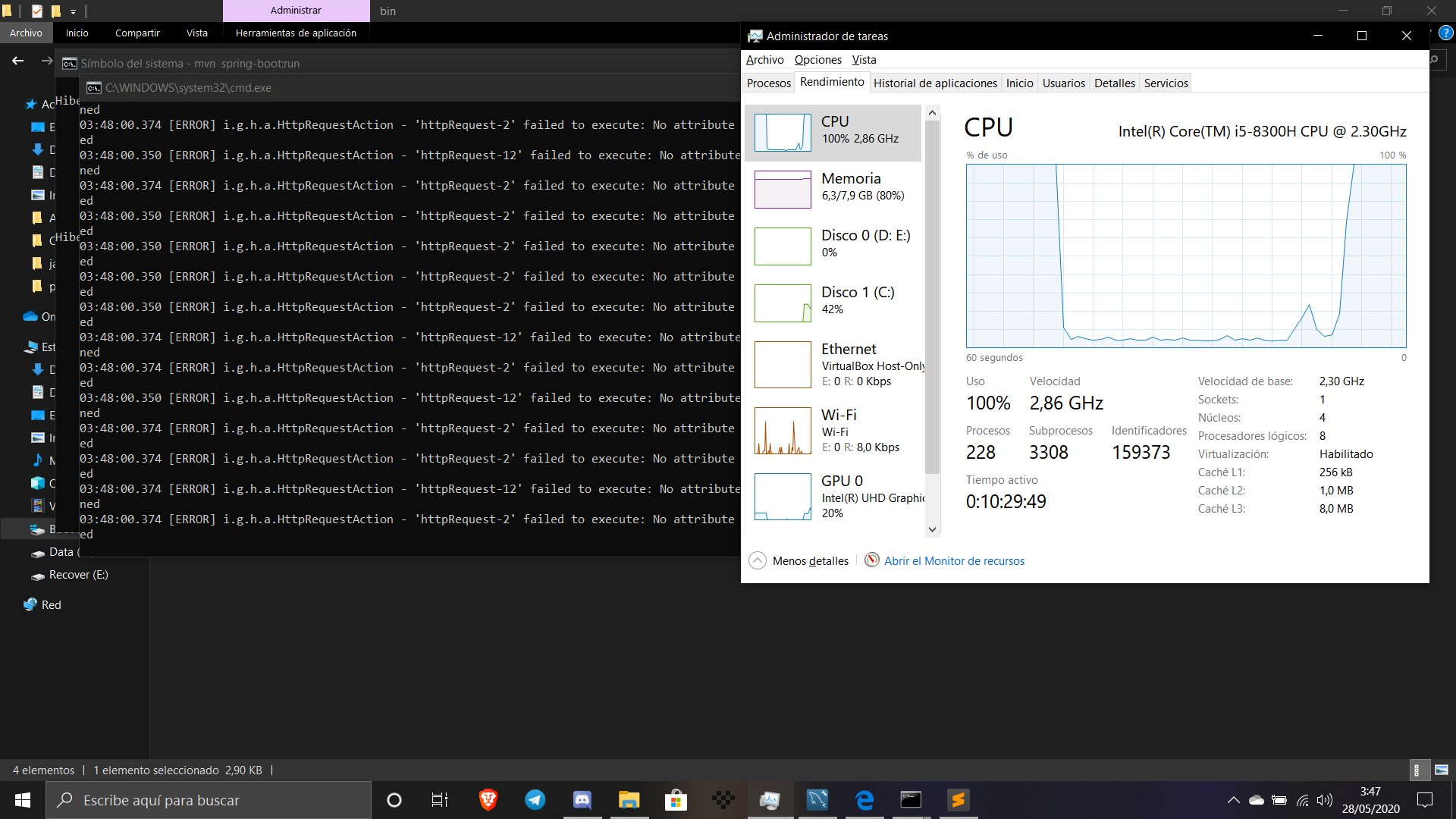
## HU-01: Agregar Paciente

Para la siguiente HU se proponen los siguientes dos escenarios:

* Creación exitosa de un paciente en el sistema:
  + El usuario accede a Home
  + El usuario se identifica en el sistema
  + El usuario crea un paciente nuevo con éxito cumplimentando todos los campos requeridos
* Creación fallida de un paciente en el sistema:
  + El usuario accede a Home
  + El usuario se identifica en el sistema
  + El usuario intenta crear un paciente sin datos de contacto (Email, Domicilio o Teléfono) pero es devuelto al formulario.

### Prueba estrés:

Se realizó una prueba de estrés que introdujo 9300 usuarios en cada escenario (Un total de 18600 usuarios), con un límite de tiempo de 10 segundos para introducirlos todos.



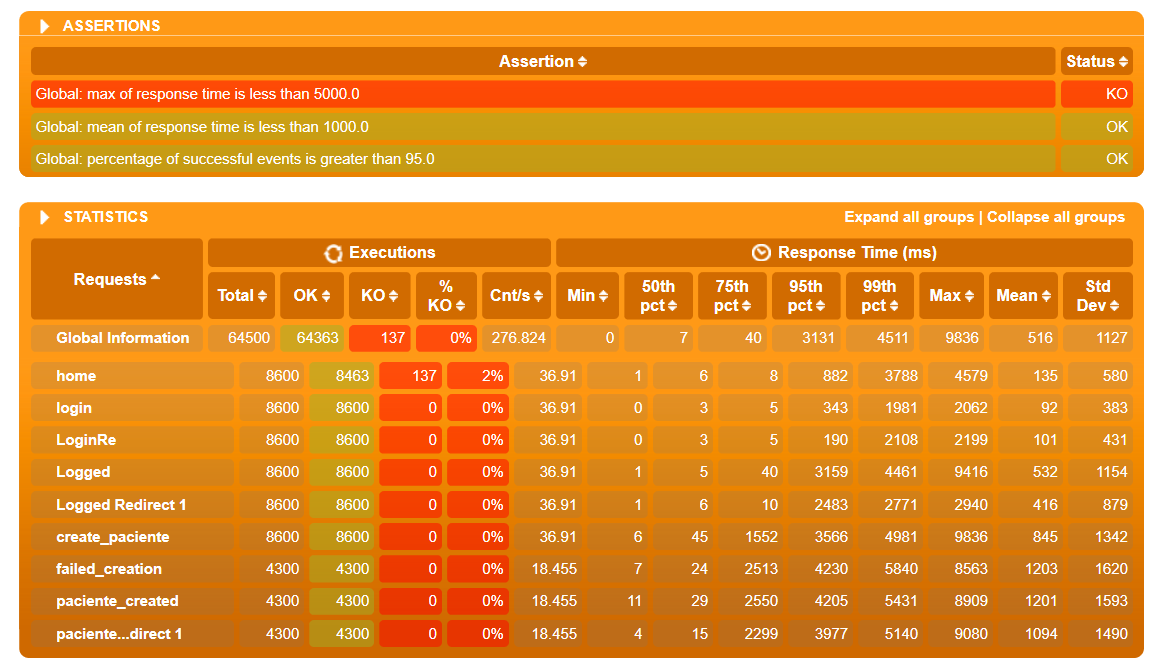
Como observamos, la CPU al alcanzar su máximo uso provoca que el sistema comience a generar tickets csrf incorrectos durante las operaciones post (Como se observa en la ejecución de Gatling) o directamente no sea capaz de generarlos. Seguramente el alto uso de memoria RAM también influyese en la generación de dichos tickets.

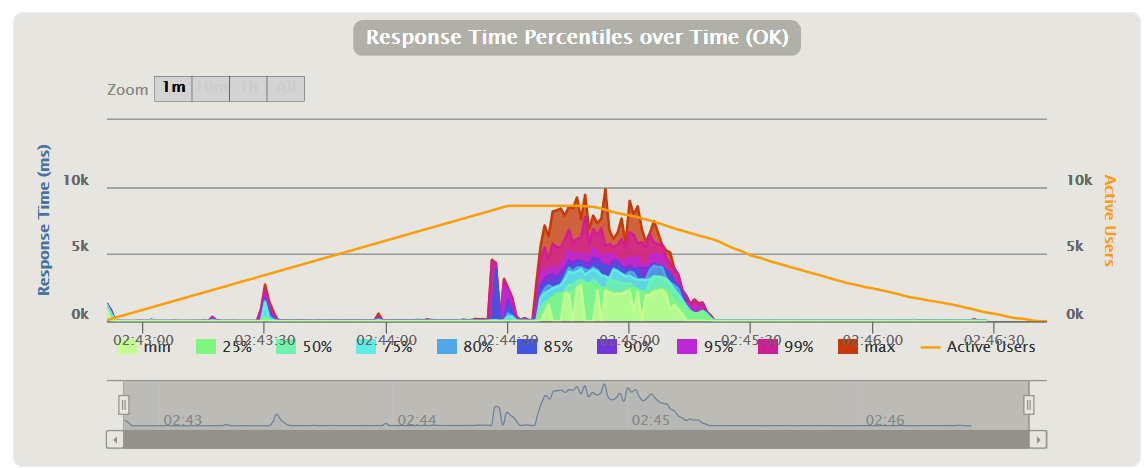
### Prueba de buen rendimiento:

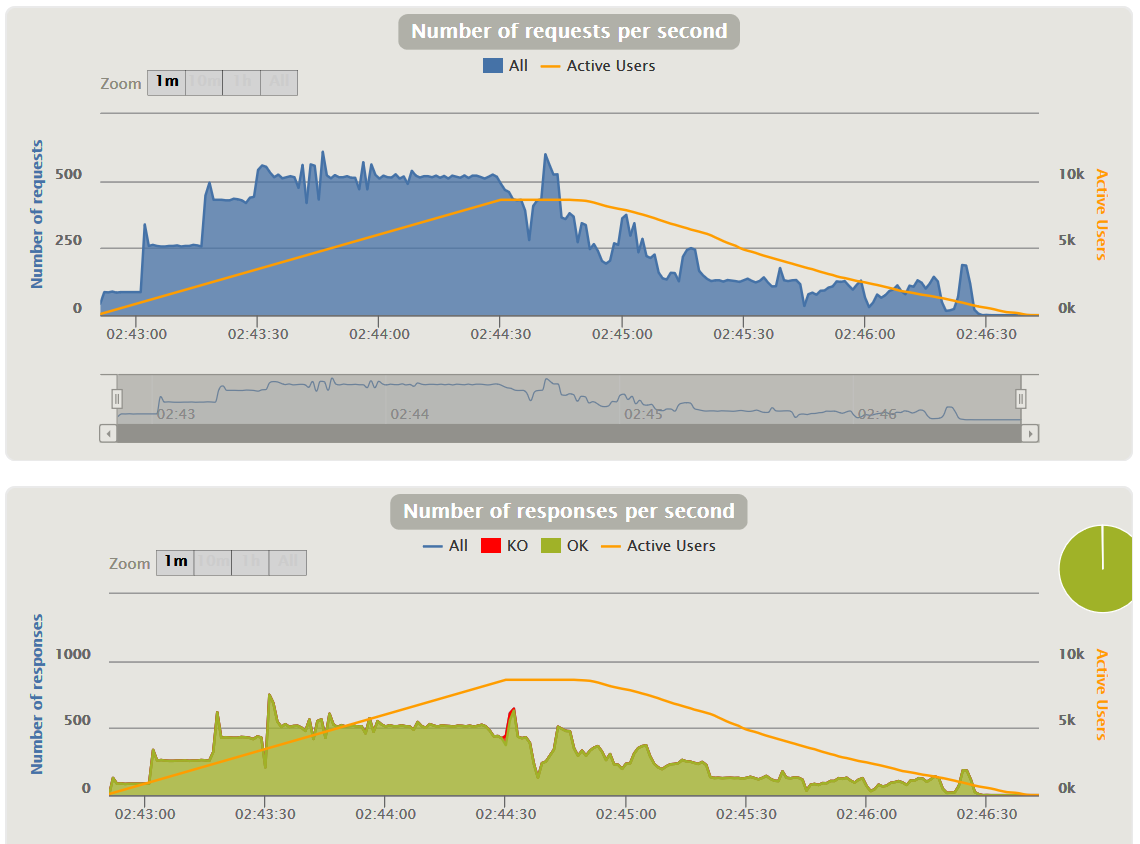
Se realizó una prueba que buscaba obtener cual era el número de usuarios máximo bajo los que el sistema era capaz de funcionar en condiciones aceptables. Definimos como condiciones aceptables estos dos criterios:

* El tiempo de respuesta medio es menor a 1 segundo (1000 ms)
* El 95% de las peticiones deben acabar en OK
* Existe un tercer assertion que comprueba que la petición que más tarde en realizarse en el sistema no supere los 5 segundos (5000ms), pero debido a posibles eventualidades no se considerará un criterio de peso para determinar si hay buen rendimiento.

Los resultados indicaron que el máximo número de usuarios que estos escenarios podían soportar con buen rendimiento era de 4300 usuarios en cada escenario (8600), con un límite de tiempo de 100 segundos para introducirlos todos.







Dado los datos generados por el report, podemos observar que sólo un 2% de las peticiones acabaron en KO (Cuando durante el pico de usuarios, los últimos intentaron acceder al Home) pero todos los usuarios pudieron realizar los escenarios. La media de tiempo de aceptación de una request fue de 516ms, siendo los valores más altos durante la ejecución de la creación de paciente. Aumentar el número de usuarios hubiera resultado en un tiempo medio de aceptación de request mayor a un segundo.

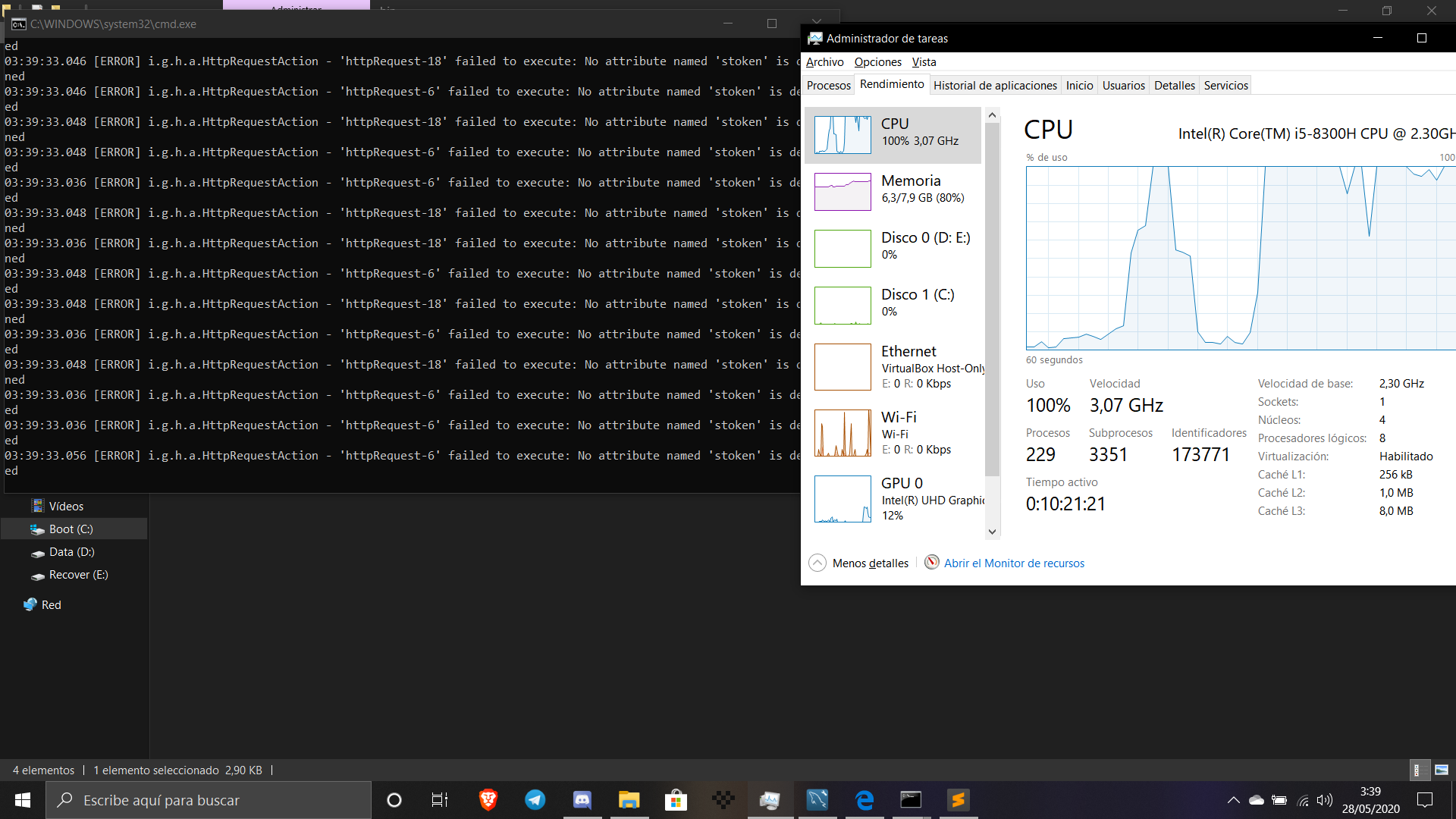
## HU-05 Mostrar Datos de Paciente:

Para la siguiente HU se proponen los siguientes escenarios:

* Acceso a los datos de un paciente:
  + El usuario accede al Home del sistema
  + El usuario se identifica en el sistema
  + El usuario accede al listado de pacientes
  + El usuario accede a los datos del paciente 1
* Acceso a los datos de un paciente no existente en BBDD:
  + El usuario accede al Home del sistema
  + El usuario se identifica en el sistema
  + El usuario accede al listado de pacientes
  + El usuario accede a los datos del paciente 1
  + El usuario intenta hacer un acceso ilegal a los datos del paciente 9999 que no existe en el sistema.

### Prueba de estrés

Se realizó una prueba de estrés que introdujo 12000 usuarios en cada escenario (24000), con un límite de tiempo de 10 segundos para introducirlos todos.



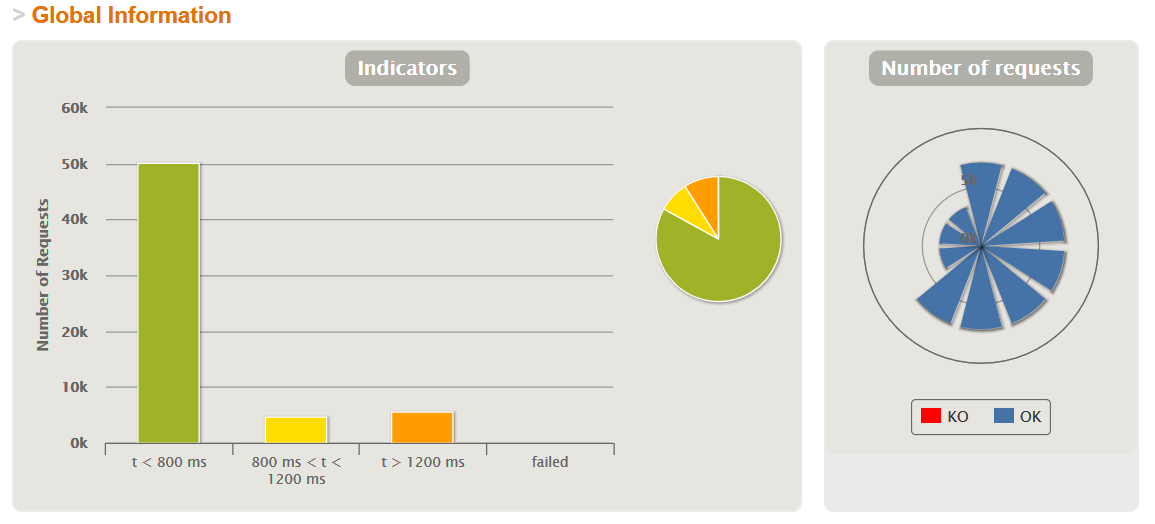
Cuando la CPU alcanzó un 100% de uso continuado, comenzó a generar incorrectamente los tickets csrf o directamente a no generarlos como se puede observar en la traza de ejecución de Gatling. Seguramente el alto uso de memoria RAM también influyese en la generación de dichos tickets.

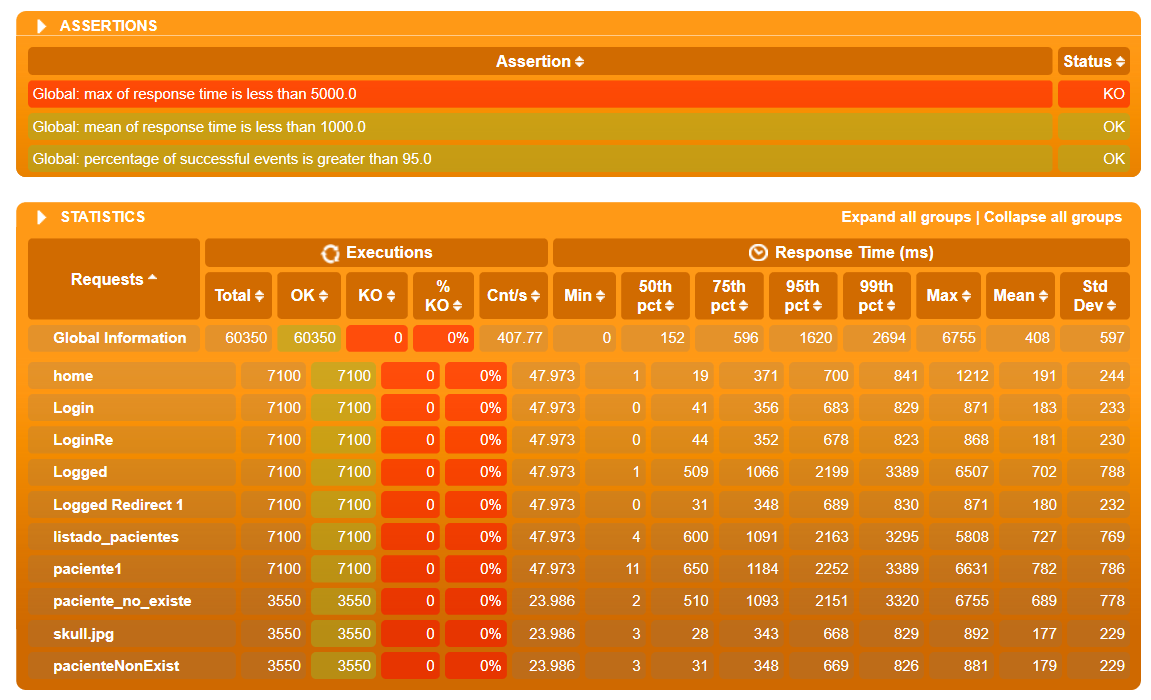
### Prueba de buen rendimiento:

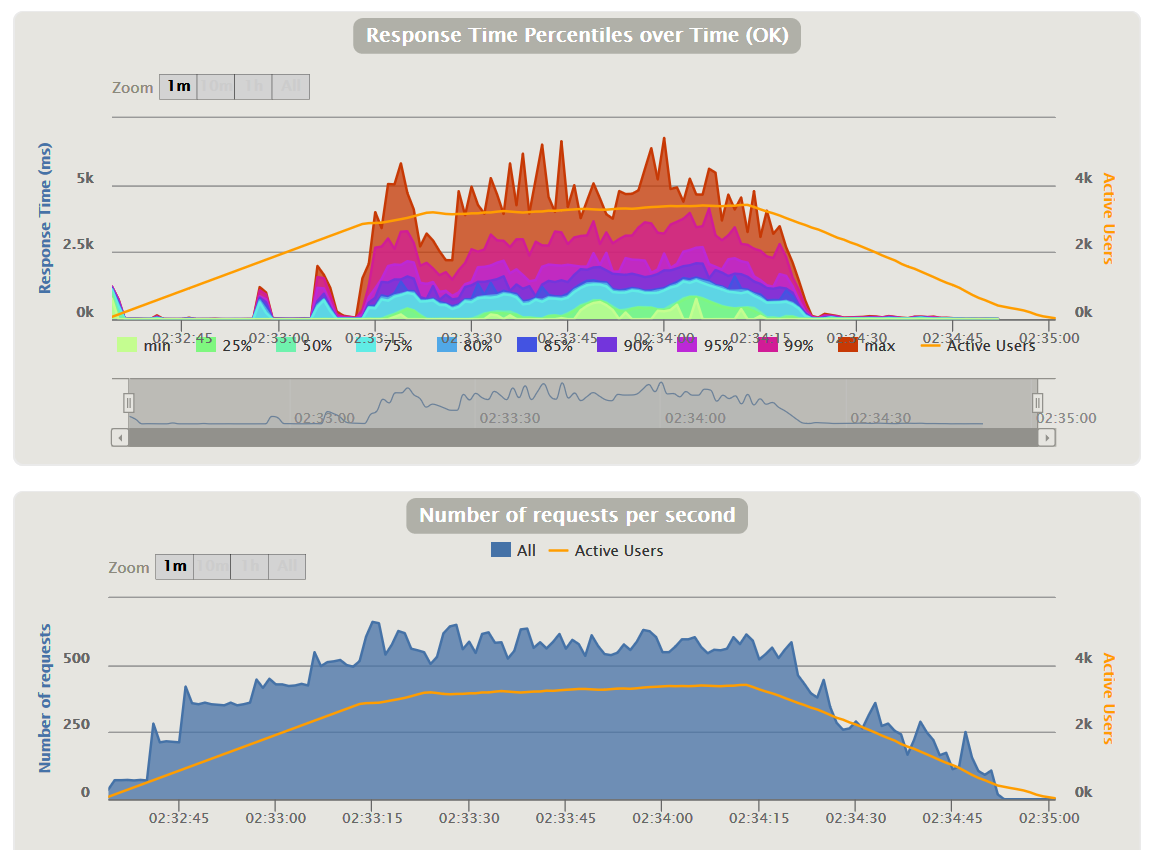
Se realizó una prueba que buscaba obtener cual era el número de usuarios máximo bajo los que el sistema era capaz de funcionar en condiciones aceptables. Definimos como condiciones aceptables estos dos criterios:

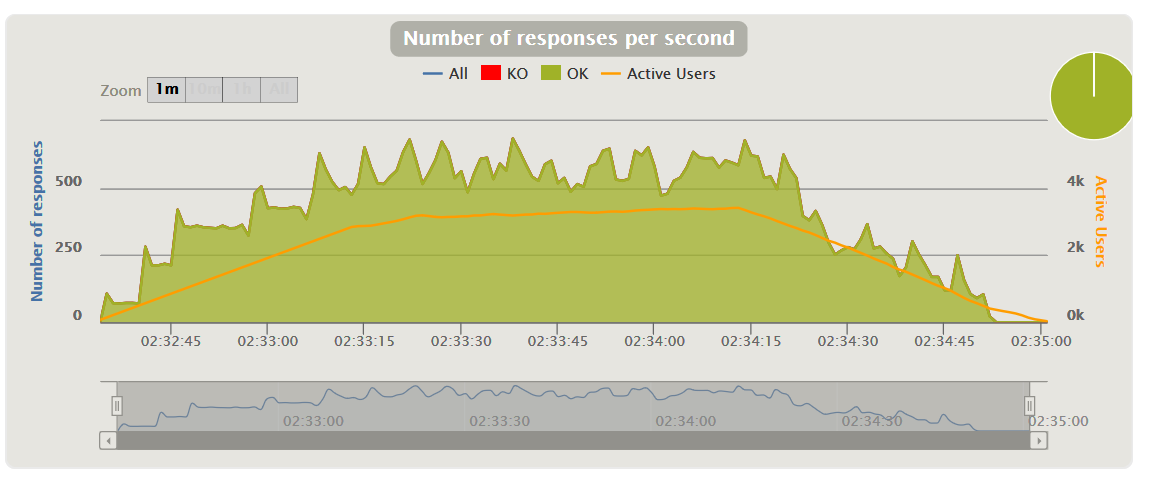
* El tiempo de respuesta medio es menor a 1 segundo (1000 ms)
* El 95% de las peticiones deben acabar en OK
* Existe un tercer assertion que comprueba que la petición que más tarde en realizarse en el sistema no supere los 5 segundos (5000ms), pero debido a posibles eventualidades no se considerará un criterio de peso para determinar si hay buen rendimiento.

Se descubrió que el número de usuarios concurrentes era de un máximo aproximado de 3550 en cada escenario (7100), con un límite de tiempo de 100 segundos para introducirlos todos.









No se rechazó ninguna request durante esta ejecución y el tiempo de asistencia medio de requests fue de 408ms, puede que parezca que existe bastante espacio para introducir más usuarios, pero otras pruebas con un número mayor comenzaban a rechazar más de un 5% de las request realizadas. Las operaciones más costosas correspondieron al envío del login y al acceso a los datos de un paciente. Aumentar el número de usuarios hubiera resultado en un tiempo medio de aceptación de request mayor a un segundo.

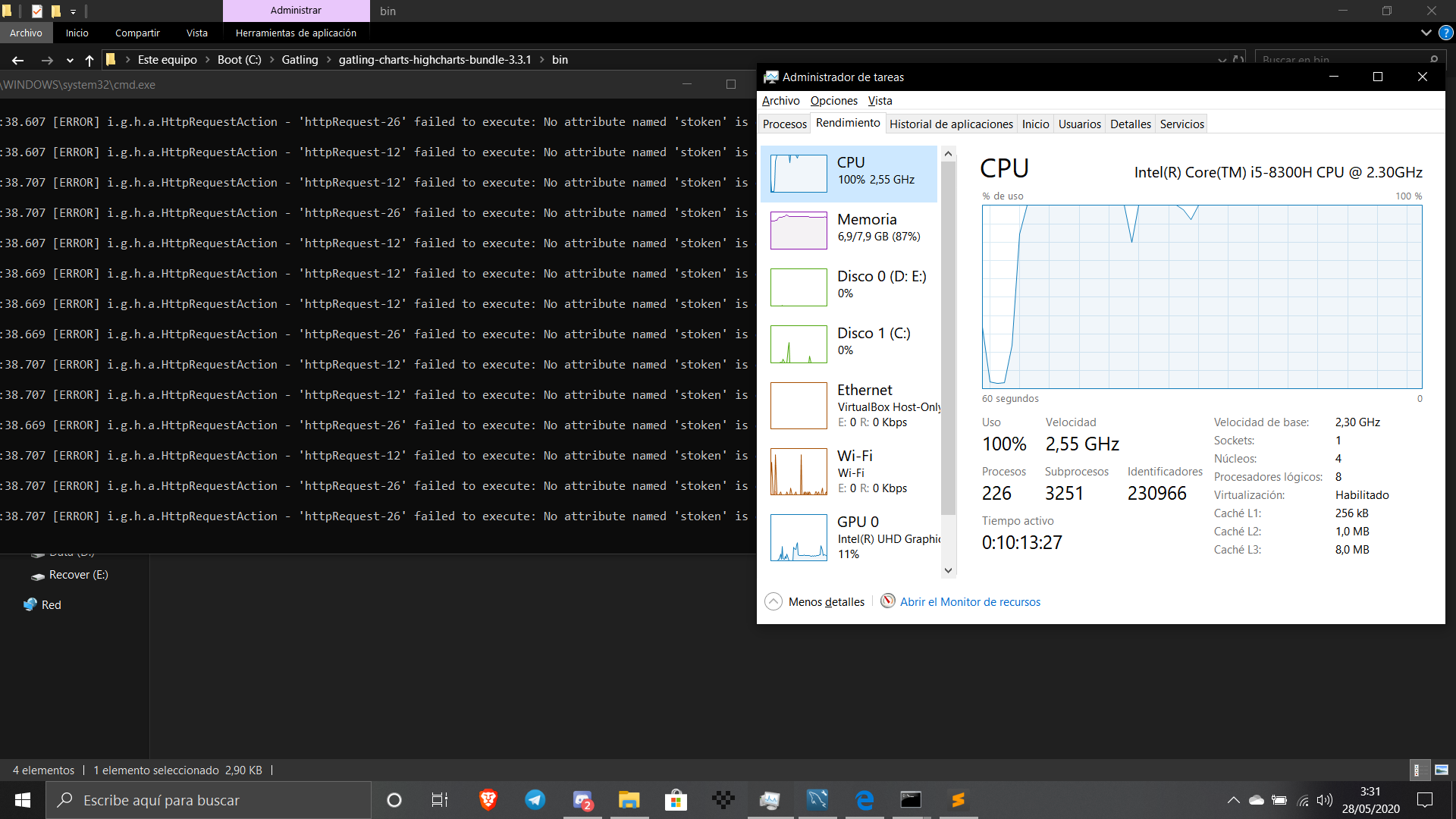
## HU-11 Crear Historia Clínica

Se definen los siguientes escenarios para esta Historia de Usuario:

* Un usuario crea una historia clínica:
  + Usuario accede a Home
  + Usuario se identifica en el sistema
  + Usuario accede a sus pacientes personales
  + Usuario accede a los datos de un paciente sin Historia Clínica
  + Usuario crea una Historia Clínica en el sistema
  + Dado que un Paciente sólo puede tener una Historia Clínica, sólo un usuario es capaz de crearla. El resto realizarían un acceso ilegal (No existe el botón en la vista) al formulario de creación y serían devueltos a la página de error.
  + El paciente sólo tiene una Historia Clínica al final de la ejecución.
* Un usuario Médico intenta crear una Historia Clínica para un paciente que no tiene asignado:
  + Usuario accede a Home
  + Usuario se identifica en el sistema
  + Usuario accede al listado de pacientes completo
  + Usuario accede a un paciente que no tiene asignado
  + Usuario intenta crear una Historia Clínica mediante un acceso ilegal (No existe el botón en la vista) y es redirigido al Home.

### Prueba de estrés

Se realizó una prueba de estrés con 25000 usuarios en cada escenario (50000), con un límite de tiempo de 10 segundos para introducirlos todos.



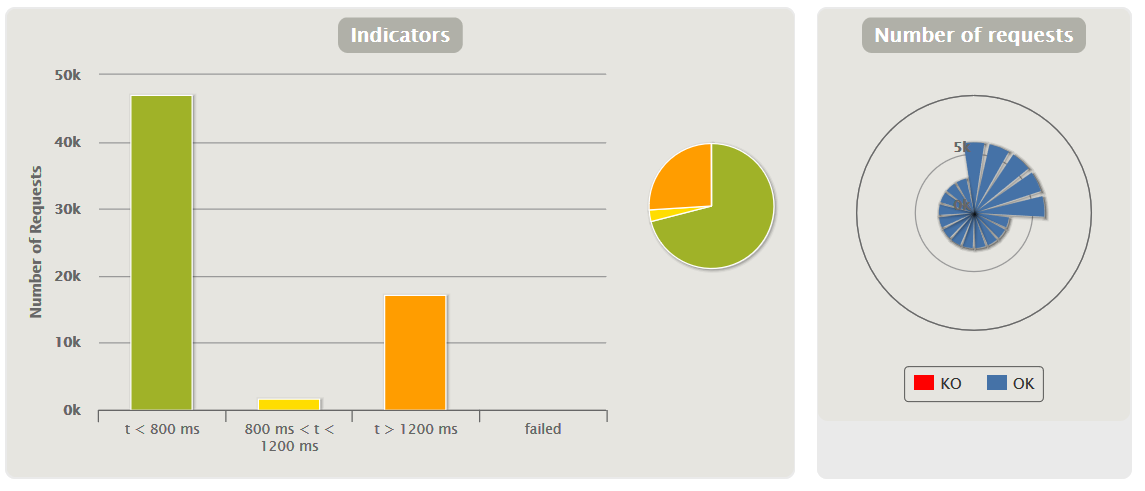
Como vemos, cuando la CPU al alcanzó un uso continuado del 100%, los tickets csrf comenzaron a fallar en su generación impidiendo así el uso de una operación post como podemos observar en la traza de Gatling. Seguramente el alto uso de memoria RAM también influyese en la generación de dichos tickets.

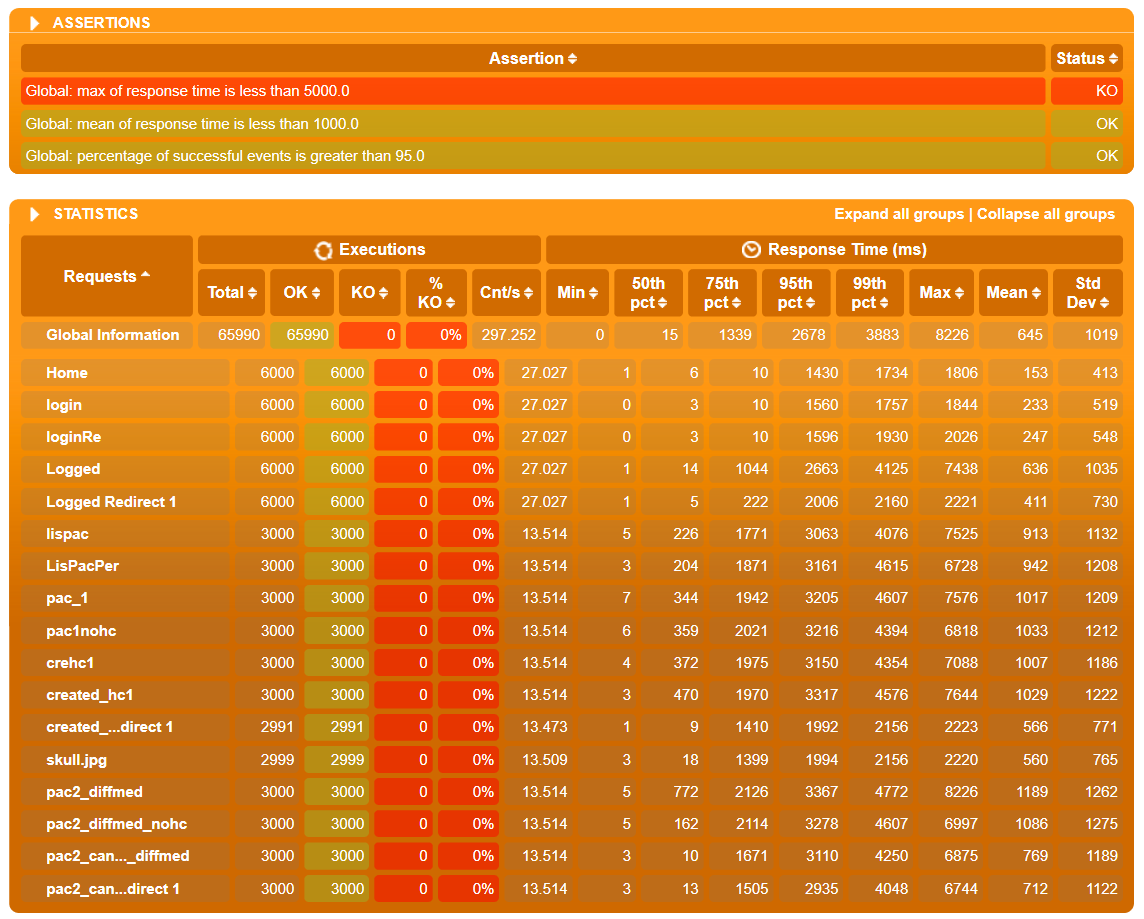
### Prueba de Buen Rendimiento:

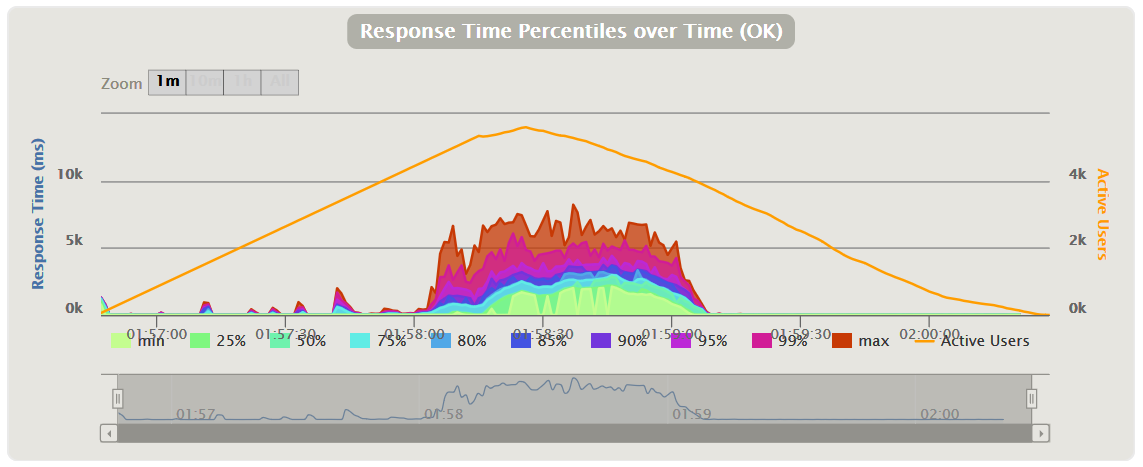
Se realizó una prueba que buscaba obtener cual era el número de usuarios máximo bajo los que el sistema era capaz de funcionar en condiciones aceptables. Definimos como condiciones aceptables estos dos criterios:

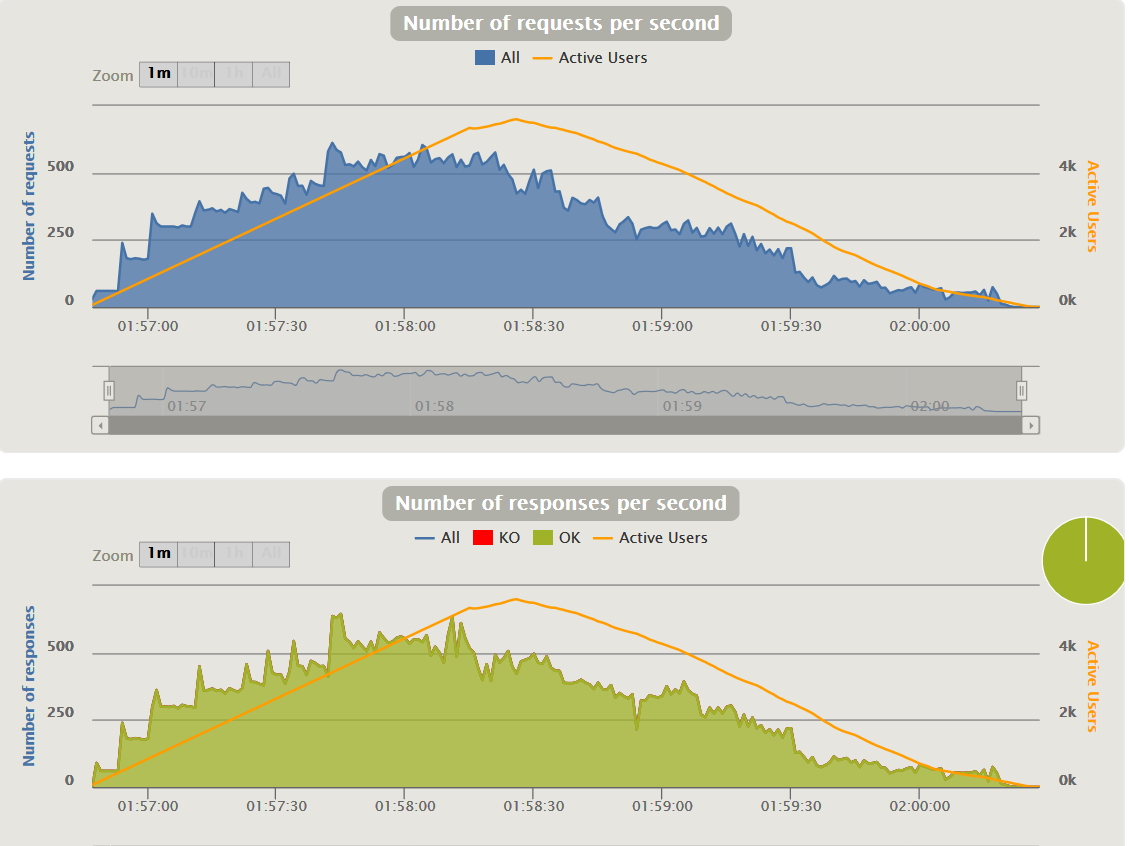
* El tiempo de respuesta medio es menor a 1 segundo (1000 ms)
* El 95% de las peticiones deben acabar en OK
* Existe un tercer assertion que comprueba que la petición que más tarde en realizarse en el sistema no supere los 5 segundos (5000ms), pero debido a posibles eventualidades no se considerará un criterio de peso para determinar si hay buen rendimiento.

Se descubrió que el número máximo de usuarios concurrentes era de aproximadamente 3000 en cada escenario (6000 usuarios), con un límite de tiempo de 100 segundos para introducirlos todos.









Todas las request fueron aceptadas, el tiempo medio de aceptación de las request fue de 645ms. Si el número de usuarios hubiese sido superior, aunque quizá si se hubieran atendido casi todas las request, el tiempo medio de aceptación hubiera superado el segundo. Observamos que 2999 usuarios del primer escenario acabaron en una página de error debido a la restricción de Historia Clínica. Las operaciones más costosas fueron el acceso a los datos de un Paciente perteneciente a un médico distinto, el acceso ilegal a la creación de una Historia Clínica para este y el acceso fallido a la creación de una Historia Clínica para un medico asignado al usuario (Fallido dado que fueron dirigidos a una página de error).