

System Design and Architecture for the IDTO Prototype

Phase II Demonstration Site (Central Florida)

www.its.dot.gov/index.htm

Draft Report — May 1, 2015

FHWA-JPO-16-274



U.S. Department of Transportation

Produced by Battelle Memorial Institute
U.S. Department of Transportation
Research and Innovative Technology Administration
Federal Transit Administration

Cover Photo: The Ohio State University

Notice

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

The U.S. Government is not endorsing any manufacturers, products, or services cited herein and any trade name that may appear in the work has been included only because it is essential to the contents of the work.

Technical Report Documentation Page

1. Report No. FHWA-JPO-16-274		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle System Design and Architecture for the IDTO Prototype – Phase II Demonstration Site (Central Florida)				5. Report Date May 1, 2015	
				6. Performing Organization Code	
7. Author(s) David Herman, Andrew Brohard, Matthew Burns, Gregory Zink, Thomas Timcho				8. Performing Organization Report No. 100026848-211A	
9. Performing Organization Name And Address Battelle 505 King Avenue Columbus, Ohio 43201				10. Work Unit No. (TRAVIS)	
				11. Contract or Grant No. DTFH61-12-D-00040 / O&ITS-13-02	
12. Sponsoring Agency Name and Address United States Department of Transportation ITS Joint Program Office Research And Innovative Technology Administration (RITA) 1200 New Jersey Avenue, SE Washington, DC 20590				13. Type of Report and Period Covered Final Report	
				14. Sponsoring Agency Code HOIT-1	
15. Supplementary Notes Mr. Ron Boenau, COTM					
16. Abstract This report documents the System Design and Architecture for the Phase II implementation of the Integrated Dynamic Transit Operations (IDTO) Prototype bundle within the Dynamic Mobility Applications (DMA) portion of the Connected Vehicle Program. This version updates the previous submission to document the revised design elements that were implemented in the demonstration of IDTO using providers in the central Florida / Orlando region. It also includes design elements for features that were designed but not fully implemented due to changes in the deployment schedule of participating partners. Finally, this report includes design elements for a D-RIDE rideshare partner that was ultimately demonstrated separate from the specific Ph. II demonstration, but integrated with the complete IDTO system.					
17. Key Words Integrated Dynamic Transit Operations, IDTO, T-CONNECT, T-DISP, D-RIDE, Dynamic Mobility Applications, DMA			18. Distribution Statement Final document. Limited Distribution. Has not been checked for Section 508 compliance.		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 107	22. Price

Table of Contents

Table of Contents	i
Chapter 1 Overview	1
Chapter 2 System of Interest	1
2.1 DISPATCHER DASHBOARD	2
2.2 WEB BROWSER	3
2.3 MOBILE DEVICE	3
2.4 MOBILE DATA TERMINAL	3
2.5 T-CONNECT OPPORTUNITY MANAGER.....	3
2.6 DATA REPOSITORY	4
2.7 TRAVELER SERVICES	4
2.8 VEHICLE LOCATION SERVICE.....	5
2.9 ROUTE AGGREGATION SERVICE	5
2.10 GEOLOCATION SERVICE	5
2.11 RIDESHARE SERVICE	5
2.12 DEMAND / RESPONSE OPERATIONS.....	5
2.13 VLS PROVIDER.....	6
2.14 ROUTE PROVIDER.....	6
2.15 FIXED ROUTE OPERATIONS	6
Chapter 3 Phase II Demonstration Site.....	7
3.1 LYNX	7
3.2 SUNRAIL.....	7
3.3 UNIVERSITY OF CENTRAL FLORIDA SHUTTLE SERVICE	7
3.4 FLEXBUS	8
3.5 ZIMRIDE	8
Chapter 4 Use Case Model.....	9
4.1 TRAVELER EXPERIENCE	9
4.1.1 Register	11
4.1.2 Login.....	15
4.1.3 Edit Profile	18
4.1.4 Plan Trip	21
4.1.5 Cancel Trip	28
4.1.6 Take a Trip.....	31
4.1.7 Logout	33
4.1.8 Deactivate Account	35
4.2 FIXED ROUTE DISPATCHER EXPERIENCE	36
4.2.1 Process T-Connect Request.....	37
4.3 SYSTEM BEHAVIOR	39
4.3.1 Monitor Active Trips.....	40
4.3.2 Notify Traveler	42

Chapter 5 Interfaces	43
5.1 TRAVELER.....	43
5.2 PRIVATE INCOMING SERVICE PROVIDER.....	44
5.3 FIXED ROUTE / T-CONNECT PROVIDER.....	46
5.4 RIDESHARE PROVIDER.....	47
5.5 DEMAND / RESPONSE PROVIDER.....	47
5.6 INCOMING FIXED ROUTE PROVIDER.....	48
5.7 ROUTE PROVIDER.....	48
Chapter 6 IDTO Data Model and Schema	50
6.1 _MIGRATIONHISTORY.....	51
6.2 BLOCKS.....	52
6.3 DISPATCHERLOGINS.....	53
6.4 DISPATCHERS.....	54
6.5 LASTVEHICLEPOSITIONS.....	55
6.6 LOGINS.....	56
6.7 MODES.....	57
6.8 PROVIDERS.....	58
6.9 PROVIDERTYPES.....	59
6.10 REGISTRATION.....	60
6.11 STEPS.....	61
6.12 TCONNECTEDVEHICLES.....	63
6.13 TCONNECTOPPORTUNITIES.....	64
6.14 TCONNECTREQUESTS.....	66
6.15 DISPATCHERLOGINS.....	67
6.16 TCONNECTSTATUS.....	68
6.17 TODOITEM.....	69
6.18 TRAVELERLOCATIONS.....	70
6.19 TRAVELERS.....	71
6.20 TRIPLEVENTS.....	73
6.21 TRIPS.....	74
6.22 WEBAPIGETTRIPUSAGES.....	76
6.23 ZIMRIDETRIPS.....	77
APPENDIX A. List of Abbreviations and Acronyms.....	A-1
APPENDIX B. Interface Messages.....	B-1
ACCOUNTINFORMATION.....	B-9
ACCOUNTLOGININFORMATION.....	B-9
ACCOUNTREGISTRATIONINFORMATION.....	B-9
ADDRESSCOMPONENTS.....	B-9
ADDRESSQUERY.....	B-9
ADDRESSQUERYRESULT.....	B-9
ADDRESSQUERYRESULTS.....	B-10
CURRENTTCONNECTIONREQUESTS.....	B-10
ETA.....	B-10
EXPECTEDLOCATIONREQUEST.....	B-10
FROMLOCATION.....	B-10
GEOMETRY.....	B-10
LEG.....	B-10
LOCATION.....	B-10

MDTRROUTE	B-11
MDTRROUTEType	B-11
MDTROUTESFORBUSSTOP	B-11
MDTROUTESFORBUSSTOPREQUEST	B-11
MDTSTOPID.....	B-11
OPTREQISO.....	B-11
OPTREQPLAN.....	B-11
OPTREQPLANCOMMON.....	B-11
OPTREQTILEZXYPNG.....	B-11
OPTREQWMS.....	B-12
POINTTIME.....	B-12
SEARCHCRITERIA	B-12
STEP	B-12
STOPID	B-12
TCONNECTIONREQUEST.....	B-12
TCONNECTIONREQUESTRESPONSE.....	B-13
TCONNECTIONSTATUS	B-13
TOLOCATION.....	B-13
TRIP	B-13
TRIPCANCELLATION	B-13
TRIPCONFIRMATION	B-14
TRIPNOTIFICATION	B-14
TRIPS	B-14
VIEWPORT.....	B-14
ZIMRIDE	B-14
TCONNECTSTATUSENUM.....	B-14
APPENDIX C. ZimRide Search API.....	C-1

List of Tables

Table 2-1. Columbus Demonstration Site Partner Roles	1
Table 6-1. _MigrationHistory Relational Database Table Columns	51
Table 6-2. _MigrationHistory Table Column Constraints.....	51
Table 6-3. _MigrationHistory Table Relationships	51
Table 6-4. Blocks Relational Database Table Columns	52
Table 6-5. Blocks Table Column Constraints	52
Table 6-6. Blocks Table Relationships.....	52
Table 6-7. DispatcherLogins Relational Database Table Columns	53
Table 6-8. DispatcherLogins Table Column Constraints.....	53
Table 6-9. DispatcherLogins Table Relationships	53
Table 6-10. Dispatchers Relational Database Table Columns	54
Table 6-11. Dispatchers Table Column Constraints	54
Table 6-12. Dispatchers Table Relationships	54
Table 6-13. LastVehiclePositions Relational Database Table Columns	55
Table 6-14. LastVehiclePositions Table Column Constraints.....	55
Table 6-15. LastVehiclePositions Table Relationships.....	55
Table 6-16. Logins Relational Database Table Columns	56
Table 6-17. Logins Table Column Constraints	56
Table 6-18. Logins Table Relationships.....	56
Table 6-19. Modes Relational Database Table Columns.....	57
Table 6-20. Modes Table Column Constraints	57
Table 6-21. Modes Table Relationships	57
Table 6-22. Providers Relational Database Table Columns	58
Table 6-23. Providers Table Column Constraints.....	58
Table 6-24. Providers Table Relationships.....	58
Table 6-25. ProviderTypes Relational Database Table Columns	59
Table 6-26. ProviderTypes Table Column Constraints.....	59
Table 6-27. ProviderTypes Table Relationships	59
Table 6-28. Registration Relational Database Table Columns	60
Table 6-29. Registration Table Column Constraints.....	60
Table 6-30. Registration Table Relationships	60
Table 6-31. Steps Relational Database Table Columns	61
Table 6-32. Steps Table Column Constraints.....	62
Table 6-33. Steps Table Relationships	62
Table 6-34. TConnectedVehicles Relational Database Table Columns	63
Table 6-35. TConnectedVehicles Table Column Constraints	63
Table 6-36. TConnectedVehicles Table Relationships.....	64
Table 6-37. TConnectOpportunities Relational Database Table Columns	64
Table 6-38. TConnectOpportunities Table Column Constraints	65
Table 6-39. TConnectOpportunities Table Relationships.....	65
Table 6-40. TConnectRequests Relational Database Table Columns	66
Table 6-41. TConnectRequests Table Column Constraints.....	66
Table 6-42. TConnectRequests Table Relationships.....	67
Table 6-43. TConnects Relational Database Table Columns	67

Table 6-44. TConnects Table Column Constraints	68
Table 6-45. TConnects Table Relationships.....	68
Table 6-46. TConnectStatus Relational Database Table Columns	68
Table 6-47. TConnectStatus Table Column Constraints.....	69
Table 6-48. TConnectStatus Table Relationships	69
Table 6-49. TodoItem Relational Database Table Columns.....	69
Table 6-50. TodoItem Table Column Constraints	70
Table 6-51. TodoItem Table Relationships	70
Table 6-52. TravelerLocations Relational Database Table Columns.....	70
Table 6-53. TravelerLocations Table Column Constraints	70
Table 6-54. TravelerLocations Table Relationships	71
Table 6-55. Travelers Relational Database Table Columns.....	71
Table 6-56. Travelers Table Column Constraints	72
Table 6-57. Travelers Table Relationships	73
Table 6-58. TripEvents Relational Database Table Columns	73
Table 6-59. TripEvents Table Column Constraints.....	73
Table 6-60. TripEvents Table Relationships	74
Table 6-61. Trips Relational Database Table Columns.....	74
Table 6-62. Trips Table Column Constraints	75
Table 6-63. Trips Table Relationships.....	76
Table 6-64. WebApiGetTripUsages Relational Database Table Columns	76
Table 6-65. WebApiGetTripUsages Table Column Constraints.....	77
Table 6-66. WebApiGetTripUsages Table Relationships.....	77
Table 6-67. ZimRideTrip Relational Database Table Columns	77
Table 6-68. ZimRideTrip Table Column Constraints	78
Table 6-69. ZimRideTrip Table Relationships.....	78

List of Figures

Figure 2-1. The IDTO Prototype.....	2
Figure 4-1. The Traveler's Experience	10
Figure 4-2. Register_ActivityGraph	11
Figure 4-3. The Registration Page	12
Figure 4-4. The Home Page.....	13
Figure 4-5. Login_ActivityGraph.....	15
Figure 4-6. The Login Page.....	16
Figure 4-7. Edit Profile_ActivityGraph	18
Figure 4-8. The Account Page.....	19
Figure 4-9. Schedule trip_ActivityGraph	21
Figure 4-10. The Trip Planning Page	22
Figure 4-11. The Results Page.....	23
Figure 4-12. The Trip Detail Page	24
Figure 4-13. The Step Detail Page.....	25
Figure 4-14. View Past Trips	26
Figure 4-15. Cancel Trip Activity Graph	28
Figure 4-16. The Scheduled Trip Detail Page.....	29

Figure 4-17. Take a Trip_ActivityGraph.....	31
Figure 4-18. Logout_ActivityGraph	33
Figure 4-19. Deactivate Account_ActivityGraph	35
Figure 4-20. Fixed Route Dispatcher Experience.....	36
Figure 4-21. Process T-Connect Request_ActivityGraph.....	37
Figure 4-22. System Behavior.....	39
Figure 4-23. Monitor Active Trips_ActivityGraph.....	40
Figure 4-24. Notify Traveler_ActivityGraph	42
Figure 5-1. I-02 Mobile Device Interface	44
Figure 5-2. I-03 Mobile Data Terminal Interface.....	45
Figure 5-3. I-08 Vehicle Location Interface	45
Figure 5-4. I-04 Dispatcher Dashboard Interface	46
Figure 5-5. I-08 Vehicle Location Interface	46
Figure 5-6. I-05 Rideshare Service Interface.....	47
Figure 5-7. I-06 Demand / Response Operations Interface	47
Figure 5-8. I-08 Vehicle Location Interface	48
Figure 5-9. I-07 Route Provider Interface.....	48
Figure 5-10. I-09 Geolocation Service Interface	49
Figure 6-1. IDTO Prototype Schema	50
Figure B-1. Messages M-01, M-02, M-03	B-1
Figure B-2. Messages M-04, M-05	B-2
Figure B-3. Message M-06.....	B-2
Figure B-4. Message M-07.....	B-3
Figure B-5. Messages M-08, M-09, M-25	B-3
Figure B-6. Messages M-10, M-11, M-12, M-13, M-14.....	B-4
Figure B-7. Message M-15.....	B-4
Figure B-8. Messages M-16, M-17, M-18, M-19.....	B-5
Figure B-9. Message M-20.....	B-6
Figure B-10. Message M-21.....	B-7
Figure B-11. Message M-22.....	B-8
Figure B-12. Message M-23.....	B-8
Figure B-13. Message M-24.....	B-8

Chapter 1 Overview

This document builds on the previously submitted software requirements and architecture for the Phase I Prototype demonstration, and expands the design and architecture that implements the functions included for the Integrated Dynamic Transit Operations (IDTO) Prototype and each partner; the interfaces between the traveler, the various partner systems and evaluation teams, and the interfaces and components internal to the core IDTO Prototype component (aka the Travel Management Coordination Center). The document is organized as follows:

Section 2: System of Interest defines the relationship between the various components that make up the IDTO Prototype and introduces the reader to each of these functional components.

Section 3: Use Case Models articulates the system use cases between the various components of the IDTO Prototype.

Section 4: Interfaces details each of the interfaces between the various services providers and the IDTO Prototype which are encompassed in the Phase I demonstration.

Section 5: Data Model and Schema provides the details of the underlying entity-relation model, including the data dictionary, primary/foreign key relationship, and other constraints.

The attached appendix lists the structure and content of all the expected messages of the interfaces described in Section 4.

Chapter 2 System of Interest

The IDTO Phase II demonstration will be conducted in central Florida. The Battelle Team that will conduct this demonstration include: LYNX, also known as the Central Florida Regional Transportation Authority, which provides public transportation services for the central Florida area; the University of Central Florida (UCF), which operates the Black and Gold shuttle service; and the Florida Department of Transportation (FDOT), which operates SunRail, a commuter rail system in central Florida. Zimride, a nationally recognized service provider for rideshare capabilities, providing dynamic rideshare service in the Orlando area, as well as in the Columbus OH area, is also included herein

The IDTO Prototype encompasses the Travel Management Coordination Center (TMCC), as described herein, as well as the mobile applications and data collection associated with the various evaluation efforts associated with this effort. LYNX's participation includes support of the T-CONNECT application by providing schedule and route information. The University of Central Florida (UCF), with its available AVL systems, will provide incoming connections to allow for protected transfers with the connecting LYNX service. SunRail, a new commuter rail service in the region, and only recently operational, will also support the demonstration of T-CONNECT in Orlando. FlexBus provides the demand/response component of T-DISP capability, however, it is likely that the procurement schedule for this service, including the software components, will not support the IDTO demonstration schedule. As such, the design for the dynamic demand/response will be maintained in this document, with a full realization of its implementation unlikely, Finally, Zimride will provide the D-RIDE capability. Table 2-1 summarizes these partners, their role and the associated applications.

Table 2-1. Columbus Demonstration Site Partner Roles

Partner	Service	Role	Application Area		
			T-CONNECT	T-DISP	D-RIDE
LYNX	Fixed Route / Fixed Schedule	T-CONNECT Provider	X		
UCF Shuttle Service	Fixed Route / Fixed Schedule	T-CONNECT Feeder	X		
Flexbus	Demand / Response	T-DISP Provider		X	
SunRail	Fixed Route / Fixed Schedule	T-CONNECT Feeder ¹	X		
Zimride	Rideshare Enabler	Rideshare Provider			X
Battelle	IDTO Prototype	System Integrator	X	X	X

Source: Battelle

¹ Sun Rail will require additional AVL capabilities to become a T-Connect Feeder

The relationship between each of the entities (partners) that will be integrated in the Phase II demonstration is shown in Figure 2-1. In the center is located the 'IDTO Prototype', or as also identified, the TMCC. The IDTO Prototype serves as the conduit through which most of the traveler interactions take place. As with the Phase I implementation, not all interactions required for a traveler's trip are mediated by the IDTO Prototype. For example, although the IDTO Traveler service will offer options for certain Demand/Response Providers, the responsibility to schedule that portion of the trip is outside the scope of the Phase II implementation and is instead handled by the traveler through other, existing applications/portals for those services. Notice too that some of the interface arrows are bi-directional, while others are unidirectional. This directionality represents the type of dialog the underlying systems of the two respective parties are expected to maintain.

The remainder of this section describes the high-level description of the roles and responsibilities of each of the components in the IDTO Prototype.

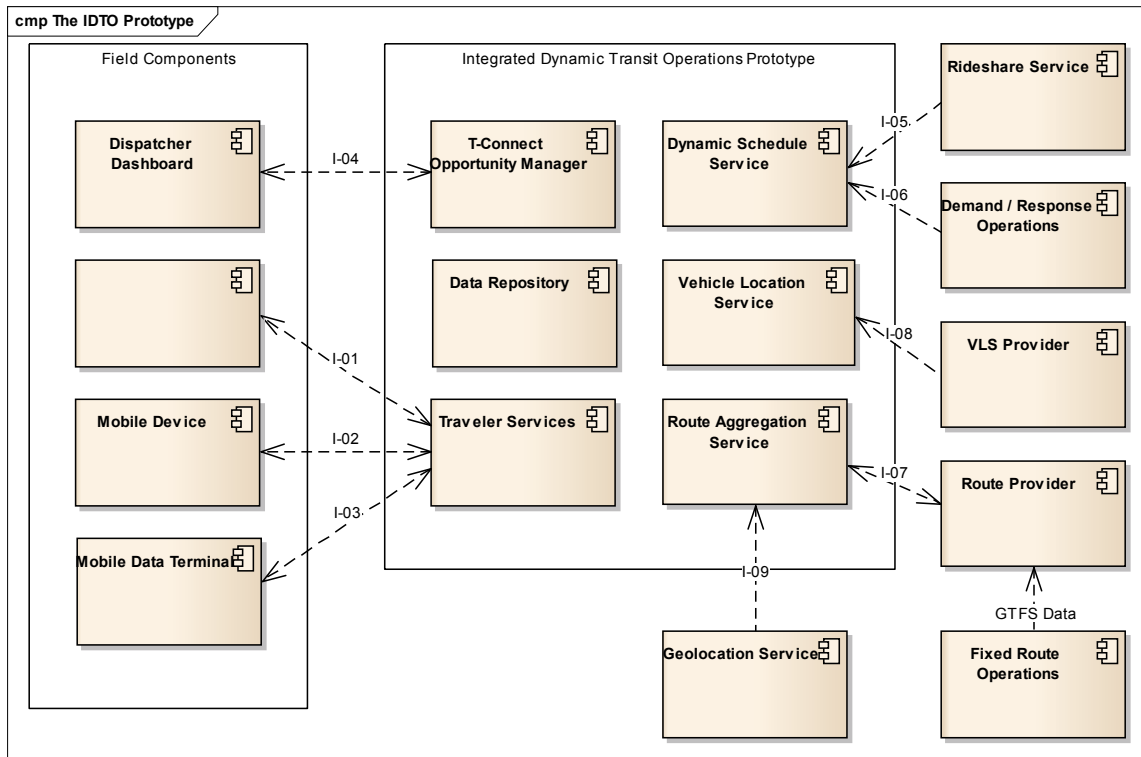


Figure 2-1. The IDTO Prototype

2.1 Dispatcher Dashboard

The Dispatcher Dashboard is the communication link between the IDTO Prototype and the Fixed Route / T-Connect Provider's operations manager. The dashboard displays T-Connect requests to the Provider's operations manager and provides the means for the operations manager to confirm or reject requests and relay that information back to the T-Connect Opportunity Manager within the IDTO Prototype. For the Phase II portion of this effort, the responsibility of relaying the status of the T-Connect request to a bus driver lies with the operations manager.

2.2 Web Browser

The IDTO user utilizes a standard web browser to perform many of the interactions with the IDTO Prototype, including creating an account, editing profile information, scheduling and maintaining trips. If the user is running the web browser on a device with location services enabled, even the progress of an active trip can be monitored.

In particular, the IDTO web applications will be tested against Microsoft's Internet Explorer 8 browser, however Apple's Safari, Google's Chrome, and Mozilla's Firefox are all expected to provide suitable results as well.

2.3 Mobile Device

Similar to Phase I of the IDTO demonstration, the primary means for a traveler to communicate with the IDTO prototype and the centerpiece of the Phase II effort is the IDTO Mobile Application residing on the traveler's mobile device.

Also similar to Phase I, the Phase II Demonstration of the IDTO mobile application will be implemented for only the Android and iOS platforms.

2.4 Mobile Data Terminal

The Mobile Data Terminal (MDT) enables a bus driver to interact with the IDTO Prototype. Not all buses will be equipped with an MDT. For the Phase I effort, only a private Demand/Response Provider's vehicles were so equipped. For those vehicles, however, the MDT will not only dynamically track the location of the vehicle, but will also allow a driver the ability to trigger a T-Connect on behalf of a traveler when the traveler's active trip is in jeopardy of missing a connection to a subsequent fixed route provider's service.

2.5 T-Connect Opportunity Manager

The T-CONNECT Opportunity Manager, or TOM, is the main component of the T-CONNECT system for IDTO. The goal of T-CONNECT is to improve rider satisfaction and reduce expected trip time for travelers by increasing the probability of automatic intermodal or intra-modal connections. T-CONNECT will protect transfers between both transit (e.g., bus, subway and commuter rail) and non-transit (e.g., shared ride modes) modes, and facilitates coordination between multiple agencies to accomplish the tasks. The TOM monitors trips scheduled by IDTO riders for T-CONNECT opportunities and automatically requests a T-CONNECT when the possibility of missing a transfer occurs. These trips are retrieved from the Data Repository, including the trip's scheduled departure time, the trip's steps, including transit authority name, Transit Vehicle Route Number, departure time, departure stop location, arrival time, and arrival stop location.

For purpose on the prototype demonstration, all T-CONNECT requests and responses are logged to the Data Repository.

The TOM enables the IDTO prototype to notify the traveler of the status of a T-CONNECT request throughout the course of the protected trip. The T-CONNECTION status is sent using notification methods on the traveler's mobile device including real-time status updates for that T-CONNECTION.

2.6 Data Repository

The IDTO Prototype utilizes both dynamic and static data in order to facilitate the functions it provides. All data is persisted, i.e. stored permanently, via the data repository. The repository manages the database and data-cache.

2.7 Traveler Services

In order to facilitate the interaction of the IDTO Backend System with the User facing components (mobile apps and web sites), the IDTO Prototype system will employ two web based interfaces. A RESTful Web Service will be deployed to provide a data interface allowing mobile apps to send and receive data to the IDTO Backend System. A second component will be the IDTO Web Server which will host the Traveler web portal.

The IDTO Web Service is a RESTful interface providing the needed functionality for the User facing mobile apps to send and receive the data needed to fulfill the traveler's needs. All interaction with the IDTO Backend System will be performed through the interfaces provided by this web service. Activities supported include:

- User account registration
- Account management
- Travel option searches
- Trip selection
- Saved trip review/cancel
- Trip Status notification
- On-trip confirmation

Data interchange by this web service will utilize the JavaScript Object Notation (JSON), a lightweight, human readable format for transferring data in a platform agnostic manner.

The IDTO Web Server serves as the component which hosts the Traveler Web Portal. While the primary use case for the system is that the Traveler will interact with the system via a mobile app, a web site is also planned which will allow the Traveler to interact with the IDTO Prototype system from any internet enabled computer or device. This Traveler Web Portal will include many of the same features available via the mobile app. By supplying a web site, the traveler can plan future travel needs, review their existing planned trips and update their account profile outside of the mobile app. This allows those travelers without smartphones or mobile internet devices to also participate in the IDTO Prototype demonstration.

2.8 Vehicle Location Service

The Vehicle Location Service (VLS) is responsible for integrating vehicle location information from a variety of external sources in a variety of forms with the traveler's active trips. This real-time location information is necessary for the TOM to determine exactly when a connection is in jeopardy, necessitating a protecting T-Connect to be issued.

2.9 Route Aggregation Service

The Route Aggregation Service (RAS) is the component responsible for abstracting the task of assembling the various legs of a given trip. The RAS is called by the IDTO web service (IWS) which provides the trip details such as start-time and location, destination location, and travel mode(s).

The RAS will abstract the various route providers. Given trip information, the RAS coordinates the relevant route providers and returns one or more potential trip routes to the IWS.

2.10 Geolocation Service

The Route Provider locates available trips based on the starting and ending geographic location (Latitude and Longitude coordinates). Most travelers will be planning their trips based on street addresses. The Geolocation Service is used to translate the addresses entered by a Traveler into the corresponding geographic location needed by the Route Provider.

2.11 Rideshare Service

The Rideshare Service is a component external to the IDTO Prototype which provides available route alternatives on a dynamic basis. Only traveler's previously subscribed to the Rideshare service will have these route alternatives presented to them during trip scheduling. Rideshare is presently only expected to fulfil an entire trip request for the traveler, however, as the prototype progresses and rideshare is adopted, this may be revisited in order to consider trips which include both rideshare and other transportation means.

2.12 Demand / Response Operations

This external component is the source of scheduling information for the Demand / Response Provider used by the IDTO Prototype. Data exchange is initiated by the IDTO Prototype as needed.

2.13 VLS Provider

This external component provides dynamic vehicle location data to the IDTO Prototype, enabling it to determine when traveler transfers are in jeopardy and thus requiring a T-Connect to be requested.

2.14 Route Provider

A Route Provider is a specific implementation of a routing service. The initial routing services will include LYNX, SunRail, UCF Black and Gold shuttle service, FlexBus, and Zimride. A routing service is considered to be either a static service (LYNX: requiring General Transit Feed Specification (GTFS) route data) or a dynamic service (FlexBus, Zimride). Route Providers are used by the Route Aggregation Service when assembling the components of a trip.

2.15 Fixed Route Operations

This external component is the source of GTFS data for a Fixed Route Provider. Ideally the provider pushes updated GTFS data to the IDTO Prototype as their published routes change, typically on a quarterly or semi-annual basis. As in Phase I, the GTFS data exchange for Phase II will be controlled by the IDTO Prototype, minimizing the impact on the external systems involved.

GTFS is a common format for public transportation schedules and associated geographic information. GTFS "feeds" allow public transit agencies to publish their transit data and allow applications such as the IDTO Prototype to consume that data in an interoperable way.

The IDTO Prototype receives schedules via GTFS feeds from all of its Fixed Route Providers, Incoming and Private Incoming, as well as the T-Connect Provider.

Chapter 3 Phase II Demonstration Site

The IDTO Phase II demonstration will be conducted in Orlando, FL. The transportation providers and their respective role in the demonstration include the LYNX as the Fixed Route / T-Connect Provider; UCF as the Incoming Fixed Route Provider; FlexBus, as a demand/response service; Zimride, a nationally recognized service provider for rideshare capabilities, as the Rideshare Provider, and finally, SunRail serves as an incoming service provider.

3.1 LYNX

LYNX, also known as the Central Florida Regional Transportation Authority, operates a fixed-route, local bus service for Orlando, Seminole and Osceola county, as well as portions of Polk and Lake county, in central Florida. In the scenarios described here, the LYNX fixed-service routes will be expanded to support the T-CONNECT application and will allow these requests to be made to the various buses serving the demonstration area.

For the Phase II demonstration, LYNX will support the T-CONNECT application by integration with their dispatching operations.

3.2 SunRail

SunRail is Central Florida's new commuter rail system, currently connecting DeBary with Sand Lake Road, near the Orlando International Airport. SunRail provides fixed-route, fixed schedule service. If real-time AVL capabilities are made available, SunRail will be used as a T-CONNECT provider for the Phase II demonstration.

3.3 University of Central Florida Shuttle Service

The University of Central Florida provides free shuttle service from 17 local student residential communities around the UCF main campus, as well as to the Health Sciences campus at Lake Nona. The Black and Gold shuttles operate on a fixed route, fixed headway schedule, and have full AVL capabilities using Zonar, an electronic GPS tracking system used internally for UCF. For the Phase II demonstration, the AVL capabilities of the UCF operated Black and Gold shuttles are expected to be used to allow incoming connections to provide for protected transfers with a connecting LYNX service.

3.4 FlexBus

FlexBus, a demonstration project sponsored by LYNX which recently completed its proof-of-concept period, is expected to provide a station to station, demand / response service using minibuses in the Central Florida area, with stations located at key activity centers throughout Altamonte Springs, Longwood, Casselberry and Maitland. Stations are located within a short walk of major destinations including SunRail stations.

FlexBus will provide the demand/response component of T-DISP capability for the demonstration of T-CONNECT in Phase II.

NOTE: The FlexBus deployment schedule ultimately did not support the necessary schedule for IDTO, and as such, any reference to this service included herein is purely conceptual.

3.5 Zimride

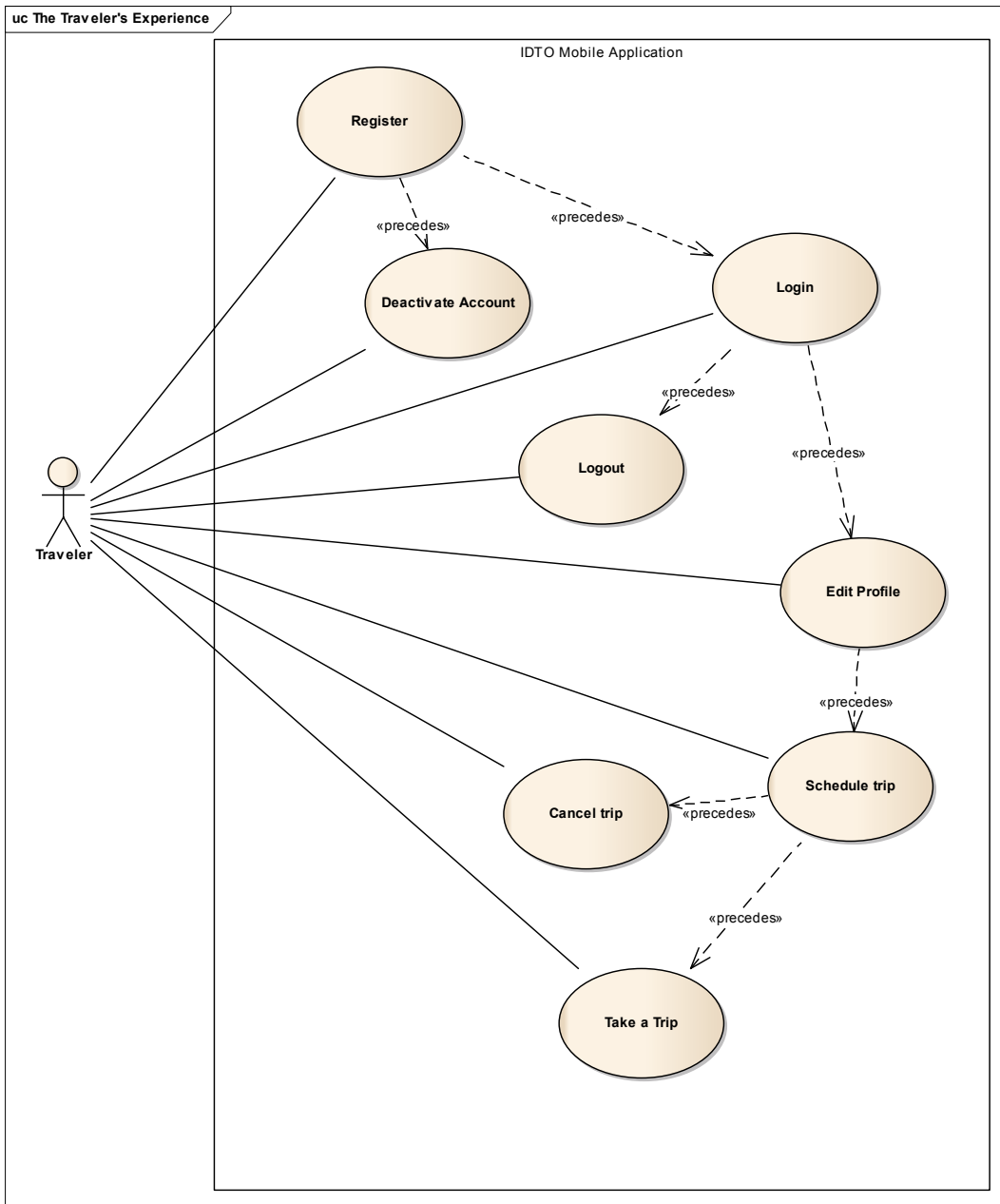
Zimride will serve as the D-RIDE provider, a capability that they presently offer in a number of campus and urbanized areas throughout the country. Through the use of social-networking tools, Zimride enables the matching of drivers and riders for both ad-hoc and pre-planned trips. The IDTO Prototype will make dynamic rideshare options available to participants through the IDTO application.

NOTE: Due to privacy concerns, Zimride was not evaluated as part of the Phase II demonstration, but instead was integrated into the latest version of IDTO but demonstrated instead with data from the Phase I providers. This did not affect the design of the service, and as such, remains in this document.

Chapter 4 Use Case Model

4.1 Traveler Experience

The following use cases, as shown in Figure 4-1, describe the interactions of the traveler with the IDTO Prototype. Most of the interactions can be made using either the IDTO mobile application on the traveler's mobile device, or through the IDTO web site, using a standard browser.

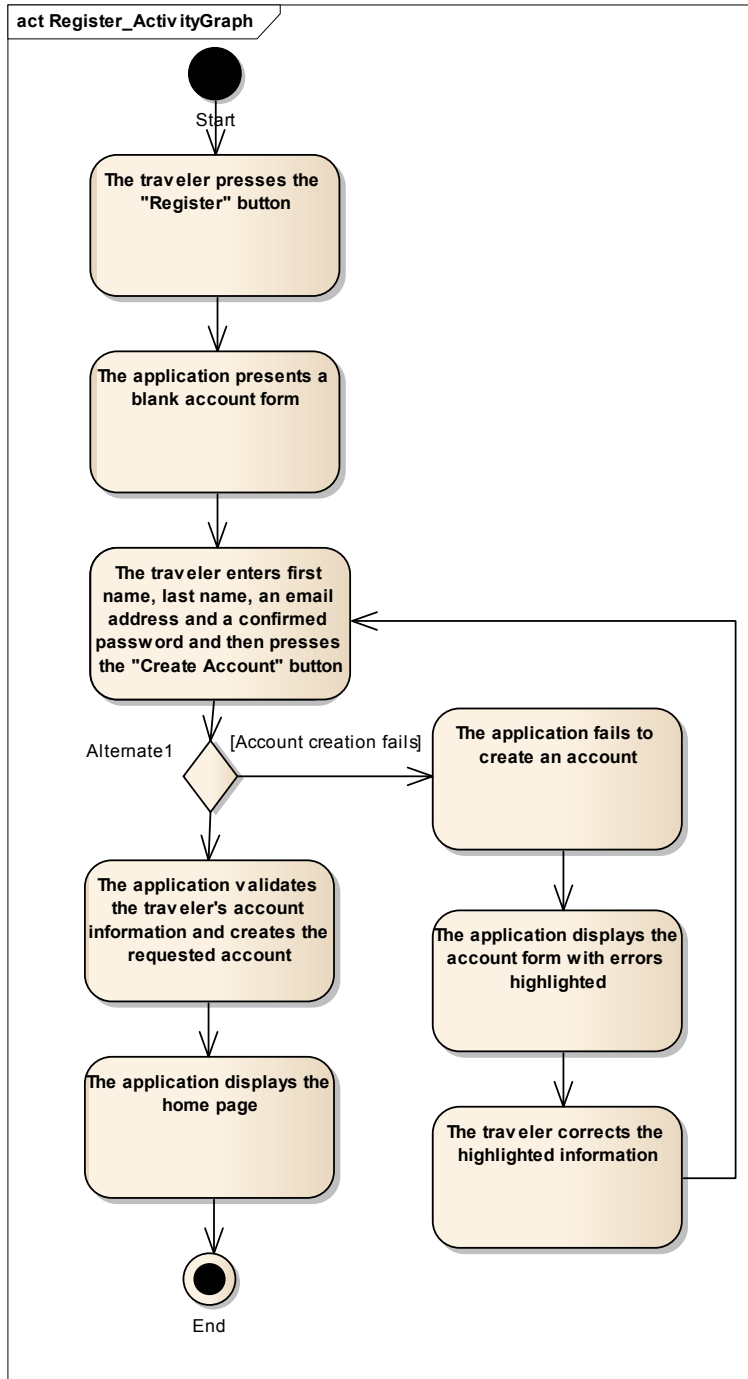


Source: Battelle

Figure 4-1. The Traveler's Experience

4.1.1 Register

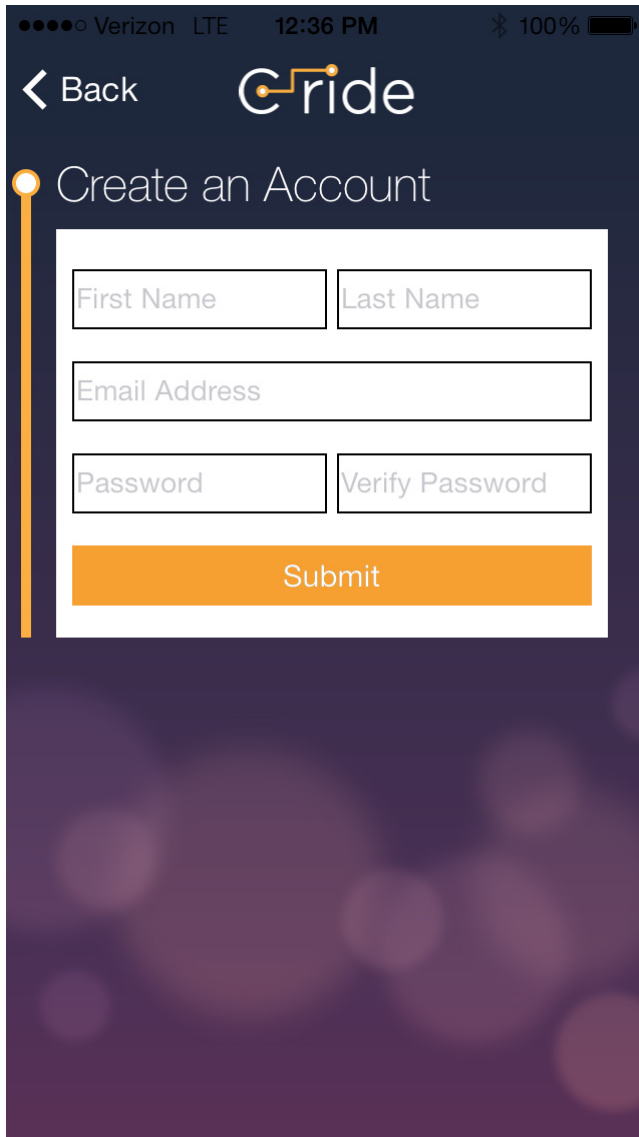
Before a traveler can interact with the IDTO prototype, he must first register an email address on the system. The steps for registering an email address and mockups for the related mobile application screens are described here.



Source: Battelle

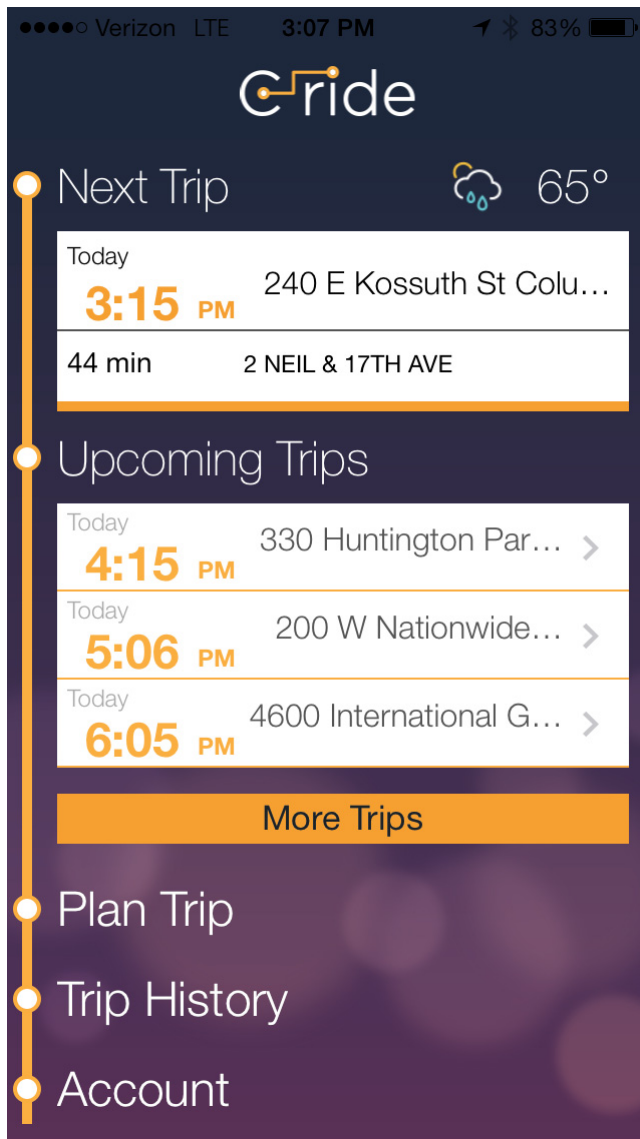
Figure 4-2. Register_ActivityGraph

U.S. Department of Transportation, Research and Innovative Technology Administration
Intelligent Transportation Systems Joint Program Office



Source: Battelle

Figure 4-3. The Registration Page



Source: Battelle

Figure 4-4. The Home Page

4.1.1.1 Register

Basic Path

Basic Path:

¹ The traveler presses the "Register" button (Figure 4-6). ² The application presents a blank account form (Figure 4-3). ³ The traveler enters first name, last name, an email address and a confirmed password and then presses the "Create Account" button. ⁴ The application validates the traveler's account information and creates the requested account. ⁵ The application displays the home page (Figure 4-4).

Alternate Paths

Account creation fails:

(at ⁴)

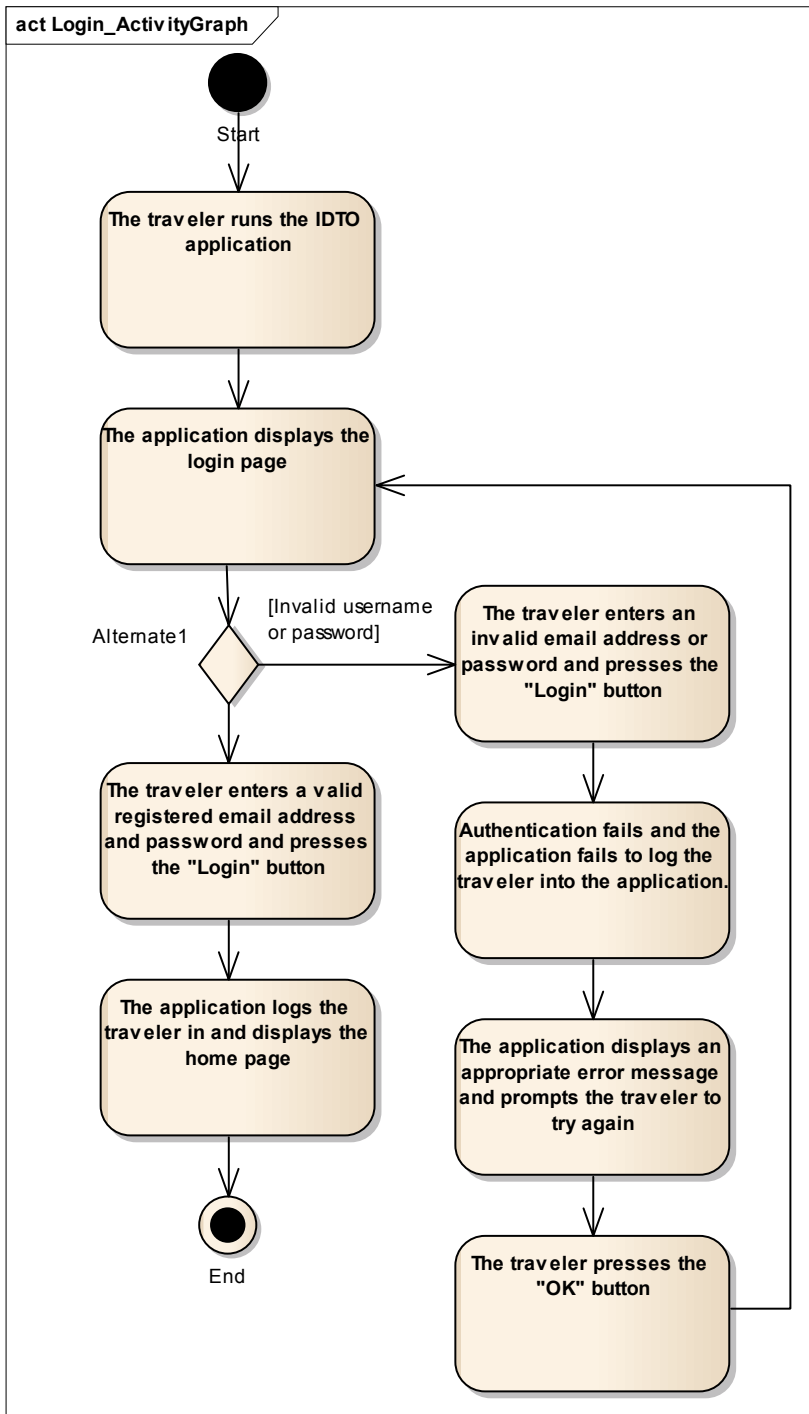
The application fails to create an account. The application displays the account form with errors highlighted. The traveler corrects the highlighted information.

Pre-condition

The traveler is not logged in to the IDTO mobile application

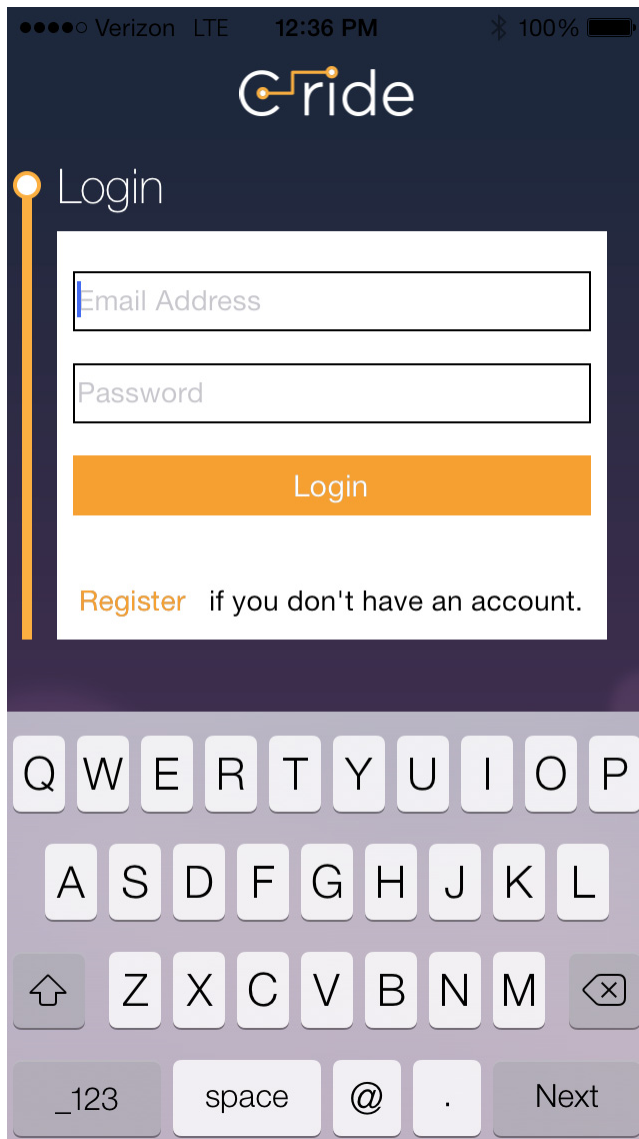
4.1.2 Login

Once an account is created, the traveler can run the mobile application and login.



Source: Battelle

Figure 4-5. Login_ActivityGraph



Source: Battelle

Figure 4-6. The Login Page

4.1.2.1 Login

Basic Path

Basic Path:

¹ The traveler runs the IDTO application. ² The application displays the login page (Figure 4-6). ³ The traveler enters a valid registered email address and password and presses the "Login" button. ⁴ The application logs the traveler in and displays the home page (Figure 4-6).

Alternate Paths

Saved credentials:

The application logs the traveler in using saved credentials.

Invalid username or password:

(at ³)

The traveler enters an invalid email address or password and presses the "Login" button. Authentication fails and the application fails to log the traveler into the application. The application displays an appropriate error message and prompts the traveler to try again. The traveler presses the "OK" button.

Exceptions

Exit application:

The traveler quits the IDTO mobile application.

Login Fails:

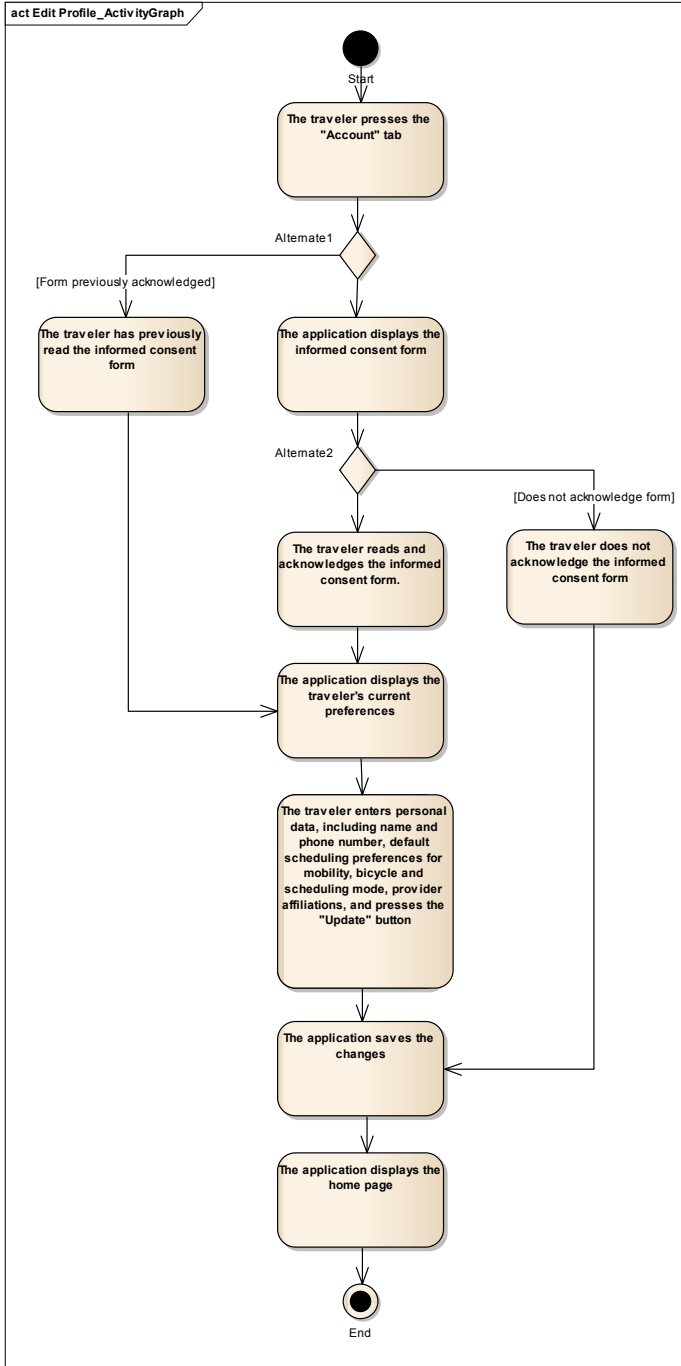
The Traveler enters an invalid email address or password for the third time. The application displays an appropriate error message.

Pre-condition

The traveler is not logged in to the IDTO mobile application

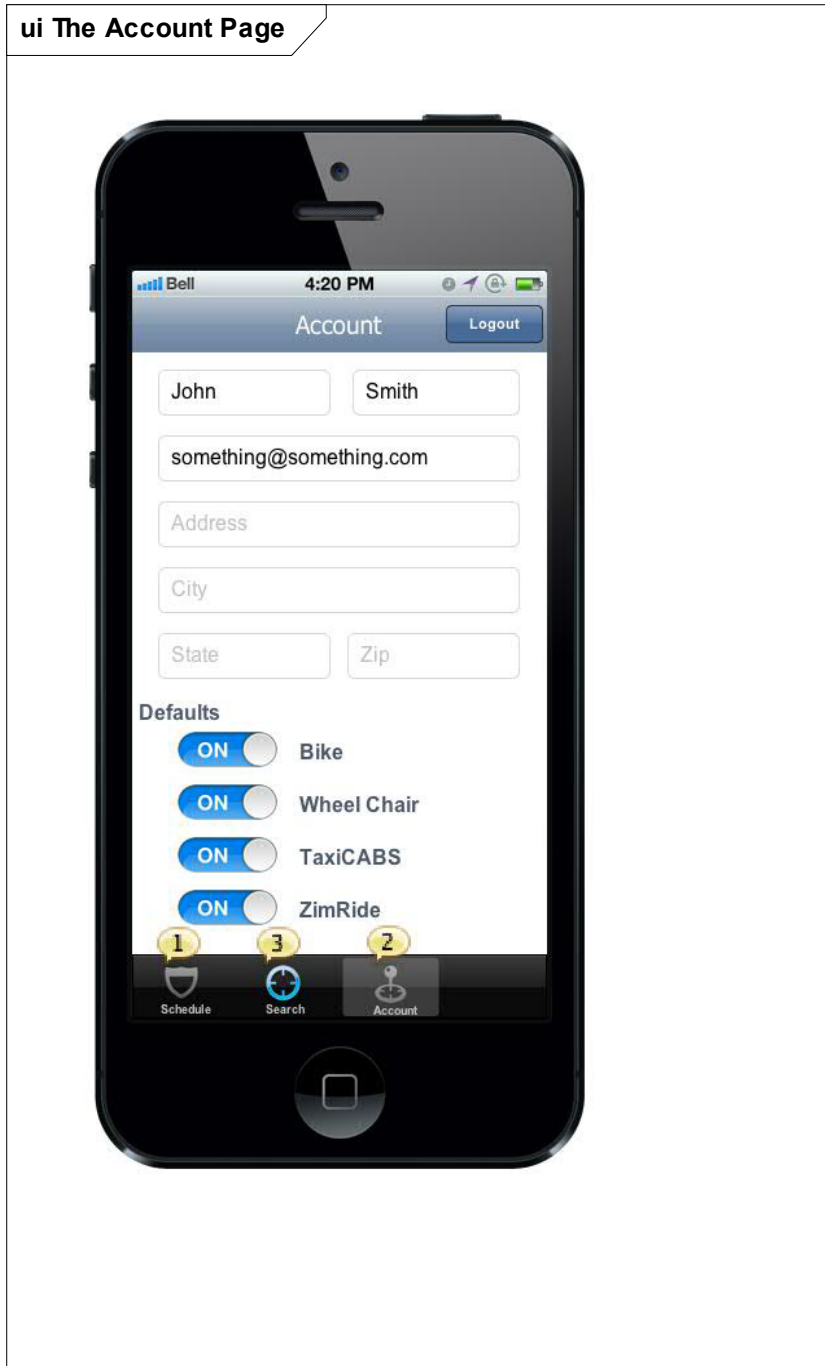
4.1.3 Edit Profile

Once the traveler has created an account and logged into the mobile application, he is able to set a number of preferences in his profile. These preferences tailor the traveler's experience to their personal situation.



Source: Battelle

Figure 4-7. Edit Profile_ActivityGraph



Source: Battelle

Figure 4-8. The Account Page

4.1.3.1 *Edit Profile*

Basic Path

Basic Path:

¹ The traveler presses the "Account" tab (Figure 4-8). ² The application displays the informed consent form. ³ The traveler reads and acknowledges the informed consent form. ⁴ The application displays the traveler's current preferences. ⁵ The traveler enters personal data, including name and phone number, default scheduling preferences for mobility, bicycle and scheduling mode, provider affiliations, and presses the "Update" button. ⁶ The application saves the changes. ⁷ The application displays the home page.

Alternate Paths

Finished editing profile:

Does not acknowledge form:

(at ³)

The traveler does not acknowledge the informed consent form.

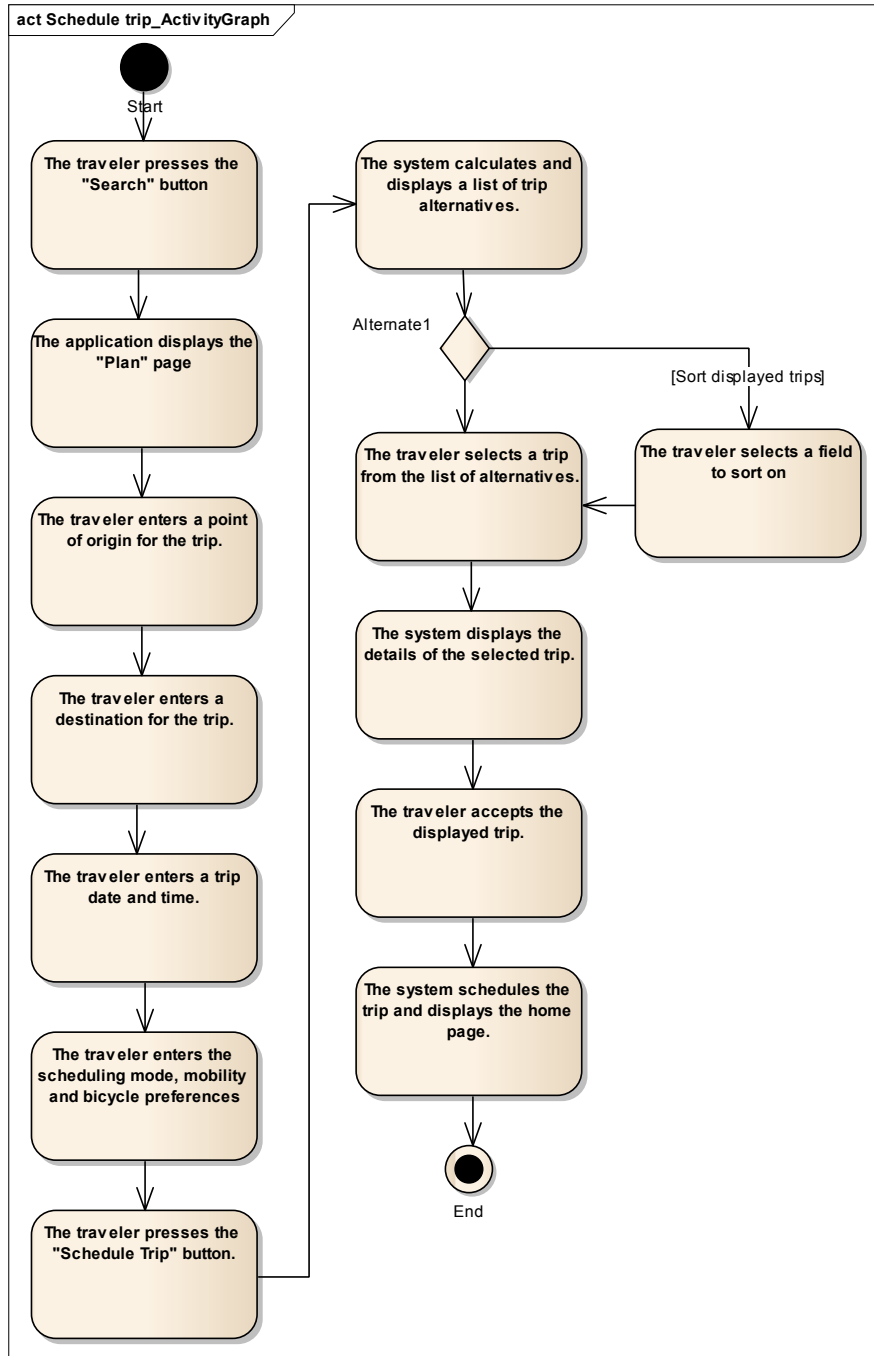
Form previously acknowledged:

(at ²)

The traveler has previously read the informed consent form.

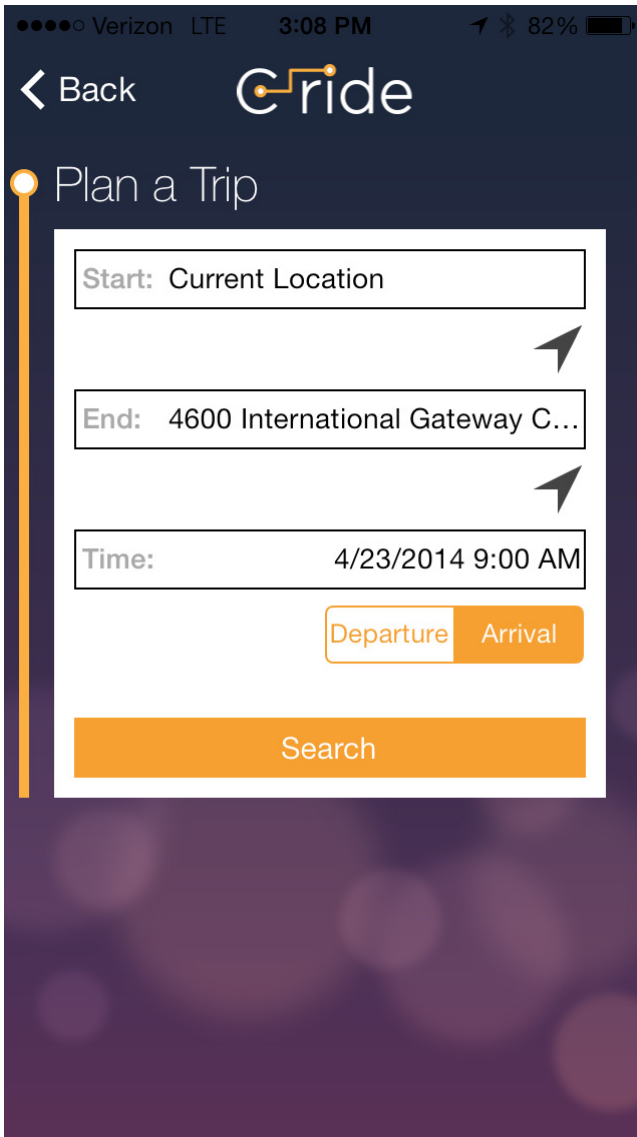
4.1.4 Plan Trip

Once the traveler's preferences are set, he is ready to plan a trip. The traveler's preferences help determine what trip providers are available to the traveler, as well as making the planning process more efficient.



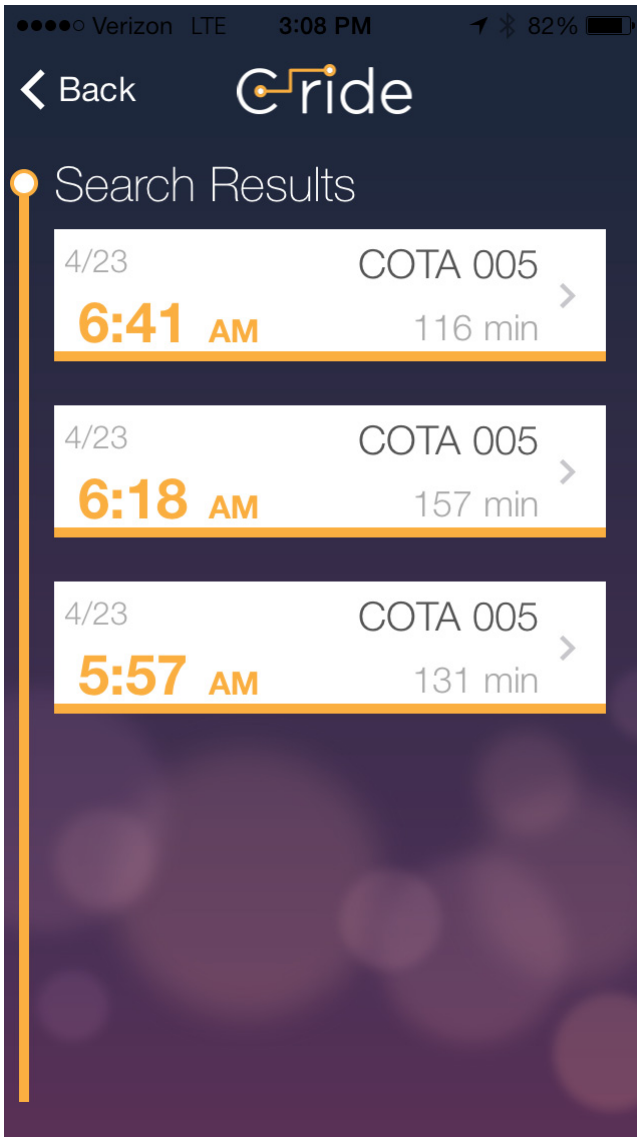
Source: Battelle

Figure 4-9. Schedule trip_ActivityGraph



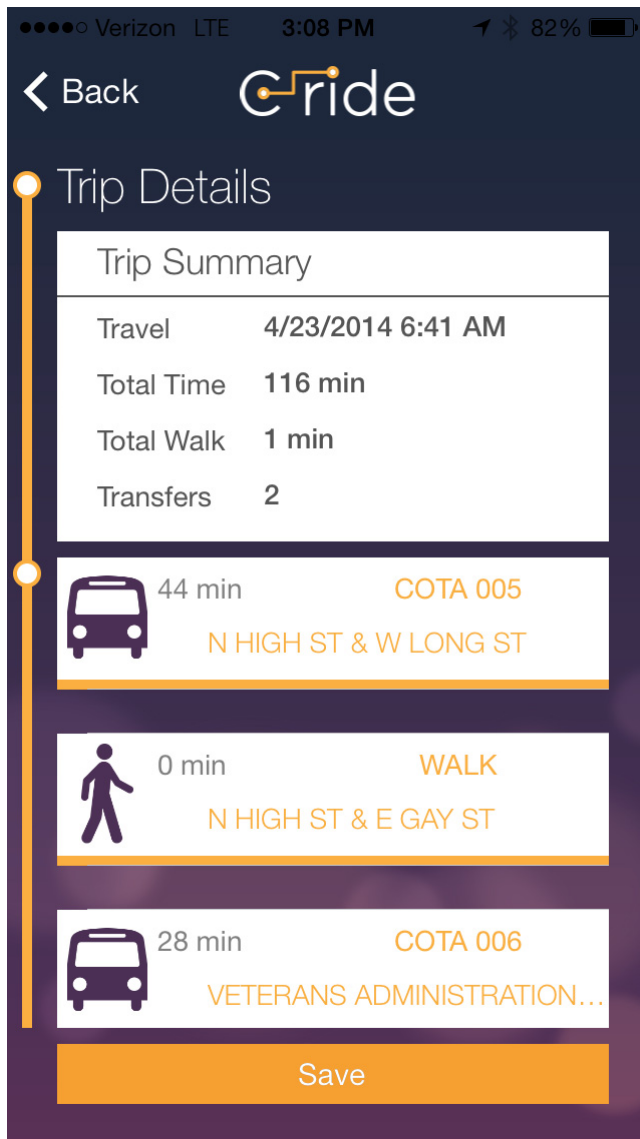
Source: Battelle

Figure 4-10. The Trip Planning Page



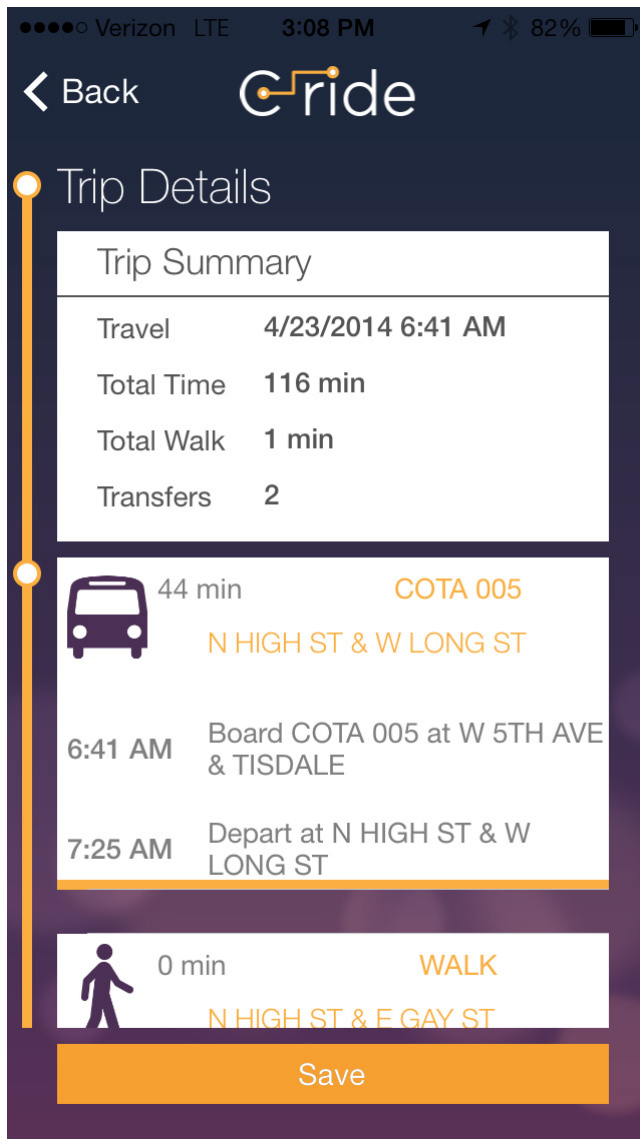
Source: Battelle

Figure 4-11. The Results Page



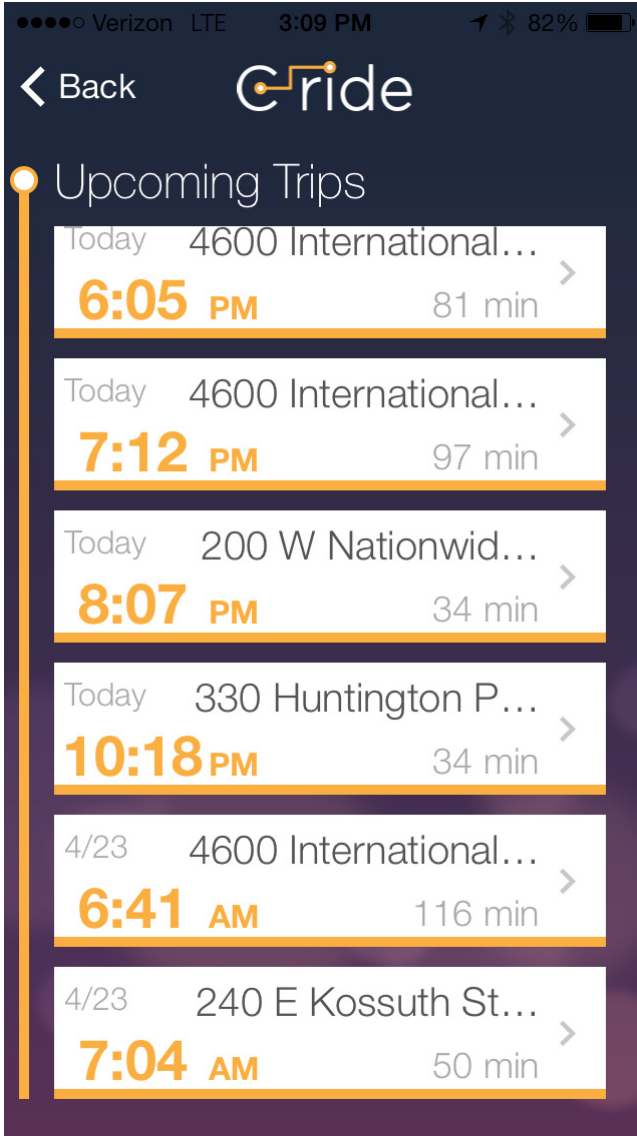
Source: Battelle

Figure 4-12. The Trip Detail Page



Source: Battelle

Figure 4-13. The Step Detail Page



Source: Battelle

Figure 4-14. View Past Trips

4.1.4.1 *Schedule trip*

This is the Schedule Trip description.

Basic Path

Basic Path:

¹ The traveler presses the "Search" button. ² The application displays the "Plan" page (Figure 4-10). ³ The traveler enters a point of origin for the trip. ⁴ The traveler enters a destination for the trip. ⁵ The traveler enters a trip date and time. ⁶ The traveler enters the scheduling mode, mobility and bicycle preferences. ⁷ The traveler presses the "Schedule Trip" button. ⁸ The system calculates and displays a list of trip alternatives (Figure 4-11). ⁹ The traveler selects a trip from the list of alternatives. ¹⁰ The system displays the details of the selected trip (Figure 4-12). ¹¹ The traveler accepts the displayed trip. ¹² The system schedules the trip and displays the home page.

Alternate Paths

Alternate1:

Previously entered destination:

The traveler selects a destination from a list of previously entered locations.

Current location is the destination:

The user selects the current location as the destination.

Previously entered point of origin:

The traveler selects a point of origin from a list of previously entered locations.

Current location is the point of origin:

The user selects the current location as the point of origin.

Select a scheduling mode:

The traveler selects a scheduling mode.

Sort displayed trips:

(at ⁹)

The traveler selects a field to sort on.

Select a mobility preference:

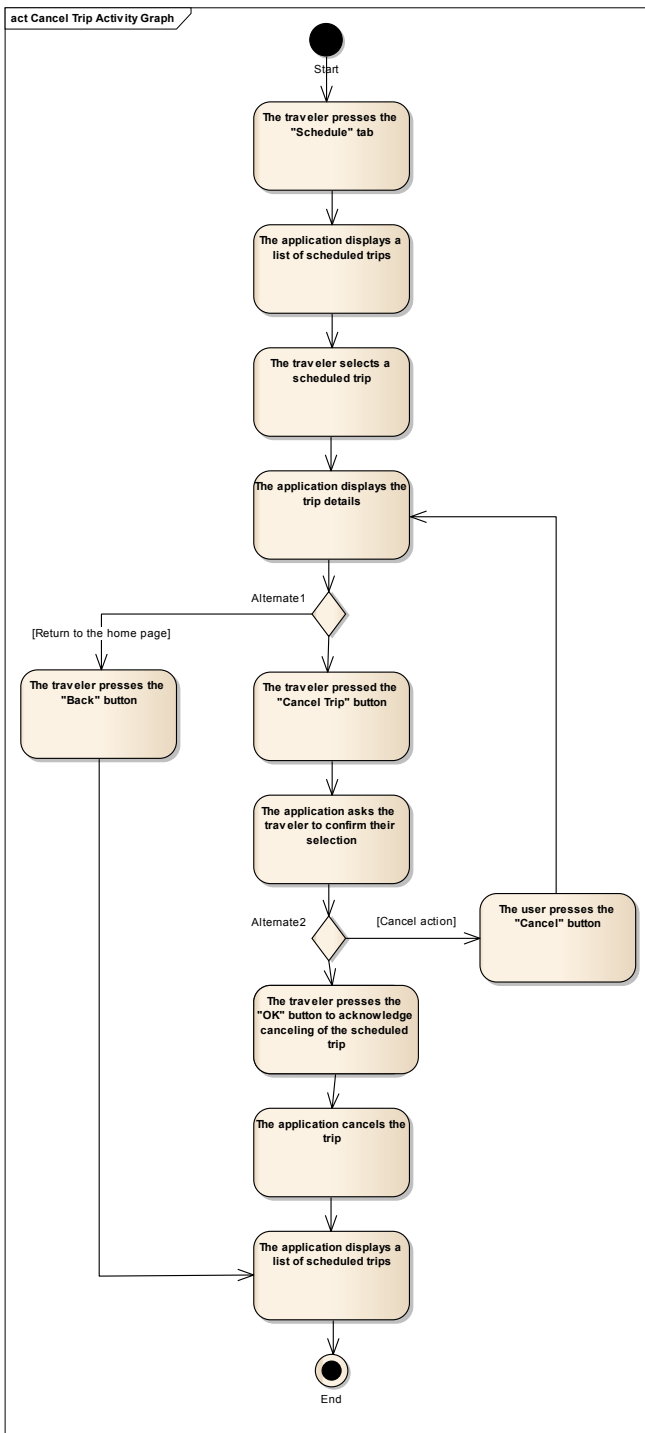
The traveler selects a mobility preference.

Select a bicycle preference:

The traveler selects a bicycle preference.

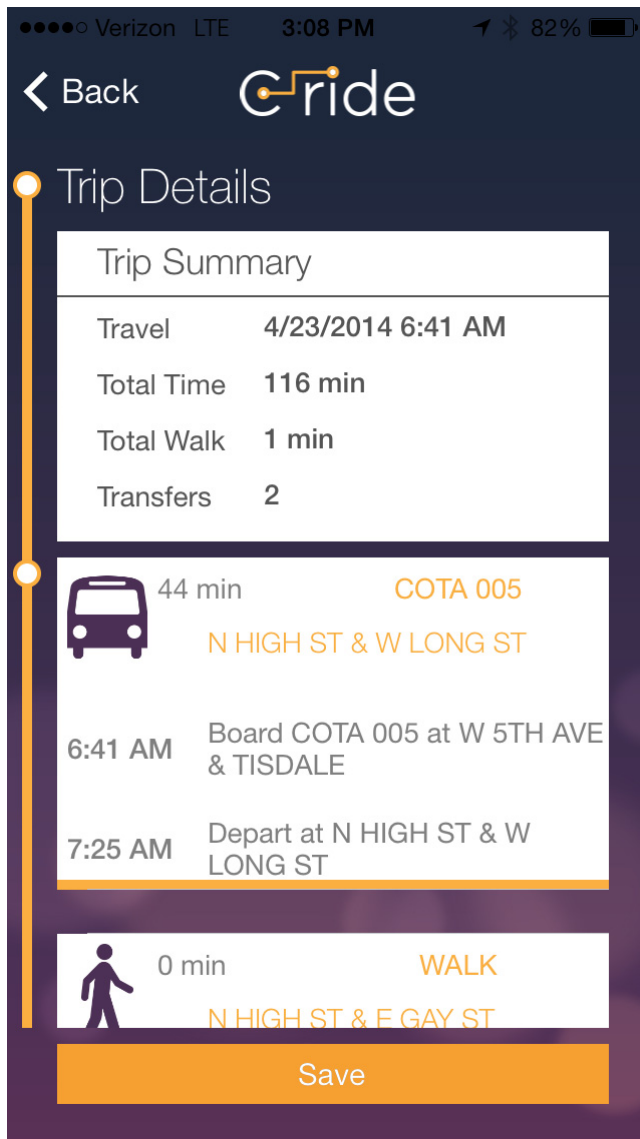
4.1.5 Cancel Trip

The traveler cancels a previously scheduled trip using the following steps.



Source: Battelle

Figure 4-15. Cancel Trip Activity Graph



Source: Battelle

Figure 4-16. The Scheduled Trip Detail Page

4.1.5.1 *Cancel trip*

Basic Path

Basic Path:

¹ The traveler presses the "Schedule" tab. ² The application displays a list of scheduled trips. ³ The traveler selects a scheduled trip. ⁴ The application displays the trip details (Figure 4-16). ⁵ The traveler pressed the "Cancel Trip" button. ⁶ The application asks the traveler to confirm their selection. ⁷ The traveler presses the "OK" button to acknowledge canceling of the scheduled trip. ⁸ The application cancels the trip. ⁹ The application displays a list of scheduled trips.

Alternate Paths

Return to the home page:

(at ⁵)

The traveler presses the "Back" button.

Cancel action:

(at ⁷)

The user presses the "Cancel" button.

Pre-condition

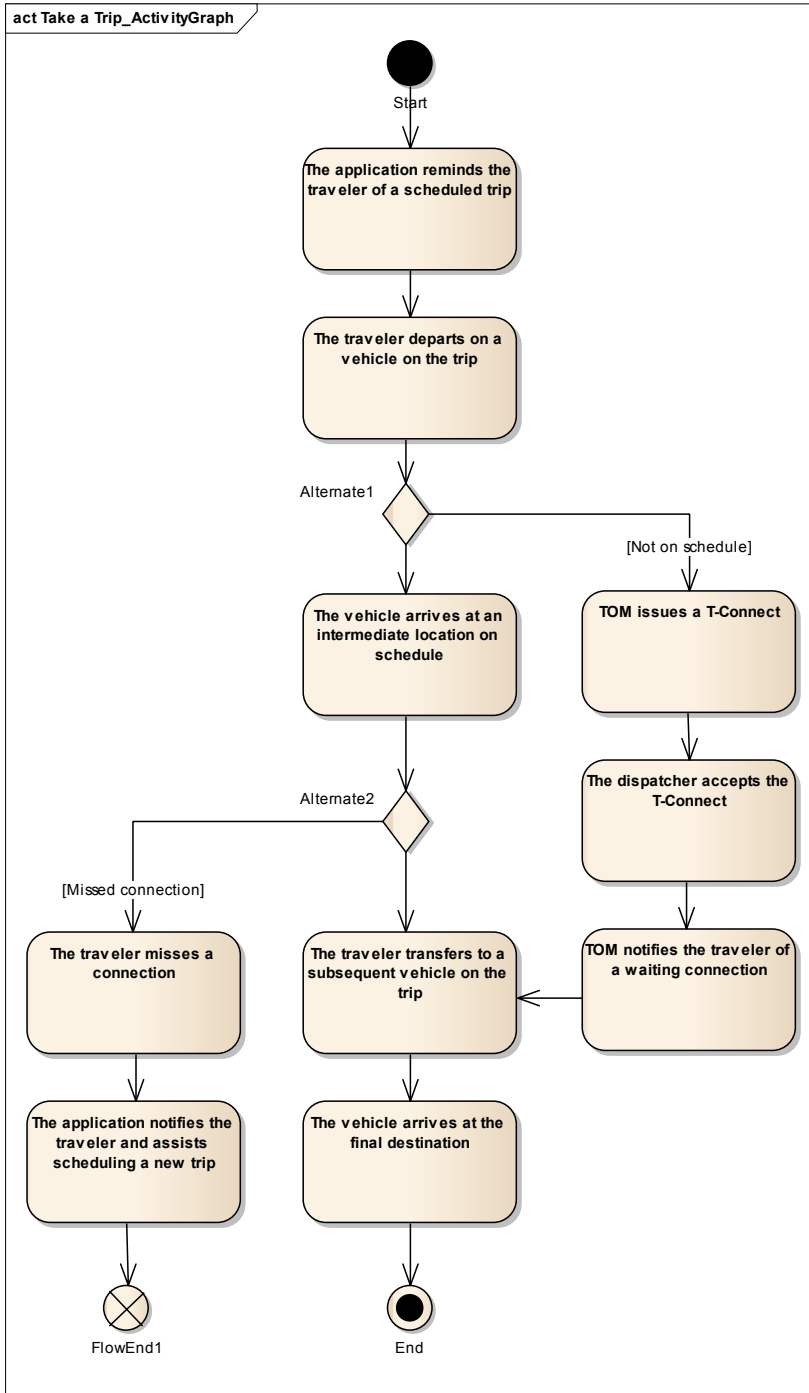
The traveler is on the home page of the application

Post-condition

The traveler is on the home page of the application

4.1.6 Take a Trip

The IDTO mobile application keeps the traveler informed of the status of his connections and transfers on the trip, and can aid the traveler in rescheduling a trip in the event a connection is missed.



Source: Battelle

Figure 4-17. Take a Trip_ActivityGraph

4.1.6.1 Take a Trip

Basic Path

Basic Path:

¹ The application reminds the traveler of a scheduled trip. ² The traveler departs on a vehicle on the trip. ³ The vehicle arrives at an intermediate location on schedule. ⁴ The traveler transfers to a subsequent vehicle on the trip. ⁵ The vehicle arrives at the final destination.

Alternate Paths

Not on schedule:

(at ³)

TOM issues a T-Connect. The dispatcher accepts the T-Connect. TOM notifies the traveler of a waiting connection.

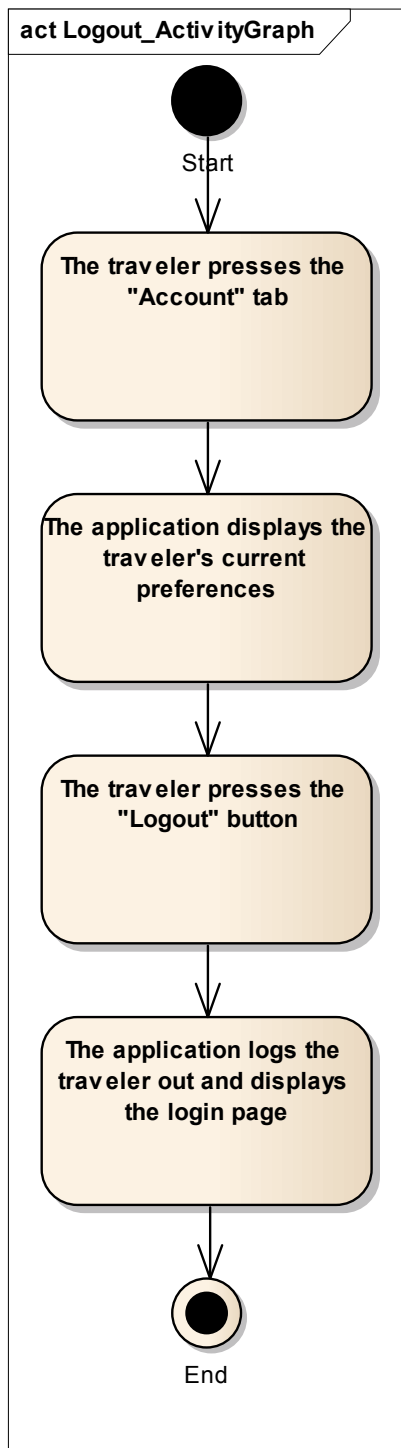
Missed connection:

(at ⁴)

The traveler misses a connection. The application notifies the traveler and assists scheduling a new trip.

4.1.7 Logout

At the end of a session the traveler logs out of the IDTO application.



Source: Battelle

Figure 4-18. Logout_ActivityGraph

U.S. Department of Transportation, Research and Innovative Technology Administration
Intelligent Transportation Systems Joint Program Office

4.1.7.1 Logout

Basic Path

Basic Path:

¹ The traveler presses the "Account" tab. ² The application displays the traveler's current preferences. ³ The traveler presses the "Logout" button. ⁴ The application logs the traveler out and displays the login page.

Pre-condition

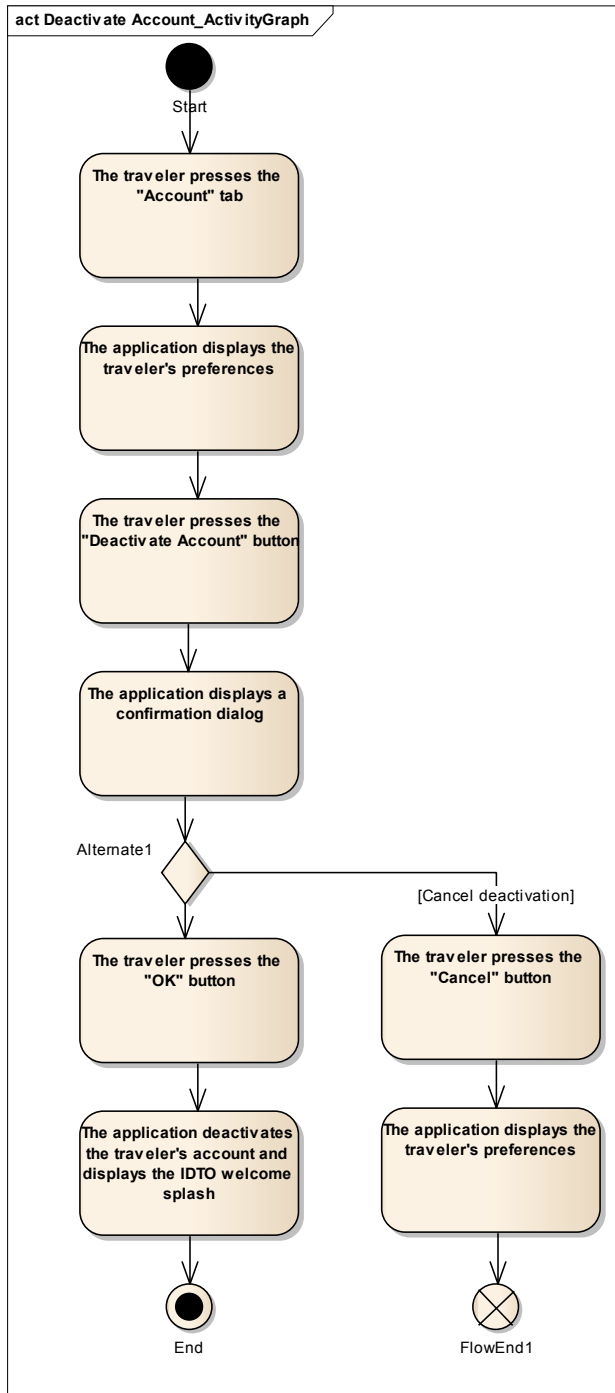
**The traveler is logged in to the IDTO
mobile application**

Post-condition

**The traveler is not logged in to the IDTO
mobile application**

4.1.8 Deactivate Account

The traveler may, at any time, cease their involvement with the IDTO Prototype and deactivate their account by following the following steps.



Source: Battelle

Figure 4-19. Deactivate Account_ActivityGraph

4.1.8.1 Deactivate Account

Basic Path

Basic Path:

¹ The traveler presses the "Account" tab. ² The application displays the traveler's preferences. ³ The traveler presses the "Deactivate Account" button. ⁴ The application displays a confirmation dialog. ⁵ The traveler presses the "OK" button. ⁶ The application deactivates the traveler's account and displays the IDTO welcome splash.

Alternate Paths

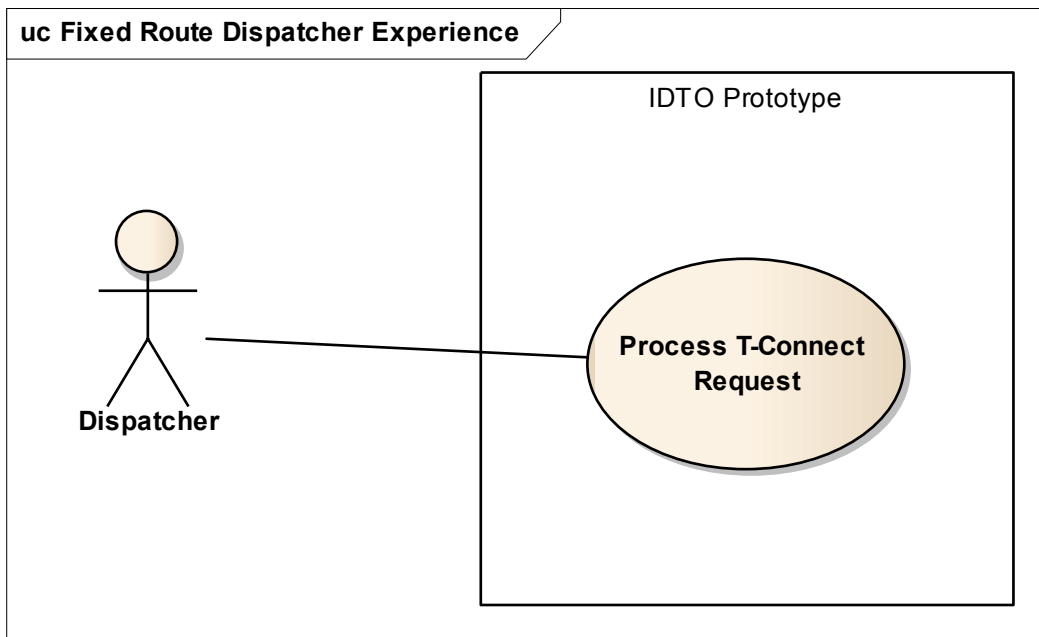
Cancel deactivation:

(at ⁵)

The traveler presses the "Cancel" button. The application displays the traveler's preferences.

4.2 Fixed Route Dispatcher Experience

For the Phase I effort, the sum total of the dispatcher's interaction with the IDTO Prototype is to accept or reject T-Connect requests presented by the system. These activities are described here.



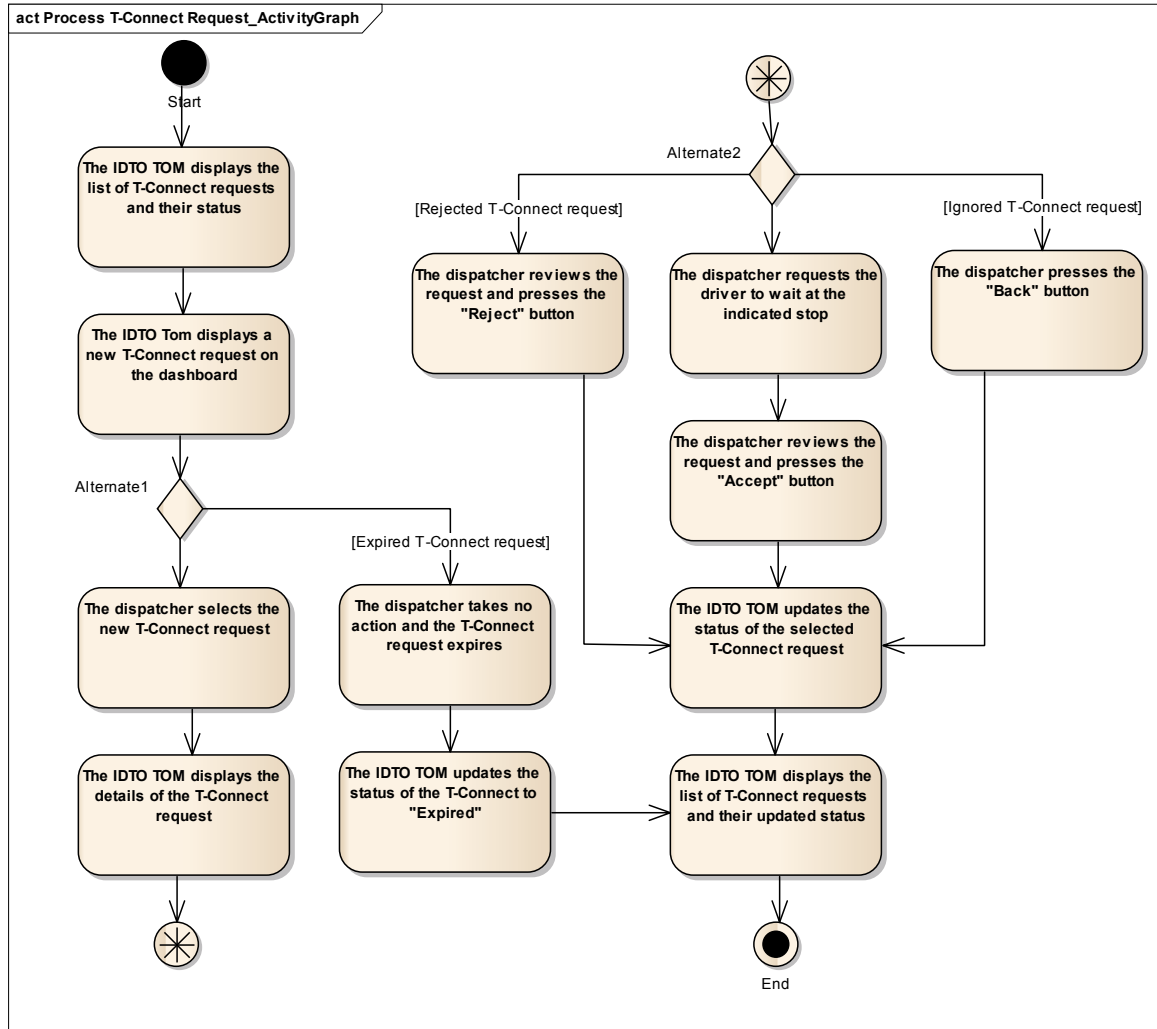
Source: Battelle

Figure 4-20. Fixed Route Dispatcher Experience

4.2.1 Process T-Connect Request

The core of the IDTO Prototype is the T-Connect. A T-Connect is a request to a Provider to hold a vehicle at a scheduled stop for a brief period of time, long enough for a traveler from a late arriving incoming connection to make the transfer. The T-Connect request gives the traveler a chance to make a connection during a scheduled trip that would otherwise not be possible.

For the Phase I implementation, the dispatcher of the Fixed Route Provider is responsible for either accepting or rejecting a T-Connect request, and subsequently relaying appropriate instructions to a vehicle driver.



Source: Battelle

Figure 4-21. Process T-Connect Request_ActivityGraph

4.2.1.1 *Process T-Connect Request*

Basic Path

Basic Path:

¹ The IDTO TOM displays the list of T-Connect requests and their status. ² The IDTO Tom displays a new T-Connect request on the dashboard. ³ The dispatcher selects the new T-Connect request. ⁴ The IDTO TOM displays the details of the T-Connect request. ⁵ The dispatcher requests the driver to wait at the indicated stop. ⁶ The dispatcher reviews the request and presses the "Accept" button. ⁷ The IDTO TOM updates the status of the selected T-Connect request. ⁸ The IDTO TOM displays the list of T-Connect requests and their updated status.

Alternate Paths

Rejected T-Connect request:

(at ⁵)

The dispatcher reviews the request and presses the "Reject" button.

Expired T-Connect request:

(at ³)

The dispatcher takes no action and the T-Connect request expires. The IDTO TOM updates the status of the T-Connect to "Expired".

No Action:

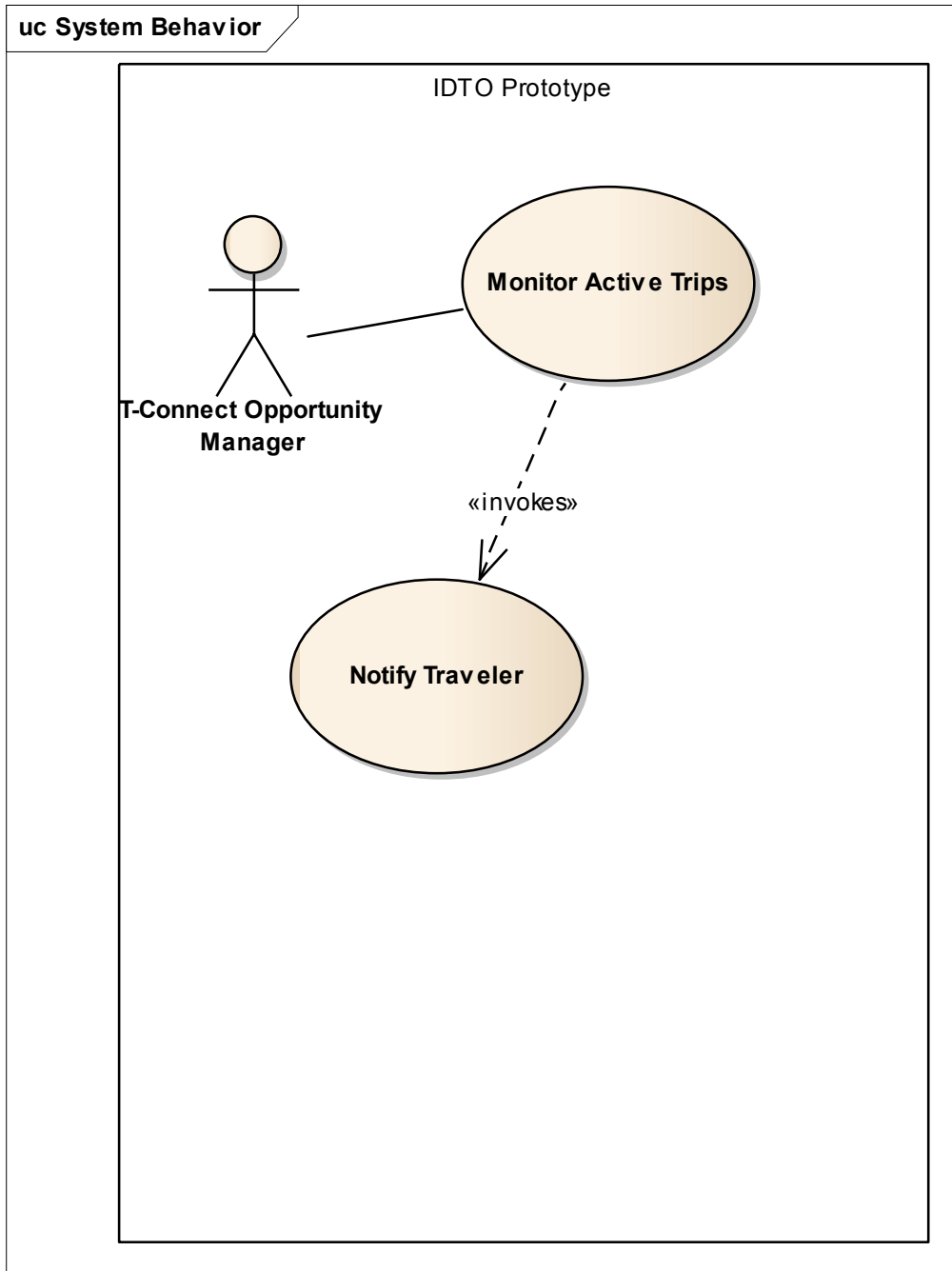
Ignored T-Connect request:

(at ⁵)

The dispatcher presses the "Back" button.

4.3 System Behavior

Many of the details of operation in the IDTO Prototype are not mediated by human intervention. These activities between the various components, internal and external to the system, are described here.

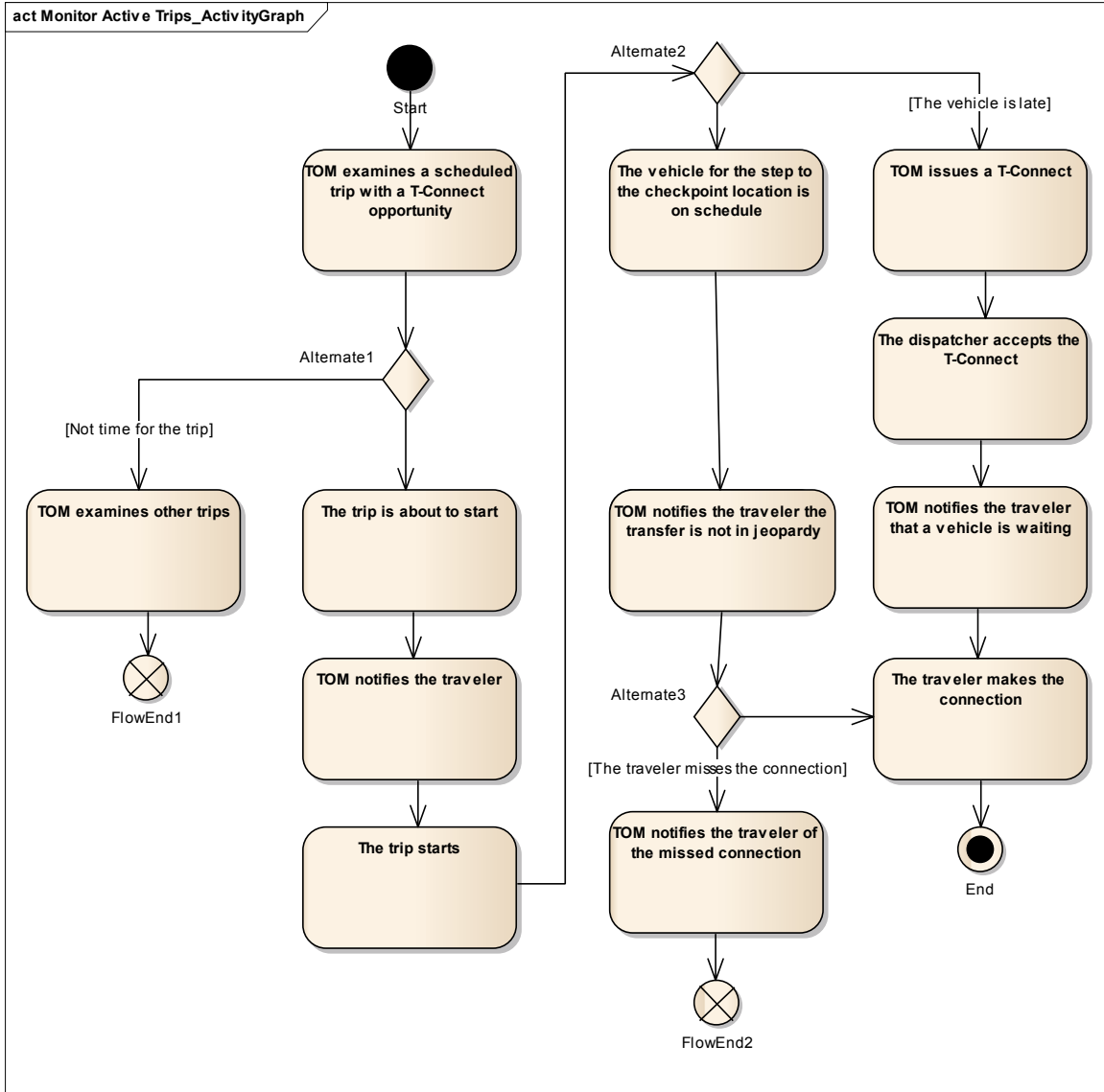


Source: Battelle

Figure 4-22. System Behavior

4.3.1 Monitor Active Trips

The TOM is the heart of the IDTO Prototype. The TOM actively monitors all scheduled trips in the IDTO Prototype, notifying the traveler of upcoming trips and issuing T-Connect requests when a traveler is observed in jeopardy of missing a connection.



Source: Battelle

Figure 4-23. Monitor Active Trips_ActivityGraph

4.3.1.1 *Monitor Active Trips*

Basic Path

Basic Path:

¹ The TOM examines a scheduled trip with a T-Connect opportunity. ² The trip is about to start. ³ The TOM notifies the traveler. ⁴ The trip starts. ⁵ The vehicle for the step to the checkpoint location is on schedule. ⁶ The TOM notifies the traveler the transfer is not in jeopardy. ⁷ The traveler makes the connection.

Alternate Paths

Not time for the trip:

(at ²)

The TOM examines other trips.

The vehicle is late:

(at ⁵)

The TOM issues a T-Connect. The dispatcher accepts the T-Connect. The TOM notifies the traveler that a vehicle is waiting.

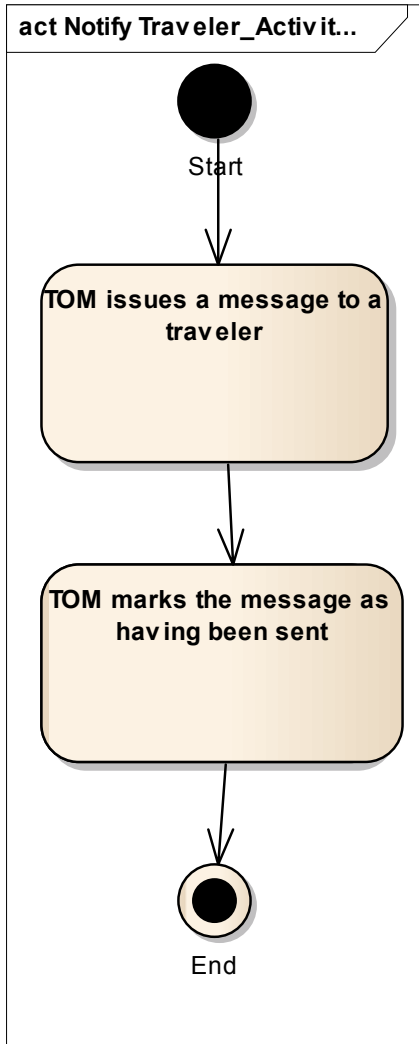
The traveler misses the connection:

(at ⁷)

The TOM notifies the traveler of the missed connection.

4.3.2 Notify Traveler

Various events noticed by the TOM are of interest to the traveler. The TOM relays messages to the traveler at opportune times to inform the traveler of these events.



Source: Battelle

Figure 4-24. Notify Traveler_ActivityGraph

4.3.2.1 Notify Traveler

Basic Path

Basic Path:

¹ The TOM issues a message to a traveler. ² The TOM marks the message as having been sent.

Chapter 5 Interfaces

The IDTO Prototype interfaces with five categories of service providers, including three variations of fixed route providers, one demand/response provider, and one rideshare provider. The general outline of the interfaces for each of these providers is described in the following sections.

5.1 Traveler

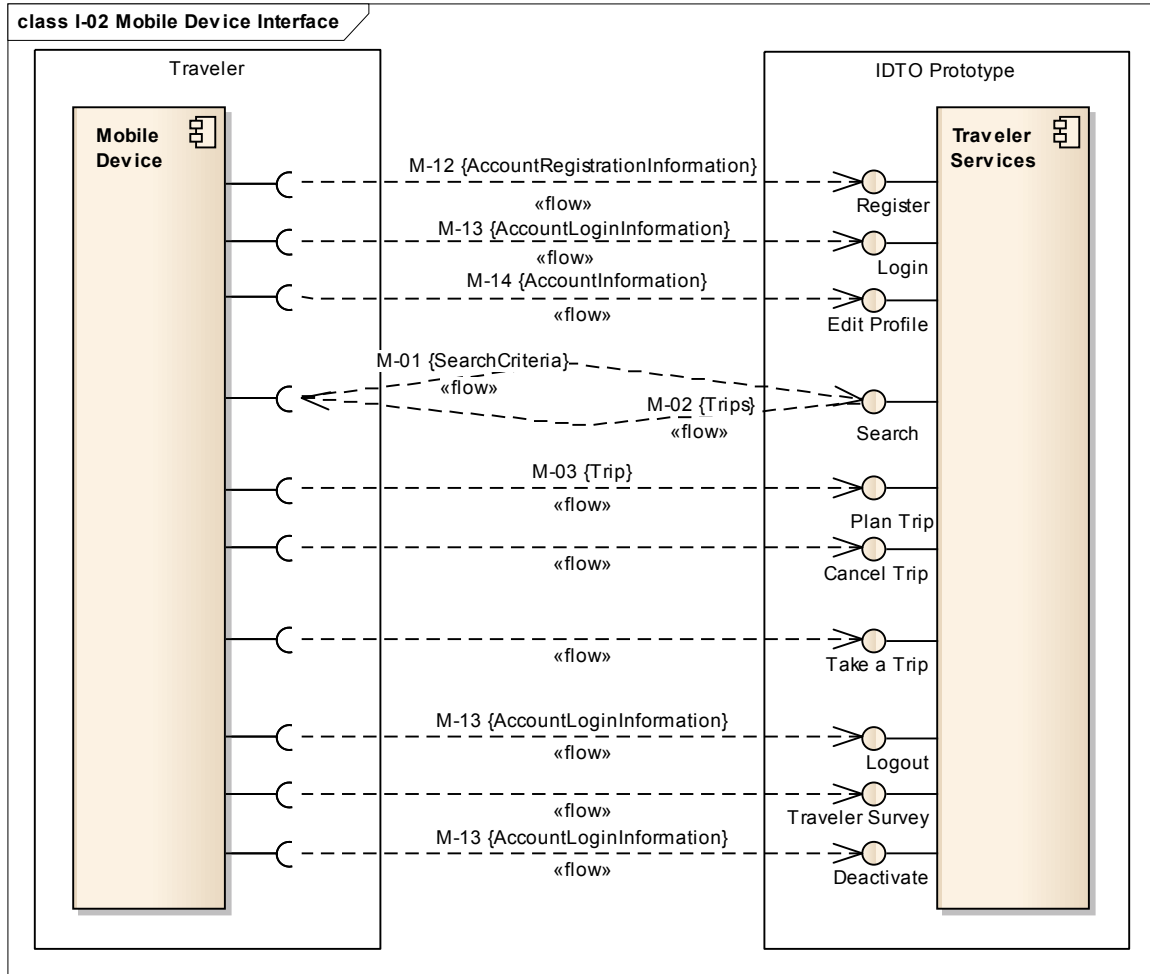
The interface between a traveler and the IDTO Prototype consists of six distinct information exchanges, listed in Figure 5-1, all through an application running on the traveler's mobile device.

The traveler must first create an IDTO user account and corresponding profile of trip preferences. The user account enables user authentication, while the preferences are used by the IDTO Prototype to facilitate trip scheduling for the traveler. This capability is referred to as Account Management and is supported via both the Traveler Smartphone APP and the Traveler Portal.

Once an account is established, the traveler is able to explore and schedule trips, enabled via both the Smartphone App and the Web Portal. Route searches from the traveler result in trip options presented back to the traveler. The traveler selects a trip which is saved and subsequently monitored within the IDTO Prototype. A critical design outcome is such that a traveler must be validated as being on the designated incoming bus in order for T-CONNECT transfers to be protected.

Once a trip is scheduled, the IDTO Prototype begins monitoring, notifying the traveler when the scheduled trip is about start, tracking the traveler through the trip via the traveler's mobile device, and providing feedback to the traveler, especially on the status of any T-Connect enabled legs of the trip.

Similar data exchanges occur between the Traveler Services component and a standard web browser, with the possible exception of location services provided back to the IDTO Prototype. Note that in either case, the traveler's mobile device or through a web browser, connectivity to the IDTO prototype must be maintained throughout an active trip for active T-Connect status to be provided.



Source: Battelle

Figure 5-1. I-02 Mobile Device Interface

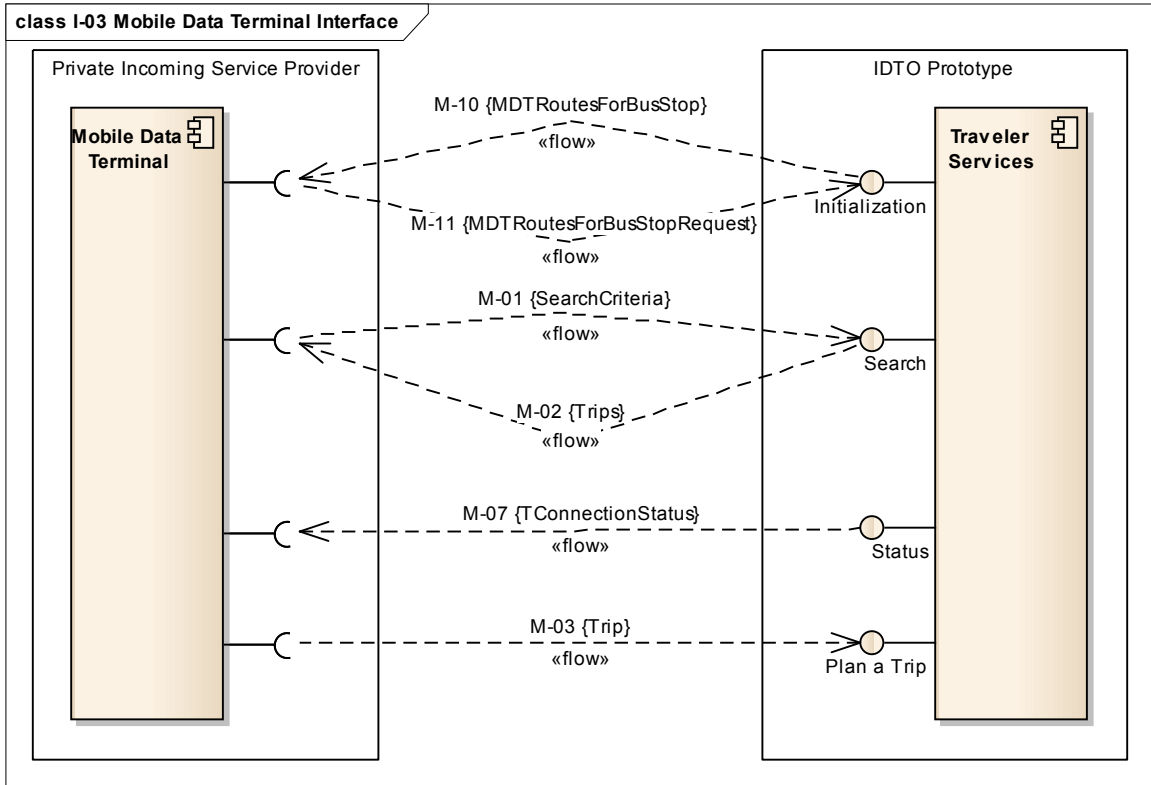
5.2 Private Incoming Service Provider

The Phase I implementation of the interface with a Private Incoming Fixed Service Provider is mediated entirely through a Mobile Data Terminal deployed in the Provider's vehicles, allowing significantly more functionality to be implemented than with the other incoming, fixed service providers. Many of the features provided to the driver are simplified trip scheduling features, allowing the driver to readily schedule trips on behalf of riders and effectively give the driver the ability to give connection protection to T-Connect Provider transfers for those riders. See Figure 5-2.

The Mobile Data Terminal interfaces with many of the same functions as the traveler's mobile device, however with a much simplified interface, limited to scheduling trips specific to the vehicle's known route, and only for immediately upcoming transfers to a T-Connect provider.

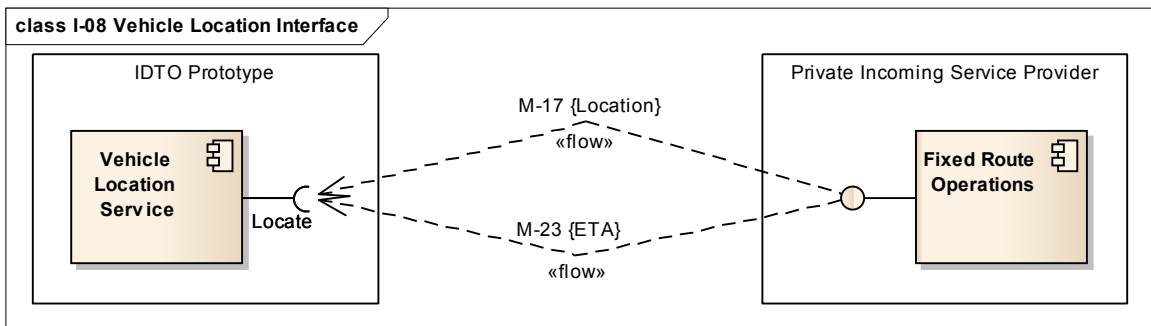
The Mobile Data Terminal does not directly allow the creation of a T-Connect request for a trip. Instead, by providing incoming vehicle data to the IDTO Prototype's Vehicle Location Service (see

Figure 5-3), the IDTO Prototype is able to make the determination if a T-Connect is warranted, and make the request on the vehicle's behalf.



Source: Battelle

Figure 5-2. I-03 Mobile Data Terminal Interface



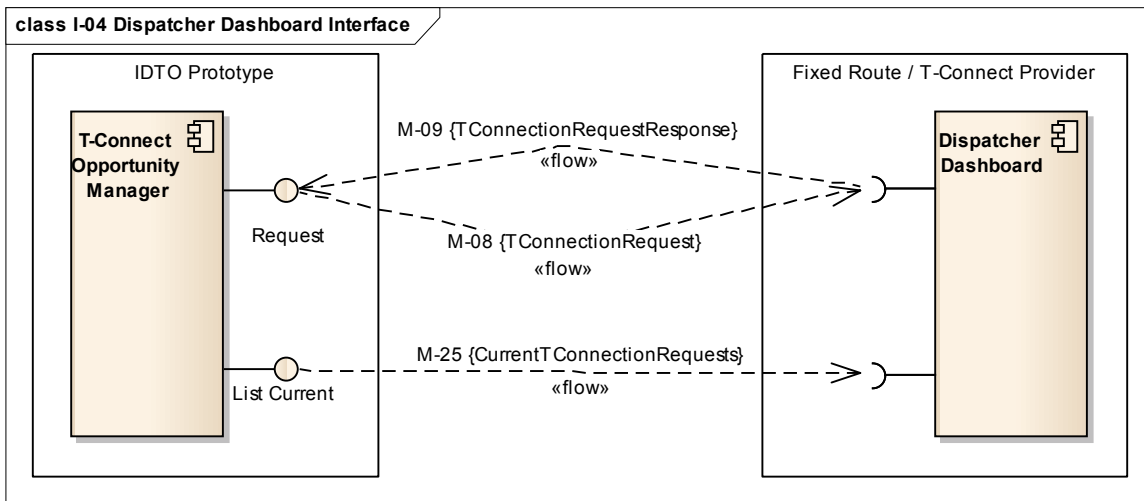
Source: Battelle

Figure 5-3. I-08 Vehicle Location Interface

5.3 Fixed Route / T-Connect Provider

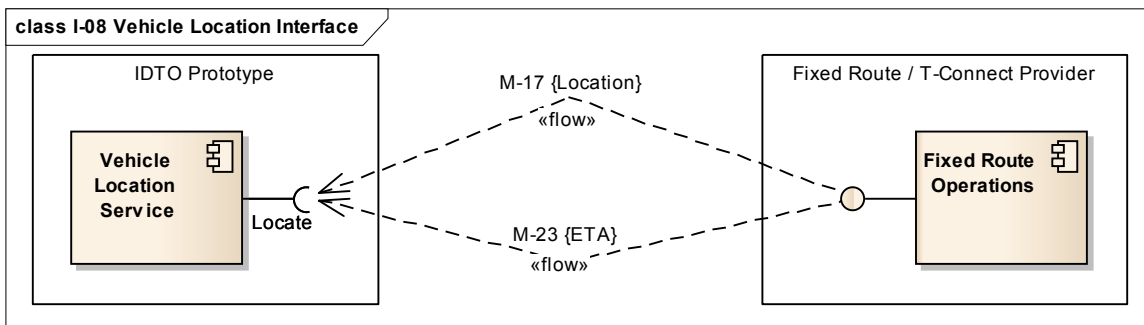
The IDTO Prototype Application interfaces with a Fixed Route provider that provides T-Connect functionality via several interfaces. The following figure identifies the specific interfaces and functionality associated with these interfaces. As shown, Bus Stop ETA from Fixed Route Operations software (e.g. Trapeze Novus ITS product) is provided to the IDTO Prototype in the form of GTFS-Realtime. The Vehicle Location Service within the Prototype persists this information in the ETA database. The T-CONNECT Opportunity Manager Service, which monitors traveler transfer requests along with incoming and outgoing buses, uses this ETA data to determine if and when a T-CONNECT is required. If a T-CONNECT is warranted, the IDTO Notification Service communicates the T-CONNECT request to the Dispatcher Dashboard, where the T-CONNECT is either accepted or rejected (see Figure 5-4). A return notification from the Dispatcher Terminal to the IDTO Notification Service completes this transaction.

In parallel with these exchanges, GTFS data is also being made available to the Route Provider (e.g. Open Trip Planner) to support traveler’s trip planning needs.



Source: Battelle

Figure 5-4. I-04 Dispatcher Dashboard Interface

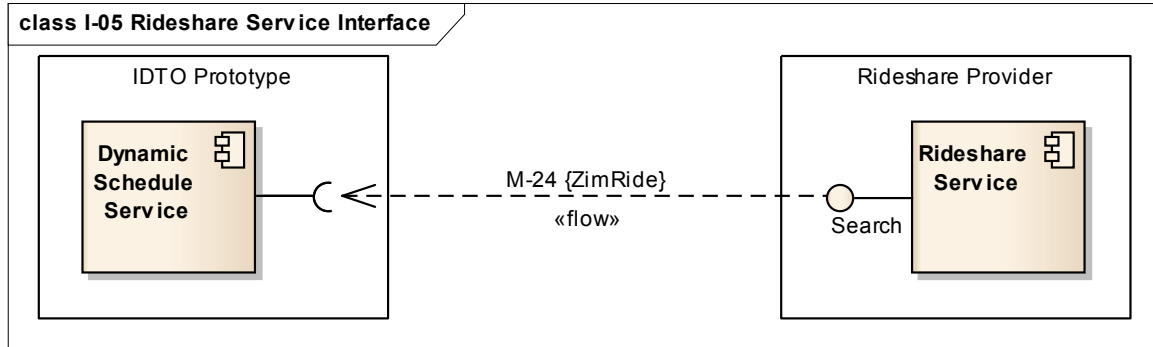


Source: Battelle

Figure 5-5. I-08 Vehicle Location Interface

5.4 Rideshare Provider

The IDTO Prototype implements a single Rideshare Provider interface. Trip data from the Rideshare Provider is pulled from the provider's interface by the Dynamic Schedule Service within IDTO Prototype in a format and mechanism that can be found in **Error! Reference source not found.** (see Figure 5-6). The IDTO Prototype subsequently makes this available for travelers to select as possible route alternatives.

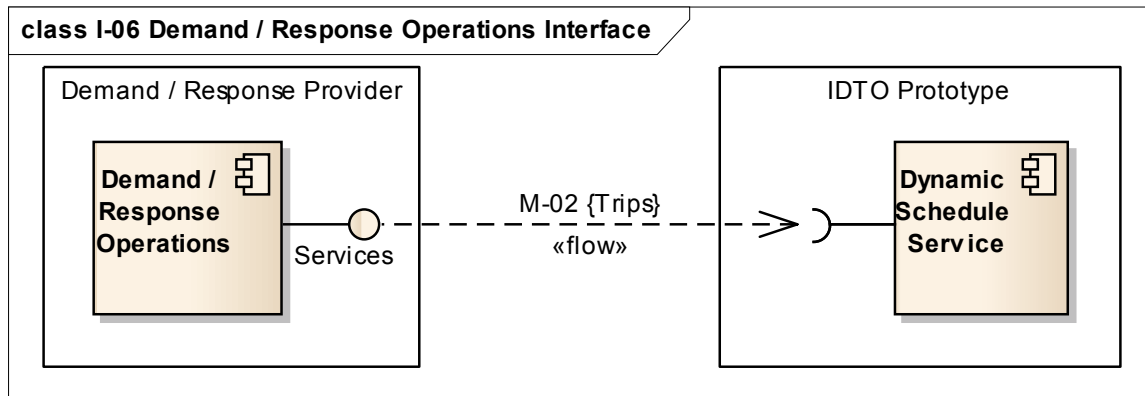


Source: Battelle

Figure 5-6. I-05 Rideshare Service Interface

5.5 Demand / Response Provider

The IDTO Prototype Application will implement a single interface with the Demand / Response provider. Figure 5-7 identifies the specific interface, and the functionality associated with this interface. As shown, Schedule data from provider's operations is provided to the IDTO Prototype in an as yet unspecified format. The Trip Collection Service within the Prototype then transforms this information to provide the traveler with possible trip options.



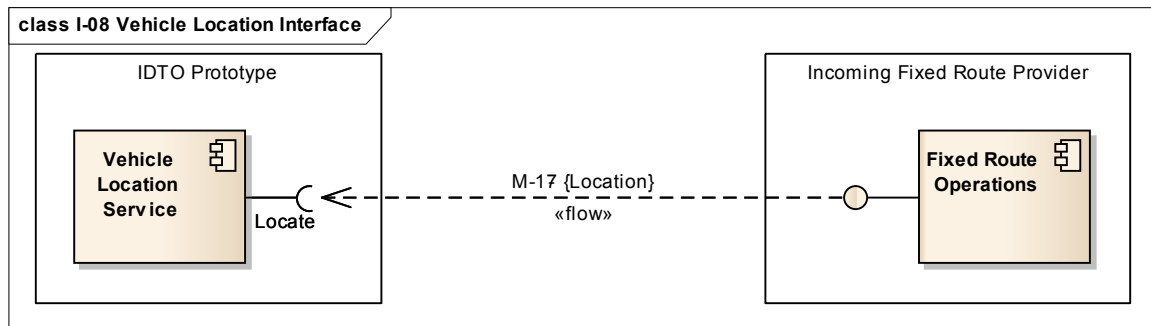
Source: Battelle

Figure 5-7. I-06 Demand / Response Operations Interface

5.6 Incoming Fixed Route Provider

The IDTO Prototype interfaces with an Incoming Fixed Route Provider slightly differently from the way it interfaces with a T-Connect Provider. The Incoming Provider need only provide the IDTO Prototype with current vehicle location data to allow the IDTO Prototype to judge the certainty a traveler has of making a subsequent connection to a T-Connect Provider, and requesting transfer protection in the form of a T-Connect request if that transfer is deemed to be threatened. As shown in Figure 5-8, Bus Location and Bus Stop ETA are provided to the IDTO Prototype from RESTful Web service calls to the Vehicle Location Service. The Vehicle Location Service persists this information in the IDTO Prototype where it is subsequently used by the T-CONNECT Opportunity Manager Service to monitor traveler transfers between buses, issuing transfer protection requests as necessary. Bus Stop ETA data is provided dynamically to the IDTO Prototype rather than through GTFS data to better judge when a protection request is justified.

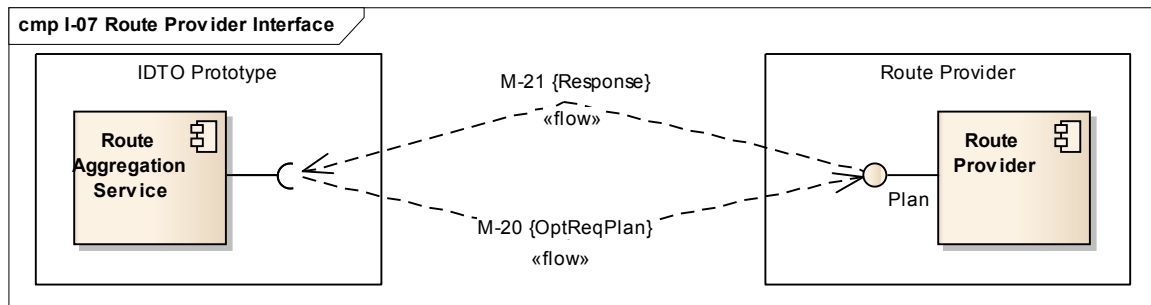
GTFS data is also being made available to the Route Provider (e.g. Open Trip Planner) to support traveler's trip planning needs.



Source: Battelle

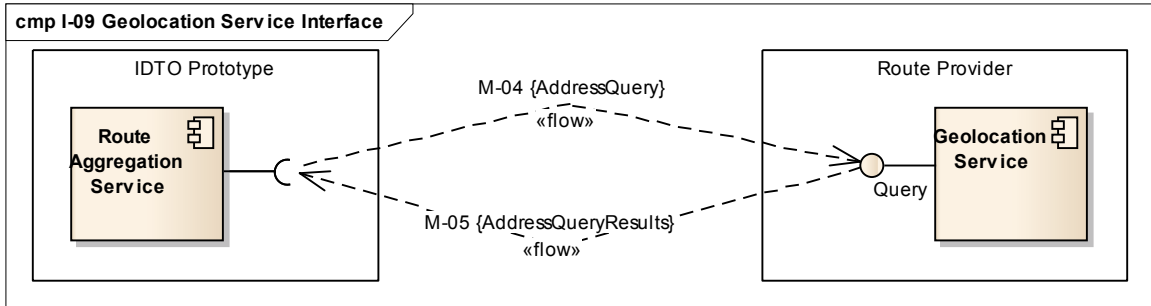
Figure 5-8. I-08 Vehicle Location Interface

5.7 Route Provider



Source: Battelle

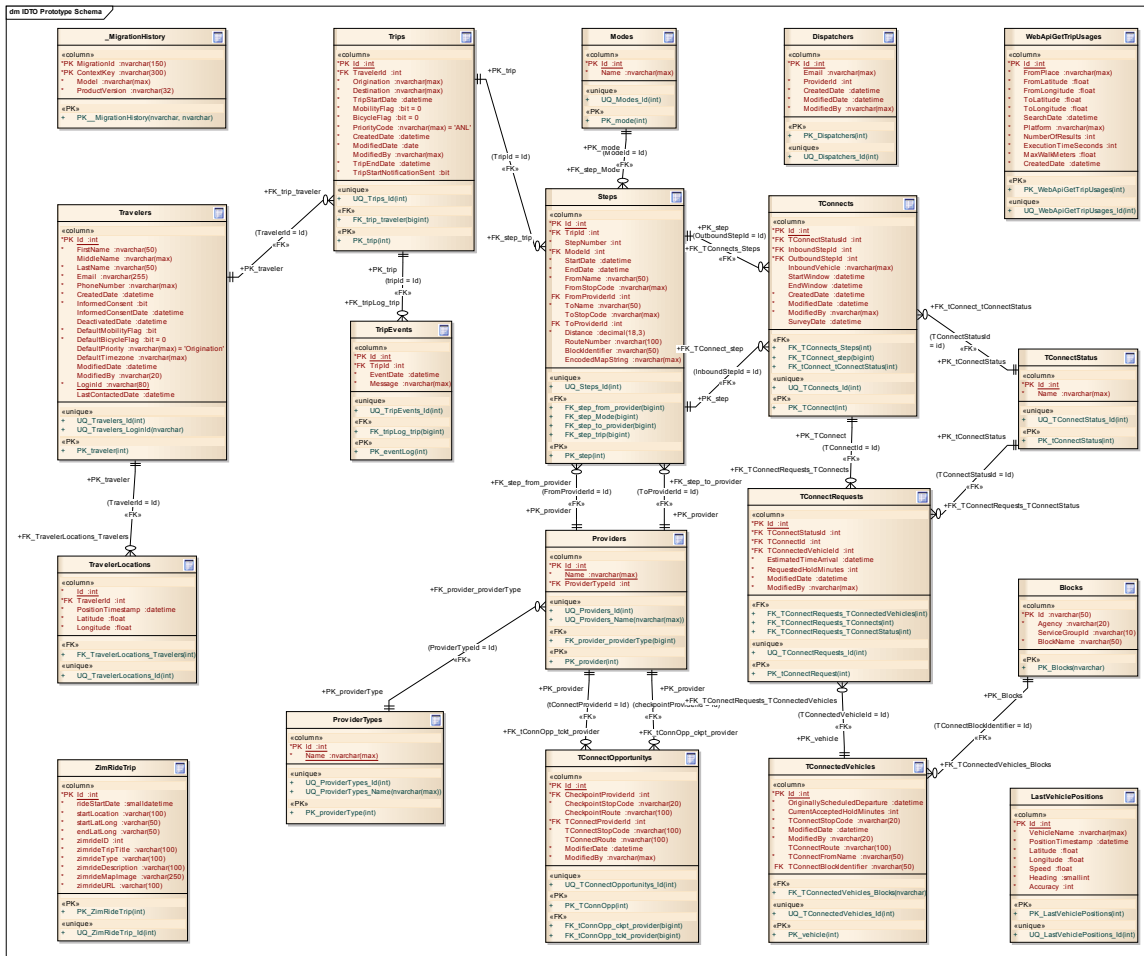
Figure 5-9. I-07 Route Provider Interface



Source: Battelle

Figure 5-10. I-09 Geolocation Service Interface

Chapter 6 IDTO Data Model and Schema



Source: Battelle

Figure 6-1. IDTO Prototype Schema

6.1 **_MigrationHistory**

Migration histories are documented in this table.

Table 6-1. _MigrationHistory Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	MigrationId	nvarchar	True	False	150	Primary key
True	ContextKey	nvarchar	True	False	300	Context key, part of primary key
False	Model	nvarchar(max)	True	False		Model name
False	ProductVersion	nvarchar	True	False	32	Product version

Source: Battelle

Table 6-2. _MigrationHistory Table Column Constraints

Name	Type	Columns
PK_ <u>_MigrationHistory</u>	Public	MigrationId ContextKey

Source: Battelle

Table 6-3. _MigrationHistory Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.2 Blocks

For identification of groups of vehicles capable of TConnections.

Table 6-4. Blocks Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	nvarchar	True	False	50	Primary key
False	Agency	nvarchar	True	False	20	Agency
False	ServiceGroupId	nvarchar	False	False	10	Service group id
False	BlockName	nvarchar	True	False	50	Block name

Source: Battelle

Table 6-5. Blocks Table Column Constraints

Name	Type	Columns
PK_Blocks	Public	Id

Source: Battelle

Table 6-6. Blocks Table Relationships

Columns	Association
(TConnectBlockIdentifier = Id)	0..* TConnectedVehicles.FK_TConnectedVehicles_Blocks
	1 Blocks.PK_Blocks

Source: Battelle

6.3 DispatcherLogins

Dispatcher identity created from initial registration. This table contains the users encrypted login credentials.

Table 6-7. DispatcherLogins Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	id	nvarchar	True	True	255	Primary key
False	_createdAt	datetimeoffset	True	False	3	Date and time offset of data creation.
False	_updatedAt	datetimeoffset	False	False	3	Date and time offset of data update.
False	_version	timestamp	True	False		Version timestamp
False	username	nvarchar(max)	False	False		Dispatcher's username
False	password	nvarchar(max)	False	False	0	Encrypted password
False	salt	nvarchar(max)	False	False	0	Salt used for the hashing algorithm used to encrypt the password

Source: Battelle

Table 6-8. DispatcherLogins Table Column Constraints

Name	Type	Columns
PK_login	Public	id
UQ_Logins_id	Public	id

Source: Battelle

Table 6-9. DispatcherLogins Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.4 Dispatchers

This table lists the identities of the users designated as dispatchers.

Table 6-10. Dispatchers Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	Email	nvarchar(max)	False	False		Dispatcher email
False	ProviderId	int	True	False		Foreign key into Provider table
False	CreatedDate	datetime	True	False		Date and time of data creation
False	ModifiedDate	datetime	True	False		Date and time of last data modification
False	ModifiedBy	nvarchar(max)	True	False		Data modified by this user

Source: Battelle

Table 6-11. Dispatchers Table Column Constraints

Name	Type	Columns
PK_Dispatchers	Public	Id
UQ_Dispatchers_Id	Public	Id

Source: Battelle

Table 6-12. Dispatchers Table Relationships

Columns	Association
None	

Source: Battelle

6.5 LastVehiclePositions

AVL data for TConnect vehicles, enabling IDTO to determine when a TConnect should be requested.

Table 6-13. LastVehiclePositions Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	VehicleName	nvarchar(max)	True	False		Vehicle identifier used by provider.
False	PositionTimestamp	datetime	True	False		Date and time that this position information was valid.
False	Latitude	float	True	False		Latitude of vehicle.
False	Longitude	float	True	False		Longitude of vehicle.
False	Speed	float	True	False		Vehicle speed.
False	Heading	smallint	True	False		Direction of vehicle travel.
False	Accuracy	int	True	False		Accuracy of position data.

Source: Battelle

Table 6-14. LastVehiclePositions Table Column Constraints

Name	Type	Columns
PK_LastVehiclePositions	Public	Id
UQ_LastVehiclePositions_Id	Public	Id

Source: Battelle

Table 6-15. LastVehiclePositions Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.6 Logins

Traveler identity created from initial registration. This table contains the users encrypted login credentials.

Table 6-16. Logins Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	id	nvarchar	True	True	255	Primary key
False	_createdAt	datetimeoffset	True	False	3	Date and time offset of data creation.
False	_updatedAt	datetimeoffset	False	False	3	Date and time offset of data update.
False	_version	timestamp	True	False		Version timestamp
False	username	nvarchar(max)	False	False		Traveler's username
False	password	nvarchar(max)	False	False		Encrypted password
False	salt	nvarchar(max)	False	False		Salt used for the hashing algorithm used to encrypt the password

Source: Battelle

Table 6-17. Logins Table Column Constraints

Name	Type	Columns
UQ_Logins_id	Public	id
PK_login	Public	id

Source: Battelle

Table 6-18. Logins Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.7 Modes

Lookup table for available modes of travel, i.e. fixed-route, ride share, etc.

Table 6-19. Modes Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	Name	nvarchar(max)	True	False		Description of this mode.

Source: Battelle

Table 6-20. Modes Table Column Constraints

Name	Type	Columns
UQ_Modes_Id	Public	Id
PK_mode	Public	Id

Source: Battelle

Table 6-21. Modes Table Relationships

Columns	Association
(ModeId = Id)	0..* Steps.FK_step_Mode
	1 Modes.PK_mode

Source: Battelle

6.8 Providers

A provider of transportation services, e.g. COTA, ZimRide, CABS, etc.

Table 6-22. Providers Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	Name	nvarchar(max)	True	True		The name of a provider.
False	ProviderTypeId	int	True	False		Foreign key into the providerType table. Indicates the kind of services this provider supplies.

Source: Battelle

Table 6-23. Providers Table Column Constraints

Name	Type	Columns
UQ_Providers_Id	Public	Id
UQ_Providers_Name	Public	Name
FK_provider_providerType	Public	ProviderTypeId
PK_provider	Public	Id

Source: Battelle

Table 6-24. Providers Table Relationships

Columns	Association
(CheckpointProviderId = Id)	0..* TConnectOpportunitys.FK_tConnOpp_ckpt_provider 1 Providers.PK_provider
(FromProviderId = Id)	0..* Steps.FK_step_from_provider 1 Providers.PK_provider
(ProviderTypeId = Id)	0..* Providers.FK_provider_providerType 1 ProviderTypes.PK_providerType
(TConnectProviderId = Id)	0..* TConnectOpportunitys.FK_tConnOpp_tckt_provider 1 Providers.PK_provider
(ToProviderId = Id)	0..* Steps.FK_step_to_provider 1 Providers.PK_provider

Source: Battelle

6.9 ProviderTypes

This table defines the valid types of service providers in the IDTO prototype:

- Fixed Route / T-Connect Provider
- Incoming Fixed Route Provider
- Private Incoming Service Provider
- Demand / Response Provider
- Rideshare Provider
- Pedestrian
- Bicycle

Table 6-25. ProviderTypes Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	Name	nvarchar(max)	True	True		The description of a provider type.

Source: Battelle

Table 6-26. ProviderTypes Table Column Constraints

Name	Type	Columns
UQ_ProviderTypes_Id	Public	Id
UQ_ProviderTypes_Name	Public	Name
PK_providerType	Public	Id

Source: Battelle

Table 6-27. ProviderTypes Table Relationships

Columns	Association
(ProviderTypeId = Id)	0..* Providers.FK_provider_providerType 1 ProviderTypes.PK_providerType

Source: Battelle

6.10 Registration

Table 6-28. Registration Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	id	nvarchar	True	True	255	
False	_createdAt	datetimeoffset	True	False	3	Date and time offset of data creation.
False	_updatedAt	datetimeoffset	True	False	3	Date and time offset of data update.
False	_version	timestamp	True	False		Version timestamp
False	handle	nvarchar(max)	False	False		Registration handle

Source: Battelle

Table 6-29. Registration Table Column Constraints

Name	Type	Columns
PK_Registration	Public	id
UQ_Registration_id	Public	id

Source: Battelle

Table 6-30. Registration Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.11 Steps

Part of a scheduled trip. A series of steps connect a scheduled trip's origination to the trip's destination.

Table 6-31. Steps Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	Tripld	int	True	False		Foreign key into trip table. Identifies which scheduled trip this step belongs to.
False	StepNumber	int	True	False		Step sequence number
False	Modeld	int	True	False		Mode of transportation
False	StartDate	datetime	True	False		Step start date and time
False	EndDate	datetime	True	False		step end date and time
False	FromName	nvarchar	True	False	50	From Name given by the Route Provider
False	FromStopCode	nvarchar(max)	False	False		From Stop Code given by the Route Provider
False	FromProviderId	int	False	False		Foreign key into the Provider table. Indicates the provider for this step.
False	ToName	nvarchar	True	False	50	To Name given by the Route Provider
False	ToStopCode	nvarchar(max)	False	False		To Stop Code given by the Route Provider
False	ToProviderId	int	False	False		Foreign key into the Provider table. This is the provider for the Stop Code
False	Distance	decimal	True	False		Distance for this step
False	RouteNumber	nvarchar	False	False	100	Route number for this step
False	BlockIdentifier	nvarchar	False	False	50	
False	EncodedMapString	nvarchar(max)	False	False		

Source: Battelle

Table 6-32. Steps Table Column Constraints

Name	Type	Columns
UQ_Steps_Id	Public	Id
FK_step_from_provider	Public	FromProviderId
FK_step_Mode	Public	ModeId
FK_step_to_provider	Public	ToProviderId
FK_step_trip	Public	TripId
PK_step	Public	Id

Source: Battelle

Table 6-33. Steps Table Relationships

Columns	Association
(OutboundStepId = Id)	0..* TConnects.FK_TConnects_Steps 1 Steps.PK_step
(FromProviderId = Id)	0..* Steps.FK_step_from_provider 1 Providers.PK_provider
(InboundStepId = Id)	0..* TConnects.FK_TConnect_step 1 Steps.PK_step
(ModeId = Id)	0..* Steps.FK_step_Mode 1 Modes.PK_mode
(ToProviderId = Id)	0..* Steps.FK_step_to_provider 1 Providers.PK_provider
(TripId = Id)	0..* Steps.FK_step_trip 1 Trips.PK_trip

Source: Battelle

6.12 TConnectedVehicles

This table maintains the data for the different vehicles in which the IDTO system will be monitoring.

Table 6-34. TConnectedVehicles Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	OriginallyScheduledDeparture	datetime	True	False		Date and time of scheduled departure
False	CurrentAcceptedHoldMinutes	int	True	False		Allowable duration for a vehicle hold.
False	TConnectStopCode	nvarchar	True	False	20	
False	ModifiedDate	datetime	True	False		Date and time of last data modification.
False	ModifiedBy	nvarchar	True	False	20	Last modifier of this data.
False	TConnectRoute	nvarchar	False	False	100	Identifying description of the route this vehicle is running.
False	TConnectFromName	nvarchar	True	False	50	Name of the source of the TConnect request.
False	TConnectBlockIdentifier	nvarchar	False	False	50	

Source: Battelle

Table 6-35. TConnectedVehicles Table Column Constraints

Name	Type	Columns
FK_TConnectedVehicles_Blocks	Public	TConnectBlockIdentifier
UQ_TConnectedVehicles_Id	Public	Id
PK_vehicle	Public	Id

Source: Battelle

Table 6-36. TConnectedVehicles Table Relationships

Columns	Association
(TConnectedVehicleId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnectedVehicles 1 TConnectedVehicles.PK_vehicle
(TConnectBlockIdentifier = Id)	0..* TConnectedVehicles.FK_TConnectedVehicles_Blocks 1 Blocks.PK_Blocks

Source: Battelle

6.13 TConnectOpportunities

This table defines every potential sequence of locations that can potentially trigger a T-Connect request. Any active trip in which one of these sequences appears is monitored closely by the T-Connect Opportunity Manager.

This table is pre-populated with all inbound checkpoint location / T-Connect departure location pairs. The provider type of the departure location must be a T-Connect provider, while the provider type at the inbound location must be an Inbound provider.

Table 6-37. TConnectOpportunities Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	CheckpointProviderId	int	True	False		Foreign key into provider table.
False	CheckpointStopCode	nvarchar	True	False	20	Identifying description of the checkpoint stop.
False	CheckpointRoute	nvarchar	False	False	100	Route identifier for the inbound vehicle.
False	TConnectProviderId	int	True	False		Foreign key into the provider table. Indicates the provider associated with this tConnect opportunity.
False	TConnectStopCode	nvarchar	True	False	100	The location of the tConnect point of origin.
False	TConnectRoute	nvarchar	False	False	100	Route of the outbound vehicle
False	ModifierDate	datetime	True	False		Date this data was last modified.
False	ModifiedBy	nvarchar(max)	True	False		Last modifier of this data.

Source: Battelle

Table 6-38. TConnectOpportunities Table Column Constraints

Name	Type	Columns
UQ_TConnectOpportunitys_Id	Public	Id
PK_TConnOpp	Public	Id
FK_tConnOpp_ckpt_provider	Public	CheckpointProviderId
FK_tConnOpp_tckt_provider	Public	TConnectProviderId

Source: Battelle

Table 6-39. TConnectOpportunities Table Relationships

Columns	Association
(CheckpointProviderId = Id)	0..* TConnectOpportunitys.FK_tConnOpp_ckpt_provider 1 Providers.PK_provider
(TConnectProviderId = Id)	0..* TConnectOpportunitys.FK_tConnOpp_tckt_provider 1 Providers.PK_provider

Source: Battelle

6.14 TConnectRequests

This table defines the connection protection requests which have been issued by the system. Records in this table will be used to track the status of the requests.

Table 6-40. TConnectRequests Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	TConnectStatusId	int	True	False		Foreign key into teh TConnectStatus table.
False	TConnectId	int	True	False		Foreign key into the TConnect table.
False	TConnectedVehicleId	int	True	False		Foreign key into the TConnectVehicle table.
False	EstimatedTimeArrival	datetime	True	False		Estimated time of arrival
False	RequestedHoldMinutes	int	True	False		Requested hold time
False	ModifiedDate	datetime	True	False		Date this data was last modified.
False	ModifiedBy	nvarchar(max)	True	False		Last modifier of this data.

Source: Battelle

Table 6-41. TConnectRequests Table Column Constraints

Name	Type	Columns
FK_TConnectRequests_TConnectedVehicles	Public	TConnectedVehicleId
FK_TConnectRequests_TConnects	Public	TConnectId
FK_TConnectRequests_TConnectStatus	Public	TConnectStatusId
UQ_TConnectRequests_Id	Public	Id
PK_tConnectRequest	Public	Id

Source: Battelle

Table 6-42. TConnectRequests Table Relationships

Columns	Association
(TConnectId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnects 1 TConnects.PK_TConnect
(TConnectStatusId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnectStatus 1 TConnectStatus.PK_tConnectStatus
(TConnectedVehicleId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnectedVehicles 1 TConnectedVehicles.PK_vehicle

Source: Battelle

6.15 DispatcherLogins

This table logs the history of all T-Connect requests generated in the IDTO Prototype.

Table 6-43. TConnects Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	TConnectStatusId	int	True	False	0	Foreign key into the tConnectStatus table. The status of the request, one of "Pending", "Accepted" or "Rejected".
False	InboundStepId	int	True	False		Foreign key into the step table.
False	OutboundStepId	int	True	False		Foreign key into the step table.
False	InboundVehicle	nvarchar(max)	False	False		Vehicle identifier for the Inbound Vehicle
False	StartWindow	datetime	False	False		Start of ETA window where a t-Connect Request will be issued.
False	EndWindow	datetime	False	False		End of ETA window where a t-Connect request will be issued.
False	CreatedDate	datetime	True	False		Date this data was created.
False	ModifiedDate	datetime	True	False		Date this data was last modified.
False	ModifiedBy	nvarchar(max)	True	False		Last modifier of this data.
False	SurveyDate	datetime	False	False		Date and time a survey was requested for this TConnect

Source: Battelle

Table 6-44. TConnects Table Column Constraints

Name	Type	Columns
FK_TConnects_Steps	Public	OutboundStepId
UQ_TConnects_Id	Public	Id
FK_TConnect_step	Public	InboundStepId
FK_tConnect_tConnectStatus	Public	TConnectStatusId
PK_TConnect	Public	Id

Source: Battelle

Table 6-45. TConnects Table Relationships

Columns	Association
(TConnectId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnects 1 TConnects.PK_TConnect
(OutboundStepId = Id)	0..* TConnects.FK_TConnects_Steps 1 Steps.PK_step
(InboundStepId = Id)	0..* TConnects.FK_TConnect_step 1 Steps.PK_step
(TConnectStatusId = Id)	0..* TConnects.FK_tConnect_tConnectStatus 1 TConnectStatus.PK_tConnectStatus

Source: Battelle

6.16 TConnectStatus

A lookup table which contains the possible status values for a TConnect.

Table 6-46. TConnectStatus Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	Name	nvarchar(max)	True	False		Description of this status code.

Source: Battelle

Table 6-47. TConnectStatus Table Column Constraints

Name	Type	Columns
UQ_TConnectStatus_Id	Public	Id
PK_tConnectStatus	Public	Id

Source: Battelle

Table 6-48. TConnectStatus Table Relationships

Columns	Association
(TConnectStatusId = Id)	0..* TConnectRequests.FK_TConnectRequests_TConnectStatus 1 TConnectStatus.PK_tConnectStatus
(TConnectStatusId = Id)	0..* TConnects.FK_tConnect_tConnectStatus 1 TConnectStatus.PK_tConnectStatus

Source: Battelle

6.17 TodoItem

Table 6-49. TodoItem Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	id	nvarchar	True	True	255	Primary key
False	_createdAt	datetimeoffset	True	False	3	Date and time offset of data creation.
False	_updatedAt	datetimeoffset	True	False	3	Date and time offset of data update.
False	_version	timestamp	True	False		Version timestamp
False	text	varchar(max)	False	False		The text of the todo item.
False	complete	bit	False	False		Completion flag.
False	handle	nvarchar(max)	False	False		

Source: Battelle

Table 6-50. TodoItem Table Column Constraints

Name	Type	Columns
PK_TodoItem	Public	id
UQ_TodoItem_id	Public	id

Source: Battelle

Table 6-51. TodoItem Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.18 TravelerLocations

This table is used to store the location of travelers. This data is used to calculate the estimated time of arrival for inbound vehicles and used in the algorithms for determining the need to issue a T-Connect Request.

Table 6-52. TravelerLocations Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
False	Id	int	True	True		Primary key
False	TravelerId	int	True	False		Foreign key into the Travelers table.
False	PositionTimestamp	datetime	True	False		Date and time this data was valid.
False	Latitude	float	True	False		The latitude of the traveler at a given moment in time.
False	Longitude	float	True	False		The longitude of the traveler at a given moment in time.

Source: Battelle

Table 6-53. TravelerLocations Table Column Constraints

Name	Type	Columns
FK_TravelerLocations_Travelers	Public	TravelerId
UQ_TravelerLocations_Id	Public	Id

Source: Battelle

Table 6-54. TravelerLocations Table Relationships

Columns	Association
(TravelerId = Id)	0..* TravelerLocations.FK_TravelerLocations_Travelers
	1 Travelers.PK_traveler

Source: Battelle

6.19 Travelers

The profile and account record for a traveler.

Table 6-55. Travelers Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	FirstName	nvarchar	True	False	50	Traveler's first name.
False	MiddleName	nvarchar(max)	False	False		Traveler's middle name or initial.
False	LastName	nvarchar	True	False	50	Traveler's last name.
False	Email	nvarchar	True	False	255	email address of the traveler, used as the unique account name for the IDTO Prototype.
False	PhoneNumber	nvarchar(max)	True	False		Traveler's phone number.
False	CreatedDate	datetime	True	False		GMT Date and time of account creation.
False	InformedConsent	bit	True	False		Indicates that the traveler has acknowledged the informed consent notice.
False	InformedConsentDate	datetime	False	False		Date and time the traveler acknowledged the informed consent notice.
False	DeactivatedDate	datetime	False	False		GMT Date and time account was deactivated, or null if active.
False	DefaultMobilityFlag	bit	True	False		Boolean value indicating whether the traveler prefers enhanced mobility services.
False	DefaultBicycleFlag	bit	True	False		Boolean value indicating whether the traveler prefers services that support transporting bicycles.

U.S. Department of Transportation, Research and Innovative Technology Administration
Intelligent Transportation Systems Joint Program Office

PK	Name	Type	Not Null	Unique	Len	Notes
False	DefaultPriority	nvarchar(max)	False	False		Preferred mode of scheduling trips, one of: "Destination": Use the trip time as the time of arrival at the trip's destination. "Origination": Use the trip time as the time of departure from the trip's point of origin.
False	DefaultTimezone	nvarchar(max)	False	False		Traveler's default time zone, as in 'US/Eastern', 'US/Central', 'US/Mountain', 'US/Arizona', 'US/Pacific', etc.
False	ModifiedDate	datetime	False	False		Date of last change to this data
False	ModifiedBy	nvarchar	False	False	20	The last modifier of this data
False	LoginId	nvarchar	True	True	80	Foreign key into the login table. Identifies the unique login associated with this traveler profile.
False	LastContactedDate	datetime	False	False		Date and time this traveler was issued any communications from the IDTO system.

Source: Battelle

Table 6-56. Travelers Table Column Constraints

Name	Type	Columns
UQ_Travelers_Id	Public	Id
UQ_Travelers_LoginId	Public	LoginId
PK_traveler	Public	Id

Source: Battelle

Table 6-57. Travelers Table Relationships

Columns	Association
(TravelerId = Id)	0..* TravelerLocations.FK_TravelerLocations_Travelers 1 Travelers.PK_traveler
(TravelerId = Id)	0..* Trips.FK_trip_traveler 1 Travelers.PK_traveler

Source: Battelle

6.20 TripEvents

This table records notable events for diagnostic and historical purposes.

Table 6-58. TripEvents Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	Tripld	int	True	False		Foreign key into trip table. Identifies the trip associated with this log event.
False	EventDate	datetime	True	False		Date and time stamp of event in UTC / Zulu time.
False	Message	nvarchar(max)	True	False		The text of the log message.

Source: Battelle

Table 6-59. TripEvents Table Column Constraints

Name	Type	Columns
UQ_TripEvents_Id	Public	Id
FK_tripLog_trip	Public	Tripld
PK_eventLog	Public	Id

Source: Battelle

Table 6-60. TripEvents Table Relationships

Columns	Association
(TripId = Id)	0..* TripEvents.FK_tripLog_trip
	1 Trips.PK_trip

Source: Battelle

6.21 Trips

A trip scheduled by a traveler, from a specified location to a specified destination, and a specific time, with parameters selected by the traveler.

Table 6-61. Trips Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key.
False	TravelerId	int	True	False		Foreign key into the traveler table. Identifies which traveler scheduled this trip.
False	Origination	nvarchar(max)	True	False		The traveler-specified starting location of the trip.
False	Destination	nvarchar(max)	True	False		The traveler-specified destination of the trip.
False	TripStartDate	datetime	True	False		The traveler specified trip time, may be an arrival or departure time, depending on the priorityCode. UTC, local timezone.
False	MobilityFlag	bit	True	False		Does this trip require mobility facilities?
False	BicycleFlag	bit	True	False		Does this trip require bicycle facilities?

PK	Name	Type	Not Null	Unique	Len	Notes
False	PriorityCode	nvarchar(max)	True	False		The scheduling priority for this trip. One of: <ul style="list-style-type: none"> • 1: Depart from point of origin no later than a specified time. • 2: "Arrive at destination no later than a specified time. • 3: Depart from point of origin no sooner than a specified time. • 4: Arrive at destination no sooner than a specified time.
False	CreatedDate	datetime	True	False		Date trip was scheduled
False	ModifiedDate	date	True	False		Date this data was last modified
False	ModifiedBy	nvarchar(max)	False	False		The last modifier of this data.
False	TripEndDate	datetime	True	False		Date and time of trip completion.
False	TripStartNotificationSent	bit	True	False		Indicates whether or not the traveler was notified at the start of this trip.

Source: Battelle

Table 6-62. Trips Table Column Constraints

Name	Type	Columns
UQ_Trips_Id	Public	Id
FK_trip_traveler	Public	TravelerId
PK_trip	Public	Id

Source: Battelle

Table 6-63. Trips Table Relationships

Columns	Association
(TravelerId = Id)	0..* Trips.FK_trip_traveler 1 Travelers.PK_traveler
(TripId = Id)	0..* Steps.FK_step_trip 1 Trips.PK_trip
(TripId = Id)	0..* TripEvents.FK_tripLog_trip 1 Trips.PK_trip

Source: Battelle

6.22 WebApiGetTripUsages

This table records the search hits executed by the IDTO mobile applications.

Table 6-64. WebApiGetTripUsages Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	FromPlace	nvarchar(max)	True	False		Name of the location where the trip is to start.
False	FromLatitude	float	True	False		Latitude of the trip's starting location.
False	FromLongitude	float	True	False		Longitude of the trip's starting location.
False	ToLatitude	float	True	False		Ending latitude of the trip.
False	ToLongitude	float	True	False		Ending longitude of the trip.
False	SearchDate	datetime	True	False		Date and time that the search was performed.
False	Platform	nvarchar(max)	True	False		Description of the users mobile device used to perform the search.
False	NumberOfResults	int	True	False		Number of hits returned to the user for this search criteria.
False	ExecutionTimeSeconds	int	True	False		Time it took to find the result set.
False	MaxWalkMeters	float	True	False		Longest distance the traveler is expected to walk for this trip.

U.S. Department of Transportation, Research and Innovative Technology Administration
Intelligent Transportation Systems Joint Program Office

PK	Name	Type	Not Null	Unique	Len	Notes
False	CreatedDate	datetime	True	False		Date and time of data creation.

Source: Battelle

Table 6-65. WebApiGetTripUsages Table Column Constraints

Name	Type	Columns
PK_WebApiGetTripUsages	Public	Id
UQ_WebApiGetTripUsages_Id	Public	Id

Source: Battelle

Table 6-66. WebApiGetTripUsages Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

Source: Battelle

6.23 ZimRideTrips

This table records the Zimride legs taken on any trip.

Table 6-67. ZimRideTrip Relational Database Table Columns

PK	Name	Type	Not Null	Unique	Len	Notes
True	Id	int	True	True		Primary key
False	rideStartDate	smalldatetime	True	False		date and time of the beginning of this leg.
False	startLocation	varchar	True	False	100	Description of starting point of the Zimride leg.
False	startLatLong	varchar	True	False	50	Starting position of the zimride leg.
False	endLatLong	varchar	True	False	50	Ending position of the zimride leg.
False	zimrideID	int	True	False		zimride identifier.
False	zimrideTripTitle	varchar	True	False	100	
False	zimrideType	varchar	True	False	100	

PK	Name	Type	Not Null	Unique	Len	Notes
False	zimrideDescription	varchar	True	False	100	
False	zimrideMapImage	varchar	True	False	250	
False	zimrideURL	varchar	True	False	100	

Source: Battelle

Table 6-68. ZimRideTrip Table Column Constraints

Name	Type	Columns
PK_ZimRideTrip	Public	Id
UQ_ZimRideTrip_Id	Public	Id

Source: Battelle

Table 6-69. ZimRideTrip Table Relationships

Columns	Association
None	No relations defined. Table included for consistency.

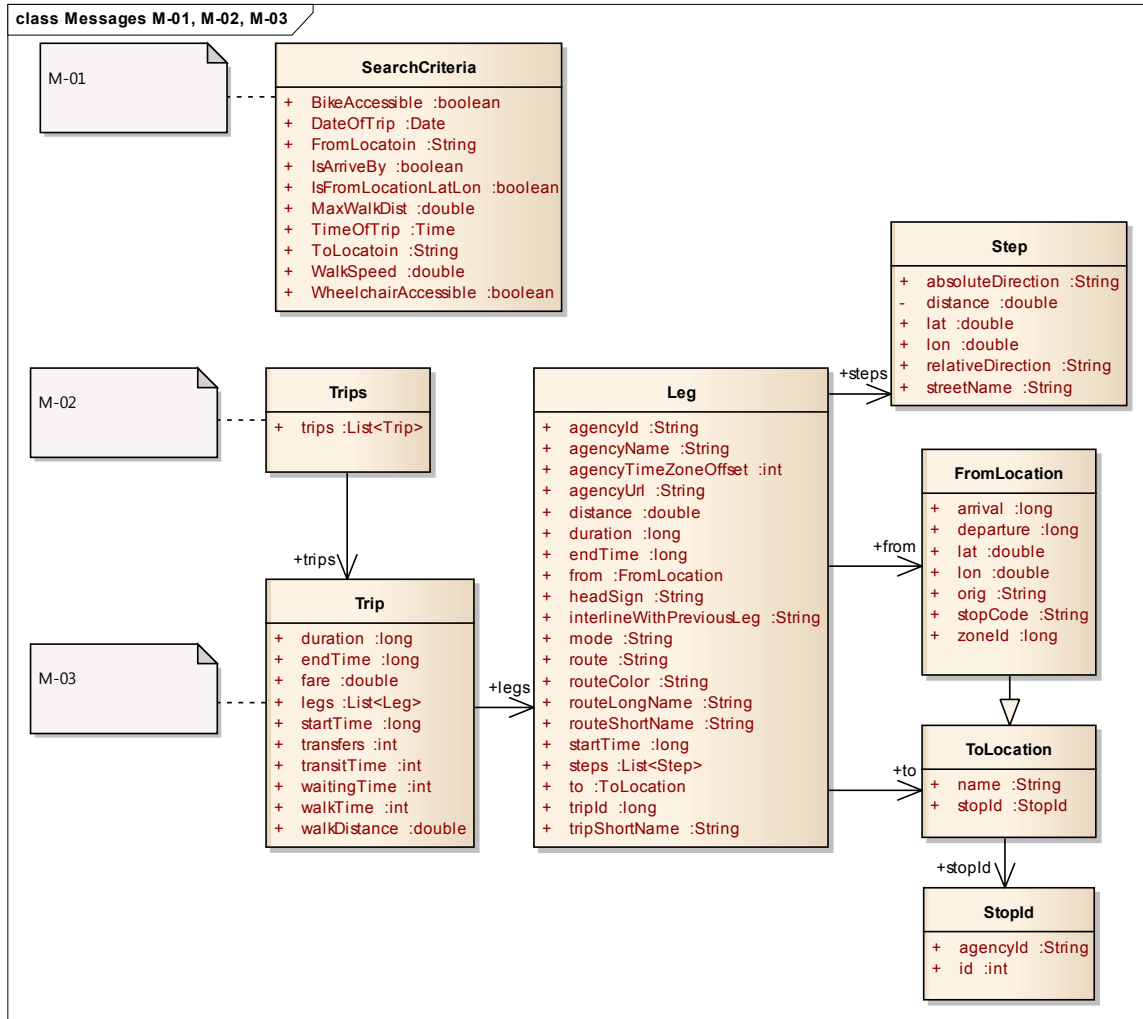
Source: Battelle

APPENDIX A. List of Abbreviations and Acronyms

<u>Acronym</u>	<u>Description</u>
AVA	Automatic Vehicle Annunciation
AVL	Automatic Vehicle Location
CAD	Computer-Aided Dispatch
ConOps	Concept of Operations
COTM	Contracting Officer's Task Manager
DMA	Dynamic Mobility Applications
DMS	Dynamic Message Sign
DOT	Department of Transportation
D-RIDE	Dynamic Ridesharing IDTO application
FHWA	Federal Highway Administration
GPS	Global Positioning System
HOT	High Occupancy Toll
HOV	High Occupancy Vehicle
IEEE	Institute of Electrical and Electronics Engineers
IDTO	Integrated Dynamic Transit Operations
ITS	Intelligent Transportation Systems
MDT	Mobile Data Terminal
MPO	Metropolitan Planning Organization
PDT	Project Development Team
PMP	Project Management Plan
RSA	Route and Schedule Adherence
SEMP	Systems Engineering Management Plan
SEP	Systems Engineering Process
SIRI	Service Interface for Real-time Information
SOW	Statement of Work
SRS	System Requirements Specifications
T-CONNECT	Transfer Connection IDTO application
TCP	Transfer Connection Protection
T-DISP	Transit Dispatch IDTO application

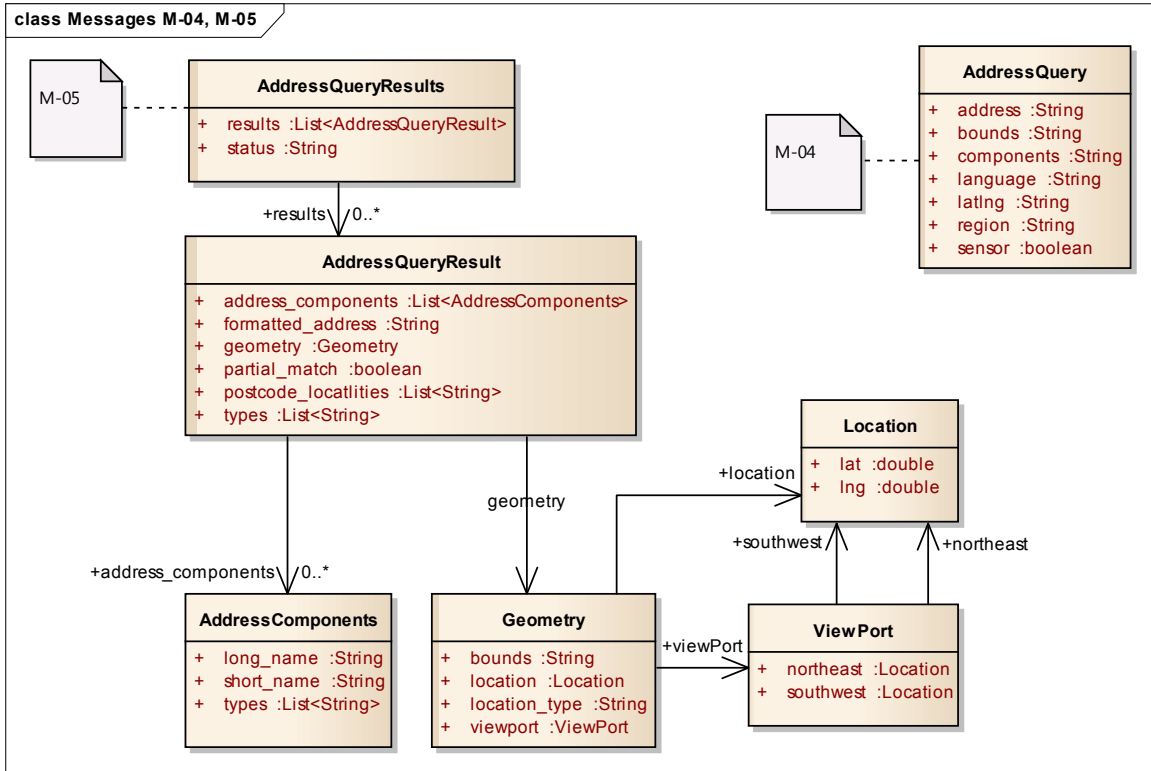
TMO	Transportation Management Organization
TRB	Transportation Research Board
US DOT	United States Department of Transportation

APPENDIX B. Interface Messages



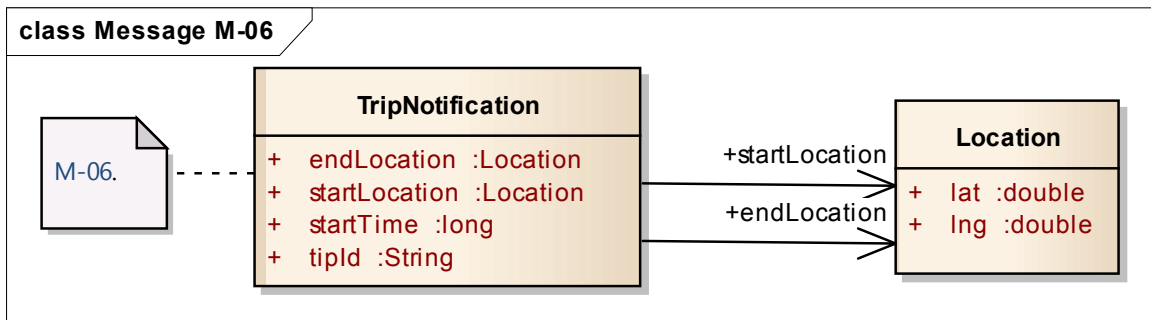
Source: Battelle

Figure B-1. Messages M-01, M-02, M-03



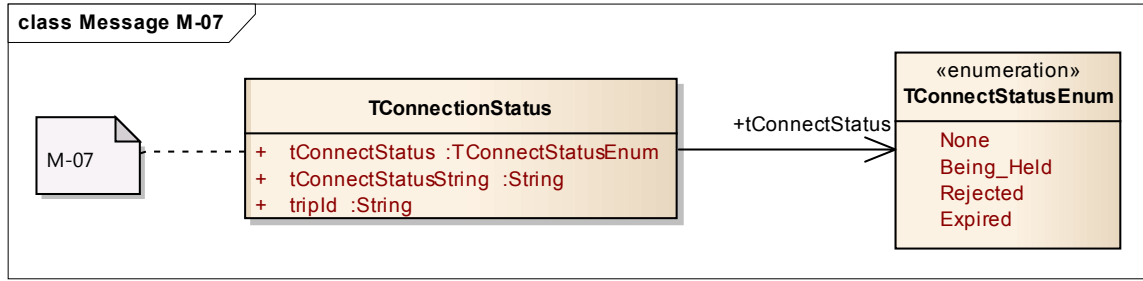
Source: Battelle

Figure B-2. Messages M-04, M-05



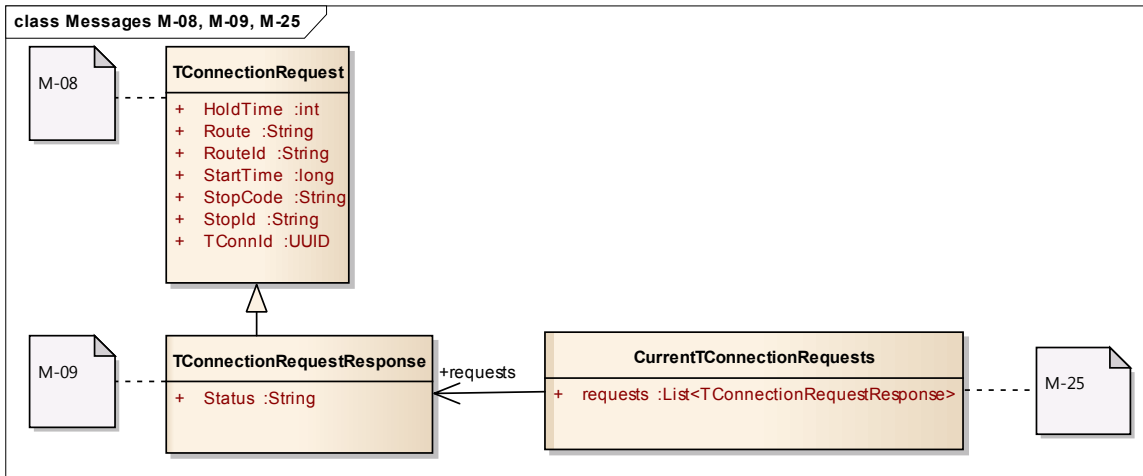
Source: Battelle

Figure B-3. Message M-06



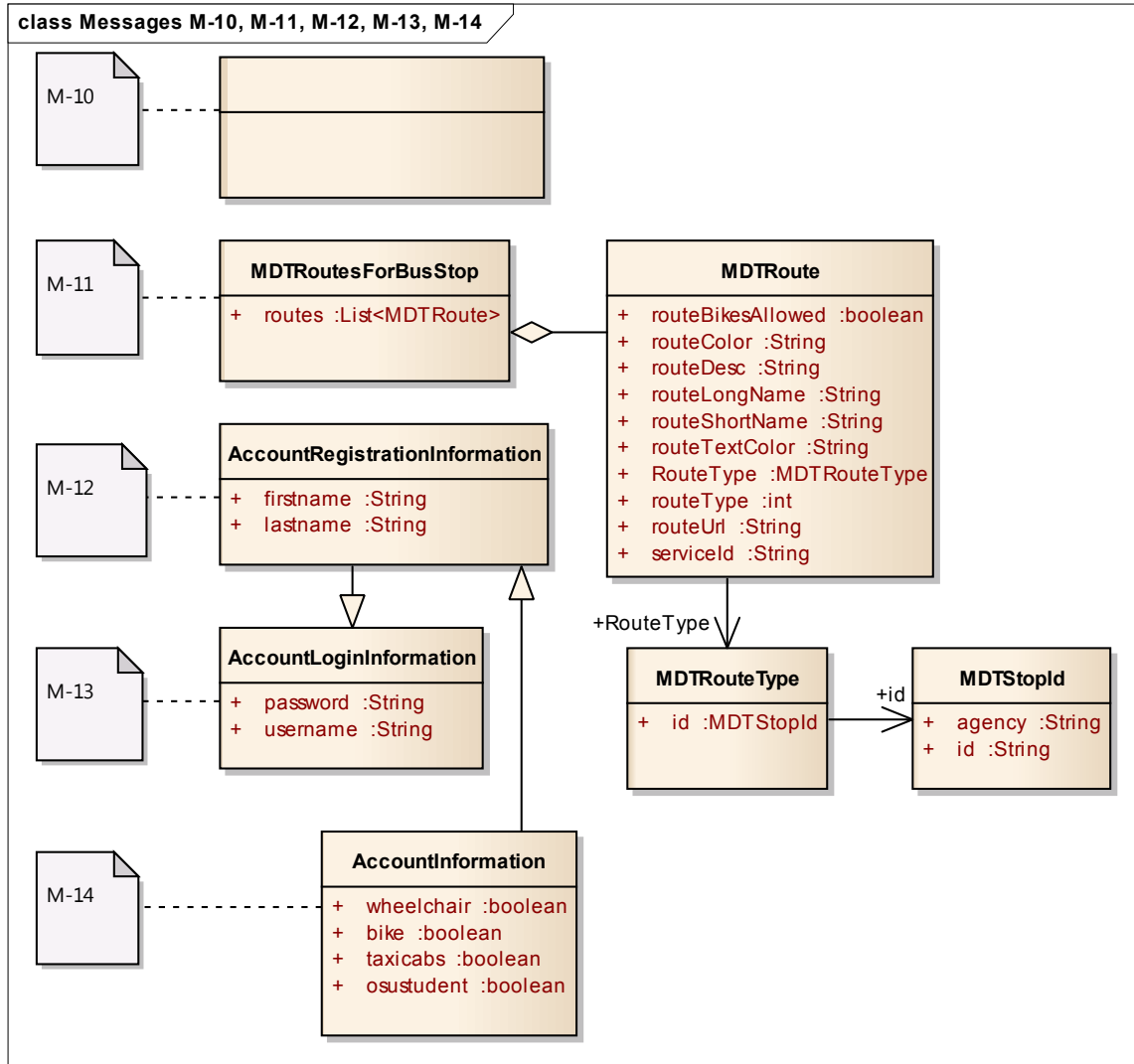
Source: Battelle

Figure B-4. Message M-07



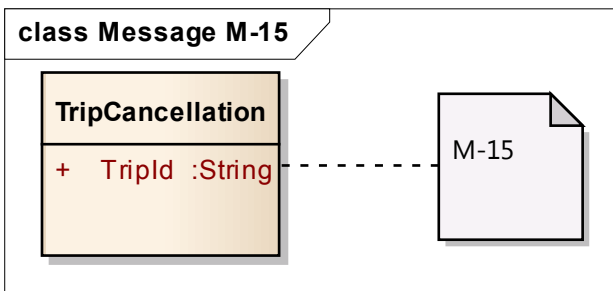
Source: Battelle

Figure B-5. Messages M-08, M-09, M-25



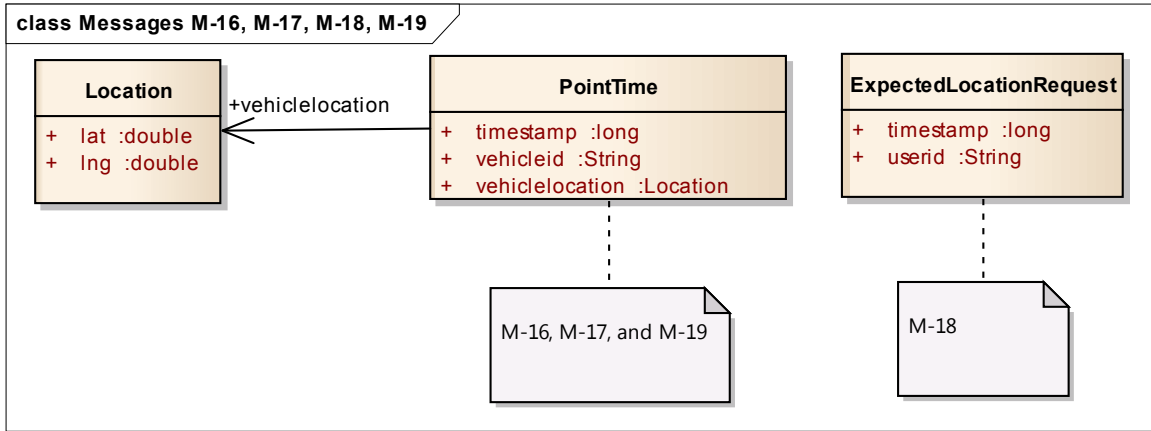
Source: Battelle

Figure B-6. Messages M-10, M-11, M-12, M-13, M-14



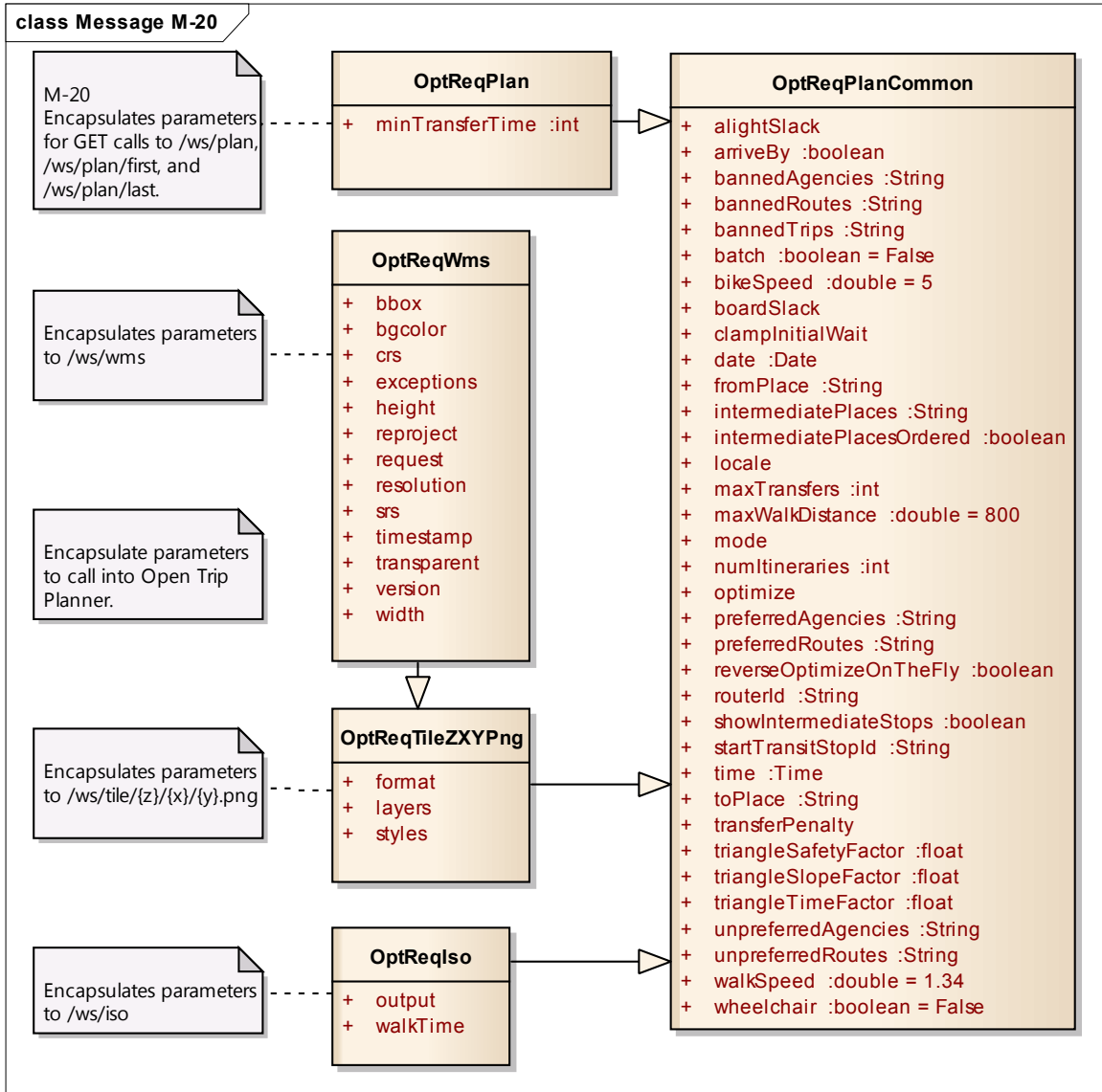
Source: Battelle

Figure B-7. Message M-15



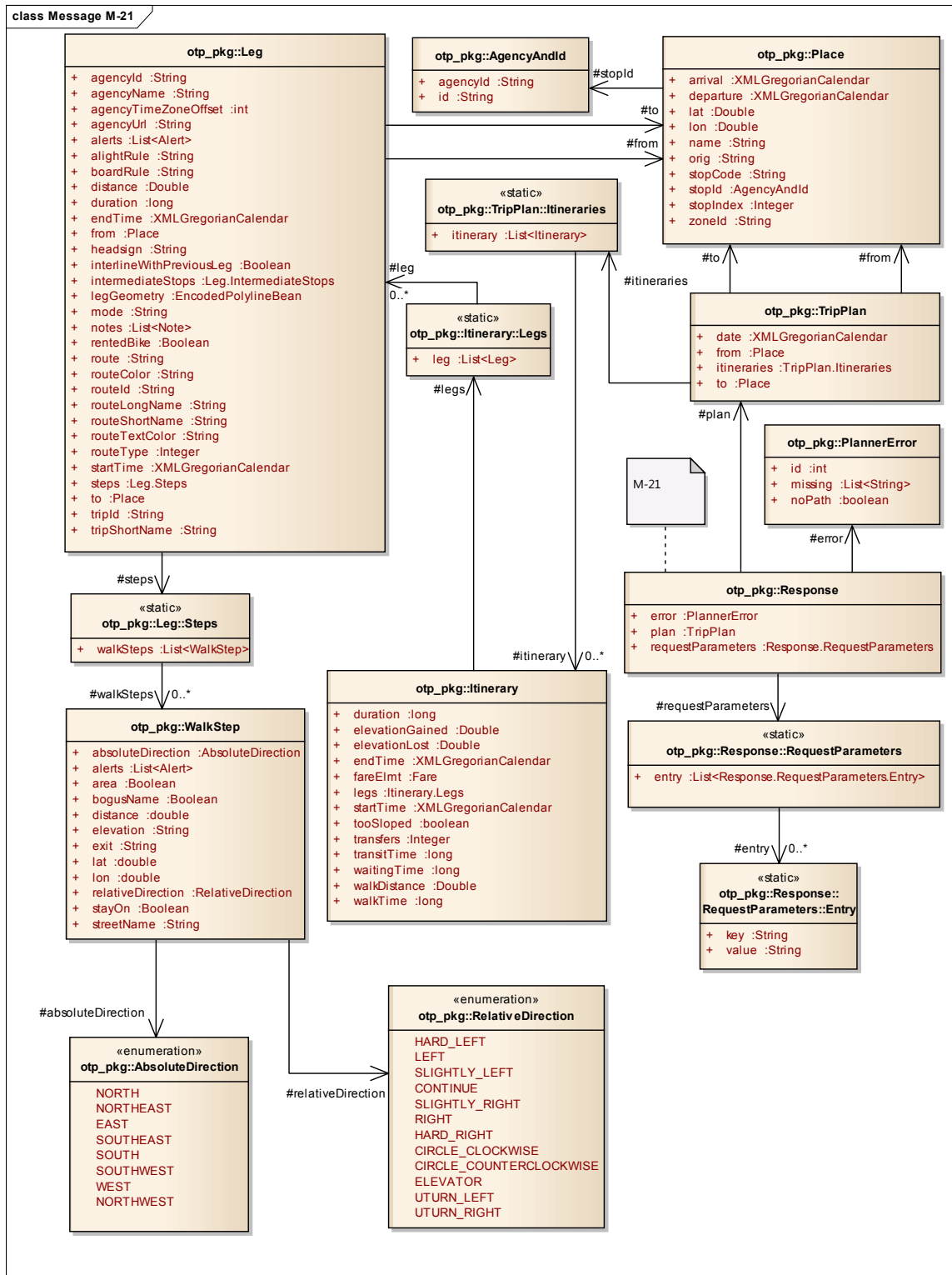
Source: Battelle

Figure B-8. Messages M-16, M-17, M-18, M-19



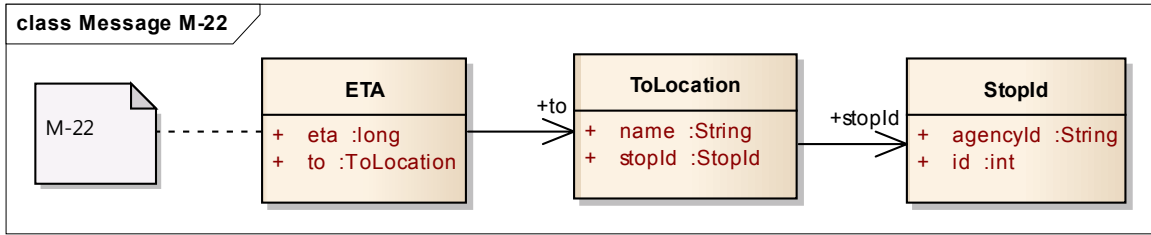
Source: Battelle

Figure B-9. Message M-20



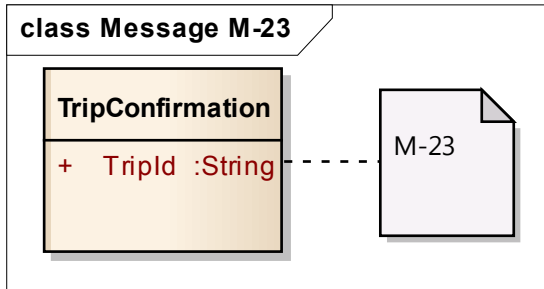
Source: Battelle

Figure B-10. Message M-21



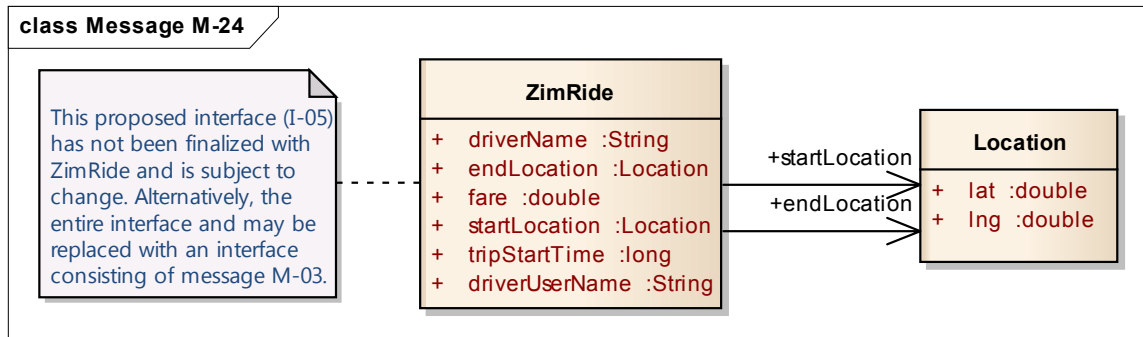
Source: Battelle

Figure B-11. Message M-22



Source: Battelle

Figure B-12. Message M-23



Source: Battelle

Figure B-13. Message M-24

AccountInformation

No description available.

AccountLoginInformation

No description available.

AccountRegistrationInformation

No description available.

AddressComponents

Note that `address_components[]` may contain more address components than noted within the `formatted_address`.

AddressQuery

Required parameters

`address` — The address that you want to geocode.

or

`latlng` — The textual latitude/longitude value for which you wish to obtain the closest, human-readable address. See Reverse Geocoding for more information.

or

`components` — A component filter for which you wish to obtain a geocode. See Component Filtering for more information. The components filter will also be accepted as an optional parameter if an address is provided.

`sensor` — Indicates whether or not the geocoding request comes from a device with a location sensor. This value must be either true or false.

Maps API for Business users must include valid client and signature parameters with their Geocoding requests.

AddressQueryResult

The only information used by the application is the location in geometry from the `addressqueryresult`.

AddressQueryResults

M-05, Geolocation Result

The result returned by the Google Geolocation Service that contains the address components plus the location coordinates in latitude and longitude for the searched address.

CurrentTConnectionRequests

No description available.

ETA

No description available.

ExpectedLocationRequest

M-18, Where Should I Be For Trip

Description

This message is a request for the location of the bus that a traveler should be on for a given trip.

FromLocation

No description available.

Geometry

No description available.

Leg

No description available.

Location

No description available.

MDTRoute

No description available.

MDTRouteType

No description available.

MDTRoutesForBusStop

No description available.

MDTRoutesForBusStopRequest

No description available.

MDTStopId

No description available.

OptReqIso

Parameters for the GET command against /ws/iso.

OptReqPlan

Holds the common parameters to make GET calls for /ws/iso, /ws/plan, /ws/plan/first, and /ws/plan/last.

OptReqPlanCommon

Holds the common parameters to make GET calls for /ws/iso, /ws/plan, /ws/wms, /ws/plan/first, and /ws/plan/last.

OptReqTileZXYPng

No description available.

OptReqWms

No description available.

PointTime

Associate a time with the location so that it is understood when the reading took place.

M-16, Bus Location Message

M-17, Vehicle Location

M-19, Where Should I Be For Trip Response Location

SearchCriteria

M01: Criteria supplied by the traveler used to search for transit opportunities. The From and To Location fields can be either an address or latitude and longitude coordinate pair. The format of the Date of Trip is MM/dd/yyyy and Time hh:mmzz. Distance is in meters and speed in meters per second.

Step

No description available.

StopId

No description available.

TConnectionRequest

M-08, T-Connection Request

Description

The T-Connection Request is generated by the TOM and sent to the Dispatcher Dashboard where the Dispatcher will either approve or reject the request.

JSON

```
{
  "TConnId": "21EC2020-3AEA-1069-A2DD-08002B30309D",
  "Route": "002",
  "RouteId": "5429",
  "StopId": "1296",
  "StopCode": "HIG13TS",
  "HoldTime": 2,
  "StartTime": 1381935831000
}
```

TConnectionRequestResponse

M-09, T-Connection Request Response

Description

This message is generated when the Dispatcher approves or rejects a T-Connection Request (M-08) using the Dispatcher Dashboard.

JSON

```
{
  "TConnId": "21EC2020-3AEA-1069-A2DD-08002B30309D",
  "Route": "002",
  "RouteId": "5429",
  "StopId": "1296",
  "StopCode": "HIG13TS",
  "HoldTime": 2,
  "Status": "Rejected",
  "StartTime": 1381935831000
}
```

TConnectionStatus

M-07, T-Connect Status

Description

This message is generated by the IDTO Traveler Services component and sent to the mobile device of the traveler when a T-Connection has been created and the status of that T-Connection. When the status of that T-Connection changes, an updated message is generated and sent to the traveler.

ToLocation

No description available.

Trip

M-03, Trip

The Trip message contains the attributes and information for a particular trip. It includes duration of the trip, start and end times, legs and steps that the traveler will take to complete the trip. Durations are in milliseconds, distance in meters, speed in meters per second, times are milliseconds since epoch.

TripCancellation

M-15, Trip Cancellation

TripConfirmation

No description available.

TripNotification

M-06, Trip Notification

Description

This is a message generated by the IDTO Traveler Services component to the mobile device of the traveler notifying them of an upcoming trip. The notification will include the details of the trip including the start time and duration.

This may end up being the same as M-03 (Trip)

Trips

M-02, Trips

The Trips message contains an array of trip (M-03) objects.

ViewPort

Viewport contains the recommended viewport for displaying the returned result, specified as two latitude,longitude values defining the southwest and northeast corner of the viewport bounding box. Generally the viewport is used to frame a result when displaying it to a user.

ZimRide

I-05, our ZimRide interface, is undefined because we really don't know what we are doing with ZimRide. Most likely we will be screen scraping to find ride opportunities (trips ?), and maybe could use M-03 for that.

TConnectStatusEnum

No description available.

APPENDIX C. ZimRide Search API

https://[zimride api url]/trips

Parameters

fromPlace=[location in format latitude, longitude]
 toPlace=[location in format latitude, longitude]
 date=[date in format yyyy-mm-dd hh:mm:ss]
 maxWalkDistance=[max walk distance between steps in meters]
 platform_id=[platform_id]

Response Example (json formatted)

Each itinerary represents a Zimride trip in the system that matched the search criteria.

```
{
  "requestParameters": {
    "toPlace": "39.989692,-83.0206883",
    "fromPlace": "40.0706879,-82.858598",
    "date": "2014-11-21",
    "maxWalkDistance": "10",
    "platform_id": "174"
  },
  "plan": {
    "date": 1416528000000,
    "from": {
      "lon": -82.858598,
      "lat": 40.0706879
    },
    "to": {
      "lon": -83.0206883,
      "lat": 39.989692
    },
    "itineraries": [
      {
        "duration": null,
        "startTime": null,
        "endTime": null,
        "walkTime": null,
        "walkDistance": 0,
        "fare": 4,
        "legs": [
          {
            "startTime": null,
            "endTime": null,
            "distance": 10.232789798394,
```

```
"mode": "ZIMRIDE",
"tripld": "877815",
"from": {
  "lat": 40.0706879,
  "lon": -82.858598
},
"to": {
  "lat": 39.989692,
  "lon": -83.0206883
},
"legGeometry": {
  "points": null,
  "levels": null,
  "length": null
},
"duration": null
}
]
}
]
}
}
```

U.S. Department of Transportation
ITS Joint Program Office-HOIT
1200 New Jersey Avenue, SE
Washington, DC 20590

Toll-Free "Help Line" 866-367-7487
www.its.dot.gov

FHWA-JPO-16-274



U.S. Department of Transportation