Lesson 1: Overview of Full-Stack Development

1.1 What is Full-Stack Development?

Full-stack development refers to the ability to work on both the frontend (user interface) and backend (server, database) of a web application. A full-stack developer is someone who can handle:

- Frontend: Building user interfaces using frameworks like Angular, React, or Vue.js
- Backend: Creating APIs, handling databases, and business logic using Django, Flask, or Express.js
- Database: Working with PostgreSQL, MySQL, MongoDB
- Version Control & DevOps: Using Git, Docker, and CI/CD pipelines

Full-stack development is essential because it enables developers to build and deploy end-to-end solutions.

1.2 Overview of Django and Angular

Django (Backend)

Django is a high-level Python web framework that encourages rapid development and clean design.

- Follows Model-View-Template (MVT) pattern
- Comes with built-in authentication, admin panel, and ORM
- Uses Django REST Framework (DRF) for API development

Angular (Frontend)

Angular is a TypeScript-based frontend framework developed by Google.

- Uses Component-Based Architecture
- Includes built-in Routing, HTTP Client, and Forms Handling
- Supports UI frameworks like Angular Material

1.3 How Backend and Frontend Interact

1. The backend (Django) provides a RESTful API that serves data in JSON format.
2. The frontend (Angular) fetches data via HTTP requests and displays it in the UI.
3. The frontend sends data back to the backend (e.g., form submissions, authentication).
4. The backend processes and stores the data in a PostgreSQL database.

This client-server model ensures scalability and modularity in full-stack applications.

1.4 Tools and Technologies

Before diving into development, youll need the following tools:

Backend:

- Python 3.x
- Django & Django REST Framework
- PostgreSQL

Frontend:

- Node.js & NPM
- Angular CLI
- TypeScript

Development Tools:

- Git & GitHub
- Postman for API Testing
- VS Code / PyCharm