

Dokumentasi Projek Pemrograman Web 2

Nama: Afrando Sharein Ramadhan

NPM: 2312105

Kelas: IF 22 C

Nama Team: BBC

Menginstall & Setup Projek NextJS

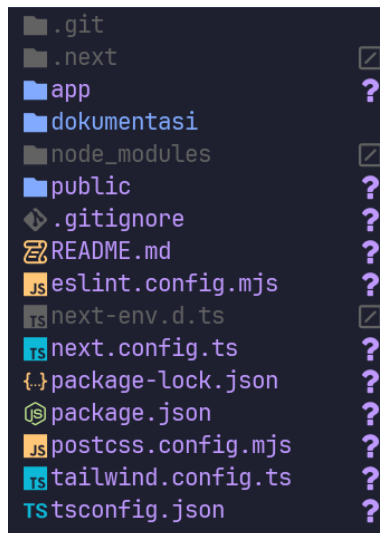
```
@afran → Final Projek npx create-next-app@latest
Need to install the following packages:
create-next-app@15.1.3
Ok to proceed? (y) y

√ What is your project named? ... bbc-askhajaya
√ Would you like to use TypeScript? ... No / Yes
√ Would you like to use ESLint? ... No / Yes
√ Would you like to use Tailwind CSS? ... No / Yes
√ Would you like your code inside a `src/` directory? ... No / Yes
√ Would you like to use App Router? (recommended) ... No / Yes
√ Would you like to use Turbopack for `next dev`? ... No / Yes
√ Would you like to customize the import alias (`@/*` by default)? ... No / Yes
Creating a new Next.js app in D:\Penyimpanan\Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya
.
```

Deskripsi Penjelasan: Pada tahap pertama saya selaku project manager / ketua, membuat & menginstall projek Tim BBC.

```
Enter name for new file or directory (dirs end with a "/" or "\"):
dokumentasi/

README.md
eslint.config.mjs
next.config.ts
package-lock.json
package.json
postcss.config.mjs
tailwind.config.ts
tsconfig.json
(5 hidden items)
```



Deskripsi Penjelasan: Setelah proyek berhasil terinstall, saya membuat folder **dokumentasi** didalam proyek yang sejajar atau satu tingkatan yang sama dengan folder **app** dan **public**.

Folder dokumentasi digunakan untuk menyimpan dokumentasi dari proses pengerjaan proyek yang sedang dikerjakan oleh masing masing anggota tim.

Menginstall & Konfigurasi Dependency Shadcn/UI

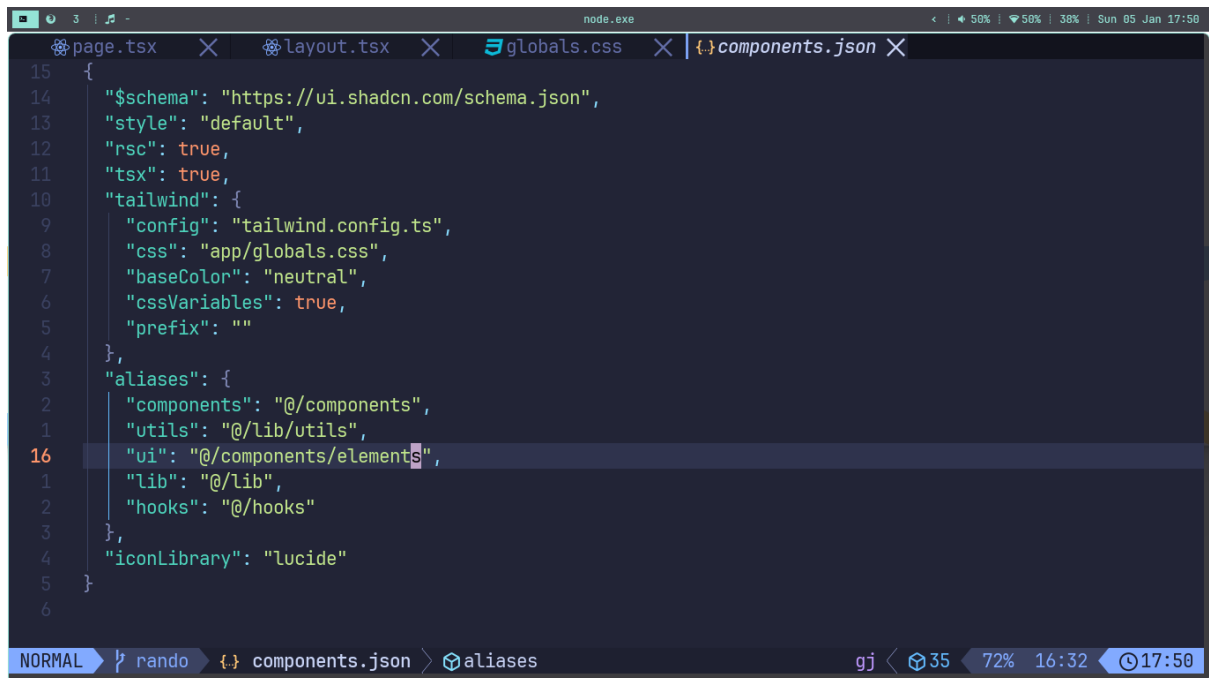
Untuk membantu pengerjaan proyek website pada sisi **frontend** yang sedang kami buat, kami menggunakan **shadcn/ui** dalam pembuatan komponen antarmuka pengguna (UI). Alasan kami memilih **shadcn/ui** berdasarkan beberapa pertimbangan diantaranya: Shadcn/ui berbasis Tailwind CSS, Fleksibilitas yang tinggi sehingga mendapat kontrol penuh untuk kustomisasi, sederhana dan efisien, dan kompatibilitas dengan Framework Modern.

```
D:\Penyimpanan Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya>npx shadcn@latest init
Need to install the following packages:
shadcn@2.1.8
Ok to proceed? (y)
✓ Preflight checks.
✓ Verifying framework. Found Next.js.
✓ Validating Tailwind CSS.
✓ Validating import alias.
```

Deskripsi Penjelasan: Saya melakukan installasi shadcn/ui dengan perintah **npx shadcn@latest init**

Setelah shadcn/ui berhasil terinstall ia akan membuat sebuah file bernama **components.json** yang berisi konfigurasi terkait components.

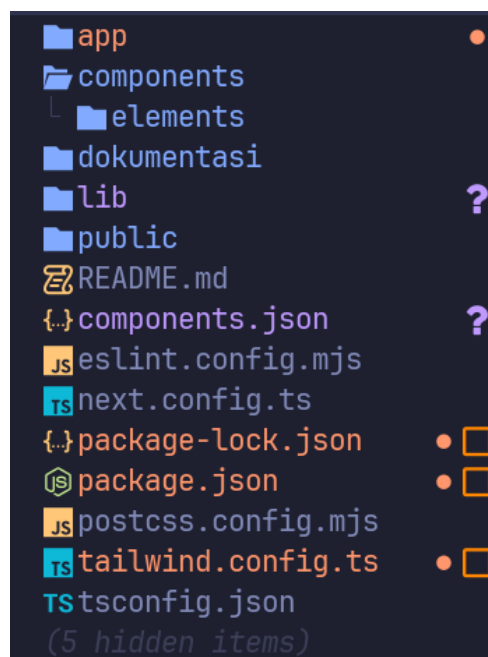




```
15 {
14   "$schema": "https://ui.shadcn.com/schema.json",
13   "style": "default",
12   "rsc": true,
11   "tsx": true,
10   "tailwind": {
9     "config": "tailwind.config.ts",
8     "css": "app/globals.css",
7     "baseColor": "neutral",
6     "cssVariables": true,
5     "prefix": ""
4   },
3   "aliases": {
2     "components": "@components",
1     "utils": "@lib/Utils",
16    "ui": "@components/elements",
1     "lib": "@lib",
2     "hooks": "@hooks"
3   },
4   "iconLibrary": "lucide"
5 }
6
```

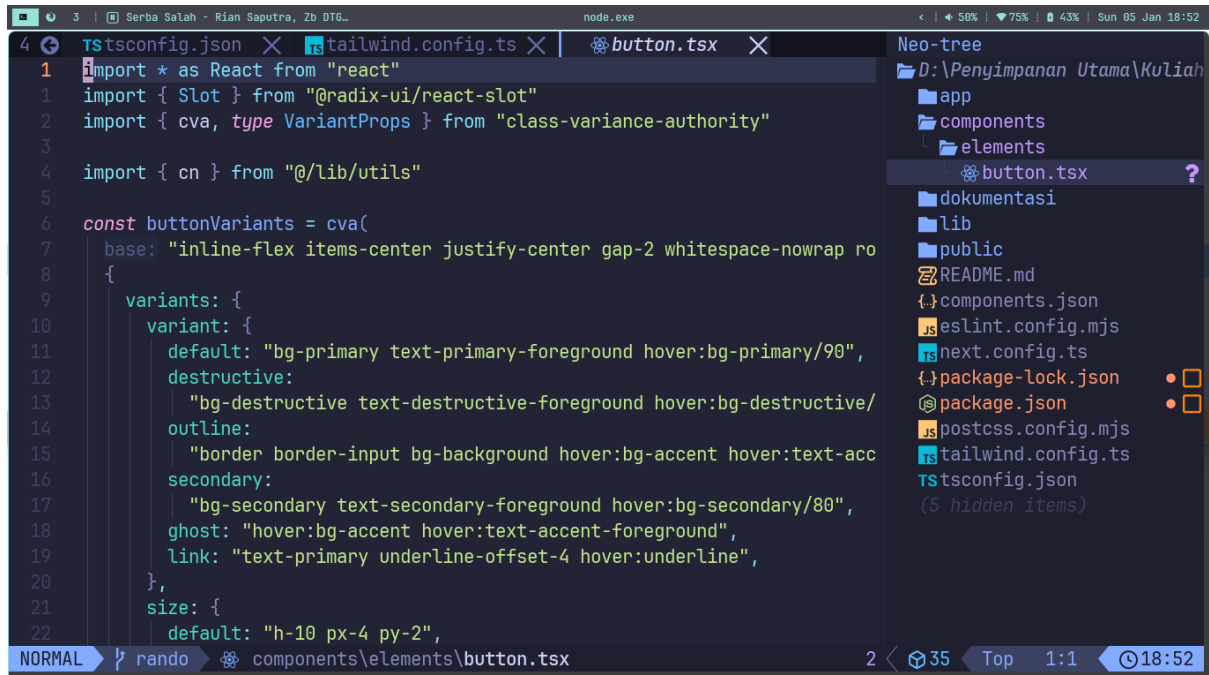
The screenshot shows a code editor with the file `components.json` open. The configuration includes Tailwind CSS settings and aliases for components, utilities, and hooks. The `ui` alias is set to `@components/elements`. The editor's status bar at the bottom shows the current file is `components.json` and the active section is `aliases`.

Deskripsi Penjelasan: Pada file **components.json** saya melakukan konfigurasi agar tiap components yang di buat oleh shadcn/ui akan berada pada folder yang telah saya sesuaikan untuk berisi components yaitu folder **components/elements**.



Deskripsi Penjelasan: Proyek yang kami buat berencana menerapkan metodologi **Atomic Design** sehingga desain sistem UI yang dibuat bersifat modular, konsisten, dan dapat digunakan kembali.

Membuat Komponen Navbar



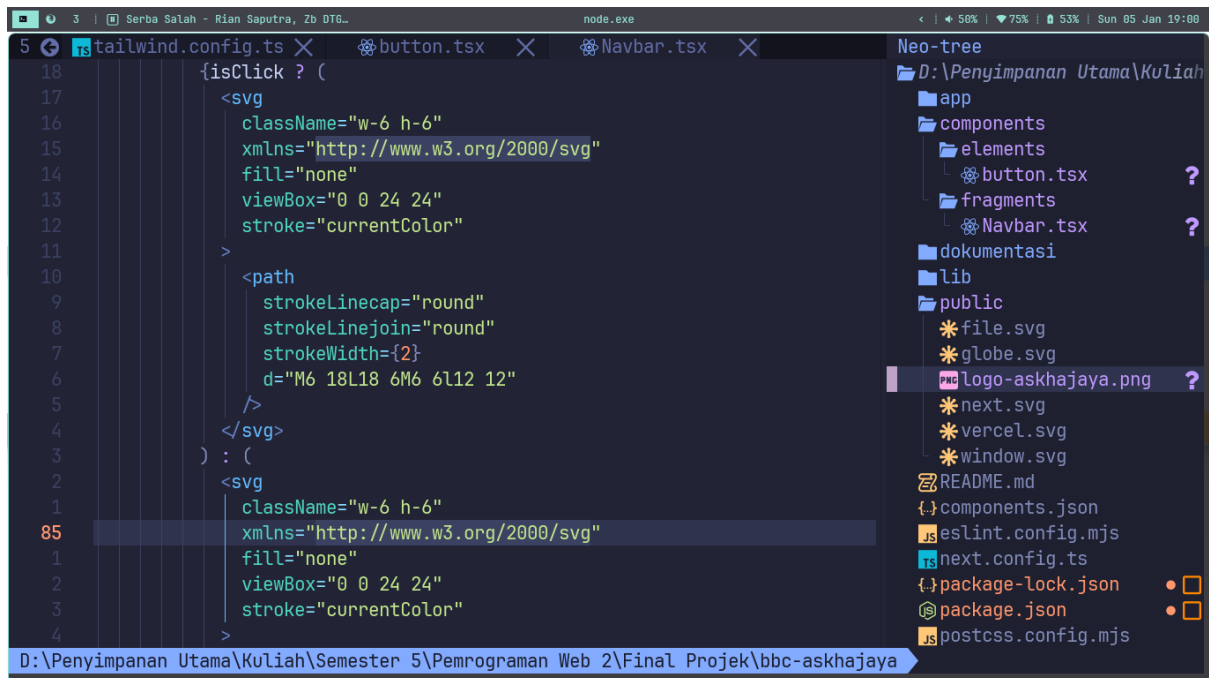
```
1 import * as React from "react"
2 import { Slot } from "@radix-ui/react-slot"
3 import { cva, type VariantProps } from "class-variance-authority"
4
5 import { cn } from "@lib/utils"
6
7 const buttonVariants = cva(
8   base: "inline-flex items-center justify-center gap-2 whitespace-nowrap rounded-
9   {
10     variant: {
11       default: "bg-primary text-primary-foreground hover:bg-primary/90",
12       destructive:
13         "bg-destructive text-destructive-foreground hover:bg-destructive/
14       outline:
15         "border border-input bg-background hover:bg-accent hover:text-ac
16       secondary:
17         "bg-secondary text-secondary-foreground hover:bg-secondary/80",
18       ghost: "hover:bg-accent hover:text-accent-foreground",
19       link: "text-primary underline-offset-4 hover:underline",
20     },
21     size: {
22       default: "h-10 px-4 py-2",
```

Deskripsi Penjelasan: Pada komponen navbar yang akan saya buat terdapat tombol **login**, oleh karena itu saya membuat komponen tombol terlebih dahulu menggunakan **shadcn/ui**.

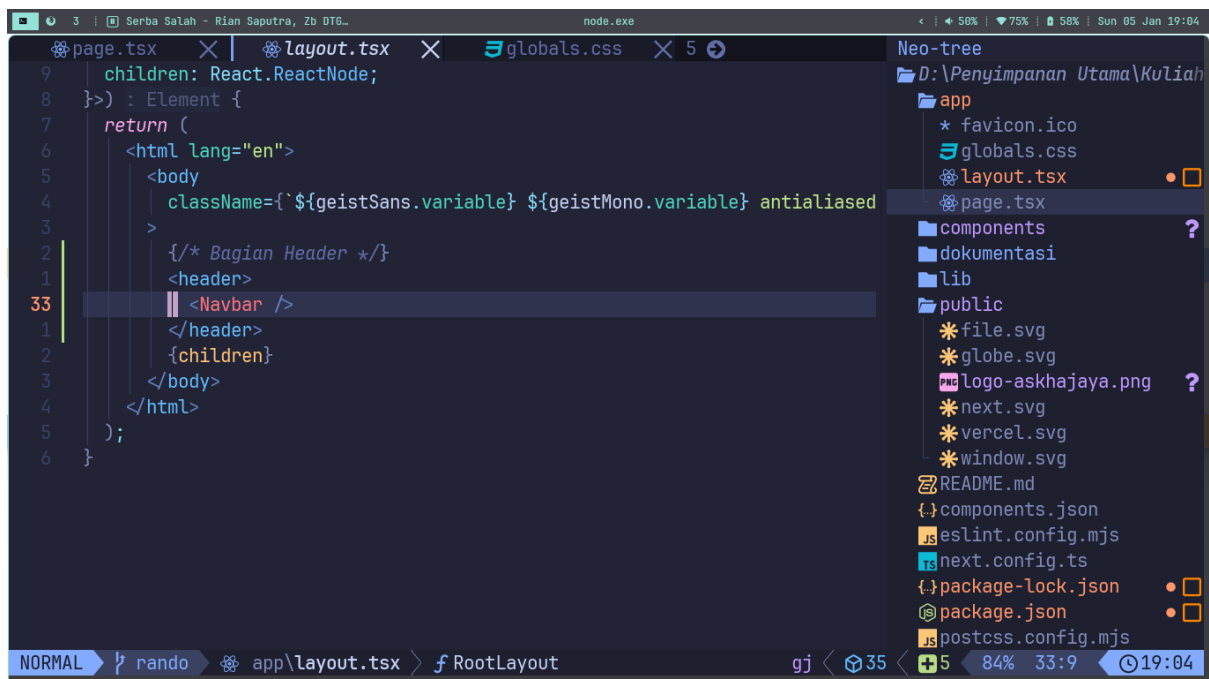
Dengan perintah **npx shadcn@latest add button**.



Deskripsi Penjelasan: Setelah membuat komponen tombol, saya lanjutkan untuk membuat sebuah file **Navbar.tsx** yang berisi kodingan untuk navbar.

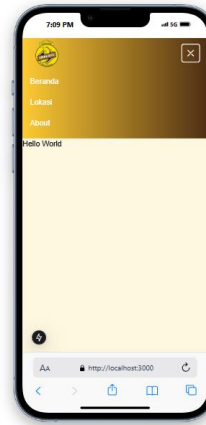
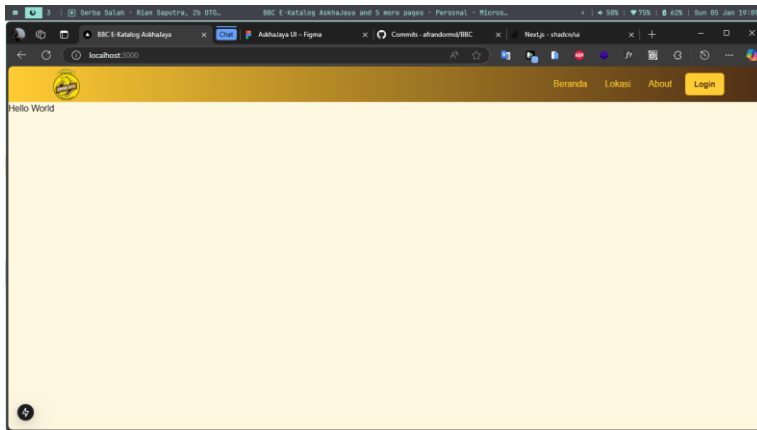


Deskripsi Penjelasan: Setelah file **Navbar.tsx** dibuat, selanjutnya saya membuat kodingan untuk komponen navbar, dan menambahkan sebuah file berupa logo di dalam folder **public**.



Deskripsi Penjelasan: Setelah komponen navbar berhasil dibuat, selanjutnya saya memanggilnya di file **layout.tsx** yang berada di dalam folder **app**. Saya meletakkan di file tersebut karena komponen navbar bersifat general, yang nantinya dipakai di masing – masing page yang akan dibuat.

Output Hasil Komponen Navbar



Deskripsi Penjelasan: Komponen navbar ini sudah bersifat responsive, sehingga ketika dibuka di layar desktop, tablet maupun smartphone tampilannya akan menyesuaikan.

Membuat Hero Section

```
1 import Image from "next/image";
2 // import { Button } from "@elements/button"
3 import { Button } from "../elements/button";
4 export default function HeroSection() : Element {
5   return (
6     <section className="w-full py-12 md:py-16 lg:py-20">
7       <div className="container px-4 md:px-6">
8         <div className="grid gap-6 lg:grid-cols-2 lg:gap-12 items-center">
9           <div className="flex flex-col justify-center space-y-4">
10             <div className="space-y-2">
11               <h1 className="text-4xl font-bold tracking-tighter sm:text-5x">
12                 Askha Jaya
13               </h1>
14               <h2 className="text-xl md:text-2xl font-medium text-gray-800">
15                 Lampung Surganya Pisang
16               </h2>
17             </div>
18             <p className="max-w-[600px] text-gray-600 md:text-lg">
19               Dengan lebih dari 4 cabang yang tersebar di berbagai wilayah
20               Bandar Lampung, Tentunya Askha Jaya selalu menghadirkan
21               pengalaman, unik dan memperkuat hubungan kepada pelanggan men
22               fokus utama.
23             </p>
24           </div>
25           <div>
26             <img alt="Hero Section Image" data-bbox="680 425 875 680" />
27           </div>
28         </div>
29       </div>
30     </section>
31   );
32 }
```

Deskripsi Penjelasan: Saya membuat file baru bernama **HeroSection.tsx** kemudian menulis kodingan untuk hero section yang berisikan copywriting di sebelah kiri dan gambar disebelah kanan.

```
HeroSection.tsx | page.tsx
6 import HeroSection from "@components/fragments/HeroSection";
5
4 export default function Home() : Element {
3   return (
2     <div>
1       <div>
7         <HeroSection />
1       </div>
2     </div>
3   );
}
```

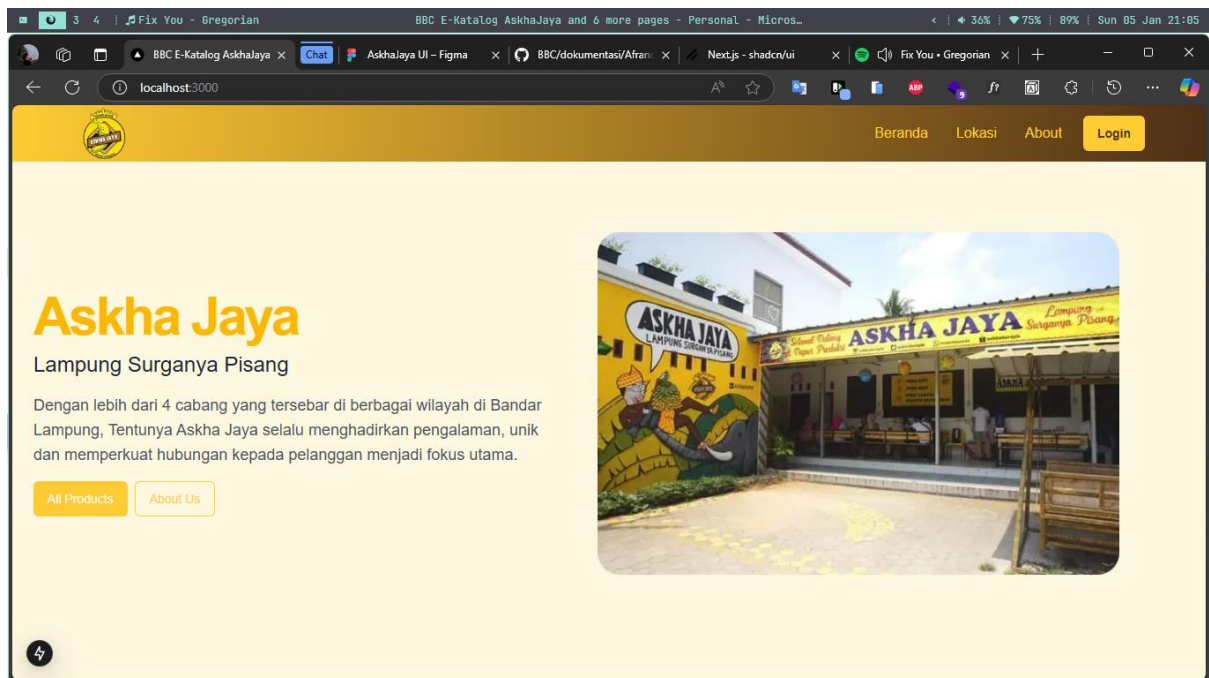
Neo-tree

- D:\Penyimpanan Utama\Kuliah
- app
- components
 - elements
 - fragments
 - HeroSection.tsx ?
 - Navbar.tsx
- dokumentasi
- lib
- public
- README.md
- components.json
- eslint.config.mjs
- next.config.ts
- package-lock.json
- package.json
- postcss.config.mjs
- tailwind.config.ts
- tsconfig.json
- (5 hidden items)

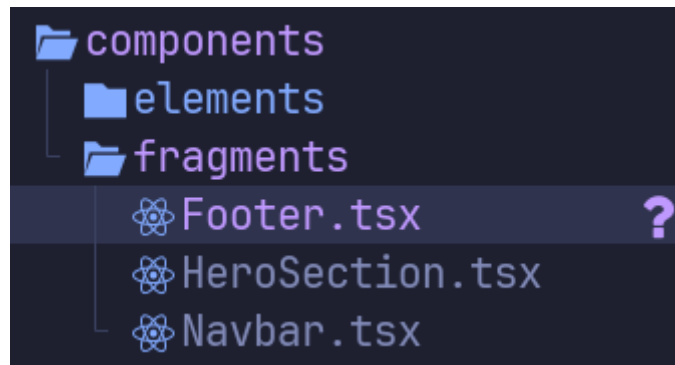
NORMAL | rando | app\page.tsx | f Home | gj | 35 | +3 | 1 | 70% | 7:7 | 21:03

Deskripsi Penjelasan: Setelah membuat hero section, saya memanggilnya di dalam file **page.tsx** yang berada di folder **app**.

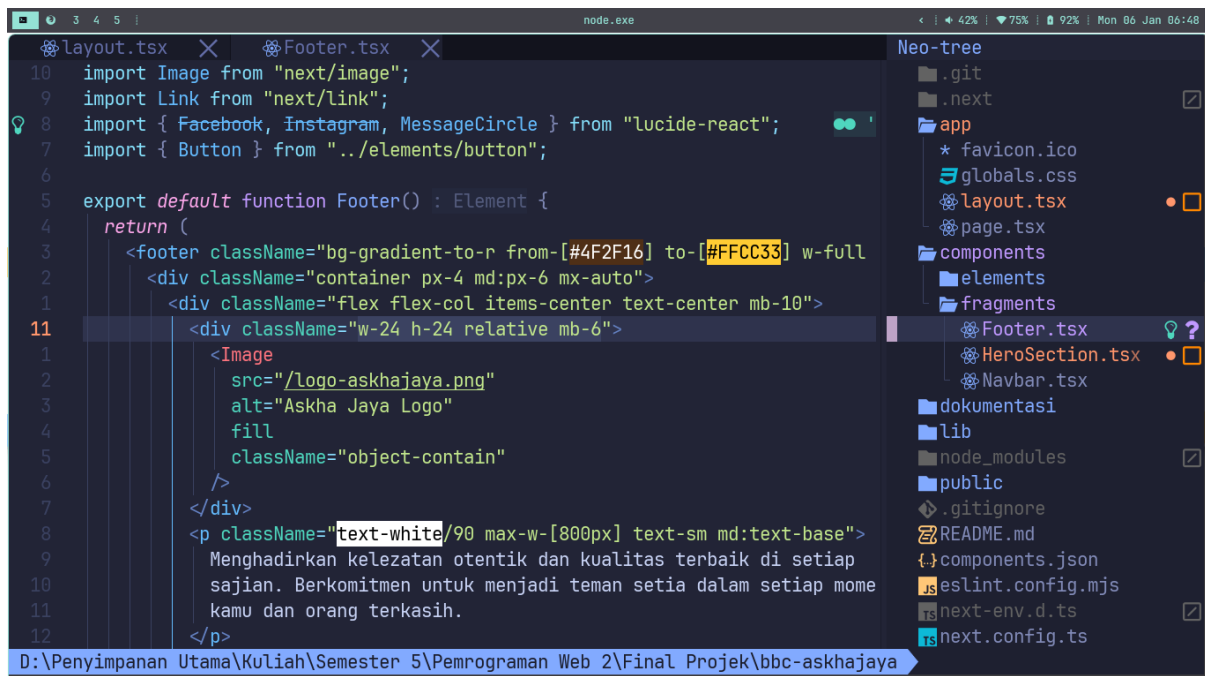
Output Hasil Hero Section



Membuat Footer



Deskripsi Penjelasan: Saya membuat sebuah file bernama **Footer.tsx** di dalam folder **components/fragments** yang nantinya file ini akan saya isi kodingan komponen footer.



```
10 import Image from "next/image";
9 import Link from "next/link";
8 import { Facebook, Instagram, MessageCircle } from "lucide-react";
7 import { Button } from "../elements/button";

5 export default function Footer() : Element {
4   return (
3     <footer className="bg-gradient-to-r from-[#4F2F16] to-[#FFCC33] w-full
2     <div className="container px-4 md:px-6 mx-auto">
1     <div className="flex flex-col items-center text-center mb-10">
11    <div className="w-24 h-24 relative mb-6">
1      <Image
2        src="/Logo-askhajaya.png"
3        alt="Askha Jaya Logo"
4        fill
5        className="object-contain"
6      />
7    </div>
8    <p className="text-white/90 max-w-[800px] text-sm md:text-base">
9      Menghadirkan kelezatan otentik dan kualitas terbaik di setiap
10     sajian. Berkomitmen untuk menjadi teman setia dalam setiap mome
11     kamu dan orang terkasih.
12   </p>
  )
}
```

Deskripsi Penjelasan: Setelah membuat file, kemudian saya mengisi file tersebut dengan kodingan komponen footer.

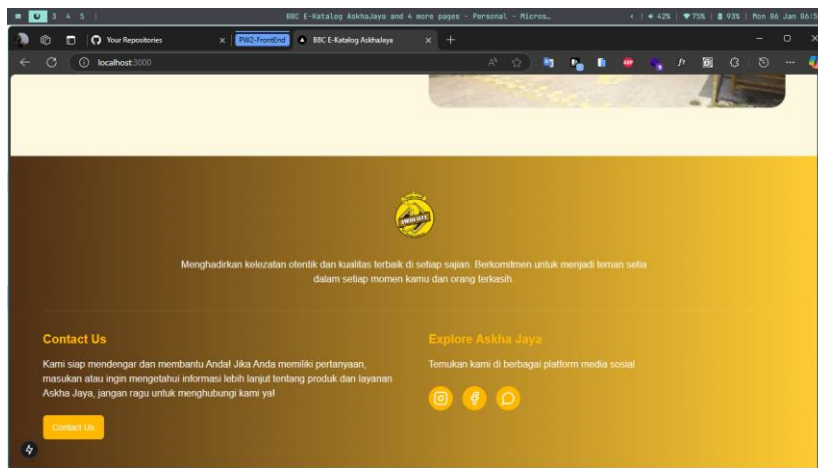

```

15 children: React.ReactNode;
14 }>) : Element {
13   return (
12     <html lang="en">
11       <body
10         className={` ${geistSans.variable} ${geistMono.variable} antialiased
9         >
8         { /* Bagian Header */}
7         <header>
6           <Navbar />
5         </header>
4         {children}
3         { /* Bagian Footer */}
2         <footer>
1           <Footer />
40        </footer>
1        </body>
2        </html>
3      );
4    }

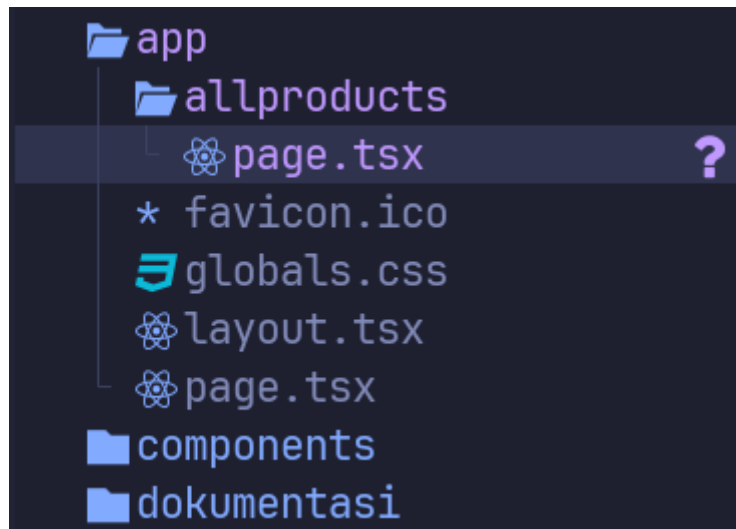
```

Deskripsi Penjelasan: Setelah komponen **Navbar** selesai dibuat, saya memanggilnya di dalam file **layout.tsx** karena komponen footer akan digunakan secara general, sama seperti komponen **Navbar** sebelumnya.

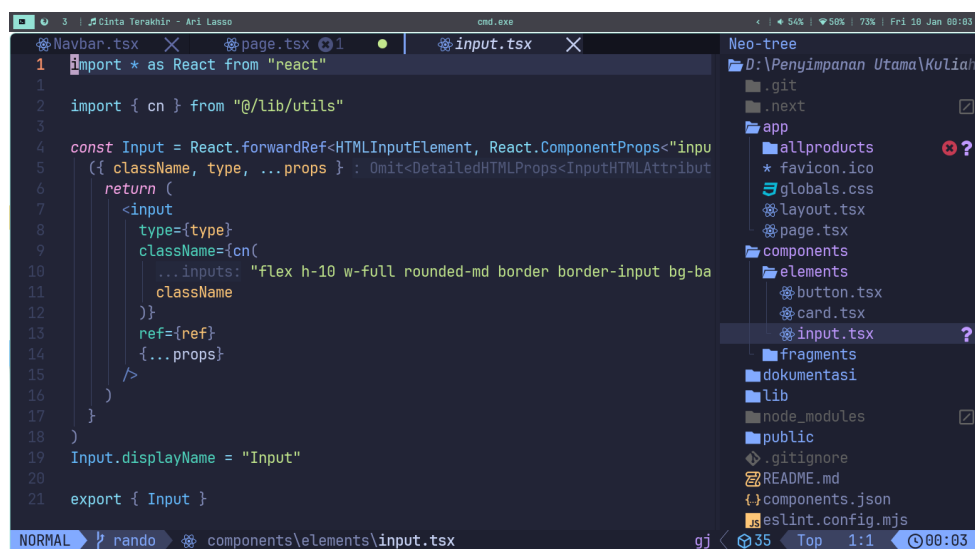
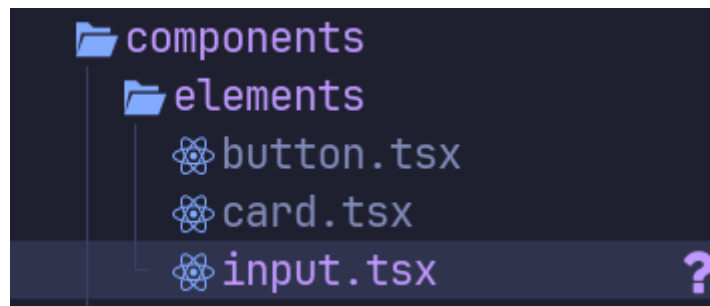
Output Footer



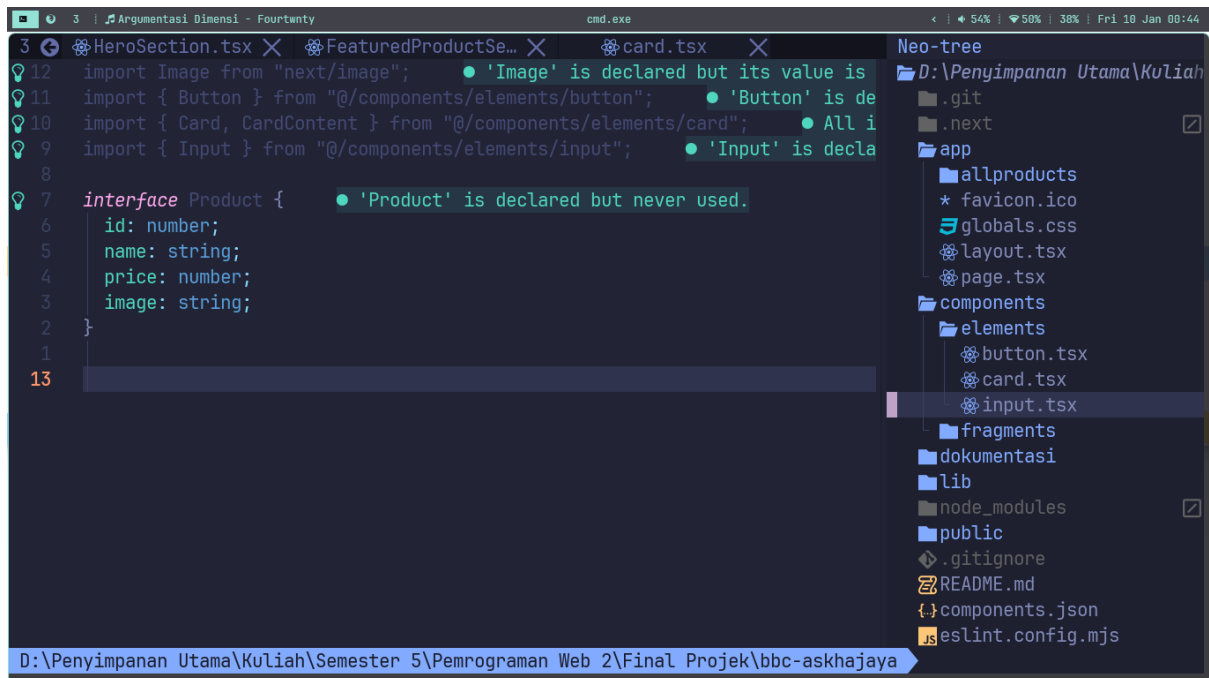
Membuat Halaman All Products



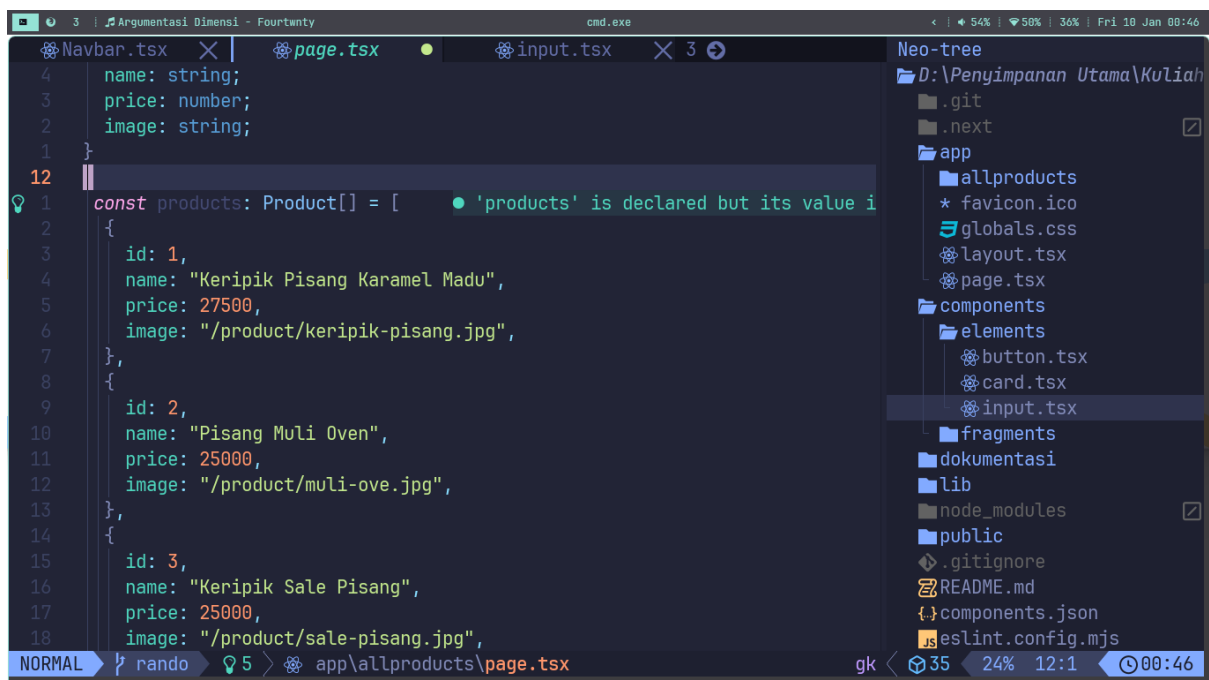
Deskripsi Penjelasan: Saya membuat sebuah folder bernama **allproducts** yang berisi file **page.tsx** yang nantinya file ini akan saya isi dengan kodingan untuk menampilkan seluruh produk yang ada di website e-katalog yang dibangun.



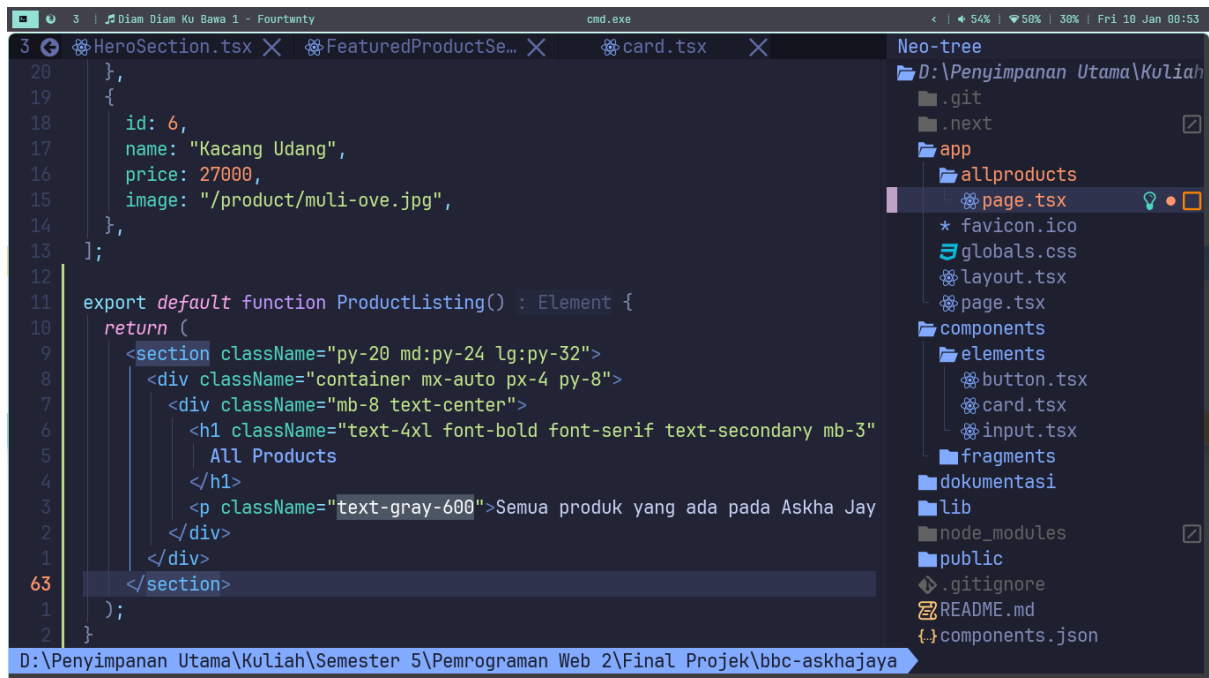
Deskripsi Penjelasan: Dikarenakan pada halaman allproducts terdapat input search untuk mencari produk, saya membuat sebuah component dengan nama **input.tsx**



Deskripsi Penjelasan: Saya membuat interface untuk mendefinisikan struktur dari data produk yang nantinya akan ditampilkan di halaman **allproducts**.



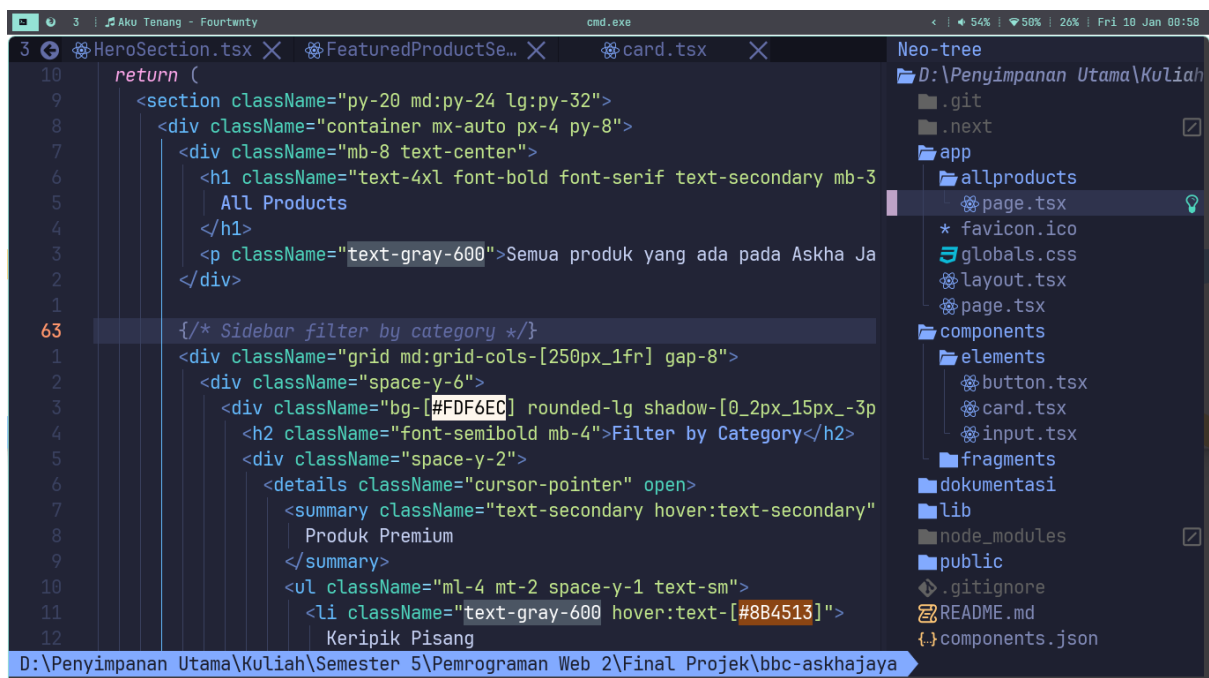
Deskripsi Penjelasan: Setelah membuat **interface**, saya membuat objek yang berisi data dummy untuk mengisi card produk sementara.



```
20  },
19  {
18    id: 6,
17    name: "Kacang Ugang",
16    price: 27000,
15    image: "/product/muli-ove.jpg",
14  },
13 ];
12
11 export default function ProductListing() : Element {
10   return (
9     <section className="py-20 md:py-24 lg:py-32">
8       <div className="container mx-auto px-4 py-8">
7         <div className="mb-8 text-center">
6           <h1 className="text-4xl font-bold font-serif text-secondary mb-3">
5             All Products
4           </h1>
3           <p className="text-gray-600">Semua produk yang ada pada Askha Jay
2         </div>
1       </div>
63    </section>
1    );
2  }

```

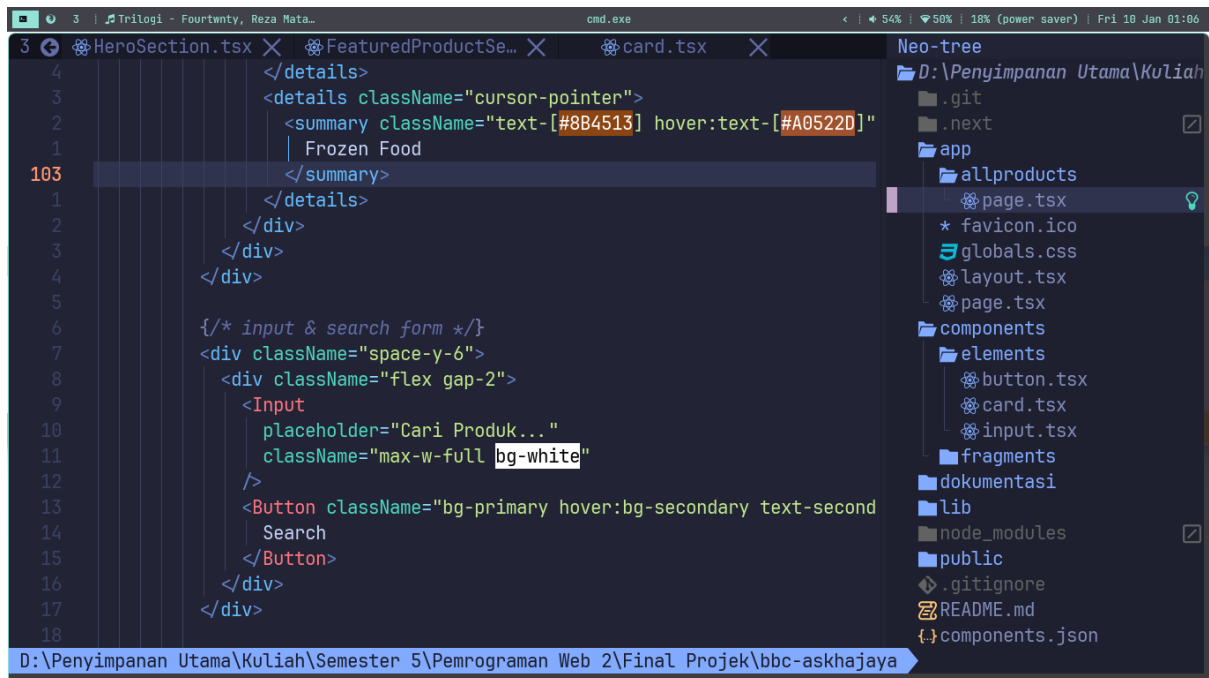
Deskripsi Penjelasan: Selanjutnya, saya membuat sebuah fungsi bernama **ProductListing** dan di dalamnya terdapat section untuk membungkus komponen pada **allproducts**.



```
10   return (
9     <section className="py-20 md:py-24 lg:py-32">
8       <div className="container mx-auto px-4 py-8">
7         <div className="mb-8 text-center">
6           <h1 className="text-4xl font-bold font-serif text-secondary mb-3">
5             All Products
4           </h1>
3           <p className="text-gray-600">Semua produk yang ada pada Askha Ja
2         </div>
63       { /* Sidebar filter by category */ }
1       <div className="grid md:grid-cols-[250px_1fr] gap-8">
2         <div className="space-y-6">
3           <div className="bg-[#FDF6EC] rounded-lg shadow-[0_2px_15px_-3px]">
4             <h2 className="font-semibold mb-4">Filter by Category</h2>
5             <div className="space-y-2">
6               <details className="cursor-pointer" open>
7                 <summary className="text-secondary hover:text-secondary">
8                   Produk Premium
9                 </summary>
10                <ul className="ml-4 mt-2 space-y-1 text-sm">
11                  <li className="text-gray-600 hover:text-[#8B4513]">
12                    Keripik Pisang

```

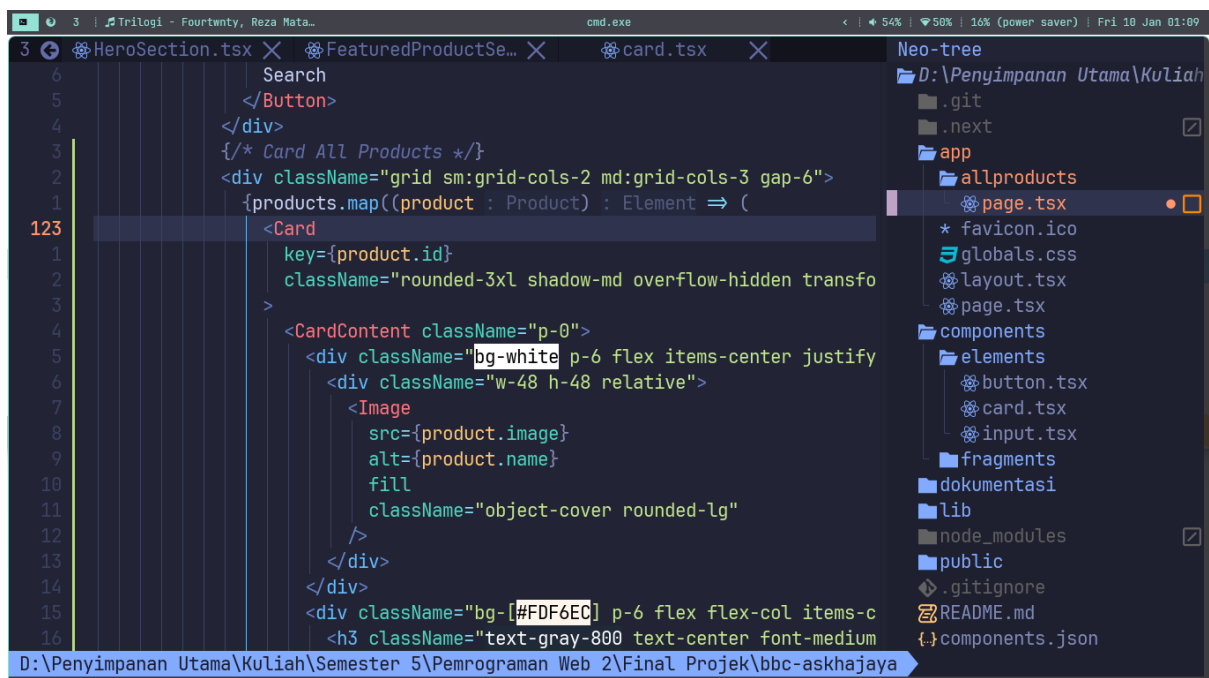
Deskripsi Penjelasan: Kemudian saya membuat sebuah sidebar yang nantinya akan digunakan untuk melakukan filterisasi produk berdasarkan kategori produknya.



```
4      </details>
3      <details className="cursor-pointer">
2        <summary className="text-[#8B4513] hover:text-[#A0522D]">
1          Frozen Food
103      </summary>
      </details>
2    </div>
3  </div>
4 </div>
5
6  {/* input & search form */}
7  <div className="space-y-6">
8    <div className="flex gap-2">
9      <Input
10        placeholder="Cari Produk..."
11        className="max-w-full bg-white"
12      />
13      <Button className="bg-primary hover:bg-secondary text-second
14        Search
15      </Button>
16    </div>
17  </div>
18
```

D:\Penyimpanan Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya

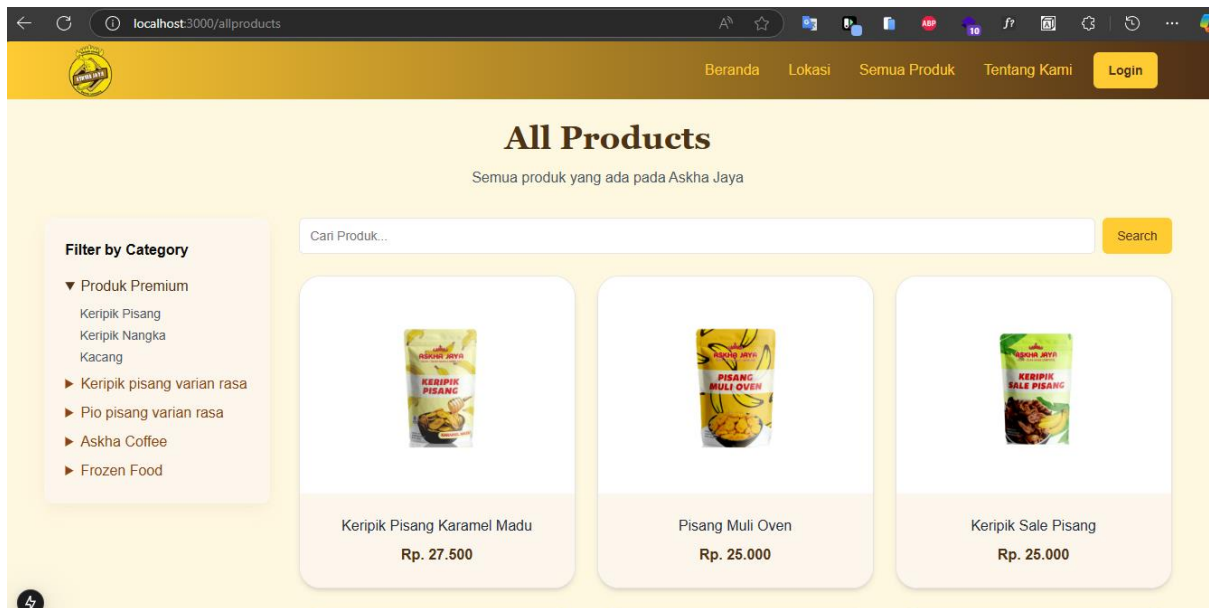
Deskripsi Penjelasan: Setelah itu saya membuat sebuah input search form yang nantinya akan digunakan untuk pencarian produk berdasarkan nama di halaman **allproducts**.



```
6      Search
5    </Button>
4  </div>
3  {/* Card All Products */}
2  <div className="grid sm:grid-cols-2 md:grid-cols-3 gap-6">
1    {products.map((product : Product) : Element => (
123  <Card
      key={product.id}
      className="rounded-3xl shadow-md overflow-hidden transfo
    >
      <CardContent className="p-0">
        <div className="bg-white p-6 flex items-center justify
          <div className="w-48 h-48 relative">
            <Image
              src={product.image}
              alt={product.name}
              fill
              className="object-cover rounded-lg"
            />
          </div>
        </div>
        <div className="bg-[#FDF6EC] p-6 flex flex-col items-c
          <h3 className="text-gray-800 text-center font-medium
16
```

D:\Penyimpanan Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya

Deskripsi Penjelasan: Selanjutnya saya membuat komponen card untuk membungkus data produk yang akan ditampilkan di halaman **allproducts**.



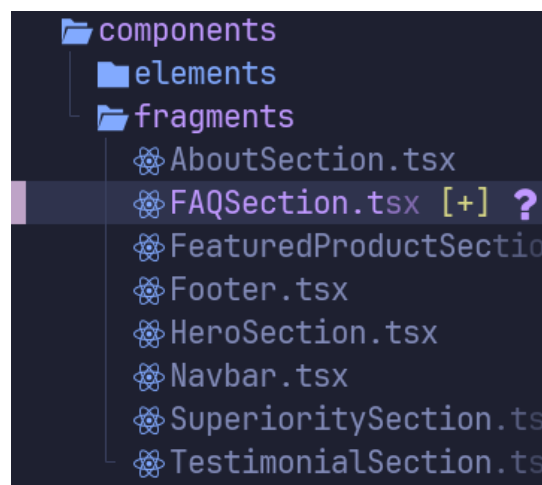
Deskripsi Penjelasan: Berikut ini output sementara dari hasil dari kodingan yang saya buat sebelumnya.

Membuat Section Frequently Ask Questions (FAQs)

```
1: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22631.4602]
(c) Microsoft Corporation. All rights reserved.

D:\Penyimpanan Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya>npm install motion
```

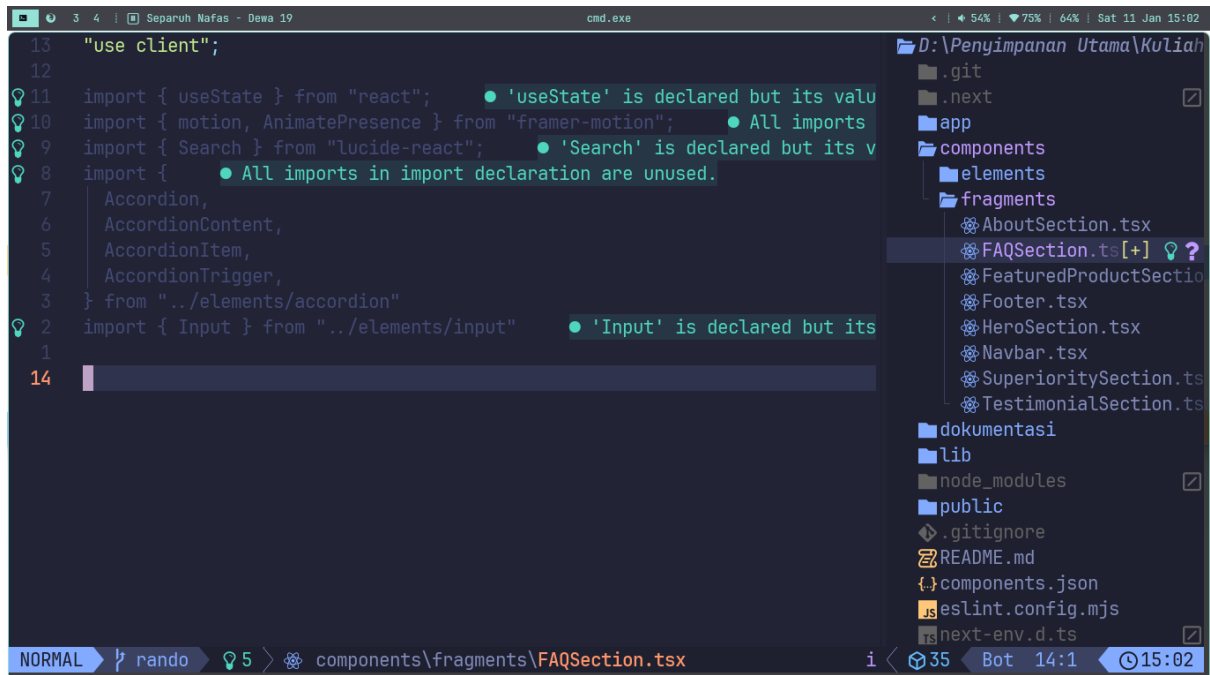
Deskripsi Penjelasan: Komponen yang akan dibuat membutuhkan animasi ketika diklik, oleh karena itu saya menginstall **framer-motion** terlebih dahulu agar dapat berfungsi dengan baik.



Deskripsi Penjelasan: Setelah itu saya membuat sebuah file bernama **FAQSection.tsx** untuk komponen section **FAQs**.

```
2: C:\Windows\system32\cmd.exe
D:\Penyimpanan Utama\Kuliah\Semester 5\Pemrograman Web 2\Final Projek\bbc-askhajaya>npx shadcn@latest add accordion
on
✓Checking registry.
✓Updating tailwind.config.ts
Installing dependencies.
```

Deskripsi Penjelasan: Saya membuat komponen accordion menggunakan shaden/ui.



```
13 "use client";
12
11 import { useState } from "react";
10 import { motion, AnimatePresence } from "framer-motion";
9 import { Search } from "lucide-react";
8 import {
7   Accordion,
6   AccordionContent,
5   AccordionItem,
4   AccordionTrigger,
3 } from "../elements/accordion"
2 import { Input } from "../elements/input"
1
14
```

Deskripsi Penjelasan: Di dalam file **FAQSection.tsx** saya mengimport terlebih dahulu package yang dibutuhkan untuk komponen **FAQs** yang akan dibuat.

```
import { Input } from "../elements/input"
const faqs = [
  {
    question: "Apa saja jenis oleh-oleh yang tersedia di toko kami?",
    answer: "Kami menyediakan berbagai jenis oleh-oleh khas daerah, termasuk..."
  },
  {
    question: "Bagaimana cara memesan produk dari e-katalog?",
    answer: "Anda dapat memesan produk dengan memilih item yang diinginkan,..."
  },
  {
    question: "Apakah ada layanan pengiriman ke luar kota atau luar negeri?",
    answer: "Ya, kami menyediakan layanan pengiriman ke seluruh Indonesia dan..."
  },
  {
    question: "Berapa lama waktu pengiriman untuk pesanan saya?",
    answer: "Waktu pengiriman bervariasi tergantung lokasi tujuan. Untuk..."
  },
  {
    question: "Apakah produk-produk yang dijual memiliki sertifikasi halal?",
    answer: "Sebagian besar produk makanan yang kami jual telah memiliki..."
  }
]
```

Deskripsi Penjelasan: Selanjutnya, saya membuat sebuah array yang berisi objek untuk pertanyaan dan jawaban yang akan di tampilkan di komponen **FAQs**.

```
export default function FAQSection() : Element {
  const [searchTerm, setSearchTerm] = useState(initialState: "")
  const filteredFaqs = faqs.filter(
    (faq : { question: string; answer: string }) : boolean =>
      faq.question.toLowerCase().includes(searchTerm.toLowerCase()) ||
      faq.answer.toLowerCase().includes(searchTerm.toLowerCase())
  )
  return (
    <section></section>
  )
}
```

Deskripsi Penjelasan: Kemudian, dibawahnya saya membuat struktu dasar untuk komponen **FAQs**.


```

return (
  <section className="py-12 bg-gray-50">
    <div className="container mx-auto px-4"> {/ container pembungkus */}
    <h2 className="text-3xl font-bold text-center mb-8">
      Pertanyaan yang Sering Diajukan
    </h2>
    <div className="max-w-xl mx-auto mb-8"> {/ komponen pencarian pertanyaan */}
    <div className="relative">
      <Search className="absolute left-3 top-1/2 transform -translate-y-1/2 text-gray-400" />
      <Input
        type="text"
        placeholder="Cari pertanyaan..."
        className="pl-10"
        value={searchTerm}
        onChange={(e : ChangeEvent<HTMLInputElement>) : void => setSearchTerm(e.target.value)}
      />
    </div> {/ penutup pencarian pertanyaan */}
  </div> {/ penutup akhir container */}
</section>
);
}

```

Deskripsi Penjelasan: Selanjutnya saya mengisi struktur dengan judul dan kolom pencarian pertanyaan.

```

19 </div>{"" ""}
18 {/ * penutup pencarian pertanyaan */}
17 <AnimatePresence>
16   {filteredFaq.length > 0 ? (
15     <motion.div
14       initial={{ opacity: 0, y: 20 }}
13       animate={{ opacity: 1, y: 0 }}
12       exit={{ opacity: 0, y: 20 }}
11       transition={{ duration: 0.3 }}
10     >
9       <Accordion
8         type="single"
7         collapsible
6         className="max-w-2xl mx-auto"
5       >
4         {filteredFaq.map((faq : { question: string; answer: string }, index : number) : Element
3           <AccordionItem key={index} value={`item-${index}`}>
2             <AccordionTrigger>{faq.question}</AccordionTrigger>
1             <AccordionContent>{faq.answer}</AccordionContent>
90          </AccordionItem>
1          )}
2        </Accordion>
3      </motion.div>
4    ) : (

```

Deskripsi Penjelasan: Selanjutnya saya membuat komponen untuk pertanyaan FAQs dengan accordion dan framer motion.

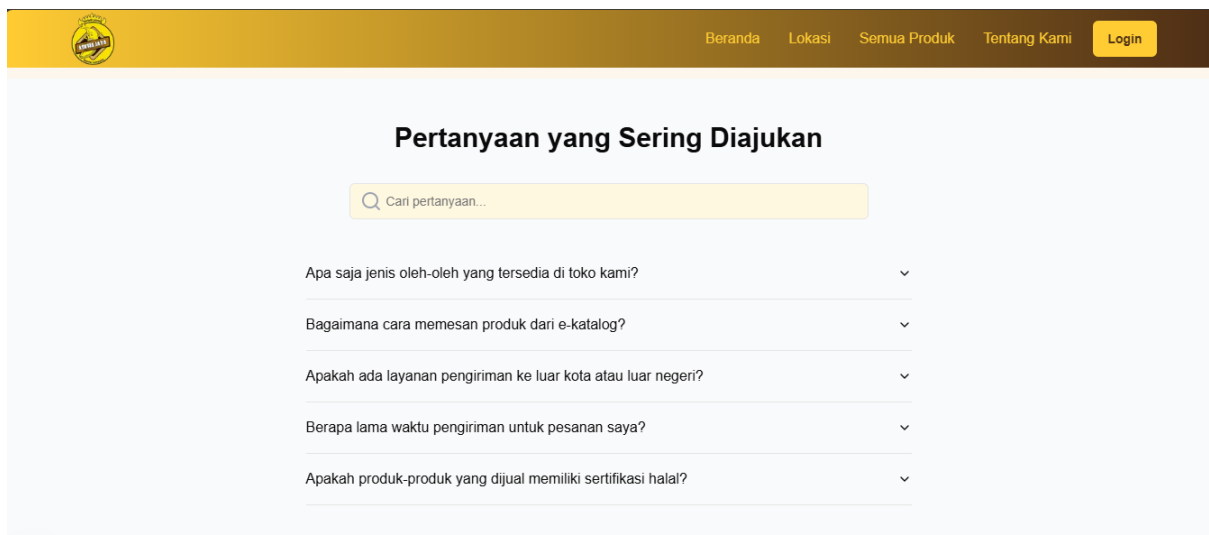
```

export default function Home() : Element {
  return (
    <
      {/* Bagian Hero Section */}
      <HeroSection />
      {/* Bagian Product Section */}
      <FeaturedProductSection />
      {/* Bagian Superiority Section */}
      <Superiority />
      {/* Bagian Testimonial Section */}
      <Testimonial />
      {/* Bagian About Section */}
      <AboutSection />
      {/* Bagian FAQs Section */}
      <FAQSection />
    </
  >);
}

```

Deskripsi Penjelasan: Setelah komponen berhasil dibuat, kemudian saya memanggilnya di file **page.tsx** utama yang berada di dalam folder **app**.

Output Komponen Section Frequently Ask Questions (FAQs)



Deskripsi Penjelasan: Berikut adalah hasil output dari komponen **FAQs** yang telah saya buat.