

# **HY359 – Project Documentation**

CSD5185

George Apostolakis

CSD5328

George Papamatthaiakis

# Overview

This project implements a role-based event management platform that supports **public and private events**, secure user authentication, real-time messaging, and AI-assisted content generation.

The system enforces strict access control, ensuring that **authorized users only** can access private data and protected endpoints.

# Project Structure & Security Model

All API calls that require authorization **validate user credentials before execution**.

Unauthorized requests are rejected early in the request lifecycle to preserve data integrity and privacy.

# Flow of Work

## User Requests

- Users may request **private events** only through **available time slots** defined by bands.
- Availability slots are visible to **all registered users**, but are consumed only upon acceptance.
- Once a private event request is **accepted**:
  - The event status changes to **requested**
  - A **Messages** button appears on the event card
  - Only the requesting user can view and interact with that private event

## LLM Integration

- Only **registered users** can access the AI Agent.
- The LLM has controlled access to the database to prevent exposure of private events.
- Guest users are explicitly restricted from AI usage for security reasons.
- The project integrates a **Groq-based LLM**, used for content generation.

# API Routes

## Route Manager

`./app.js` — Server entry point & route manager

## Admin Routes

`./routes/admin.js`

- `/getReviews` — Fetch all user reviews
- `/details` — Verify admin credentials
- `/removeUser` — Remove a user
- `/reviewStatus` — Update review status
- `/bandsPerCity` — Retrieve bands grouped by city
- `/numOfEvents` — Total number of events
- `/numOfUsers` — Total number of registered users
- `/numberOfEarning` — Platform earnings
- `/reviewDeletion` — Delete a review

## Band Routes

`./routes/band.js`

- `/createEvent` — Create a public event
- `/setAvailability` — Define availability slots
- `/removeAvailability` — Remove availability slot
- `/updateRequest` — Accept or reject user requests
- `/updateBand` — Update band credentials
- `/details` — Verify band credentials

# API Routes (continued)

## Database Routes

`./routes/db.js`

- /dropdb
- /insertRecords
- /users
- /initdb

## General Routes

`./routes/general.js`

- /getBands — Fetch all bands
- /seePublicEvents — Fetch public events
- /messageHistory — Retrieve message history
- /sendMessage — Send a message
- /register — Register user or band
- /getPrivateEvents — Fetch private events

## LLM Routes

`./routes/llm.js`

- /generate — Generate content using LLM

## User Routes

`./routes/user.js`

- `/details` — Verify user credentials
- `/seeAvailability` — View band availability
- `/requestBand` — Request a private event
- `/review` — Submit a review (pending status)
- `/update-user` — Update user credentials



# Database Backend

These files handle database initialization, data insertion, role-specific queries, event and review management, and user authentication data:

```
./resource.js
```

```
./database.js
```

```
./databaseInsert.js
```

```
./databaseQueriesAdmin.js
```

```
./databaseQueriesBands.js
```

```
./databaseQueriesBoth.js
```

```
./databaseQueriesEvents.js
```

```
./databaseQueriesUsers.js
```

```
./dbConfig.js
```

# Database Frontend

- `./public/admin.html` — Admin control panel
- `./public/band.html` — Band dashboard
- `./public/bandAddEvent.html` — Event creation
- `./public/bandCommunity.html` — Band community page
- `./public/eventRequest.html` — Private event request form
- `./public/guest.html` — Guest landing page
- `./public/index.html` — Main page
- `./public/llm.html` — AI prompt interface
- `./public/messageIndexBumAss.html` — Messaging interface
- `./public/register.html` — Registration page
- `./public/reviews.html` — Band reviews
- `./public/userReviewPage.html` — User review management
- `./public/userSeesBandsPage.html` — Band listing for users
- `./public/userUpdate.html` — User profile update

# CSS Styling

`./public/CSS/availability.css`

`./public/CSS/bandCards.css`

`./public/CSS/bandFilter.css`

`./public/CSS/llm.css`

`./public/CSS/login.css`

`./public/CSS/message.css`

`./public/CSS/navigation.css`

`./public/CSS/osm.css`

`./public/CSS/publicEventCards.css`

`./public/CSS/reviewCards.css`

`./public/CSS/style.css`

# Frontend JavaScript

These scripts handle API communication, UI rendering, authentication checks, messaging, and input validation:

```
./public/JS/admin.js
```

```
./public/JS/band.js
```

```
./public/JS/button.js
```

```
./public/JS/guest.js
```

```
./public/JS/loginCheck.js
```

```
./public/JS/map.js
```

```
./public/JS/messaging.js
```

```
./public/JS/user.js
```

```
./public/JS/validation.js
```