

## Use Cases ----- 3ngineer\_342

### UC1: Search Connections

**Primary Actor:** Traveller (end user)

**Goal and Interest:** Find train trip options matching provided criteria (direct, 1-stop, 2-stop).

#### Preconditions:

- System (backend) is running.
- Routes CSV has been loaded into memory at server startup, or frontend has fetched all connections.

#### Main Success Scenario:

1. User enters any subset of search criteria (departure, arrival, departure time, arrival time, train type, days of operation, first/second class prices).
2. Frontend builds a Query object and either:
3. Backend controller receives the request and calls `findAllConnections(routes, query)`.
4. `findAllConnections` filters routes, constructs Trip objects and computes prices, duration, and change times.
5. Backend returns a JSON array of Trip objects.
6. Frontend displays results in ResultsTable.

#### Alternate Flows:

- **Empty Query:** No filters , returns all trips.
- **Partial Input:** Returns only trips matching given filters.
- **No Matches:** System returns empty array , frontend displays “No trips found.”
- **Invalid Numeric Filter:** Non-numeric price input leads to no match; backend currently ignores invalid numeric filters.

#### Postconditions:

- UI displays filtered Trip list or “No trips found.”
-

## Use Case UC-02: View Results

**UC2:** View Results

**Primary Actor:** Traveller

**Goal and Interest:** Inspect returned trips with key details (departure, arrival, times, duration, prices, change times).

### Preconditions

- A result set ( $\geq 0$  items) is available.

### Main Success Scenario

1. Frontend renders results in ResultsTable.
2. Each row displays Departure, Arrival, Departure/Arrival Time, Duration, Train Type, Days of Operation, First/Second Class Prices, and Change Times.
3. If results are empty, the message “No trips found” is displayed.

### Alternate Flows

- **Malformed Duration:** Some trip entries have malformed times; duration computed client-side when necessary.

### Postconditions

- Traveller can view trip data and optionally sort or inspect further.

---

---

## Use Case UC-03: Sort Results

**UC3:** Sort Results

**Primary Actor:** Traveller

**Goal and Interest:** Re-order displayed trips by selected fields (duration, first class price, second class price).

### Preconditions

- Results are displayed in the ResultsTable.

### Main Success Scenario

1. User clicks column header.
2. ResultsTable toggles sort order (ascending/descending) for that field.
3. Table re-renders with rows sorted accordingly.

### Alternate Flows

- **Missing Duration Field:** Duration calculated from segment times before sorting.

#### **Postconditions**

- Table displays trips sorted per user selection.
- 

### **Use Case UC-04: Load Connections**

**UC4:** Load Connections

**Primary Actor:** System

**Goal:** Load CSV data into memory for fast search queries.

#### **Preconditions**

- data/train\_connections.csv exists and is accessible.

#### **Main Success Scenario**

1. Controller calls loadCSVData() in connectionsController.js during module initialization.
2. loadCSVData() reads train\_connections.csv synchronously, parses rows, and returns an array of TripSegment objects.
3. Controller stores this array in the routes variable for later searches.

#### **Alternate Flows**

- **File Missing/Read Error:** Current implementation throws error (startup fails).
  - Recommended improvement: catch error, log it, and exit gracefully or continue with empty routes.

#### **Postconditions**

- routes holds the parsed TripSegment array in memory.
- 

### **Use Case UC-05: Filter Locally**

**Name:** Filter Locally

**Primary Actor:** Traveller

**Goal:** Filter a list (allConnections) to find matching trips

#### **Preconditions**

- HomePage has populated allConnections.

### **Main Success Scenario**

1. SearchForm sends query fields to HomePage.handleSearch.
2. handleSearch iterates through allConnections and applies string and numeric filters.
3. Matching trips are stored in the results state and displayed to the user.

### **Alternate Flows**

- **Not Loaded Yet:** If allConnections not yet loaded, search yields no results.

### **Postconditions**

- results state contains the filtered subset of all connections.
- 
-