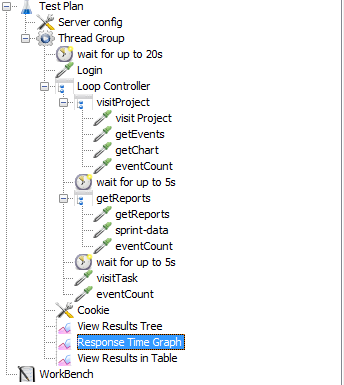
### Tasker application performance test

Following scenario was used:

1. User is logging in ( **login** )
2. Enters project page ( **visitProject**)
3. Waits up to 5 seconds ( random )
4. enters sprint report page ( getReports )
5. Waits up to 5 seconds ( random )
6. Visits page with one task ( visitTask)
7. Repeat 5 times from step 2

For each of visited page there are extra ajax request which are not counted towards full page load

In atachement there is enclosed jmeter configuration file which only needs server configuration

Each scenario was repeated for 5 , 10 and 15 simileous threads - similuation of multiple users clicking in applicaiton at once( clicking a lot )

The most time was for visitTask , where there might be some optimalization done in future (UPDATE : optimalization has been done )

Charts represent average time how much it took time to load page + load all ajax requests

Additionaly when test was running , extra user ( me ) was clicking on application in different places to test and read other sets of data ( charts can be seen as more irregular )

At the begining of test initial loading of users was alwyas slow

During testing of application application was stable and had no errors ( just slower with high load )

Test were run normal and 2x stronger instance in cloud ( tiktalik vps which had not that much capacity)

Test were then run on local machine and was running smoothly with approx 45 threads at once

System settings:

**Cloud:**

Each Std Unit has 1 GB RAM i and CPU power comparable with procesor of 1000MHz.

Hosting : [https://tiktalik.com](https://tiktalik.com/)

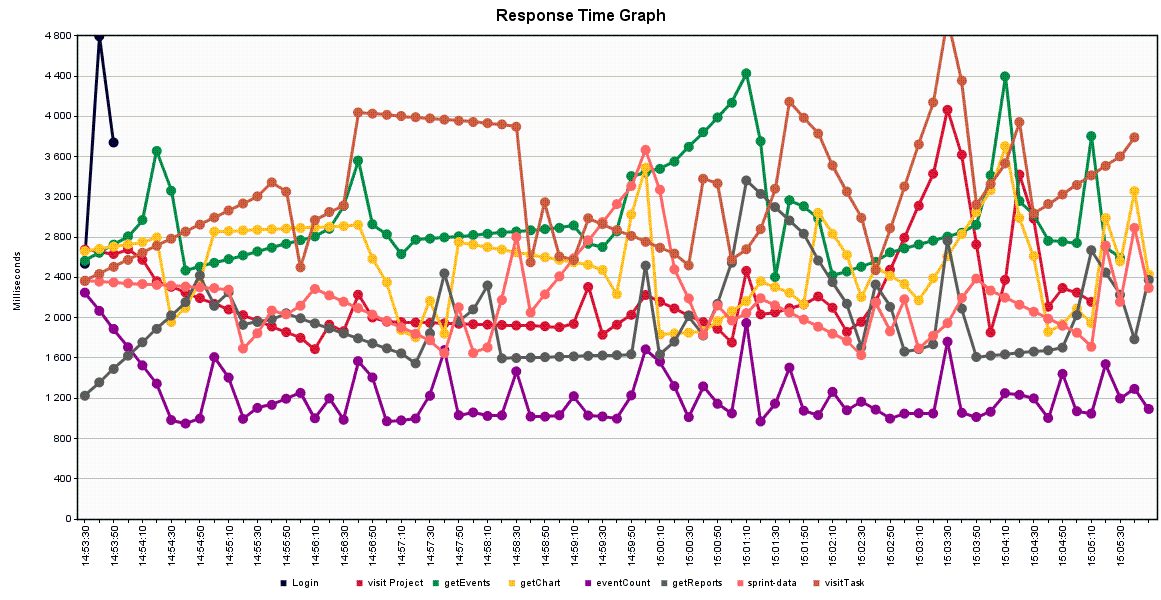
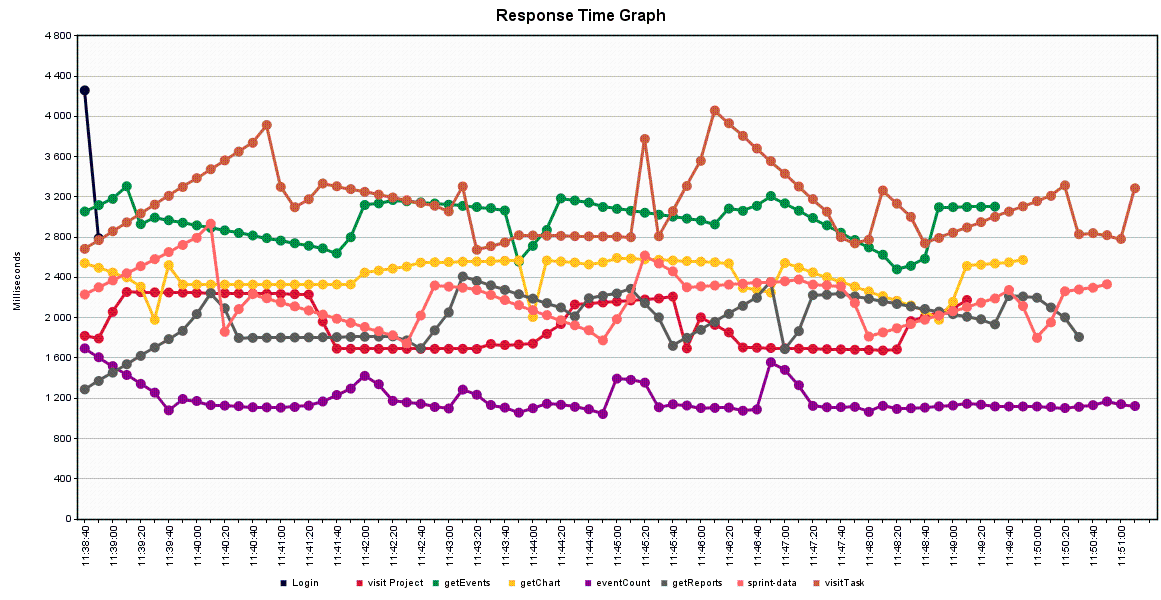
**PC**Procesor: i5-240 @3.1GHz

Ram: 16 GB RAM

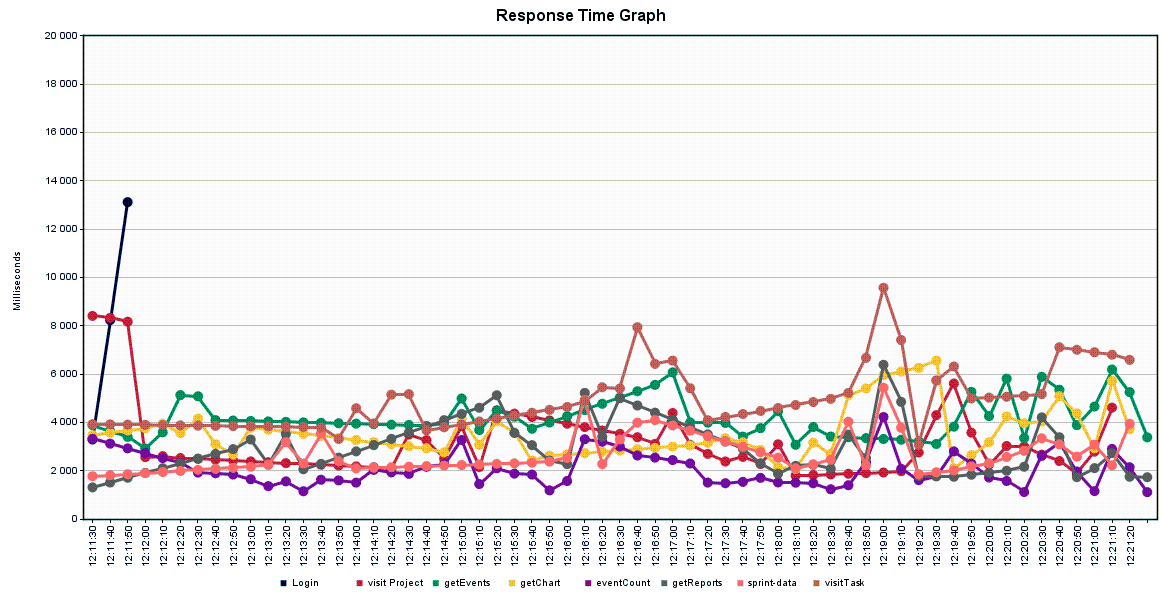
Drive SSD

**Tasker-perf 4 x Std Unit**

5 threads at once  - avarage longest time to open page ~ 3s (7s including all REST data )

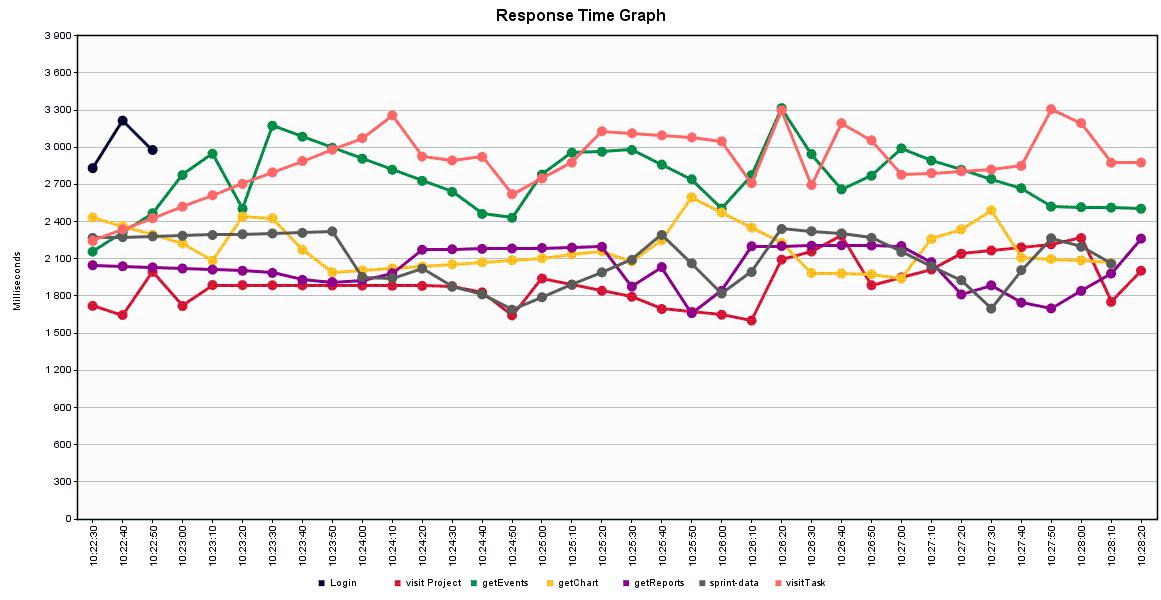
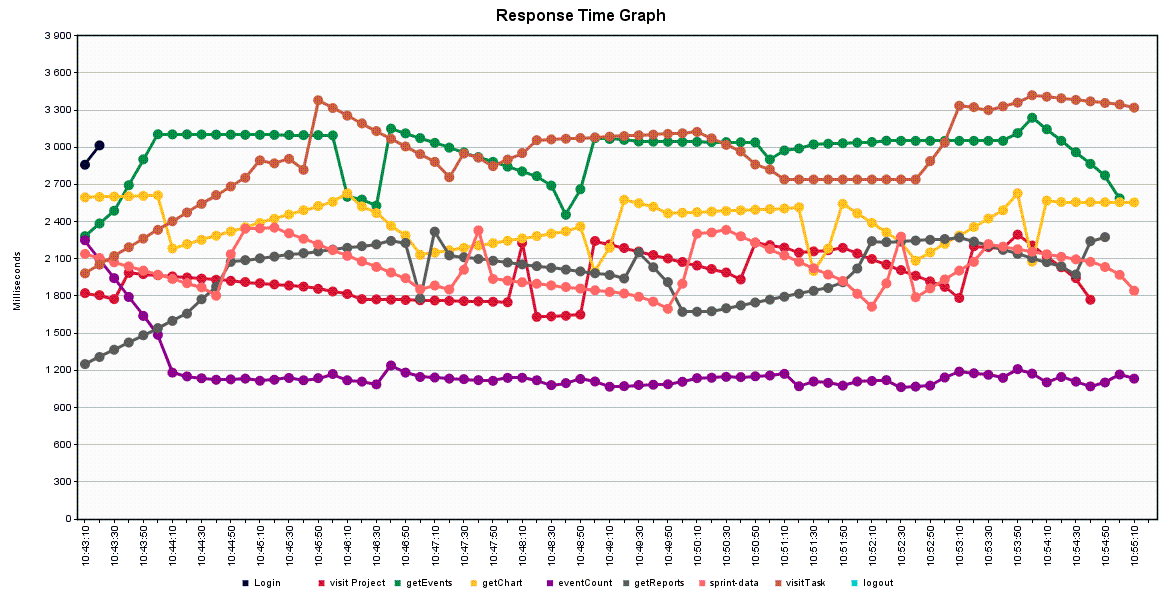
  
10 threads at once  - avarage longest time to open page ~ 3s ( 8s including all REST data )   

15 threads at once  - avarage longest time to open page ~ 6s ( 9s including all REST data )

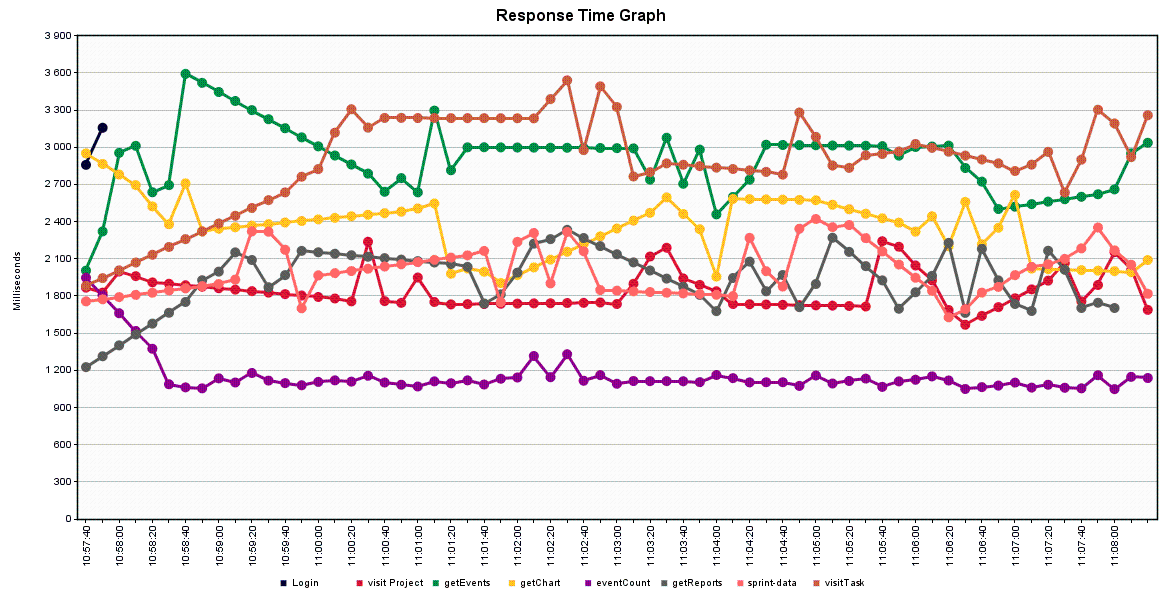


**Tasker-perf 8 x Std Unit**

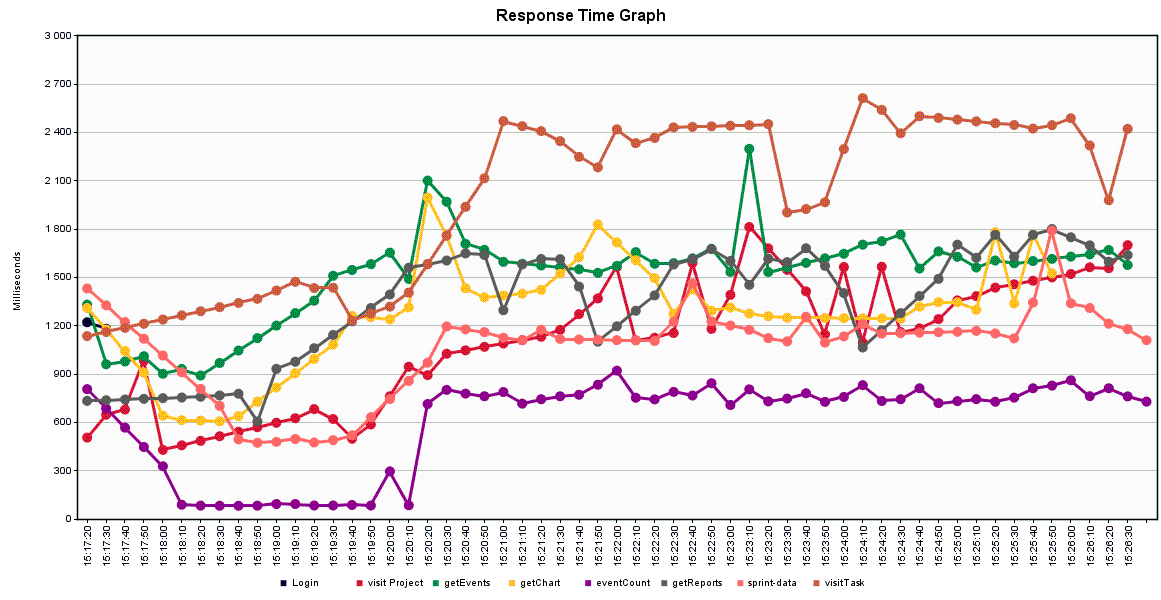
5 threads at once  - avarage longest time to open page ~ 2.5s (5s including all REST data )



10 threads at once  - avarage longest time to open page ~ 2.5s (6s including all REST data )   

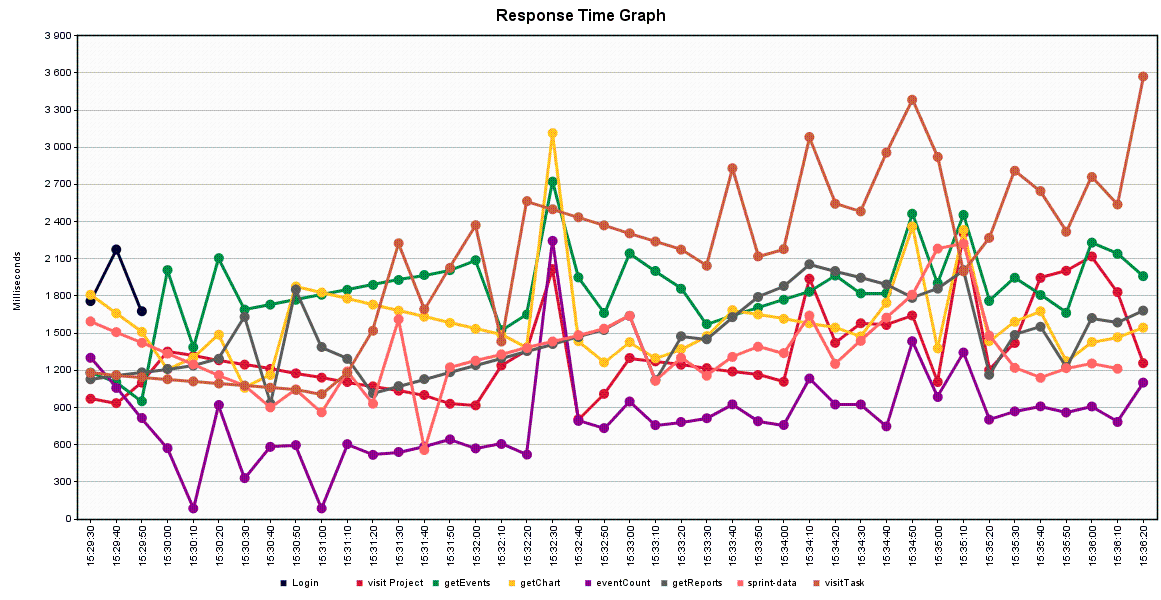


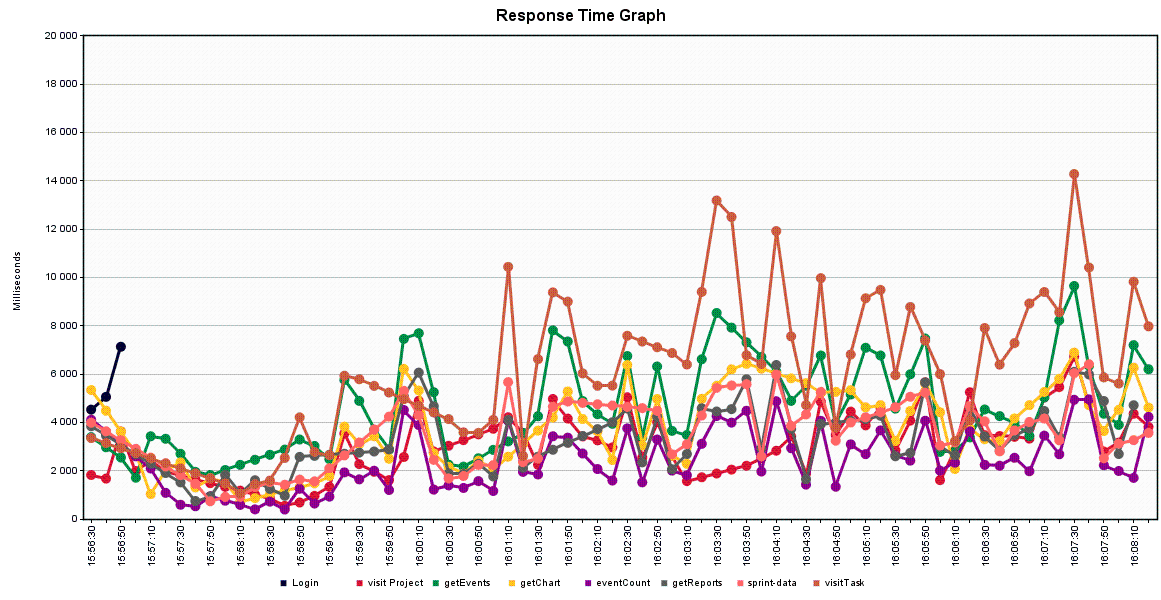
15 threads at once  - avarage longest time to open page ~ 3s (6s including all REST data )     

**PC**

15  threads at once  - avarage longest time to open page~ 2.5s (4s including all REST data )

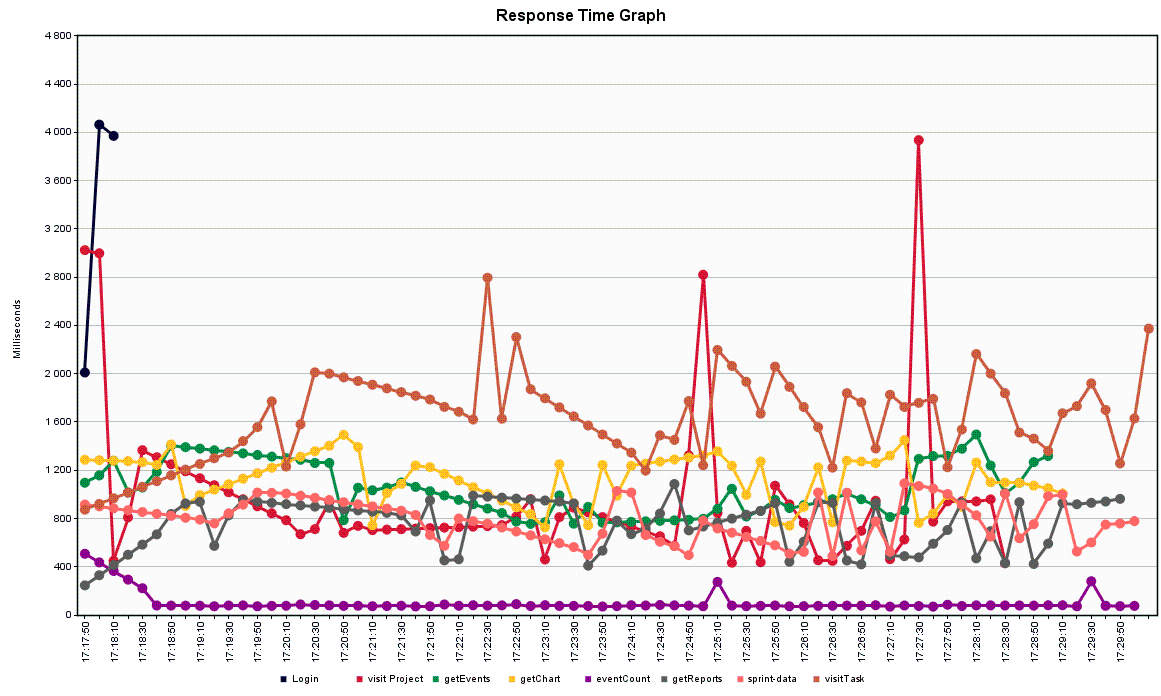
25  threads at once  - avarage longest time to open page ~ 3s (5s including all REST data )

  
45 threads at once  - avarage longest time to open page ~ 13s (15s including all REST data )    



After performing tests above , some parts of code was refactored

For 4 st units , avarage longest time to pen page was down to 2,5 seconds ( 4s including ajax data ) vs 6s ( 9 with including all REST data ) on old code

Improve in performance 2-3 times

Test reexecution on normal production machine :