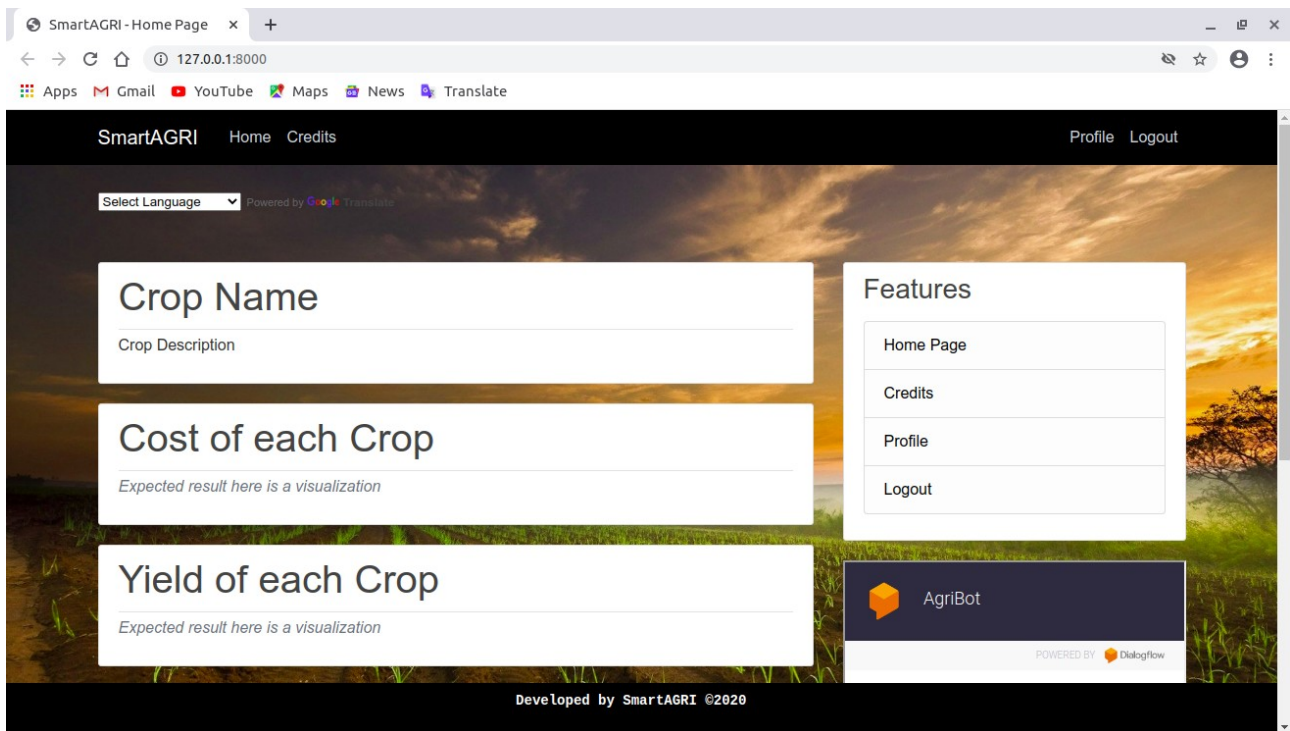


# SmartAGRI users Manual

To setup the web app, use the **configurations.txt** file.

After successfully setting up the web app it should look like this,



And in the terminal it should look like this,

```
nss2000@ns00: ~/GovTechThon20/project
(env) nss2000@ns00:~/GovTechThon20/project$ python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
November 05, 2020 - 11:27:38
Django version 3.0.2, using settings 'django.project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[05/Nov/2020 11:27:43] "GET / HTTP/1.1" 200 7057
```

To achieve the status of ADMIN, follow the below steps,

```
python manage.py createsuperuser
```

Type the above command in the virtual environment.

It will look like this,

```
(env) nss2000@ns00:~/GovTechThon20/project$ python manage.py createsuperuser
Username (leave blank to use 'nss2000'): admin1
Email address: admin1@email.com
Password:
Password (again):
Superuser created successfully.
(env) nss2000@ns00:~/GovTechThon20/project$
```

After performing the above step, the below command,

```
python manage.py runserver
```

And open the url,

```
http://127.0.0.1:8000/
```

Use the Credentials to login to the Admin page of the webapp.

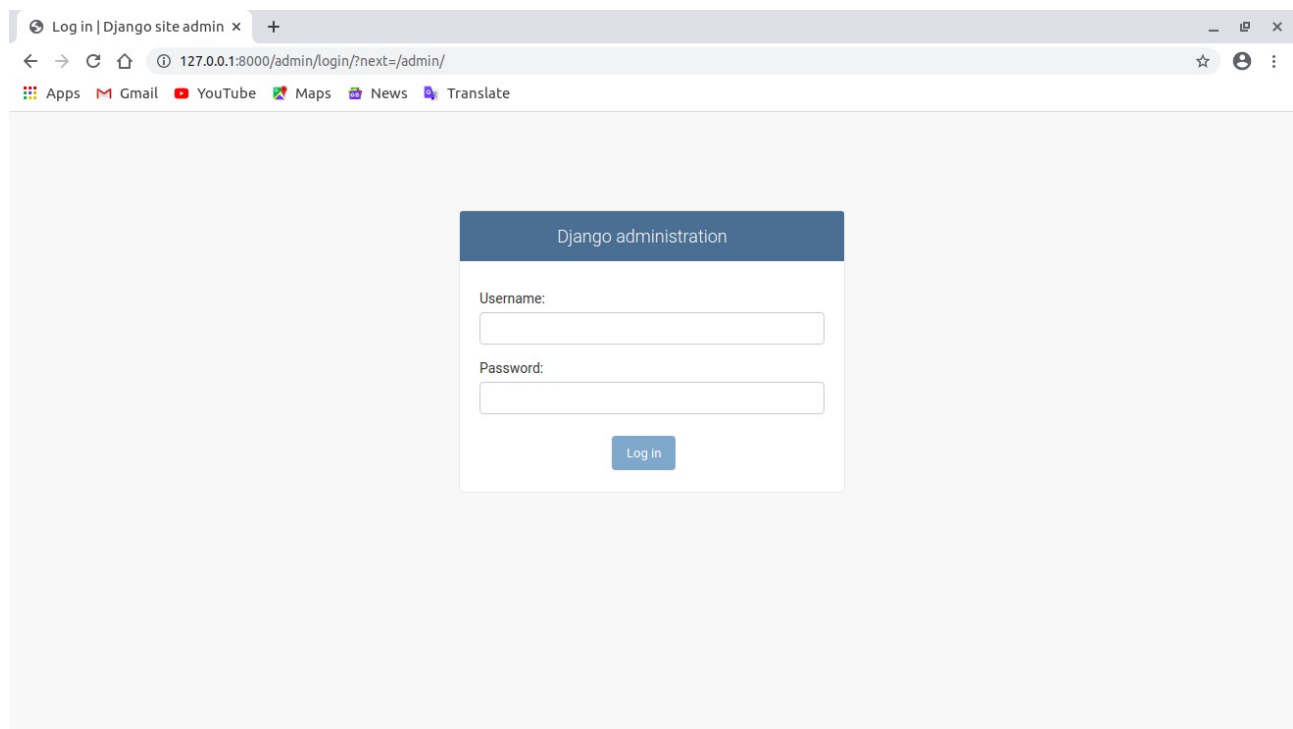
Use the url,

```
http://127.0.0.1:8000/
```

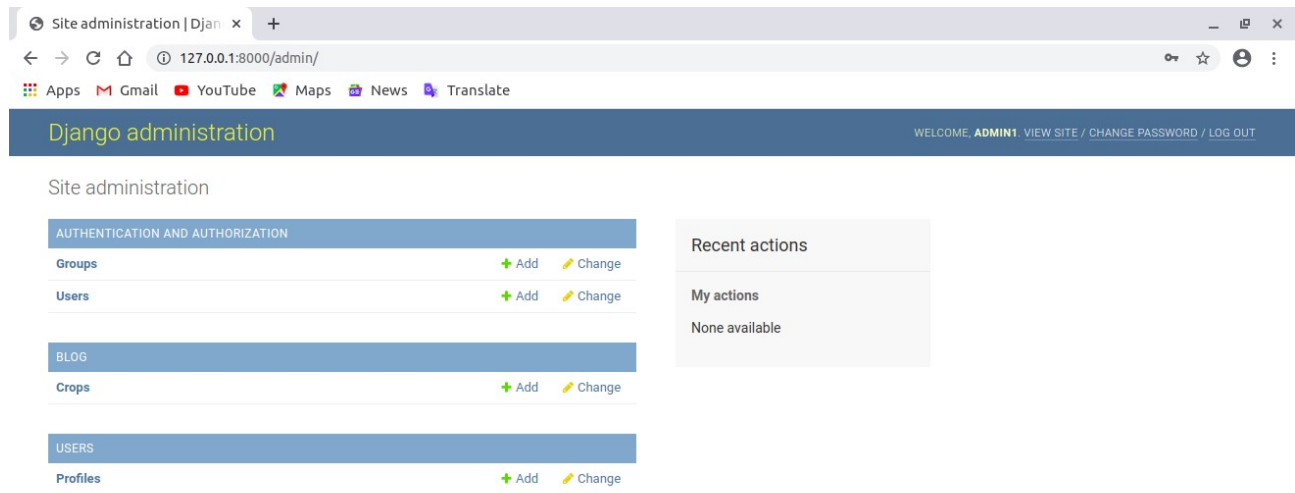
Now type 'admin' at the end of the above url, like this,

```
http://127.0.0.1:8000/admin
```

The browser should look like this,

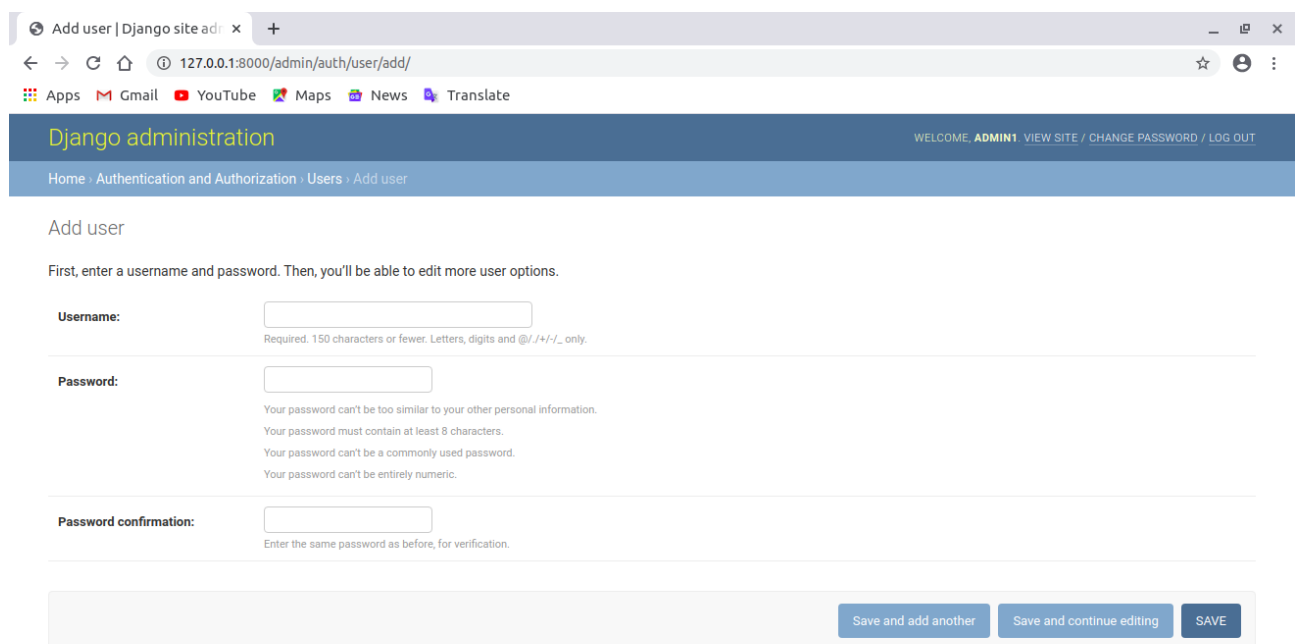


After logging in with the credentials used to create a super user shown in above steps, The browser should look like this,



This is how any user can achieve the status of an ADMIN.  
Also once the user is an admin he/she can add users using this UI provided freely by Django.  
For further knowledge of how Django has this feature and other features follow this link below,  
<https://docs.djangoproject.com/en/3.1/>

Users can be deleted, added and their information could be updated, it lloks like this,



In the above pic the ADMIN can be seen creating a user.

Once a user is created it looks like this,

Select user to change

ADD USER +

Q  Search

Action: ----- Go 0 of 4 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	admin	admin@email.com			✓
<input type="checkbox"/>	admin1	admin1@email.com			✓
<input type="checkbox"/>	adminSmartAgri	djangoblog2020@gmail.com			✓
<input type="checkbox"/>	test2020	test2020@email.com			✗

4 users

**FILTER**

By staff status

All  
Yes  
No

By superuser status

All  
Yes  
No

By active

All  
Yes  
No

If a user info is to be deleted,

Select user to change

ADD USER +

Q  Search

Action: Delete selected users Go 1 of 4 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input checked="" type="checkbox"/>	admin	admin@email.com			✓
<input type="checkbox"/>	admin1	admin1@email.com			✓
<input type="checkbox"/>	adminSmartAgri	djangoblog2020@gmail.com			✓
<input type="checkbox"/>	test2020	test2020@email.com			✗

4 users

**FILTER**

By staff status

All  
Yes  
No

By superuser status

All  
Yes  
No

By active

All  
Yes  
No

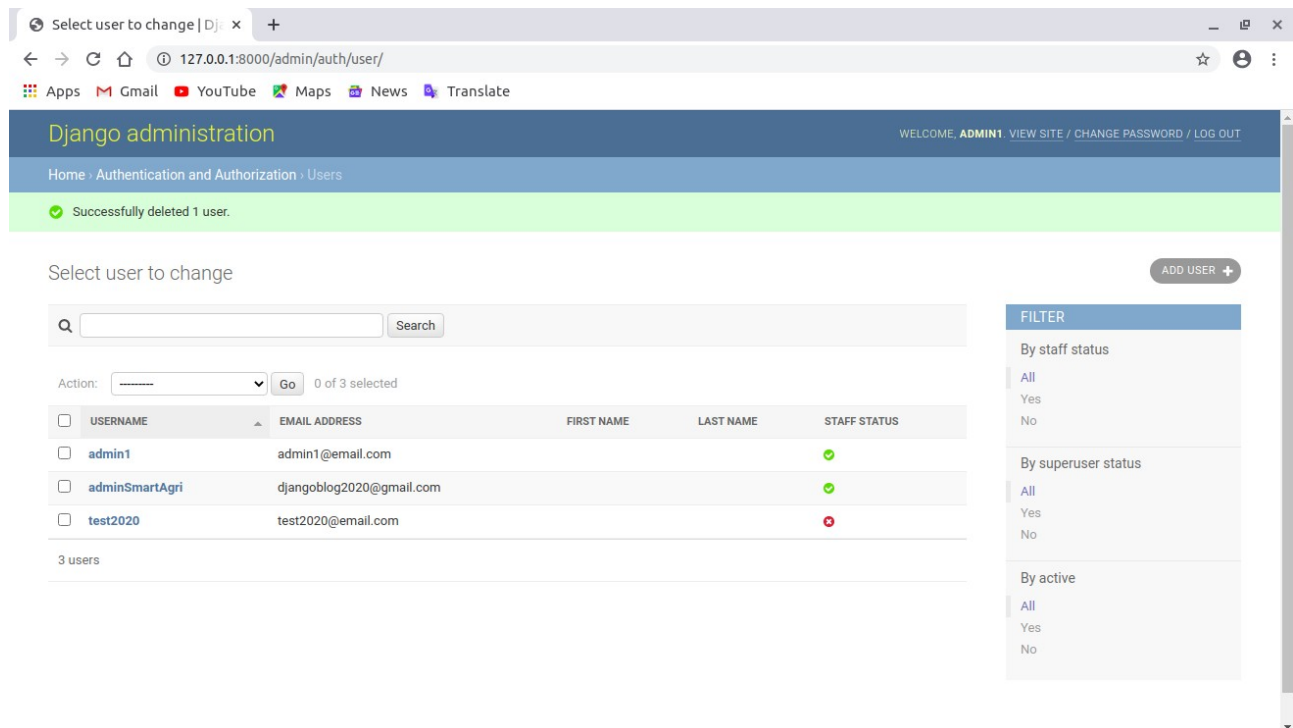
Select the user as shown above and click the **Go** to delete.

It will prompt for conformation as if the ADMIN is not sure to delete then he/she can click, **No take me back** button and if he/she is sure to delete user then the **Yes, I'm Sure** button is to be clicked.

Yes, I'm sure

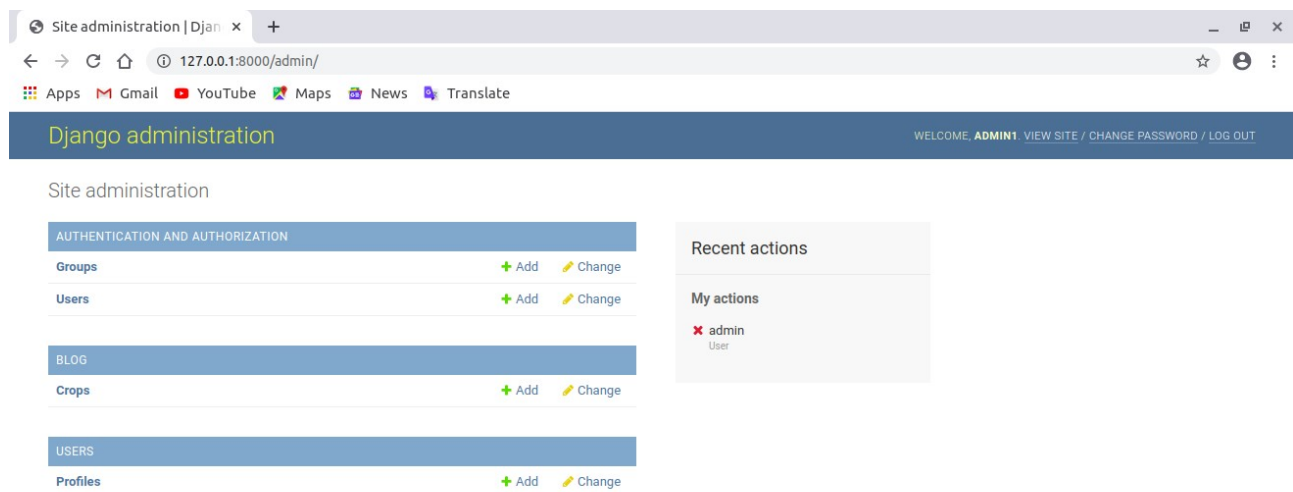
No, take me back

Once user is deleted it will look like this,



With respect to this project, the '**models.py**' file in the '**blog**' folder is set create data in this manner,

Click the **Crops** in the **BLOG** as shown in the below pic,



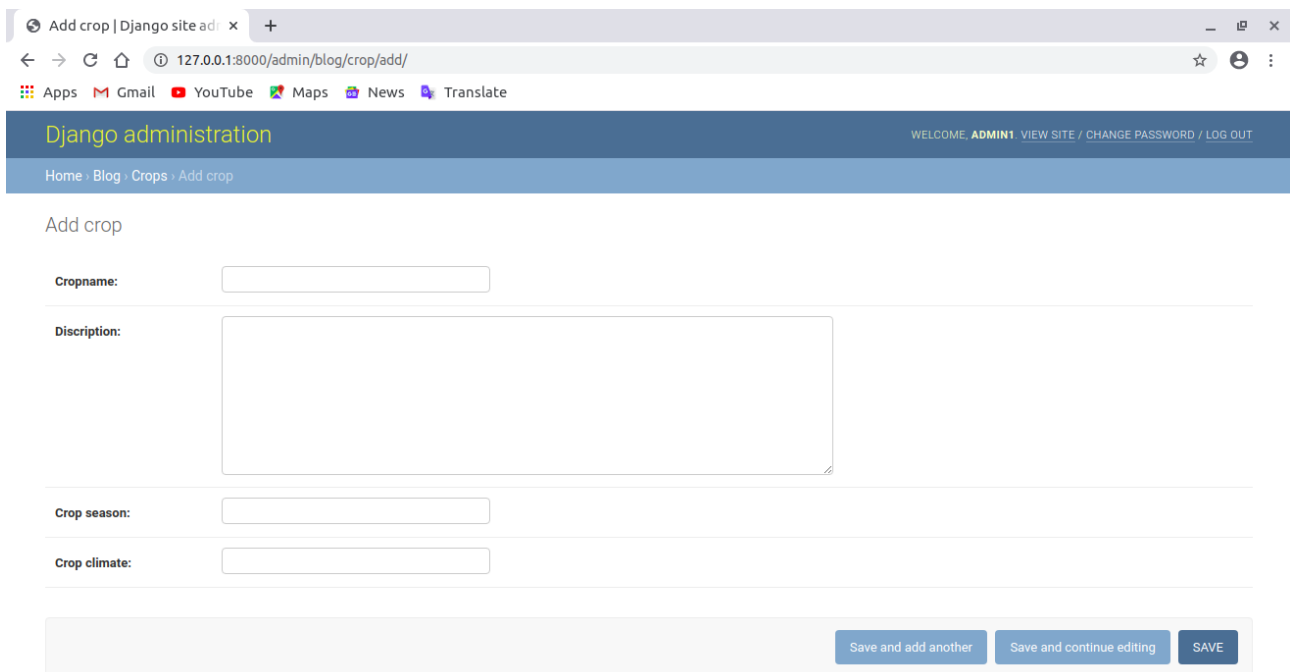
After the previous step the below will be seen,



The screenshot shows the Django administration interface. The top header is dark blue with 'Django administration' on the left and 'WELCOME, ADMIN1 VIEW SITE / CHANGE PASSWORD / LOG OUT' on the right. Below the header is a light blue breadcrumb trail: 'Home > Blog > Crops'. The main content area has a title 'Select crop to change' and an 'ADD CROP +' button in the top right corner. Below the title is a list showing '0 crops'.

Click the **ADD CROP +** at the right corner to add data,

This is how it will look to add data,



The screenshot shows the 'Add crop' form in the Django administration interface. The browser address bar shows '127.0.0.1:8000/admin/blog/crop/add/'. The page title is 'Add crop | Django site admin'. The breadcrumb trail is 'Home > Blog > Crops > Add crop'. The form has the following fields: 'Cropname:' (text input), 'Discription:' (text area), 'Crop season:' (text input), and 'Crop climate:' (text input). At the bottom right, there are three buttons: 'Save and add another', 'Save and continue editing', and 'SAVE'.

The above steps show the actions of an ADMIN and how he/she can manage data, including all operations such as Authorization, Authentication and update, delete and create user data as well as the web site data.

All this is achievable due to DJANGO, as it is open source, user friendly and easy to program.