Vaccine Slot Notifier

Semester- VII

Summer Internship Report

Submitted by

Name: Karm Patel

Enroll No: 180170107077

COMPUTER ENGINEERING DEPARTMENT

VISHWAKARMA GOVERNMENT ENGINEERING COLLEGE

CHANDKHEDA

External Guide:

Prof. K. K. Rana

ASSISTANT PROFESSOR

VGEC, Chandkheda.

Internal Guide:

Prof. K. K. Rana

ASSISTANT PROFESSOR

VGEC, Chandkheda.

Gujarat Technological University

Academic Year 2021-22

Vishwakarma Government Engineering College, chandkheda

Computer Department

CERTIFICATE

Date: 11/06/2021

This is to certify that the Summer Internship Report entitled **Vaccine Slot Notifier** Submitted by Enroll No: **180170107077** Name**: Karm Patel** towards the fulfillment of Subject: Summer Internship (3170001) of Gujarat Technological University is the record of work carried out by him under our supervision and guidance in the Academic Year 2021-22.

Internal Guide	Head of Department
Prof. K. K. Rana	Prof M. T. Savaliya
Assistant Professor	Associate Professor
VGEC Chandkheda	VGEC Chandkheda

TABLE OF CONTENTS

ACKNOWLEDGEMENT

With great pleasure, I take this opportunity to express my deep sense of gratitude and indebtedness to my renowned and esteemed guide **Prof. K. K. Rana** Assistant Professor, Department of Computer Engineering, Vishwakarma Government Engineering College, Chandkheda for his consummate knowledge, due criticism, invaluable guidance and encouragement which has enabled us to give present shape to this work.

I am heavily indebted to HOD **M.T Savaliya**, Professor & Head, Department of Computer Engineering, Vishwakarma Government Engineering College, Chandkheda, for his everlasting willingness to extend his profound knowledge and experience in the preparation of this report. Any attempt to define this indebtedness would be incomplete.

Finally, I would like to thank our friends and family for their support and patience, and other faculty members of the department for their everlasting willingness to extend their support and help in the completion of this work. Especially to our parents who without their encouragement and financial support, this would not have been possible.

Yours Sincerely,
Karm Patel (180170107077)

Abstract

Vaccination is the only solution for this COVID-19 pandemic. Indian Government started vaccination for age group 18-44 on 1st May 2021. Due to undecided time of availability of vaccine slots, less number of vaccine slots, and a large number of vaccine takers, people wasting too much time checking the Co-win server for vaccine slots. To avoid these difficulties I proposed a vaccine slot notifier web app that takes needed details from users and notifies them through mail when vaccine slots are available in his/her area. I deployed a web app on Amazon Web Service (Cloud computing platform) Elastic Cloud Computing (EC2) instance. Till date, more than 250 users registered on this web app and used the service of this application.

List of Tables

List of Figures

1. Introduction

Cowin is the government website portal to handle vaccine-related information like booking of slots, a dashboard for vaccination, reports for vaccination, FAQs, an admin portal, etc. Users can register in the portal via entering identity details and uploading proofs . One can find vaccine slots in the Co-win portal by entering either state and district or pin-code of his/her area. If a vaccine slot is open in his/her area then one has to select a time and vaccination center to book an appointment. There are two categories of problems, first is, as slot booking speed by people is more than the refresh speed of indicator to show vaccine slots. So though vaccination slots booked, an indicator shows old count since it is not refreshed, so after selecting the vaccination center, time and entering captchas, if one clicks on the book button, a message "slots are not available" appeared on the screen. Now one already invested 1-2 minutes on this booking so he/she is not able to book other slots also as all slots are booked within a minute. The second is, Government didn't decide on a specific time to update slots on the Co-win portal. So concerned people are trying a whole day to book a slot, which really tedious task. To address this problem people need some technical solutions to make their work easy.

As one can not control people’s crowd to book slots or can not increase indicator speed of the Co-win portal. So solution of 1st problem is difficult to tackle. But to solve the second problem, I proposed a vaccine slot notifier which notifies with basic details of slots like Pincode, district, slot place, date, etc through mail to the user. This system reduces the manual effort of people as well as saves time. As I made user flexible webpage and deployed it to an AWS server, anyone can easily access it and subscribe to this service.

2. System Analysis

Before building a solution for any problem requirements of users must be clear in minds of developers. The study of the current system is also a crucial thing as developers can know what’s going on in the current system, technologies used, features, problems, and weaknesses of the current system.

2.1. Study of current System

As Vaccination is a crucial thing in this pandemic period and people have high concerns to save their life, many technical private organization or individual techies proposed their system to make this vaccination drive flexible to users. Among one of the famous solutions is under45.in created by Chennai-based developer Berty Thomas which alerts users via telegram messaging app. A user needs to submit her/his state and district to get alerts on Telegram when vaccination slots become available at a center near them through a dedicated channel. The message has information on which center has available slots, the number of slots available, the date of availability, and which brand of vaccine (Covishield or Covaxin) is available there.

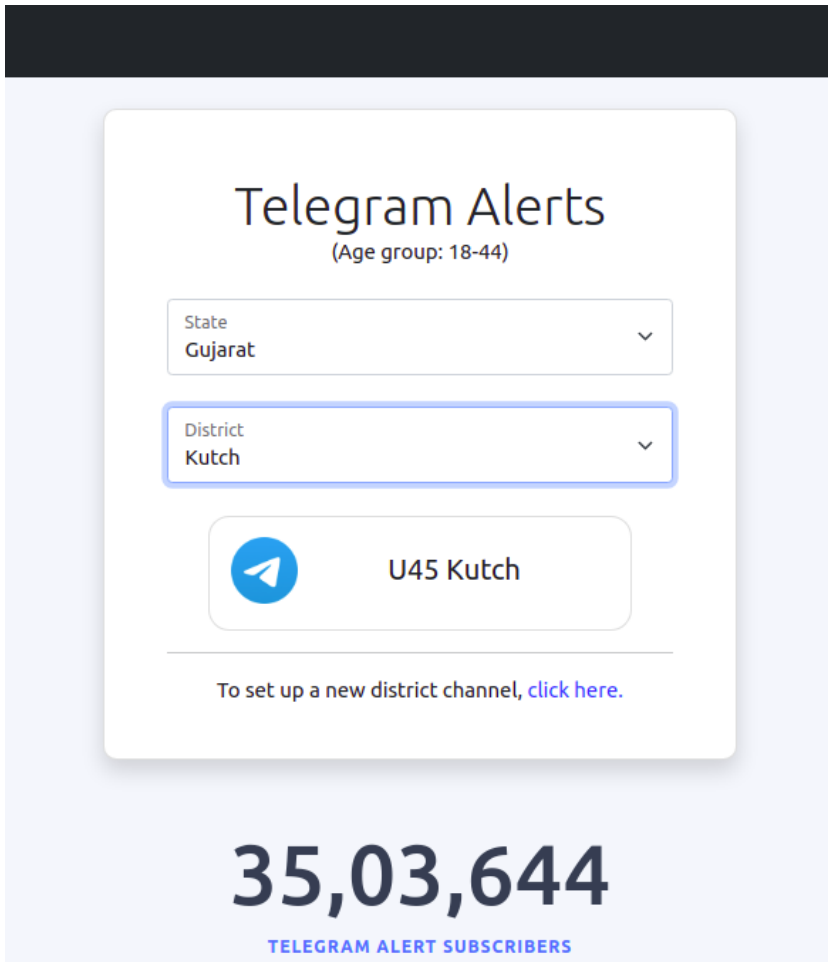

Figure 1 - under45.in subscribe form

another similar website like vaccinateme.in, findslot.in, and getjab.in also tried to give solutions to the problem by notifying through SMS, mail, or WhatsApp [1].

2.2. Problem and weakness of Current System

All these websites gave amazing effort to solve the problem of vaccination drive, but still, there are some glitches in every system. Under45.in does not provide service for all states & districts. findslot.in is depends on public API for Co-Win which had stooped to give real-time data for vaccine slots, so unfortunately this organization stopped their service. getjab.in crossed their user limit so they can't be able to take new users.

2.3. Requirement analysis of New System

By analyzing the current system and its weaknesses, I concluded the following requirements for users.

- The system should be user-friendly. it should use very common services like Gmail, SMS, or WhatsApp, so maximum users can take advantage of it.
- The system should notify the user as soon as possible after the availability of demanded slot in the area.

- The system should not depend on external API as API can stopping the service. In short, There should be minimum 3rd parties between the System and the Co-Win portal.
- The system should have a strong server and bandwidth to handle a large number of users.

2.4. Design: Analysis, Design Methodology, and Implementation Strategy

After analyzing system requirements I can broadly classify design in the following steps.

1. **Scrapping real-time data from Co-Win server:**
   - To collect vaccine slots data we need one pipe which brings data from the Co-Win server to our application. In short, we need to make a web scrapper by considering network bottlenecks like requests limit per minute.
   - This scrapper should run 24 hours to check iteratively on the server to check whether vaccine slots are available or not.
   - If available then the scrapper should notify a user through the mail with slot details.

2. **Make a webpage to get user data:**
   - To use this system there is necessary to have a front end that is accessed by a user.
   - In this system, I will use a web page to collect user data like email, Pincode, state name, district name, gender, age group, etc.
   - A web page should also provide unsubscribe facility, so once a user gets the vaccine he/she can unsubscribe from the service to avoid unnecessary mails.

3. **Deploy the System on a cloud platform.**
   - The system should be deployed on a cloud server so users from anywhere and anytime access it.

3. System Modelling

3.1: Data Flow Diagram

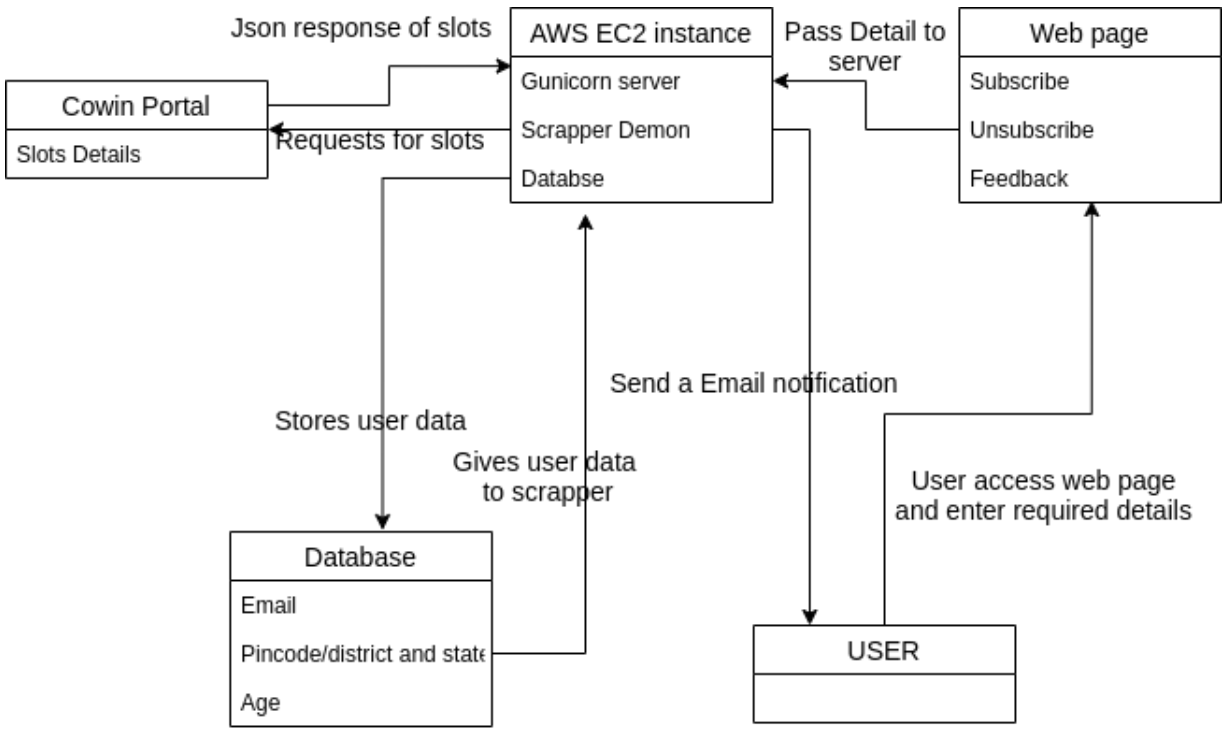

Figure 2 - Data flow diagram

3.2: System Architecture

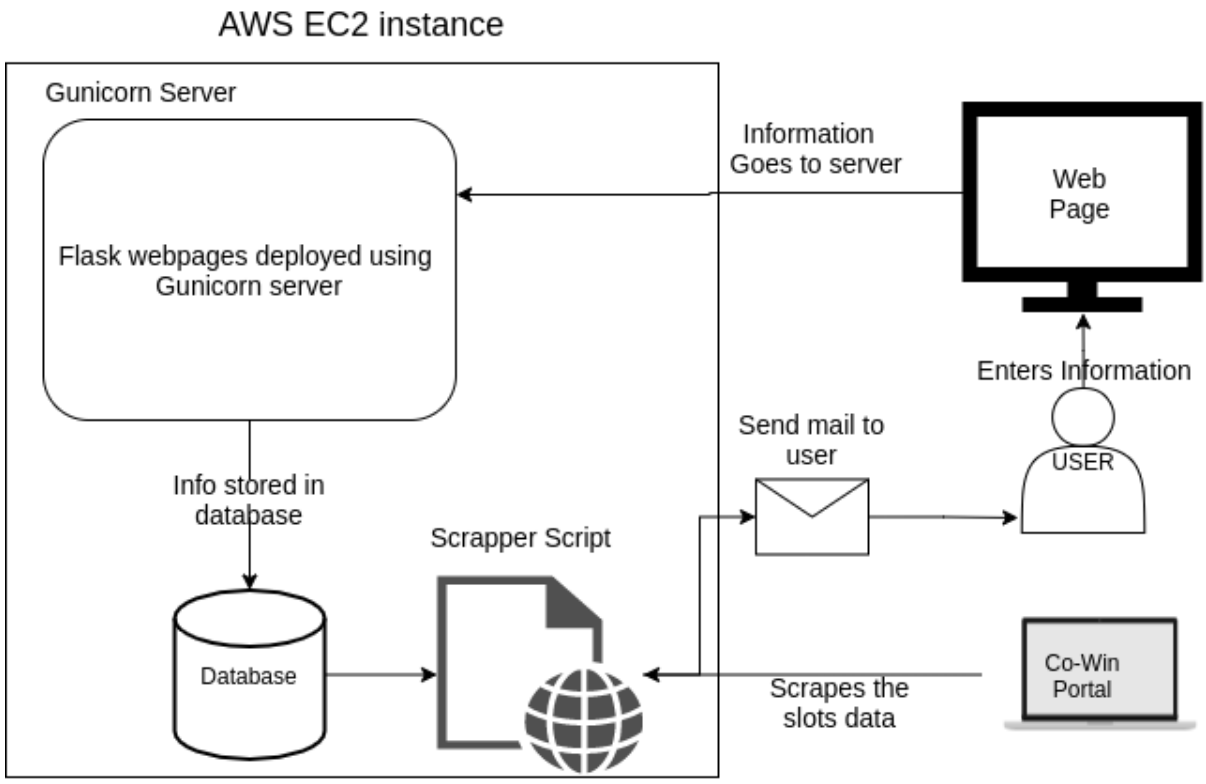

Figure 3 - System Architecture

4. Implementation

- As I described in section 2.4 that to implement this system I divided the process into mainly 3 steps. Implementation of these steps is discussed below.

4.1: Implementation steps and snapshots:

1. Scrapping real-time data from Co-win server

- I have used python libraries named requests and beautiful soap to scrape the data from the Co-win server.
- This process is iterative which means this script will make a request in an interval of 5 seconds to a server in order to get a JSON response.
- As per the policy of the Co-win server if I reduce this time interval then my server gets blocked.
- In short, this is a python script that will run in the background for 24 hours and check whether slots are available or not.
- If slots are available then it will send mail to a user using the flask-mail service.

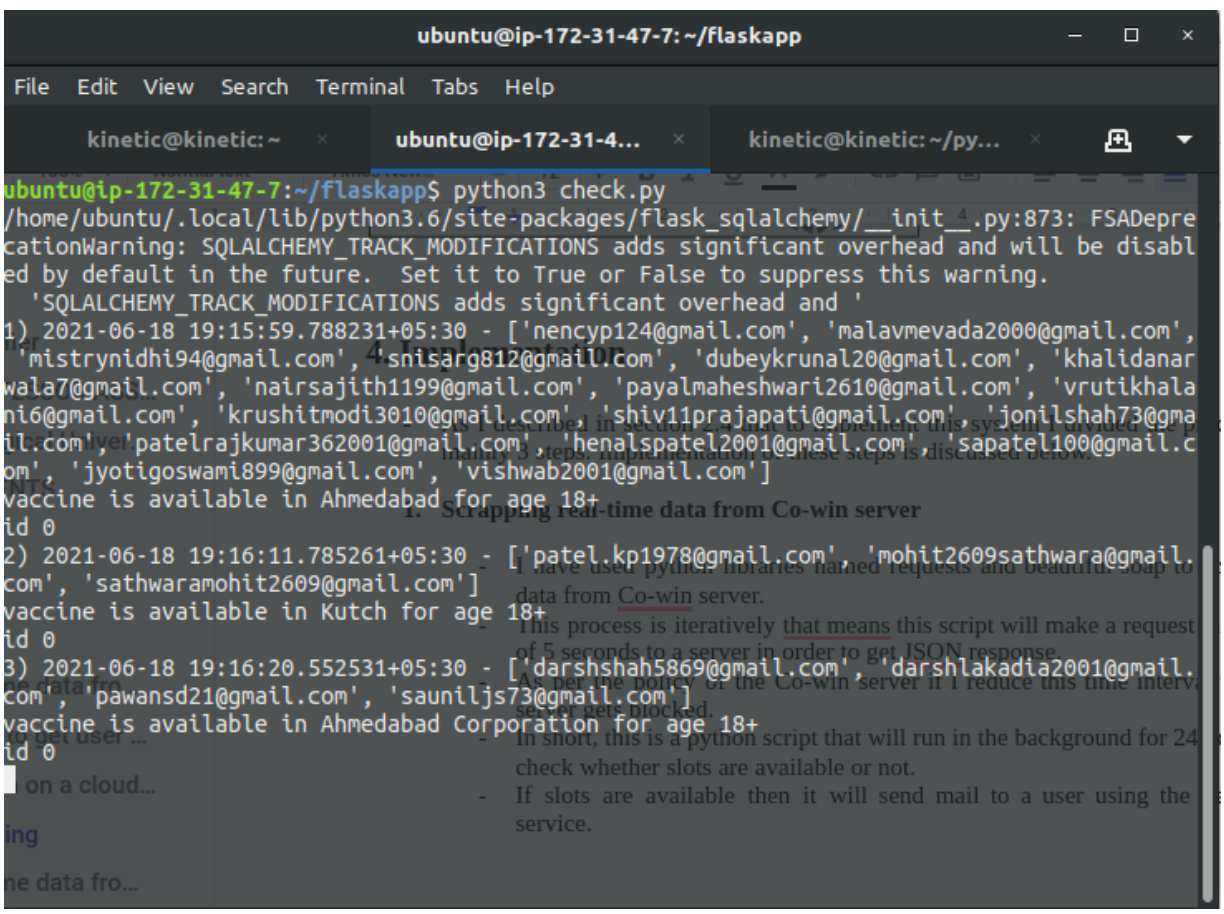

Figure 4 - scrapper demon logs

Figure 5 - Gmail notification sample

2. **Make a webpage to get user data::**

- To make a web page I have used Html, CSS, javascript, jquery, and bootstrap technologies for the frontend and Flask framework for the backend.
- To store the user data I am using flask database service sql-alchemy.
- A webpage is classified into 3 subpages

1. **Subscribe :**
   - This is the main page that collects the required details from the user.
   - As shown in the figure subscribe page contains fields like email, search by, Pincode, state, and age group. where search by option can be used to select option for searching between Pincode or district as co-win also provide such options.
   - Districts dropdown menu is dependent on state i.e. dropdown data changes as a user selects state.

2. **Unsubscribe:**
   - After satisfied with this service user can unsubscribe by entering an email id to avoid unnecessary emails.
3. **Feedback:**
   - If a user wants to report bugs, to give comments or feedback then this option can be used.

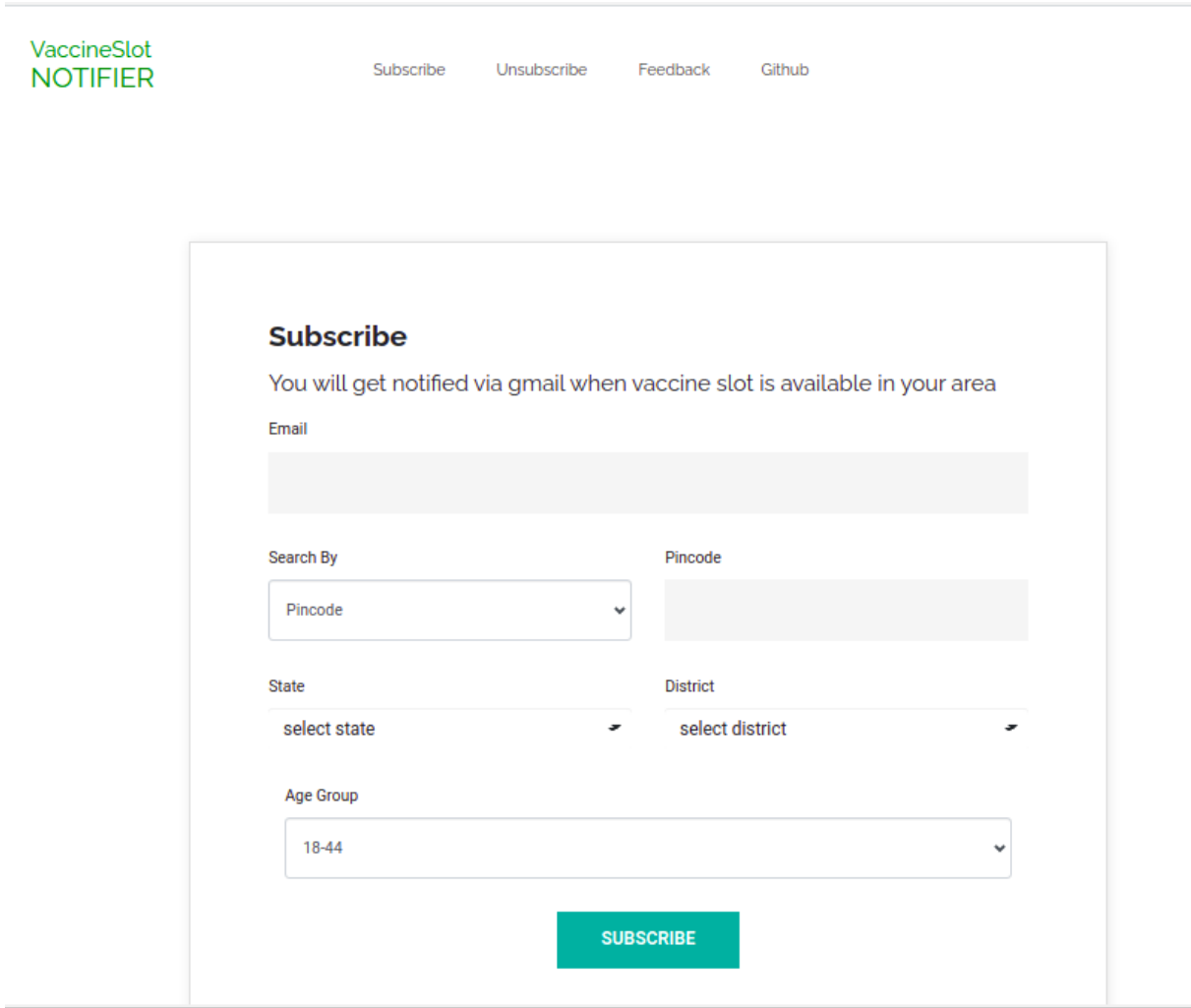

Figure 6 - Subscribe Form

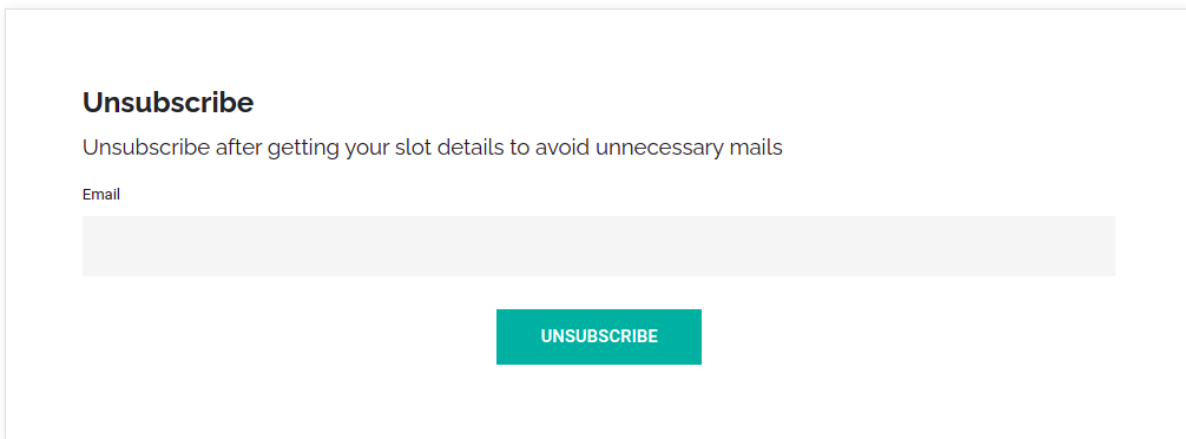

Figure 7 - Unsubscribe Form

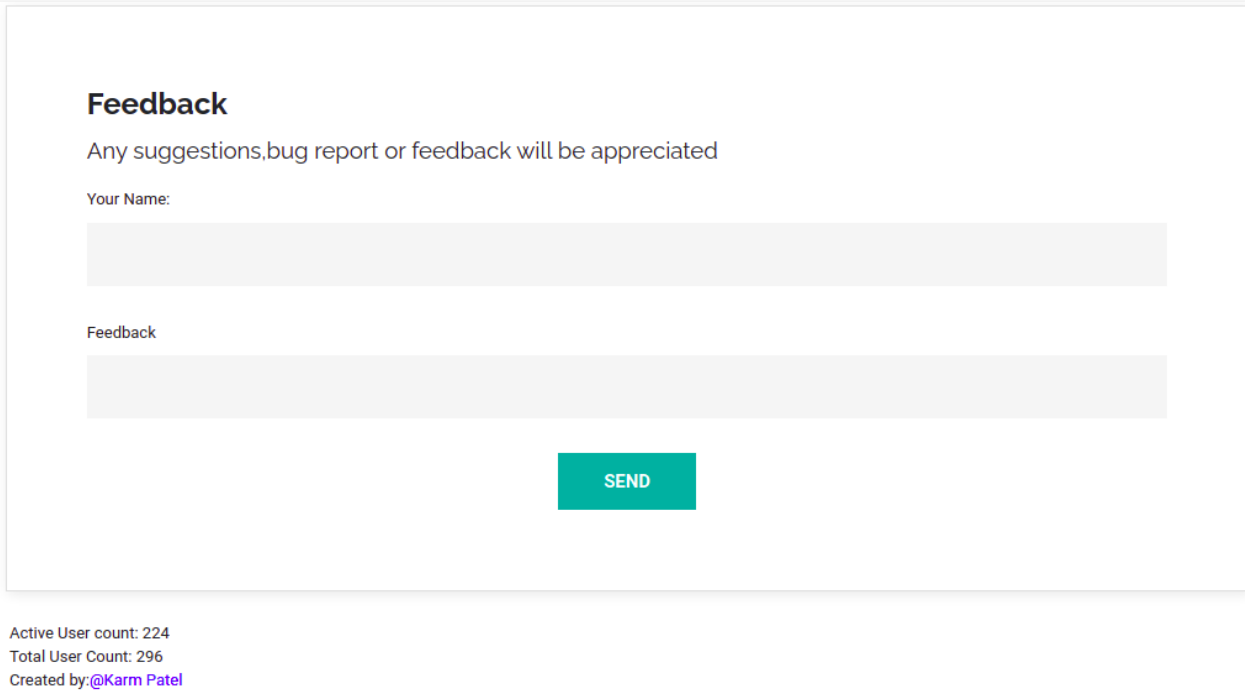

Figure 8 - Feedback Form

3. **Deploy the System on a cloud platform.**

   - Without deploying service to cloud platform system is useless, because no one can access service except developers.
   - I have a used **Amazon Web Service** as a cloud platform and using an EC2 instance to deploy a web server.
   - **Gunicorn** server is used to host flask web pages.

4.2: Database Table structure:

- SQL (Structured Query Language) database is used to store the data as it is simple to store and implements for structured data.
- As this is not a complex project in terms of database, only 1 schema table is used which structure is as follows.

id	INTEGER
by	VARCHAR(20)
email	VARCHAR(20)
pin	VARCHAR(20)
state	VARCHAR(20)
district	VARCHAR(20)

min_age	VARCHAR(20)

5. Conclusion and Future Scope

A vaccine slot notifier is an example of how problems of society can be solved using technology. This system reduces manual work and saves people time. However, while building this type of system we should follow a technical policy of the host website for example, in this case, we should follow the scrapping policy as Co-win mentioned. The primary goal of the system is to help people not stealing vaccine slots from the server. Moreover, techies also tried to make scripts to book vaccine slots automatically however, after adding the captchas feature some of these scripts failed. but strill some exist, this is unethically from my point of view. By making this kind of script we are grabbing vaccine slots from the hands of normal people.

Below are some improvements that can be applied in the future release of this product.

1. Scrapper is using only 1 CPU so if users increases then the time between two check request of the same place will be increased. By using multicores this problem can be solved.
2. Many users remain subscribed even if they don’t need service, auto unsubscribe can be applied using some algorithms.
3. Another drawback is anyone can unsubscribe by entering one’s email id. Security can be improved by implementing a login/signup feature.
4. A current system using only 1 IP address to scrape a Co-win server, by applying proxy IPs of India we can reduce interval between two requests.