

# Package co.edu.uniandes.admin

## Class Summary

[AdminServer](#)

[LoadTestSpecification](#)

---

co.edu.uniandes.admin

## Class AdminServer

```
java.lang.Object
|
+--co.edu.uniandes.admin.AdminServer
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class AdminServer
extends java.lang.Object
```

### Author:

Carlos Eduardo Gómez Montoya

### Author:

Cristian David Sierra Barrera

### Author:

Harold Enrique Castro Barrera

## Constructors

### AdminServer

```
public AdminServer()
    Constructor
```

## Methods

### closeConnection

```
public void closeConnection()
```

Close connections between the clients and the server. servidor. Primero se cierran los flujos y luego se cierra el socket.

## executeOneTest

```
public void executeOneTest(Scenario toTest,  
                           java.lang.String type,  
                           java.lang.String priority,  
                           int test)
```

Execute one test for the params given.

**Parameters:**

toTest - Scenario  
type - of load  
priority - to execute  
number - of test to execute

---

## executeTests

```
public void executeTests(Scenario toTest,  
                          java.lang.String type,  
                          java.lang.String priority)
```

Execute the scenario according the times indicated.

**Parameters:**

toTest - Scenario

---

## getMachines

```
public java.util.ArrayList getMachines()
```

---

## getNameBenchmark

```
public java.lang.String getNameBenchmark()
```

---

## getNameOutputFile

```
public java.lang.String getNameOutputFile()
```

---

## getNumberOfPrimes

```
public int getNumberOfPrimes()
```

---

## getParamBenchmark

```
public int getParamBenchmark()
```

---

## getPrefixNameMachines

```
public java.lang.String getPrefixNameMachines()
```

---

## getPriorities

```
public java.util.ArrayList getPriorities()
```

---

## getProfiles

```
public java.util.HashMap getProfiles()
```

---

## getScenarios

```
public java.util.ArrayList getScenarios()
```

---

## getTestsClient

```
public java.util.ArrayList getTestsClient()
```

---

## getTestsScenarios

```
public int getTestsScenarios()
```

## installVirtualMachines

```
public void installVirtualMachines()
```

Copy the client to Hypervisor and Clone based on the maximum scenario.

---

## load

```
public void load(java.lang.String name)
```

Load configuration of Tests

---

## main

```
public static void main(java.lang.String[] args)  
    throws java.lang.Exception
```

Main method, execute the server functions

**Throws:**

java.lang.Exception -

---

## run

```
public void run()
```

Execute the tests

---

## runByTests

```
public void runByTests()
```

Execute by test with all scenarios.

---

## setMachines

```
public void setMachines(java.util.ArrayList machines)
```

---

## setNameBenchmark

```
public void setNameBenchmark(java.lang.String nameBenchmark)
```

---

## setNameOutputFile

```
public void setNameOutputFile(java.lang.String nameOutputFile)
```

---

## setNumberOfPrimes

```
public void setNumberOfPrimes(int numberOfPrimes)
```

---

## setParamBenchmark

```
public void setParamBenchmark(int paramBenchmark)
```

---

## setPrefixNameMachines

```
public void setPrefixNameMachines(java.lang.String prefixNameMachines)
```

---

## setPriorities

```
public void setPriorities(java.util.ArrayList priorities)
```

---

## setProfiles

```
public void setProfiles(java.util.HashMap profiles)
```

---

## setScenarios

```
public void setScenarios(java.util.ArrayList scenarios)
```

---

## setTestsClient

```
public void setTestsClient(java.util.ArrayList testsClient)
```

---

## setTestsScenarios

```
public void setTestsScenarios(int testsScenarios)
```

---

## setupMachinesWithScenario

```
public void setupMachinesWithScenario(Scenario toTest)
```

Modify the virtual Machines according to Scenario

**Parameters:**

toTest -

---

## uninstallVirtualMachines

```
public void uninstallVirtualMachines()
```

Delete the clients from Hypervisor.

---

co.edu.uniandes.admin

# Class LoadTestSpecification

```
java.lang.Object
|
+--co.edu.uniandes.admin.LoadTestSpecification
```

---

< [Fields](#) > < [Constructors](#) > < [Methods](#) >

---

```
public class LoadTestSpecification
extends java.lang.Object
```

**Author:**

Carlos Eduardo Gómez Montoya

**Author:**

Cristian David Sierra Barrera

**Author:**

Harold Enrique Castro Barrera

## Fields

### TAG\_BENCHMARK

```
public static final java.lang.String TAG_BENCHMARK
```

---

### TAG\_CLIENT\_TESTS

```
public static final java.lang.String TAG_CLIENT_TESTS
```

---

### TAG\_CORES

```
public static final java.lang.String TAG_CORES
```

---

### TAG\_MACHINE\_NAME\_PREFIX

```
public static final java.lang.String TAG_MACHINE_NAME_PREFIX
```

---

### TAG\_NAME

```
public static final java.lang.String TAG_NAME
```

---

### TAG\_NAME\_PROFILE

```
public static final java.lang.String TAG_NAME_PROFILE
```

---

### TAG\_NUMBER\_TEST

```
public static final java.lang.String TAG_NUMBER_TEST
```

---

### TAG\_OUTPUT\_FILE

```
public static final java.lang.String TAG_OUTPUT_FILE
```

---

### TAG\_PRIORITIES

```
public static final java.lang.String TAG_PRIORITIES
```

---

## TAG\_PROFILES

```
public static final java.lang.String TAG_PROFILES
```

---

## TAG\_RAM

```
public static final java.lang.String TAG_RAM
```

---

## TAG\_SCENARIOS

```
public static final java.lang.String TAG_SCENARIOS
```

## Constructors

### LoadTestSpecification

```
public LoadTestSpecification()
```

## Methods

### loadConfiguration

```
public static void loadConfiguration(AdminServer admin,  
                                     java.lang.String nameFile)
```

Method to load information about Execution Tests

**Parameters:**

scenariosListToLoad - List of Scenarios

hardwareProfilesListToLoad - List of Hardware Profiles

**Returns:**

filename output of tests.



# Package co.edu.uniandes.benchmarks

## Class Summary

### [Benchmarking](#)

co.edu.uniandes.benchmarks

## Class Benchmarking

```
java.lang.Object
|
+--co.edu.uniandes.benchmarks.Benchmarking
```

< [Constructors](#) > < [Methods](#) >

```
public class Benchmarking
extends java.lang.Object
```

#### Author:

Carlos Eduardo Gómez Montoya

#### Author:

Cristian David Sierra Barrera

#### Author:

Harold Enrique Castro Barrera

## Constructors

### Benchmarking

```
public Benchmarking()
```

## Methods

### executeBenchmarking

```
public static void executeBenchmarking(java.lang.String name,
                                         int param)
```

Execute the type of benchmarking in the test

#### Parameters:

name - of benchmark  
param - of benchmark

---

## primeCalculatorIntegers

```
public static void primeCalculatorIntegers(int n)
```

Calculator n primes.

**Parameters:**

n - number of primes to find Algorithm used from

<http://beginnersbook.com/2014/01/java-program-to-display-first-n-or-first-100-prime-numbers/>

# Package co.edu.uniandes.conf

## Class Summary

### [Configuration](#)

co.edu.uniandes.conf

## Class Configuration

```
java.lang.Object
|
+--co.edu.uniandes.conf.Configuration
```

< [Constructors](#) > < [Methods](#) >

```
public class Configuration
    extends java.lang.Object
```

#### Author:

Carlos Eduardo Gómez Montoya

#### Author:

Cristian David Sierra Barrera

#### Author:

Harold Enrique Castro Barrera

## Constructors

### Configuration

```
public Configuration(java.lang.String filename)
```

## Methods

### getLabelLogFile

```
public java.lang.String getLabelLogFile()
```

This method returns the label of each entry in the log file.

#### Returns:

String The label.

## getLogFileName

```
public java.lang.String getLogFileName()
```

This method returns the log file name.

**Returns:**

String The log file name.

---

## getPathLog

```
public java.lang.String getPathLog()
```

This method returns the directory path where the log file will be stored.

**Returns:**

String The pathLog.

---

## getPort

```
public int getPort()
```

This method returns the port number.

**Returns:**

int The port number.

---

## getVirtualBoxHome

```
public java.lang.String getVirtualBoxHome()
```

This method returns the directory path where VirtualBox installed is.

**Returns:**

String The directory path of VirtualBox.

---

## getVirtualMachinesHome

```
public java.lang.String getVirtualMachinesHome()
```

This method returns the directory path where the virtual machines stored are.

**Returns:**

String The directory path of the virtual machines repository.

# Package co.edu.uniandes.model

## Class Summary

### [HardwareProfile](#)

Class to represent a Hardware Profile

### [Scenario](#)

Class to represent a Scenario

---

co.edu.uniandes.model

## Class HardwareProfile

```
java.lang.Object
|
+--co.edu.uniandes.model.HardwareProfile
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class HardwareProfile
extends java.lang.Object
```

Class to represent a Hardware Profile

**Author:**

Carlos Eduardo Gómez Montoya

**Author:**

Cristian David Sierra Barrera

**Author:**

Harold Enrique Castro Barrera

## Constructors

### HardwareProfile

```
public HardwareProfile(java.lang.String name,
                        int RAM,
                        int cores)
```

Constructor

**Parameters:**

RAM - in the Hardware Profile

cores - in the Hardware Profile

## Methods

### getCores

```
public int getCores()
```

---

### getRAM

```
public int getRAM()
```

---

### toString

```
public java.lang.String toString()
```

**Overrides:**

toString in class java.lang.Object

---

co.edu.uniandes.model

## Class Scenario

```
java.lang.Object  
|  
+--co.edu.uniandes.model.Scenario
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class Scenario  
extends java.lang.Object
```

Class to represent a Scenario

**Author:**

Carlos Eduardo Gómez Montoya

**Author:**

Cristian David Sierra Barrera

**Author:**

Harold Enrique Castro Barrera

## Constructors

## Scenario

```
public Scenario(java.util.ArrayList nProfiles,  
                int nNumberTests)
```

Constructor

**Parameters:**

nTest - Number of test to execute this scenario.

nProfiles - Profiles of the Scenario

## Methods

### averageTime

```
public double averageTime()
```

Average time of Scenario

**Returns:**

---

### getNumberTests

```
public int getNumberTests()
```

---

### getProfiles

```
public java.util.ArrayList getProfiles()
```

---

### getSizeScenario

```
public int getSizeScenario()
```

Return the number of machines in the scenario

**Returns:**

number of profiles

---

## getTimeByTest

```
public double getTimeByTest(int t)
```

Return the time in the t position

**Parameters:**

t - position or t test

**Returns:**

Time of that Test

---

## getTimes

```
public double[] getTimes()
```

---

## setNumberTests

```
public void setNumberTests(int numberTests)
```

---

## setTime

```
public void setTime(double[] times)
```

---

## setTimebyTest

```
public void setTimebyTest(long startTime,  
                           long endTime,  
                           int test)
```

Set time in the array according to the test executed.

---

## toString

```
public java.lang.String toString()
```

**Overrides:**

toString in class java.lang.Object



# Package co.edu.uniandes.resources

## Class Summary

### [VirtualBoxVM](#)

Class to represent a VirtualBox Virtual Machine

### [VirtualMachine](#)

Class to represent a Virtual Machine

---

co.edu.uniandes.resources

## Class VirtualBoxVM

```
java.lang.Object
|
+--VirtualMachine
    |
    +--co.edu.uniandes.resources.VirtualBoxVM
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class VirtualBoxVM
extends VirtualMachine
```

Class to represent a VirtualBox Virtual Machine

**Author:**

Carlos Eduardo Gómez Montoya

**Author:**

Cristian David Sierra Barrera

**Author:**

Harold Enrique Castro Barrera

## Constructors

### VirtualBoxVM

```
public VirtualBoxVM(java.lang.String name,
                    int nTests,
                    Configuration conf)
```

## Methods

## registerVirtualMachine

```
public void registerVirtualMachine()
```

---

## unregisterVirtualMachine

```
public void unregisterVirtualMachine()
```

---

## vmModifyHardwareProfile

```
public void vmModifyHardwareProfile()
```

Modify the Virtual Machine with the Hardware Profile

**Overrides:**

[vmModifyHardwareProfile](#) in class [VirtualMachine](#)

---

## vmTurnOff

```
public void vmTurnOff()
```

Turn off the Virtual Machine

**Overrides:**

[vmTurnOff](#) in class [VirtualMachine](#)

---

## vmTurnOn

```
public void vmTurnOn()
```

Turn on the Virtual Machine

**Overrides:**

[vmTurnOn](#) in class [VirtualMachine](#)

---

co.edu.uniandes.resources

# Class VirtualMachine

```
java.lang.Object
|
+--co.edu.uniandes.resources.VirtualMachine
```

## Direct Known Subclasses:

[VirtualBoxVM](#)

---

< [Constructors](#) > < [Methods](#) >

---

public abstract class **VirtualMachine**  
extends java.lang.Object

Class to represent a Virtual Machine

### Author:

Carlos Eduardo Gómez Montoya

### Author:

Cristian David Sierra Barrera

### Author:

Harold Enrique Castro Barrera

## Constructors

### VirtualMachine

```
public  VirtualMachine(java.lang.String name,  
                        int nTests)
```

Constructor

#### Parameters:

name - of Virtual Machine

## Methods

### getName

```
public java.lang.String getName()
```

---

## getProfile

```
public HardwareProfile getProfile()
```

---

## getTimeByTest

```
public double getTimeByTest(int test)
```

---

## getTimesSignUp

```
public double[] getTimesSignUp()
```

---

## reinitialize

```
public void reinitialize()
```

---

## setName

```
public void setName(java.lang.String name)
```

---

## setProfile

```
public void setProfile(HardwareProfile profile)
```

---

## setTimeSignUpByTest

```
public void setTimeSignUpByTest(long startTime,  
                                long endTime,  
                                int test)
```

---

## setTimesSignUp

```
public void setTimesSignUp(double[] timesSignUp)
```

---

## vmModifyHardwareProfile

```
public abstract void vmModifyHardwareProfile()
```

Modify the Virtual Machine with the Actual Profile

---

## vmTurnOff

```
public abstract void vmTurnOff()
```

Turn off the Virtual Machine

---

## vmTurnOn

```
public abstract void vmTurnOn()
```

Turn on the Virtual Machine

# Package co.edu.uniandes.util

## Class Summary

### [Constants](#)

Constants

### [LoggerUtil](#)

### [Services](#)

Services in the Server

---

co.edu.uniandes.util

## Class Constants

```
java.lang.Object
|
+--co.edu.uniandes.util.Constants
```

---

< [Fields](#) > < [Constructors](#) >

---

```
public class Constants
extends java.lang.Object
```

Constants

#### Author:

Carlos Eduardo Gómez Montoya

#### Author:

Cristian David Sierra Barrera

#### Author:

Harold Enrique Castro Barrera

## Fields

### CSV

```
public static final java.lang.String CSV
```

---

### MILLON

```
public static final double MILLON
```

---

## PROPERTIES\_FILENAME

```
public static final java.lang.String PROPERTIES_FILENAME
```

---

## SEPARATOR\_BENCHMARKING

```
public static final java.lang.String SEPARATOR_BENCHMARKING
```

---

## SPACE

```
public static final java.lang.String SPACE
```

---

## TIME\_STABILIZATION

```
public static final int TIME_STABILIZATION
```

---

## TYPE\_FILES

```
public static final java.lang.String TYPE_FILES
```

---

## VBOX

```
public static final java.lang.String VBOX
```

## Constructors

### Constants

```
public Constants()
```

---

```
co.edu.uniandes.util
```

## Class LoggerUtil

```
java.lang.Object
|
+--co.edu.uniandes.util.LoggerUtil
```

---

---

```
public class LoggerUtil
extends java.lang.Object
```

## Constructors

### LoggerUtil

```
public LoggerUtil()
```

## Methods

### getLoggerInfo

```
public static org.apache.log4j.Logger getLoggerInfo( java.lang.String
classname,
                                                    java.lang.String
directoryPath,
                                                    java.lang.String name)
                                                    throws java.io.IOException
```

This method return a logger to write in an info file and the console.

**Parameters:**

classname - It is the name of the class that appears in each entry of the log file.  
directoryPath - It is the path of the directory where the log file will be stored.  
name - It is the name of the log file.

**Returns:**

Logger It is the configured logger to use in any part of the source code.

**Throws:**

java.io.IOException -

---

co.edu.uniandes.util

## Class Services

```
java.lang.Object
|
+--co.edu.uniandes.util.Services
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class Services
extends java.lang.Object
```

Services in the Server



**Author:**

Carlos Eduardo Gómez Montoya

**Author:**

Cristian David Sierra Barrera

**Author:**

Harold Enrique Castro Barrera

## Constructors

## Services

```
public Services()
```

## Methods

### execute

```
public static void execute(java.lang.String command)
```

Execute terminal command in Host

**Parameters:**

command -

---

### pause

```
public static void pause(long n)
```

Pause n seconds the execution.

**Parameters:**

n - seconds.

# INDEX

## A

[averageTime](#) ... 15  
[AdminServer](#) ... 1  
[AdminServer](#) ... 1

## B

[Benchmarking](#) ... 9  
[Benchmarking](#) ... 9

## C

[closeConnection](#) ... 1  
[Configuration](#) ... 11  
[Configuration](#) ... 11  
[Constants](#) ... 22  
[Constants](#) ... 23  
[CSV](#) ... 22

## E

[execute](#) ... 25  
[executeBenchmarking](#) ... 9  
[executeOneTest](#) ... 2  
[executeTests](#) ... 2

## G

[getCores](#) ... 14  
[getLabelLogFile](#) ... 11  
[getLogFileName](#) ... 12  
[getLoggerInfo](#) ... 24  
[getMachines](#) ... 2  
[getName](#) ... 19  
[getNameBenchmark](#) ... 2  
[getNameOutputFile](#) ... 2  
[getNumberOfPrimes](#) ... 3  
[getNumberTests](#) ... 15  
[getParamBenchmark](#) ... 3  
[getPathLog](#) ... 12  
[getPort](#) ... 12  
[getPrefixNameMachines](#) ... 3  
[getPriorities](#) ... 3  
[getProfile](#) ... 20  
[getProfiles](#) ... 3  
[getProfiles](#) ... 15  
[getRAM](#) ... 14  
[getScenarios](#) ... 3  
[getSizeScenario](#) ... 15  
[getTestsClient](#) ... 3  
[getTestsScenarios](#) ... 3  
[getTimeByTest](#) ... 16  
[getTimeByTest](#) ... 20  
[getTimes](#) ... 16  
[getTimesSignUp](#) ... 20  
[getVirtualBoxHome](#) ... 12  
[getVirtualMachinesHome](#) ... 12

## H

[HardwareProfile](#) ... 13  
[HardwareProfile](#) ... 13

## I

[installVirtualMachines](#) ... 4

## L

[load](#) ... 4  
[loadConfiguration](#) ... 8  
[LoadTestSpecification](#) ... 6  
[LoadTestSpecification](#) ... 8  
[LoggerUtil](#) ... 23  
[LoggerUtil](#) ... 24

## M

[main](#) ... 4  
[MILLON](#) ... 22

## P

[pause](#) ... 25  
[primeCalculatorIntegers](#) ... 10  
[PROPERTIES\\_FILENAME](#) ... 23

## R

[registerVirtualMachine](#) ... 18  
[reinitialize](#) ... 20  
[run](#) ... 4  
[runByTests](#) ... 4

## S

[setMachines](#) ... 4  
[setName](#) ... 20  
[setNameBenchmark](#) ... 5  
[setNameOutputFile](#) ... 5  
[setNumberOfPrimes](#) ... 5  
[setNumberTests](#) ... 16  
[setParamBenchmark](#) ... 5  
[setPrefixNameMachines](#) ... 5  
[setPriorities](#) ... 5  
[setProfile](#) ... 20  
[setProfiles](#) ... 5  
[setScenarios](#) ... 5  
[setTestsClient](#) ... 6  
[setTestsScenarios](#) ... 6  
[setTime](#) ... 16  
[setTimebyTest](#) ... 16  
[setTimeSignUpByTest](#) ... 20  
[setTimesSignUp](#) ... 21  
[setupMachinesWithScenario](#) ... 6  
[Scenario](#) ... 14  
[Scenario](#) ... 15  
[SEPARATOR\\_BENCHMARKING](#) ... 23  
[Services](#) ... 24  
[Services](#) ... 25  
[SPACE](#) ... 23

## T

[toString](#) ... 14  
[toString](#) ... 16  
[TAG BENCHMARK](#) ... 7  
[TAG CLIENT TESTS](#) ... 7  
[TAG CORES](#) ... 7  
[TAG MACHINE NAME PREFIX](#) ... 7  
[TAG NAME](#) ... 7  
[TAG NAME PROFILE](#) ... 7  
[TAG NUMBER TEST](#) ... 7  
[TAG OUTPUT FILE](#) ... 7  
[TAG PRIORITIES](#) ... 7  
[TAG PROFILES](#) ... 8  
[TAG RAM](#) ... 8  
[TAG SCENARIOS](#) ... 8  
[TIME STABILIZATION](#) ... 23  
[TYPE FILES](#) ... 23

## U

[uninstallVirtualMachines](#) ... 6  
[unregisterVirtualMachine](#) ... 18

## V

[vmModifyHardwareProfile](#) ... 18  
[vmModifyHardwareProfile](#) ... 21  
[vmTurnOff](#) ... 18  
[vmTurnOff](#) ... 21  
[vmTurnOn](#) ... 18  
[vmTurnOn](#) ... 21  
[VBOX](#) ... 23  
[VirtualBoxVM](#) ... 17  
[VirtualBoxVM](#) ... 17  
[VirtualMachine](#) ... 19  
[VirtualMachine](#) ... 19