The materials required for this project will be:

1. Raspberry Pi
2. 8GB SD card
3. USB mini charger (Comes with almost all smart phones now)
4. USB Mouse and Keyboard
5. HDMI Cable
6. Monitor (Any Size)
7. A router, depending on where you are.

The 1st step in making a server on the Pi is to download an appropriate Disk Image. Choose which ever one you are comfortable with. Personally I went with the Raspbian (recommended by my teacher). This distribution is easy to pick and not all difficult to navigate through.

Distributions can be found here:

http://www.raspberrypi.org/downloads

To write the image onto your SD card, use Win32DiskManager (Simple and straightforward):

http://sourceforge.net/projects/win32diskimager/

The 2nd Step is to make sure your Pi is updated after the initial boot-up:

```
sudo apt-get update
sudo apt-get upgrade
```

Internet Access on the Raspberry PI

If you connect the PI to your local network which has a ADSL/CABLE router or mode, then it should have DHCP on. DHCP will provide it an IP address and this will be displayed on-screen when the PI boots up before the log in screen. If it doesnt, you need to get an IP and add this in manually. To do this, log in to the PI using the default username: pi and password: Raspberry and type the following command:

```
sudo vi /etc/network/interfaces
```

Using VI: Edit the file using by pressing 'a' to insert text. Once, edited, press escape to come out of editing mode.

The IP parameters should be as follows with your specific IP address information:

```
iface eth0 inet static
address 192.168.1.100
gateway 192.168.1.1
```

```
netmask 255.255.255.0
network 192.168.1.0
broadcast 192.168.1.255
```

NOTE: Dont forget to comment/remove line stating:

```
iface eth0 inet dhcp
```

Additionally you will need to set up DNS in order to get out on the internet. This setting should be done in the resolve.conf file as follows:

```
sudo vim/etc/resolv.conf
```

This should say something like:

```
nameserver 8.8.8.8
nameserver 192.168.1.1
```

to test network connectivity, at the command line, type in

```
ping <your favourite serach engine.com>
```

Press CTRL+C to stop pinging.

Update Asterisk Install Pre-requisites

To start with, update the system and install the required componants by copying the command below:

```
sudo apt-get update
```

```
sudo apt-get -y install make gcc g++ libxml2 libxml2-dev ssh libncurses5 libncursesw5
libncurses5-dev libncursesw5-dev linux-libc-dev sqlite libnewt-dev libusb-dev zlib1g-dev
libmysqlclient15-dev libsqlite0 libsqlite0-dev bison openssl libssl-dev libeditline0 libeditline-dev
libedit-dev mc sox libedit2 libedit-dev curl libcurl4-gnutls-dev apache2 libapache2-mod-php5
php-pear openssh-server build-essential openssh-client zlib1g zlib1g-dev libtiff4 libtiff4-dev
libnet-telnet-perl mime-construct libipc-signal-perl libmime-types-perl libproc-waitstat-perl
mpg123 libiksemel-dev php5 php5-cli mysql-server php5-mysql php-db libapache2-mod-php5
php5-gd php5-curl mysql-client vim
```

Dowload and Extract Asterisk

To Download and Extract asterisk, enter the commands as below:

```
cd /usr/src
```

```
sudo wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-1.8-current.tar.gz
sudo tar xvfz asterisk-1.8-current.tar.gz
cd asterisk-1.8*
```

Compile and Install Asterisk
Using the following commands:

```
sudo ./configure
sudo make
sudo make install
sudo make samples
sudo make config
```

Finally, restart the PI by typing the fillowing command:
Sudo int6
Next you should give Asterisk permissions
sudo useradd -c "Asterisk PBX" -d /var/lib/asterisk asterisksudo chown -R asterisk:asterisk /var/run/asterisksudo chown -R asterisk:asterisk /var/log/asterisksudo chown -R asterisk:asterisk /var/lib/php/session/
The next step is to find a softphone program to use in order to communicate between two devices and configure the sip.conf and extensions.conf files within Asterisk.
The sip.conf file should look like this according to the research and images I’ve seen:

```
[general]
context = default
bindport = 5060
bindaddr = 0.0.0.0
tcpbindaddr = 0.0.0.0
tcpenable = yes

[1001]
type = friend
callerid = User One <1001>
secret = 1001
host = dynamic
canreinvite = no
dtmfmode = rfc2833
mailbox = 1001
disallow = all
allow = ulaw
transport = udp

[1002]
type = friend
callerid = User Two <1002>
secret = 1002
```

```
host = dynamic
canreinvite = no
dtmfmode = rfc2833
mailbox = 1002
disallow = all
allow = ulaw
transport = udp
```

Unfortunately mine looks nothing like that and this is my largest obstacle at the moment.