**WEBSITE BLOCKER**

**A PROJECT REPORT**

**Submitted by**

**SUBHASHIS TRIPATHY [Reg No: RA2112703010020]**

**AYYAPPA REDDY[RegNo:RA2112703010030]**

**CM NIRMAL [ RA2112703010011]**

**Under the guidance of**

**Dr. R. LAKSHMINARAYANAN**

**(Assistant Professor, Department of Networking and Communications,School of**

**Computing)**

**in partial fulfillment of the requirements for the degree of**

**BACHELOR OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE AND ENGINEERING**

**with specialization in CYBER SECURITY AND DIGITAL FORENSIC**



**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,**

**KATTANKULATHUR- 603 203**

***SRM INSTITUTE OF SCIENCE AND TECHNOLOGY***

***KATTANKULATHUR - 603203***

***(Under Section 3 of UGC Act, 1956)***

***BONAFIDE CERTIFICATE***

***Certified that 21CSC203P project report titled “WEBSITE BLOCKER" is the bonafide work of “SUBHASHIS TRIPATHY "[Reg No: RA2112703010020] and AYYAPPA REDDY [Reg No: RA2112703010030]” and NIRMAL [Reg No: RA2112703010011], who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.***

***Dr. R. LAKSHMINARAYANAN DR. ANNAPURANI PANAIYAPPAN .K***

***GUIDE HEAD OF THE DEPARTMENT***

***Assistant Professor Professor***

***Networking and Communications Networking and Communications***

***INTERNAL EXAMINER EXTERNAL EXAMINER***

***Abstract***

**Website Blocker** is a program that blocks access to websites either permanently or on a regular basis. We can ban all websites from undesirable categories to use the internet securely.

The goal of the Website Blocker python project is to ban certain websites from being accessed from any device. This project will assist the user in avoiding distractions by preventing them from accessing web pages on their smartphone.

The user may input numerous websites to block in this Python Website Blocker Project, and then hitting the block button will verify the condition that if the website has already been banned, it will print 'already blocked', else it will block all of the websites and print 'blocked'.

The objective of Python website blocker is to block some certain websites which can distract the user during the specified amount of time.

In this, we will block the access to the list of some particular websites during the working hours so that the user can only access those websites during the free time only.

The working time in this python application is considered from 9 AM to 5 PM. The time period except that time will be considered as free time.

***TABLE OF CONTENTS***

**ABSTRACT**

**1. INTRODUCTION**

**2. PROJECT ANALYSIS**

**2.1 PROBLEM STATEMENT**

**2.2 PROPOSE OF THE PROJECT**

**2.3 ADVANTAGES**

**2.4 DISADVANTAGES**

**3. OBJECTIVES**

**4. PROGRAM ARCHITECTURE OF WEBSITE BLOCKER**

**4.1** About the program

**4.2** Program Architecture

## **4.3** Configuration

## 4.4 Requirements

**5.DIAGRAMS**

**5.1 USE CASE DIAGRAM**

**5.2** CLASS DIAGRAM

5.3 ARCHITECTURE DIAGRAM

**6. FLOW CHART OF THIS PROJECT**

**7. WORKING OF THIS PROJECT**

**8. MODULES**

**9. CODE EXPLANATIONS**

***10*. MAIN PROGRAM**

***CHAPTER 1***

***INTRODUCTION***

​

A website blocker is a software program that intentionally restricts access to particular sites. The purpose of website blockers is to deter users from using distracting websites instead of completing their work. Website blocks can be installed on a particular browser as an extension, or sometimes on a computer hard drive. A typical blocker allows users to input particular links into a list. All standard browsers have this feature built, which can usually be access through the advanced browser settings menu. It is also possible to configure an IP address the same way, if desired, blocking a particular site from all computers that share the same network.

    It has been empirically shown that when people are attempting to work, but have to inhibit performing a desired task, they work less productively (Bucciol, Houser, & Piovesan, 2013). This project showed with moderate effect sizes that people make more mistakes, and the mistakes are higher impact, when they are working while avoiding a desired activity compared to controls. This was shown with adults (Bucciol, Houser, & Piovesan, 2013), as well as with children (Bucciol, Houser, & Piovesan, 2011).

   A preliminary randomized controlled study looked at blocking websites while having students complete tedious, attentionally demanding tasks (Marotta & Acquisti, 2017). Results were measured by what volume of work students completed. Marotta and colleagues found students who were told they would have their favorite websites blocked (Facebook in this case) had higher overall productivity by a significant margin. Students who had the option to use an installed blocker, but did not have it pre-set up with their websites for them did not show the same effects of increased productivity compared to controls. However, it should be noted that there was an inverse relationship to student satisfaction; students that had the blocker forced on them were less happy afterwards than students without (Marotta & Acquisti, 2017).

A related field that has not been explored is the use of Ad-Blocking software. Ad Blockers do not need to be programmed with specific sites, instead automatically blocking various commercials and banners one encounters when surfing the internet. The potential use of ad blockers as an assistive technology is unknown, as current literature in the area is marketing focus (IE trying to make ads more distracting, not less) (Hsieh & Chen, 2011).

***CHAPTER 2***

***PROJECT ANALYSIS***

***2.1 PROBLEM STATEMENT***

This is real world program which blocks certain distracting website like Facebook, Youtube etc during your work hours.

What we are going to in this program is that we will pass the link of websites which you think is distracting and the time that you are working on your computer and program will block those website.

***2.2 PROPOSED OF PROJECT***

* To block the website we need to use website blocker
* To block the unwanted website
* To block corrupted website
* To block distracting website for sometime.

***2.3) ADVANTAGES***

1) Unwanted Sites - One major problem with websites is that a large amount do not feature what you might call G-rated content; i.e. content for all ages. Plus for many people, the content is explicit, hateful, hurtful and very much unwanted.

Examples are porn sites, gambling sites, sites for people to voice their dislike over other races, religions, sexual orientation and more. Not only do you get crude language and x-rated photos on the sites, you can pick up computer bugs or unwanted cookies from the sites, attract nuts and place yourself in danger if you interact with others there. So for all ages, these sites can be dangerous and unwanted.

* Fun Sites - Many fun sites like those offering interesting interactions with forums, fun safe games like cards or gaming like the kinds played with Xbox or Nintendo systems, are great - -but time consuming. This may be alright for time off, but for students who should be studying and adults and teens who should be working, they can cause:

-         Loss in work production

-         Loss in homework time

-         Loss in family time

-         Low grade

-         Low work reviews = lower pay

Additional stress and loss in personal rest-and-relation time also results. So time needs monitored on these types of sites.

* Unsafe Sites - Some websites might look like they have a lot to offer, but once you click them on, you can run into technical trouble. For example, you might get scripts that automatically try to take over your computer in different ways, banners and pop-up ads that try and do insert tracking devices into your systems and much more.

***2.4 DISADVANTAGES***

* outside interference in the source code of the page, possible interception of data and traffic.
* slowing down the website loading speed - the ad blocker needs to process and block advertising elements in the code.
* slowdown of the browser speed, freezes may occur in rare cases, and there may also be an increased use of PC resources.
* the need for timely update of extensions, otherwise vulnerabilities in the browser security system may appear.
* the need to pay in case of using paid extensions.
* some sites are ad-supported or their owners earn mostly only from ads, so they may show notifications like “you seem to be using a blocker” and ask you to disable the extension or add their site to the list of exclusions.
* the owners of some sites calculate the use of blockers and block the site or part of its content for such visitors, demanding to disable the blocker - in return, they offer disabling ads for money, that is, paid access to their content or viewing with ads.

***CHAPTER 3***

***OBJECTIVES***

* The objective of Python website blocker is to block some certain websites which can distract the user during the specified amount of time.
* In this, we will block the access to the list of some particular websites during the working hours so that the user can only access those websites during the free time only.
* The working time in this python application is considered from 9 AM to 5 PM. The time period except that time will be considered as free time.
* To block unwanted website
* To block corrupted website to safe or secure your system.

***CHAPTER4***

***PROGRAM ARCHITECTURE OF WEBSITE BLOCKER***

***4.1) About the program :***

What we are going to in this program is that we will pass the link of websites which you think is distracting and the time that you are working on your computer and program will block those website.

***4.2) Program Architecture:***

1. Every system have **host** file whether it is Mac, Windows or Linux.  
   **Host** file in Mac and Linux :

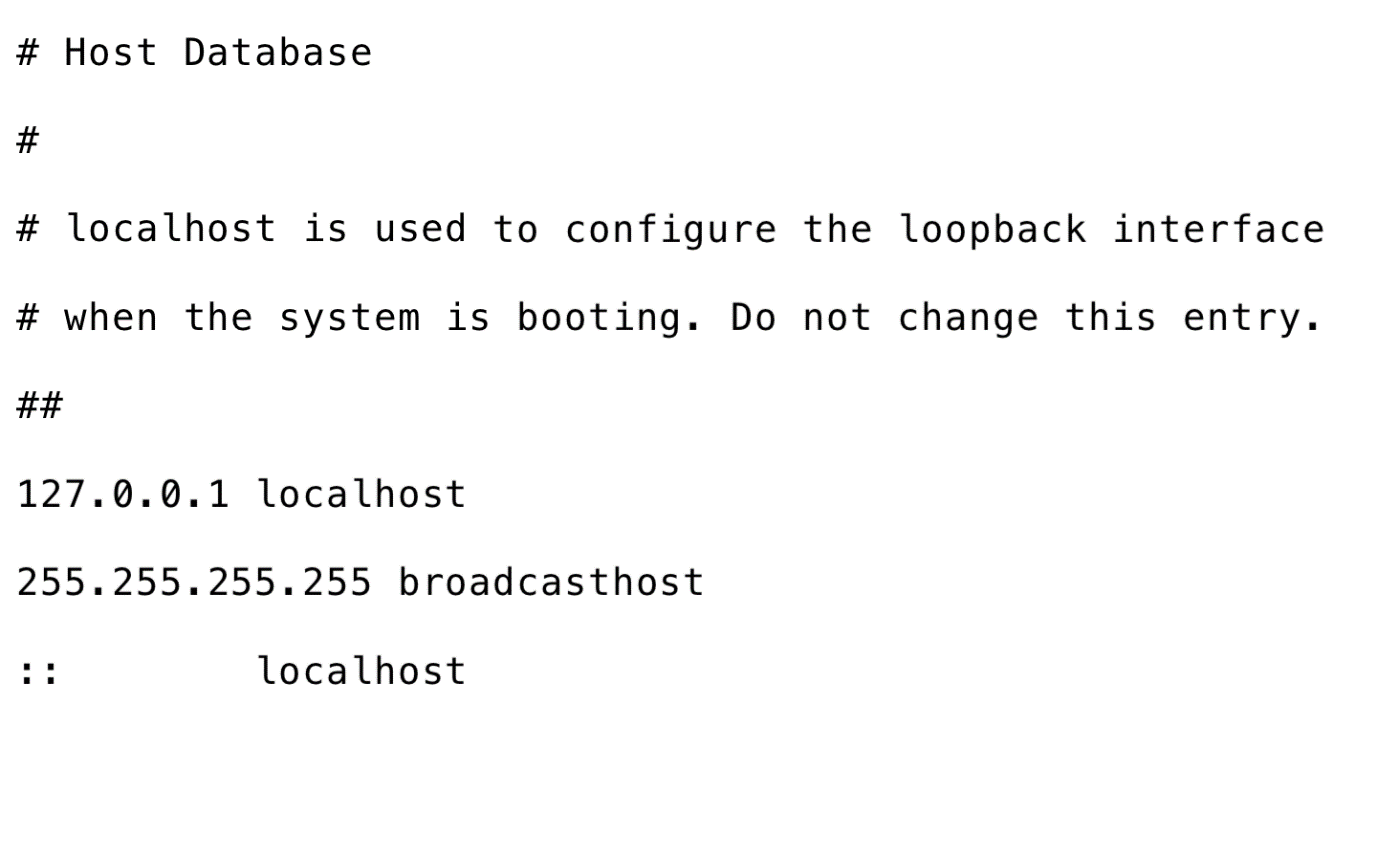
/etc/hosts

**Host** file in Windows:

C:\Windows\System32\drivers\etc

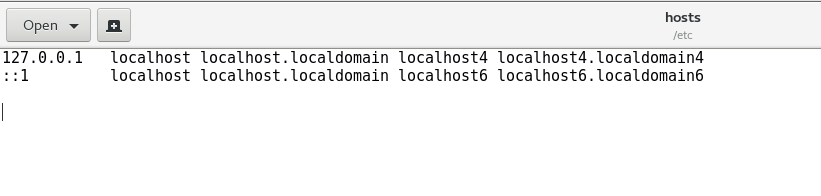
1. **Working of host file:**Host is an operating system file which maps hostnames to IP addresses. In this program we will be mapping hostnames of websites to our localhost address. Using python file handling manipulation we will write the hostname in hosts.txt and remove the lines after your working hours.

**Host file in Mac:**



***4.3) Configuration***

Since the hosts file contains the mapping between the host names and IP addresses, let's look at the content of our hosts file first stored as **/etc/hosts** as we use CentOS 7 (Linux).



As we can see in the above image, the mappings are saved for the localhost with the IP address 127.0.0.1.

Similarly, we can redirect any hostname back to the localhost IP (127.0.0.1). It will block the access to that hostname and redirect that to localhost on each request.

For testimony, let's edit the content of hosts file and add some of the following lines to it. To make changes to the **/etc/hosts** file, we will need to change its permissions.

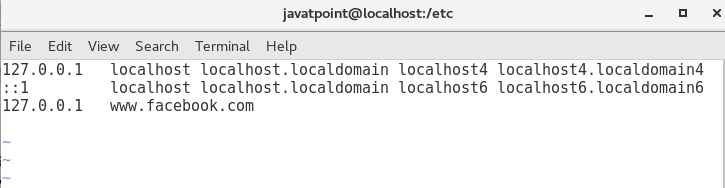
Run the following command on the terminal to change the permissions.

1. $ sudo chmod 777 /etc/hosts

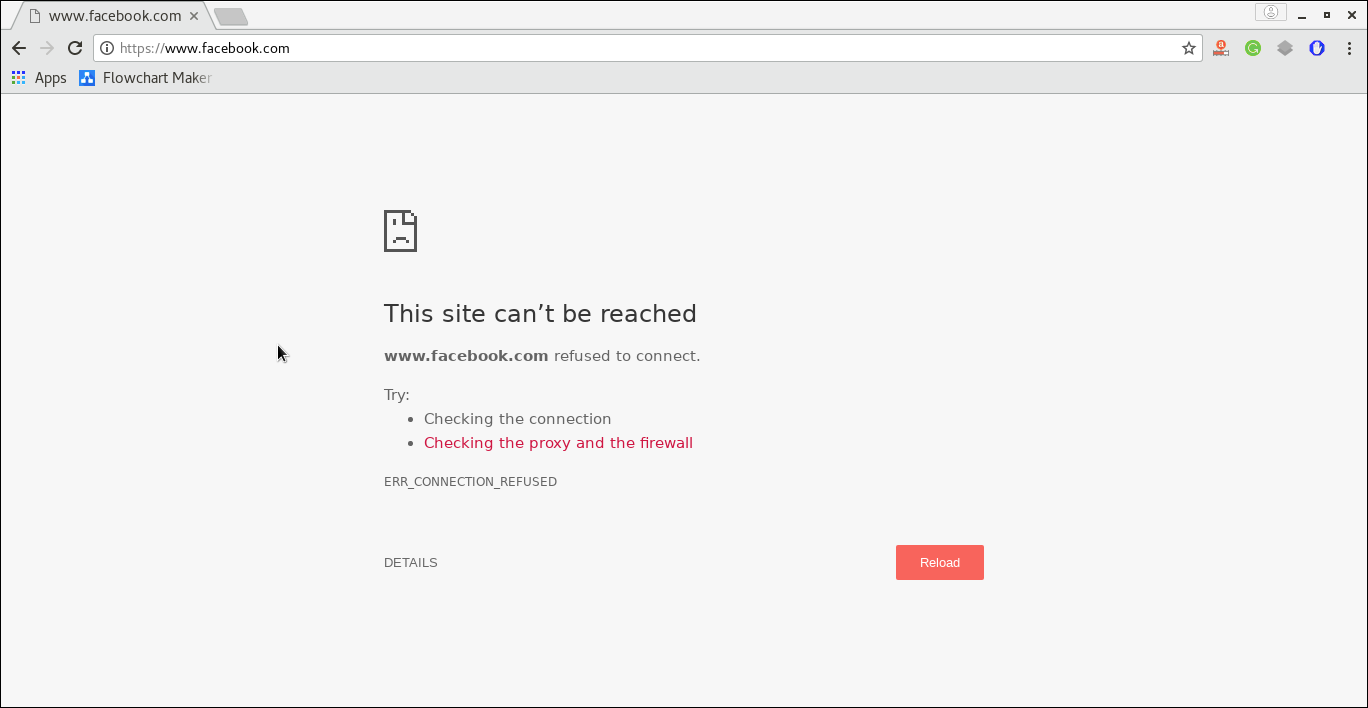
Now add the following line to the file to block the access to facebook.com (for example).

1. 127.0.0.1       www.facebook.com

As shown in the image below.



Save the file and try to open the www.facebook.com using the browser.



As shown in the above figure, it is refused to connect.

We have completed our task by manually editing the hosts file. We haven't achieved our objective yet. Our objective is to block access to some particular websites (for example, facebook) during working hours only (9 AM to 5 PM).

It needs a python script which keeps adding the necessary lines to the hosts file during a particular period.

In this section of the tutorial, we will build such python script which keeps editing the hosts file during the working hours. We will also deploy that script at the OS startup so that it doesn't need any external execution.

## *4.4) Requirements*

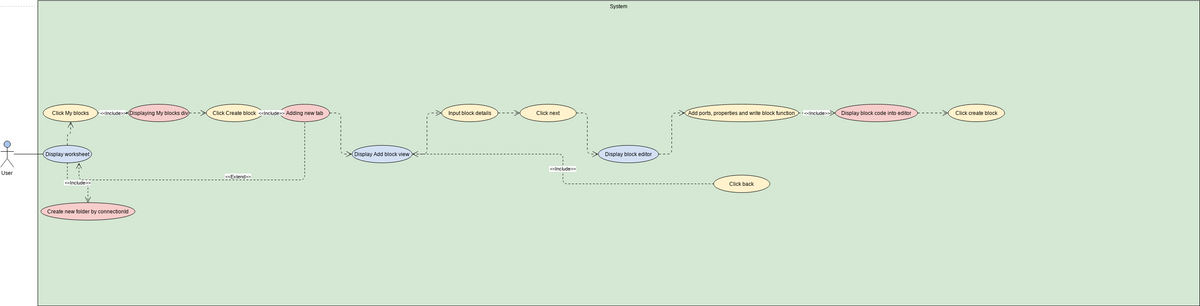
We need to know the following python modules to build the python website blocker.

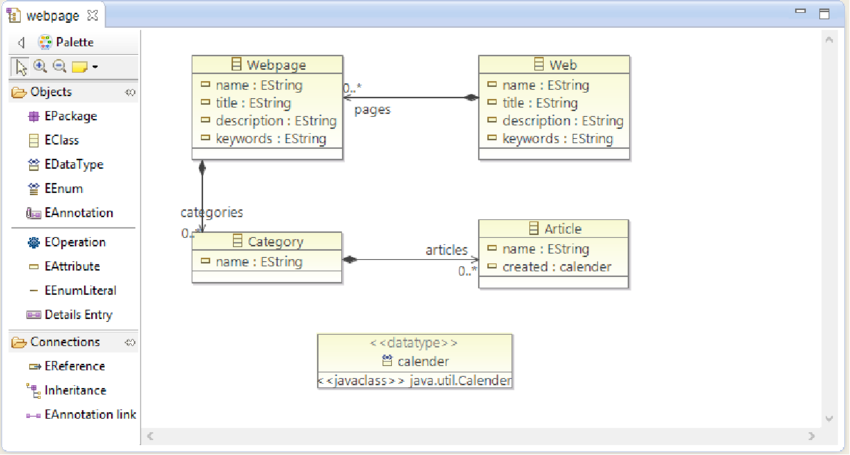
1. **file handling:** file handling is used to do the modifications to the hosts file.
2. **time:** The time module is used to control the frequency of the modifications to the hosts file.
3. **datetime:** The datetime module is used to keep track of the free time and working time

***CHAPTER 5***

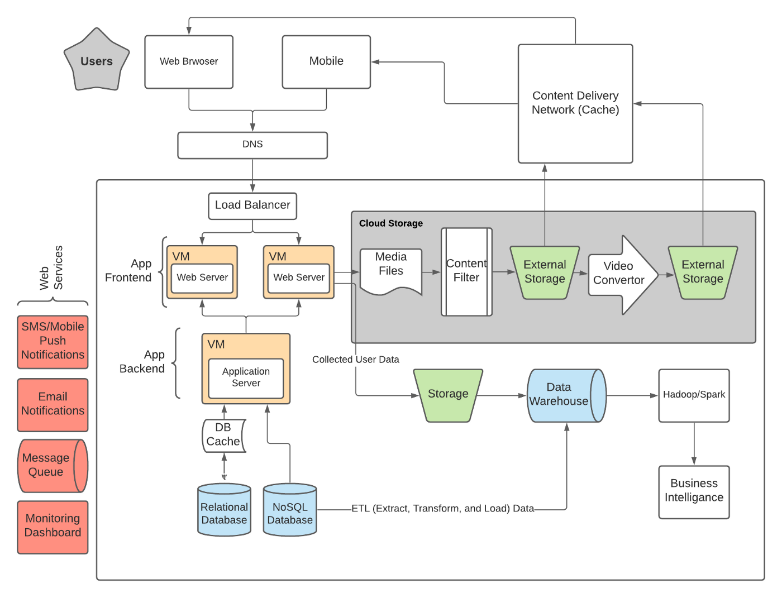
## *DIAGRAMS*

***5.1) USE CASE DIAGRAM***



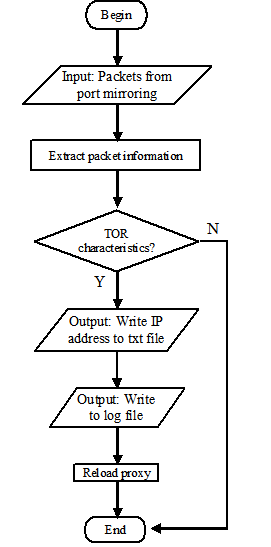
***5.2) Class Diagram***

***5.3) Architecture Diagram: -***



***CHAPTER 6***

***FLOWCHART OF PROJECT***



***CHAPTER 7***

***WORKING OF THE PROJECT***

1. Every system have **host** file whether it is Mac, Windows or Linux.  
   **Host** file in Mac and Linux :

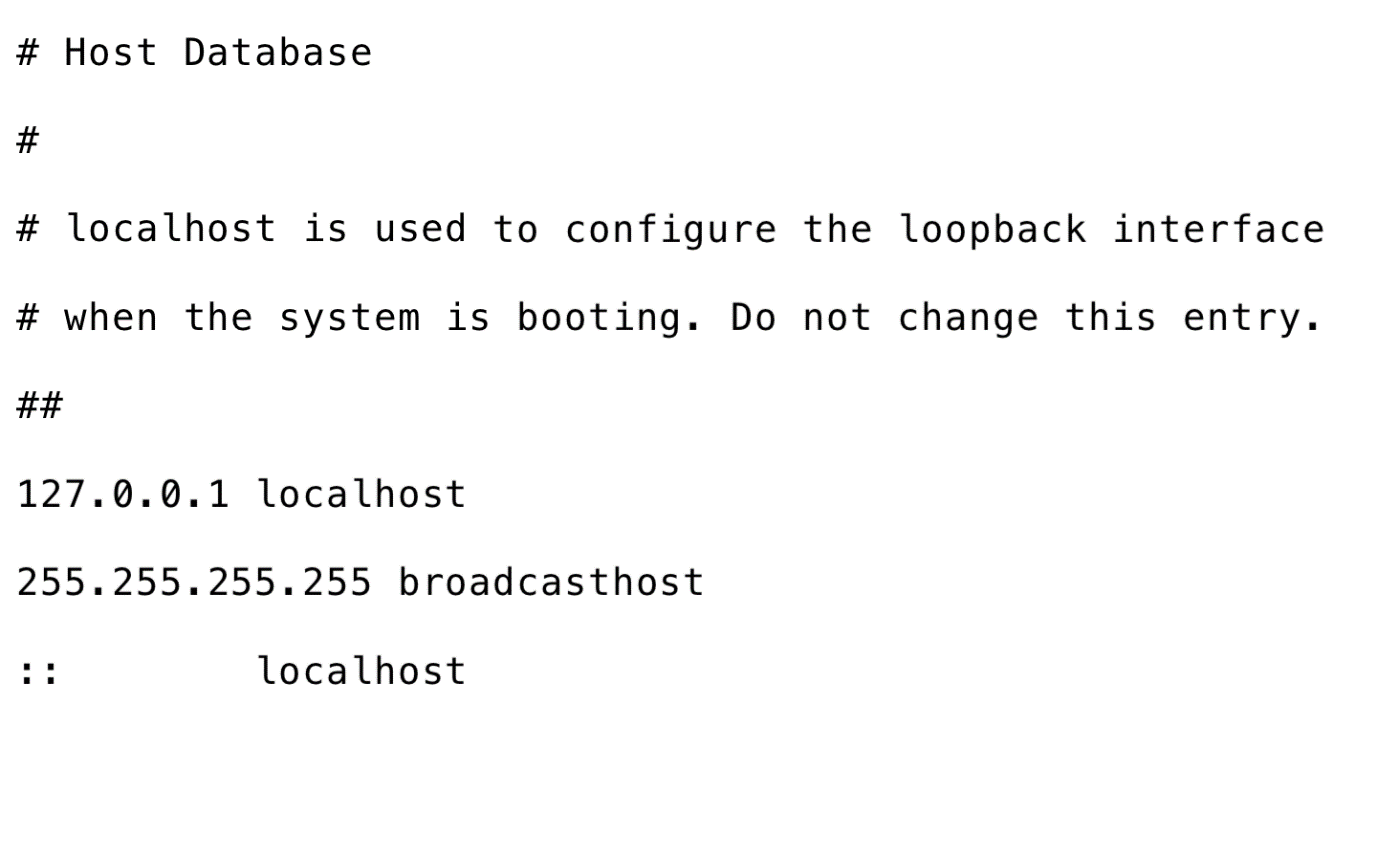
/etc/hosts

**Host** file in Windows:

C:\Windows\System32\drivers\etc

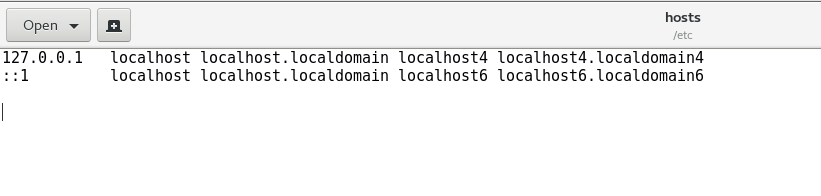
1. **Working of host file:**Host is an operating system file which maps hostnames to IP addresses. In this program we will be mapping hostnames of websites to our localhost address. Using python file handling manipulation we will write the hostname in hosts.txt and remove the lines after your working hours.

**Host file in Mac:**



***7.3) Configuration***

Since the hosts file contains the mapping between the host names and IP addresses, let's look at the content of our hosts file first stored as **/etc/hosts** as we use CentOS 7 (Linux).



As we can see in the above image, the mappings are saved for the localhost with the IP address 127.0.0.1.

Similarly, we can redirect any hostname back to the localhost IP (127.0.0.1). It will block the access to that hostname and redirect that to localhost on each request.

For testimony, let's edit the content of hosts file and add some of the following lines to it. To make changes to the **/etc/hosts** file, we will need to change its permissions.

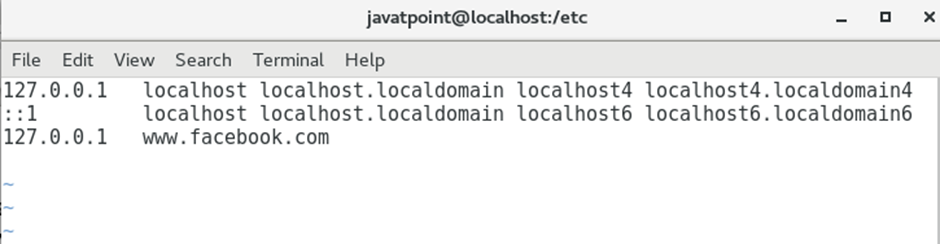
Run the following command on the terminal to change the permissions.

1. $ sudo chmod 777 /etc/hosts

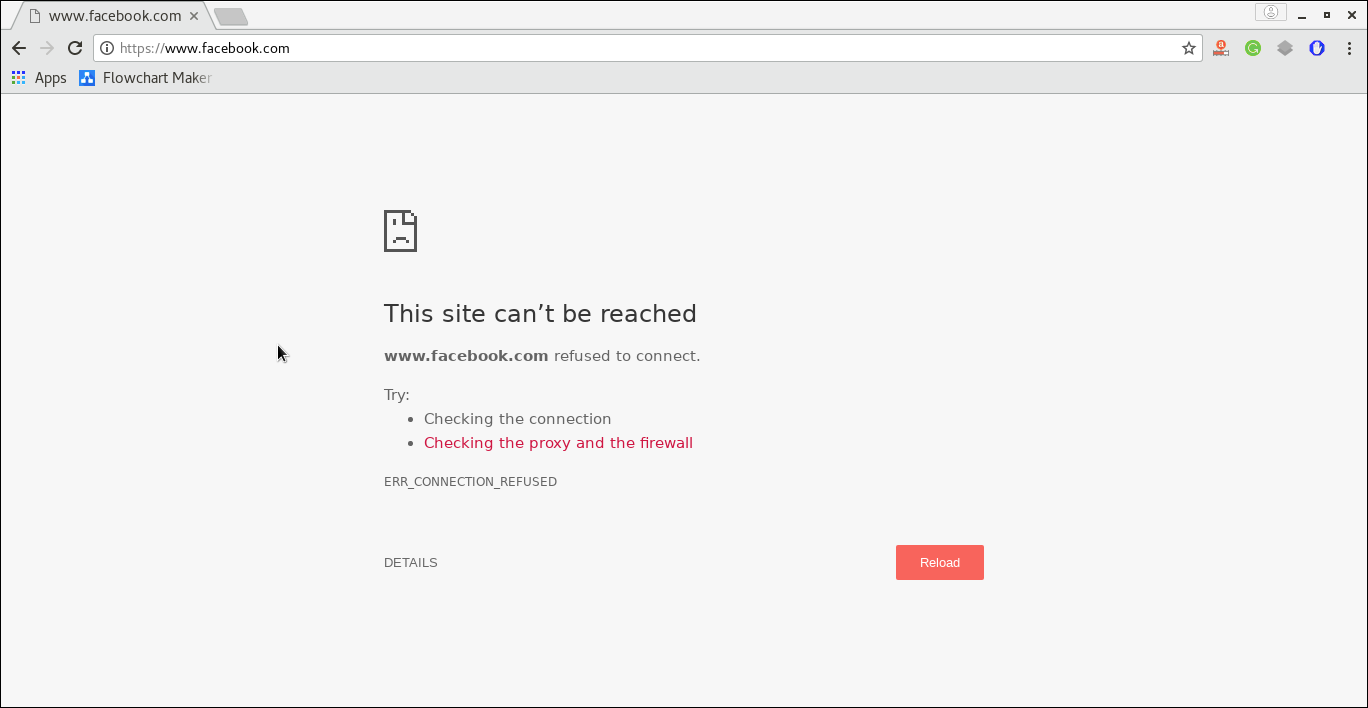
Now add the following line to the file to block the access to facebook.com (for example).

1. 127.0.0.1       www.facebook.com

As shown in the image below.



Save the file and try to open the www.facebook.com using the browser.



As shown in the above figure, it is refused to connect.

We have completed our task by manually editing the hosts file. We haven't achieved our objective yet. Our objective is to block access to some particular websites (for example, facebook) during working hours only (9 AM to 5 PM).

It needs a python script which keeps adding the necessary lines to the hosts file during a particular period.

In this section of the tutorial, we will build such python script which keeps editing the hosts file during the working hours. We will also deploy that script at the OS startup so that it doesn't need any external execution.

## *7.4) Requirements*

We need to know the following python modules to build the python website blocker.

1. **file handling:** file handling is used to do the modifications to the hosts file.
2. **time:** The time module is used to control the frequency of the modifications to the hosts file.
3. **datetime:** The datetime module is used to keep track of the free time and working time

***CHAPTER 8***

***Modules***

1)BLOCK DIVISION: -

To block a particular website

2)UNBLOCK DIVISION: -

To unblock a particular website

***CHAPTER 9***

***Code Explanations***

### ***Project File Structure***

Let us have a look at the steps we will be performing to create the Python Website Blocker project

1. Importing the required library
2. Creating GUI window
3. Hostpath and IP address.
4. Creating Block function to block websites
5. Creating Unblock function to unblock websites
6. Creating labels and buttons
7. Main command

Let us start building the python project following the given steps

#### **1. Importing the required library:**

We will only require Tkinter Module to create GUI in Python.

#importing required library

from tkinter import \*

* Import \* means importing all the methods and functions from the tkinter library.

#### **2. Creating GUI Window:**

window = Tk()

window.geometry('650x400')

window.minsize(650,400)

window.maxsize(650,400)

window.title(" DataFlair Website Blocker")

heading=Label(window, text ='Website Blocker' , font ='arial')

heading.pack()

* tk()- helps us create an empty window where we can add labels and buttons. We have created a space named window.
* Geometry() – this function is used to give size to the window.
* minsize(), maxsize() – this function is for giving the minimum and maximum size to the window.
* title() – provides an appropriate title to the window.
* Here we have created a Label named heading which is for giving a heading to the window. We have used Label() to create the label and pack() to pack the heading label to the window.

#### **3. Host Path and IP Address:**

host\_path ='C:\Windows\System32\drivers\etc\hosts'

ip\_address = '127.0.0.1'

* As we want the websites to be blocked in our system we need to add those websites in the host file. So here we give our local ip address and path of the host file.

#### **4. Create Block Function:**

def Blocker():

website\_lists = enter\_Website.get(1.0,END)

Website = list(website\_lists.split(","))

with open (host\_path , 'r+') as host\_file:

file\_content = host\_file.read()

for web in Website:

if web in file\_content:

display=Label(window, text = 'Already Blocked' , font = 'arial')

display.place(x=200,y=200)

pass

else:

host\_file.write(ip\_address + " " + web + '\n')

Label(window, text = "Blocked", font = 'arial').place(x=230,y =200)

* We have created this block function so that when we add any website to the text area and click on the block button, the website will be added to the host file and will be blocked. If the website is already in the host file then a label ‘Already Blocked ’ will be displayed.
* get() – this method is used to get the text that is added to the enter\_website label.
* open() – this is for opening the host file. Here we have opened the host file in r+ mode which is read plus write mode.
* split() – this method is used to separate the content of the text area. Here we used (,) as our delimiter to separate the website list.
* After opening the file, with help of a loop we check if the website entered is already present then display label “Already Blocked” and if it is not present add it to the file using write() method and then display label as “Blocked”.

#### **5. Unblock Function:**

def Unblock():

website\_lists = enter\_Website.get(1.0,END)

Website = list(website\_lists.split(","))

with open (host\_path , 'r+') as host\_file:

file\_content = host\_file.readlines()

for web in Website:

if web in website\_lists:

with open (host\_path , 'r+') as f:

for line in file\_content:

if line.strip(',') != website\_lists:

f.write(line)

Label(window, text = "UnBlocked", font = 'arial').place(x=350,y =200)

pass

else:

display=Label(window, text = 'Already UnBlocked' , font = 'arial')

display.place(x=350,y=200)

* The Unblock() function is created so that we can unblock a website that is already blocked and is present in the host file. If a website is blocked and we click on the Unblock button a label “Unblocked” is displayed. If a website is already Unblocked and isn’t a part of the host file then display label – “Already Unblocked”.
* get() – this method is used to get the text that is added to the enter\_website label.
* open() – this is for opening the host file. Here we have opened the host file in r+ mode which is read plus write mode.
* split() – this method is used to separate the content of the text area. Here we used (,) as our delimiter to separate the website list.
* We read the content of the file and if the entered website is present we rewrite all the content in the file except the one we need to unblock and display the label “Unblocked. And if there is no such website in the host file we simply display the label – “Already Unblocked”.

#### **6. Creating Labels and Buttons:**

label1=Label(window, text ='Enter Website :' , font ='arial 13 bold')

label1.place(x=5 ,y=60)

enter\_Website = Text(window,font = 'arial',height='2', width = '40')

enter\_Website.place(x= 140,y = 60)

* Here we have created one Label using the Tkinter library. One label is to display text -”Enter Website:” and the text area is to provide a text area to enter websites.
* The method Label() is used to create a label.
* For the text area, we use the method Text().
* To place both of these to the window we use the place() method and inside it we give the values to x and y.
* We can specify the text we want to display, font,height, width,bg,fg for each of these.

block\_button = Button(window, text = 'Block',font = 'arial',pady = 5,command = Blocker ,width = 6, bg = 'royal blue1', activebackground = 'grey')

block\_button.place(x = 230, y = 150)

unblock\_button = Button(window, text = 'UnBlock',font = 'arial',pady = 5,command = Unblock ,width = 6, bg = 'royal blue1', activebackground = 'grey')

unblock\_button.place(x = 350, y = 150)

* Here we have created 2 buttons one for block and another for unblock. We make use of the Button() method to create these buttons which is an inbuilt method of the Tkinter Module. We make use of the place() method to place these buttons on the window.
* Command= function\_name – This syntax is used inside the Button() method to specify what task a button will do when it is clicked. In this case, block and unblock are the two tasks.
* We can also specify a number of things like text on the button, font, bg,fg etc.

#### **7. Main Command:**

window.mainloop()

* The mainloop function is used to run the window and display the output on the window.
* While working with the host file, to run the code we need to open the code as admin in the command prompt and then run to make the changes in the hostfile. If we try to run the file directly it will show an error – “Permission not granted.”

***CHAPTER 10***

***MAIN PROGRAM***

#import library

from tkinter import \*

#initialize window

root = Tk()

root.geometry('500x300')

root.resizable(0,0)

root.title("TechVidvan - Website Blocker")

#heading

Label(root, text = 'WEBSITE BLOCKER' , font = 'arial 20 bold').pack()

Label(root,text = 'TechVidvan' , font = 'arial 20 bold').pack(side = BOTTOM)

#path of our host file ang ip adsdress

host\_path = 'C:\Windows\System32\drivers\etc\hosts'

ip\_address = '127.0.0.1'

#ENTER WEBSITE

Label(root, text= 'Enter Website :', font = 'arial 13 bold').place(x=5,y=60)

Websites = Text(root, font = 'arial 10', height ='2', width = '40', wrap = WORD,padx = 5, pady=5)

Websites.place(x = 140, y =60)

#block function

def Blocker():

website\_lists = Websites.get(1.0,END)

Website = list(website\_lists.split(","))

with open (host\_path ,'r+') as host\_file:

file\_content = host\_file.read()

for website in Website:

if website in file\_content:

Label(root, text = 'Already Blocked' , font = 'arial 12 bold').place(x=200,y=200)

pass

else:

host\_file.write(ip\_address + " " + website + '\n')

Label(root, text = "Blocked", font = 'arial 12 bold').place(x=230,y =200)

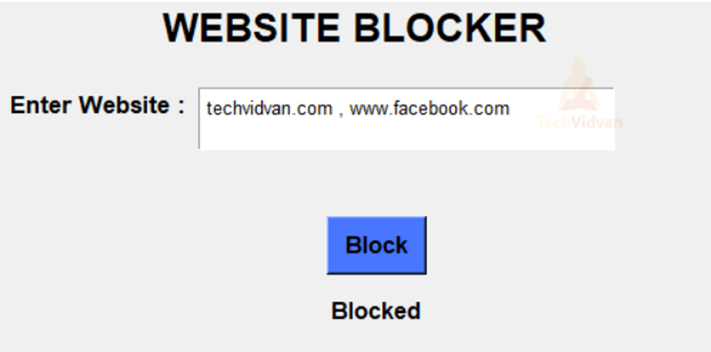
block\_btn = Button(root, text = 'BLOCK' , font = 'arial 12 bold', command = Blocker, width = 6 , bg = 'royal blue1', activebackground = 'sky blue')

block\_btn.place(x = 230, y =150)

root.mainloop

***OUTPUT: -***

***BLOCK DIVISION :-***



***UNBLOCK DIVISION: -***



***Result: -*** Successfully learned many things so far while doing this project and implemented and the outputs are executed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
|  | | | | |
|  | | | | |
|  | |  | | * 1. CM.Nirmal   2. Subhashis Tripathy   3. Ayyapa Reddy |
|  | |  | |  |
|  | |  | | RA2112703010011  RA2112703010020  RA2112703010030 |
|  | |  | |  |
|  | |  | | Faculty of Engineering & Technology |
|  | |  | | WEBSITE BLOCKER |
|  | |  | |  |
|  |  | | Pycharm | |
|  |  | | 11/11/2022 | |