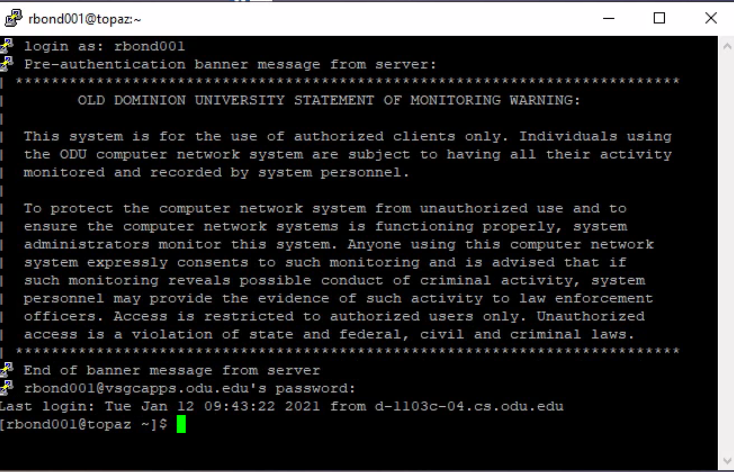
VSGCAPPS Website Details(vsgcapps.odu.edu)

# Access

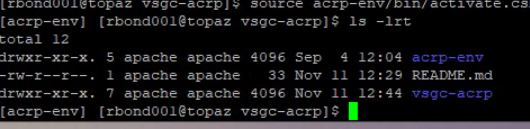
* To connect to server use putty to ssh into the machine by giving host name as vsgcapps.odu.edu
* The machine [topaz.prod.odu.edu](http://topaz.prod.odu.edu/) is available for you. Try to login in with ODU credentials.



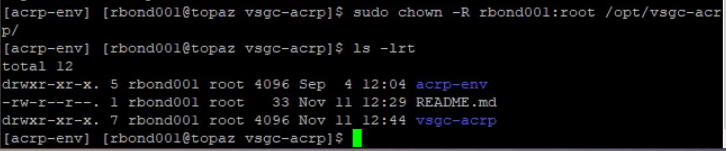
* Once you are successfully logged in you will see something like this.
* Follow the steps below to go the root directory of the project.

# Project files

1. cd /
2. cd opt
3. cd vsgc-acrp
4. Activate virtual environment to make any changes in the project ( source acrp-env/bin/activate.csh)
5. cd vsgc-acrp
6. If you are having any permissions issue give ( ls -lrt ) and check the permissions.

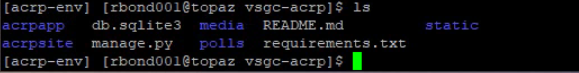


* If it is in apache change the ownership by giving ( sudo chown -R rbond001:root /opt/vsgc-acrp/ )
* Then the permissions should change to root.



* Once you are done with your changes change the permissions back by giving ( sudo chown -R apache:apache /opt/vsgc-acrp/ )

1. Under vsgc-acrp we have all the project files.



1. Acrpsite directory is main project folder to change any settings or urls.

In **settings.py** you have configurations for the Django project.

In **urls.py** you have url declarations for the project.

However you can list all the urls of the project here or you can create a individual files under app folders and can give the path here to get the urls from app url’s ( I used in this way )



1. Media is the file storage. The uploaded files will be saved here.
2. Check this link ( <https://docs.djangoproject.com/en/3.1/howto/static-files/> ) to see how static files work.
3. Here we have two apps( acrpapp , polls ) under one single project (acrpsite) because we can create multiple apps under one project using Django.

For acrpapp :

In the First Project we have two forms

1)<https://vsgcapps.odu.edu/acrpsite/>

2) <http://vsgcapps.odu.edu/application/>

For polls : <https://vsgcapps.odu.edu/graward/>

The main model is same for both apps.

**models.py**: To change any row or column in database or to show how the form looks you have to change in models.py

If you have changed any thing is models.py you have to run the below commands to display.

1. Python manage.py makemigrations
2. Python manage.py migrate

**forms.py:** This is the file I created.Here I import the classes from models.py to exclude and include particular fields and can-do styling to display html page.

**views.py:** Form data sent back to a Django website is processed by a view.

( <https://docs.djangoproject.com/en/3.1/topics/http/views/> ) Check this link to see how functions are written in views.py

**Urls.py**: This is the file I created, form the main project project url I have given this path to include app urls.py. All the urls for that particular app is given here.

* All the html files are in templates.
* Template tags and filters are in template tags folders.
* Json files are in fixtures folder.

1. Once you finished all the changes give the following command in the server to see the changes.

( Sudo systemctl restart httpd24-httpd )

If restart doesn’t work, try enabling It by giving ( Sudo systemctl enable httpd24-httpd ) and restart it.

1. If you are running in local ,no need to use the above command just give python manage.py runserver.

# Main Commands

To check Django version: python -m django –version

If you want to create new project: django-admin startproject mysite

If you want to create new app: python manage.py startapp polls

Below are commands to run after any modifications:

1. Python manage.py makemigrations
2. Python manage.py migrate
3. Python manage.py runserver

Makemigrations and migrate commands are mainly used if there is any modification in models otherwise you can just give Python manage.py runserver.

# Database:

* You can see the database details in settings.py under acrpsite folder.
* There are two databases in settings file one is for local and one is for server.
* For local the host name is given as “root”.
* If you are working in server, comment the local database details.
* And If you are working in local, comment the server database details.
* Below pictures shows the tables of entire project.



Acrpapp\_applicant: This table has data of applicant form <http://vsgcapps.odu.edu/application/>

Acrpapp\_designapp: This table has data of main form <https://vsgcapps.odu.edu/acrpsite/>

Acrpapp\_emp:This table has titles and questions of faculty form.

Acrpapp\_response:This table has the scores of faculty form.

Acrpapp\_teammember:This table has the data of the applicant teammembers which the user will fill in this form <https://vsgcapps.odu.edu/acrpsite/>

Acrpapp\_user\_profile:This table has the status if faculty form.

Auth\_group:This table shows if we added any group in admin page.

Auth\_user:The number of users we added in admin page.

Polls\_applicant:This table had the data of the second app form.

Polls\_faculty:This table has the emails the user filled in the second app form.

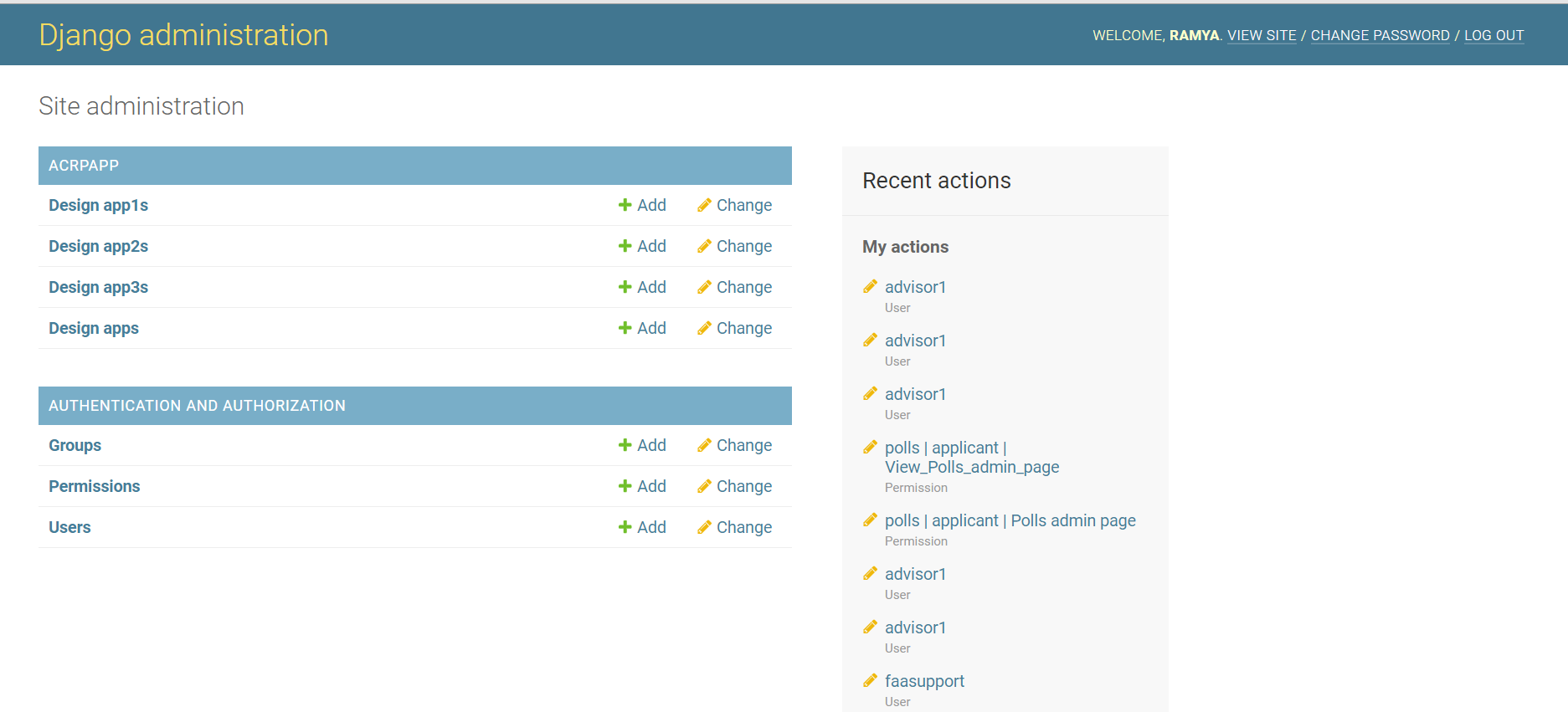
Polls\_recommendation\_fields:This table has the data of recommender which user mentions.

# AdminPage:

For overview of admin page look into this link <https://docs.djangoproject.com/en/3.1/ref/contrib/admin/>

First,you have a run a command python manage.py createsuperuser.Then it will ask for username and password after completing it you have to run the server.

And use <https://vsgcapps.odu.edu/admin/> or <http://127.0.0.1:8000/admin/> when you are using local. and login the credentials you created.



This is how the admin page looks like.

In this project, I have given some permissions in admin.py in acrpapp (1st) project