**1. SERVER SETUP – WINDOWS SERVER**

**PRE-REQUISITIES: Should have installed nodejs, python3.6**

**1.1 UPGRADE OTREE**

**1.1.1 In PowerShell or CMD:** **pip3 install -U otree-core.** Also, **pip3 install numpy**

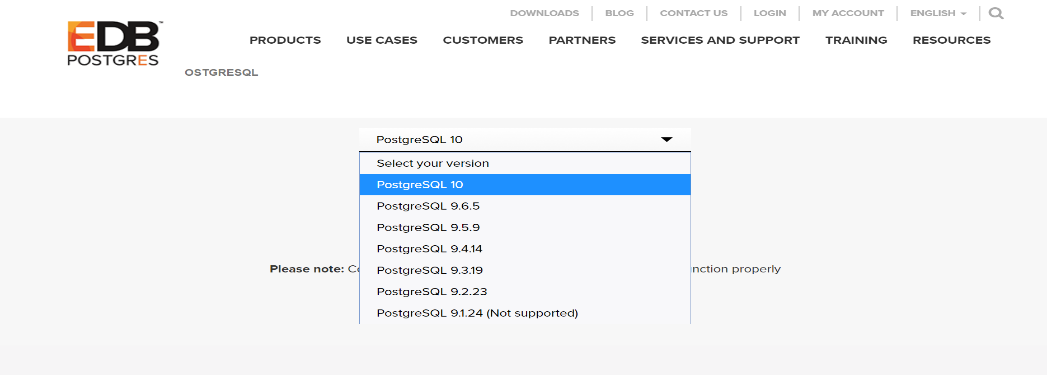
Better to use this at least twice a week for constant updating of otree bugfixes



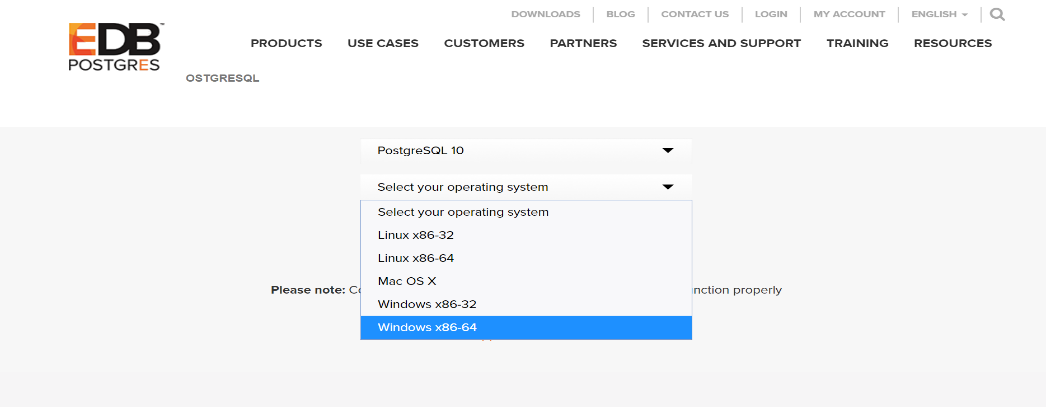
**1.2 INSTALL POSTGRESQL DATABASE**

**1.2.1 Download from the link given:** [**Postgres**](http://www.enterprisedb.com/products-services-training/pgdownload#windows)

1.2.1.1 Select the latest version of Postgres



1.2.1.2 Select the appropriate operating system based on the computer you are going to install (either 32-bit or 64-bit Windows system). I am using a 64-bit Windows operating system hence selecting Windows x86-64

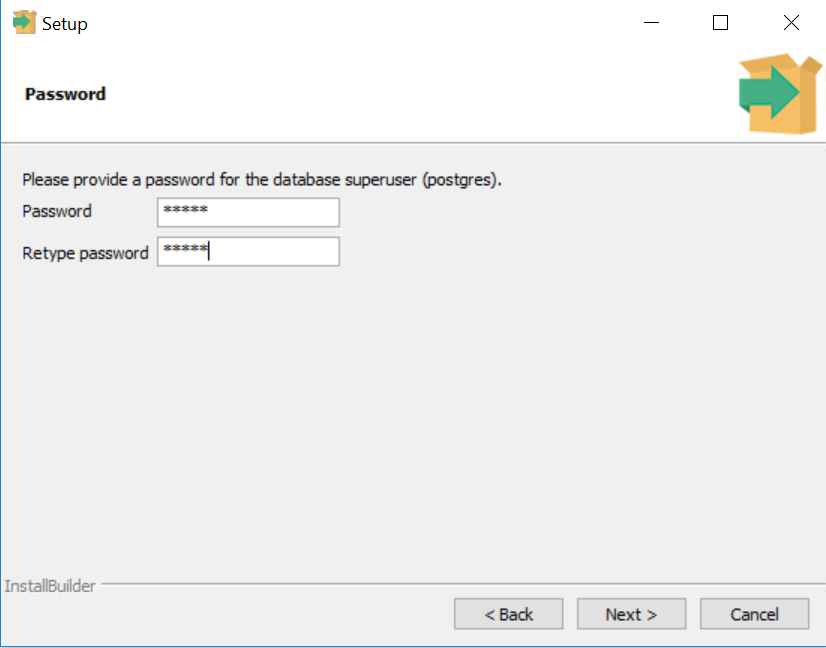


1.2.1.3 Now click on **Download Now** to start the download

**1.2.2 Install Postgres**

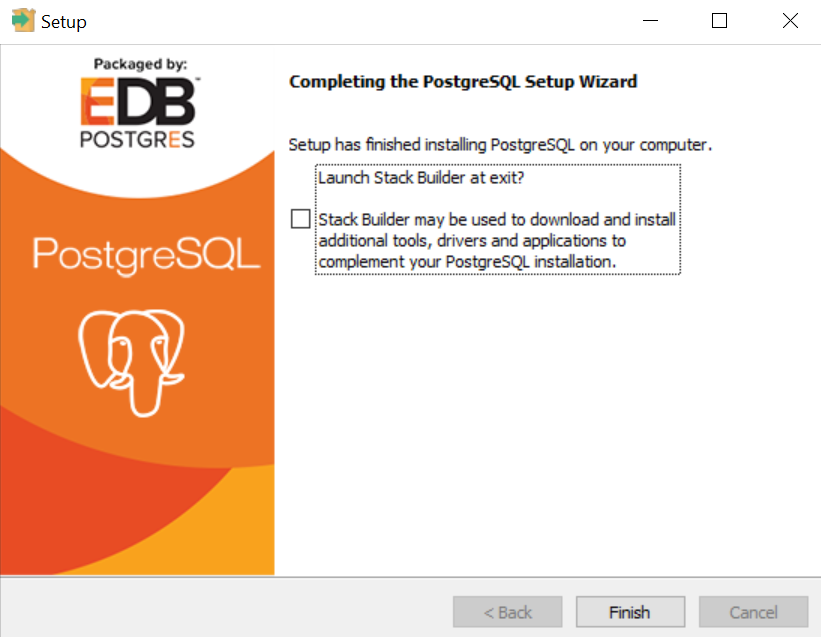
1.2.2.1 Run the installer and click on the **Next** button till you reach the password setup

1.2.2.2 Type the desired **password** to be set for the database root user and click on **Next** button till the setup is installed



1.2.2.3 Note down the root password as it will be used to login later

1.2.2.4 Uncheck the Stack Builder Launcher and click **Finish** button

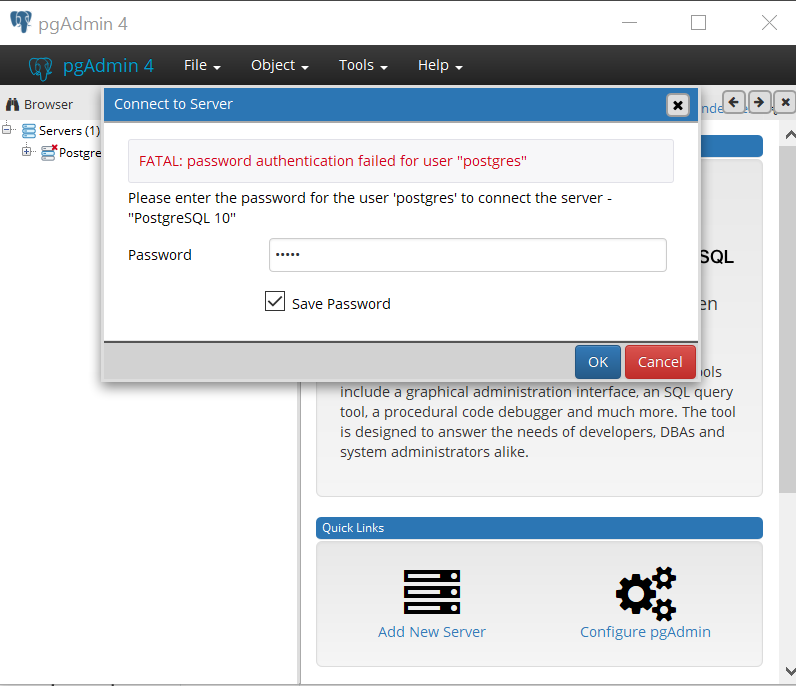


**1.2.3 Setup the Postgres Database**

1.2.3.1 Launch the **pgAdmin4** application from the Windows start menu

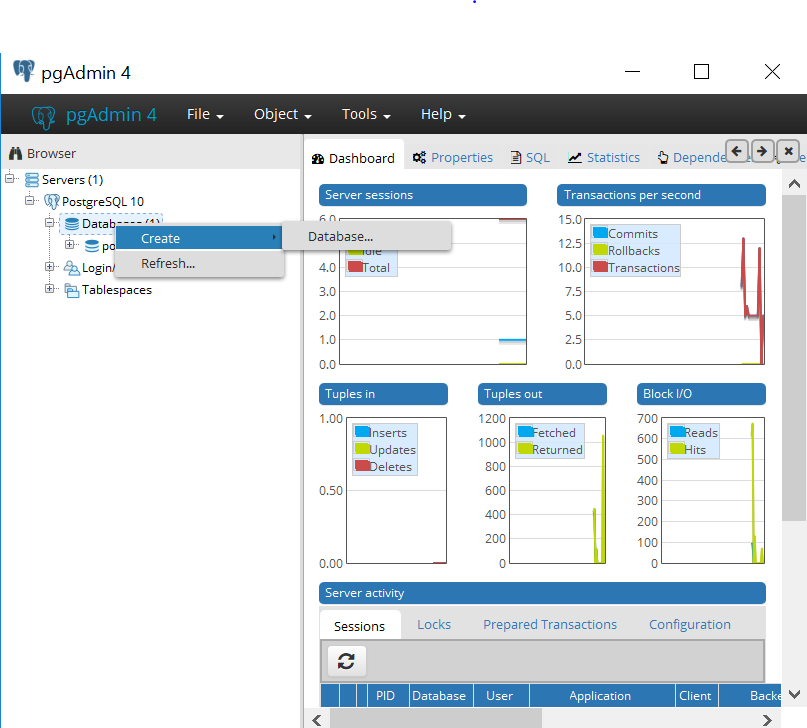
1.2.3.2 Double click on **Server** and **PostgreSQL 10** for the password prompt

1.2.3.3 Enter your root password set before for logging in, click on **Save Password** checkbox and then click **OK** button

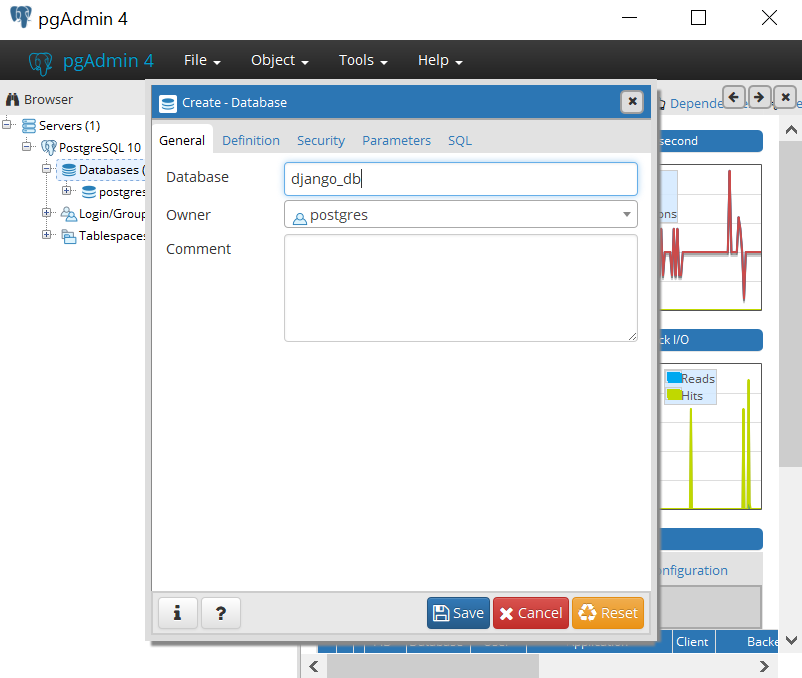


1.2.3.4 Now right click on **Database** inside PostgreSQL 10

1.2.3.5 Select **Create** and then select **Database**



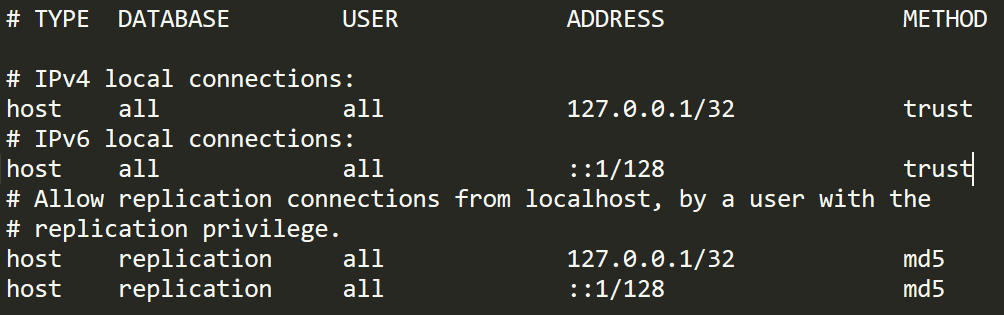
1.2.3.6 Now provide the name for the database as **django\_db** and click on **Save** button



1.2.3.7 Now just simply double click on django\_db you have created for establishing connection. You should see a database connected message at the right bottom corner

1.2.3.8 Now open the **pg\_hba.conf** file under **C:\Program Files\PostgreSQL\10\data** using any notepad

1.2.3.9 On the lines for **IPv4** and **IPv6** change the **METHOD** from **md5** to **trust**

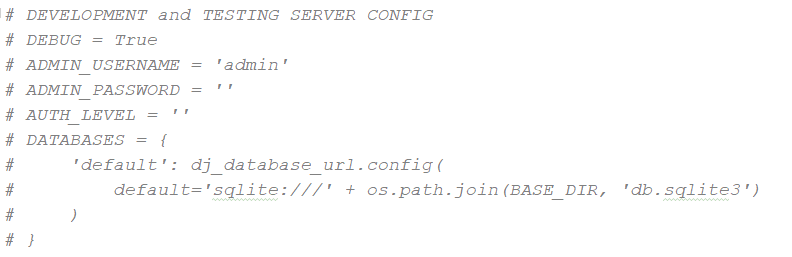


1.2.3.10 In the PowerShell or CMD: **pip3 install psycopg2**

1.2.3.11 Now go to the **settings.py** file inside the otree game and uncomment the lines shown below - (This section is needed only for deploying the lab)



1.2.3.12 Now in the same file comment the lines shown below – (This section is needed only for development and testing purpose)



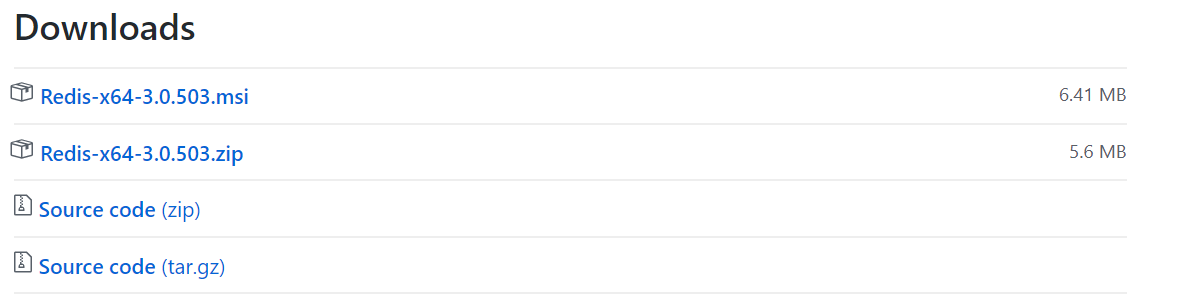
1.2.3.15 Now reset the database. In the PowerShell or CMD go to the respective otree game directory and then type: **otree resetdb.** If all the above steps went well this should work

Now the **PostgreSQL 10 for Windows** database has been installed and setup for our use.

**1.3 INSTALL REDIS**

**1.3.1 Download the Redis from the given link:** [**Redis**](https://github.com/MSOpenTech/redis/releases)

1.3.1.1 Scroll through the bottom of the page for the download section and click on the latest **.msi** release file to start the download

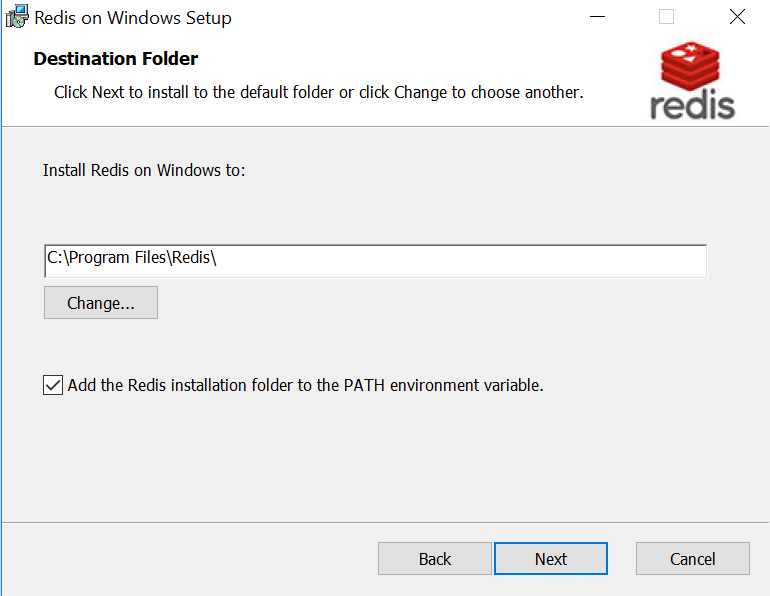


**1.3.2 Install Redis**

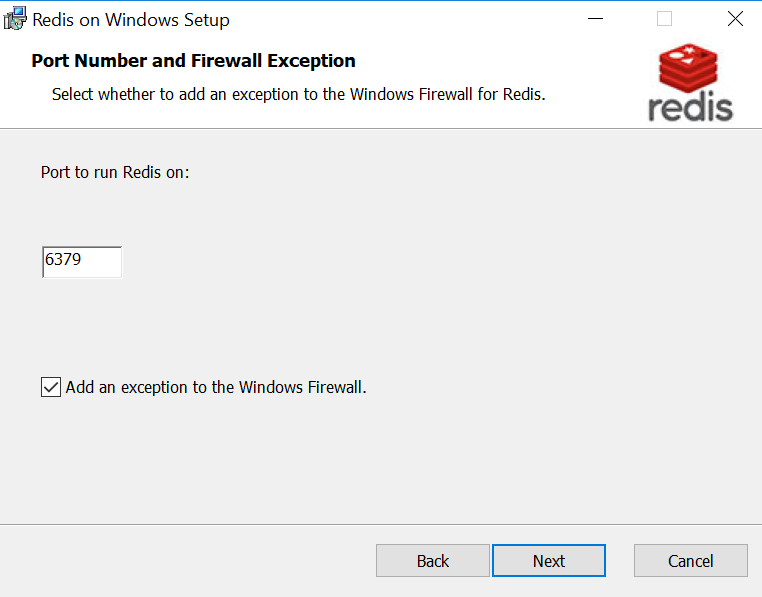
1.3.2.1 Double click on the downloaded installer file

1.3.2.2 Click on **Next** button, **Accept** the agreement and again click on **Next** button

1.3.2.3 Now select the **Add Redis Installation folder to the PATH environment variable** and click on **Next** button



1.3.2.4 Verify the port number as **6379**, select the **Add an exception to the Windows Firewall** and click on **Next** button



1.3.2.5 Now click on **Next** again then **Install** and then **Finish**

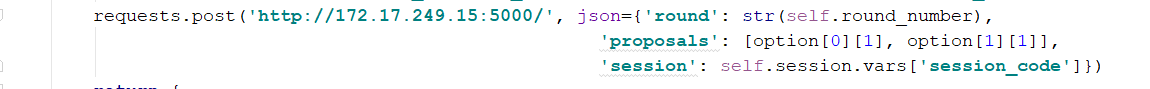
1.3.2.6 In the new PowerShell or CMD type: **redis-cli ping**. It should reply with **PONG**

Now the **Redis for Windows** has been installed and setup for our use.

**1.4 SETUP GAME**

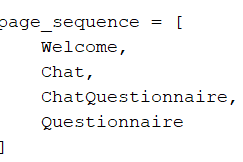
**1.4.1 CHANGES TO BE MADE**

**1.4.1.1 Change IP Address**: In the **views.py** inside **TwoPlayerGame/my\_trust** and **AIvsHumanChat/AIvsHumanChat** folder under the class Send, change the IP Address to the current system IP Address on the line shown below in the snippet



**1.4.1.2 Using Chatbot:** If we are using chatbot, then uncomment two lines **Chat, ChatQuestionnaire** under the **page\_sequence** variable at the last in the **AIvsHumanChat/questionnaire1** **views.py** file. Also change the IP Address in the **send**() and **say**() inside **Chat.html** file in the same game folder under the templates section as shown below

Views.py

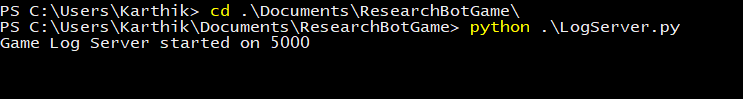


Chat.html



**1.4.2 LAUNCH THE GAME LOG SERVER**

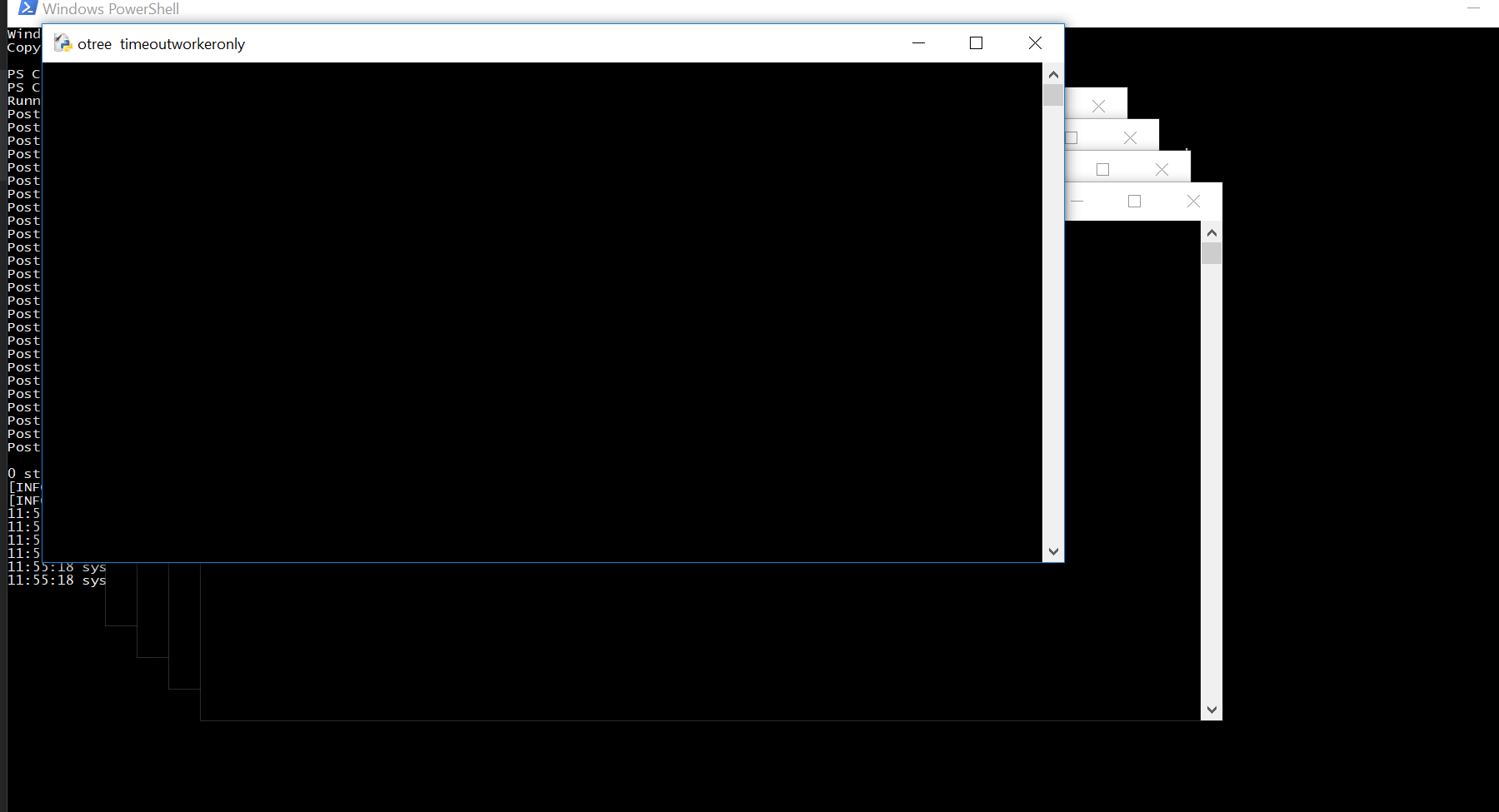
**1.4.2.1 In the PowerShell or CMD:** Go the folder where LogServer.py is present in the project and then type **python LogServer.py**



**1.4.2.2 Similarly in the new PowerShell or CMD:** Go the folder where LogServer.py is present in the project and then type **node chatbot.js**

**1.4.3 RUN THE PRODUCTION SERVER**

**1.4.3.1 In the new PowerShell or CMD:** Go the game project folder and then type **otree runprodserver 0.0.0.0:8000** or **otree runprodserver --port=8000**

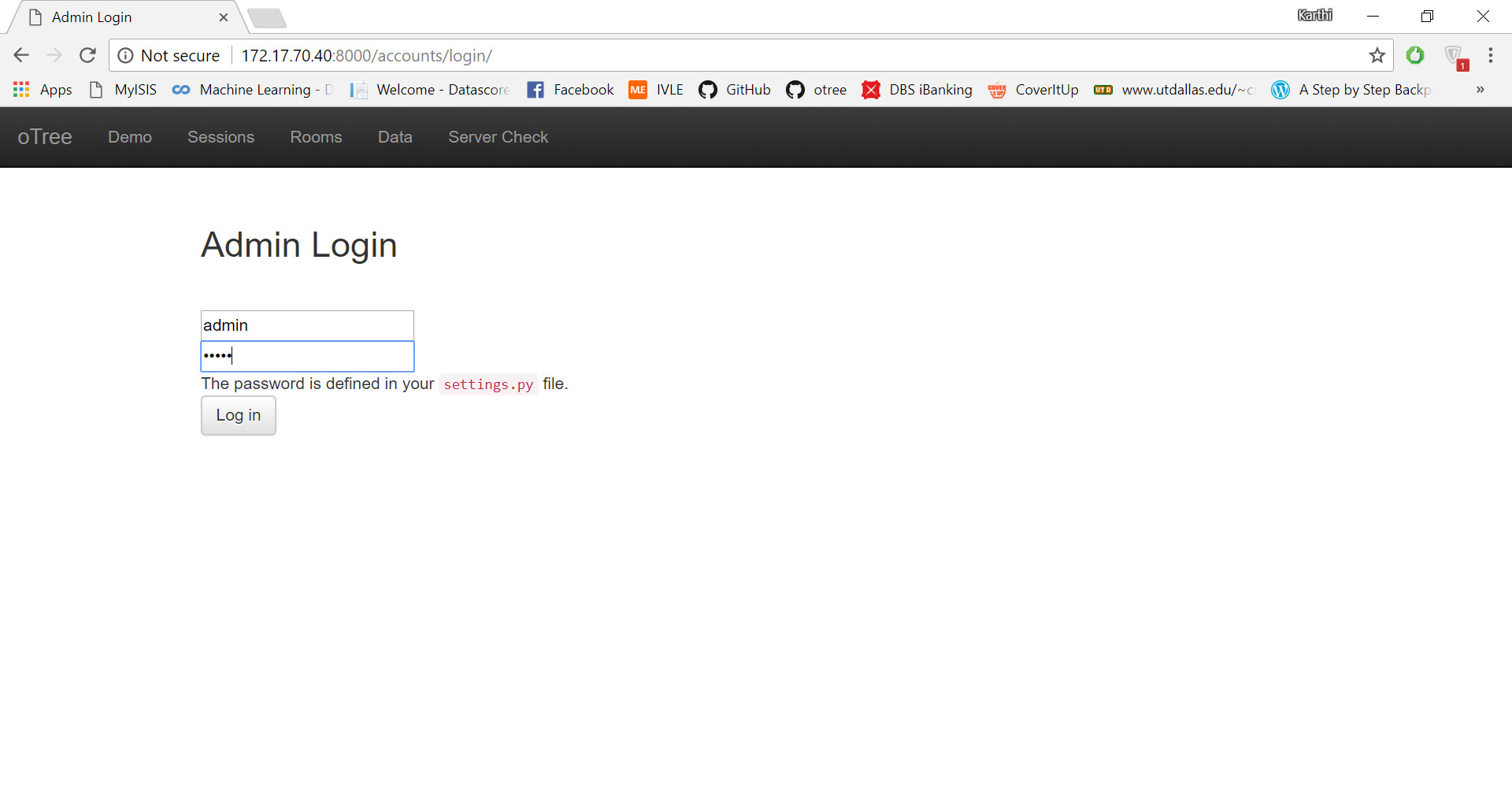


**1.4.4. LAUNCH THE GAME**

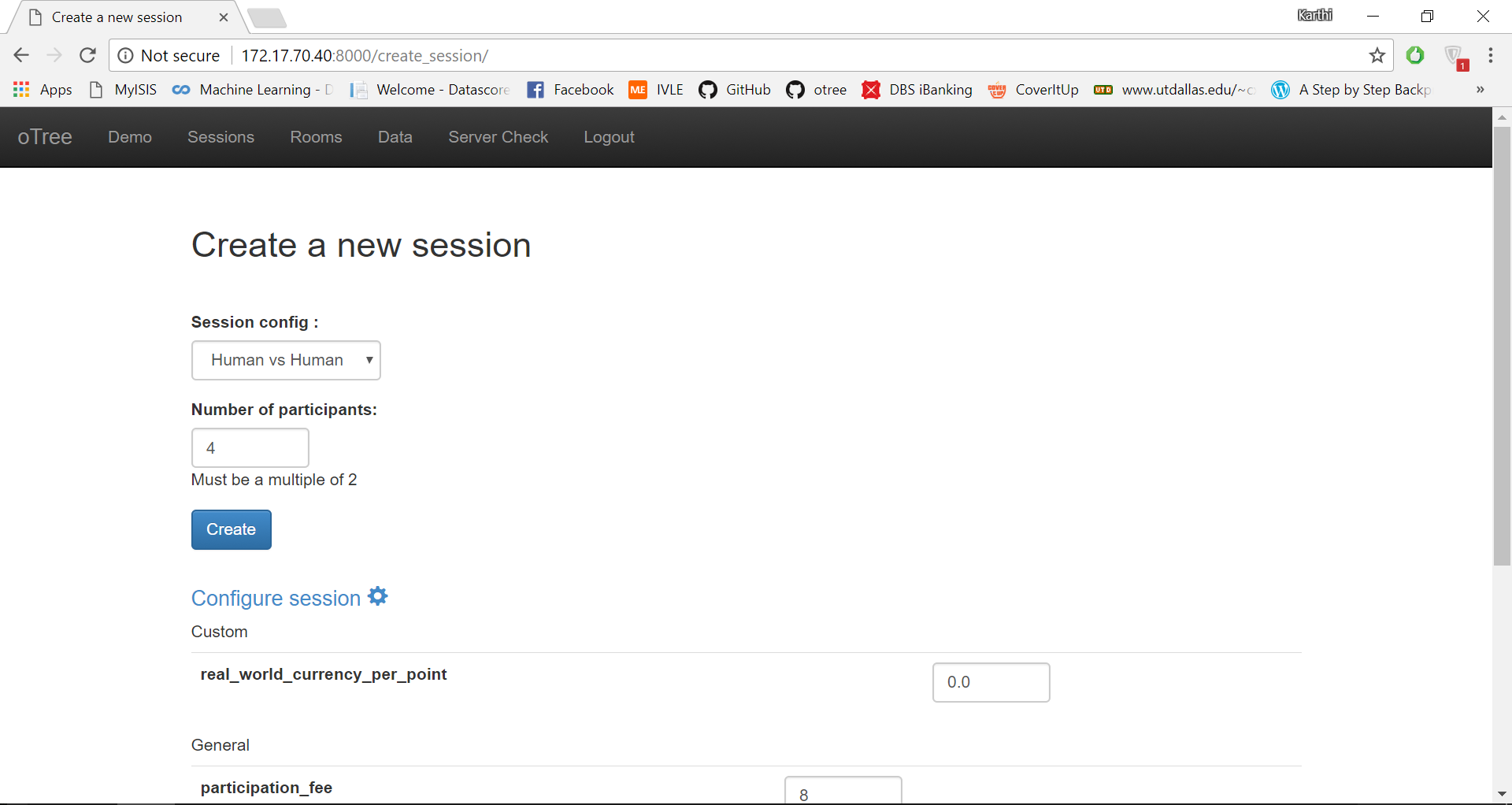
**1.4.4.1 Open the browser and type: http://{ur\_ip\_address}:{server\_port\_running}/accounts/login**.

For example: http://172.17.249.15:8000/accounts/login

**1.4.4.2 Now enter the user name and password** set in the setting.py file. Should be the following: **username: admin** and **password: hello**

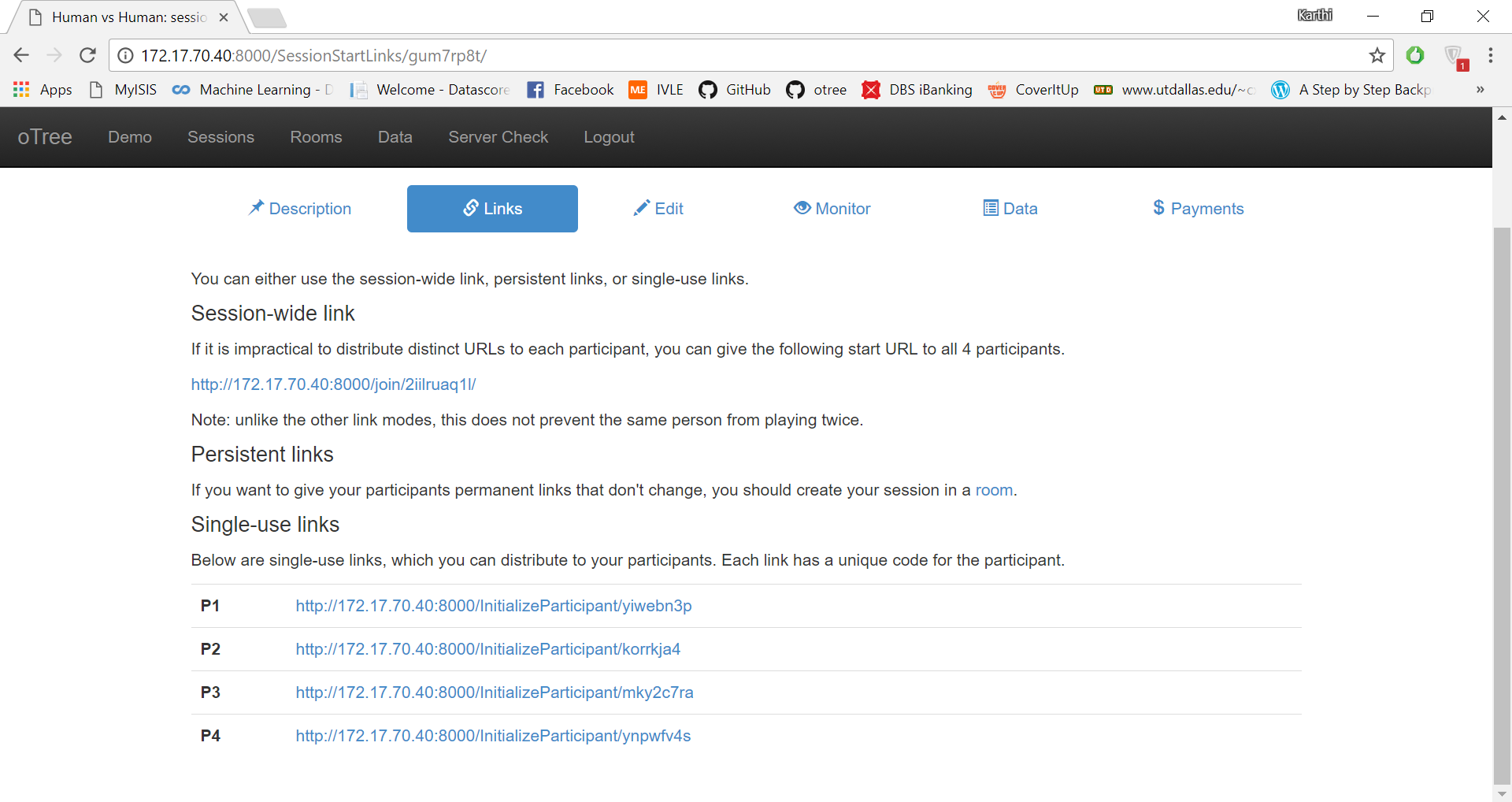


**1.4.4.3 Create Session:** Now click on the **Create Session** button and select the game and input the number of participants as required and Create



1.4.4.4 Now participant links will be generated. Copy the session wide link and add **?participant\_label=PC1** to the link. Here the value for participant label should be unique for each participant.

Example: The link to given to participant 1 would be **http://172.17.70.40:8000/join/2iilruaq1l?participant\_label=PC1** and for participant 2 would be **http://172.17.70.40:8000/join/2iilruaq1l?participant\_label=PC2** and so on for the example shown below.



**1.5 CREATE A WINDOWS FIREWALL RULE**

**1.5.1 Open Windows Firewall**

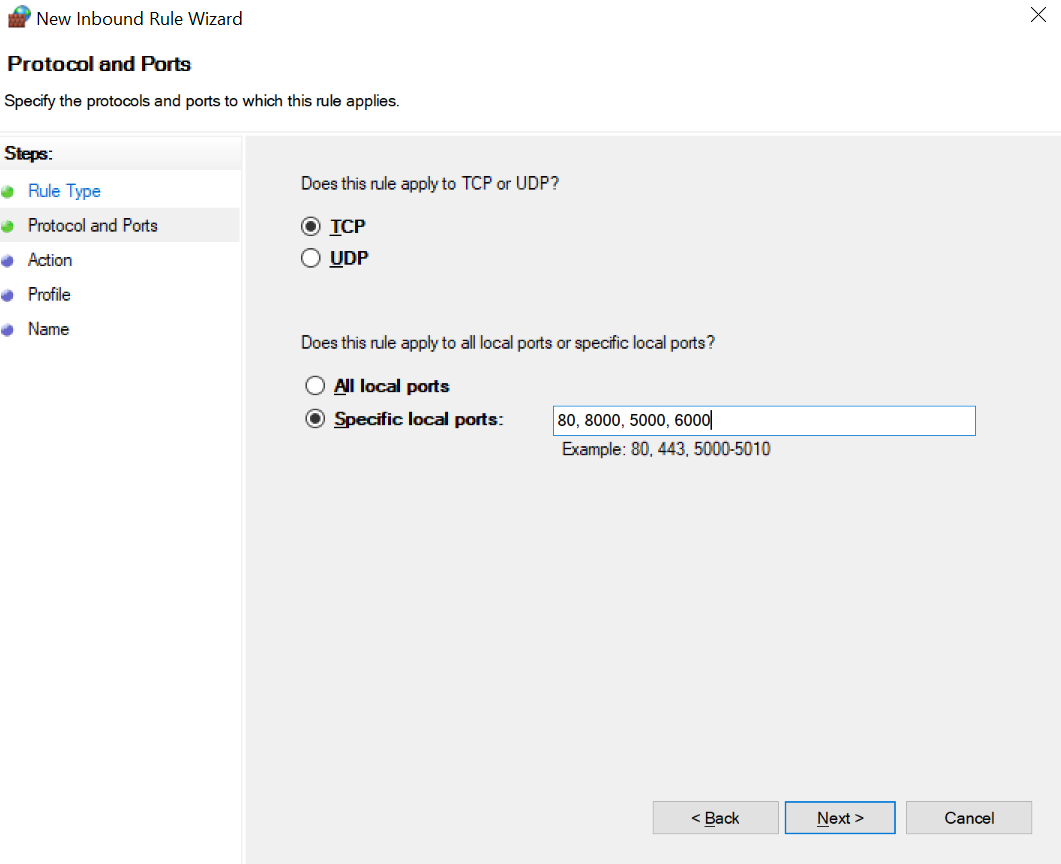
1.5.1.1 In the Windows start search **firewall** and click on **Windows Firewall with Advanced Security** application

**1.5.2 Create a Port Rule**

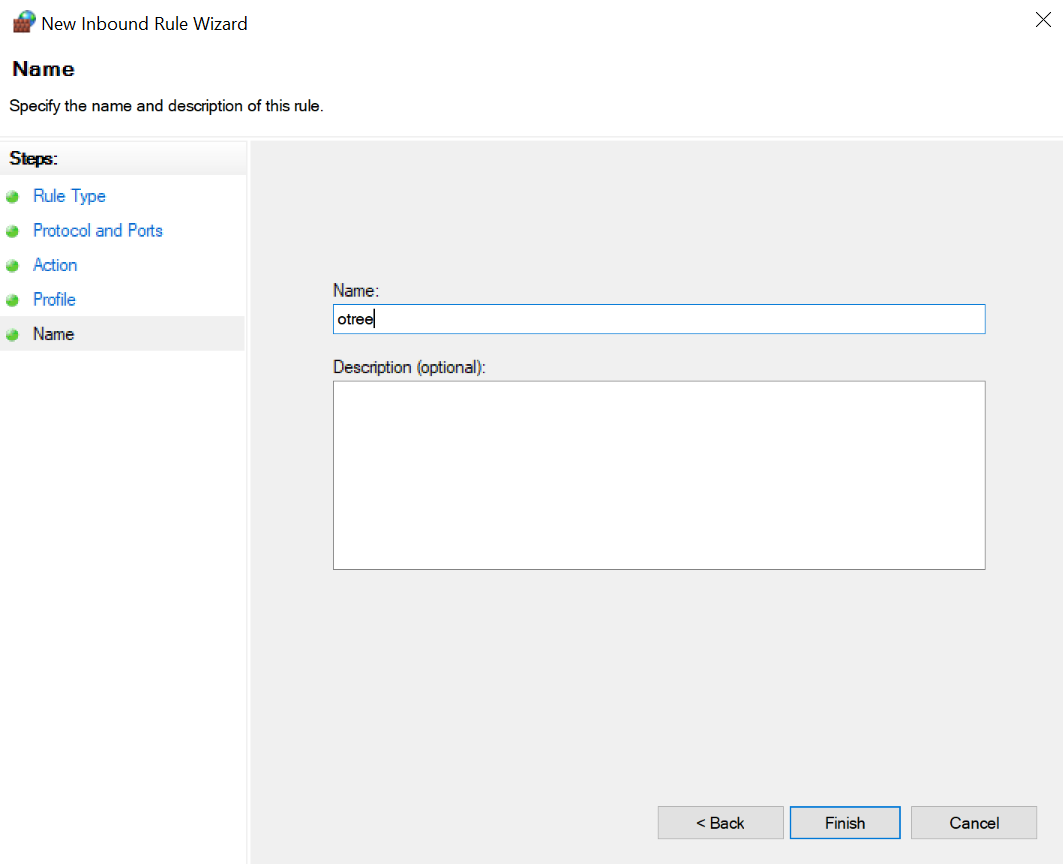
1.5.2.1 Now click on **Inbound Rules -> New Rule**

1.5.2.2 Select **Port** and click **Next** button

1.5.2.3 Type the port numbers: **8000, 5000, 4000** and click **Next** button



1.5.2.4 Click on **Next -> Next**. Provide any name and click on **Finish** button

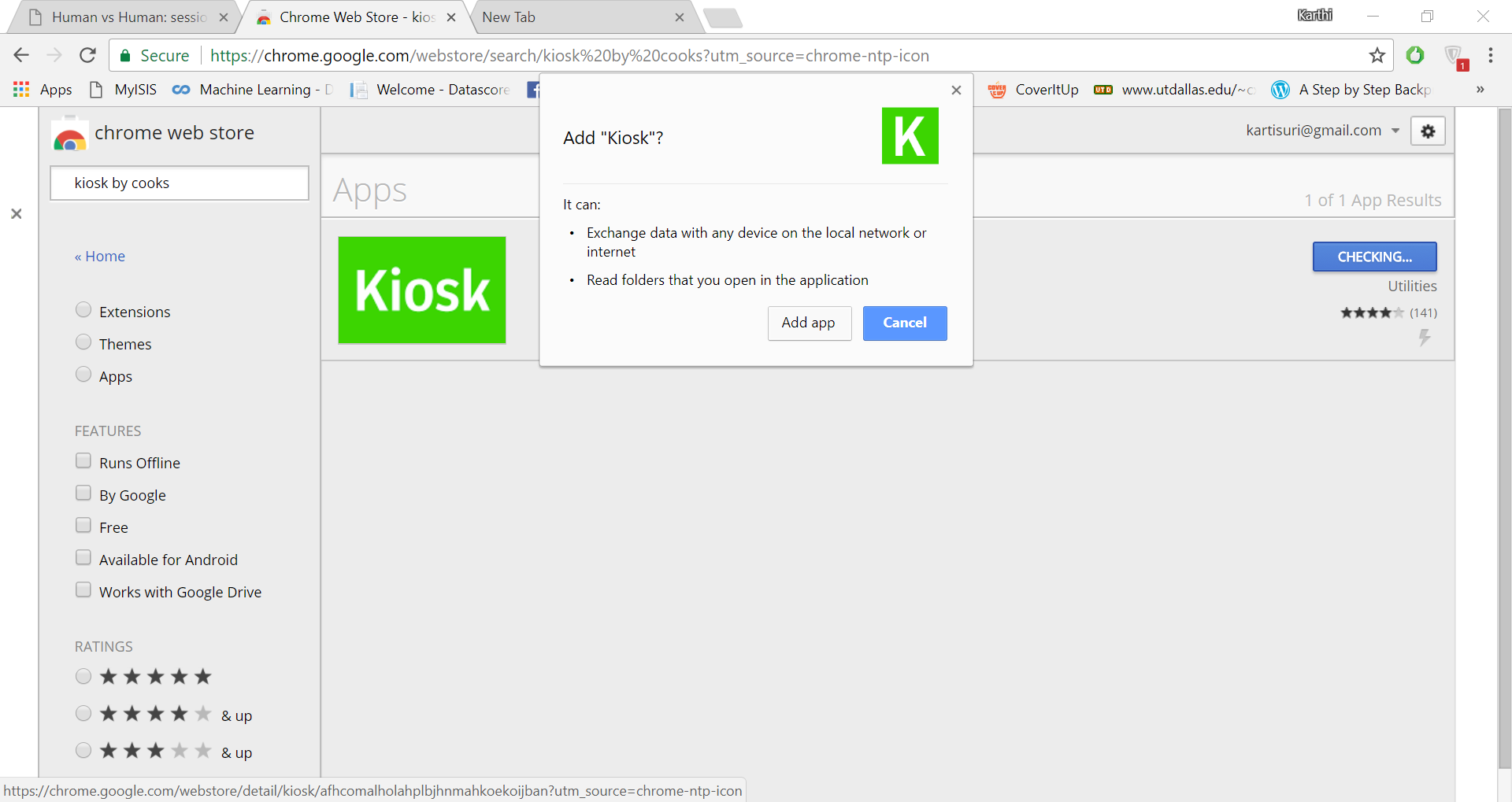


Now new **Windows Firewall Rule** has been created to allow access for other computers to communicate to the Server.

**1.6 LAUNCH THE GAME IN KIOSK ON DIFFERENT COMPUTERS**

**1.6.1 Download the Kiosk Application for Chrome**

**1.6.1.1 Open the Chrome browser**->App Store->Web Store->Type “Kiosk by cooks”->Press Enter -> Add to Chrome -> Add app



1.6.1.2 Launch the Kiosk App and type the correct URL for the participant. Give the admin name as **admin** and password as **hello** and then launch

**Note:**

1. **Alt + F4** to close the Kiosk
2. Next time when you launch kiosk screen will be black because of old URL. Hence press **Ctrl + Alt + a** to get into the setup page. Provide the name **admin** and password you gave at the first i.e. **hello**. Type the new URL and launch again
3. After the game is finished, do not forgot to export the data as excel or csv.