# Borda Count Final Report

## Team Members and Roles

| | |
|---|---|
| Prof. Korok Ray | Client |
| Amith Gopi | Supervisor (SPOC) |
| Sneha Singh | Team Member (Scrum Master) |
| Shaista Ambreen | Team Member (Product Owner) |
| Sunanda Saha | Team Member |
| Upasana Mishra | Team Member |

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

TEXAS A&M UNIVERSITY, COLLEGE STATION

## Summary

In this project, we are tasked with developing a responsive website, which could be used for retrieving the optimal poll from the users. This website eliminates the need of sending bulk emails to a large list of users to gather their preferences and having to sort through each person to figure out the best preference.

The application implements an aggregation algorithm called Borda Count, which is a method used for aggregating individual preferences into a social preference. This is an area of Economics called Social Choice Theory, which is connected to Game Theory. In this algorithm, the administrator creates an event, which is a choice among K objects (e.g., times for a meeting). The N different users can then rank the K objects, which is a strict ordering, and the website uses the Borda Count method to deliver the top choice, and this result is visible to the administrator.

It is a legacy project wherein the Fall 2021 CSCE 606 class performed the initial implementation of the website using Python Django. They had implemented a single-admin system with a sum-based algorithm to output the group’s final preferences. As part of the expansion, the requirements for our team were to replace the existing Borda Count algorithm to Cumulative Voting Algorithm and to implement the multi-admin functionality to accommodate multiple admins.

The application is developed using the Django framework. The story points of the project are monitored using Pivotal tracker. It is also deployed on Heroku and uploaded on GitHub. The stakeholders for this project are Dr. Korok Ray, Amith Gopi and Dr. Philip Ritchey.

## User Stories Implemented

Below are the user stories and the sub-tasks that we planned and completed successfully.

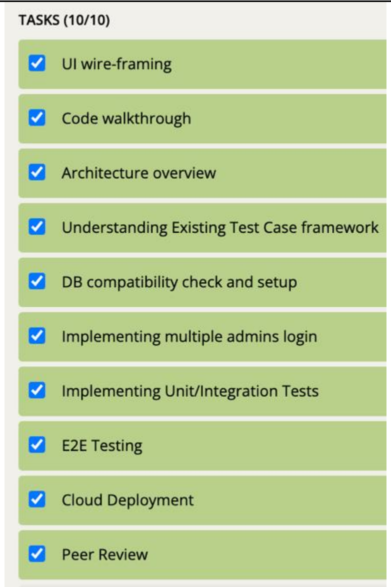
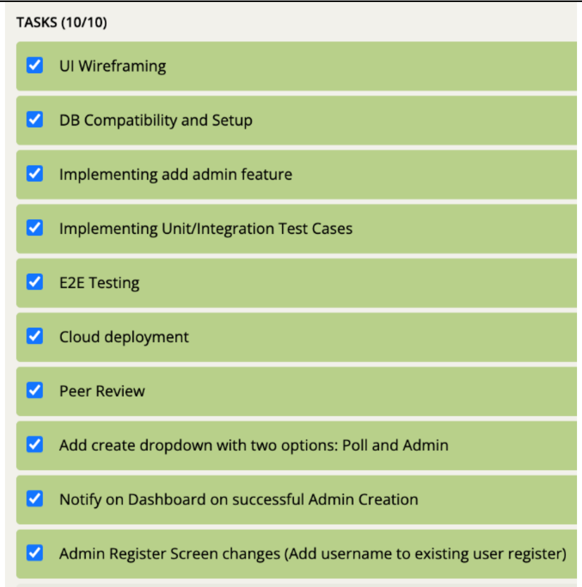
## User stories related to Multi-Admin Functionality:

**Feature: Multi Admin Login (Points - 4)**

- As multiple administrators,
- So that anyone can log into the application and view/create polls on the dashboard
- I want to allow multiple admins to enter personal credentials on the login page

**Feature: Adding New Admin (Points - 4)**

- As an existing administrator,
- So that multiple admins can manage the application
- I want to add a new admin with random credentials which can be reset later.

| **Title** | **Multi Admin Login** | **Adding New Admin** |
|---|---|---|
| **Tasks** |  TASKS (10/10)<br>✓ UI wire-framing<br>✓ Code walkthrough<br>✓ Architecture overview<br>✓ Understanding Existing Test Case framework<br>✓ DB compatibility check and setup<br>✓ Implementing multiple admins login<br>✓ Implementing Unit/Integration Tests<br>✓ E2E Testing<br>✓ Cloud Deployment<br>✓ Peer Review |  TASKS (10/10)<br>✓ UI Wireframing<br>✓ DB Compatibility and Setup<br>✓ Implementing add admin feature<br>✓ Implementing Unit/Integration Test Cases<br>✓ E2E Testing<br>✓ Cloud deployment<br>✓ Peer Review<br>✓ Add create dropdown with two options: Poll and Admin<br>✓ Notify on Dashboard on successful Admin Creation<br>✓ Admin Register Screen changes (Add username to existing user register) |

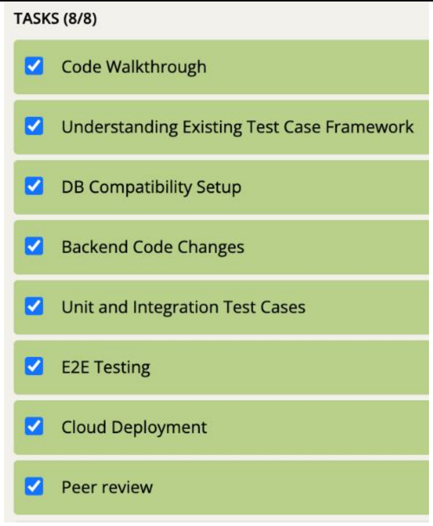
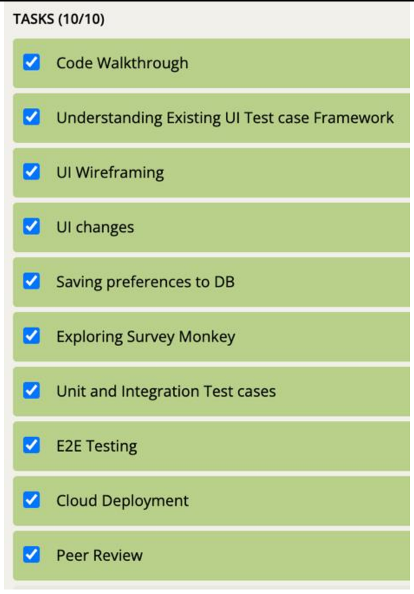
## <u>User stories related to Cumulative Borda Count Algorithm:</u>

**Feature: Cumulative Voting Backend Changes (Points - 4)**

- As an admin,
- So that I can get the highest preference,
- I want to replace the existing algorithm with Cumulative Voting.

**Feature: Cumulative Voting UI Changes (Points - 4)**

- As a user,
- So that I can give a number of points to each alternative,
- I want to replace the ranking with the intensity of preferences.

| **Title** | **Cumulative Voting Backend Changes** | **Cumulative Voting UI Changes** |
|---|---|---|
| **Tasks** |  TASKS (8/8)<br>✓ Code Walkthrough<br>✓ Understanding Existing Test Case Framework<br>✓ DB Compatibility Setup<br>✓ Backend Code Changes<br>✓ Unit and Integration Test Cases<br>✓ E2E Testing<br>✓ Cloud Deployment<br>✓ Peer review |  TASKS (10/10)<br>✓ Code Walkthrough<br>✓ Understanding Existing UI Test case Framework<br>✓ UI Wireframing<br>✓ UI changes<br>✓ Saving preferences to DB<br>✓ Exploring Survey Monkey<br>✓ Unit and Integration Test cases<br>✓ E2E Testing<br>✓ Cloud Deployment<br>✓ Peer Review |

## Iteration wise progress

**Iteration 0: (2 points)**

- Organized the team and developed a meeting schedule
- Scheduled meeting with the client and supervisor to discuss on the user requirements for the project

**Iteration 1: (2 points)**

- Had meeting with the client to acquire better understanding of the project and its requirements.
- Code walkthrough and completed the initial local setup of the project
- Came up with user stories for the application and developed UI wireframes.
- Presented client with the live demo of the wireframe and working prototype of the multi admin functionality.
- Acquainted ourselves with the Django framework.

**Iteration 2: (4 points)**

- Developed both the user interface and backend for implementing multi admin feature.
- Delivered multi admin feature via two different methods mentioned below:
    - The existing admin creates a new admin with username and password. The newly created admin can then login and change their passwords.
    - Second approach is that the new admin registers as a user with their preferred credentials and then an admin gives them admin permissions.
- Created interface for admin to add a new admin and to give admin permissions to a user
- Implemented UI changes to deliver notification after a new admin is added.
- Performed unit and integration testing of the implementation and deployed the changes to the Heroku.
- Demonstrated the actual working multi admin functionality to the client on Heroku.

**Iteration 3: (4 points)**

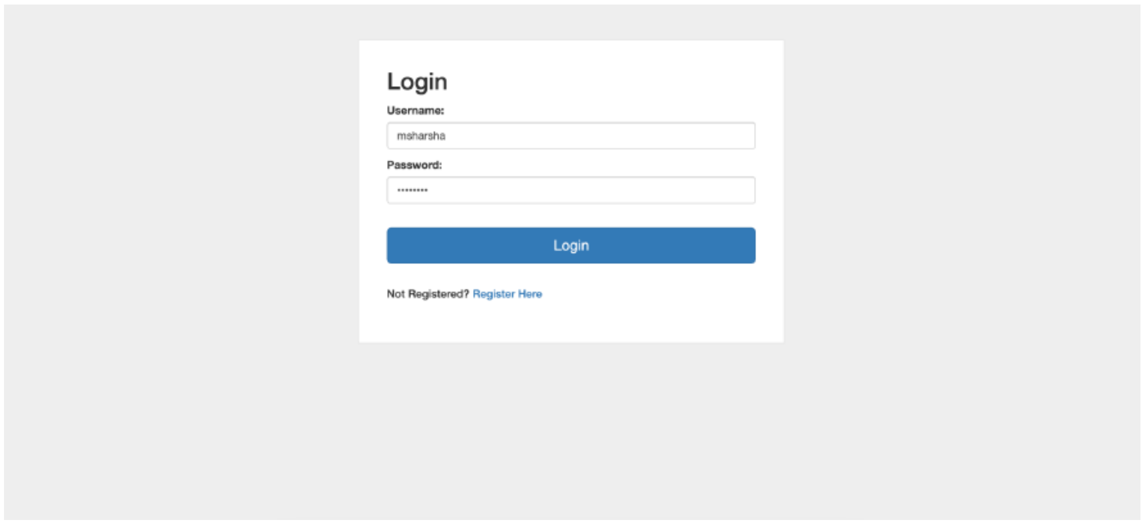
- Discussed the queries with supervisor for implementing the new Cumulative Borda-Count Algorithm.
- Discussed the validations for the new algorithm with the supervisor.
- Scheduled a meeting with previous team to discuss their approach to update the preference collecting mechanism.
- Generated the UI wireframes to collect the preferences and made backend changes to update the aggregator.
- The Database Schema for the new changes is designed.

**Iteration 4: (4 points)**

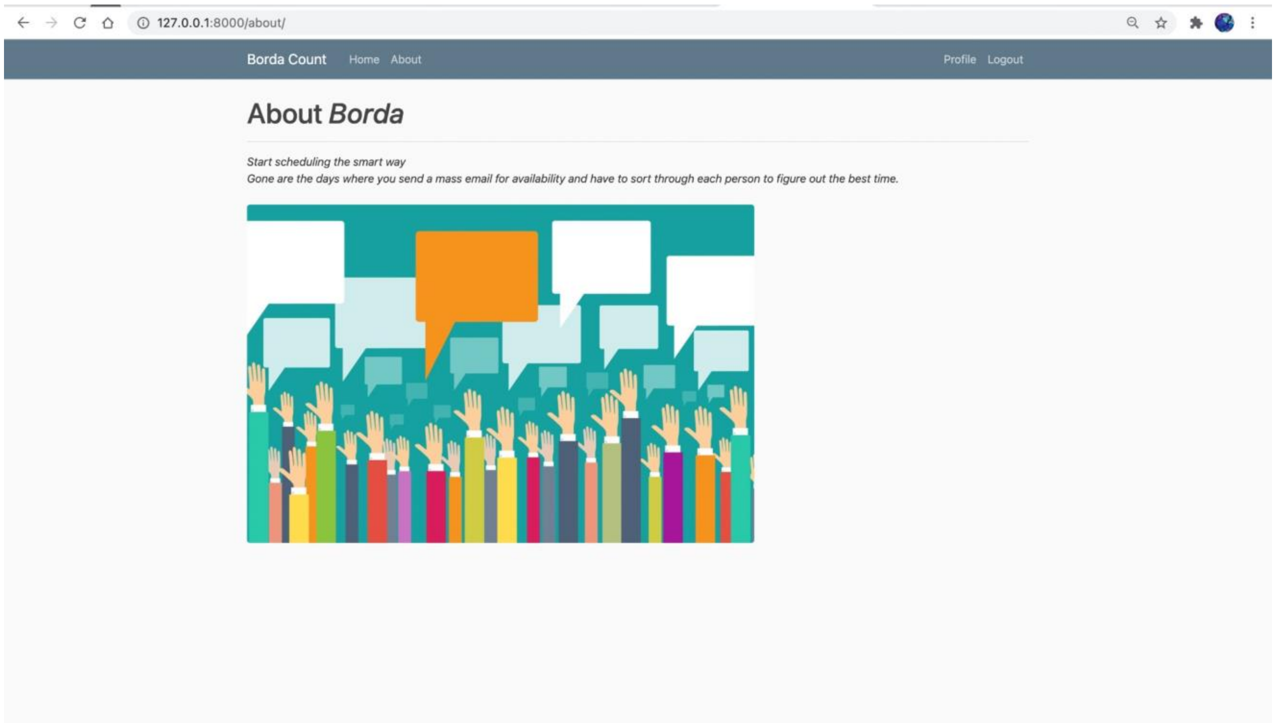
- Implemented the new Cumulative Algorithm with the required backend and UI changes.
- Added the necessary backend validations for the new algorithm.
- Added the UI implementations for the necessary validations implemented in the backend.
- Implemented Text Box Preferences on UI as well as Backend.
- The user can submit their preferences by filling in their choices in the text boxes.
- Performed unit and integration testing and deployed the changes to Heroku.
- Merged all the changes to the parent repository.
- Presented client with the live demo of the working prototype.

## User Interface

### Login Page



### About Page

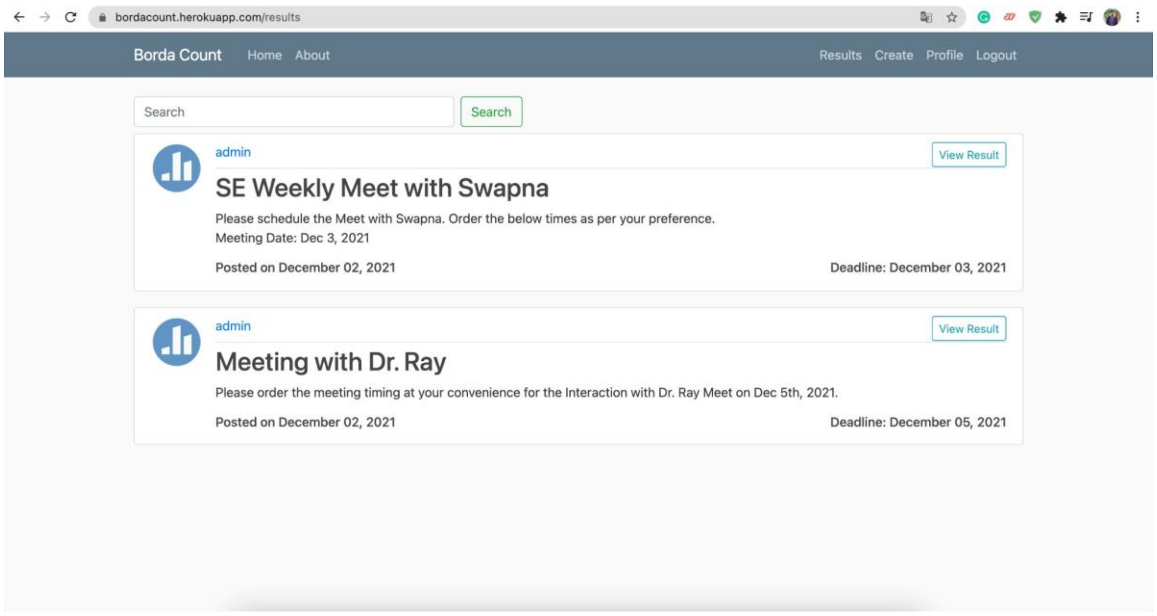


About *Borda*

*Start scheduling the smart way*
*Gone are the days where you send a mass email for availability and have to sort through each person to figure out the best time.*

## Admin flow

Once the login, upon clicking on **results** he can view all the polls that admins created.



bordacount.herokuapp.com/results

Borda Count Home About Results Create Profile Logout

Search Search

admin View Result

SE Weekly Meet with Swapna

Please schedule the Meet with Swapna. Order the below times as per your preference.
Meeting Date: Dec 3, 2021

Posted on December 02, 2021 Deadline: December 03, 2021

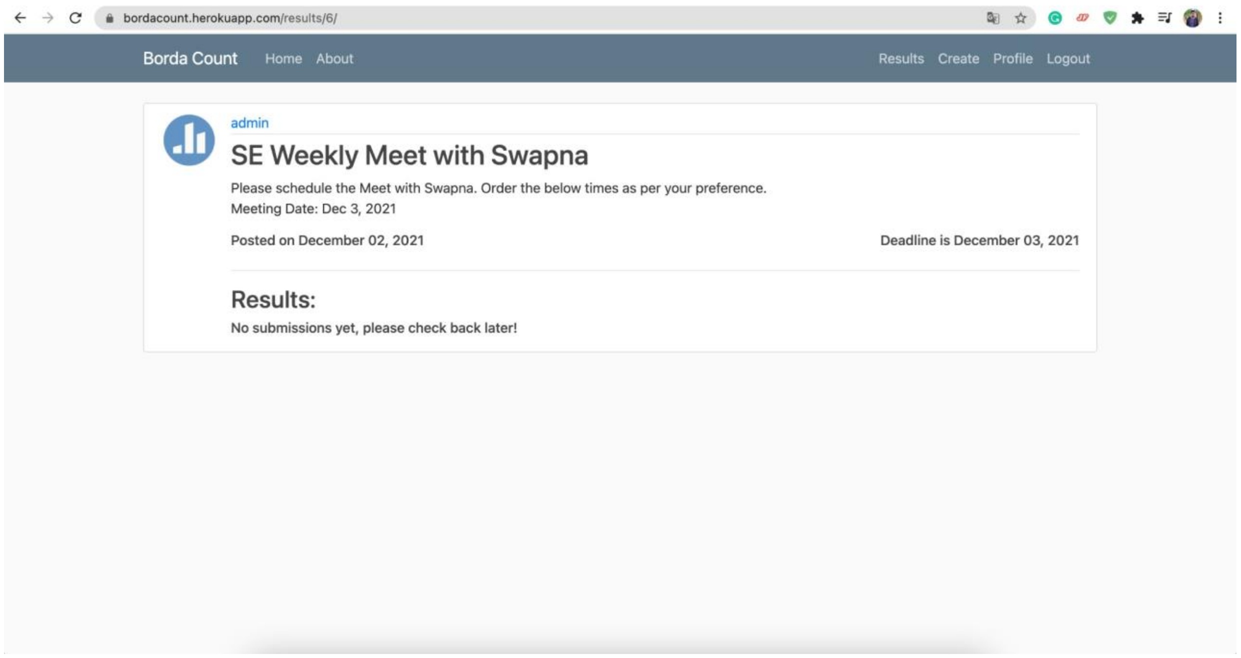
admin View Result

Meeting with Dr. Ray

Please order the meeting timing at your convenience for the Interaction with Dr. Ray Meet on Dec 5th, 2021.

Posted on December 02, 2021 Deadline: December 05, 2021

Upon clicking on the **View Results,** they can view the results for any poll like below. Below case when the admin has created a poll and none of the user have responded.



bordacount.herokuapp.com/results/6/

Borda Count Home About Results Create Profile Logout

admin

SE Weekly Meet with Swapna

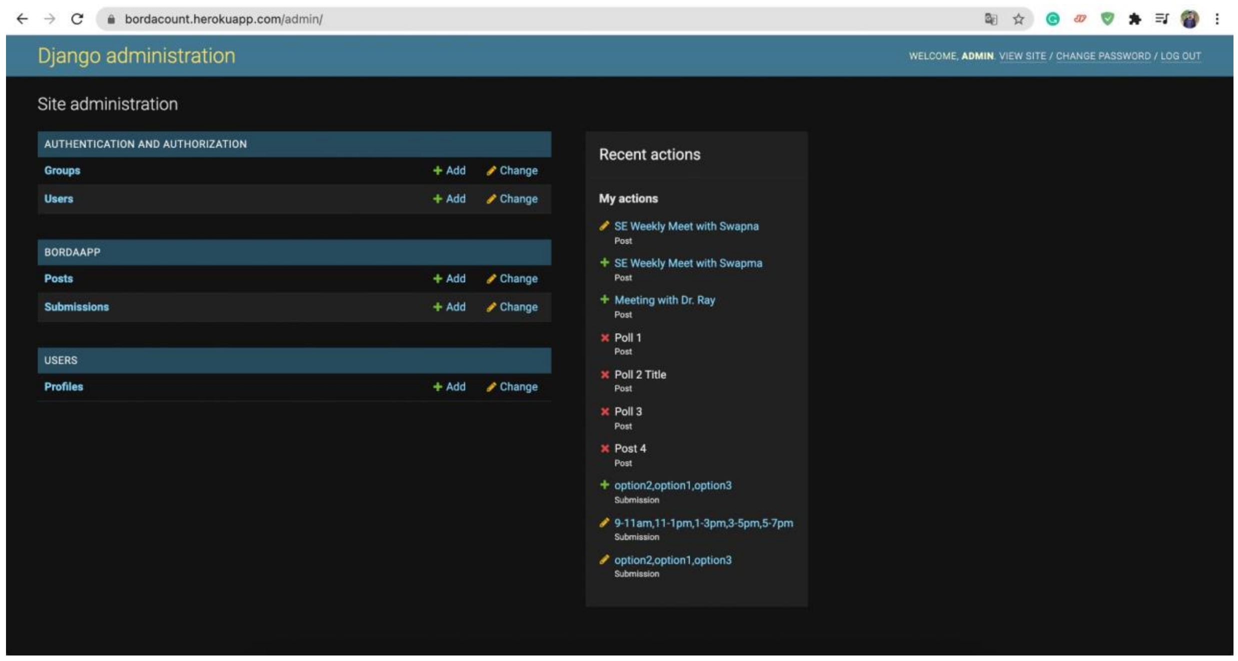
Please schedule the Meet with Swapna. Order the below times as per your preference.
Meeting Date: Dec 3, 2021

Posted on December 02, 2021 Deadline is December 03, 2021

Results:

No submissions yet, please check back later!

Also, they can create their own poll using **Create** button,



bordacount.herokuapp.com/admin/

Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

| AUTHENTICATION AND AUTHORIZATION | | |
|---|---|---|
| Groups | + Add | Change |
| Users | + Add | Change |

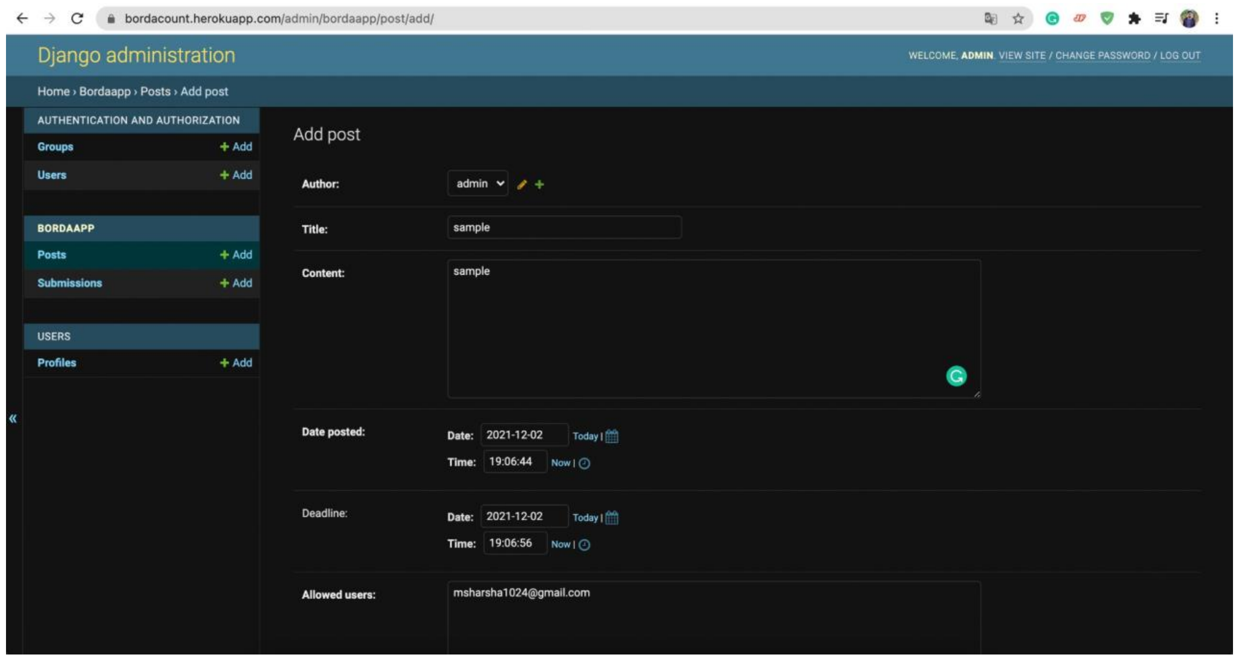
| BORDAAPP | | |
|---|---|---|
| Posts | + Add | Change |
| Submissions | + Add | Change |

| USERS | | |
|---|---|---|
| Profiles | + Add | Change |

Recent actions

My actions

- SE Weekly Meet with Swapna — Post
- SE Weekly Meet with Swapma — Post
- Meeting with Dr. Ray — Post
- Poll 1 — Post
- Poll 2 Title — Post
- Poll 3 — Post
- Post 4 — Post
- option2,option1,option3 — Submission
- 9-11am,11-1pm,1-3pm,3-5pm,5-7pm — Submission
- option2,option1,option3 — Submission



bordacount.herokuapp.com/admin/bordaapp/post/add/

Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Bordaapp › Posts › Add post

| AUTHENTICATION AND AUTHORIZATION | |
|---|---|
| Groups | + Add |
| Users | + Add |
| **BORDAAPP** | |
| Posts | + Add |
| Submissions | + Add |
| **USERS** | |
| Profiles | + Add |

Add post

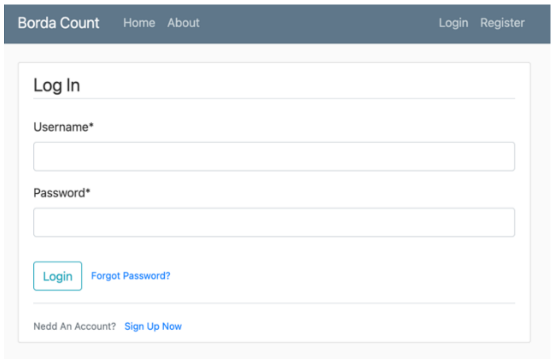
| | |
|---|---|
| Author: | admin |
| Title: | sample |
| Content: | sample |
| Date posted: | Date: 2021-12-02 Today; Time: 19:06:44 Now |
| Deadline: | Date: 2021-12-02 Today; Time: 19:06:56 Now |
| Allowed users: | msharsha1024@gmail.com |

Upon clicking the save button, the poll is published. At this stage, all the allowed users should be able to see the poll on their dashboard.

## Multi-Admin Functionality Flow:

### First Approach (Admin creates another admin):

Admin – Login


Borda Count Home About Login Register

Log In

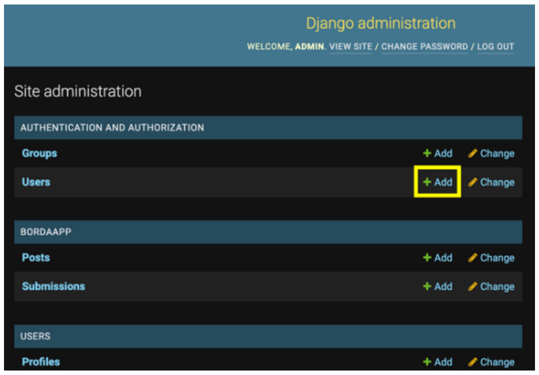
Username*

Password*

Login Forgot Password?

Nedd An Account? Sign Up Now

Click “Add” to add new user


Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

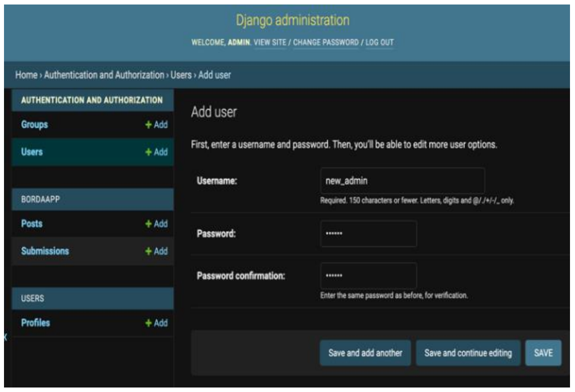
Site administration

| AUTHENTICATION AND AUTHORIZATION | | |
|---|---|---|
| Groups | + Add | Change |
| Users | + Add | Change |

| BORDAAPP | | |
|---|---|---|
| Posts | + Add | Change |
| Submissions | + Add | Change |

| USERS | | |
|---|---|---|
| Profiles | + Add | Change |

Admin gives credentials for new user


Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users › Add user

Add user

First, enter a username and password. Then, you'll be able to edit more user options.

Username: new_admin

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

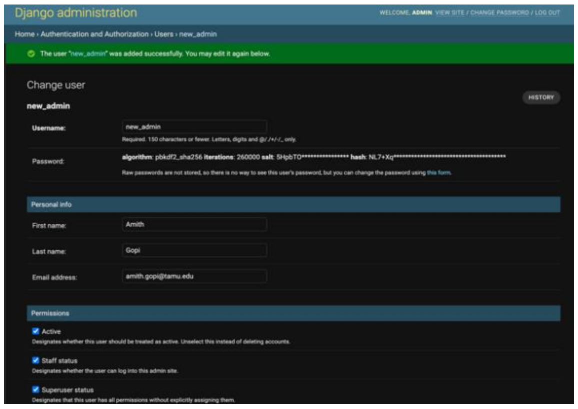
Password: ••••••

Password confirmation: ••••••

Enter the same password as before, for verification.

Save and add another | Save and continue editing | SAVE

User Added


Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users › new_admin

The user “new_admin” was added successfully. You may edit it again below.

Change user

new_admin

HISTORY

Username: new_admin

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password: **algorithm**: pbkdf2_sha256 **iterations**: 260000 **salt**: SHpbTO**************** **hash**: NL7+Xq****************************************

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using this form.

Personal info

First name: Amith

Last name: Gopi

Email address: amith.gopi@tamu.edu

Permissions

[✓] Active

Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

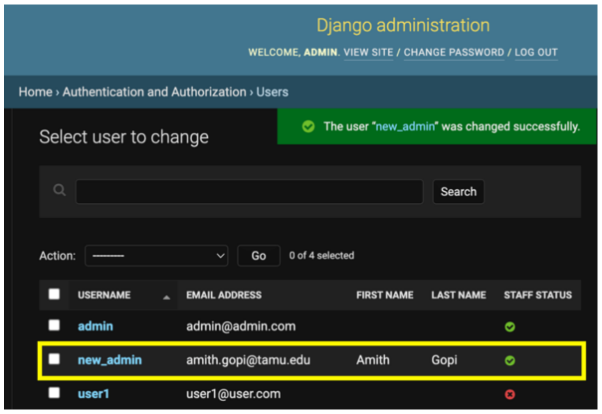
[✓] Staff status

Designates whether the user can log into this admin site.

[✓] Superuser status

Designates that this user has all permissions without explicitly assigning them.

Added user appears on list


Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users

The user “new_admin” was changed successfully.

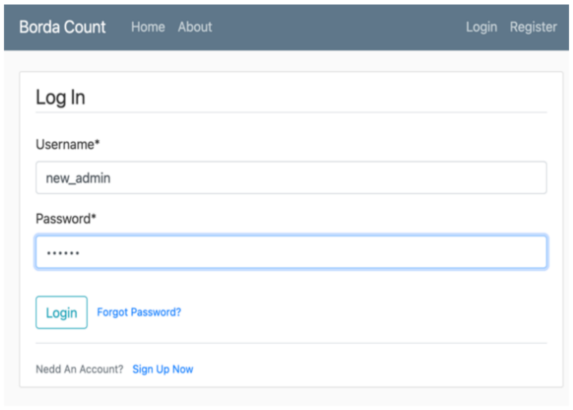
Select user to change

Search

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

| | USERNAME | EMAIL ADDRESS | FIRST NAME | LAST NAME | STAFF STATUS |
|---|---|---|---|---|---|
| ☐ | admin | admin@admin.com | | | ✓ |
| ☐ | new_admin | amith.gopi@tamu.edu | Amith | Gopi | ✓ |
| ☐ | user1 | user1@user.com | | | ✗ |

User logs in using added credentials


Borda Count Home About Login Register

Log In

Username*

new_admin

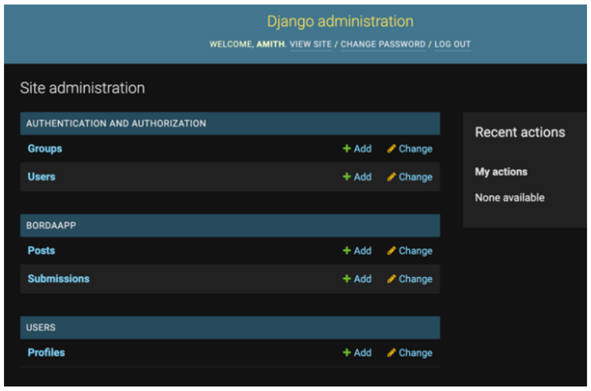
Password*

••••••

Login Forgot Password?

Nedd An Account? Sign Up Now

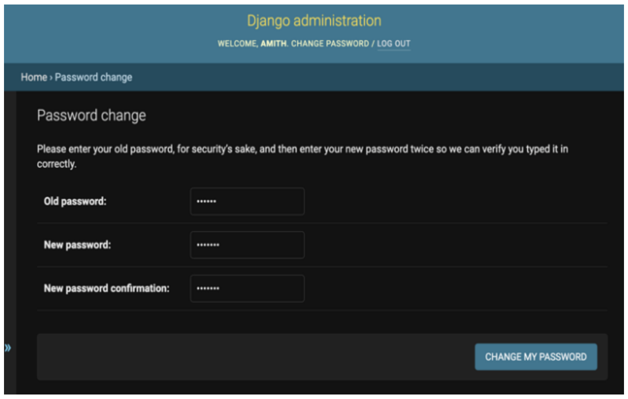
User homepage



Site administration

| | | |
|---|---|---|
| AUTHENTICATION AND AUTHORIZATION | | |
| Groups | + Add | Change |
| Users | + Add | Change |
| BORDAAPP | | |
| Posts | + Add | Change |
| Submissions | + Add | Change |
| USERS | | |
| Profiles | + Add | Change |

User changes the password



Password change

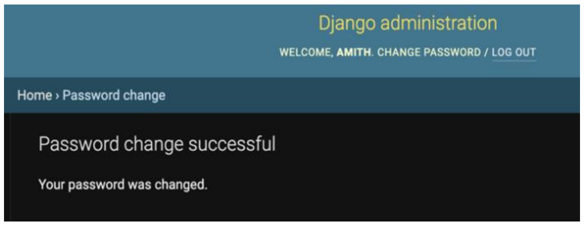
Please enter your old password, for security's sake, and then enter your new password twice so we can verify you typed it in correctly.

Old password: ••••••

New password: •••••••

New password confirmation: •••••••

Password changed succesfully

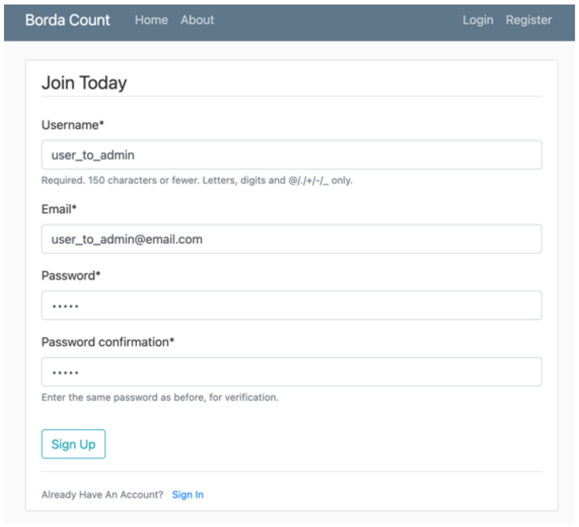


Password change successful

Your password was changed.

**Second Approach (Admin grants existing user admin permissions):**

New User SignUp



Borda Count Home About Login Register

Join Today

Username*

user_to_admin

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

user_to_admin@email.com

Password*

•••••

Password confirmation*

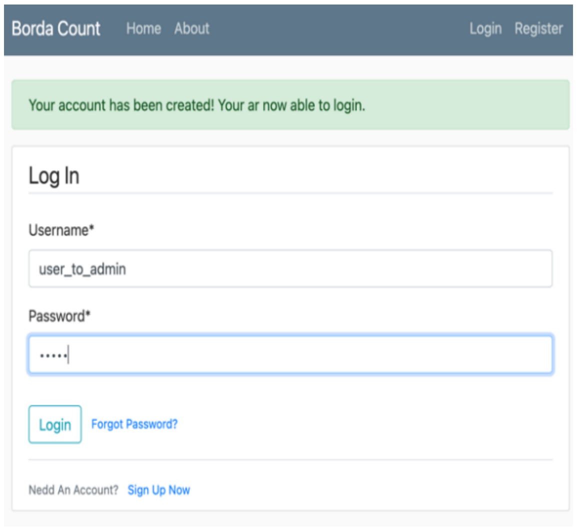
•••••

Enter the same password as before, for verification.

Sign Up

Already Have An Account? Sign In

User Login



Borda Count Home About Login Register

Your account has been created! Your ar now able to login.

Log In

Username*

user_to_admin

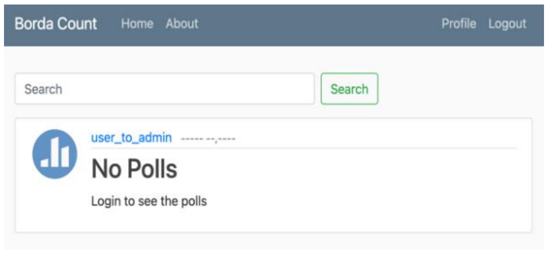
Password*

•••••

Login Forgot Password?

Nedd An Account? Sign Up Now

Homepage



Borda Count Home About Profile Logout

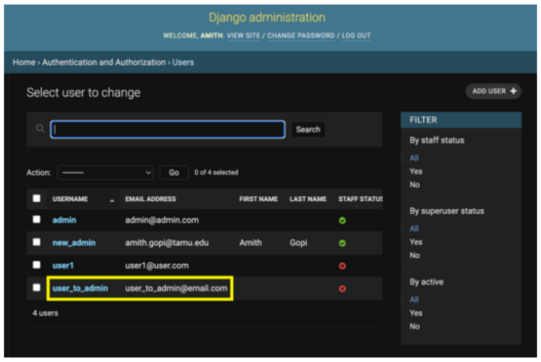
Search Search

user_to_admin ----- --,----

No Polls

Login to see the polls

User present in the list of users



Django administration

WELCOME, AMITH. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users

Select user to change

ADD USER +

Search

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

| USERNAME | EMAIL ADDRESS | FIRST NAME | LAST NAME | STAFF STATUS |
|---|---|---|---|---|
| admin | admin@admin.com | | | ✓ |
| new_admin | amith.gopi@tamu.edu | Amith | Gopi | ✓ |
| user1 | user1@user.com | | | ✗ |
| user_to_admin | user_to_admin@email.com | | | ✗ |

4 users

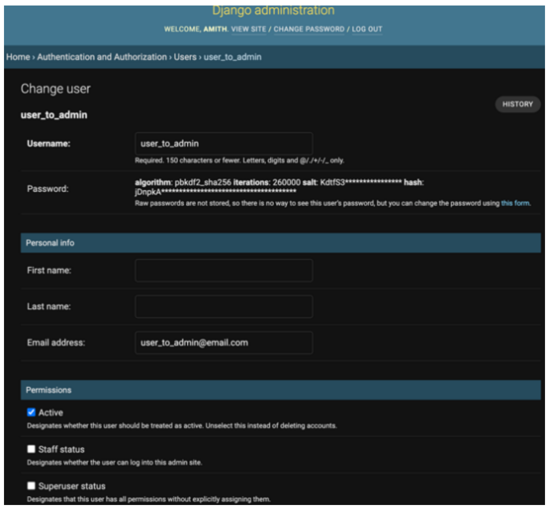
FILTER

By staff status: All, Yes, No

By superuser status: All, Yes, No

By active: All, Yes, No

Change user info page



Django administration

WELCOME, AMITH. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users › user_to_admin

Change user

user_to_admin

HISTORY

Username: user_to_admin

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password: **algorithm**: pbkdf2_sha256 **iterations**: 260000 **salt**: KdtfS3**************** **hash**: jDnpkA****************************************

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using this form.

Personal info

First name:

Last name:

Email address: user_to_admin@email.com

Permissions

☑ Active

Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

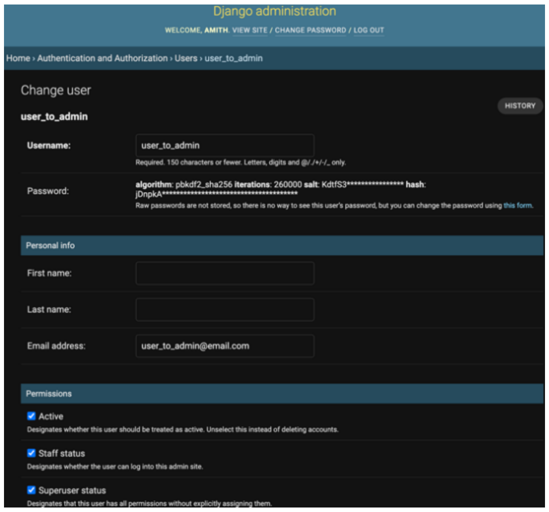
☐ Staff status

Designates whether the user can log into this admin site.

☐ Superuser status

Designates that this user has all permissions without explicitly assigning them.

Add the user as admin



Django administration

WELCOME, AMITH. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users › user_to_admin

Change user

user_to_admin

HISTORY

Username: user_to_admin

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password: **algorithm**: pbkdf2_sha256 **iterations**: 260000 **salt**: KdtfS3**************** **hash**: jDnpkA****************************************

Raw passwords are not stored, so there is no way to see this user's password, but you can change the password using this form.

Personal info

First name:

Last name:

Email address: user_to_admin@email.com

Permissions

☑ Active

Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

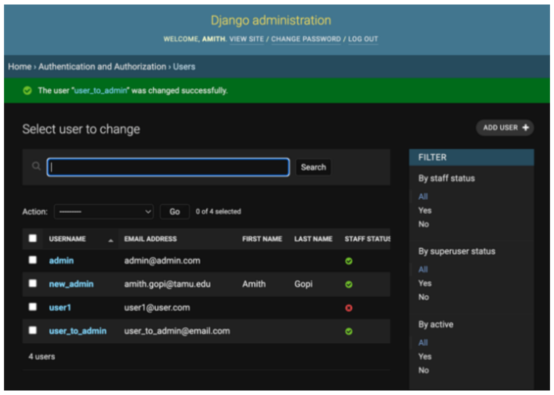
☑ Staff status

Designates whether the user can log into this admin site.

☑ Superuser status

Designates that this user has all permissions without explicitly assigning them.

User changed to admin successfully



Home › Authentication and Authorization › Users

The user "user_to_admin" was changed successfully.

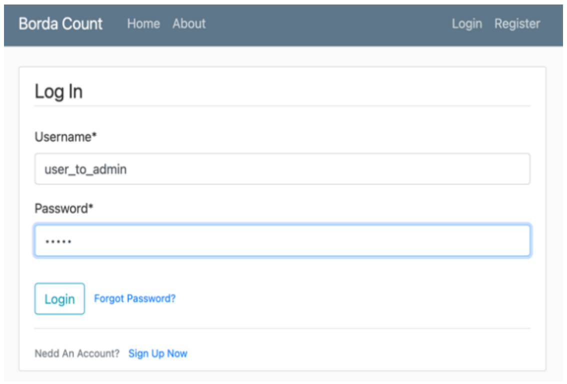
Select user to change

Action: 0 of 4 selected

| USERNAME | EMAIL ADDRESS | FIRST NAME | LAST NAME | STAFF STATUS |
|---|---|---|---|---|
| admin | admin@admin.com | | | ✓ |
| new_admin | amith.gopi@tamu.edu | Amith | Gopi | ✓ |
| user1 | user1@user.com | | | ✗ |
| user_to_admin | user_to_admin@email.com | | | ✓ |

4 users

User can now log in as admin

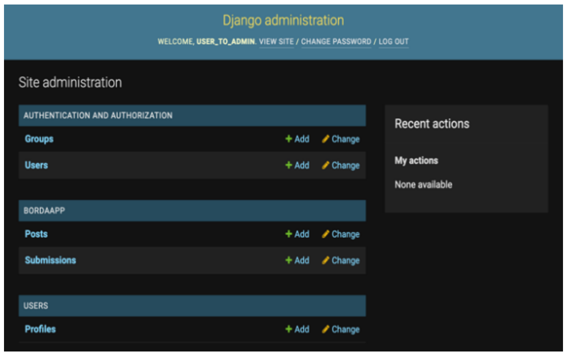


Log In

Username*
user_to_admin

Password*
•••••

Admin Homepage



Site administration

AUTHENTICATION AND AUTHORIZATION
Groups
Users

BORDAAPP
Posts
Submissions

USERS
Profiles

Recent actions

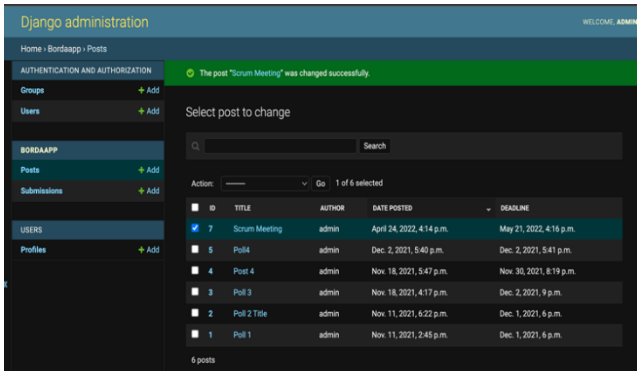
My actions

None available

Django administration
WELCOME, USER_TO_ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Add
Change

## Cumulative Borda Count Implementation Flow:

Poll is created by admin



Django administration

WELCOME, ADMIN

Home › Bordaapp › Posts

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

BORDAAPP

Posts + Add

Submissions + Add

USERS

Profiles + Add

The post "Scrum Meeting" was changed successfully.

Select post to change

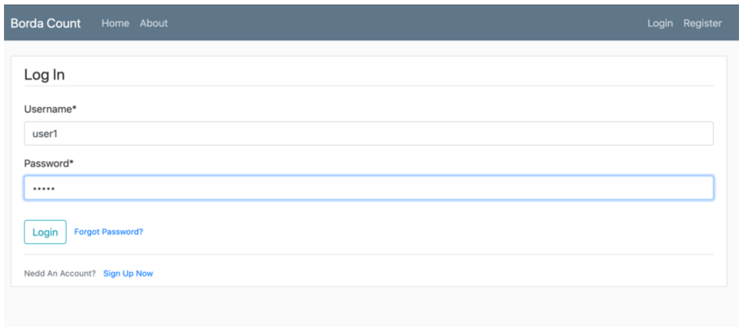
Search

Action: ——— Go 1 of 6 selected

| | ID | TITLE | AUTHOR | DATE POSTED | DEADLINE |
|---|---|---|---|---|---|
| ✓ | 7 | Scrum Meeting | admin | April 24, 2022, 4:14 p.m. | May 21, 2022, 4:16 p.m. |
| | 5 | Poll4 | admin | Dec. 2, 2021, 5:40 p.m. | Dec. 2, 2021, 5:41 p.m. |
| | 4 | Post 4 | admin | Nov. 18, 2021, 5:47 p.m. | Nov. 30, 2021, 8:19 p.m. |
| | 3 | Poll 3 | admin | Nov. 18, 2021, 4:17 p.m. | Dec. 2, 2021, 9 p.m. |
| | 2 | Poll 2 Title | admin | Nov. 11, 2021, 6:22 p.m. | Dec. 1, 2021, 6 p.m. |
| | 1 | Poll 1 | admin | Nov. 11, 2021, 2:45 p.m. | Dec. 1, 2021, 6 p.m. |

6 posts

User logs in to take the poll



Borda Count Home About Login Register

Log In

Username*

user1

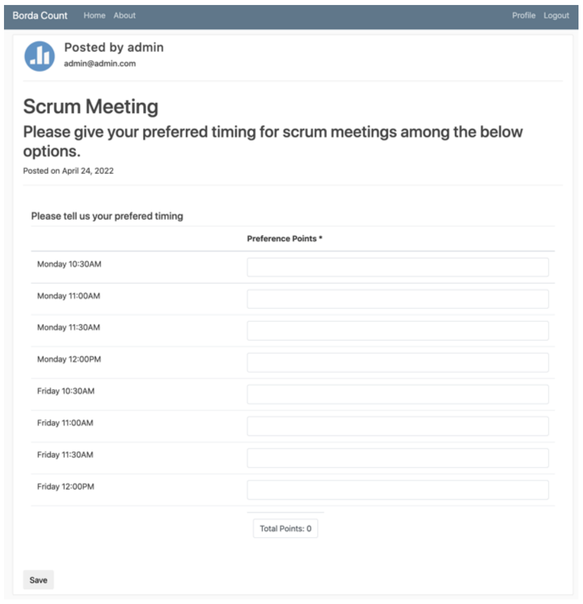
Password*

•••••

Login Forgot Password?

Nedd An Account? Sign Up Now

Poll as it appears to the user



Borda Count Home About Profile Logout

Posted by admin
admin@admin.com

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

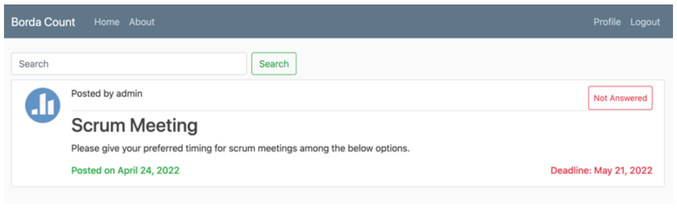
Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | |
| Monday 11:00AM | |
| Monday 11:30AM | |
| Monday 12:00PM | |
| Friday 10:30AM | |
| Friday 11:00AM | |
| Friday 11:30AM | |
| Friday 12:00PM | |

Total Points: 0

Save

Poll as visible on the user dashboard



Borda Count Home About Profile Logout

Search Search

Posted by admin

Not Answered

Scrum Meeting

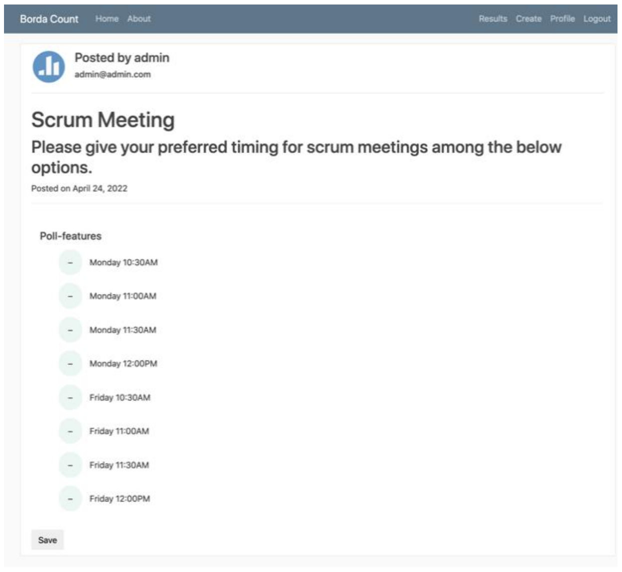
Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

Deadline: May 21, 2022

## Comparison between old and new UI:

Previous screen with old Borda Count Algorithm



Borda Count Home About Results Create Profile Logout

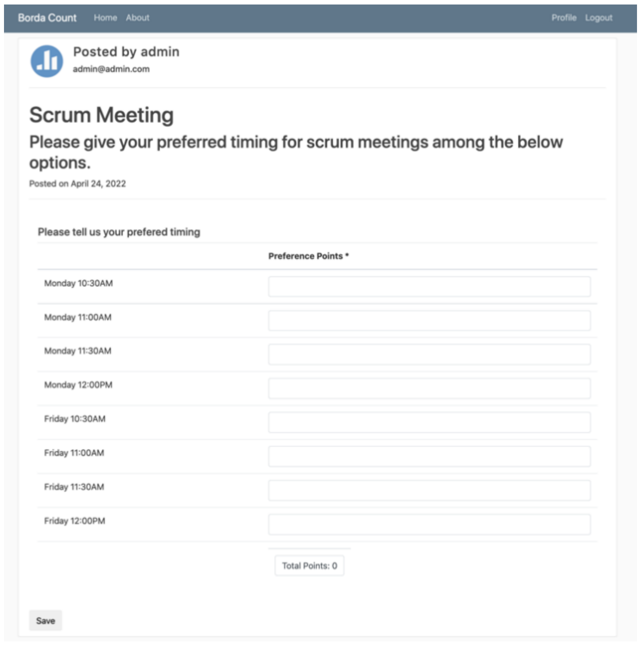
Posted by admin
admin@admin.com

Scrum Meeting
Please give your preferred timing for scrum meetings among the below options.
Posted on April 24, 2022

Poll-features
- Monday 10:30AM
- Monday 11:00AM
- Monday 11:30AM
- Monday 12:00PM
- Friday 10:30AM
- Friday 11:00AM
- Friday 11:30AM
- Friday 12:00PM

Save

New Screen with Cumulative Borda Count Algorithm



Borda Count Home About Profile Logout

Posted by admin
admin@admin.com

Scrum Meeting
Please give your preferred timing for scrum meetings among the below options.
Posted on April 24, 2022

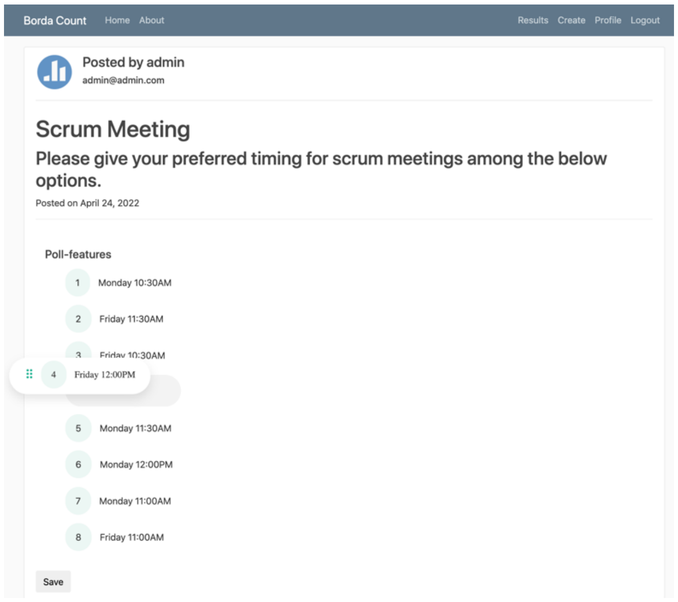
Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | |
| Monday 11:00AM | |
| Monday 11:30AM | |
| Monday 12:00PM | |
| Friday 10:30AM | |
| Friday 11:00AM | |
| Friday 11:30AM | |
| Friday 12:00PM | |

Total Points: 0

Save

Previously user needed to drag and drop preferences



Borda Count Home About Results Create Profile Logout

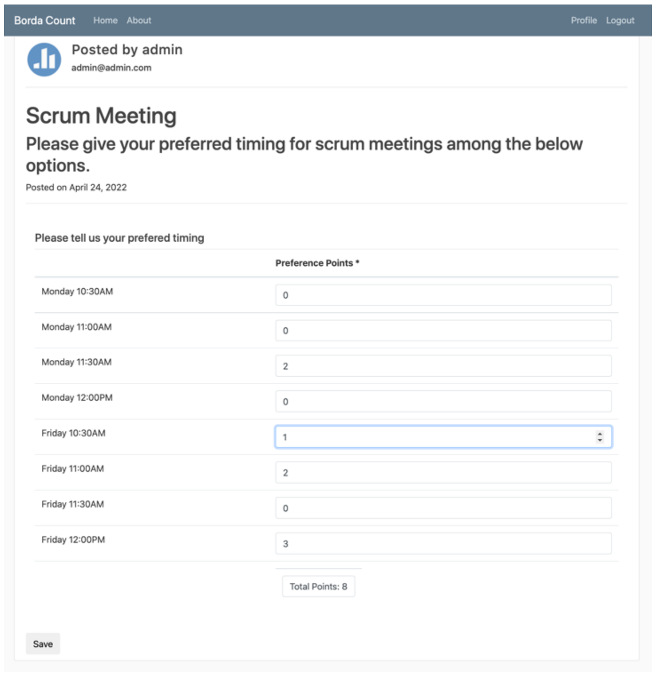
Posted by admin
admin@admin.com

Scrum Meeting
Please give your preferred timing for scrum meetings among the below options.
Posted on April 24, 2022

Poll-features
1 Monday 10:30AM
2 Friday 11:30AM
3 Friday 10:30AM
4 Friday 12:00PM
5 Monday 11:30AM
6 Monday 12:00PM
7 Monday 11:00AM
8 Friday 11:00AM

Save

Currently user needs to allocate points to the preferences



Borda Count Home About Profile Logout

Posted by admin
admin@admin.com

Scrum Meeting
Please give your preferred timing for scrum meetings among the below options.
Posted on April 24, 2022

Please tell us your prefered timing

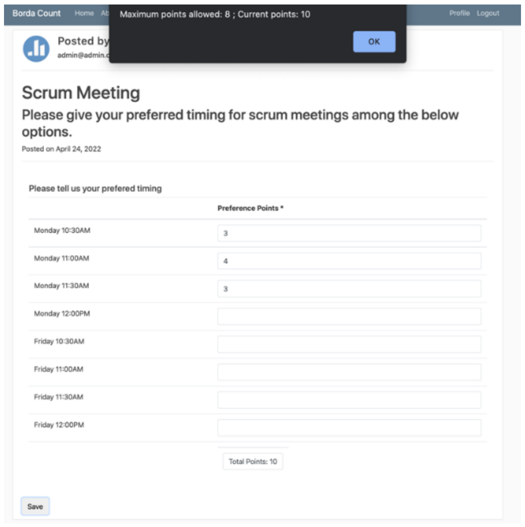
| | Preference Points * |
|---|---|
| Monday 10:30AM | 0 |
| Monday 11:00AM | 0 |
| Monday 11:30AM | 2 |
| Monday 12:00PM | 0 |
| Friday 10:30AM | 1 |
| Friday 11:00AM | 2 |
| Friday 11:30AM | 0 |
| Friday 12:00PM | 3 |

Total Points: 8

Save

## Required validations:

When total number of points exceeds the total points allocated to the user (Total points is equal to the number of alternatives)



Maximum points allowed: 8 ; Current points: 10

Posted by admin@admin.c

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

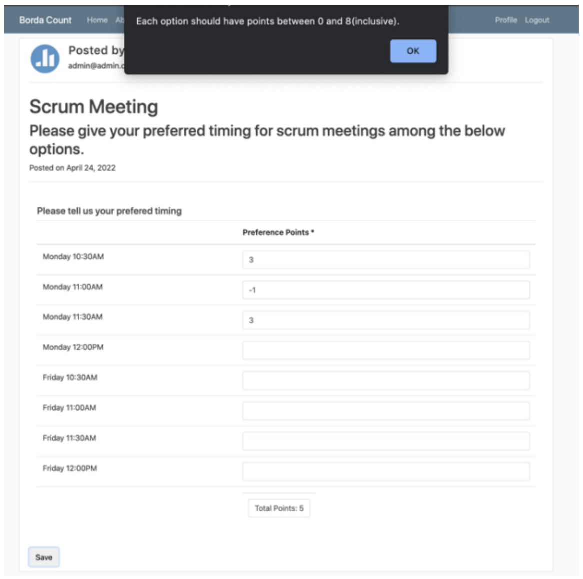
Posted on April 24, 2022

Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | 3 |
| Monday 11:00AM | 4 |
| Monday 11:30AM | 3 |
| Monday 12:00PM | |
| Friday 10:30AM | |
| Friday 11:00AM | |
| Friday 11:30AM | |
| Friday 12:00PM | |

Total Points: 10

When negative points is allocated to a preference



Each option should have points between 0 and 8(inclusive).

Posted by admin@admin.c

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

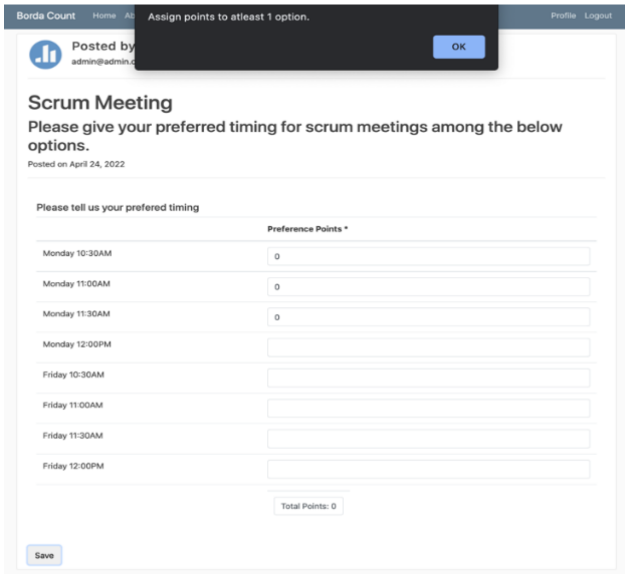
Posted on April 24, 2022

Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | 3 |
| Monday 11:00AM | -1 |
| Monday 11:30AM | 3 |
| Monday 12:00PM | |
| Friday 10:30AM | |
| Friday 11:00AM | |
| Friday 11:30AM | |
| Friday 12:00PM | |

Total Points: 5

When no points are assigned to any option



Assign points to atleast 1 option.

Posted by admin@admin.c

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

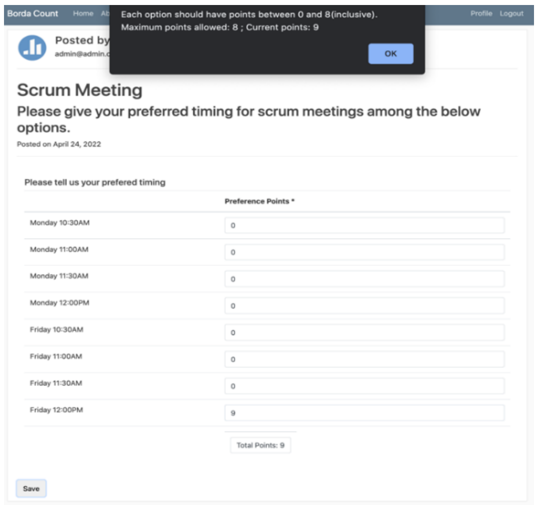
Posted on April 24, 2022

Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | 0 |
| Monday 11:00AM | 0 |
| Monday 11:30AM | 0 |
| Monday 12:00PM | |
| Friday 10:30AM | |
| Friday 11:00AM | |
| Friday 11:30AM | |
| Friday 12:00PM | |

Total Points: 0

When points allocated to an option exceeds the total points allocated to the user



Each option should have points between 0 and 8(inclusive).
Maximum points allowed: 8 ; Current points: 9

Posted by admin@admin.c

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

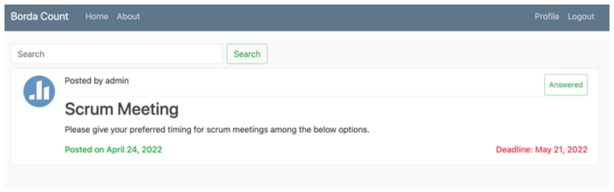
Please tell us your prefered timing

| | Preference Points * |
|---|---|
| Monday 10:30AM | 0 |
| Monday 11:00AM | 0 |
| Monday 11:30AM | 0 |
| Monday 12:00PM | 0 |
| Friday 10:30AM | 0 |
| Friday 11:00AM | 0 |
| Friday 11:30AM | 0 |
| Friday 12:00PM | 9 |

Total Points: 9

**Results:**

After the user has completed the poll



Posted by admin

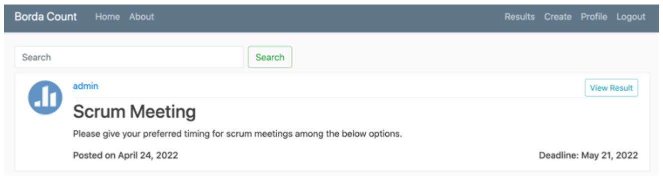
Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

Deadline: May 21, 2022

The admin can view the results



admin

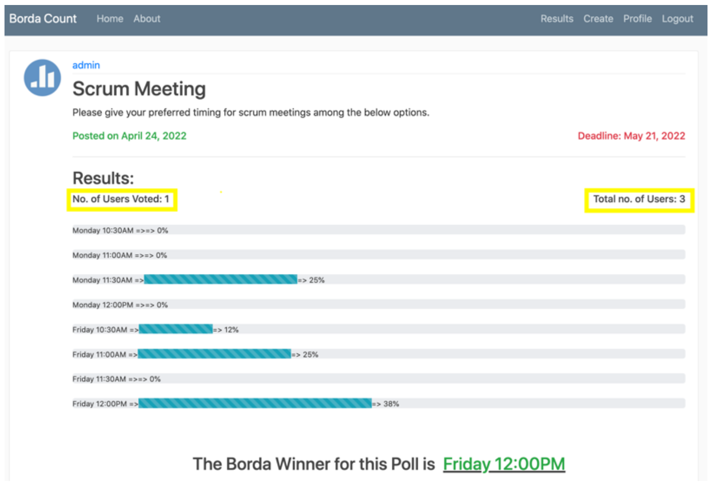
Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

Deadline: May 21, 2022

Results when only a percent of users have taken the poll



admin

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

Deadline: May 21, 2022

Results:

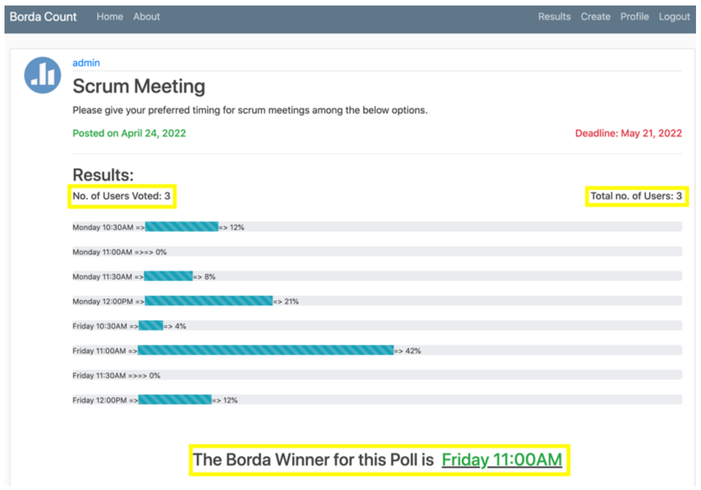
No. of Users Voted: 1

Total no. of Users: 3

```
Monday 10:30AM =>=> 0%
Monday 11:00AM =>=> 0%
Monday 11:30AM => => 25%
Monday 12:00PM =>=> 0%
Friday 10:30AM => => 12%
Friday 11:00AM => => 25%
Friday 11:30AM =>=> 0%
Friday 12:00PM => => 38%
```

The Borda Winner for this Poll is Friday 12:00PM

User needs to allocate points to the preferences



admin

Scrum Meeting

Please give your preferred timing for scrum meetings among the below options.

Posted on April 24, 2022

Deadline: May 21, 2022

Results:

No. of Users Voted: 3

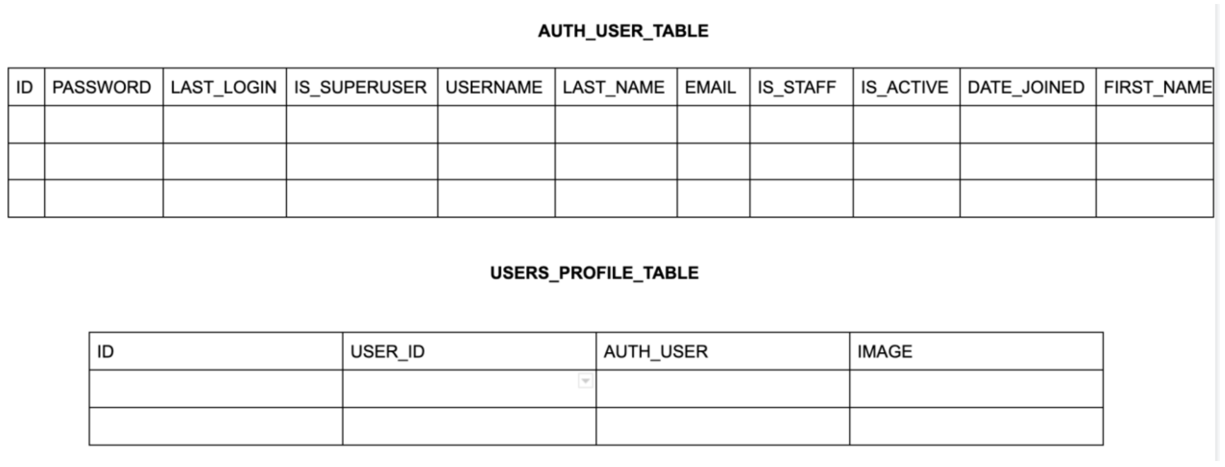
Total no. of Users: 3

```
Monday 10:30AM => => 12%
Monday 11:00AM =>=> 0%
Monday 11:30AM => => 8%
Monday 12:00PM => => 21%
Friday 10:30AM => => 4%
Friday 11:00AM => => 42%
Friday 11:30AM =>=> 0%
Friday 12:00PM => => 12%
```

The Borda Winner for this Poll is Friday 11:00AM

Borda Winner for this poll gives an insight to the admin about the preferred time chosen by all the users.

## **Database design diagram:**



**AUTH_USER_TABLE**

| ID | PASSWORD | LAST_LOGIN | IS_SUPERUSER | USERNAME | LAST_NAME | EMAIL | IS_STAFF | IS_ACTIVE | DATE_JOINED | FIRST_NAME |
|---|---|---|---|---|---|---|---|---|---|---|
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

**USERS_PROFILE_TABLE**

| ID | USER_ID | AUTH_USER | IMAGE |
|---|---|---|---|
|  |  |  |  |
|  |  |  |  |

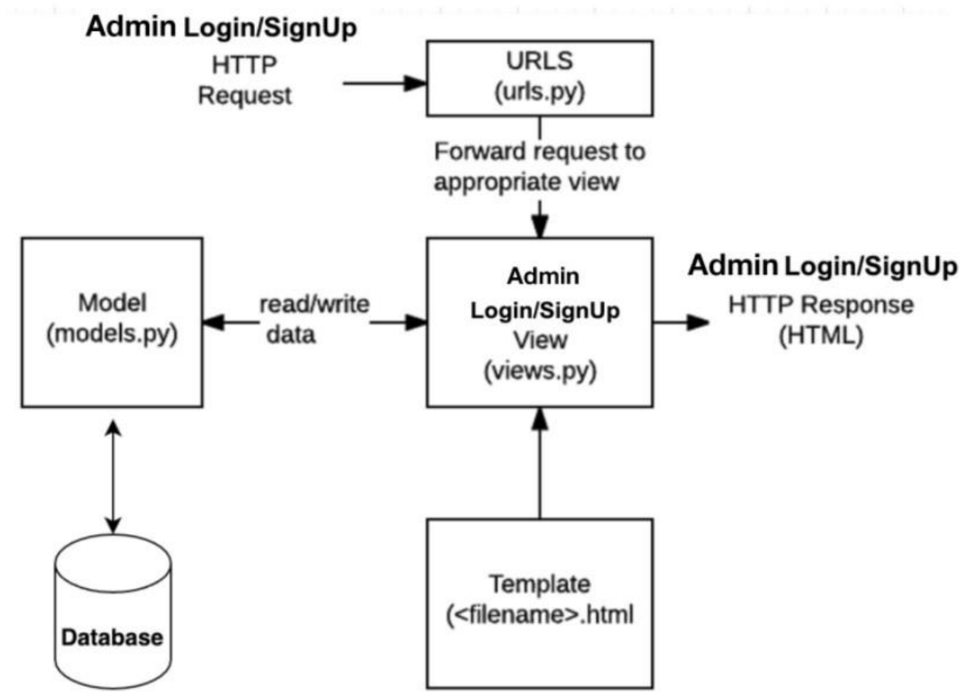
Post design:

| ID | AUTHOR | TITLE | CONTENT | DATE_POSTED | DEADLINE | ALLOWED_USERS | ANSWERED_USERS | OPTIONS |
|---|---|---|---|---|---|---|---|---|
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Submission design:

| ID | OPTIONS | PREFERENCES | POST_ID | SUBMITTED_BY | SUBMITTED_DATE |
|---|---|---|---|---|---|
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Design diagram:

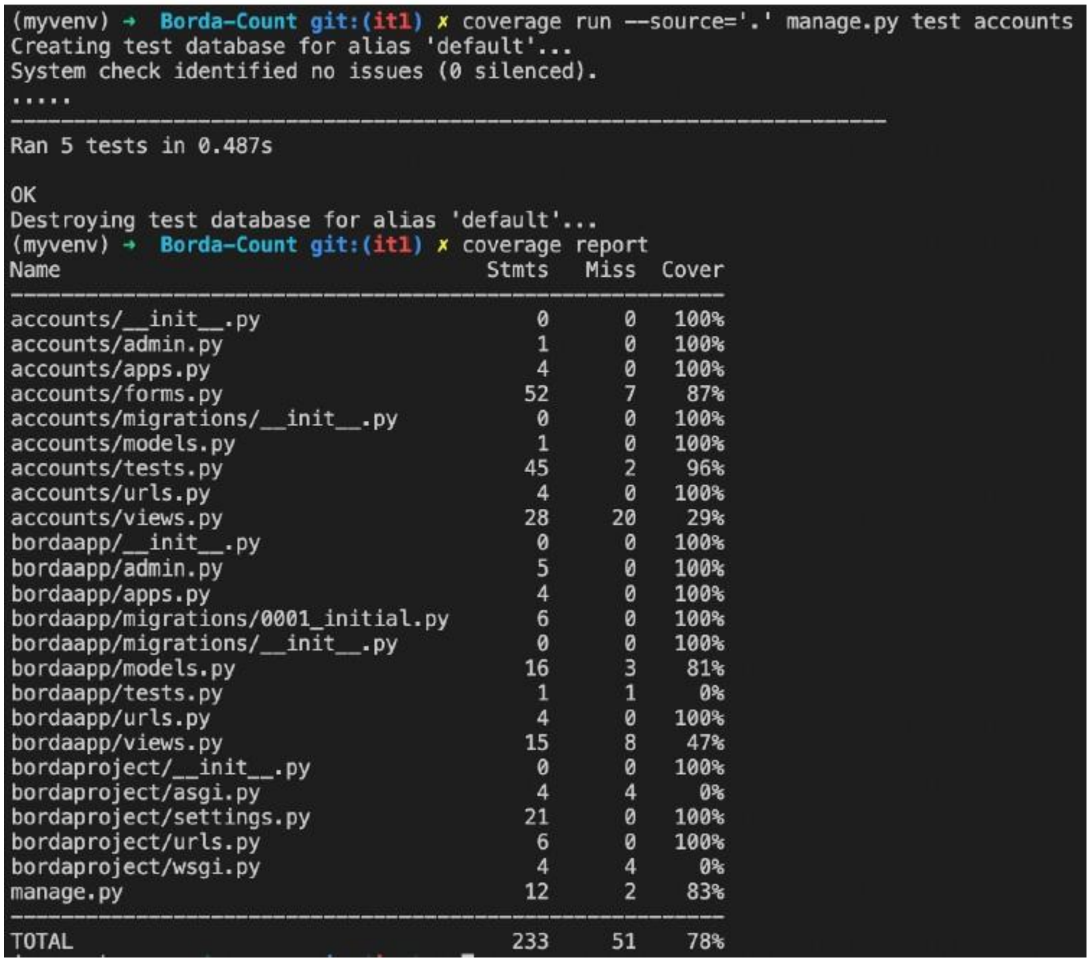


Django uses the Model View Template (MVT) pattern. MVT has views for receiving HTTP requests and returning HTTP responses. When an admin/user creates/submits/views a poll, the respective HTTP requests will be sent. Using urls.py, the application eventually calls the respective view. These views use the models (Post & Submission) that interact with the database. In this way, the admin can create and view polls and users can submit their preferences.

## Testing:

Test case 1: User email validation - Check if the given email is in valid format.
Test case 2: Password validation - Check if the password is meeting the requirements.
Test case 3: Authentication validation- Check if the user is providing the correct password.
Test case 4: Sign up validation - Check if the user is a new user not an existing one.
Test case 5: User sign in validation - Check if the user is providing the correct username.

### Code Coverage



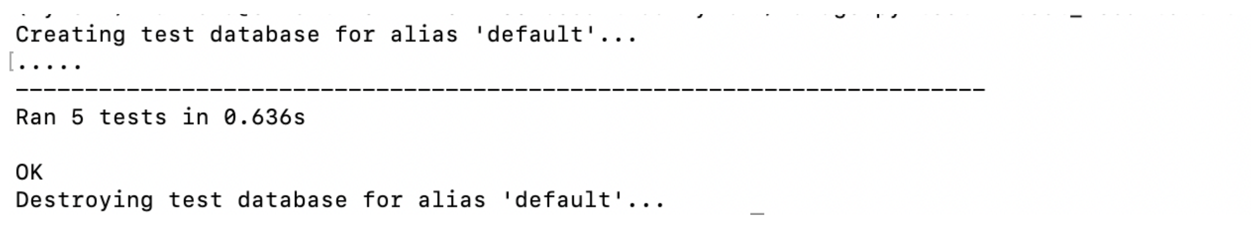
```
(myvenv) → Borda-Count git:(it1) ✗ coverage run --source='.' manage.py test accounts
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.....
----------------------------------------------------------------------
Ran 5 tests in 0.487s

OK
Destroying test database for alias 'default'...
(myvenv) → Borda-Count git:(it1) ✗ coverage report
Name                                      Stmts   Miss  Cover
-------------------------------------------------------------
accounts/__init__.py                          0      0   100%
accounts/admin.py                             1      0   100%
accounts/apps.py                              4      0   100%
accounts/forms.py                            52      7    87%
accounts/migrations/__init__.py               0      0   100%
accounts/models.py                            1      0   100%
accounts/tests.py                            45      2    96%
accounts/urls.py                              4      0   100%
accounts/views.py                            28     20    29%
bordaapp/__init__.py                          0      0   100%
bordaapp/admin.py                             5      0   100%
bordaapp/apps.py                              4      0   100%
bordaapp/migrations/0001_initial.py           6      0   100%
bordaapp/migrations/__init__.py               0      0   100%
bordaapp/models.py                           16      3    81%
bordaapp/tests.py                             1      1     0%
bordaapp/urls.py                              4      0   100%
bordaapp/views.py                            15      8    47%
bordaproject/__init__.py                      0      0   100%
bordaproject/asgi.py                          4      4     0%
bordaproject/settings.py                     21      0   100%
bordaproject/urls.py                          6      0   100%
bordaproject/wsgi.py                          4      4     0%
manage.py                                    12      2    83%
-------------------------------------------------------------
TOTAL                                       233     51    78%
```

Test case 6 : Post Object validation - Check if the post object is getting created successfully.
Test case 7 : Object creation : Check if Post Object is created with correct values.
Test case 8 : Submission Object validation - Check if the submission object is getting created successfully.
Test case 9 : User Authorization : Check if a user has access to submit preferences to a post.

Test case 10 : Borda Count Validation : Verify that the Borda Count algorithm runs successfully.

Test report:



```
Creating test database for alias 'default'...
.....
----------------------------------------------------------------------
Ran 5 tests in 0.636s

OK
Destroying test database for alias 'default'...
```

CodeClimate Link for the project: https://codeclimate.com/github/shaista5555/Borda-Count/progress/maintainability

## <u>**Challenges Faced:**</u>

- Borda count algorithm was new to everyone across the team, so we went through Wikipedia and other learning sources to understand the algorithm.
- The team members were not very familiar with the Django framework, so they had to learn and adapt to it quickly.
- Challenge with AWS Cloud9 implementation – We faced an issue with the version of SQLite3 while setting up the project in AWS Cloud9. It was solved by following a procedure to upgrade the version of Django.
  Following is the link to the solution: Solution to version upgrade of Sqllite3
- Deployment Issue: On altering the existing Submission model to add a new attribute, the existing data started throwing exceptions, as the field was not present for old rows. We tried adding the default argument to solve the issue, but it still did not help. So after checking the production data and confirming it was raw and meaningless, we unmigrated the existing migration files, deleted all existing migration files under 'bordaapp/migrations', recreated migrations and pushed them using the below command:

  heroku run python manage.py migrate bordaapp zero
  delete all the migration files under 'bordaapp/migrations'
  heroku run python manage.py makemigrations
  git add .
  git commit -m <msg>
  git push heroku main
  heroku run python manage.py migrate

## Customer Interaction

- **02/23/2022:** We met in-person with our customer and received the overview of the project.
- **02/27/2022:** We scheduled weekly scrum calls on zoom with our customer for every Wednesday from 11:30am – 12:00pm.
- **03/02/2022 MoMs**:
    - Supervisor to clarify requirements about multiple admins
    - Team to check the legacy code.
    - Team to create wireframe for adding new Admin.
    - Team to explore the DB changes that needs to be done.
- **03/23/2022 MoMs:**
    - Team demoed working UI prototype as a plan for multi-admin functionality.
    - Discussed queries about the future work for the project, Cumulative Borda Count.
- **03/30/2022 MoMs:**
    - Team demoed multi-admin functionality using two approaches on Heroku.
    - Supervisor to forward it to Prof. Ray and gather further feedback.
- **04/06/2022 MoMs:**
    - Discussed the queries about the new Cumulative Borda-Count Algorithm.
- **04/20/2022 MoMs:**
    - Team demoed the implemented version of Cumulative Borda-Count.
    - Discussed the validations we implemented for the new algorithm and asked for further inputs.
- **05/10/2022 MoMs:**
    - Team to give final walk-through of Multi-admin functionality and Cumulative Borda Count to Prof. Ray and his team.

## Software Development Process

We have followed BDD methodology. Following BDD has helped speed up the development process. With BDD, as its name suggests, focus is on behavior rather than implementation itself. BDD enhances quality of the code, thus reducing the maintenance cost and minimizing project risk.

## Configuration Management

We have created separate branches for implementing the user requirements and once implemented we merged the changes to the parent repository. Deployed the code to Heroku and tested the application by creating 4 users and 2 admins. Demo was presented to the client post every iteration.

The credentials for the initial user and admin is held by Prof. Korok Ray and will by provided to the future teams. No other credentials are needed.

## Repo Content

**Installation:** (also present in Readme.txt)

- To create Virtual Environment (First Only, skip this step and activate Virtual Environment if you have the setup already.)

  `python3 -m venv myvenv`
- To activate Virtual Environment:

  `source myvenv/bin/activate`
- To update pip (First time only, skip this step if you're not setting up the application)

  `python -m pip install --upgrade pip`
- To install necessary packages for the applications, use:

  `pip install -r requirements.txt` Note: requirements.txt has the required packages for this application. Do `pip freeze > requirements.txt` to add newly installed packages.
- To run migrations:

  `python manage.py migrate` Note: Migrations are to be run for the first time and only when there's a change in models.py
- To run the server:

  `python manage.py runserver` Visit http://127.0.0.1:8000/ to see the Application running.

### Deployment

The web application is deployed to Heroku using git as a version control. Following commands are used:

```
heroku login
heroku git:remote -a bordacount
git push heroku main
heroku logout
```

## Future Scope

- Add email validations and forward them emails once a new poll is created for them.
- Change the UI for accepting input for allowed users and options to make poll creation easy and convenient.
- Add features to allow poll creation for distributed lists/user groups, rather than individual users.
- Although we have covered the basic input handling at all the places, we can add a feature to do comprehensive input handling.
- We can try using other variants of Borda Count algorithm with this application.

## Project Resources

***Heroku Deployment***
The application till today is deployed on Heroku at this link: https://bordacount.herokuapp.com/

***Pivotal Tracker: https://www.pivotaltracker.com/n/projects/2556139***

***Source code: https://github.com/msharsha/Borda-Count***
Final source code till today can be found under the main branch.

***Slack: https://softwareengin-nat9577.slack.com/archives/C033ZEWBT9B***

***Project Demo and Poster Video:*** *https://youtu.be/W21kQuD2QHs*

THANK YOU