

# POLITEKNIK NEGERI BANYUWANGI

## Laporan Final Project: Cine-Booking App

Mobile Application Development  
Semester Ganjil 2025/2026 [cite: 4]

Disusun Oleh Kelompok 07:

Dino Febiyan	NIM001	Backend Engineer & Data Seeder.
Cheryl Aurellya Bangun Jaya	NIM002	Frontend Engineer (Home Module).
Rusydi Jabir Al Awfa	362458302044	Frontend Engineer (Seat Matrix Module).
Mohammad Faisal	NIM004	Logic Controller (The Brain).
Semua		QA Lead, Auth & Profile

## BAB 1

### Pendahuluan & Pembagian Kerja

#### 1.1 Deskripsi Aplikasi

Aplikasi Smart E-Kantin ini dibangun menggunakan Flutter dengan arsitektur yang memisahkan logic, UI, dan backend services.

#### 1.2 Tabel Pembagian Tanggung Jawab

Berikut adalah detail pembagian tugas sesuai instruksi ujian:

#### 1.3 Tabel Pembagian Tanggung Jawab (sementara)

Berikut adalah ringkasan pembagian tugas (sementara) yang dapat diperbaiki nanti:

Anggota 1: Backend — Firebase Setup, Data Seeding — (inisial: ani)

Anggota 2: UI Engineer — Slicing Login & Home — (inisial: budi)

Anggota 3: Auth & Nav — Login Logic & Routing — (inisial: citra)

Anggota 4: Logic — Cart State & Discount — (inisial: dani)

Anggota 5: QA Lead — Testing & Merge — (inisial: eka)

## BAB 2

### Bukti Keaslian Kode (Strict Mode)

Sesuai instruksi ujian untuk mencegah penggunaan code generator otomatis[cite: 8].

#### 2.1 Watermark Code

Berikut adalah bukti penggunaan suffix inisial pada variabel Widget dan Fungsi:

Penjelasan: Variabel `buttonLogin_budi` menunjukkan kode dibuat oleh Budi (UI Engineer) [cite: 9].

#### 2.2 Logic Trap (Diskon NIM)

Implementasi logika bisnis unik “Ganjil 5%, Genap Gratis Ongkir”[cite: 10].

```
// Contoh snippet kode (Logic Trap)
void hitungDiskon_dani(String nim) {
    int lastDigit = int.parse(nim.characters.last);
    if (lastDigit % 2 != 0) {
        // Logika Ganjil
        print("Diskon 5%");
    } else {
        // Logika Genap
        print("Gratis Ongkir");
    }
}
```

## BAB 3

### Arsitektur Backend

Dikerjakan oleh Backend Architect[cite: 23].

#### 3.1 Struktur Database (Firestore)

Menggunakan 3 Collection utama: `Users`, `Products`, dan `Transactions`[cite: 13, 15, 17].

#### 3.2 Data Seeding

Bukti seeding minimal 10 produk dummy ke Firebase[cite: 27].

## BAB 4

### Implementasi Antarmuka & Auth

Dikerjakan oleh UI Engineer dan Auth Controller[cite: 28, 33].

#### 4.1 Halaman Login & Register

Validasi form diterapkan: Password minimal 6 karakter dan Email wajib domain kampus[cite: 36].

#### 4.2 Navigasi

Menggunakan `Navigator.pushReplacement` untuk routing antar halaman[cite: 34].

## BAB 5

### State Management & Logic

Dikerjakan oleh Transaction Logic Specialist[cite: 37].

#### 5.1 State Keranjang

State management menggunakan (Provider/Bloc/GetX) untuk fitur Add to Cart dan Remove from Cart[cite: 38, 39].

#### 5.2 Perhitungan Checkout

Total harga dihitung berdasarkan logika diskon NIM dan stok di Firebase berkurang otomatis (Transaction Write)[cite: 41].

## BAB 6

### Integrasi & Git Branching

Dikerjakan oleh QA Lead[cite: 42].

#### 6.1 Strategi Branching

Struktur branch yang digunakan sesuai instruksi[cite: 49, 50, 51, 52, 53]:

- **feature/backend-setup**
- **feature/ui-widgets**
- **feature/auth-nav**
- **feature/cart-state**
- **feature/testing-docs**

## BAB 7

### Pengujian & Penutup

#### 7.1 Handling Data

Bukti fitur `ListView.builder` tidak error saat data kosong dan implementasi `LoadingIndicator`[cite: 44, 45].

#### 7.2 Link Video Demo

Berikut adalah link video demo aplikasi yang diunggah ke Google Drive/YouTube:

- Link: [Masukkan Link Di Sini]