This Spring Shall Be Challenged

Does it has to be Spring all the time?

Holger Steinhauer, FOSDEM 2021, Kotlin DevRoom

About Me

I keep it short

- Developer for over 15 years
- More than a decade on JVM
- Kotlin lover since 2019
- Founder of steinhauer.software
- Co-Organiser of Virtual KUG, KUG Berlin & Java Advent
- Podcaster

Agenda

- A Opinionated View On Current Projects
- The “Challenge”
- The Contenders
- The Findings
- Conclusion

A Opinionated View On Projects

A Opinionated View On Projects

- Decisions made by managers
  - Based on Google results
- Silver bullet thinking
- Companies look for framework followers, not software developers

The “Challenge”

The “Challenge”

- Simple REST endpoint
- Return list of available articles with meta-data from a database
- JSON output
- Needs authentication

The Contenders

The Contenders

- Should
  - Easy to use and on the resources
  - Support Gradle (ok, and Maven)
  - Have Kotlin examples (duh)
  - Provide bootstrapping
  - Be Open Source
  - Support auth with OAuth, Sessions and more
  - Easy metric collection (e.g. Micrometer)

The Contenders'

…little helpers

- Ebean (https://ebean.io) as ORM
- Koin DI (https://insert-koin.io) as DI container
- Gatling (https://gatling.io/) for Performance Tests
- Caddy as http server (https://caddyserver.com/)

The Contenders

Spring Boot

Pros	Cons
Well known	Heavy use of reflection
Good documentation	Loads of dependencies
Huge community	Lots of Java baggage
Backed by VMWare	Slow startup

The Contenders

Ktor

Pros	Cons
Pure Kotlin	
Good documentation	
Active community	
Backed by Jetbrains	

The Contenders

Micronaut

Pros	Cons
Similar to Spring	Brings some Java baggage
Uses (K)APT	
Fast Startup	
Backed by Object Computing, Inc	

The Contenders

Jooby

Pros	Cons
Really fast start up	Spotty documentation
Minimal set up	Small team
Kotlin-esque DSL	Not many extension with 2.x
Thin extension layer	Java baggage

Findings

Findings

Jar Size And Startup

Framework	JAR Size	Startup Time	Heap Usage
Spring	27 MB	~ 3sec	~ 200MB
Ktor	22 MB	~ 1sec	**~ 82MB**
Micronaut	22 MB	~ 1sec	~ 92MB
Jooby	**13 MB**	**< 1sec**	~ 140MB

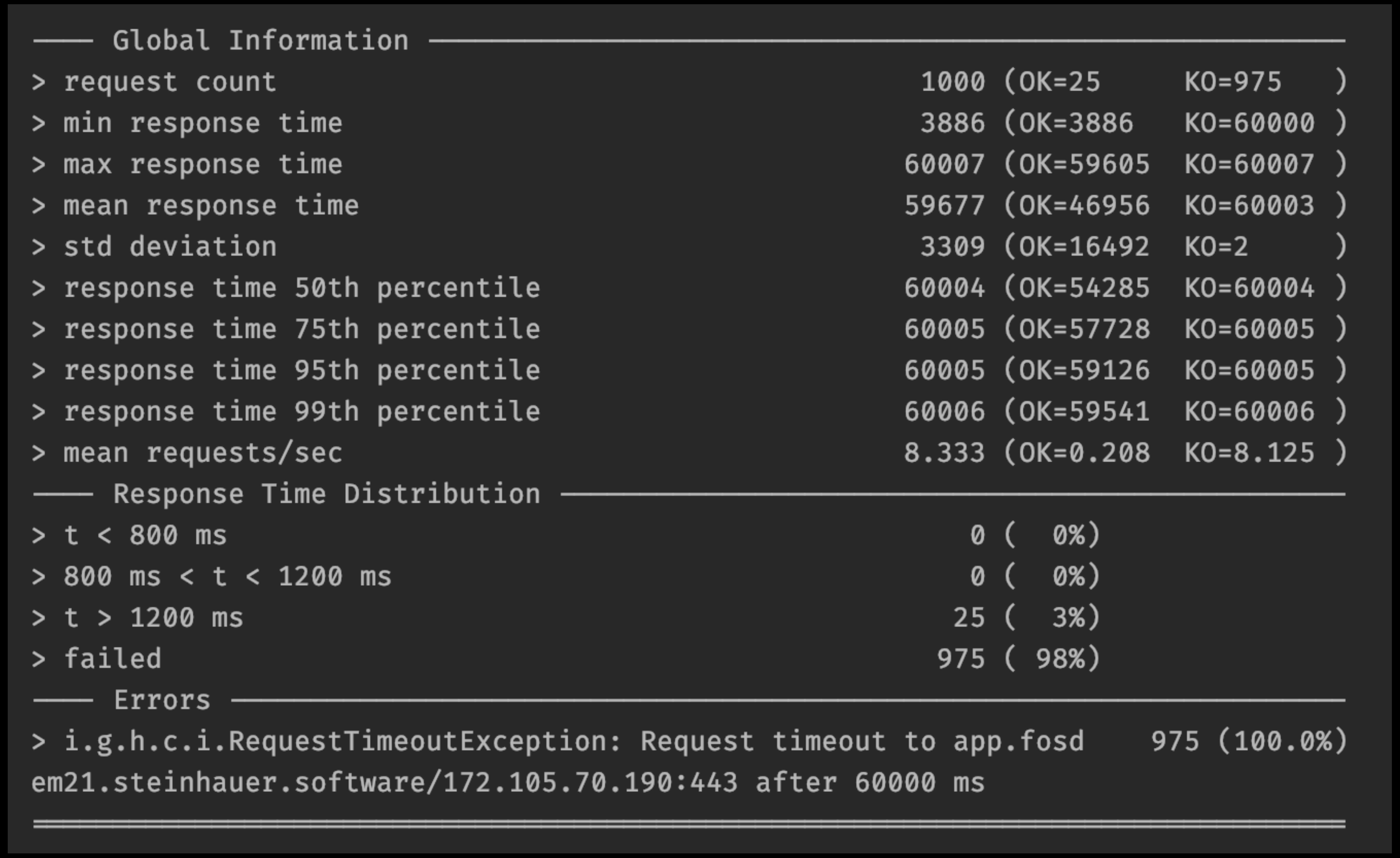
Findings

Memory With Requests

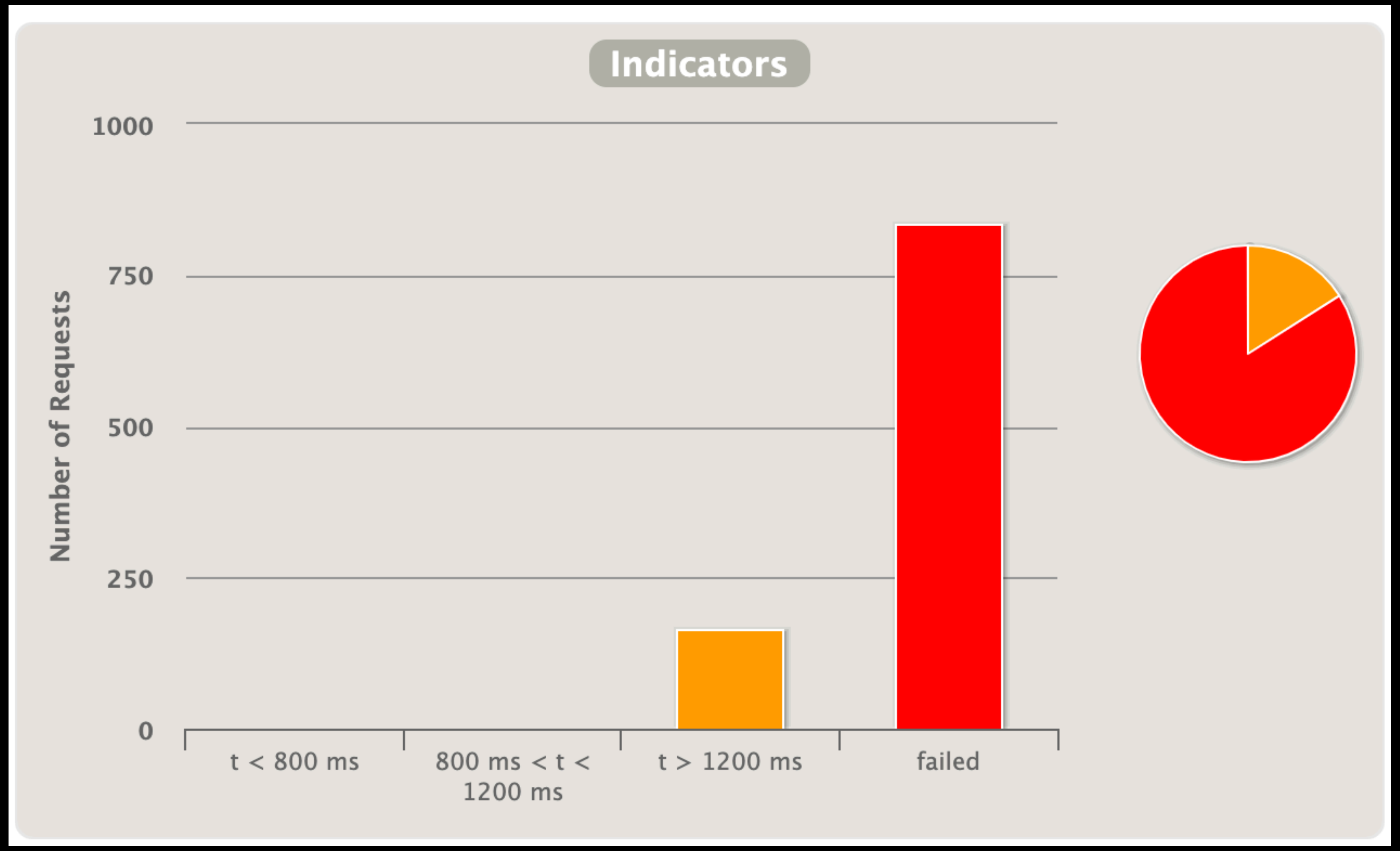
Framework	Heap Usage (Initial Request)	Heap Usage (After Gatling)
Spring	~53MB	~ 221MB
Ktor	**~ 34MB**	**~ 144MB**
Micronaut	~ 70MB	~ 544MB
Jooby	~ 72MB	~ 314MB

Findings

Spring Load Test

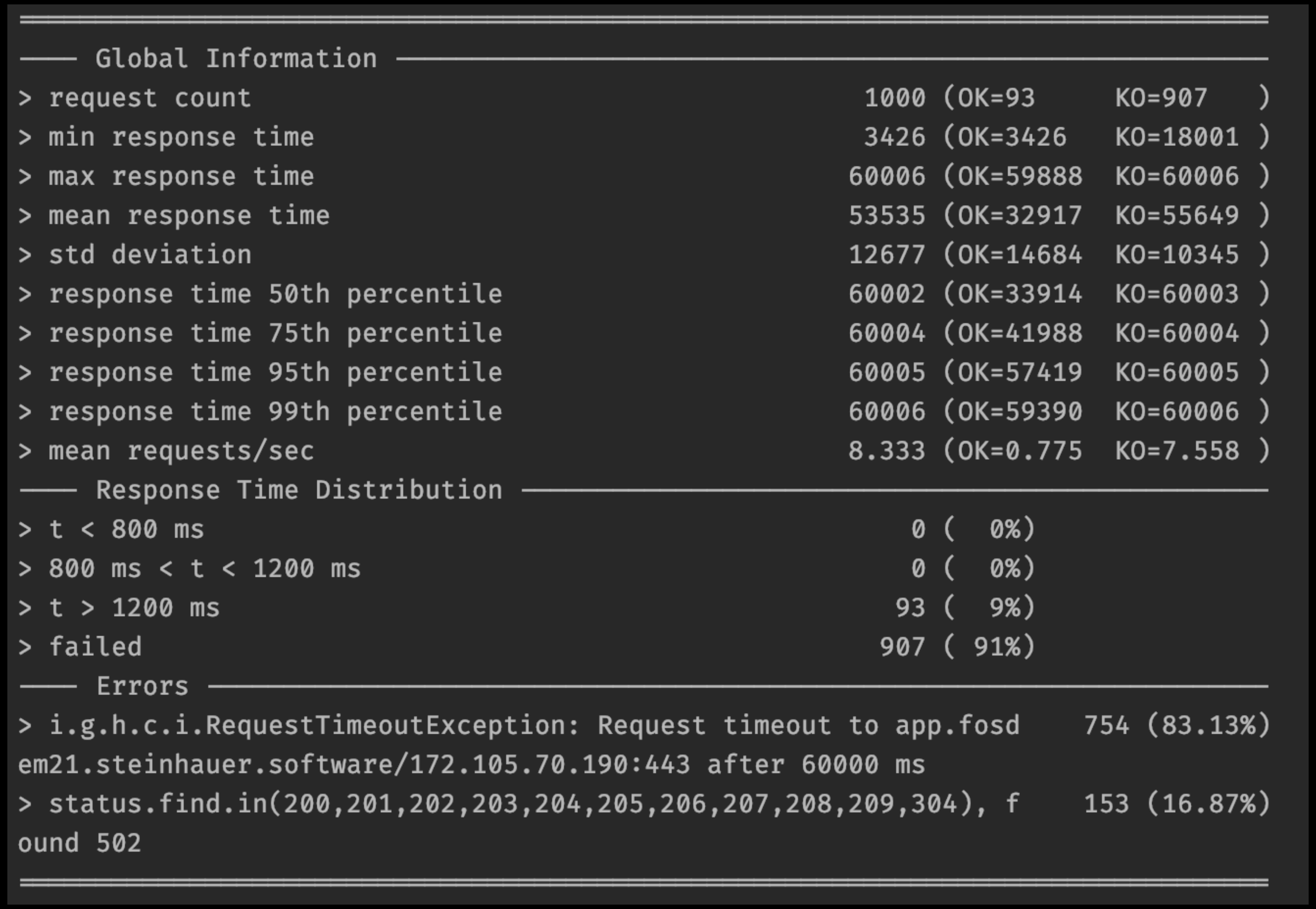


```
---- Global Information --------------------------------------------------------
> request count                                       1000 (OK=25     KO=975    )
> min response time                                   3886 (OK=3886   KO=60000  )
> max response time                                  60007 (OK=59605  KO=60007  )
> mean response time                                 59677 (OK=46956  KO=60003  )
> std deviation                                       3309 (OK=16492  KO=2      )
> response time 50th percentile                      60004 (OK=54285  KO=60004  )
> response time 75th percentile                      60005 (OK=57728  KO=60005  )
> response time 95th percentile                      60005 (OK=59126  KO=60005  )
> response time 99th percentile                      60006 (OK=59541  KO=60006  )
> mean requests/sec                                  8.333 (OK=0.208  KO=8.125  )
---- Response Time Distribution ------------------------------------------------
> t < 800 ms                                             0 (  0%)
> 800 ms < t < 1200 ms                                   0 (  0%)
> t > 1200 ms                                           25 (  3%)
> failed                                               975 ( 98%)
---- Errors --------------------------------------------------------------------
> i.g.h.c.i.RequestTimeoutException: Request timeout to app.fosd    975 (100.0%)
em21.steinhauer.software/172.105.70.190:443 after 60000 ms
================================================================================
```

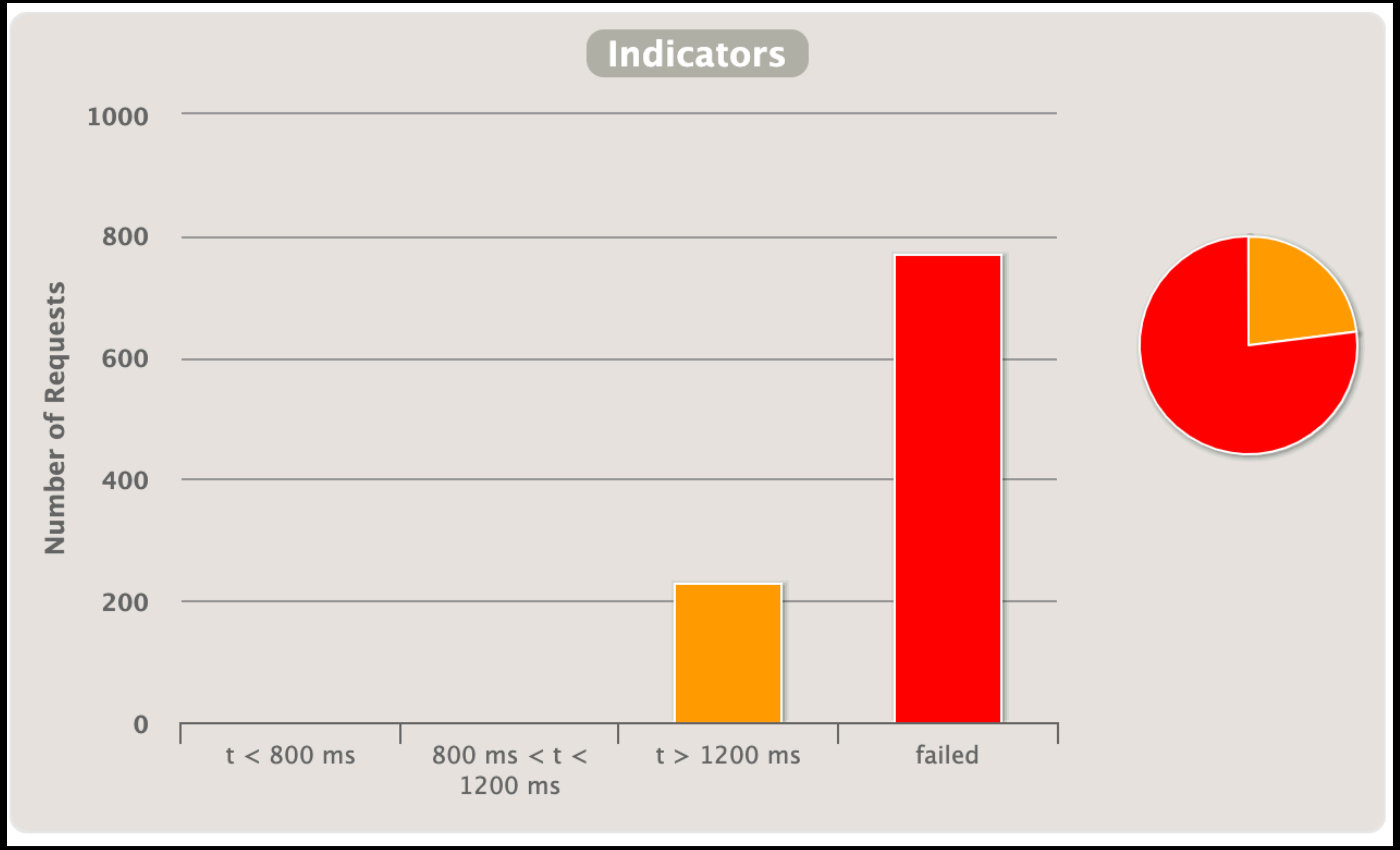


Findings

Ktor Load Test

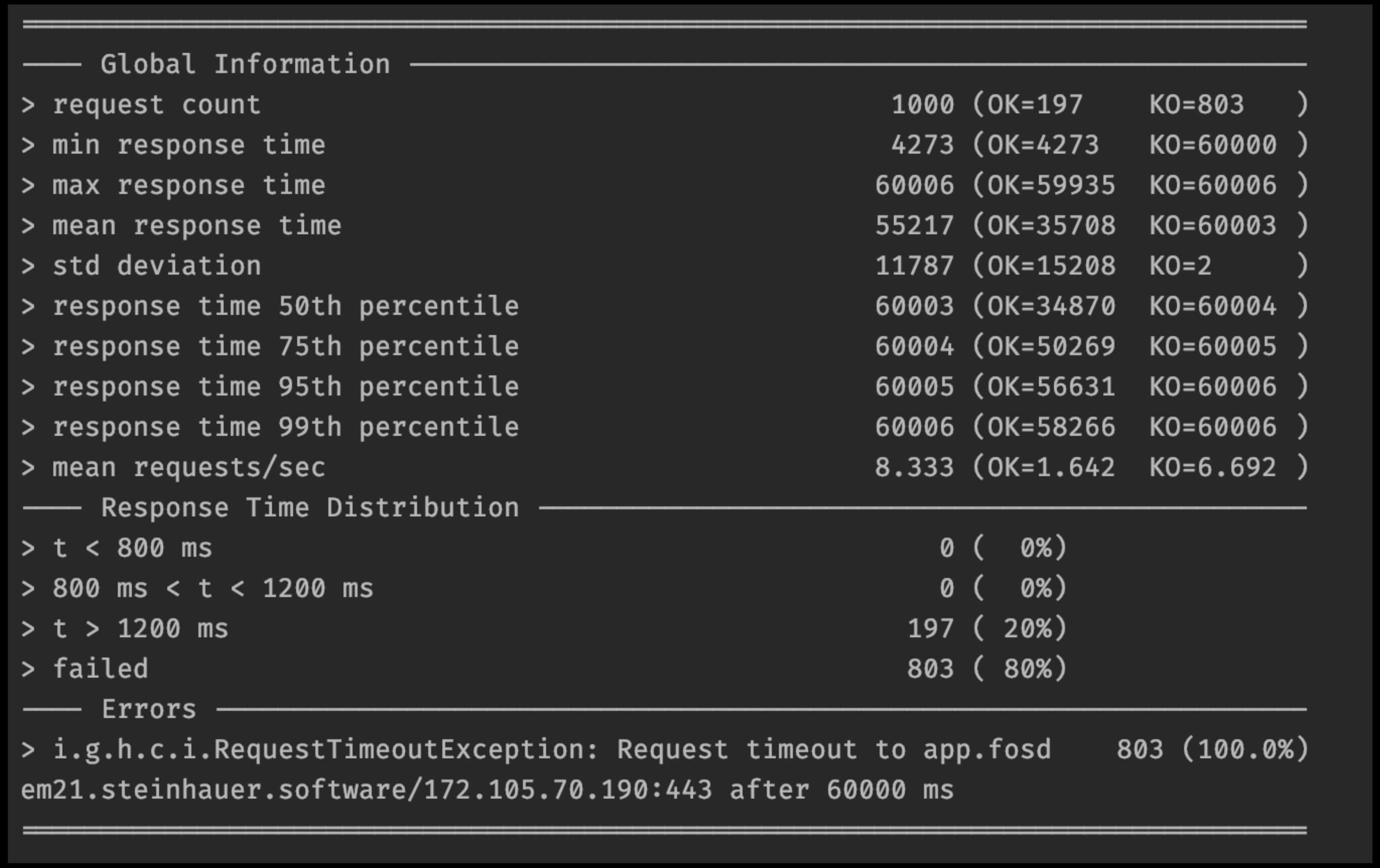


```
================================================================================
---- Global Information --------------------------------------------------------
> request count                                       1000 (OK=93     KO=907   )
> min response time                                   3426 (OK=3426   KO=18001 )
> max response time                                  60006 (OK=59888  KO=60006 )
> mean response time                                 53535 (OK=32917  KO=55649 )
> std deviation                                      12677 (OK=14684  KO=10345 )
> response time 50th percentile                      60002 (OK=33914  KO=60003 )
> response time 75th percentile                      60004 (OK=41988  KO=60004 )
> response time 95th percentile                      60005 (OK=57419  KO=60005 )
> response time 99th percentile                      60006 (OK=59390  KO=60006 )
> mean requests/sec                                  8.333 (OK=0.775  KO=7.558 )
---- Response Time Distribution ------------------------------------------------
> t < 800 ms                                             0 (  0%)
> 800 ms < t < 1200 ms                                   0 (  0%)
> t > 1200 ms                                           93 (  9%)
> failed                                               907 ( 91%)
---- Errors --------------------------------------------------------------------
> i.g.h.c.i.RequestTimeoutException: Request timeout to app.fosd    754 (83.13%)
em21.steinhauer.software/172.105.70.190:443 after 60000 ms
> status.find.in(200,201,202,203,204,205,206,207,208,209,304), f    153 (16.87%)
ound 502
================================================================================
```

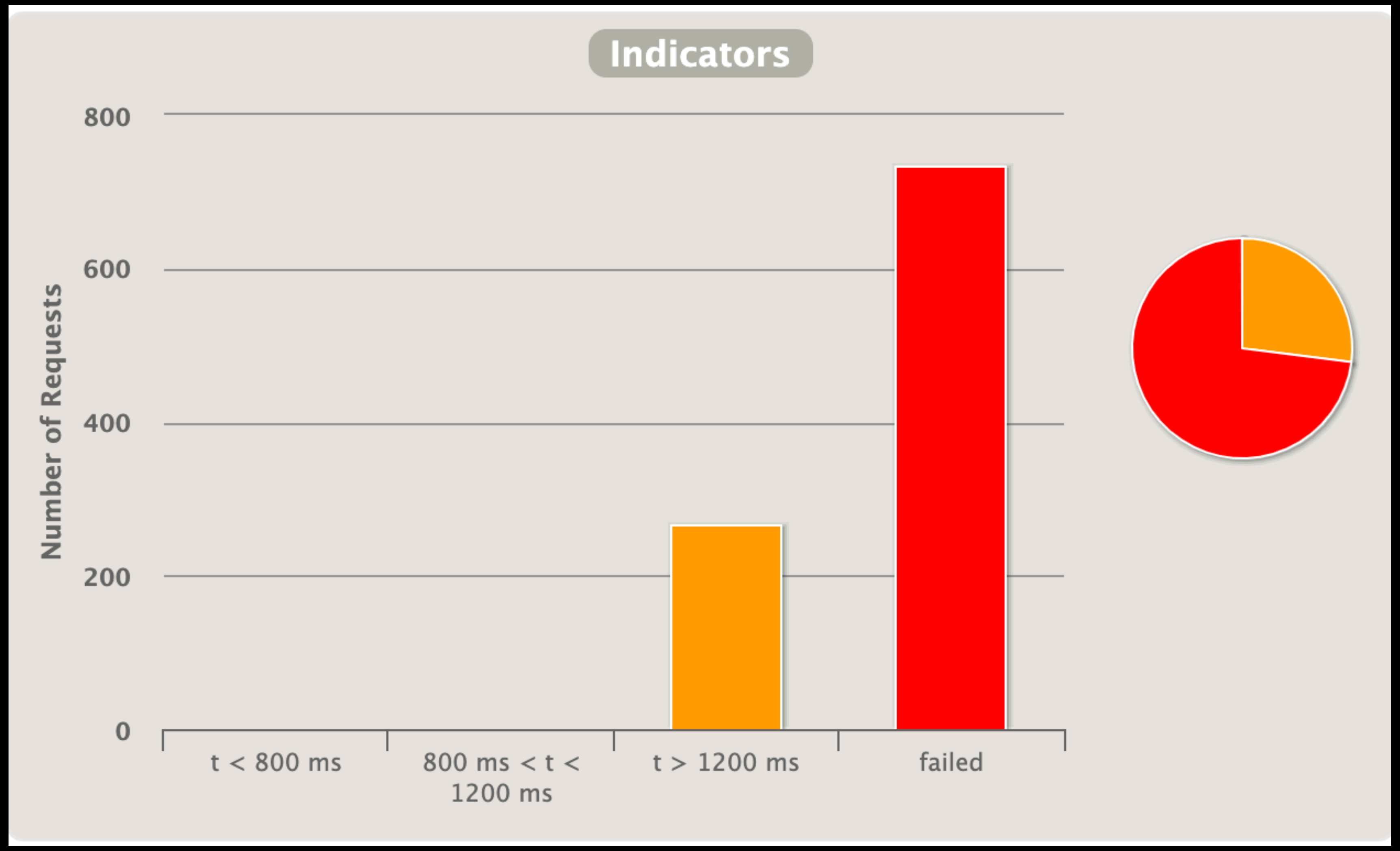


Findings

Micronaut Load Test

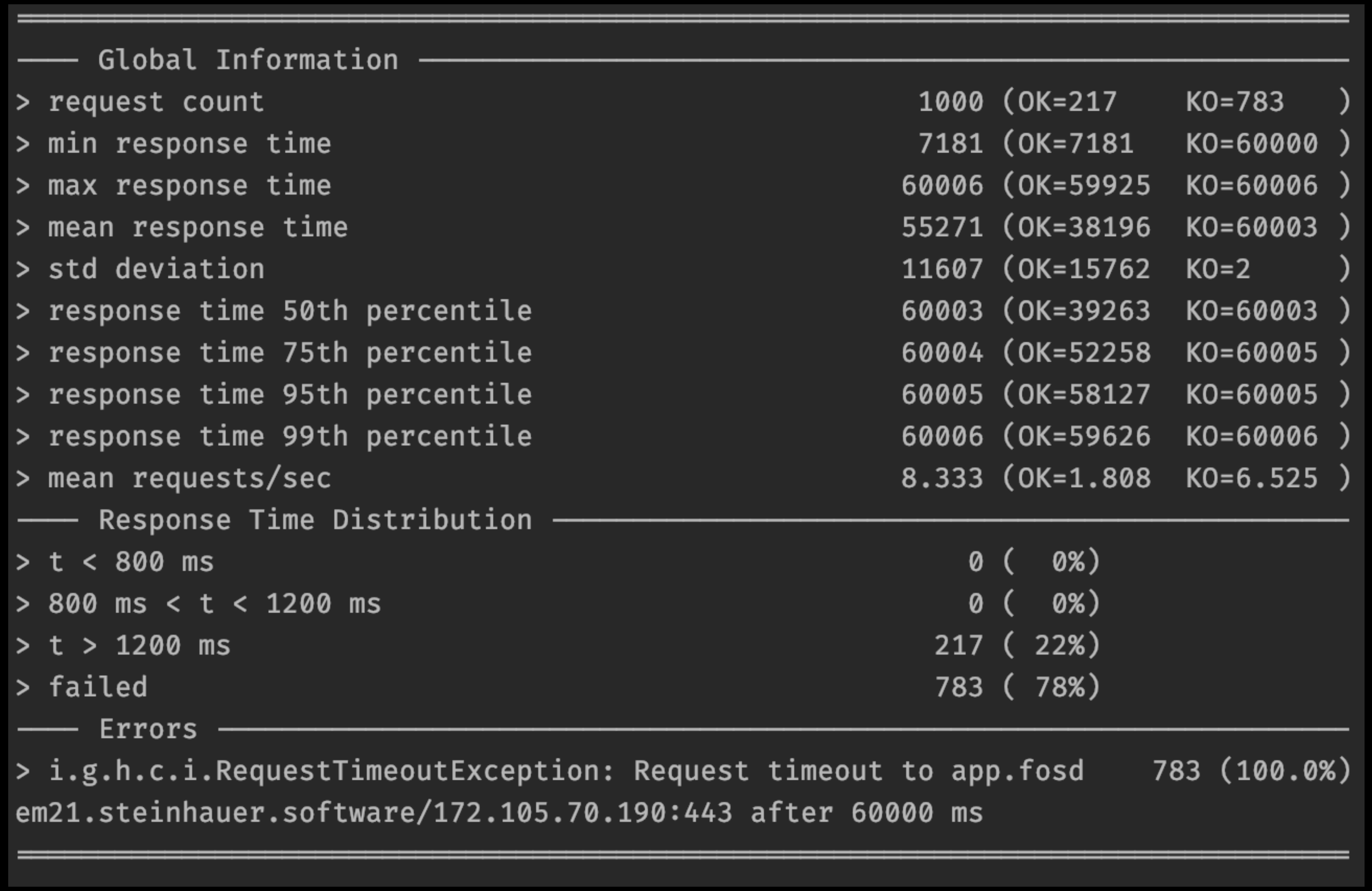


```
================================================================================
---- Global Information --------------------------------------------------------
> request count                                       1000 (OK=197    KO=803   )
> min response time                                   4273 (OK=4273   KO=60000 )
> max response time                                  60006 (OK=59935  KO=60006 )
> mean response time                                 55217 (OK=35708  KO=60003 )
> std deviation                                      11787 (OK=15208  KO=2     )
> response time 50th percentile                      60003 (OK=34870  KO=60004 )
> response time 75th percentile                      60004 (OK=50269  KO=60005 )
> response time 95th percentile                      60005 (OK=56631  KO=60006 )
> response time 99th percentile                      60006 (OK=58266  KO=60006 )
> mean requests/sec                                  8.333 (OK=1.642  KO=6.692 )
---- Response Time Distribution ------------------------------------------------
> t < 800 ms                                             0 (  0%)
> 800 ms < t < 1200 ms                                   0 (  0%)
> t > 1200 ms                                          197 ( 20%)
> failed                                               803 ( 80%)
---- Errors --------------------------------------------------------------------
> i.g.h.c.i.RequestTimeoutException: Request timeout to app.fosd    803 (100.0%)
em21.steinhauer.software/172.105.70.190:443 after 60000 ms
================================================================================
```

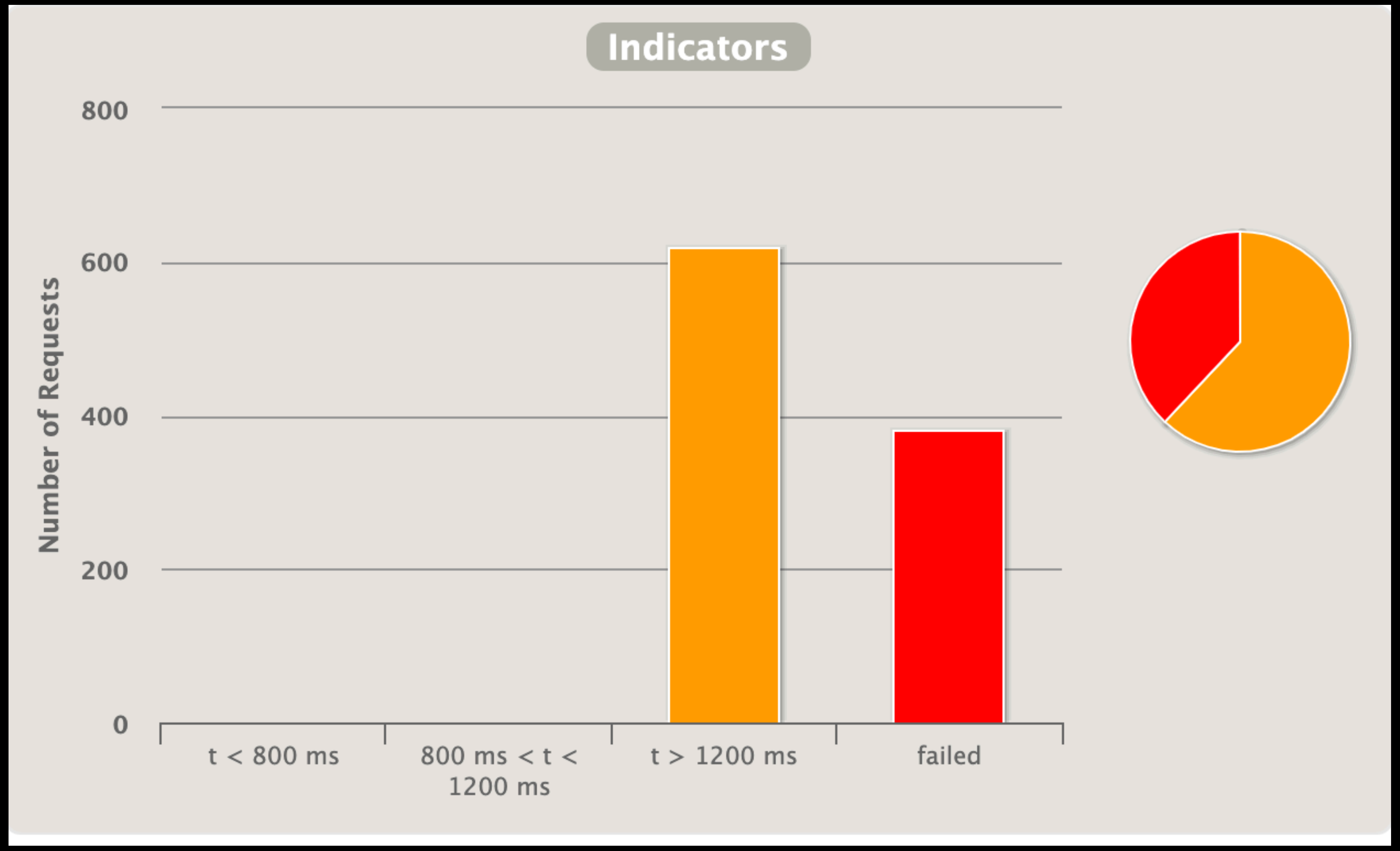


Findings

Jooby Load Test



```
================================================================================
---- Global Information --------------------------------------------------------
> request count                                       1000 (OK=217    KO=783    )
> min response time                                   7181 (OK=7181   KO=60000 )
> max response time                                  60006 (OK=59925  KO=60006 )
> mean response time                                 55271 (OK=38196  KO=60003 )
> std deviation                                      11607 (OK=15762  KO=2      )
> response time 50th percentile                      60003 (OK=39263  KO=60003 )
> response time 75th percentile                      60004 (OK=52258  KO=60005 )
> response time 95th percentile                      60005 (OK=58127  KO=60005 )
> response time 99th percentile                      60006 (OK=59626  KO=60006 )
> mean requests/sec                                  8.333 (OK=1.808  KO=6.525  )
---- Response Time Distribution ------------------------------------------------
> t < 800 ms                                             0 (  0%)
> 800 ms < t < 1200 ms                                   0 (  0%)
> t > 1200 ms                                          217 ( 22%)
> failed                                               783 ( 78%)
---- Errors --------------------------------------------------------------------
> i.g.h.c.i.RequestTimeoutException: Request timeout to app.fosd    783 (100.0%)
em21.steinhauer.software/172.105.70.190:443 after 60000 ms
================================================================================
```



The Conclusion

Spring Might Be Just Fine

Good feature / resource balance

Ktor Is Solid

Pure Kotlin, Great Community, Fast Growth

Micronaut Is The Faster Spring

Similar approach, annotation processing boost speed

Jooby’s Promising

Fast, Resilient, Easy on the Resources

Questions?

Thank you

Come again

- Code: https://github.com/daincredibleholg/this-spring-shall-be-challenged
- LinkedIn: https://www.linkedin.com/in/holgersteinhauer/
- Twitter: https://twitter.com/hfsteinhauer
- Company: https://steinhauer.software
- Virtual KUG: https://www.meetup.com/Virtual-Kotlin-User-Group/
- KUG Berlin: https://www.meetup.com/kotlin-berlin/
- Podcast: https://anchor.fm/coding-with-holger
- Java Advent: https://www.javaadvent.com