Implement API with a single POST request endpoint. The endpoint receives address details and calculates the transport price based on the route details. The API must be guarded, which means not everyone should have access it, hence implement also an **Authentication**.

If possible write also Unit / Feature tests.

Example request body:

{

"addresses": [

{

"country": “DE”,

"zip": “10115",

"city": “Berlin"

},

{

"country": “DE”,

"zip": “20095",

"city": “Hamburg"

}

]

}

**addresses** is the array containing **address objects**. Address object, has following properties:

**country**: country code of the address

**zip**: zip code of the address

**city**: city of the address

Request validation rules:

1. **addresses:** must be an array containing at least 2 items. There is no max limit for the items count.
2. **address object:** country, zip, and city combination must exist in the database

MongoDB database configuration:

Connection uri: **mongodb+srv://admin:mH2Kgdobov0UhUrn@cluster0.xviag.mongodb.net/**

Database name: interview

Database consists of two collections:

c**ities**: collection of cities that transport price can be calculated for. Key attributes:

**name:** Name of the city

**zipCode:** zip code of the city

c**ountry:** country code where the city is located at

**vehicleTypes:** collection of vehicle types. Consist of data that are used for price calculation. Key attributes:  
**number:** number of vehicle type, will be added to the price response

**cost\_km:** cost of the vehicle per kilometer

**minimum**: minimum price of the transportation for the vehicle

Business logic:

Calculate the total distance between all addresses in **kilometers**. Use google direction api for that. Api key: **AIzaSyA\_z4H4vBv0Mn8og2T4c2\_iWqJrfiLAIqY** Then get vehicle types from the database and calculate the price for each vehicle type. You have the **cost\_km** attribute for the vehicle type, that indicates the price for 1 km. You should calculate the total price for the vehicle. Additionally you must check if the calculated price is not less than **minimum.**

Example response:

**vehicle\_type** is the **number** of the vehicle type that price is calculated for. **price** is resulting price value for that vehicle type.

[

{

"vehicle\_type": 12,

"price": 69

},

{

"vehicle\_type": 0,

"price": 119

},

{

"vehicle\_type": 11,

"price": 51.24

},

{

"vehicle\_type": 1,

"price": 179

},

{

"vehicle\_type": 2,

"price": 169

},

{

"vehicle\_type": 3,

"price": 179

},

{

"vehicle\_type": 4,

"price": 179

},

{

"vehicle\_type": 5,

"price": 179

},

{

"vehicle\_type": 6,

"price": 179

},

{

"vehicle\_type": 7,

"price": 259

},

{

"vehicle\_type": 8,

"price": 179

},

{

"vehicle\_type": 9,

"price": 179

},

{

"vehicle\_type": 10,

"price": 339

}

]