

CSCI-SHU 213 - Project Part 3

CSCI-SHU 213 - Project Part 3: Web Application

1. Objective

In Part 3, you will build a **secure and usable web application** that implements the Air Ticket Reservation System on top of your Part 2 schema.

The application must:

- Handle authentication and sessions.
- Perform server-side validation of all inputs.
- Support the use cases for public users, customers, booking agents, and airline staff described below.

2. Application Behavior and Use Cases

2.1 Public Users

Public users who are not signed in may:

- Register for a new account (customer, booking agent, or airline staff).
- Sign in with valid credentials.
 - On successful sign-in, create a session and store session variables
 - On failure, report errors clearly.
 - Passwords must be stored as secure hashes, not plain text.
- Search for upcoming flights by origin or destination airport, city, and date.
- Look up flight status for in-progress flights by providing the airline and flight number.

2.2 Customers

After signing in, a customer may:

- Review purchased flights (default view shows upcoming flights) with filters such as date ranges, origin, and destination.
- Search for flights and purchase tickets, with server-side enforcement of capacity and pricing.
- Review spending:
 - Default view: total spending in the last 12 months plus a bar chart for the last 6 months.
 - Custom view: specify a date range, with total spending and a month-by-month bar chart.

2.3 Booking Agents

After signing in, a booking agent may:

- View flights they have purchased on behalf of customers, filtered by date ranges and routes.
- Search for flights and purchase tickets only for airlines they are authorized to represent.
- Access analytics:
 - Commission totals for the last 30 days.
 - Average commission per ticket over the same period.
 - Number of tickets sold.
 - Bar charts showing:
 - Top 5 customers by number of tickets (last 6 months).
 - Top 5 customers by commission (last year).

2.4 Airline Staff

Default (Admin and Operator)

After signing in, an airline staff member sees information for the airline they work for.

- Default view: all flights operated by that airline in the next 30 days, with filters for date ranges and routes.
- Staff can access passenger lists for flights and view all flights taken by a specific customer on their airline.
- Analytics include:
 - Top booking agents by month and year (by tickets and by commission).
 - Most frequent customer in the last year.
 - Tickets sold per month.
 - Delay vs. on-time statistics.
 - Top destinations for the last 3 months and last year.

Admin

Staff with Admin permission can:

- Add new airports and airplanes.
- Create new flights.
- Associate booking agents with the airline.

Operator

Staff with Operator permission can:

- Update the status of flights.

3. Anti-Automation Challenge

Implement **exactly one** of the following twists in your web application. While you may choose any twist, selecting the same one you used in Part 1&2 is highly recommended to ensure consistency across the project stages.

1. **Code-share flights**
2. **Seat classes**
3. **Multi-airport cities and aliases**

4. Security, Integrity, and Implementation

- Server side validation
 - All business rules must be enforced on the server side.
 - Client-side validation may help usability, but server-side checks are mandatory.
- Prevent SQL injection
 - Use prepared statements (if supported) or thoroughly sanitize inputs to prevent SQL injection.
- Use sessions
 - Authenticate and authorize every protected request using sessions.
- UI&Design
 - The user interface should be straightforward and usable.
 - Polished design is welcome but not required.

5. Deliverables

You must submit:

1. **Source code** of the application.
2. A **manifest** listing all files with brief descriptions.
3. A **short document** mapping each user-facing feature to the database queries it issues.
4. (If working in pairs) A **concise summary** of each member's contributions.

Additional requirements for demo:

- For development and testing, running the application locally is acceptable.
- For the demo, you must either:
 - Bring a host machine to the check-off session.
 - Or make the application reachable on the web.
- Be prepared to demonstrate test scenarios and answer questions about your implementation.
- During the check-off session, please ensure that the application is executed in a way that minimizes runtime bugs or can effectively handle them if they occur. **Debugging on-site during the check-off session is not recommended.**

6. Evaluation

Part 3 will be evaluated on:

- Correctness and completeness of implemented features.
 - Security of sessions and input handling.
 - Clarity and usability of the user experience and analytics.
 - Code quality and maintainability.
 - Convincing implementation of the chosen anti-automation challenge.
-