**Theory: Social Authentication, Email, and OTP Sending API**

In modern web applications, authentication and user verification are key components. Django offers several ways to handle authentication, including integration with social media logins (such as Google and Facebook) and methods for sending OTPs (One Time Passwords) and emails via third-party APIs.

**Key Concepts:**

1. **Social Authentication in Django:**
   * Social authentication allows users to log in using their existing accounts from platforms like Google, Facebook, and Twitter. This eliminates the need for creating and remembering new usernames and passwords.
   * **django-allauth** is a popular third-party package that makes it easy to integrate social authentication into Django projects. It supports multiple providers, including Google, Facebook, Twitter, and others.
2. **Sending Emails in Django:**
   * Django has built-in email support. You can configure SMTP settings in settings.py to send emails.
   * For more robust email handling (e.g., templates, attachments), you can use third-party services like **SendGrid**, **Mailgun**, or **Amazon SES**.
3. **Sending OTPs:**
   * OTPs are used to verify a user's identity during actions like registration, password reset, or login.
   * **Twilio** is a powerful API that allows sending OTPs via SMS, email, and even voice calls