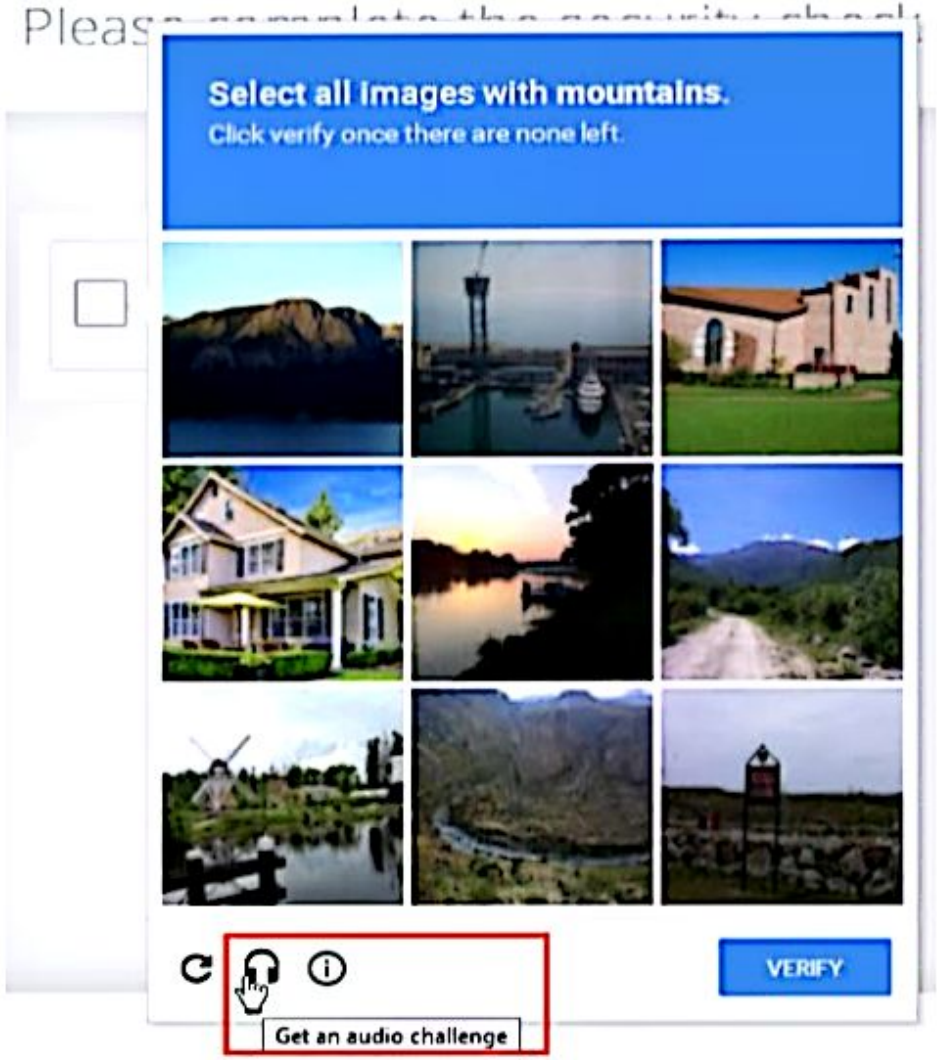
Project no:-18

INT-213 REPORT

Title:-Image base captcha using python

1. N.Bhanu prakash 11907168
2. N.venkateswarlu 11915167
3. A.Sai Charan. 11915357



* Intro

a) Captcha is used to human Verification

b) Image captcha is more secure than text based Captcha.

* Requirements

a) Requirements of this project is to display some images.

b) Based on the images user has to select that image which specify the question.

c) After clicking the submit button, Answers should be verify with the user response.

d) If captcha is more complex then user can skip that for the next captcha.

Design and GUI

a) We python to write the code and tkinter for implementation our Graphical user interface.

b) We used PIL for putting the images on the GUI.

Tkinter - is a python library uses for GUI.

PIL - Python Image Library.

- From tkinter import
- From PIL import Image, ImageTk.

Coding

* Python – To write the code.
* SQLITE3 – To handle the database for the project
  import .sqlite3.
* We make use of classes in our coding part so that we have to write less code.
* Also we uses the inheritance of the classes for reusing the code again and easy access to the require data.

Classes

* Image class
  → For a single image and its attributes.
* Group class
  → It will consist of its attributes and 9 images.
* Captcha class
  → It inherited by Group class and the use of the captcha is displaying, user response and validation of response.

Captcha class

captcha.py

captcha (Group.Group)

- __init__ (self, root, group_id)
- image_clicked (self, index, i, j)
- next_captcha (self)
- reset (self)
- show_captcha (self)
- show_welcome_page (self)
- verify (self)

Group class

Group.py

Group

- __init__ (self, id)
- get_answer_list (self)
- get_answer_list (self)
- get_image_list (self)
- set_answer_for (self, index, value)
- set_response_for (self, index)
- set_statement (self)

Files

- main.py
- captcha.PY
- Group.PY
- image.PY
- Database.fdb

Features

* Reset
→ User can reset what ever he/she selected

* Next captcha
→ if the captcha is more complex then user has a choice to skip that captcha.

* Verify
→ this is a button to verify the image selection.

Home Page - Select or create a | Untitled8 - Jupyter Notebook

localhost:8888/notebooks/Untitled8.ipynb?kernel_name=python3

jupyter Untitled8 Last Checkpoint: 21 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Code

In [2]:
```
from tkinter import *
import tkinter.messagebox
from captcha.image import imagecaptcha
import string
import random
import os
import pyttsx3


image_captcha = ''
def genarate_captcha():
    data_set = list(string.ascii_letters+strings.digits)
    s=''
    for i in range(6):
    a=random.choice(data_set)
    s=s+a
    data_set.remove(a)


    global image_captcha
    image_chaptcha=s
    return s


auido_captcha = ''
def genarate_digit_captcha():
    s1='';s=''
```

Type here to search

ENG 18:51 01-11-2020

localhost:8888/notebooks/Untitled8.ipynb?kernel_name=python3

jupyter Untitled8 Last Checkpoint: 21 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Code

```
auido_captcha = ''
def genarate_digit_captcha():
    s1='';s=''
    for i in range(4):
    a=str(random.randint(0,9))
    s=s+a
    s1=s1+a+" "


    global audio_captcha
    audio_chaptcha=s
    return s1
def genarate_first_image():
    img =imagecaptcha()

    s = generate_captcha()

    value = img.generate(s)
    img.write(s,"c1.png")
    os.startfile('c1.png')


def generate_audio_captcha():
    s=generate_digit_captcha()

    voiceEngine = pyttsx3.init()
    voiceEngine.setproperty('rate',170)
    voiceEngine.setproperty('volume',1.0)
```

Type here to search

ENG 18:51

Home Page - Select or create a | Untitled8 - Jupyter Notebook

localhost:8888/notebooks/Untitled8.ipynb?kernel_name=python3

jupyter Untitled8 Last Checkpoint: 21 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run Code

```
    global audio_captcha
    audio_chaptcha=s
    return s1
def genarate_first_image():
    img =imagecaptcha()

    s = generate_captcha()

    value = img.generate(s)
    img.write(s,"c1.png")
    os.startfile('c1.png')


def generate_audio_captcha():
    s=generate_digit_captcha()

    voiceEngine = pyttsx3.init()
    voiceEngine.setproperty('rate',170)
    voiceEngine.setproperty('volume',1.0)

    voices =voiceEngine.getproperty('voices')
    voiceEngine.setproperty('voice',voices[1].id)

    voiceEngine.say(s)
    voiceEngine.runAndWait()
    voiceEngine.say(s)
    voiceEngine.runAndWait()
```

Type here to search