Gymkhana Nominations Portal

Summer Project - Programming Club

Ashish Kumar

31 May 2017

1 Acknowledgement

I would like to thank Programming Club for giving me the opportunity to create the portal and learn about backend web development in Django.

2 Repository

The project repository can be found **here** and the live project is hosted **here**.

2.1 Mentors

- Kunal Kapila
- Yash Srivastav
- Pratham Verma

3 Description

The main aim of the project is to develop a portal through which post-holders can make nominations for their child posts and send it to their parent post for approval and so on.When it reaches the top authority for approval he/she can release nominations to users. The target group can apply for the same. The portal can keep track of each and every process the application goes through, e.g release, submission, approval from higher authority, interview remarks, final selection and increment in power if selected. The detailed documentation can be found *here*.

4 Timeline

4.1 Week 1

1. We started off by creating a simple web-app about a library through which normal users can issue a book and depending on the availability, librarian (aka admin) can either reject or accept the issuance.Furthermore users were also notified in notification section. This was meant as an exercise for the actual project, and to get familiar with django.

2. Having learned the basics of django,started learning about using signals in django, deploying website on server and integrating IITk Authentication.

3. Decided to use django-filter for better and fast extration of data and added filters for books and authors. Final code till now looks clean and user friendly.

4. Started working on the main project and created a basic one level version of the portal but without forms.

4.2 Week 2

1. Started learning and creating Dynamic forms. Compleated the basic version in which form-creator can add multiple questions with specific type like short answer, multiple correct, integertype, etc. However struggled with how to save and store their filled answers effectively.

2. After getting the detailed flowchart from mentors, We started working on the models of Portal. Created Post model with infinite nesting, each one having a parent post, and thus extending the portal to multi-level.

3. Merged *Dynamic Forms* as a separate app in project and linked with 'create nomination'. Improved the filled forms saving and displaying model by using the *Json* library of python to store filled-answers.