Mohammad Belal Hossain

Teaching Profile

Asia/Dhaka (UTC+6) · Open to remote teaching engagements

Email: md.b.hossain@gmail.com (mailto:md.b.hossain@gmail.com)
LinkedIn: linkedin.com/in/tomdbelalh
Professional Knowledge Hub: *[URL to be confirmed at deployment]*

Teaching Philosophy

Core principle: Judgment transferred, not just answers.

My approach to teaching is based on the belief that clarity is part of professional duty — not a luxury added after success, but part of how success happens. This commitment has guided me through periods of career volatility, including times when my own financial stability wasn't guaranteed.

Fundamentals first: I teach fundamentals before frameworks. Understanding why a solution works matters more than memorizing how to implement it. This approach creates developers who can adapt to new contexts, not just repeat patterns they've seen.

Documentation as teaching: Technical documentation isn't separate from teaching — it's teaching in written form. The same clarity that helps stakeholders understand architecture decisions helps developers learn from them.

Living examples: Abstract concepts are best taught through living examples. WINWINSP and GrowSharely aren't just products — they're teaching material that demonstrates architecture decisions, foundation-first delivery, and documentation practices in real contexts.

No shortcuts: I don't teach exam dumps or interview tricks. I teach the skills that actually matter for long-term careers: clear communication, documented decisions, verifiable quality, and professional reliability.

Subjects I Teach

1. Software Architecture & Design

Topics covered: Foundation-first delivery — why and how. Architecture decision records (ADRs) — templates and governance. Domain modelling for business systems. Business-rules-driven design. SaaS architecture patterns. Legacy system analysis and stabilisation. Multi-project context and RBAC.

Evidence: Seven accepted ADRs for GrowSharely. WINWINSP architecture methodology. 150+ projects including architecture work.

Teaching formats: 6-week structured course. 2-day intensive workshop. 1-on-1 architecture coaching.

2. Laravel & PHP Development

Topics covered: Laravel beyond CRUD (operational business systems). Domain modelling in Laravel. Testing with PHPUnit (409 tests in reference product). Foundation-first delivery in Laravel. Multi-project contexts. Legacy Laravel codebase stabilisation.

Evidence: GrowSharely built in Laravel. 409 tests (1,508 assertions). Laravel expertise across multiple projects.

Teaching formats: 8-week structured course. 3-day hands-on workshop. Code review and mentoring.

3. Technical Documentation

Topics covered: Multi-layer documentation (business, architecture, implementation). Architecture decision records — writing and governance. Handbook documentation standards. Documentation that stakeholders understand. Living documentation alongside code. Documentation as quality assurance.

Evidence: WINWINSP documentation template. Multi-layer documentation for GrowSharely. Six published engineering lessons.

Teaching formats: 4-week structured course. 2-day workshop with templates. 1-on-1 documentation coaching.

4. Legacy Code Recovery

Topics covered: Forensic analysis of inherited codebases. Stabilisation before modernisation. Risk assessment and migration planning. Rebuilding stakeholder trust. Technical debt management. Recovery roadmaps.

Evidence: Project recovery work in 150+ project portfolio. Legacy stabilisation expertise. WINWINSP lessons from real delivery.

Teaching formats: 6-week structured course. 2-day workshop with case studies. 1-on-1 recovery consulting.

5. International Remote Delivery

Topics covered: Written communication as quality assurance. Cross-timezone collaboration. Documentation-driven development. Remote client relationship management. Expectations and cultural awareness. Reliability and trust building.

Evidence: International clients (Australia, Canada, Singapore, UK). Remote delivery methodology. WINWINSP documentation practices.

Teaching formats: 4-week structured course. 1-day workshop. 1-on-1 coaching.

Who I Mentor

Early-Career Developers (0-2 years experience)

Focus areas: Fundamentals first — why before how. Professional communication skills. Documentation habits. Testing and quality practices. Career navigation.

Teaching approach: Structured exercises with clear explanations. Code review with teaching comments. Gradual responsibility increase. Emphasis on long-term skill building.

Evidence: Mentored developers throughout career. Sustained commitment through career volatility. Clarity in explanation treated as professional duty.

Mid-Career Developers (3-7 years experience)

Focus areas: Architecture decision-making. Legacy codebase navigation. Technical documentation. Client communication. Portfolio development.

Teaching approach: Case study analysis. Architecture review sessions. Documentation template sharing. Career strategy discussions.

Evidence: 150+ projects across career stages. Architecture and delivery leadership. Documented engineering practice.

Senior Developers & Architects (8+ years experience)

Focus areas: Architecture practice refinement. Documentation methodology. Legacy system recovery planning. Engineering leadership. Professional portfolio development.

Teaching approach: Peer consultation. Architecture collaboration. Documentation review. Knowledge sharing.

Evidence: WINWINSP methodology creation. Engineering handbook authorship. Architecture decision record practice.

Engineering Teams & Companies

Focus areas: Architecture decision record adoption. Documentation standards. Legacy stabilisation planning. Testing discipline. Remote delivery practices.

Teaching approach: Team workshops. Architecture reviews. Documentation template customization. Ongoing consulting.

Evidence: WINWINSP reusable methodology. Documentation templates. International delivery experience.

Teaching Formats

1-on-1 Mentoring

Best for: Individual skill development. Career guidance. Personalized feedback. Portfolio development.

Format: Weekly 60-minute sessions. Code review between sessions. Progress tracking. Flexible curriculum based on goals.

Pricing: Sliding scale based on ability to pay. Pro bono slots available for demonstrated commitment. Corporate sponsorship encouraged.

Corporate Training

Best for: Team skill development. Architecture practices. Documentation standards. Legacy codebase planning.

Format: 2-4 hour sessions. Customised to team context. Follow-up resources provided. Optional ongoing consulting.

Topics: Foundation-first delivery. Architecture decision records. Technical documentation. Legacy stabilisation. Remote collaboration.

Pricing: Based on team size and duration. Custom quotes for multi-session engagements. Follow-up mentoring discounts.

Online Courses

Best for: Self-directed learners. International participants. Flexible scheduling. Budget-conscious learners.

Format: 6-8 week structured courses. Weekly video lessons. Assignments with feedback. Community discussion. Certificate of completion.

Proposed courses: Foundation-First Software Architecture (6 weeks). Architecture Decision Records in Practice (4 weeks). Technical Documentation That Works (4 weeks). Legacy Code Recovery (6 weeks).

Pricing: Accessible pricing for international markets. Early-bird discounts. Group discounts. Scholarship slots available.

Architecture Coaching

Best for: Senior developers transitioning to architecture. Teams adopting architecture practices. Companies planning legacy system recovery.

Format: 90-minute coaching sessions. Architecture document review. Decision-making support. Ongoing consulting available.

Focus: Real project challenges. Practical decision frameworks. Documentation standards. Stakeholder communication.

Pricing: Per-session or package rates. Corporate sponsorship available. Pro bono for open-source projects.

Developer Mentoring (Roadmap Planning)

Best for: Career navigation. Skill development planning. Portfolio building. Job transition preparation.

Format: Initial 90-minute assessment. 60-minute follow-up sessions. Personalized development plans. Resource recommendations.

Focus: Strengths and gaps analysis. Career path planning. Skill development priorities. Portfolio and documentation review.

Pricing: Sliding scale based on ability. Pro bono for demonstrated commitment. Package rates for ongoing coaching.

Teaching Materials

Available resources: WINWINSP documentation template. Architecture decision record templates. Engineering lessons (six published). GrowSharely as living example. Technical documentation standards.

Under development: Foundation-First Architecture course. ADR adoption workbook. Legacy recovery field guide. Remote delivery handbook.

Evidence of Teaching Commitment

Documented commitment: Sustained mentoring throughout career (including periods of personal financial instability). WINWINSP documentation as teaching material. Six engineering lessons published for reuse. Clarity in explanation treated as professional duty.

Portfolio evidence: 150+ projects with documented decisions. Architecture decision records with teaching value. Multi-layer documentation for learning. Living examples from real delivery.

Contact for Teaching Engagements

Email: md.b.hossain@gmail.com

LinkedIn: linkedin.com/in/tomdbelalh

Professional Knowledge Hub: [URL to be confirmed at deployment]

When inquiring, please include:

- Teaching format (1-on-1, corporate, online course)
- Subject area (architecture, Laravel, documentation, etc.)
- Audience level (early-career, mid-career, senior, team)
- Timeline and availability
- Budget range (or request sliding scale consideration)

Pro bono policy: Limited pro bono slots available for demonstrated commitment. Priority for developers in early-career phases. Corporate sponsorship encouraged for sponsored slots. Sliding scale based on ability to pay.

Version 1 — 15 July 2026. Professional teaching profile based on verified portfolio information and real delivery experience.