

# HPN Calendar Network

TUMS Version!

Practical Assignment 1

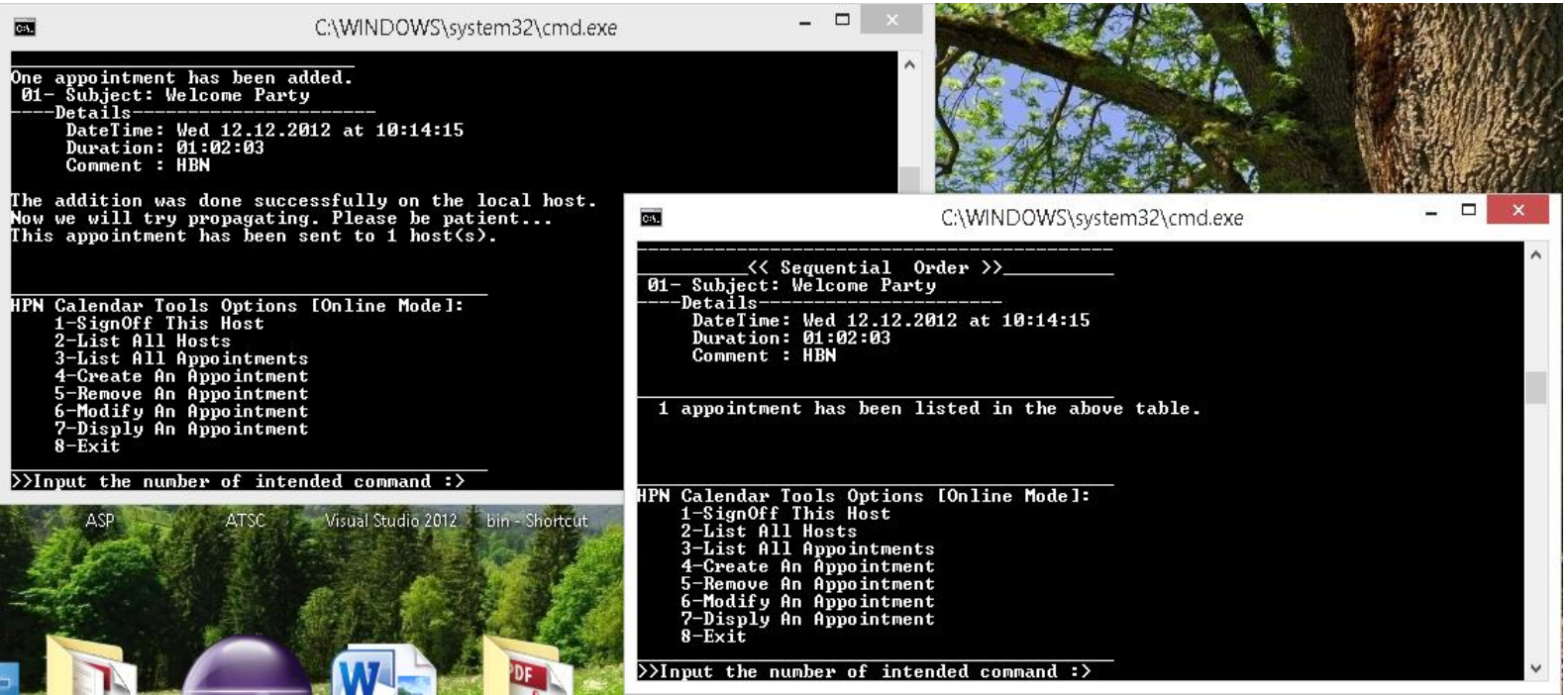
Winter Semester 2013

Group 09

<<Members>>

Mahdi Jaberzadeh Ansari

I. A picture of execution both the Java and C#



## II. Usage and User Interface

This project was written in JAVA and C#. In the AP1 folder you can see calendar1 folder that contains JAVA implementation of the assignment and in the calendar2 you can find the C# implementation of this assignment.

Both these folders contain the source code and executable shell files which start the binary version by double click!

- **How to execute JAVA version?**

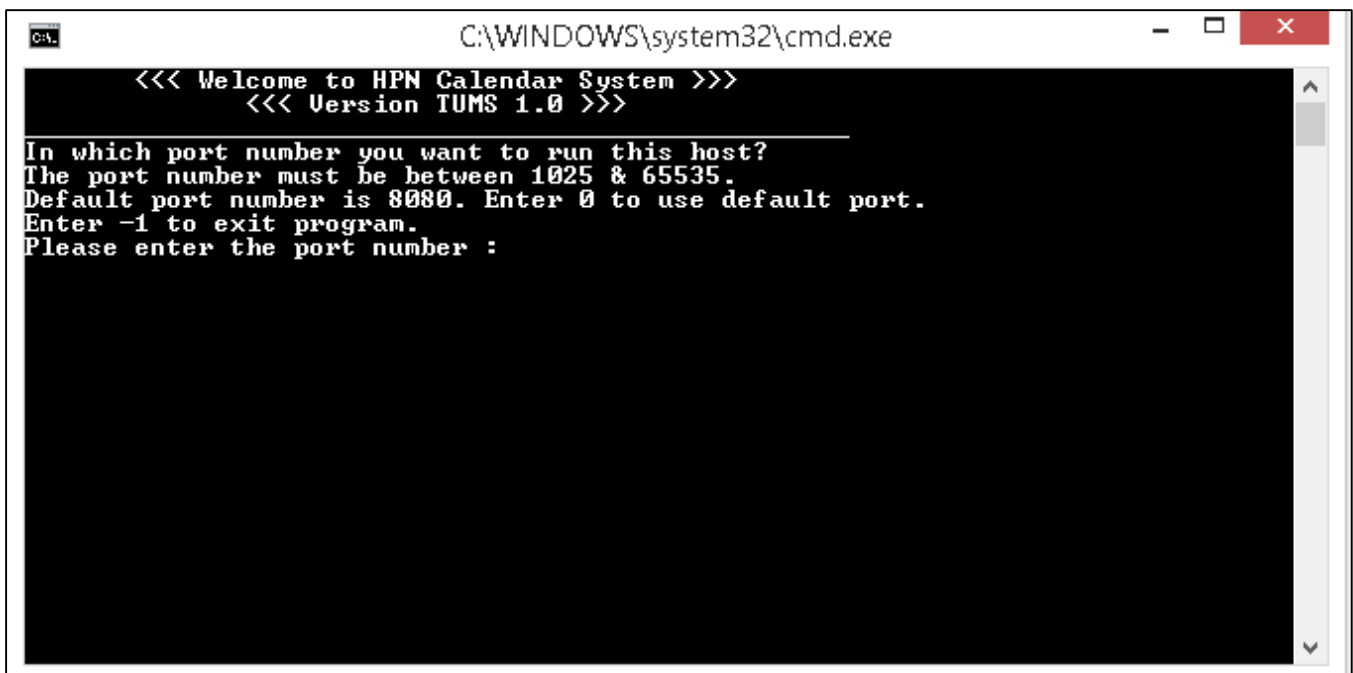
For executing the java version you must put "Calendar1.jar" and "start.bat" in the same folder and then click on the "start.bat" file, in the first start it will make a "database.hpn" file (of course if it's not exist in the current folder!) that will store the appointments for the future executions and providing the "Persistent Storage" future!

- **How to execute the C# version?**

For executing the C# version you must put "Calendar2.exe" and "XmlRpcCS.dll" and "start.bat" in the same folder and then click on the "start.bat" file. Same as above it will make a "database.hpn" file if it's not exist in the current folder, or read it and load the old appointments from that have been stored from previous run!

- **Steps to use the program :**

1. Click on "start.bat" file to see the following window! Please note the commands and user interface in both implementations are the same.



```
C:\WINDOWS\system32\cmd.exe

<<< Welcome to HPN Calendar System >>>
<<< Version TUMS 1.0 >>>

In which port number you want to run this host?
The port number must be between 1025 & 65535.
Default port number is 8080. Enter 0 to use default port.
Enter -1 to exit program.
Please enter the port number :

```

2. Enter the port number that you want your host to run and then press the Enter key. If you enter the 0 it will use port number 8080 as the default port. And if you enter -1 the program will close! The port that you want to enter must be greater than 1024 and less than 65535.

3. After pressing the enter if the server can run on the port that you have entered, it will show the IP address of the current machine and then ask :  
*"Do you want to create a new Calendar Network? [Y/N]:"*  
 If it is the first host you must press the "Y" because it must make the calendar network and the others must enter "N" to have the options to connect to the Calendar Network!
4. Press "Y" then you will see the following figure.

```

C:\WINDOWS\system32\cmd.exe
<<< Welcome to HPN Calendar System >>>
<<< Version TUMS 1.0 >>>

In which port number you want to run this host?
The port number must be between 1025 & 65535.
Default port number is 8080. Enter 0 to use default port.
Enter -1 to exit program.
Please enter the port number : 8080
The port number has assigned to : 8080
The XML-RPC server has checked : Ok.
The host has run on this address : http://172.16.1.100:8080/
Do you want to create a new Calendar Network? [Y/N] : y

HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> _

```

5. In the above picture you can see the following options :

```

HPN Calendar Tools Options [Online Mode]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :>

```

6. At the first line you can see the project is in its online mode!
7. Then you have some options! That is better to start with (3). To check how many appointments has been loaded from local data base!
8. If you try now may be you seeing an empty list!
9. So after check the numbers of current appointments on the system try to create one appointment.
10. Now we will try to work with each part and you can follow the next figures.

- **List of all appointments:**

We have 2 methods for listing the appointments:

1. We can sort them based on sequential number

```
HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 3
Please select the type of the order of the list
1-Order Based On Squential Number
2-Order Based On Date Parameter
Please enter the number of your choice : 1
-----
! <<< List Of All Appointments >>> !
-----
<< Sequential Order >>
02- Subject: Birthday Mahdi
-----
Details
DateTIme: Wed 15.05.1985 at 12:12:45
Duration: 03:04:05
Comment : Jaberzadeh

03- Subject: Tina Birthday
-----
Details
DateTIme: Mon 26.12.1988 at 14:15:09
Duration: 05:06:09
Comment : Boroukhian

05- Subject: +ukds
-----
Details
DateTIme: Sun 12.09.1367 at 00:12:23
Duration: 02:03:01
Comment : nlf

06- Subject: tin
-----
Details
DateTIme: Wed 05.10.1988 at 14:32:45
Duration: 01:02:01
Comment : moon
```

List of appointments based on sequential number

Now in the list of appointments we have 4 appointments with the sequential number: 2,3,5,6. With this method we can understand that the appointment 1 and 4 has been removed already and we cannot have sequential number 1 or 4 when we want to add a new appointment, since our sequential numbers are unique.

2. Also we can sort them based on their date parameter

```

HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 3

Please select the type of the order of the list.
1-Order Based On Squential Number
2-Order Based On Date Parameter
Please enter the number of your choice : 2

! <<< List Of All Appointments >>> !
-----
<< Date Order >>
01- Subject: +ukds
---Details-----
SequNum : 05
DateTime: Sun 12.09.1367 at 00:12:23
Duration: 02:03:01
Comment : mlf

02- Subject: Birthday Mahdi
---Details-----
SequNum : 02
DateTime: Wed 15.05.1985 at 12:12:45
Duration: 03:04:05
Comment : Jaberzadeh

03- Subject: tin
---Details-----
SequNum : 06
DateTime: Wed 05.10.1988 at 14:32:45
Duration: 01:02:01
Comment : moon

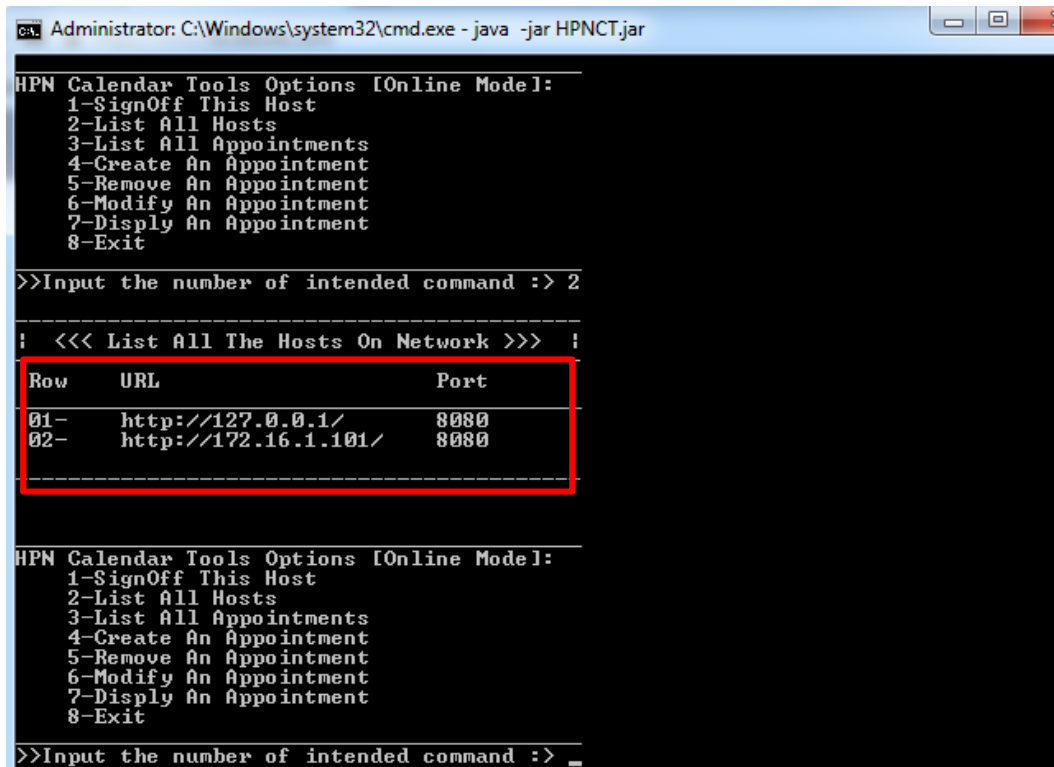
04- Subject: Tina Birthday
---Details-----
SequNum : 03
DateTime: Mon 26.12.1988 at 14:15:09
Duration: 05:06:09
Comment : Boroukhian

```

List of appointments based on date parameter

- **List of all hosts on the network :**

We can see the list of all hosts which are now available in our calendar network. Please note the first member of the host list is refer to the current host and the rows 2 until the end are refer to other hosts!



```
Administrator: C:\Windows\system32\cmd.exe - java -jar HPNCT.jar

HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 2

! <<< List All The Hosts On Network >>> !

Row    URL                                Port
01-    http://127.0.0.1/                  8080
02-    http://172.16.1.101/              8080

HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 
```

- *Add an appointment*

```
HPN Calendar Tools Options [Online Model]:
1-SignOff This Host
2-List All Hosts
3-List All Appointments
4-Create An Appointment
5-Remove An Appointment
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 4

! <<< Add new appointment procedure >>> !

Please enter the date of the appointment in the following format [dd.mm.yyyy] :
09.07.2013
Please enter the time of the appointment in the following format [hh:mm:ss], use
24 hours format : 15:30:00
Please enter the header/subject of the appointment : Meeting
Please enter the comment/body of the appointment : Boss
Please enter the hours, minutes and seconds of the appointment's duration one by
one.
These parameters must be equal or greater that zero, and less than 60.
Please enter the hours of the duration [0-59] : 2
Please enter the minutes of the duration [0-59] : 30
Please enter the seconds of the duration [0-59] : 00
The addition was done successfully on the local host.
Now we will try propagating. Please be patient...
This appointment has been sent to 1 host(s).
```

So now we are adding one appointment in our calendar:

```
Please enter the date of the appointment in the following format [dd.mm.yyyy] : 09.07.2013
Please enter the time of the appointment in the following format [hh:mm:ss], use 24 hours format :
15:30:00
Please enter the header/subject of the appointment : Meeting
Please enter the comment/body of the appointment : Boss
Please enter the hours, minutes and seconds of the appointment's duration one by one.
These parameters must be equal or greater that zero, and less than 60.
Please enter the hours of the duration [0-59] : 2
Please enter the minutes of the duration [0-59] : 30
Please enter the seconds of the duration [0-59] : 00
```

One appointment has been added.

07- Subject: Meeting

----Details-----

DateTime: Tue 09.07.2013 at 15:30:00

Duration: 02:30:00

Comment : Boss

The addition was done successfully on the local host.

There is no other host in the network to propagate this new appointment.



So this host has been added a new appointment with sequential number 7 in our calendar and this appointment will be send to other hosts which are in our calendar network. When we get list of appointments from our calendar we will see this new appointment with its details

```
>>Input the number of intended command :> 3
Please select the type of the order of the list.
    1-Order Based On Squential Number
    2-Order Based On Date Parameter
Please enter the number of your choice : 1

!   <<< List Of All Appointments >>>   !
-----
      << Sequential Order >>
02- Subject: Birthday Mahdi
---Details-----
    DateTime: Wed 15.05.1985 at 12:12:45
    Duration: 03:04:05
    Comment : Jaberzadeh

03- Subject: Tina Birthday
---Details-----
    DateTime: Mon 26.12.1988 at 14:15:09
    Duration: 05:06:09
    Comment : Boroukhian

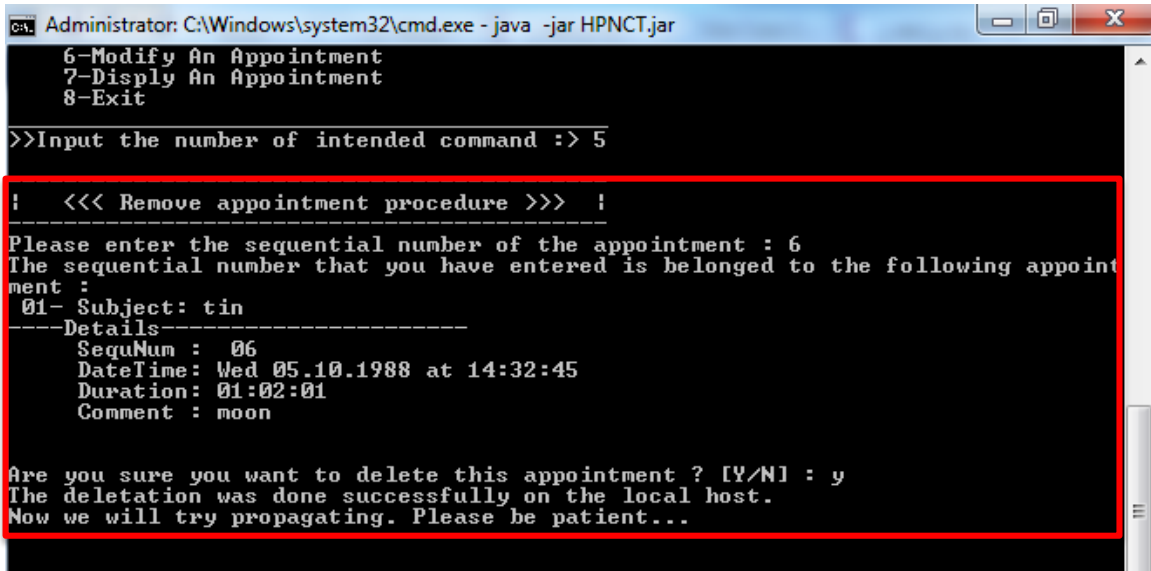
05- Subject: +ukds
---Details-----
    DateTime: Sun 12.09.1367 at 00:12:23
    Duration: 02:03:01
    Comment : mlf

06- Subject: tin
---Details-----
    DateTime: Wed 05.10.1988 at 14:32:45
    Duration: 01:02:01
    Comment : moon

07- Subject: Meeting
---Details-----
    DateTime: Tue 09.07.2013 at 15:30:00
    Duration: 02:30:00
    Comment : Boss
```

- Remove an appointment

When our users wants to remove an appointment the program will ask it to choose the appointment based on its sequential number and after that it will just get the confirmation based on information of chosen appointment from the host and then it will remove the chosen appointment and at last it will propagate this message to all of online hosts.



```
Administrator: C:\Windows\system32\cmd.exe - java -jar HPNCT.jar
6-Modify An Appointment
7-Disply An Appointment
8-Exit

>>Input the number of intended command :> 5

! <<< Remove appointment procedure >>> !
-----
Please enter the sequential number of the appointment : 6
The sequential number that you have entered is belonged to the following appointment :
01- Subject: tin
-----Details-----
      SequNum : 06
      DateTime: Wed 05.10.1988 at 14:32:45
      Duration: 01:02:01
      Comment : moon

Are you sure you want to delete this appointment ? [Y/N] : y
The deletation was done successfullly on the local host.
Now we will try propagating. Please be patient...
```

As you can see the appointment 6 has been removed from our calendar.

- *Modify appointments*

Whenever our host wants to modify a specific appointment, it has to choose the appointment base on its sequential number and our calendar will get a confirmation from host and start modifying that appointment from beginning. After the modifying it will ask the host if it is sure about changing this appointment or not, and after pressing yes this message will propagate in our calendar network.

```

-----
! <<< Modify an appointment procedure >>> !
-----
Please enter the sequential number of the appointment : 7
The sequential number that you have entered is belonged to the following appointment :
01- Subject: Meeting
-----Details-----
    SequNum : 07
    DateTime: Tue 09.07.2013 at 15:30:00
    Duration: 02:30:00
    Comment : Boss

Are you sure you want to modify this appointment ? [Y/N] : y
So please enter new parameters for this appointment.
Please enter the date of the appointment in the following format [dd.mm.yyyy] :

```

```

Are you sure you want to modify this appointment ? [Y/N] : y
So please enter new parameters for this appointment.
Please enter the date of the appointment in the following format [dd.mm.yyyy] :
01.01.2013
Please enter the time of the appointment in the following format [hh:mm:ss], use
24 hours format : 09:00:00
Please enter a new header/subject for this appointment : Meeting
Please enter a new comment/body for the appointment : friends
Please enter the hours, minutes and seconds of the appointment's duration one by
one.
These parameters must be equal or greater that zero, and less than 60.
Please enter the hours of the duration [0-59] : 4
Please enter the minutes of the duration [0-59] : 0
Please enter the seconds of the duration [0-59] : 0

```

```

Are you sure you want to save the changes for this appointment ? [Y/N] : y
The modification was done successfully on the local host.
Now we will try propagating. Please be patient...

```

- *Display an appointment*

```
! <<< Show an appointment procedure >>> !  
-----  
Please enter the sequential number of the appointment : 2  
The sequential number that you have entered is belonged to the following appointment :  
  
01- Subject: Birthday Mahdi  
-----Details-----  
    SequNum : 02  
    DateTime: Wed 15.05.1985 at 12:12:45  
    Duration: 03:04:05  
    Comment : Jaberzadeh
```

- *Sign off*

With sign off this host will be removed from the network and will be go on its sign off mode

```
! <<< Signing Off This Host >>> !  
-----  
Please wait to unjoin from all hosts on calendar network!  
All the host was removed successfully!  
Now this machine will work on offline mode!  
  
-----  
This host is offline, sign On for more options.  
You can just see last appointments that has  
been saved on hard disk in last connection.  
  
HPN Calendar Tools Options[Offline Model]:  
    1-SignOn/Connect To The Network  
    2-List All Appointments  
    3-Show An Appointment  
    4-Exit  
  
-->>Input the number of intended command :>
```

### III. Source code explanation

In this section we want to explain the source code. Actually at first we divided to 2 subgroups to act on the project in the both C# and JAVA sections simultaneously, but unfortunately the C# subgroup was not as powerful as the JAVA subgroup and we decided to work with each other on JAVA at first step and then switch on C# section.

Maybe it has some advantages and disadvantages. At first all group members learned both implementation, but when we want to switch to implement the C# section we didn't have enough time as much as the previous section! Since we had decided to change the JAVA implementation to C#, we spent too many time to make a neat and trustable code in JAVA.

Based on what you suggest us to use Apache XML-RPC library<sup>1</sup> at first we have searched the Internet to find an example for server and client implementation that use XML-RPC. We found many results but all of them use an old library of XML-RPC that called *Helma XML-RPC* and for example in the following links there are some implementations but none of them work with XML-RPC 3.0 that we have downloaded from Apache website!

- <http://tldp.org/HOWTO/XML-RPC-HOWTO/xmlrpc-howto-java.html>
- <http://www.tutorialspoint.com/xml-rpc/index.htm>
- <http://oreilly.com/catalog/progxmlrpc/chapter/ch03.html>

As we mentioned above, none of them work truly because they used some functions that had been dropped in the current version. But we found the Helma.jar file in some websites that archive the files in the internet. It was our start point for remote calling!

After more search we find a good example that implements the client and the server of XML-RPC 3.0 in the following website:

- <http://oreilly.com/catalog/progxmlrpc/chapter/ch03.html>

So we use the codes of this website as the core of our client and the server in Java section.

After we could implement our java program and have tested it completely we have switched to implement C# section. In this part we start with WCF that you have mentioned in your assignment sheet! But it was some time consuming and we didn't have time to spend!

So we have switched to use XML-RPC.NET<sup>2</sup> that is called cooking computing library! But after some pretest for connecting Java and C# we found out that it is impossible to call functions by this library same as the way we had used in Apache library. Because it has no handler for functions same as Apache, and we couldn't find any way to refer to a function in the across point to connect Java to C# and vice versa.

---

1) <http://ws.apache.org/xmlrpc/>

2) <http://xml-rpc.net/>

So we have searched the Internet again and found the following .pdf file that explains a research about connecting Java and C# programs by XML-RPC!

- [http://www.c-lab.de/fileadmin/user\\_upload/Ueber\\_Uns/Services\\_Downloads/C-LAB\\_Reports/2006/C-LAB-TR-2006-1\\_Evaluation\\_of\\_XML-RPC.pdf](http://www.c-lab.de/fileadmin/user_upload/Ueber_Uns/Services_Downloads/C-LAB_Reports/2006/C-LAB-TR-2006-1_Evaluation_of_XML-RPC.pdf)

It was what we need to! And in this pdf it has shown how to make client and server in both Java and C# to call a function in C# and Java respectively. The library that they have used in the C# section was some old and unfortunately has not been developed after 2003! So it has some problems sometimes with the current .Net framework 4.5! But as we said before this library use handler to recognize functions in the C# same as what we use in Apache library!

The library is downloadable from the following link:

- <http://xmlrpccs.sourceforge.net/>

It made some problems for us in usage! For example we couldn't stop the serving on the server in sign off mode so we developed some additional classes to support it, or we couldn't to send the user defined types (Objects) and we had to just use primitive types like integer or Boolean and string!

But that was fine! Because we didn't need to make 2 different proxies for calling functions in C# and Java! In the other hand in this way if we accept the handler name as a standard for remote calling! We can just provide a schema and interfaces to share our servicing functions with others to call our functions remotely because the goal of developing XML-RPC was this!

But if we used the XML-RPC.NET library or some others we might store the type of the hosts in the Network to call them with different proxies!!!! And we think it is far from the main goal of this assignment if using of the XML-RPC technology to achieve platform independent way was one of its purposes! And in this way we wanted to approach this goal that call the functions in both Java and C# in an implementation independent way!

Now it's the time to go in to the code explanation section.

Our code is full of comments so we do not explain the code line by line and we focus on the functionality and out code organization!

## A. Java Implementation

The code has packaged in 7 packages as follow:

Package Name	Explanation
<b>hpn.main</b>	This package contain the 'HpnCalendarTools' class that have the <i>main</i> function.
<b>hpn.settings</b>	This package contain 3 classes that store some default values.

<b>hpn.console.scanner</b>	This package contains a class that manages the reading user inputs.
<b>hpn.console.file</b>	This package contains a class that provides the file reading and writing functionality but not parsing contains.
<b>hpn.calendar</b>	This package has 6 classes that manage storing the appointments on the main memory and provide some functionality same as add, remove, modify, listing that are related to appointments. Also managing the sequential number and loading from local database and updating the local database is placed in this package.
<b>hpn.apache.xml.client</b>	This package has 5 classes that manage the functionality to sign in and sign off the host to the Calendar Network. And also some classes that keep a list of all hosts that are registered in the Network. Especially managing the user interface and executing the user instructions are held by 'HpnXmlRpcClient' class.
<b>hpn.apache.xml.webserver</b>	This package contains 3 classes that handle the server section to response the requests of local and foreign hosts.

The classes of the first 4 packages are clear and we explain the classes of each 3 last packages of the above list in the following table.

1) Package 'hpn.calendar'

class	Description
<b>SequentialNumber</b>	This class manages the sequential Number. It provides some functionality to produce a unique sequential number each time that a new appointment want to create. Also it provides some function to set the sequential number to an especial value when the host start or sign in to the network calendar. In this way we do not need to use any database to generate any unique id.
<b>Appointment</b>	This class provides a blue print for appointment object. It contains the properties like date, header, duration, and comment and some functions to set and get this properties.
<b>CalendarTools</b>	This interface introduce the public functions that are related to the calendar itself, same as 'add New Appointment', 'remove Appointment', 'modify Appointment', 'synchronize Request'. These functions will call by XML-RPC to manage the calendar. Note the sign in and sign off functionalities does not matter to calendar and they are collected in client package.
<b>Calendar</b>	This class is one of the most important classes of this system that implement the functions that are needed to manage the calendar locally or externally. Please note that in this system we execute both local requests and external requests for adding, modifying, removing and synchronizing throw XML-RPC. So many of the core functions for this purpose have gathered in this class.
<b>AppointmentComparator</b>	We provide an extra functionality to order the list of appointments based on sequential number or based on their dates. These 2 classes are needed for sorting the appointment list!
<b>ListOrder</b>	

## 2) Package 'hpn.apache.xml.client'

class	Description
<b>HostUrl</b>	This class provides a blue print for other host objects. In fact we store the IP address and port number of each host as a HostUrl object. This class contains some function to provide validation check for the url address of the hosts and their port number.
<b>HostsList</b>	This class keeps a list of all hosts in its self. Actually this class provide 3 functions 'joinRequest', 'addMe', 'removeMe' that are listed in 'CalendarNetwork' interface to provide functionality of joining and rupturing for external hosts. You can read more about the role of these functions in the last row of this table! Also this class provides some functions to list and modify the hosts for local user. Please note we reserve the first element of the host list for storing the url address and port number of the local host! Because the local user must send its requests to the local server by XML-RPC, and then if the result of any request was OK, then it will use the list of other hosts for propagating!
<b>HpnXmlRpcClient</b>	Actually it is the core of the client. All expected functions that are listed in the 'HpnClientFunctionality' class are implemented here to execute the local user commands!
<b>HpnClientFunctionality</b>	This is an abstract class that collects the name of all important functionality that we expected for the local client. In the other word it contains all functions that are needed to communicate with the user and executing the local user commands.
<b>CalendarNetwork</b>	This interface contains the signature of 3 methods that are needed to support joining and rupturing the external hosts. <b>joinRequest</b> : When a remote host need to join to the network can call this function by using the handler "CalendarNetwork.joinRequest" in any other host that are in the network currently and then will receive a list of all other hosts. <b>addMe</b> : If a host can call the joinRequest function in one of the hosts and it receive the list of all hosts successfully then it will call the "Calendar.synchronizeRequest" to receive the appointments list and the excepted sequential number. And then this new host must call this function on all other hosts that it has received the list of them in previous section. <b>removeMe</b> : When a host need to rupture the calendar network must call this function on all other hosts one by one!

## 3) Package 'hpn.apache.xml.webserver'

class	Description
<b>HpnXmlRpcRequestHandlerFactory</b>	These classes provide the needed functionality to make a XML-RPC host to receive and execute the functions on this host and send back the response to the clients that send the requests. This package is important but not as much as previous packages!
<b>HpnXmlRpcServer</b>	
<b>ServerStatus</b>	



## B. C# Implementation

In the C# section all things are the same just we do not have 'HpnXmlRpcRequestHandlerFactory' class and we do not have 'HpnXmlRpcServer' with the same codes as the previous time! Because in the C# we use XmlRpcCS.dll file that packages all the functionality that we needed to make the both client and server in C#! So we just made the 'HpnXmlRpcServer' class to simulate the class that we had in our Java implementation.

Also we provide 'date' and 'Integer' classes to simulate the classes we have in Java language to reduce the changing in our code!