ADDENDUM NO. 1

Date: September 3, 2015

Bid Date: October 1, 2015
3:00 P.M. (Local Time)

Bid Name: Fixed Route Scheduling Software

Bid No.: RTSX-160004-DS

NOTE: This Addendum has been issued to the holders of record of the specifications.

The original Specifications remain in full force and effect except as revised by the following changes which shall take precedence over anything to the contrary:

1. I received an email request to provide editable access to the MS-Office-formatted Tables 1 through 5. This Addendum #1 has been issued to advise that the requested format (also changed to 8-1/2" x 14" legal size) has been uploaded separately for proposer's use.

ACKNOWLEDGMENT: Each Proposer shall acknowledge receipt of this Addendum No. 1 by his or her signature below, **and shall attach a copy of this Addendum to its proposal.**

CERTIFICATION BY PROPOSER

The undersigned acknowledges receipt of this Addendum No. 1 and the Proposal submitted is in accordance with information, instructions, and stipulations set forth herein.

PROPOSER: _____

BY: _____

DATE: _____

Table 1 TSS Functional Requirements

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
1. General System Requirements			
1.1.	Basic function of the scheduling system shall be to define transit routes, service level requirements, and produce schedules efficiently and quickly meeting planning needs, supporting local government requirements, and public expectations. Additionally, scheduling system shall also be able to produce vehicle itineraries and driver runs, complying with union agreements (see Appendix 8.) and RTS work practices with a focus on minimizing vehicle requirements and driver operational costs.		
1.2.	Scheduling system must fully support related and proposed future ITS systems (TIS, AVL, APC, etc.).		
1.3.	Scheduling system satisfies RTS's current operating policies, procedures, practices, and work rules.		
1.4.	All components of the scheduling system, e.g. schedule planning, route definitions, runtimes data, schedule building, vehicle itinerary generation, driver run creation, weekly off day assignments, driver run bidding, day-to-day operational management, and driver pay calculations, must fully integrate within themselves, without any need of data import/export within these components.		
1.5.	All components will be compliant with National Transportation Communications for ITS Protocol (NTCIP) and the Transit Communications Interface Profiles (TCIP) and follow the provisions of FTA National Architecture Policy on Transit Projects.		
1.6.	All components must conform in every respect to the standards, guidelines, and regulations established by Federal and Florida state laws.		
1.7.	Technical support shall be available via phone, e-mail, web site resources, or on-site support. The proposer must respond to support critical operational issues within 24 hours. Ability to provide telephone technical support during business hours (between 8:00 am to 5:00 pm ET) for non-emergency issues.		
1.8.	All equipment and interfaces shall meet or exceed all Americans with Disability Act (ADA) requirements at the time of implementation.		
2. Operating Environment			
2.1.	The system shall be using three or four tier web/application/database, or based on Microsoft Windows client-server architecture.		
2.2.	Operate within a Microsoft Windows Server environment and shall provide a 32-bit and 64-bit Windows 7 operating system or above or Web-based client. (Specify which of these is available and the degree of functionality each provides.).		
2.3.	The system must be in use at five transit systems at least as large as RTS for like size and complexity (fixed route minimum peak vehicle requirements).		
2.4.	Data communications shall be based on standard open protocols that conform to the Open Systems Interconnection (OSI) seven-layer model. These protocols shall include IEEE 802.3 Ethernet with Transmission Control Protocol (TCP)/Internet Protocol (IP) for LAN communications, wireless LAN, and the use of IP for Wide Area Network (WAN) communications.		
2.5.	All workstations shall use Dynamic Host Configuration Protocol (DHCP) for IP address assignment.		
2.6.	Support Microsoft Windows 7 enterprise and above for application tier.		
2.7.	Allow users to extract data from the database using third-party software and/or programs with the export/import utility being intuitive and easy to use.		
2.8.	Support minimum of 20 concurrent users with <1 second GUI response time for system operations and with full database integrity maintained.		
2.9.	Provide consistency among data (decimals, dates) entered, stored, and displayed.		
2.10.	All software must be written in industry standard programming language(s).		
2.11.	Ability to conform to the latest version of National Transportation Communications for ITS Protocol (NTCIP) Standard 1404 on Scheduling/Runcutting (SCH) Objects.		
2.12.	All components of the scheduling system must fully conform to COG IT standards, allowing standard functions, including those related with screen sizing, cut/paste, row/columns/screens suppression/hiding/resizing/printing, data transfers to spread sheets, pdf formats, etc.		
2.13.	System architecture design shall be based on open standards, provide incremental growth path to permit expansion as needed, and provide application interface with various internal and external software systems to enhance data access.		
2.14.	The system must track connectivity between the various parts of the back-end system and email a list of addresses if a suspected problem occurs. This		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	includes monitoring server connection, memory usage, database connection, communication middleware connection, number of applications connected, and web gateway connection.		
3. Database Standard			
3.1.	The system shall be based on the Microsoft Structured Query Language (SQL) Server relational database management system (RDBMS). The system's database shall be open, COTS, and fully documented to allow 3rd party reporting tools, such as Crystal Reports, to be used, as well as common database management and analysis tools (e.g., Microsoft Access, Excel). (NO interfacing license fees or access fees to import to or export data from external applications or systems). Proposer will provide at no additional cost a data dictionary (including primary keys, table names, table relationships, table descriptions, table indexes, field names, field descriptions, field data types, field default value settings, field validation rules, and any other component of the data structure), data flow, and entity relationship diagram/schema for all data tables.		
3.2.	The system RDBMS must provide high availability, data protection for failover, and disaster recovery.		
3.3.	All data must be stored in a manner so that any data changes must be made only once.		
3.4.	Provide separate simultaneous database environment to support training and/or testing needs.		
3.5.	The system shall have the capability for simultaneous access of data by multiple users with record-level locking to prevent data conflicts.		
3.6.	Provide access to all database tables and fields to authorized users.		
3.7.	Data shall be retained and manipulated as relational files using common database routines for definition and access. All parameters needed for administration shall be available through system administrator screen. Tools shall be provided for performance measurement and analysis.		
3.8.	Support standard SQL/Open Database Connectivity (ODBC)/Java Database Connectivity (JDBC) access in order to create custom reports.		
3.9.	Provide well-documented tools for system administration, including all required tasks related to data import, backup, rollback, archiving, restoration/recovery, and purging, performance monitoring, and installation of system upgrades. These actions shall be made possible by a Graphical User Interface (GUI) or via the command line for automating tasks.		
3.10.	The system shall back-up the data on a regular basis and provide a clear, simple procedure for restoring data from one of the daily back-ups in the event that undesired changes are accidentally made.		
3.11.	Data and database schema more than five years old shall automatically be moved to a secondary database(s) and deleted from the primary database.		
3.12.	Users must have the ability to easily export to a comma separated value (CSV) format any tabular data displays simply by selecting the data grid desired. Subsets of a full data table (i.e. filtered or sorted values) must be exported with the same conditions applied to the exported values.		
3.13.	Enough data storage shall be provided to keep at least five years of historical data.		
3.14.	If there is a catastrophic failure that results in the loss of data, the proposer shall provide a means to retrieve the corrupted data without disruption to system operations.		
3.15.	The historical database shall be read-only.		
3.16.	It shall not be necessary to shut down the database to perform a successful transfer to a historical information database.		
3.17.	Archived data shall be structure in a way that is optimized for later retrieval, analysis, and reporting.		
3.18.	The system shall consist of a backup or mirrored server that will take over operations should the main server fail. The mirrored server shall enable failover operations to be carried out in a seamless manner requiring minimal manual intervention.		
3.19.	The system shall maintain records of all versions of the back-end programs, user interfaces, configuration files, and executables that are either received from the server or created and that are successfully loaded and running.		
4. Interfaces			
4.1.	The system shall provide a tested interface for data exchange with (as applicable) AVL, Fleet-Net Operations system, UTA APC units, etc. and must have previously been integrated with at least three systems in a production environment. Please specify manufacturer and model of installations. Interfaces rely on standard programming languages for integration. No licensing restrictions on interfacing data to or from scheduling database. RTS is the owner of all software data.		
4.2.	The system must automatically integrate running times at all levels (route, pattern, time point, stop, day of week, time of day) from AVL proposer and		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	offer graphical, querying, and reporting interfaces to analyze the data for schedule building purposes.		
4.3.	The system must automatically provide all necessary schedule information to Fleet-Net Operations software for timekeeping and payroll processing and to track items like bid assignments, overtime, scheduled work days, scheduled days off, cancelled service, etc.		
4.4.	The software shall be compatible with or offer functionality similar to Time Table Publisher, which formats data quickly for printing of schedules. Provide a description of the process and examples. No licensing restrictions on interfacing data to or from scheduling database. RTS is the owner of all software data.		
4.5.	The system shall provide a General Transit Feed Specification (GTFS) interface to export scheduling output files used by the Google Transit trip planner (http://www.google.com/transit), as well as GTFS-real-time. No licensing restrictions on interfacing data to or from scheduling database. RTS is the owner of all software data.		
4.6.	The software shall be able to interface with Microsoft Office 2010.		
4.7.	Ability to provide functionality for selective copying/sharing of data across scheduling system databases.		
4.8.	Ability to keep uniform look-and-feel and uniform functionality across all elements of the application.		
4.9.	External actions (performed by third party applications) can be initiated from the client, application, or data tier.		
5. Client Functionality			
5.1.	The system shall use Microsoft Windows system properties for fonts and colors wherever applicable and be provided within the traditional Microsoft Windows paradigm (e.g., dock-able windows, standard windows keyboard shortcuts (i.e. CTRL+C, CTRL+V, etc.), drop-down menus, toolbars, mouse clicks, etc.).		
5.2.	Ability to provide keyboard commands/shortcuts for all commonly used or repetitive application functions.		
5.3.	The system shall support the Universal Naming Convention (UNC) and long filenames.		
5.4.	The system shall be able to allow for unlimited "undo" / "redo" steps to the previous command, or warn users if certain changes cannot be "undone".		
5.5.	The system shall be able to perform all printing functions via the Windows Print Manager.		
5.6.	Ability to adjust the display of columns and rows.		
5.7.	Groups, window layouts, configuration settings, and screen preferences shall be savable on a user-by-user basis.		
5.8.	An on-line help feature shall be included with the system. It must provide context-sensitive help information for commands, menu items, and screen options.		
5.9.	The system shall include toolbars (iconic command buttons) that are configurable by a system administrator.		
5.10.	The system must have configurable lists with the ability to show lists related to a primary list. For instance, a list of runs with a related list of trips.		
5.11.	User shall be able to resize, hide or show table columns any time and modify the displayed font.		
5.12.	The grid control for configurable lists shall support click-and-drag column resizing, drag-and-drop column reordering, data sorting (both ascending and descending), easy selection of columns to be viewed (add/delete columns).		
5.13.	Configurable lists must include options to print and export the content of the list (into Excel format for example).		
5.14.	The system must include tools to select objects (i.e., trips, blocks, stops, etc.) based on user-defined characteristics.		
5.15.	The system shall include the ability to save selection criteria.		
5.16.	Software shall have auto-save & auto-recovery, versioning, and file archiving.		
5.17.	The system must include tools to group objects according to criteria specified by the user. Then, it must be possible to produce statistics by groups, for example the total distance of trips grouped by route.		
5.18.	The system shall be able to display and print histograms based on the characteristics of objects.		
5.19.	Pre-programmed function keys (hot keys) shall be an integral part of the user interface, permitting easy access to and manipulation of data for the user.		
5.20.	The system must support data import and export of the main entities data in standard Windows file formats including Microsoft Excel, XML, or comma-delimited American Standard Code for Information Interchange (ASCII) text files.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
5.21.	If display content is larger than the display, scroll bars shall be used to enable rapid viewing of all display content. For tabular displays that must be scrolled, the row and column headings of the table shall be stationary so that these headings can be viewed regardless of the scroll position.		
5.22.	The system shall be capable of displaying time units in hours and/or minutes; time shall be able to be expressed in AM/PM (APX) and military (24-hour) time.		
5.23.	The system shall provide the ability to easily add user-defined fields in support of additional or new data requirements.		
5.24.	The system shall be have both tabular and graphical views as appropriate.		
5.25.	The system shall be able to turn on and off graphical views of data.		
5.26.	The system shall be able to have all tabular views with multi-column sorting unless logically inappropriate.		
5.27.	The system shall provide easy user navigation.		
6. Configuration Parameters			
6.1.	The system shall retain all configuration and preference changes when software upgrades are applied, both on the server and on workstations.		
6.2.	The application shall retain all configuration and preference changes for each user, regardless of where the user logs on.		
6.3.	Changes to parameter files shall be immediately reflected in current data.		
6.4.	Configuration of the proposed program via parameter files, option screens, and the like shall be sufficient to tailor the proposed system to meet RTS requirements and business rules. Source-code-level customization shall not be necessary.		
7. Data Validation			
7.1.	The system shall perform data validation and check for data reasonableness at the point of data entry before allowing the data to be processed or used by the system and shall reject any invalid data. This includes all input data, parameters, and commands whether collected automatically or entered by a user. Checks include things like invalid running times, trips, etc. Provide details.		
7.2.	The system shall include the ability to define validation criteria, for instance that block duration shall not exceed 16 hours.		
7.3.	When unreasonable input data or results are detected, diagnostic messages clearly describing the problem shall be generated. These messages shall be unabbreviated English text and shall not require the use of a reference document for interpretation or diagnostic or other complex data or descriptions intended for maintenance personnel.		
7.4.	Diagnostic data shall be logged for later retrieval by the system administrator.		
7.5.	The system's user interface must be uniform and consistent in its use of menus, buttons, function keys, and screen designs. Each screen or window must have its own unique identifier providing the user with a clear indication of the screen function.		
8. Data Entry			
8.1.	The system shall respond to all user input actions indicating whether the action was accepted, was not accepted, or is pending. For multi-step procedures, the system shall provide feedback at each step. Indications such as text messages, color changes, and blinking shall provide this feedback.		
8.2.	The amount of data required to enter shall be minimal. The system shall insert any data that is already known (e.g., date, time, user identification, vehicle identification, Operator ID) and provide default values where appropriate.		
8.3.	When data entry of a field is limited to a known set of valid responses, the list of valid responses shall be presented to the user in the form of a scrollable list.		
8.4.	For all data fields if only a portion of a data value needs to be changed, only that portion of the value shall need to be entered.		
8.5.	Data entry can be cancelled at any time and this action shall cause the process to be terminated and the data values cleared.		
8.6.	Data entry can be suspended at any time by requesting a different display or window. This action shall cause the process to be suspended and the data value shall remain unchanged until display or window is returned to.		
8.7.	Critical actions initiated by users, such as a deletion, shall be performed only after a warning message and request for confirmation are issued to the initiating user and the confirmation of the intended action is received from the initiating user.		
9. Geographic Information System			
9.1.	The RTS GIS software standard is Environmental Systems Research Institute (ESRI). This will be used as the basis for any and all system processing,		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	storage, and display of geographic data including basemap(s), bus vector data, and AVL data. The proposed software shall be able to integrate with RTS GIS software and consume RTS GIS feature classes. Specifically, the application shall provide an integrated GIS that can display a centerline street network from most common GIS data sources (shapefiles, etc.). Therefore, RTS shall be able to create within the TSS software network components (stops, time points, routes, route variants, etc.) that automatically sync with RTS's transit geodatabases without the need to import/export any data. Fully document process and integration of GIS and scheduling data.		
9.2.	The application must allow for the localization of stops/stations, public locations/landmarks on the street network.		
9.3.	All GIS components of the scheduling system must fully integrate (read and write) with RTS's GIS for purposes of maintaining bus stops inventory, time points, bus route definitions and street paths, with full integration with underlying ESRI basemaps and centerline data. The intention of this requirement is to disallow data duplication. The mentioned data are to be written to and maintained within RTS GIS database. Integration is at the database level.		
9.4.	The system shall be supplied with a geocoding feature built-in which can be used to geocode time points, bus stops, and other pertinent routing locations.		
9.5.	The geocode feature shall permit the user to define bus stops using a variety of methods, including direct entry of GPS determined coordinates and setting the stop location with a mouse click.		
9.6.	The system shall be capable of allowing stops to be properly positioned at intersections.		
9.7.	The system shall be capable of displaying all trip patterns, or fixed portions of flexible trip patterns, on a map for visual display.		
9.8.	Route and pattern trace definitions, modifications, stop inventory information, including stop sequencing in patterns shall be fully integrated with the existing RTS GIS system for maintenance of stops and related assets, including maintenance of these data by writing into RTS's GIS geodatabase. At the point of consumption, these data shall be used directly from RTS GIS via geodatabase reads, without the need for file based import of data or the need to move between different software.		
9.9.	The system must allow for the multiple locations of bus stops to be displayed, including but not limited to - a location on the centerline network and a public location along but not on the street network. Multiple locations available as GIS feature classes from RTS GIS shall be used.		
9.10.	The system shall display fixed-route paths from RTS GIS based on their scheduled stops, including branching (expected turn-by-turn route, with stops displayed).		
9.11.	The system must be able to automatically calculate and store distances, times, and speeds for deadhead route segments, entire trip patterns, and selected trip pattern segments defined by the user using the street network and GIS feature classes.		
9.12.	The application must be able to compute stop-to-stop distances based on the street network information. The resulting itineraries shall be editable by the user.		
9.13.	In case stop distances computed by the GIS do not reconcile with times computed by the scheduling system, the scheduling system shall be able to export a list of such segments for review within GIS, as well as maintain a second set of stop to stop distances.		
9.14.	The application shall allow for integrated editing and easy replacement of map files.		
9.15.	Functionalities shall be available to automatically re-localize objects following an update of the underlying GIS map network data. Explain the process.		
9.16.	System shall be able to assign sequence number to newly created or maintained bus stops stations in GIS. Workflow is 1) stop feature is created in GIS; 2) scheduling system assigns sequence; 3) scheduling system reads/uses from GIS.		
9.17.	The GIS component shall have the capability to export the street network data to external systems or applications in shapefile format.		
9.18.	The system must allow for the definition of fare zones by route, system-wide, time of day, and day of week.		
9.19.	The map display shall support zoom in, zoom out, and pan.		
9.20.	The map display must show one-ways, forbidden turns, barriers.		
9.21.	The GIS component shall have an automatic coordinate converter to convert coordinate locations from state plane (x-y) to latitude/longitude, and from, latitude/longitude to state plane (x-y).		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
9.22.	The system must allow and maintain a log of the street network edits (new street segments, street segment modification, street segment deletion) posted to GIS such that upon a upgrade to underlying GIS street data, segments previously custom edited but included parts of newly delivered data are identifiable and are discarded following succession rules to the new segments.		
9.23.	The GIS component shall be seamlessly accessible from all other components.		
9.24.	The system must be able to work with spatial data in WGS84 coordinate system (latitude/longitude).		
10. System Performance			
10.1.	The system shall be designed for and capable of 24 hour per day, 7 day per week operation.		
11. Security			
11.1.	The system provides the system administrator with the following group-level security features: • Control over a group member's access to each specific subsystem of the application. • Control over a group member's access to specific screens and fields within such screens. • Control over a group member's access to specific commands and functions. • Control over a group member's rights to view, add, modify, or delete specific data elements and records. • Control over a group member's right to add, modify, or delete screens, menus, database columns, and reports.		
11.2.	The system provides the system administrator with the following user-level security features: • Control over an individual user's access to specific screens (windows). • Control over an individual user's access to specific commands and functions. • Control over an individual user's rights to view, add, modify, or delete specific data elements and records. • Control over an individual user's rights to add, modify, or delete screens, menus, database elements, and reports.		
11.3.	Each level of access shall be capable of specifying read/write, read, or no access to each identified system function.		
11.4.	A minimum of three user-access levels shall be supported by the system: • Information user: these users shall have read-only access to the system. • Editor: these users shall have full access to specific system functions as determined by the system administrator. • Administrator: these users shall have unrestricted access to system functions and shall have special privileges required to administer overall access security and to maintain the system. A secure method shall be provided for the system administrator to change passwords and user identifications and establish functional partitions that identify a subset of all system data, including events that users are permitted to access.		
11.5.	Users without proper minimum authorization shall be denied access to all system functions and data, as well as all system resources such as servers, printers, workstations, etc.		
11.6.	Access to system functions and capabilities shall be based upon each user's authorization level and not the physical workstation.		
11.7.	The logon/logoff status of a user shall be unaffected by any failure recovery procedure in the system.		
11.8.	The system shall support single sign on and integrates with COG active directory (Windows Domain authentication) so that additional passwords are not required.		
11.9.	The system logs all login attempts, successful logins and logoffs to the system (including the time, date, workstation ID, and user ID of the login attempts).		
11.10.	Ability to record and date/time stamp all transactions and include user ID and rollback any change within a 30 day window.		
11.11.	The activity log shall be real-time and accessible.		
12. General			
12.1.	A warranty period of five years that begins after all RTS's data has been converted and the entire system is accepted in writing or used in a production (operational) context by RTS.		
12.2.	Ability to provide a complete online help facility based on industry standards that allow users to search by keyword or by task.		
12.3.	Ability to provide established quality assurance procedures and documentation.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
12.4.	Ability to provide software manuals as electronic files in MS Office and PDF formats on DVD or equivalent.		
12.5.	Ability to provide comprehensive plans and be responsible for testing, repairing, and complete installation of the software.		
12.6.	Ability to provide on-site hands on training to RTS staff.		
12.7.	Ability to provide documentation for each function.		
13. Scheduling - General			
13.1.	Ability to create, maintain, and manage multiple numbers of schedule databases for handling current, previous, future, and simulation types of schedules.		
13.2.	The system shall be able to create new and update existing routes, patterns, time points, and stops. Describe/illustrate the graphical methods of viewing and working with schedules.		
13.3.	The system shall have the capability for the creation of multiple schedules of up to nine service frequencies for each route.		
13.4.	The system shall be capable of allowing the user to assign stop amenities (e.g., bench, shelter, etc.) to each stop and other supplemental data.		
13.5.	The system shall have the capability of browsing single or multiple records at one time and on a single screen.		
13.6.	Ability to perform a batch copy and renaming of schedule data completely including patterns, running times, trips, blocks, runs, and rosters from one sign-up to another sign-up schedule and from one service group to another service group schedule.		
13.7.	The system shall have the capability to freeze versions of a schedule once finalized.		
13.8.	User defined and configurable lists for all components, including trips, headways, blocks, patterns, etc.		
13.9.	Ability to create alternative schedules during same period to evaluate different scenarios.		
13.10.	The system shall have the capability to track the effect of service changes in terms of hours and miles by route, pay period, and location and how this affects the cost of service.		
13.11.	Ability to accommodate 99 or more time points and stops per pattern (specify if a maximum exists for any parameter).		
13.12.	Ability to accommodate infinite patterns (or variants) on a route. Patterns defined as distinct bus stop sequences, including the designation of selected stops in each trip pattern as schedule time points and whether a trip pattern is inbound or outbound.		
13.13.	Ability to generate patterns within a line/route without redefining running time and time points.		
13.14.	The system shall be able to accommodate loop routings, i.e., routes that operate in one direction where the origin and destination points are the same.		
13.15.	Ability to establish a sequential listing of time points by route/line and direction.		
13.16.	The system shall permit the specification of any time point for sorting trips on the screen or on reports to be printed.		
13.17.	Ability to define time points that are not actual bus stops.		
13.18.	The system shall have the capability to save, retrieve, and modify multiple versions of trips, running times, blocks, and routings.		
13.19.	The system shall have the capability to support the development of multiple schedules by time of day, time of year, and various day types.		
13.20.	The system shall have the capability for time points to be added, deleted, inverted, or modified even where scheduled trips already exist. Such modifications will cause the scheduled trips to be updated automatically.		
13.21.	Define and assign route types (i.e. regular, express, special, etc.).		
13.22.	Ability to assign multiple vehicle groups to a route.		
13.23.	Ability to develop feasible connections with other carriers (e.g. other service proposers).		
13.24.	The system shall be able to store and display both original route and detour route information.		
13.25.	Automatically or manually assign trip, block, run, and roster numbers based on user-specified sorting order and criteria.		
13.26.	The system must be able to support a 36 hour operational day.		
13.27.	The system must allow for the definition of all pay time components. This includes report time, travel time, wait time, deadhead time, platform time, clear time, scheduled overtime, spread time, make-up time, etc.		
13.28.	The system shall offer solutions to gradually increase or decrease bus service frequencies to meet demand by time of day, offering easy "what-if' analysis.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
13.29.	The system shall allow trips, blocks, routes, and associated non-revenue time to be flagged by funding source. Funding amount can be defined by percentage or hour.		
13.30.	The system shall offer special event capabilities to completely plan special event transit services in a few hours. It shall quickly and optimally determine vehicle and driver requirements, integrate and determine the impact of the service on RTS's regular service, and quickly determine all costs and requirements needed for special events.		
14. Trip Building			
14.1.	Capable of automatic or manual trip building for each route, using the designated sequence of trip patterns during defined time periods.		
14.2.	Able to copy trips as follows: • One schedule to a new schedule (i.e., fall to spring) • One or many trips from one schedule to another schedule • One trip within the current schedule • Special trips from one schedule to a new schedule (i.e., sporting events, special events, etc.)		
14.3.	Through the connection of trips, the system shall automatically calculate the vehicle requirements for twelve or more specific user defined time periods of the day (such as AM peak, Base, PM peak and evening) for each service.		
14.4.	Ability to create or modify trips individually, for a specified headway, for a time interval, for a specified number of vehicles and synchronize with the passing times of other trips at common timing points.		
14.5.	The system shall have the capability to unhook all trips at one time.		
14.6.	Ability to create trips based on the passing time at any one of the trip timing point and ability to modify time point information in existing trips, such as 1) change time point passing time at any one of the trip time points, 2) change the name of time point, and 3) change run time between time points, etc.		
14.7.	Ability to modify selected trips (for example, single trip pattern or all trip in a single direction) for instance to change the origin, the destination, the running time, and the trip pattern.		
14.8.	Ability to modify trips based on a specified time point and shift all time points in a trip (positively or negatively) by a user specified amount of time in one step.		
14.9.	Define and manage transfer connections by viewing trips from more than one route together.		
14.10.	Ability to clearly display graphically and/or in tabular format the times of all trips and routes traveling through a common time point.		
14.11.	Ability to provide both tabular and graphical view of trips, including nodes, trip times, block numbers, assignment numbers, deadhead and recovery time, as well as the ability to manage, create, and make changes directly to trips in the graphical application. The system shall have the capability to sequence trips so that they are displayed in a logical progression on the screen and on reports.		
14.12.	Ability to identify/define trips within the software structure to denote revenue and nonrevenue trips.		
14.13.	Ability to automatically generate return/connected trips with user specified minimum/maximum layover time, which may be location or time of day specific.		
14.14.	Ability to vary routings on service patterns to cater to specific capacity limitations, such as number of platforms available at each station.		
14.15.	Ability to build conflict-free timetables system wide using configurable parameters for different service plans.		
14.16.	Allow user to assign specific vehicle types and/or vehicle characteristics to trips, blocks, routes, and time of day basis and have ability to override or enforce vehicle assignments at scheduling level.		
14.17.	Ability to allow for different running times between time points by direction, time of day, day of week, route.		
14.18.	The system shall have the capability to override the automatic assignment of running times for individual trips. A log of overrides shall be created listing the user and when the changes were made.		
14.19.	The system shall have the capability to maintain running time overrides between subsequent schedules.		
14.20.	Ability to support the definition of base running times and speeds for various days and time periods between two timepoints, designated stops along		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	each trip pattern, and for deadhead segments and apply to multiple routes where they are in common (i.e., new routes using the two timepoints would automatically inherit the defined base running time.		
14.21.	When adjusting running times by time of day by route the software will maintain intervals between buses at stops (either through interlining or manipulation of layover amounts) which serve multiple routes.		
14.22.	The proposed system shall have the capability to show the distance between time points and the associated speeds.		
14.23.	Capable of calculating the running time between any pair of time points on any trip pattern, taking into account running speeds and via integration with AVL data.		
14.24.	Allow running times to be manually input or adjusted for trips, patterns, or between stops or timepoints.		
14.25.	Ability to allow for variations on deadhead running times as a function of varying running speeds by time of the day.		
14.26.	Ability to automatically detect interlining opportunities between different routes at common endpoints without manual intervention.		
14.27.	Ability to change the running time between 2 time points and automatically recalculate all trips that were built using that running time.		
14.28.	Ability to control the displayed timing points in the various displays.		
14.29.	Ability to allow or prohibit interlining between routes via a user defined parameter.		
14.30.	Ability to create route groups for interlining.		
14.31.	The system shall notify the user when an assigned routing is missing a running time.		
14.32.	Ability to define interlined routes before assigning blocks.		
14.33.	Ability to allow dispatching from more than one garage or location (Ability to assign multiple garages/yards to a route, multiple routes to a garage/yard, etc. per user's discretion).		
14.34.	Specify destination sign code for each trip.		
14.35.	The system shall have the capability for creating trip exceptions due to special events or days (e.g., trips that only operate on certain days of year).		
14.36.	Ability to define minimum/maximum layovers globally, at trip start, at trip end, by place, by route, by schedule type, by time of day, and by combinations of these, and as a fixed value or as a percentage.		
14.37.	The system shall have the capability for the connection of trips using all of the following or combinations thereof: one route at a time, multiple routes, individual trips, trips of a specific routing or running time, any unconnected or unhooked trips from another route, specified interlines.		
15. Blocking			
15.1.	Ability to automatically generate blocking scenarios with respect to pre-defined criteria and business rules to achieve optimized vehicle blocking solutions that meet the RTS service and operational requirement. Options shall be provided to block based on criteria like minimizing the number of vehicles or total vehicle hours required. Manual blocking shall also be allowed. Provide a detailed description of blocking optimizing algorithms.		
15.2.	Ability to accept, reject, or edit automatic blocking solutions so RTS can simulate different block solutions for the same trip data before committing to one.		
15.3.	Ability to create, delete, and modify deadhead trips automatically to make trip connections and to automatically assign yard pull-in and pull-out deadhead trips upon completion of block.		
15.4.	Ability to add a control point between deadhead trips to allow a vehicle to hold safely on a pocket or spur between revenue trips to avoid same location conflicts on the revenue trips.		
15.5.	Able to load simultaneously multiple vehicle schedule scenarios (for comparison, for copying trips from one to the other, etc.).		
15.6.	Ability to display blocking graphically on a horizontal time scale and display information such as timepoints, layover time, revenue time, non-revenue time, mileage, and assignment numbers on each block.		
15.7.	Ability to modify trip times within the blocking module with the changes automatically reflected in blocking.		
15.8.	Ability to find the most efficient next trip on line(s) to be added to an existing block (including interlining potential) if manual blocking is selected.		
15.9.	Ability to block by service day type and vehicle groups.		
15.10.	Ability to view performance statistics for each block.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
15.11.	Ability to allow users to freeze certain blocks manually so that the automated blocking process will not change them.		
15.12.	Ability to optimize blocking without changing more than a specifiable amount of the sign-on and sign-off times of the assignments and allow for optimization without having to do a full runcutting and thus leaving no need to redo the assignments.		
15.13.	Ability to allow the user to set a limit on the number of vehicles allowed at once at a given time point (for example, there is a constraint on the number of buses that can be accommodated at any time at a transit center).		
15.14.	Ability to maintain seasonal or inactive routes.		
15.15.	Ability to maintain variations of service or day's exceptions by day of week (i.e. extra Friday night service).		
15.16.	Ability to maintain and assign alternate/calendar-based schedules (for example schedules for no school, holidays, early opening or late closing, etc.). Specify the maximum number of alternate schedules.		
16. Runcutting			
16.1.	Ability to define as soft rules the preferences for operator workday characteristics.		
16.2.	Ability to cut runs past midnight.		
16.3.	Capable of cutting single-piece or multi-piece work assignment runs, with each piece being a sequence of route trips from a particular block.		
16.4.	Ability to configure via parameters in compliance with RTS's rules/regulations and union contract provisions (hard and soft rules) including but not limited to: Min./max. work times, part-time operator work restrictions, spread times, spread premiums, report times, travel time, overtime, and guaranteed time. Provide a detailed description of automatic runcutting optimizing algorithms.		
16.5.	All rules, definitions, costs, optimizers, and parameters for runcutting are user defined and can be changed by the user directly from the user interface in the future without programming or proposer assistance.		
16.6.	Ability to define multiple operator workday types.		
16.7.	Maintain accurate summary statistics by time of day on total, revenue, and non-revenue (deadhead, layover, travel, recovery) mileage and time.		
16.8.	Software shall flag when a manual or automatically generated runcut violates any RTS rule/regulation, and list which rule/regulation is violated on each run.		
16.9.	Ability to define target ratios for the number of runs for specific run types, in order to provide a balanced solution (easier to roster).		
16.10.	Ability to reflect run classification, numbering, and other RTS specific parameters in the runcutting algorithm, as well as to define driver preferences as soft rules.		
16.11.	Ability to allow for either manual or automatic runcutting at user's discretion. Manual runcutting can be done at any time during a run cut by route, block, or run.		
16.12.	Ability to automatically create cost efficient and legal runcuts that are equal to or better than (based on non-revenue time share of total time) RTS's current runcutting solutions based on comparable service plan.		
16.13.	Ability to compare/evaluate the runcutting solutions (for same or different block solution) during the optimization process and compare two runcutting solutions on a single screen. Comparison output will provide information on pay time, vehicle requirements, etc.		
16.14.	Allow the shifting of trip relief times to optimize runcuts.		
16.15.	Ability to prohibit relief at a given location for part of the day, or for a specified direction.		
16.16.	Ability to allow for multiple operating yards in a single runcutting solution and assign a yard to a division.		
16.17.	Ability to enforce, warn, or ignore violations of work rules during construction of runs.		
16.18.	Ability to lock and unlock schedule by users (controlled by user level security).		
16.19.	Ability to accept user input to freeze certain runs manually and proceed to optimize on the remaining uncut work.		
16.20.	Ability to analyze cost with respect to union contract provisions, including, but not limited to, pay rates, work rules, management requirements, facility specific rules.		
16.21.	Ability to optimize scheduling parameters such as platform hours, minimum overtime, use of part time operators, etc. by cost or operators counts.		
16.22.	Ability to define different types of reliefs (on-street, pull-in to garage, car relief) and to prioritize relief points for trips at terminals and limit relief		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	points to designated places.		
16.23.	Allow user to identify certain time points as relief points between operator assignments (between routes and to and from relief points) and automatically build the travel time into the assignments.		
16.24.	The system shall have the capability to cut runs based on the inclusion of operator breaks at designated break and meal points.		
16.25.	Ability to automatically build travel time into assignments based on relief point utilization.		
16.26.	Ability to calculate travel time based on the vehicle schedule (i.e., drivers traveling from the depot to start place, etc.).		
16.27.	Ability to provide facility to uncut runs.		
16.28.	Ability to optimize a runcut solution on a replacement set of blocks that match, within a specified time range, the sign-on/off time of an existing specified runcut scenario.		
16.29.	The name of each runcut must be unique when saved, with a validation to prevent duplicate runcut identifiers.		
16.30.	Unlimited ability to save and retrieve previous run cuts.		
16.31.	Ability to maintain notes/comments at the following levels and to selectively address them to the drivers, the scheduler and/or the public: such as, but not limited to timepoints, specific passing times, or trips.		
16.32.	Ability to provide analysis/statistical tools that automatically summarize a runcut by cost and all cost components (including operators and hours [pay-to-platform]) required to operate the service.		
16.33.	Ability to conduct "what if" scenario analysis based on prospective rules and parameter changes that control the automatic runcutting algorithm. Modifications applied during scenario testing must not affect the integrity of existing files. Those saved files must be able to become the working files at a later date if the new work rules are adopted.		
16.34.	Ability to view percentages of runs by run type.		
16.35.	Ability to modify trips and blocks from within the runcut tools.		
16.36.	Ability to display runcutting and blocking side by side.		
16.37.	Ability to display runcutting solution graphically using horizontal time scale and allow modifications from graphical interface.		
16.38.	Ability to allow interruption to the runcutting processes at any time.		
16.39.	The system shall automatically calculate person and vehicle requirements for multiple user defined periods of the day.		
16.40.	Ability to support flexible sign-on/off matrix varying by place, time of the day, type of relief, vehicle group, type of run.		
16.41.	Ability to maintain a calendar of schedules in production.		
16.42.	Ability to create vehicle and operator schedule statistics for a specified interval of dates.		
16.43.	Any changes made at any level must propagate throughout the system with necessary integrity checks.		
16.44.	No duplication of data in the scheduling system, including the geocoding database.		
16.45.	Provide built-in tools for importing and analyzing historical runtime and on-time performance data, and adjusting runtimes for future schedule changes.		
16.46.	Ability to compare scheduling data, including all critical operational data, e.g. number of trips, blocks, runs, operational cost between different schedule sets, for a given day or range of days as well as maintain and provide reports on deviations including comparison of costs, number of trips added/cancelled, on specific days, or range of days, on route/division basis.		
16.47.	Ability to define non-driving operator assignments, and assignments that include a driving element and a non-driving element within a day.		
16.48.	Able to manually accept, reject, or edit automatic run cuts by route, vehicle type, block, or run.		
16.49.	Ability to create interfaces based on a calendar that records which vehicle and crew schedules are in effect for each date of a period.		
16.50.	The system shall not allow concurrent modifications to individual run cuts.		
16.51.	The system shall have the capability to optimize a run cut either by an entire location or by redoing a selected number of user defined runs.		
16.52.	Ability to easily query individual runs for all related information.		
16.53.	Ability to maintain run pieces without supporting block data such as spare work and garage work.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
16.54.	The system shall have the capability to manually add open pieces of work to previously cut runs.		
16.55.	The system shall be able to cut runs accommodating service and day of week variations within the schedule.		
16.56.	The system shall have the capability to cut runs for special events or days of operation and special service conditions.		
17. Rostering			
17.1.	Capable of both cafeteria-style and agency-developed rostering.		
17.2.	Ability to handle multiple rosters at same time (including extraboard), with allocation of operator workdays to a roster, based on preset criteria (for example: AM work or PM work; or workday type – straight, split; etc.).		
17.3.	Ability to override any pay component.		
17.4.	Ability to run multiple rostering scenarios.		
17.5.	Ability to define rules and parameters to meet company and government rules like minimum rest time, target rest time, and free periods.		
17.6.	Capable of building rosters automatically or manually.		
17.7.	Ability to swap roster day assignments between positions.		
17.8.	Ability to swap full week's assignments between positions.		
17.9.	Ability to build separate rosters for different groups of employees (i.e., part-time rosters).		
17.10.	Ability to detect whether assignments are below some RTS-specified number of hours of elapsed time between the end of one day's assignment and the beginning of the next day's assignment.		
17.11.	Ability to create rosters by division/yard, by combining weekday, Friday, Saturday, and Sunday runs and assigning weekly operator runs.		
17.12.	Ability to optimize work rosters based on similarity such as on-duty and off-duty time per user preference.		
17.13.	Ability to favor restricting roster position to a single operator workday, a single route, a single group of routes, etc.		
17.14.	Ability to allow automatic renumbering Saturday and Sunday runs to match weekday run numbers on all regular rosters created.		
17.15.	Ability to provide roster statistics including efficiency and cost (such as total weekly pay) of each roster and for all rosters by division/yard and by system.		
17.16.	Ability to sort weekly assignments by line on screen for visual analysis.		
17.17.	Ability to define rules to control days off, weekend off, working time customizable to factors such as working time and categories of run.		
17.18.	The system shall have the capability to automatically assign days off for rostered work based on work rules and regulations.		
17.19.	Ability to control runs via patterns to match requirement, such as having early runs before a day off and late runs after a day off.		
17.20.	Ability to handle four and five day working week.		
17.21.	Ability to display roster assignments by: • Workday type • Workday ID • Run ID • Run type • Varying start times within a single bid • Time between pieces of a run • Run start/end time • Route • Total weekly pay time • Full work days, etc. The display will allow for filtering based on include-only or exclude all bids that contain a specific characteristic like start time or route number.		
17.22.	Ability to define and take into account special service in effect for holidays and other occasions.		

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
17.23.	The system must be able to create extra work assignments that are not reflected in fixed route scheduling solution. These assignments would be used for Specials/Charters, Stand-by assignments, and other types of casual work.		
17.24.	The system must be able to create user-defined categories of extra work (i.e., meetings, marketing, training, etc.).		
17.25.	The system must allow definition of an extra work validity period (i.e., date range, days of week).		
17.26.	The system must be able to optionally associate customer's information with extra work created.		
17.27.	The system shall have the capability to validate the transitions between rosters for two consecutive bid periods.		
17.28.	The system must be able to support the creation of extra work in real-time or in advance.		
17.29.	Ability to optimize rosters based on cost parameters while in compliance with RTS's hard and soft rules, and government regulations.		
17.30.	The system shall have the capability to automatically generate one roster at a time or all rosters.		
18. Reporting			
18.1.	Ability to generate reports using standard database access tools with ease.		
18.2.	Ability to produce reports on platform, deadhead, and revenue miles and hours by user defined parameters such as trips, blocks, routes, time periods, service days, garage/operator, funding source, jurisdictions/community.		
18.3.	Ability to easily export reports into CSV, XLSX, DOCX, PDF, HTML and XML formats.		
18.4.	Ability to retrieve an exported file into the application for further editing.		
18.5.	The application shall include a set of standard reports and a configurable reporting tool. Reports available as standard (provide samples and details for each): • Headways[1] • Block paddles • Driver assignments • Detailed driver run sheet (including sign-on, sign-off, pull-in, and pull-out) • Summary and actual miles, hours, and running times for revenue and non-revenue service (route, block, trip, run, system-wide). This information shall be available to be displayed by route, day of week, and time of day with subtotals rolled into summaries for the system as whole. • Vehicle/car requirement and utilization summaries by division, assignment sheet, hour, manifest, weekly pay sheet (as user defined format) • Platform-to-Pay time • Trip length • Service start and end times • Dispatch pullout and pull-in sheets • Comparison of schedules (as user defined format) • Time between routes at stops and time points • Route frequency • Public timetables • Mileage and platform hours data by route and day • Bus stop list by route/pattern • Bus stop list for entire system • Bus stop list by user-defined subdivision • Timepoint list by route/pattern		

[1] This report shall present a list of buses (by block numbers) that service a particular route including time points along the route, the pull out/pull in times, and the departure times of each bus.

Item #	Scheduling System Functional Requirements	Meets Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	• User overrides of running times • Turn movements for an entire trip pattern, including any notes. • Run costing summary • User defined reports		
18.6.	Ability to generate yard capacity.		
18.7.	All the above reports shall be able to be aggregated by time, day, week, month, quarter, and year.		
18.8.	The daily reports shall provide statistics broken down on an hourly basis along with daily totals. The weekly reports shall provide statistics broken down on a daily basis along with weekly totals. The monthly report shall provide statistics broken down on a daily basis along with weekly and monthly totals and so forth.		
18.9.	Query features shall be available to filter reports based on time interval, hour, day, week, month, year and Year To Date (YTD) and there shall be the capability to compare specified data for given time intervals, dates, weeks, months, years or YTD.		
18.10.	Reports must provide statistical summary statistics, like minimum, maximum, mean, medium, count, and 1st/5th/25th/75th/95th/99th percentiles, as well as more advanced metric aggregates.		
18.11.	Drill down, drill through, or drill anywhere capabilities shall be available.		
18.12.	It shall be possible to create sub queries within reports for enhanced data analysis and interrogation.		
18.13.	Conditional formatting shall be permitted based on user defined criteria.		
18.14.	The reporting tool shall allow authorized users to create new (i.e., define their own report templates) and to edit/configure existing (i.e., modify the templates provided by the proposer) report formats and add custom text and messages.		
18.15.	The report generation tool shall allow users to choose 'a-la-carte' data elements and put them together in any combination necessary to build and customize reports to suit specific needs.		
18.16.	A Wizard shall be available to report writers to specify the type of report they want to create and then combine data elements to achieve the desired output.		
18.17.	User must be able to modify formatting, column aggregation, and sorting at the report creation level.		
18.18.	Data elements can be used as reported data, filters, or report sections.		
18.19.	The user must be able to schedule any report to be generated on a recurring basis (daily, weekly, monthly, yearly) and distributed to a designated list of email addresses.		
18.20.	Users must be able to toggle between tabular reports, charts, or combination reports easily.		
18.21.	Reports generated shall be fully modifiable and configurable by RTS staff without intervention or support by the proposer.		
18.22.	Support common report writing tools like Crystal Report.		
18.23.	Requests for reports shall be acknowledged within 10 seconds with an indication that the report is being processed.		
18.24.	After the deployment and implementation of the system, if need arises to create additional reports, under the maintenance agreement the selected proposer shall provide in addition to all of its standard reports up to 25 additional and customizable reports as requested by RTS.		
18.25.	Data for reports can be made available to report writers through data views created by database administrators. These data elements shall have additional metadata associated with them allowing users with no knowledge of the back end database to retrieve the data they need.		

Table 2 Mandatory Component Price Schedule

Item	Description	Estimated Quantity	Unit Cost	Hours and Hourly Rates	Total Cost	Reoccurring Cost (Y/N)?
Software – the cost of the software and the appropriate number of user licenses offered in the price must be stated by the proposer. It is the responsibility of the proposer to understand RTS operations in sufficient detail to determine the number of user licenses required to run the solution in the RTS environment.						
1	Database software					
2	TSS software license					
Services (supply & installation) – all costs associated with the full installation and implementation of the system (inclusive of cable and wiring and system design). Supplemental costs associated with user assessment, installation, database conversion, etc., must be detailed if separate and not included in the software price above.						
1	Desktop computers					
2	Servers					
Data Acquisition and Conversion Costs – if the proposer must acquire databases, street maps, or other items necessary to support installation, these costs shall be identified here. It shall also include existing schedule data conversion and import into new system.						
1	[insert item]					
Related Third Party Software Costs – all other software necessary to operate the TSS systems or to support maintenance of the system recommended by the proposer shall be identified. All such products shall be purchased by the proposer and licensed to RTS.						
1	Interface with RTS APC units (consumable schedule export)					
2	Interface with TIS (consumable schedule export)					
3	Interface with RTS's existing Operations software solution (provide and consume information).					
Training – if training costs are not included in the software purchase or licensing costs, proposals must identify all costs associated with all required training.						
1	Software functionality					
2	System administration					
Documentation – if documentation costs are separate and not included they must be identified below						
1	User					
2	System					
4	Configuration and troubleshooting					
Testing – if testing costs are not included in the software purchase or licensing costs, proposals must identify all costs associated with all required testing.						
1	[insert item]					

Maintenance and Support – Transfer information from *Table 3 Five Year Maintenance and Support Costs*						
1	TSS software					
2	TSS hardware					
Life Cycle Replacement Projection – 10-year budget outlook with replacement costs and lifecycle of products.						
2	[insert item]					
Other Costs – any other cost not identified above shall be identified and indicated by the proposer. This includes an itemized list of spare parts.						
1	[insert item]					
TSS Cost						

Table 3 Five Year Maintenance and Support Costs

Five Year Maintenance and Support – one year maintenance and technical support price shall be included (no charge [N/C]). Identify all ongoing costs related to maintenance and support.					
	Year 1	**Year 2**	**Year 3**	**Year 4**	**Year 5**
Hardware					
[insert item]	N/C				
Software					
[insert item]	N/C				
Software Maintenance Costs					
Hardware Maintenance Costs					
Total Maintenance Costs					

Table 4 Future Price Schedule

Item	Cost
[insert item]	

Table 5 Operations Software Functional Requirements

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
1. General			
1.1.	As applicable, software shall meet the same requirements for the TSS solution as documented in sections 1, 2, 3, 4, 5, 6, 7, 8, 10, and 11 of *Table 1 TSS Functional Requirements*.		
1.2.	The system will import all information stored in RTS's existing Operations software solution provided by Fleet-Net.		
2. Data Integration, Import, and Export			
2.1.	Interface to/from AVL system for real-time data exchange of work and vehicle assignments and to perform extra pay validation (i.e., late pull-in, logoff).		
2.2.	Interface to/from payroll/HR system/sign-in-terminal for all actions associated with hours worked, employee accrual balances, and pay codes along with employee demographic information.		
2.3.	Interface with the TSS solution.		
2.4.	The system shall contain and allow updating of employee records that include, but are not limited to: qualifications, seniority, hire date, license number and expiration date, birth date, address, email, and telephone number(s), emergency contact, etc.		
2.5.	Access to employee data must be controlled with system's built-in security features to allow confidential information to be accessed only by authorized users.		
2.6.	Ability to provide appropriate tools to export and import data that includes, but is not limited to: employee records, absences, vacations, etc.		
3. Regular Bid Process			
3.1.	The system must be able to accommodate current and future labor agreement work rules, and practices.		
3.2.	The system shall verify work rules and practices specified by the labor agreement are met throughout the bidding process.		
3.3.	The system shall enable RTS to define the entire bid process with all phases and the bid schedule for all operating personnel (i.e., bidding timetable identifying a specific bid time for each active employee participating in the bid).		
3.4.	The system shall allow bidding for rosters by garage/division location and/or system-wide specific.		
3.5.	The system shall allow for cafeteria style picking and pre-built roster position picking.		
3.6.	The system can handle different types of operators (5-day or 4-day rosters, part-timers, etc.) for either cafeteria or roster position picking.		
3.7.	Ability to make work force calculation/utilization report.		
3.8.	Validate driver choices to determine whether each driver meets minimum rest time requirements between runs.		
3.9.	Ability to allow only the system administrator to input system parameters for setting the number of full-time extraboard and part-time operators.		
3.10.	The system shall restrict operators to picking assignments based on eligibility criteria as per labor agreement and validate driver choices to determine whether each driver meets requirements of min./max. assigned work load and has been trained for the selected type of service.		
3.11.	The system must be able to show the pay details and work times of the work that can be bid.		
3.12.	The system shall allow operating personnel to choose work based on a selection of criteria. The criteria would then restrict the view of open		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	work viewed by the operating personnel. Operating personnel could deselect the criteria to view all work available to be picked.		
3.13.	The system shall assign the work picked by operating personnel at the time the work is selected and disallow that work from being available to any further picking.		
3.14.	The system shall allow the user to validate the work left to be picked, by day, by garage/division to the numbers of operators left to pick work assignments.		
3.15.	The system shall warn users at a point in the bidding process when work combinations (split runs) violate the fatigue rules under the provision of labor agreement or regulations.		
3.16.	The regular run selection process applies to run selection for holidays and special service days.		
3.17.	The system shall accommodate a separate set of schedules for run selection of holidays and special service days.		
3.18.	The system shall allow designated trippers to be picked for the duration of the bidding cycle as overtime assignments consistent with the operator's regular picked work and as per labor agreement.		
3.19.	The system shall be able to limit the number of trippers that can be picked by the operator per day.		
3.20.	The system shall provide a report of picked work to each operator.		
3.21.	The system shall support roll-back functions.		
3.22.	The system shall contain automation such that the regular roster position bidding and the vacation replacement bidding can be processed automatically, according to the labor agreement rules.		
3.23.	The system shall allow the operators to bid for their roster position (submit their pick choices) through the web, and see their picked work with notification.		
3.24.	The system shall allow the vacation replacement operators to bid for their work weeks (submit their pick choices) through the web, and see their picked work with notification.		
3.25.	The system shall allow seniority "bumps" to enable operating personnel to change their work assignment in the event that their picked work assignment has changed.		
3.26.	The system must be able to print a confirmation slip with the work and absences that were successfully bid.		
3.27.	The system must be able to support the modification of employee bid assignments to accommodate employees who cease employment with RTS or graduate from training and need a bid assignment until the next bid occurs.		
3.28.	The system must support the temporary holding-down of work when the bid employee is not available for a qualifying period.		
3.29.	The system must be able to recalculate the seniority ranking after importing new employee information or on demand.		
4. Vacation Bid			
4.1.	The system shall verify all work rules and practices specified by the labor agreement are met throughout the vacation bidding process.		
4.2.	The system shall allow specifying the number of vacation weeks or days offered for bidding, based on system parameters for the number of operating personnel picking per day throughout the vacation bid by location.		
4.3.	The system shall allow specific system parameters for both part-time and fulltime operating personnel.		
4.4.	The system shall make the following information available to the operating personnel: employee name, ID number, seniority date, vacation days available, days carried from previous year.		
4.5.	The system must display available quotas when the absence bid is entered and, if a quota has been exhausted, prevent the employee from		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	selecting the period.		
4.6.	The system shall interface from the HR system the vacation quotas and balances for each operating personnel.		
4.7.	The system shall allow defining the week of vacation (example: beginning on Saturday and ending the next Friday, less scheduled days off).		
4.8.	The system shall allow vacation to be selected purely by seniority (work or vacation order).		
4.9.	The system shall permit operating personnel who return to active service, at any time in the year, to select vacation if not already done so, according to what unselected weeks are left in the system.		
4.10.	The system shall provide a report of vacation assignments to each operator.		
4.11.	The system shall contain automation such that the vacation bidding can be processed automatically, according to the labor agreement rules.		
4.12.	The system shall allow the operators to bid through the web, and see their assigned vacations with notification.		
4.13.	The system must be able to support the configuration of a stand-alone absence bid or an absence bid that is conducted at the same time as a work bid.		
5. Mark-ups / Daily Schedule Preparation			
5.1.	The system shall provide the ability to generate and view daily schedules for every day specified within a desired day range (example: complete booking, next 2 weeks, etc.). This includes preparing for the next operational day which will consider all work that will be operated, assign the bid employee, open work due to absences and work rule violations, and prepare the day's extraboard and overtime volunteer lists.		
5.2.	The system shall provide the ability to maintain and apply different effective schedules on different specified dates.		
5.3.	Ability to manage daily operations not only for the planned regular service schedules, but also schedules for special events with minimum or no manual manipulation.		
5.4.	The system shall provide capability of updating daily schedules for specified dates.		
5.5.	The system shall provide a graphical /calendar representation of work days.		
5.6.	The system shall provide the capability to define absence types with corresponding absence codes.		
5.7.	The system shall provide tools for the system administrator to define absence quotas per type of absence. The quota is to be defined over flexible day range (example: per day, per week, per month, per year, per rolling 26 weeks, etc.). The quotas must be able to be defined for a range of dates allowing different quota values for each week/day.		
5.8.	The system shall provide tools to define different absence quotas for specific employees.		
5.9.	The system will warn the user if assigning an absence to operating personnel exceeds the corresponding quota.		
5.10.	The system must be able to accept absence requests and have them processed (granted/denied) by a dispatcher or a supervisor.		
5.11.	The system shall provide tools for the system administrator to define special temporary work patterns and apply them to specific employees over a specified date range (example: progressive return after long-term sick leave).		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
6. Extraboard/Open Work Assignment Process			
6.1.	The system shall verify all work rules and practices specified by the labor agreement are met throughout the process.		
6.2.	The system shall maintain paid and unpaid time off balances and update them through an interface with the HR system. And shall be used to verify whether time-off can be granted to an employee or not.		
6.3.	The system shall allow operators to create a personal profile noting preferred OT assignments, days off, known routes, and other variables.		
6.4.	The system shall control the number of open "slots" for day off requests using paid or unpaid absence codes by day of week.		
6.5.	The system administrator can control and modify the number of slots available for a single day.		
6.6.	The daily listing of open work shall be ordered as required by labor agreement and its order shall be modifiable by the system administrator.		
6.7.	The system shall create lists of extraboard operators ordered in a manner consistent with the labor agreement and work practice.		
6.8.	The system shall provide the ability to divide a run or tripper into various segments determined by the user and assign to the extraboard operators as regular time or overtime.		
6.9.	The system shall validate that any subdivided run or tripper is consistent with the labor agreement and work practice.		
6.10.	The system shall automatically assign open full runs to extraboard operators ordered in a manner consistent with the labor agreement and practice and with regard to the daily rotation of the extraboard.		
6.11.	The system shall automatically "Call" assignments with specific start times throughout the day as part of the daily extraboard assignments, in a manner consistent with the labor agreement and practice and with regard to the daily rotation of the Extraboard. This process shall also allow manual intervention.		
6.12.	The system shall allow extraboard operators to be scheduled on weekly hold-downs when replacing an operator for a full week's absence.		
6.13.	The system shall allow automation of the hold-down bidding process according to labor agreement rules.		
6.14.	The system shall allow users to move extraboard operators between the a.m. and p.m. extraboard as necessary.		
6.15.	The system shall allow changes to the operator bid to be made one week at a time or from a point in time to the end of the current bid.		
6.16.	The system shall determine the number of runs and trippers remaining after all available extraboard operators are assigned.		
6.17.	The system shall be possible for the remaining runs and trippers to be assigned in a manner that is consistent with the labor agreement and practice.		
6.18.	The system shall determine the number of remaining extraboard operators after all available runs and trippers are assigned.		
6.19.	The system shall allow the remaining extraboard operators to be assigned in a manner that is consistent with the labor agreement and practice.		
6.20.	The system shall allow proper absence codes to be used when un-assigning regular run or tripper operators from picked work.		
6.21.	The system shall allow extraboard operators to be unassigned from any work assigned during the mark up process without a formal absence code up to a specified time each day to be determined.		
6.22.	The system shall allow that any work formerly assigned would revert to the open work set.		
6.23.	The extraboard operator would revert to being available for work assignments.		
6.24.	The system must be able to optimize open and unassigned work into efficient work combinations in order to minimize overtime, unproductive time, and spread penalties in compliance with labor agreement.		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
6.25.	The system must allow for a centralized markup of the extraboard assignments.		
6.26.	The system shall provide an ongoing summary in real-time of all unassigned work to be covered.		
6.27.	The system shall provide the user with total daily hours scheduled, the hours at overtime, the hours at regular time, any hours cancelled, and any hours left to be assigned.		
6.28.	The system shall allow users to "drill down" on any indicator to see the specifics of work left to be assigned and work that is canceled.		
6.29.	The system shall provide summary statistics by location and/or at the system-level.		
6.30.	The system shall allow the option to disallow operators to work overtime in excess of the maximum number of hours prescribed, less time allowed for recovery on any run or tripper for any given day.		
6.31.	The maximum number of hours prescribed shall be controlled by the system administrator.		
6.32.	The system shall allow the assignment of overtime and be consistent with the labor agreement and practice with regard to the rotation of overtime.		
6.33.	The system shall generate lists of operators on their day off and include, when possible, those operators' overtime preferences according to the operator's profile on record.		
6.34.	The system shall match operators that have not signed up for overtime, but are available for overtime according to their picked work schedule with work that is unfilled and available for overtime.		
6.35.	The system shall provide the ability to assign work to operators on modified run or other special situations and track their assignments.		
6.36.	The system shall allow scheduled work to be modified or canceled with a reason code and explanation. Cancelled work must be shared with trip planning software (if implemented) so no itineraries are generated using cancelled trip information.		
6.37.	The system shall allow an employee to be transferred from one division to another.		
6.38.	The system shall manage work exchange requests between operators.		
6.39.	The system shall allow to automatically generate a file that can be used by an Interactive Voice Response (IVR) for telephonic query concerning an individual's next day work assignment		
7. Dispatch and Pay			
7.1.	The system shall display all work runs with their assigned operator for the day. The work can be viewed in a calendar and table view.		
7.2.	The system shall display unassigned work assignments.		
7.3.	The system shall manage weekly and daily changes to planned rosters.		
7.4.	The system shall display absent drivers.		
7.5.	The system will support the modification of work times.		
7.6.	The system shall display all extraboard and standby operators for the day, with their status (off, vacations, etc.).		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
7.7.	The system shall allow the clerk to view who must sign-on in the next time interval (time interval to be configurable) with their expected report time.		
7.8.	The system shall interface with an existing sign-on device so the system gets the information of who signs on (and when) as it happens.		
7.9.	The system shall show in real-time the work that has been completed and the work that still needs to be completed.		
7.10.	The system shall allow the clerk to sign-on/off an operator directly.		
7.11.	The system will provide a printed receipt confirming a successful sign-on/sign-off operation.		
7.12.	The system will support the generation of employee messages; some of which may restrict the automated sign-on/sign-off processes.		
7.13.	The system shall provide a sign-on web panel to be available on RTS' computers or devices in each garage /division.		
7.14.	The system will provide the ability to perform sign-on/sign-off functions by the employee who will identify themselves to the system using an electronic ID card or employee number and PIN (personal identification number).		
7.15.	The system must notify the clerk if an operator is late for sign-on through visual and audible cues.		
7.16.	The system shall support the assignment and un-assignment of work and allow the clerk to reassign work to another operator (regular, extraboard, standby) if the originally assigned operator is late for sign-on or is unable to complete their assigned run. Employee availability, qualifications, and work assignment rules will be considered during these processes. These assignment/un-assignment activities must be logged in the database.		
7.17.	The system shall alter any work if any labor agreement rule is not respected when reassigning work runs.		
7.18.	If an operator is unavailable for work, the system must allow the dispatcher to record a reason (sick, injury, etc.) and an expected return date.		
7.19.	The system must allow the clerk to create ad-hoc work such as (but not limited to): trippers, shuttle, utility, relay, coach move, or exchange.		
7.20.	The system must be able to allow service adjustments and schedule changes for individual time points and stops due to detours, etc.		
7.21.	The system shall provide a monitoring function displaying all active work runs with their start and end times and assigned operator.		
7.22.	The system shall provide a sign-out web panel to be available on RTS' computers or devices in each garage/division.		
7.23.	The system shall provide the ability for the operators signing out to log delays/overtime.		
7.24.	The system will provide detour notification for the dispatcher and the employee.		
7.25.	The system will be able to apply filters to the daily work to show only certain types of work, show open work only, or show assignment exceptions only.		
7.26.	The system must be able to receive driver and vehicle modifications resulting from plug buses and other assignment changes from the AVL system.		
7.27.	The system must be able to receive real-time updates of driver and vehicle incidents from the AVL system.		
8. Daily/Weekly Timekeeping			
8.1.	The system shall verify all work rules and practices specified in the labor agreement are met throughout the timekeeping process.		
8.2.	They system shall provide for each user up-to-date information on hours worked, vacation, sick leave, etc.		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
8.3.	Track work performed other than scheduled assignments (i.e. relief supervisor/dispatcher).		
8.4.	The system must be able to consider the employee activities (work assignments, absences, additional pay) and apply the necessary rules to generate timekeeping transactions that reflect RTS documented timekeeping policies.		
8.5.	The system must be able to generate hours, pay rate, and dollars for each timekeeping transaction.		
8.6.	The system must be able to track paid hours (actual work times) by operator in a manner consistent with the labor agreement and practice for both part-time and full-time operating personnel, including but not limited to platform time at straight pay, platform time at overtime pay, report time, spread time, allowed time (make up time to daily/weekly guarantee), and excess time (exceeding run pay for late pull-ins, traffic, etc.). Absence code pay hours must be paid in compliance with the labor agreement and practice, including but not limited to sick leave pay, training pay, other special pay or non-platform pay hours, and vacation leave pay (single day or weekly). Extraboard operator pay hours include call time hours or other non-platform assignment hours and run and tripper platform time segments.		
8.7.	The system shall be able to report all platform time scheduled and worked.		
8.8.	The system must be able to perform labor account distributions for each timekeeping transaction.		
8.9.	The system must be able to track and report unpaid time by operator in a manner consistent with the labor agreement and practice for both part-time and full-time operating personnel.		
8.10.	The system must be able to track and report when an assigned run is reassigned to another operator: time worked by replacement operator, time when regular operator resumed run, and reason for reassignment.		
8.11.	The system must be able to track and code extra service hours by operating personnel, created inside or outside the scheduling system for special events, charters, etc.		
8.12.	The system must be able to accumulate actual pay hours, by operator, by day for those operators with a daily guarantee and by week for those operators with a weekly guarantee.		
8.13.	The system must automatically process actual pay hours and transmit actual pay hours to the payroll system for gross to net payroll processing.		
8.14.	The system must incorporate RTS's existing accounting structures with applicable pay rules and pay codes.		
8.15.	The system must be able to post and export timekeeping transactions in accordance with RTS payroll system's import specification.		
8.16.	The system shall allow a day to be "closed" according to the applicable procedures.		
8.17.	The system must be able to provide an audit trail of each timekeeping transaction that has been modified or added.		
8.18.	The system must have the ability to generate and print payroll audit reports.		
9. Reporting			
9.1.	Ability to generate discrepancy report that shows scheduled run data versus actual run data on a daily basis.		
9.2.	Define, track, and display employee statistics including hours worked, overtime, days off, etc.		
9.3.	Ability to generate overtime discrepancy comparison report that shows the picked work time and the actual work time on a daily basis.		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
9.4.	Ability to generate incident/accident reports that also contain the type of incident/accident.		
9.5.	The system must be able to report to management in real time each day's work product. The work product consists of the distribution and cost of open or unassigned runs and trippers. The amount of daily overtime, call-time, and other unproductive time.		
9.6.	The system must provide inquiry tools that display operational characteristics/statistics including: absences, accidents, incidents, extraboard utilization, overtime utilization, and missed service.		
9.7.	The system shall provide operator utilization statistics and reports by user-defined parameters as well as ad hoc utilization reports.		
9.8.	The system shall have the ability to report part-time operator productivity (can be based on a maximum number of weekly assignments) for daily, weekly, monthly or other periods as requested.		
9.9.	Data used to report utilization statistics and reports shall be able to be downloaded in the form of an XML file, comma delimited, Crystal Report, spreadsheet, or in tab delimited formats.		
9.10.	The system must produce system statistics at the following levels: service type, garage and system, vehicle block, run and trips within runs, and tripper and trips with trippers.		
9.11.	The system must be able to report the discrepancy between schedule and actual vehicle mileage and hours based on division, line or block level.		
9.12.	The system must be able to interface with vehicle maintenance/AVL applications for tracking actual vehicle miles by vehicle.		
10. Operator Service Web Kiosk			
10.1.	The system shall provide kiosk module for the operators to have access to their day assignment, manifest, driver paddle.		
10.2.	The kiosk shall be a web application, to be run on RTS's computers at garage /division dispatch rooms.		
10.3.	Employees using the web portal must validate themselves using an electronic ID card or by entering their employee number and PIN.		
10.4.	The access to the web module must be secured with username and password.		
10.5.	It shall be possible for each operator to change his/her password.		
10.6.	The system shall allow the operators to request an absence or a vacation for desired date(s).		
10.7.	The system shall display the number of daily available day off quotas and daily vacation quotas.		
10.8.	The system shall allow the operators to specify their overtime availability, specified as the range(s) of dates and day(s) of the week when available for overtime.		
10.9.	The system shall allow the operators to specify general work preferences per day of week for desired type of work (example: earliest start or finish, latest start or finish, shortest, highest pay). The system must prevent employees from changing their choices after their designated bidding deadline.		
10.10.	The system shall allow the operators to view the status of all work (pieces) for a given day (unassigned, late, etc.)		
10.11.	The system shall allow the operators to specify desired specific choices for daily open work pieces for a given day.		
10.12.	The system shall allow the operators to enter delays for operated work.		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
10.13.	The system shall be able to capture a vehicle service request.		
11. Daily Operation Metrics			
11.1.	The system shall provide a web page displaying live operational metrics.		
11.2.	This display shall be configurable so the user can select which metrics to view.		
12. Vehicle Assignment			
12.1.	The system shall provide the ability to import the vehicle availability.		
12.2.	The system shall provide a display of vehicle locations within the division.		
12.3.	The system must be able to receive real-time updates for vehicle availability.		
12.4.	The system must support the definition of a parking map by day of week and time of day.		
12.5.	The system must support the definition of pull-out sequences and parking restrictions within the parking map.		
12.6.	The system shall display the vehicle already assigned to the work runs and the available unassigned vehicles, by types (wheel chair lift, articulated bus, bike rack, etc.), as well as vehicles that are not available due to maintenance activities.		
12.7.	The system shall allow a garage/division personnel to assign a vehicle to a work run, or to change the vehicle if already assigned while taking into consideration the required vehicle characteristics and availability.		
12.8.	If there is an accident/mechanical issue, the system shall allow the supervisor to introduce a vehicle change and, if necessary, manage related replacement trips accordingly (e.g., a standby operator drives the new vehicle to the accident location and brings the damaged one back to the depot).		
12.9.	The system must provide tools that can identity a shortage of available vehicles.		
12.10.	The system must be able to natively provide daily and scheduled vehicle requirements to the fleet management solution.		
12.11.	The system must provide tools to split/un-split blocks and perform vehicle change-outs.		
13. Employee Performance Management			
13.1.	The system shall provide a function to monitor employee performance based on RTS's rules and practices.		
13.2.	The system must be able to allow viewing of an employee's timekeeping transactions, work assignments, messages, allowance history, and absences.		
13.3.	The system must be able to show a consolidated view of an employee's employment history including: vehicle assignment history, employee certifications, and qualifications and equipment issues.		
13.4.	The system shall allow defining of performance rules and indicators in the system. Such rules can be (but not limited to): · Warning after a defined number of absences of a specific type (example: 5 late absences) within a user-specified period (year, days, rolling days, etc.), discipline meeting after a number of warnings (example: 2), award certificate if defined period of time without incident, extra vacation day(s) after a number of days without incident.		
13.5.	The system must be able to support point-based, hours-based, and occurrence-based step discipline. This feature must incorporate step		

Item #	Operations System Functional Requirements	Meet Requirement "Out of the Box"? (Yes/No)	Comments/ Explanation of Functionality
	discipline rules and recommend the appropriate administrative action.		
13.6.	The system shall keep track of the operators' performance according to the defined indicators, and provide messages when performance conditions are triggered.		
13.7.	The system must provide tools to support safe driving programs.		
13.8.	The system must provide tools to support perfect/good attendance incentive programs.		
13.9.	The system must provide tools that support the tracking of absences based on days, occurrences, and patterns.		
13.10.	The system must provide tools that support the tracking of late sign ins and sign outs based on days, occurrences, and patterns.		
13.11.	The system shall allow RTS to generate and print employee performance reports.		
13.12.	The system shall allow the attachment of files, such as scanned documents, to employee's performance record.		
13.13.	The system shall have a method to incorporate historical performance data into employee's record.		