

# D&T II

# D03: Performance testing

# **Performance test suite**

# Group 20

**Candelario Luna, Luis**

**Carrasco Márquez, Antonio**

**Gil Guerrero, Luis**

**Márquez Orellana, Francisco**

**Morales Moreno, Sergio**

**Moreno Ruiz, Juanma**

Content

[Introduction: 5](#_Toc8582305)

[Performance tests: 6](#_Toc8582306)

[Use case 1: 6](#_Toc8582307)

[As a not authenticated actor I want to register as a Rookie 6](#_Toc8582308)

[Analysis results: 12](#_Toc8582309)

[Use case 2: 13](#_Toc8582310)

[As a not authenticated actor I want to list the positions and the companies available and navigate to their respective companies and positions 13](#_Toc8582311)

[Analysis results: 17](#_Toc8582312)

[Use case 3 18](#_Toc8582313)

[As a not authenticated actor I want to search for a position using a single key word. 18](#_Toc8582314)

[Analysis results: 24](#_Toc8582315)

[Use case 4 & 5: 25](#_Toc8582316)

[As a not authenticated user I want to register as a Company. 25](#_Toc8582317)

[As an authenticated user I want to edit my personal data. 25](#_Toc8582318)

[Analysis results: 30](#_Toc8582319)

[Use case 6: 31](#_Toc8582320)

[As a company I want to manage my positions, which means listing, showing, creating, updating and deleting them. 31](#_Toc8582321)

[Analysis results: 35](#_Toc8582322)

[Use case 7: 36](#_Toc8582323)

[As a company I want to manage my problems, which means listing, showing, creating, updating and deleting them. 36](#_Toc8582324)

[Analysis results: 41](#_Toc8582325)

[Use case 8: 42](#_Toc8582326)

[As a company I want to manage the applications to my positions, which means listing them grouped by status, showing and updating them. 42](#_Toc8582327)

[Analysis results: 48](#_Toc8582328)

[Use case 9 & 10: 49](#_Toc8582329)

[As a rookie I want to manage my applications, which includes listing them grouped by status, showing them, creating them, and updating them. 49](#_Toc8582330)

[As a rookie I want to be assigned an arbitrary problem when I create an application. 49](#_Toc8582331)

[Analysis results: 55](#_Toc8582332)

[Use case 11 & 12: 56](#_Toc8582333)

[As an admin I want to create user accounts for new administrators. 56](#_Toc8582334)

[As an admin I want to display a dashboard with the following information: 56](#_Toc8582335)

[Analysis results: 62](#_Toc8582336)

[Use case 13 & 14: 63](#_Toc8582337)

[As a rookie I want to manage my curricula, which includes listing, showing, creating, updating, and deleting them. 63](#_Toc8582338)

[As a rookie I want to attach a curriculum to the application that is a copy of the original. 63](#_Toc8582339)

[Analysis results: 68](#_Toc8582340)

[Use case 15: 69](#_Toc8582341)

[As a rookie I want to manage my finder, which involves updating the search criteria, listing its contents, and clearing it. 69](#_Toc8582342)

[Analysis results: 74](#_Toc8582343)

[Use case 16: 75](#_Toc8582344)

[As an authenticated actor I want to manage my social profiles, which includes listing, showing, creating, updating, and deleting them 75](#_Toc8582345)

[Analysis results: 80](#_Toc8582346)

[Use case 17: 81](#_Toc8582347)

[As an authenticated actor I want to manage my messages, which includes listing them grouped by tag, showing them, sending a message to an actor, deleting a message that I got. 81](#_Toc8582348)

[Analysis results: 86](#_Toc8582349)

[Use case 18, 19, 20 & 21: 87](#_Toc8582350)

[As an admin I want to broadcast a notification message to the actors of the system. The message must have tag “SYSTEM” by default. 87](#_Toc8582351)

[As an admin I want to launch a process that flags the actors of the system as spammers or not-spammers. 87](#_Toc8582352)

[As an admin I want to Ban an actor with the spammer flag. 87](#_Toc8582353)

[As an admin I want to unban an actor who was banned previously. 87](#_Toc8582354)

[Analysis results: 93](#_Toc8582355)

[Use case 22: 94](#_Toc8582356)

[As an auditor I want to log into the system, self-assign a position and log out. 94](#_Toc8582357)

[Analysis results: 99](#_Toc8582358)

[Use case 23: 100](#_Toc8582359)

[As an auditor I want to manage my audits which includes listing them, showing them, creating them, updating, and deleting them. 100](#_Toc8582360)

[Analysis results: 105](#_Toc8582361)

[Use case 24: 106](#_Toc8582362)

[As and admin I want to notify my users about of the rebranding of the system only one time. 106](#_Toc8582363)

[Analysis results: 111](#_Toc8582364)

[Use case 25: 112](#_Toc8582365)

[As an admin I want to create user accounts for new auditors. 112](#_Toc8582366)

[Analysis results: 117](#_Toc8582367)

[Use case 26 & 27: 118](#_Toc8582368)

[Launch a process to compute an audit score for every company. 118](#_Toc8582369)

[Display a dashboard with the following information : 118](#_Toc8582370)

[Analysis results: 123](#_Toc8582371)

[Use case 28: 124](#_Toc8582372)

[As an actor who is not authenticated I want to: 124](#_Toc8582373)

[Analysis results: 129](#_Toc8582374)

[Use case 29: 130](#_Toc8582375)

[As a provider I want to create and delete my catalogue of items. 130](#_Toc8582376)

[Analysis results: 135](#_Toc8582377)

[Use case 30: 136](#_Toc8582378)

[As a provider I want to update and show my catalogue of items. 136](#_Toc8582379)

[Analysis results: 141](#_Toc8582380)

[Use case 31: 142](#_Toc8582381)

[As a provider I want to create and delete my sponsorships. 142](#_Toc8582382)

[Analysis results: 147](#_Toc8582383)

[Use case 32: 148](#_Toc8582384)

[As a provider I want to update and show my sponsorships. 148](#_Toc8582385)

[Analysis results: 153](#_Toc8582386)

[Conclussion: 154](#_Toc8582387)

# Introduction:

To test all user cases for Acme-Rookies we used Jmeter with the following method:

* Group some requisites to do the testing to be more efficient.
* A test with 100 users and 10 loops to see if there is any error in the code.
* A test with a number between 130 and 180 users to see the maximum users that supports the actions without any problem relative to the user experience, in this case we are going to check the time spend to do the actions, looking at the 90% Line results.
* A test with the first errors encountered and the most probably reason that made that errors occur. Checking the computer performance we will observe the processor, memory, disk and network behavior.

All the test were made with the same computer in the pre-production virtual machine**, with 4 GB of ram and a 2 core processor.**

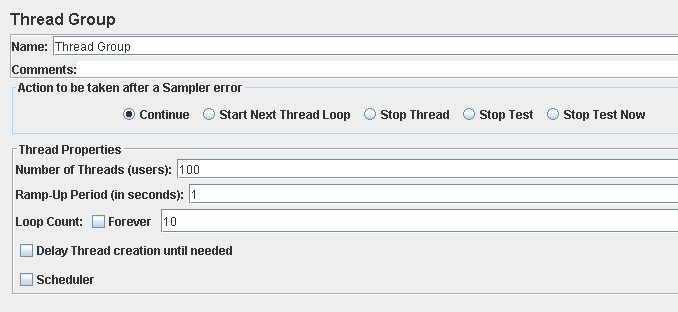
The computer specifications are:

* CPU: Intel core I5 7200u
* Mainboard: Medion D15KHN
* Memory: 8 GB DDR3
* Graphics: Nvidia Geforce GTX 950M
* 245 GB SSD

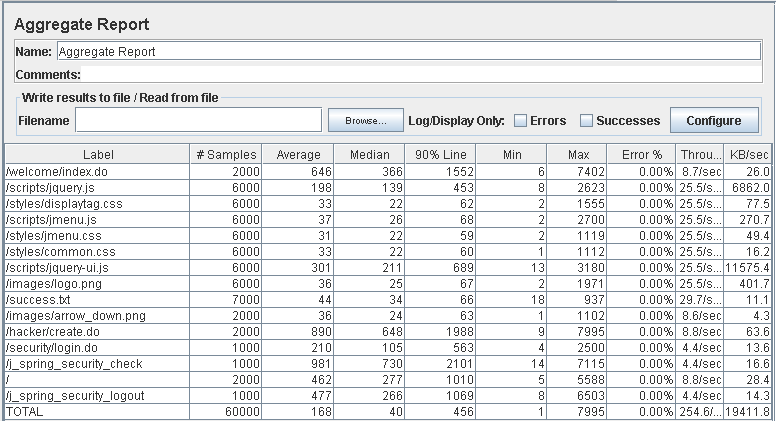
# Performance tests:

## Use case 1:

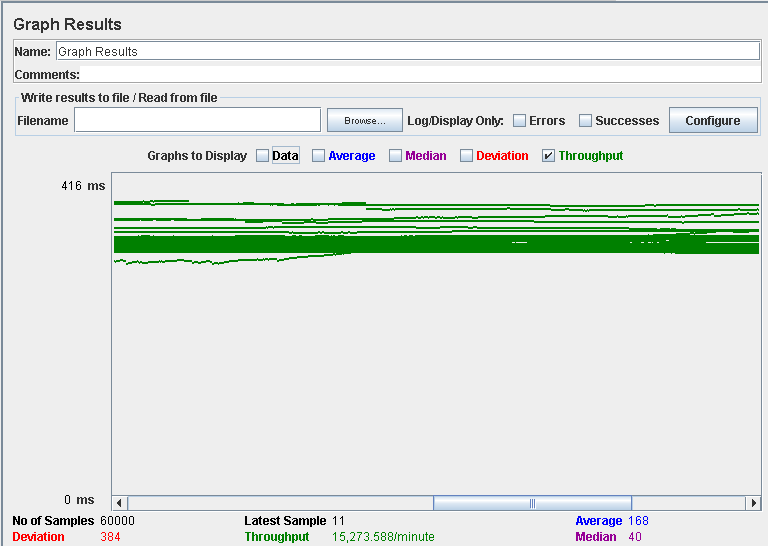
### As a not authenticated actor I want to register as a Rookie

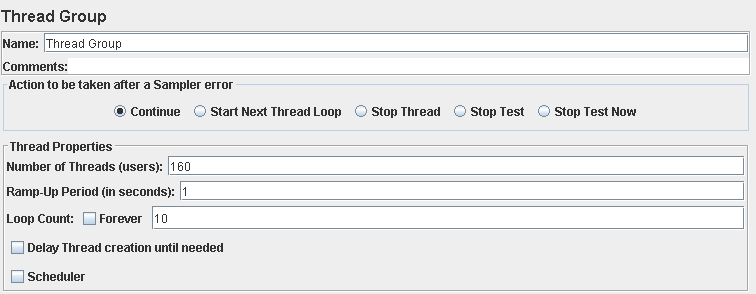


Performance test 90% results: Total 9,879s.

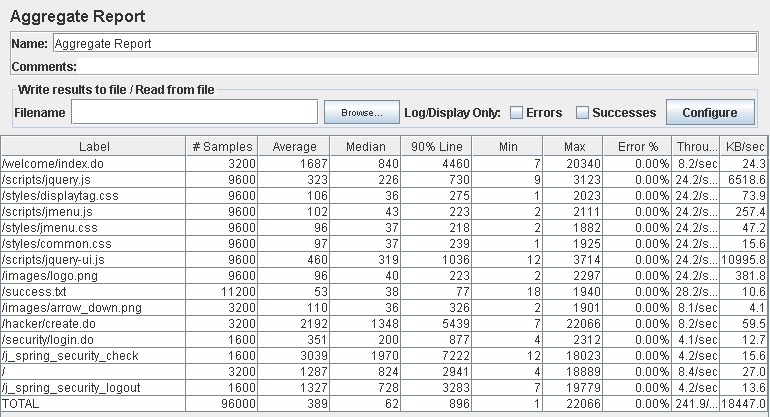


Performance thread results: 15,273 per minute



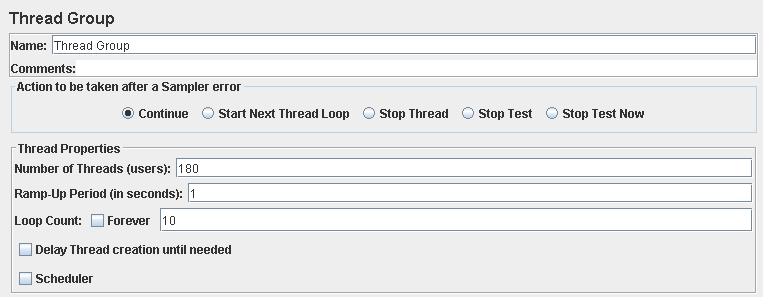


Performance test 90% results: Total 20,851s

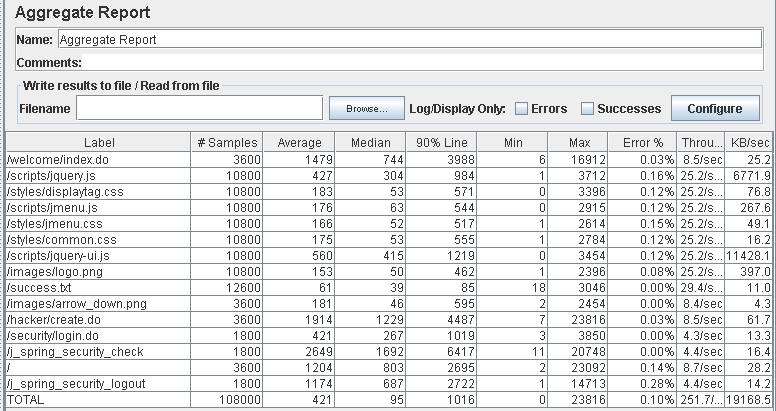


Performance thread results: 14,513 per minute

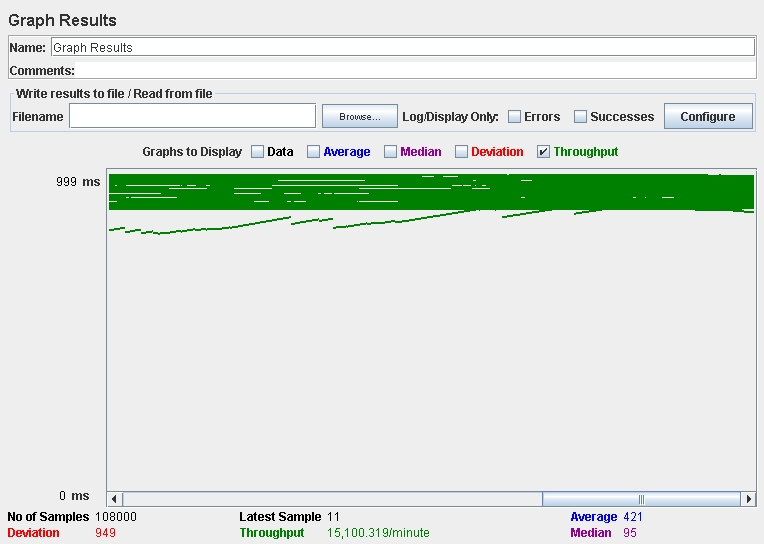




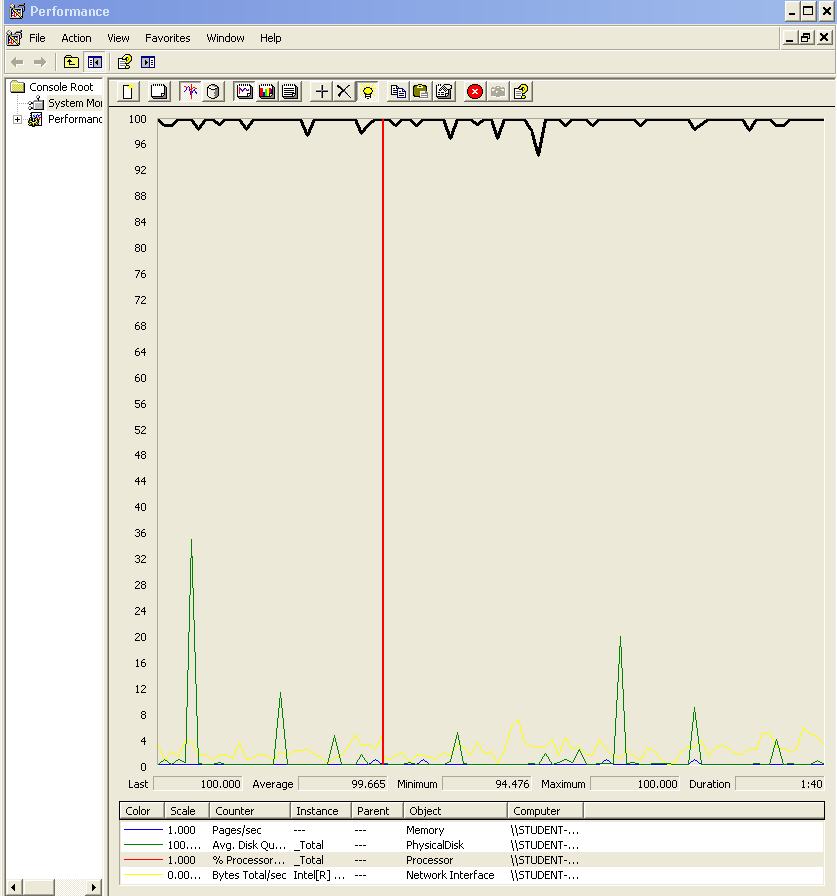
Performance test 90% results: Total 26,860s



Performance thread results: 15,100 per minute



Computer performance:



### Analysis results:

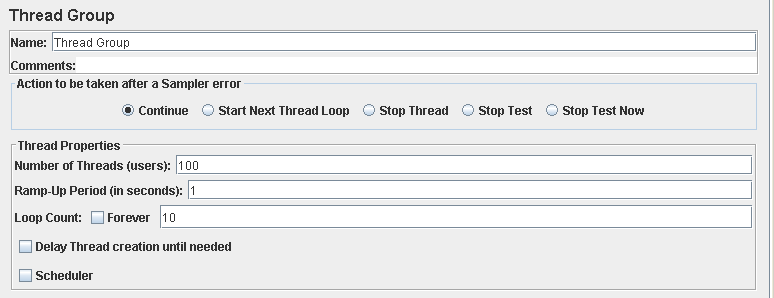
100 users and 10 loops: the application runs perfectly

160 users and 10 loops: the application runs without errors but the times are really high (7200ms).

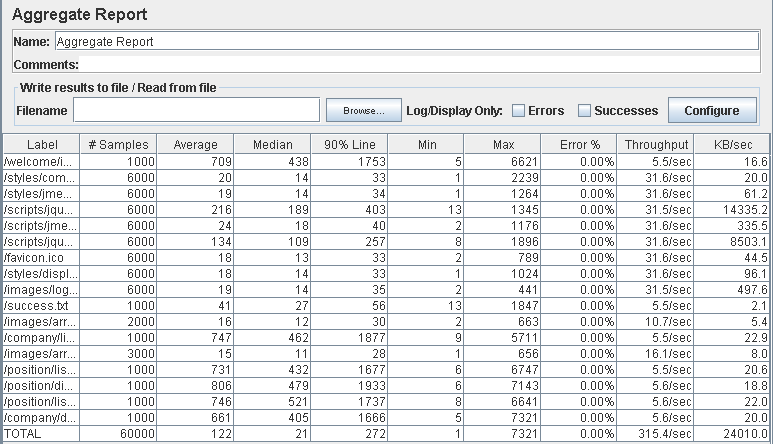
180 users and 10 loops: the application begins to have errors we believe is a processors bottleneck problem.

## Use case 2:

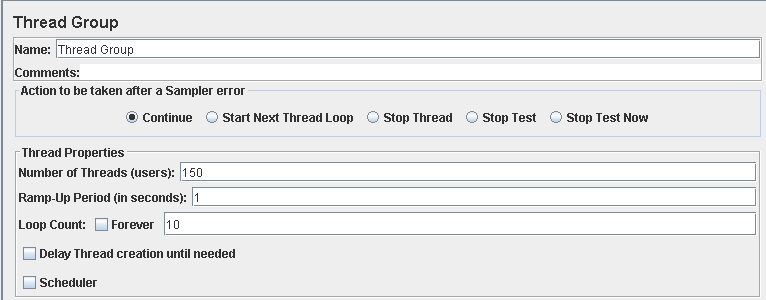
### As a not authenticated actor I want to list the positions and the companies available and navigate to their respective companies and positions



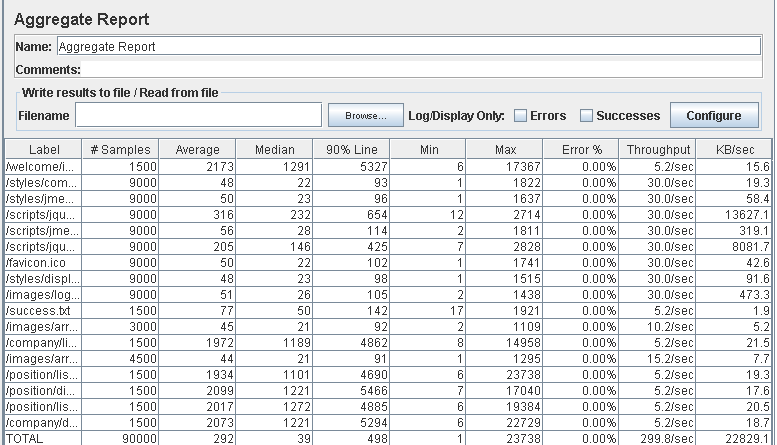
Performance test 90% results: Total 11,625s.



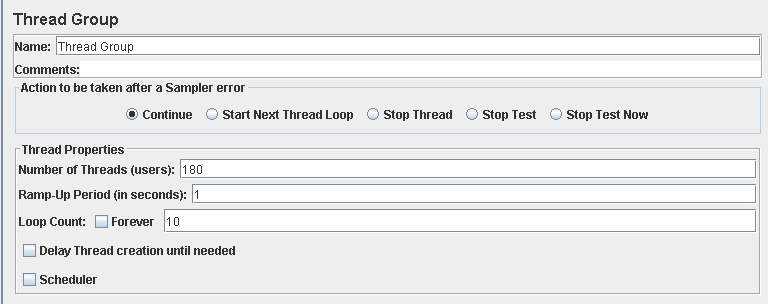
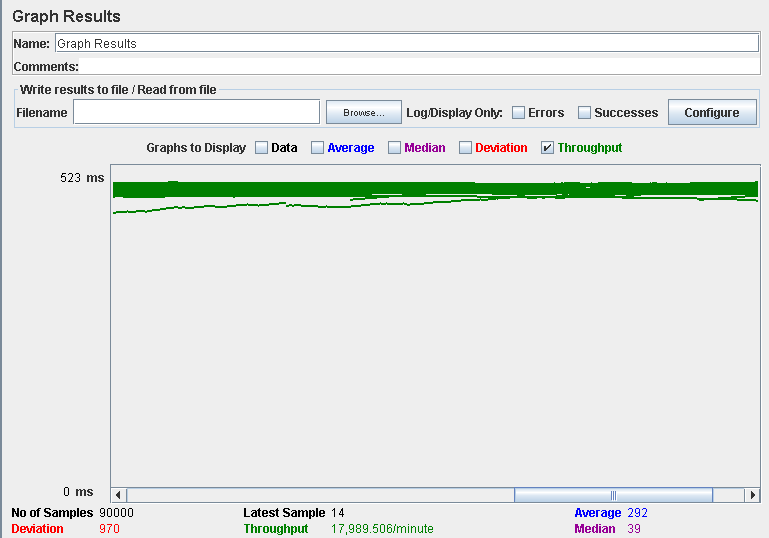
Performance thread results: 18,922 per minute.



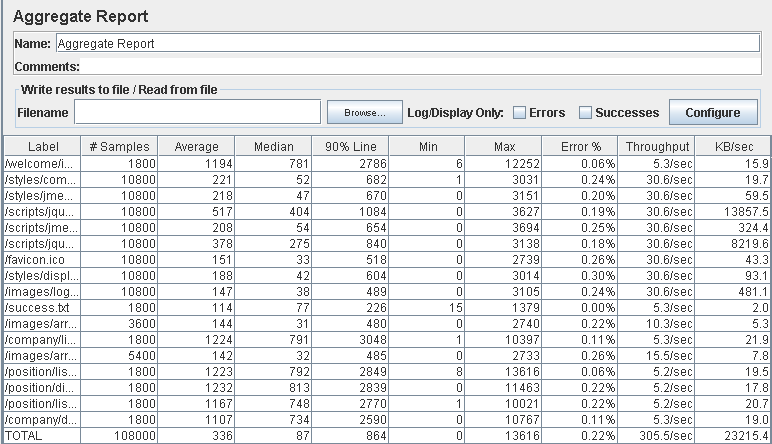
Performance test 90% results: Total 32,536s.



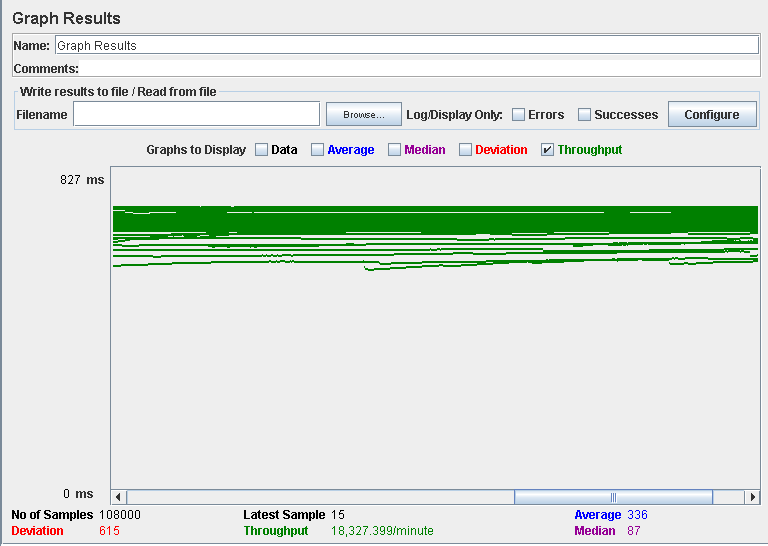
Performance thread results: 17,989 per minute.

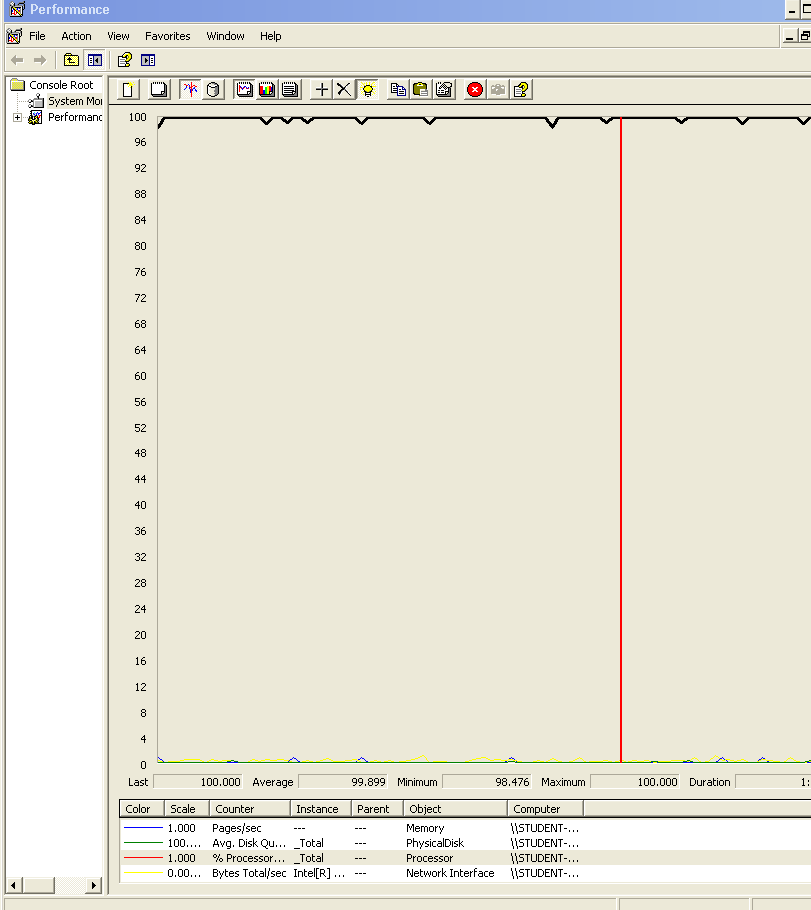


Performance test 90% results: Total 22,829s.



Performance thread results: 18,327 per minute.



Computer performance:

### Analysis results:

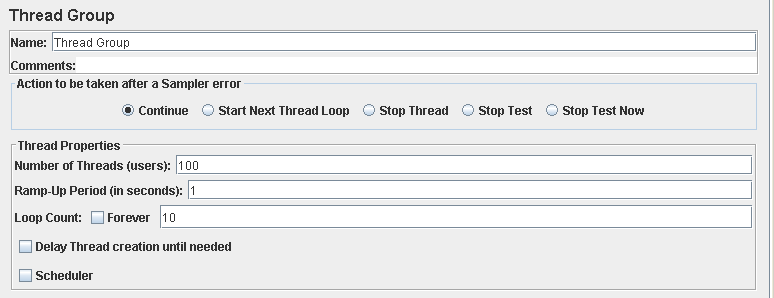
100 users and 10 loops: the application runs perfectly

150 users and 10 loops: the application runs without errors but the 90% line shows that listing the positions and companies requires a lot of time.

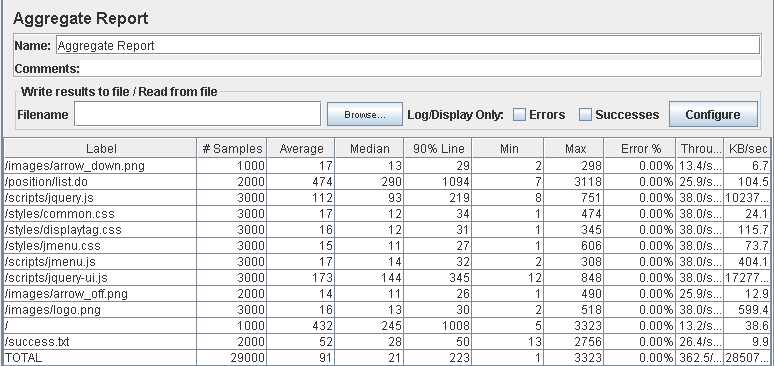
180 users and 10 loops: it begins to happen some errors, after reviewing the computer analysis results, we believe is a processor bottleneck problem.

## Use case 3

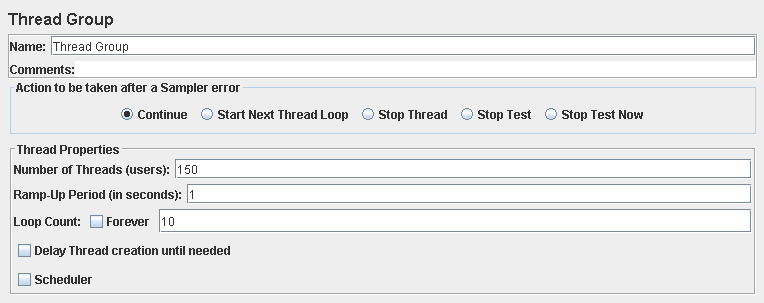
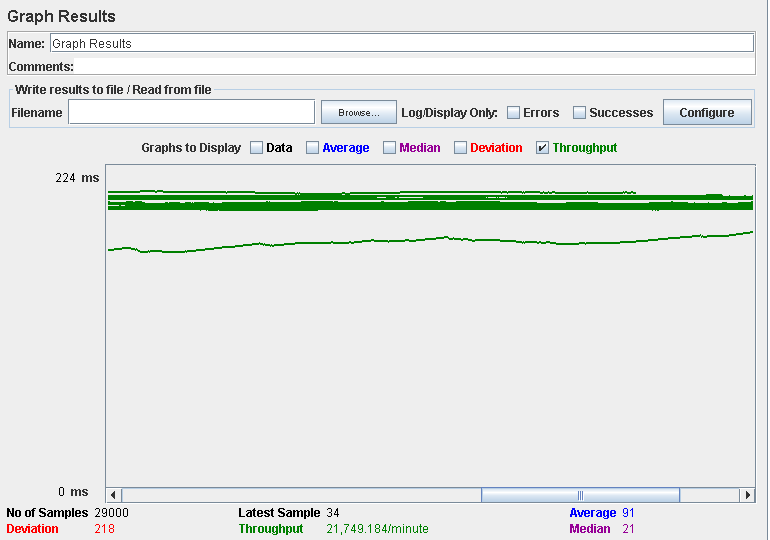
### As a not authenticated actor I want to search for a position using a single key word.



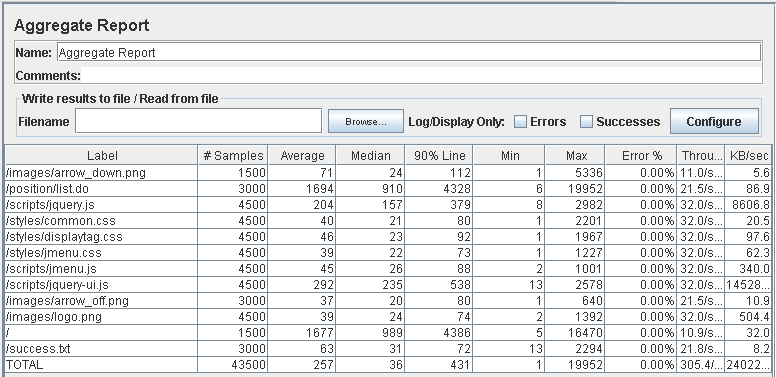
Performance test 90% results: Total 3,148s.



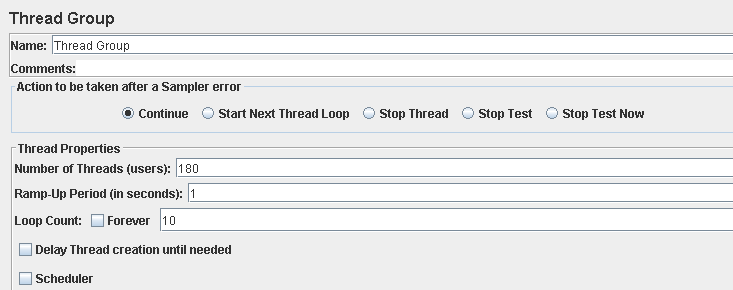
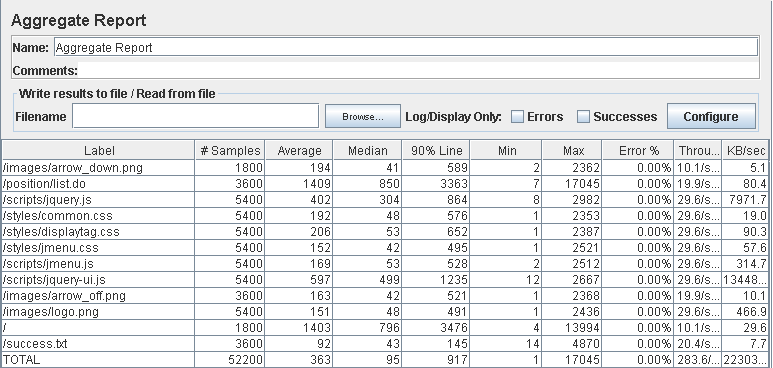
Performance thread results: 21,749 per minute.



Performance test 90% results: Total 10,302s.

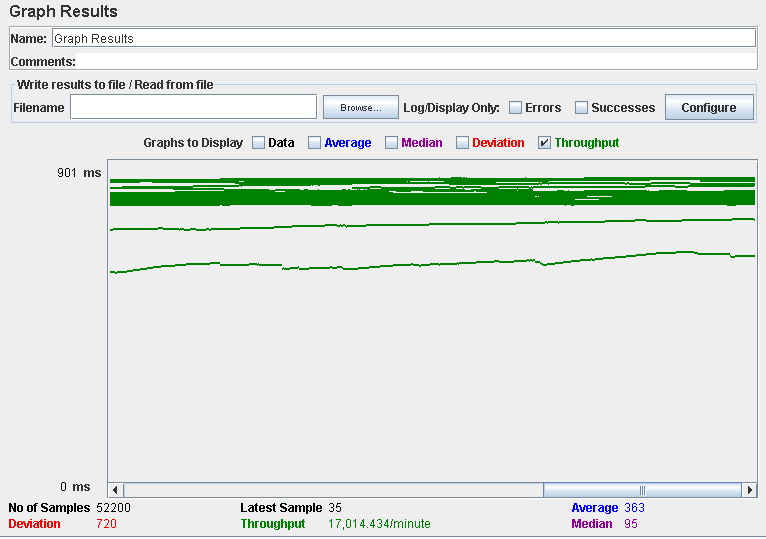
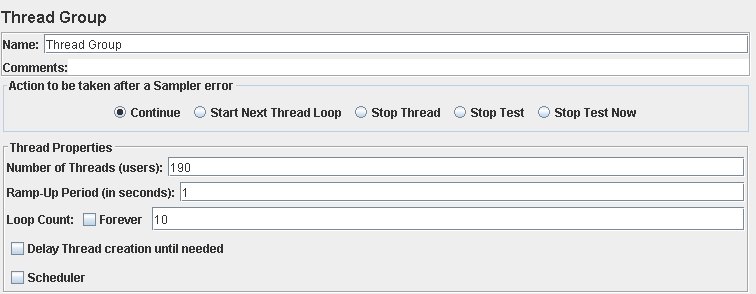


Performance thread results: 18,326 per minute.

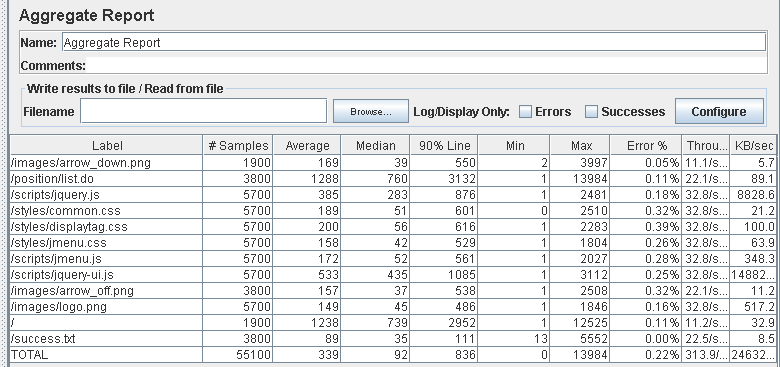


Performance test 90% results: Total 13,832

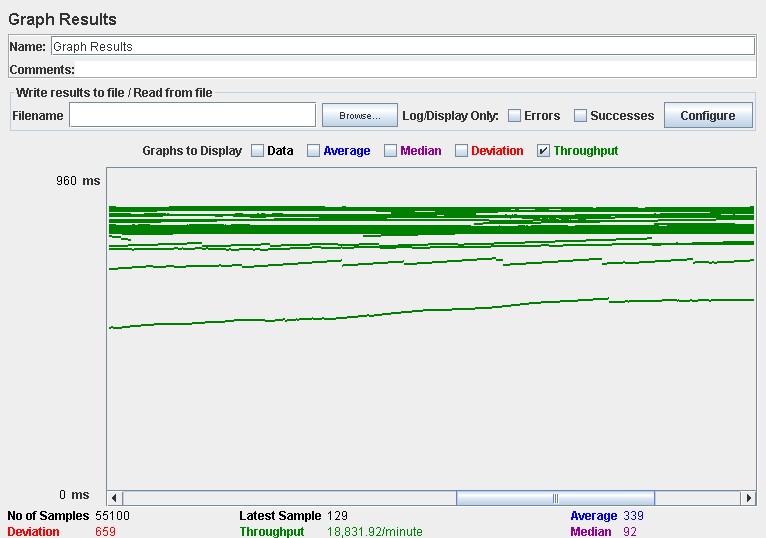
Performance thread results: 17,014 per minute



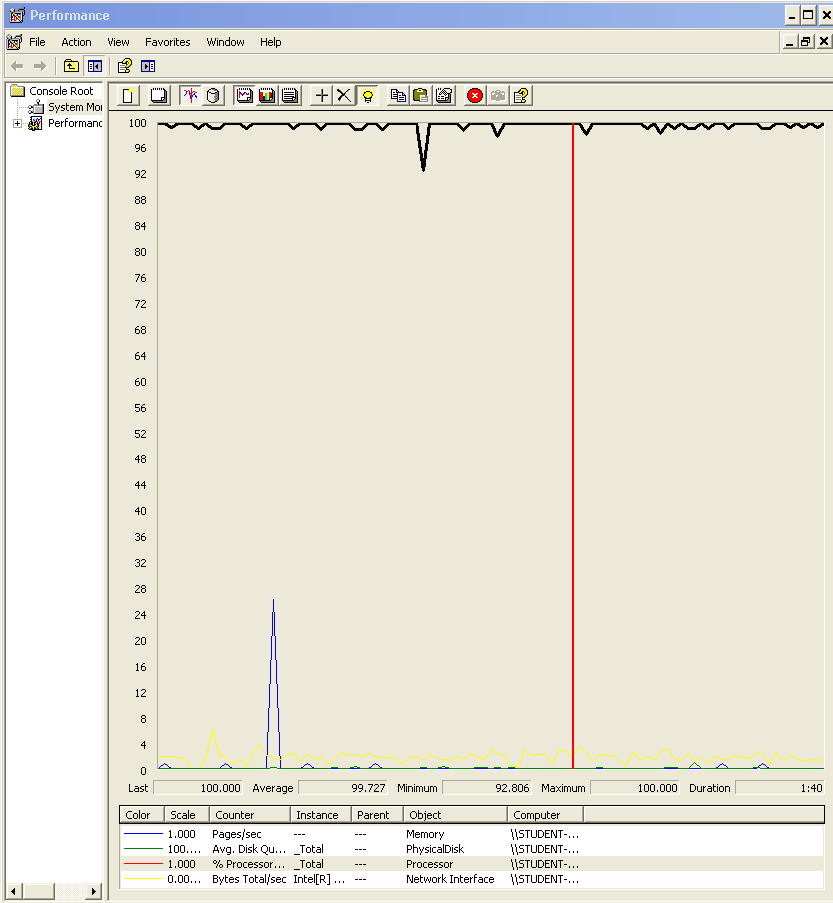
Performance test 90% results: Total 12,037



Performance thread results: 18,931 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs perfectly

150 users and 10 loops: the application runs without errors but the 90% line shows that listing the finder and going to the welcome view requires a lot of time.

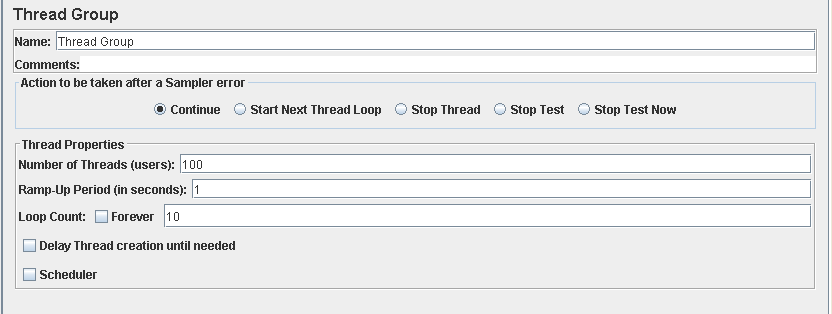
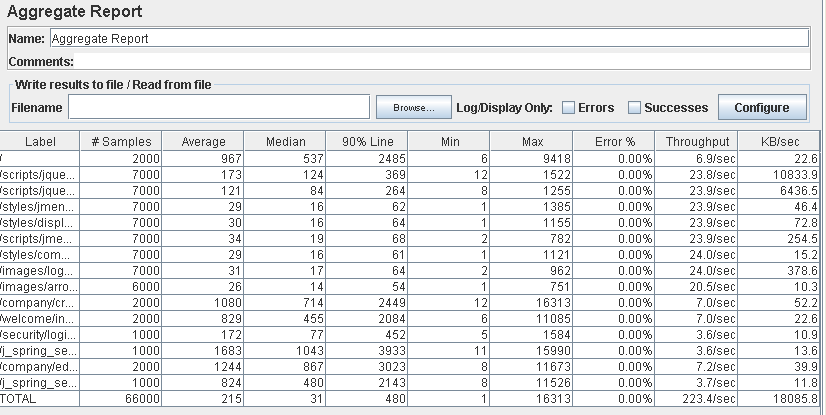
180 users and 10 loops: the application runs almost the same as with 150 users and 10 loops

190 users and 10 loops: it begins to appear some errors, we captured the computer performance for that test and we think that it was a bottleneck problem with the processor.

## Use case 4 & 5:

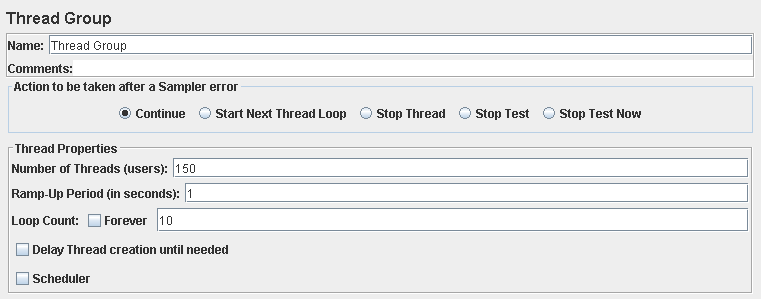
### As a not authenticated user I want to register as a Company.

### As an authenticated user I want to edit my personal data.

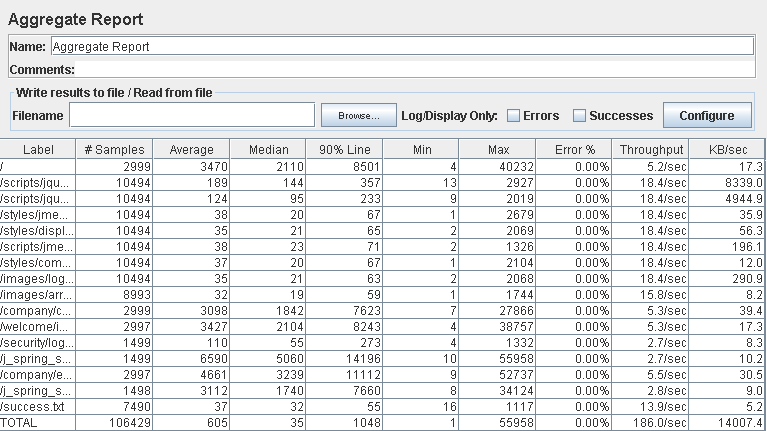


Performance test 90% results: Total 17,575s.

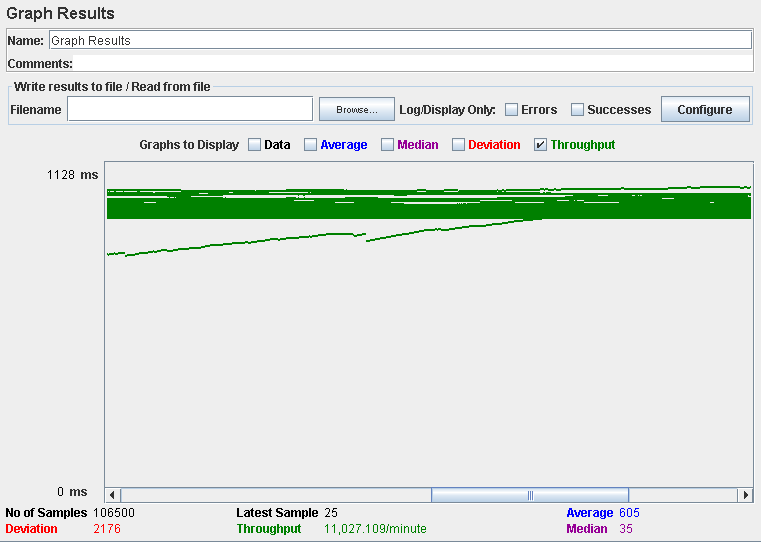
Performance thread results: 13,404 per minute.

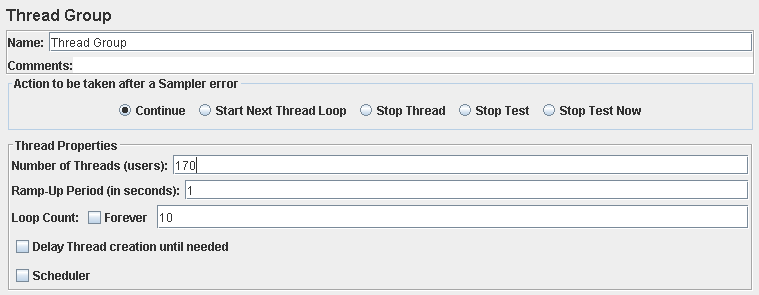


Performance test 90% results: Total 58,645s.

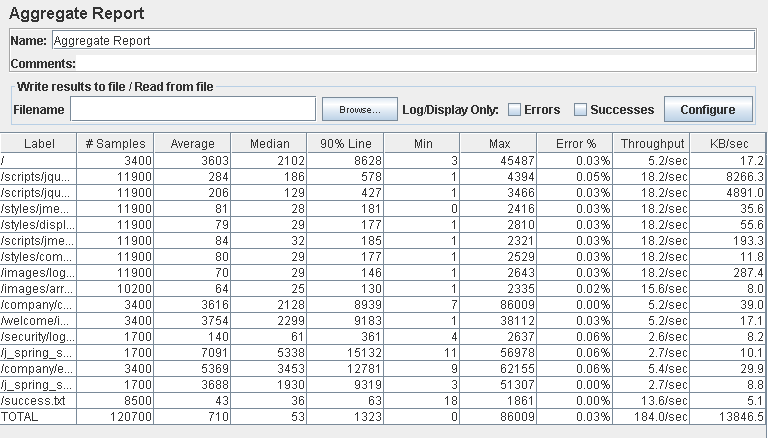


Performance thread results: 11,027 per minute.





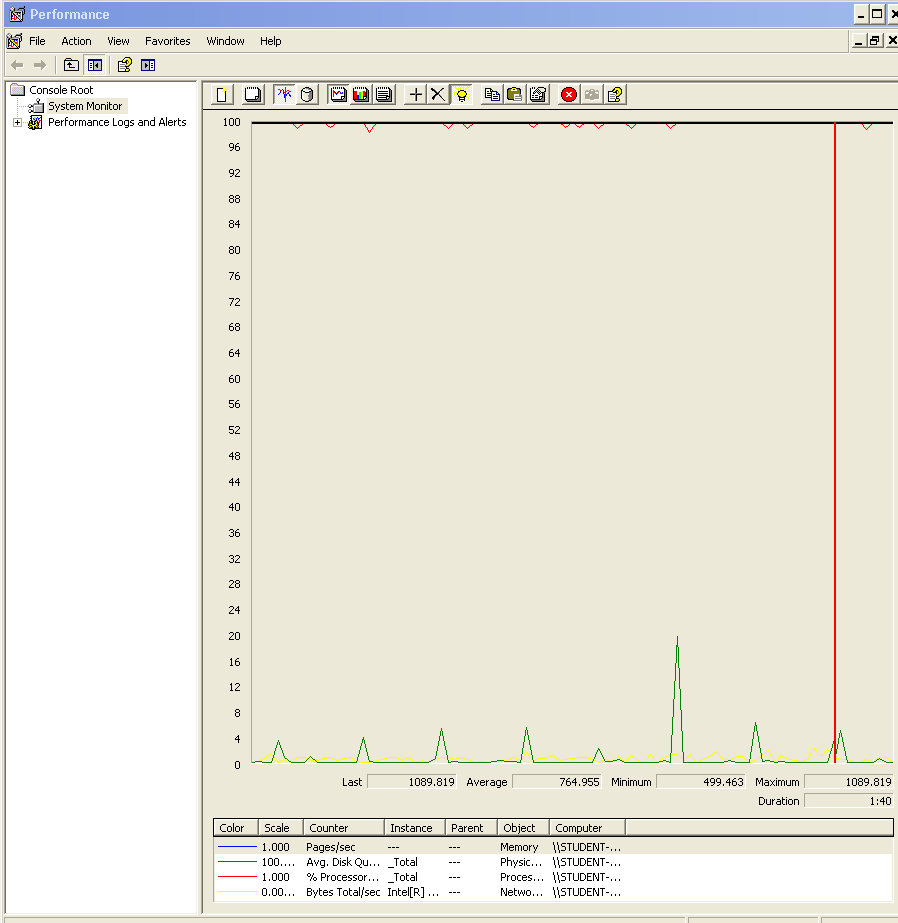
Performance test 90% results: Total 67,730s



Performance thread results: 11,670 per minute.



Computer performance:



### Analysis results:

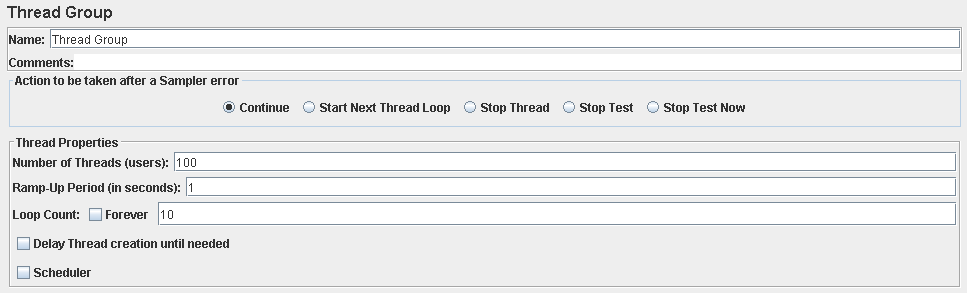
100 users and 10 loops: the application runs without errors but the edit and security check times are a bit high

150 users and 10 loops: the application runs without errors but the times are really high.

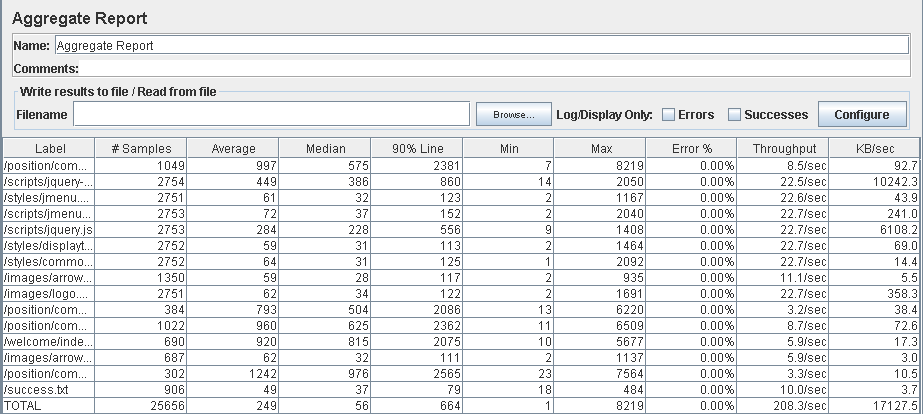
170 users and 10 loops: the application begins to have problems, the computer performance shows complications, we think is a memory bottleneck problem.

## Use case 6:

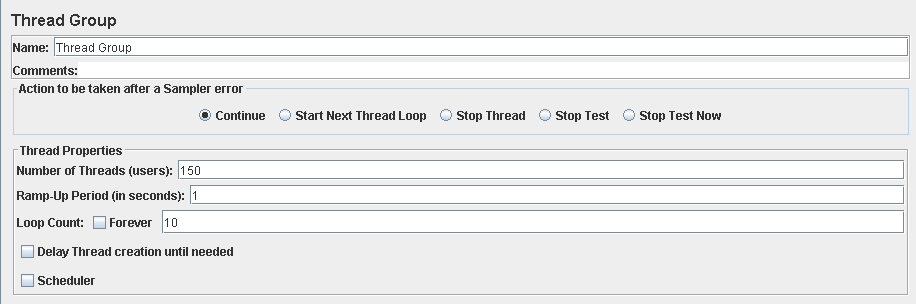
### As a company I want to manage my positions, which means listing, showing, creating, updating and deleting them.



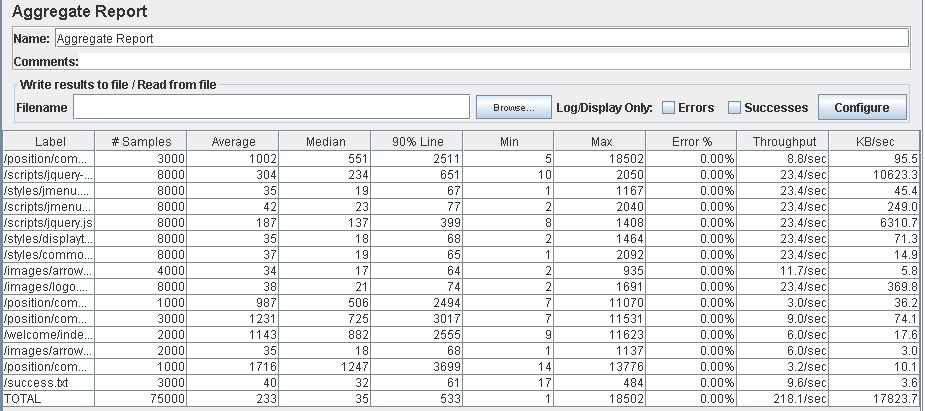
Performance test 90% results: Total 13,727s.



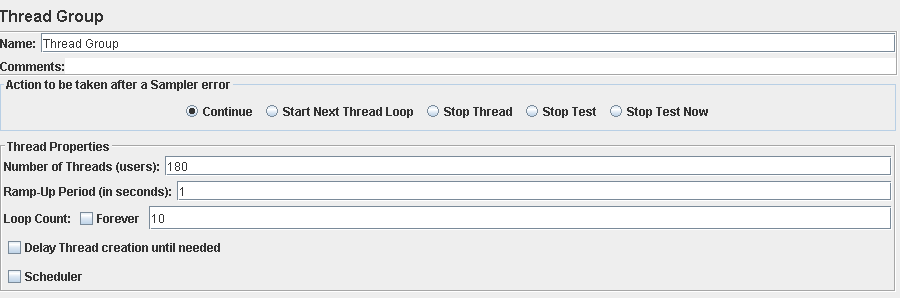
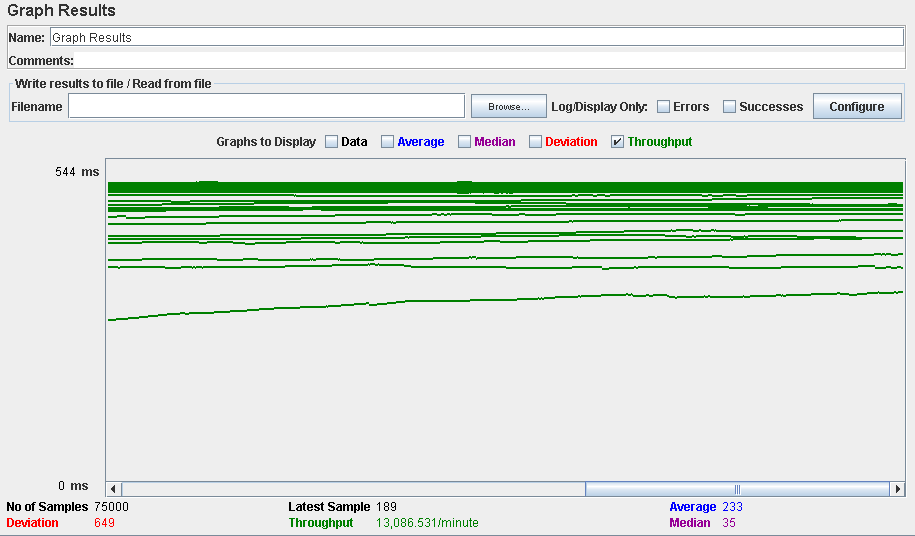
Performance thread results: 12,836 per minute.



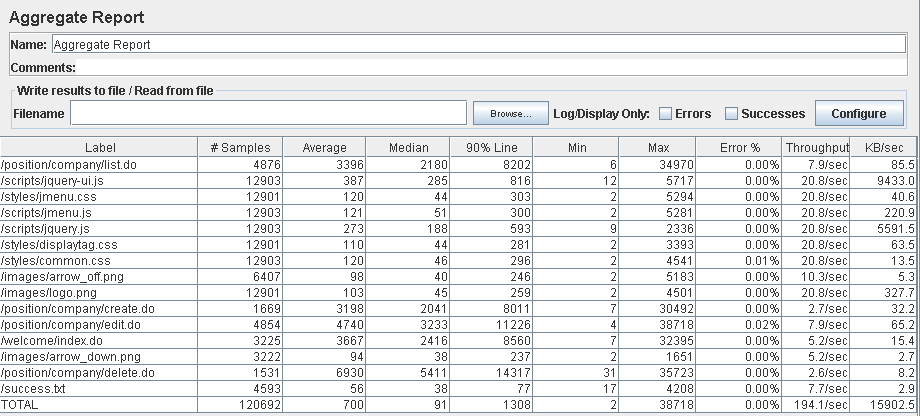
Performance test 90% results: Total 13,570s.



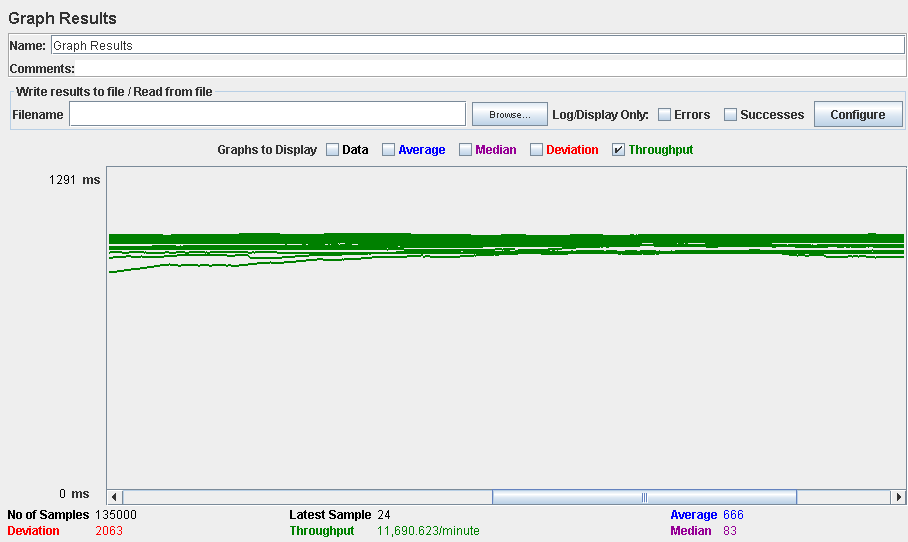
Performance thread results: 13,088 per minute.



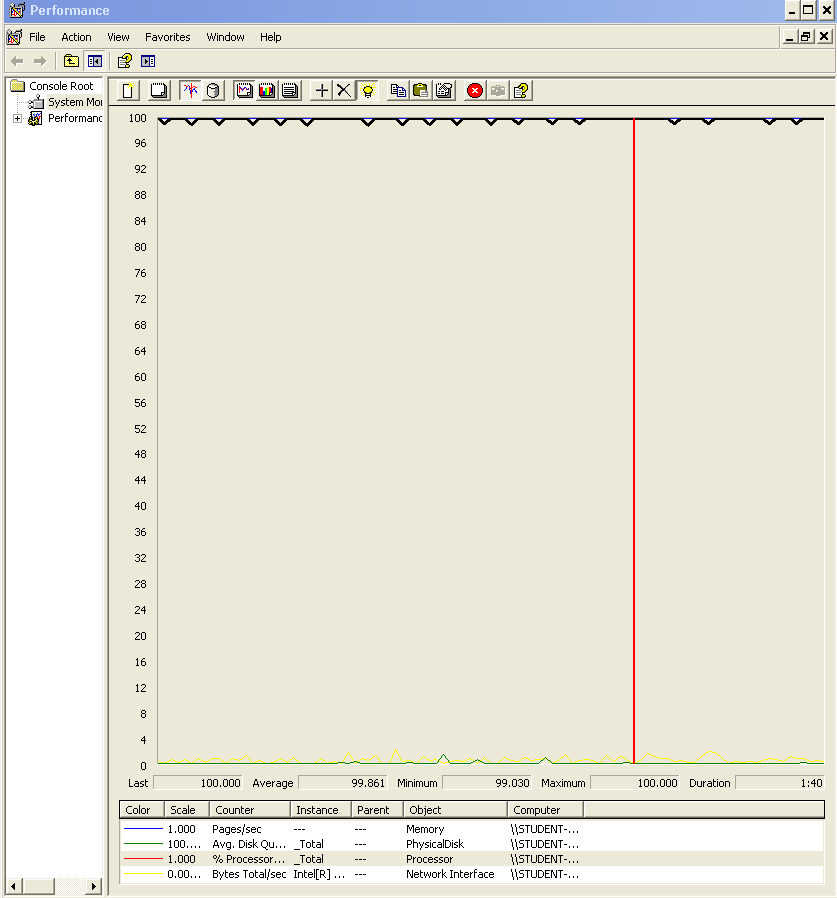
Performance test 90% results: Total 53,814s.



Performance thread results: 11,690 per minute.



Computer performance:



### Analysis results:

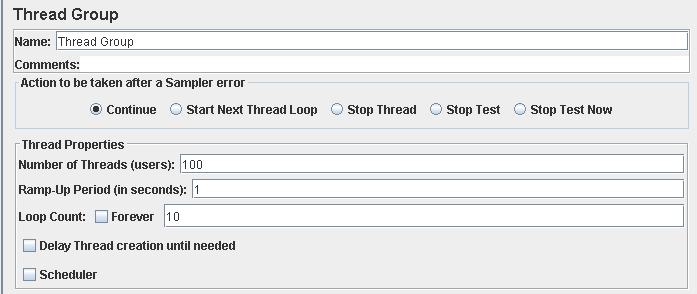
100 users and 10 loops: the application runs perfectly

150 users and 10 loops: the application runs without errors but the times are a bit high (3600ms).

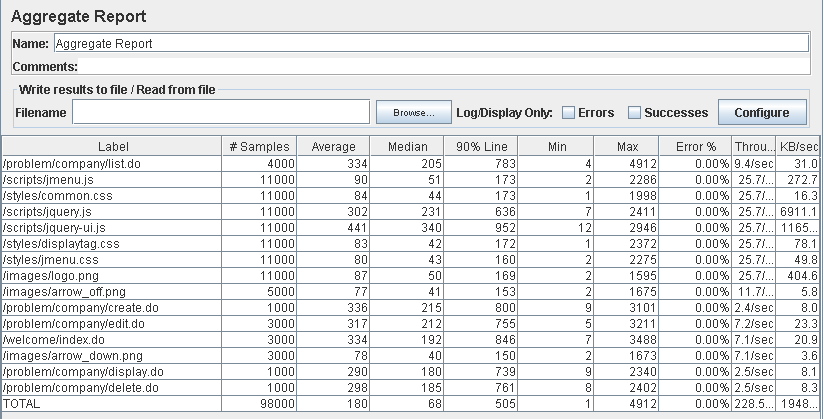
180 users and 10 loops: the application begins to have errors, we believe is a memory or a processors bottleneck problem.

## Use case 7:

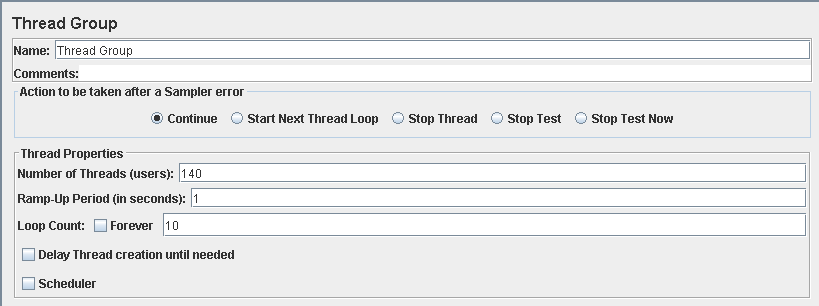
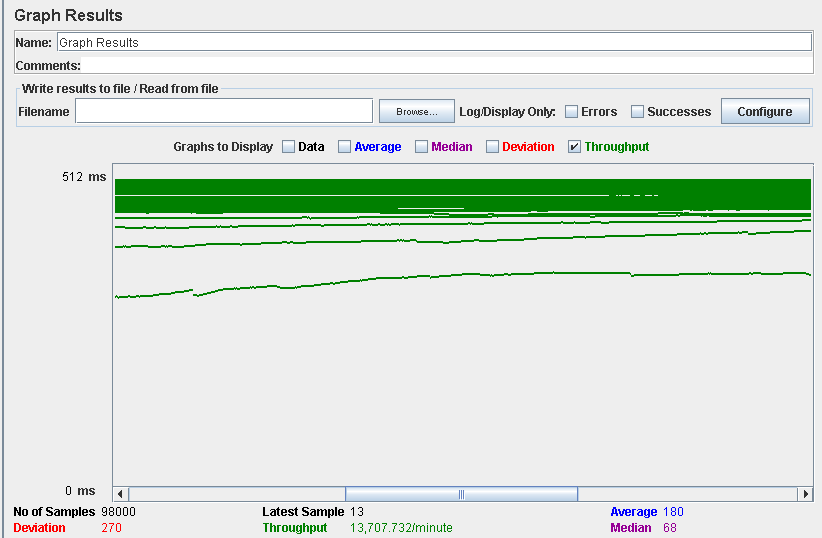
### As a company I want to manage my problems, which means listing, showing, creating, updating and deleting them.



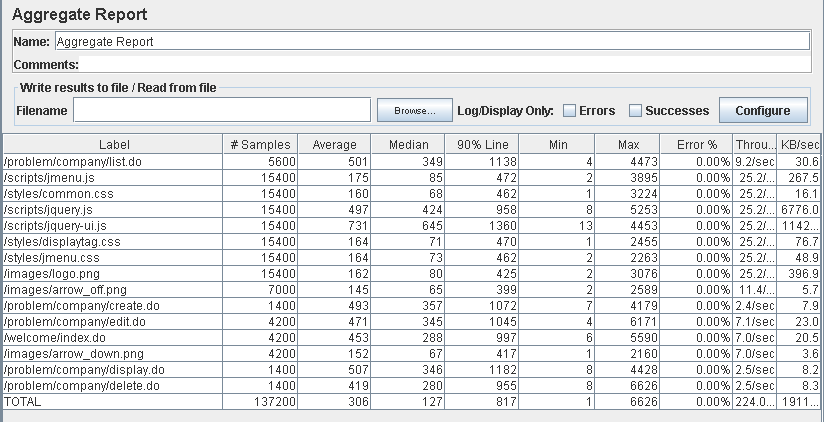
Performance test 90% results: Total 7,422s.



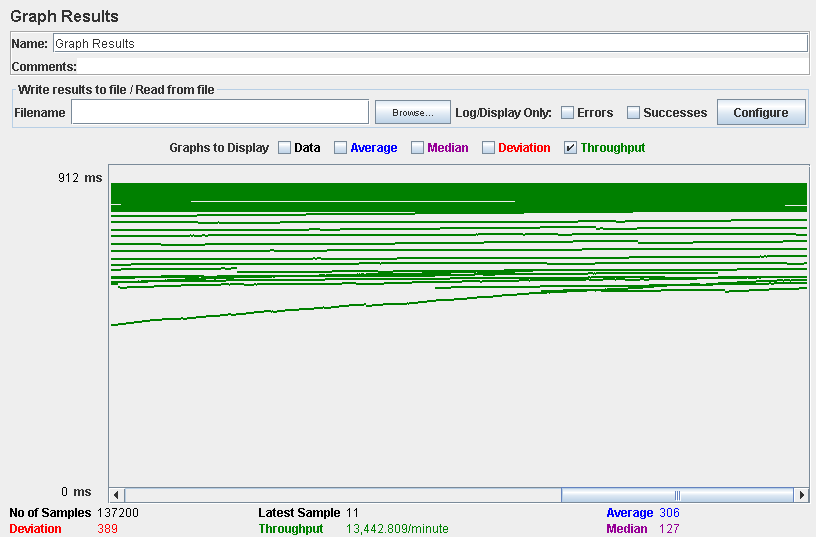
Performance thread results: 13,707 per minute.

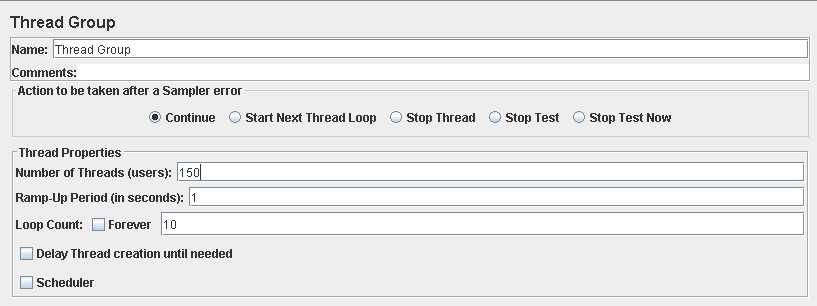


Performance test 90% results: Total 11,814s.

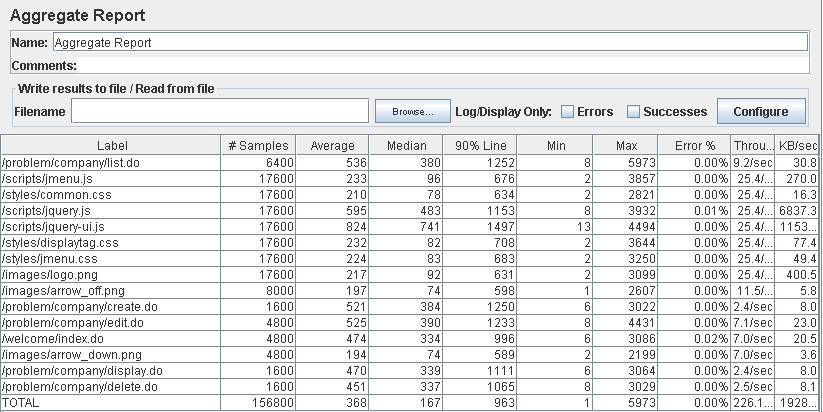


Performance thread results: 13,442 per minute.





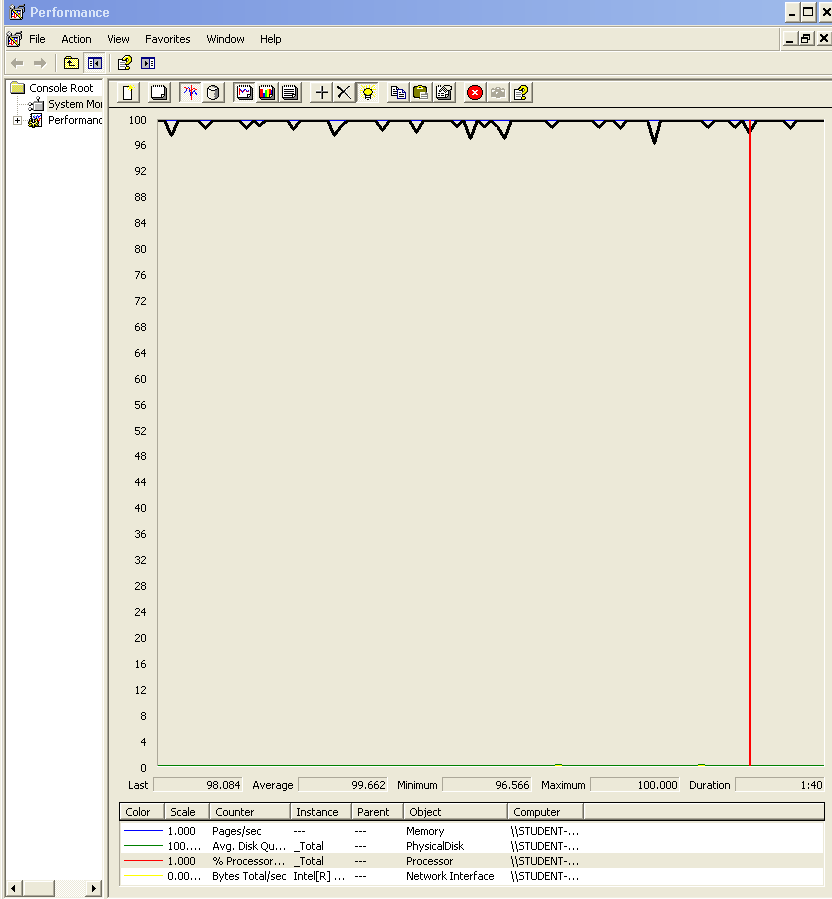
Performance test 90% results: Total 14,076s.



Performance thread results: 13,565 per minute.



Computer performance:



### Analysis results:

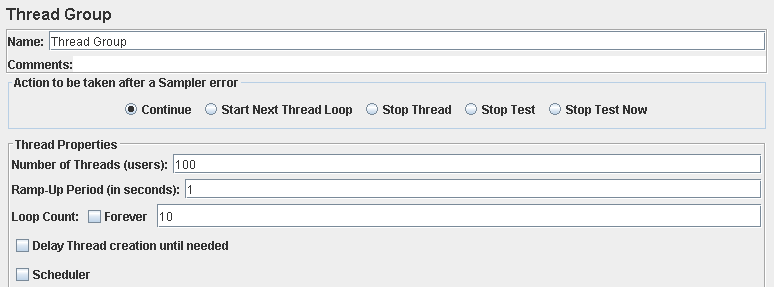
100 users and 10 loops: the application runs perfectly

140 users and 10 loops: the application runs perfectly too.

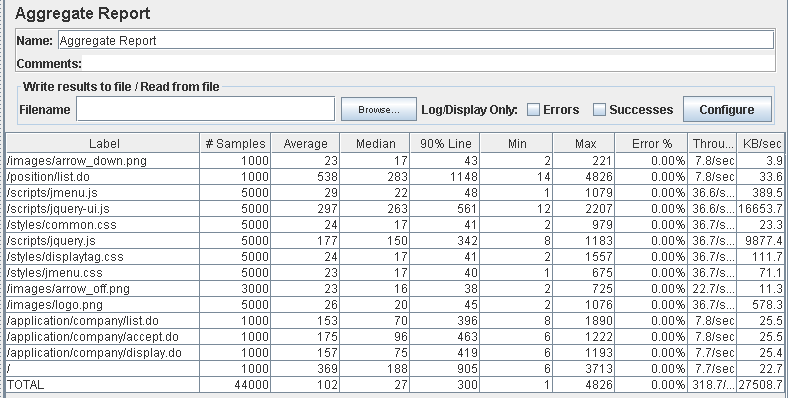
150 users and 10 loops: the application begins to have errors, the computer performance shows a memory bottleneck problem.

## Use case 8:

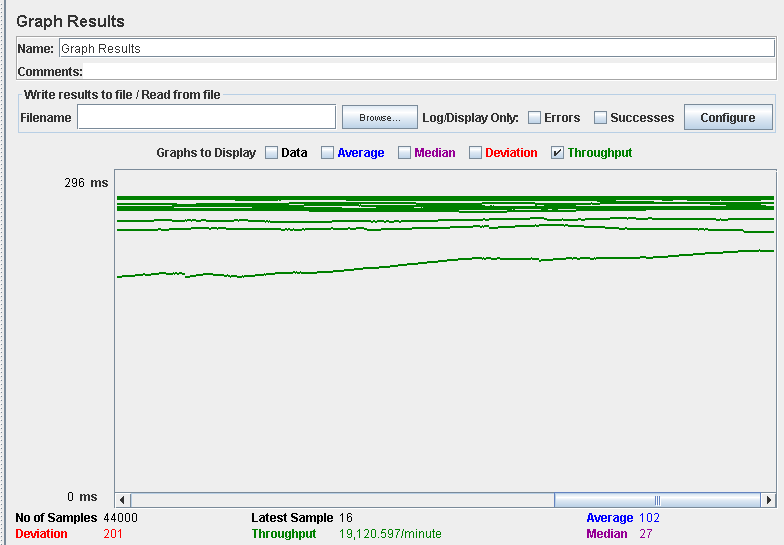
### As a company I want to manage the applications to my positions, which means listing them grouped by status, showing and updating them.

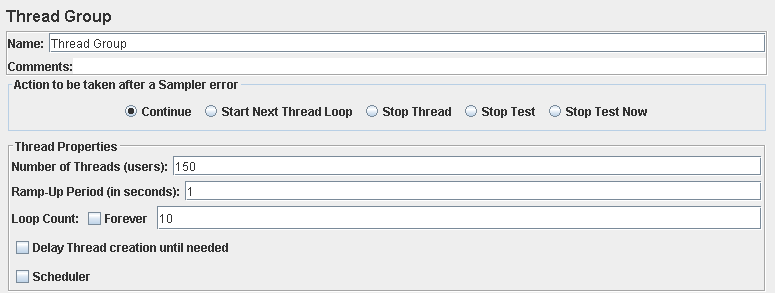


Performance test 90% results: Total 4,067s.

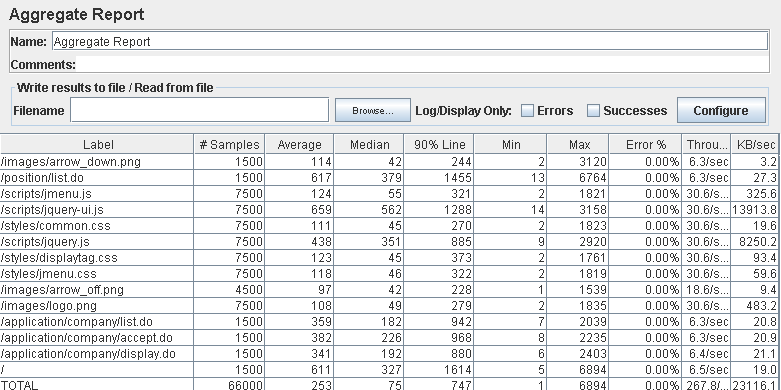


Performance thread results: 19,120 per minute

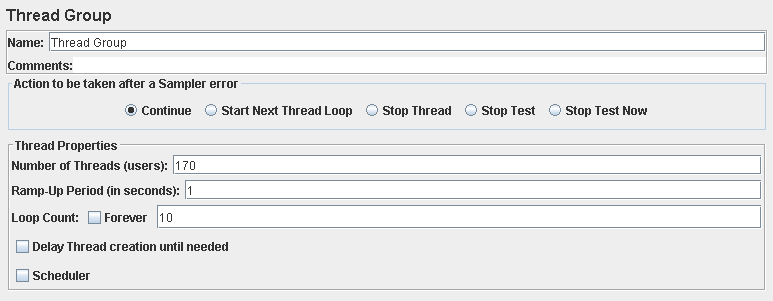
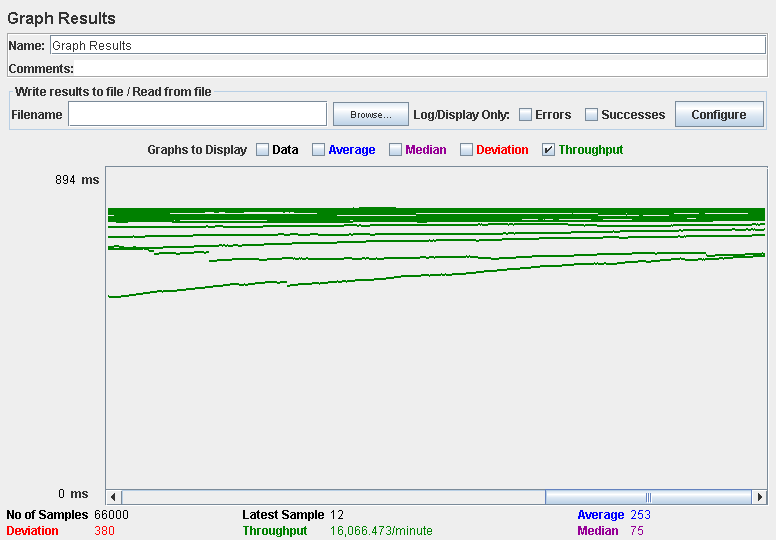




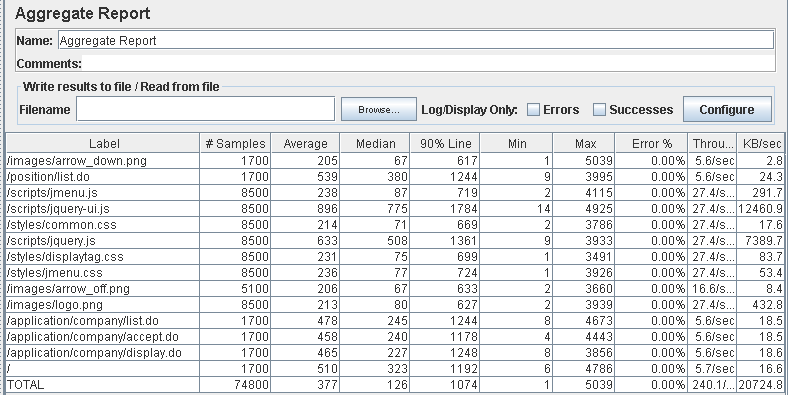
Performance test 90% results: Total 8,164s.



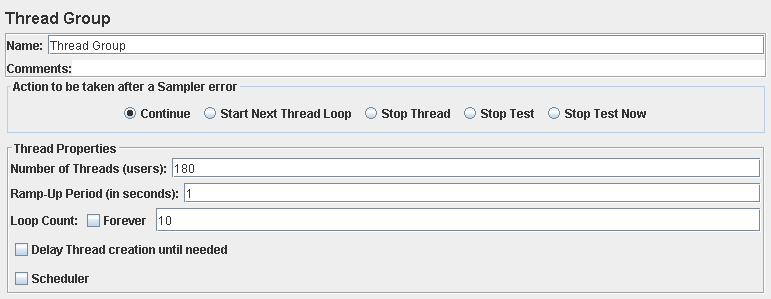
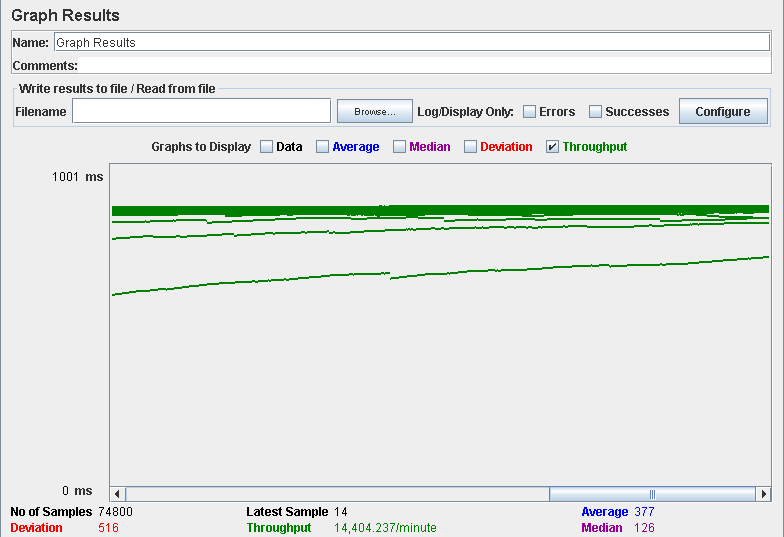
Performance thread results: 16,066 per minute.



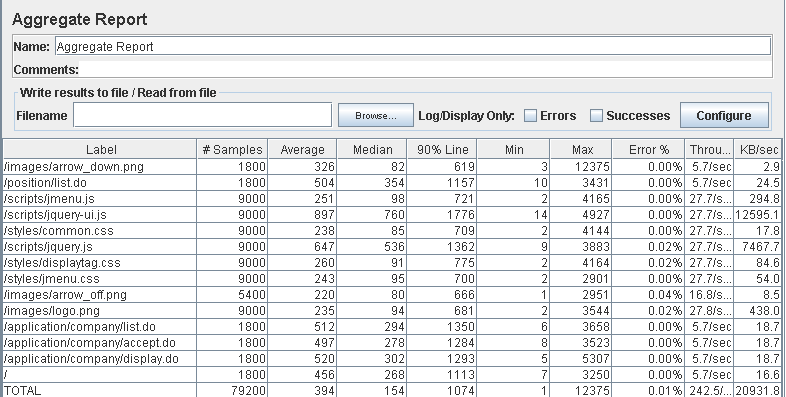
Performance test 90% results: Total 13,850s.



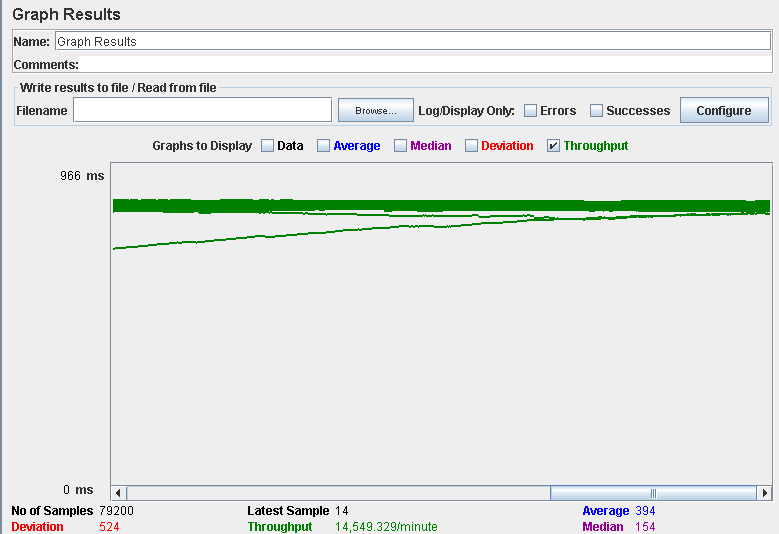
Performance thread results: 14,404 per minute.



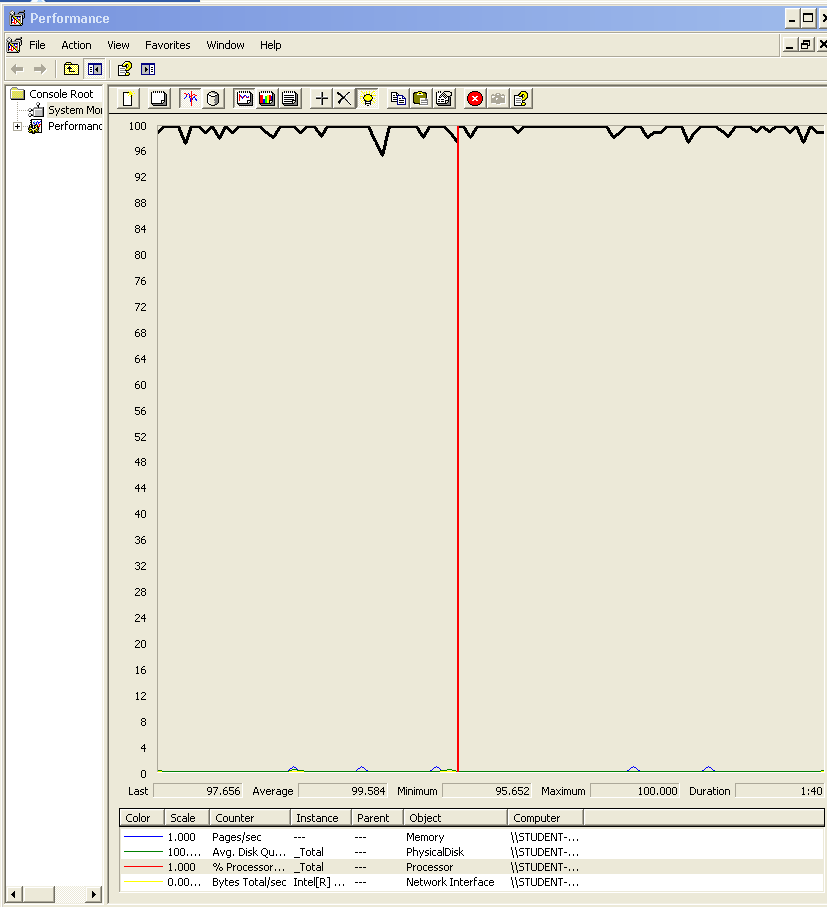
Performance test 90% results: Total 14,206s.



Performance thread results: 14,549 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs perfectly

150 users and 10 loops: the application runs perfectly too.

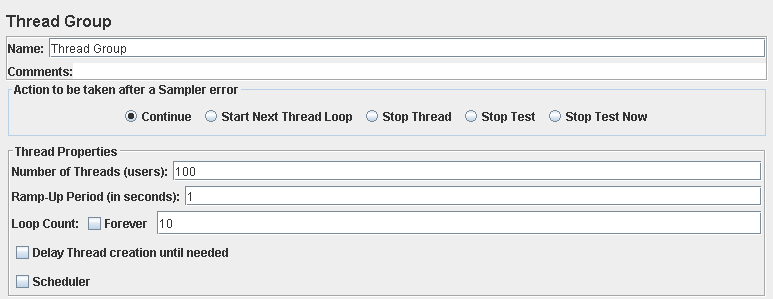
170 users and 10 loops: the application runs perfectly too.

180 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

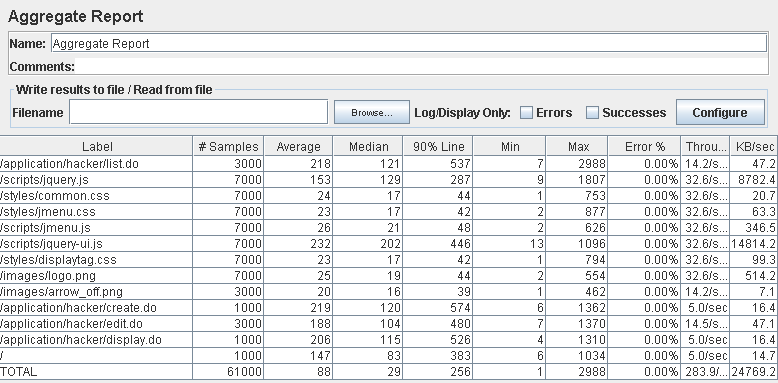
## Use case 9 & 10:

### As a rookie I want to manage my applications, which includes listing them grouped by status, showing them, creating them, and updating them.

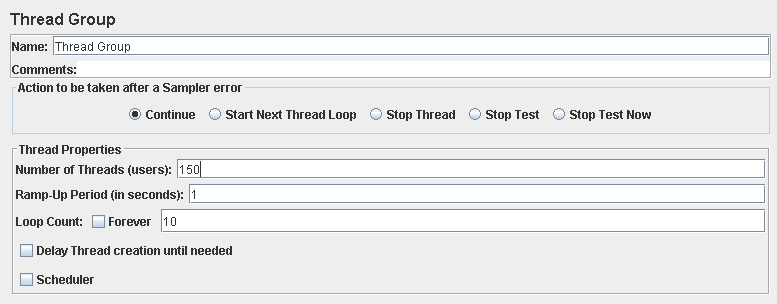
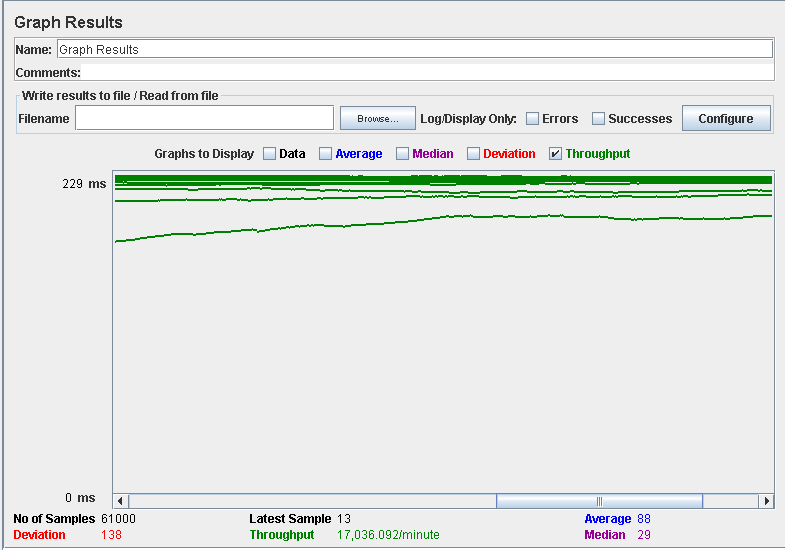
### As a rookie I want to be assigned an arbitrary problem when I create an application.



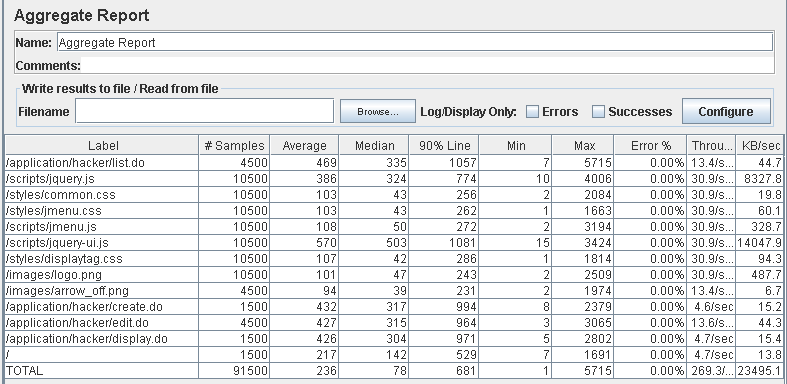
Performance test 90% results: Total 3,531s.



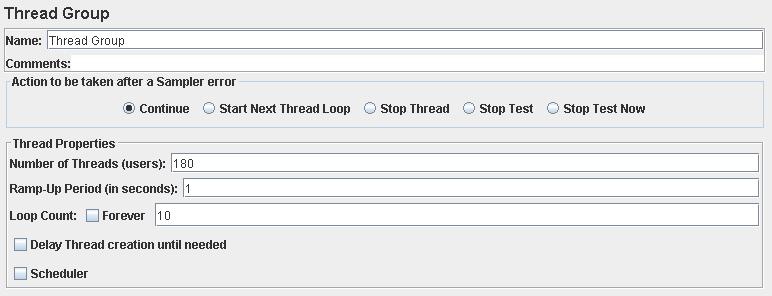
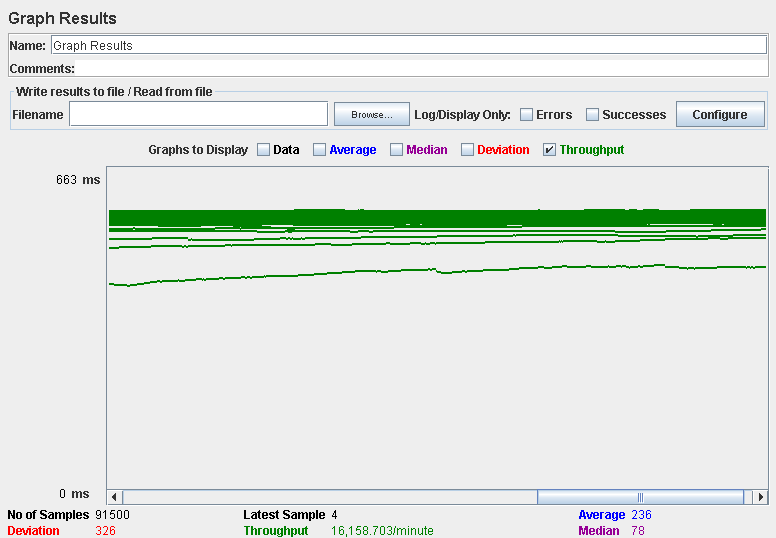
Performance thread results: 17,036 per minute.



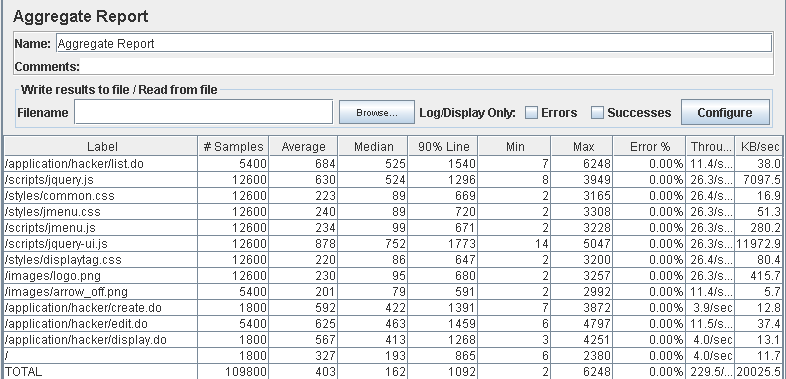
Performance test 90% results: Total 5,745s.



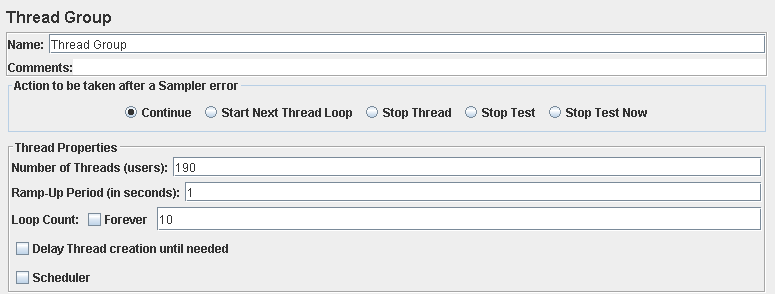
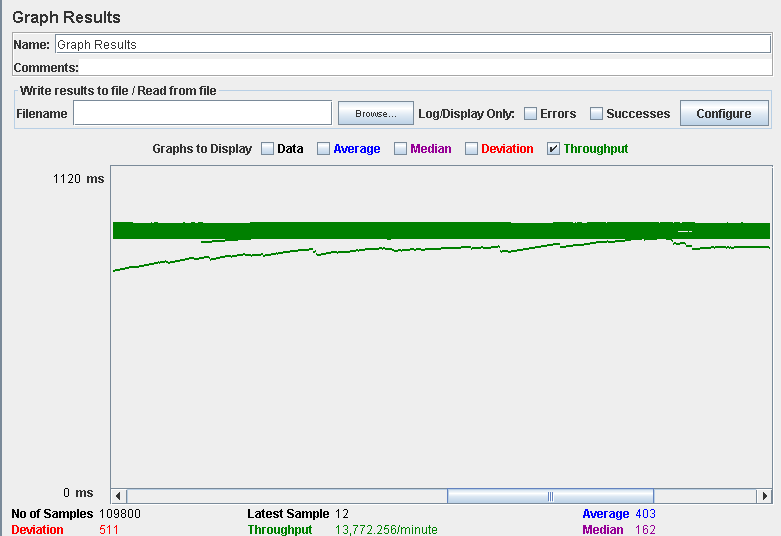
Performance thread results: 16,158 per minute.



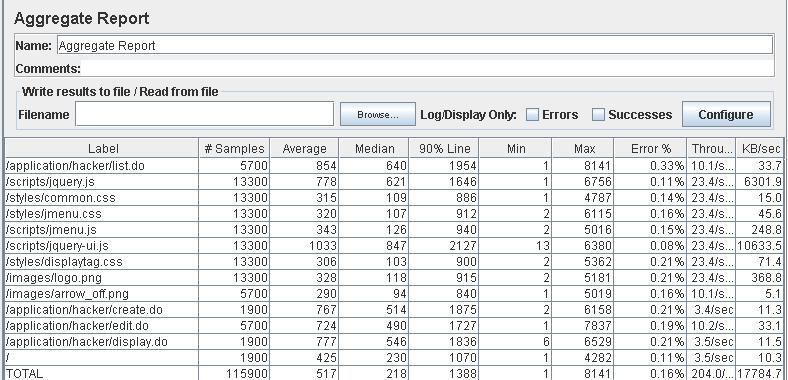
Performance test 90% results: Total 10,085s.



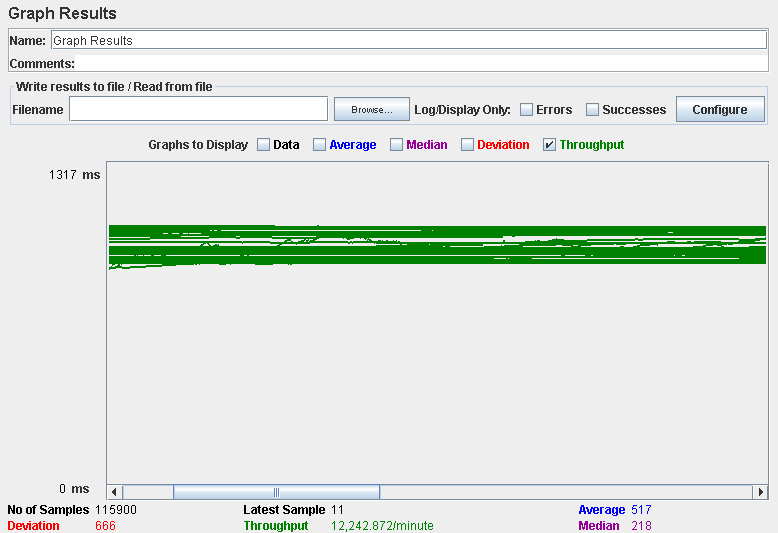
Performance thread results: 13,772 per minute.



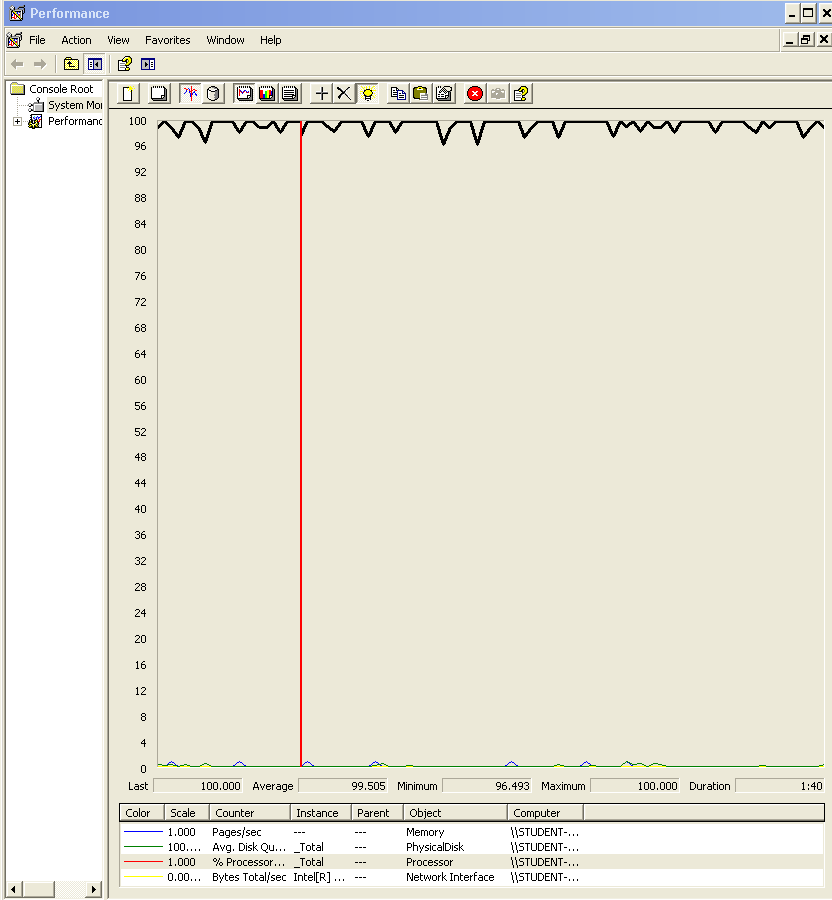
Performance test 90% results: Total 16,452s.



Performance thread results: 12,242 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application runs perfectly.

180 users and 10 loops: the application runs perfectly.

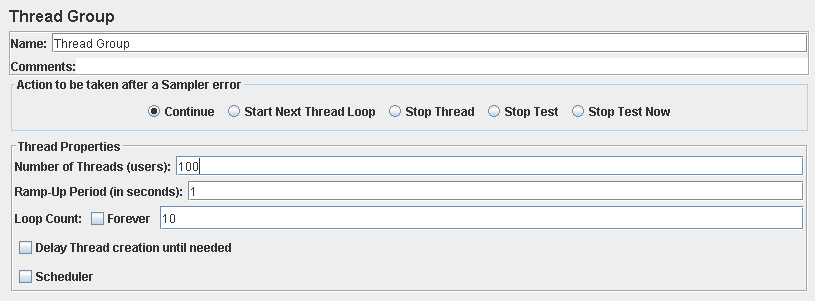
190 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

## Use case 11 & 12:

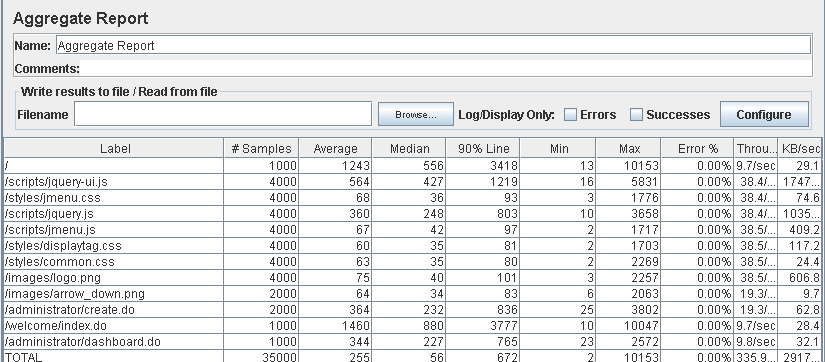
### As an admin I want to create user accounts for new administrators.

### As an admin I want to display a dashboard with the following information:

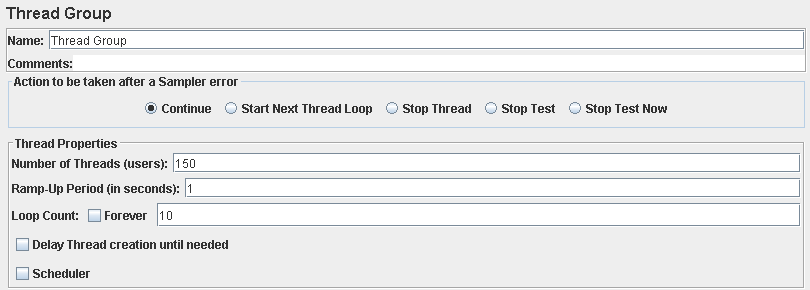
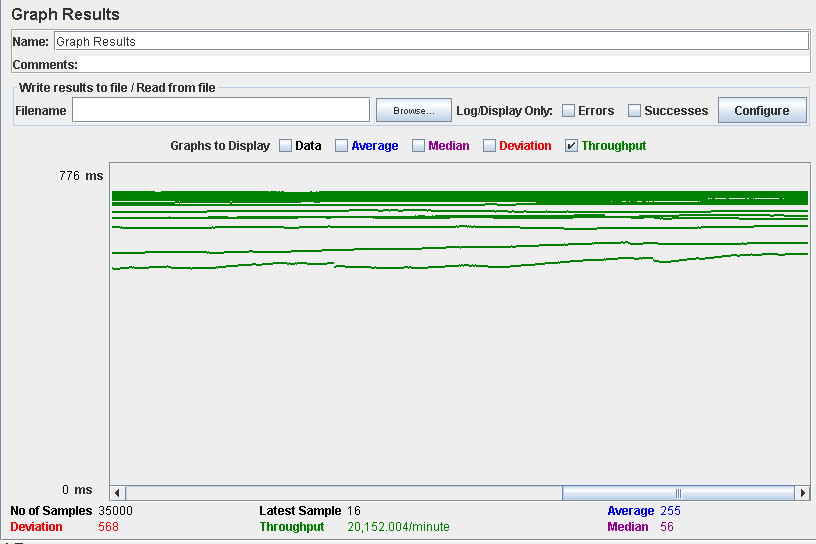
* The average, the minimum, the maximum, and the standard deviation of the number of positions per company.
* The average, the minimum, the maximum, and the standard deviation of the number of applications per rookie.
* The companies that have offered more positions.
* The rookies who have made more applications.
* The average, the minimum, the maximum, and the standard deviation of the salaries offered.
* The best and the worst position in terms of salary.
* The minimum, the maximum, the average, and the standard deviation of the number of curricula per rookie.
* The minimum, the maximum, the average, and the standard deviation of the number of results in the finders.
* The ratio of empty versus non-empty finders.



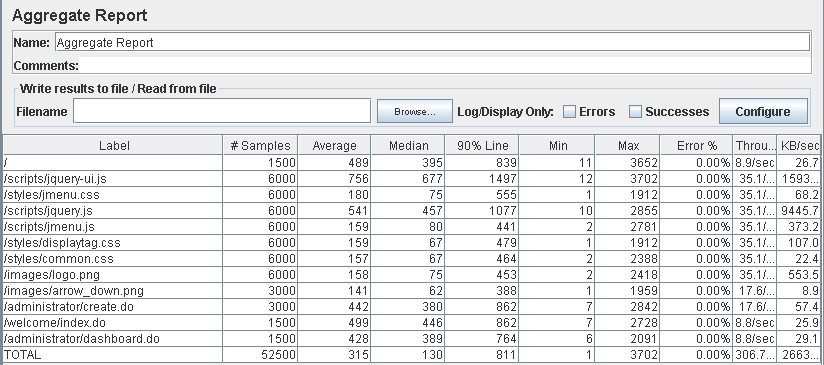
Performance test 90% results: Total 11,393s.



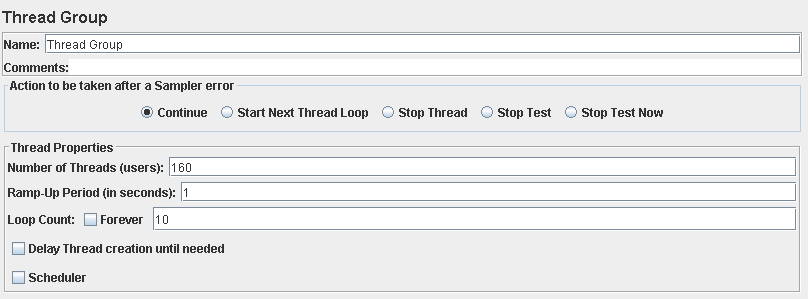
Performance thread results: 20,152 per minute.



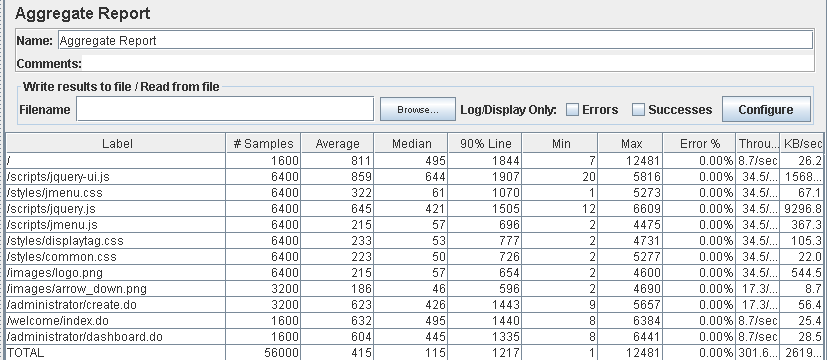
Performance test 90% results: Total 7,902s.



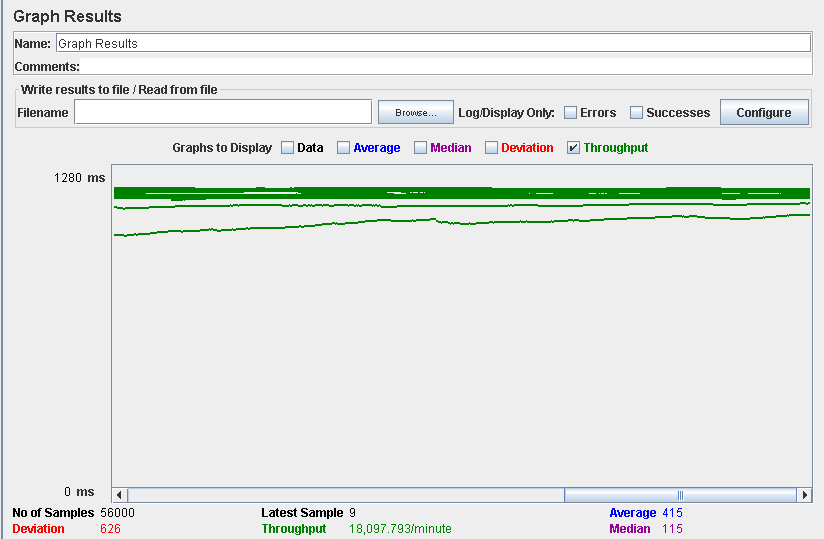
Performance thread results: 18,283 per minute.

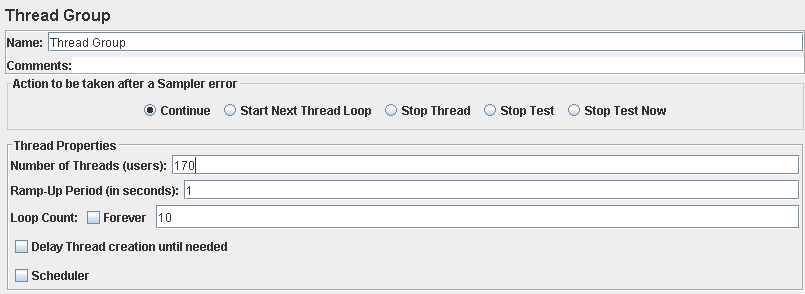


Performance test 90% results: Total 13,993s.

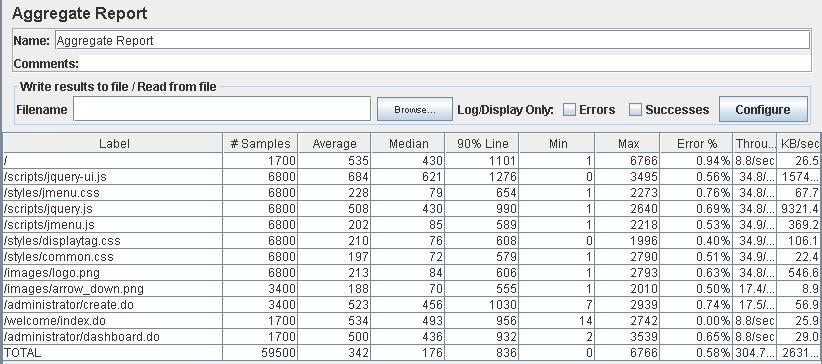


Performance thread results: 18,907 per minute.





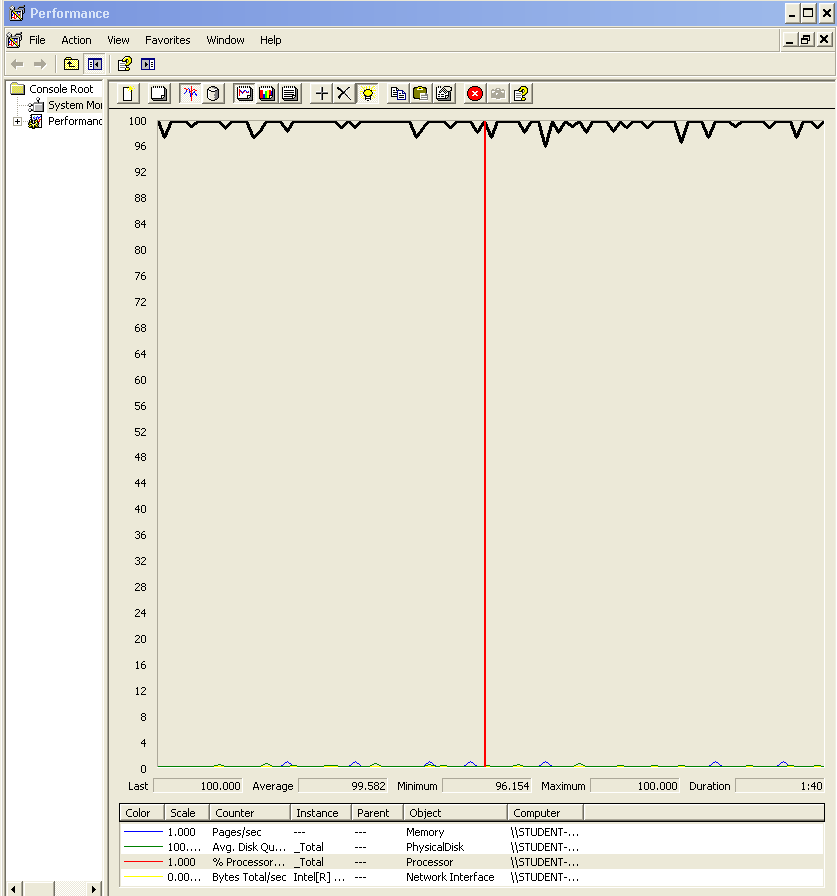
Performance test 90% results: Total 10,104s.



Performance thread results: 18,400 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs well, the times are a bit high sometimes butt is fixed somehow with 150 users, could be that the computer was doing another tasks under the system.

150 users and 10 loops: the application runs perfectly.

160 users and 10 loops: the application runs perfectly.

170 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

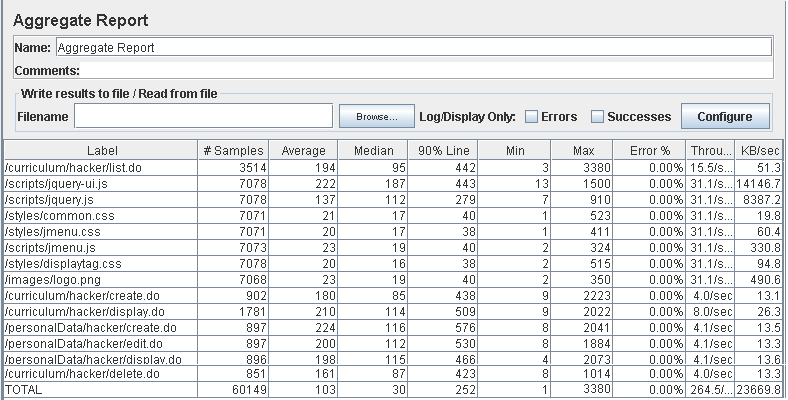
## Use case 13 & 14:

### As a rookie I want to manage my curricula, which includes listing, showing, creating, updating, and deleting them.

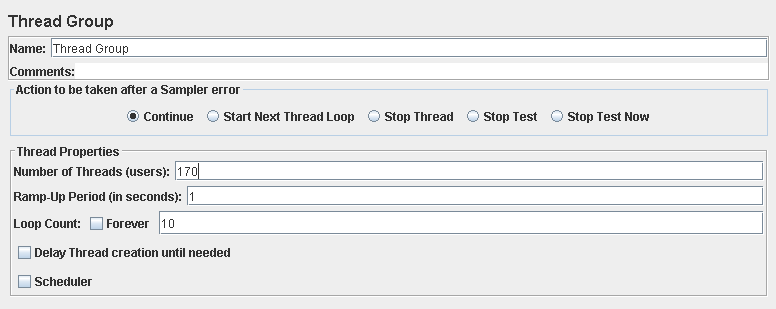
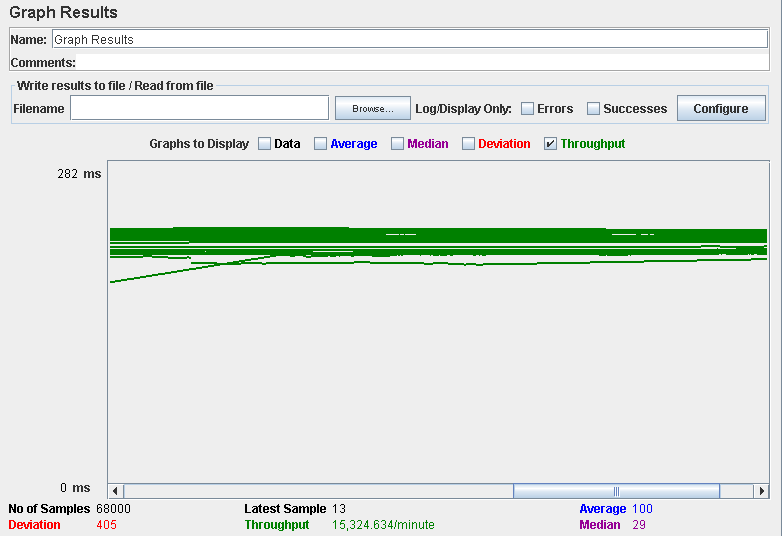
### As a rookie I want to attach a curriculum to the application that is a copy of the original.



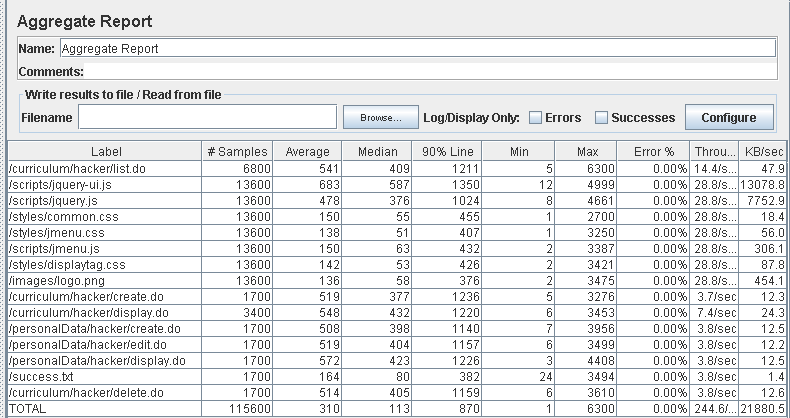
Performance test 90% results: Total 4,302s.



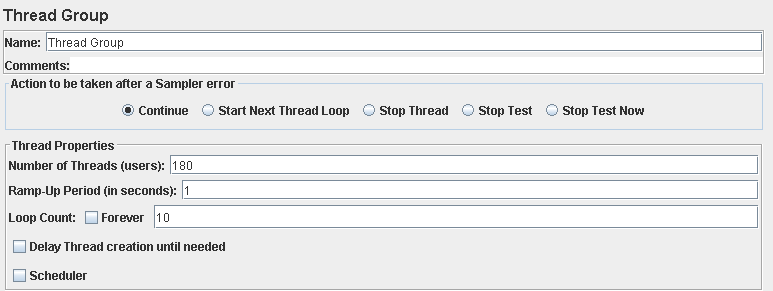
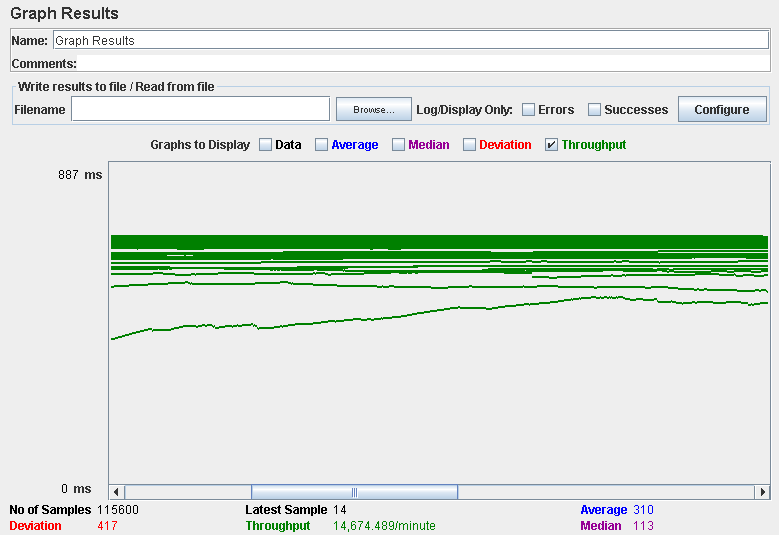
Performance thread results: 15,324 per minute.



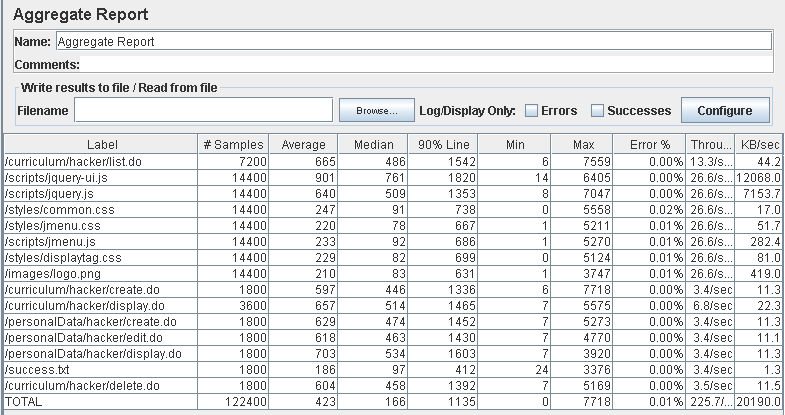
Performance test 90% results: Total 14,270s.



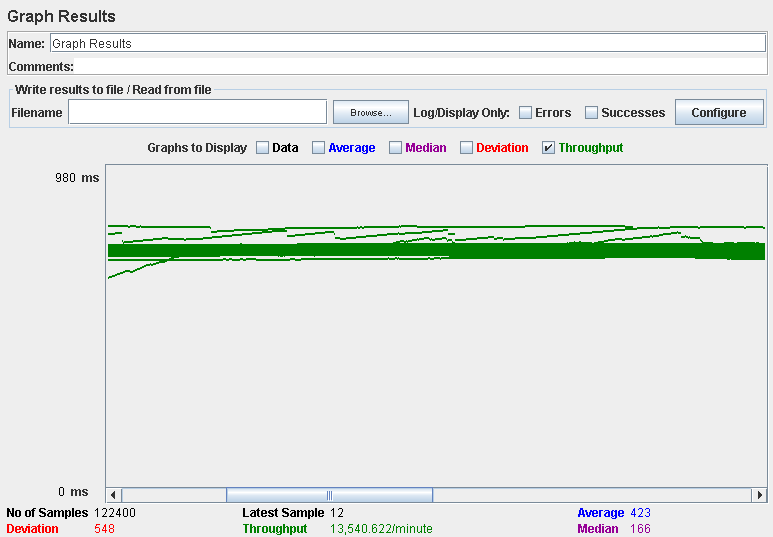
Performance thread results: 14,674 per minute.



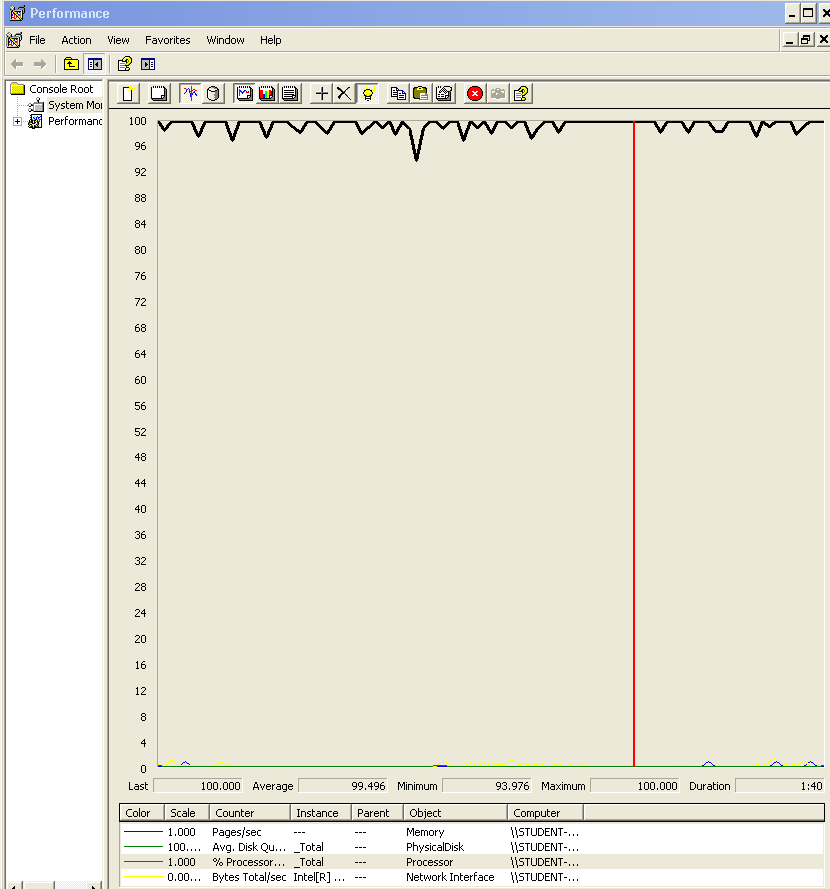
Performance test 90% results: Total 17,227s.



Performance thread results: 13,540 per minute.



Computer performance:



### Analysis results:

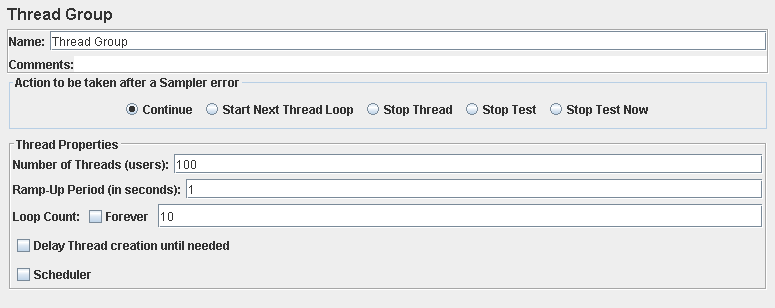
100 users and 10 loops: the application runs perfectly.

170 users and 10 loops: the application runs perfectly.

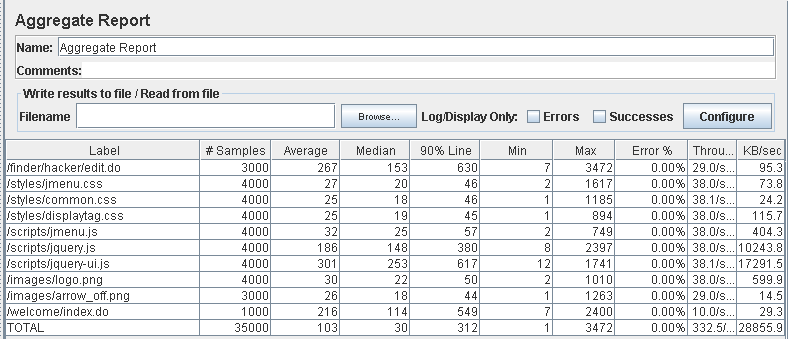
180 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

## Use case 15:

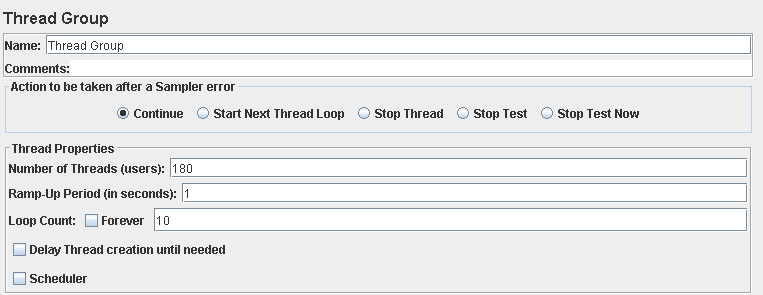
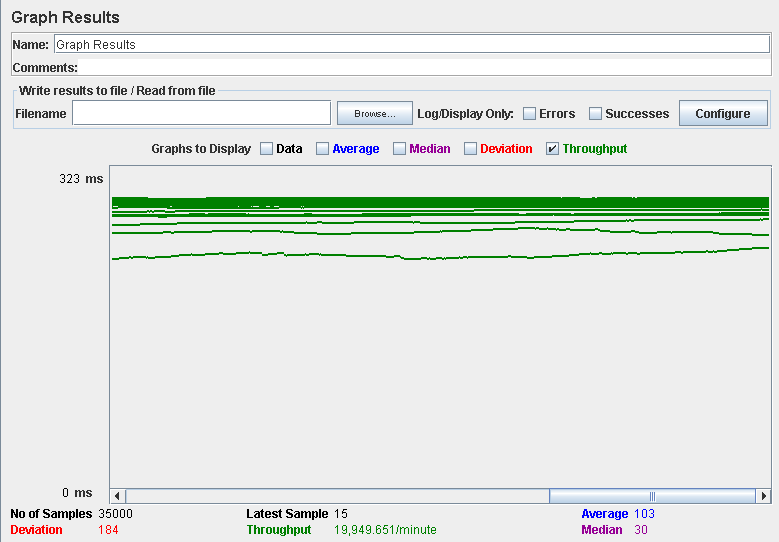
### As a rookie I want to manage my finder, which involves updating the search criteria, listing its contents, and clearing it.



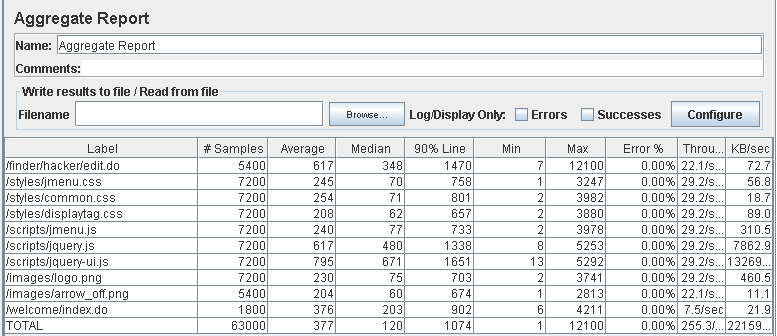
Performance test 90% results: Total 2,464s.



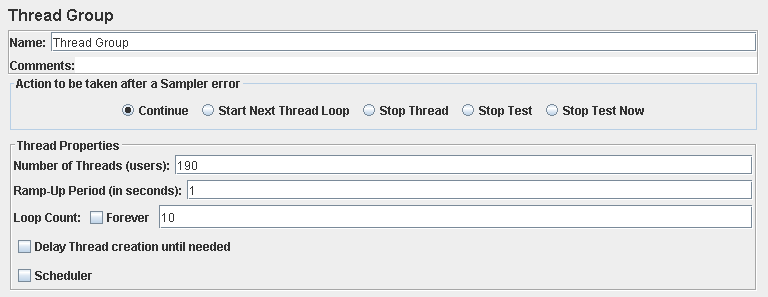
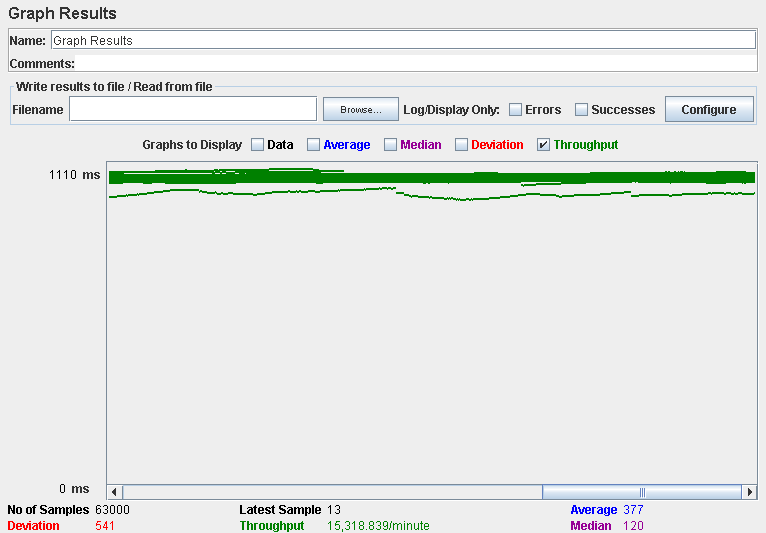
Performance thread results: 19,949 per minute.



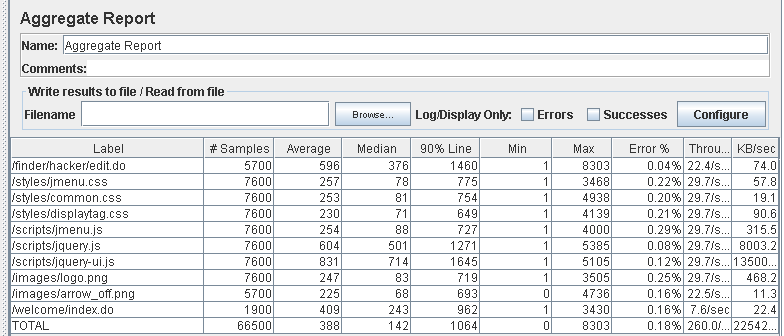
Performance test 90% results: Total 9,587s.



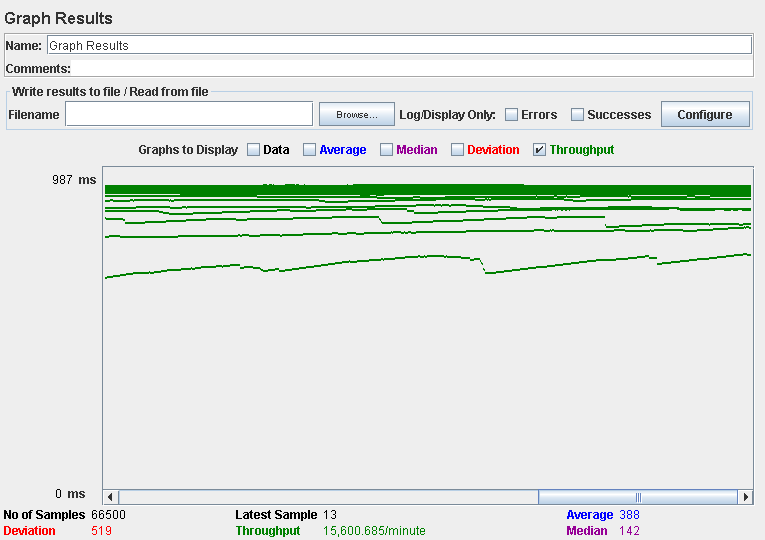
Performance thread results: 15,318 per minute.



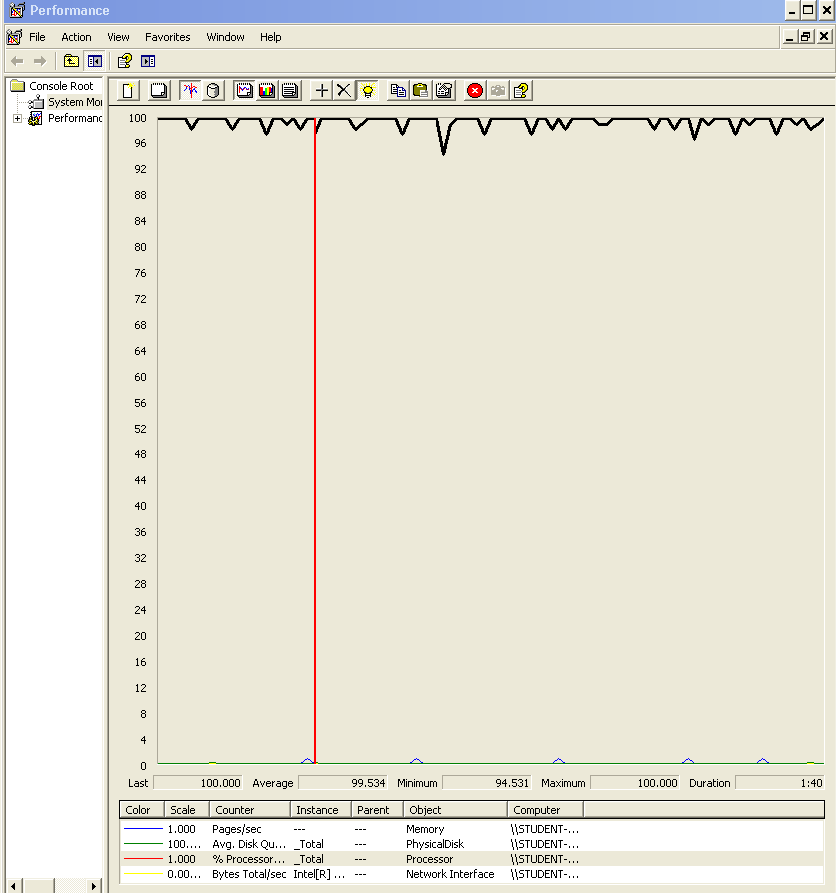
Performance test 90% results: Total 9.655s.



Performance thread results: 15,600 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs perfectly.

180 users and 10 loops: the application runs perfectly.

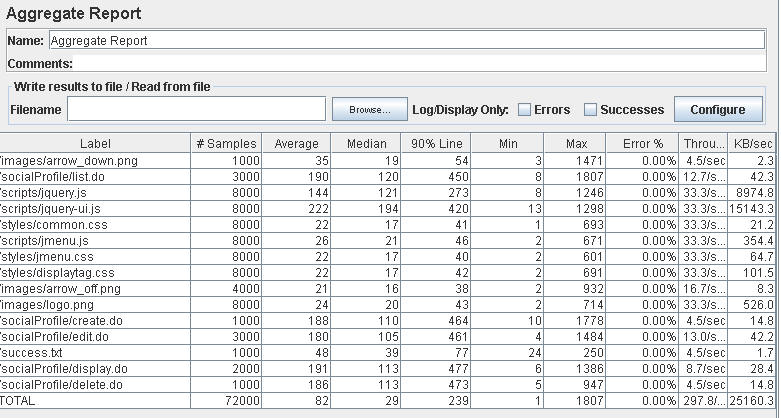
190 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

## Use case 16:

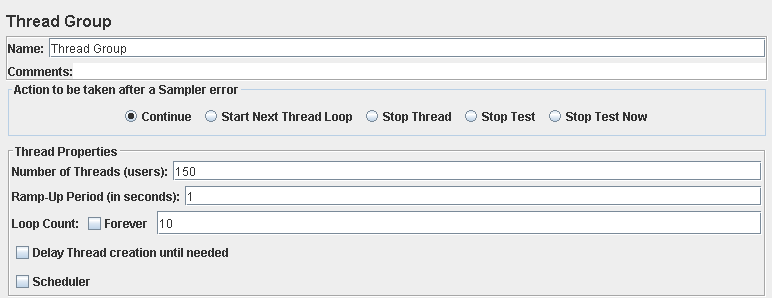
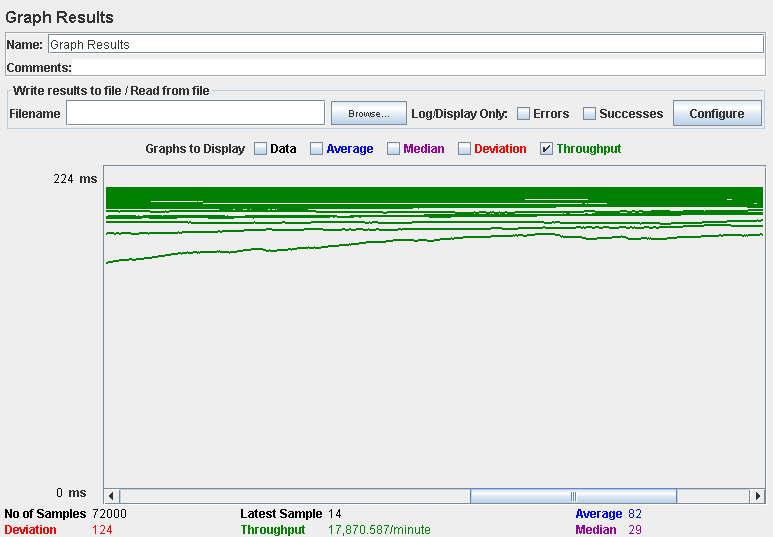
### As an authenticated actor I want to manage my social profiles, which includes listing, showing, creating, updating, and deleting them



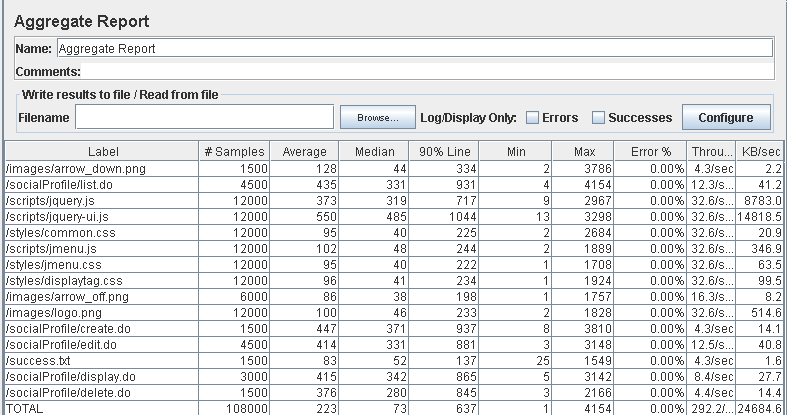
Performance test 90% results: Total 3,399s.



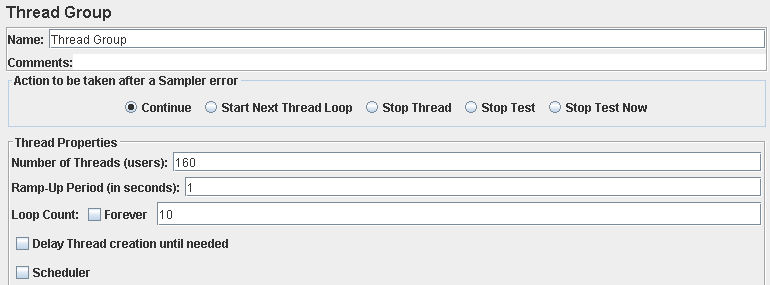
Performance thread results: 17,780 per minute.



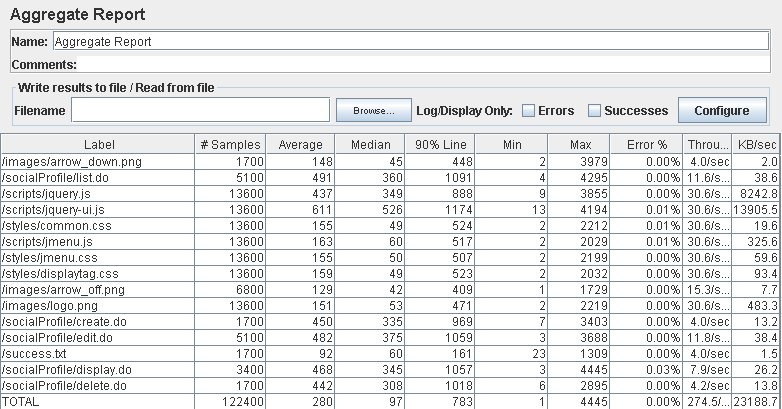
Performance test 90% results: Total 8,147s.



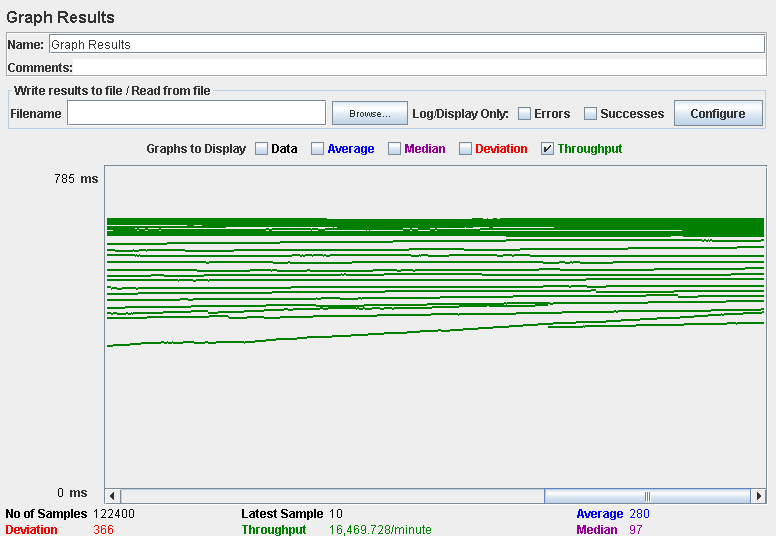
Performance thread results: 17,531 per minute.



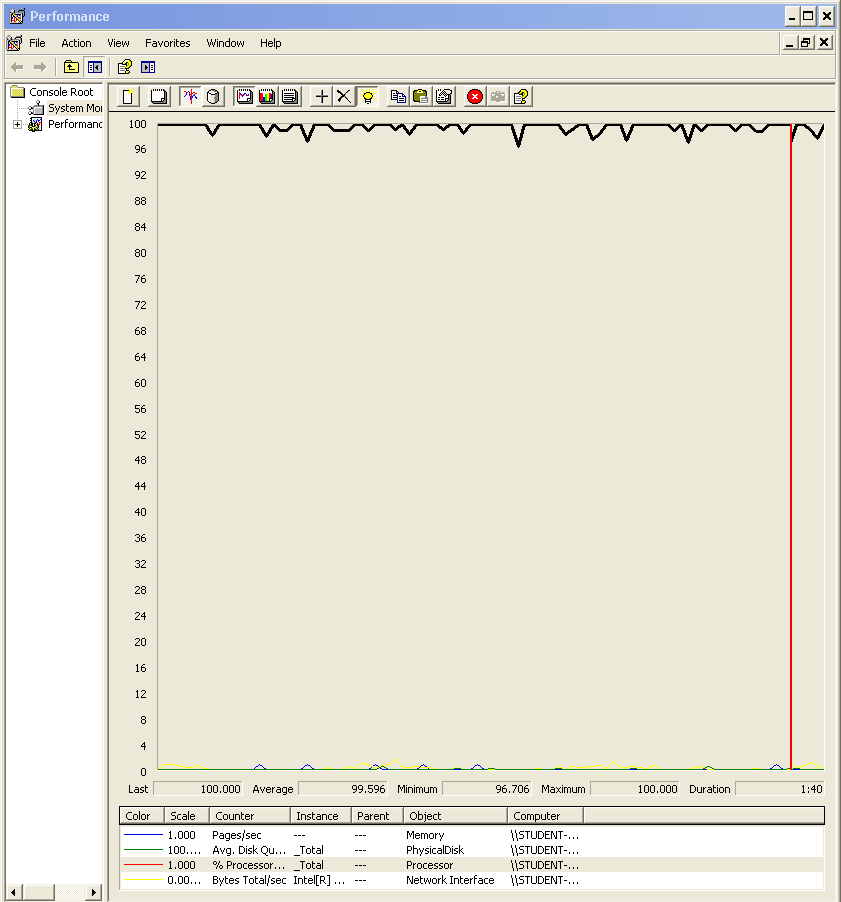
Performance test 90% results: Total 10,816s.



Performance thread results: 16,469 per minute.



Computer performance:



### Analysis results:

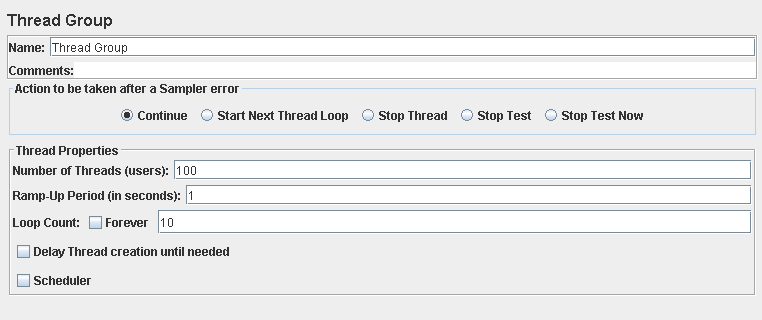
100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application runs perfectly.

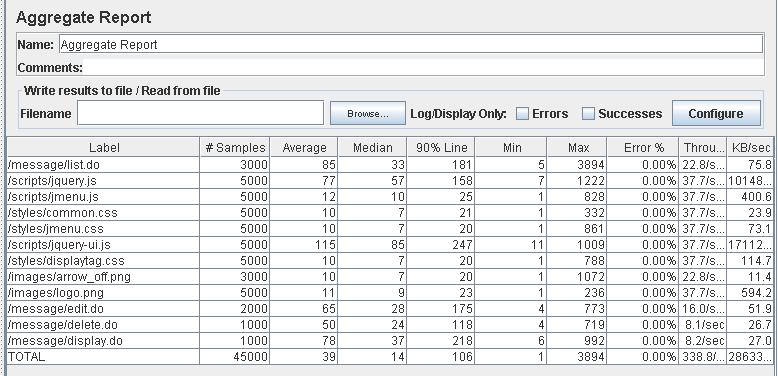
160 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

## Use case 17:

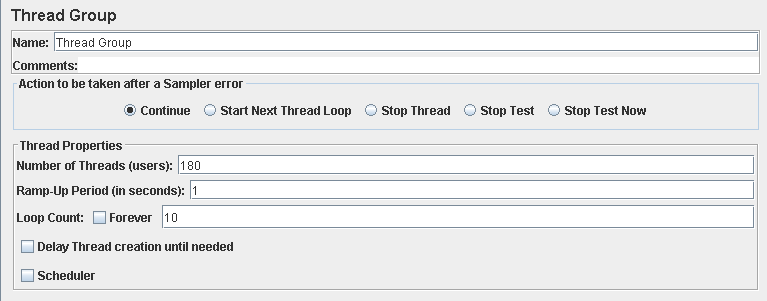
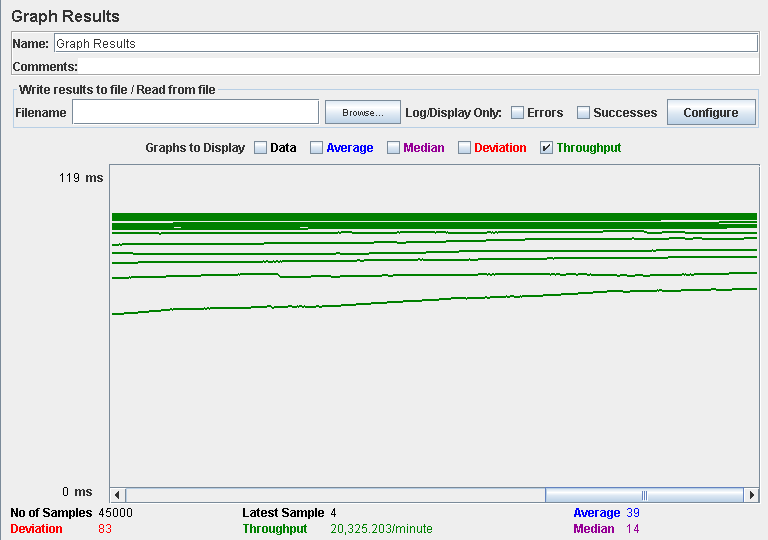
### As an authenticated actor I want to manage my messages, which includes listing them grouped by tag, showing them, sending a message to an actor, deleting a message that I got.



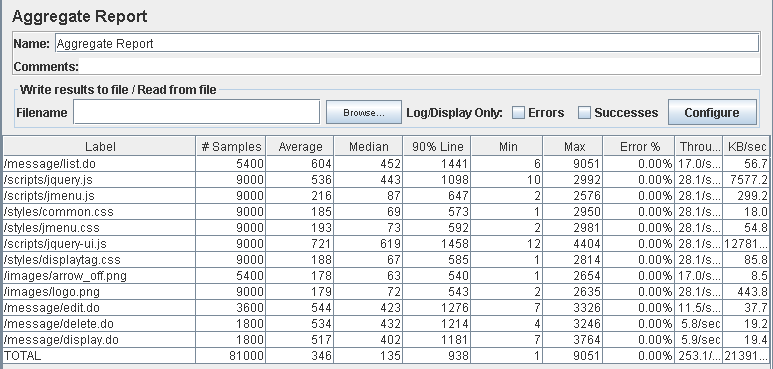
Performance test 90% results: Total 1,226s.



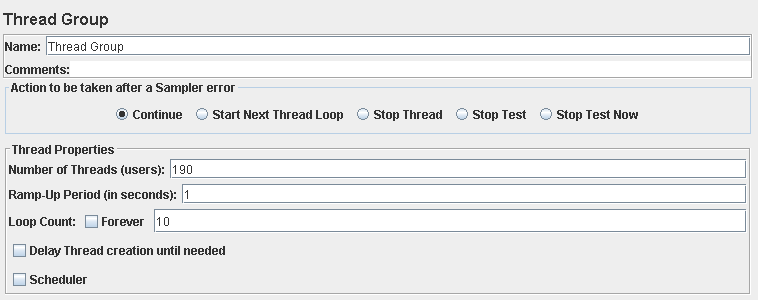
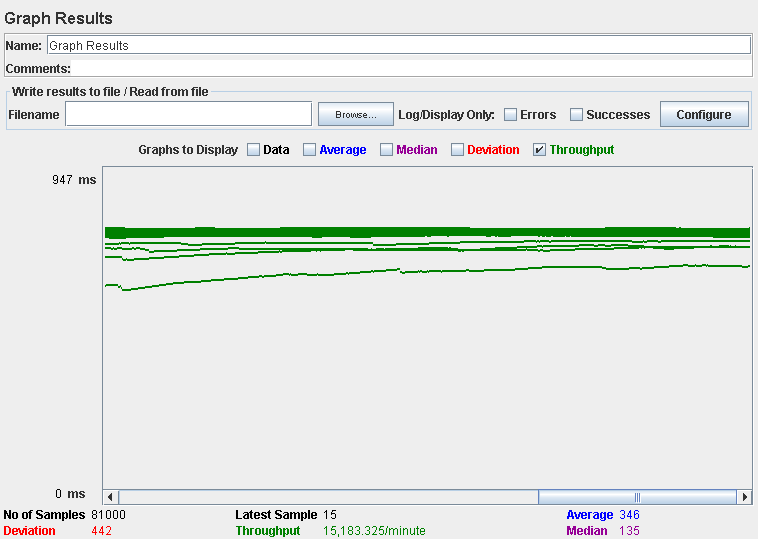
Performance thread results: 20,325 per minute.



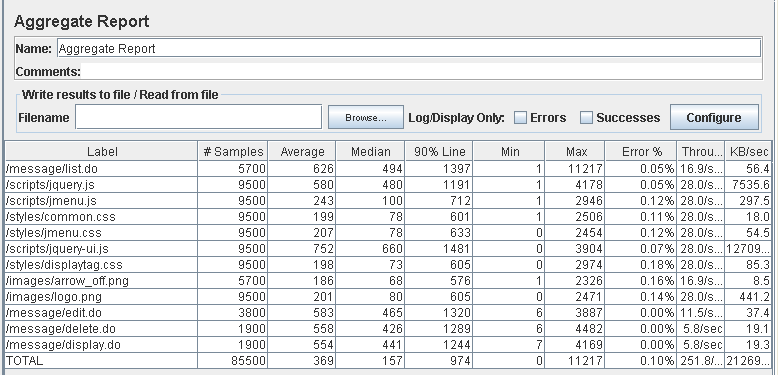
Performance test 90% results: Total 11,148s.



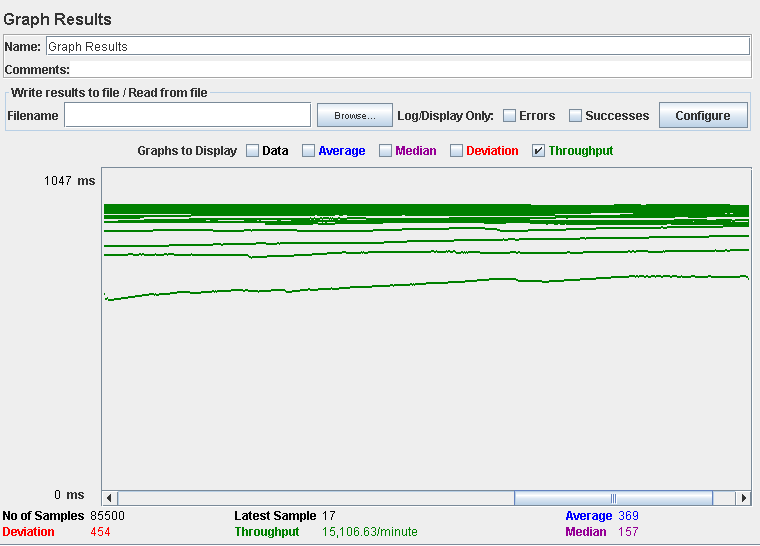
Performance thread results: 15,183 per minute.



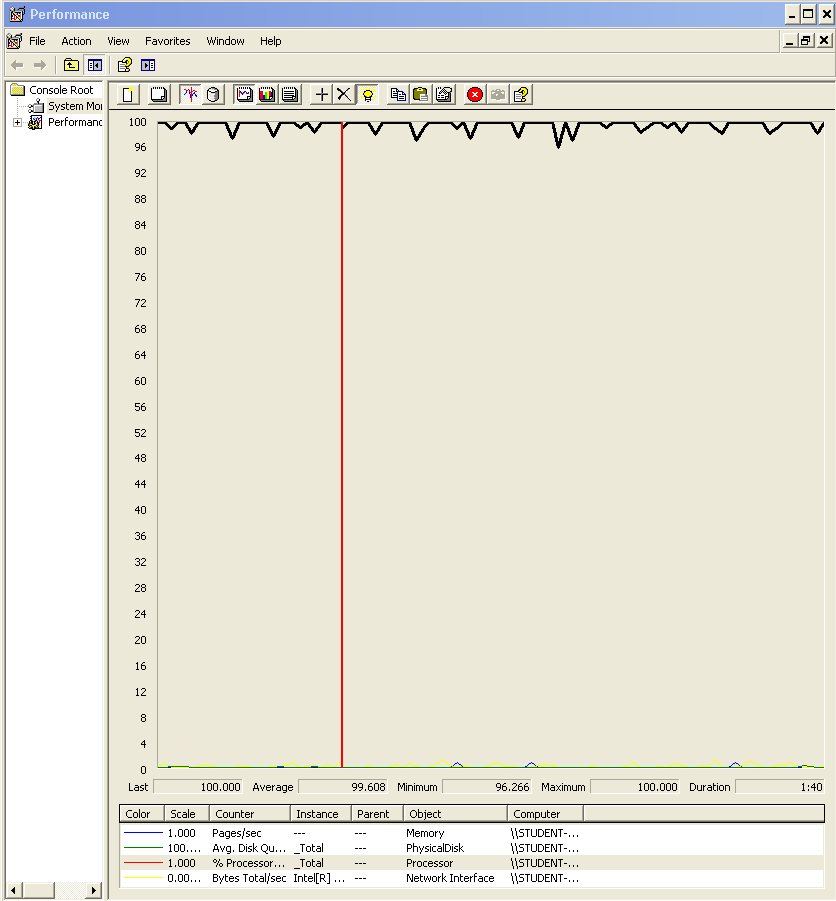
Performance test 90% results: Total 11,654s.



Performance thread results: 15,106 per minute.



Computer performance:



### Analysis results:

100 users and 10 loops: the application runs perfectly.

180 users and 10 loops: the application runs perfectly.

190 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

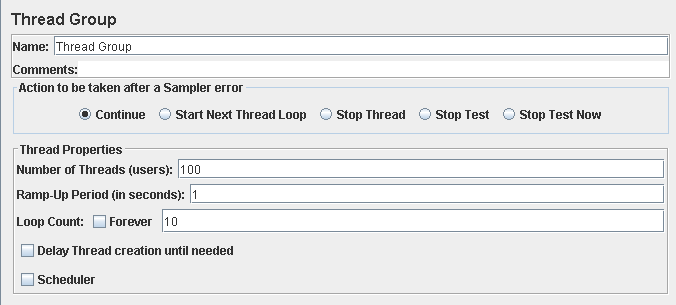
## Use case 18, 19, 20 & 21:

### As an admin I want to broadcast a notification message to the actors of the system. The message must have tag “SYSTEM” by default.

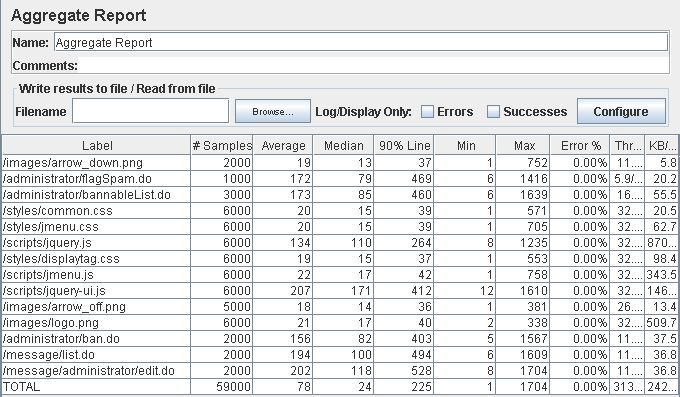
### As an admin I want to launch a process that flags the actors of the system as spammers or not-spammers.

### As an admin I want to Ban an actor with the spammer flag.

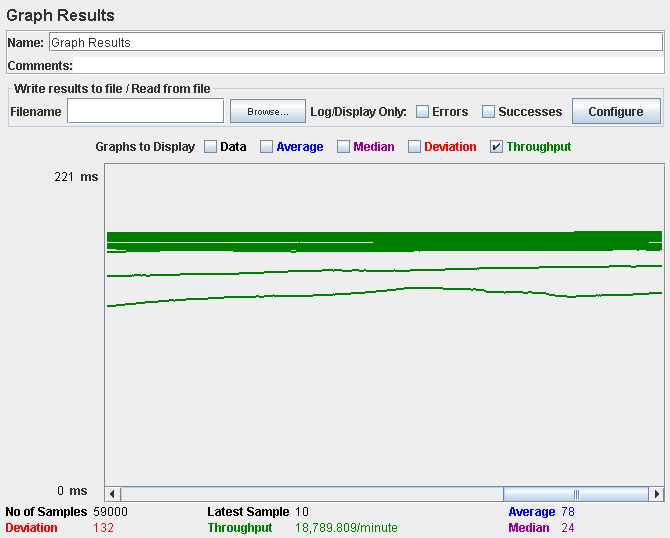
### As an admin I want to unban an actor who was banned previously.



Performance test 90% results: Total 3,300s.

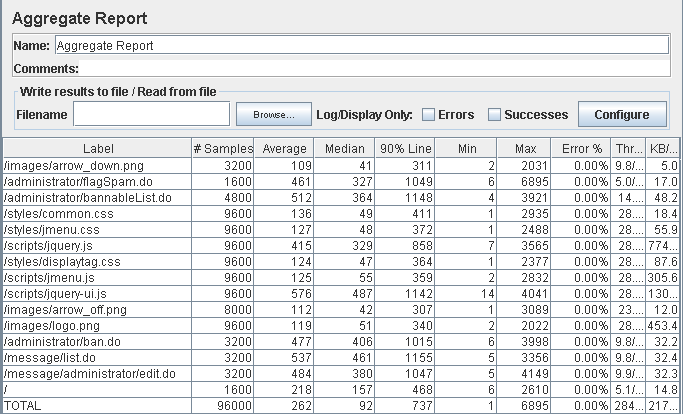


Performance thread results: 18,789 per minute.

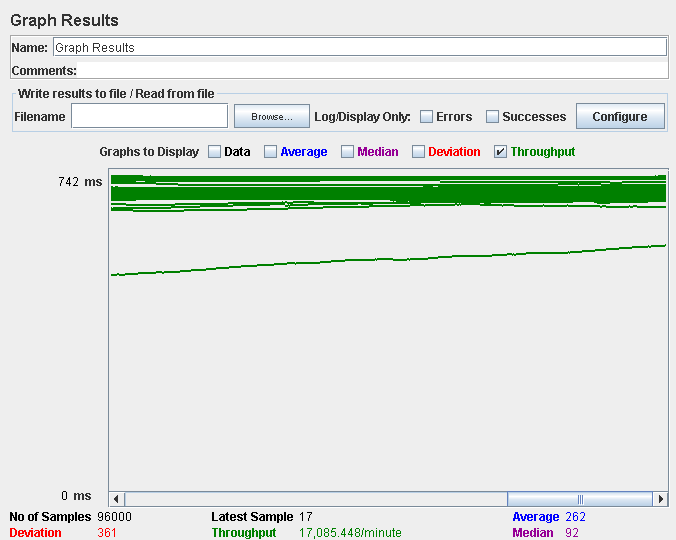


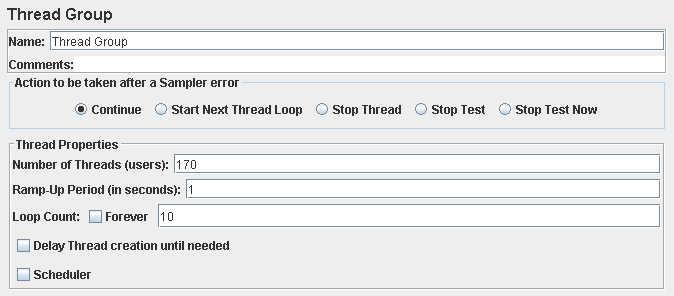


Performance test 90% results: Total 9,987s.

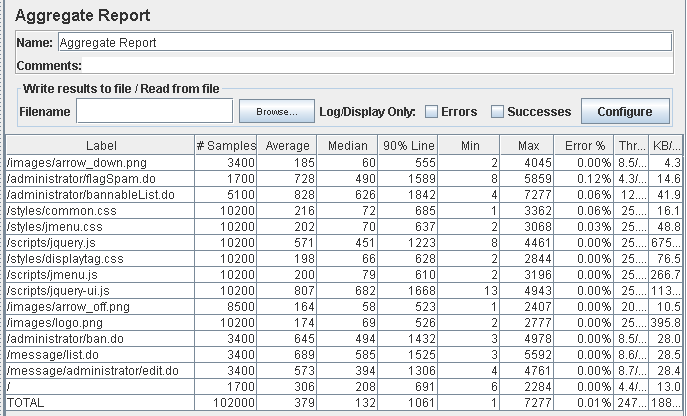


Performance thread results: 17,085 per minute.

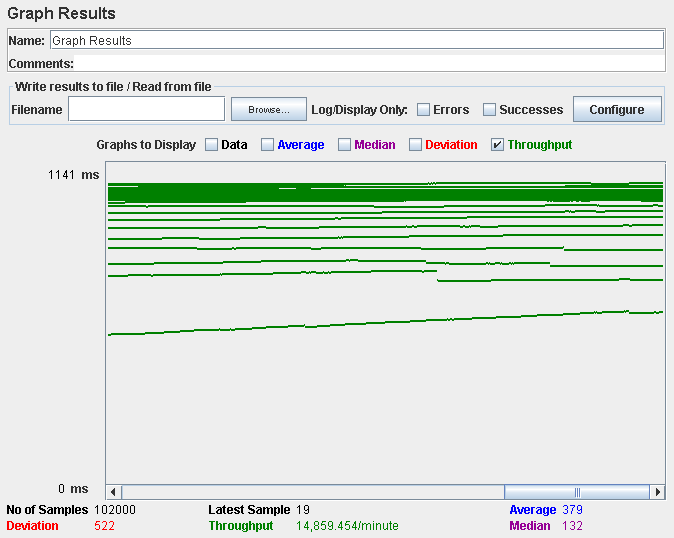




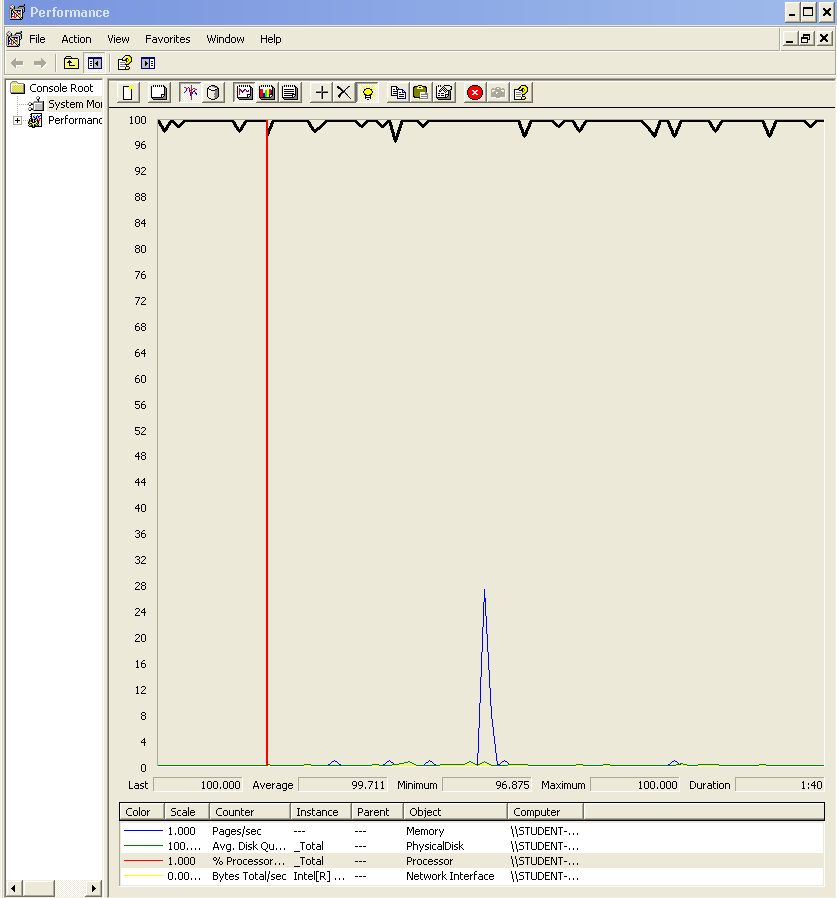
Performance test 90% results: Total 15,440s.



Performance thread results: 14,859 per minute.



Computer performance:



# Analysis results:

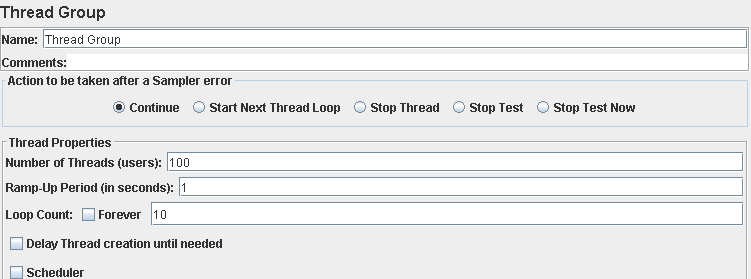
100 users and 10 loops: the application runs perfectly.

160 users and 10 loops: the application runs perfectly.

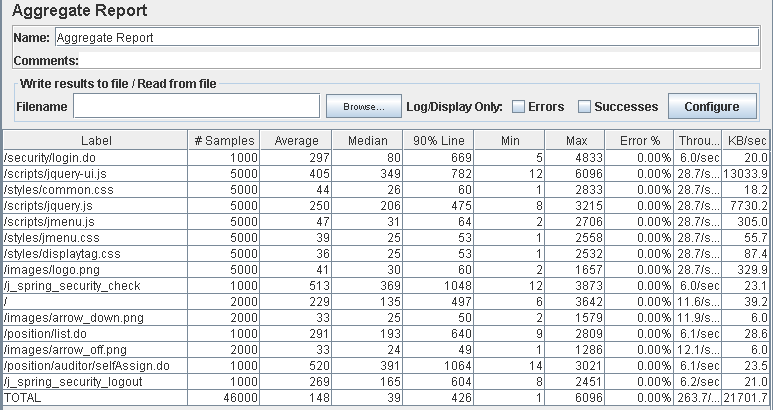
170 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

# Use case 22:

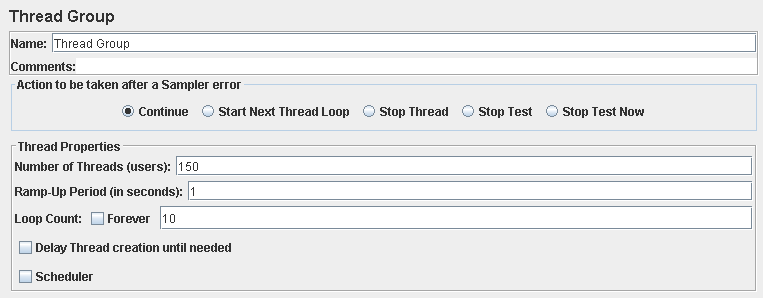
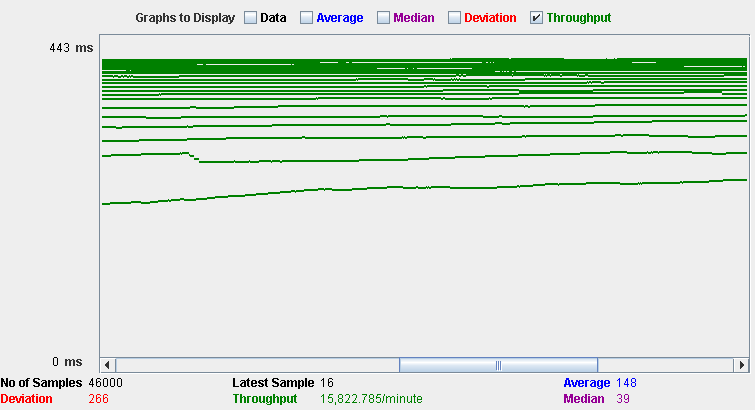
### As an auditor I want to log into the system, self-assign a position and log out.



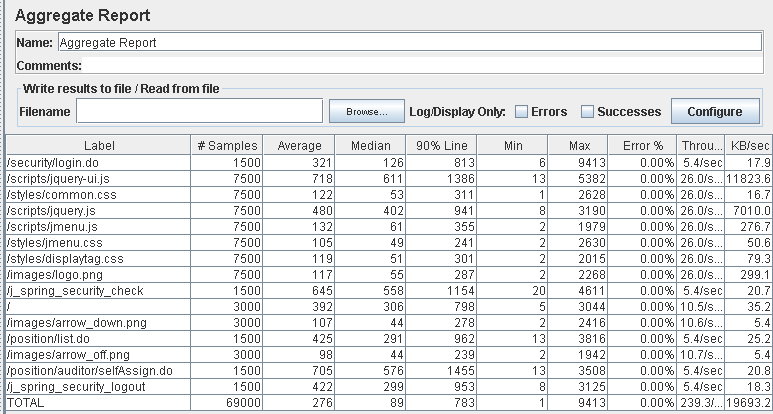
Performance test 90% results: Total 6,595s.



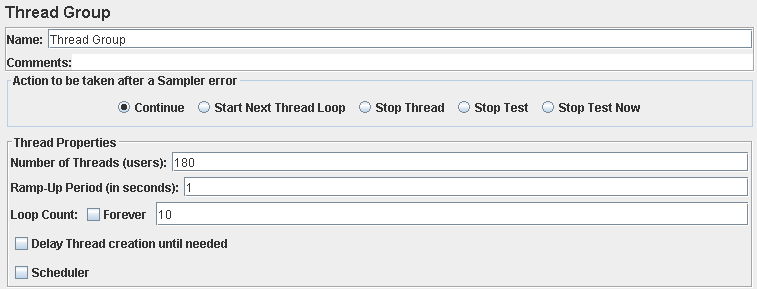
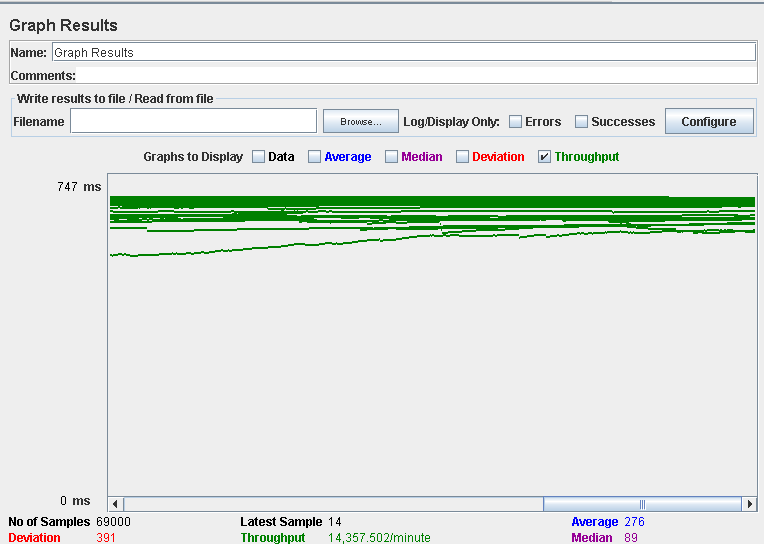
Performance thread results: 15,822 per minute.



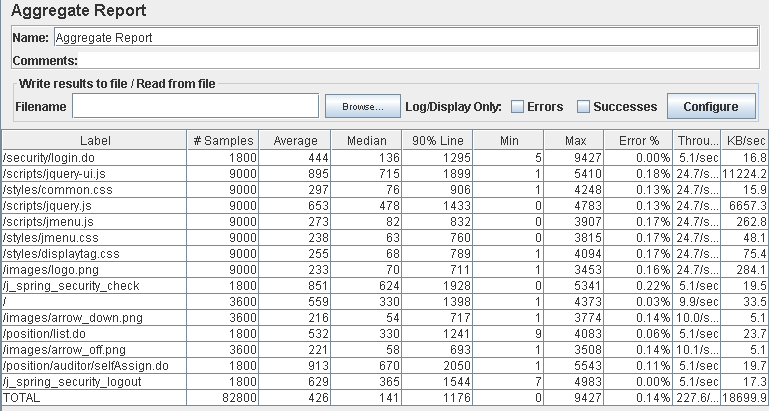
Performance test 90% results: Total 11,257s.



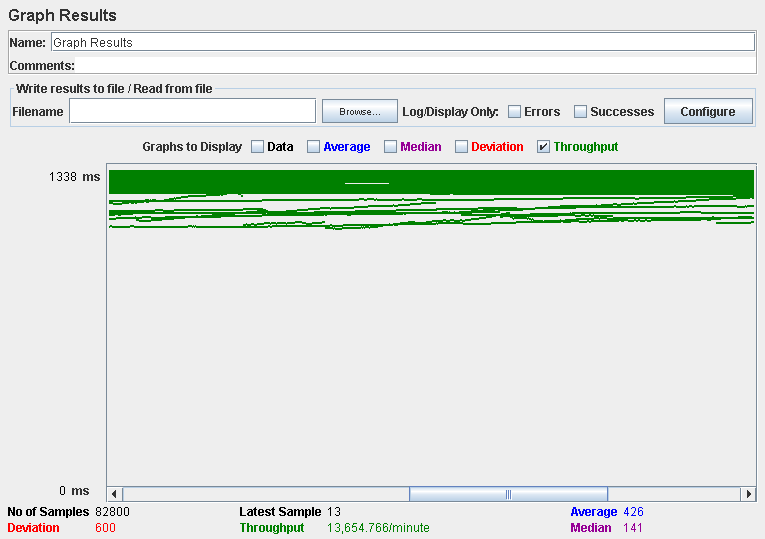
Performance thread results: 14,357 per minute.



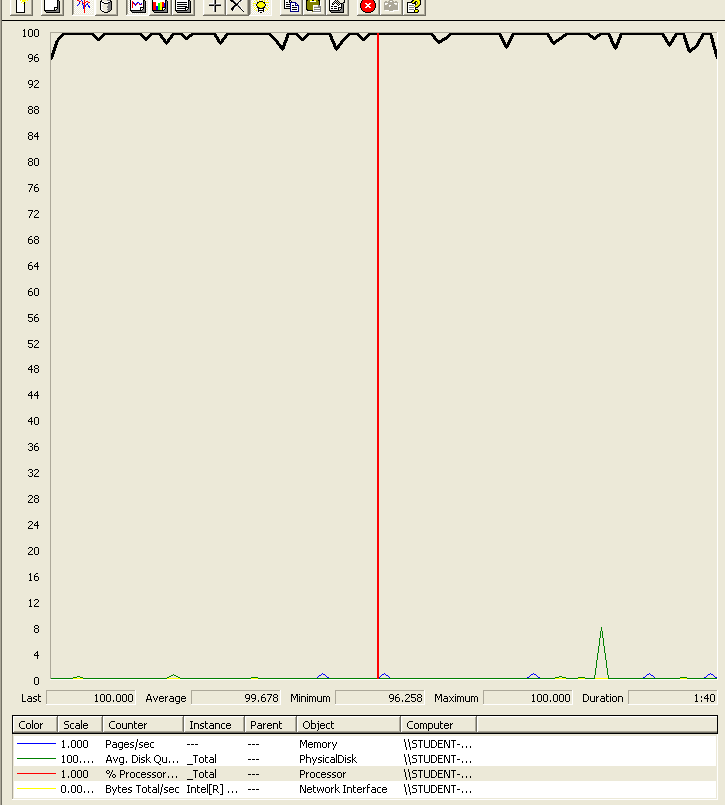
Performance test 90% results: Total 19,372s.



Performance thread results: 13,654 per minute.



Computer performance:



# Analysis results:

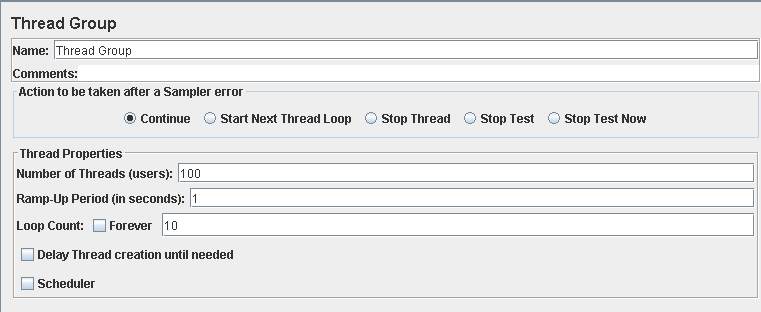
100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application runs perfectly.

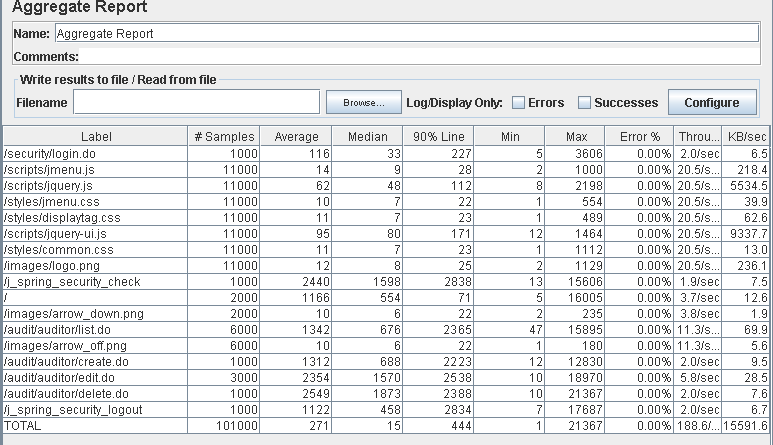
180 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

# Use case 23:

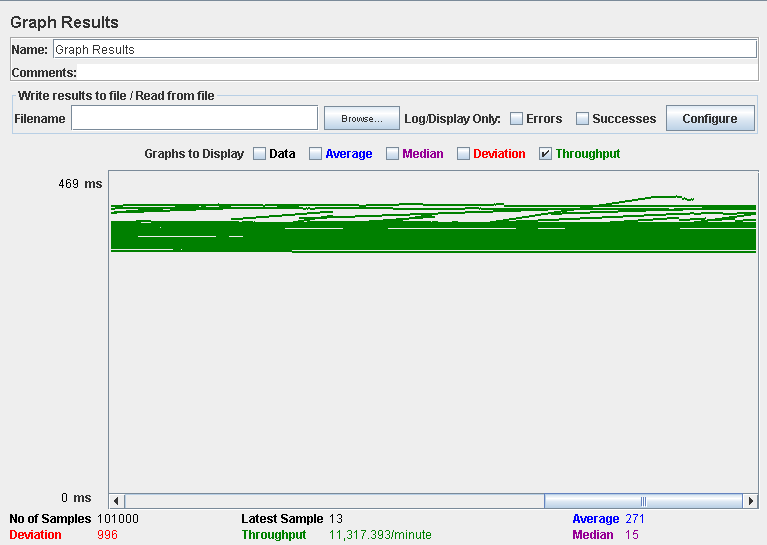
### As an auditor I want to manage my audits which includes listing them, showing them, creating them, updating, and deleting them.

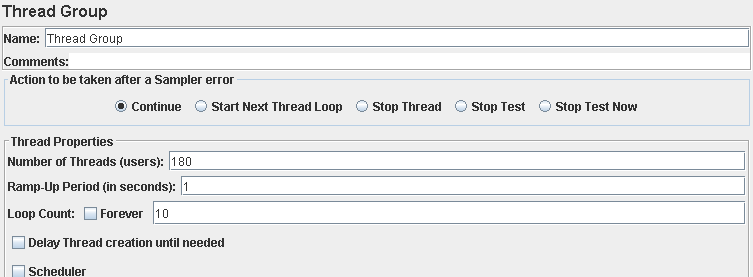


Performance test 90% results: Total 16,376s.

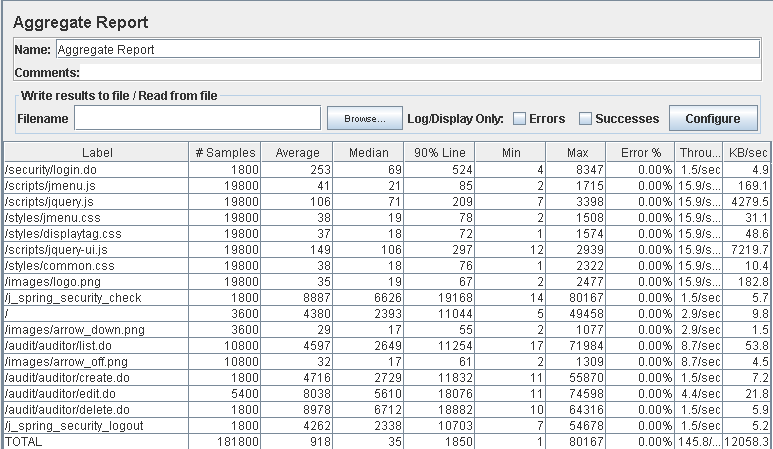


Performance thread results: 11,317 per minute.

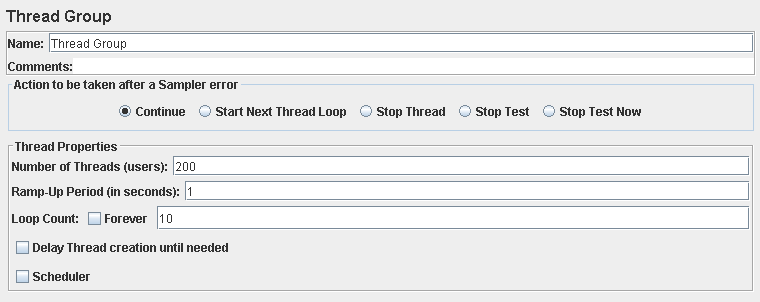
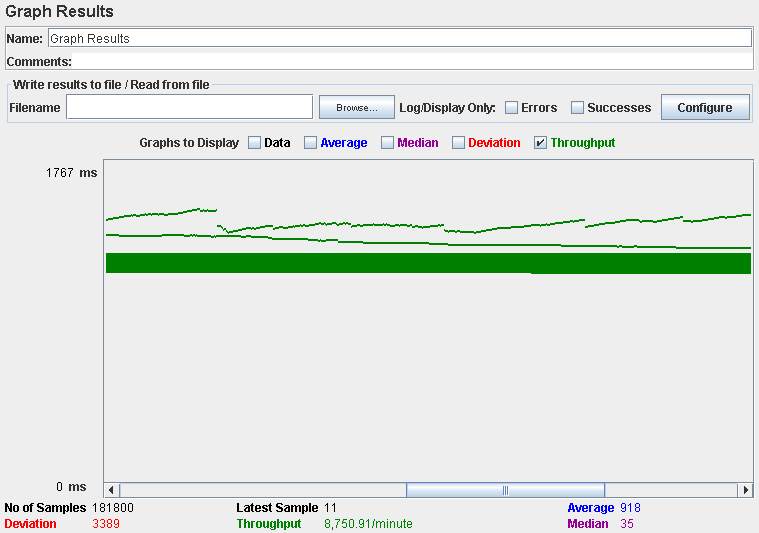




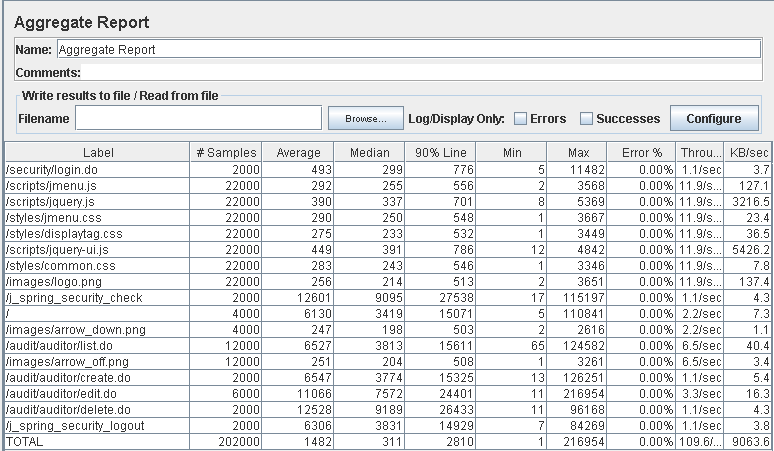
Performance test 90% results: Total 101,234s.



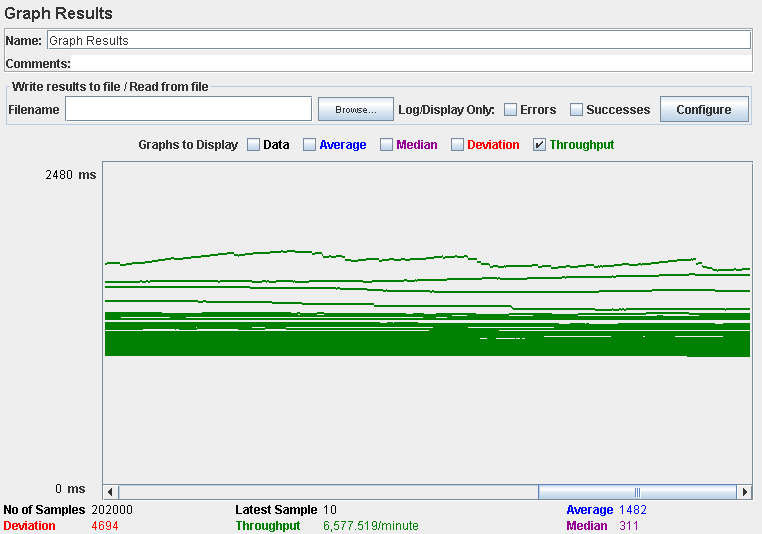
Performance thread results: 8,750 per minute.



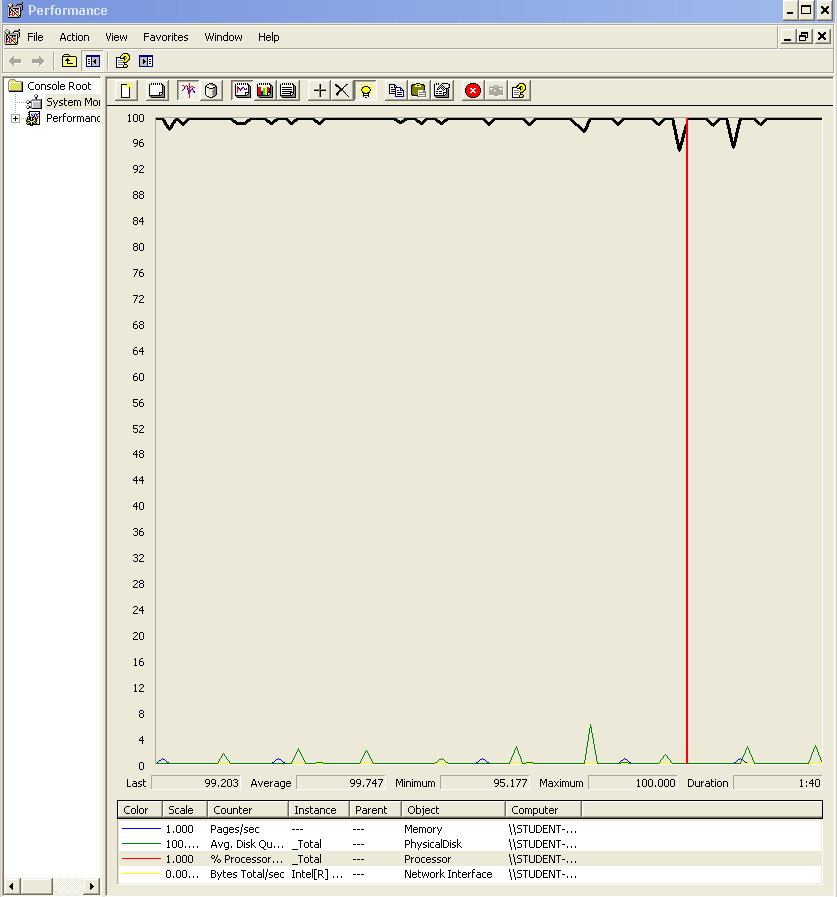
Performance test 90% results: Total 134,372s.



Performance thread results: 6,577 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

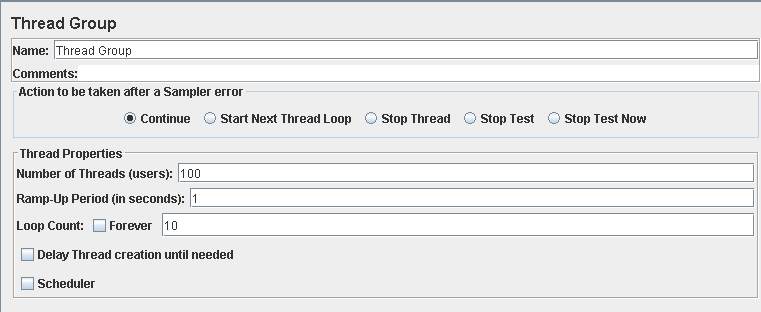
180 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

200 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

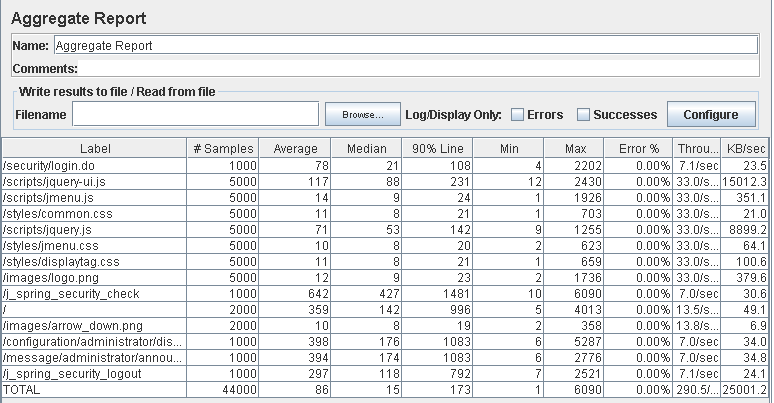
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 24:

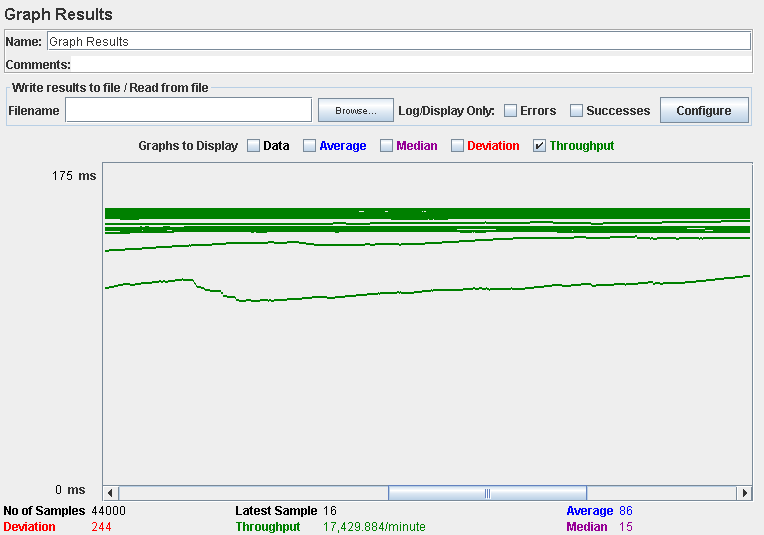
### As and admin I want to notify my users about of the rebranding of the system only one time.

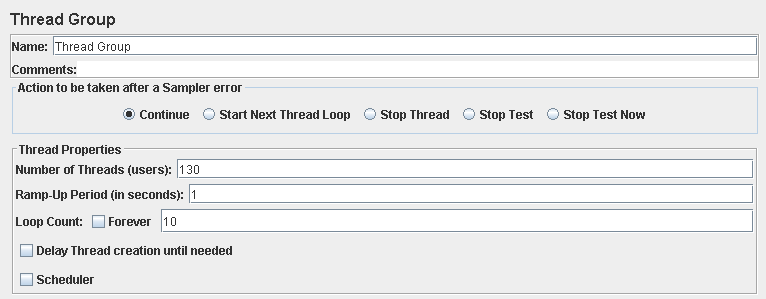


Performance test 90% results: Total 6,063s.

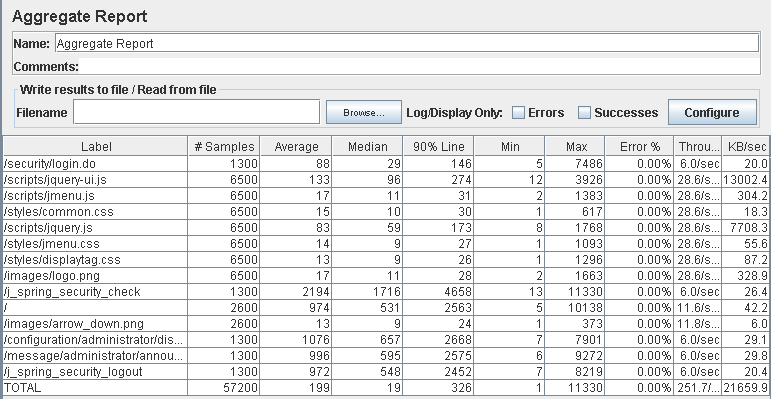


Performance thread results: 17,429 per minute.

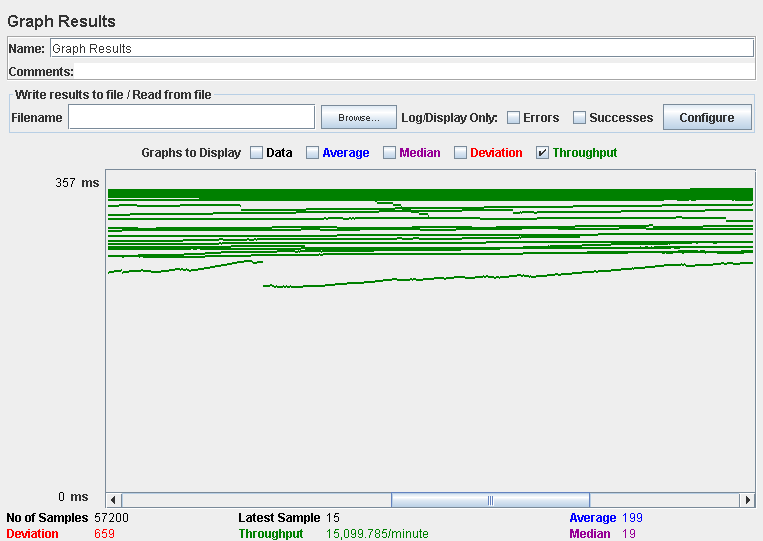


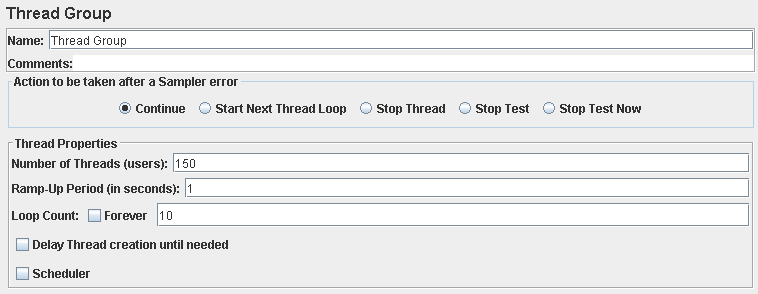


Performance test 90% results: Total 15,675s.

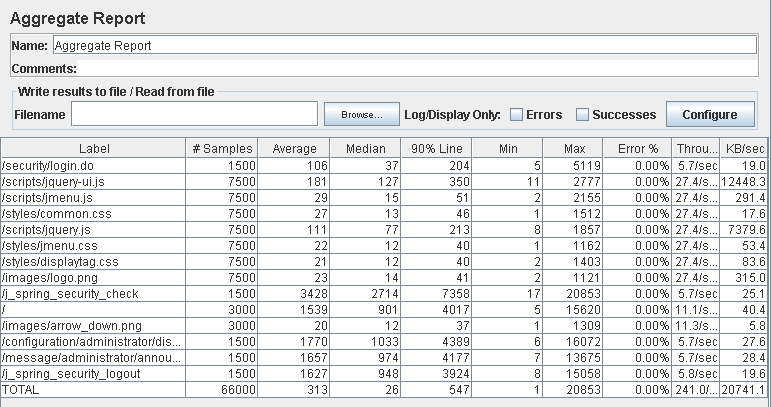


Performance thread results: 15,099 per minute.





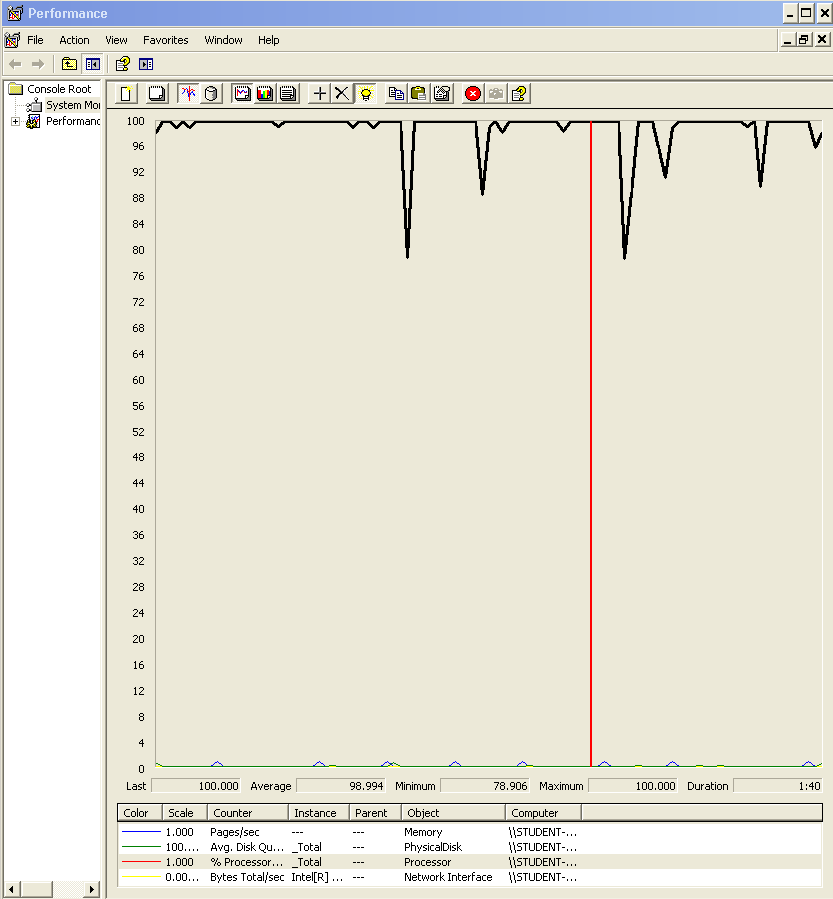
Performance test 90% results: Total 24,887s.



Performance thread results: 14,458 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

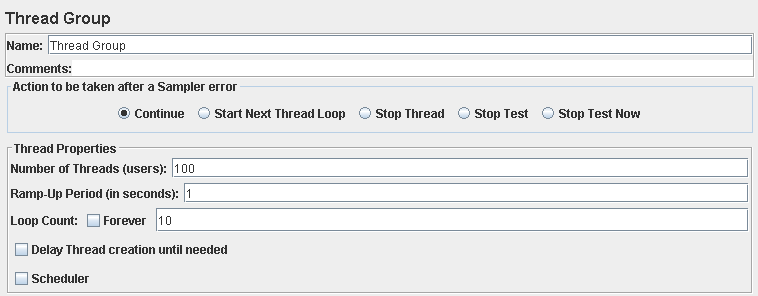
130 users and 10 loops: the application doesn’t have errors but the login time is a bit high.

150 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

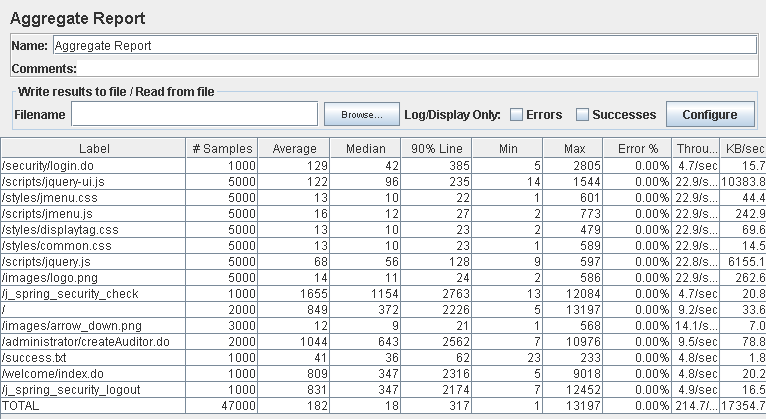
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 25:

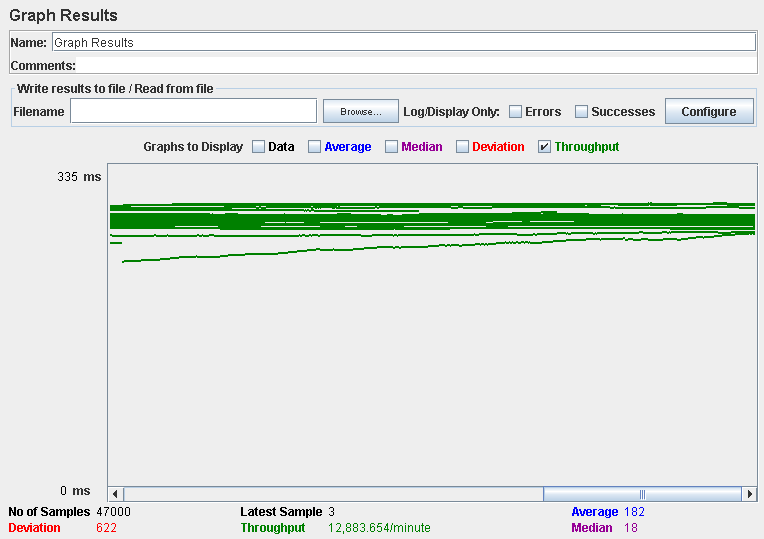
### As an admin I want to create user accounts for new auditors.

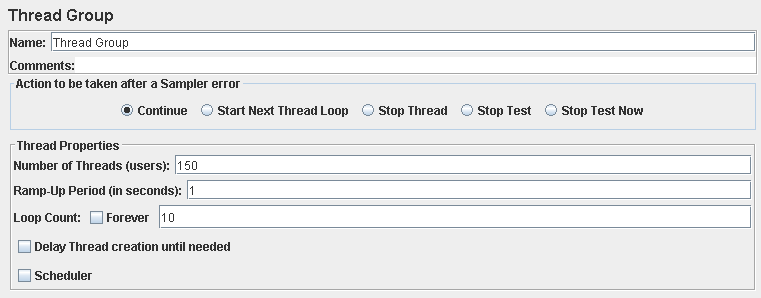


Performance test 90% results: Total 12,970s.

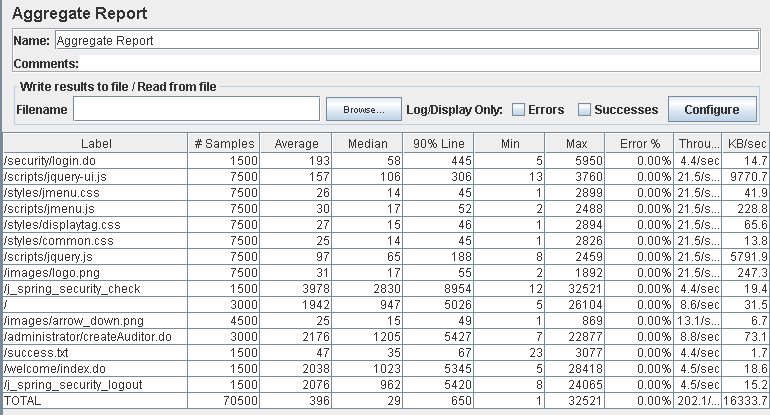


Performance thread results: 12,883 per minute.

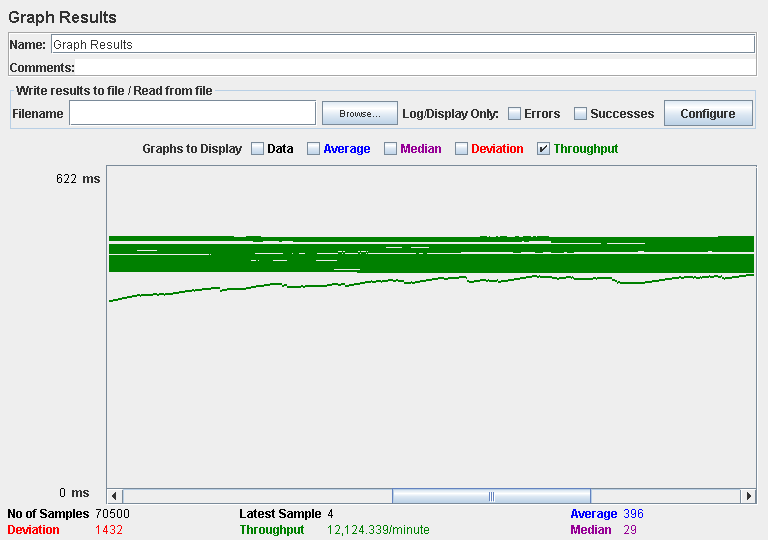


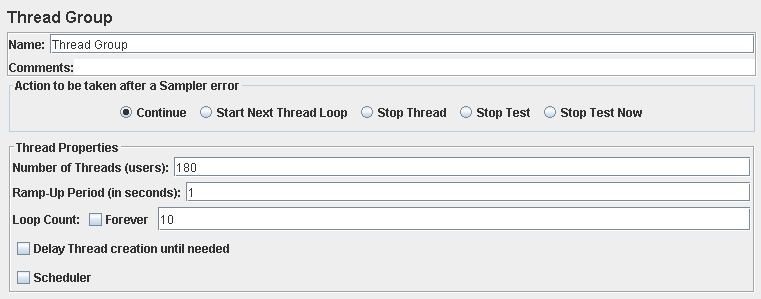


Performance test 90% results: Total 34,470s.

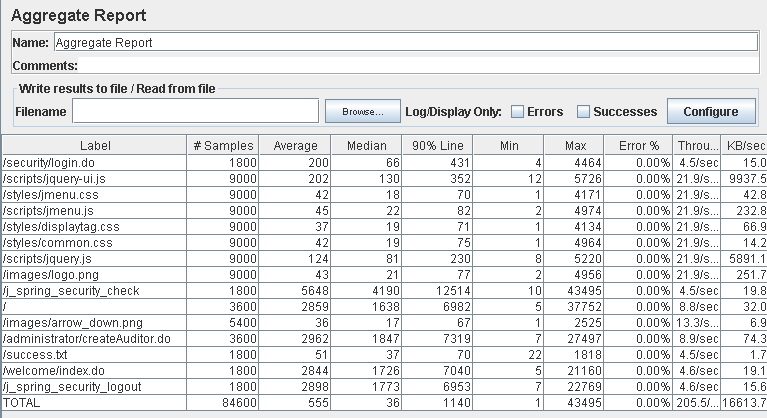


Performance thread results: 12,124 per minute.

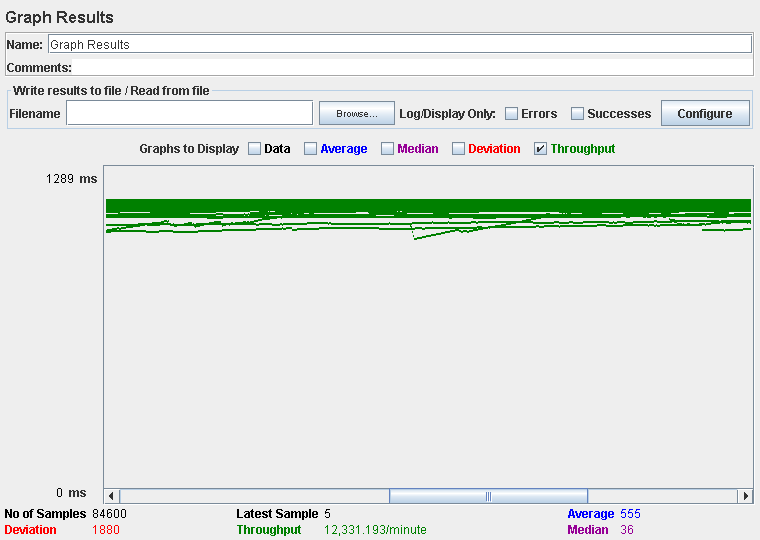




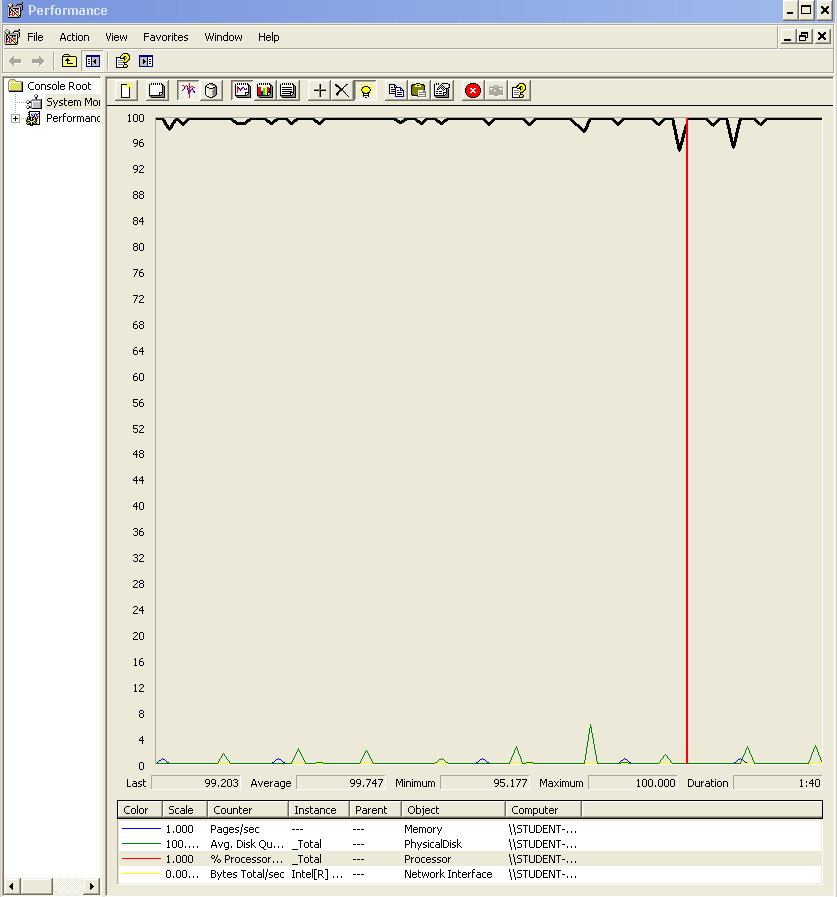
Performance test 90% results: Total 42,333s.



Performance thread results: 12,331 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

180 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

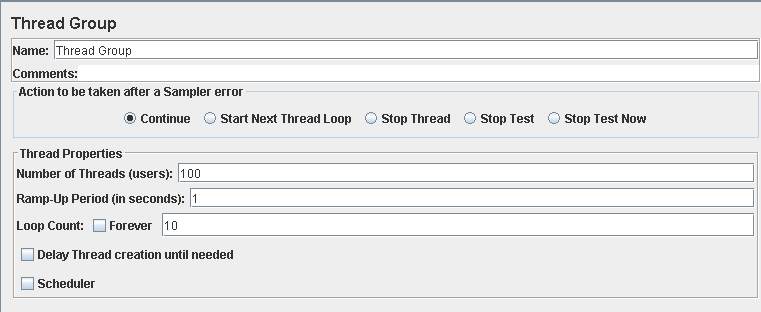
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 26 & 27:

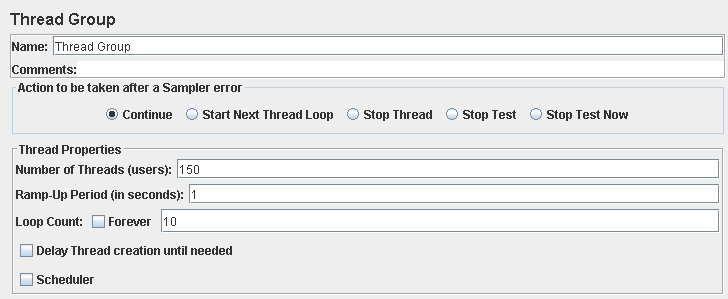
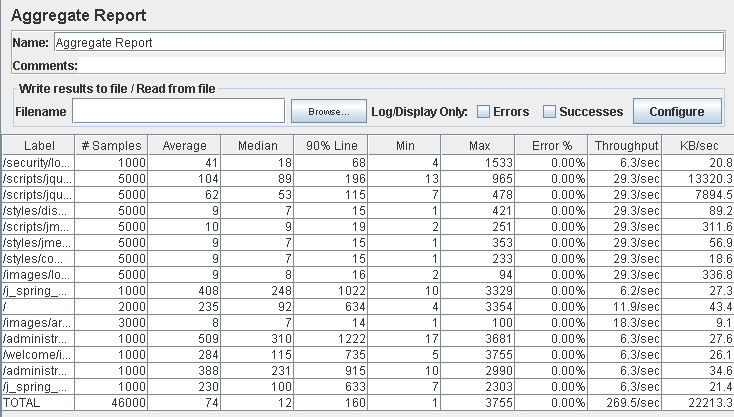
### Launch a process to compute an audit score for every company.

### Display a dashboard with the following information :

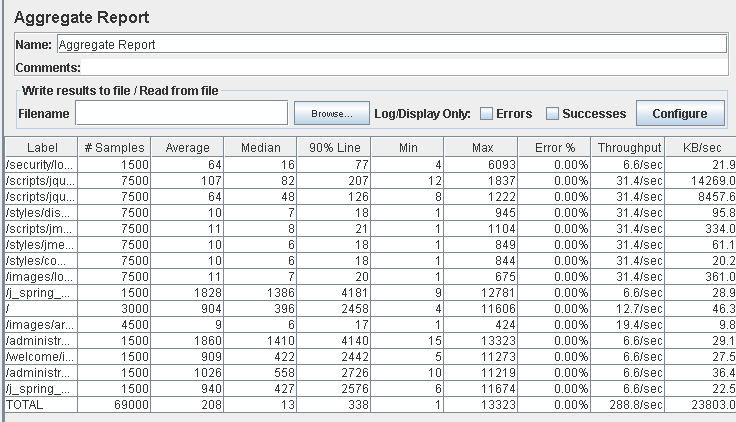
* The average, the minimum, the maximum, and the standard deviation of the audit score of the positions stored in the system.
* The average, the minimum, the maximum, and the standard deviation of the audit score of the companies that are registered in the system.
* The companies with the highest audit score.
* The average salary offered by the positions that have the highest average audit score.
* The minimum, the maximum, the average, and the standard deviation of the number of items per provider.
* The top-5 providers in terms of total number of items provided.
* The average, the minimum, the maximum, and the standard deviation of the number of sponsorships per provider.
* The average, the minimum, the maximum, and the standard deviation of the number of sponsorships per position.
* The providers who have a number of sponsorships that is at least 10% above the average number of sponsorships per provider.



Performance test 90% results: Total 5,634s.

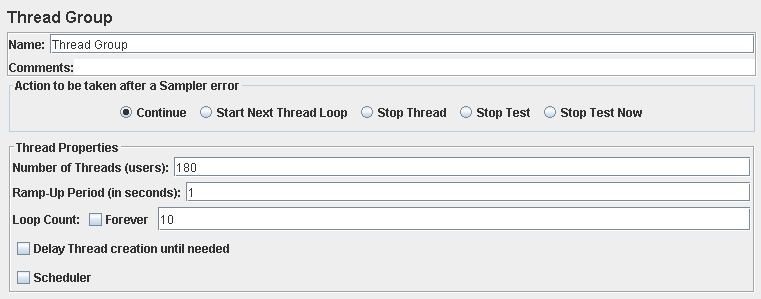


Performance test 90% results: Total 19,045s.

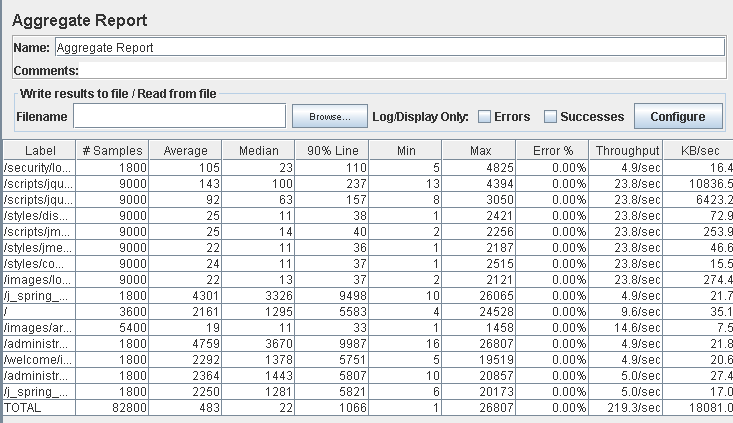


Performance thread results: 17,325 per minute.





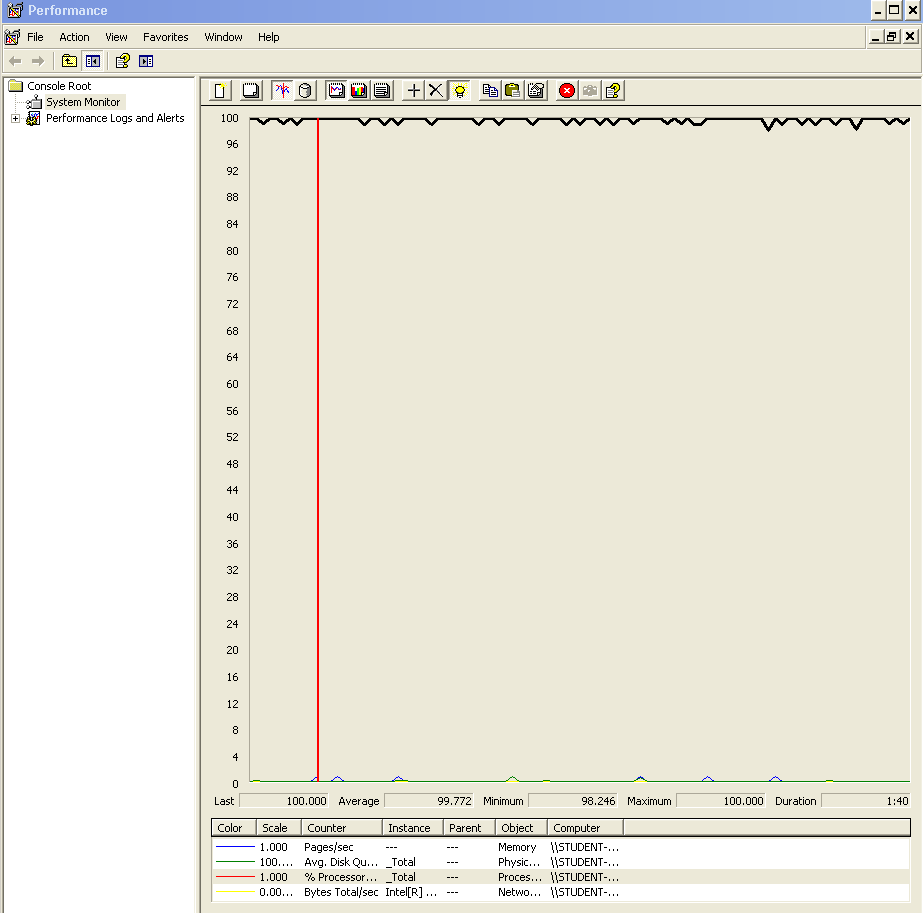
Performance test 90% results: Total 62,217s.



Performance thread results: 13,159 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application doesn’t have errors but the times are high.

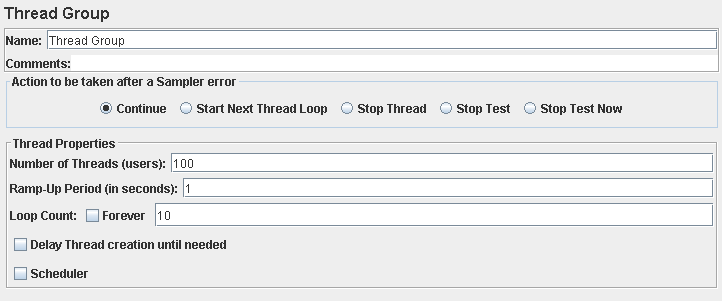
180 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

The computer performance analysis shows it could be a processors bottleneck problem.

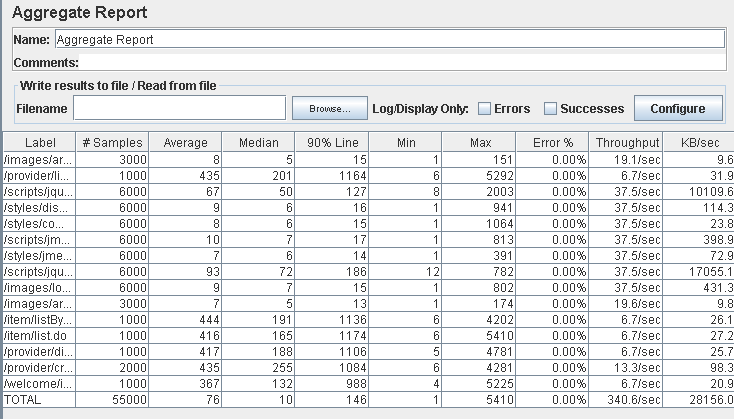
# Use case 28:

### As an actor who is not authenticated I want to:

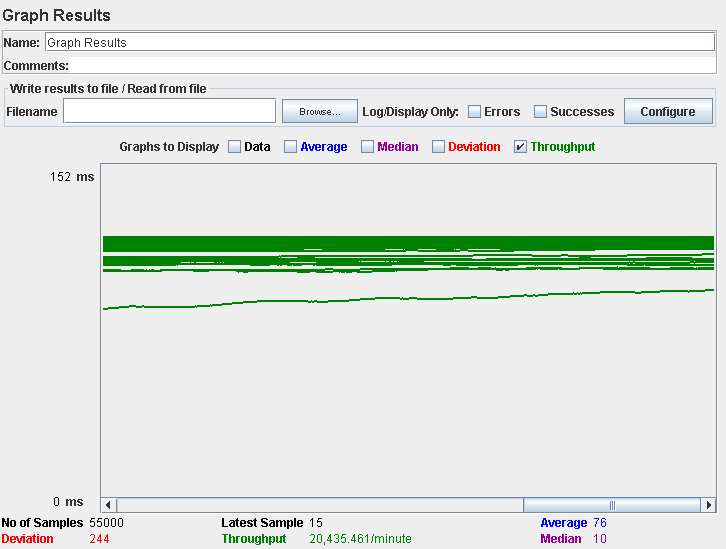
* Browse the list of providers and navigate to their items.
* Browse the list of items and navigate to their providers.
* Register to the system as a provider.



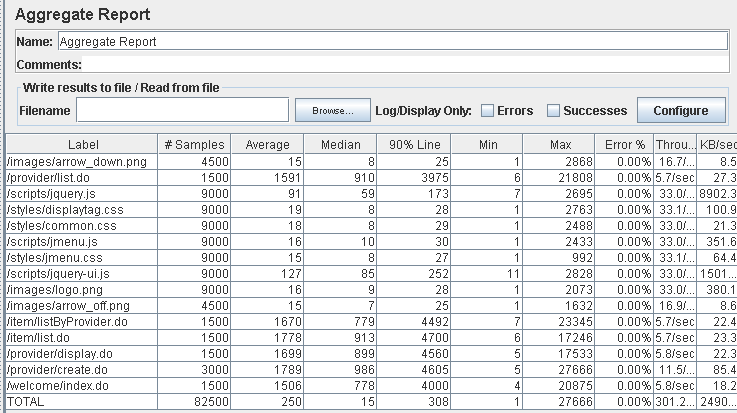
Performance test 90% results: Total 7,177s.



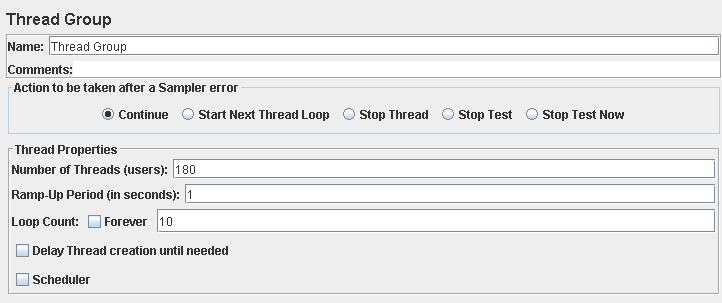
Performance thread results: 20,435 per minute.



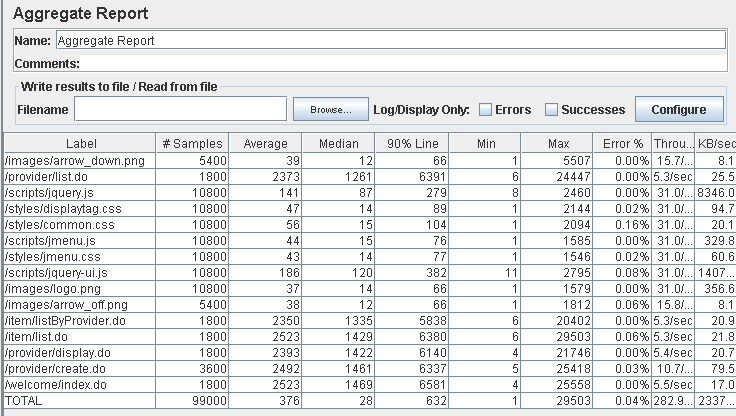
Performance test 90% results: Total 26,949s.



Performance thread results: 18,070 per minute.



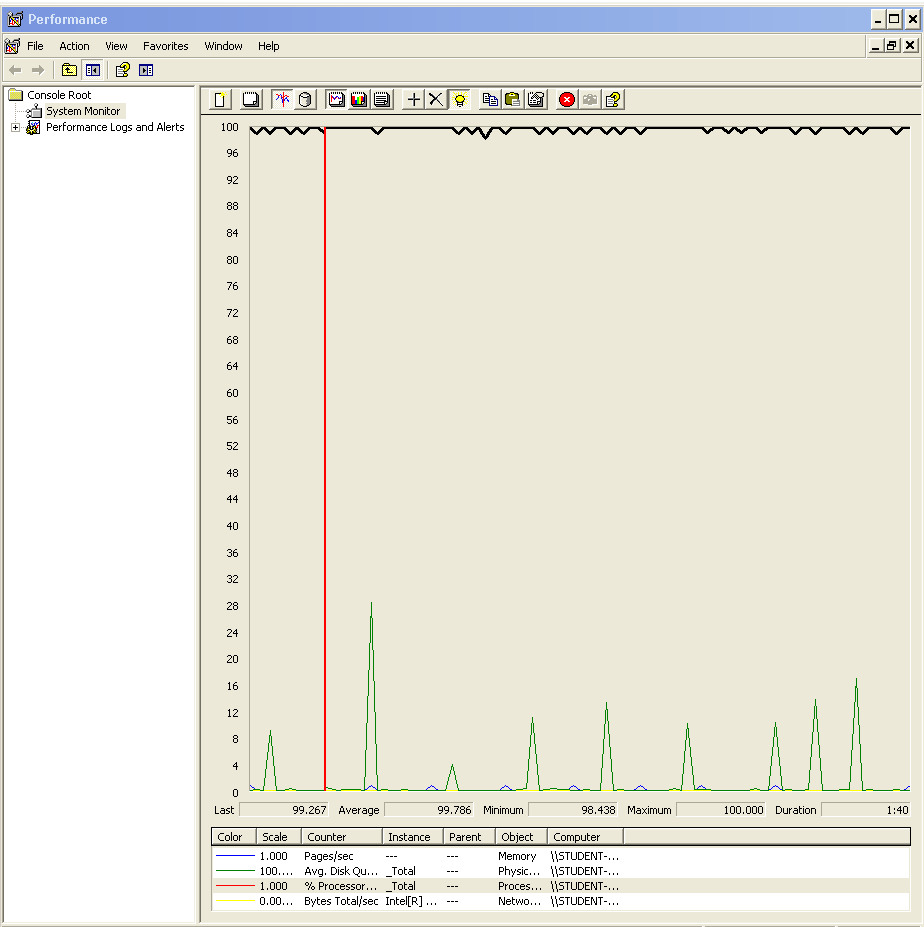
Performance test 90% results: Total 38,806s.



Performance thread results: 16,973 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

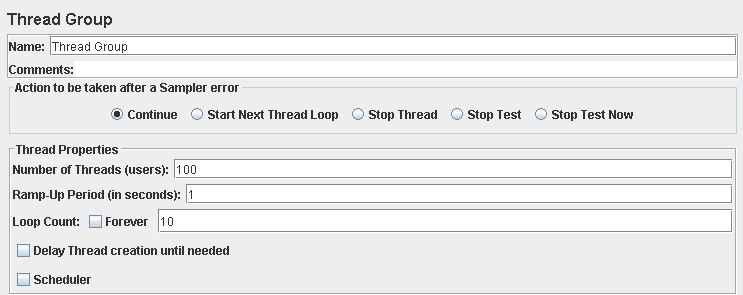
150 users and 10 loops: the application doesn’t have errors but the times are high.

180 users and 10 loops: the application begins to have some errors and the times are incredibly high.

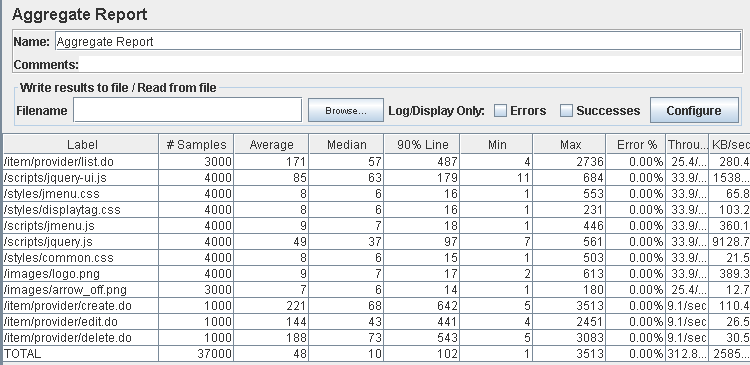
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 29:

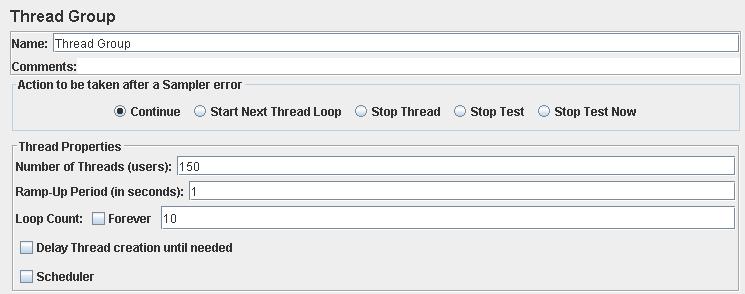
### As a provider I want to create and delete my catalogue of items.



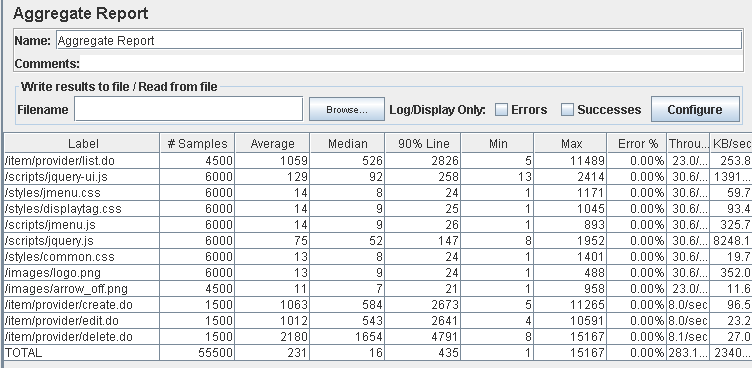
Performance test 90% results: Total 2,485s.



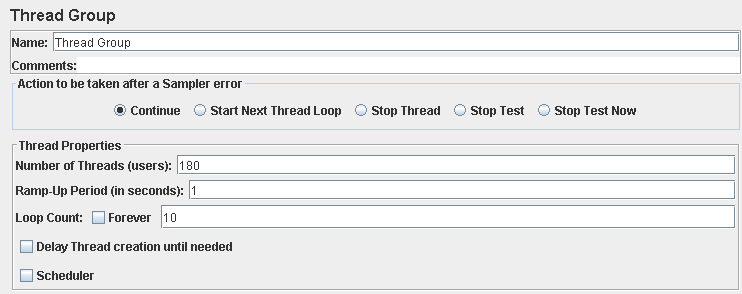
Performance thread results: 18,765 per minute.



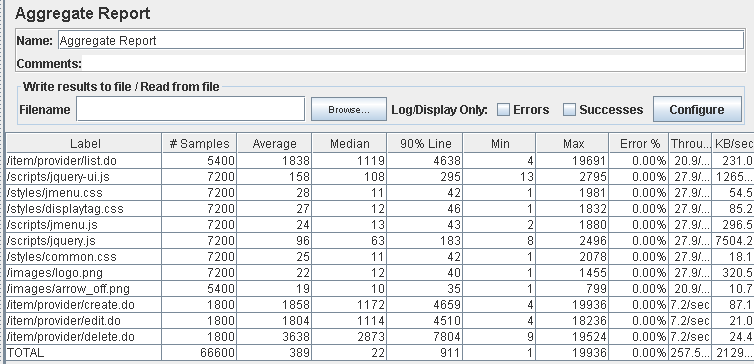
Performance test 90% results: Total 13,480s.



Performance thread results: 16,984 per minute.



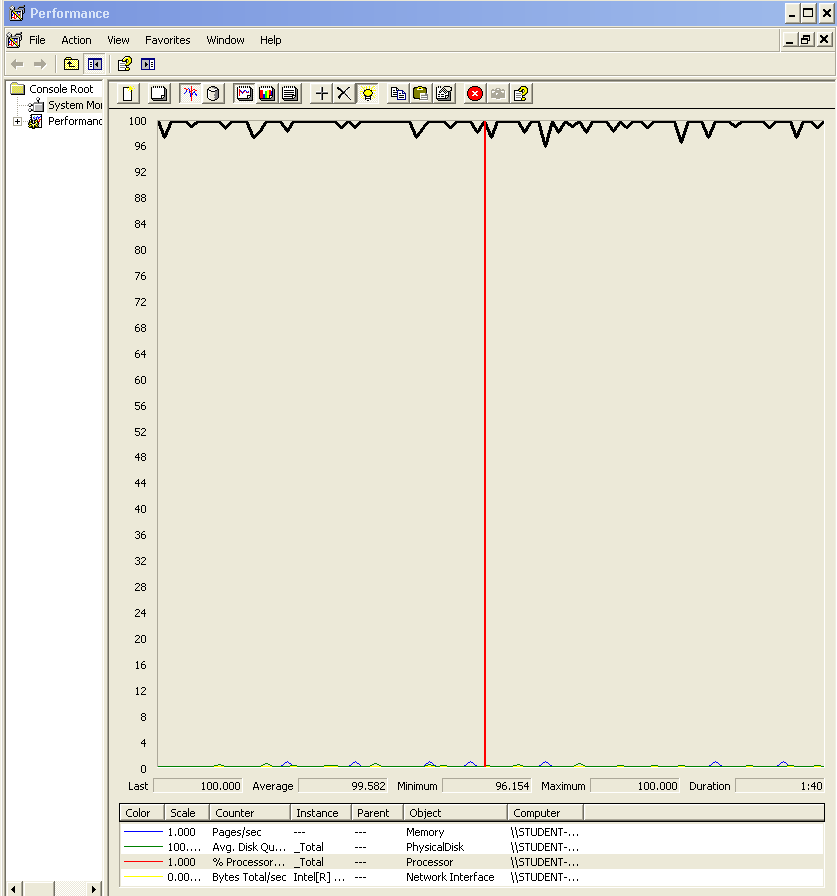
Performance test 90% results: Total 22,337s.



Performance thread results: 15,449 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

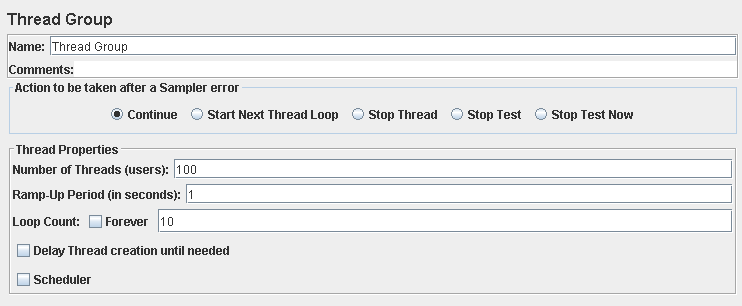
150 users and 10 loops: the application doesn’t have errors but the time of the delete action is high.

180 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

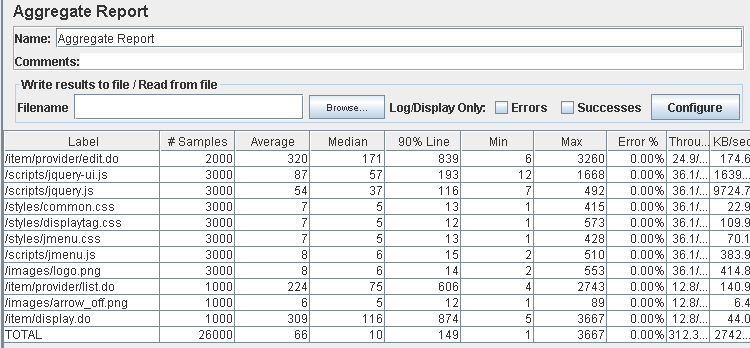
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 30:

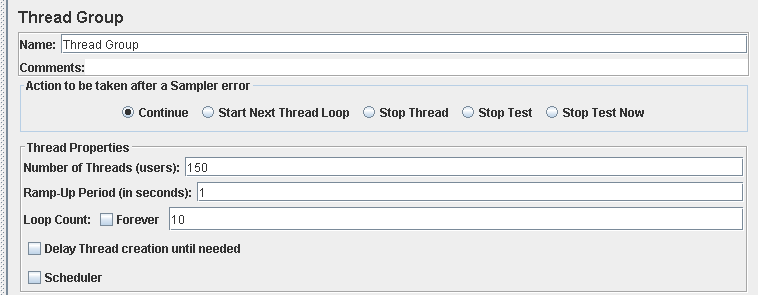
### As a provider I want to update and show my catalogue of items.



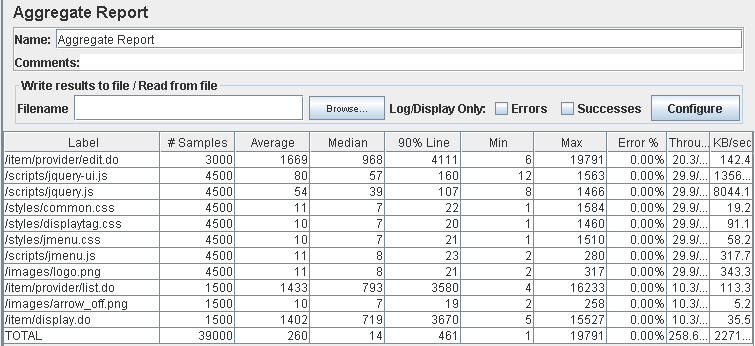
Performance test 90% results: Total 2,707s.



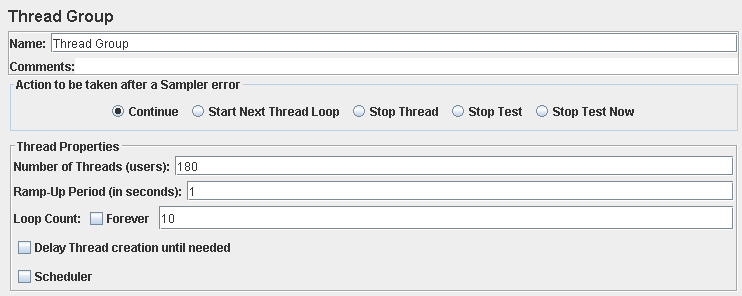
Performance thread results: 18,737 per minute.



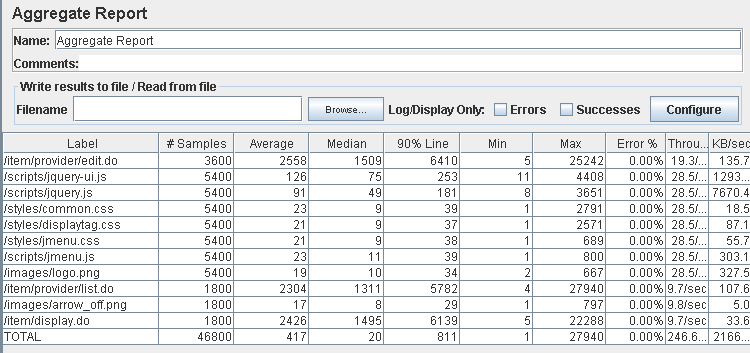
Performance test 90% results: Total 11,754s.



Performance thread results: 15,515 per minute.



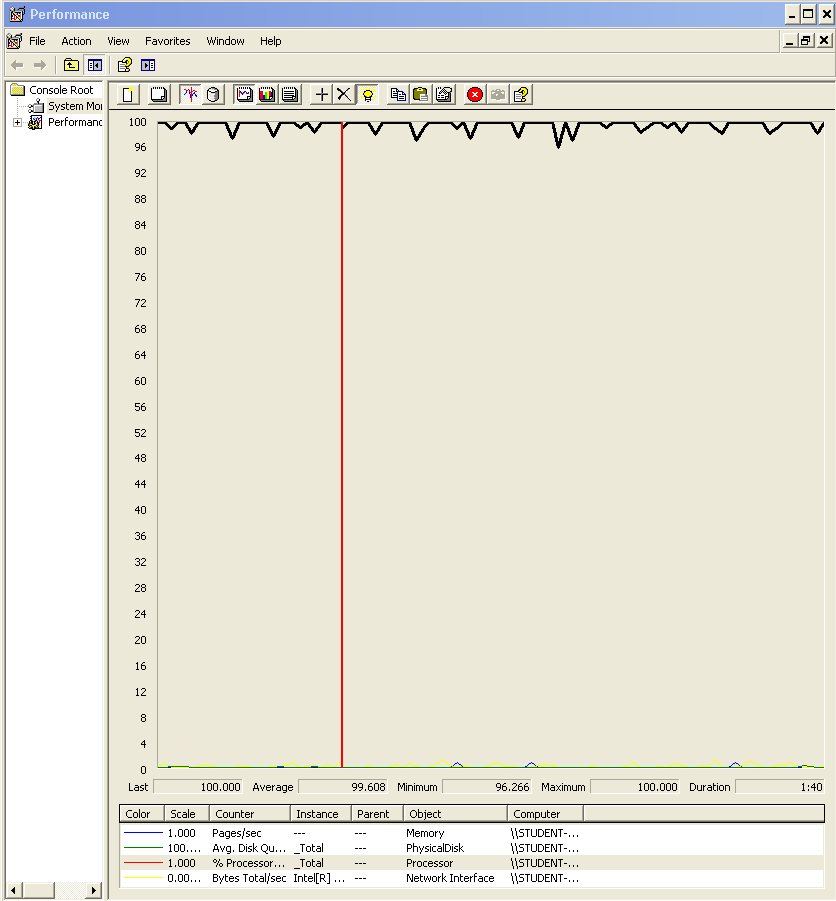
Performance test 90% results: Total 18,981s.



Performance thread results: 14,796 per minute.



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

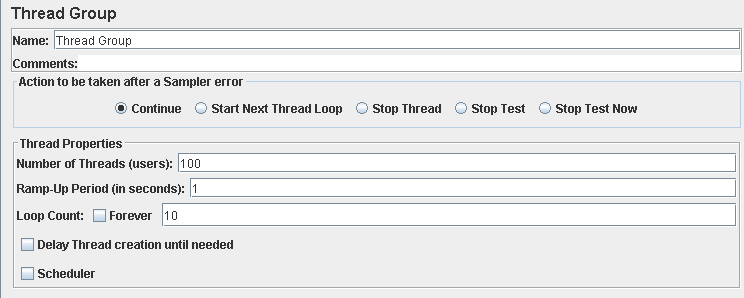
150 users and 10 loops: the application doesn’t have errors but the times are high.

180 users and 10 loops: the application doesn’t have errors but the times are incredibly high.

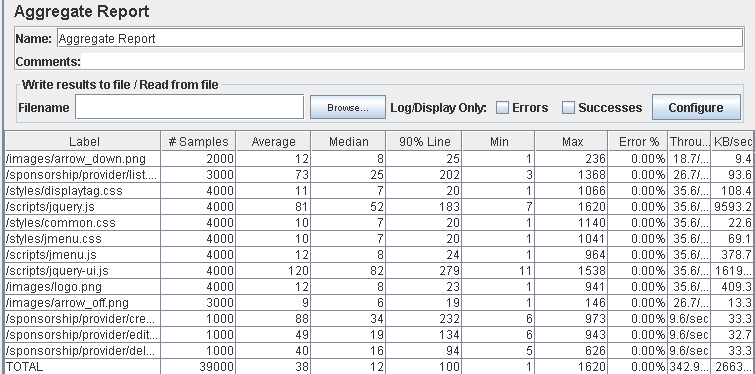
The computer performance analysis shows it could be a processors bottleneck problem.

# Use case 31:

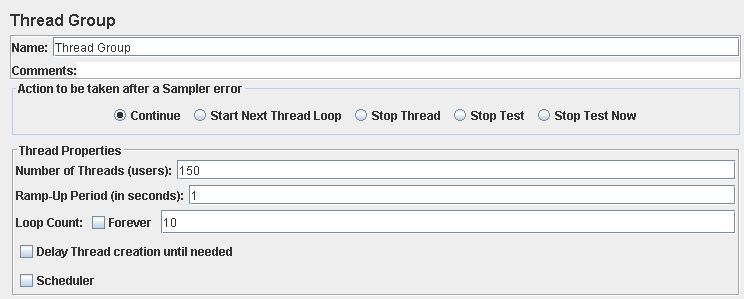
### As a provider I want to create and delete my sponsorships.



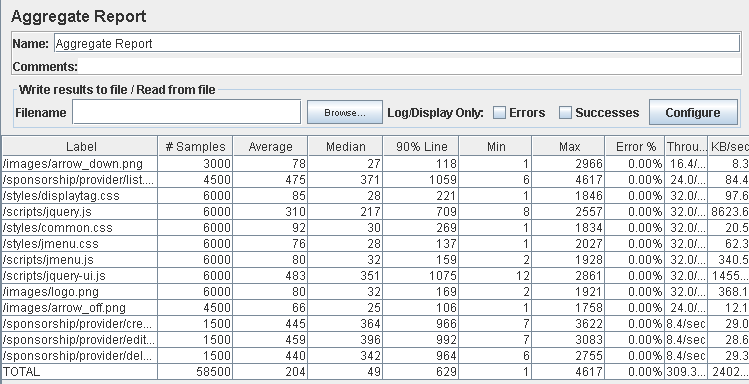
Performance test 90% results: Total 1,275s.



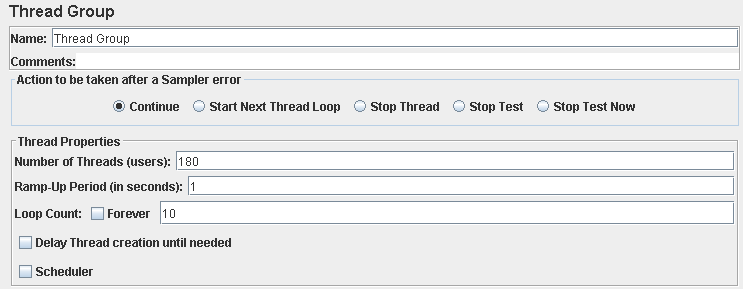
Performance thread results: 20,574 per minute.



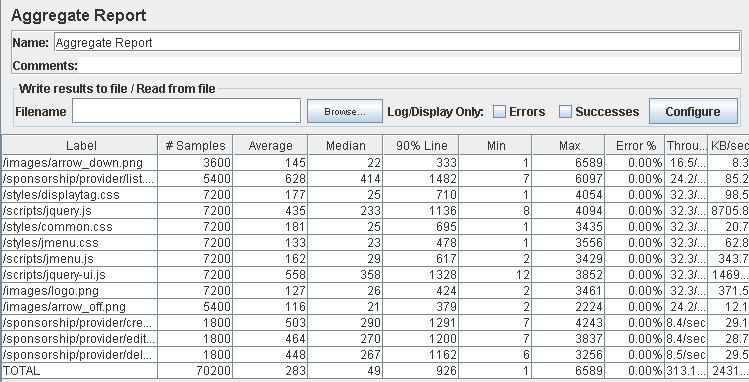
Performance test 90% results: Total 6,945s.



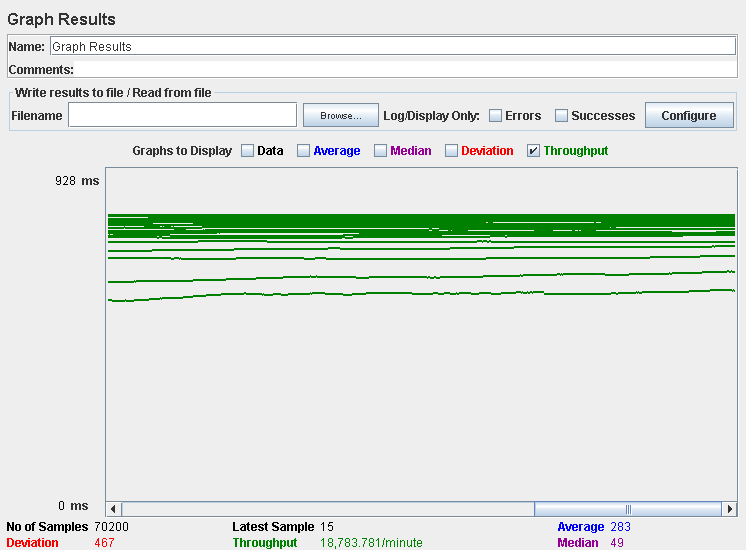
Performance thread results: 18,556 per minute.



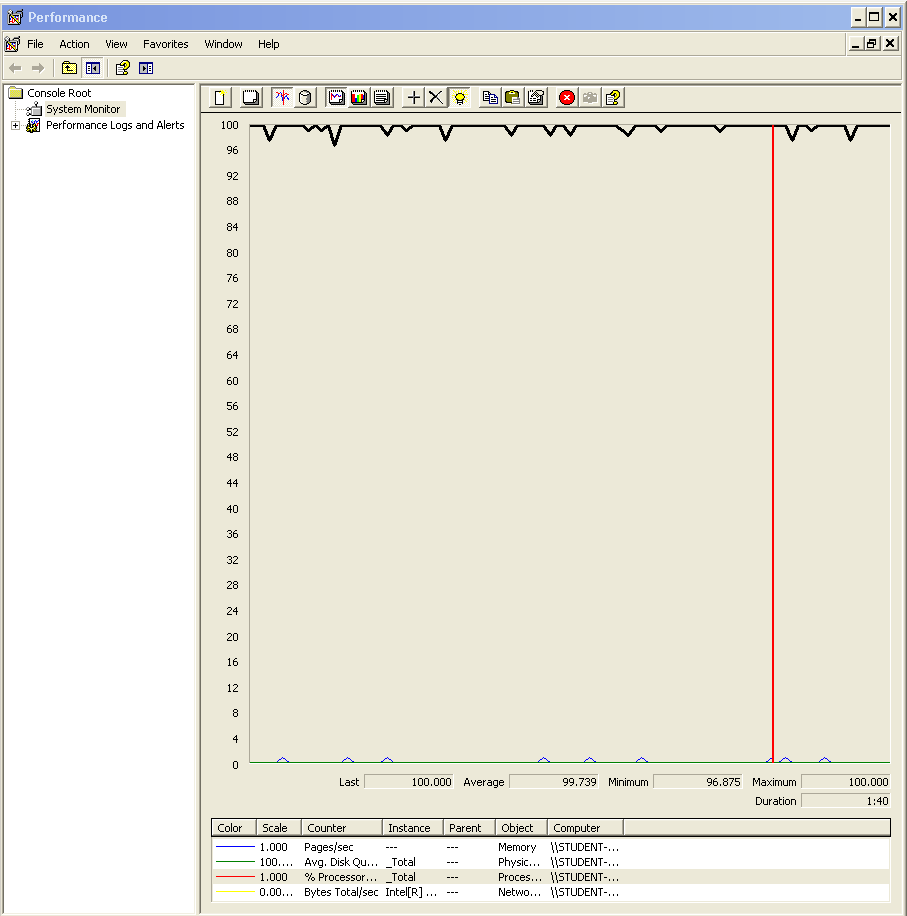
Performance test 90% results: Total 11,235s



Performance thread results: 18,783 per minute



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

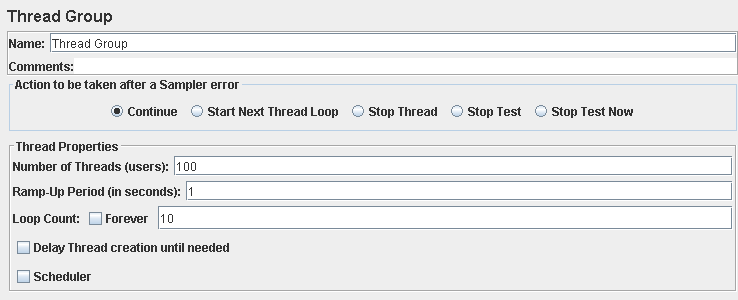
150 users and 10 loops: the application runs perfectly.

180 users and 10 loops: the application runs perfectly.

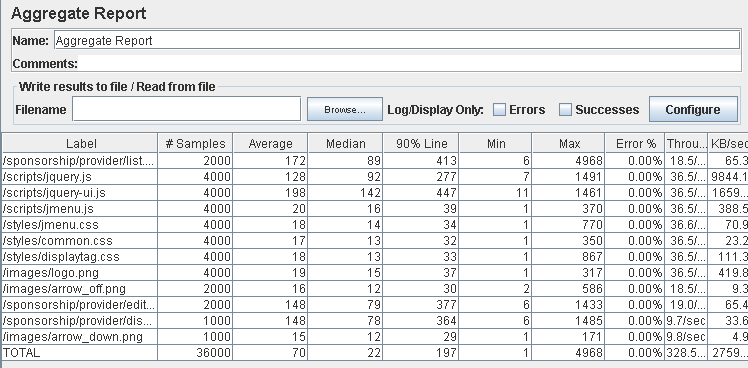
A test with more users doesn’t have sense since other test goes with errors with a larger number.

# Use case 32:

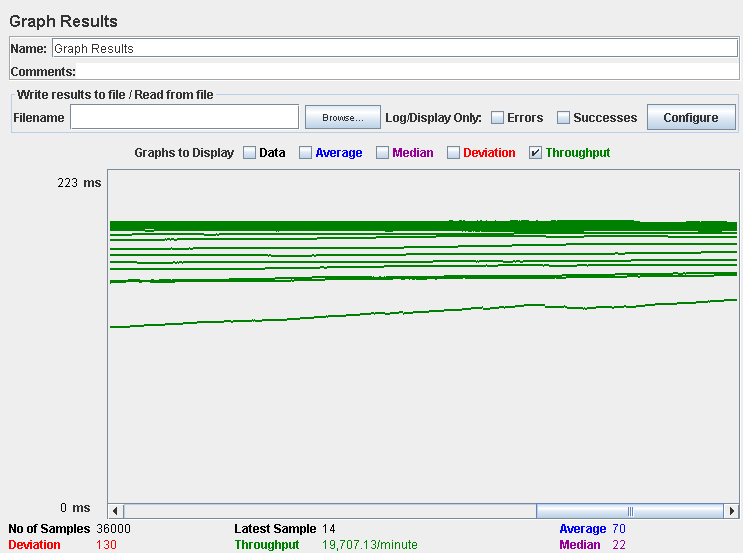
### As a provider I want to update and show my sponsorships.



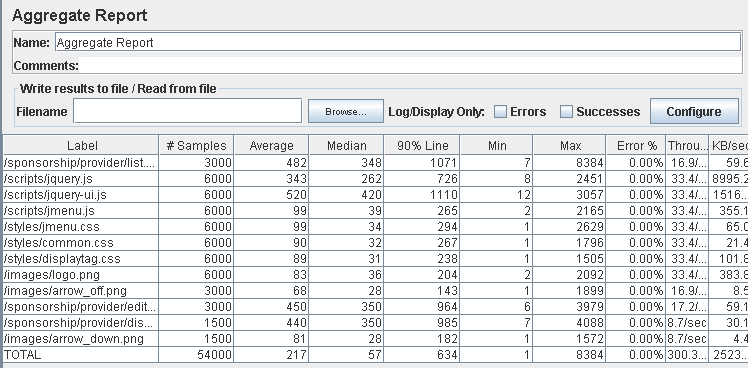
Performance test 90% results: Total 2,070s



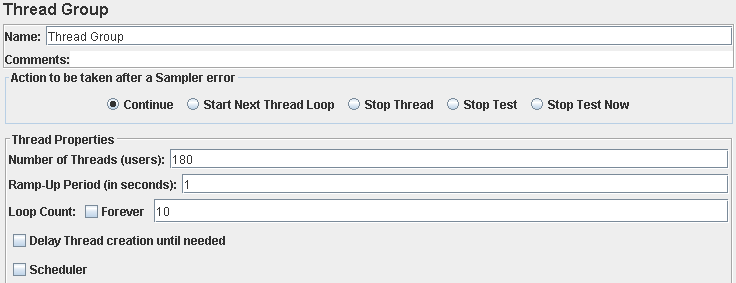
Performance thread results: 19,707 per minute



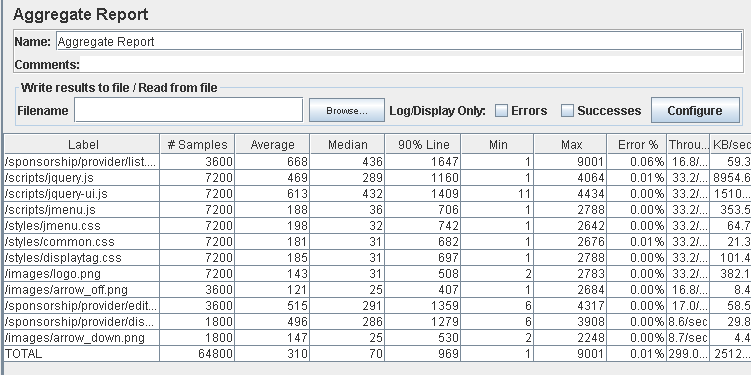
Performance test 90% results: Total 6,449s



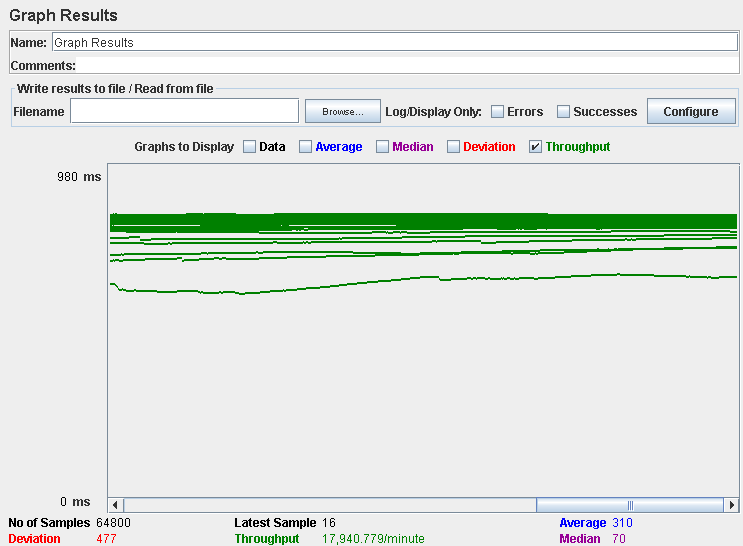
Performance thread results: 18,019 per minute



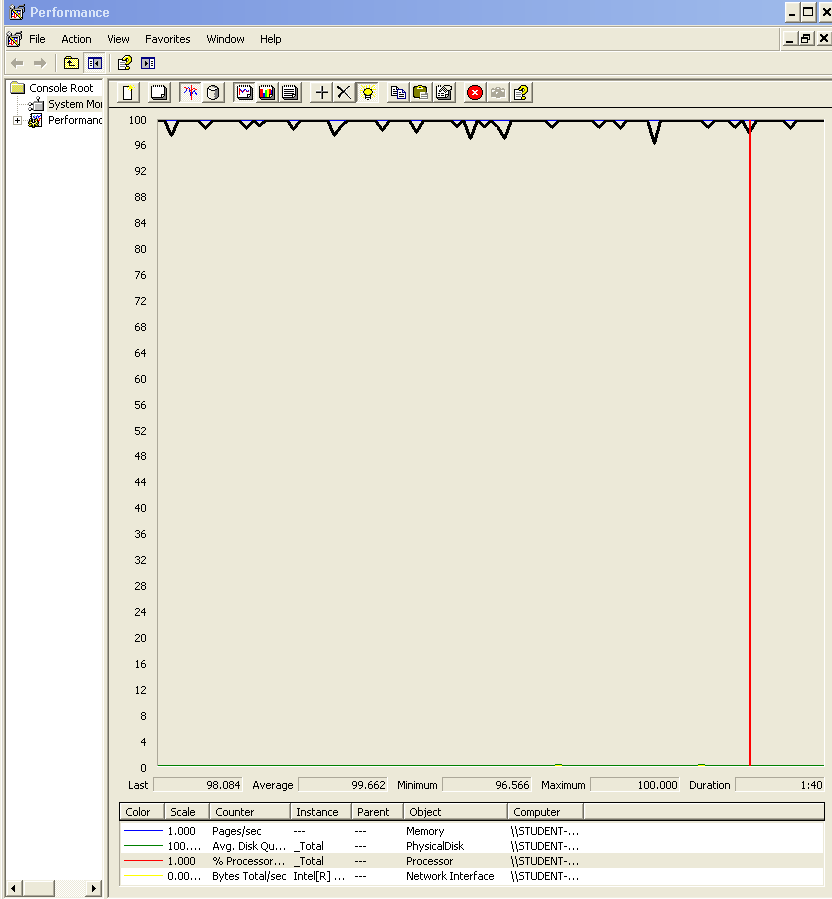
Performance test 90% results: Total 11,126s



Performance thread results: 17,940 per minute



Computer performance:



# Analysis results:

100 users and 10 loops: the application runs perfectly.

150 users and 10 loops: the application runs perfectly.

180 users and 10 loops: the application begins to have errors, the computer performance analysis shows it could be a processors bottleneck problem.

# Conclussion:

After the analysis of all the test realized, we can say that for 100 users and 10 loops all of our tests were successful, without any errors or excessive answer time.

It seems that from 150 users the system could encounter some errors due to the processor or memory (in most cases was the processor), either because the answer time was more than 3 seconds or because there wasn’t a success answer from the website.

We believe that with 125 users the system will answer correctly without errors or excessive answers times that could affect the user experience.

Changing the processor for a better one with more cores could mean an improvement in the number of concurrent users and after this change maybe a memory upgrade would be the way to go to continue increasing the allowed users.