

Contrato de comunicación

Base Url: **REST**

REGISTRAR UN USUARIO -> /userRegister

```
{
  Message_Type: USER_REGISTER,
  Message_Data: {
    Name: {
      F_Name: "nombre_1", -- string
      M_Name: "nombre_2", -- string -- en caso de existir, si no, enviar ""
      L_Name: "apellido_1"
    }
    Email: "email", -- string
    Enrolled: "true-false", -- bool
    University_Name: "nombre_U", -- en caso de que enrolled sea true, si no enviar un ""
    Username: "user", -- string
    Password: "user- pwd" -- tiene que cumplir las condiciones de seguridad
    Carnet: "carnet" -- string -- en caso de no existir, pasar ""
    Phone_Number: #telefono --int
  }
}
-- returns response
{
  Message_Type: RESPONSE
  Message_Data: REGISTERED
}
```

BUSCAR VUELOS -> /flightSearch

```
{  
  Message_Type: FLIGHT_SEARCH  
  Message_Data:{  
    Starts: [Airport code, City_Name]  
    Ends: [Airport code, City_Name]  
  }  
}
```

Returns – List of flights

```
{  
  Message_Type:LIST_OF_FLIGHTS  
  Message_Data:  
    [  
      {Starts: [Airport code, City_Namem, Coutry], Ends: [Airport code, City_Name,  
        Country], Trip_ID: “trip_id”},  
      .  
      .  
      .  
    ]  
}
```

Reserva de vuelos -> FlightBooking

```
{  
  Message_Type: FLIGHT_BOOKING  
  Message_Data: {  
    Trip_ID: “trip_ID”  
  }  
}
```

Returns response

```
{  
Message_Type: RESPONSE  
Message_Data: BOOKED  
}
```

PROMOCIONES – PROMOS

```
{  
Message_Type: PROMOS  
Message_Data: GET_PROMOS  
}
```

Returns – list of promos

```
{  
Message_Type: LIST_OF_PROMOS  
Message_Data: - list of promos  
  [  
    {  
      Promotion_Name: "promoName",  
      Promotion_ID: "promoID",  
      Start_Date: "startDate",  
      End_Date: "endDate",  
      Starts:  
        [  
          Airport code,  
          City_Name,  
          Coutry  
        ],  
      Ends:  
        [  
          Airport code,  
          City_Name,
```

```
        Country
      ],
      Trip_ID: "trip_id"
    }
  .
  .
  .
]
}
```

Chequeo de pasajeros - Trip

```
{
Message_Type: TRIP
Message_Data:
  {
    Trip_ID: "tripID"
  }
}
```

Returns trip info

```
{
Message_Type: TRIP_INFO
Message_Data:
  {
    Plane_ID: "planeID"
    Type_ "planeType"
    Number_of_Seats: "#seats"
  }
}
```

Check-In de un vuelo - CHECK

```
{
```

Message_Type: CHECK

Message_Data: -

```
{  
  Trip_ID: "tripID"  
  Seat_Number: "seat#"  
  Type: "seat_Type"  
  UserName: "username"  
  Passport: "#pasaporte"  
  – se supone que el cliente ya debe estar registrado, solo hace falta el user o el pasaporte,  
  si se llena uno, dejar el otro en ""  
}
```

Returns paying info

```
{  
  Message_Type: PAYING  
  Message_Data:  
    {  
      Type: "seat_class"  
      Price: "seatPrice"  
      Boarding_ID: "BoardingID"  
    }  
}
```

Paying-Method - PAYING

```
{  
  Message_Type: PAYING  
  Message_Data:  
    {  
      Boarding_ID: "board_ID"  
      Card_Number: "card#"  
    }  
}
```

```
        In_Card_Name: "cardName"
        Exp.Date: "expDate"
    }
}
```

Returns paid response

```
{
    Message_Type: RESPONSE
    Message_Data: PAID
}
```

Registro de vuelos - FREGISTER

```
{
    Message_Type: FLIGHT_REGISTER
    Message_Data:
        {
            Plane_ID: "planeID"
            Number_of_Seats: "#seats"
            Type: "seatType" – maybe a range of seats . dunno
            Starts: [Airport code, City_Name, Coutry]
            Ends: [Airport code, City_Name, Country]
        }
}
```

Returns RESPONSE

```
{
    Message_Type: RESPONSE
    Message_Data: "Successfully Registered"
}
```

```
{  
  Message_Type:  
  Message_Data:  
}
```