



University of New York Tirana

Develop a Theater Web Application

Professor: Elton Ballhysa

Department of Computer Science

Worked:

Denada & Danja Bali

Deadline: 20 February 2023

Contents

Web Application	3
General Info	3
Functionality for Anonymous Users	3
Functionality for Logged Customer	3
API Functionality	3
Technology	3
ER Database Diagram	4
UML Sequence Diagram	5
Difficulties We Ecountered	6

Theater Web Application

We developed a web application using Java Server Faces (JSF) for the web tier and Java Persistence API (JPA) for the data tier. The web application uses Oracle SQL developer as the database and runs on Tomcat as the application server. We used Eclipse as the IDE for development.

We designed the database schema using an Entity Relationship (ER) diagram that defines the relationships between the various entities in the application. The database schema includes entities for theater, accounts, halls, shows, plays, seats, customers, and reservations.

An anonymous user can search for shows on a particular date, check the details of a play, and check the seat occupancy of a particular show. A logged-in customer can perform all the functionality of an anonymous user, as well as check and update their profile page, reserve up to five seats for a particular show, and browse past reservations.

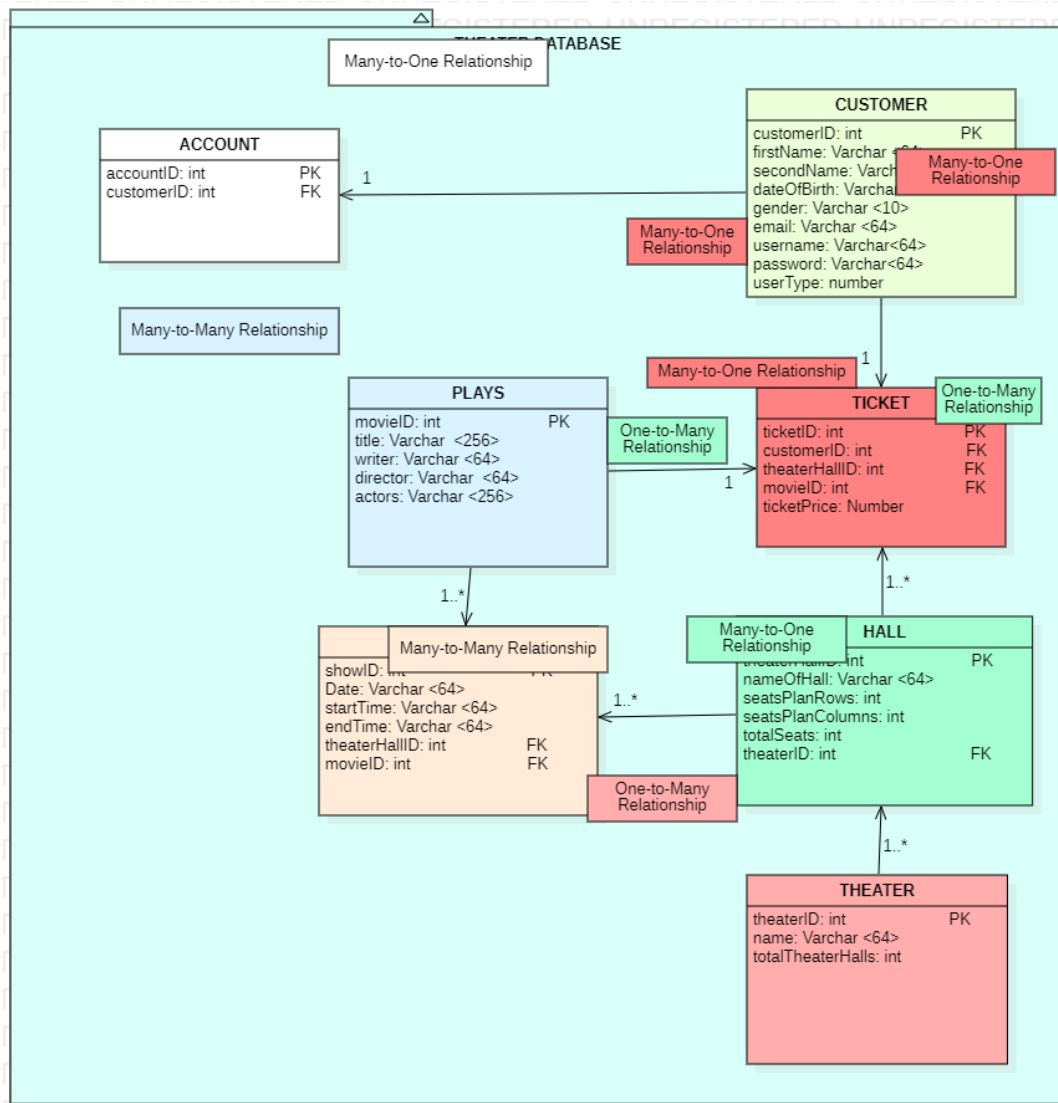
We implemented two theater halls with different seat layouts in the application. Each show includes information about the play, date, time, and theater hall. The seat plan for each theater hall is defined in the database and can be displayed to the user.

The API functionality includes a "Retrieve All Plays" API that returns information about all currently scheduled shows, including the number of occupied and free seats for each show. The "Retrieve Show Details" API returns information about a specific show, including the date, time, play title, writer, director, players, and the occupancy status of each seat.

Conclusion: In conclusion, we developed a B2C Theater Web Application that allows users to manage theater showrooms, seat plans, availability, and reservation process. The application includes an API that returns information about theater shows and seat plans. An anonymous user can search for shows and check seat occupancy, while a logged-in customer can reserve seats and check their past reservations. The application is built using JSF and JPA, and the database is implemented using Oracle SQL.

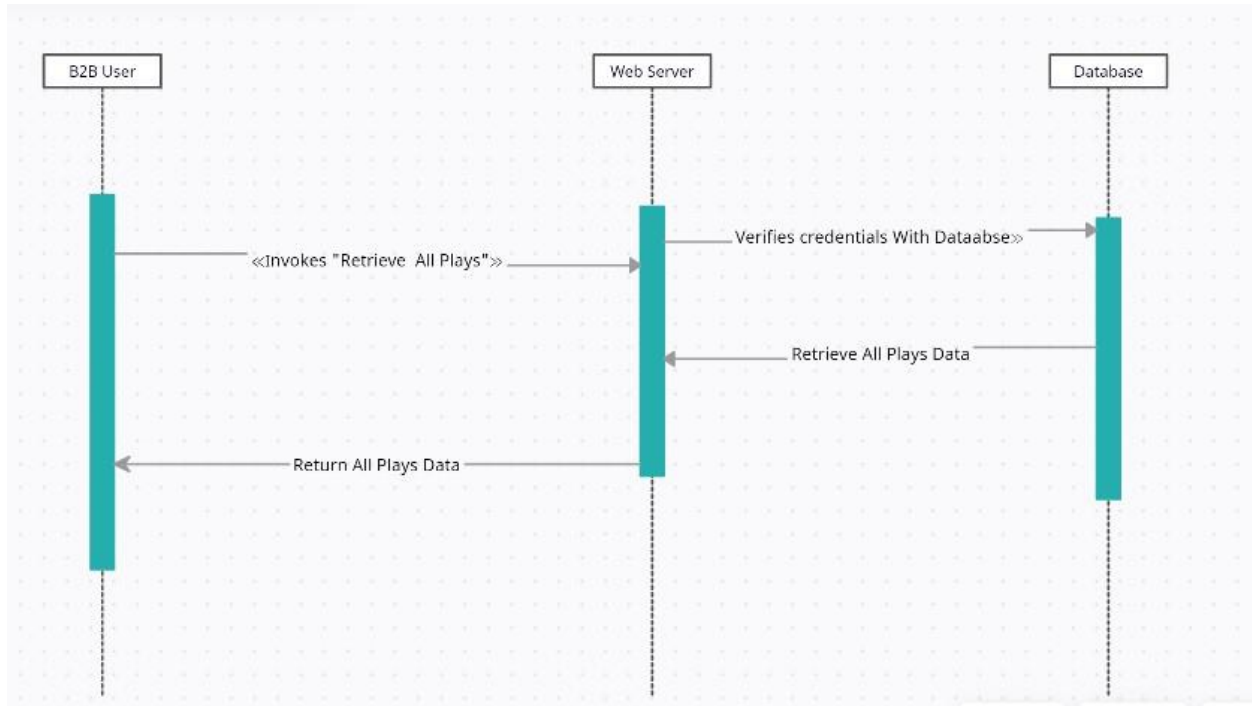
Please see the attached ER diagram for the database schema.

Theater Web Application



Theater Web Application

Please see the attached sequence diagram for API functionalities.



Legend:

1. B2B User invokes "Retrieve All Plays" API with credentials
2. Web Server verifies credentials with Database
3. Web Server retrieves all plays data from Database
4. Web Server returns all plays data to B2B

User Difficulties We Encountered!

During the implementation and development of this assignment, there were several difficulties that we encountered. Some of these difficulties were:

Complexity of the seat layout: Implementing the seat plan and availability functionality was challenging. This required complex solution for checking and reserving seats, as well as rendering the seat map in an intuitive way for the end user.

Security vulnerabilities: Since the application deals with sensitive user data, such as customer login information and reservation details, it is crucial for us to ensure the security of the application. Involving the implementation of a secure login/authentication mechanisms, protecting against database injection attacks, and implementing secure data storage and transmission.

Integration challenges: Integrating different technologies, such as JSF, JPA lead to compatibility issues or unexpected behavior in the begging stages of development. This required a lot of troubleshooting and debugging to ensure that the different components of the application worked together seamlessly.

Scalability and performance: The optimization of database queries and server-side processing to ensure that the application can handle proper volume of users and data.

User interface design: Developing a user-friendly interface required a lot of thought and planning to ensure the interface is easy to navigate and provides all the necessary functionality for both anonymous and logged-in users.

Overall, the successful implementation of this assignment required careful planning, attention to detail, and a lot of testing to ensure that the application is reliable, secure, and easy to use

Theater Web Application

for end users. Lastly, but most importantly, it benefited us by learning new technologies which we are looking forward to improve and work with in our careers.