Use case name:	Use Case 1: Enter Journey Origin and Destination	
Scenario:	Passenger planning a journey needs to specify starting point and destination	
Triggering event:	Passenger opens the transportation app to plan a journey	
Brief description:	Passenger inputs their current location or desired starting point and their intended destination to initiate journey planning	
Actors:	Passenger	
Related use cases:	UC2 (Specify Arrival/Departure Time), UC25 (Calculate Optimal Routes)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	Passenger has access to the transportation app App is functional and connected to the network	
Postconditions:	- Origin and destination are stored in the system - System is ready to calculate route options	
Flow of activities:	**Actor**	**System**
	1. Opens transportation app 2. Selects origin input field 3. Enters or selects origin location 4. Selects destination input field 5. Enters or selects destination 6. Confirms location entries	1.1. Displays main interface 2.1. Activates location input interface 3.1. Validates and stores origin location 4.1. Activates destination input interface 5.1. Validates and stores destination 6.1. Prepares for route calculation
Exception conditions:	- Invalid location entered: System prompts for valid location - Network connectivity issues: System displays offline message - GPS location unavailable: System requests manual location entry	

Use case name:	Use Case 2: Specify Arrival/Departure Time
Scenario:	Passenger wants to plan journey for specific time requirements
Triggering event:	Passenger needs to arrive at destination by specific time or depart at specific time
Brief description:	Passenger sets preferred departure time or required arrival time for journey planning optimization
Actors:	Passenger

Related use cases:	UC1 (Enter Journey Origin and Destination), UC25 (Calculate Optimal Routes)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Origin and destination have been specified - Current time is available to the system	
Postconditions:	- Time preference is stored in the system - System can calculate time-optimized routes	
Flow of activities:	**Actor**	**System**
	1. Selects time preference option 2. Chooses "Depart at" or "Arrive by" 3. Sets desired time and date 4. Confirms time selection	1.1. Displays time selection interface 2.1. Configures time input based on selection 3.1. Validates time against current time 4.1. Stores time preference for route calculation
Exception conditions:	- Past time selected: System prompts for future time - Invalid time format: System requests correct format - Time too far in future: System warns about schedule availability	

Use case name:	Use Case 3: View Route Options	
Scenario:	Passenger reviews available transportation routes for their journey	
Triggering event:	System completes route calculation based on passenger inputs	
Brief description:	System presents multiple route alternatives with details like duration, cost, and transportation modes	
Actors:	Passenger	
Related use cases:	UC1 (Enter Journey Origin and Destination), UC25 (Calculate Optimal Routes), UC4 (Select Preferred Route)	
Stakeholders:	Passengers, Transportation Service Provider, Route Planning System	
Preconditions:	- Origin and destination are specified - Route calculation is completed - Transportation schedules are available	
Postconditions:	- Route options are displayed to passenger - Passenger can make informed route selection	
Flow of activities:	**Actor**	**System**
	1. Waits for route calculation 2. Reviews displayed route options	1.1. Processes route calculation request 2.1. Displays multiple route alternatives 3.1. Provides detailed information for each route

	3. Examines route details (time, cost, modes) 4. Compares different options	4.1. Highlights key differences between routes
Exception conditions:	- No routes available: System suggests alternative times or locations - Route calculation timeout: System retries or suggests manual refresh - Incomplete route data: System displays available information with warnings	

Use case name:	Use Case 4: Select Preferred Route	
Scenario:	Passenger chooses their preferred route from available options	
Triggering event:	Passenger decides on best route after reviewing options	
Brief description:	Passenger selects one route from the presented alternatives to proceed with journey planning	
Actors:	Passenger	
Related use cases:	UC3 (View Route Options), UC5 (Save Route as Favorite), UC9 (Purchase Digital Ticket/Pass)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Multiple route options are displayed - Route details are available for comparison	
Postconditions:	- Selected route is stored as active journey plan - System prepares for next steps (ticketing, navigation)	
Flow of activities:	**Actor**	**System**
	1. Reviews route options 2. Selects preferred route 3. Confirms route selection	1.1. Maintains display of route alternatives 2.1. Highlights selected route 3.1. Stores selected route as active plan 4.1. Prepares ticketing and navigation options
Exception conditions:	- Route no longer available: System suggests similar alternatives - Route details changed: System updates information and requests reconfirmation - Selection timeout: System maintains options for re-selection	

Use case name:	Use Case 5: Save Route as Favorite
Scenario:	Passenger wants to save frequently used route for quick future access
Triggering event:	Passenger identifies route they use regularly

Brief description:	Passenger marks a route as favorite to enable quick selection in future journey planning	
Actors:	Passenger	
Related use cases:	UC4 (Select Preferred Route), UC28 (Store User Preferences)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Route has been selected - Passenger has user account in system	
Postconditions:	- Route is saved in passenger's favorites - Route appears in quick-access favorites list	
Flow of activities:	**Actor**	**System**
	1. Selects "Save as Favorite" option 2. Provides name/label for favorite route 3. Confirms saving action 4. Updates user's favorite routes list	1.1. Displays favorite saving interface 2.1. Validates favorite name 3.1. Stores route in user's favorites
Exception conditions:	- Duplicate favorite name: System suggests alternative name - Storage limit reached: System prompts to remove old favorites - User not logged in: System prompts for authentication	

Use case name:	Use Case 6: Check Real-time Arrival Information	
Scenario:	Passenger wants current information about vehicle arrival times	
Triggering event:	Passenger needs to know when next vehicle will arrive	
Brief description:	Passenger views live updates on vehicle arrival times at specific stops or stations	
Actors:	Passenger	
Related use cases:	UC22 (Update Vehicle Location), UC7 (Request Proximity Notification)	
Stakeholders:	Passengers, Transportation Service Provider, Vehicle Operators	
Preconditions:	- Vehicle tracking system is operational - Real-time data is available - Passenger has selected route or stop	
Postconditions:	- Current arrival information is displayed - Information updates automatically	
Flow of activities:	**Actor**	**System**

	1. Requests real-time arrival information 2. Views current arrival predictions 3. Monitors updates to arrival times 4. Selects destination input field 5. Enters or selects destination 6. Confirms location entries	1.1. Accesses vehicle tracking data 2.1. Displays live arrival time 3.1. Continuously updates arrival predictions 4.1. Highlights any delays or changes
Exception conditions:	- Vehicle tracking unavailable: System displays scheduled times with warning - Data connection lost: System shows last known information - No vehicles in service: System displays service status	

Use case name:	Use Case 7: Request Proximity Notification	
Scenario:	Passenger wants to be alerted when vehicle approaches their location	
Triggering event:	Passenger wants hands-free monitoring of vehicle approach	
Brief description:	Passenger sets up automatic notifications for when their vehicle is approaching the pickup location	
Actors:	Passenger	
Related use cases:	UC6 (Check Real-time Arrival Information), UC22 (Update Vehicle Location)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Vehicle tracking is available - Passenger has notification permissions enabled - Route or stop is selected	
Postconditions:	- Proximity notification is activated - Passenger will receive alert when vehicle approaches	
Flow of activities:	**Actor**	**System**
	1. Selects proximity notification option 2. Sets notification distance/time preference 3. Confirms notification request	1.1. Displays notification settings 2.1. Configures notification parameters 3.1. Activates proximity monitoring 4.1. Monitors vehicle location for notification trigger
Exception conditions:	- Notifications disabled: System requests permission - Vehicle tracking unavailable: System cannot set notification - Multiple vehicles on route: System clarifies which vehicle to track	

Use case name:	Use Case 8: View Available Fare Options	
Scenario:	Passenger wants to see different ticket types and pricing	
Triggering event:	Passenger needs to understand fare structure before purchasing	
Brief description:	Passenger browses different ticket types, passes, and pricing options available for their journey	
Actors:	Passenger	
Related use cases:	UC9 (Purchase Digital Ticket/Pass), UC34 (Suggest Optimal Pass Type)	
Stakeholders:	Passengers, Transportation Service Provider, Payment System	
Preconditions:	- Route is selected - Fare information is available in system - Current pricing is loaded	
Postconditions:	- Fare options are displayed with details - Passenger can make informed purchase decision	
Flow of activities:	**Actor**	**System**
	1. Requests fare information 2. Reviews different ticket types 3. Compares pricing and validity periods 4. Examines terms and conditions	1.1. Retrieves current fare structure 2.1. Displays ticket options with descriptions 3.1 Shows pricing comparison and recommendations 4.1 Provides detailed fare terms
Exception conditions:	- Fare information unavailable: System displays last known rates with warning - Special pricing periods: System highlights promotional rates - Route-specific restrictions: System shows applicable limitations	

Use case name:	Use Case 9: Purchase Digital Ticket/Pass
Scenario:	Passenger buys digital transportation ticket or pass
Triggering event:	Passenger decides to purchase ticket after reviewing options
Brief description:	Passenger selects ticket type, provides payment information, and completes purchase transaction
Actors:	Passenger
Related use cases:	UC8 (View Available Fare Options), UC27 (Process Fare Payment), UC10 (Use Digital Pass for Validation)
Stakeholders:	Passengers, Transportation Service Provider
Preconditions:	- Fare options are displayed - Payment system is operational

	- Passenger has valid payment method	
Postconditions:	- Digital ticket/pass is purchased and stored - Payment is processed successfully - Ticket is ready for use	
Flow of activities:	**Actor**	**System**
	1. Selects desired ticket/pass type 2. Chooses payment method 3. Enters payment information 4. Confirms purchase 5. Receives digital ticket	1.1. Confirms selection and displays price 2.1. Displays payment interface 3.1. Validates payment details 4.1. Processes payment transaction 5.1. Generates and stores digital ticket
Exception conditions:	- Payment declined: System suggests alternative payment methods - Transaction timeout: System allows retry or cancellation - Ticket generation failure: System refunds payment and reports error	

Use case name:	Use Case 10: Use Digital Pass for Validation	
Scenario:	Passenger validates their digital ticket for journey	
Triggering event:	Passenger boards vehicle or enters station requiring validation	
Brief description:	Passenger presents digital QR code or NFC ticket for scanning and validation	
Actors:	Passenger	
Related use cases:	UC9 (Purchase Digital Ticket/Pass), UC31 (Verify Ticket Validity)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Digital ticket/pass is purchased and stored - Validation system is operational - Ticket is within validity period	
Postconditions:	- Ticket is validated for journey - Journey record is created - Remaining ticket value/uses are updated	
Flow of activities:	**Actor**	**System**
	1. Opens digital ticket in app 2. Presents ticket to validator 3. Waits for validation confirmation 4. Proceeds with journey	1.1. Displays QR code or activates NFC 2.1. Scans/reads ticket information 3.1 Verifies ticket validity 4.1. Records journey and updates ticket status 5.1. Displays validation confirmation
Exception conditions:	- Invalid ticket: System displays error and denies access - Expired ticket: System prompts for renewal - Technical scanning issues: System provides alternative validation method	

<table>
<tr><td><b>Use case name:</b></td><td colspan="2">Use Case 11: Enable Accessibility Filters</td></tr>
<tr><td><b>Scenario:</b></td><td colspan="2">Passenger with accessibility needs configures appropriate filters</td></tr>
<tr><td><b>Triggering event:</b></td><td colspan="2">Passenger requires accessible transportation options</td></tr>
<tr><td><b>Brief description:</b></td><td colspan="2">Passenger activates profile settings to filter routes and options based on accessibility requirements</td></tr>
<tr><td><b>Actors:</b></td><td colspan="2">Passenger</td></tr>
<tr><td><b>Related use cases:</b></td><td colspan="2">UC12 (View Accessible Route Options), UC28 (Store User Preferences)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Passengers, Transportation Service Provider</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">- Passenger has user account
- Accessibility options are available in system
- Passenger profile can be modified</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">- Accessibility preferences are saved
- Future route calculations include accessibility filters
- Profile reflects accessibility needs</td></tr>
<tr><td rowspan="2"><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td>1. Accesses accessibility settings
2. Selects required accessibility features
3. Configures specific requirements
4. Saves accessibility profile</td><td>1.1. Displays accessibility options
2.1. Records selected accessibility needs
3.1. Validates and stores detailed preferences
4.1. Updates user profile with accessibility settings</td></tr>
<tr><td><b>Exception conditions:</b></td><td colspan="2">- Limited accessibility data: System warns about potential route limitations
- Conflicting requirements: System requests clarification
- Profile save failure: System retries and confirms successful save</td></tr>
</table>

Use case name:	Use Case 12: View Accessible Route Options
Scenario:	Passenger views routes filtered for accessibility compliance
Triggering event:	Route calculation includes accessibility requirements
Brief description:	System displays routes that meet passenger's accessibility needs, highlighting accessible features
Actors:	Passenger
Related use cases:	UC11 (Enable Accessibility Filters), UC3 (View Route Options), UC25 (Calculate Optimal Routes)
Stakeholders:	Passengers, Transportation Service Provider
Preconditions:	- Accessibility filters are enabled - Route calculation is completed

	- Accessibility data is available for routes	
Postconditions:	- Accessible routes are displayed with accessibility details - Non-accessible routes are filtered out or marked - Passenger can select appropriate accessible route	
Flow of activities:	**Actor**	**System**
	1. Requests route options 2. Reviews accessible route options 3. Examines accessibility features 4. Compares accessible alternatives	1.1. Applies accessibility filters to route calculation 2.1 Displays routes meeting accessibility criteria 3.1. Highlights specific accessible features 4.1. Provides accessibility details for each route
Exception conditions:	- No accessible routes available: System suggests alternative times or locations - Limited accessibility information: System displays available data with warnings - Accessibility features temporarily unavailable: System provides current status	

Use case name:	Use Case 13: Request Alternative Routes During Disruption	
Scenario:	Passenger needs new route options when service is disrupted	
Triggering event:	Service disruption affects passenger's planned route	
Brief description:	Passenger seeks alternative route options when their original route is affected by delays, cancellations, or other disruptions	
Actors:	Passenger	
Related use cases:	UC26 (Push Service Disruption Alerts), UC25 (Calculate Optimal Routes), UC3 (View Route Options)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Service disruption is occurring - Original route is affected - Alternative routes exist	
Postconditions:	- Alternative routes are calculated and displayed - Passenger can select new route - Disruption impact is minimized	
Flow of activities:	**Actor**	**System**
	1. Receives disruption notification 2. Requests alternative routes 3. Reviews new route options 4. Selects alternative route	1.1. Detects service disruption affecting route 2.1. Calculates alternative route options 3.1. Displays alternatives avoiding disrupted services 4.1. Updates journey plan with new route

Exception conditions:	- No alternatives available: System suggests waiting or different travel times - Multiple disruptions: System prioritizes least affected routes - Real-time data unavailable: System uses last known service status

Use case name:	Use Case 14: Book Group Transportation	
Scenario:	Passenger reserves transportation space for multiple travelers	
Triggering event:	Passenger needs to travel with a group requiring coordination	
Brief description:	Passenger reserves space for multiple people traveling together as a group	
Actors:	Passenger	
Related use cases:	UC33 (Generate Group QR Code), UC9 (Purchase Digital Ticket/Pass)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Group travel options are available - Route supports group bookings - Passenger has authority to book for group	
Postconditions:	- Group reservation is confirmed - Space is reserved for specified number of travelers - Group ticket/pass is generated	
Flow of activities:	**Actor**	**System**
	1. Selects group travel option 2. Specifies number of travelers 3. Provides group details 4. Confirms group booking 5.	1.1. Displays group booking interface 2.1 Checks availability for group size 3.1. Records group information 4.1. Reserves space and generates group booking
Exception conditions:	- Insufficient space available: System suggests alternative times or routes - Group size exceeds limits: System requests splitting into smaller groups - Group booking restrictions: System explains limitations and alternatives	

Use case name:	Use Case 15: Renew Transit Pass
Scenario:	Passenger extends validity of existing transportation pass
Triggering event:	Transit pass is approaching expiration or has expired
Brief description:	Passenger extends the validity period of their existing transit pass through renewal process
Actors:	Passenger

<table>
<tr><td><b>Related use cases:</b></td><td colspan="2">UC9 (Purchase Digital Ticket/Pass), UC27 (Process Fare Payment)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Passengers, Transportation Service Provider</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">- Passenger has existing transit pass
- Renewal option is available for pass type
- Payment system is operational</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">- Transit pass validity is extended
- Payment is processed for renewal
- Pass remains active without interruption</td></tr>
<tr><td><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td></td><td>1. Selects pass renewal option
2. Chooses renewal period
3. Confirms renewal terms
4. Completes renewal payment</td><td>1.1. Displays current pass details and renewal options
2.1. Calculates renewal cost
3.1. Processes renewal payment
4.1. Updates pass validity period</td></tr>
<tr><td><b>Exception conditions:</b></td><td colspan="2">- Pass not eligible for renewal: System explains restrictions and alternatives
- Payment failure: System retries payment or suggests alternative methods
- Renewal processing error: System maintains current pass status and retries</td></tr>
</table>

<table>
<tr><td><b>Use case name:</b></td><td colspan="2">Use Case 16: Request Refund for Unused Ticket</td></tr>
<tr><td><b>Scenario:</b></td><td colspan="2">Passenger seeks refund for ticket they cannot or did not use</td></tr>
<tr><td><b>Triggering event:</b></td><td colspan="2">Passenger has unused ticket eligible for refund</td></tr>
<tr><td><b>Brief description:</b></td><td colspan="2">Passenger submits refund application for eligible unused tickets according to refund policy</td></tr>
<tr><td><b>Actors:</b></td><td colspan="2">Passenger</td></tr>
<tr><td><b>Related use cases:</b></td><td colspan="2">UC32 (Calculate Refund Eligibility), UC27 (Process Fare Payment)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Passengers, Transportation Service Provider, Financial Department, Payment System</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">- Passenger has unused ticket
- Ticket is within refund eligibility period
- Refund policy allows refund for ticket type</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">- Refund request is submitted and processed
- Eligible refund amount is calculated
- Refund is issued to original payment method</td></tr>
<tr><td><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td></td><td>1. Selects refund request option
2. Selects ticket for refund
3. Provides refund reason
4. Confirms refund request</td><td>1.1. Displays refund policy and eligible tickets
2.1 Verifies ticket eligibility for refund
3.1. Calculates refund amount based on policy
4.1. Processes refund to original payment method</td></tr>
</table>

Exception conditions:	- Ticket not eligible for refund: System explains policy restrictions - Refund period expired: System denies refund with explanation - Processing error: System queues request for manual review

<table>
<tr><td><b>Use case name:</b></td><td colspan="2">Use Case 17: View Multimodal Journey Options</td></tr>
<tr><td><b>Scenario:</b></td><td colspan="2">Passenger explores combinations of different transportation modes</td></tr>
<tr><td><b>Triggering event:</b></td><td colspan="2">Single transportation mode insufficient for complete journey</td></tr>
<tr><td><b>Brief description:</b></td><td colspan="2">Passenger views journey options that combine multiple transportation modes</td></tr>
<tr><td><b>Actors:</b></td><td colspan="2">Passenger</td></tr>
<tr><td><b>Related use cases:</b></td><td colspan="2">C1 (Enter Journey Origin and Destination), UC25 (Calculate Optimal Routes), UC3 (View Route Options)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Passengers, Multiple Transportation Providers, Multimodal Integration System</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">- Multiple transportation modes are available
- Integration between different systems exists
- Route calculation supports multimodal planning</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">- Multimodal journey options are displayed
- Transfer points and timing are clearly shown
- Passenger can select integrated multimodal route</td></tr>
<tr><td rowspan="2"><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td>1. Requests multimodal options
2. Reviews combined transportation options
3. Examines transfer points and timing
4. Compares multimodal alternatives</td><td>1.1. Calculates routes using multiple transportation modes
2.1. Displays integrated journey options
3.1. Highlights transfer locations and schedules
4.1. Shows total journey time and cost for each option</td></tr>
<tr><td><b>Exception conditions:</b></td><td colspan="2">- Limited multimodal integration: System shows available combinations with warnings
- Transfer timing issues: System highlights potential connection problems
- Mode unavailable: System recalculates without unavailable transportation mode</td></tr>
</table>

Use case name:	**Use Case 18: Monitor Journey Progress**
Scenario:	Passenger tracks their location and progress along selected route
Triggering event:	Passenger begins journey and wants real-time progress tracking

Brief description:	Passenger receives step-by-step guidance and tracks their current location along the planned route	
Actors:	Passenger	
Related use cases:	UC4 (Select Preferred Route), UC22 (Update Vehicle Location), UC6 (Check Real-time Arrival Information)	
Stakeholders:	Passengers, Transportation Service Provider	
Preconditions:	- Route is selected and journey has begun - GPS/location services are available - Real-time tracking is enabled	
Postconditions:	- Journey progress is continuously monitored - Passenger receives navigation guidance - Arrival predictions are updated based on progress	
Flow of activities:	**Actor**	**System**
	1. Starts journey monitoring 2. Follows route guidance 3. Monitors progress updates 4. Receives arrival predictions	1.1. Activates GPS tracking and route guidance 2.1. Provides turn-by-turn directions 3.1. Updates location and calculates remaining journey 4.1. Adjusts arrival time based on current progress
Exception conditions:	- GPS signal lost: System uses last known location and requests manual update - Route deviation: System recalculates route from current location - Service delays: System updates journey plan with new timing	

Use case name:	Use Case 19: Authenticate in Driver Interface
Scenario:	Driver logs into system with secure credentials
Triggering event:	Driver begins work shift and needs system acces
Brief description:	Driver provides authentication credentials to access driver-specific system functions
Actors:	Passenger
Related use cases:	UC20 (Access Assigned Route Information)
Stakeholders:	Drivers, Transportation Service Provider, Security System
Preconditions:	- Driver has valid credentials - Driver interface system is operational - Driver is authorized for current shift
Postconditions:	- Driver is authenticated and logged into system - Driver-specific interface is available

	- Driver activities can be tracked and recorded	
Flow of activities:	**Actor**	**System**
	1. Opens driver interface application 2. Enters driver credentials 3. Provides additional authentication if required 4. Accesses driver dashboard	1.1. Displays authentication screen 2.1. Validates credentials against driver database 3.1. Verifies driver authorization for current shift 4.1. Loads driver-specific interface and route information
Exception conditions:	- Invalid credentials: System denies access and logs attempt - Driver not scheduled: System displays schedule information and denies access - System maintenance: System displays maintenance message and estimated availability	

Use case name:	Use Case 20: Access Assigned Route Information	
Scenario:	Driver views detailed route and schedule information for their shift	
Triggering event:	Driver needs route details to begin or continue service	
Brief description:	Driver accesses comprehensive information about their assigned route including stops, timing, and special instructions	
Actors:	Driver	
Related use cases:	UC19 (Authenticate in Driver Interface), UC21 (Follow Turn-by-Turn Navigation)	
Stakeholders:	Drivers, Transportation Service Provider, Route Planning Department	
Preconditions:	- Driver is authenticated in system - Route assignment exists for driver - Route information is current and available	
Postconditions:	- Driver has access to complete route information - Route details are displayed in driver interface - Driver can begin route execution	
Flow of activities:	**Actor**	**System**
	1. Requests assigned route information 2. Reviews route details and schedule 3. Examines stop locations and timing 4. Notes special instructions or alerts	1.1. Retrieves driver's current route assignment 2.1. Displays comprehensive route information 3.1. Activates proximity monitoring 4.1. Monitors vehicle location for notification trigger
Exception conditions:	- No route assigned: System displays schedule and contacts dispatcher	

	- Route information incomplete: System shows available information and requests update - Route changes: System highlights changes and requests acknowledgment

<table>
<tr><td><b>Use case name:</b></td><td colspan="2"><b>Use Case 21: Follow Turn-by-Turn Navigation</b></td></tr>
<tr><td><b>Scenario:</b></td><td colspan="2">Driver receives navigation guidance for assigned route</td></tr>
<tr><td><b>Triggering event:</b></td><td colspan="2">Driver begins route execution and needs navigation assistance</td></tr>
<tr><td><b>Brief description:</b></td><td colspan="2">Driver receives step-by-step navigation instructions to follow the assigned route efficiently</td></tr>
<tr><td><b>Actors:</b></td><td colspan="2">Driver</td></tr>
<tr><td><b>Related use cases:</b></td><td colspan="2">UC20 (Access Assigned Route Information), UC22 (Update Vehicle Location)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Drivers, Transportation Service Provider, Navigation System</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">- Route information is loaded
- GPS navigation is available
- Vehicle location tracking is active</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">- Driver receives continuous navigation guidance
- Route adherence is monitored
- Navigation adjusts for real-time conditions</td></tr>
<tr><td rowspan="2"><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td>1. Starts route navigation
2. Follows navigation instructions
3. Receives route updates
4. Confirms arrival at stops</td><td>1.1. Activates GPS navigation for assigned route
2.1. Provides turn-by-turn directions
3.1 Adjusts navigation for traffic or route changes
4.1 Records stop arrivals and departure times</td></tr>
<tr><td><b>Exception conditions:</b></td><td colspan="2">- GPS signal lost: System uses last known location and provides general directions
- Route blocked: System calculates alternative path to next stop
- Navigation system failure: System provides basic route information for manual navigation</td></tr>
</table>

Use case name:	Use Case 22: Update Vehicle Location
Scenario:	System automatically tracks vehicle position via GPS
Triggering event:	Vehicle is in service and location tracking is active

Brief description:	System continuously monitors and updates vehicle location for real-time passenger information and operational tracking	
Actors:	System	
Related use cases:	UC6 (Check Real-time Arrival Information), UC21 (Follow Turn-by-Turn Navigation), UC35 (Record Vehicle Timing Data)	
Stakeholders:	Passengers, Drivers, Transportation Service Provider, Operations Control	
Preconditions:	- Vehicle has GPS tracking capability - Vehicle is in active service - Communication system is operational	
Postconditions:	- Vehicle location is continuously updated - Real-time information is available to passengers - Location data is stored for analysis	
Flow of activities:	**Actor**	**System**
		1. Receives GPS location data from vehicle 2. Validates and processes location information 3. Updates vehicle position in real-time database 4. Distributes location updates to passenger information systems
Exception conditions:	- GPS signal unavailable: System uses last known location with timestamp - Communication failure: System queues location updates for transmission when connection restored - Invalid location data: System filters out erroneous readings	

Use case name:	**Use Case 23: Report Traffic Incident**
Scenario:	Driver logs unexpected road conditions or incidents
Triggering event:	Driver encounters traffic incident affecting route
Brief description:	Driver reports traffic incidents, road conditions, or other issues that may affect service or passenger safety
Actors:	Driver
Related use cases:	UC26 (Push Service Disruption Alerts), UC30 (Coordinate Emergency Response)

Stakeholders:	Drivers, Transportation Service Provider, Operations Control, Emergency Services	
Preconditions:	- Driver is authenticated and on active route - Incident reporting system is available - Driver observes reportable incident	
Postconditions:	- Incident is recorded in system - Appropriate alerts are generated - Operations team is notified for response	
Flow of activities:	**Actor**	**System**
	1. Observes traffic incident or road condition 2. Selects incident type and severity 3. Provides additional incident details 4. Submits incident report	1.1. Provides incident reporting interface 2.1. Records incident details and location 3.1. Validates and stores incident report 4.1. Notifies operations control and generates alerts
Exception conditions:	- Emergency situation: System immediately escalates to emergency response - Duplicate incident report: System consolidates with existing reports - Communication failure: System queues report for transmission when connection restored	

Use case name:	**Use Case 24: Complete Route Session**	
Scenario:	Driver marks route as completed at end of service	
Triggering event:	Driver finishes assigned route or shift	
Brief description:	Driver formally ends their route session, recording completion and any final notes	
Actors:	Driver	
Related use cases:	UC35 (Record Vehicle Timing Data), UC29 (Generate Traffic Analytics)	
Stakeholders:	Drivers, Transportation Service Provider, Operations Management	
Preconditions:	- Driver has active route session - Route service is ending - Driver interface is accessible	
Postconditions:	- Route session is marked complete - Final timing and performance data is recorded - Driver is logged out of active route	
Flow of activities:	**Actor**	**System**

	Selects route completion option Confirms route completion Provides any final notes or issues Logs out of route session	1.1 Displays route completion interface 2.1 Records final route timing and statistics 3.1. Stores completion notes and driver feedback 4.1. Closes route session and updates driver status
Exception conditions:	- Incomplete route: System requests confirmation and reason for early completion - System issues during completion: System saves available data and allows manual completion - Emergency route termination: System records emergency completion with appropriate flags	

Use case name:	**Use Case 25: Calculate Optimal Routes**
Scenario:	System processes journey parameters to determine best options
Triggering event:	Passenger submits origin, destination, and preferences
Brief description:	System analyzes multiple factors to calculate and rank optimal route options for passenger journey
Actors:	System
Related use cases:	UC1 (Enter Journey Origin and Destination), UC2 (Specify Arrival/Departure Time), UC3 (View Route Options)
Stakeholders:	Passengers, Transportation Service Provider, Route Optimization System
Preconditions:	- Origin and destination are specified - Transportation schedules are available - Route calculation algorithms are operational
Postconditions:	- Multiple route options are calculated and ranked - Route details include timing, cost, and mode information - Results are ready for passenger review
Flow of activities:	**System**
	1. Receives route calculation request with parameters 2. Analyzes available transportation options 3. Calculates multiple route alternatives 4. Ranks routes based on optimization criteria 5. Prepares route options for display
Exception conditions:	- No viable routes found: System suggests alternative times or nearby locations - Calculation timeout: System provides partial results or requests retry - Data unavailable: System uses cached information with appropriate warnings

Use case name:	**Use Case 26: Push Service Disruption Alerts**
Scenario:	System notifies affected users about service issues
Triggering event:	Service disruption is detected or reported
Brief description:	System automatically sends notifications to passengers affected by service disruptions, delays, or cancellations
Actors:	System
Related use cases:	UC23 (Report Traffic Incident), UC13 (Request Alternative Routes During Disruption)
Stakeholders:	Passengers, Transportation Service Provider, Operations Control
Preconditions:	• Service disruption is detected or reported • Affected passengers can be identified • Notification system is operational
Postconditions:	• Affected passengers receive disruption alerts • Alternative options are suggested where possible • Service status is updated system-wide
Flow of activities:	**System**
	1. Detects or receives service disruption report 2. Identifies affected routes and passengers 3. Generates appropriate alert messages 4. Sends notifications to affected passengers 5. Updates service status information
Exception conditions:	Notification system failure: System logs alerts for manual distribution Passenger contact information unavailable: System updates public displays and app notifications False alarm: System sends correction notification and updates status

Use case name:	Use Case 27: Process Fare Payment
Scenario:	System securely handles payment transactions
Triggering event:	Passenger initiates ticket purchase or pass renewal
Brief description:	System processes payment transactions securely through integrated payment gateway
Actors:	System
Related use cases:	UC9 (Purchase Digital Ticket/Pass), UC15 (Renew Transit Pass), UC16 (Request Refund for Unused Ticket)
Stakeholders:	Passengers, Transportation Service Provider, Payment Gateway, Financial Institution

Preconditions:	Payment request is initiated Payment gateway is operational Valid payment information is provided
Postconditions:	Payment is processed successfully or declined Transaction record is created Appropriate confirmation or error message is generated
Flow of activities:	**System**
	1. Receives payment request with transaction details 2. Validates payment information 3. Submits transaction to payment gateway 4. Receives payment confirmation or decline 5. Records transaction and generates confirmation
Exception conditions:	Payment declined: System provides decline reason and suggests alternatives Gateway timeout: System retries transaction or allows manual retry Transaction processing error: System logs error and initiates refund if necessary

Use case name:	Use Case 28: Store User Preferences
Scenario:	System saves personalized settings for future use
Triggering event:	User modifies preferences or settings
Brief description:	System securely stores user preferences including accessibility settings, favorite routes, and notification preferences
Actors:	System
Related use cases:	UC5 (Save Route as Favorite), UC11 (Enable Accessibility Filters)
Stakeholders:	Passengers, Transportation Service Provider, Data Management System
Preconditions:	User has account in system Preference data is valid Storage system is operational
Postconditions:	User preferences are stored securely Preferences are available for future sessions User experience is personalized based on stored preferences
Flow of activities:	**System**
	1. Receives user preference update request 2. Validates preference data 3. Encrypts sensitive preference information 4. Stores preferences in user profile database 5. Confirms successful storage
Exception conditions:	Storage system unavailable: System queues preferences for later storage Invalid preference data: System requests correction or uses default values Storage quota exceeded: System prompts user to remove old preferences

Use case name:	**Use Case 29: Generate Traffic Analytics**
Scenario:	System analyzes transportation patterns for optimization
Triggering event:	Scheduled analytics generation or on-demand analysis request
Brief description:	System processes collected transportation data to generate insights for service optimization and planning
Actors:	System
Related use cases:	UC35 (Record Vehicle Timing Data), UC22 (Update Vehicle Location)
Stakeholders:	Transportation Service Provider, Operations Management, Planning Department
Preconditions:	Sufficient operational data is available Analytics algorithms are operational Data processing resources are available
Postconditions:	Analytics reports are generated Insights are available for decision making Trends and patterns are identified
Flow of activities:	**System**
	1. Collects operational data from various sources 2. Processes data using analytics algorithms 3. Identifies patterns and trends 4. Generates reports and visualizations 5. Distributes analytics to stakeholders
Exception conditions:	Insufficient data: System generates partial analytics with data quality warnings Processing failure: System retries analysis or generates error report Data quality issues: System filters problematic data and notes limitations

Use case name:	Use Case 30: Coordinate Emergency Response
Scenario:	System manages response to incidents affecting service
Triggering event:	Emergency incident is reported or detected
Brief description:	System coordinates emergency response procedures including notifications, resource allocation, and communication with emergency services
Actors:	System
Related use cases:	UC23 (Report Traffic Incident), UC26 (Push Service Disruption Alerts)
Stakeholders:	Drivers, Transportation Service Provider, Security System
Preconditions:	Emergency incident is detected or reported Emergency response protocols are defined Communication systems are operational
Postconditions:	Emergency response is initiated Appropriate authorities are notified Service adjustments are implemented

Flow of activities:	**System**
	1. Receives emergency incident report 2. Assesses incident severity and type 3. Initiates appropriate emergency protocols 4. Notifies emergency services and operations control 5. Implements service adjustments and passenger notifications
Exception conditions:	Communication system failure: System uses backup communication methods Unclear incident severity: System escalates to highest appropriate response level Multiple simultaneous emergencies: System prioritizes based on severity and resource availability

Use case name:	Use Case 31: Verify Ticket Validity
Scenario:	System checks if digital ticket is valid for use
Triggering event:	Passenger presents ticket for validation
Brief description:	System verifies that presented digital ticket is authentic, current, and valid for the requested journey
Actors:	Driver
Related use cases:	UC10 (Use Digital Pass for Validation), UC9 (Purchase Digital Ticket/Pass)
Stakeholders:	Passengers, Transportation Service Provider, Validation System, Revenue Protection
Preconditions:	Digital ticket is presented for validation Validation system is operational Ticket database is accessible
Postconditions:	Ticket validity is determined Validation result is recorded Appropriate access is granted or denied
Flow of activities:	**System**
	1. Scans or reads digital ticket information 2. Verifies ticket authenticity and format 3. Checks ticket validity period and usage limits 4. Validates ticket for current journey/zone 5. Records validation attempt and result
Exception conditions:	Invalid ticket: System denies access and logs security event Expired ticket: System provides expiration information and renewal options System connectivity issues: System uses offline validation if available

Use case name:	**Use Case 32: Calculate Refund Eligibility**

<table>
<tr><td><b>Scenario:</b></td><td colspan="2">System determines if ticket qualifies for refund based on policy</td></tr>
<tr><td><b>Triggering event:</b></td><td colspan="2">Passenger requests refund for unused ticket</td></tr>
<tr><td><b>Brief description:</b></td><td colspan="2">System evaluates ticket against refund policy criteria to determine eligibility and calculate refund amount</td></tr>
<tr><td><b>Actors:</b></td><td colspan="2">System</td></tr>
<tr><td><b>Related use cases:</b></td><td colspan="2">UC16 (Request Refund for Unused Ticket), UC27 (Process Fare Payment)</td></tr>
<tr><td><b>Stakeholders:</b></td><td colspan="2">Passengers, Transportation Service Provider, Financial Department</td></tr>
<tr><td><b>Preconditions:</b></td><td colspan="2">Refund request is submitted
Ticket information is available
Refund policy rules are current</td></tr>
<tr><td><b>Postconditions:</b></td><td colspan="2">Refund eligibility is determined
Refund amount is calculated if eligible
Decision rationale is documented</td></tr>
<tr><td rowspan="2"><b>Flow of activities:</b></td><td><b>Actor</b></td><td><b>System</b></td></tr>
<tr><td>1. Starts route navigation
2. Follows navigation instructions
3. Receives route updates
4. Confirms arrival at stops</td><td>1. Receives refund request with ticket details
2. Retrieves applicable refund policy rules
3. Evaluates ticket against eligibility criteria
4. Calculates refund amount based on policy
5. Documents decision and rationale</td></tr>
<tr><td><b>Exception conditions:</b></td><td colspan="2">Policy rules unclear: System escalates to manual review
Ticket information incomplete: System requests additional information
Calculation error: System uses conservative approach and flags for review</td></tr>
</table>

Use case name:	Use Case 33: Generate Group QR Code
Scenario:	System creates single validation code for group travel
Triggering event:	Group transportation booking is confirmed
Brief description:	System generates unified QR code that validates transportation for entire group
Actors:	System
Related use cases:	UC14 (Book Group Transportation), UC31 (Verify Ticket Validity)
Stakeholders:	Passengers, Transportation Service Provider, Group Travel Coordinator
Preconditions:	Group booking is confirmed and paid Group size is within system limits QR code generation system is operational

Postconditions:	Group QR code is generated and linked to booking Code is available for group leader Validation system recognizes group code
Flow of activities:	**System**
	1. Receives group booking confirmation 2. Validates group booking details 3. Generates unique QR code for group 4. Links QR code to group booking record 5. Delivers QR code to group leader
Exception conditions:	Code generation failure: System retries generation with new parameters Group booking invalid: System prevents code generation and reports error Delivery failure: System provides alternative code delivery methods

Use case name:	**Use Case 34: Suggest Optimal Pass Type**
Scenario:	System recommends most cost-effective pass based on usage patterns
Triggering event:	Passenger views fare options or system analyzes usage
Brief description:	System analyzes passenger travel patterns and recommends the most economical pass type
Actors:	System
Related use cases:	UC8 (View Available Fare Options), UC28 (Store User Preferences)
Stakeholders:	Passengers, Transportation Service Provider, Revenue Management
Preconditions:	Passenger usage data is available Multiple pass types exist Recommendation algorithms are operational
Postconditions:	Optimal pass recommendation is generated Cost savings analysis is provided Recommendation is presented to passenger
Flow of activities:	**System**
	1. Analyzes passenger travel history and patterns 2. Evaluates available pass types and pricing 3. Calculates cost scenarios for different pass types 4. Identifies most cost-effective option 5. Generates recommendation with savings analysis
Exception conditions:	Insufficient usage data: System provides general recommendations based on stated travel plans Equal cost options: System recommends based on flexibility and convenience factors Recommendation algorithm failure: System presents all options without specific recommendation

Use case name:	**Use Case 35: Record Vehicle Timing Data**
Scenario:	System logs actual arrival/departure times against schedule
Triggering event:	Vehicle arrives at or departs from scheduled stops
Brief description:	System automatically records vehicle timing data for performance analysis and schedule optimization
Actors:	System
Related use cases:	UC22 (Update Vehicle Location), UC24 (Complete Route Session), UC29 (Generate Traffic Analytics)
Stakeholders:	Transportation Service Provider, Operations Management, Schedule Planning
Preconditions:	Vehicle location tracking is active Schedule data is available Timing recording system is operational
Postconditions:	Timing data is recorded in database Performance metrics are updated Data is available for analysis
Flow of activities:	**System**
	1. Detects vehicle arrival at scheduled stop 2. Records actual arrival time 3. Compares with scheduled time 4. Detects vehicle departure from stop 5. Records departure time and calculates dwell time 6. Stores timing data for analysis
Exception conditions:	Location detection failure: System estimates timing based on last known position Schedule data unavailable: System records actual times without comparison Data storage failure: System queues timing data for later storage