Architecture and Diagrams (Assignment 4 – Manager Role)

Initial Component Diagram (pre-implementation, MVC-style)

[Browser / Client]

|

v

[Express Router]

■■■ /manager (GET, POST)

|

v

[Manager Controller]

- Loads schedule

- Validates form data

- Sends updates to DB

|

v

[Mongoose Models]

- Schedule

- SongReport

- DJ

|

v

[MongoDB]

Initial Sequence Diagram (pre-code expected flow)

Client -> Router: GET /manager

Router -> Manager Controller: fetch dashboard data

Controller -> MongoDB: get schedule, featured DJ, reports, stats

MongoDB --> Controller: send data

Controller -> EJS View: inject data into template

EJS -> Client: rendered HTML

Schedule Add Flow (pre-code expectation)

Client -> Router: POST /manager/add-schedule

Router -> Manager Controller: validate fields

Controller -> MongoDB: save new schedule entry

MongoDB --> Controller: updated schedule list

Controller --> Client: JSON (updated schedule + stats)

Updated Diagrams (post-implementation)

System Architecture After Implementation

[Client Browser]

|

v

[Express App]

- Middleware: sessions, parsing, static hosting

- /manager routes

|

v

[Manager Route Handlers]

- Add DJ to schedule (POST)

- Remove DJ from schedule (POST)

- Render dashboard (GET)

|

v

[Database Layer]

- Schedule Model

- SongReport Model

- DJ Model

|

v

[MongoDB]

Updated Sequence (actual implemented behavior)

Add-Schedule Flow

Client -> Router: POST /manager/add-schedule

Router -> Manager Controller: validate

Controller -> MongoDB (Schedule): insert new DJ slot

MongoDB --> Controller: return updated schedule list

Controller -> View Logic (JSON): {schedule, stats}

View JS --> Client DOM: update table, featured DJ, stats

Remove-Schedule Flow

Client -> Router: POST /manager/remove-schedule

Router -> Manager Controller: delete slot by index

Controller -> MongoDB: save updated schedule array

MongoDB --> Controller: send updated data

Controller --> Client: JSON {schedule, stats}

Client JS: rebuild schedule table + featured section

Song Search Flow

Client -> JS: user types in search bar

JS -> DOM: filter elements

(display only matches)