A Job Portal – HIREME based on Django and PostgreSQL

Project basic working directory structure-

```
.gitignore
job/
   static/
   templates/
   __init__.py
   admin.py
   apps.py
   forms.py
   models.py
   tests.py
   urls.py
   views.py
Jobportal/
         __init__.py
         settings.py
         urls.py
         wsgi.py
manage.py
README.md
requirements.txt
```

Requirements

- Python 3.6+

Test Locally

- `git clone https://github.com/ag602/Jobportal.git`
- `cd Jobportal`
- `Make virtual env`
- `pip install -r requirements.txt`
- `Setup postgres on the system***`
- `python3 manage.py makemigrations` and `python3 manage.py migrate`
- `python manage.py runserver`

PAYPAL REST API Setup

- Log into the Developer Dashboard(https://www.paypal.com/signin?returnUri=https%3A%2F%2Fdeveloper.paypal.com%2Fdeveloper%2Fapplications) with your PayPal account credentials. If you don't have an account, you can click on the sign up option

- On My Apps & Credentials, use the toggle to switch between live and sandbox testing apps
- Navigate to the REST API apps section and click Create App
- Type a name for your app and click Create App. The app details page opens and displays your credentials
- Copy and save the client ID and secret for your app
- Review your app details and save your app
- Go to views.py
- Change SandboxEnvironment to LiveEnvironment and add the credentials(PAYPAL_SECRET & PAYPAL_CLIENT_ID) generated above
- For more help, refer to - https://github.com/paypal/Payouts-Python-SDK

Libraries/Frameworks used

- For notifications: django-notifications-hq
- For Personal Messages: djangorestframework
- For Paypal Payments: django-paypal
- For Paypal Payouts: Payouts-Python-SDK
- For Social Login: social_django

Future Work

- Help texts to guide the customers for clarity of a feature
- How it Works? - Should deliver more information for users to know how actually the site works
- Payment Modes, apart from Paypal should work
- Credit management feature more robust and trustable
- Email alerts for a particular search and sorting in Jobs
- Location search Google Maps and Places API Credits
- Share the Job with friends on social media
- Similar Jobs in Job Page
- Credit transaction history
- Messages Dashboard and code improvisation

`*** FOR INSTALLING POSTGRES ON UBUNTU -` https://www.digitalocean.com/community/tutorials/how-to-install-and-use-postgresql-on-ubuntu-16-04

How it works

Django is a high-level Python Web framework encouraging rapid development and pragmatic, clean design. A web application framework is a toolkit of components all web applications need. The goal here is to allow developers to instead of implementing the same solutions over and over again, focus on the parts of their application that are new and unique to their project.

A simple Django flow is: URLS -> VIEWS -> TEMPLATE

All Templates and their use -

- *base.html* – Contains navbar(header) and footer. Is extended on all templates.
- *home.html* – home page
- *chat/index/message.html* – all contain user-to-user messaging frontend
- *baseprofile.html* – contains sidebar used in dashboard and extended to all dashboard templates containing sidebar
- *notifications.html* – notification page
- *appliedjobs.html* – Page contains applied jobs
- *credits.html* – credits page
- *bookmarks.html* – All bookmarks page
- *manage_candidates.html* – All candidates for the jobs you posted
- *post_a_job.html* – Post job page
- *profile.html* – Dashboard home page
- *profile_edit.html* – Edit profile page
- *manage_jobs.html* – Manage your job postings in a single page
- *seefreelancerprofile.html* – See the profile of freelancer
- *404 and 500.html* – Error pages
- *job_list*.html – List all jobs
- *job_page*.html – Specific job page
- *pages*-login*.html – login page*
- *pages_register.html* – First register page
- *pages_register_2.html – Second register page*
- *social_login.html – Google/linkedin auth page*
- *orders.html – List all transactions including current wallet balance*
- *checkout_confirmation.html – 2nd checkout page*
- *checkout_page.html* – Coupon Apply, plan select page,1st checkout page
- *invoice.html – Invoice page*
- *pricing_plans.html* – All plans in 1 page
- *order_confirmation.html* – Order confirmation page. Displays a success msg
- *withdraw_amount_page.html* – Withdraw page 1. Write amt to withdraw
- *withdraw_successful.html* – withdraw success page

- *email-job-accepted-freelancer.html* – Email when job applied. Needs to improve
- *contact.html* – Contact us page
- *privacypolicy and terms.html*