"How to Build Highly Scalable Backend Microservices with AI?" - Complete Cohort Design

I'm absolutely THRILLED to design this comprehensive cohort with you! This is going to be an incredible learning journey that transforms participants from microservices beginners to AI-powered architecture experts. Let me break down this ambitious 8-week program that will have your learners building production-ready, scalable systems by the end.

🎯 Cohort Overview & Structure

Duration: 8 weeks

Format: Passive → 4x Recorded → Active → 2x Live + Recorded → Final Live Celebration

Target Audience: Backend developers, system architects, DevOps engineers

Prerequisites: Basic understanding of APIs, databases, and cloud concepts

Class Size: 25-30 minimum participants for optimal interaction

---

📚 Complete Session-by-Session Breakdown

Week 1: LIVE SESSION - Foundations & Architecture Patterns

"Setting the Foundation for Scalable Microservices"

Learning Objectives:

- Understand microservices architecture principles vs monoliths
- Learn domain-driven design (DDD) for service boundaries
- Master the single responsibility principle for microservices
- Explore real-world case studies of successful implementations

Live Session Content (2 hours):

1. **Opening & Introductions** (20 min)
2. **Microservices Evolution & Why They Matter** (30 min)
3. **Domain-Driven Design Workshop** (40 min) - Interactive service boundary mapping
4. **Design Patterns Deep Dive** (25 min) - API Gateway, Service Registry, Circuit Breaker
5. **Q&A & Week 2 Preview** (5 min)

Learning Materials:

- **Video Library:** "Microservices Design Patterns" (45 min compilation)
- **Reading:** Custom eBook "From Monolith to Microservices" (Chapter 1-3)
- **Interactive Diagrams:** Service boundary decision trees
- **Case Study:** Netflix's microservices transformation

Assignment 1: "Design Your Domain"

- Map a real-world application (e-commerce, social media, or banking) into microservices
- Identify 5-7 service boundaries using DDD principles
- Create service interaction diagrams
- **Deliverable:** Architecture diagram + justification document (500 words)
- **Due:** End of Week 2

Resources Provided:

- Lucidchart templates for architecture diagrams
- Discord server for peer collaboration

- Office hours booking link for 1:1 mentoring

Week 2: RECORDED - Data Management & Database Strategies

"Database-Per-Service Pattern & Data Consistency"

Learning Objectives:

- Implement database-per-service pattern effectively
- Handle distributed transactions with SAGA pattern
- Master event-driven data synchronization
- Design for eventual consistency

Recorded Content (90 minutes total):

1. **Database Isolation Principles** (25 min)
2. **SAGA Pattern Implementation** (30 min) - Orchestration vs Choreography
3. **Event Sourcing & CQRS** (25 min)
4. **Practical Demo:** Building a distributed order system (10 min)

Learning Materials:

- **Hands-on Labs:** 4 interactive coding exercises with auto-grading
- **Code Repository:** Sample implementations in Node.js, Python, and Java
- **Tool Access:** Docker Compose environments for local development
- **Reference Guide:** Data consistency patterns cheat sheet

Assignment 2: "Build Your First Microservice"

- Create a User Management service with proper database isolation
- Implement event publishing for user registration/updates
- Add basic CRUD operations with validation
- **Deliverable:** GitHub repository with documentation
- **Due:** End of Week 3

Bonus Challenges:

- Add integration tests with Testcontainers
- Implement soft deletion patterns
- Create API versioning strategy

Week 3: RECORDED - API Design & Communication Patterns

"RESTful APIs, GraphQL, and Inter-Service Communication"

Learning Objectives:

- Design RESTful APIs following OpenAPI specifications
- Implement GraphQL for complex data requirements
- Master synchronous vs asynchronous communication
- Build robust error handling and retries

Recorded Content (105 minutes total):

1. **API Design Best Practices** (20 min)
2. **OpenAPI/Swagger Deep Dive** (25 min)
3. **GraphQL Federation** (30 min)
4. **Message Queues & Event Streaming** (20 min)
5. **Error Handling Strategies** (10 min)

Learning Materials:

- **Interactive API Designer:** Build APIs visually
- **GraphQL Playground:** Hands-on query building
- **Message Queue Simulator:** RabbitMQ and Kafka environments
- **Postman Collections:** Pre-built API test suites

Assignment 3: "Multi-Service Communication"

- Extend Week 2's service to include Product Catalog service

- Implement both REST and GraphQL endpoints
- Add message-based communication for inventory updates
- **Deliverable:** Two connected services with comprehensive API docs
- **Due:** End of Week 4

Peer Review Component:

- Review 2 other participants' API designs
- Provide constructive feedback using provided rubric
- Receive feedback on your own implementation

Week 4: RECORDED - Security, Authentication & Authorization

"Zero-Trust Security in Distributed Systems"

Learning Objectives:

- Implement JWT-based authentication at scale
- Design role-based access control (RBAC) systems
- Master service-to-service security with mTLS
- Build API gateway security layers

Recorded Content (110 minutes total):

1. **Authentication Patterns in Microservices** (25 min)
2. **JWT Implementation & Best Practices** (20 min)
3. **mTLS for Service-to-Service Security** (25 min)
4. **API Gateway Security Features** (20 min)
5. **Security Testing & Vulnerability Scanning** (20 min)

Learning Materials:

- **Security Toolkit:** Pre-configured auth services and policies
- **Hands-on Labs:** OAuth2 flow implementation

- **Vulnerability Scanner:** Integrated SAST/DAST tools
- **Security Checklist:** Production-ready security audit guide

Assignment 4: "Secure Your Microservices"

- Add comprehensive authentication to previous services
- Implement RBAC with multiple user roles
- Set up service-to-service security
- **Deliverable:** Security documentation + penetration test results
- **Due:** End of Week 5

Expert Guest Session (Optional 30-min):

- Security expert from a Fortune 500 company
- Real-world breach case studies and prevention

Week 5: LIVE SESSION - AI Integration & Perplexity API

"Adding Intelligence to Your Microservices"

Learning Objectives:

- Integrate AI services into microservices architecture
- Master Perplexity AI API for real-time information
- Design AI-driven decision-making services
- Handle AI service scaling and error scenarios

Live Session Content (2.5 hours):

1. **AI in Microservices Overview** (20 min)
2. **Perplexity API Deep Dive** (40 min) - Live coding demo
3. **Building an AI-Powered Recommendation Service** (60 min) - Interactive workshop
4. **Scaling AI Workloads** (20 min)
5. **Error Handling for AI Services** (20 min)

6. **Q&A & Live Troubleshooting** (10 min)

Learning Materials:

- **Perplexity API Cookbook:** 20+ practical examples
- **AI Integration Patterns Guide:** Comprehensive reference
- **Model Selection Framework:** Decision trees for AI service choices
- **Monitoring AI Services:** Custom dashboards and alerts

Assignment 5: "Build an Intelligent Service"

- Create a Customer Support AI service using Perplexity API
- Implement smart routing based on query analysis
- Add fallback mechanisms for AI failures
- **Deliverable:** Production-ready AI microservice with monitoring
- **Due:** End of Week 6

Live Coding Challenge:

- 45-minute build challenge during the session
- Create a real-time sentiment analysis service
- Winners get bonus learning credits

Week 6: RECORDED - Monitoring, Logging & Observability

"Full-Stack Observability for Production Systems"

Learning Objectives:

- Implement distributed tracing across services
- Build comprehensive logging strategies
- Create actionable monitoring dashboards
- Set up intelligent alerting systems

Recorded Content (115 minutes total):

1. **The Three Pillars: Metrics, Logs, Traces** (25 min)
2. **Implementing Distributed Tracing** (30 min)
3. **Prometheus & Grafana Setup** (25 min)
4. **ELK Stack for Centralized Logging** (25 min)
5. **Alert Management & On-Call Best Practices** (10 min)

Learning Materials:

- **Observability Stack:** Pre-configured Prometheus + Grafana + Jaeger
- **Log Analysis Tools:** Kibana dashboards and saved searches
- **Alerting Playbooks:** Runbook templates for common scenarios
- **Performance Baseline:** Benchmarking tools and methodologies

Assignment 6: "Instrument Your Services"

- Add comprehensive observability to all previous services
- Create custom Grafana dashboards
- Implement distributed tracing with correlation IDs
- **Deliverable:** Observability demo video (5 minutes) + dashboard exports
- **Due:** End of Week 7

Troubleshooting Simulation:

- Participants debug pre-broken services using observability tools
- Gamified experience with leaderboards

Week 7: RECORDED - Performance Optimization & Caching

"Building for Scale: Caching, CDNs, and Performance"

Learning Objectives:

- Implement multi-level caching strategies
- Design for horizontal scaling patterns

- Optimize database queries and API performance
- Handle traffic spikes with auto-scaling

Recorded Content (120 minutes total):

1. **Caching Strategies Deep Dive** (30 min)
2. **Redis for Distributed Caching** (25 min)
3. **Database Optimization Techniques** (25 min)
4. **Load Balancing & Auto-Scaling** (25 min)
5. **Performance Testing & Benchmarking** (15 min)

Learning Materials:

- **Performance Testing Suite:** JMeter scripts and load generators
- **Caching Laboratory:** Redis cluster setup with various patterns
- **Optimization Toolkit:** Database query analyzers and profilers
- **Scaling Simulators:** Traffic spike simulation environments

Assignment 7: "Optimize for Scale"

- Implement comprehensive caching across your service ecosystem
- Conduct load testing and identify bottlenecks
- Optimize based on results and document improvements
- **Deliverable:** Performance analysis report + optimized codebase
- **Due:** End of Week 8

Performance Competition:

- Teams compete to optimize a standard microservices application
- Winner gets recognition and additional resources

Week 8: RECORDED - AWS Deployment & DevOps Pipeline

"Production Deployment with AWS ECS/EKS and CI/CD"

Learning Objectives:

- Deploy microservices to AWS using ECS and EKS
- Build complete CI/CD pipelines
- Implement blue-green deployment strategies
- Set up production monitoring and alerting

Recorded Content (130 minutes total):

1. **AWS Container Services Overview** (20 min) - ECS vs EKS vs Fargate
2. **Containerization Best Practices** (25 min)
3. **CI/CD Pipeline Construction** (35 min) - GitHub Actions + AWS CodePipeline
4. **Blue-Green Deployments** (25 min)
5. **Infrastructure as Code** (25 min) - Terraform and CloudFormation

Learning Materials:

- **AWS Sandbox Accounts:** Individual practice environments
- **Infrastructure Templates:** Terraform modules for common patterns
- **CI/CD Pipeline Library:** GitHub Actions workflows and scripts
- **Deployment Checklists:** Production readiness validation guides

Assignment 8: "Deploy to Production"

- Deploy your complete microservices ecosystem to AWS
- Implement full CI/CD pipeline with automated testing
- Set up production monitoring and alerts
- **Deliverable:** Live demo URL + infrastructure code + deployment documentation
- **Due:** Final Live Session (Week 9)

Capstone Project Integration:

- All previous assignments culminate in a complete, production-ready system
- Peer evaluation of final deployments

Week 9: FINAL LIVE CELEBRATION - Demo Day & Certification

"Showcase Your Microservices Mastery"

Celebration Session (3 hours):

1. **Welcome & Cohort Achievements** (15 min)
2. **Participant Demo Presentations** (90 min) - 5 min each + Q&A
3. **Industry Expert Panel** (45 min) - Q&A with hiring managers
4. **Networking Session** (20 min) - Virtual breakout rooms
5. **Certification Ceremony** (15 min)
6. **Alumni Network Introduction & Next Steps** (15 min)

Demo Requirements:

- 5-minute presentation of capstone project
- Live demonstration of key features
- Architecture walkthrough with lessons learned
- Q&A from peers and experts

Certification Criteria:

- Complete all 8 assignments (minimum 80% quality score)
- Participate in at least 6 weeks of activities
- Successfully deploy final project to production
- Peer evaluation score of 4.0+ (out of 5.0)

📋 Assessment & Grading Framework

Assignment Scoring Rubric (100 points each):

- **Technical Implementation** (40 points): Code quality, architecture decisions, best practices
- **Documentation** (25 points): Clear README, API docs, deployment guides

- **Innovation** (20 points): Creative solutions, going beyond requirements
- **Peer Collaboration** (15 points): Code reviews, forum participation, helping others

Certification Requirements:

- **Minimum Overall Score:** 640/800 points (80%)
- **No Single Assignment Below:** 60 points (75%)
- **Participation Score:** 90%+ (forum posts, peer reviews, live session attendance)
- **Final Project Demo:** Successfully deployed and demonstrated

Recognition Levels:

- 🎖️ **Microservices Master** (90%+): Top 20% of cohort
- 🏅 **Architecture Expert** (85%+): Strong technical execution
- 🎓 **Certified Graduate** (80%+): Successfully completed all requirements

---

🛠️ Technology Stack & Tools Provided

Development Environment:

- **Cloud IDE:** GitPod workspaces for each participant
- **Container Registry:** Private Docker Hub repos
- **Code Repository:** GitHub organization with template repos
- **Communication:** Dedicated Discord workspace with channels for each requirement

AWS Resources (Per Participant):

- **Sandbox Account:** Free credits which AWS provides for new account creation
- **ECS/EKS Clusters:** Pre-configured for deployments
- **RDS Instances:** PostgreSQL and MongoDB options
- **ElastiCache:** Redis clusters for caching exercises
- **CloudWatch:** Monitoring and alerting setup

Development Tools:

- **API Testing:** Postman Pro team license
- **Monitoring Stack:** Prometheus, Grafana, Jaeger hosted instances
- **CI/CD:** GitHub Actions credits + AWS CodePipeline access
- **Security:** SAST/DAST scanning tools integrated

AI & Integration Services:

- **Perplexity API Credits:** What Perplexity Pro provides, so try to get Perplexity pro even for just the 2 months of the cohort
- **OpenAI API Access:** For comparison and experimentation
- **Message Queues:** Confluent Cloud Kafka clusters
- **Databases:** MongoDB Atlas, PostgreSQL on AWS RDS

📚 Comprehensive Learning Materials Package

Custom eBook Series (400+ pages total):

1. **"Microservices Architecture Fundamentals"** - Foundations and patterns
2. **"AI-Powered Backend Services"** - Integration strategies and best practices
3. **"Production Deployment Guide"** - AWS deployment and DevOps practices
4. **"Security & Performance Playbook"** - Advanced optimization techniques

Video Library (50+ hours):

- **Recorded Sessions:** All sessions available for review
- **Bonus Content:** Industry expert interviews and case studies
- **Quick Reference:** 5-minute "how-to" videos for common tasks
- **Troubleshooting Guides:** Step-by-step problem-solving videos

Interactive Labs & Simulations:

- **Architecture Simulator:** Visual microservices design tool
- **Performance Tester:** Load testing environments
- **Security Scanner:** Vulnerability assessment tools
- **Deployment Simulator:** Practice deployments without AWS costs

Code Repositories & Templates:

- **Starter Templates:** 10+ microservice boilerplates in multiple languages
- **Reference Implementations:** Complete example applications
- **Security Templates:** Auth services and security middleware
- **Infrastructure Code:** Terraform and Helm charts for common patterns

---

🎯 Expected Learning Outcomes

By Week 4, participants will:

- Design and implement secure microservices with proper database isolation
- Create comprehensive API documentation and testing suites
- Understand and apply security best practices including mTLS and JWT
- Build services that communicate effectively using multiple patterns

By Week 8, participants will:

- Deploy production-ready microservices to AWS with full observability
- Integrate AI capabilities using Perplexity API and other services
- Implement comprehensive caching and performance optimization
- Set up complete CI/CD pipelines with automated testing and deployment

Final Capstone Capabilities:

- **End-to-End System:** Complete microservices application with 5-7 services
- **AI Integration:** Intelligent features using multiple AI APIs

- **Production Deployment:** Live system running on AWS with proper monitoring
- **Professional Portfolio:** GitHub repos and documentation suitable for job applications

---

🤝 Community & Support Structure

Mentorship Program:

- **Lead Instructor:** Available for 2 office hours per week
- **Teaching Assistants:** 3 TAs for technical support and code reviews
- **Industry Mentors:** 5 senior engineers from major tech companies for career guidance
- **Peer Mentoring:** Buddy system pairing experienced with newer developers

Support Channels:

- #general: Main discussion and announcements
- #technical-help: Q&A and troubleshooting
- #code-reviews: Peer review and feedback
- #career-advice: Job search and professional development
- #showcase: Share wins and project updates

Weekly Check-ins:

- **Monday:** Week kickoff and goal setting
- **Wednesday:** Mid-week progress check and Q&A
- **Friday:** Week wrap-up and peer sharing

🏆 Post-Cohort Value & Alumni Network

Immediate Benefits:

- **Verified Certificate:** LinkedIn-displayable credential with skill validation

- **GitHub Portfolio:** 8 production-ready projects demonstrating expertise
- **Industry Connections:** Network with 30+ peers and 5+ industry mentors
- **Job Placement Support:** Resume reviews, interview prep, and job referrals

Ongoing Alumni Benefits:

- **Alumni Discord:** Permanent access to graduate network (500+ members)
- **Monthly Tech Talks:** Industry updates and advanced topics
- **Job Board Access:** Exclusive opportunities from partner companies
- **Continued Learning:** 20% discount on future advanced cohorts

Career Advancement Tracking:

- **6-Month Follow-up:** Career progression and salary increase surveys
- **Success Stories:** Alumni spotlight featuring promotions and new roles
- **Referral Program:** Alumni earn rewards for successful referrals

---

💰 Investment & Value Proposition

Cohort Pricing Structure:

- **Standard Enrollment:** $6.99
- **Early Bird (4 weeks prior):** $4.99 (28.61% savings)
- **Student Discount:** $3.99 with valid .edu email
- **Group Discounts:** 15% off for 3+ enrollments from same company OR group. For that, DM @YourAKShaw OR email hitme@yourakshaw.com.

What's Included (Value: $8,000+):

- **50+ Hours Premium Content:** Live sessions + recorded materials
- **AWS Practice Credits:** $200 per participant ($6,000 total value)
- **API Access:** Perplexity AI and other premium service credits ($300 value)

- **Development Tools:** Postman Pro, GitPod, monitoring tools ($500+ value)
- **Expert Access:** 1:1 mentoring and industry connections (Priceless)
- **Career Support:** Resume reviews, interview prep, job referrals ($1,200+ value)

🚀 Pre-Cohort Preparation & Onboarding

2 Weeks Before Start:

- **Welcome Package:** Digital handbook, schedule, and technology setup guides
- **Environment Setup:** AWS account creation, tool installations, access provisioning
- **Prerequisite Assessment:** Skills check and personalized prep recommendations
- **Cohort Introductions:** Discord server access and icebreaker activities

1 Week Before Start:

- **Technical Bootcamp:** Optional 2-hour session covering prerequisites
- **Mentor Matching:** Assignment of industry mentors based on career goals
- **Learning Path Customization:** Adjust content based on experience levels
- **Final Tech Check:** Ensure all tools and environments are working

Day of First Session:

- **30-Min Pre-Session:** Final Q&A and technical troubleshooting
- **Cohort Kickoff:** Official welcome and expectation setting
- **Success Planning:** Individual goal setting with assigned mentor

---

This cohort represents the cutting edge of backend development education, combining practical microservices expertise with AI integration skills that are in massive demand. Every element is designed to transform participants into highly skilled, industry-ready professionals who can architect and build the scalable systems that power today's most successful companies.

The structured progression from foundational concepts through advanced deployment ensures that every participant, regardless of starting skill level, emerges with production-ready capabilities and a portfolio that showcases real expertise. The combination of live interaction, hands-on practice, peer collaboration, and industry mentorship creates an unparalleled learning experience that delivers lasting career transformation.

Ready to launch this incredible journey? 🚀