1.1.1.1 Management of the person with special needs planned routes

The person with special needs will access to a display consisting of several .NET forms, and between them, a JavaScript script showing a LEAFLET map where the starting and final route point are set using the computer mouse.

The form asks for the following data:

1. A symbolic name for the route/itinerary (hospital, school, home, etc.), and the symbolic name for the starting and final destination point.
2. Dates (period) when each specific route must be active or activated (when it is going to be made).
3. Hours of the day (from-to) when the path is usually performed.
4. Weekdays: Within a date range, the user can travel only certain days (eg. only Monday or Mondays and Wednesdays, etc).
5. Type of day: Within the week, the user indicate to the system what kind of day a certain route is done (weekdays, Saturdays, holidays)

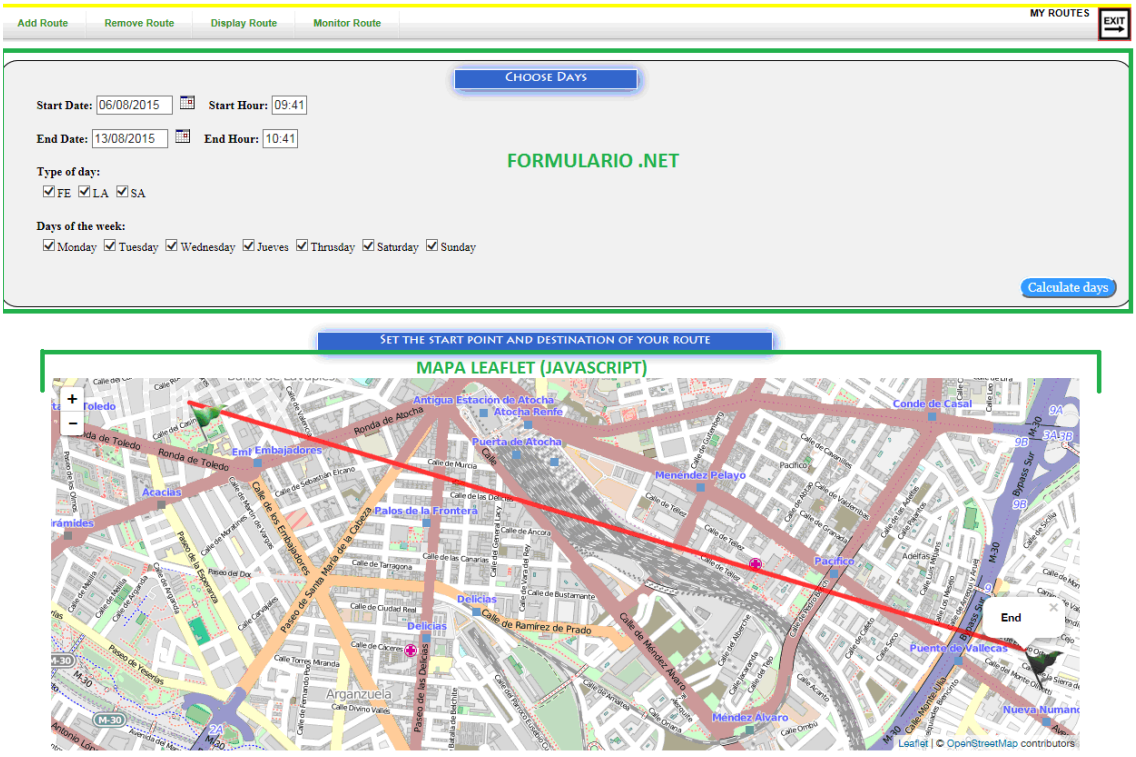


Figure 1. Route planning display

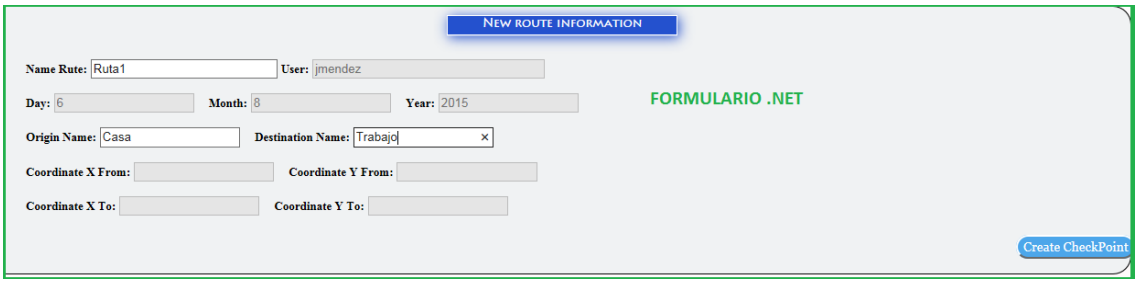


Figure 2. Indicating the route general data

Subsequently, once clicking on "`Create CheckPoint`" the system will call the EMT Opendata web service…

https://openbus.emtmadrid.es:9443/emt-proxy-server/last/geo/GetRouteLinesRoute.php

…providing the trip preferences and the starting and final points. This service returns a JSON object with route details (distance, distance to bus stops along the route, information about the various sections of route, etc.) and leads us to the display where to establish the different Checkpoints of the route.

Once in the aforementioned display, the different Checkpoints are established (checkpoint.aspx) and object elements are stored in the client, which contains data on the route and are drawn on the map as layers (JavaScript script).

NOTE: Everything drawn on the map can be consider a layer; there are on route Checkpoint layers, but also at bus stops, or in a walking intinerary, in a bus itinerary, etc. each one with its own information

On route checkpoint	
Stop checkpoint	
Bus stop checkpoint	
Walking checkpoint	
Bus checkpoint	

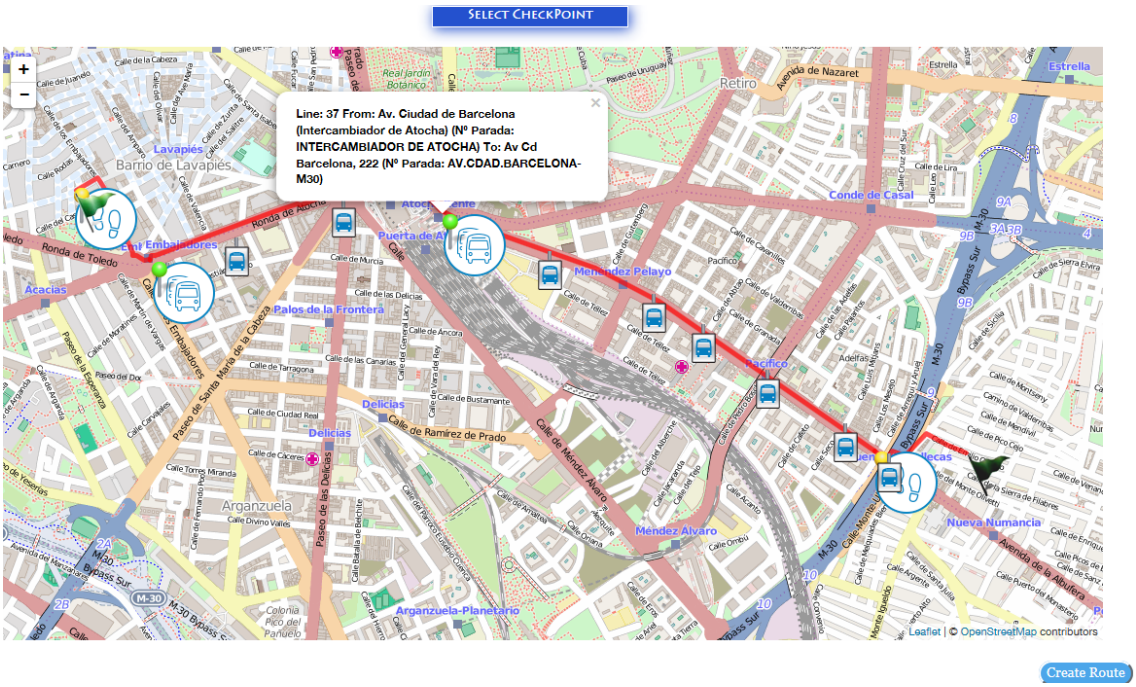


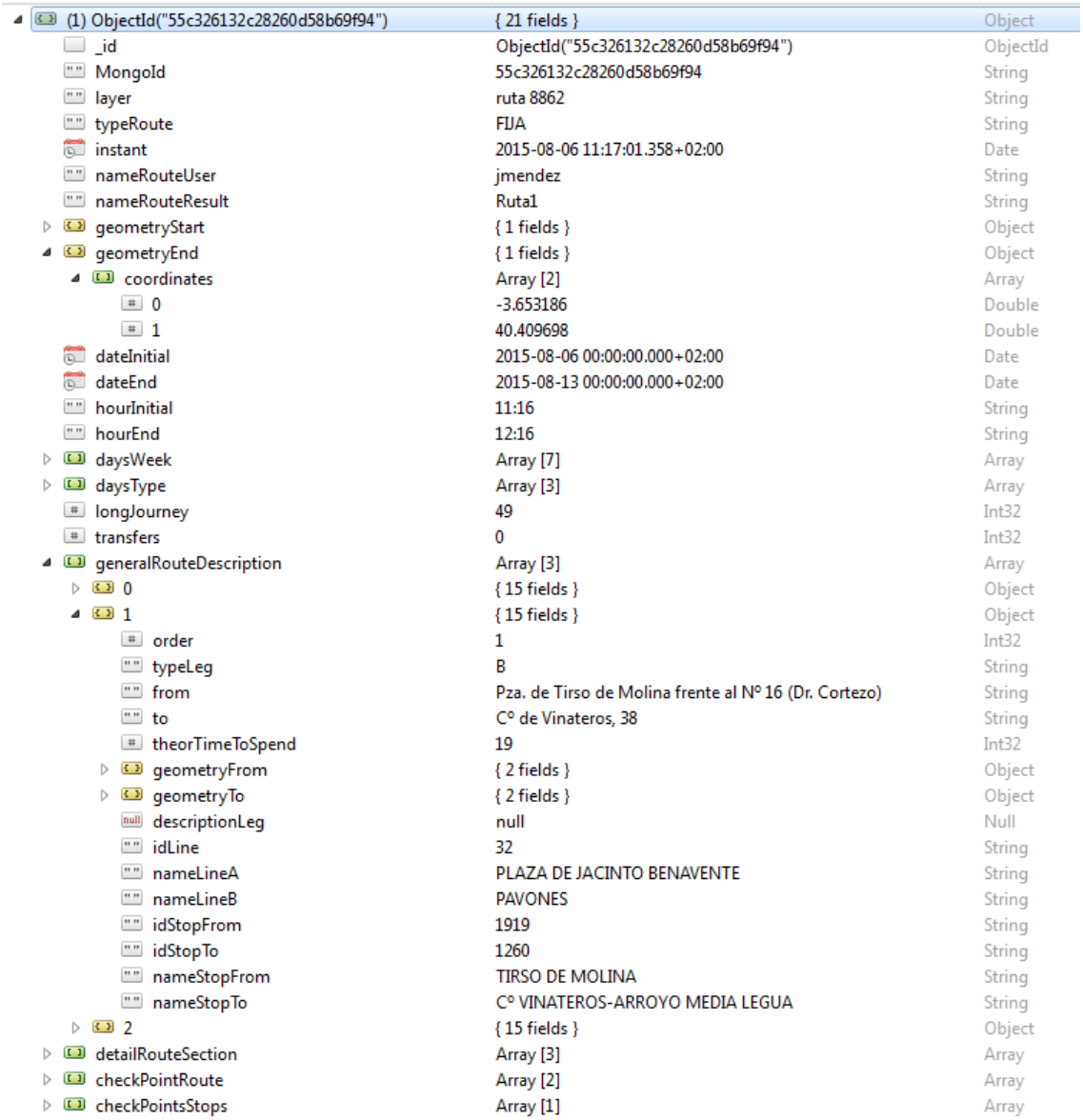
Figure 3. Creating the route checkpoints for the CEP

Once the details of the selected route are painted on the screen, the different checkpoints can be established by clicking on the map (in client mode); the checkpoint will be set as “on route” if clicking out of a layer, otherwise checkpoints can be selected as bus stops, etc. It is also possible to remove checkpoints by clicking on each Checkpoint layer.

Figure 4. Creating and removing checkpoints

Checkpoints will be later used by the CEP to release data on average time of buses passing by at a specific location on a selected route in a certain day and time slot. Alarms may be set in case of “no appearance of the person with special needs" or "time deviation according to forecasted route", for instance.

By clicking on “`Create Route”,` once all the checkpoints are set up, all route data will be stored in a MongoDB collection with the following structure:



(1) ObjectId("55c326132c28260d58b69f94")	{ 21 fields }	Object
_id	ObjectId("55c326132c28260d58b69f94")	ObjectId
MongoId	55c326132c28260d58b69f94	String
layer	ruta 8862	String
typeRoute	FIJA	String
instant	2015-08-06 11:17:01.358+02:00	Date
nameRouteUser	jmendez	String
nameRouteResult	Ruta1	String
geometryStart	{ 1 fields }	Object
geometryEnd	{ 1 fields }	Object
coordinates	Array [2]	Array
0	-3.653186	Double
1	40.409698	Double
dateInitial	2015-08-06 00:00:00.000+02:00	Date
dateEnd	2015-08-13 00:00:00.000+02:00	Date
hourInitial	11:16	String
hourEnd	12:16	String
daysWeek	Array [7]	Array
daysType	Array [3]	Array
longJourney	49	Int32
transfers	0	Int32
generalRouteDescription	Array [3]	Array
0	{ 15 fields }	Object
1	{ 15 fields }	Object
order	1	Int32
typeLeg	B	String
from	Pza. de Tirso de Molina frente al Nº 16 (Dr. Cortezo)	String
to	Cº de Vinateros, 38	String
theorTimeToSpend	19	Int32
geometryFrom	{ 2 fields }	Object
geometryTo	{ 2 fields }	Object
descriptionLeg	null	Null
idLine	32	String
nameLineA	PLAZA DE JACINTO BENAVENTE	String
nameLineB	PAVONES	String
idStopFrom	1919	String
idStopTo	1260	String
nameStopFrom	TIRSO DE MOLINA	String
nameStopTo	Cº VINATEROS-ARROYO MEDIA LEGUA	String
2	{ 15 fields }	Object
detailRouteSection	Array [3]	Array
checkPointRoute	Array [2]	Array
checkPointsStops	Array [1]	Array

Figure 5. MongoDB structure for checkpoints list

Any stored route can be selected and removed by using the command **“Remove Route”**. This function simply deletes that specific MongoDB collection.

It is also possible to visualize the subscribed routes from **“Display Route”.** This function recovers the collection for that specific route, stores it in hidden control son the server and from the client side, using Javascript, shows it on the map (LEAFLET).

The route monitoring function allows viewing, in real time, the user location and displaying the various alarms that are occurring in case of any deviation from the planned route. This functionality uses information supplied not only by the person with special needs Interface Prototype tracking but also by the alerts comming from the Context Event Processor running.