Thinking Craftsman

Thinking Craftsman
Consulting and Learning Programs

2025

Nitin Bhide

nitinbhide@thinkingcraftsman.in

Thinking Craftsman – Consulting and Learning

About Nitin Bhide

Feedback about Nitin’s Consultancy and Trainings on LinkedIn

Consulting

Learning Programs

Knowledge Sessions (Tech Talks)

Contact Nitin

Thinking Craftsman

Nitin Bhide

Transformative Tech Leader, Consultant, and Visionary Thinking Craftsman

- With over 30+ years of **pioneering experience** in software development, a deep mastery of 9+ programming languages across diverse domains, including web development, computational geometry, algorithms, edu tech, fintech, legal tech, and video analytics.
- 20+ years of Agile/DevOps expertise,
- Consulting for both MNCs and startups and has honed the skills in crafting robust software architectures and writing impeccable code
- As a Mentor and Executive Committee member of the Smart India Hackathon (2022, 2023) and a Professor of Practice at Chandigarh University, passionate about teaching, mentoring, and inspiring the next generation of software developers.
- Despite extensive experience, Love for coding remains undiminished

What People say about Nitin

Anindya,
Senior Software Engineer

Nitin is by far one of the best Teachers I have met. He is a ocean of knowledge when it comes to software development, software design, software quality and pretty much everything to do with "software". I had a awesome time learning and interacting with him. I think anybody would be extremely lucky to have him as a mentor.

Ajay

Nitin is truly a craftsman at heart. In current context, Nitin is working in Software engineering - how should a software design/ coding be and how it should not be. The messages are always backed by solid real life project examples and the research paper which recommended a practice. Nitin has a very keen eye for code - in few minutes he can make out purpose and quality of code (including any smell).

Why Choose Nitin’s Consultancy /Training Programs?

Learn from a Versatile Industry Expert

- Gain insights from an industry veteran proficient in 9 programming languages.
- Experience across diverse domains including algorithms, computational geometry, web and mobile applications, fintech (options trading), edutech, video analytics, and AI.

Real-World Experience and Practical Advice

- Benefit from practical experiences and advice drawn from various industries.
- Learn strategies and solutions that have been tested and proven in real-world scenarios.

Deep Understanding Beyond the Basics

- Not just the 'how', but the 'why' behind every concept.
- Develop a deeper understanding that empowers participants to apply knowledge effectively.

Hands-On Practice and Research-Backed Insights

- Receive advice and guidance grounded in hands-on practice and thorough research.
- Enhance your skills with practical, actionable insights.

Consulting

Consulting Areas

Design/Architecture

- Software Architecture/Design
- Tech stack modernization
- Software Architecture/Design/Code reviews
- Multi programming language integrations
- Enterprise application architectures
- Application Scaling, Security and Performance
- New product development

Processes and Practices

- Release process and Version Control Best Practices
- DevOps pipeline setup
- Engineering and deployment of GenAI/LLM
- Mentoring the Senior Developers/Architects/Product Managers

Pricing Options

- Man-hour based
- Monthly Retainer (different options)

Please email/call for the details

Comprehensive Learning programs

Comprehensive Learning programs
to evolve a Software Developer into
The Thinking Craftsman

Programs for Software Developers

Basic/Foundation Programs

- Coding Best Practices (C++, Java)
- Understanding Version Control
- Fundamental & Solid Design Principles
- Programming Under the Hood
- C++ Deep Dive
- C++ 20 Easy Parts
- Understanding Internet Programming(Web Basics)
- CSS Basics

Practitioner Programs

- Multiple facets of design
- Design for Testability
- Designing Good API
- Effective Code Review
- Secure Programming
- Web Performance Optimization

Experienced Developer Programs

- Design Patterns
- Code Cleanup & Refactoring
- Performance Optimization
- Design Workshop

Programs for All Experience Levels and Domains

Soft skills and Misc Programs

- DevOps Basics
- Agile Basics
- Kanban – Why and How ?
- Mindmapping
- How to Develop Good Check list ?

Tech Talks / Knowledge sessions

- Short 1-hour sessions to introduce teams to different aspects of software development and technology landscape

Foundation programs

(For ALL developers but mainly for Beginners)

Programs : Foundation (for everyone)

	Coding Best Practices	Understanding Version Control
Eligibility	At least 6 months programming experience	At least 6 months programming experience
Contents	• Object life cycle management (initialization/cleanup, • Data structure usage and selection, • Basic Ref count, • Basics error handling	• Why we need version control • Basic concepts in version control (checkout, push/pull, update, merge, commits, logs) • Advanced Concepts (branching, merging, tagging, conflict resolution) • Best Practices to follow
Type	Classroom + hands on exercise	Classroom + hands on exercise using Git
Recommended Batch Size	15 – 25	15 - 25
Duration	1 day	4 hours
Sticker Price	*Email or call for the details*	*Email or call for the details*

Programs : Foundation (for everyone)..

	Fundamental Design Principles	SOLID Design principles
Eligibility	At least 6 months programming experience	At least 2-3 yrs programming experience
Contents	Following principles will be covered • Fail fast • Tell, don’t ask • Design by contract • Minimizing impact of change	• Open Closed Principle • Liskov substitution principles • Dependency inversion principle • Interface segregation principle • Acyclic dependency principle
Type	Classroom	
Recommended Batch Size	15-25	
Duration	1 day	
Sticker Price	*Email or call for the details*	

Programs : Foundation (for all C++ developers)

	Programming Under the Hood	C++ Deep Dive
Eligibility	None	Knows basic C++ syntax
Contents	• Basic concepts of programming, • Calling conventions • Compiler and linker, • lib/DLL etc	• Concept of preprocessor, • Virtual functions, • Function pointers, • Type information, • Exceptions and stack unwind, • Various compiler options
Type	Classroom	Classroom + hands on exercises
Recommended Batch Size	15 – 25	
Duration	1 day	
Sticker Price	*Email or call for the details*	

Programs : Foundation (for everyone)

	Understanding Internet (Web Basics)	Basics of Cascading Style Sheets
Eligibility	At least 6 months web programming experience	At least 6 months web programming experience
Contents	• Common internet protocols • HTTP protocol • HTTP verbs (GET, POST, PUT etc) • HTTP Headers • Security (SSL, TLS) • Session/Authentication/ Authorization	• Style sheets (CSS1, CSS2, CSS3) • Selector and Declaration blocks • CSS rules • Box model and positioning scheme • Layouts • How to design page with CSS • Some tips/recommended practices
Type	Classroom	Classroom
Recommended Batch Size	15 – 25	15 - 25
Duration	4 hours	2 hours
Sticker Price	*Email or call for the details*	*Email or call for the details*

Thinking Craftsman

Programs for practitioners

(for developers with some experience)

Programs for Practitioners

	Multiple Facets of Design	Design for Testability
Eligibility	At least 2-3 yrs programming experience 'SOLID Design principles' program	At least 2-3 yrs programming experience Multiple facets of Design
Contents	Basics of various aspects of design like • methodologies, • design for change, • design for performance, • algorithm design, • UI design etc are covered	• Way of Testivus • What is testability ? • What is 'design for testability' and why we need it ? • Common mistake which make design difficult to test. And how to fix those. • TDD & Design for testability.
Type	Classroom	
Recommended Batch Size	15-25	
Duration	1 day	
Sticker Price	*Email or call for the details*	

Programs for Practitioners ..

	Designing Good API
Eligibility	At least 2-3 yrs programming experience
Contents	•What is an API ? •Common API design problems •How to avoid common problems ? •API design considerations •Discussion of API +ve/-ve of API that participants have used
Type	Classroom
Recommended Batch Size	15-25
Duration	4 hours
Sticker Price	*Email or call for the details*

Programs for Practitioners ..

	Effective Code Review
Eligibility	At least 2-3 yrs programming experience
Contents	Defect discovery by systematic analysis of code • Analyzing assumptions, • Dependencies, • Complexity • Code smells etc
Type	Classroom
Recommended Batch Size	15-25
Duration	4 hours
Sticker Price	*Email or call for the details*

Programs for experienced developers

Thinking Craftsman

Programs for Experienced Developers

	Design Patterns	Performance Optimization
Eligibility	At least 2-3 yrs programming experience. ‘SOLID Design principles’ program	At least 2-3 yrs programming experience
Contents	Commonly used design patterns • Creational patterns (factory patterns, singleton) • Structural Patterns (proxy, composite) • Behavioral Patterns(iterator, command, etc). • Patterns in CAA frameworks	• Code profiling • Interpreting profiler results, • Various strategies for performance optimization
Type	Classroom	Classroom
Recommended Batch Size	15 – 25	15 - 25
Duration	2 to 3 Days	1 days
Sticker Price	*Email or call for the details*	*Email or call for the details*

Programs for Experienced Developers ..

	Code Cleanup & Refactoring
Eligibility	At least 2-3 yrs programming experience. ‘Effective Code Review program’, ‘Design Patterns’ program
Contents	• What is Refactoring • When to refactor ? • Typical refactoring steps. • Refactoring methods/functions • Refactoring class members • Refactoring conditional logic • Refactoring class hierarchy
Type	Classroom
Recommended Batch Size	15 – 25
Duration	1 Days
Sticker Price	*Email or call for the details*

Programs for Experienced Developers

	Design Workshop/Assessment
Eligibility	All design related learning programs
Contents	• Design problem will be given to participants. • Participants will be divided in teams of 2-3. • Participants have to develop the design for the problem and then have to present and **defend** their design. • The participants design skills be assessed on various design aspects considered, flexibility, cohesiveness, patterns used, intentional and unintentional violation of SOLID principles etc
Type	Classroom
Recommended Batch Size	15 - 20
Duration	1 To 3 day
Sticker Price	This is a customized program. Pricing will be decided on depending on number of participants and number of days required.

Soft Skills and Other Misc Programs

Soft Skills and Other Misc. Programs

	Mindmapping	Agile Basics
Eligibility	Anyone can do this program (Software Developers, QA Engineers, Manager, IT, Admin, HR etc)	Anyone can do this program
Contents	• What is mindmaping ? • What can you do with mindmaping ? • Hands on exercises on using mindmaping for planning, brainstorming, idea generation, decision making, etc etc • Group mindmaping.	• What is the problem that we are trying to solve • Doing Agile vs Being Agile • Agile Manifesto (Values and Principles) • Few Agile insights (from manufacturing)
Type	Classroom + Hands on exercises	Classroom
Recommended Batch Size	15 – 25	15 - 25
Duration	4 hours	4 hours
Sticker Price	*Email or call for the details*	*Email or call for the details*

Soft Skills and Other Misc. Programs

	Understanding Kanban	Effective Checklist Workshop
Eligibility	Anyone can do this program	Anyone can do this program
Contents	• What is the problem that we are trying to solve • History of Kanban • Kanban for Knowledge work • Work practices (what is and what is not Kanban) • Designing a Kanban board • Kanban metrics • Common mistakes	• Why some checklists work and some don’t • WHO Safe Surgery Checklist • Characteristics of Good and Bad checklists • How to make effective checklist ? • How checklists make you smarter ?
Type	Classroom	Classroom + Hands on exercises
Recommended Batch Size	15 – 25	15-25
Duration	4 hours	4 to 6 hours
Sticker Price	*Email or call for the details*	*Email or call for the details*

Knowledge Sessions

Knowledge Sessions

Knowledge sessions are an hour long, short introductory sessions on various topics.

Knowledge Session gives an overview of the topic

Knowledge Sessions try to broaden the horizon of participants

List of Knowledge Sessions

- Hidden algorithms of Internet (Error Correction code, TCP handshake, Exponential backoff)
- Life of a Google query
- Costliest bugs in the history of computing
- Cache is everywhere (including in your kitchen)
- Unreasonable effectiveness of Graph algorithms
- Object Oriented Containers- Applying Principles of OO to design of Docker containers
- Modern API design
- Merkle Trees – Data structure behind block chain and Git
- GUI design patterns – ModelView, MVC, MVP, MVVM

- Amateur Programmer to Professional Software Developer
- Characteristics of a Great Software Developer
- Daily Habits of Highly Effective Developers
- Psychology of Successful Agile
- Algorithms of Brain
- “Do, Doing, Done” is not Kanban
- Protecting Yourself and Others from Cyber Crimes
- Tech Warriors: Collaborating with Pune Police During COVID-19"

Contact Nitin

Thinking Craftsman

Contact

Nitin Bhide

http://thinkingcraftsman.in

Email : nitinbhide@thinkingcraftsman.in
Mobile : +91 98220 34694

Blog

LinkedIn