*Specification Draft (version 3)*

Request/Response

**Request pattern:**

{

command: ~The Command~, -> Int

data: ~Data If Required~ -> JSON Object

}

**Response pattern:**

{

command: ~The Command~, -> Int

code: ~Response Code~, -> Int

data: ~Data If Required~ -> JSON Object

}

**The list of requests/responses:**

// Action: Log in the User

{

command: LOGIN,

data: ~Credentials~ -> Credentials

}

{

command: LOGIN,

code: ~ResponseCode~,

data: {

error: ~Error Message~ -> String

}

}

// Action: Add the last created route to the database

{

command: ADD\_ROUTE,

data: {

total\_distance: ~Kilometers~ -> Double

polyline: ~Google Maps Polyline~, -> String

stops: [ ~Cities~ ] , -> City[]

}

}

{

command: ADD\_ROUTE,

code: ~Response Code~,

data: {

id: ~Route ID~ -> Int

}

}

// Action: Get a complete information of a route from the database

{

command: GET\_ROUTE,

data: {

id: ~Route ID~ -> Int

}

}

{

command: GET\_ROUTE,

code: ~Response Code~

data: {

route: ~Route~ -> Route

}

}

// Action: Get a brief list of routes information

{

command: GET\_ROUTE\_LIST,

data: { <empty> } // may contain some filters later

}

{

command: GET\_ROUTE\_LIST,

code: ~Response Code~,

data: {

routes\_list: [ -> JSON array

route: { -> JSON object

id: ~Route ID~, -> Int

info: ~Route Info~ -> RouteInfo  
 }

]

}

}

Glossary

blue - to be used for the current version

*grey* - for the future development

**Common data types:**

Credentials {

name: ~User Name~, -> String

password: ~User Password~ -> String

}

Route {

id: ~Route ID~, -> Int

info: ~Short Route Info~, -> RouteInfo

polyline: ~Google Maps Polyline~, -> String

stops: [ ~Cities~ ] -> City[]

path:Path; // \*\_\*

}

RouteInfo {

origin: ~From City~, -> City

destination: ~To City~, -> City

total\_distance: ~Kilometers~ -> Double

}

City {

place\_id: ~Google Maps Place~ -> String

formatted\_address: ~City Name~ -> String

location: ~Location~ -> Location

population:Int;

}

Location {

lat: ~Latitude~ -> Double

lng: ~Longitude~ -> Double

}

Path { // \*\_\*

roads:Road[];

???

}

Road { // \*\_\*

id: ~Road ID/Place ID~, -> String

name: ~Road Name~, -> String

// M05 (Kiev-Odessa), M06 (Kiev-Lviv) etc.

statistics: ~Road Statistics~ -> ???

// quality, traffic etc.

???

}

**Response codes:**

OK = 200

BadRequest = 400

Unauthorized = 401

Forbidden = 403

NotFound = 404

**Request codes:**

LOGIN = 0

ADD\_ROUTE = 1

GET\_ROUTE = 2

GET\_ROUTE\_LIST = 3

**\*\*Add your argumented propositions here.\*\***