1.Summary

This application is for GPS tracking of vehicles.

It shows location of vehicles in real time and simulates tracks between date/times and reports all information such as stops, idle, accelerate.

For more information, please check this link

<https://play.google.com/store/apps/details?id=com.jo.spectrum&hl=en_US>

2.Functionality

-Monitor Page

Display vehicles on the map and show their information in real time.

The user can select any vehicle and display it individually on the map.

-Replay Page

Tracks history of selected vehicles in given dates/times.

Select date and replay GPS track points and trip log, it’s information such as speeding, max speed, Harsh accelerate, Harsh Brake. You can navigate this page.

-Reports Page

Report a variety of data in Chart and Table format.

For selected vehicle and selected date, it shows any information of mileage and Stops and Max Speed, Fuel etc.

-Activity page

Once the user register his device, the user can activate his device on this page.

The user should fill out his device id and plate number, country, payment information etc.

-Update Device

Update Device information

-Routing page

This page provides Direction Service. In other word, routing service.

3.Purpose of the project

I have been working on the Spectrum Tracking Project for 3 years.

But I was not working on the back end and worked on only front end such as web front end, mobile apps.

So I don’t know well the back end and table schema.

I think they used Node.js and MongoDB as back end.

May be, it is better in speed.

And they also used PHP. But I don’t know what php framework they implemented.

Now we should clone this application into new project.

I have only mobile app source code and know the API.

So it can be help for you.

In new project, you should build front end based on Baidu map sdk.

Because it is similar with Google Map API, if you are familiar with Google Maps API, you can build it easily.

I don’t mind to what you would prefer javascript frame work and back end.

But should be flexible and extendable.

There no technological issue specially as you see.

There are several pages but don’t need all.

Just focus on above 6 pages.

4. API

@POST("auth/login")

/\*\* param

email,

password

\*\*/

Call<ResponseBody> authLogin(

@Body HashMap<String, Object> body)

-----------------------------------------------------------------------------------------------------------------------

@POST("auth/logout")

/\*\*

\*\*/

Call<ResponseBody> authLogout();

------------------------------------------------------------------------------------------------------------------------

@POST("auth/register")

/\*\* param

email,

password,

firstName,

lastName

\*\*/

Call<ResponseBody> authRegister(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@POST("auth/verify")

/\*\* param

email,

password,

firstName,

lastName

\*\*/

Call<ResponseBody> authVerify(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@POST("auth/reset-password")

/\*\* param

email,

password,

code

\*\*/

Call<ResponseBody> resetPassword(

@Body HashMap<String, Object> body);

@POST("auth/resend-password-reset")

/\*\* param

email

\*\*/

Call<ResponseBody> authResendPasswordReset(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@GET("assets")

Call<ResponseBody> assets();

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/getAllTrackersWeb")

Call<ResponseBody> getAllTrackersWeb(

@Header("X-CSRFToken") String xCSRFToken

);

------------------------------------------------------------------------------------------------------------------------

@GET("trackers/{id}")

Call<ResponseBody> trackers\_id(

@Path("id") String trackerId);

------------------------------------------------------------------------------------------------------------------------

@GET("users/{id}")

Call<ResponseBody> users\_id(

@Path("id") String userId);

------------------------------------------------------------------------------------------------------------------------

@GET("trackers/getTrackerBySpectrumId/{id}")

Call<ResponseBody> getTrackerBySpectrumId(

@Path("id") String spectrumId);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/modify")

/\*\* param

id,

photoStatus(true/false),

\*\*/

Call<ResponseBody> modify\_tracker(

@Body HashMap<String, Object> body

);

-----------------------------------------------------------------------------------------------------------------------

@POST("trackers/registerPhoneTracker")

/\*\* param

reportingId (email),

userId

\*\*/

Call<ResponseBody> registerPhoneTracker(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("assets/createPhoneAsset")

/\*\* param

trackerId ,

name(firstName),

spectrumId(email)

driverName(firstName)

userId

\*\*/

Call<ResponseBody> createPhoneAsset(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/unShareTrakcer")

/\*\* param

reportId ,

email

\*\*/

Call<ResponseBody> unShareTrakcer(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/shareTrakcer")

/\*\* param

reportId ,

email,

flag

\*\*/

Call<ResponseBody> shareTrakcer(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/getShareUsers")

/\*\* param

reportId ,

\*\*/

Call<ResponseBody> getShareUsers(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("users/setShareFlag")

/\*\* param

reportId ,

flag

\*\*/

Call<ResponseBody> setShareFlag(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/getTrackersByTrackerIds")

/\*\* param

tracker\_ids ,

\*\*/

Call<ResponseBody> getAllTrackers(

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("asset-logs")

/\*\* param

reporting\_id(email),

dateTime,

lat,

lng,

ACCStatus,

currentTripMileage,

speedInMph,

trackerModel

\*\*/

Call<ResponseBody> postUserLocation(

@Header("X-SpectrumTracking-TrackerEndpointKey") String apiKey,

@Body HashMap<String, Object> body

);

------------------------------------------------------------------------------------------------------------------------

@POST("users/getUserFromId")

/\*\* param

reporting\_id(email),

dateTime,

lat,

lng,

ACCStatus,

currentTripMileage,

speedInMph,

trackerModel

\*\*/

------------------------------------------------------------------------------------------------------------------------

Call<ResponseBody> getUserInfo(

@Header("X-CSRFToken") String xCSRFToken

);

------------------------------------------------------------------------------------------------------------------------

@GET("assets/{id}/logs")

Call<ResponseBody> assets\_logs(

@Path("id") String trackerId,

@Query("startDate") String startDate,

@Query("endDate") String endDate,

@Query("animation") Boolean animation);

-----------------------------------------------------------------------------------------------------------------------

@GET("triplog/{id}/logs")

Call<ResponseBody> trip\_logs(

@Path("id") String reportingId,

@Query("startDate") String startDate,

@Query("endDate") String endDate);

------------------------------------------------------------------------------------------------------------------------

@GET("alertlog/{id}/logs")

Call<ResponseBody> event\_logs(

@Path("id") String reportingId,

@Query("startDate") String startDate,

@Query("endDate") String endDate);

------------------------------------------------------------------------------------------------------------------------

@POST("https://app.spectrumtracking.com/php/route.php")

/\*\* param

traction\_type(email),

card\_holder\_name,

card\_number,

card\_expiry,

card\_cvv,

\*\*/

Call<ResponseBody> tokenizeCreditCard(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@GET("auth")

Call<ResponseBody> doAuth();

------------------------------------------------------------------------------------------------------------------------

@PUT("users/{id}")

Call<ResponseBody> updateCreditCardInfoSecondary(

@Path("id") String userId,

@Query("token\_type") String tokenType,

@Query("token\_number") String tokenNumber,

@Query("cardholder\_name") String cardholderName,

@Query("card\_type") String cardType,

@Query("exp\_date") String expDate);

@PUT("users/updateAppReviewReminder/{email}")

Call<ResponseBody> addReviewNumber(

@Path("email") String email

);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/register")

/\*\* param

spectrumId(tracker\_id),

userId,

email,

plan

\*\*/

Call<ResponseBody> trackerRegister(

@Header("X-CSRFToken") String xCSRFToken,

@Body HashMap<String, Object> body);

-----------------------------------------------------------------------------------------------------------------------

@POST("orders/generateToken")

/\*\* param

card\_holder\_name,

auth,

card\_number,

card\_expiry,

card\_cvv,

\*\*/

Call<ResponseBody> generateToken(

@Body HashMap<String, Object> body);

-----------------------------------------------------------------------------------------------------------------------

@POST("assets")

/\*\* param

name,

trackerId,

driverName,

spectrumId,

\*\*/

Call<ResponseBody> createAssets(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@POST("orders/payment")

/\*\* param

card\_number,

card\_expiry,

card\_cvv,

current\_code,

amount,

items,

auth,

productService,

sendHtml

\*\*/

Call<ResponseBody> ordersPayment(

@Body HashMap<String, Object> body);  
------------------------------------------------------------------------------------------------------------------------

@POST("trackers/setGeoFence")

/\*\* param

latGeo,

lngGeo,

radiusGeo,

assetId,

plateNumber,

driverName

\*\*/

Call<ResponseBody> setGeoFence(

@Body HashMap<String, Object> body);

---------------------------------------------------------------------------------------------------------------------

@POST("trackers/modify")

/\*\* param

id,

driverName,

operation

\*\*/

Call<ResponseBody> modify(

@Header("X-CSRFToken") String xCSRFToken,

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/getImageUrl")

/\*\* param

name

\*\*/

Call<ResponseBody> getImageUrl(

@Body HashMap<String, Object> body);

------------------------------------------------------------------------------------------------------------------------

@POST("trackers/imageUpload")

/\*\* param

filename,

state,

File

\*\*/

Call<ResponseBody> imageUpload(

@Body RequestBody body);

------------------------------------------------------------------------------------------------------------------------

@GET("trackers/byAssetName/{id}")

Call<ResponseBody> alarm(

@Path("id") String trackerId);

------------------------------------------------------------------------------------------------------------------------

@GET("api-interface/coord2AddressMapbox/{latLng}")

/\*\* param

{lng},{lat}

\*\*/

Call<ResponseBody> coord2AddressMapbox(

@Path("latLng") String latLng);

-----------------------------------------------------------------------------------------------------------------------

@GET("api-interface/coord2AddressMapboxAll/{latLng}")

/\*\* param

{lng lat},{lng lat},…

\*\*/

Call<ResponseBody> coord2AddressMapboxAll(

@Path("latLng") String latLng);

------------------------------------------------------------------------------------------------------------------------

@GET("v1/geo/reverse-geocodes")

Call<ResponseBody> getAddress(

@Query("latitude") Double lat,

@Query("longitude") Double lon,

@Query("key") String AppID

);

----------------------------------------------------------------------------------------------------------------------