Test cases Document for Backend testing

Extended System test cases of "Test Cases_reviewed.pdf"
Date: 7.6.2016
Last modifications: 22.6.2016
Performed by mzeen
Version: 0.2 in development, on commit 449e3b5

Sr.no	Use Case	Function Being Tested	Initial System State	Input	Expected Output	Test execution documentation
1	IC_C1	SF_S3: Send list of stops	Stop table exists in database	./stops	List of stops with stopID, stop name, line passing through the stops, location of the stop, bus schedule of the stop is displaying. Schedule contains line name, lineID, stopID, arriving time and timestamp.	Passed
2	IC_C2	SF_S5: Send properties of bus			*Not implemented in this iteration. SF_S2 can be used instead (see below)*	
3		SF_S6: Send GPS-Data	GPS data was sent to database	./busses/{busId}/	Current GPS Data coordinates with timestamp	Passed
4	IC_C3	SF_S6: Send GPS-Data			See above	
5	IC_B1	SF_S1: Send list of lines		./lines	List of Lines with properties lineID, name, routeID, timestamp and busses running on that specific line	Passed
6		SF_S2: Send list of busses	Bus table exists in database	./busses	List of all busses with their busID, number plate, color, picture and the line they serve	Passed
7		SF_S4: Send list of routes	Route table exists in database	./routes	List of all routes with routeID, route coordinates in geojson format and timestamp	Passed
8		SF_S7: Store bus and line	Bus and line tables exist in database	./updateBusStatus Method: POST Content type: application/json	Result status 200, Bus and line information stored	Passed; tested with lineID = 3, busID = 2, with initially bus 2 on line 1.

#	IC	SF	Precondition	Request	Expected Result	Output

				Content:
				{
				"lineId":number,
				"busId":number
				}
						Output of post is 200, calling ./lines reveals that bus 2 is now on line 3.

9	IC_B2	SF_S13: Store number of available seats			*For next iteration*	
10	IC_B3	SF_S4: Send list of routes			See above	
11	IC_B4	SF_S3: Send list of stops			See above	

12	-	SF_S8: Store GPS data	Bus table exists in database	./realTimeData
				
				Method: POST
				Content type: application/json
				Content:
				{
				"busId":number,
				"position": {
				"type": "Point",
				"coordinates": [double, double]},
				"timestamp": timeStamp
				}
					Result status 200, GPS Data of bus is stored, timestamp not changed	Passed;
						With busID = 1, coordinates [15.0, -3.4] and timeStamp = 1098692636000 the output of POST is 200, calling ./busses/1 reveals that coordinates have been changed to [15.0, -3.4]. The timeStamp remains 1098692634000.

13	-	SF_S9: Send latest timestamp	Line, bus, route and stops tables exist in database	./update	Last timestamps for bus, route, line and stops	Passed
14	IC_B5	SF_S10: Notification of custom stop request	CitizenApp sends new Custom Stop Request	./customStops Method: POST Content type: application/json Content: { "lineId":number, "lat":double, "lon":double, "numberOfPersons":number, "deviceId":String, "userName":String, "userAddress":String,	Result status 200; New custom stop with assigned requestId, lineID, pickUpTime, location, number of personson, deviceID, information of passenger included name, address, and assistance, status and timestamp	Passed with the following request data; { "lineId":2, "lat":6.5, "lon":2.2, "numberOfPersons":21, "deviceId":"oihjwo", "userName":"Devil", "userAddress":"Hell", "userAssistance":[3,8],

19				"userAssistance":[number,number, …], "pickUpTime":Timestamp }		"pickUpTime":"2016-06-21 13:10:00" }
14			Request from CitizenApp with lineId has been stored, DriverApp requests pending customStops for given lineId	./customStops?lineId={lineId}	All custom stop requests for the given line with pickUpTime after current time and status pending, each with all their properties (see above)	Passed with lineId = 1 and get all the specified output.
15			DriverApp requests pending customStops for given lineId; no requests for this line	./customStops?lineId={lineId}	Empty list	Passed
16			DriverApp requests pending customStops for given lineId; requests for that line have pickUpTime before current time	./customStops?lineId={lineId}	Empty list	Passed
17		SF_S11: Response to custom stop request	DriverApp received custom stop requests for the given line, sends an update of status on a custom stop request	./customStops/{requestId} Method: POST Content type: application/json Content: { "status":number }	Result status 200; Status of given request updated as given	Passed
18			CitizenApp has the requestId	./customStops?requestId={requestId}&deviceId={deviceId}	Custom stop request for the given requestId with all properties	Passed
19	IC_B6	SF_S11: Response to custom stop request	DriverApp send status as complete	*See above with status complete*		

Author: Muhammad Zeeshan, Ricarda Rosemann
Status: Complete
Reviewer: Hafiz
Review Status: Complete