

Struktur Komponen dan Implementasi Pojok MBKM

File dan Struktur Komponen

Komponen Utama

Berikut adalah struktur komponen dan hubungan antar file yang akan diimplementasikan:

```
src/
├── components/
│   ├── sections/
│   │   └── PojokMBKM.tsx           # Section utama yang ditampilkan di homepage
│   └── pojok-mbkm/
│       ├── JournalCard.tsx        # Card untuk preview jurnal
│       ├── JournalGrid.tsx        # Grid layout untuk koleksi jurnal
│       ├── JournalFilter.tsx      # Filter berdasarkan kategori
│       ├── JournalSearch.tsx      # Komponen pencarian
│       ├── JournalPagination.tsx  # Navigasi halaman
│       ├── JournalDetail.tsx      # Tampilan detail jurnal
│       ├── JournalMedia.tsx       # Galeri media jurnal
│       └── AuthorInfo.tsx         # Info penulis
├── lib/
│   ├── types/
│   │   └── journal.ts             # TypeScript interfaces
│   └── data/
│       └── dummyJournals.ts       # Data dummy untuk testing
└── app/
    ├── page.tsx                   # Halaman utama (termasuk section PojokMBKM)
    └── pojok-mbkm/
        └── [id]/
            └── page.tsx            # Halaman detail jurnal
```

Flow dan Interaksi Komponen

- PojokMBKM.tsx** (Parent Component):
 - Menampilkan judul section dan tagline
 - Mengatur state untuk filter dan pencarian
 - Mengintegrasikan komponen **JournalSearch**, **JournalFilter**, **JournalGrid** dan **JournalPagination**

- Mengelola fetch data jurnal (dari data dummy)
2. **JournalFilter.tsx**:
 - Menampilkan opsi filter berdasarkan kategori
 - Mengirim event ketika filter berubah ke parent component
 3. **JournalSearch.tsx**:
 - Menyediakan input search dengan debounce
 - Mengirim query search ke parent component
 4. **JournalGrid.tsx**:
 - Menerima array jurnal dari parent
 - Menampilkan jurnal dalam grid layout responsif
 - Merender **JournalCard** untuk setiap item jurnal
 5. **JournalCard.tsx**:
 - Menampilkan preview jurnal (judul, ringkasan, penulis, dll)
 - Menyediakan link "Baca Selengkapnya" ke halaman detail
 6. **JournalPagination.tsx**:
 - Menampilkan navigasi halaman
 - Mengirim event perubahan halaman ke parent component
 7. **app/pojok-mbkm/[id]/page.tsx**:
 - Mengambil detail jurnal berdasarkan ID dari URL parameter
 - Merender **JournalDetail** dengan data jurnal
 8. **JournalDetail.tsx**:
 - Menampilkan konten lengkap jurnal
 - Mengintegrasikan **JournalMedia** dan **AuthorInfo**
 9. **JournalMedia.tsx**:
 - Menampilkan galeri foto dalam jurnal
 10. **AuthorInfo.tsx**:
 - Menampilkan info penulis dan biografi singkat

API Data dan State Management

Untuk MVP dengan data dummy:

typescript

```
// Contoh penggunaan di PojokMBKM.tsx
import { useState, useEffect } from 'react';
import { getFilteredJournals } from '@lib/data/dummyJournals';
import { Journal, JournalFilterOptions, PaginationOptions } from '@lib/types/journal';

// State management
const [journals, setJournals] = useState<Journal[]>([]);
const [filterOptions, setFilterOptions] = useState<JournalFilterOptions>({});
const [pagination, setPagination] = useState<PaginationOptions>({ page: 1, limit: 9 });
const [totalJournals, setTotalJournals] = useState(0);

// Fetch data berdasarkan filter dan pagination
useEffect(() => {
  const filteredJournals = getFilteredJournals(
    filterOptions.category,
    filterOptions.searchQuery,
    filterOptions.authorId
  );

  setTotalJournals(filteredJournals.length);

  // Apply pagination
  const start = (pagination.page - 1) * pagination.limit;
  const end = start + pagination.limit;
  setJournals(filteredJournals.slice(start, end));
}, [filterOptions, pagination]);
```

Rencana Implementasi Berkelanjutan

Tahap 1: MVP

- Implementasi komponen dengan data statis/dummy
- Layout dasar dan styling UI
- Fungsionalitas filter kategori

Tahap 2: Enhanced Features

- Implementasi pencarian
- Pagination
- Tampilan detail jurnal
- Responsivitas penuh

Tahap 3: Full Integration

- Integrasi dengan backend/API
- Dark/light mode integration
- Animasi dan transisi
- Loading states dan error handling