

Sprawozdanie

Rekonfigurowalność e-systemów

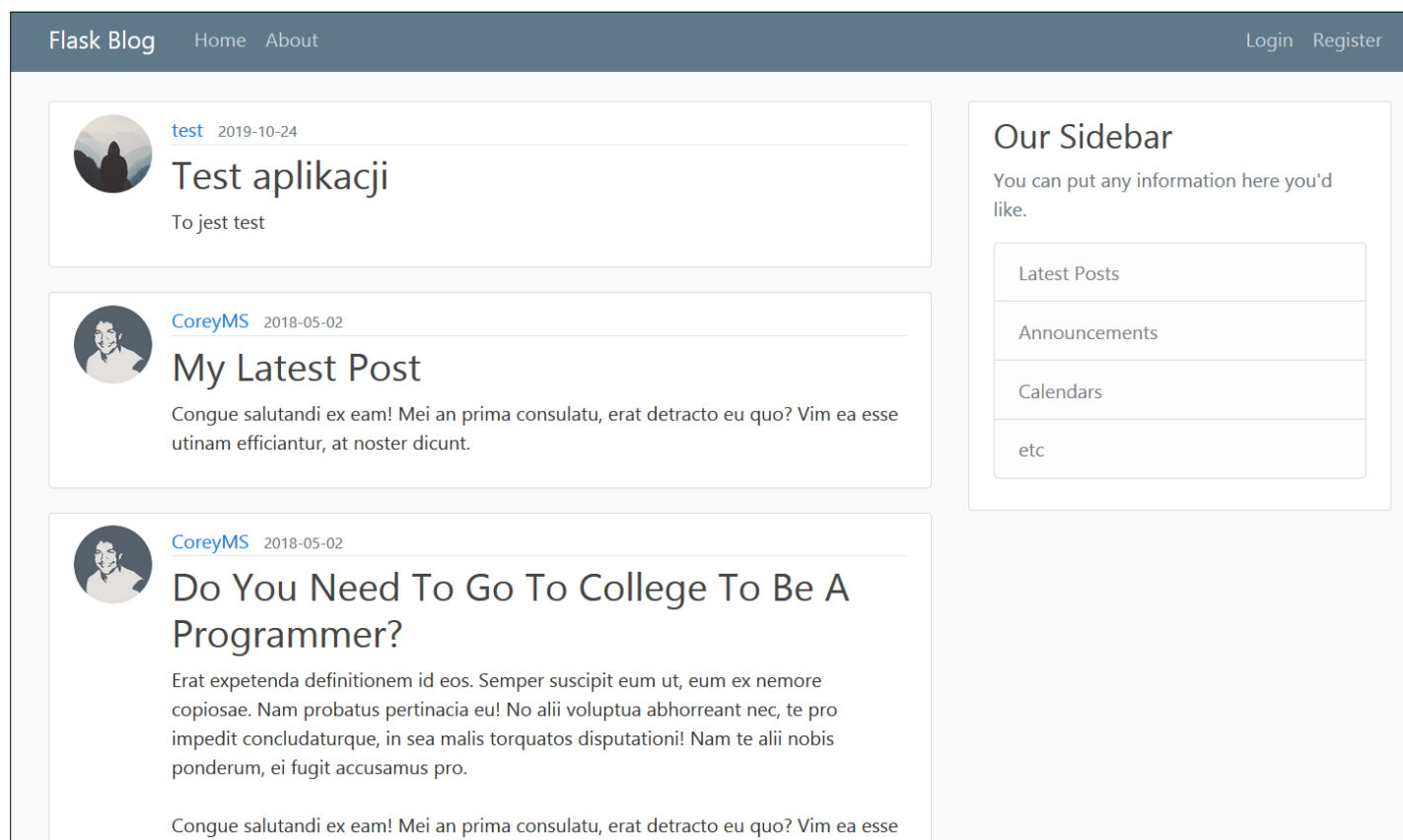
Mateusz Nimirski 226019

Wojciech Czarnecki 235714

Wstęp

Celem naszego projektu było zbadanie aplikacji webowej komunikującej się z bazą danych wykorzystując mapowanie obiektowo-relacyjne.

Badaną aplikacją webową był prosty blog umożliwiający zakładanie kont, logowanie oraz dodawanie, usuwanie i modyfikowanie postów. Dane kont oraz dotychczasowych postów są przechowywane w bazie danych SQLite.



Rysunek 1 Strona główna aplikacji

Aplikacja została napisana w języku Python z wykorzystaniem frameworku Flask. Komunikacja z bazą danych jest realizowana z wykorzystaniem SQLAlchemy ORM.

Aplikacja została postawiona na serwerze wykorzystując Linode. Parametry serwera to:

Lokalizacja: Frankfurt De

System operacyjny: Ubuntu 18.10

RAM: 1GB

Pamięć dysku twardego: 25GB

Ilość rdzeni: 1

Do przeprowadzenia pomiarów został wykorzystany program JMeter wraz z dodatkowymi wtyczkami.

Pomiary

Pierwszym etapem badania aplikacji było wykonanie testu poprawnego działania aplikacji poprzez wysłanie pojedynczego zapytania przez jednego użytkownika.

Nazwa	Wartość	URL Encode?	Content-Type	Include Equals?
-------	---------	-------------	--------------	-----------------

Rysunek 2 Wykorzystane zapytanie http do testów

Sample #	Start Time	Thread Name	Etykieta	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(n
1	18:07:36.394	Users 1-1	Home page	91		7539	117	91	

Rysunek 3 Wynik testu dla jednego zapytania pojedynczego użytkownika

```
<!DOCTYPE html>
<html>
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6f
y4IWVTh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

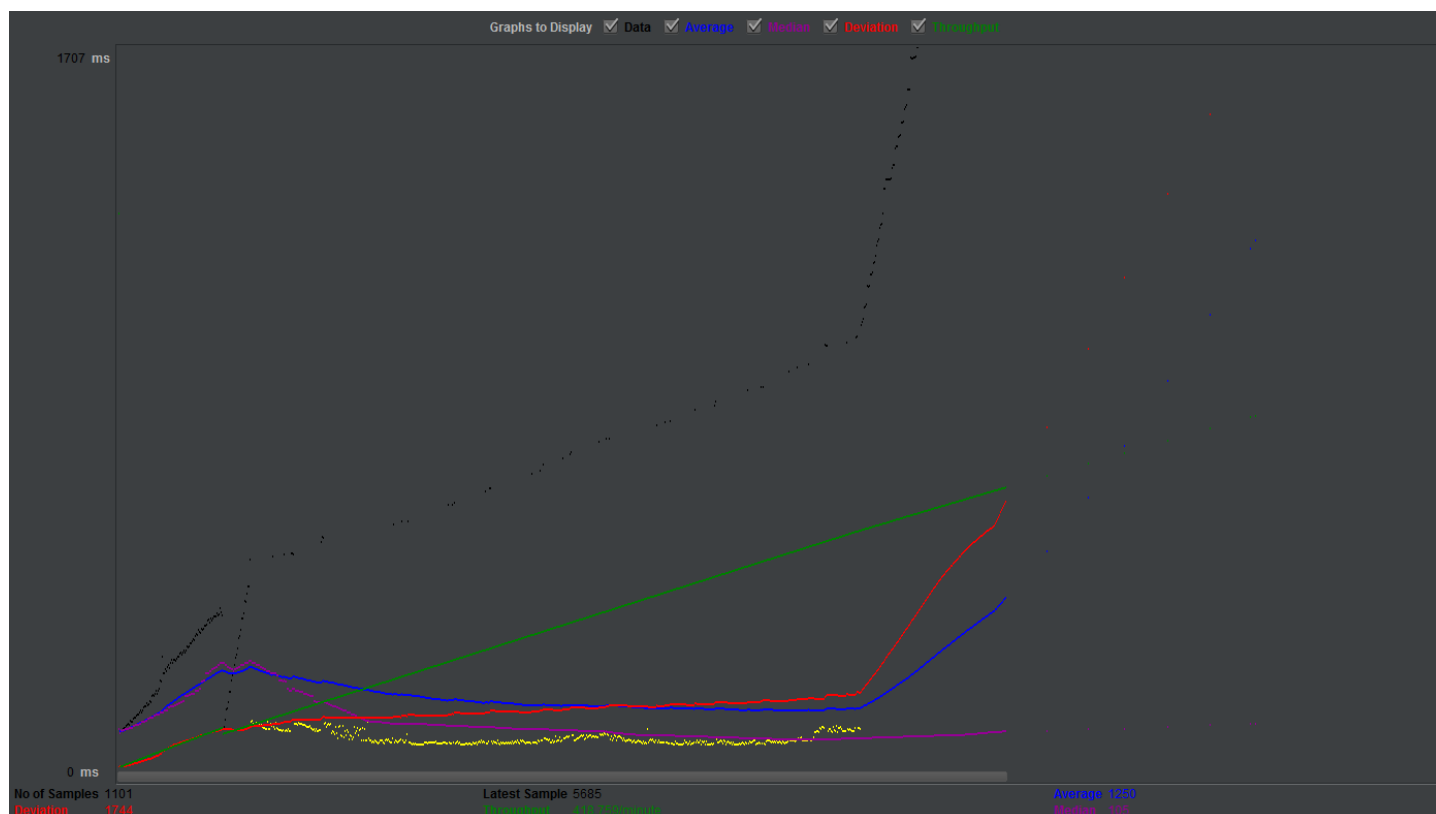
  <link rel="stylesheet" type="text/css" href="/static/main.css">

  <title>Flask Blog</title>
</head>
<body>
  <header class="site-header">
    <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">
      <div class="container">
        <a class="navbar-brand mr-4" href="/">Flask Blog</a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-controls="navbarToggle" aria-expanded="false"
aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarToggle">
```

Rysunek 4 Response body prawidłowej odpowiedzi.

Powyższe rysunki pokazują prawidłowe działanie aplikacji dla jednego zapytania pojedynczego użytkownika

Następnie zostały przeprowadzone testy dla większej ilości jednoczesnych użytkowników oraz większej ilości wysyłanych komunikatów przez pojedynczego użytkownika



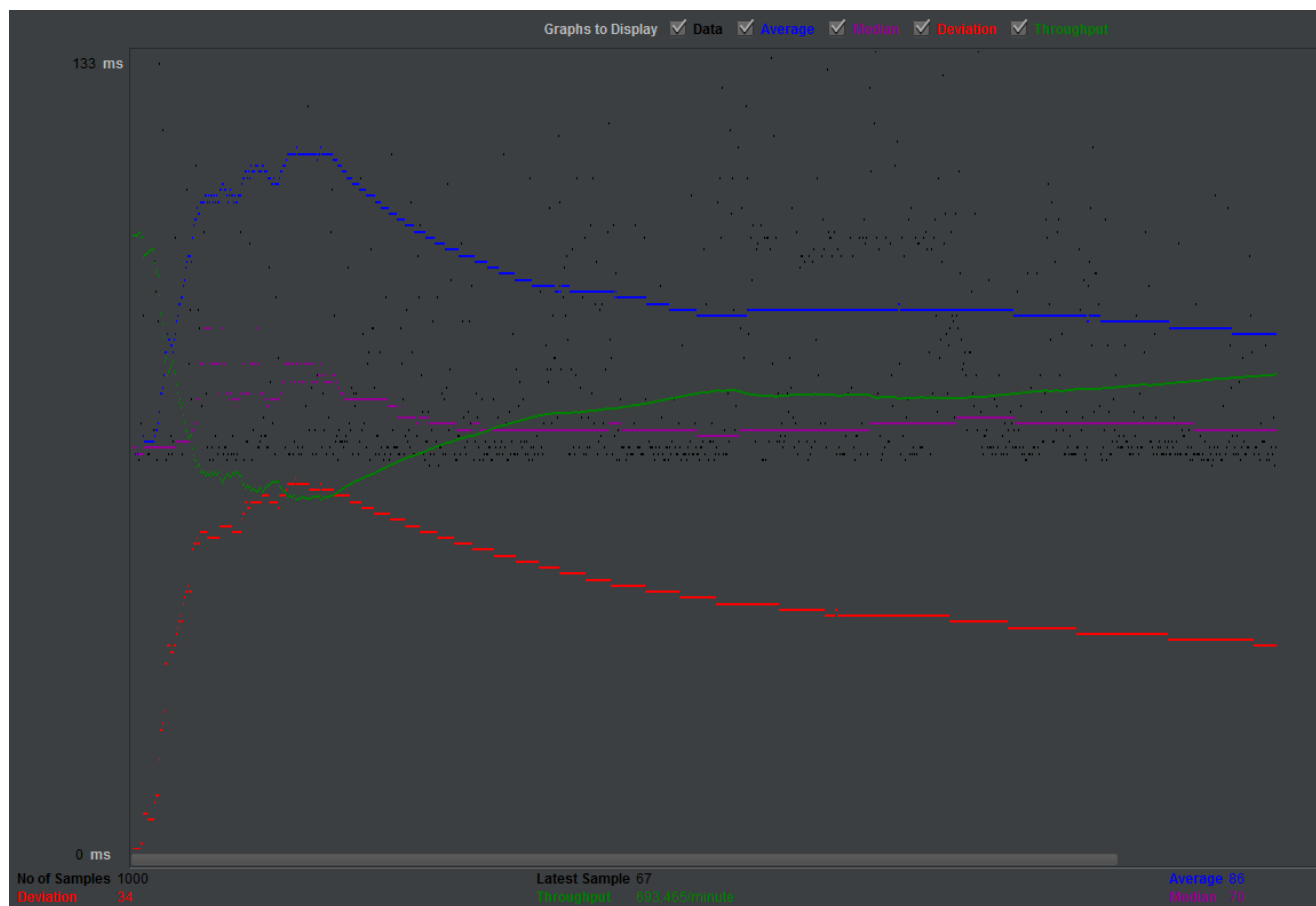
Rysunek 3 Wynik testu dla 1000 użytkowników wysyłających 1 zapytanie w ciągu jednej sekundy

Etykieta	# Liczba próbek	Średnia	Minimum	Maksimum	Odch. std.	% błędów	Przepustowość	KB/sec	Wysłano KB/sec	bitów średnio
Home page	1101	1250	54	5685	1744,14	49,68%	7,0/sec	34,04	0,40	4993,8
RAZEM	1101	1250	54	5685	1744,14	49,68%	7,0/sec	34,04	0,40	4993,8

Rysunek 4 Podsumowanie testu dla 1000 użytkowników

Sample #	Start Time	Thread Name	Url	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
943	18:10:08.477	Users 1-403	Home page	3668		7539	117	3668	31
1002	18:10:08.478	Users 1-404	Home page	4433		7539	117	4433	28
911	18:10:08.479	Users 1-405	Home page	3239		7539	117	3239	28
910	18:10:08.480	Users 1-406	Home page	3233		7539	117	3233	28
1003	18:10:08.480	Users 1-407	Home page	4434		7539	117	4434	28
909	18:10:08.481	Users 1-408	Home page	3228		7539	117	3228	28
807	18:10:08.482	Users 1-409	Home page	1855		7539	117	1855	43
879	18:10:08.483	Users 1-410	Home page	2829		7539	117	2829	40
130	18:10:08.485	Users 1-411	Home page	111		2441	0	0	46
129	18:10:08.486	Users 1-412	Home page	110		2441	0	0	39
139	18:10:08.487	Users 1-413	Home page	109		2441	0	0	38
145	18:10:08.488	Users 1-414	Home page	108		2441	0	0	37
134	18:10:08.489	Users 1-415	Home page	107		2441	0	0	36
131	18:10:08.491	Users 1-417	Home page	105		2441	0	0	34
132	18:10:08.491	Users 1-416	Home page	105		2441	0	0	34
133	18:10:08.492	Users 1-418	Home page	104		2441	0	0	33
137	18:10:08.493	Users 1-419	Home page	103		2441	0	0	33
146	18:10:08.494	Users 1-420	Home page	103		2441	0	0	33
138	18:10:08.495	Users 1-421	Home page	101		2441	0	0	31
153	18:10:08.496	Users 1-422	Home page	106		2441	0	0	36
135	18:10:08.497	Users 1-424	Home page	99		2441	0	0	29
136	18:10:08.497	Users 1-423	Home page	99		2441	0	0	29
149	18:10:08.499	Users 1-425	Home page	98		2441	0	0	27
140	18:10:08.500	Users 1-426	Home page	96		2441	0	0	25
141	18:10:08.501	Users 1-427	Home page	95		2441	0	0	24
142	18:10:08.502	Users 1-428	Home page	94		2441	0	0	23
147	18:10:08.503	Users 1-429	Home page	93		2441	0	0	22
143	18:10:08.504	Users 1-430	Home page	92		2441	0	0	21
144	18:10:08.504	Users 1-431	Home page	92		2441	0	0	21
146	18:10:08.505	Users 1-432	Home page	91		2441	0	0	20
151	18:10:08.506	Users 1-433	Home page	96		2082	0	0	25
154	18:10:08.508	Users 1-434	Home page	94		2441	0	0	23
808	18:10:08.509	Users 1-435	Home page	1835		7539	117	1835	26
152	18:10:08.510	Users 1-436	Home page	92		2441	0	0	20
155	18:10:08.511	Users 1-437	Home page	91		2441	0	0	20
156	18:10:08.512	Users 1-439	Home page	90		2441	0	0	20
157	18:10:08.512	Users 1-438	Home page	90		2441	0	0	21
158	18:10:08.513	Users 1-440	Home page	90		2441	0	0	20
160	18:10:08.514	Users 1-441	Home page	89		2441	0	0	21
159	18:10:08.516	Users 1-443	Home page	87		2441	0	0	20
165	18:10:08.516	Users 1-442	Home page	92		2441	0	0	20

Rysunek 5 Fragment tabeli przedstawiający moment, w którym zapytania zwracają nieprawidłową odpowiedź dla 1000 użytkowników.



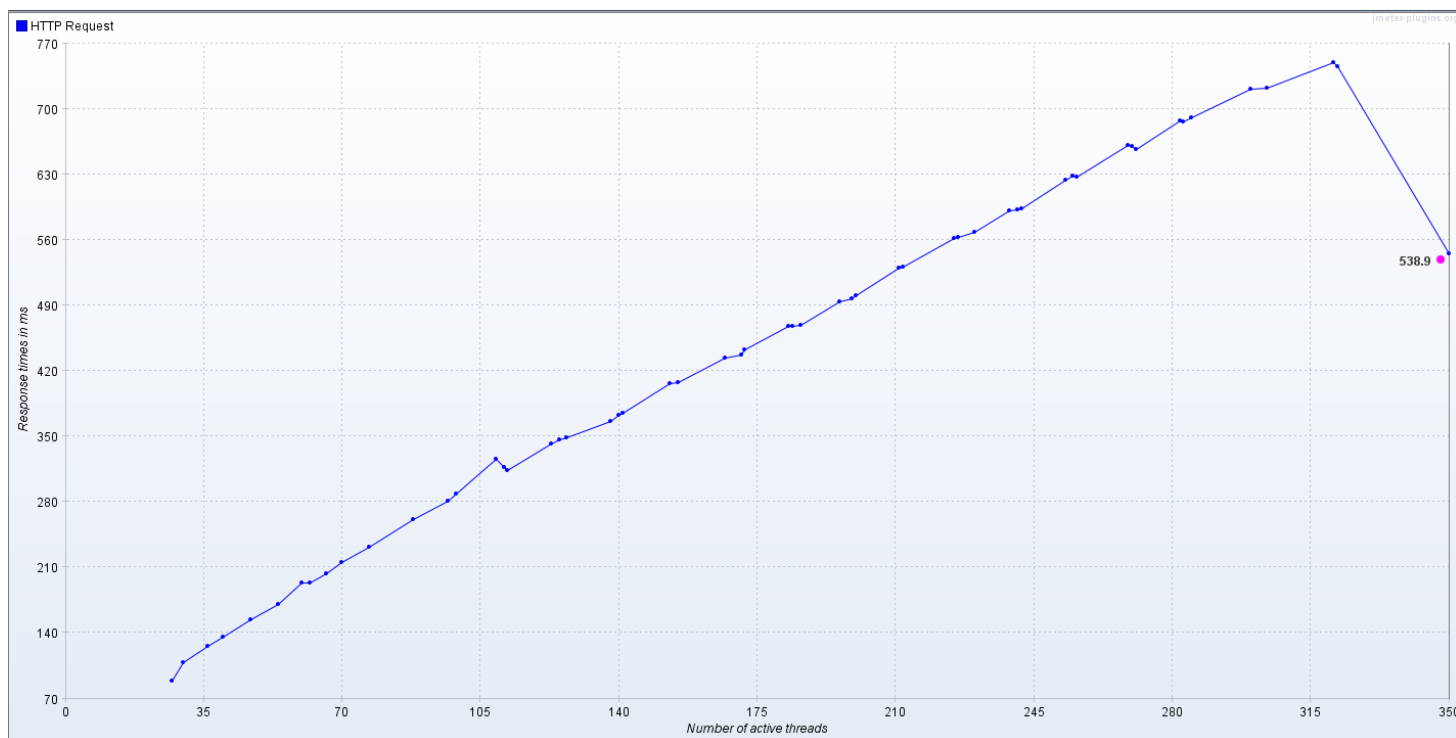
Rysunek 6 Wynik testu dla 1 użytkownika wysyłającego 1000 zapytań w ciągu jednej sekund

Etykieta	# Liczba próbek	Średnia	Minimum	Maksimum	Odch. std.	% błędów	Przepustowość	KB/sec	Wysłano KB/sec	bitów średnio
Home page	1000	86	64	290	34,76	0,00%	11,6/sec	85,09	1,32	7539,0
RAZEM	1000	86	64	290	34,76	0,00%	11,6/sec	85,09	1,32	7539,0

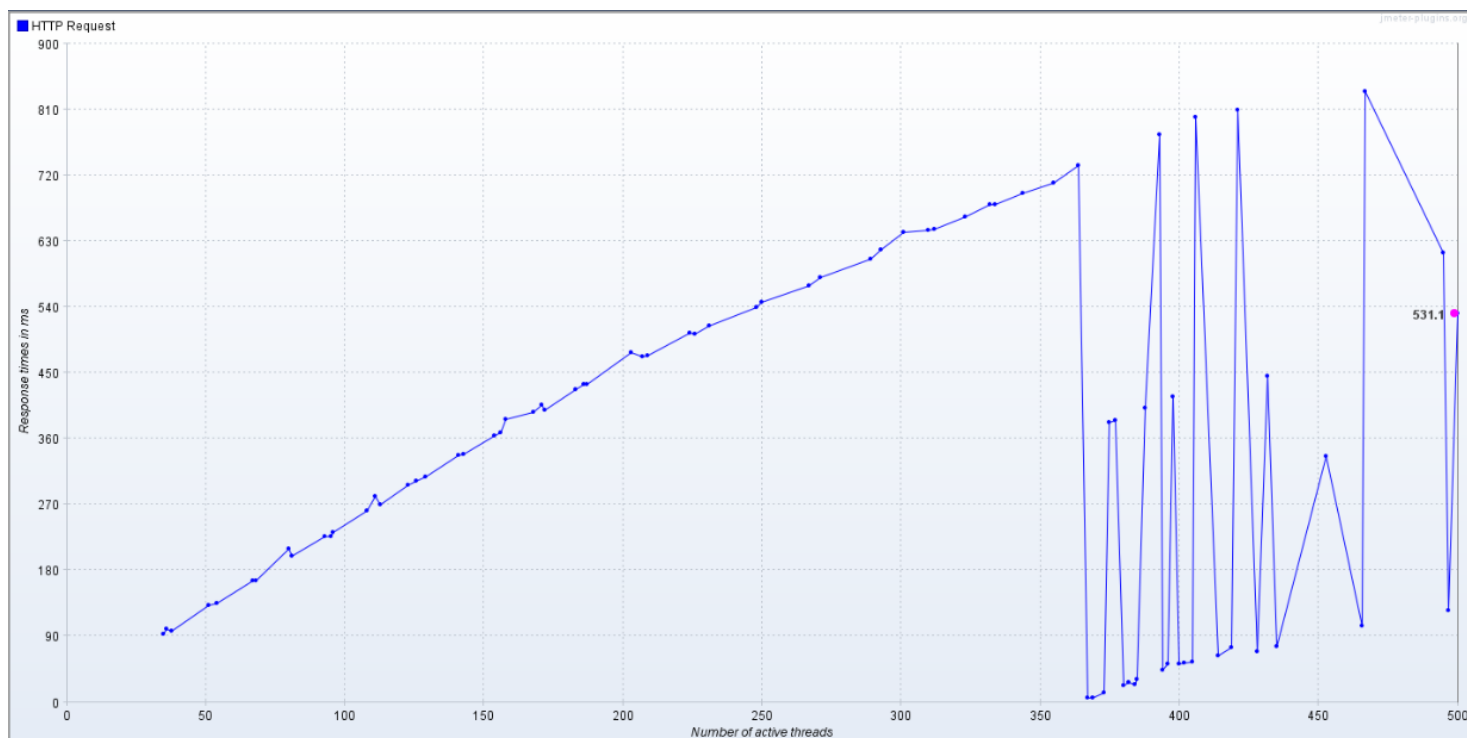
Rysunek 7 Podsumowanie testu dla 1 użytkownika wysyłającego 1000 zapytań w ciągu jednej sekund

Powyższe pomiary pokazują różnice zachowania aplikacji pomiędzy wysyłaniem wielu zapytań przez pojedynczego użytkownika a wysyłaniem jednego zapytania przez wielu użytkowników. W przypadku wysyłania wielu zapytań przez pojedynczego użytkownika aplikacja działa prawidłowo w przypadku wysyłania pojedynczego zapytania przez wielu użytkowników. Różnica ta jest spowodowana najprawdopodobniej przez zastosowanie bazy danych SQL Lite, która realizowana jest jako plik umieszczony na serwerze. W przypadku, gdy wielu użytkowników próbuje pobrać z niej dany plik, jest blokowany w celu zachowania spójności danych i zapytanie nie zostaje wykonane prawidłowo. Na rysunku 5 można również zaobserwować że dla niektórych zapytań serwer zwraca prawidłowe odpowiedzi w niektórych przypadkach pomimo wcześniejszych błędów.

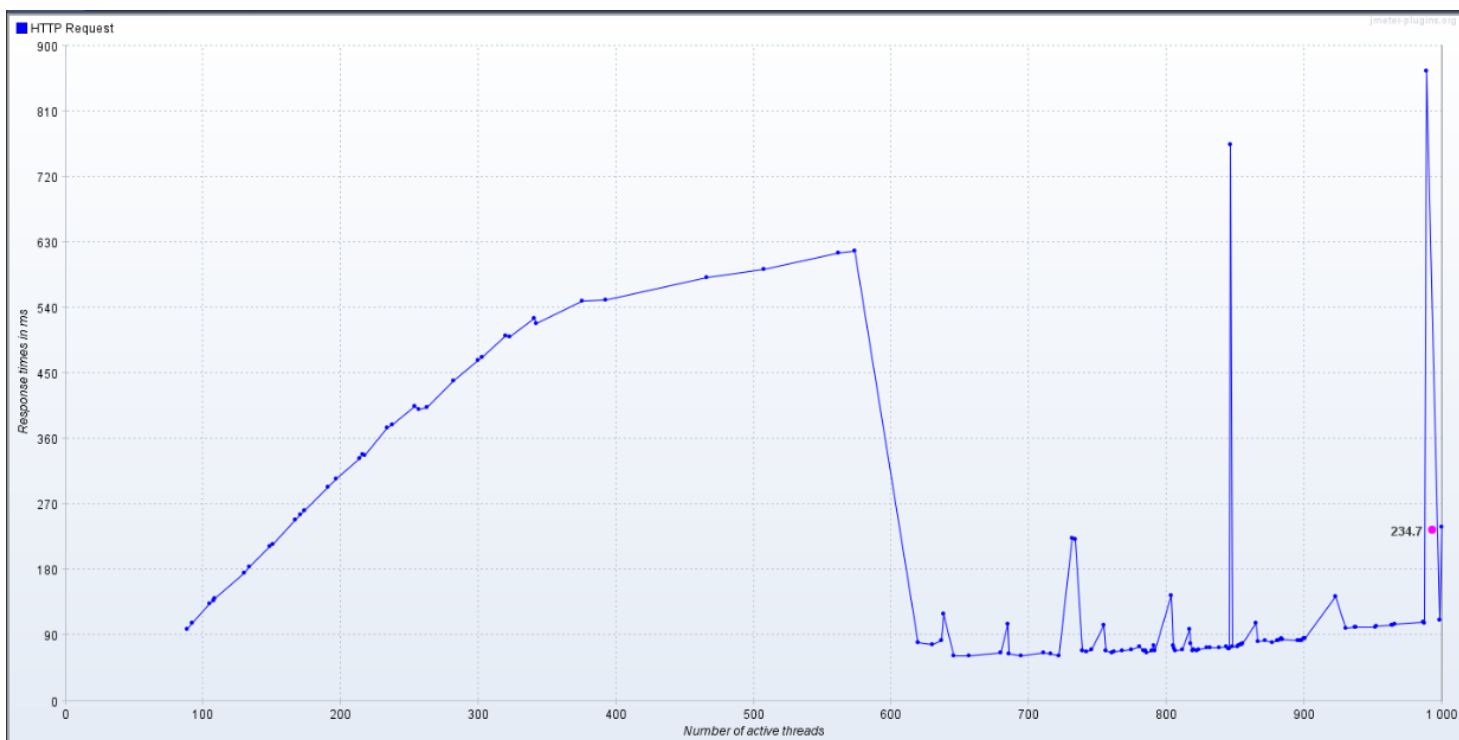
Następnym etapem badania naszej aplikacji było wykonanie dodatkowych pomiarów oraz dokładniejsze przebadanie odpowiedzi serwera.



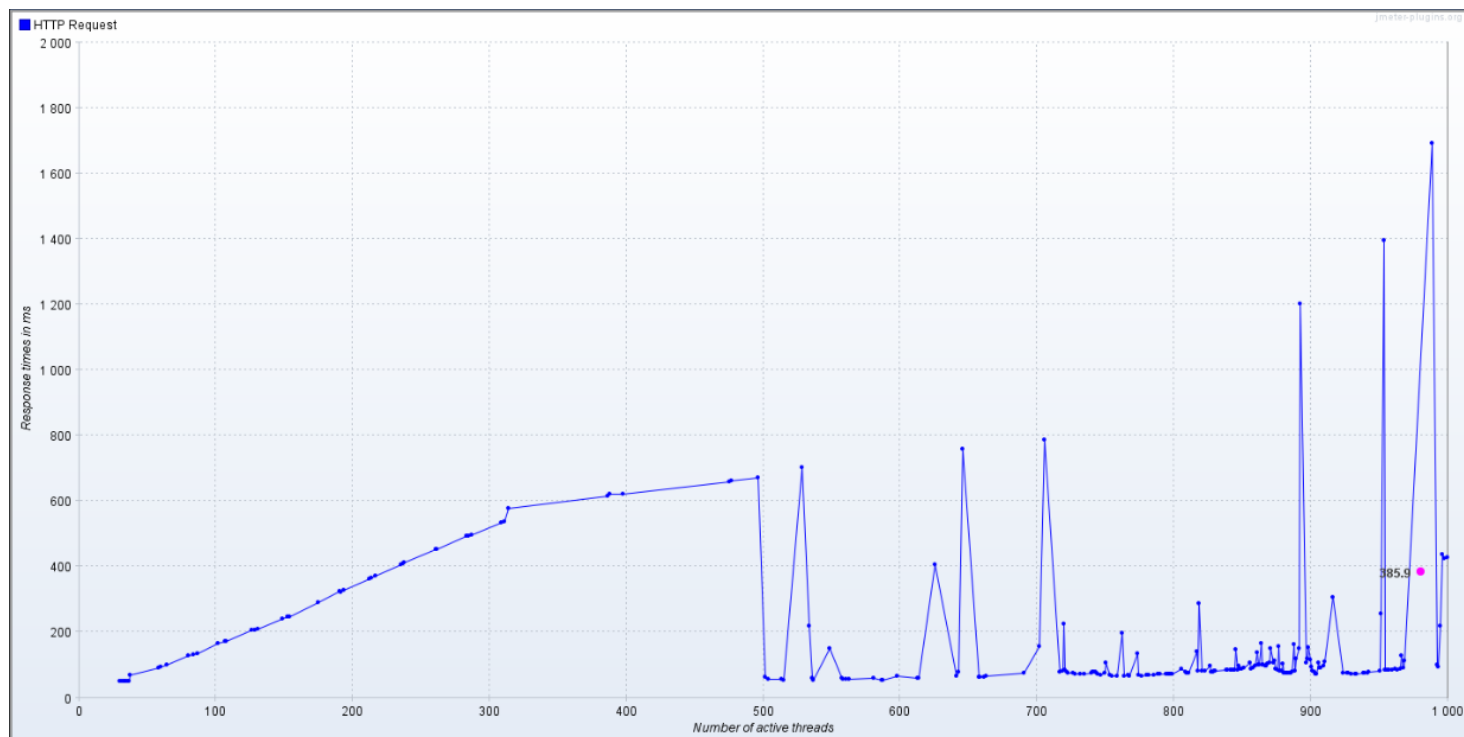
Rysunek 8. Wykres zależności czasu odpowiedzi od liczby jednocześnie użytkowników (max 350 użytkowników).



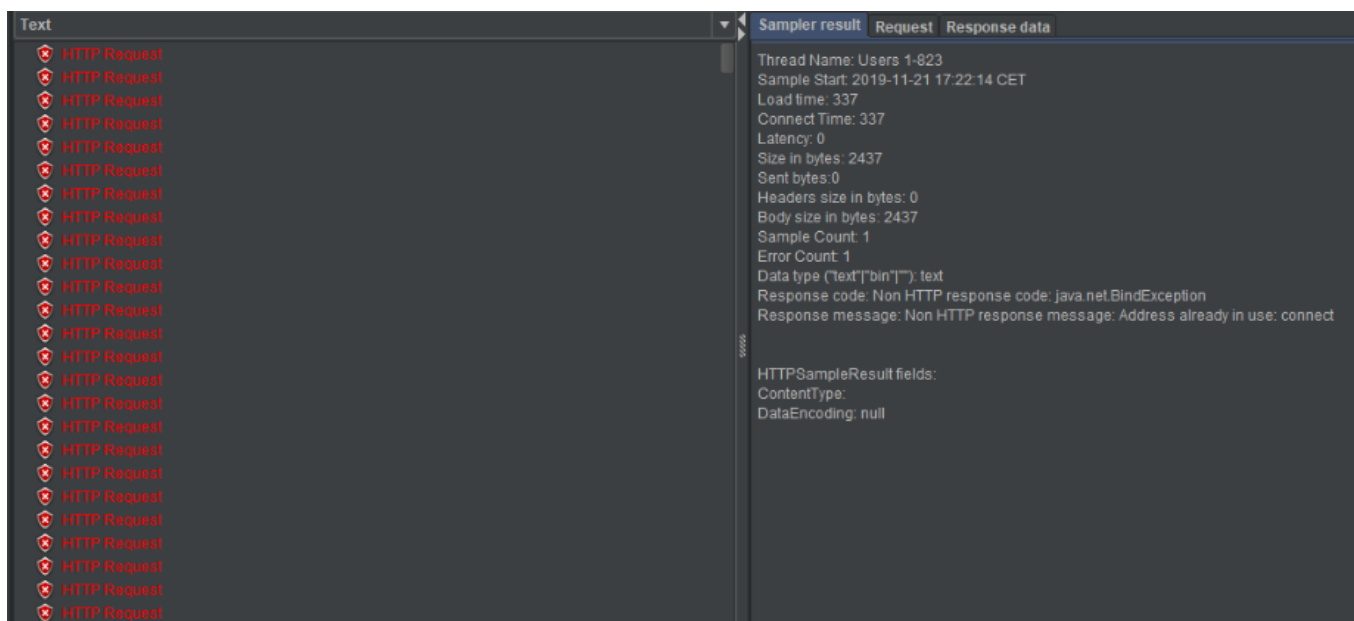
Rysunek 9. Wykres zależności czasu odpowiedzi od liczby jednocześnie użytkowników (max 500 użytkowników).



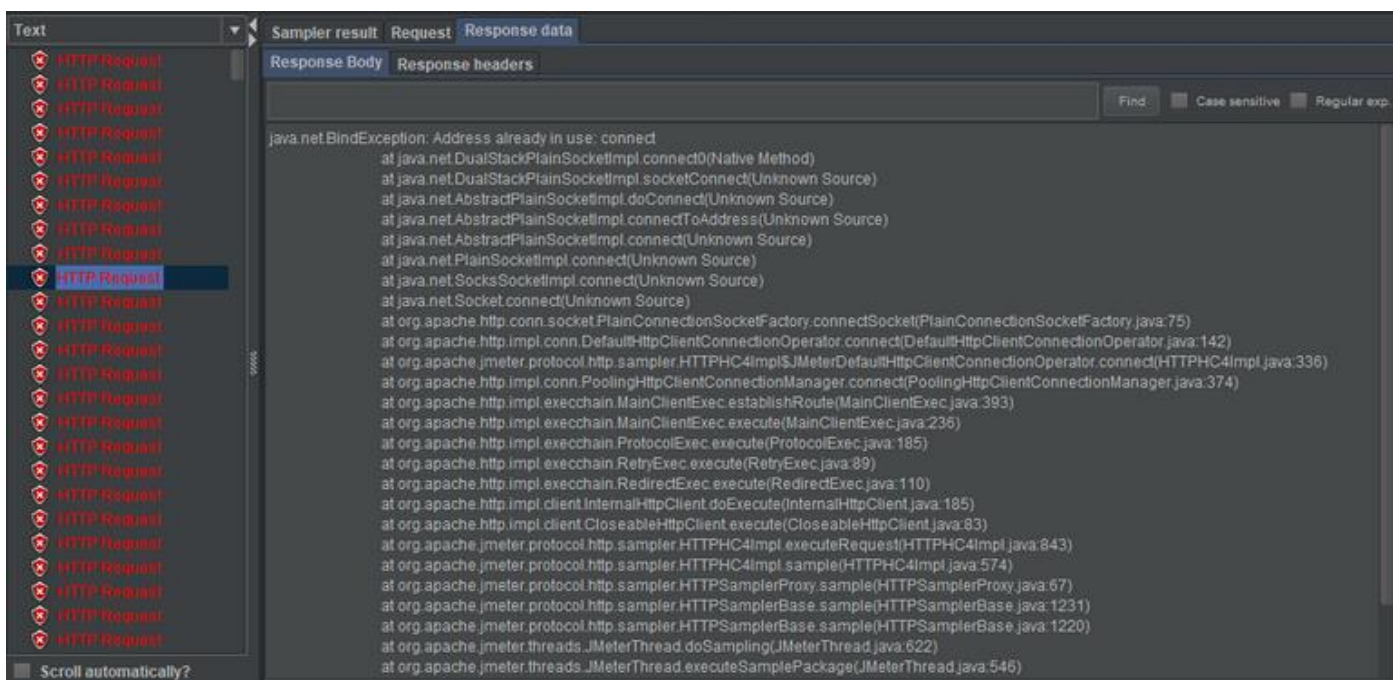
Rysunek 10. Wykres zależności czasu odpowiedzi od liczby jednoczesnych użytkowników (max 1000 użytkowników).



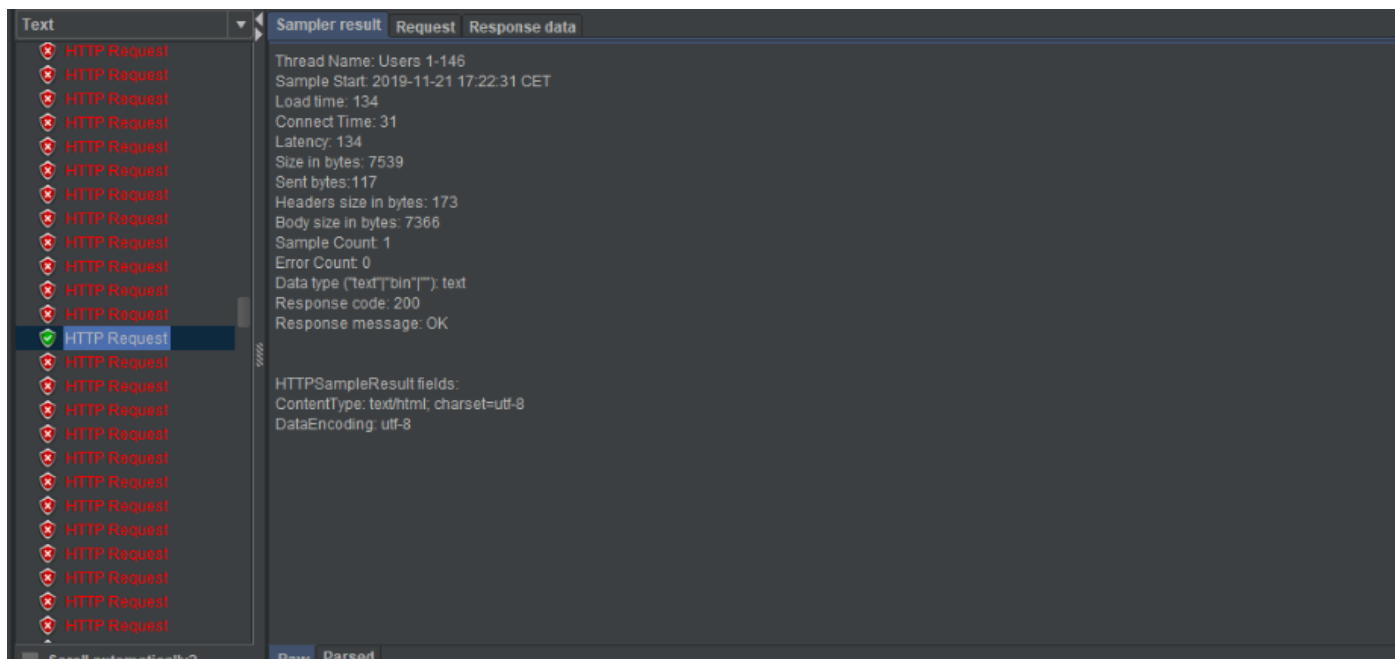
Rysunek 11 Wykres zależności czasu odpowiedzi od liczby jednoczesnych użytkowników (max 1000 użytkowników).



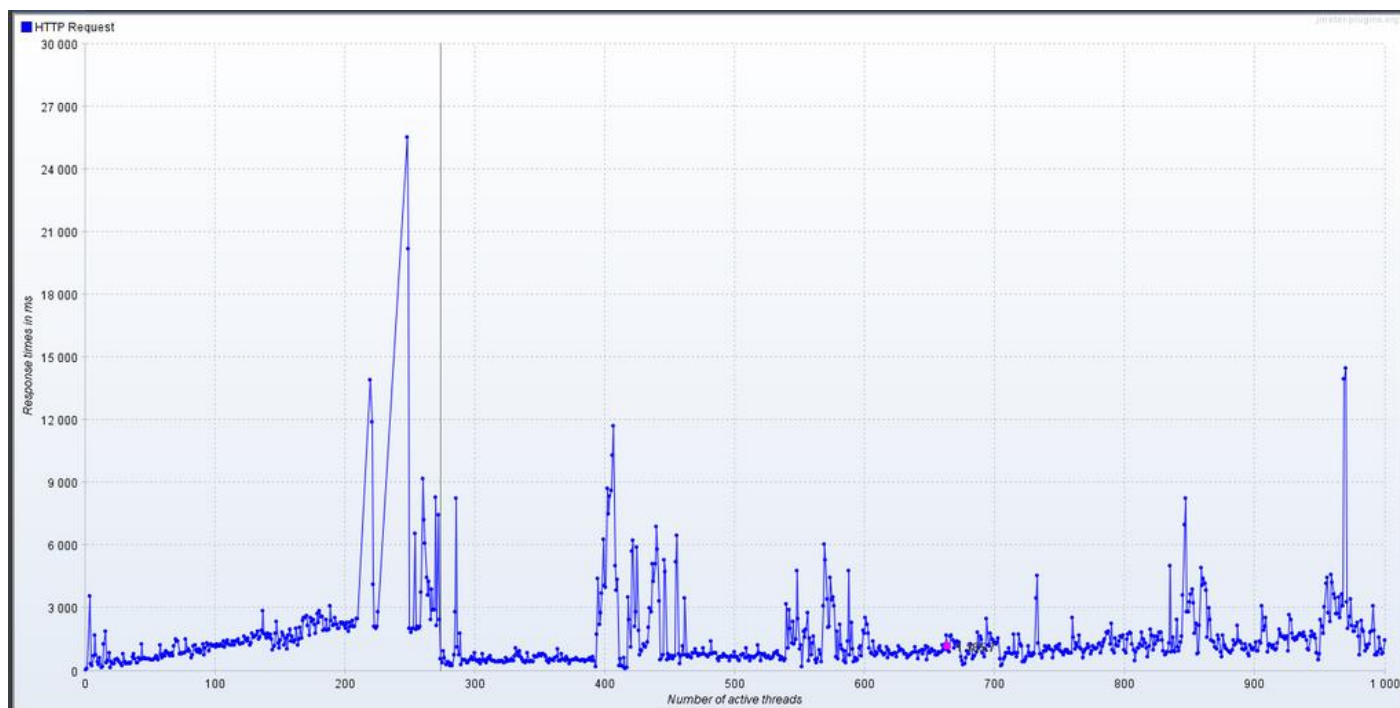
Rysunek 12 Komunikaty błędów zwracane przez serwer po przekroczeniu krytycznej ilości użytkowników



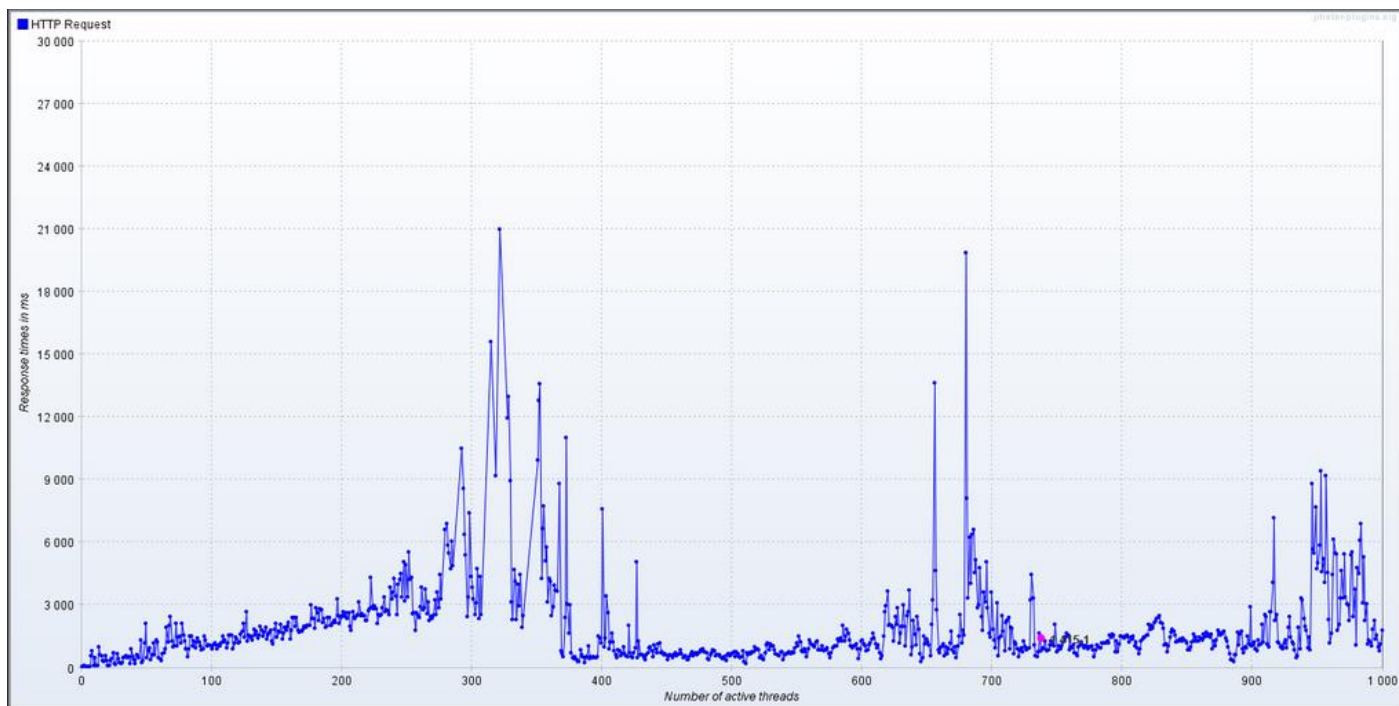
Rysunek 13 Response body zwracane przez serwer po przekroczeniu krytycznej ilości użytkowników



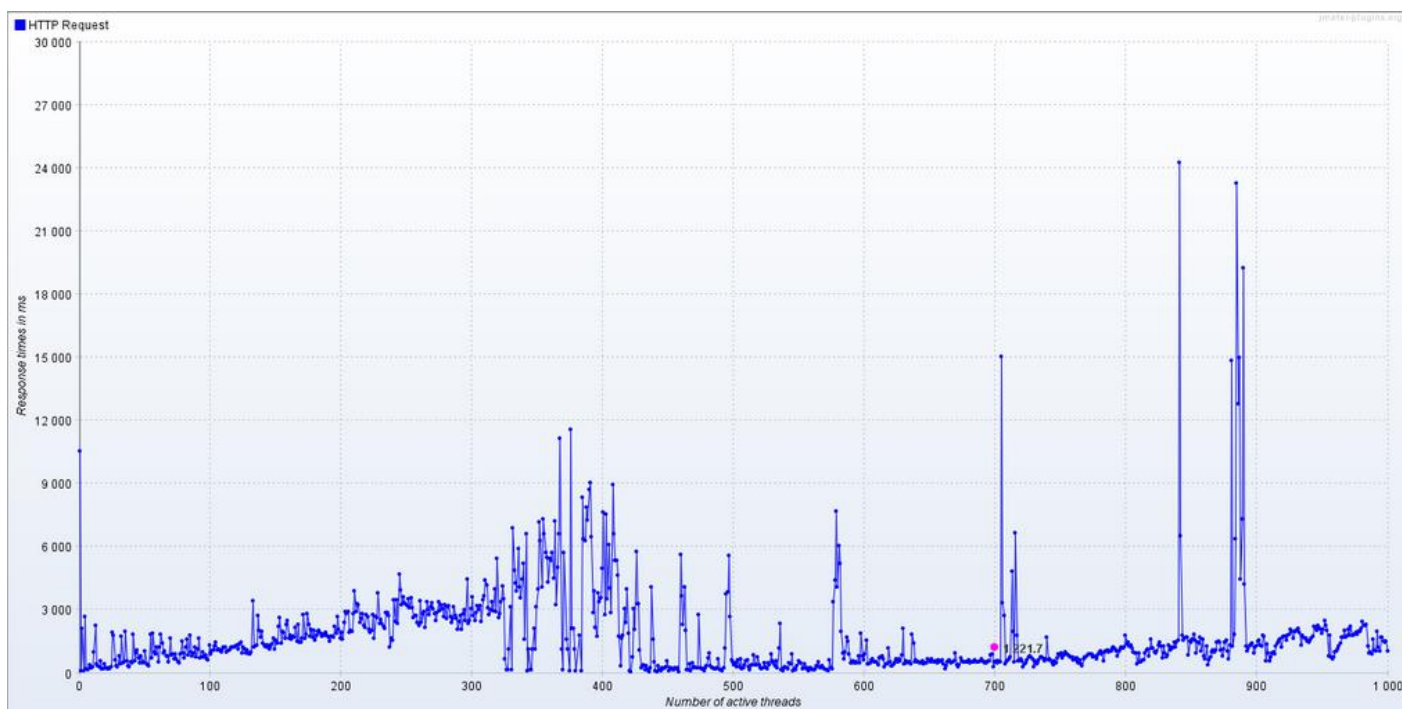
Rysunek 14 Prawidłowa odpowiedź serwera na zapytanie



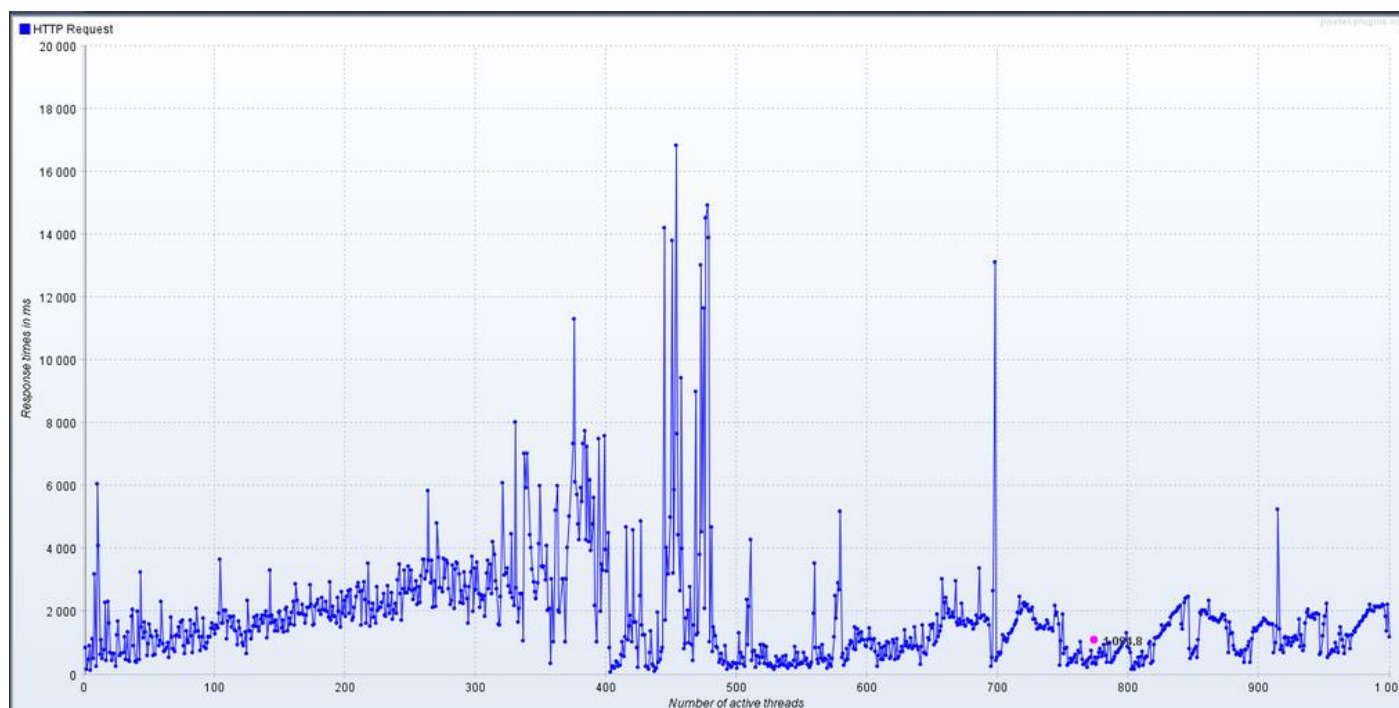
Rysunek 15 Wykres czasu odpowiedzi od przypyływu 1 użytkownik/s.



Rysunek 16 Wykres czasu odpowiedzi od przyływu 2 użytkowników/s.



Rysunek 17 Wykres czasu odpowiedzi od przyływu 5 użytkowników/s.



Rysunek 18 Wykres czasu odpowiedzi od przyływu 10 użytkowników/s.

Powyższe wykresy przedstawiają zależności czasu odpowiedzi od liczby użytkowników. Wykres 8 przedstawia zmiany czasu odpowiedzi dla maksymalnie 350 użytkowników. Liczba ta nie przekracza maksymalnej możliwej ilości użytkowników. Na wykresie można zaobserwować liniowy wzrost czasu odpowiedzi od ilości użytkowników. Jedynie ostatni punkt pomiarowy odbiega od przewidywalnego. Wynika to z wykorzystanej metody pomiarowej, która dla maksymalnej ilości użytkowników wysyła w nieskończoność zapytania, co zaburza średni czas odpowiedzi.

Wykres 9. przedstawia czas odpowiedzi dla maksymalnie 500 użytkowników. Liczba ta przekracza maksymalną ilość jednoczesnych użytkowników, a na wykresie można zaobserwować zaburzenia czasu odpowiedzi. Czas niektórych odpowiedzi jest zauważalnie krótszy, ale odpowiedzi te nie są prawidłowe. Ekstremalne różnice pomiędzy poszczególnymi punktami pomiarowymi wynikają z różnic czasu odpowiedzi dla prawidłowych i nieprawidłowych komunikatów. Nieprawidłowe komunikaty zwracane są prawie natychmiastowo, natomiast prawidłowe odpowiedzi są odsyłane po znacznie dłuższym czasie

Wykresy 10 i 11 przedstawiają czas odpowiedzi dla 1000 jednoczesnych użytkowników. Można na nich zaobserwować, że czas prawidłowych odpowiedzi nadal wzrasta wraz ze wzrostem jednoczesnych użytkowników pomimo tego, że część odpowiedzi serwera nie jest prawidłowa. Porównując oba wykresy można również zauważyć, że moment przekroczenia maksymalnej ilości użytkowników nie jest stały. Dla 10. wykresu jest to około 580 użytkowników, natomiast dla 11. wykresu jest to około 500.

Poprzez nieprawidłowy komunikat rozumiemy komunikaty przedstawione na rysunku 12. i 13. Komunikaty te pokazują, że przyczyną nieprawidłowego zachowania aplikacji jest wyczerpanie dostępnych lokalnych portów. Zapytania są kierowane na porty wykorzystane już przez inne zapytania, w wyniku czego otrzymywany jest komunikat o tym, że dany port jest już zajęty. W przypadku prawidłowej odpowiedzi zwracany jest komunikat o kodzie 200, a w ciele odpowiedzi znajduje się kod HTML strony.

Podsumowanie

Podczas realizacji projektu mieliśmy okazję zapoznać się ze sposobami analizy aplikacji internetowych i narzędziem JMeter. W trakcie projektu postawiliśmy aplikację na serwerze. Uzyskane przez nas wyniki pokazują, że aplikacja nie działa prawidłowo dla dużej ilości jednoczesnych użytkowników. Najprawdopodobniej działanie aplikacji można usprawnić poprzez zmianę bazy danych z `sqlite` na `MySQL` oraz odpowiednie zamykanie niewykorzystywanych socketów niestety nie zdążyliśmy dokonać rekonfiguracji aplikacji w celu zbadania naszej hipotezy.