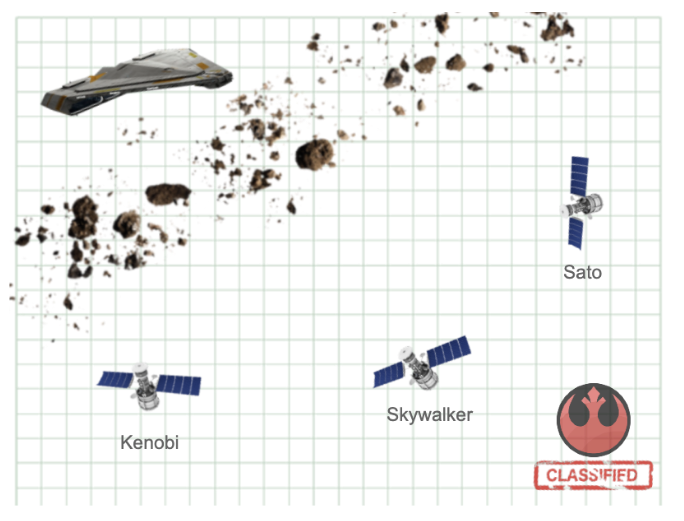
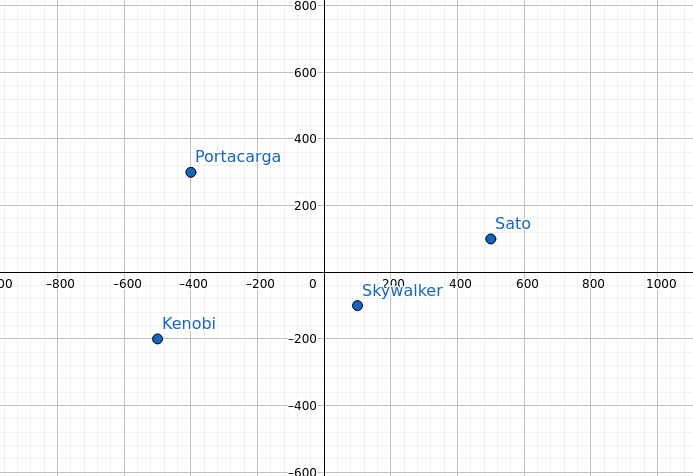
**Documentación Challenge Quasar, para el ingreso de C Emmanuel Moreno Torres a Mercado Libre.**

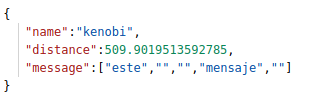
El sistema debe calcular la posición y el mensaje secreto de la nave portacarga imperial que está emitiendo un llamado de auxilio a 3 satélites de la Alianza Rebelde.



La imagen anterior se puede representar en plano cartesiano de la siguiente forma:

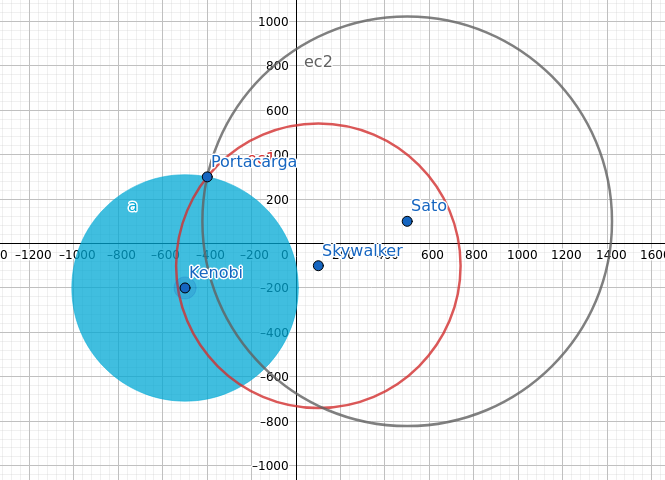


La información disponible que se tiene para dicho cálculo es la distancia y el mensaje incompleto del portacarga imperial hacia cada satélite, representado de la siguiente manera:



Donde el campo “**name**” representa el satélite a donde se está enviando la información, “**distance**” es la distancia a la que se encuentra la nave portacarga imperial del satélite, y “**message**” un arreglo de cadenas con el mensaje incompleto, éste debe de tener la misma longitud en cada uno de los 3 llamados.

Para realizar el cálculo de las coordenadas del portacarga imperial, se decidió graficar una circunferencia al rededor de cada satélite, usando como radio la distancia al portacarga imperial, así, el punto de intersección de los 3 círculos, es la posición del portacarga imperial.



En éste ejemplo, usaremos los valores:

name: "skywalker"

distance: 640.3124237432849

message: ["","es","","","secreto"]

name: "sato"

distance: 921.9544457292888

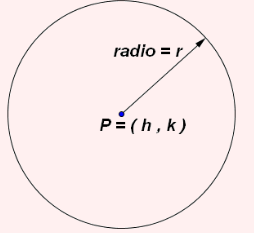
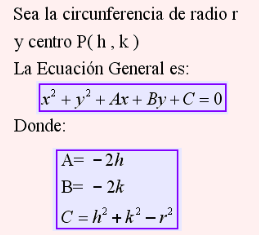
message: ["este","","un","",""]

name: "kenobi"

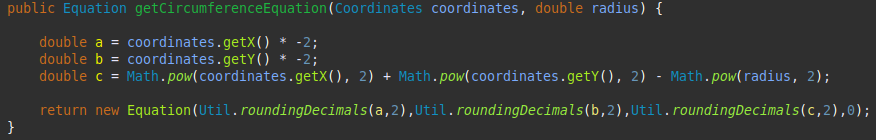
distance: 509.9019513592785

message: ["este","","","mensaje",""]

Para comenzar, se debe obtener la ecuación general de la circunferencia de cada satélite.



Ésto se hace con un método genérico (X² y Y² al siempre ser 1, se omitirán en el mapeo de los pojos en todo el sistema).



Una vez teniendo las ecuaciones de la circunferencia de los 3 satélites, se calculan las intersecciones entre ellos, que en éste caso son Kenobi → Skywalker, Skywalker → Sato y Kenobi → Sato.

Para éste ejemplo, la ecuación de la circunferencia Kenobi es la siguiente:

x² + y² +1000x + 400y + 30000

Y la de Skywalker es la siguiente:

x² + y² - 200x + 200y - 390000

Para la intersección entre dos circunferencias, primero se suman las ecuaciones generales de la circunferencia, ésto se hace multiplicando la segunda ecuación (-1) y efectuando la suma.

x² + y² +1000x + 400y + 30000

-x² - y² + 200x - 200y + 390000

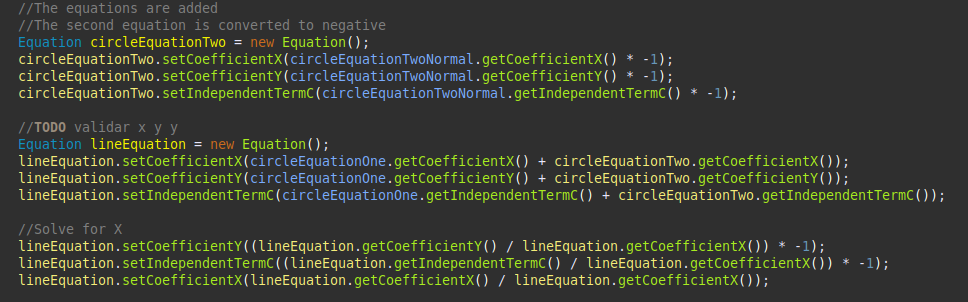
----------------------------------------------

1200x + 200y + 420000

Luego se despeja x:

1200x = - 200y -420000

x = - 0.16666y - 350

Dentro del método genérico el código hasta éste punto es el siguiente:

Ahora la variable x se sustituye en la primera ecuación de la circunferencia que obtuvimos, en éste caso la del satélite “kenobi”

(- 0.16666y - 350)² + y² +1000(- 0.16666y - 350) + 400y + 30000

Se resuelve el término cuadrático (- 0.16666y - 350)² mediante la formula:

(a + b)² = a² + 2ab + b²

Por lo tanto:

(- 0.16666y - 350)² = 0.0277y² + 116.662y + 122500

Y se resuelve la segunda operación 1000(- 0.16666y – 350) multiplicando los términos:

1000(- 0.16666y – 350) = -166.66y – 350000

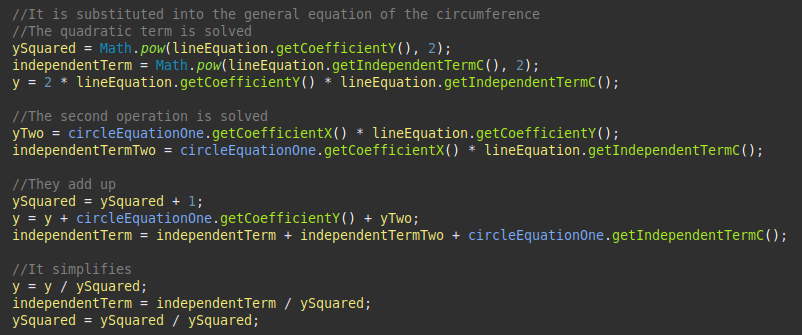
Con lo cual quedaría la ecuación completa de la siguiente manera:

0.0277y² + 116.662y + 122500 + y² - 166.66y – 350000 + 400y + 30000

Y ya resuelta y simplificada:

1.0277y² + 350.002y – 197500 → y² + 340.57y – 192176.71

Dentro del método genérico el código hasta éste punto es el siguiente:



Dada una ecuación de segundo grado:

y² + 340.57y – 192176.71

Se procede a resolver por medio de la fórmula general.

- b ± √(b² -4ac)

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

2a

Se obtiene el discriminante:

(340.57)² – 4 (1) (-192176.71) = 891606.1461

Se continua con el desarrollo de la fórmula:

- 340.57 ± √(891606.1461)

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

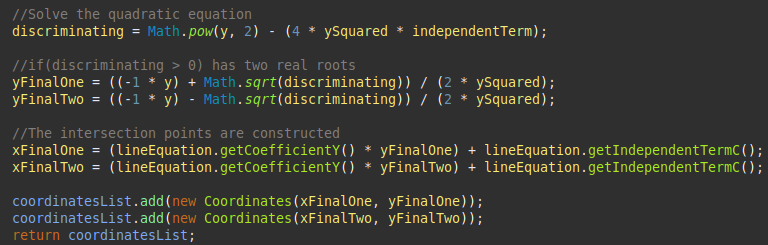
2

Se resuelven las 2 incógnitas, obteniendo los dos puntos de intersección entre las circunferencias:

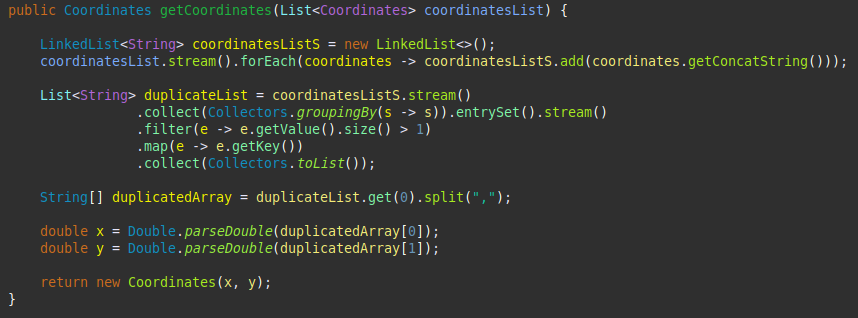
x1 = - 400 x2 = - 243

y1 = 300 y2 = - 640

Dentro del método genérico el código hasta éste punto es el siguiente:



Al tener los puntos de intersección, éstos se agregan a una lista, y ya teniendo los cálculos de los las intersecciones de las 3 circunferencias, se procede a compararlos, y el punto que más se repita, es el punto donde se encuentra la nave portacarga imperial.



**Diagrama ER**

El sistema cuenta con seguridad JWT implementada con Spring security, para lo cual se designaron 2 roles: “ADMIN” y ”USER”, con el cual el administrador puede crear otros usuarios de tipo USER para que puedan consumir los endpoints del sistema.

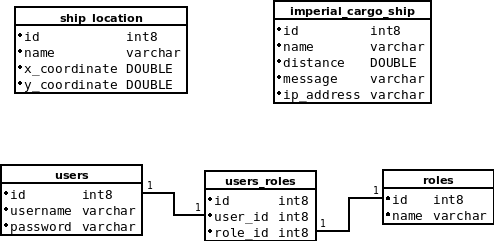


Tabla: ship\_location

Descripción: Contiene las coordenadas y los satelites principales del sistema.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clave | Nombre | Campo | Tipo | Descripción |
| PK | Identificador | id | int8 | Número identificador del registro |
|  | Nombre | name | varchar | Nombre del satélite al que va dirigido |
|  | Distancia | distance | DOUBLE | Distancia del satélite |
|  | Mensaje incompleto | message | varchar | Arreglo de cadenas, del mensaje incompleto |
|  | Dirección IP | ip\_address | varchar | Dirección IP del cliente |

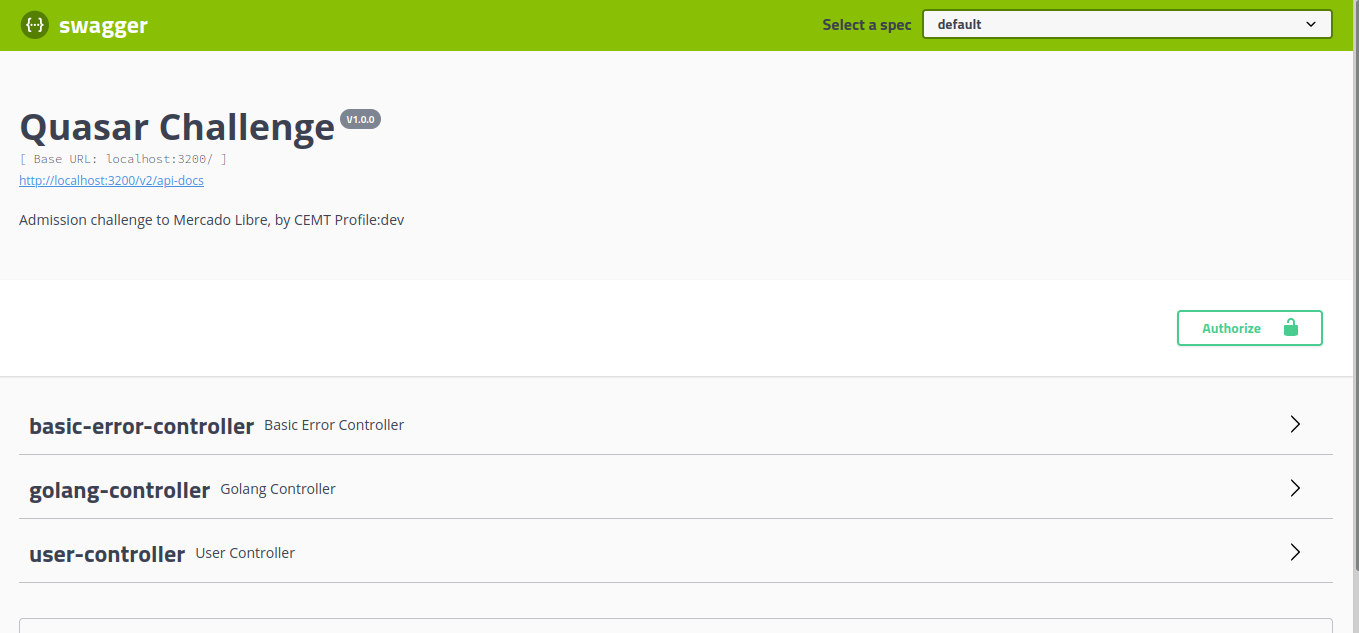
Tabla: imperial\_cargo\_ship

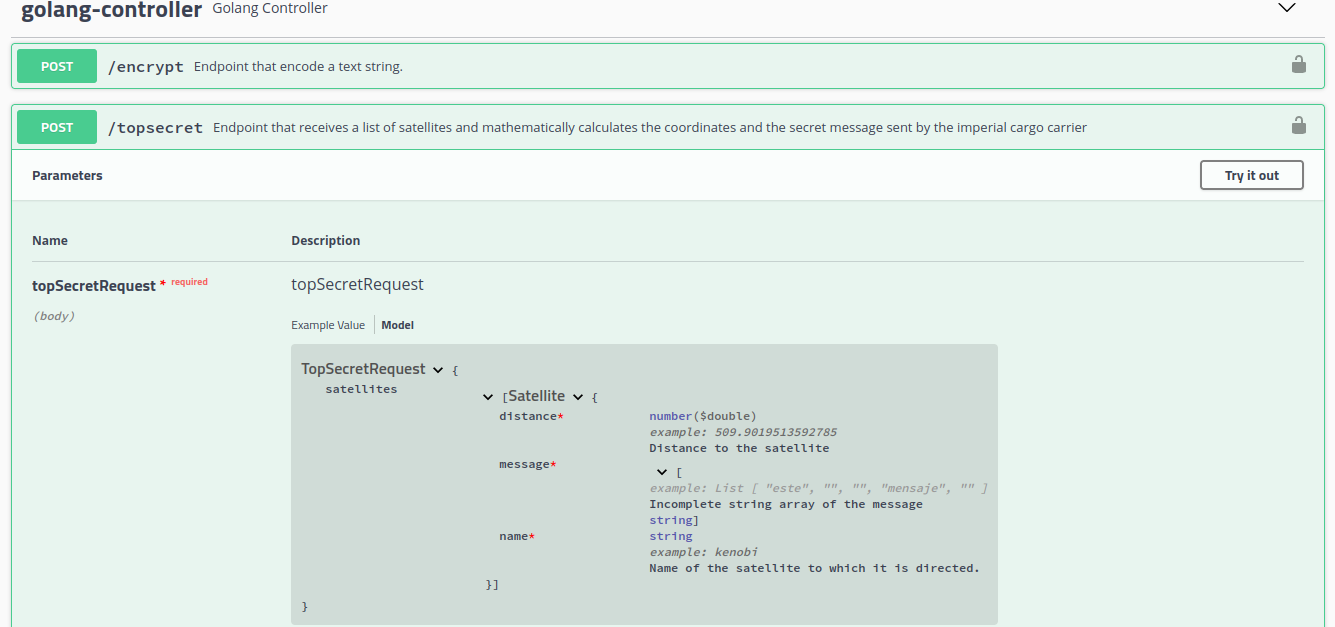
Descripción: Contiene la información del portacarga por cada satélite, para hacer los llamados independientes.

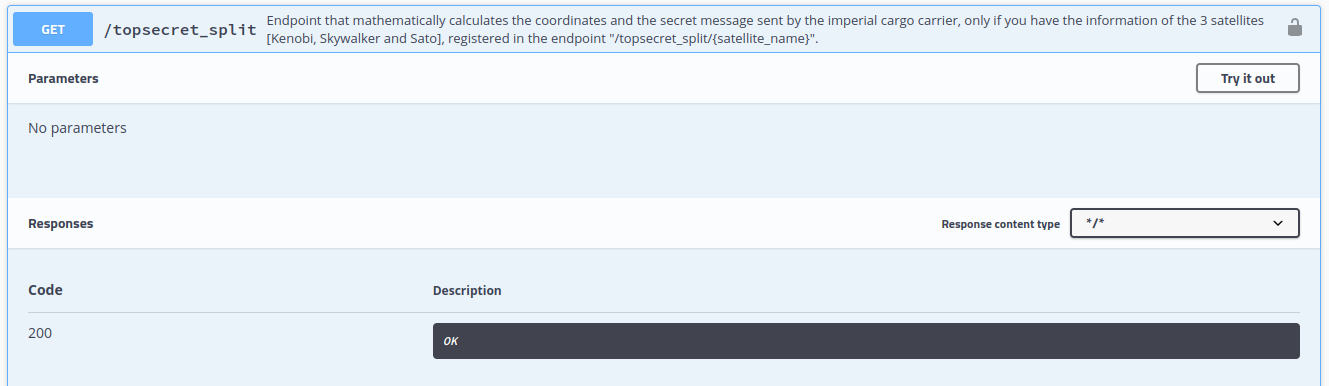
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clave | Nombre | Campo | Tipo | Descripción |
| PK | Identificador | id | int8 | Número identificador del registro |
|  | Nombre | name | varchar | Nombre del satélite |
|  | Coordenada en x | x\_coordinate | DOUBLE | Coordenada en x de la nave |
|  | Coordenada en y | y\_coordinate | DOUBLE | Coordenada en y de la nave |

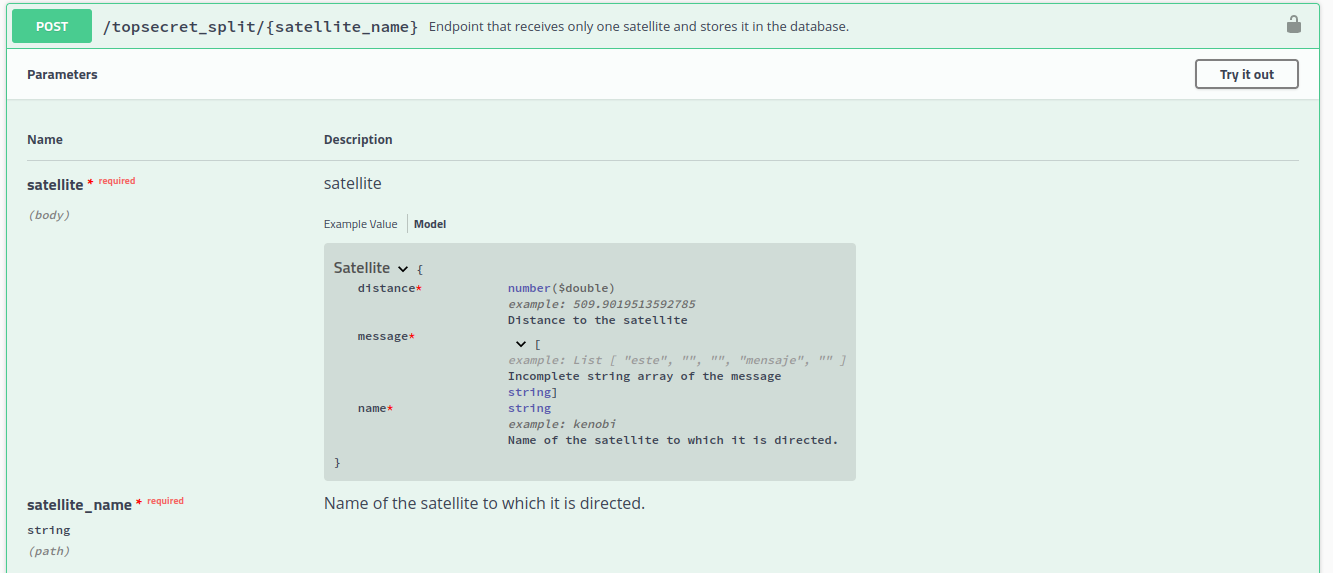
**Swagger2**

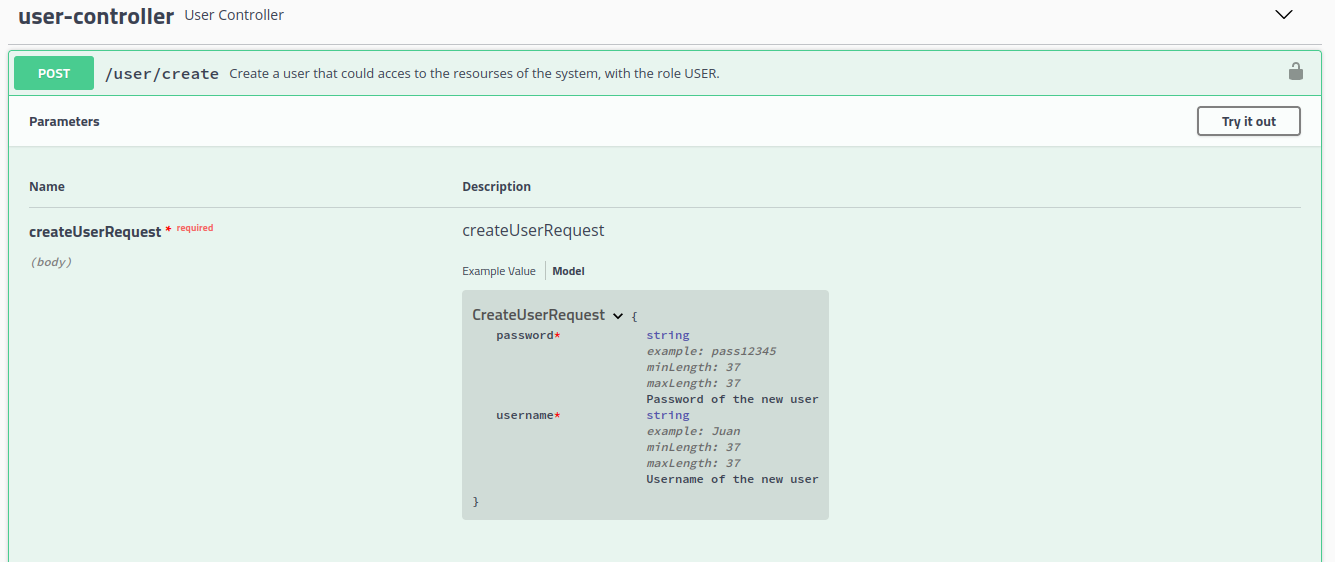
La documentación de los endpoints se realizó con la librería Swagger2





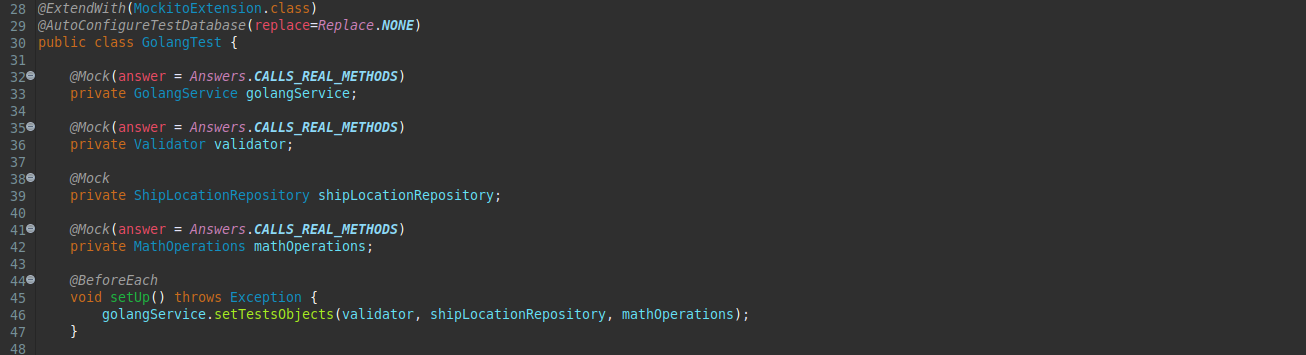






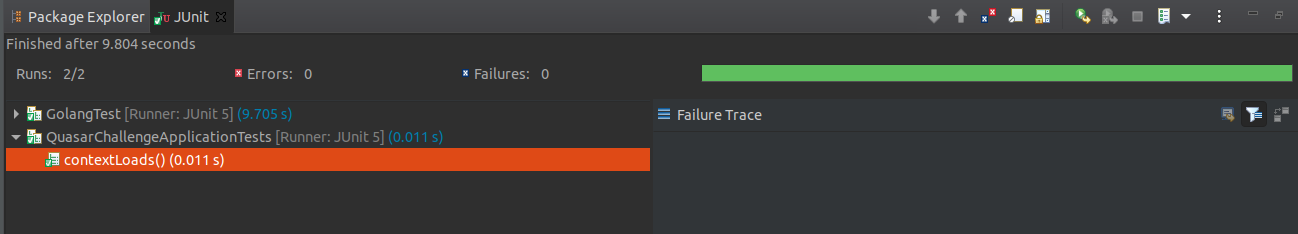
**Testing**

La ejecución de pruebas unitarias se realizó con Junit y Mockito, de la siguiente manera:





Y éstas corrieron de forma satisfactoria:





**Documentación adicional**

La documentación relacionada a **Javadoc** se encuentra en la ruta del proyecto:

*/doc/index.html*

Dentro del siguiente archivo se puede encontrar información relevante al funcionamiento del sistema:

*/README.md*

El proyecto se encuentra desplegado en la URL (incluyendo swagger):

*https://quasarchallenge.uc.r.appspot.com/swagger-ui.html*

Se puede importar la colección de postman desde el archivo:

*/Challenge.postman\_collection.json*