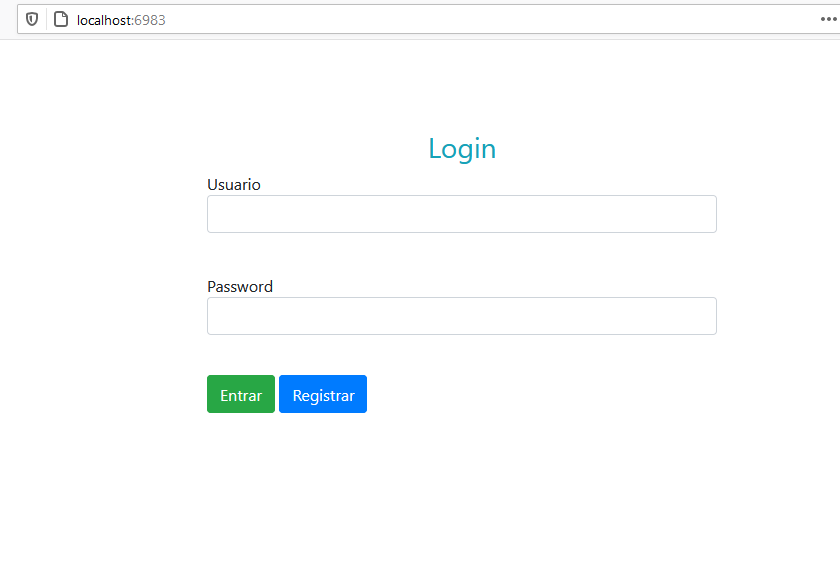
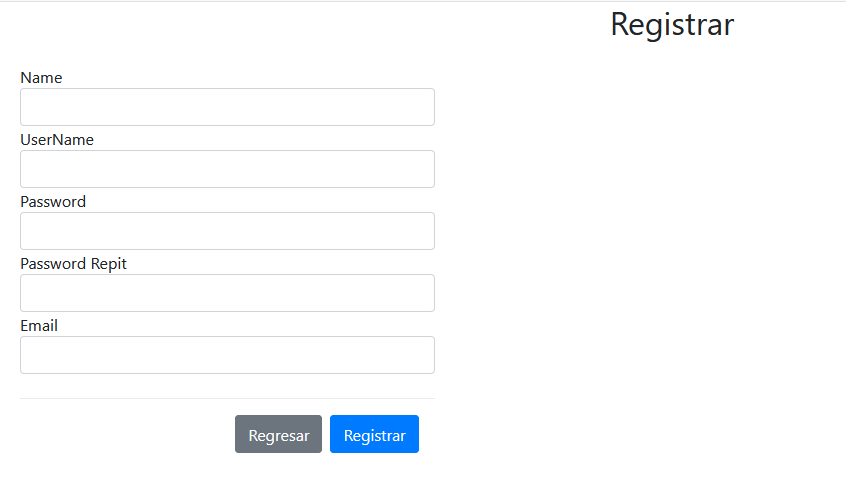
Documentación Portal

Usuario Administrador autogenerado: **Admin** password: **123456**

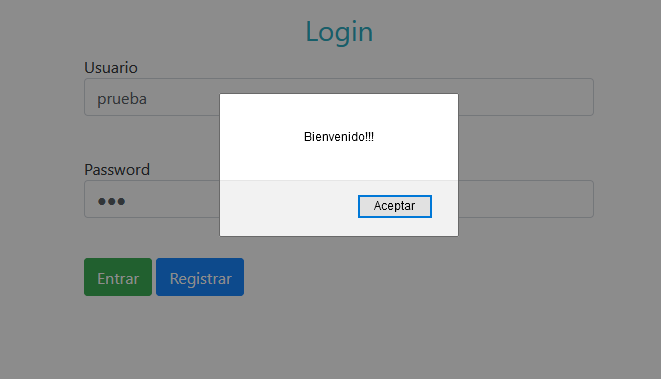
El ingreso por el portal se realiza a través del login, donde se coloca el UserName, dentro se validan los permisos y si es un usuario Administrador ingresa y se cargan los roles pertinentes, de igual forma que el perfil de los viajeros.



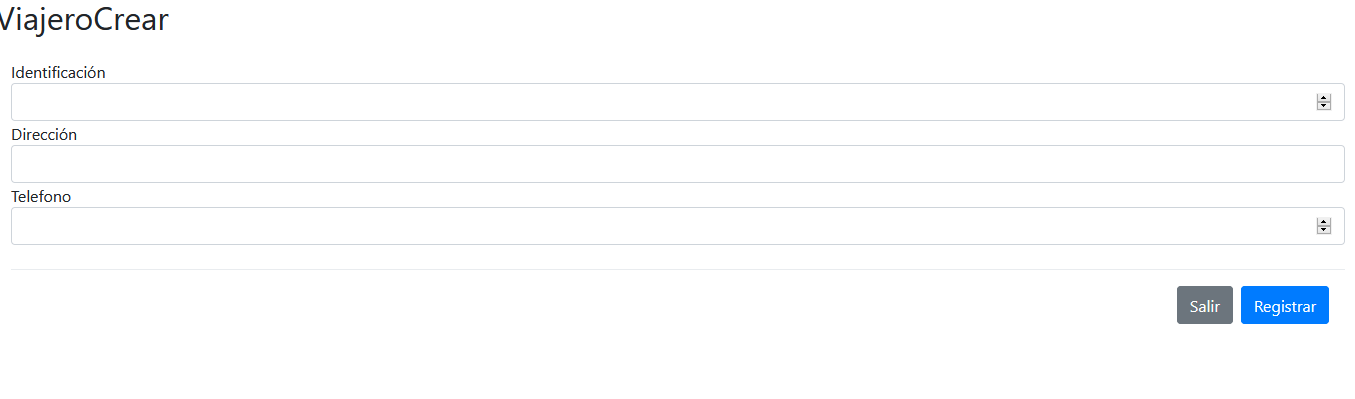
Para los viajeros nuevos, está el botón de registro.

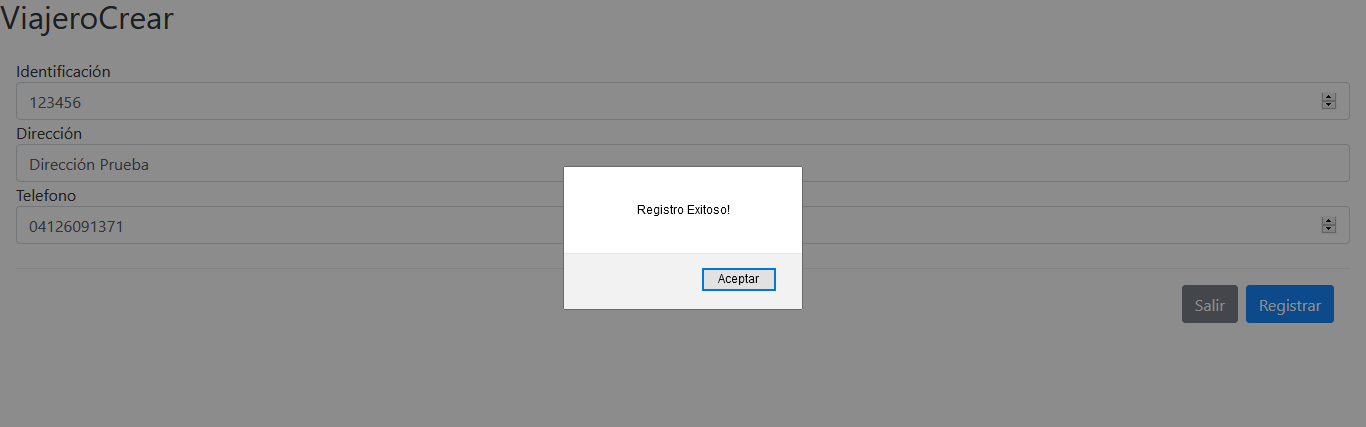


Donde colocamos, los datos básicos para el registro, una vez registrado, nos indica que se registró con éxito, devolviéndonos al login, para poder ingresar.

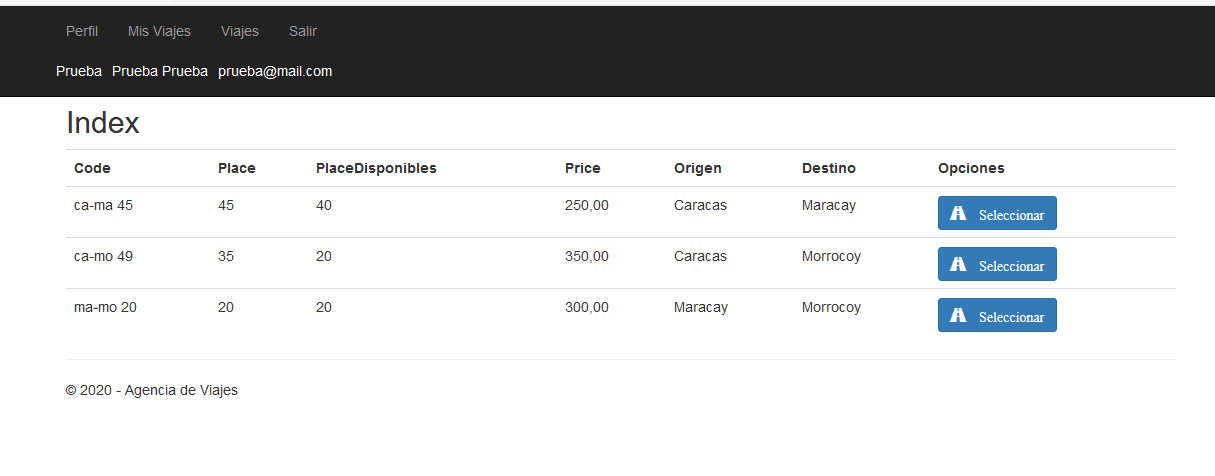


Una vez registrado es un usuario que ingresa por primera vez, nos solicitara los datos restantes para poder ingresar.



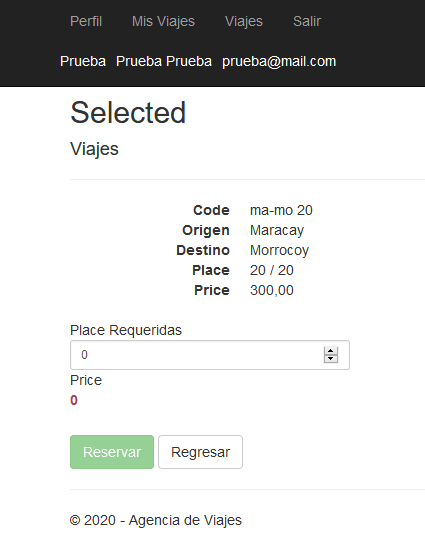


Una vez registrado, podemos visualizar todos los viajes disponibles.

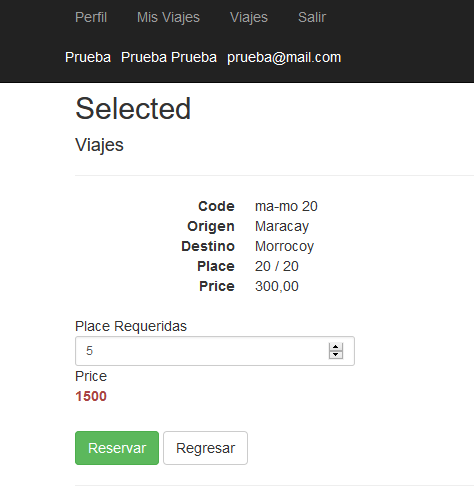


Podemos visualizar en el menú, el perfil, los viajes (Reservas que posee el viajero), los viajes disponibles y la opción de salir.

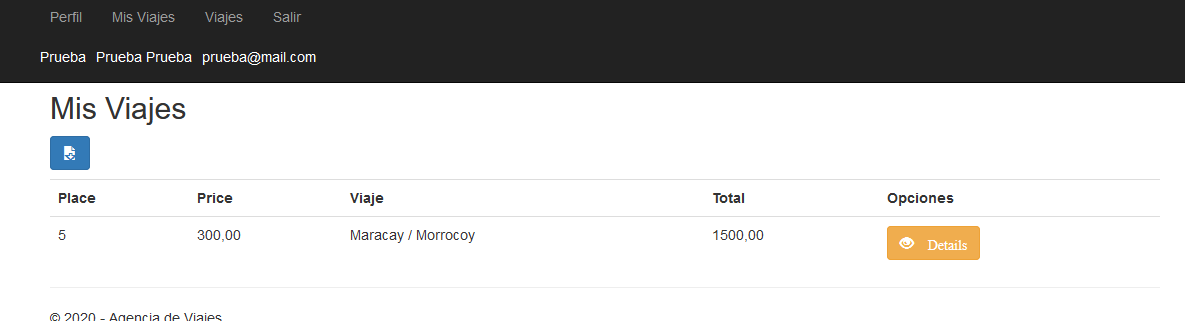
Al seleccionar la opción de seleccionar se despliega la ventana que nos permite realizar la reserva.



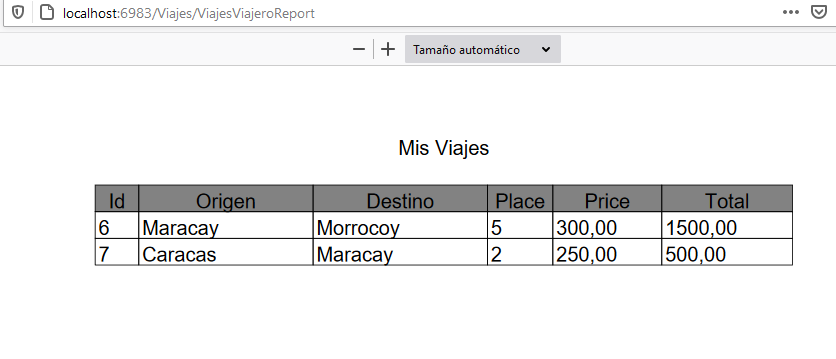
Cuando colocamos las plazas se activa el botón y actualiza el precio



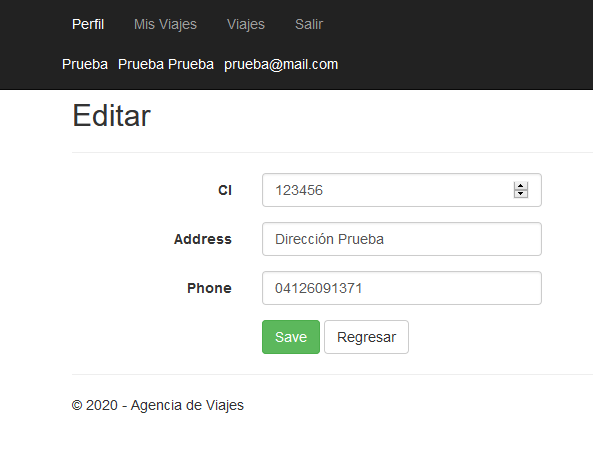
Luego de hacer la reserva nos envía a la vista de los viajes



Donde podemos ver los detalles y un reporte del viaje en PDF.

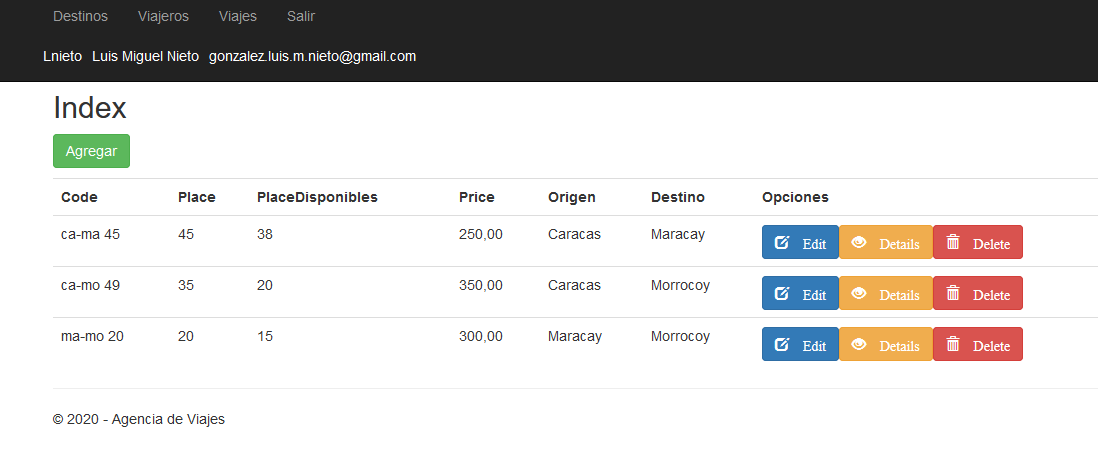


En la sección de perfil podemos cambiar los datos

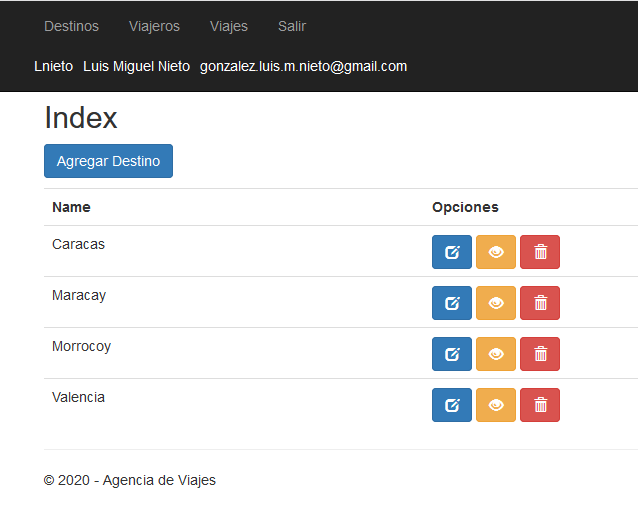


Usuario Administrador

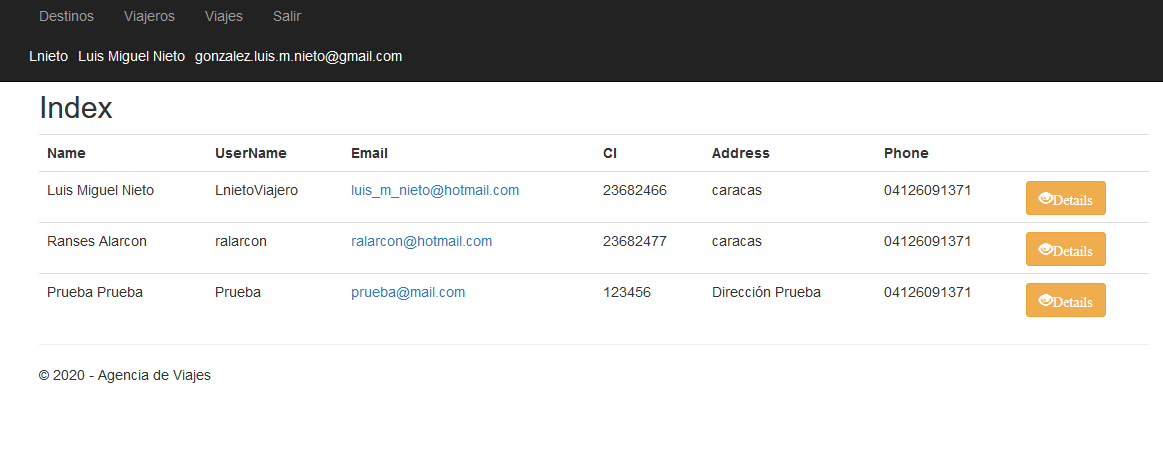
Al ingresar podemos ver todos los viajes disponibles, donde podemos agregar, editar, ver el detalle o eliminar.



En el menú tiene las opciones del destino donde se puede agregar, editar, ver el detalle y eliminar los destinos.



Viajeros, donde puede visualizar los datos de los viajeros y los detalles.



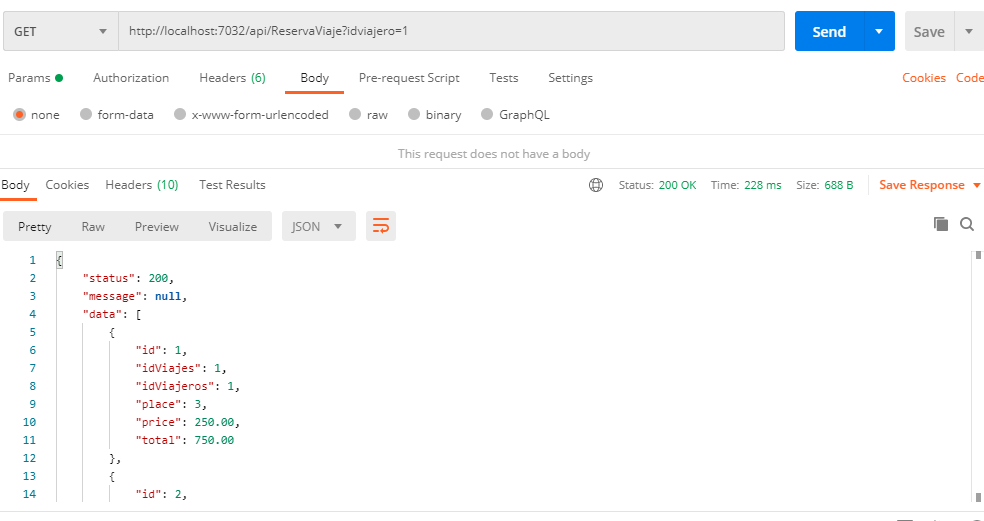
Documentación Api

Reserva de Viajes:

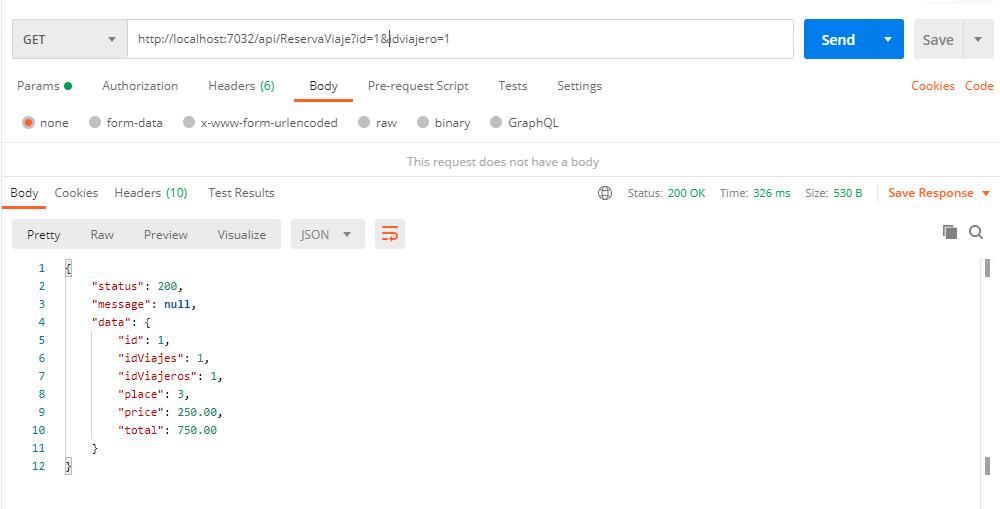
EndPoint: /api/ReservaViaje

Métodos:

* GET (ALL): Parámetros idviajero, trae la lista de todas las reservas de los viajes que tiene el viajero.



* GET: Parámetros id, idviajero; trae la reserva del viaje en específico que tiene el viajero.



* POST : parámetro Objeto ReservaViajero en el body

{

            "id": 0,

            "idViajes": 2,

            "idViajeros": 1,

            "place":3

 }

* PUT: parámetro Objeto ReservaViajero en el body

{

            "id": 5,

            "idViajes": 2,

            "idViajeros": 1,

            "place":3

 }

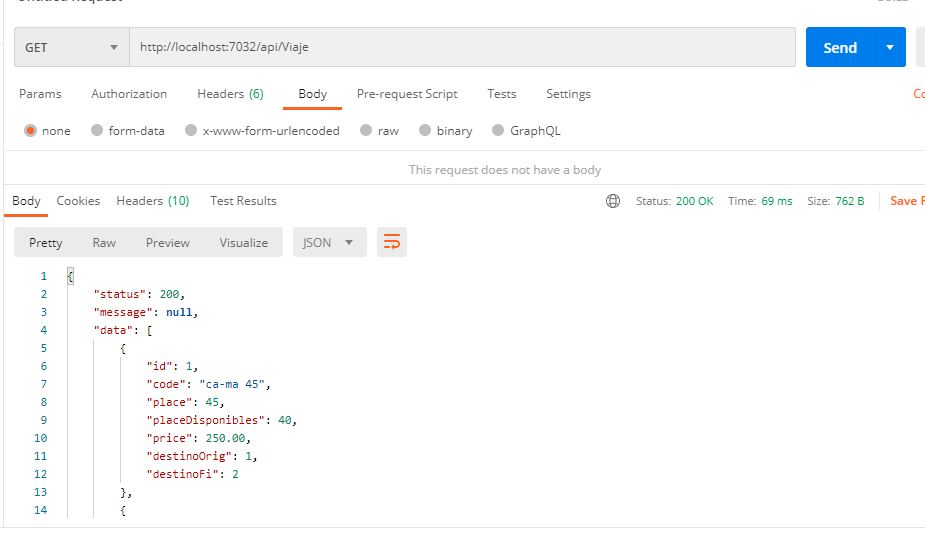
* DELETE: parámetro Id de la Reserva a eliminar.

Viajes:

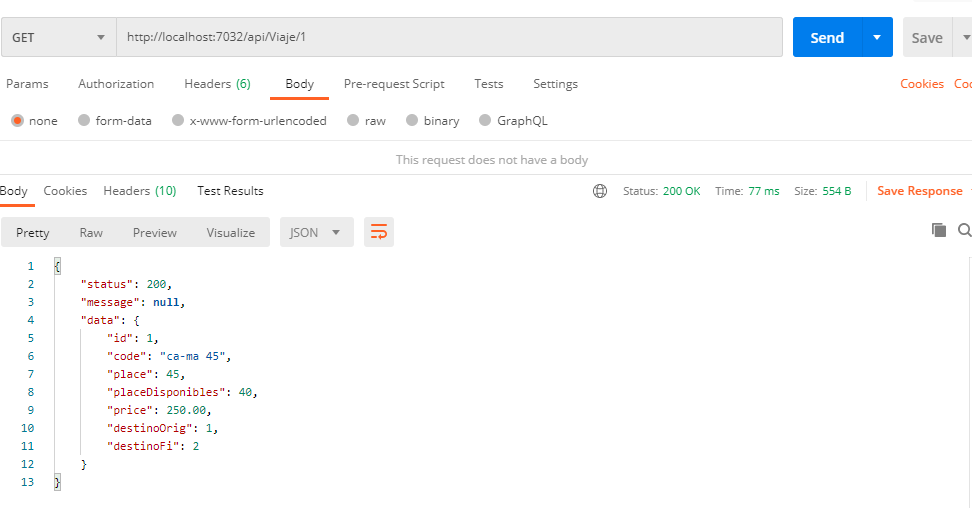
EndPoint: /api/Viaje

Métodos:

* GET (ALL): trae la lista de todos los viajes disponibles.



* GET: Parámetros id; trae el viaje en específico.



* POST : parámetro Objeto viaje en el body

{

        "id":0,

        "code": "ca-ma 45",

        "place": 45,

        "placeDisponibles": 45,

        "price": 250.00,

        "destinoOrig": 1,

        "destinoFi": 2

    }

* PUT: parámetro Objeto viaje en el body

{

        "id":5,

        "code": "ca-ma 45",

        "place": 45,

        "placeDisponibles": 45,

        "price": 250.00,

        "destinoOrig": 1,

        "destinoFi": 2

    }

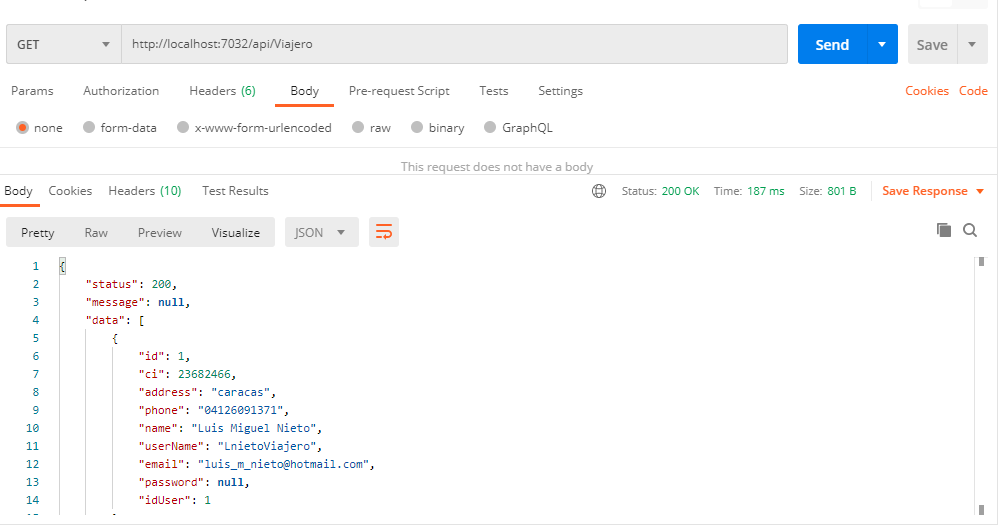
* DELETE: parámetro Id del viaje a eliminar.

Viajero:

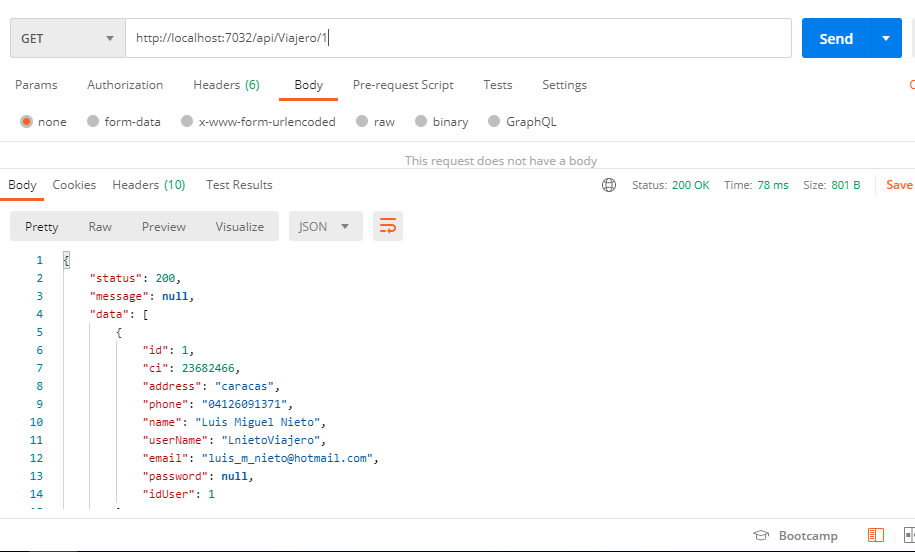
EndPoint: /api/Viajero

Métodos:

* GET (ALL): trae la lista de todos los viajeros disponibles.



* GET: Parámetros id; trae el viajero en específico.



* POST : parámetro Objeto viajero en el body

 {

            "id": 0,

            "ci": 23682466,

            "address": "caracas",

            "phone": "04126091371",

            "name": "Luis Miguel Nieto",

            "userName": "LnietoViajero",

            "email": "luis\_m\_nieto@hotmail.com",

            "password": “**123456”**,

            "idUser": 1

        }

* PUT: parámetro Objeto viajero en el body

 {

            "id": 1,

            "ci": 23682466,

            "address": "caracas",

            "phone": "04126091371",

            "name": "Luis Miguel Nieto",

            "userName": "Lnieto",

            "email": "luis\_m\_nieto@hotmail.com",

            "password": **null**,

            "idUser": 1

        }

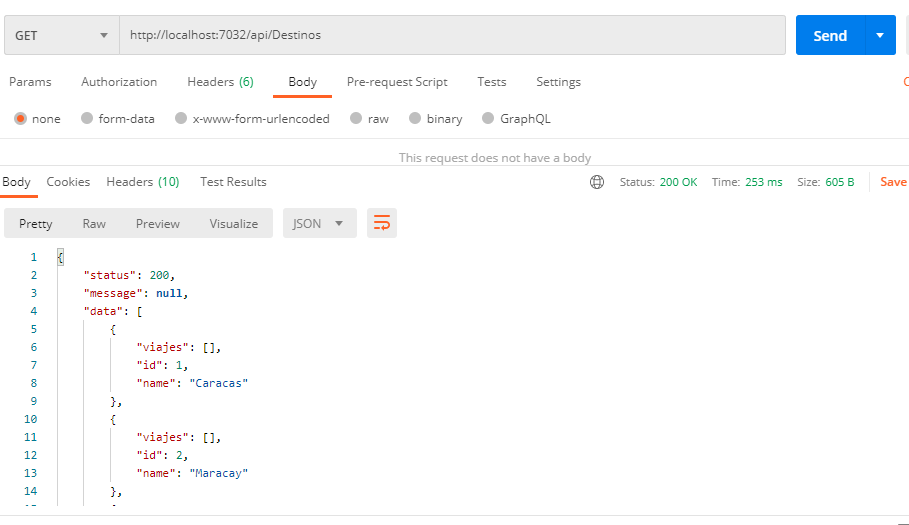
* DELETE: parámetro Id del viajero a eliminar.

Destino:

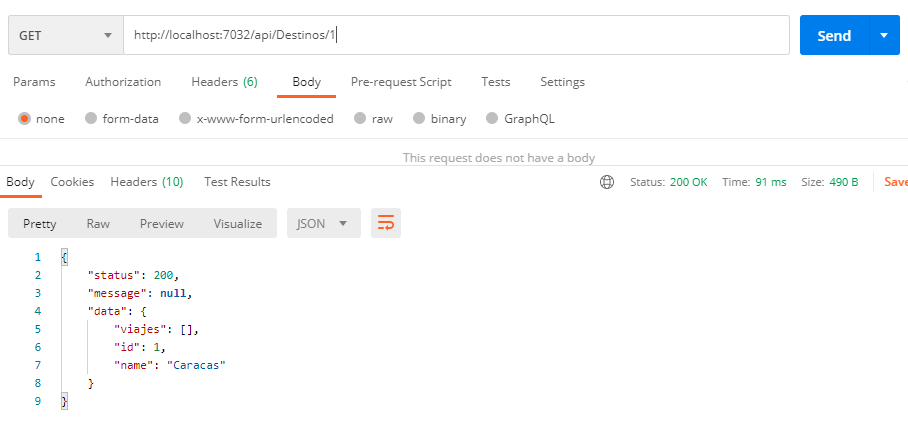
EndPoint: /api/Destinos

Métodos:

* GET (ALL): trae la lista de todos los Destinos disponibles.



* GET: Parámetros id; trae el Destino en específico.



* POST : parámetro Objeto Destino en el body

 {

        "id": 0,

        "name": "Caracas"

  }

PUT: parámetro Objeto Destino en el body

{

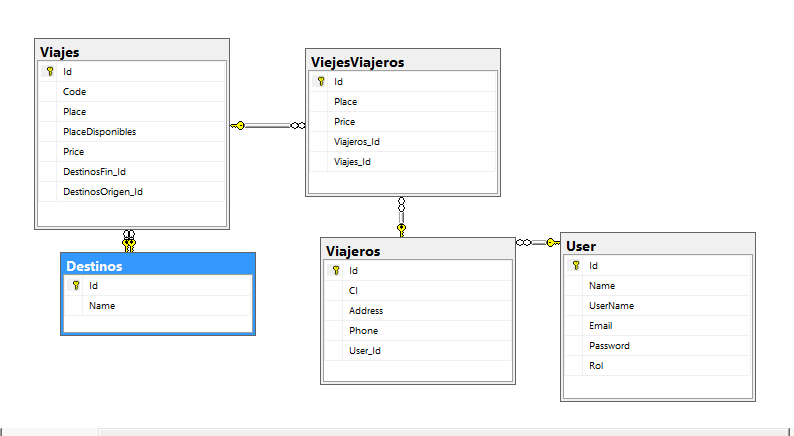
        "id": 1,

        "name": "Valencia"

    }

* DELETE: parámetro Id del Destino a eliminar.

Diseño de la Base de datos



El sistema utiliza E.F. 6.0 code first por lo que solo se debe configurar la dirección y dar los permisos necesarios y se generara automáticamente la Base de datos.