

Assignment Description

Advanced Django Rest Framework Internship Project.

Project Requirements

1. API Creation:

- Create a Django project and configure it with Django Rest Framework.
- Design and implement a RESTful API to manage a collection of books.
- The API should support the following operations:
 - Retrieve a list of all books.
 - Retrieve a single book by its ID.
 - Create a new book.
 - Update an existing book.
 - Delete a book.

2. Partial Update of Records with Validation:

- Extend the API to allow partial updates of book records.
- Implement an endpoint to update specific fields of a book record with proper validation.
- Validate input data based on predefined criteria (e.g., length, format, etc.).
- Return appropriate error responses for invalid data.

3. User Signup with Email Verification:

- Implement user signup functionality with email verification.
- Users should be able to create an account by providing their email and password.
- Send a verification email to the user's provided email address.
- Implement a verification process where users can verify their email addresses by clicking a link in the email.
- Allow access to certain endpoints only after successful email verification.

4. Advanced Filtering:

- Implement advanced filtering options for the book collection.
- Allow users to filter books based on multiple criteria (e.g., title, author, publication year, etc.).
- Implement pagination to limit the number of results per page.
- Provide sorting options for the filtered results.

5. Pagination:

- Implement pagination for the API endpoints that return a list of books.
- Allow users to specify the number of results per page and navigate through the pages.
- Include metadata in the API response to provide information about pagination (e.g., total number of books, current page, next page, previous page, etc.).

6. File Upload:

- Extend the API to support file uploads, such as book cover images or PDF files.
- Implement an endpoint for uploading files and associate them with book records.
- Handle file validation, including size limits, file type checks, and secure storage.
- Provide appropriate error responses for invalid or failed file uploads.

Deliverables

1. Codebase:
 - Create a Git repository and maintain the codebase throughout the project.
 - Ensure that the code is well-organized, follows best practices, and is properly documented.
 - Include a README file that explains how to set up and run the project.
2. API Documentation:
 - Document the API endpoints, their functionalities, and the expected request/response formats.
 - Use tools like Swagger or Django Rest Swagger to generate API documentation.
 - Include the API documentation in the project repository.
3. Demo and Presentation:
 - Prepare a detailed presentation to demonstrate your project and showcase its advanced features.
 - Present the API endpoints, authentication process, filtering options, pagination, file upload functionality, and any additional functionality you have implemented.
 - Discuss any challenges faced during the project and how you overcame them.

Additional Guidelines

- Strive for a clean, efficient, and well-tested codebase.
- Utilize advanced DRF features such as serializers, viewsets, permissions, filters, and pagination.
- Implement email sending functionality using celery.

Note: If a candidate is unable to complete all the given topics within the assignment, they should still submit the project with a note indicating the completed points. We value the candidate's effort and understanding of the concepts, even if they were unable to implement all the features.