Shuttle Schedule Viewer System — Project Report

**Student Name:** Nida Sohail  
**Project Type:** Web Application  
**Technologies Used:** HTML, CSS, JavaScript, Node.js, Express, MySQL, MongoDB

Project Overview

The Shuttle Schedule Viewer System is a web-based tool developed to simplify the process of viewing, managing, and updating campus shuttle routes and their schedules. The application is designed for both students and administrators.

Objectives

* Allow users to view all shuttle routes and their respective stops.
* Display accurate weekday, weekend, and special day shuttle timings.
* Provide a secure admin panel for managing shuttle schedules.
* Prevent unauthorized access to the admin dashboard.

System Features

**1. User Interface**

* **Route Selection Dropdown:** Lets users choose a route.
* **Stops Viewer:** Displays the list of stops in correct sequence.
* **Schedule Viewer:** Shows shuttle times for weekdays, weekends, and special days.

**2. Admin Dashboard**

* **Secure Login:** Only registered admins can access admin dashboard .
* **Session Handling:** Admin sessions are protected using express-session.
* **Schedule Editor:** Admins can edit and update schedules in MongoDB.
* **Logout Feature:** Ends session and redirects to homepage securely.

**3. Security**

* Admin login is required to access sensitive pages.
* Sessions are stored server-side and protected by a secret key.
* Direct URL access to admin pages is blocked unless the admin is logged in.

Technologies Used

| **Component** | **Tech Stack** |
| --- | --- |
| Frontend | HTML, CSS, JavaScript |
| Backend | Node.js, Express |
| Database | MySQL (for routes & stops)   MongoDB (for schedules) |
| Session Management | express-session |

Database Design

**MySQL:**

* routes — Stores route IDs and names.
* stops — Contains stop details.
* route\_stops — Links routes and stops with sequence numbers.
* admins — Stores email and password for login.

**MongoDB:**

* schedules — Stores weekday, weekend, and special schedules for each route.

Functional Workflow

1. **User View:**
   * Select a route → View stops and schedule.
2. **Admin Login:**
   * Admin enters email/password → Authenticated via MySQL → Session created.
3. **Admin Actions:**
   * View route stops.
   * Update schedule: weekday, weekend, and special days.
   * Click Save → Updates MongoDB.
   * Logout → Ends session and redirects to home.

Testing & Results

* Tested login session persistence.
* Verified data update flows for schedule saving.
* Prevented unauthorized access to admin panel.
* Ensured real-time schedule visibility after updates.

Conclusion

The Shuttle Schedule Viewer System meets the intended goals of making campus shuttle schedules transparent and editable only by authorized administrators. The integration of both **MySQL** and **MongoDB** demonstrates a hybrid data management approach, allowing structured route data alongside flexible schedule storage