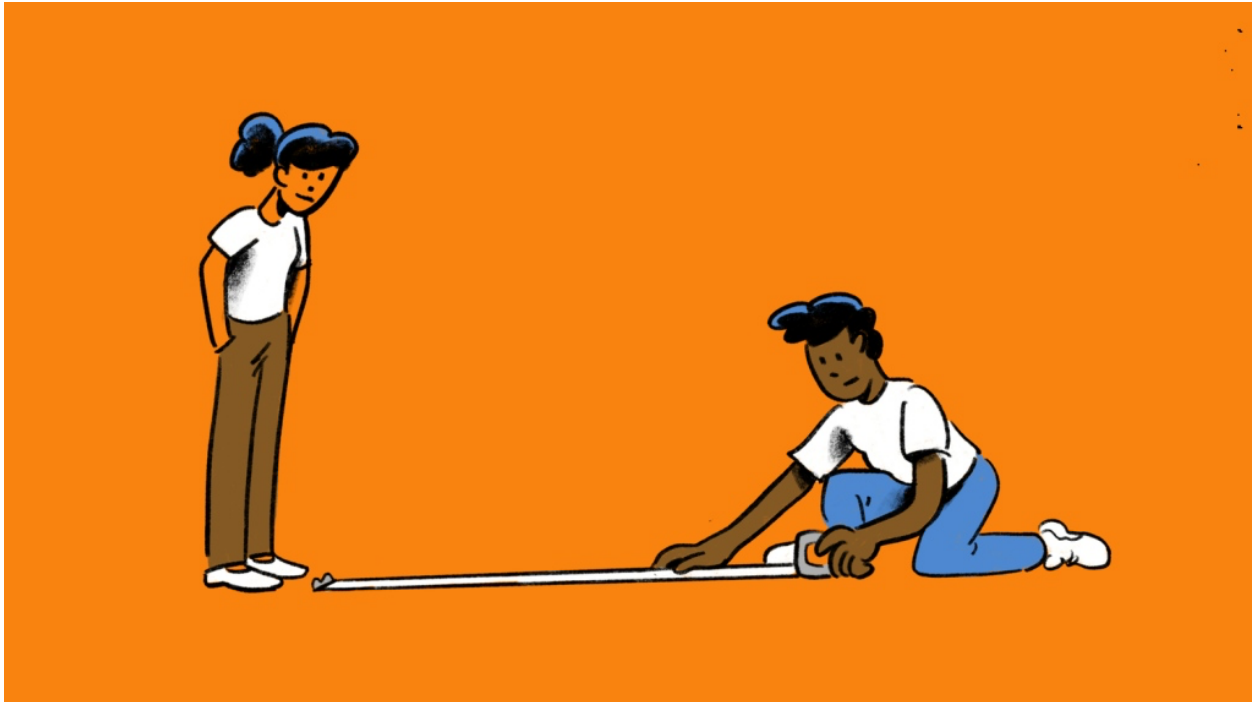


DEPLOYMENT MANUAL



Prerequisites:

1. Ensure that you have the necessary credentials and access to a server.
2. Install Git on your local machine to clone the repository.

Step 1: Clone the Repository

```
```bash
git clone < >
cd <your_project_directory>
```
```

Next step-

Ho x | re: x | ha: x | axi x | Pa x | Lo x | Wi x | Ba x | ISO ISC x | Me x | ap x | Me x | ny x | Cl: x | iB: x | htr x | +

github.com/htmw/2023S-Team4/tree/main

🔍 Type / to search

+ - 🔍 📁 📧

htmw / 2023S-Team4

< > Code Issues Pull requests Actions Projects Wiki Security Insights

🔖 2023S-Team4 Public

👁 Watch 2 🍴 Fork 0 ⭐ Star 0

main 3 branches 0 tags

hanith2646 Add files via upload

| | | |
|---|----------------------|--------------|
| 📁 .github/workflows | Create android.yml | |
| 📁 api | Add files via upload | |
| 📁 covid-predictor | django | |
| 📁 documents | Add files via upload | |
| 📁 presentations | Add files via upload | |
| 📁 retrospective | Add files via upload | |
| 📁 sprint05 | Add files via upload | |
| 📁 sprint06 | Add files via upload | |
| 📁 team members_group_4 | Add files via upload | |
| 📄 .DS_Store | django | |
| 📄 SPRINT 3 | Create SPRINT 3 | 7 months ago |
| 📄 Team Agreement_group4.pdf | Add files via upload | 9 months ago |
| 📄 cs 619 sars presentation final (100)... | MAIN PPT 1 | 9 months ago |
| 📄 f1 | created f1 file | 7 months ago |

Go to file Add file <> Code

Local Codespaces

📄 Clone

HTTPS SSH GitHub CLI

https://github.com/htmw/2023S-Team4.git

Use Git or checkout with SVN using the web URL.

📄 Open with GitHub Desktop

📄 Download ZIP

📄 Code 55% faster with AI pair programming.

Start my free trial Don't show again

About

Spring 2023 - Fall 2023

🔍 Activity

⭐ 0 stars

👁 2 watching

🍴 0 forks

Report repository

Releases

No releases published

Create a new release

Packages

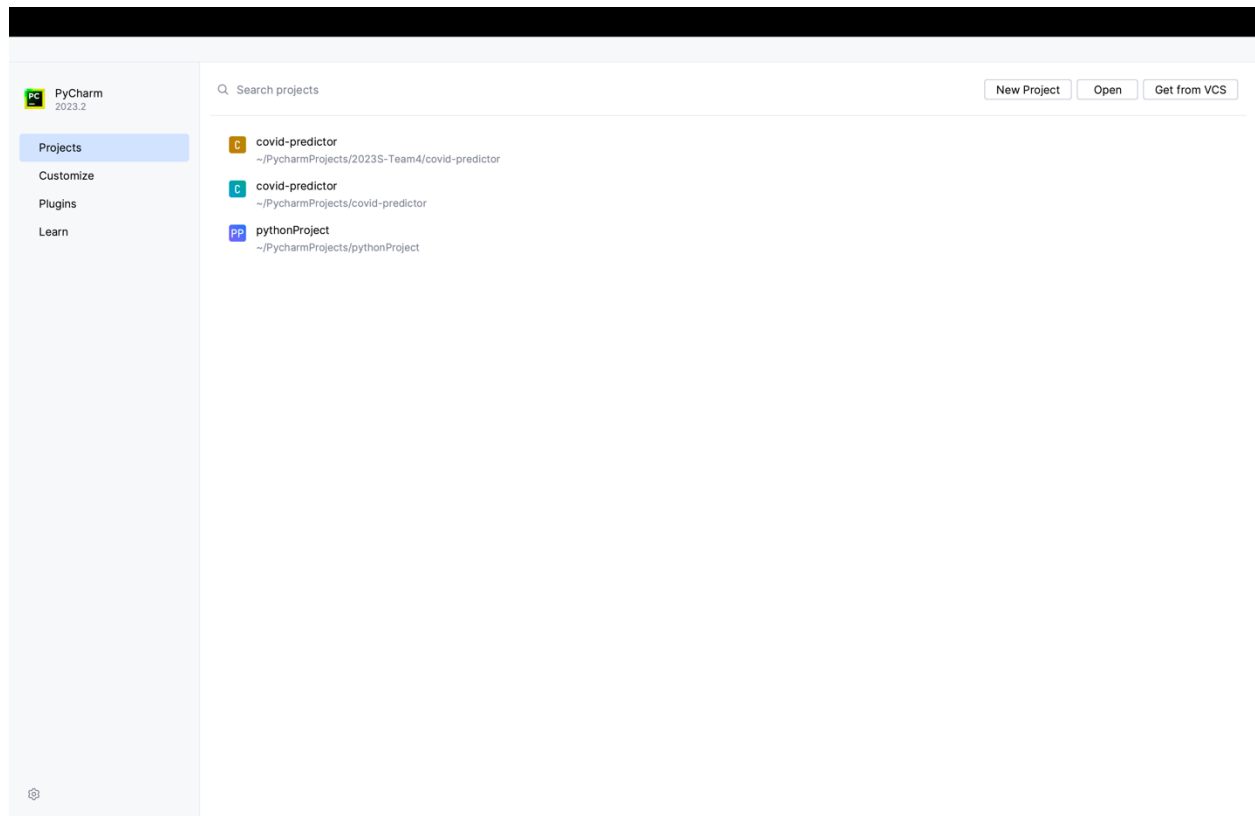
No packages published

Publish your first package

Contributors 5

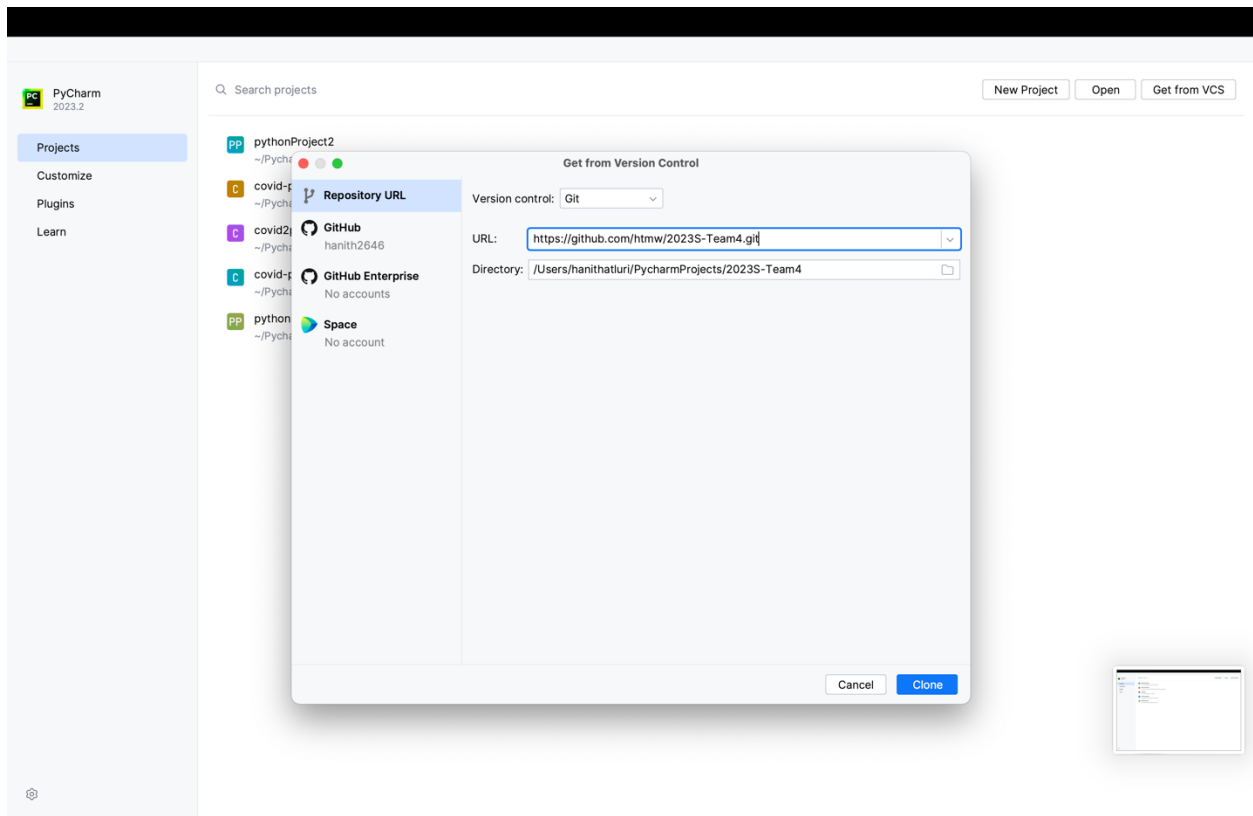
👤 👤 👤 👤 👤

Next step-



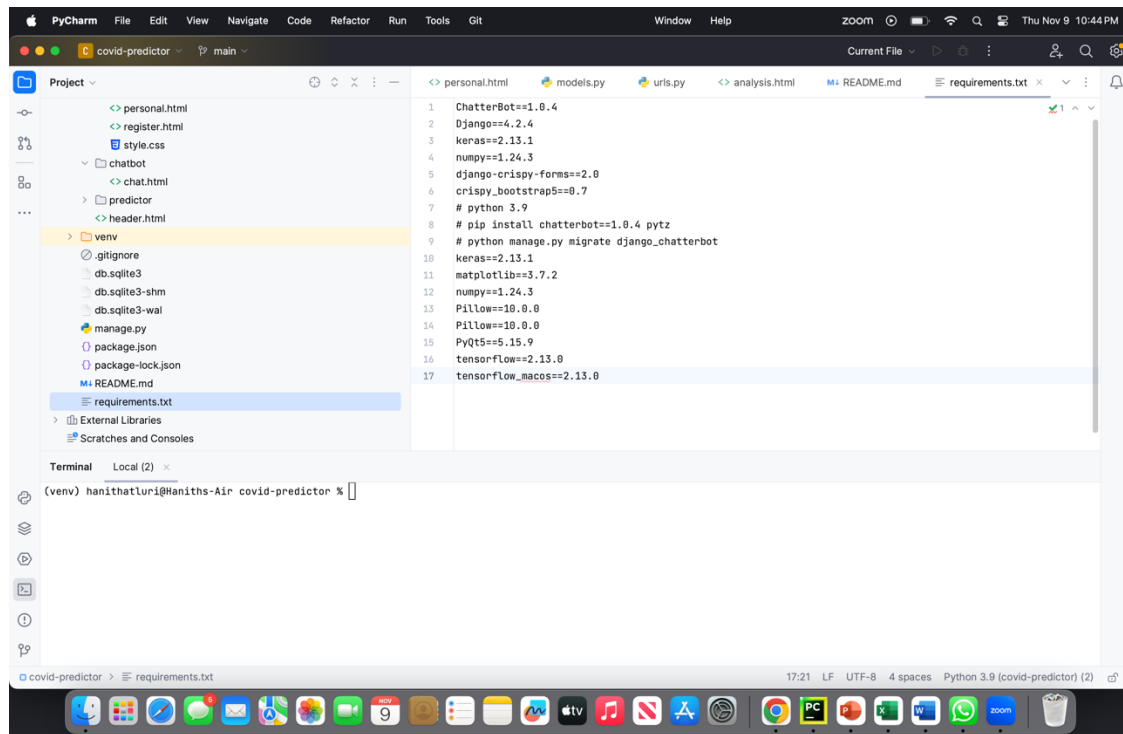
Next step-

Copy and paste the link in the open fromVCS and then click clone to clone the code from the github to your desired folder

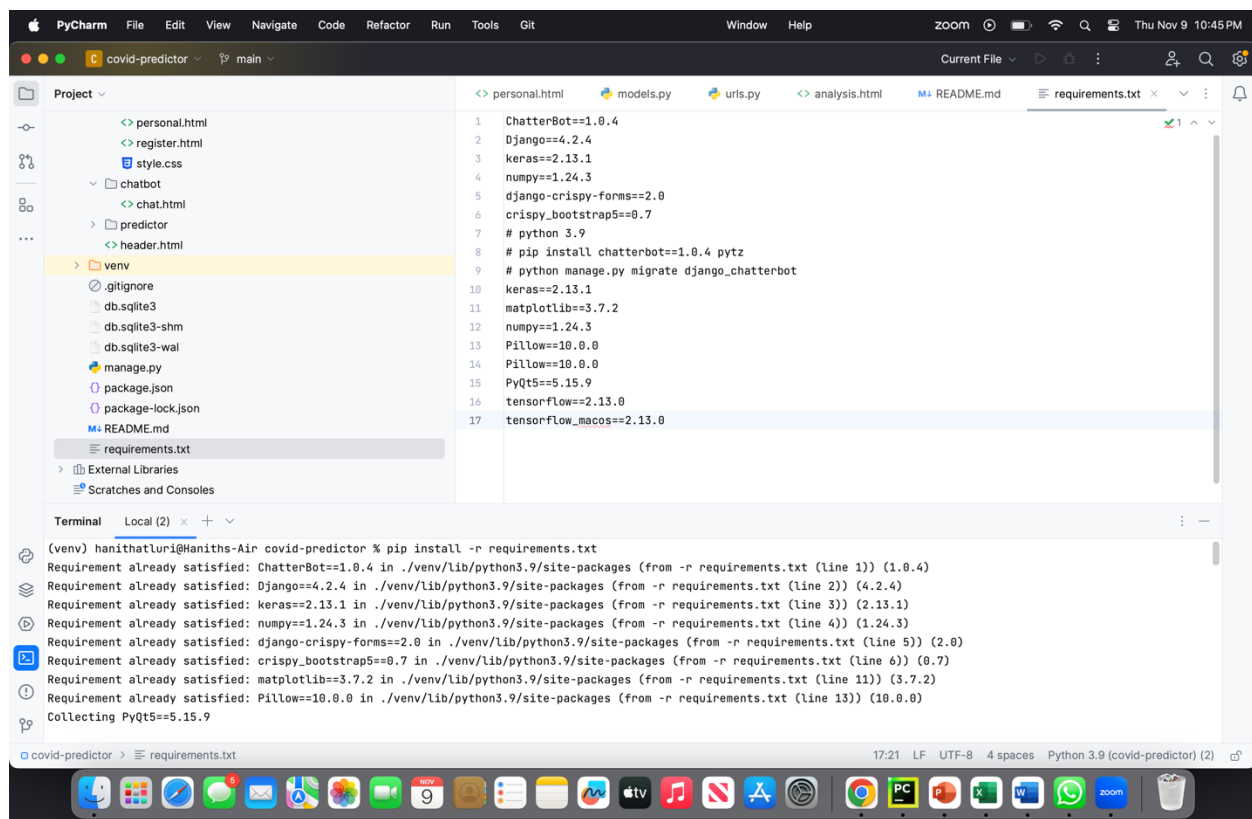


Step 2: Install Dependencies

```
```bash
pip install -r requirements.txt
```
```



Open terminal and then install all dependencies



Step 3: Database Migration

```
```bash
python manage.py migrate
```
```

Step 4: Static Files Collection

```
```bash
python manage.py collectstatic
```
```

Step 5: Configure Django Settings

Update your Django settings for production. Update the `DEBUG` setting to `False` and set the `ALLOWED_HOSTS` to the domain or IP address of your server.

```
```python
settings.py

DEBUG = False

ALLOWED_HOSTS = ['your_domain_or_ip']
```
```

Step 6: Set Up a Web Server

Choose a web server to serve your Django application. Common choices include Nginx or Apache. Configure the server to serve your Django app using a WSGI server such as Gunicorn.

Step 7: Configure Database Connection

Update your database settings in the Django settings file to point to your production database. You might use PostgreSQL, MySQL, or another database supported by Django.

Step 8: Set Up ChatterBot

Configure ChatterBot by creating a ChatterBot instance and training it if necessary. Update your Django settings to use the configured ChatterBot instance.

```
```python
settings.py

CHATTERBOT = {
```

```

'name': 'YourChatterBotInstance',
'trainer': 'chatterbot.trainers.ChatterBotCorpusTrainer',
'training_data': [
 'chatterbot.corpus.english',
 # Add additional training data if needed
],
}
'''

```

### Step 9: Set Up Keras and TensorFlow

Ensure that Keras and TensorFlow are configured correctly. If using GPU, make sure GPU drivers are installed on your server.

### Step 10: Configure Crispy Forms and Bootstrap

Update your Django settings to use crispy forms with Bootstrap.

```

'''python
settings.py

CRISPY_TEMPLATE_PACK = 'bootstrap5'
'''

```

### Step 11: Secure Your Application

Implement security measures such as HTTPS, and ensure that your server and application are properly configured.

### Step 12: Test the Deployment

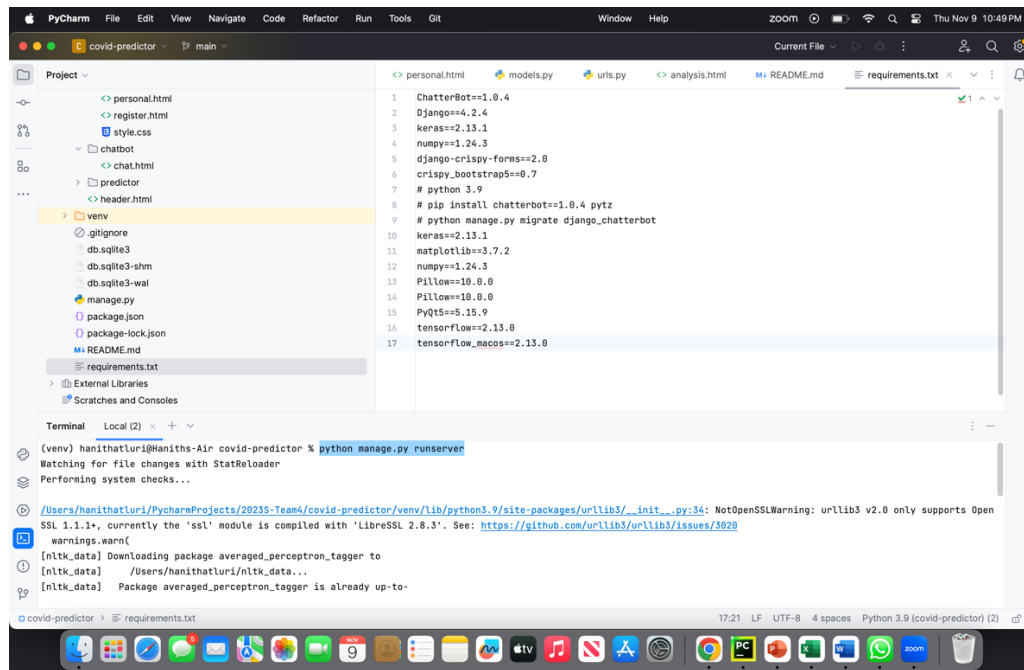
Run your web server and test your application in a production environment. Check for any errors or issues and resolve them.

### Step 13: Monitoring and Maintenance

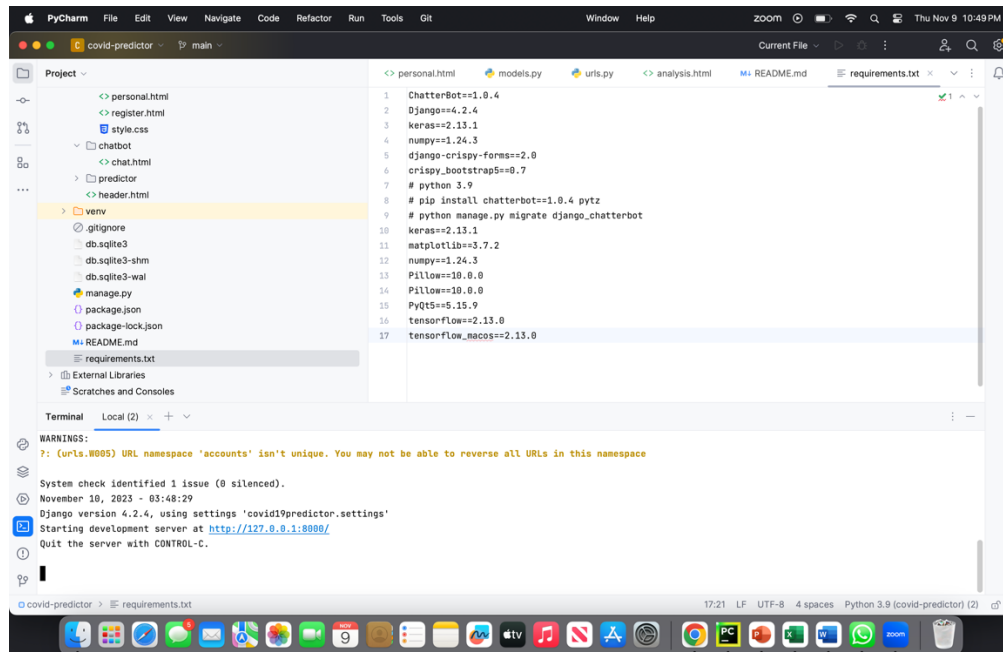
Set up monitoring tools to track the performance of your application. Implement a maintenance plan for regular updates and backups.

### Step- 14: to runserver

Use this command line to run server- **python manage.py runserver**

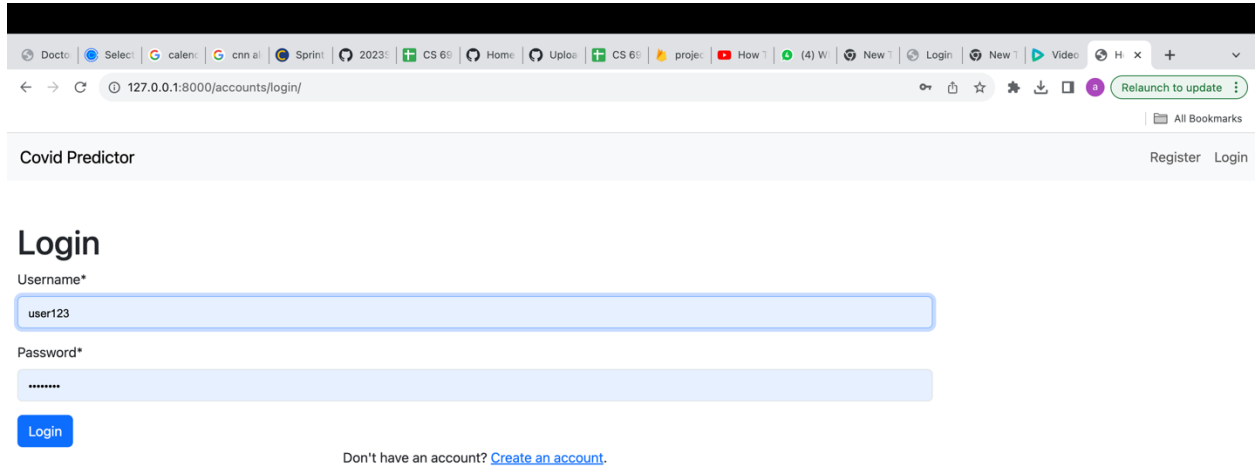


Then application will be started in unique sever-id click on it to to redirect from the default browser





Finally application will be running on default browser



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/accounts/login/`. The browser's tab bar includes several open tabs, and the address bar features a 'Relaunch to update' button. The page header shows the application name 'Covid Predictor' on the left and 'Register Login' on the right. The main content area is titled 'Login' and contains two input fields: 'Username\*' with the value 'user123' and 'Password\*' with masked characters. A blue 'Login' button is positioned below the password field. At the bottom, a link for 'Create an account' is provided for users who do not have an account.

Covid Predictor Register Login

## Login

Username\*

Password\*

Login

Don't have an account? [Create an account.](#)