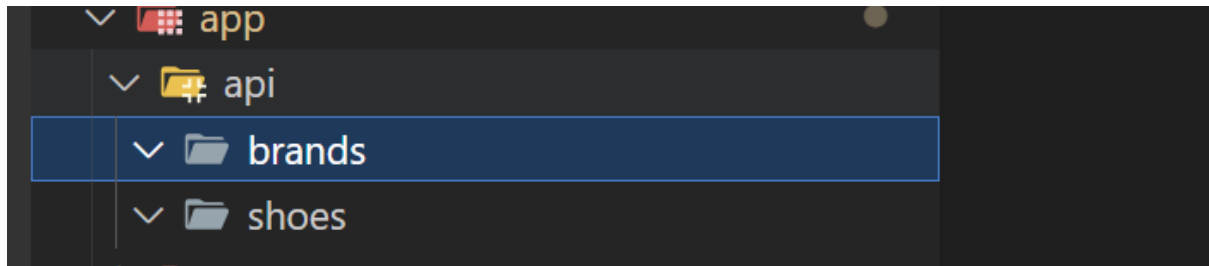


Nama : Farid Hambali

NPM : 22312112

Sesi 1



Membuat folder api yang berisi brands dan shoes

```
app > api > brands > route.ts > GET
1 import { NextResponse } from 'next/server';
2 export async function GET() {
3   const brands = [
4     { id: 'adidas', name: 'Adidas', logo: '/images/adidas-logo.png' },
5     { id: 'ortuseight', name: 'Ortuseight', logo: '/images/ortus-logo.png' },
6     { id: 'salomon', name: 'Salomon', logo: '/images/salomon-logo.png' },
7     { id: 'nike', name: 'Nike', logo: '/images/nike-logo.png' },
8     { id: 'newbalance', name: 'New Balance', logo: '/images/newbalance-logo.png' },
9     { id: 'on', name: 'On', logo: '/images/on-logo.png' },
10  ];
11
12  return NextResponse.json(brands);
13 }
```

API untuk data merek sepatu

```

app > api > shoes > route.ts > GET
1  import { NextResponse } from 'next/server';
2
3  export async function GET(request: Request) {
4    try {
5      const { searchParams } = new URL(request.url);
6      const brand = searchParams.get('brand'); // Mendapatkan parameter "brand" dari URL
7
8      // Data sepatu dengan harga dan ukuran
9      const shoes: Record<string, Array<{ id: string; name: string; image: string; price: number; sizes: string[] }>> = {
10        adidas: [
11
12        ],
13        ortuseight: [
14
15        ],
16        nike: [
17
18        ],
19        newbalance: [
20
21        ],
22        salomon: [
23
24        ],
25        on: [
26
27        ]
28      ];
29
30      const shoeData = brand && shoes[brand] ? shoes[brand] : [];
31
32      return NextResponse.json(shoeData);
33    } catch (error) {
34      console.error('Error handling request:', error);
35      return NextResponse.json({ error: 'An error occurred while processing your request.' }, { status: 500 });
36    }
37  }
38
39  }
40

```

API untuk data sepatu dari setiap merek

```

page.tsx app  route.ts ...brands  route.ts ...shoes  page.tsx ...[id] U X  Untitled-1
app > brands > [id] > page.tsx > Shoe
1  "use client";
2  import { useEffect, useState } from 'react';
3
4  type Shoe = {
5    id: string;
6    name: string;
7    image: string;
8    price: number;
9    sizes: string[];
10 };

```

Halaman dinamis untuk sepatu per merek

```
app > brands > [id] > page.tsx > BrandDetail
```

```
11
12 export default function BrandDetail({ params }: { params: { id: string } }) {
13   const [shoes, setShoes] = useState<Shoe[]>([]);
14   const [selectedShoe, setSelectedShoe] = useState<Shoe | null>(null);
15   const [isModalOpen, setIsModalOpen] = useState<boolean>(false);
16
17   useEffect(() => {
18     const fetchShoes = async () => {
19       const res = await fetch(`/api/shoes?brand=${params.id}`);
20       const data = await res.json();
21       setShoes(data);
22     };
23
24     fetchShoes();
25   }, [params.id]);
26
27   const handleShoeClick = (shoe: Shoe) => {
28     setSelectedShoe(shoe);
29     setIsModalOpen(true); // Show the modal
30   };
31
32   const closeModal = () => {
33     setIsModalOpen(false);
34   };
35 }
```

Membuat komponen untuk menampilkan detail merek berdasarkan id

```
app > brands > [id] > page.tsx > BrandDetail
```

```
12 export default function BrandDetail({ params }: { params: { id: string } }) {
13
14   return (
15     <main className="container mx-auto p-6">
16       <h1 className="text-4xl font-extrabold text-gray-800 text-center mb-8">Sepatu {params.id}</h1>
17
18       <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-8">
19         {shoes.map((shoe) => (
20           <div
21             key={shoe.id}
22             className="group relative bg-white rounded-lg shadow-lg overflow-hidden cursor-pointer hover:scale-105 transition-all duration-300"
23             onClick={() => handleShoeClick(shoe)}
24           >
25             <img
26               src={shoe.image}
27               alt={shoe.name}
28               className="w-full h-56 object-cover group-hover:scale-110 transition-transform duration-300"
29             />
30
31             <div className="p-4">
32               <h2 className="text-lg font-semibold text-gray-800 group-hover:text-blue-500 transition-colors duration-300">{shoe.name}</h2>
33               <p className="mt-2 text-xl font-bold text-black">Rp {shoe.price.toLocaleString()}</p>
34             </div>
35           </div>
36         ))}
37       </div>
38
39       {/* Modal for selected shoe */}
40       {isModalOpen && selectedShoe && (
41         <div className="fixed inset-0 bg-black bg-opacity-50 flex justify-center items-center z-50">
42           <div className="bg-white p-6 rounded-lg shadow-xl max-w-lg w-full">
43             <button
44               className="absolute top-4 right-4 text-black font-bold text-2xl"
45               onClick={closeModal}
46             >
47               &times;
48             </button>
49
50             <div className="flex flex-col items-center">
51               <img
52                 src={selectedShoe.image}
53                 alt={selectedShoe.name}
54                 className="w-72 h-72 object-cover mb-4 transition-transform duration-300 transform-hover:scale-110"
55               />
56             <h2 className="text-2xl font-semibold text-gray-800">{selectedShoe.name}</h2>
57             <p className="text-2xl font-semibold text-black">Rp {selectedShoe.price.toLocaleString()}</p>
58           </div>
59         </div>
60       )}
61     </main>
62   );
63 }
```

menampilkan daftar sepatu berdasarkan **ID merek** dan menyediakan modal untuk melihat detail sepatu yang dipilih.

```

app > page.tsx > [🔗] Brand
1  "use client";
2  import { useEffect, useState } from 'react';
3  import Link from 'next/link';
4
5  type Brand = {
6    id: string;
7    name: string;
8    logo: string;
9  };

```

Membuat halaman home pertama kali

```

app > page.tsx > [🏠] Home
10
11 export default function Home() {
12   const [brands, setBrands] = useState<Brand[]>([]);
13
14   useEffect(() => {
15     const fetchBrands = async () => {
16       const res = await fetch('/api/brands');
17       const data = await res.json();
18       setBrands(data);
19     };
20
21     fetchBrands();
22   }, []);

```

menampilkan daftar merek sepatu dengan data yang diambil dari API

```

app > page.tsx > [🏠] Home
11 export default function Home() {
23   return (
24     <main className="container mx-auto p-6">
25       <h1 className="text-4xl font-extrabold text-gray-800 text-center mb-10">Step Up Your Game Discover the Best Sports Shoes Here!</h1>
26       <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-8">
27         {brands.map((brand) => (
28           <Link
29             key={brand.id}
30             href={`/brands/${brand.id}`}
31             className="group relative bg-white shadow-xl rounded-lg overflow-hidden hover:scale-105 transition-transform duration-300"
32           >
33             <img
34               src={brand.logo}
35               alt={brand.name}
36               className="mx-auto w-28 h-28 object-contain p-4 transition-transform duration-300 group-hover:scale-110"
37             />
38             <div className="absolute inset-0 bg-gradient-to-t from-black to-transparent opacity-40 group-hover:opacity-0 transition-opacity duration-300">
39               <h2 className="mt-4 text-xl font-semibold text-gray-800 group-hover:text-blue-500 text-center">{brand.name}</h2>
40             </div>
41           </Link>
42         ))}
43       </div>
44     </main>
45   );

```

Menambahkan daftar merek Sepatu di halaman home

```

app > layout.tsx > ...
1  import type { Metadata } from "next";
2  import localFont from "next/font/local";
3  import Link from "next/link"; // Import Link from Next.js
4  import "./globals.css";
5
6  // Menambahkan font lokal
7  const geistSans = localFont({
8    src: "./fonts/GeistVF.woff",
9    variable: "--font-geist-sans",
10   weight: "100 900",
11 });
12 const geistMono = localFont({
13   src: "./fonts/GeistMonoVF.woff",
14   variable: "--font-geist-mono",
15   weight: "100 900",
16 });

```

Menambahkan font di layout.tsx

```

app > layout.tsx > ...projek akhir\BAD_VIBRATIONS\app\page.tsx
16  });
17  export const metadata: Metadata = {
18    title: "Sepatu Olahraga Keren",
19    description: "Tampilan modern untuk katalog sepatu olahraga.",
20  };

```

Mengganti nama website menjadi Sepatu olahraga keren

```

app > layouts.tsx > RootLayout
21 export default function RootLayout({
22
23 }) {
24   return (
25     <html lang="en">
26       <head>
27         <link rel="icon" href="/favicon.ico" />
28       </head>
29       <body
30         className={` ${geistSans.variable} ${geistMono.variable} antialiased bg-gray-100 text-gray-800`>
31         >
32           <div className="min-h-screen bg-gray-200">
33             <div>
34               <div>
35                 <div>
36                   <div>
37                     <div>
38                       <div>
39                         <div>
40                           <div>
41                             <div>
42                               <div>
43                                 <div>
44                                   <div>
45                                     <div>
46                                       <div>
47                                         <div>
48                                           <div>
49                                             <div>
50                                               <div>
51                                                 <div>
52                                                   <div>
53                                                     <div>
54                                                       <div>
55                                                         <div>
56                                                           <div>
57                                                             <div>
58                                                               <div>
59                                                                 <div>
60                                                                  <div>
61                                                                   <div>
62                                                                    <div>

```

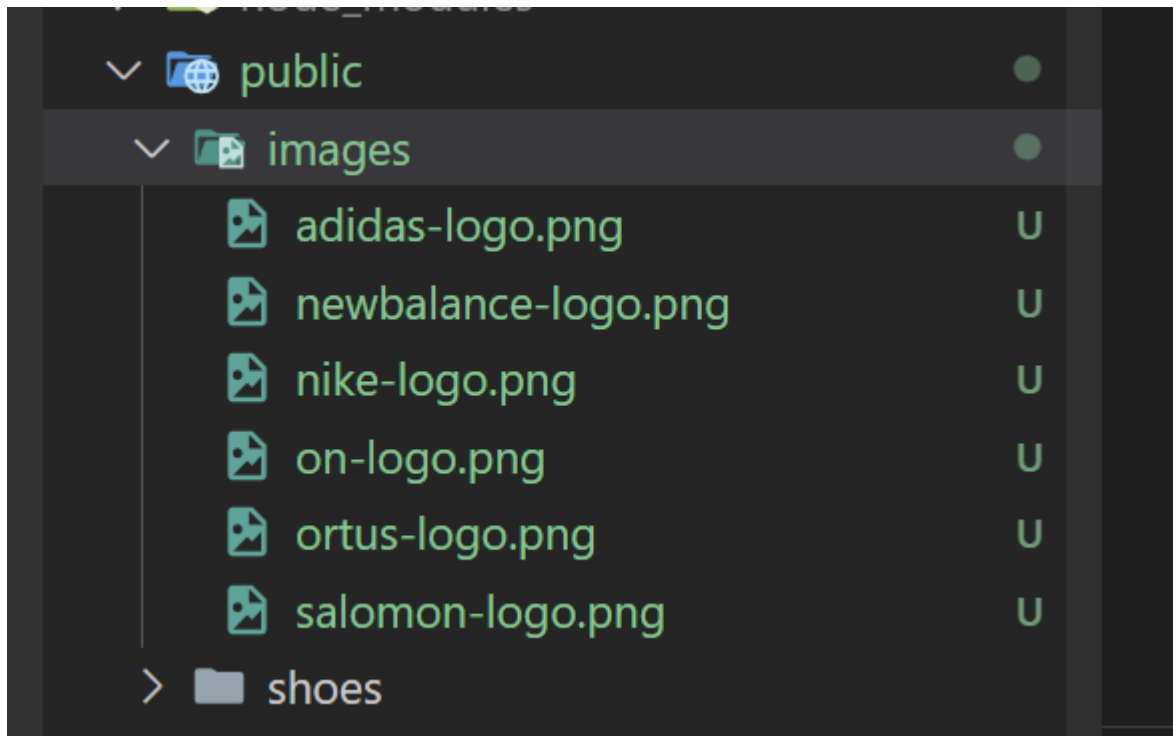
Menambahkan navbar, menu home,about,contact dan copyright

```

app > globals.css > body
16 @media (prefers-color-scheme: dark) {
17
18 }
19 /* Mengatur dasar font dan background */
20 body {
21   color: var(--foreground);
22   background: var(--background);
23   font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
24   line-height: 1.6;
25   transition: background 0.3s ease, color 0.3s ease;
26 }

```

Mengatur font dan background dasar



Menambahkan folder public yang berisi images dan shoes

Sesi 2

```
/* Efek transisi untuk elemen */
a, button {
  transition: color 0.3s ease, background-color 0.3s ease;
}
```

Efek transisi untuk elemen

```
button {
  padding: 12px 24px;
  font-size: 16px;
  font-weight: 600;
  border: none;
  border-radius: 8px;
  background-color: var(--primary-color);
  color: #fff;
  cursor: pointer;
  box-shadow: var(--shadow-light);
  transition: background-color 0.3s ease, box-shadow 0.3s ease;
}
```

Menambahkan desain tombol

```
button:hover {
  background-color: var(--secondary-color);
  box-shadow: var(--shadow-dark);
}

button:disabled {
  background-color: #d1d5db;
  cursor: not-allowed;
}
```

Menambahkan desain tombol

```
a {
  color: var(--primary-color);
  text-decoration: none;
  font-weight: 600;
  transition: color 0.3s ease;
}

a:hover {
  color: var(--secondary-color);
}
```

Menambahkan desain link

```
h1, h2, h3, h4, h5, h6 {
  font-family: 'Poppins', sans-serif;
  font-weight: 700;
  color: var(--foreground);
  text-transform: uppercase;
  letter-spacing: 0.05em;
}
```

Mengatur header dengan efek


```

.card {
  background-color: var(--card-bg);
  border-radius: 12px;
  box-shadow: var(--shadow-light);
  padding: 20px;
  margin-bottom: 20px;
  transition: box-shadow 0.3s ease, transform 0.3s ease;
}

.card:hover {
  box-shadow: var(--shadow-dark);
  transform: translateY(-5px);
}

```

Menambahkan Kartu dengan bayangan dan sudut melengkung

```

.grid-container {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));
  gap: 20px;
  padding: 20px;
}

.grid-item {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  padding: 20px;
  border-radius: 12px;
  box-shadow: var(--shadow-light);
  background-color: var(--card-bg);
  transition: box-shadow 0.3s ease, transform 0.3s ease;
}

.grid-item:hover {
  box-shadow: var(--shadow-dark);
  transform: translateY(-5px);
}

```

Menambahkan grid responsive untuk layout