SKYPARK ITECH
"The Future of Innovation"

Software Development & System Design Internship

(15 Days | Practical-Oriented | Industry Ready)

INTERNSHIP SYLLABUS

Internship Objectives

- Build **strong coding foundations**
- Understand **real-world system design**
- Learn **architecture documentation**
- Apply knowledge through a **mini project**
- Prepare students for **interviews & real projects**

Day 1 – Introduction to Software Engineering

- What is software development lifecycle (SDLC)
- Agile vs Waterfall
- Roles: Developer, Tester, Architect
- Tools overview (Git, IDEs, Docs)

Practical:

Install development tools

Simple "Skypark World" program

GitHub account setup

Day 2 – Programming Fundamentals

- Variables, data types
- Conditional statements
- Loops & control flow
- Writing clean code

Practical:

Simple logic programs

Code review basics

Day 3 – Functions & Modular Programming

- Functions & methods
- Code reusability
- File structure & folders
- Naming conventions

Practical:

Convert programs into functions

Mini task-based coding

Day 4 – Data Structures Basics

- Arrays, Lists
- Stack & Queue (conceptual)
- Time complexity basics
- Why data structures matter

Practical:

Implement array/list programs

Performance comparison examples

Day 5 – Introduction to System Design

- What is system design?
- Client–Server architecture
- Monolithic vs Microservices
- Real-world examples (Swiggy, Amazon)

Practical:

Draw system flow diagram

Design a simple login system

Day 6 – Object-Oriented Programming (OOP)

- Class & Object
- Encapsulation, Inheritance
- Polymorphism
- Real-time examples

Practical:

Build a Student / Employee class

Apply OOP concepts

Day 7 – Database & Data Flow

- What is a database?
- SQL vs NoSQL (concepts)
- CRUD operations
- Data flow in applications

Practical:

Design database schema

Perform basic CRUD queries

Day 8 – Backend Logic & APIs

- What is backend?
- API concepts
- Request & response flow
- Error handling

Practical:

Create simple backend logic

Test APIs using tools

Day 9 – Architecture Documentation

- UML diagrams
- Flowcharts
- ER diagrams
- Writing technical documentation

Practical:

Create architecture document

System design explanation practice

Day 10 – Software Testing & Optimization

- Types of testing
- Debugging techniques
- Performance optimization
- Code refactoring

Practical:

Debug faulty code

Improve existing program

Day 11 – Mini Project Planning

- Problem statement selection
- Requirement analysis
- Architecture planning
- Technology stack selection

Practical:

Finalize project idea

Create system design document

Day 12 – Project Development (Phase 1)

- Core module implementation
- Backend logic
- Database integration

Practical:

Implement main features

Day 13 – Project Development (Phase 2)

- Additional modules
- Error handling
- Optimization

Practical:

Complete project functionality

Day 14 – Testing & Deployment

- Test cases
- Bug fixing
- Deployment basics
- Documentation finalization

Practical:

Run full testing

Prepare final report

Day 15 – Presentation & Assessment

- Project presentation
- System design explanation
- Viva & evaluation
- Career guidance