

Enables the show manager to find out what sales each salesperson has made. Medium priority, as this does not directly impact the auditorium's functioning.

4.5.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*
- *The system asks the show manager for the ID of the salesperson*
- *The show manager enters the ID of the salesperson*

- *The system finds the salesperson if the ID is valid and prints all transactions made by this salesperson. It also prints the total revenue collected by this salesperson and the commission they are eligible to receive.*

4.5.3 Functional Requirements

Salesperson ID Validity - The ID entered by the show manager must be a valid ID, or else the system returns to the Show Manager Menu

4.6 Show Manager - Print Balance Sheet

4.6.1 Description and Priority

Enables the show manager to print the auditorium's balance sheet, or list of transactions. This includes all ticket bookings and cancellations, and expenditures input by the audit clerk. Medium priority, as this does not directly impact the auditorium's functioning.

4.6.2 Stimulus/Response Sequences

- *The show manager logs in and navigates to the menu option for this feature.*
- *The system asks the show manager for the start and end DateTimes for the search window*
- *The show manager enters these details sequentially*
- *The system prints the transaction ID, name and value (credit or debit) for each transaction in this time window, before returning to the main menu.*

4.6.3 Functional Requirements

DateTime Validity - The date time input must be in the format specified, and the starting time must be before the ending time for the search window.

4.7 Spectator or Salesperson - View seat availability for a particular show

4.7.1 Description and Priority

Enables spectators and show managers to view the availability of seats for any show of their choosing. High priority as this is essential for customers to be able to make bookings.

4.7.2 Stimulus/Response Sequences

- *The salesperson logs in and navigates to the menu option for this feature.*
- *The system prompts the salesperson for show name*
- *The salesperson enters the show name*
- *The system displays all shows matching this show name, and gives the salesperson an option to select one from this list*
- *The salesperson selects a particular screening of a show*

- *The system prints details about which seats are available to be bought, and details about the price and category of each seat.*

4.7.3 Functional Requirements

Validity of show name: The show name entered must be valid, or else the system prints an empty list to choose from.

4.8 Salesperson - Book and print tickets

4.8.1 Description and Priority

Enables salespeople to make sales and print tickets; the transaction details are all stored automatically. High priority as this is essential for customers to be able to use the auditorium.

4.8.2 Stimulus/Response Sequences

- *The salesperson logs in and navigates to the menu option for this feature.*
- *The system prompts the salesperson for show name*
- *The salesperson enters the show name*
- *The system displays all shows matching this show name, and gives the salesperson an option to select one from this list*
- *The salesperson selects a particular screening of a show, and enters the seat numbers to be booked*
- *The system generates a transaction ID (or a ticket ID) and prints it to the screen, along with the booking details*
- *The system asks for booking confirmation*
- *The salesperson confirms if payment has been received, and rejects otherwise.*

4.8.3 Functional Requirements

Validity of show name: The show name entered must be valid, or else the system prints an empty list to choose from.

Availability of seat: The seat number entered must be available, or else the system forfeits and returns to the salesperson menu.

4.9 Salesperson - Cancel tickets

4.9.1 Description and Priority

Enables salespeople to cancel tickets; the transaction details are all stored automatically. High priority as this is essential for customers to be able to use the auditorium.

4.9.2 Stimulus/Response Sequences

- *The salesperson logs in and navigates to the menu option for this feature.*

- *The system prompts the salesperson for show name*
- *The salesperson enters the show name*
- *The system displays all shows matching this show name, and gives the salesperson an option to select one from this list*
- *The salesperson selects a particular screening of a show, and enters the seat numbers to be cancelled*
- *The system finds the transaction ID, and prints the transaction details with the amount to be refunded to the screen*
- *The salesperson issues the refund*

4.9.3 Functional Requirements

Validity of show name: The show name entered must be valid, or else the system prints an empty list to choose from.

Unavailability of seat: The seat number entered must be booked, or else the system forfeits and returns to the salesperson menu.

Within the cancellation window: The cancellation must be done more than 3 days before the show screening, and only then does the system offer a refund.

4.10 Audit Clerk - Add expenses to ledger

4.10.1 Description and Priority

Enables audit clerks to add expenses to the auditorium ledger. Medium priority as this will not cause immediate problems in case of failure.

4.10.2 Stimulus/Response Sequences

- *The audit clerk logs in and navigates to the menu option for this feature.*
- *The system prompts the salesperson for expense details - name and value (credit or debit)*
- *The audit clerk enters these details sequentially*
- *The system creates a transaction ID for this expense, and adds this to the ledger.*
- *A success message is printed.*

4.10.3 Functional Requirements

Valid Login: The audit clerk must be able to login with the correct details.

4.11 Login

4.11.1 Description and Priority

Enables the different types of users to log in and start using the system. HIGH priority as the system cannot be used without being logged in.

4.11.2 Stimulus/Response Sequences

- *The system asks for what class of users the user belongs to - show managers, salespersons or audit clerks*
- *The user selects one of the three*
- *The system asks for login ID and password*
- *The user enters these sequentially*
- *The system verifies these details, and starts up if these details are correct.*

4.11.3 Functional Requirements

Valid Login: The user must enter valid details.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 Response Time

The software returns the desired output for different users in different amounts of time,for eg- it will take more time for the manager to generate the balance sheets than a spectator to check the seat availability.

5.1.2 System Dependability

It determines the fault tolerance of the system. If the system gets some strange or invalid input or the system faces any random failure, then the user must be informed about it.

5.1.3 Prominent Results

The result displayed must be prominent and the output in the form of .csv/.xls files must be clear and unambiguous.

5.2 Security Requirements

5.2.1 User Profile should be Unique

Any user account registered for the software should be unique and no fake accounts should be present. Only one user account per User ID should be allowed.

5.2.2 Secure Login

The system should be secure from malicious or forced login to access the software.

5.3 Software Quality Attributes

5.3.1 Reliability

The system should be reliable i.e. it should consistently work as required without failing so often that the auditorium cannot function.

5.3.2 File Accessibility

The application must be allowed to create, view and modify files on the system to store transaction and login details across multiple runs of the system.

5.3.3 Maintainability

The application should be easy to extend. The code should be written in such a way that it favors the implementation of new functions.

5.3.4 Portability

The application should be portable to different platforms i.e. it should be adaptable in different platforms.

6. Other Requirements

Appendix A: Glossary

We have used defined terms intuitively across the project, so we won't define terms separately. A few terms like the different user classes may need defining for someone who's new to this software, but they too have been more or less defined, in terms of what functionality the system offers each class of user.

Appendix B: Analysis Models

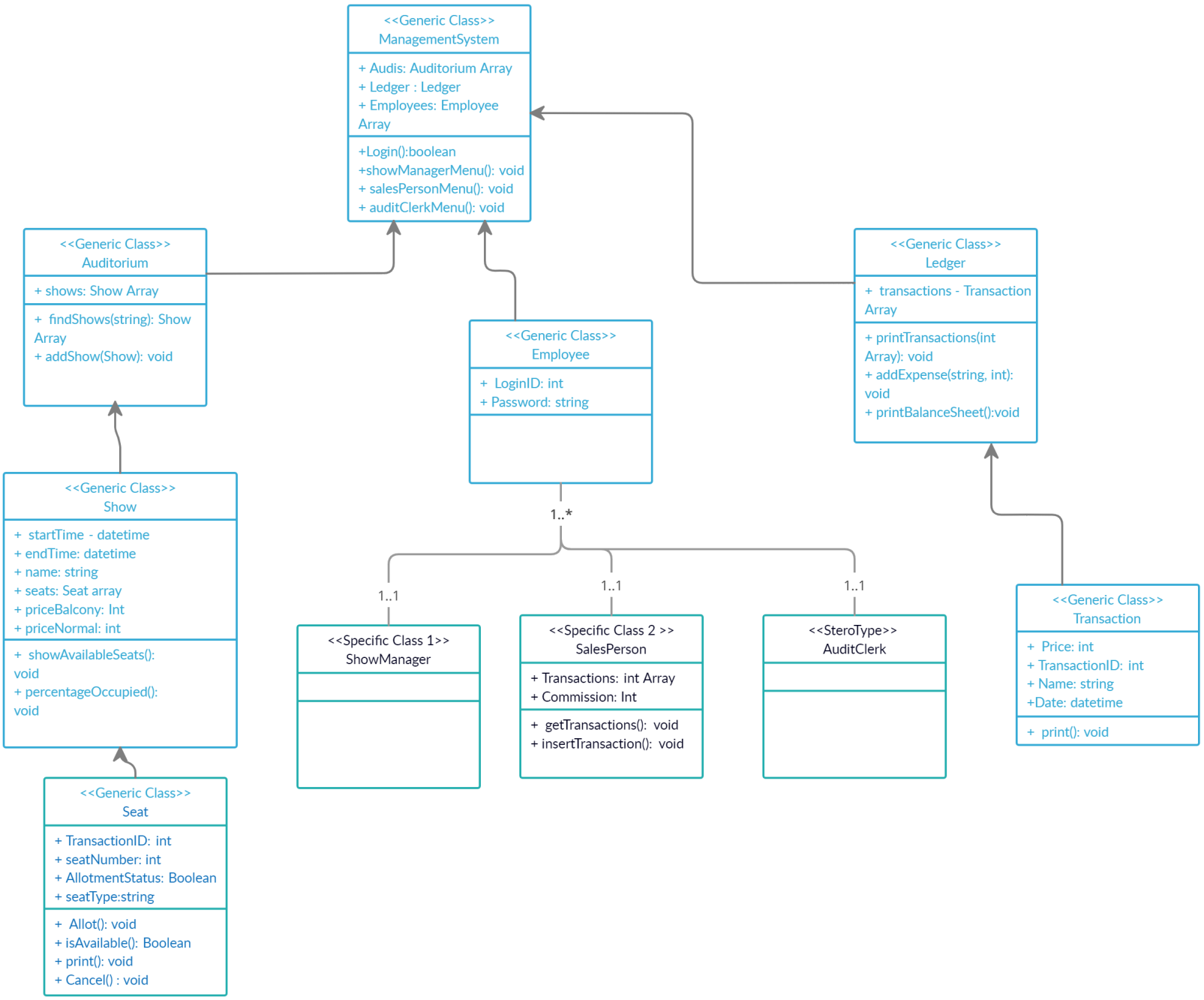

Figure 2: Class diagram