Apache HTTP Server

a documentation Manual

30 Baldovino, Britanny Meneses

05 Cabrera, Gian Luigi Madarang

08 Culanag, Ian Jemuel Nacorda

34 Elegado, Angel Nica Ticwala

16 Ishii, Satoyoshi Lumalang

17 Jumawid, Ronnie Jr. Sabado

25 Supnet, Dominic Ethan Marasigan

TABLE OF CONTENTS

PREREQUISITE

Before anything else, we need to make sure that apache is installed. To do so we begin by typing,

```
webtech@webtech2018:~$ sudo apt-get update
```

```
webtech@webtech2018:~$ sudo apt-get install apache2
```

After making sure that apache is installed we can now start with the Virtual hosts

SETTING UP APACHE VIRTUAL HOST

CREATING A DIRECTORY

Apache has its default web content under the /var/www/html, but for this project we will specify another location for our content for it to be separate from the default content.

We will set individual directories under the /var/www directory.

```
webtech@webtech2018:~$ sudo mkdir -p /var/www/group3a.org/public_html
webtech@webtech2018:~$ sudo mkdir -p /var/www/group3b.org/public_html
webtech@webtech2018:~$ sudo mkdir -p /var/www/group3c.org/public_html
```

Because of these commands we have generated 3 directories

```
webtech@webtech2018:/var/www$ ls
group3a.org  group3b.org  group3c.org  html
```

And under the generated directories contains public_html that would store the content of the project

```
webtech@webtech2018:/var/www/group3a.org$ ls
public_html
```

GRANTING PERMISSIONS

Now that we have constructed the directory for our content we have to change the ownership of the said directory. We can do that by typing the command:

```
sudo chown -R $USER:$USER /var/www/group3a.org/public_html
sudo chown -R $USER:$USER /var/www/group3b.org/public_html
sudo chown -R $USER:$USER /var/www/group3c.org/public_html
```

The $USER represents the current user logged in

```
sudo chmod -R 755 /var/www
```

This command ensures that the web server has the permissions it need to server the content under the directory

CREATING FILES TO BE SERVED

The content we'll use for the three virtual hosts are the advocacy sites that the group has created. For our repository, we used GitHub, and in order to clone that same repository in Ubuntu Server, we must first install Git:

```
webtech@webtech2018:~$ sudo apt-get update
webtech@webtech2018:~$ sudo apt-get install git
```

After installation, we clone the Git repository we have on GitHub by executing the command:

```
$ git clone https://github.com/culanag/group3.git
```

Doing so would clone the repository to our server. The directories are then transferred to the appropriate subdirectories (/var/www/www/group3a.org/public_html, etc..) To do so, we copy the corresponding directory of each:

```
$ sudo cp -r group3/9358B-G3-1 /var/www/www.group3a.org/
```

And then we delete the current public_html which is set earlier by default

```
$ sudo rm -r public_html
```

And finally rename the directory to 'public_html' using the command below

```
$ sudo mv 9358B-G3-1 public_html
```

The process is repeated for the other advocacy sites.

CREATE VIRTUAL HOSTS FILES

Apache has a default virtual host file named 000-default.conf. We would use that as a basis for our virtual host files

```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

Begin by copying the 000-default.conf

```
webtech@webtech2018:~$ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-avai
lable/group3a.org.conf
webtech@webtech2018:~$ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-avai
lable/group3b.org.conf
webtech@webtech2018:~$ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-avai
lable/group3c.org.conf
```

Next is to edit each file

group3a.conf should contain the ff:

```
ServerName www.group3a.org
ServerAlias group3a.org
ServerAdmin webmaster@localhost
DocumentRoot /var/www/group3a.org/public_html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

group3b.conf should contain the ff:

```
ServerName www.group3b.org
ServerName group3b.org
ServerAdmin webmaster@localhost
DocumentRoot /var/www/group3b.org/public_html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

group3c.conf should contain the ff:

```
ServerName www.group3c.org
ServerAlias group3c.org
ServerAdmin webmaster@localhost
DocumentRoot /var/www/group3c.org/public_html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

Enable the Virtual Host File

After creating the virtual host file we used the a2ensite tool to enable the sites

```
webtech@webtech2018:~$ sudo a2ensite group3a.org.conf
webtech@webtech2018:~$ sudo a2ensite group3b.org.conf
webtech@webtech2018:~$ sudo a2ensite group3c.org.conf
```

Then disable 000-default.conf

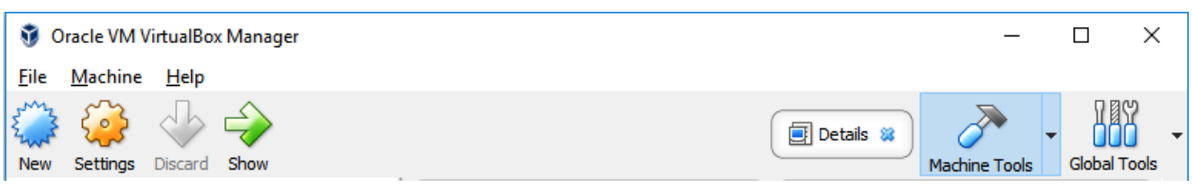
```
webtech@webtech2018:~$ sudo a2dissite 000-default.conf
```

Next is to reload the server

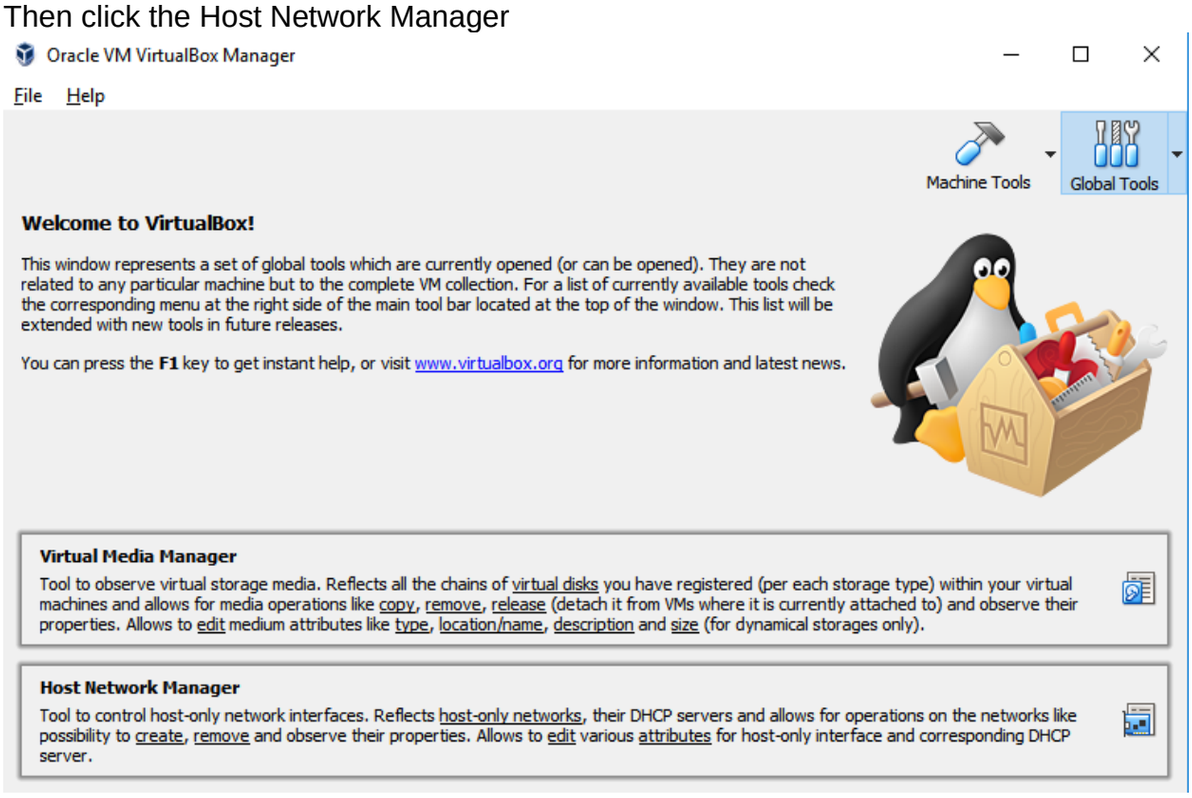
```
webtech@webtech2018:~$ sudo service apache2 reload
```

Set up local host file

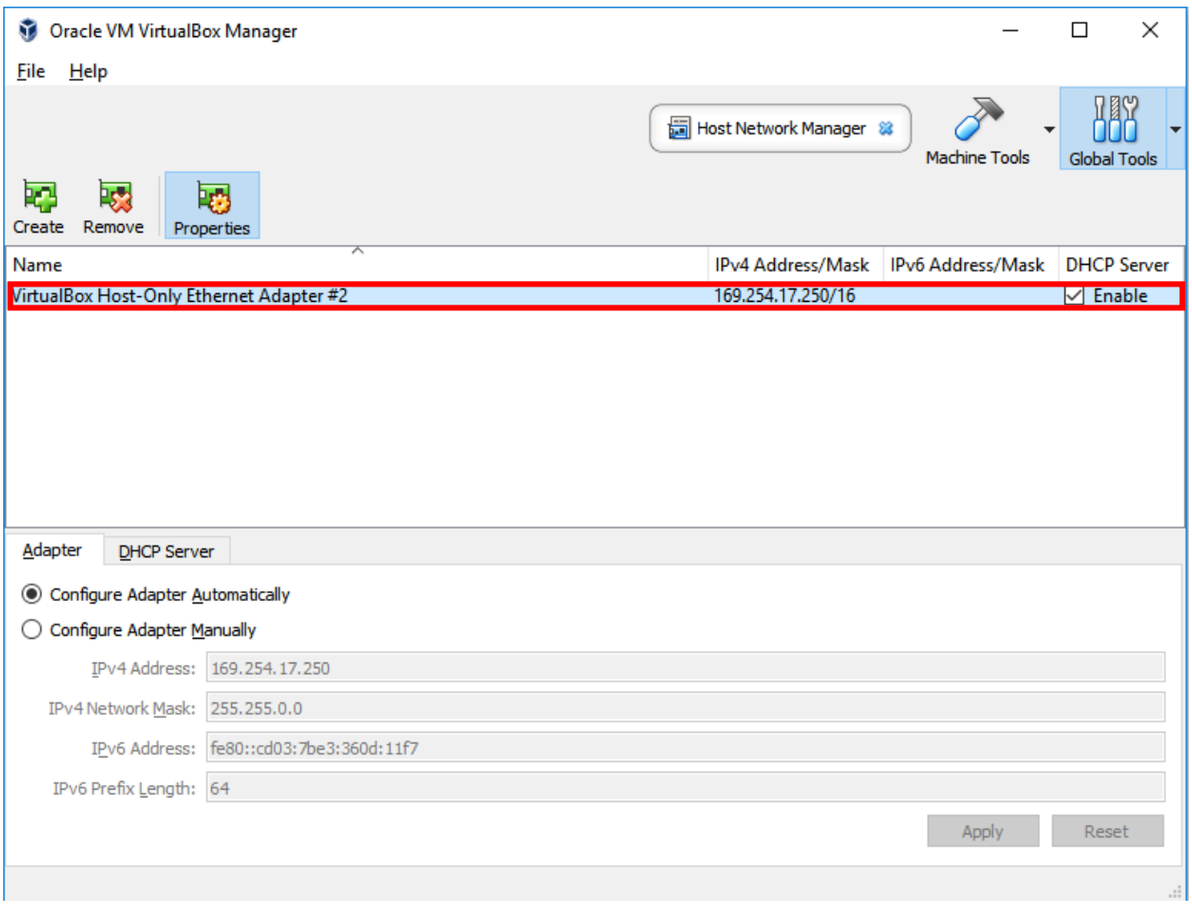
If you are using the option Host-only Adapter then make sure to go to global tools found on the upper right of your vm



Then click the Host Network Manager



Afterwards you will reach this section

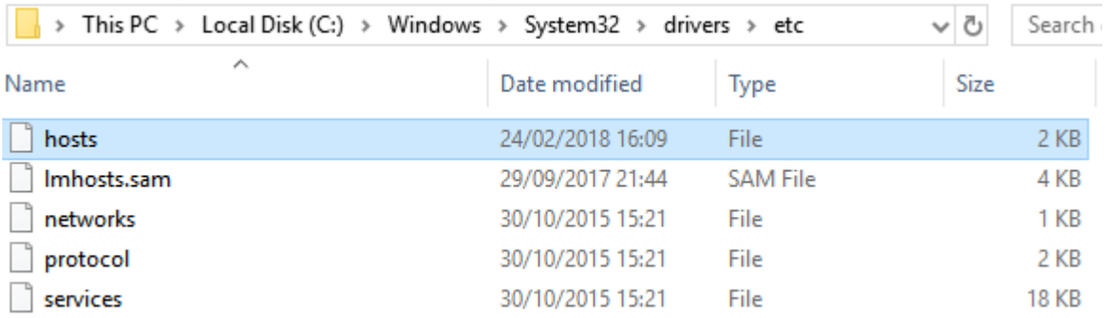


Make sure to create a Host-Only network and enable the DHCP server so that you will be given an IP address

In the VM type hostname to check IP address of your machine

```
webtech@webtech2018:~$ hostname -I
169.254.17.249
```

After knowing the needed info open your host file

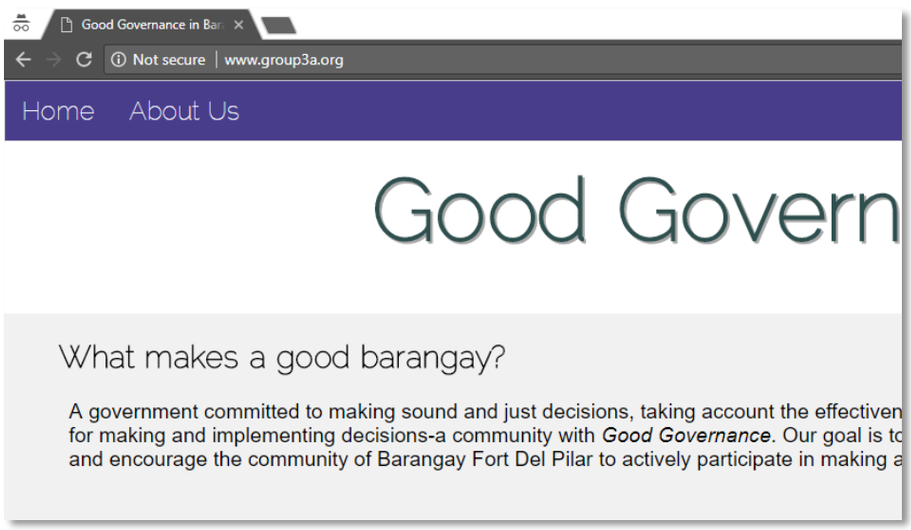


After opening your host file edit it and add the IP address of your VM and the domain name

```
169.254.17.249 www.group3a.org
169.254.17.249 www.group3b.org
169.254.17.249 www.group3c.org
```

Testing the url

After editing your host file test it by opening your browser and input the domain name you specified in the ServerName in your virtual host file



If successful, the advocacy website shall now display when the domain name is typed in the browser.

ENABLING COMPRESSION

In order to serve compressed content to clients, we modify the configuration file of our website. We begin by typing:

```
$ sudo nano /etc/apache2/sites-available/www.group3a.org.conf
```

This will execute nano and take us to the .conf file, we then add the directive

```
AddOutputFilterByType DEFLATE text/html text/css
```

to compress html and css content types.

```
<VirtualHost *:80>
        ServerAdmin 2160301@slu.edu.ph
        ServerName www.group3a.org
        ServerAlias group3a.org
        DocumentRoot /var/www/www.group3a.org/public_html

        AddOutputFilterByType DEFLATE text/html text/css

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

And then we save the file, reload the server to apply (as specified before).

ENABLING CACHING
//gian ethan
CONTENT NEGOTIATION

ACCESS CONTROL

ENABLING SERVER-SIDE INCLUDES

The first thing needed to be done is to enable the Includes module. It can be enabled with the command:

```
webtech@webtech2018:~$ sudo a2enmod include
```

We then open the configuration file:

```
$ sudo nano /etc/apache2/sites-available/webtech1.ssi.org.conf
```

We add these lines inside the configuration file; take note that Includes havs to be added at the end of Options. Another thing to consider is that the AddType and AddOutputFules directives for .shtml are needed for older versions of Ubuntu.

```
<Directory /var/www/webtech1.ssi.org/public_html>
        Options Indexes FollowSymLinks MultiViews Includes
        AllowOverride None
        Order allow,deny
        allow from all
        AddType text/html .shtml
        AddOutputFilter INCLUDES .shtml
</Directory>

<IfModule mod_dir.c>
        DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
```

Again, for changes to take effect, the server must be restarted.

We will then be testing if it works. We'll be creating an SSI test file with the contents:

```
<html>
<head>
<title>SSI Test Page</title>
</head>
<body>
<!--#echo var="DATE_LOCAL" -->
</body>
</html>
```

this would be save in /var/www/ssi-test.shtml
Typing http://127.0.0.1/ssi-test.shtml
SSL/TLS ENCRYPTION

Create virtual hosts mapped to the domain names webtech1.negotiage.org

Begin by creating a directory that will store the content of the domain.

```
webtech@webtech2018:~$ sudo mkdir /var/www/webtech1.negotiate.org/public_html
```

After typing that go to the created directory by typing

```
webtech@webtech2018:~$ cd /var/www/webtech1.negotiate.org/public_html
```

Next is to create files to be served do this by typing

```
webtech@webtech2018:/var/www/webtech1.negotiate.org/public_html$ sudo nano content.html
```

And another one

```
webtech@webtech2018:/var/www/webtech1.negotiate.org/public_html$ sudo nano content.txt
```

And here is the content of the created files

```
<!DOCTYPE html>
<html>
<head>
<title>
content.html
</title>
<body>
<h1>This is content.html</h1>
</body>
</head>
</html>
```

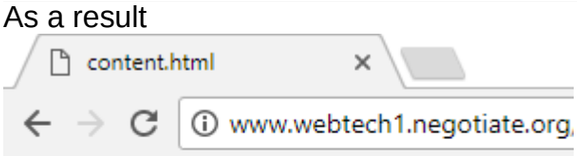
```
This is content.txt
```

Next would be configuring the host hosts to enable clients to negotiate with the servers, using the HTTP Accept header
We start by creating a virtual host file as shown at the first part of this documentation.

This is the created virtual host file

```
 <Directory /var/www/webtech1.negotiate.org/public_html/>
        <IfModule mode_negotiation.c>
        Options Indexes FollowSymLinks MultiViews
                        DirectoryIndex content
        AddType text/html .html
        AddType text/plain .txt
        DirectoryIndex Priority text/html text/plain
        </IfModule>
 </Directory>
```

As a result



This is content.html

Next would be configuring the host hosts to enable clients to negotiate with the servers, using the HTTP Accept-language header

First is to edit the global configuration file of apache, to do so type

```
webtech@webtech2018:~$ sudo vi /etc/apache2/mods-available/negotiation.conf
```

Then comment out the following to allow our virtual host to have it's own configuration

```
<IfModule mod_negotiation.c>

        # LanguagePriority allows you to give precedence to some languages
        # in case of a tie during content negotiation.
        #
        # Just list the languages in decreasing order of preference. We have
        # more or less alphabetized them here. You probably want to change this.
        #
        #LanguagePriority en ca cs da de el eo es et fr he hr it ja ko ltz nl nn no pl pt pt-BR ru s
v tr zh-CN zh-TW

        #
        # ForceLanguagePriority allows you to serve a result page rather than
        # MULTIPLE CHOICES (Prefer) [in case of a tie] or NOT ACCEPTABLE (Fallback)
        # [in case no accepted languages matched the available variants]
        #
        #ForceLanguagePriority Prefer Fallback

</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Afterwards save by typing :x and restart your apache. Make sure to configure the virtual host file of your domain

This is the added configuration in the webtech1.negotiation.org.conf
We made use of the MultiView for our negotiation with the client we specified the Directory Index as language,html so that the server will serve the file and find the specified extension appropriate for the client's browser language settings

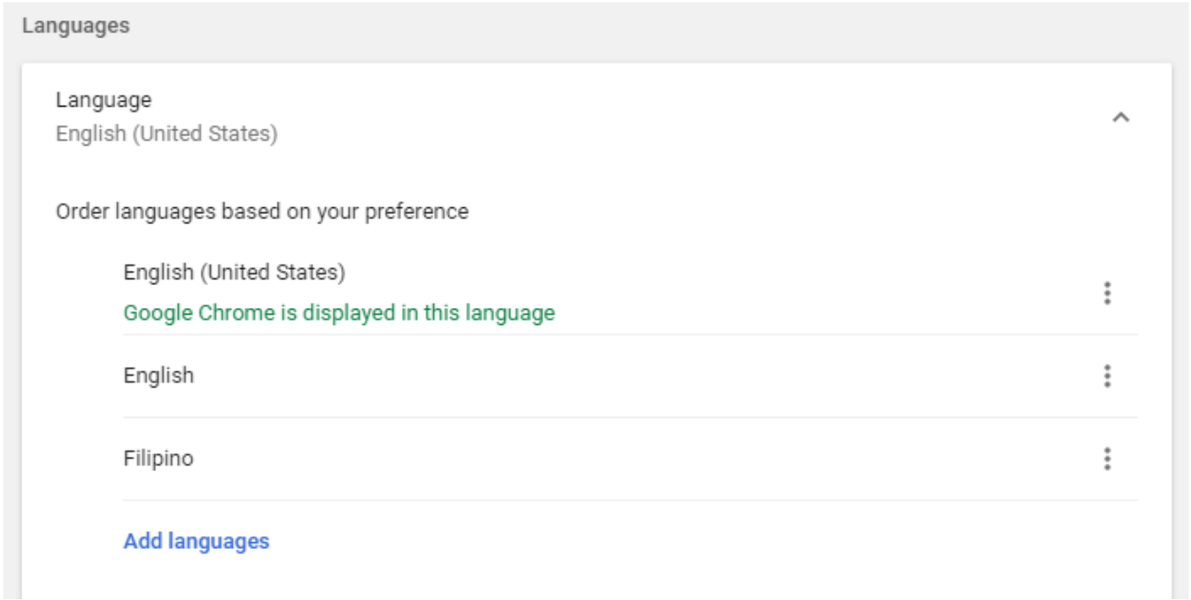
```
<Directory /var/www/webtech1.negotiate.org/public_html/>

        Options Indexes FollowSymLinks MultiViews
                        #DirectoryIndex language.html
        AllowOverride None
        Order allow,deny
        allow from all
</Directory>
```

And this is the result of our configuration

This is our English version of our website

We can add languages in our client in our settings in chrome, also we can prioritize our language



Here we have a client web browser that prioritizes English as the language of the web resources that will be served to the client