**Laporan Praktikum: Implementasi Material Design Components (MDC) dengan Flutter**

**MATA KULIAH VISUAL PROGRAMMING**

**MATERI PRAKTIKUM 8**

****

**Disusun Oleh :**

**Muhammad Aditya Ridwan 0806022310026**

**Rabu, 20 November 2024**

**BAB I**

**Pendahuluan**

**1.1 Latar Belakang**

Flutter adalah framework pengembangan aplikasi lintas platform yang populer karena kemampuannya membangun antarmuka modern dengan performa tinggi. Material Design Components (MDC) adalah panduan dari Google yang dirancang untuk menciptakan antarmuka yang konsisten dan estetis. Dalam codelabs MDC-101 hingga MDC-104, pengembang diajarkan cara menggunakan MDC dalam membangun aplikasi Shrine, sebuah aplikasi toko online dengan desain responsif, interaktif, dan modern.

**1.2 Tujuan**

1. Memahami dan mengimplementasikan konsep Material Design Components dalam Flutter.
2. Mempelajari pengaturan tema, tipografi, tata letak, responsivitas, dan animasi.
3. Membuat aplikasi Shrine sebagai hasil akhir dengan prinsip Material Design.

**BAB II**

**Alat dan Bahan**

**2.1 Alat**

1. Laptop/PC dengan spesifikasi mendukung pengembangan Flutter.
2. Aplikasi editor kode seperti Visual Studio Code atau Android Studio.
3. Emulator Android/iOS atau perangkat fisik untuk testing.

**2.2 Bahan**

1. **Software Development Kit (SDK)**: Flutter SDK versi terbaru.
2. **Library Tambahan**:
   * google\_fonts: Untuk menambahkan font Material Design.
   * lottie: Untuk implementasi animasi interaktif.
3. **File Codelabs**: File proyek dari [MDC-101 hingga MDC-104](https://codelabs.developers.google.com/codelabs/mdc-101-flutter).
4. **Resource Gambar dan Animasi**: Gambar produk dan file animasi JSON.

**BAB III**

**Tahap Pengerjaan**

**3.1 MDC-101: Menambahkan Tema dan Tipografi**

1. **Menentukan Tema Warna**
   * Membuat tema warna utama (primary), sekunder (secondary), dan warna latar belakang aplikasi.
   * Menggunakan ColorScheme dari Flutter untuk konsistensi warna di seluruh aplikasi.
2. **Menyesuaikan Tipografi**
   * Menambahkan font khusus menggunakan pustaka google\_fonts.
   * Menyesuaikan ukuran, berat, dan gaya teks sesuai dengan panduan Material Design.
3. **Mengatur Tema Global**
   * Menerapkan tema dan tipografi yang telah ditentukan secara global melalui ThemeData.

**3.2 MDC-102: Membuat Tata Letak**

1. **Merancang Struktur Halaman Login**
   * Menggunakan widget seperti TextField untuk input data pengguna.
   * Menambahkan tombol login dengan gaya Material Design.
2. **Menyusun Tampilan Produk dengan Grid**
   * Membuat tata letak daftar produk menggunakan GridView untuk tampilan rapi dan simetris.
   * Menentukan jumlah kolom dalam grid berdasarkan ukuran layar.
3. **Menggunakan Card dan Icon**
   * Menampilkan produk dalam komponen Card untuk menonjolkan elemen desain.
   * Menambahkan ikon untuk tombol aksi seperti menambah ke keranjang.

**3.3 MDC-103: Menyesuaikan Responsivitas**

1. **Menambahkan Media Query**
   * Menentukan ukuran grid secara dinamis berdasarkan ukuran layar menggunakan MediaQuery.
   * Menyesuaikan elemen UI seperti margin dan padding agar terlihat rapi di berbagai perangkat.
2. **Menerapkan Breakpoints**
   * Membuat tata letak adaptif dengan menentukan breakpoint untuk ukuran perangkat kecil, sedang, dan besar.

**3.4 MDC-104: Animasi dan Interaksi**

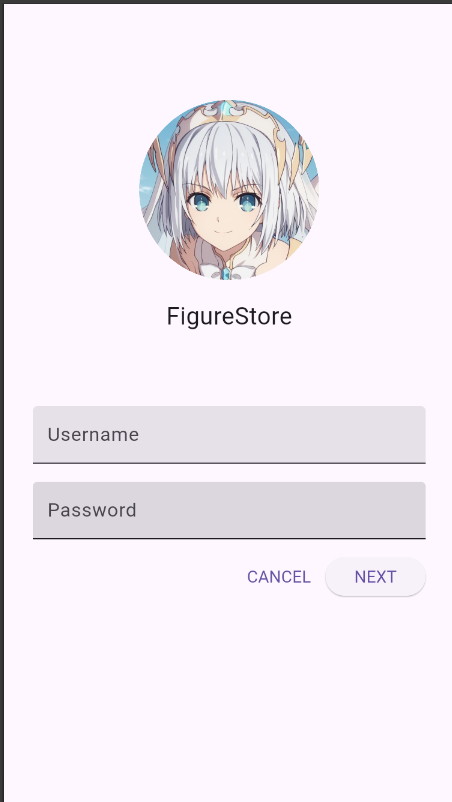
1. **Menambahkan Animasi Interaktif**
   * Menggunakan pustaka lottie untuk menampilkan animasi onboarding yang menarik.
   * Memastikan animasi berjalan dengan lancar di semua perangkat.
2. **Meningkatkan Interaksi Pengguna**
   * Menambahkan snackbar untuk memberikan umpan balik setelah pengguna melakukan aksi seperti menekan tombol.
   * Menambahkan navigasi antar halaman menggunakan Navigator.
3. **Melakukan Penyesuaian Akhir**
   * Memastikan transisi antar halaman berjalan mulus dengan menambahkan animasi sederhana.
   * Mengoptimalkan pengalaman pengguna dengan menyesuaikan waktu loading dan respon UI.

**BAB IV**

**Hasil dan Pembahasan**

**A.Component Basic**

**Gambar 4.1 Halaman Login**

****

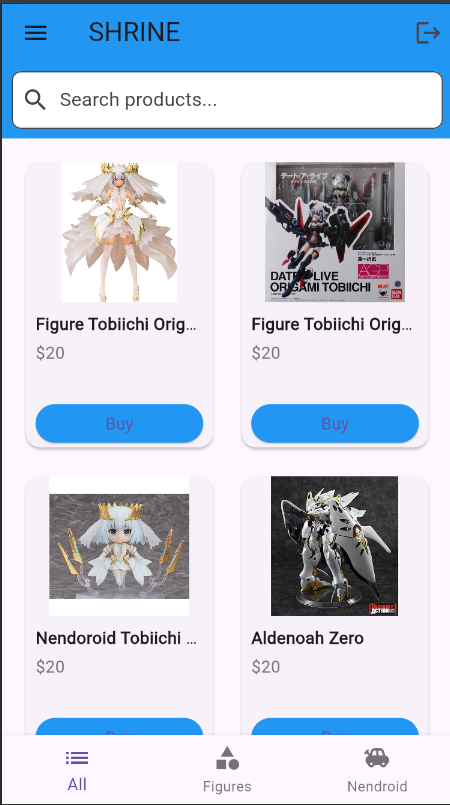
Pada tahap ini, halaman *login* dengan tata letak sederhana berhasil dibuat, seperti yang ditampilkan pada Gambar 4.1. Halaman ini menggunakan widget CircleAvatar untuk menambahkan gambar profil di bagian atas. Di bawahnya, terdapat teks nama aplikasi (*FigureStore*) yang menggunakan widget Text dengan tema tipografi Material Design.

Bagian form terdiri atas dua kolom input, yaitu *Username* dan *Password*, yang dibuat menggunakan widget TextFormField. Properti InputDecoration diterapkan untuk menambahkan label teks pada setiap kolom input, sehingga pengguna dapat memahami fungsinya.

Tombol navigasi disusun menggunakan widget TextButton dengan teks *CANCEL* dan widget ElevatedButton dengan teks *NEXT*. Tombol *CANCEL* berfungsi untuk menghapus input yang dimasukkan pada TextFormField, sedangkan tombol *NEXT* berfungsi untuk melanjutkan proses login.

**B. Material Structur**

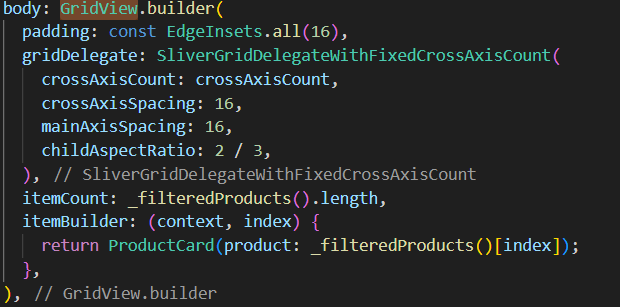
**Gambar 4.1 Halaman Utama**



Pada tahap ini, aplikasi berhasil menampilkan halaman utama dengan struktur Material Design. Tata letak halaman ini menggunakan kombinasi widget seperti AppBar, Drawer, dan BottomNavigationBar.

1. **AppBar**  
   Widget AppBar digunakan di bagian atas halaman untuk menampilkan judul aplikasi. Properti seperti backgroundColor dan elevation diterapkan untuk menyesuaikan warna dan kedalaman bayangan.
2. **Drawer**  
   Widget Drawer ditambahkan di sisi kiri untuk memberikan akses ke menu navigasi. Dalam Drawer, beberapa elemen seperti ListTile digunakan untuk membuat menu dengan ikon dan label teks.
3. **BottomNavigationBar**  
   Widget BottomNavigationBar digunakan untuk navigasi di bagian bawah layar. Tombol navigasi ini memungkinkan pengguna berpindah antar halaman dengan cepat. Setiap tombol dibuat menggunakan properti BottomNavigationBarItem dengan ikon dan label yang relevan.

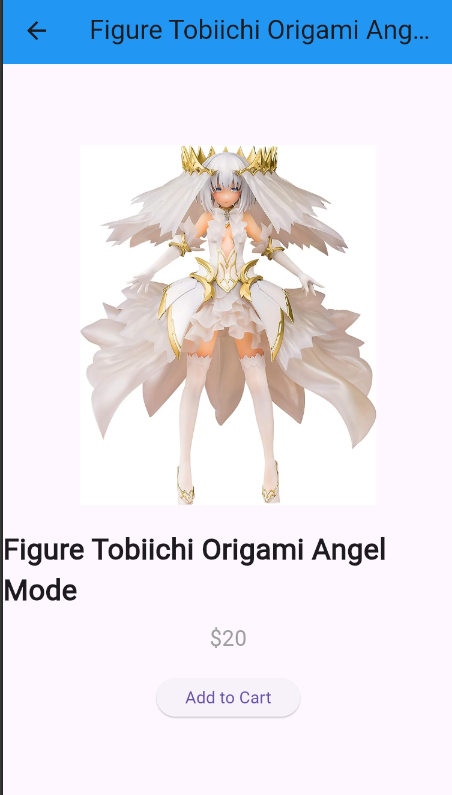
**Gambar 4.2 Halaman Utama**

****

Pada tahap ini, halaman grid berhasil dirancang untuk menampilkan konten dalam format *card* menggunakan widget GridView. Tata letak ini efisien dan responsif terhadap berbagai ukuran layar.

1. **GridView**Widget GridView dengan properti crossAxisCount digunakan untuk menentukan jumlah kolom dalam grid. Elemen grid dirancang dengan spasi yang rapi menggunakan properti mainAxisSpacing dan crossAxisSpacing.
2. **Card**Masing-masing elemen dalam grid dibuat menggunakan widget Card. Widget ini menampilkan gambar produk dengan teks deskripsi di bagian bawah. Properti seperti elevation dan shape diterapkan untuk menambahkan dimensi visual dan membuat setiap *card* lebih menarik.

**Gambar 4.3 Halaman Produk**

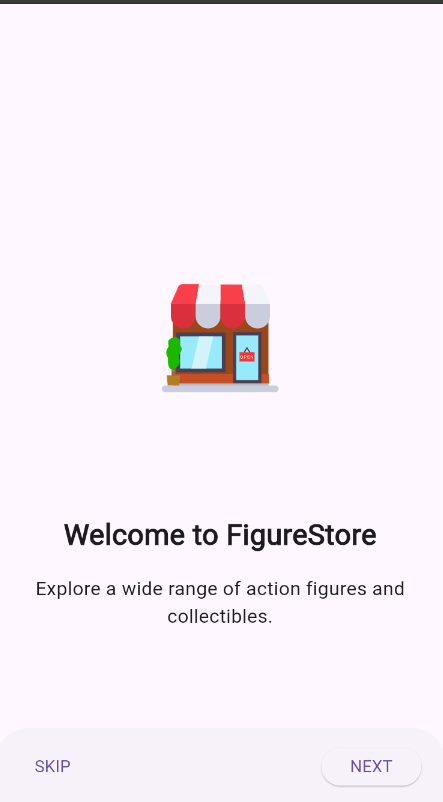
****

Pada tahap ini, halaman detail produk berhasil dibuat, seperti yang ditunjukkan pada Gambar 4.1. Halaman ini menggunakan **widget Scaffold** sebagai struktur utama untuk menyusun elemen-elemen di layar. Elemen-elemen yang digunakan pada halaman ini meliputi:

* **AppBar**  
  Bagian atas halaman menggunakan AppBar yang berfungsi untuk menampilkan judul produk, yaitu "Figure Tobiichi Origami Angel Mode". Warna latar belakang AppBar diatur menggunakan warna biru untuk memberikan tampilan yang konsisten dengan tema aplikasi.
* **Gambar Produk**  
  Bagian utama halaman menampilkan gambar produk menggunakan widget Image. Gambar ini diletakkan di bagian atas halaman dengan ukuran yang disesuaikan untuk memperlihatkan detail produk secara jelas.
* **Informasi Produk**  
  Nama dan harga produk ditampilkan menggunakan widget Text.
  + **Nama Produk** ditampilkan dalam font besar dan tebal untuk menarik perhatian pengguna.
  + **Harga Produk** menggunakan warna abu-abu untuk memberikan kesan profesional dan minimalis.
* **Tombol Aksi**  
  Terdapat tombol "Add to Cart" yang berfungsi untuk menambahkan produk ke keranjang belanja. Saat tombol ditekan, pesan "Item added to cart" akan muncul di bagian bawah layar menggunakan fitur SnackBar.

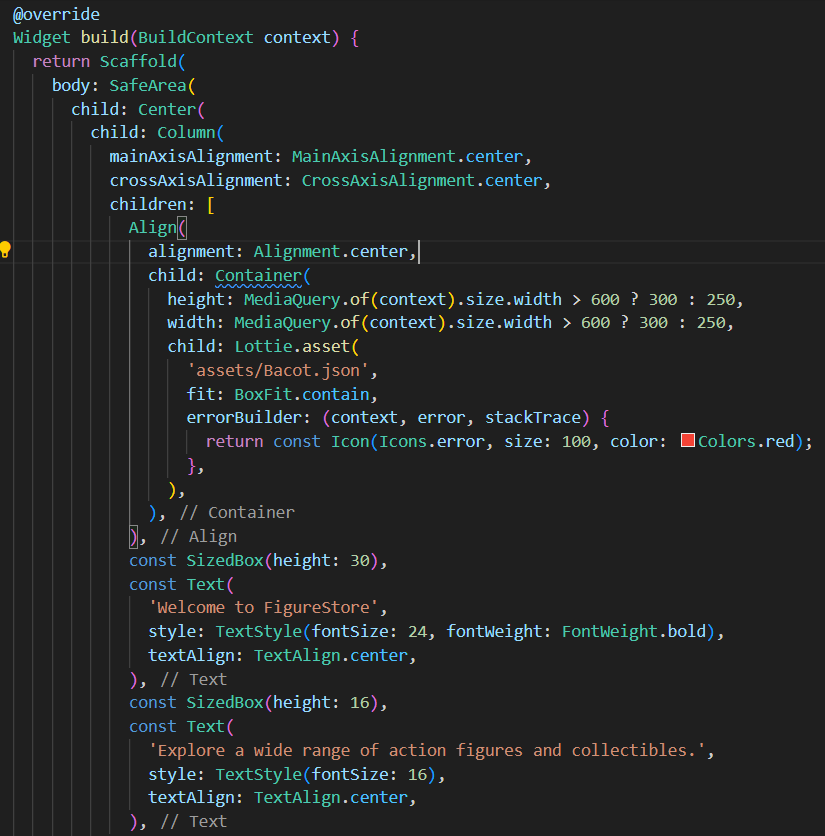
**C. Halaman Onboard**

**Gambar 4.1 Halaman Onboard 1**

****

Pada halaman pertama onboarding, pengguna diperkenalkan dengan aplikasi FigureStore. Halaman ini menampilkan animasi interaktif, teks sambutan, dan desain yang responsif.

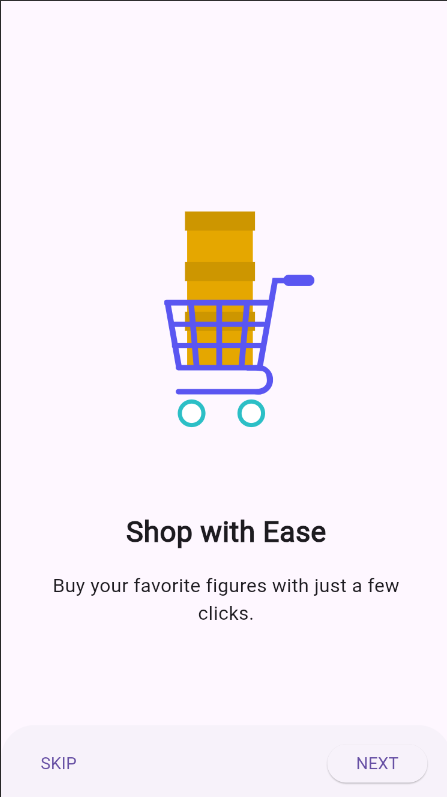
**Kode Halaman Onboard 1**

****

Kode untuk halaman ini menggunakan widget Column untuk menyusun elemen secara vertikal, sementara animasi ditampilkan menggunakan Lottie.asset. Berikut adalah fitur utama pada kode:

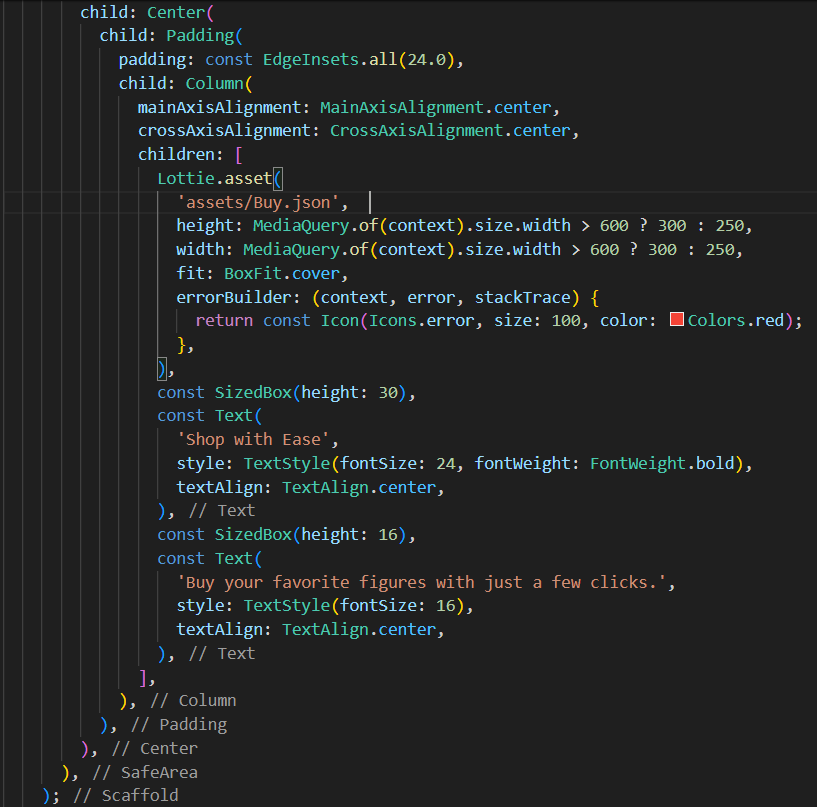
1. Animasi Lottie: Berasal dari file assets/Bacot.json, animasi diatur agar proporsional terhadap layar menggunakan MediaQuery.
2. Teks Sambutan: Ditampilkan menggunakan Text widget dengan ukuran font 24 (judul) dan 16 (deskripsi).
3. Tata Letak Responsif: Menggunakan widget Center untuk memastikan semua elemen berada di tengah layar.

**Gambar 4.3 Halaman Onboard 2**

****

Halaman kedua onboarding berfokus pada kemudahan belanja. Halaman ini menampilkan animasi yang menggambarkan aktivitas belanja dan teks yang menjelaskan fitur.

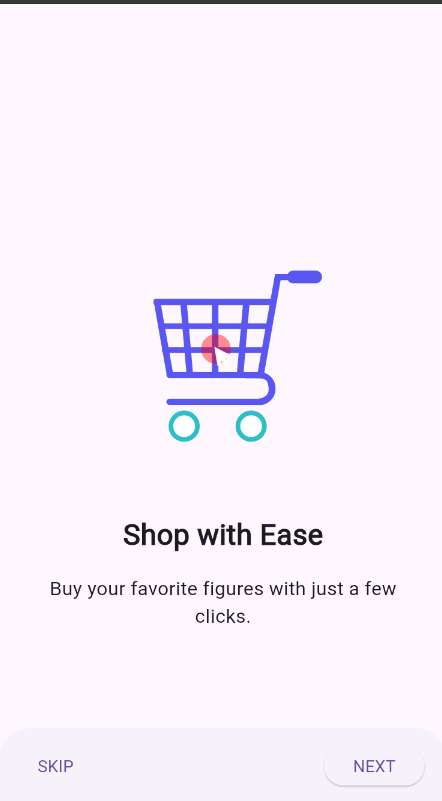
**Kode Halaman Onboard 2**

****

Kode untuk halaman ini menggunakan Padding untuk menjaga jarak antar elemen dan Lottie.asset untuk animasi. Fitur utama pada kode:

1. Animasi Belanja: File animasi dari assets/Buy.json ditampilkan menggunakan widget Lottie.asset.
2. Judul dan Deskripsi: Menggunakan Text widget dengan ukuran font 24 untuk judul dan 16 untuk deskripsi.
3. Penyesuaian Layout: Padding 24 dp digunakan untuk memberikan ruang yang cukup antar elemen.

**Gambar 4.5 Halaman Onboard 3**

****

Halaman onboarding terakhir menonjolkan keamanan dan kecepatan aplikasi. Halaman ini menampilkan animasi tentang keamanan dan teks yang memberikan rasa percaya kepada pengguna.

**Kode Halaman Onboard 3**

****

Kode untuk halaman ini memiliki elemen berikut:

1. Animasi Keamanan: File animasi dari assets/Secure.json ditampilkan dengan widget Lottie.asset.
2. Teks Keamanan: Menggunakan widget Text untuk menampilkan judul dan deskripsi dengan ukuran font yang sama seperti halaman sebelumnya.
3. Responsivitas: Tata letak yang responsif menggunakan MediaQuery untuk mengatur ukuran elemen berdasarkan ukuran layar perangkat.

**BAB V**

**Kesimpulan**

Berdasarkan praktikum yang telah dilakukan, saya berhasil mencapai tujuan yang telah ditetapkan, yaitu untuk memahami dan mengimplementasikan prinsip dasar Material Design dalam pengembangan aplikasi Flutter. Selama lima tahap praktikum, saya telah mempelajari dan berhasil mengimplementasikan berbagai komponen Material Design dengan baik.

Meskipun saya belum sepenuhnya mampu mengklasifikasikan produk yang telah terfilter berdasarkan kategori dalam praktikum kali ini, secara keseluruhan, praktikum ini telah memberikan pemahaman yang lebih mendalam mengenai penerapan prinsip Material Design dalam Flutter. Hal ini tidak hanya memperkaya pengetahuan saya tentang Material Design dalam konteks pengembangan aplikasi berbasis Flutter, tetapi juga meningkatkan keterampilan saya dalam menciptakan aplikasi yang lebih efektif dan efisien.