

# Rencana Judul Topik Tugas Akhir Bimbingan Ardiles Sinaga

## Tujuan Tiap Judul (dengan konteks migrasi dari monolith)

No	Judul	Fokus Riset setelah migrasi dari monolith
1	Refactoring Arsitektur Aplikasi TAPATUPA Menjadi Multi-Kabupaten Berbasis Web	Menganalisis dan memecah modul <i>kabupaten, asset, sewa</i> dari monolith menjadi service <i>independen</i> yang mendukung multi-tenant (multi kabupaten). Fokus pada <i>domain-driven design</i> dan pemisahan data.
2	Implementasi Arsitektur Database Master-Slave untuk Optimalisasi Kinerja Sistem TAPATUPA Terdistribusi	Mengonversi koneksi database tunggal di monolith menjadi sistem replikasi (MySQL master-slave). Menguji performa query read/write antar service (misalnya, Sewa dan Aset).
3	Integrasi Layanan Pembayaran Online Bank Sumut pada Sistem TAPATUPA Berbasis Microservice	Memisahkan modul pembayaran dari monolith menjadi service <i>independen</i> ( <i>service-payment</i> ) dan menghubungkannya melalui API dengan <i>service-sewa</i> . Mengimplementasikan callback dan sinkronisasi status pembayaran.
4	Penerapan API Gateway dan Keamanan Antar Layanan Menggunakan JWT pada Sistem TAPATUPA	Membangun <i>API Gateway</i> sebagai layer utama yang menghubungkan seluruh service hasil refactoring (Auth, Sewa, Pembayaran, Aset). Menambahkan <i>JWT + internal service token</i> untuk keamanan dan otorisasi.

## Peta Pengembangan (Dari Monolith ke Microservice)

Tahap 0: Aplikasi TAPATUPA Monolith (Laravel)

```
├─ Modul Auth  
├─ Modul Aset  
├─ Modul Sewa  
├─ Modul Pembayaran  
└─ 1 Database tunggal (MySQL)
```

Tahap 1: Refactoring

```
├─ Service Auth  
├─ Service Aset  
├─ Service Sewa  
└─ Service Payment
```

```

    |--- API Gateway
    |   |
    |   --- Database: master-slave (MySQL)

```

Setiap kelompok akan mengekstrak 1 modul utama dari sistem monolith dan mengembangkan service-nya sendiri.

#### **Timeline Implementasi (Mis: 1 Semester / 14 Minggu)**

Minggu	Kegiatan	Mahasiswa Terkait
1–2	Analisis sistem monolith, identifikasi domain dan batas modul	Semua
3–5	Refactoring modul (Aset, Sewa, Auth, Pembayaran) jadi service terpisah	1, 3
4–6	Setup MySQL master-slave dan migrasi database	2
6–9	Integrasi antar service melalui API Gateway internal	4
8–11	Integrasi pembayaran online (Bank Sumut)	3
10–12	Implementasi JWT dan token antar service	4
12–14	Testing integrasi sistem & analisis performa	Semua

#### **Manfaat Ilmiah dan Teknis**

Bidang	Inovasi Penelitian
Arsitektur Sistem	Migrasi sistem pemerintahan dari monolith ke microservice berbasis open-source (Laravel).
Database	Optimisasi query dan skalabilitas dengan replikasi master-slave.
Integrasi Layanan Keuangan	Model transaksi daring resmi (Bank Sumut) untuk penyewaan aset daerah.
Keamanan Sistem	Lapisan autentikasi JWT dan komunikasi antar-service yang aman.

#### **Output Akhir**

- Sistem TAPATUPA v2 (berbasis microservice, dockerized).
- 4 laporan penelitian per kelompok (masing-masing fokus sesuai judul).
- 1 laporan integrasi & dokumentasi sistem gabungan.
- Demo aplikasi end-to-end:
  - Login (Auth)
  - Sewa aset (Sewa + Aset)
  - Bayar (Payment)
  - Akses publik lewat Gateway (JWT-secured).

**Hal-hal yang perlu diukur pada masing-masing penelitian:**

**1. Judul 1 – Refactoring Arsitektur Aplikasi TAPATUPA Menjadi Multi-Kabupaten Berbasis Web**

**Fokus:** Transformasi dari sistem **monolith** menjadi **multi-service** dengan dukungan multi-kabupaten (multi-tenant).

**Aspek yang Diukur:**

Aspek	Indikator Pengukuran	Alat / Metode
Kinerja Sistem	- Waktu respon per permintaan sebelum & sesudah refactoring.- Beban CPU/memori server per 100 request.	Apache Benchmark (ab), Postman Monitor, Grafana.
Skalabilitas	- Kemampuan menambah kabupaten baru tanpa downtime.- Lama waktu deployment service baru (menit).	Docker Compose test, log waktu deploy.
Struktur Arsitektur	- Jumlah coupling antar modul (dihitung dari dependensi kode).- Jumlah service independen yang terbentuk.	Static Code Analysis (PHPStan, SonarQube).
Maintainability	- Waktu yang dibutuhkan untuk mendeploy update minor (per modul).	Catatan pengujian manual.

**Output Analisis:**

- Perbandingan monolith vs microservice (response time, CPU usage).
- Diagram dependency sebelum & sesudah refactor.

**2. Judul 2 – Implementasi Arsitektur Database Master-Slave untuk Optimalisasi Kinerja Sistem TAPATUPA Terdistribusi**

**Fokus:** Penerapan **replikasi MySQL master-slave** agar beban baca/tulis terbagi dan konsistensi data terjaga.

**Aspek yang Diukur:**

Aspek	Indikator Pengukuran	Alat / Metode
Performa Query	- Rata-rata waktu query SELECT di master vs slave.- Latency replikasi (delay waktu update tersalin ke slave).	MySQL Performance Schema, SHOW SLAVE STATUS.
Throughput Sistem	- Jumlah transaksi/menit (TPS) sebelum dan sesudah replikasi.	JMeter / Apache Benchmark.

Konsistensi Data	- Jumlah ketidaksesuaian data antara master dan slave (opsional).	SQL diff check script.
Ketersediaan (Availability)	- Waktu sistem tetap aktif saat master down (failover time).	Simulasi downtime.

#### Output Analysis:

- Grafik waktu query read/write.
- Analisis kecepatan sinkronisasi data antar node.
- Studi banding sebelum–sesudah replikasi.

### 3. Judul 3 – Integrasi Layanan Pembayaran Online Bank Sumut pada Sistem TAPATUPA Berbasis Microservice

**Fokus:** Pemisahan modul **pembayaran** menjadi *service-payment* dan integrasi API Bank Sumut (QRIS/Virtual Account).

#### Aspek yang Diukur:

Aspek	Indikator Pengukuran	Alat / Metode
Keandalan Transaksi	- Persentase transaksi sukses vs gagal.- Waktu respons API Bank Sumut vs sistem internal.	Log service-payment, curl latency test.
Sinkronisasi Data	- Akurasi status pembayaran antara sistem dan bank (callback vs database).	Audit log per transaksi.
Keamanan Transaksi	- Validasi signature/token pada webhook callback.- Hasil pengujian endpoint tanpa autentikasi.	OWASP ZAP / Postman Security Test.
Kepuasan Pengguna (opsional)	- Skor kepuasan pengguna internal (admin/pemda) terhadap otomatisasi pembayaran.	Kuesioner (Likert scale).

#### Output Analysis:

- Grafik latency integrasi (ms).
- Statistik transaksi berhasil/gagal.
- Dokumentasi callback & signature verification.

### 4. Judul 4 – Penerapan API Gateway dan Keamanan Antar Layanan Menggunakan JWT pada Sistem TAPATUPA

**Fokus:** Membangun *API Gateway* sebagai *central entry point* dengan autentikasi **JWT** untuk pengguna dan **service token** antar microservice.

### **Aspek yang Diukur:**

<b>Aspek</b>	<b>Indikator Pengukuran</b>	<b>Alat / Metode</b>
Keamanan	- Jumlah serangan (request tidak sah) yang berhasil ditolak.- Hasil uji penetrasi API (Unauthorized Access).	OWASP ZAP, Postman Security Collection.
Performa Gateway	- Overhead waktu respon akibat validasi JWT (ms).- Throughput request per detik (RPS).	JMeter, Grafana metrics.
Reliabilitas	- Rata-rata uptime gateway selama pengujian beban.	Uptime Monitor.
Autentikasi Antar Service	- Keberhasilan pertukaran data antar-service dengan token internal yang valid.	Log antar service (Auth ↔ Sewa ↔ Payment).

### **Output Analisis:**

- Grafik waktu respon API dengan/ tanpa JWT.
- Log validasi antar-service.
- Analisis tingkat keamanan endpoint.

### **Hubungan Indikator Antar Penelitian**

<b>Hubungan</b>	<b>Keterangan</b>
1 → 2	Hasil refactoring mempengaruhi desain database master-slave (struktur & koneksi service).
2 → 3	Replikasi database mempengaruhi transaksi pembayaran (kecepatan update status pembayaran).
3 → 4	Integrasi Bank Sumut memerlukan gateway aman untuk callback dan validasi token.
4 → 1	Gateway dan JWT menjadi fondasi utama agar sistem multi-kabupaten tetap aman dan terpusat.