

Struktur Backend Sistem Pembelajaran SAT dan UTBK Berbasis Agentic AI

1. Arsitektur Layanan Mikro (Microservices)

- Auth Service: OAuth2, JWT dengan refresh token dan revocation.
- User Service: Manajemen profil, RBAC.
- Data Service: Pemisahan PostgreSQL Data Service untuk soal, user, aktivitas.
- Vector DB Service: Retrieval embedding dan pencarian semantic (Qdrant/Chroma/Milvus).
- Embedding Service: Transformasi teks ke vector menggunakan InstructorXL / text-embedding-3-small.
- Agent Orchestrator Service: Orkestrasi Planner, Generator, Evaluator dalam pipeline Agentic AI.
- Question Service: Bank soal, generator adaptif, caching hasil.
- Assessment Service: Manajemen sesi ujian, adaptive scoring.
- Analytics Service: Logging aktivitas, analitik performa, dashboard.
- Notification Service: Notifikasi email/real-time.
- CMS/Admin Panel Service: UI dan API untuk upload materi, edit soal, manajemen konten.
- Message Broker Service: Komunikasi event-driven yang terpusat. (Kafka/RabbitMQ/NATS)

2. Desain API Gateway dan Komunikasi Antar Layanan

- API Gateway (FastAPI / gRPC gateway) menangani routing permintaan, rate limiting, dan autentikasi.
- Komunikasi sinkron (gRPC/REST) untuk layanan inti.
- Asinkron (Event Bus): RabbitMQ/Kafka/NATS untuk tugas event-driven (pembuatan soal baru, feedback, indexing).
- Service Discovery: Consul atau Kubernetes DNS untuk registrasi dan health checks.
- Feature Flags: Pengelolaan rilis bertahap pada tingkat API.

3. Database dan Penyimpanan

- PostgreSQL: Relasional untuk data pengguna, sesi, konfigurasi.
- Qdrant/Chroma/Milvus: Vector database untuk semantic search.
- Redis: Caching embedding dan metadata sesi.
- Blob Storage (S3-compatible): Penyimpanan materi PDF dan aset.
- Data Encryption at Rest: AES-256 TDE pada semua penyimpanan.

4. Orkestrasi dan Containerization

- Docker: Kontainer untuk setiap microservice.
- Kubernetes (Helm charts): Deployment, service discovery, auto-scaling.

- Consul/Eureka: Registry dan health checks.
- CI/CD Integration: Pipeline otomatis build dan deploy.

5. Keamanan Backend

- TLS mutual antara microservices.
- WAF & API Rate Limiting di API Gateway.
- JWT dengan refresh token dan revocation list.
- RBAC untuk membatasi akses API sesuai peran.
- Audit Trail untuk pencatatan perubahan dan akses.
- Data Masking untuk sensitif data masking di log.

6. CI/CD dan DevOps

- Pipeline GitHub Actions: linting, unit/integration tests, image build.
- Docker Registry (GitHub Container Registry).
- Deployment otomatis ke staging dan production via Helm.
- Rollback otomatis jika health checks gagal.

7. Monitoring dan Logging

- Metrics: Prometheus + Grafana untuk pemantauan kinerja.
- Tracing: Jaeger/OpenTelemetry untuk distribusi trace.
- Logging: ELK Stack (Elasticsearch, Logstash, Kibana) untuk log terpusat.
- Alerts: Integrasi Alertmanager untuk notifikasi insiden.

8. Strategi Testing dan Skalabilitas

- Load Testing: Menggunakan k6, Locust, atau JMeter untuk 10k TPS.
- Stress Testing: Skenario spike dan endurance.
- Chaos Engineering: Simulasi kegagalan layanan untuk resiliency.
- Horizontal Scaling: Kubernetes HPA berdasarkan metrik.