**Demonstration Flow Design of T-CONNECT for IDTO Project**

* **Relevant Routes**:

The following **routes** and **stops** are of our interests, whose trip-update information should be parsed into “TransitETA” table as a high priority.

|  |  |  |  |
| --- | --- | --- | --- |
| Agency | Route | Stop | Stop\_id |
| Bart | Pittsburg/Bay Point o/from SFIA/Millbrae | Pittsburg |  |
| Tri\_Delta | 300 EB | Bart Central Platform (Pittsburg) | 818889 |
|  | 300 EB | Hillcrest Park & Ride station | 817754 |
|  | 300 WB | Hillcrest Park & Ride station | 817754 |
|  | 300 WB | Bart Central Platform (Pittsburg) | 818889 |
|  | 380 WB | Hillcrest Park & Ride station | 817754 |
|  | 388 WB | Hillcrest Park & Ride station | 817754 |
|  | 383 CW | Hillcrest Park & Ride station | 817754 |
|  | 383 CCW | Hillcrest Park & Ride station | 817754 |
|  | 390 EB | Bart Central Platform (Pittsburg) | 818889 |
|  | 390 EB | Hillcrest Park & Ride station | 817754 |
|  | 390 WB | Hillcrest Park & Ride station | 817754 |
|  | 390 WB | Bart Central Platform (Pittsburg) | 818889 |
|  | 391 EB | Bart Central Platform (Pittsburg) | 818889 |

As discussed before, a column of Scheduled Time of Arrival (“STA”) has been added to the table.

This version of demo will be able to show the CONNECTIONs taken place in the following transfer trips, which will possibly be extended in future.

|  |  |  |  |
| --- | --- | --- | --- |
| Direction | From Route | To Route | Transfer stop\_id |
| Westbound tide, AM | 380 WB | 300 WB | 817754 |
| Westbound tide, AM | 383 CCW | 300 WB | 817754 |
| Westbound tide, AM | 388 WB | 300 WB | 817754 |
| Eastbound tide, PM | BART | 300 EB | 818889 |
| Eastbound tide, PM | BART | 390 EB | 818889 |
| Eastbound tide, PM | BART | 391 EB | 818889 |

* **Notes**:
  + The main algorithm to trigger push notification is based on possible failure of the scheduled connection (transfer on an interval not more than 10 minutes).
  + Add a table of “Tconnect2” to include information of the trips having requested connection protection service; And add a table of “TconnUsers” to include all users who have subscribed for the service.

**Table Tconnect2**

Table schema:

CREATE TABLE `Tconnect2` (

`event\_id`, varchar(4) “yymmddnnn”

`event\_req\_time `,

`this\_route\_agency `,

`this\_route\_id `,

`this\_route\_arr\_stop `,

`this\_trip\_id `,

`next\_route\_agency `,

`next\_route\_id `,

`next\_route\_dep\_stop`,

`next\_trip\_id `,

`requested\_holding\_time`,

`event\_state`, varchar(4) NOT NULL

PRIMARY KEY (`event\_id`, `event\_state`),

) ENGINE=InnoDB DEFAULT CHARSET=latin1

**Table TconnUsers**

Table schema:

CREATE TABLE `Tconnect2` (

`event\_id`,

`user\_token`,

`user\_state `, varchar(4) NOT NULL

PRIMARY KEY (`user\_token`),

) ENGINE=InnoDB DEFAULT CHARSET=latin1

* + The code of `event\_status`:

|  |  |  |
| --- | --- | --- |
| Value  (‘zyxs’ decimal) |  | Note |
| 0000 | Request not starts |  |
| 0001 | Request start |  |
| 0011 | Approval by dispatcher |  |
| 0021 | Rejected by dispatcher | x is the digit stands for dispatch operation |
| 0031 | Time out at waiting list |
| 0041 | Ready for dispatching but not done |
| 00x2 | The first notification has been pushed to user. |  |
| 01x2 | The transfer-to bus has left after the requested holding. | Y is the digit stands for bus reaction |
| 02x2 | The transfer-to bus has left without a holding service. |
| 0yx3 | The second notification has been pushed to user. |  |
| 1yx3 | Service Success according to passenger’s feedback | z is the digit stands for passenger’s feedback |
| 2yx3 | Service Failure according to passenger’s feedback |
| 3yx3 | Time out for passenger’s feedback |

* **Demo Flow for All Involved Parties**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Passenger | Server | Dispatch Center | Bus Driver |
| 10:01 | O/D Input, request trip planning |  |  |  |
| Confirm Trip to subscribe notification service | Receive and store (to MySQL) information of planned trip (\*) |  |  |
|  | Get ETA of BART at “Pittsburg/Bay Point”, 10:17 for example |  |  |
| 10:12 |  | <5 min. before arrival>  Update “event\_state” in the Tconnect table **(0001)** |  |  |
|  |  | Receive new request and push a message dialog |  |
| 10:14 |  |  | Ask driver to hold until 10:20 by calling |  |
|  |  |  | Approve connection protection |
|  |  | Press “Approve” or “Reject” button; Update “event\_state” **(00x1);** |  |
|  | Request the second push notification; Update “event\_state” **(00x2);** |  |  |
| Receive the first push notification: “Upon your request, the Bus 300 will wait at XXX until TT:TTam.” |  |  |  |
| 10:20 |  |  |  | Finish waiting and departing |
| 10:21 |  | <1 min. after departure>  Update “event\_state” **(01x3)** |  |  |
| Receive the second push notification: “The bus 300 on your trip has left from XXX.” |  |  |  |
| 10:22 | Answer the question on web “Was the last service successful?” |  |  |  |
|  | Update “event\_state” **(zyx3)** |  |  |

* **Working Flow for System Components (TCONNECT2)**



* **Information Design of Dispatch Terminal**

1. The content of pop-up message at dispatch terminal

The information items below are necessary:

* + “The bus line requested to hold: ”
  + “The vehicle id: ”
  + “Requested holding time, until: ”

The response buttons include

* + “Approve” (in green)
  + “Reject” (in orange)

2. The columns of event record table

Client Seq.: “mmddyysss”, e.g. “101316001”

From Trip: “route id\_stop name”, e.g. “380WB\_Hillcrest”

To Trip: “route\_id\_stop\_name”, e.g. “300WB\_Hillcrest”

Request time: “hh:mm”, e.g. “09:35” (local time)

Status: “Approved” / “Rejected” / “Time out”