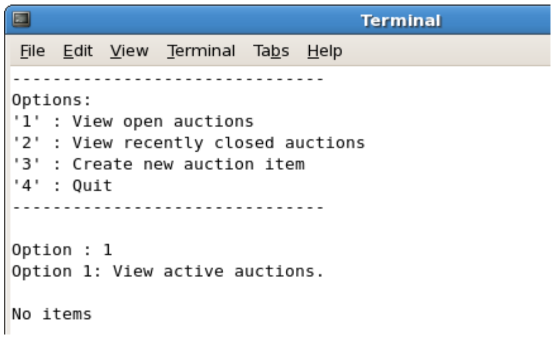
DAS4 Assessed Exercise
0603112

1 Overall Design

1.1 ClientUI

Usage: To Run the ClientUI:
- Open 3 terminals and locate to the src folder of the Project
- In one terminal enter 'rmi registry'
- In another, enter 'make' to start the Server
- In another, enter 'make run_client' to start the ClientUI

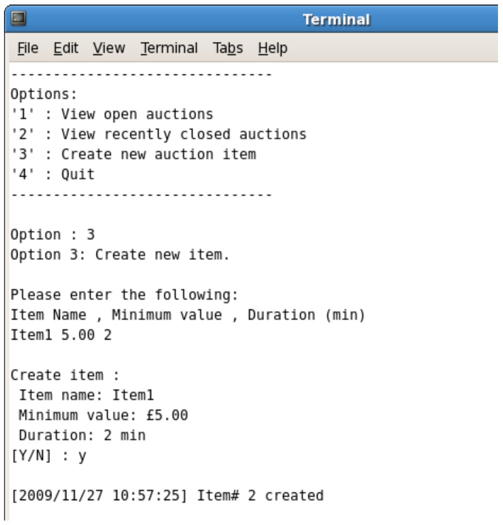


```
------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option : 1
Option 1: View active auctions.

No items
```

Illustration 1: ClientUI

The Client user interface prompts the user for an option. The interface allows the user to View active auctions, View inactive auctions, Create an auction. If there are currently auctions open, it allows the user to bid on an auction.

1.2 Creating an Auction

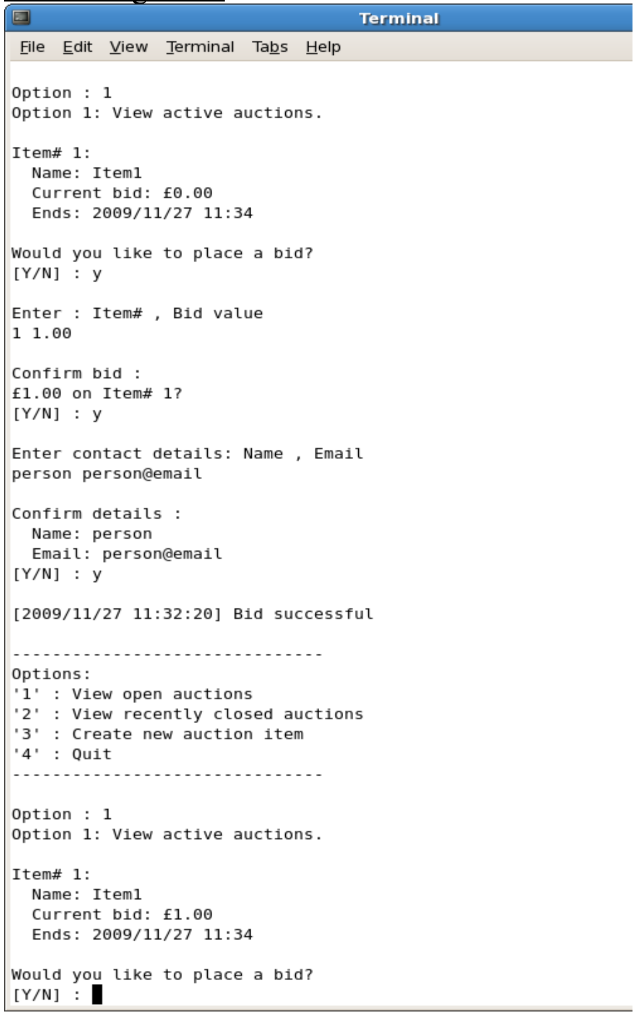


```
------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option : 3
Option 3: Create new item.

Please enter the following:
Item Name , Minimum value , Duration (min)
Item1 5.00 2

Create item :
 Item name: Item1
 Minimum value: £5.00
 Duration: 2 min
[Y/N] : y

[2009/11/27 10:57:25] Item# 2 created
```

Illustration 2: CreateAuctionUI

To create an auction, the user must enter an item name, a minimum acceptable value and a duration in minutes. A full date/time was initially to be entered, but this required the user to enter many variables that seemed unnecessary for a test interface. The date and time may be changed in the user interface if it were to be used for auctions lasting more than a number of minutes.

1.3 Placing a Bid

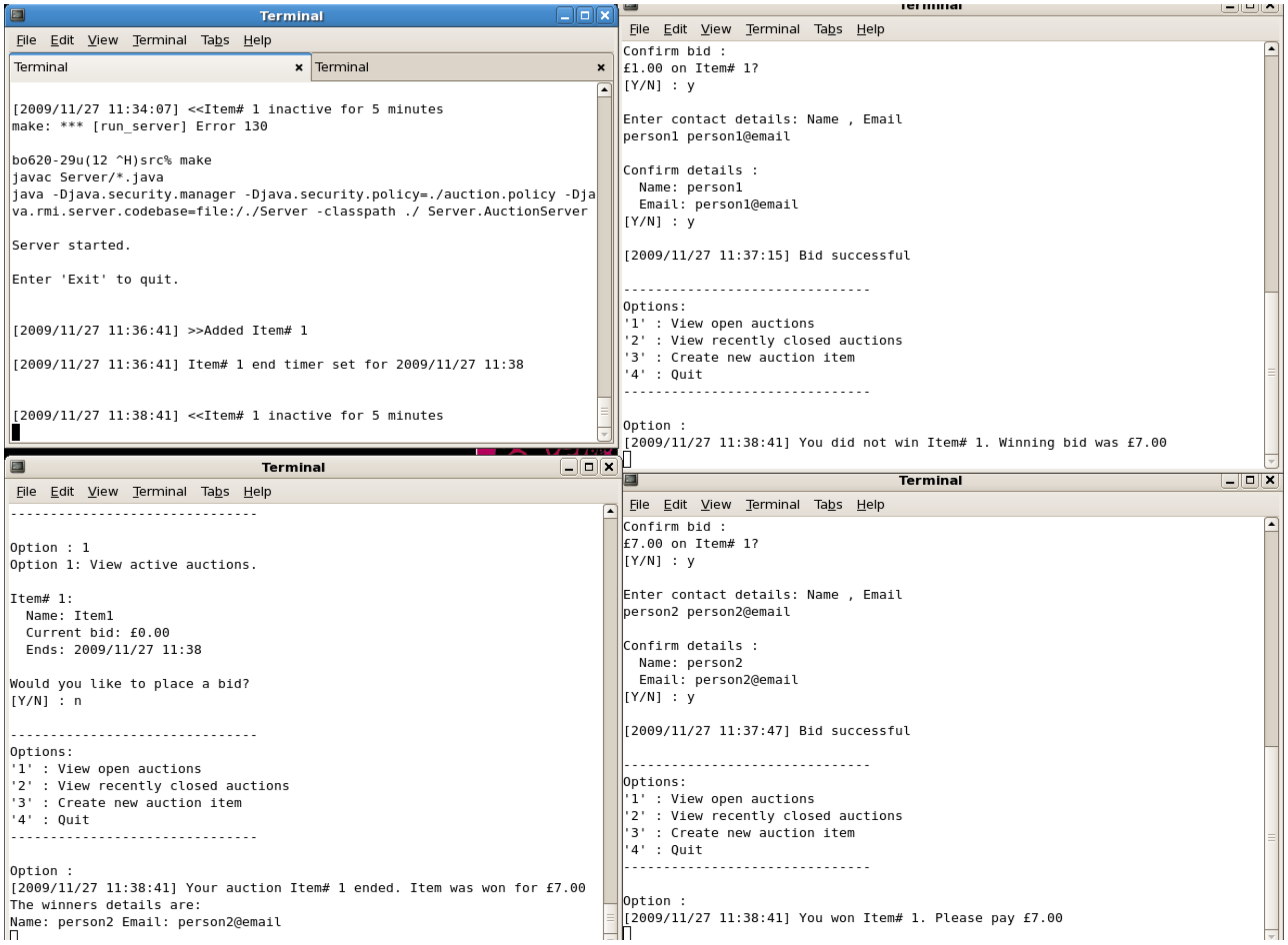


```
Option : 1
Option 1: View active auctions.

Item# 1:
  Name: Item1
  Current bid: £0.00
  Ends: 2009/11/27 11:34

Would you like to place a bid?
[Y/N] : y

Enter : Item# , Bid value
1 1.00

Confirm bid :
£1.00 on Item# 1?
[Y/N] : y

Enter contact details: Name , Email
person person@email

Confirm details :
  Name: person
  Email: person@email
[Y/N] : y

[2009/11/27 11:32:20] Bid successful

-------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
-------------------------------

Option : 1
Option 1: View active auctions.

Item# 1:
  Name: Item1
  Current bid: £1.00
  Ends: 2009/11/27 11:34

Would you like to place a bid?
[Y/N] :
```

Illustration 3: PlaceBidUI

To place a bid, the user must first check if there are any auctions to bid on. If so, they are given instructions on creating the bid. The required information to place a bid is the item number of the item they wish to bid on, the value of their bid, plus the users name and email address. If the bid was successful, they may view the bid in the active auction list. A bid is unsuccessful if the bid parameters were invalid or the user was outbidded before completing their bid request.

1.4 Callbacks



```
[2009/11/27 11:34:07] <<Item# 1 inactive for 5 minutes
make: *** [run_server] Error 130

bo620-29u(12 ^H)src% make
javac Server/*.java
java -Djava.security.manager -Djava.security.policy=./auction.policy -Dja
va.rmi.server.codebase=file:/./Server -classpath ./ Server.AuctionServer

Server started.

Enter 'Exit' to quit.


[2009/11/27 11:36:41] >>Added Item# 1

[2009/11/27 11:36:41] Item# 1 end timer set for 2009/11/27 11:38


[2009/11/27 11:38:41] <<Item# 1 inactive for 5 minutes
```

```
Confirm bid :
£1.00 on Item# 1?
[Y/N] : y

Enter contact details: Name , Email
person1 person1@email

Confirm details :
  Name: person1
  Email: person1@email
[Y/N] : y

[2009/11/27 11:37:15] Bid successful

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:38:41] You did not win Item# 1. Winning bid was £7.00
```

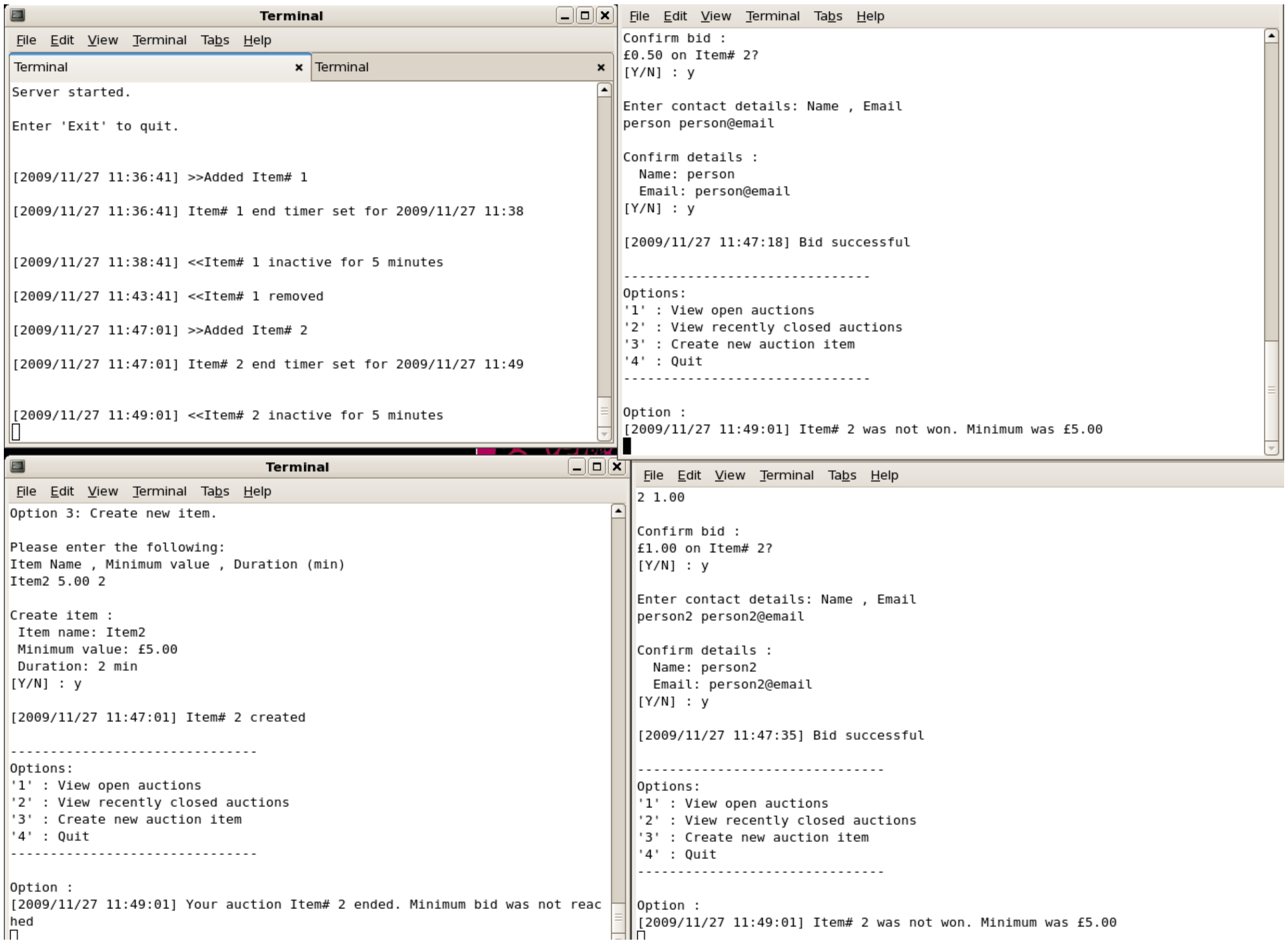
```
------------------------------

Option : 1
Option 1: View active auctions.

Item# 1:
  Name: Item1
  Current bid: £0.00
  Ends: 2009/11/27 11:38

Would you like to place a bid?
[Y/N] : n

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:38:41] Your auction Item# 1 ended. Item was won for £7.00
The winners details are:
Name: person2 Email: person2@email
```

```
Confirm bid :
£7.00 on Item# 1?
[Y/N] : y

Enter contact details: Name , Email
person2 person2@email

Confirm details :
  Name: person2
  Email: person2@email
[Y/N] : y

[2009/11/27 11:37:47] Bid successful

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:38:41] You won Item# 1. Please pay £7.00
```

Illustration 4: Callbacks – Top left:Server, Bottom left:Auctioneer, Top right:Losing Bid, Bottom right: Winning Bid

If a client bids on an item, then they are notified about the end result. They may be notified that they won the item; that they did not win the item; or that no one one the item, in which case the minimum value of the item is revealed.

An auctioneer is notified about the end result of their item – either that it was won, in which case they are told the winners contact details; or that the minimum value was not reached.



```
Server started.

Enter 'Exit' to quit.


[2009/11/27 11:36:41] >>Added Item# 1

[2009/11/27 11:36:41] Item# 1 end timer set for 2009/11/27 11:38


[2009/11/27 11:38:41] <<Item# 1 inactive for 5 minutes

[2009/11/27 11:43:41] <<Item# 1 removed

[2009/11/27 11:47:01] >>Added Item# 2

[2009/11/27 11:47:01] Item# 2 end timer set for 2009/11/27 11:49


[2009/11/27 11:49:01] <<Item# 2 inactive for 5 minutes
```

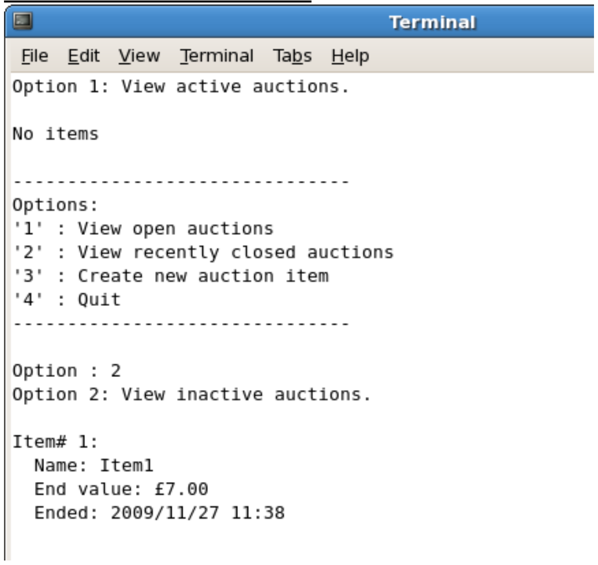
```
Confirm bid :
£0.50 on Item# 2?
[Y/N] : y

Enter contact details: Name , Email
person person@email

Confirm details :
  Name: person
  Email: person@email
[Y/N] : y

[2009/11/27 11:47:18] Bid successful

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:49:01] Item# 2 was not won. Minimum was £5.00
```

```
Option 3: Create new item.

Please enter the following:
Item Name , Minimum value , Duration (min)
Item2 5.00 2

Create item :
 Item name: Item2
 Minimum value: £5.00
 Duration: 2 min
[Y/N] : y

[2009/11/27 11:47:01] Item# 2 created

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:49:01] Your auction Item# 2 ended. Minimum bid was not reac
hed
```

```
2 1.00

Confirm bid :
£1.00 on Item# 2?
[Y/N] : y

Enter contact details: Name , Email
person2 person2@email

Confirm details :
  Name: person2
  Email: person2@email
[Y/N] : y

[2009/11/27 11:47:35] Bid successful

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option :
[2009/11/27 11:49:01] Item# 2 was not won. Minimum was £5.00
```

Illustration 5: Callbacks - Minimum not reached Top left:Server, Bottom left:Auctioneer, Top/Bottom right:Losing Bids

1.5 View Inactive Items



```
Option 1: View active auctions.

No items

-------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
-------------------------------

Option : 2
Option 2: View inactive auctions.

Item# 1:
  Name: Item1
  End value: £7.00
  Ended: 2009/11/27 11:38
```

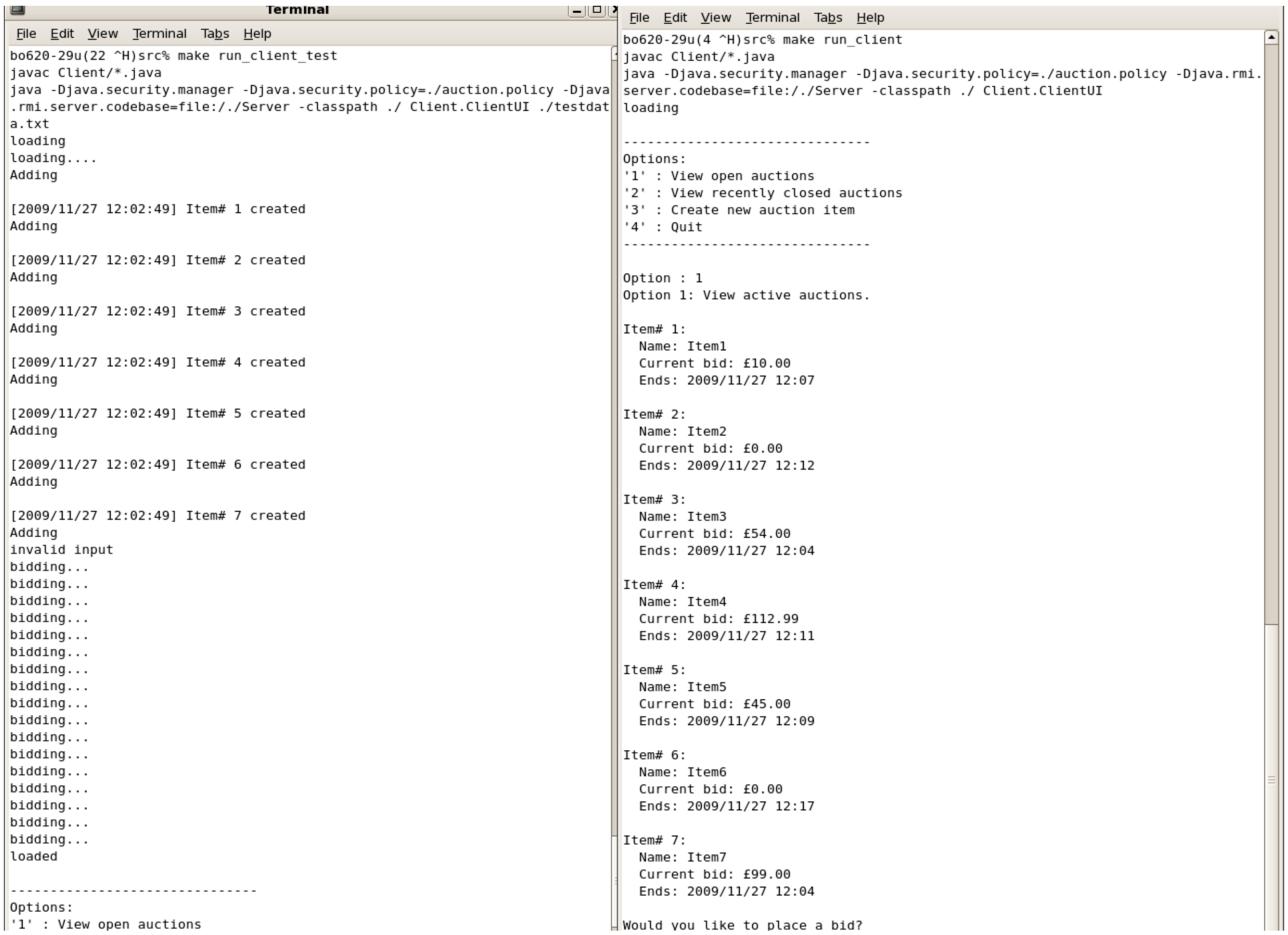
Illustration 6: Inactive Auctions

A user may view recently closed items. Closed items are kept for a set time period of 5 minutes. This time period may be configured on the server, but it is not implemented in the interface.

2 Testing

2.1 Test Driver

Usage: To Run the TestDriver:
- Open 3 terminals and locate to the src folder of the Project
- In one terminal enter 'rmi registry'
- In another, enter 'make' to start the Server
- In another, enter 'make run_client_test' to start the ClientUI with test data


Terminal
File Edit View Terminal Tabs Help
File Edit View Terminal Tabs Help

```
bo620-29u(22 ^H)src% make run_client_test
javac Client/*.java
java -Djava.security.manager -Djava.security.policy=./auction.policy -Djava
.rmi.server.codebase=file:/./Server -classpath ./ Client.ClientUI ./testdat
a.txt
loading
loading....
Adding

[2009/11/27 12:02:49] Item# 1 created
Adding

[2009/11/27 12:02:49] Item# 2 created
Adding

[2009/11/27 12:02:49] Item# 3 created
Adding

[2009/11/27 12:02:49] Item# 4 created
Adding

[2009/11/27 12:02:49] Item# 5 created
Adding

[2009/11/27 12:02:49] Item# 6 created
Adding

[2009/11/27 12:02:49] Item# 7 created
Adding
invalid input
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
bidding...
loaded

------------------------------
Options:
'1' : View open auctions
```

```
bo620-29u(4 ^H)src% make run_client
javac Client/*.java
java -Djava.security.manager -Djava.security.policy=./auction.policy -Djava.rmi.
server.codebase=file:/./Server -classpath ./ Client.ClientUI
loading

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option : 1
Option 1: View active auctions.

Item# 1:
  Name: Item1
  Current bid: £10.00
  Ends: 2009/11/27 12:07

Item# 2:
  Name: Item2
  Current bid: £0.00
  Ends: 2009/11/27 12:12

Item# 3:
  Name: Item3
  Current bid: £54.00
  Ends: 2009/11/27 12:04

Item# 4:
  Name: Item4
  Current bid: £112.99
  Ends: 2009/11/27 12:11

Item# 5:
  Name: Item5
  Current bid: £45.00
  Ends: 2009/11/27 12:09

Item# 6:
  Name: Item6
  Current bid: £0.00
  Ends: 2009/11/27 12:17

Item# 7:
  Name: Item7
  Current bid: £99.00
  Ends: 2009/11/27 12:04

Would you like to place a bid?
```

Illustration 7: Test Driver

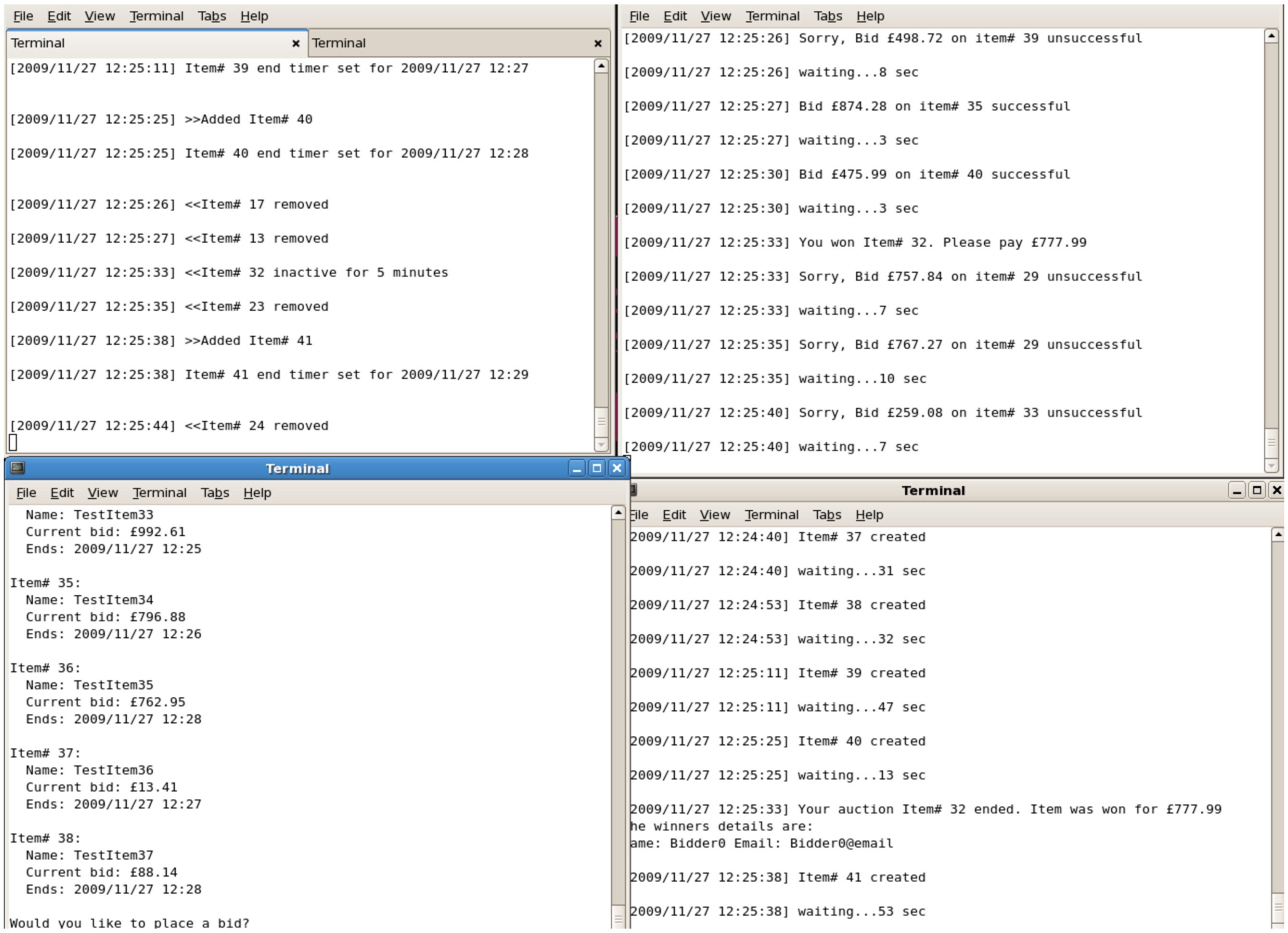
I created a test driver that populates the server with initial auction items and bids. This aided testing the user interface by providing several auctions to bid on.

2.2 Automated Test

Usage: To Run the automated test:

- Open 4 terminals and locate to the src folder of the Project
- In one terminal enter 'rmi registry'
- In another, enter 'make' to start the Server
- In another, enter './TestAuctions n' to start the test auction thread
- In another, enter './TestBidder n' to start the test bidders thread
    - NOTE: In each case, n is the number of threads to create

I created scripts to automate the auctioneering and bidding process. These take in an integer as a parameter which is the number of bidders or auctioneers to create. I used these to run in the background as I tested the user interface, and to check that the system was behaving as expected, and to act as individual clients.



```
[2009/11/27 12:25:11] Item# 39 end timer set for 2009/11/27 12:27


[2009/11/27 12:25:25] >>Added Item# 40

[2009/11/27 12:25:25] Item# 40 end timer set for 2009/11/27 12:28


[2009/11/27 12:25:26] <<Item# 17 removed

[2009/11/27 12:25:27] <<Item# 13 removed

[2009/11/27 12:25:33] <<Item# 32 inactive for 5 minutes

[2009/11/27 12:25:35] <<Item# 23 removed

[2009/11/27 12:25:38] >>Added Item# 41

[2009/11/27 12:25:38] Item# 41 end timer set for 2009/11/27 12:29


[2009/11/27 12:25:44] <<Item# 24 removed
```

```
  Name: TestItem33
  Current bid: £992.61
  Ends: 2009/11/27 12:25

Item# 35:
  Name: TestItem34
  Current bid: £796.88
  Ends: 2009/11/27 12:26

Item# 36:
  Name: TestItem35
  Current bid: £762.95
  Ends: 2009/11/27 12:28

Item# 37:
  Name: TestItem36
  Current bid: £13.41
  Ends: 2009/11/27 12:27

Item# 38:
  Name: TestItem37
  Current bid: £88.14
  Ends: 2009/11/27 12:28

Would you like to place a bid?
```

```
[2009/11/27 12:25:26] Sorry, Bid £498.72 on item# 39 unsuccessful

[2009/11/27 12:25:26] waiting...8 sec

[2009/11/27 12:25:27] Bid £874.28 on item# 35 successful

[2009/11/27 12:25:27] waiting...3 sec

[2009/11/27 12:25:30] Bid £475.99 on item# 40 successful

[2009/11/27 12:25:30] waiting...3 sec

[2009/11/27 12:25:33] You won Item# 32. Please pay £777.99

[2009/11/27 12:25:33] Sorry, Bid £757.84 on item# 29 unsuccessful

[2009/11/27 12:25:33] waiting...7 sec

[2009/11/27 12:25:35] Sorry, Bid £767.27 on item# 29 unsuccessful

[2009/11/27 12:25:35] waiting...10 sec

[2009/11/27 12:25:40] Sorry, Bid £259.08 on item# 33 unsuccessful

[2009/11/27 12:25:40] waiting...7 sec
```

```
2009/11/27 12:24:40] Item# 37 created

2009/11/27 12:24:40] waiting...31 sec

2009/11/27 12:24:53] Item# 38 created

2009/11/27 12:24:53] waiting...32 sec

2009/11/27 12:25:11] Item# 39 created

2009/11/27 12:25:11] waiting...47 sec

2009/11/27 12:25:25] Item# 40 created

2009/11/27 12:25:25] waiting...13 sec

2009/11/27 12:25:33] Your auction Item# 32 ended. Item was won for £777.99
he winners details are:
ame: Bidder0 Email: Bidder0@email

2009/11/27 12:25:38] Item# 41 created

2009/11/27 12:25:38] waiting...53 sec
```

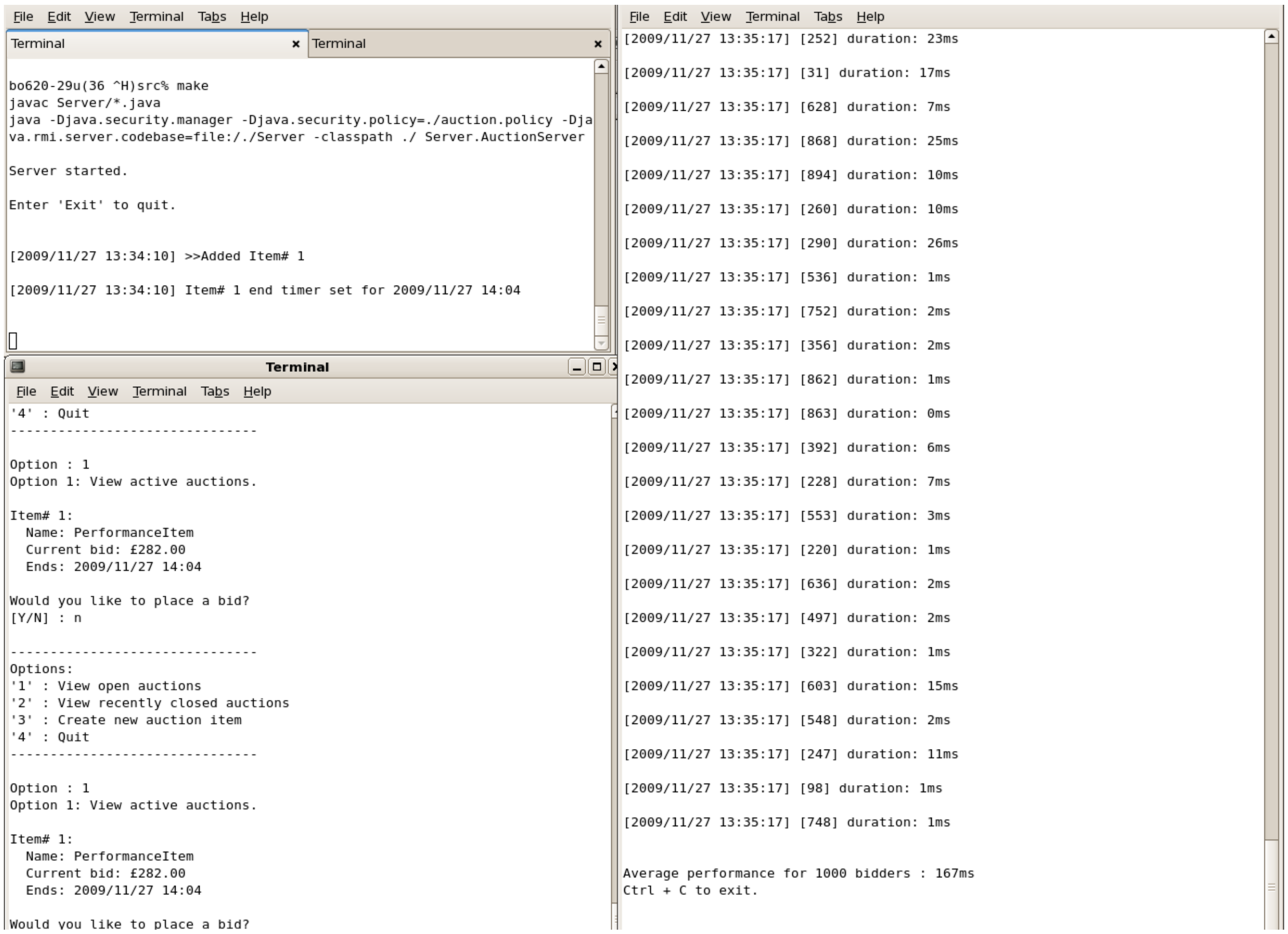
Illustration 8: Automated tests - Left: Server, Top right: Create random bids, Bottom right: Create random auctions

2.3 Performance Test

Usage: To Run the Automated Auction Threads:

- Open 3 terminals and locate to the src folder of the Project
- In one terminal enter 'rmi registry'
- In another, enter 'make' to start the Server
- In another, enter './PerformanceTest n' to start the test auction thread
    - NOTE: n is the number of threads to create

The performance test creates one item, and a number of threads that constantly place bids of increasing value for 1 minute. The test measures the duration of each method call.


File Edit View Terminal Tabs Help
Terminal
Terminal
Terminal
File Edit View Terminal Tabs Help
File Edit View Terminal Tabs Help

```
bo620-29u(36 ^H)src% make
javac Server/*.java
java -Djava.security.manager -Djava.security.policy=./auction.policy -Dja
va.rmi.server.codebase=file:/./Server -classpath ./ Server.AuctionServer

Server started.

Enter 'Exit' to quit.


[2009/11/27 13:34:10] >>Added Item# 1

[2009/11/27 13:34:10] Item# 1 end timer set for 2009/11/27 14:04
```

```
'4' : Quit
------------------------------

Option : 1
Option 1: View active auctions.

Item# 1:
  Name: PerformanceItem
  Current bid: £282.00
  Ends: 2009/11/27 14:04

Would you like to place a bid?
[Y/N] : n

------------------------------
Options:
'1' : View open auctions
'2' : View recently closed auctions
'3' : Create new auction item
'4' : Quit
------------------------------

Option : 1
Option 1: View active auctions.

Item# 1:
  Name: PerformanceItem
  Current bid: £282.00
  Ends: 2009/11/27 14:04

Would you like to place a bid?
```

```
[2009/11/27 13:35:17] [252] duration: 23ms

[2009/11/27 13:35:17] [31] duration: 17ms

[2009/11/27 13:35:17] [628] duration: 7ms

[2009/11/27 13:35:17] [868] duration: 25ms

[2009/11/27 13:35:17] [894] duration: 10ms

[2009/11/27 13:35:17] [260] duration: 10ms

[2009/11/27 13:35:17] [290] duration: 26ms

[2009/11/27 13:35:17] [536] duration: 1ms

[2009/11/27 13:35:17] [752] duration: 2ms

[2009/11/27 13:35:17] [356] duration: 2ms

[2009/11/27 13:35:17] [862] duration: 1ms

[2009/11/27 13:35:17] [863] duration: 0ms

[2009/11/27 13:35:17] [392] duration: 6ms

[2009/11/27 13:35:17] [228] duration: 7ms

[2009/11/27 13:35:17] [553] duration: 3ms

[2009/11/27 13:35:17] [220] duration: 1ms

[2009/11/27 13:35:17] [636] duration: 2ms

[2009/11/27 13:35:17] [497] duration: 2ms

[2009/11/27 13:35:17] [322] duration: 1ms

[2009/11/27 13:35:17] [603] duration: 15ms

[2009/11/27 13:35:17] [548] duration: 2ms

[2009/11/27 13:35:17] [247] duration: 11ms

[2009/11/27 13:35:17] [98] duration: 1ms

[2009/11/27 13:35:17] [748] duration: 1ms


Average performance for 1000 bidders : 167ms
Ctrl + C to exit.
```

Illustration 9: PerformanceTest - Top left:Server, Bottom left:ClientUI, Right: PerformanceTest with 10000 bidders

Performance results:

Number of Bidders	Average Method Call Duration (ms)
1	3
10	15
100	140
1000	144
10000	163

The average duration is measured by each bidder thread storing their total number of method calls and their total method call duration. After 1 minute, the test is stopped and the collective durations and method calls are added. The average is calculated by the totalDuration/totalCalls.

3 Potential Extensions

The system supports simple client/server distribution. However, there are many areas in which it could be extended.

To improve the availability of the system, there could be support for multiple distributed servers. This would allow distributed auction lists to store auction items, in the case that there is distributed storage space. This could solve any storage problems that might be introduced by a drastic increase of auctions. There could also be replication managers to back up the auction lists to ensure that a failure on any server does not affect the running of the system.

To improve fault tolerance, there could be better support for notifying clients. At the moment if a client disconnects from the rmiregistry, then it is still bound to the registry and an exception is thrown when the server tries to notify the client. To improve this, callbacks should not be used to notify the client and emails should be sent to the contact address, or user accounts could be introduced to allow users to disconnect and still receive their notifications.

3 Testing the System

Please follow the usage instructions as described in sections 1.1, 2.1, 2.2 and 2.3 to test the respective parts of the system.

NOTE: the paths in the scripts and makefile are relative, and so should work so long as the terminal is located to the 'src' directory.