

Personal Voice Assistant as Helping-Hand for Searching Online Events and Registrations for them

SYNOPSIS OF THE PROJECT

Objective/Aim of the project:

To help online event seekers to search for online events from some of the popular sites, which serve as a platform for such types of online events, so that the user will not have to search for such types of sites here and there, wasting his/her time. Also, to remind the user of the event deadlines so that he/she will not miss any opportunity for his/her interest. The user can search the events with the help of an assistant (her name is Anna) and save the events of his/her choice later, if he/she is not sure of registering at that moment, so that the assistant can send a reminder via email.

Software Requirements:

- Operating System: Windows 10
- Programming languages: Html, Python
- Database : MS Excel
- Software: IDLE

Libraries, Modules and Packages used:

- pandas
- hashlib
- flask_ngrok -- run_with_ngrok
- flask -- Flask, request
- speech_recognition
- gtts -- gTTS
- playsound
- time -- strftime
- os
- re
- webbrowser
- bs4 -- BeautifulSoup
- requests
- smtplib
- openpyxl

- datetime -- date

Concepts Used:

- Front-end Development
- Web-Scraping
- Data Analysis
- Virtual Voice assistant
- Building API using python

Navigation Structure:

- Index Page -
 - 1) Consist of
 - *Activate Anna* button which is used to open Login page and thus activating anna
 - Link to SignUp page
 - 2) Technicalities behind page
 - HTML language is used for front-end development
- SignUp Page -
 - 1) Consist of
 - Registration form
 - 2) Technicalities behind page
 - HTML is used for front-end development, which makes sure that all fields are filled before submitting, for implementing html on webpage, flask package is used
 - POST method is used for posting data to another page
 - Data contains name, email and password of a user
- Login Page -
 - 1) Consist of
 - Login Form
 - 2) Technicalities behind page
 - HTML is used for front-end development
 - POST method is used for posting data to another page
 - Data contains email and password of a user used while registration
- Registration Successful Message page
 - 1) Consist of
 - *Start Conversation* button which triggers our voice assistant to finally start the process of searching for events

2) Technicalities behind page

- request module from flask is used to fetch data from the form
- "Registration successful!" message is shown when all the data posted gets saved in "Registered_Users.xlsx" file using openpyxl python library and also users dictionary , first variables are assigned to data and then this data is appended to the file
- Before saving hashlib is used to encrypt password to ensure security of the user account
- To create session email of user is assigned as session_email

● Login Successful Message page

3) Consist of

- *Start Conversation* button

4) Technicalities behind page

- request module from flask is used to fetch data
- "Login Successful" message is shown when password is correct after checking from "Registered_Users.xlsx" file, again hashlib and openpyxl is used for this purpose
- First the data is extracted in users_data dictionary so that we can get name of logged in user to set as session email
- To create session email of user's email is assigned as session_email

● Conversation Interrupted Message Page

1) Consist of

- *Save Event* Button - leads to save event page
- *Activate Again* Button - leads to Login Page

● Save Events Page

1) Consist of

- Save Event form

2) Technicalities behind page

- Data is saved in "User_Events.xlsx" file
- Form leads to "You are Logged Out" message page which contains options for saving events and login again through buttons.

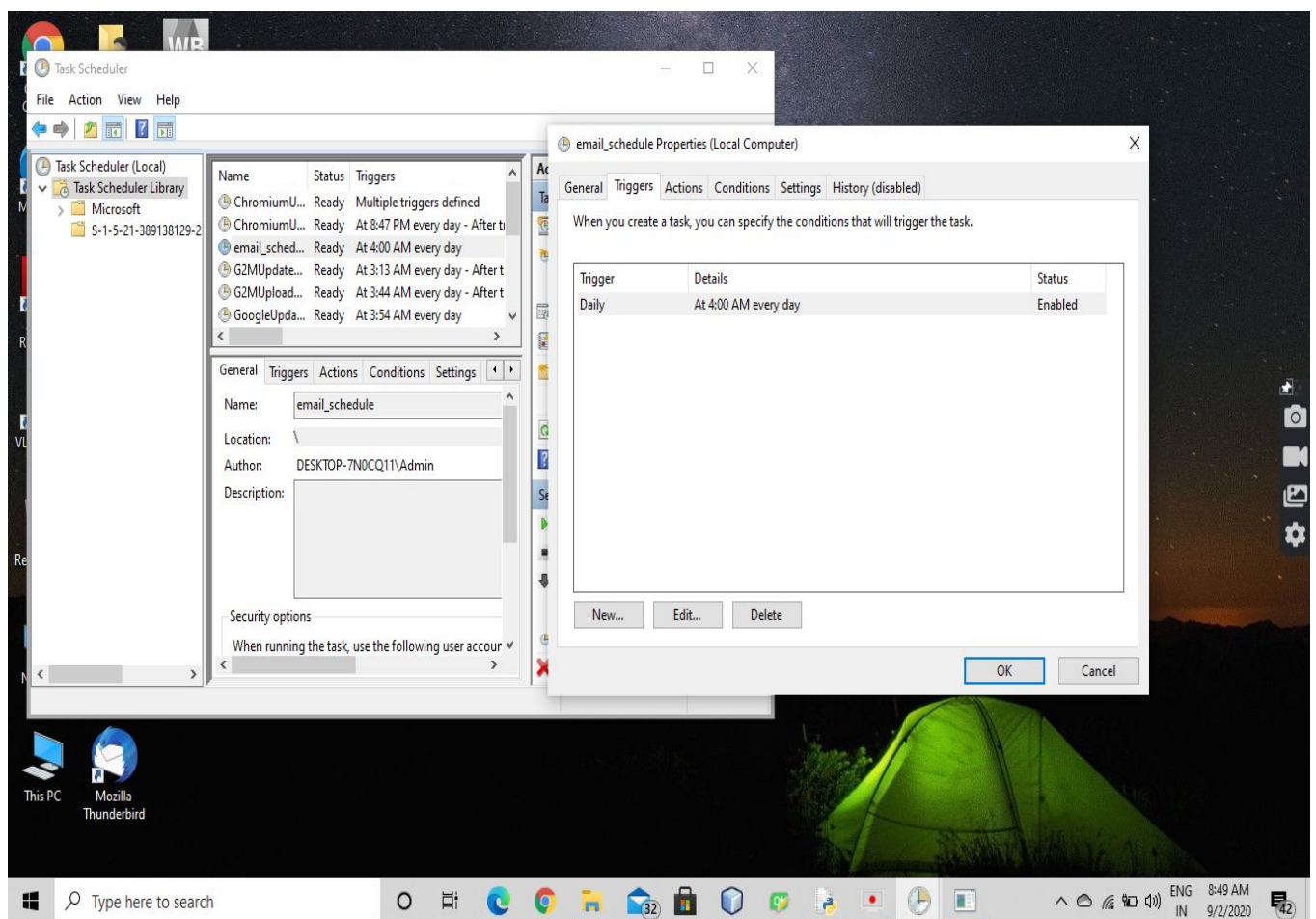
● You are logged out message page

1) Consist of

- *Save Events* button
- *Login Again* button

- Web scraping is used for getting competitions from myGov.com and the it is converted to dataframe using pandas library
- Projects are obtained from “Projects.xlsx” file using pandas
- time module is used for time sleep and getting date in specific format
- Task Scheduler is used for running pp2.py file daily at 4:00 AM
- Data analysis is used for getting events for a date for an email
- smtplib is used for connecting to mail server and sending reminder mails to user and then those events get deleted from the “User_Events.xlsx” file

Task Scheduler showing scheduled time as 4:00 AM

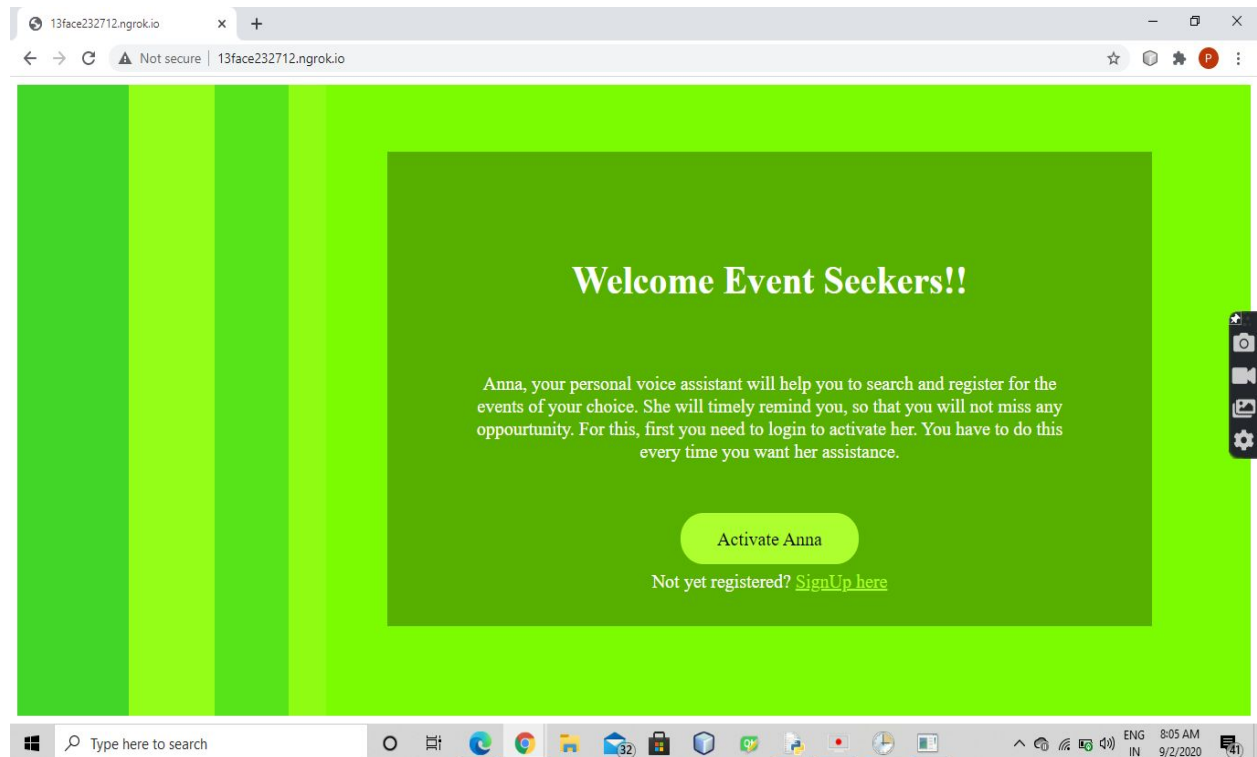


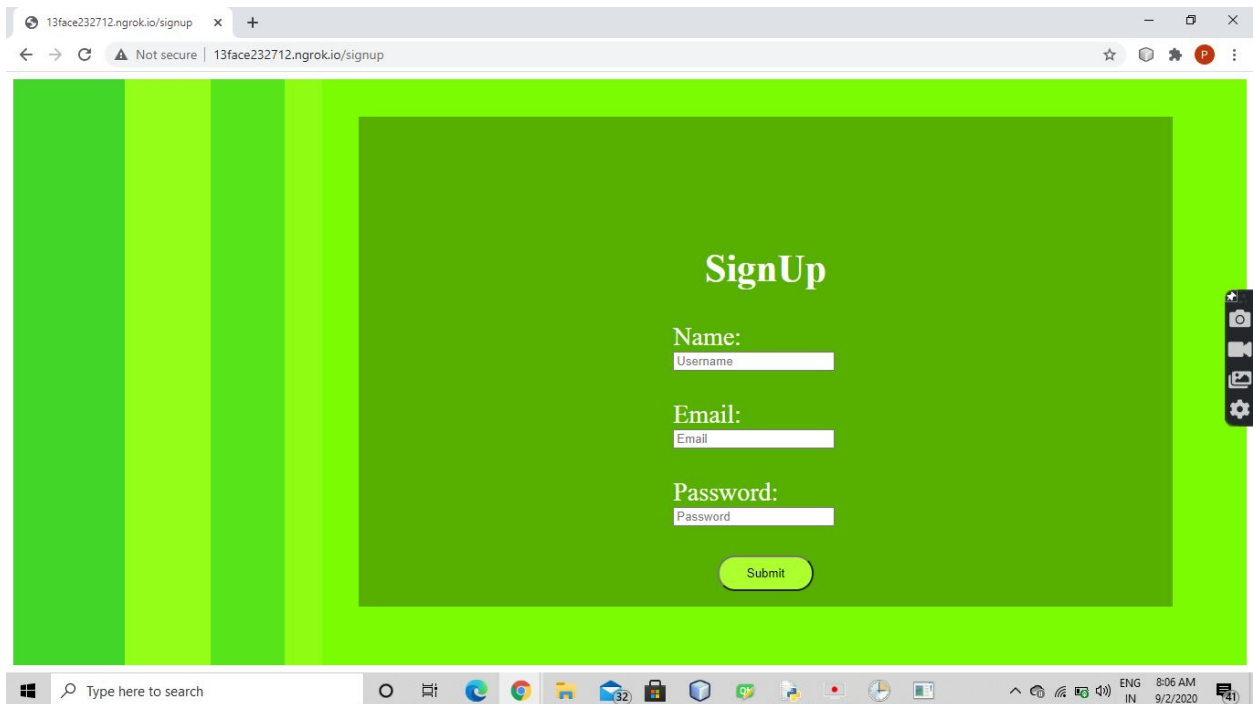
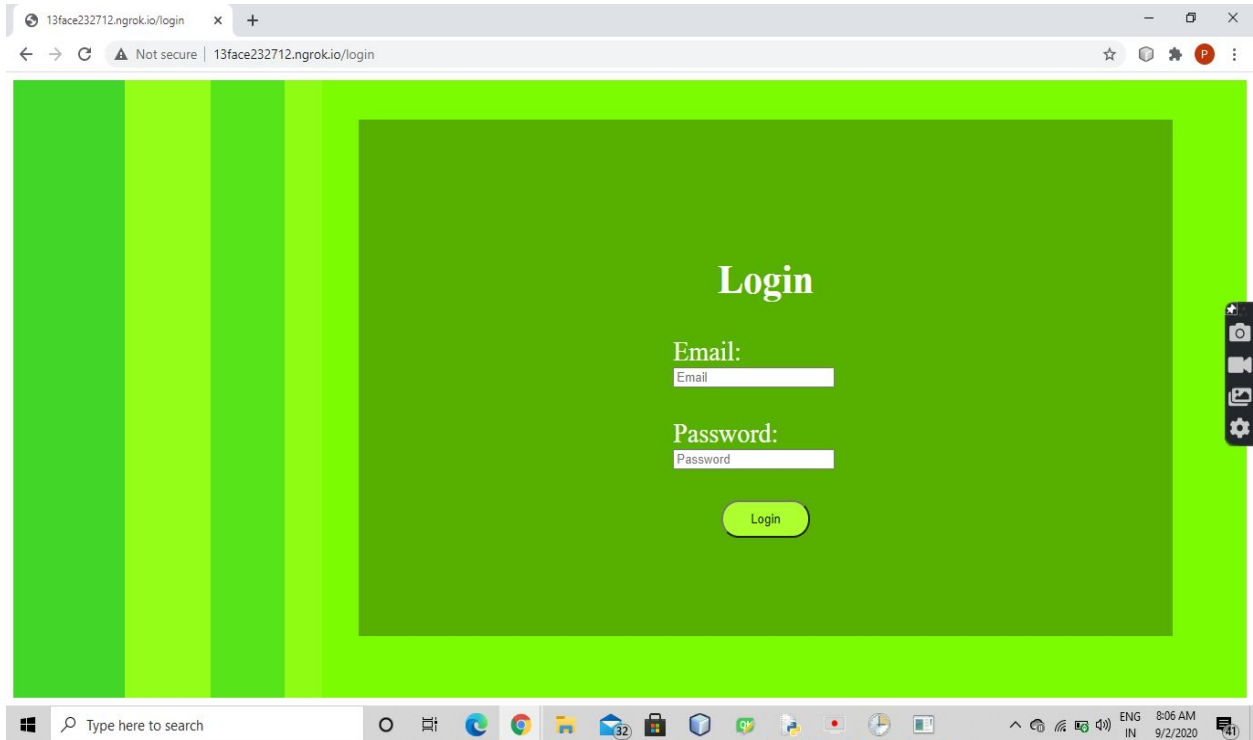
Anna helping user to search Events

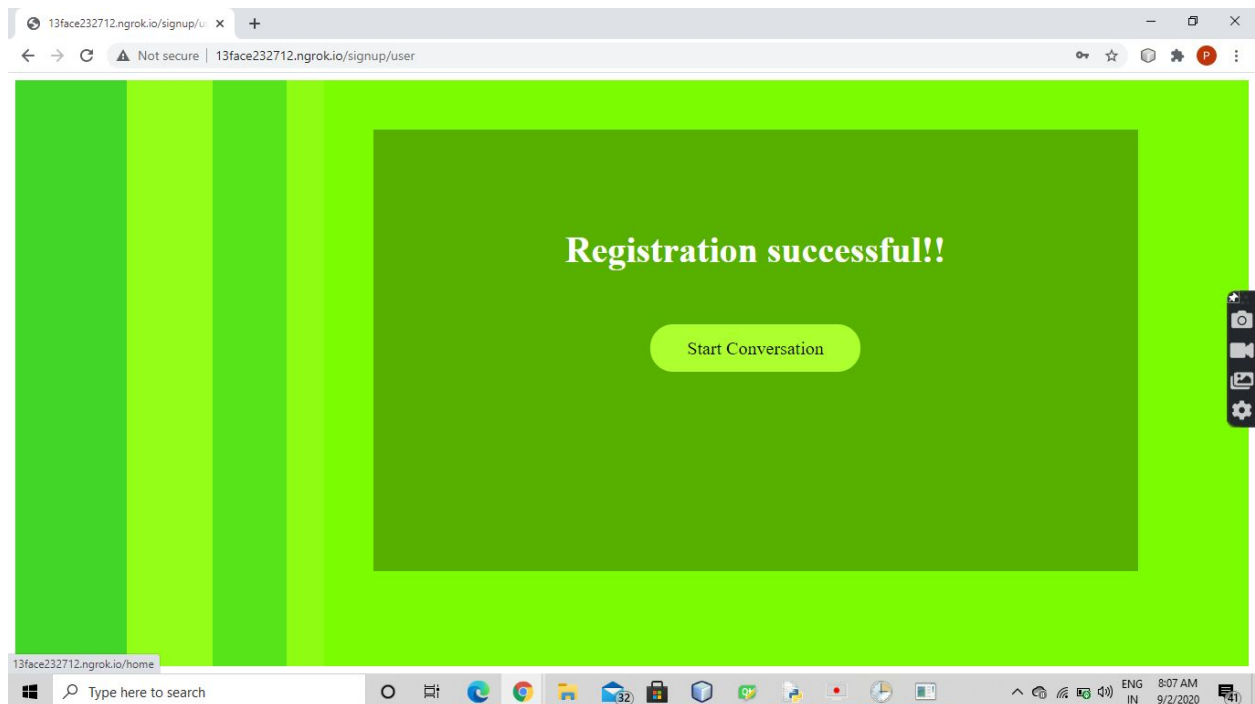
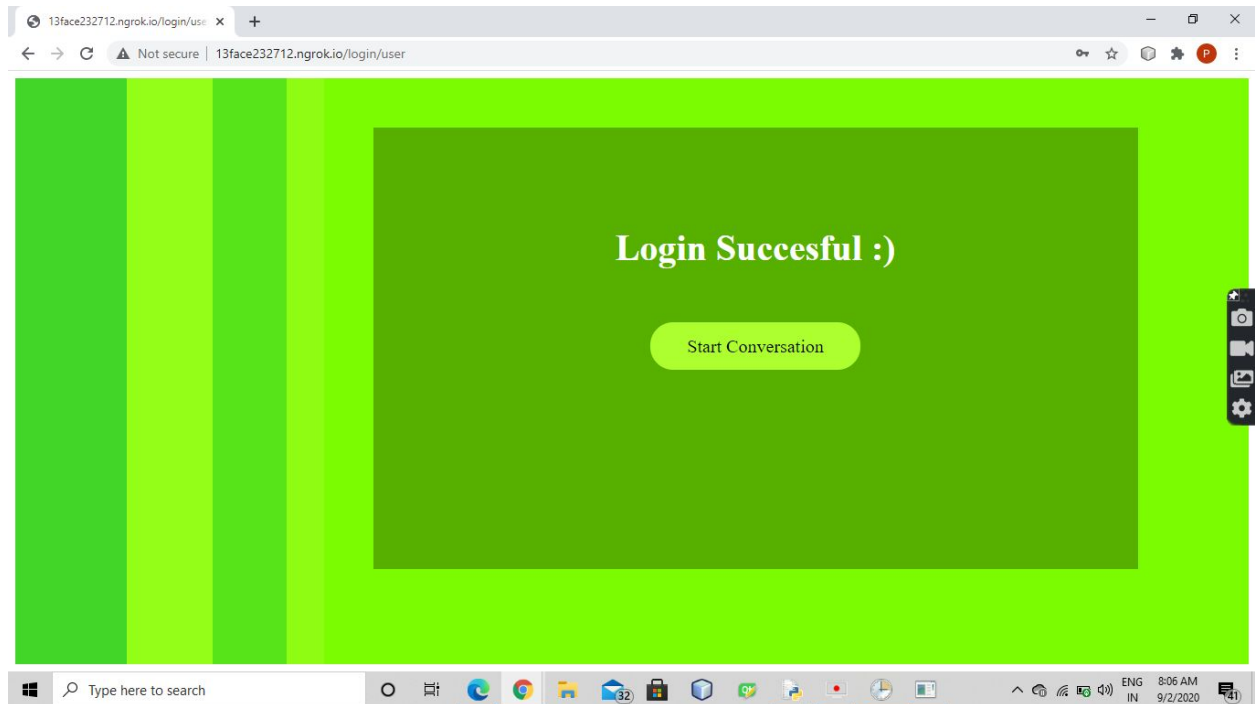
```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
I cannot hear you
timeout
127.0.0.1 - - [02/Sep/2020 07:55:52] "[37mGET /home HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 07:55:57] "[37mGET /user/events HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 07:56:10] "[37mPOST /user/data HTTP/1.1][0m" 200 -

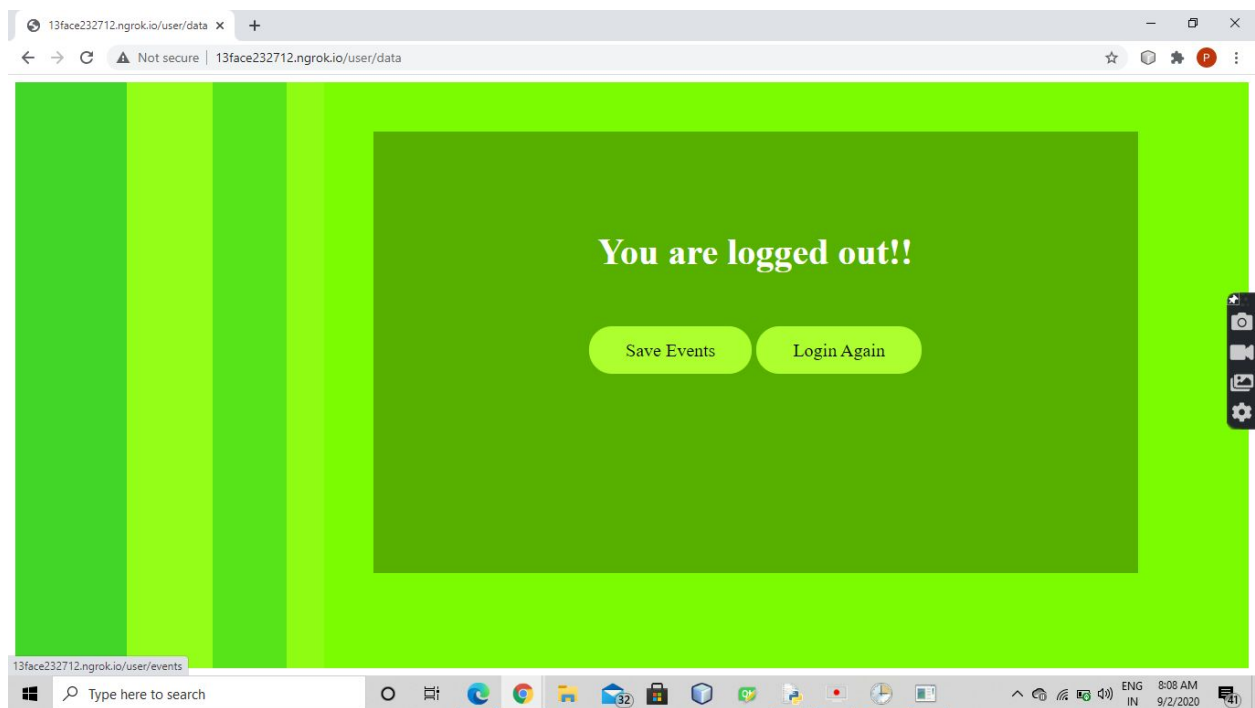
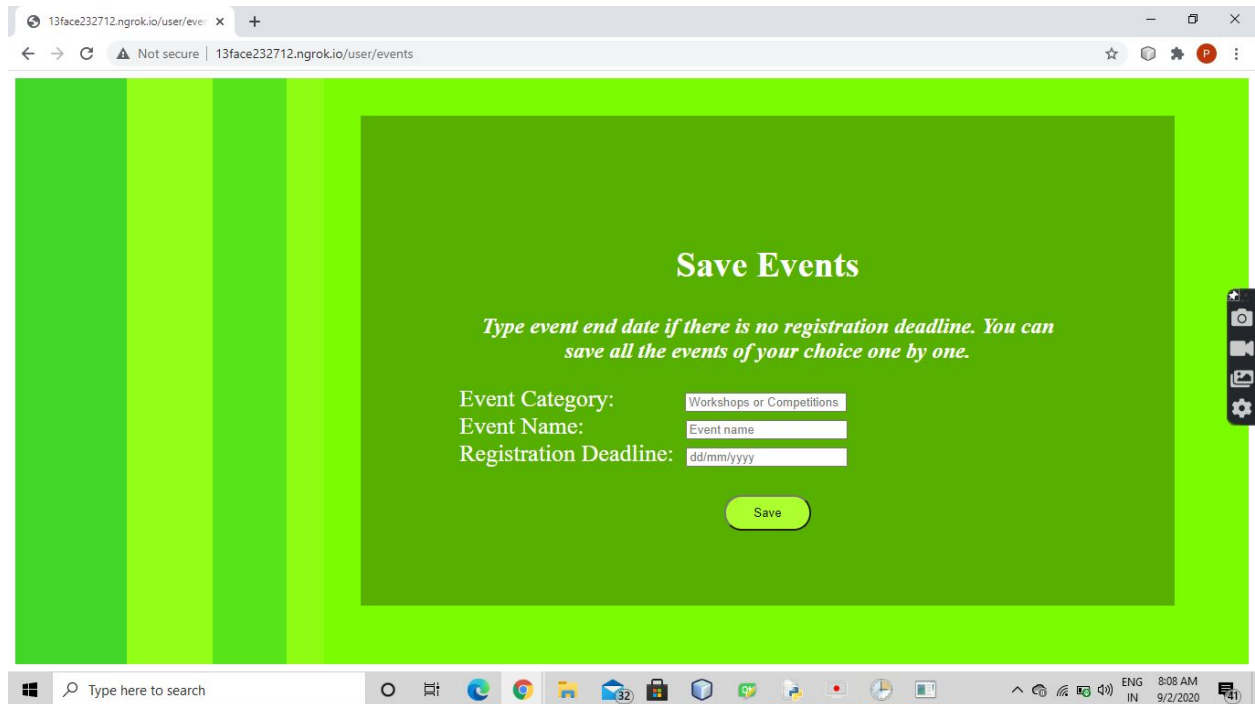
===== RESTART: C:\Python36\pp2.py =====
>>>
===== RESTART: C:\Python36\pp2.py =====
Traceback (most recent call last):
  File "C:\Python36\pp2.py", line 43, in <module>
    book.save('User_Events.xlsx')
  File "C:\Python36\lib\site-packages\openpyxl\workbook\workbook.py", line 392, in save
    save_workbook(self, filename)
  File "C:\Python36\lib\site-packages\openpyxl\writer\excel.py", line 291, in save_workbook
    archive = ZipFile(filename, 'w', ZIP_DEFLATED, allowZip64=True)
  File "C:\Python36\lib\zipfile.py", line 1090, in __init__
    self.fp = io.open(file, filemode)
PermissionError: [Errno 13] Permission denied: 'User_Events.xlsx'
>>>
===== RESTART: C:\Python36\pp2.py =====
>>>
===== RESTART: C:\Python36\project_minipython.py =====
* Serving Flask app "project_minipython" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Running on http://13face232712.ngrok.io
* Traffic stats available on http://127.0.0.1:4040
127.0.0.1 - - [02/Sep/2020 08:05:49] "[37mGET / HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 08:05:51] "[33mGET /favicon.ico HTTP/1.1][0m" 404 -
127.0.0.1 - - [02/Sep/2020 08:05:59] "[37mGET /signup HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 08:06:10] "[37mGET /login HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 08:06:30] "[37mPOST /login/user HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 08:06:45] "[37mGET /signup HTTP/1.1][0m" 200 -
127.0.0.1 - - [02/Sep/2020 08:06:58] "[37mPOST /signup/user HTTP/1.1][0m" 200 -
Hello Priya Fel, I am Anna, your personal assistant, to help you with your online events registration process.
Tell me, what do you want me to search? Competitions, Workshops, or Projects
```

Navigation Pages









----- end -----