Tech Trek: A Journey through Full-Stack Development to Industry Mastery

Workshop Agenda

Overall Agenda

Concepts Practically Covered in the Workshop:

1. **Solution Design for Real-World Problems:**
    - Identifying and designing solutions for practical, real-world challenges.
    - Gathering requirements and creating a design plan.
    - Developing a comprehensive testing plan.
2. **Application Development:**
    - Freezing the tech stack to ensure consistency and compatibility.
    - Building a backend service.
    - Developing a frontend service.
    - Understanding and applying industry best practices for development:
        - Writing unit tests for reliability.
        - Considering UI/UX principles for user-friendly design.
        - Implementing design patterns and principles.
3. **Making the Application Industry-Ready:**
    - **Containerization:** Dockerizing the application for consistency across environments.
    - **Hosting:** Deploying the application:
        - **Docker Way:** Hosting a monolithic application.
        - **Kubernetes (K8s) Way:** Deploying microservices.
    - Understanding industry best practices for DevOps:
        - **Security Considerations:** Implementing security measures in the deployment process.
        - **NOC & SOC Monitoring:** Ensuring the application is monitored effectively for operational continuity and security.

Assignment Tasks after the workshop:

1. **Choose Your Use Case:**
    - Develop a full-stack application following the workshop steps.
    - **Add Persistence:** Integrate a database (SQL or NoSQL) with your backend service.
    - **Deployment Model:** Choose between a monolithic or microservices architecture and plan to host your application using free services.
    - **Showcase Your Work:** Once completed, add the GitHub repository link and any deployed application links to your resume.
    - **Community Engagement:** Share your completed project on social media using the hashtag **#full-stack-industry-maestry** and tag us.

Session wise Agenda

Session-1: The Digital Odyssey - Understanding Computer Science & Software Development

Introduction:

- **Overview of the Workshop Agenda:** Brief outline of what will be covered in the workshop.
- **Icebreaker Activity:** Engage participants with a quick activity to understand their backgrounds and expectations.

Computer Science Fundamentals:

- **Important Subjects in Computer Science:**
  - Overview of key topics like algorithms, data structures, and their relevance.
- **Brief Touch on Algorithms and Data Structures:**
  - Explain the significance of algorithms and data structures in solving problems efficiently.
- **Debugging a Simple Algorithm - Live:**
  - Demonstrate debugging a straightforward algorithm to illustrate problem-solving techniques.

Software Industry Overview:

- **Software Development Lifecycle (SDLC):**
  - Discuss stages like requirements gathering, design, development, testing, deployment, and maintenance.
- **Types of Roles in Tech:**
  - Overview of various roles including developer, tester, DevOps engineer, and more.
- **Emerging Trends:**
  - Briefly introduce trends like Artificial Intelligence (AI), Blockchain, and their impact on the industry.

Q&A Session:

- **Encourage Questions:** Open the floor for participants to ask questions about the industry, different roles, and emerging trends.

Problem Statement & Requirements of the To-Do Application:

- **Requirements Gathering:**
  - Explain the process of gathering requirements for the to-do application project, setting the stage for the hands-on development.

Session-2: Diving Into Development - Building Your First Application

Prerequisites for the Session:

- Python
- Node.js
- Any text editor of your choice (VSCode, Sublime, etc.)

Theory:

- **REST API Standards:** Overview of best practices for designing RESTful APIs.
- **Coding Best Practices:** Tips for writing clean, maintainable, and efficient code.
- **Testing and Code Coverage:** Importance of unit testing and how to measure code coverage.
- **System Design of the Application:** High-level architecture and design considerations for the application.

Hands-On:

- **Backend Development:** Build a simple backend todo app with CRUD APIs using FastAPI.
- **Unit Testing:** Write unit tests for the backend todo app to ensure functionality and reliability.
- **Frontend Development:** Create a simple React frontend that consumes the backend APIs.

Session-3: Bringing Ideas to Life - Deployment and Real-World Operation

Prerequisites for the Session:

- Docker
- Kubectl
- K9s
- Kind

Theory:

- **Docker and Containerization:** Introduction to containerization, Docker basics, and why it's crucial for modern development.
- **Manifest Files (Configuration-Driven Frameworks):** Understanding the structure and purpose of YAML and JSON files in Kubernetes.
- **Monolithic vs Microservice Architectures:** Comparison of monolithic and microservice architectures, and when to use each.
- **Kubernetes:** An overview of Kubernetes and its growing importance in the industry.

Practical:

- **Containerization:** Create Dockerfiles, build Docker images for your application, and push them to Docker Hub.
- **Kubernetes Cluster Setup:** Set up a simple Kubernetes cluster using Kind.
- **Kubernetes Deployment:** Create Kubernetes manifest files for both frontend and backend services.
- **Scaling:** Learn how to scale frontend and backend services by tweaking the deployment configurations.

Advanced Topics (High-Level Overview):

- **CI/CD Integration:** Introduction to continuous integration and deployment practices.
- **Monitoring and Logging:** Overview of tools and practices for monitoring and logging in Kubernetes.
- **Security Best Practices:**
    - **RBAC (Role-Based Access Control):** Understanding and implementing RBAC to secure your Kubernetes environment.

Session-4: Beyond Coding - Building Your Professional Identity

Resume Preparation:

- **What to Include in Your Resume:**
    - Essential sections: Contact information, summary, skills, experience, education, and projects.
    - **Keywords:** Importance of using role-specific keywords to pass ATS (Applicant Tracking Systems).
- **Sample Resume:** Review and analyze a sample resume to highlight best practices.
- **ATS Resume Checker:** Introduction to tools that help ensure your resume is ATS-friendly.
- **Adding Projects:** How to effectively showcase your projects, including adding GitHub links.

Interview Preparation:

- **Stages of a Typical Technical Interview:**
    - **Aptitude Tests:** Common types and how to prepare.
    - **Coding Tests:** Practice platforms like LeetCode, HackerRank, and InterviewBit.
    - **Coding Interviews:** Tips on problem-solving and communication during coding rounds.
    - **HR Interviews:** What to expect and how to prepare for cultural fit and behavioral questions.
- **Self-Introduction & Personal Profile Questions:** Crafting a compelling personal introduction and answering common profile questions.
- **Preparing for HR Interviews:**
    - Understanding the company’s values and culture.
    - Strategies for answering behavioral questions effectively.
- **Learning About the Company:** Researching the company’s background, mission, and recent news to tailor your responses.

Creating Your Professional Identity:

- **Platforms to Create a Profile On:**
  - **LinkedIn, Naukri, InstaHyre:** Setting up profiles on key professional platforms.
  - **Importance of Keywords:** How to use role-specific keywords to enhance visibility to recruiters.
- **Job Opportunities:**
  - **Tech and Non-Tech Roles:** Understanding different career paths and opportunities.
- **Soft Skills:** Emphasizing the importance of communication, teamwork, and problem-solving abilities.
- **General Guidance:** Tips on continuous learning and career development.

Job Experience - 15 Minutes:

- Job experience sharing by the resource persons

Q&A Session:

- Open floor for participants to ask questions and seek queries