**Evidencias test-drive**

**Consideraciones iniciales:**

Por default el sistema al iniciar crea un Auto.

Se asumió que la duración de la reserva es de 1 día. Ya que en el enunciado no se aclaraba la duración mínima y además se indicaba que desde que se devuelve el auto hasta que se entrega al próximo cliente debe pasar un tiempo, el cual no aclaraba. Por lo que se asumió que desde las 10hs que se entrega el auto a la empresa hasta las 18hs que se entrega el auto al próximo cliente, es el tiempo de procesamiento de datos.

Tampoco se aclaraba que cuando se creaba una reserva que se debía enviar la duración que uno quería reservar, por lo que se entiende que el sistema la define.

Al crear la reserva el sistema automáticamente define la hora de retiro y de entrega del auto, como el día de entrega.

Todos los endpoints en /api/v1/cars, /api/v1/reservation y /api/v1/users

Tienen las operaciones basicas CRUD

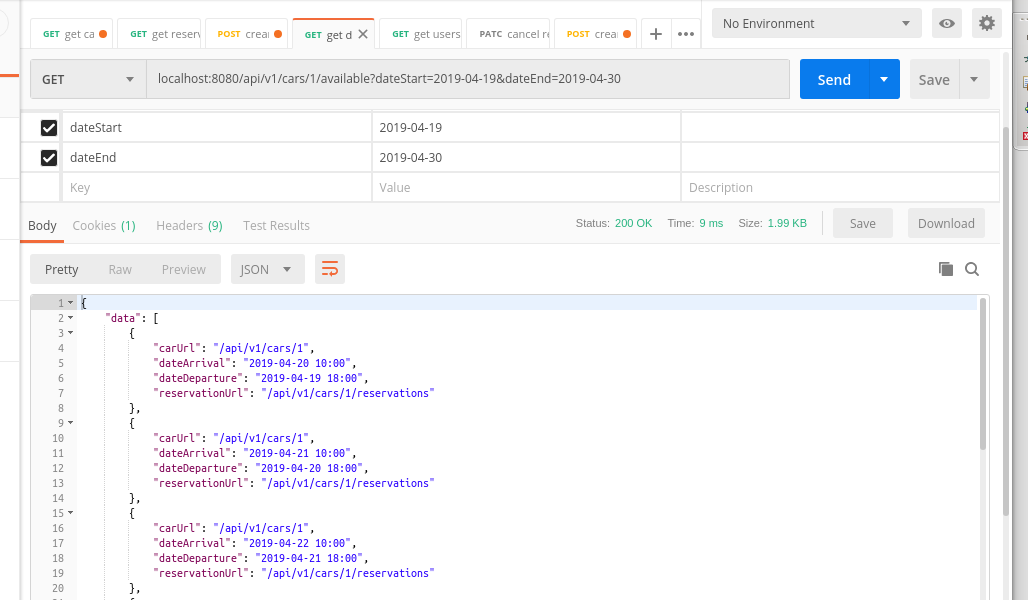
De una reserva solo se puede modificar el estado por canceled.

Todos los getAll muestran los resultados paginados.

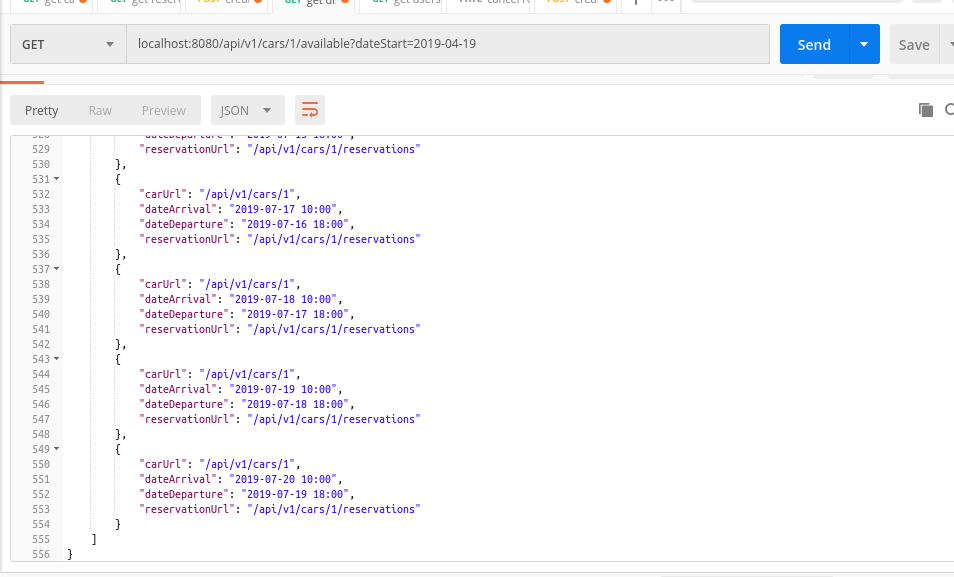
**Endopints Importantes:**

Para ver la disponibilidad de un auto, la fecha desde y hasta no son obligatorias.

Si no se indica la fecha hasta se calcula como 3 meses.



Sin fecha hasta:

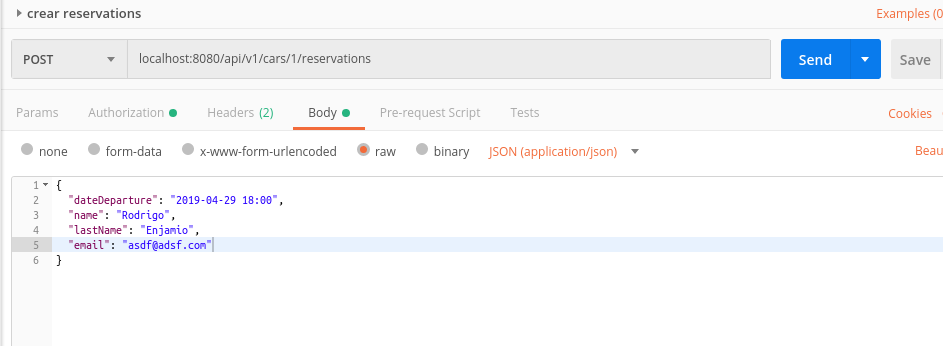


Crear una reserva para el auto id 1:

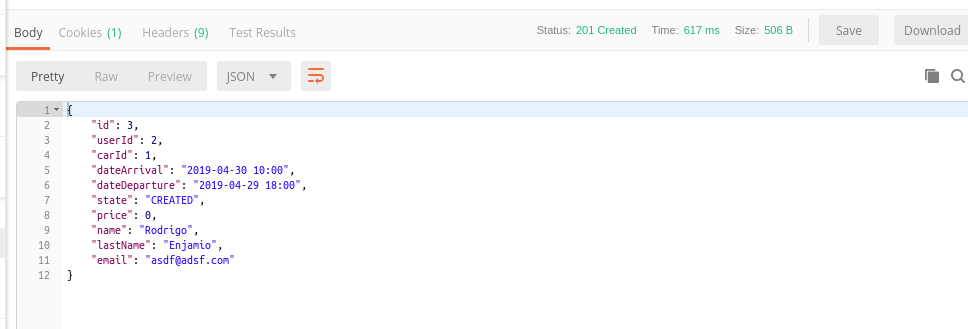
[/api/v1/cars/{idCar}/reservations](http://localhost:8080/swagger-ui.html" \l "/operations/reservation-controller/reservationCarUsingPOST)

o

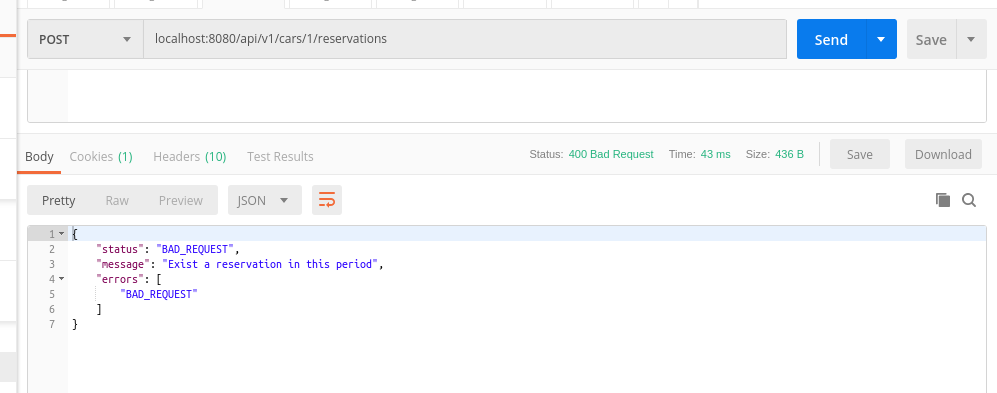
[/api/v1/reservations](http://localhost:8080/swagger-ui.html" \l "/operations/reservation-controller/reservationCar2UsingPOST) (y en el json enviar el campo carId={idCar})



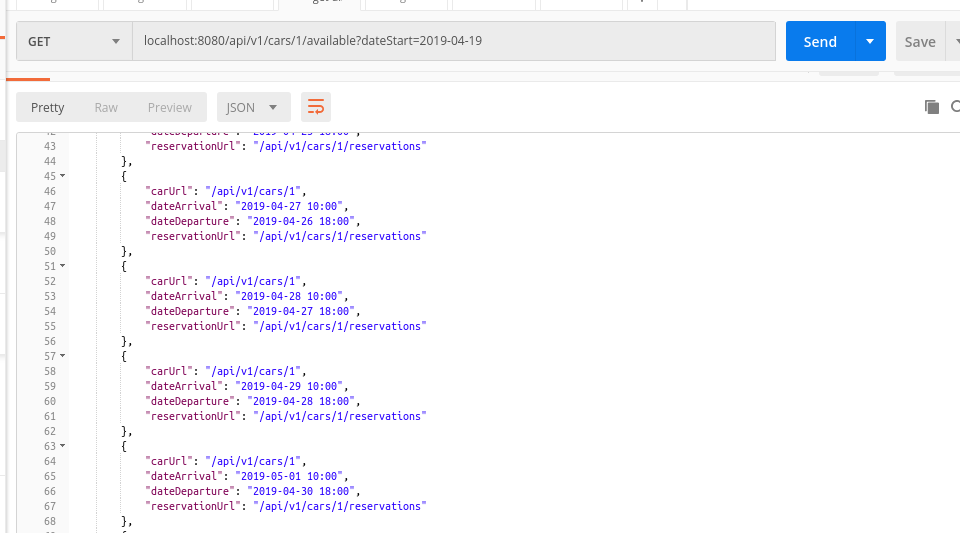
Respuesta:



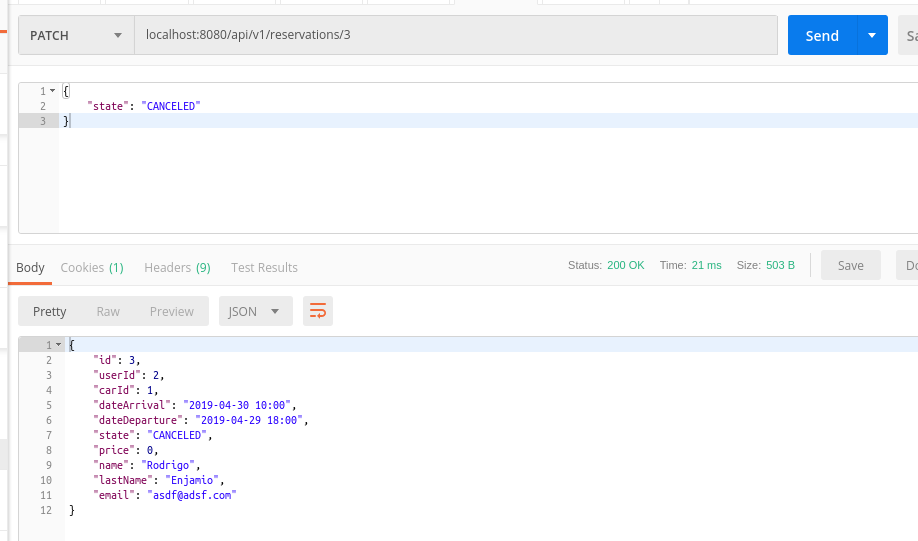
Crear una reserva en la misma fecha



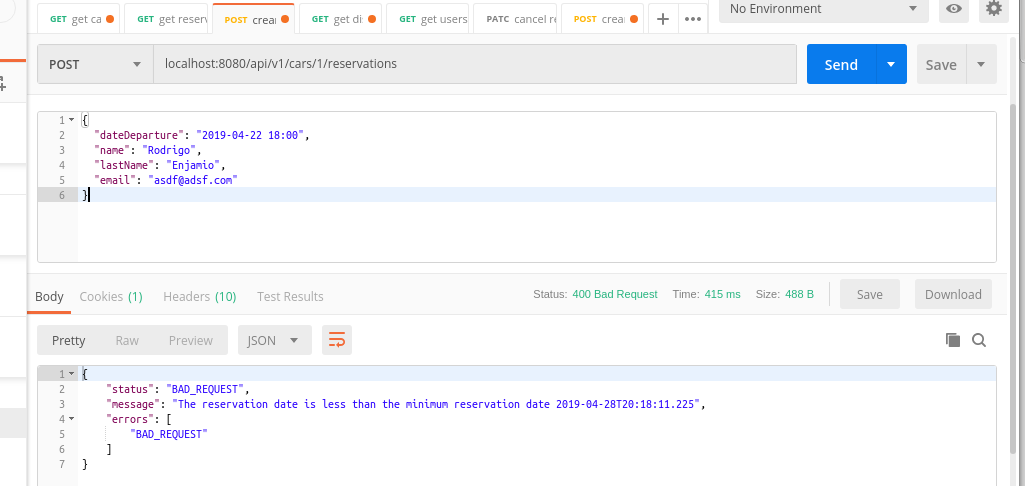
Disponibilidad con una reserva creada el 29-04-2019



Cancelar una reserva



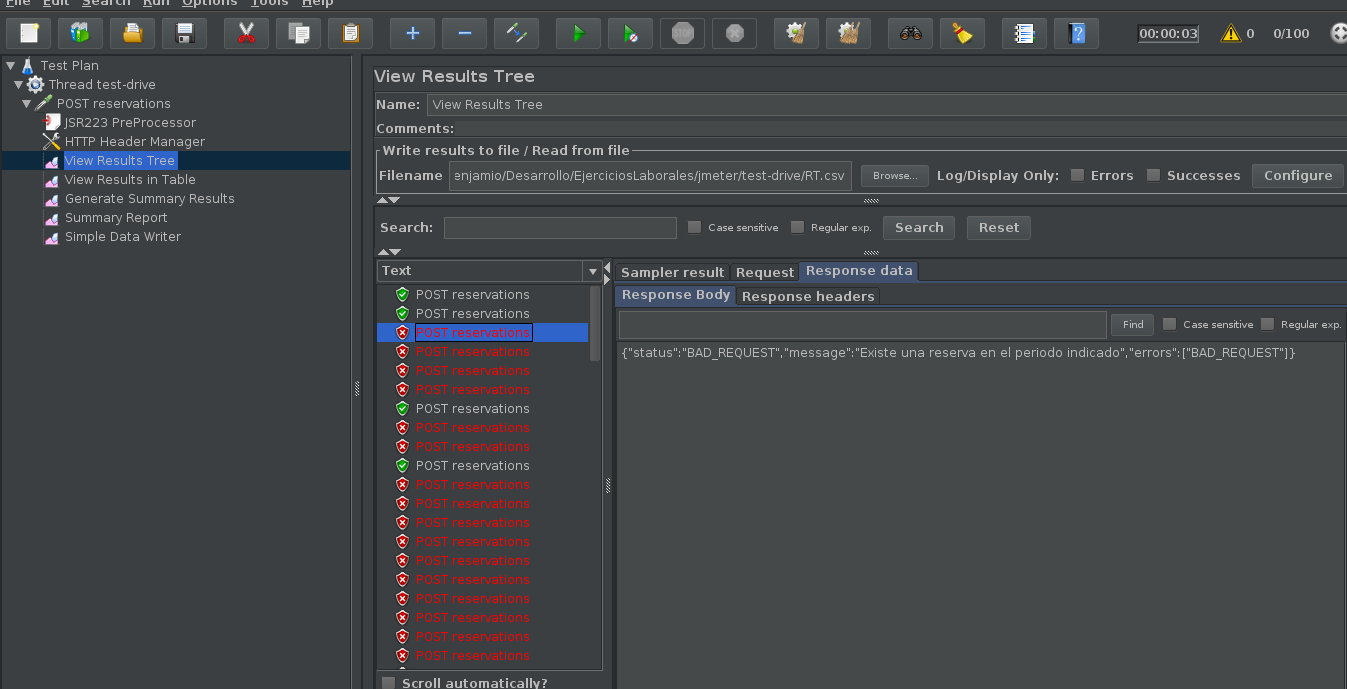
Crear una reserva con fecha menor a una semana



Crear una reserva con fecha mayor a 1 meses:



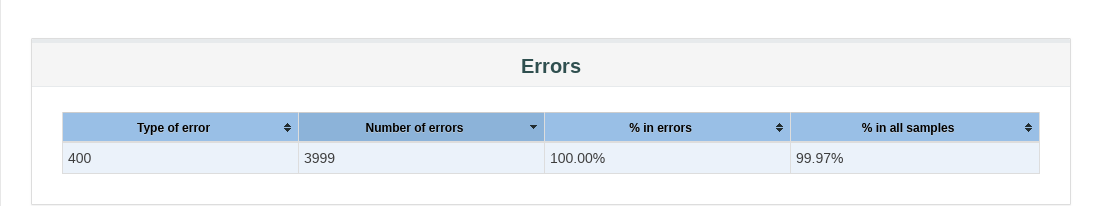
Para verificar la concurrencia ejecute un test con 1000 request cada 1 segundo en los cuales en 4 genero la misma reserva.



Para solucionar este problema hice que el método sea syncronized y para asegurarme cree un constraint en la BD por los campos dateDeparture y car\_id.

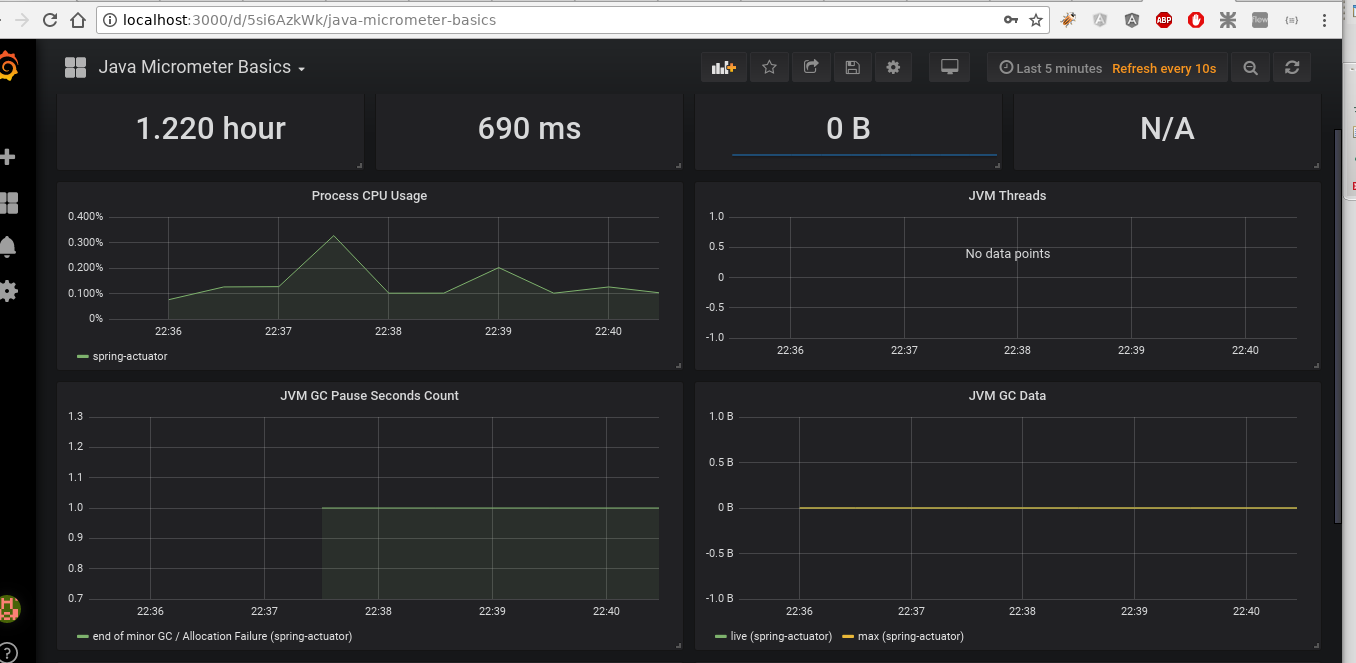
Realice una prueba con 4000 request y no se produzco el error.





Monitorización

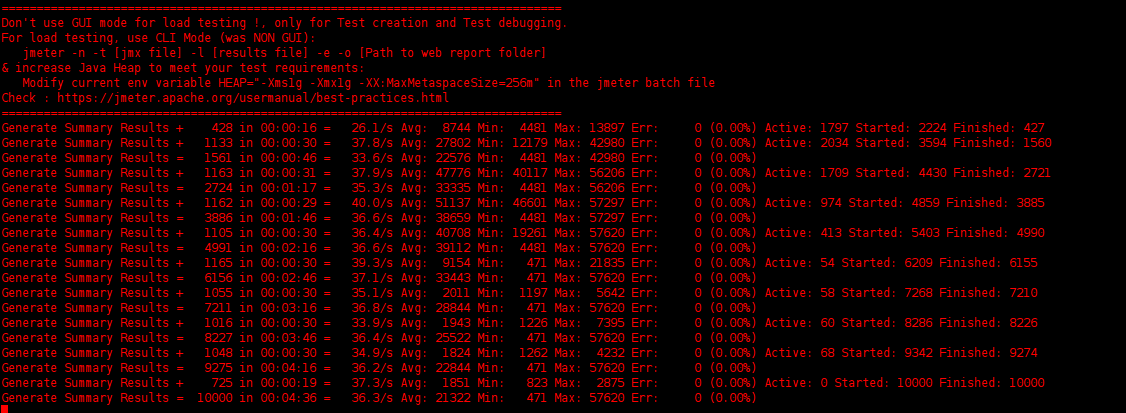
Prometheus y Grafana



Test de concurrencia con jmeter:

GET a /api/v1/cars/1/available

10000 Request cada 1 segundo, sin errores.



Actuator:

<http://localhost:8080/actuator>

Swagger:

<http://localhost:8080/swagger-ui.html>

Coverage:

