Image based Captcha using Python

A project report submitted in partial fulfilment for the Award

Of

BACHELOR OF TECHNOLOGY

IN

Computer Science & Engineering

By

Siddarth N S(12104917)

Naveed Mohammed(12105088)

SECTION-K21PK

LOVELY PROFESSIONAL UNIVERSITY,

PUNJAB INDIA

Acknowledgement

The satisfaction that accompanies the successful completion of this project would be in complete without the mention of the people who made it possible, without whose constant guidance and encouragement would have made efforts go in vain. I consider myself privileged to express gratitude and respect towards all those who guided us through the completion of this project.

I convey thanks to my project guide Ms. Archana Chhabra of Computer Science and Engineering Department for providing encouragement, constant support and guidance which was of a great help to complete this project successfully.

Last but not the least, we wish to thank our parents for financing our studies in this college as well as for constantly encouraging us to learn engineering. Their personal sacrifice in providing this opportunity to learn engineering is gratefully acknowledged.

Abstract

CAPTCHA : A CAPTCHA is a type of challenge–response test used in computing to determine whether or not the user is human.

CAPTCHAs are, by definition, fully automated, requiring little human maintenance or intervention to administer, producing benefits in cost and reliability. The most common, typical CAPTCHA Code is a text based image with distorted letters of different fonts, blurry or confusing backgrounds with random lines. The user is asked to reenter the letters or numbers to obtain services. If the user fails, then the access is denied.

CAPTCHA technique is basically a challenge response test which involves a computer (server) initiating a task for the user to complete. If the user completes it successfully then the user is considered as “human” else it is treated as a “web–bot”.

Abbreviations

- GUI : Graphical User Interface
- CAPTCHA : Completely Automated Public Turning test to tell Computers and Humans Apart
- AD : Advertisement
- Email : Electronic mail

Contents

1.Introduction

This section gives a scope description and overview of Everything included in this Project Report. Also, the purpose for this document is described and system overview along with goal and vision are listed.

1.1Purpose

Captcha using Python Project. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with users. This document is primarily intended to anyone who wants to get an overview of how Captcha works, its outcomes and possible usages in future.

1.2System Overview

A CAPTCHA system presents a visitor with an obscured word, words, or phrase. The obscuring is usually achieved by warping the words, distorting the background, or segmenting the word by adding lines.Users are asked to decode the image and select the image in the correct order (they may or may not be case sensitive) before submitting the form. Upon form submission, the response is verified, and users are either taken to the next step or presented with an error.

1.3 Problem Statement

Nowdays on every platform Bot is main reason for very high traffic on internet. "Bot" generally refers to any program that is set to automatically complete some process, whether it's posting news on Twitter or leaving spam in website comment sections. Used correctly, these programs are fairly useful, but they can also be used to generate useless/ad-ridden/malicious content, overwhelm a site with signups, rig online poll results, scrape email addresses, or do any number of other unpleasant things.

It's just best not to let them in. To overcome this problem we need Captcha system. CAPTCHA motto goes, to create a task that is "Easy for people, hard for bots."

1.4 Goal & Vision

The main goal of CAPTCHA is to put forth a test which is simple and straight forward for any human to answer but for a computer, it is almost impossible to solve. CAPTCHAs can be used by websites that offer services like online polls and registration forms.
Web-based email services like Gmail, Yahoo and Hotmail offer free email accounts for their users.
However, upon each sign-up process, CAPTCHAs are used to prevent spammers from using a bot to generate hundreds of spam mail accounts.

2. Requirements Specification

2.1Hardware Requirements

To access this application, you only need a PC/Laptop with Python install on it. Also check if python GUI working on it or not.

2.2Software Requirements

For this system to work you need python compiler to write the code and execute it. You also need to install the necessary package for python GUI to work. After executing you can use this software in new window which will appear after executing the code.

3.**Procedure**

First open the python compiler.

After opening python compiler, type the code and attach all images required by the program.

After typing all the code, execute it.

After executing new window will open where registration number, password and verify captcha option will be present. Type the required detail and after that type the captcha shown and verify it.

If you are verified then you are logged in otherwise you need to fill all the details again.

4. Constraints and Assumptions

Image Captcha only available for image not for any other meanlike audio or any mathematical calculations. It only supports English language and will not work with any other language. It is note case sensitive so writing in any case will work. Since this project is of small level so only limited number of captchas are available.

5. Code

```
import random import string from tkinter import

Tk, Label, Entry, Button, END from PIL import

ImageTk, Image from captcha.image import

ImageCaptcha
```

```
from tkinter import *

from PIL import Image, ImageTk

from random import *


opt = {1: "bird", 6: "cake", 11: "road"}

i = randrange(1, 16, 5)

img_ran = {1: "b1.jpg",      2: "b2.jpg", 3: "b3.jpg", 4:

"b4.jpg",5: "b5.jpg", 6: "c1.jpg", 7: "c2.jpg", 8: "c3.jpg",

  9: "c4.jpg", 10: "c5.jpg", 11: "r1.jpg", 12: "r2.jpg", 13:

"r3.jpg", 14: "r4.jpg", 15: "r5.jpg"}

i1 = randrange(1, 16, 1)

i2 = randrange(1, 16, 1)

i3 = randrange(1, 16, 1)

i4 = randrange(1, 16, 1)

i5 = randrange(1, 16, 1)

i6 = randrange(1, 16, 1)
```

```
i7 = randrange(1, 16, 1)
i8 = randrange(1, 16, 1)
i9 = randrange(1, 16, 1)
iS = [i1, i2, i3, i4, i5, i6, i7, i8, i9]
iT = []

for x in range(9):
    if (iS[x] == i or iS[x] == i + 1 or iS[x] == i + 2 or iS[x]
== i + 3 or iS[x] == i + 4):
        iT.append(iS[x])

root = Tk()
root.title("Image Captcha Verification")
_l1 = Label(root, text="Select all images which contain a
" + opt[i])
_l1.grid(row=0, column=3, columnspan=10)
img = ImageTk.PhotoImage(Image.open(img_ran[i1]))
imglabel1 = Label(root, image=img, bg="black",
width=150, height=150)
imglabel1.grid(row=2, column=0, rowspan=5,
columnspan=5)
var1 = IntVar()
c1 = Checkbutton(root, variable=var1,
highlightthickness=0, bd=0, offvalue=0, onvalue=i1)
c1.grid(row=6, column=4)
img1 = ImageTk.PhotoImage(Image.open(img_ran[i2]))
imglabel2 = Label(root, image=img1, bg="black",
width=150, height=150)
imglabel2.grid(row=2, column=5, rowspan=5,
columnspan=5)
var2 = IntVar()
```

```
c2 = Checkbutton(root, variable=var2,
highlightthickness=0, bd=0, offvalue=0, onvalue=i2)
c2.grid(row=6, column=9)

img2 = ImageTk.PhotoImage(Image.open(img_ran[i3]))
imglabel3 = Label(root, image=img2, bg="black",
width=150, height=150)
imglabel3.grid(row=2, column=10, rowspan=5,
columnspan=5)
var3 = IntVar()
c3 = Checkbutton(root, variable=var3,
highlightthickness=0, bd=0, offvalue=0, onvalue=i3)
c3.grid(row=6, column=14)
img3 = ImageTk.PhotoImage(Image.open(img_ran[i4]))
imglabel4 = Label(root, image=img3, bg="black",
width=150, height=150)
imglabel4.grid(row=7, column=0, rowspan=5,
columnspan=5)
var4 = IntVar()
c4 = Checkbutton(root, variable=var4,
highlightthickness=0, bd=0, offvalue=0, onvalue=i4)
c4.grid(row=11, column=4)

img4 = ImageTk.PhotoImage(Image.open(img_ran[i5]))
imglabel5 = Label(root, image=img4, bg="black",
width=150, height=150)
imglabel5.grid(row=7, column=5, rowspan=5,
columnspan=5)
var5 = IntVar()
c5 = Checkbutton(root, variable=var5,
highlightthickness=0, bd=0, offvalue=0, onvalue=i5)
c5.grid(row=11, column=9)
```

```
img5 = ImageTk.PhotoImage(Image.open(img_ran[i6]))
imglabel6 = Label(root, image=img5, bg="black",
width=150, height=150)
imglabel6.grid(row=7, column=10, rowspan=5,
columnspan=5)
var6 = IntVar()
c6 = Checkbutton(root, variable=var6,
highlightthickness=0, bd=0, offvalue=0, onvalue=i6)
c6.grid(row=11, column=14)
img6 = ImageTk.PhotoImage(Image.open(img_ran[i7]))
imglabel7 = Label(root, image=img6, bg="black",
width=150, height=150)
imglabel7.grid(row=12, column=0, rowspan=5,
columnspan=5)
var7 = IntVar()
c7 = Checkbutton(root, variable=var7,
highlightthickness=0, bd=0, offvalue=0, onvalue=i7)
c7.grid(row=16, column=4)

img7 = ImageTk.PhotoImage(Image.open(img_ran[i8]))
imglabel8 = Label(root, image=img7, bg="black",
width=150, height=150)
imglabel8.grid(row=12, column=5, rowspan=5,
columnspan=5)
var8 = IntVar()
c8 = Checkbutton(root, variable=var8,
highlightthickness=0, bd=0, offvalue=0, onvalue=i8)
c8.grid(row=16, column=9)


img8 = ImageTk.PhotoImage(Image.open(img_ran[i9]))
```

```
imglabel9 = Label(root, image=img8, bg="black",
width=150, height=150)
imglabel9.grid(row=12, column=10, rowspan=5,
columnspan=5)
var9 = IntVar()
c9 = Checkbutton(root, variable=var9,
highlightthickness=0, bd=0, offvalue=0, onvalue=i9)
c9.grid(row=16, column=14)
def verCap():
    l_var = [var1.get(), var2.get(), var3.get(), var4.get(),
var5.get(), var6.get(), var7.get(), var8.get(), var9.get()]
    l_var1 = []
    l_var2 = []
    for i in range(9):
        if (l_var[i] != 0):
            l_var2.append(l_var[i])
            if (l_var[i] in iT):
                l_var1.append(l_var[i])
    if (len(l_var1) == len(iT) == len(l_var2)):
        imglabel1.destroy()
        c1.destroy()
        imglabel2.destroy()
        c2.destroy()
        imglabel3.destroy()
        c3.destroy()
        imglabel4.destroy()
        c4.destroy()
        imglabel5.destroy()
        c5.destroy()
        imglabel6.destroy()
        c6.destroy()
        imglabel7.destroy()
        c7.destroy()
        imglabel8.destroy()
        c8.destroy()
        imglabel9.destroy()
        c9.destroy()
        la = Label(root, text="Captcha Verified!", width=25,
height=10)
        la.grid(row=1, column=1)
        but.destroy()
        _l1.destroy()
    else:
        imglabel1.destroy()
        c1.destroy()
        imglabel2.destroy()
        c2.destroy()
        imglabel3.destroy()
        c3.destroy()
        imglabel4.destroy()
        c4.destroy()
        imglabel5.destroy()
        c5.destroy()
        imglabel6.destroy()
        c6.destroy()
        imglabel7.destroy()
        c7.destroy()
        imglabel8.destroy()
        c8.destroy()
        imglabel9.destroy()
        c9.destroy()
        lb = Label(root, text="Uh Oh! Captcha Mismatch!
Try Again!", width=45, height=10)
        lb.grid(row=1, column=1)
    but.destroy()
    _l1.destroy()
but = Button(root, width=15, height=3, bd=5, fg="black",
text="Verify", command=verCap)
but.grid(row=17, column=8)
```

6. GUI Design for Frontend

- User interface

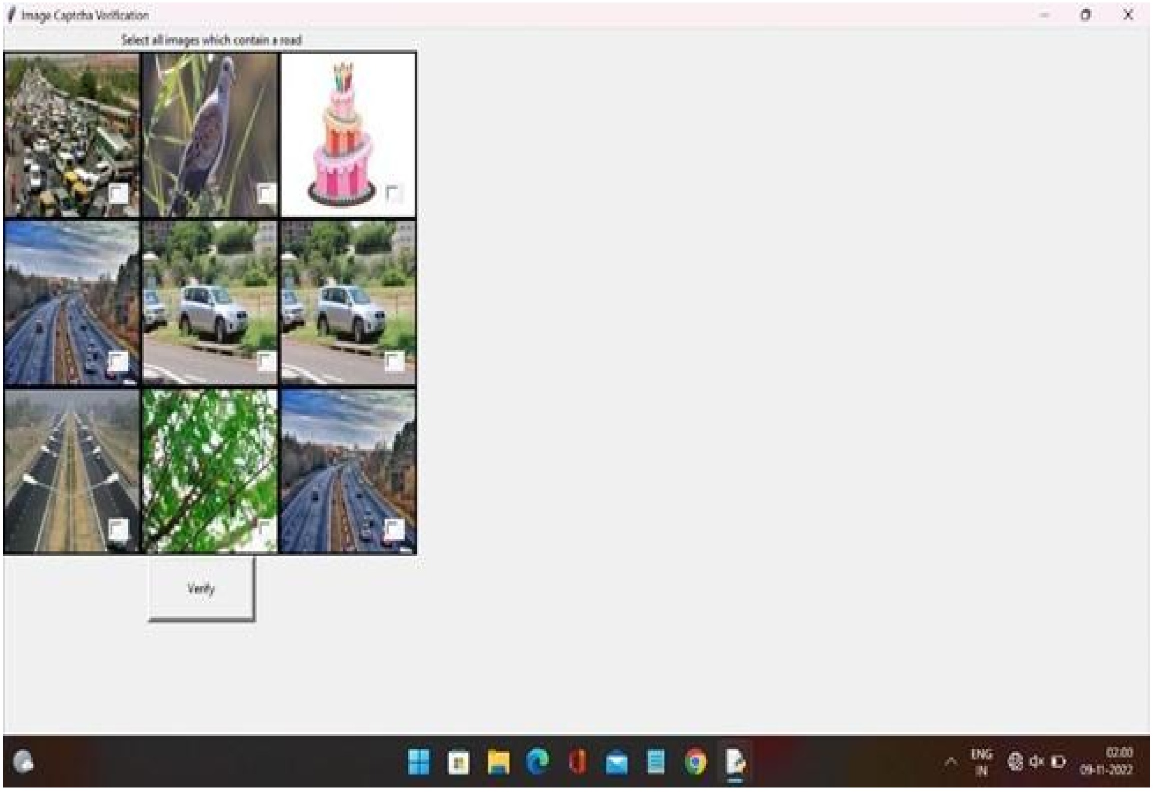


Image Captcha Verification

Select all images which contain a road

- **OUTPUT:**

- Captcha Not Verified:

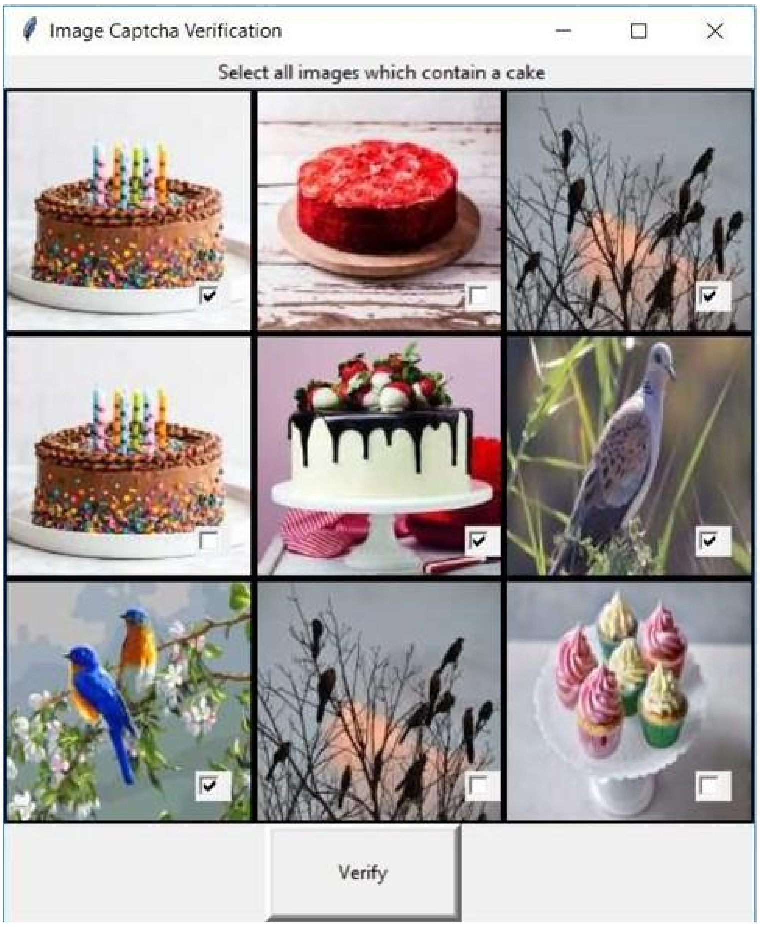
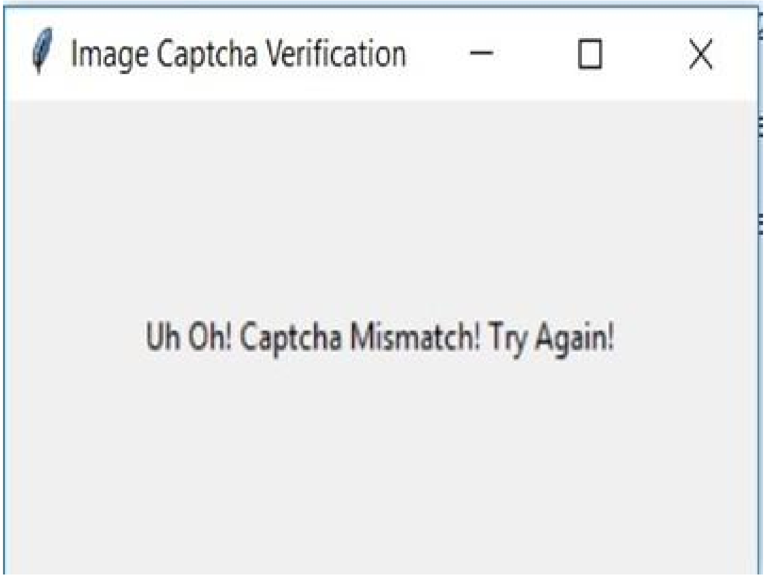
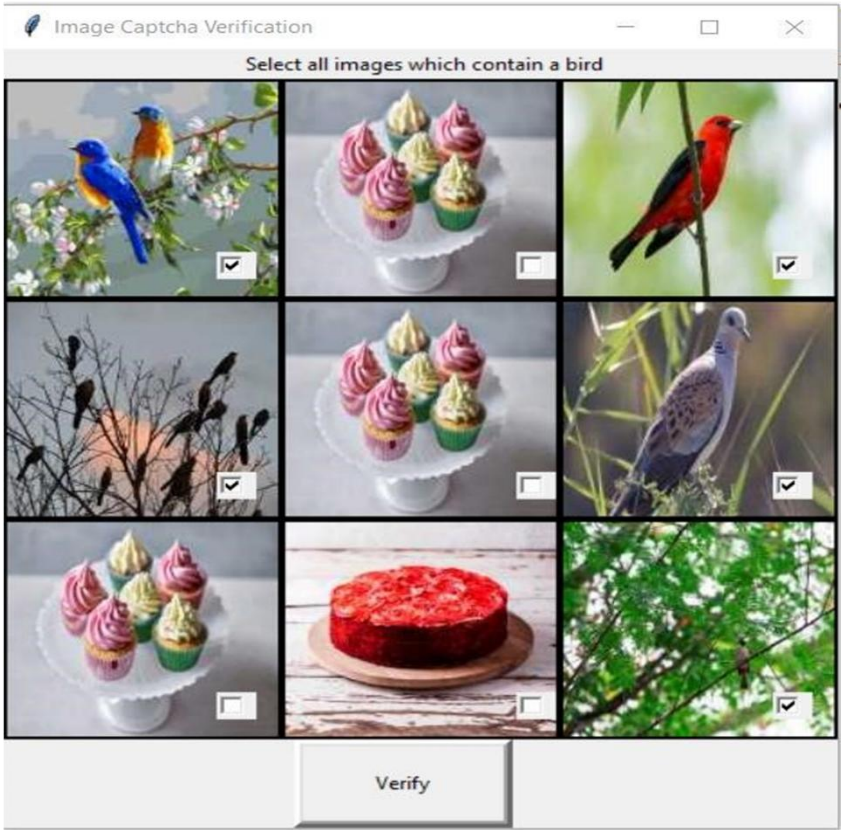


Image Captcha Verification
Select all images which contain a cake
Verify

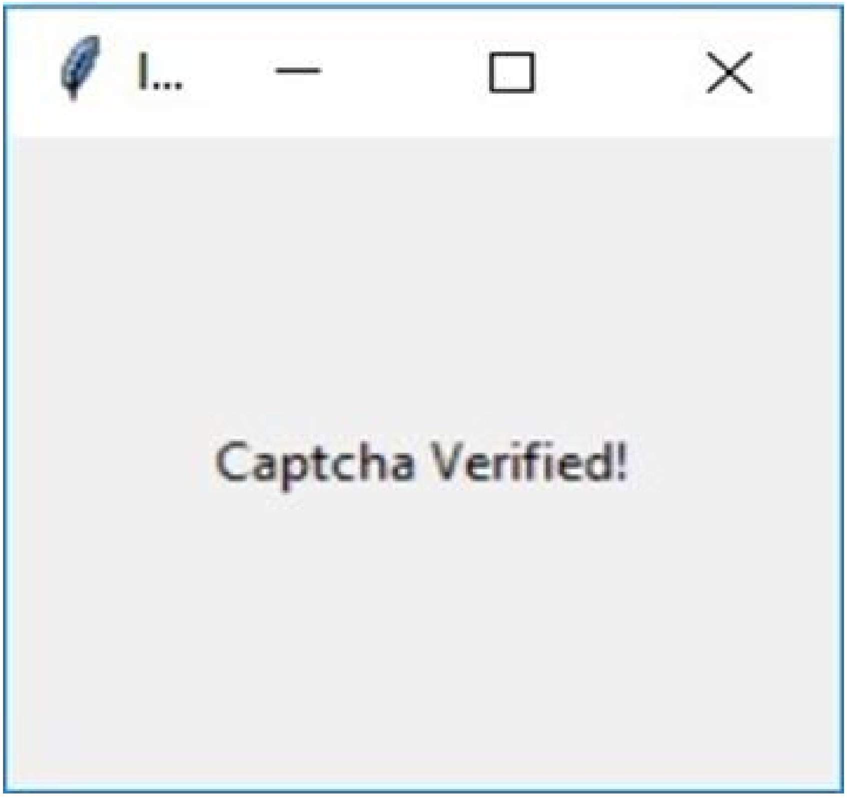


Uh Oh! Captcha Mismatch! Try Again!

- Captcha Verified:



Select all images which contain a bird



Captcha Verified!

7.Conclusion

The captcha helps us to avoid both spam and bot present on different platform. It helps different platform to avoid fake accounts or post . The experience of developing this project also helped us learning lot about python and python GUI. It also simplify the problem of redundant accounts on any platform.

It helps us to learn how to code in python and we are able to learn more about different module present in python. It also proved beneficial for us because we were able to design GUI in python .

8. References

- https://en.wikipedia.org/wiki/CAPTCHA
- https://electricalfundablog.com/captc ha-code- works-design-types