Project Document - IPO Web App Development

Workspace Login URL : https://bluestock.in/blueflow/m
Login Details : You will receive over mail
Work Policy : https://bluestock.in/hr/work-policy.html

Project: IPO Web Application & REST API Development
Company: Bluestock Fintech
Website : www.bluestock.in
Mail Id : hello@bluestock.in
WA Contact :9225220170
Date: 01/06/2025
Department : #BF-SOFT-DEV

Project Manager - Yash Kale
Mail id - yashkale@bluestock.in
WA No : 9209550273

Important :-

Check UI UX Design- https://www.figma.com/design/aNuWMnlnjXdsVgCP350ftH/IPO-Web-App---M2-J1?node-id=0-1&t=JkmueTInQSBeLW6n-1

Check System Design: - https://www.figma.com/board/g9bjreevYNJkfMuwRacyaP/System-Design?t=rhom7O3DRl5pdHkG-1

Assets (Logo,SVG, PNG Etc : https://drive.google.com/drive/folders/1yH9Y_mIqqEkZXtzhqHSuFtEwFOr8BXH5?usp=drive_link

Introduction
This report outlines the production-level software development tasks assigned to our interns at Bluestock Fintech. The project involves developing an IPO web application and design REST API for the Bluestock website/app and our clients' websites/apps. These tasks are designed to enhance the interns' practical

knowledge and contribute to our ongoing projects. Each task aligns with our company's goals and the interns' skills and career aspirations.

Objective:

Develop a web application and REST API that provides IPO-related information to the public. The application will display information such as company logo, name, price band, opening and closing dates, issue size, issue type, listing date, status, IPO price, listing price, listing gain, current market price (CMP), and current return. Additionally, it will include downloadable RHP and DRHP PDFs.

Data Source : Add dummy IPO Data in database which shown on upcoming ipo webpage (refer ui ux design)

Software Requirements

Backend:

- SDK: Python (Version 3.12.3)
- Framework: Django (Version 5.0.6) - `pip install Django`
- API: Django REST Framework (Version 3.15.1) - `pip install djangorestframework`
- Tools: Postman for API testing, Git & GitHub for version control

Frontend:

- Technologies: HTML, CSS, plain JavaScript (no NodeJS)
- Framework: Bootstrap 5 (via CDN link)
- Database: PostgreSQL
- IDE: Visual Studio Code (VS Code)

1. Prerequisites

Interns are required to familiarize themselves with the following concepts before starting the project:

What is Stock Market ? - https://youtu.be/Zt6bVL39b9M?si=efZqf77wkWaOrnAQ
What is IPO ? - https://youtu.be/gWrR5qPEmWE?si=QAf19b_i-uSMn3vV

PROJECT FEATURES

Public User Interface:

- Home page: List of IPOs (upcoming, ongoing, listed)

- IPO detail page with full IPO data
- Download RHP & DRHP PDFs
- Search & Filter by status or name

Admin Panel:
- Login-secured interface
- Create, update, delete IPO data
- Upload PDF files (RHP, DRHP)
- Manage logos and all IPO fields

API:
- /api/ipo/ — List IPOs
- /api/ipo/<id>/ — IPO details
- Filtering, search & sort support

This information will be available on both the Bluestock website/app and clients' websites/apps.

PROJECT STRUCTURE

```
ipo_project/
├── ipo_app/
│   ├── models.py
│   ├── views.py
│   ├── serializers.py
│   ├── urls.py
│   └── admin.py
├── ipo_project/
│   ├── settings.py
│   ├── urls.py
├── media/  # PDF & logo uploads
├── templates/
│   └── home.html, detail.html, ...
├── static/
├── db.sqlite3 / PostgreSQL
└── manage.py
```

Development Overview

Important Notes:-
- This is a production-level project; therefore, write quality code and test features regularly.
- Apply necessary security rules to ensure data protection and integrity.
- Utilize ChatGPT for testing code, writing features, and designing the database schema.

1. References

https://youtu.be/s7aINQPGNDM?si=ydfab7GMUaMCg91W

https://www.youtube.com/watch?v=C1NgOmoOszc&list=PLjVLYmrlmjGcyt3m6rt21nfjhYSWP_Ue_

https://youtu.be/5EY6JFptZgw?si=nwZ5DE7nVc4TXs8X

9.Deadlines

Date	Work Status
-	Complete All Frontend (Client Side, Admin Dashboard)
-	Complete Entire Project

//Deadline for entire project submission updated in workspace

Submission Guidelines
- Do not commit buggy or error-prone code.
- Thoroughly test your code on your local machine before committing.
- Commit changes to the main branch and push them.

10. **Resources for This Project Assets**
 (Logo,SVG, PNG Etc) :
https://drive.google.com/drive/folders/
11Q6kapopPp4tVn82QKRCX8cpHYHO5GCK?usp=drive_link
(Request Access)

Figma UI/UX Prototype:

Please visit link for complete details -
https://www.figma.com/design/aNuWMnlnjXdsVgCP350ftH/IPO-Web-App---M2-J1?node-id=0-1&t=JkmueTInQSBeLW6n-1

BF Bluestock Fintech

Search 🔍   V Hi, Vishal ⌄ 🔔

MENU

⊞ Dashboard

🛒 Manage IPO

📄 IPO Subscription

◎ IPO Allotment

OTHERS

⚙ Settings

API Manager

👤 Accounts

Help

Upcomming IPO | Dashboard   Register IPO

Company	Price Band	Open	Close	ISSUE SIZE	ISSUE Type	Listing Date	Status	Action	Delete/View
Adani Power	₹ 329 - 136	2023-06-03	2024-06-05	45530.15 Cr.	Book Built	2023-06-10	Ongoing	Update	🗑 👁
VBL LTD	₹ 229 - 136	2024-06-03	2024-06-05	1330.15 Cr.	Book Built	2018-06-10	Comming	Update	🗑 👁
Tata Motor	₹ 12549 - 136	2024-06-03	2024-06-05	1340.15 Cr.	Book Built	2016-06-10	New Listed	Update	🗑 👁
HDFC LTD	₹ 1244 - 136	2024-06-03	2024-06-05	830.15 Cr.	Book Built	2029-06-11	Comming	Update	🗑 👁
Tata Motor	₹ 629 - 136	2024-06-01	2024-06-05	820.15 Cr.	Book Built	2023-06-10	Ongoing	Update	🗑 👁
VBL LTD	₹ 629 - 136	2024-06-03	2024-06-05	130.15 Cr.	Book Built	2024-06-10	Comming	Update	🗑 👁
Tata Motor	₹ 6729 - 136	2024-06-03	2024-06-05	170.15 Cr.	Book Built	2027-06-10	New Listed	Update	🗑 👁
VBL LTD	₹ 1629 - 136	2024-06-03	2024-06-05	130.15 Cr.	Book Built	2022-06-10	Comming	Update	🗑 👁
Tata Motor	₹ 2329 - 136	2024-06-03	2024-06-05	130.15 Cr.	Book Built	2023-06-10	New Listed	Update	🗑 👁
VBL LTD	₹ 329 - 136	2024-06-03	2024-06-05	130.15 Cr.	Book Built	2021-06-10	Comming	Update	🗑 👁

‹ 1 2 ... 9 10 ›

System Design:

View More: -

https://www.figma.com/board/g9bjreevYNJkfMuwRacyaP/System-Design?node-id=0-1&p=f
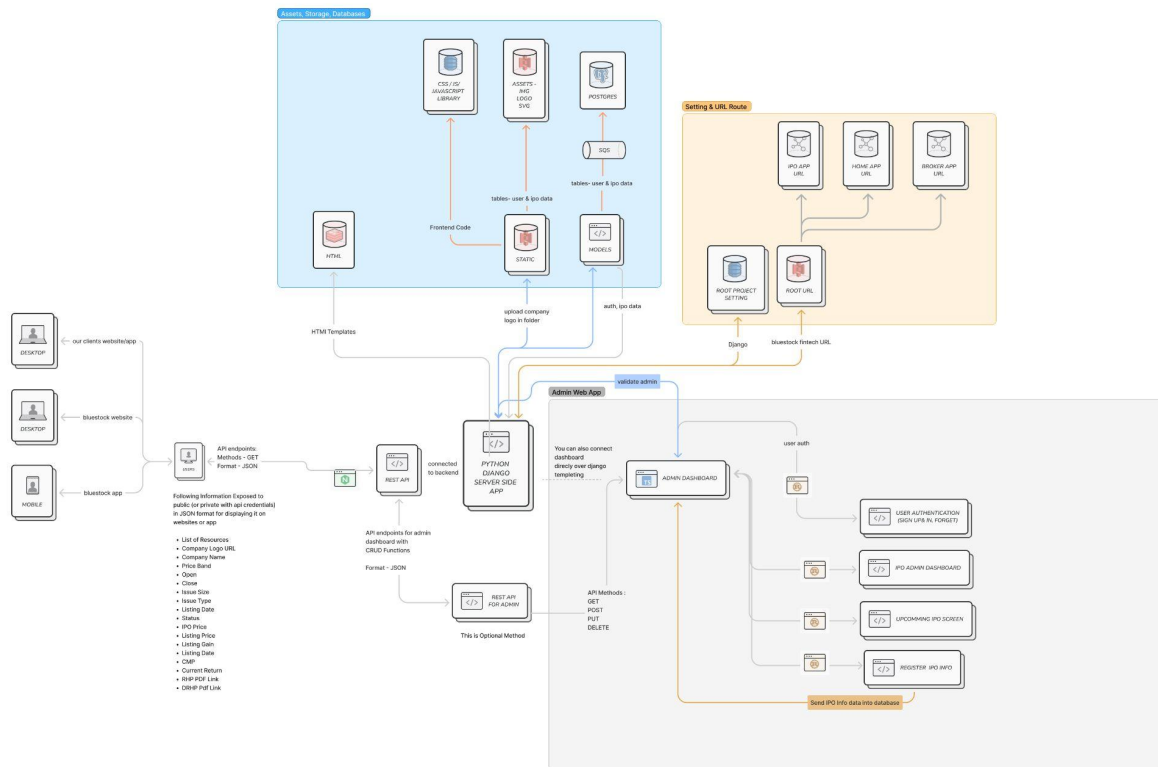
System Design : IPO Web Application & REST API
By Bluestock Fintech

Django Models

```python
from django.db import models

class IPO(models.Model):
    STATUS_CHOICES = [
        ('upcoming', 'Upcoming'),
        ('ongoing', 'Ongoing'),
        ('listed', 'Listed'),
    ]

    company_name = models.CharField(max_length=255)
    logo = models.ImageField(upload_to='logos/')
    price_band = models.CharField(max_length=100)
    open_date = models.DateField()
    close_date = models.DateField()
    issue_size = models.CharField(max_length=100)
    issue_type = models.CharField(max_length=100)
    listing_date = models.DateField(null=True, blank=True)
    status = models.CharField(max_length=20, choices=STATUS_CHOICES)
    ipo_price = models.FloatField(null=True, blank=True)
```

```python
    listing_price = models.FloatField(null=True, blank=True)
    current_market_price = models.FloatField(null=True, blank=True)
    rhp_pdf = models.FileField(upload_to='docs/', null=True, blank=True)
    drhp_pdf = models.FileField(upload_to='docs/', null=True, blank=True)

    @property
    def listing_gain(self):
        if self.ipo_price and self.listing_price:
            return round(((self.listing_price - self.ipo_price) / self.ipo_price) * 100, 2)
        return None

    @property
    def current_return(self):
        if self.ipo_price and self.current_market_price:
            return round(((self.current_market_price - self.ipo_price) / self.ipo_price) * 100, 2)
        return None

    def __str__(self):
        return self.company_name
```

ER Diagram Description:

Entities and Attributes:

1. **Companies**
   - **company_id**: INT (Primary Key, Auto-Increment)
   - **company_name**: VARCHAR(255) (Not Null)
   - **company_logo**: VARCHAR(255) (URL or file path)
2. **IPOs**
   - **ipo_id**: INT (Primary Key, Auto-Increment)
   - **company_id**: INT (Foreign Key referencing Companies.company_id)
   - **price_band**: VARCHAR(50)
   - **open_date**: DATE
   - **close_date**: DATE
   - **issue_size**: VARCHAR(100)
   - **issue_type**: VARCHAR(50)
   - **listing_date**: DATE
   - **status**: ENUM('Upcoming', 'Open', 'Closed', 'Listed')
   - **ipo_price**: DECIMAL(10,2)
   - **listing_price**: DECIMAL(10,2)
   - **listing_gain**: DECIMAL(5,2)
   - **current_market_price**: DECIMAL(10,2)

- ○ **current_return**: DECIMAL(5,2)
3. **Documents**
    - ○ **document_id**: INT (Primary Key, Auto-Increment)
    - ○ **ipo_id**: INT (Foreign Key referencing IPOs.ipo_id)
    - ○ **rhp_pdf**: VARCHAR(255) (URL or file path)
    - ○ **drhp_pdf**: VARCHAR(255) (URL or file path)

Relationships:

1. **Companies to IPOs**
    - ○ **Type**: One-to-Many
    - ○ **Description**: One company can have multiple IPOs, but each IPO is associated with exactly one company.
    - ○ **Foreign Key**: company_id in IPOs references company_id in Companies.
    - ○ **On Delete**: CASCADE (if a company is deleted, all related IPOs are deleted).
2. **IPOs to Documents**
    - ○ **Type**: One-to-Many
    - ○ **Description**: One IPO can have multiple documents, but each document is associated with exactly one IPO.
    - ○ **Foreign Key**: ipo_id in Documents references ipo_id in IPOs.
    - ○ **On Delete**: CASCADE (if an IPO is deleted, all related documents are deleted).

```
[Companies] --(1:N)-- [IPOs] --(1:N)-- [Documents]
    |                       |                   |
    | company_id (PK)       | ipo_id (PK)       | document_id (PK)
    | company_name          | company_id (FK)   | ipo_id (FK)
    | company_logo          | price_band        | rhp_pdf
                            | open_date         | drhp_pdf
                            | close_date        |
                            | issue_size        |
                            | issue_type        |
                            | listing_date      |
                            | status            |
                            | ipo_price         |
                            | listing_price     |
                            | listing_gain      |
                            | current_market_price |
                            | current_return       |
```