# NYCPS Transportation Management System - Solution Design (RFP R1804)

## 1. Overview

This document outlines the functional design for the NYCPS Office of Pupil Transportation (OPT) Transportation Management System. The system aims to provide a single, integrated platform for near real-time bus location tracking, student ridership recording, dynamic route optimization, and seamless communication among all stakeholders (OPT staff, school administrators, bus vendors (SBCs), drivers, parents/caregivers, and students). The solution will replace legacy systems and serve as the new system of record for GPS, routing, and ridership data.

## 2. Core Functional Components

The solution is comprised of the following core functional components:

2.1. GPS Data Ingestion & Processing Pipeline

* **Responsibilities:**
  + Receive location, time, and motion data (speed, start/stop, idle, heavy braking) from GPS-enabled devices on buses at least every 60 seconds.
  + Process and validate incoming data streams for accuracy and timeliness (targeting near real-time availability within 30 seconds for 99% of streams).
  + Associate incoming data with specific devices, buses, drivers, and routes based on login/assignment information from the Bus Driver Module.
  + Calculate Estimated Time of Arrival (ETA) for upcoming stops and final destinations based on real-time location, traffic conditions, and route data.
  + Detect geofence entry/exit events (for up to 100 geofences per device) and trigger alerts/notifications as configured in the OPT Administrative Module.
  + Provide processed, near real-time location and ETA data via internal APIs to other system components (Routing Engine, Notification System, User Modules).
  + Handle device status reporting (e.g., low battery, connectivity loss).
  + Store raw and processed GPS data for immediate access (12 months) and long-term archival (7 years).
* **Key Data Inputs:** GPS coordinates, timestamps, speed, motion events (start/stop/idle/braking), device ID, driver ID, route ID, vehicle ID, geofence definitions.
* **Key Data Outputs:** Processed real-time bus location, ETAs, geofence alerts, historical track data ("breadcrumbs"), device status alerts.
* **Interactions:** Bus Driver Module (receives device/driver/route association), Dynamic Routing Engine (provides real-time data for route adjustments/analysis), Notification System (triggers alerts), User Modules (provides data for map displays/ETAs), Reporting & Analytics Subsystem (provides historical data), Data Archival Storage.

2.2. Ridership Tracking Module

* **Responsibilities:**
  + Receive and record student boarding and alighting events, including student ID, timestamp, location (GPS coordinates), bus ID, route ID, and driver/attendant ID.
  + Support multiple methods of capture: automated scanning (QR code/barcode via device camera or dedicated reader), manual entry by driver/attendant (as backup).
  + Handle distinct processes for different student categories (SE, GE, Pre-K/EI) as required.
  + Associate ridership events with specific students, stops, routes, and buses.
  + Track student presence/absence against the expected roster for each stop.
  + Flag students whose parents/caregivers have opted out of ridership tracking (if feature enabled by DOE).
  + Make ridership status available in near real-time to relevant User Modules (Driver, School Admin, OPT Admin, Parent/Caregiver).
  + Provide data for ridership reports and performance metrics (e.g., on-time pickups/drop-offs).
  + Store ridership data for immediate access (12 months) and long-term archival (7 years).
* **Key Data Inputs:** Student ID (from scan/manual entry), timestamp, GPS location, route/stop sequence, device ID, driver/attendant ID, opt-out flags.
* **Key Data Outputs:** Real-time ridership status (boarded, alighted, absent), historical ridership records, data feeds for reporting.
* **Interactions:** Bus Driver Module (receives scan/manual entry), GPS Pipeline (receives location context), Student Management System (links to student records/rosters), User Modules (displays status), Notification System (triggers board/alight alerts), Reporting & Analytics Subsystem.

2.3. Dynamic Routing Engine

* **Responsibilities:**
  + Generate optimized initial bus routes (AM, PM, Alt PM, Field Trip, After-School, etc.) based on student data (home/school locations, session times, special needs, exceptions like STH/Foster/Dual Custody), stop locations, vehicle capacity/type, contract rules, school calendars, and configurable OPT business rules/parameters (e.g., travel time guidelines, distance constraints, medical alert codes, weighted capacity calculations).
  + Integrate and utilize NYC LION ArcGIS base maps and allow authorized OPT users to edit map layers (street speeds, new roads, temporary roadblocks, travel restrictions).
  + Dynamically adjust active routes in near real-time based on live inputs: traffic conditions, road closures, weather alerts, GPS data (actual path vs. planned), real-time ridership updates (e.g., student absence notifications from Parent App), alerts from drivers (e.g., breakdown). Dynamic adjustments must prioritize student experience and service stability, respecting OPT scheduling/sequencing where possible but allowing necessary deviations.
  + Calculate and update ETAs dynamically based on real-time conditions.
  + Suggest optimal re-sequencing or route consolidation/splitting based on ongoing analysis.
  + Support "what-if" scenario planning for route optimization.
  + Handle specific routing logic for different student populations (GE, SE, Pre-K/EI, STH, etc.) and programs (Fall, Summer, After-School, Field Trips) separately but within the integrated system.
  + Generate route sheets and turn-by-turn navigation data (visual/audio) for the Bus Driver Module.
  + Identify potential routing conflicts (e.g., incorrect vehicle type for student needs, late arrivals, route overcrowding, guideline violations) and generate alerts for OPT Admin Module.
  + Integrate with Student Management System for student data, Stop Management functions for stop data/rules, and Session Time management.
  + Interface with upstream NYCPS systems (Contracts, Vehicle Info) and downstream systems (Payment processing, Reporting, etc.).
  + Support route check-in/check-out by routers and supervisor approval workflows.
  + Retain 7 years of historical routing data for analysis and replay.
* **Key Data Inputs:** Student data (from Student Management), stop data/rules, school session times/calendars, vehicle data (capacity/type/availability), contract rules, OPT routing parameters, real-time GPS data, real-time traffic feeds, real-time ridership updates, driver alerts, map data (LION ArcGIS, OPT edits).
* **Key Data Outputs:** Optimized route plans, dynamic route adjustments, turn-by-turn navigation data, route sheets, ETAs, routing conflict alerts, data feeds for User Modules and Reporting.
* **Interactions:** GPS Pipeline (receives real-time data), Ridership Module (receives updates), Student Management System (student data), User Modules (displays routes/ETAs, receives alerts/requests), Notification System (receives/triggers alerts), External Systems (Map data providers, NYCPS systems).

2.4. Notification & Communication System

* **Responsibilities:**
  + Manage and deliver automated, near real-time notifications to relevant stakeholders via their respective modules (or SMS/email as fallback).
  + Trigger notifications based on system events: bus approaching pickup/drop-off, student board/alight, route delays, missed pickups, vehicle breakdown, geofence alerts, routing conflicts, changes in student/route information.
  + Allow OPT Administrators to configure notification types, content, triggers, frequency, and recipients (including opt-in/opt-out management for parents/students).
  + Enable OPT Administrators to initiate manual alerts (including robocalls) for specific events (e.g., weather emergencies, accidents, major traffic issues) targeting specific groups (e.g., parents on specific routes, all drivers).
  + Facilitate two-way communication between OPT Administrators/Routers and Bus Drivers (via their respective modules).
  + Facilitate communication/feedback submission from Parent/Student/School modules to the OPT Admin module.
  + Support multi-language communication based on user preferences.
  + Integrate with external collaboration/messaging platforms used by NYC agencies if required.
* **Key Data Inputs:** Event triggers (from GPS, Ridership, Routing, User Modules), user contact preferences, opt-in/out status, manually initiated alert content/targets (from OPT Admin), configured alert rules.
* **Key Data Outputs:** Notifications (in-app, SMS, email, robocall), message logs.
* **Interactions:** All other core components (receives event triggers), User Modules (delivers notifications, receives messages), External messaging platforms.

2.5. User Modules

**General Requirements:** Mobile-first design (iOS/Android native apps for Parent/Student, Driver), responsive web design for Admin modules and optional Parent/Student web access. WCAG 2.0 AA compliant. Multi-language support (9+ official NYCPS languages). Role-based access control (RBAC) managed via OPT Admin module. Troubleshooting/FAQ sections.

* **2.5.1. Parent / Caregiver and Student Module:**
  + Separate Parent/Caregiver and Student access levels via signup/login.
  + View map display of assigned bus route(s) and near real-time bus location.
  + View ETAs for pickup/drop-off.
  + Receive configurable "approaching bus" notifications.
  + Receive student board/alight notifications (opt-in/out).
  + *Parent Only:* Indicate student absence for the day.
  + *Parent Only:* Request updates to student info (address, alternate PM drop-off) - subject to OPT approval workflow.
  + *Student Only:* Display scannable QR code for boarding/alighting.
  + Submit feedback/report issues to OPT Admin.
  + *Offline Capability:* Ability to view static route/stop info if offline.
* **2.5.2. Bus Driver Module:**
  + Secure login (potentially with biometric option, credential management).
  + Driver/Attendant ID capture, association with assigned route(s) and vehicle.
  + Display assigned route (map & text view) with turn-by-turn navigation (audio/visual), dynamically updated by Routing Engine.
  + Simplified interface for ridership recording (scan QR/barcode, manual entry for exceptions). Display student roster for current stop.
  + Receive alerts (traffic, weather, breakdown, student conduct, messages from OPT Admin/Router).
  + Send alerts/messages to OPT Admin/Router (breakdown, delay, misconduct, etc.).
  + *Potential Add-on:* Display driver behavior monitoring feedback/alerts (speeding, braking, etc.).
  + *Offline Capability:* Ability to view assigned route map/sequence and record ridership data offline (for later sync).
* **2.5.3. School Module:**
  + View near real-time location/ETA for all buses serving the school.
  + View assigned routes and student roster information for the school.
  + Isolate specific routes to view vehicle location and student ridership status.
  + Receive near real-time alerts for issues affecting school routes (delays, missed pickups).
  + View school-specific KPIs (buses active, students en route, on-time performance).
  + Report issues/problems to OPT Admin module.
* **2.5.4. OPT Administrative Module:**
  + System-wide map view displaying all buses, routes, drivers, students in near real-time. Ability to filter/search.
  + Differentiate users by persona/scope (RBAC).
  + Manage user authentication/authorization for all modules.
  + Comprehensive communication tools (initiate/receive alerts/messages to/from all other modules).
  + View system-wide KPIs and near real-time event 'Newsfeed'.
  + Route replay functionality.
  + Configure GIS elements (geofences, speed limits, street segments) for Routing Engine.
  + Configure system-wide alerts and notification rules.
  + Access detailed reporting interface (canned & custom reports).
  + Manage device inventory and ticketing system interface.
  + Override functions (e.g., driver authentication, route deviations - with audit trail).
  + Search/view driver details, route history, GPS history, etc.
* **2.5.5. SBC Administrative Module:** (As agreed post-award)
  + View near real-time location/ETA for all buses associated with the SBC.
  + View assigned routes and potentially basic driver/vehicle association for their fleet.
  + Receive alerts relevant to their operations.
  + (Potential) View basic performance KPIs for their fleet.
  + (Potential) Interface with ticketing system for device issues.
  + (Potential) Suggest route/headcount changes for OPT review.
  + (Potential) View/print route sheets.

2.6. Student Management (Backend) System

* **Responsibilities:**
  + Serve as the central repository for student transportation-related data.
  + Maintain student profile information relevant to transportation (ID, name, grade, school, home address(es), contact info, special needs codes (IEP/medical/ambulatory status), eligibility status, exception types (STH, Foster, etc.), assigned routes/stops).
  + Synchronize necessary data with upstream NYCPS systems (ATS, NPSIS, IEP systems, DOHMH IFSP data, potentially NYCSA) via APIs or other secure methods.
  + Integrate seamlessly with the Parent/Caregiver module for receiving update requests (address changes, absences) and enrollment updates.
  + Integrate with School Admin module for receiving updates (student join/leave).
  + Provide authoritative student data (including rosters, special needs) to the Dynamic Routing Engine and Ridership Module.
  + Manage student-stop assignments based on rules/requests.
  + Manage parent/caregiver opt-out status for ridership tracking.
  + Ensure data integrity and consistency across integrated systems.
  + Store historical student transportation data (7 years).
* **Key Data Inputs:** Data feeds from NYCPS systems, updates from Parent/Caregiver/School modules, stop assignment data, opt-out flags.
* **Key Data Outputs:** Authoritative student roster/profile/needs data for Routing and Ridership, data for User Modules, historical student data for Reporting.
* **Interactions:** NYCPS Enterprise Systems (ATS, NPSIS, etc.), Parent/Caregiver Module, School Module, Dynamic Routing Engine, Ridership Module, Reporting & Analytics Subsystem.

2.7. Reporting & Analytics Subsystem

* **Responsibilities:**
  + Collect and store historical data from all other components (GPS tracks, ridership events, route plans/actuals, notifications, user actions, system logs, KPIs).
  + Provide an interface within the OPT Administrative Module for generating canned and customized reports.
  + Support reporting requirements mandated by NYC Council legislation (route durations, on-time performance, delays, complaints, etc.).
  + Enable filtering, sorting, and aggregation of data by various attributes (SBC, garage, school, district, route type, student type, time period, etc.).
  + Calculate and display system KPIs (real-time and historical).
  + Allow export of reports and underlying data in standard formats for use in third-party tools (e.g., Excel, BI platforms).
  + Support ad-hoc querying capabilities for authorized OPT users.
  + Potentially feed a dedicated data warehouse/data mart for more advanced analytics.
  + Ensure reporting access is controlled via RBAC.
* **Key Data Inputs:** Historical data streams from all other system components.
* **Key Data Outputs:** Canned reports, custom reports, dashboard visualizations, data exports.
* **Interactions:** All other core components (receives data), OPT Administrative Module (provides reporting interface), potentially external BI tools/data warehouse.

2.8. Device Management & Inventory System

* **Responsibilities:**
  + Maintain a comprehensive inventory of all GPS-enabled devices (tablets/phones) and associated hardware (ID readers, mounts).
  + Track device lifecycle attributes (make, model, serial number, assignment to SBC, location, status - active/in repair/lost, installation/removal dates, repair history).
  + Support tracking of portable devices assigned dynamically to drivers/routes/vehicles daily.
  + Integrate with ticketing system for managing installation, repair, replacement, and transfer requests.
  + Provide data for OPT Admin module dashboards/reports regarding device status and inventory levels.
  + Support remote device management capabilities (e.g., remote data wipe for lost/stolen devices, configuration updates) via an MDM solution.
* **Key Data Inputs:** Device serial numbers, assignment data, status updates from field/ticketing system, MDM commands.
* **Key Data Outputs:** Device inventory reports, device status dashboards, data feeds for OPT Admin Module.
* **Interactions:** OPT Administrative Module, Ticketing System Interface, MDM Solution, potentially Bus Driver Module (for device check-in/out).

2.9. Ticketing System Integration Interface

* **Responsibilities:**
  + Provide a robust, bi-directional interface (likely API-based) between the vendor's ground support ticketing system and the NYCPS ticketing system.
  + Facilitate automated creation, updating, tracking, and closure of tickets related to GPS device installation, repair, transfer, removal, and system support requests initiated from either system.
  + Transmit necessary ticket details between systems (ticket ID, requestor, device ID, vehicle ID, location, issue description, status, resolution details, timestamps).
  + Ensure data consistency between the two ticketing systems.
* **Key Data Inputs:** Ticket creation/update requests (from NYCPS or vendor system), status changes, resolution details.
* **Key Data Outputs:** Ticket data synchronization messages, confirmation receipts.
* **Interactions:** Vendor Ground Support Ticketing System, NYCPS Ticketing System, OPT Administrative Module (for reporting/visibility), Device Management & Inventory System.

## 3. Cross-Cutting Concerns

* **Security & Compliance:** Authentication, Authorization (RBAC), Encryption, Audit Logging, Data Privacy (FERPA, NY Ed Law 2-d, etc.), Compliance with NYC3/OTI/DIIT policies will be implemented across all components.
* **Scalability & Performance:** Components will be designed to handle peak loads and scale dynamically. Near real-time data processing and low latency user interfaces are critical.
* **Availability & Reliability:** High availability architecture (targeting 99.99%-99.999999% depending on component) and disaster recovery plans (meeting RPO/RTO targets) are essential.
* **Usability & Accessibility:** All user-facing modules will adhere to WCAG 2.0 AA standards, support multiple languages, and follow modern UX design principles.