Becoming a Software Engineer

The Blueprint

O ver the next few weeks, we'll embark on an exciting adventure into the world of software engineering. This series is designed to give you a solid foundation and a glimpse into the many facets of the tech industry. We'll be meeting twice a week for interactive sessions where you'll get to learn both the technical and non-technical aspects of being a software engineer. Think of it as a hands-on guidebook that covers everything from writing clean code to understanding team dynamics and planning your career.
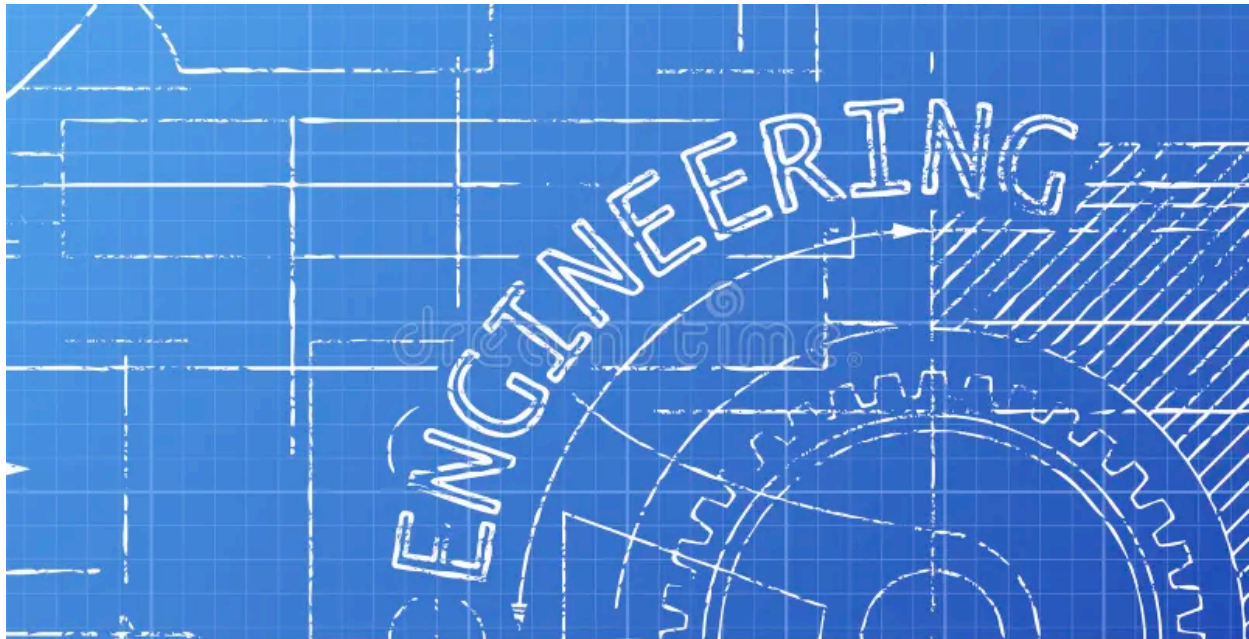
Each session will be a mix of discussions, coding exercises, and practical projects. We'll start with the basics, like setting up your development environment and understanding version control with Git. From there, we'll dive into front-end and back-end development, exploring how they come together to create a full-stack application. Along the way, we'll also talk about Agile methodologies, effective communication, and how to navigate your early career in tech.

Week 1: Introduction and Fundamentals

Session 1: Orientation & Basics

    Overview:
- Personal introduction, discuss background, and learn about the mentee's goals.
- Explain the agenda and structure of the mentorship.

    Key Topics:
- Overview of software engineering roles and responsibilities.
- Introduction to the software development lifecycle (SDLC).
- Basic principles of coding: clean code, version control (Git basics), and IDE setup.

    Practical:
- Set up the development environment (IDE, GitHub/GitLab repository).
- Simple coding exercise to assess current skill level.

Session 2: Project Planning & Version Control

    Overview:
- Introduction to project planning and Agile methodologies (Scrum/Kanban).
- Discuss the importance of version control and collaborative development.

    Key Topics:
- Detailed walkthrough of Git: branching, committing, merging, and pull requests.
- Overview of a typical software project structure.

    Practical:
- Guide the mentee through creating a new project in a repository.
- Hands-on exercise in branching and merging in Git.

Week 2: Front-End Development

Session 3: Front-End Basics

Overview:
- o Introduction to front-end technologies and their role in full-stack development.

Key Topics:
- o HTML/CSS basics and their importance in web development.
- o Introduction to JavaScript: syntax, basic DOM manipulation.

Practical:
- o Build a simple static webpage using HTML and CSS.
- o Add basic interactivity with JavaScript (e.g., a button click event).

Session 4: Advanced Front-End Concepts

Overview:
- o Dive deeper into modern front-end development practices.

Key Topics:
- o Introduction to a front-end framework/library (React, Angular, or Vue).
- o Component-based architecture and state management.

Practical:
- o Build a small, interactive application using the chosen framework/library.
- o Explore debugging and inspecting tools for front-end development.

Week 3: Back-End Development

Session 5: Back-End Basics

Overview:
- o   Introduction to back-end development and server-side technologies.

Key Topics:
- o   Overview of RESTful APIs and their role in web applications.
- o   Introduction to back-end languages and frameworks (Node.js, Python/ Django, Ruby on Rails, etc.).

Practical:
- o   Set up a basic server and create a simple API endpoint.
- o   Demonstrate how to handle HTTP requests and responses.

Session 6: Data Management

Overview:
- o   Discuss the importance of databases in web applications.

Key Topics:
- o   Introduction to SQL and NoSQL databases.
- o   Basic CRUD operations and database schema design.

Practical:
- o   Create and interact with a simple database.
- o   Integrate database operations into the API created in Session 5.

Week 4: Full-Stack Integration & Career Development

Session 7: Full-Stack Project

Overview:
- o   Bring together front-end and back-end skills into a cohesive project.

Key Topics:
- o   Discuss the architecture of full-stack applications.
- o   Introduction to deploying web applications.

Practical:
- o   Develop a simple full-stack application (e.g., a to-do list or a blog).
- o   Guide the mentee through deploying the application to a cloud service (Heroku, AWS, etc.).

Session 8: Non-Technical Skills & Career Guidance

Overview:
- o   Focus on the non-technical aspects of a software engineering career.

Key Topics:
- o   Effective communication and collaboration in a team.
- o   Navigating code reviews and feedback.
- o   Career development: building a portfolio, resume writing, and preparing for technical interviews.

Practical:
- o   Mock code review session.
- o   Review the mentee's resume and provide feedback.
- o   Conduct a mock technical interview or whiteboard session.

Additional Activities:

Pair Programming: Encourage pair programming during sessions to provide hands-on learning.

Homework: Assign small, manageable tasks or readings between sessions to reinforce learning.

Q&A: Leave time in each session for the mentee to ask questions or discuss challenges they're facing.

Notes

Notes

Notes

Notes