|  |
| --- |
| **Use Case Name:** Create new passenger account |
| **Actor:** Passenger  **Description:**   1. Input user ID, name, email, password & confirm password. 2. Check User ID. (Unique). 3. Store new passenger account into the database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. URL of “Create account” hyperlink page.   **Post Condition:**   1. **Successful**- Create user account. 2. **Unsuccessful**- Stay in the same page, do not create user account & show error message. 3. **Exception**- Stay in the same page if possible. |

|  |
| --- |
| **Use Case Name:** User login to the system |
| **Actor:** User  **Description:**   1. Input user ID & password. 2. Check user ID & password. (Authentication) 3. Save ID and password. (Local Storage) 4. Remember password.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. URL of “Log In” hyperlink page.   **Post Condition:**   1. **Successful**- Log in the user into the system. 2. **Unsuccessful**- Stay in the same page, do not login the user & show error message. 3. **Exception**- Stay in the same page if possible. |

|  |
| --- |
| **Use Case Name:** Passenger Booking a seat |
| **Actor:** Passenger  **Description:**   1. Input Way (Arrival/Departure), Route, Timeslot & Date. 2. Check the data. (Valid) 3. Store new booking into database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Passenger logged in to the system. 2. Charge the cost from next semester’s fee.   **Post Condition:**   1. **Successful**- Book a seat for the passenger. 2. **Unsuccessful**- Stay in the same page, do not book a seat & show error message. 3. **Exception**- Stay in the same page if possible. |

|  |
| --- |
| **Use Case Name:** Checking past travels |
| **Actor:** Passenger  **Description:**   1. Retrieve and show all past travels from database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Passenger logged in to the system.   **Post Condition:**   1. **Successful**- Show data about user’s past travels. 2. **Unsuccessful**- Stay in the same page, do not show any data & show error message. 3. **Exception**- Stay in the same page if possible |

|  |
| --- |
| **Use Case Name:** Seeing number of passengers |
| **Actor:** Admin  **Description:**   1. Input Way (Arrival/Departure), Route, Timeslot & Date. 2. Retrieve, calculate and show the number of passengers from database   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system.   **Post Condition:**   1. **Successful**- Show number of passengers. 2. **Unsuccessful**- Stay in the same page, do not show any data & show error message. 3. **Exception**- Stay in the same page if possible. |

|  |
| --- |
| **Use Case Name:** Adding a vehicle |
| **Actor:** Admin  **Description:**   1. Input brand, model, registration number & vehicle ID. 2. Check vehicle ID and registration number. (Unique) 3. Store a new vehicle into database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system. 2. The vehicle does not exists.   **Post Condition:**   1. **Successful**- Add a vehicle. 2. **Unsuccessful**- Stay in the same page, do not add a vehicle & show error message. 3. **Exception**- Stay in the same page if possible |

|  |
| --- |
| **Use Case Name:** Deleting a vehicle |
| **Actor:** Admin  **Description:**   1. Input vehicle number. 2. Check data. (Valid) 3. Delete the vehicle from database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system. 2. The vehicle exists.   **Post Condition:**   1. **Successful**- Delete the vehicle & show success message. 2. **Unsuccessful**- Stay in the same page, do not delete the vehicle & show error message. 3. **Exception**- Stay in the same page if possible |

|  |
| --- |
| **Use Case Name:** Adding a route |
| **Actor:** Admin  **Description:**   1. Input route ID & name. 2. Check route ID. (Unique) 3. Store a new route into database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system. 2. The route does not exists.   **Post Condition:**   1. **Successful**- Add a route & show success message. 2. **Unsuccessful**- Stay in the same page, do not add a route & show error message. 3. **Exception**- Stay in the same page if possible |

|  |
| --- |
| **Use Case Name:** Deleting a route |
| **Actor:** Admin  **Description:**   1. Input route ID. 2. Check route ID.(Valid) 3. Delete the route from database.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system. 2. The route exists.   **Post Condition:**   1. **Successful**- Delete the route & show success message. 2. **Unsuccessful**- Stay in the same page, do not delete the route & show error message. 3. **Exception**- Stay in the same page if possible |

|  |
| --- |
| **Use Case Name:** Setting vehicle for a travel |
| **Actor:** Admin  **Description:**   1. Input Way (Arrival/Departure), Route, Timeslot, Date & vehicle number. 2. Check data. (Valid) 3. Store vehicle info to that particular travel.   **Exception:**   1. Required page not found. 2. No database connection.   **Precondition:**   1. Admin logged in to the system.   **Post Condition:**   1. **Successful**- Set up the vehicle for that travel & show success message. 2. **Unsuccessful**- Stay in the same page, do not set up the vehicle for the travel & show error message. 3. **Exception**- Stay in the same page if possible. |