

Praktikum 4 - Matakuliah Pilihan 1 (Web)

Program Studi: Teknik Informatika

Lakukan praktikum dibawah ini, dan buat screenshot untuk pembuktian mengerjakan setiap poin dengan mengisi tabel dibawah, kemudian tunjukan hasil akhir dari men-share repository github yang telah dibuat.

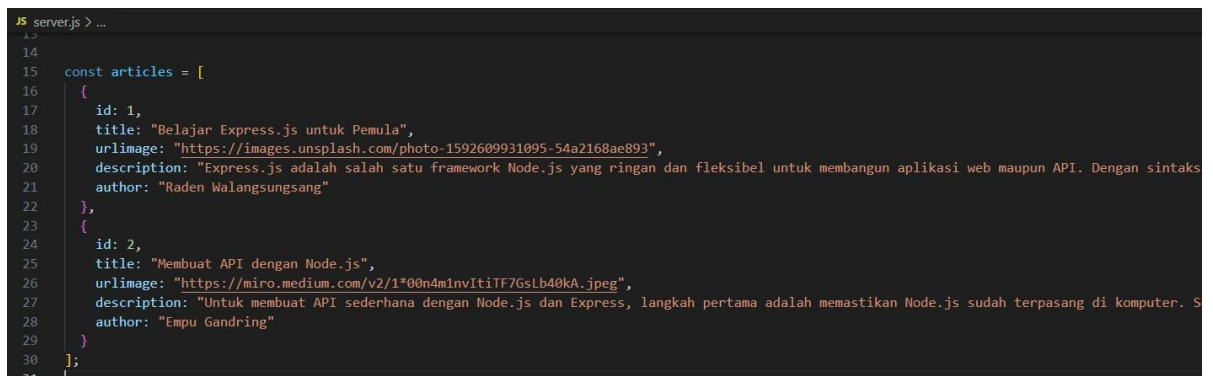
A. Membuat Endpoint test untuk artikel dengan Express.js

1. Lakukan langkah yang sama seperti praktikum 3, menyiapkan express js (npm init, npm install express, buat file [server.js](#))
2. Buat file [server.js](#) dengan port di 8001, tidak perlu ada router. Seperti pada gambar dibawah ini



```
JS server.js X
JS server.js > ...
1 // server.js
2 const express = require("express");
3 const app = express();
4 const port = 8001;
5
6 app.listen(port, () => {
7   console.log(`Server berjalan di http://localhost:${port}`);
8 });
9
```

3. Pada file [server.js](#) tambahkan array-objects dengan nama articles seperti pada gambar dibawah ini



```
JS server.js > ...
13
14
15 const articles = [
16   {
17     id: 1,
18     title: "Belajar Express.js untuk Pemula",
19     urlimage: "https://images.unsplash.com/photo-1592609931095-54a2168ae893",
20     description: "Express.js adalah salah satu framework Node.js yang ringan dan fleksibel untuk membangun aplikasi web maupun API. Dengan sintaks",
21     author: "Raden Walangsungsang"
22   },
23   {
24     id: 2,
25     title: "Membuat API dengan Node.js",
26     urlimage: "https://miro.medium.com/v2/1*00n4m1nvItiTF7GsLb40kA.jpeg",
27     description: "Untuk membuat API sederhana dengan Node.js dan Express, langkah pertama adalah memastikan Node.js sudah terpasang di komputer. S",
28     author: "Empu Gandring"
29   }
30 ];
31
```

4. Untuk urlimage, dan description bisa cari berita dari web media lainnya, tidak harus sama.
5. Kemudian tambahkan endpoints baru (/api/test/getarticle) untuk mengakses artikel

```

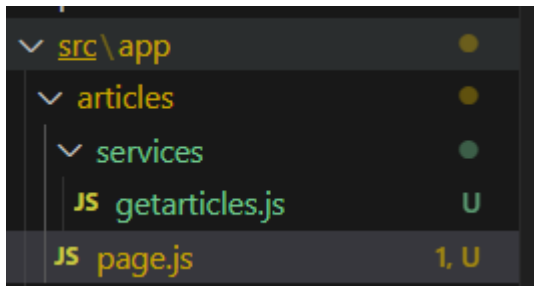
31
32 // Endpoint GET /api/test/getarticle
33 app.get("/api/test/getarticle", (req, res) => {
34   res.json({
35     status: "success",
36     data: articles
37   });
38 });

```

6. Jalankan [server.js](#) dan pastikan bisa mengakses /api/test/getarticle

B. Membuat Service mengambil data dari API Endpoints di Next.js

1. Buat folder baru Latihan4 didalamnya install [next.js](#) seperti pada praktikum 2
Ketik : `npx create-next-app@latest praktikum2`
2. Buka terminal baru, Jalankan dengan **npm run dev**.
3. (Pastikan [server.js](#) pada point A tidak dihentikan/distop)
Port 8001 untuk API / Express
Port 8000 untuk Frontend / Nextjs
4. Buat folder articles di dalam src/app sehingga menjadi /src/app/articles
5. Buat folder [page.js](#) kosong di dalamnya sehingga menjadi /src/app/articles/[page.js](#)
6. Buat folder services di dalam folder articles sehingga menjadi /src/app/articles/services, kemudian di dalamnya buat file getarticles.js
7. Pastikan folder hierarchy seperti pada gambar dibawah ini



8. Pada file [getarticles.js](#) buat kode program seperti ini, file ini berfungsi untuk mengambil data ke api endpoints dengan menggunakan nextjs

```

src > app > articles > services > JS getarticles.js > ...
1  export async function getArticles() {
2    try {
3      const res = await fetch("http://localhost:8001/api/test/getarticle");
4      if (!res.ok) {
5        throw new Error("Gagal mengambil data artikel");
6      }
7      const data = await res.json();
8      return data.data; // sesuai struktur di server.js { status, data }
9    } catch (error) {
10     console.error("Error fetching articles:", error);
11     return [];
12   }
13 }

```

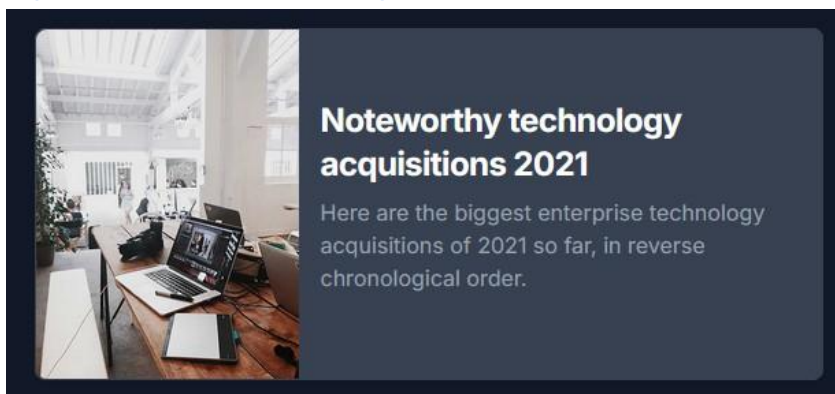
9. Pastikan API bisa diakses lewat browser.
10. Kemudian pada src/app/articles/[page.js](#) ketik kode program seperti dibawah ini

```
src > app > articles > JS page.js > ArticlesPage > articles.map() callback
1  import { getArticles } from "../services/getarticles";
2
3  export default async function ArticlesPage() {
4    const articles = await getArticles()
5
6    return (
7      <div className="p-4">
8        <h1 className="text-xl font-bold mb-4">Daftar Artikel</h1>
9        <ul className="space-y-2">
10         {articles.map((article) => (
11           <li key={article.id} className="border p-3 rounded">
12             <h2 className="font-semibold">{article.title}</h2>
13             <img src={article.urlimage} alt={article.title} className="w-40 my-2" />
14             <p>{article.description}</p>
15             <small className="text-gray-500">By: {article.author}</small>
16           </li>
17         )>
18       </ul>
19     </div>
20   );
21 }
```

11. Kode program diatas untuk menampilkan artikel yang diambil dari API services dengan nama getArticles()
12. Setelah kedua program selesai dibuat, cek di browser dengan mengunjungi link <http://localhost:8000/articles>

C. Experiment 1

1. Ganti namafile untuk [page.js](#) dan [getarticles.js](#) dengan ekstensi *.tsx sehingga menjadi page.tsx dan getarticles.tsx, kemudian apakah yang terjadi?
2. Ambil component flowbite Horizontal Card pada link berikut <https://flowbite.com/docs/components/card/>



3. Modifikasi src/app/articles/page.tsx dengan mengganti baris kode dengan flobite horizontal components.

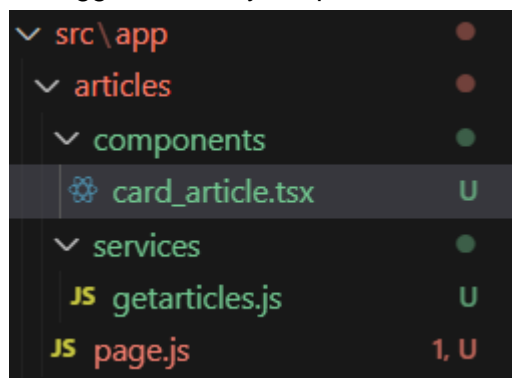
```
<li key={article.id} className="border p-3 rounded">
  ...
</li>
```

D. Experiment 2

1. Gunakan Dynamic components seperti pada Praktikum 2. Dengan menempatkan flowbite ke folder components di dalam /src/app/articles/components
2. Kemudian pada page.tsx cukup dipanggil saja dengan di dalam panggil nama komponennya dan mengirim parameter seperti pada gambar dibawah ini

```
{articles.map((article) => (  
  <li key={article.id} className="border p-3 rounded">  
    <Card img={article.urlimage} title={article.urlimage} desc={article.description} author={article.author} />  
  </li>  
))}
```

3. Pastikan dynamic component tersimpan di folder components dibawah articles sehingga strukturnya seperti ini:



E. Github + Visual Code

1. Buat proyek di Github dengan nama **Latihan4**

git init

git add .

git commit -m "first commit"

git branch -M main

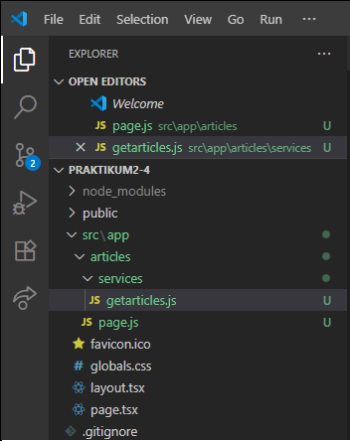
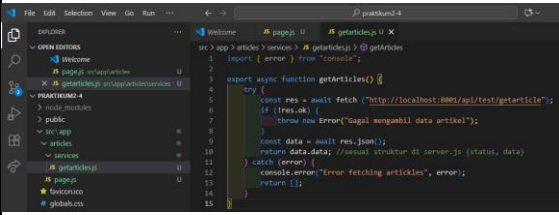
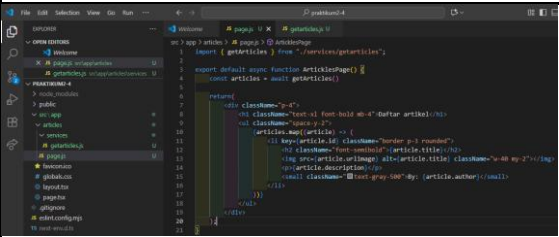
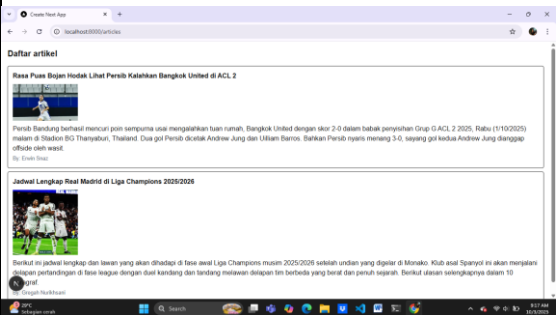
git remote add origin https://github.com/agunghakase/Latihan4.git

git push -u origin main

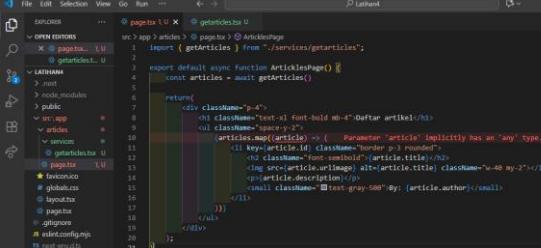
Hasil Pengerjaan

No.	Instruksi	Screenshot	Kendala/Saran
A.	Instalasi dan Konfigurasi		
1.	<p>1.Lakukan langkah yang sama seperti praktikum 3, menyiapkan express js (npm init, npm install express, buat file server.js)</p> <p>2.Buat file server.js dengan port di 8001, tidak perlu ada router. Seperti pada gambar dibawah ini</p> <p>3.Pada file server.js tambahkan array-objects dengan nama articles seperti pada gambar dibawah ini</p> <p>4.Untuk urlimage, dan description bisa cari berita dari web media lainnya, tidak harus sama.</p> <p>5.Kemudian tambahkan endpoints baru (/api/test/getarticle) untuk mengakses artikel</p> <p>6. Jalankan server.js dan pastikan bisa mengakses /api/test/getarticle</p>	 <pre> C:\WINDOWS\system32\cmd. X + v C:\Users\HP\APIpraktikum4titho>npm init -y Wrote to C:\Users\HP\APIpraktikum4titho\package.json: { "name": "apipraktikum4titho", "version": "1.0.0", "description": "", "main": "index.js", "scripts": { "test": "echo \"Error: no test specified\" && exit 1" }, "keywords": [], "author": "", "license": "ISC" } C:\Users\HP\APIpraktikum4titho>npm install express added 68 packages, and audited 69 packages in 7s 16 packages are looking for funding run 'npm fund' for details found 0 vulnerabilities C:\Users\HP\APIpraktikum4titho> </pre>  <pre> 1 // server.js 2 const express = require("express"); 3 const app = express(); 4 const port = 8001; 5 6 app.listen(port, () => { 7 console.log(`Server berjalan di http://localhost:\${port}`); 8 }); </pre>  <pre> 11 const articles = [12 { 13 id: 1, 14 title: "Rasa Puan Bujur Mada Lihat Peris Kelatan Bungk United di KOL F", 15 urlimage: "https://cdn-production-images.kly.akamai.net/mPw0x5h6gphub", 16 description: "Peris Bandung berhasil mencuri poin sempurna usai mengalahkan t", 17 author: "Erwin Sma", 18 }, 19 { 20 id: 2, 21 title: "Jadwal Lengkap Real Madrid di Liga Champions 2025/2026", 22 urlimage: "https://cdn-production-images.kly.akamai.net/v5e1pD68avg374e", 23 description: "Berikut ini jadwal lengkap dan lawan yang akan dihadapi di fase", 24 author: "Gregah Nurikhsani", 25 }, 26]; </pre>  <pre> 10 const articles = [11 // ... 12]; 13 14 // Endpoint GET /api/test/getarticle 15 app.get("/api/test/getarticle", (req, res) => { 16 res.json({ 17 status: "success", 18 data: articles 19 }); 20 }); </pre>  <pre> { "status": "success", "data": [{ "id": 1, "title": "Rasa Puan Bujur Mada Lihat Peris Kelatan Bungk United di KOL F", "urlimage": "https://cdn-production-images.kly.akamai.net/mPw0x5h6gphub", "description": "Peris Bandung berhasil mencuri poin sempurna usai mengalahkan t", "author": "Erwin Sma" }, { "id": 2, "title": "Jadwal Lengkap Real Madrid di Liga Champions 2025/2026", "urlimage": "https://cdn-production-images.kly.akamai.net/v5e1pD68avg374e", "description": "Berikut ini jadwal lengkap dan lawan yang akan dihadapi di fase", "author": "Gregah Nurikhsani" }] } </pre>	Tidak ada

<p>2.</p>	<p>1.Buat folder baru Latihan4 didalamnya install next.js seperti pada praktikum 2 Ketik : npx create-next-app@latest praktikum2</p> <p>2.Buka terminal baru, Jalankan dengan npm run dev.</p> <p>3.(Pastikan server.js pada point A tidak dihentikan/distop) Port 8001 untuk API / Express Port 8000 untuk Frontend / Nextjs</p>	 <pre> C:\WINDOWS\system32\cmd. X + v Microsoft Windows [Version 10.0.26100.6584] (c) Microsoft Corporation. All rights reserved. C:\Users\HP>npx create-next-app@latest praktikum2-4 Need to install the following packages: create-next-app@15.5.4 Ok to proceed? (y) y ✔ Would you like to use TypeScript? ... No / Yes ✔ Which linter would you like to use? » ESLint ✔ 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? (recommended) ... No / Yes ✔ Would you like to customize the import alias '@/*' by default? ... No / Yes Creating a new Next.js app in C:\Users\HP\praktikum2-4. Using npm. Initializing project with template: app-tw Installing dependencies: - react - react-dom - next Installing devDependencies: - typescript - @types/node - @types/react - @types/react-dom - @tailwindcss/postcss - tailwindcss - eslint C:\WINDOWS\system32\cmd. X + v Installing dependencies: - react - react-dom - next Installing devDependencies: - typescript - @types/node - @types/react - @types/react-dom - @tailwindcss/postcss - tailwindcss - eslint - eslint-config-next - @eslint/eslintrc added 331 packages, and audited 332 packages in 1m 135 packages are looking for funding run `npm fund` for details found 0 vulnerabilities Initialized a git repository. Success! Created praktikum2-4 at C:\Users\HP\praktikum2-4 C:\Users\HP> C:\Users\HP\praktikum2-4>npm run dev -- -p 8000 > praktikum2-4@0.1.0 dev > next dev --turbo -p 8000 ▲ Next.js 15.5.4 (Turbopack) - Local: http://localhost:8000 - Network: http://10.172.142.246:8000 ✓ Starting... ✓ Ready in 2.1s ○ Compiling /articles ... ✓ Compiled /articles in 5.4s GET /articles 200 in 6585ms ○ Compiling /favicon.ico ... ✓ Compiled /favicon.ico in 540ms GET /favicon.ico?favicon.0b3bf435.ico 200 in 1144ms P5 C:\Users\HP\APIpraktikum4titho> node server.js Server berjalan di http://localhost:8001 </pre>	<p>Tidak ada</p>
-----------	---	--	------------------

	<p>4. Buat folder articles di dalam src/app sehingga menjadi /src/app/articles</p> <p>5. Buat folder page.js kosong di dalamnya sehingga menjadi /src/app/articles/page.js</p> <p>6. Buat folder services di dalam folder articles sehingga menjadi /src/app/articles/services, kemudian di dalamnya buat file getarticles.js</p> <p>7. Pastikan folder hierarchy seperti pada gambar dibawah ini.</p> <p>8. Pada file getarticles.js buat kode program seperti ini, file ini berfungsi untuk mengambil data ke api endpoints dengan menggunakan nextjs</p>	 	
<p>3.</p>	<p>9. Pastikan API bisa diakses lewat browser.</p> <p>10. Kemudian pada src/app/articles/page.js ketik kode program seperti dibawah ini</p> <p>11. Kode program diatas untuk menampilkan artikel yang diambil dari API services dengan nama getArticles()</p> <p>12. Setelah kedua program selesai dibuat, cek di browser dengan mengunjungi link http://localhost:8000/articles</p>	 <pre> C:\Users\HP\praktikum2-4>npm run dev -- -p 8000 > praktikum2-4@0.1.0 dev > next dev --turbo --port 8000 ▲ Next.js 15.5.4 (Turbo) - Local: http://localhost:8000 - Network: http://10.172.142.246:8000 ✓ Starting... ✓ Ready in 2.1s ○ Compiling /articles ... ✓ Compiled /articles in 5.4s GET /articles 200 in 6585ms ○ Compiling /favicon.ico ... ✓ Compiled /favicon.ico in 540ms GET /favicon.ico?favicon.0b3bf435.ico 200 in 1144ms </pre> <p>npm run dev -- -p 8000</p> 	<p>Tidak ada</p>

1. Ganti namafile untuk page.js dan getarticles.js dengan ekstensi *.tsx sehingga menjadi page.tsx dan getarticles.tsx, kemudian apakah yang terjadi?



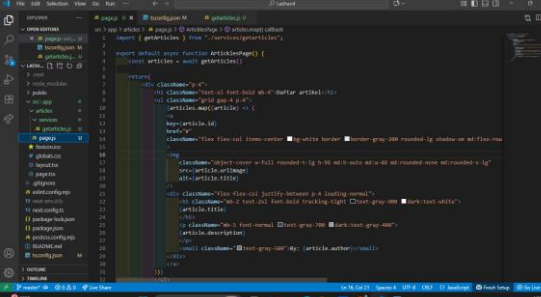
```
import { useState } from 'react';
import { useRouter } from 'next/router';
import { getArticles } from '../services/getarticles';

export default function ArticlePage() {
  const [articles, setArticles] = useState<Article[]>();

  return (
    <div className="p-4">
      <h2 className="text-2xl font-bold mb-4">Daftar Artikel</h2>
      <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-4">
        {articles.map((article) => (
          <div key={article.id} className="border p-3 rounded">
            <div className="font-size-16">{article.title}</div>
            <img src={article.urlImage} alt={article.title} className="w-40 h-20" />
            <p>{article.description}</p>
            <p>{article.author}</p>
          </div>
        ))}
      </div>
    </div>
  );
}
```

Setelah mengganti file page.js dan getarticles.js menjadi page.tsx dan getarticles.tsx, Next.js otomatis menggunakan TypeScript. Akibatnya, kode yang tidak memiliki deklarasi tipe data akan menampilkan error. Solusinya, perlu menambahkan definisi tipe seperti type Article agar kode bisa dijalankan tanpa error.

2. Ambil component flowbite Horizontal Card pada link berikut <https://flowbite.com/docs/components/card/>

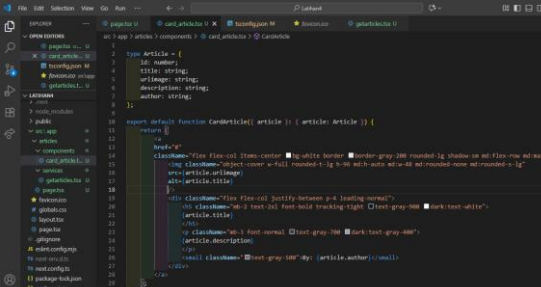
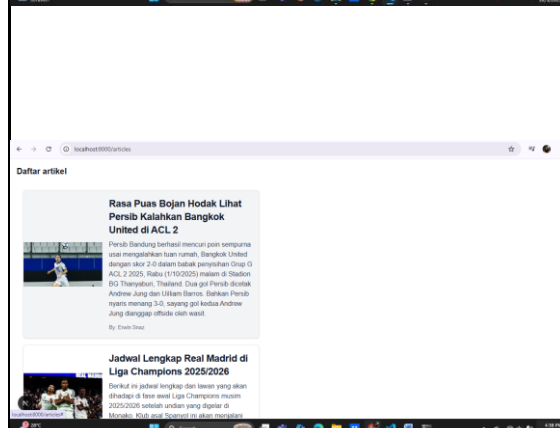


```
import { useState } from 'react';
import { useRouter } from 'next/router';
import { getArticles } from '../services/getarticles';

export default function ArticlePage() {
  const [articles, setArticles] = useState<Article[]>();

  return (
    <div className="p-4">
      <h2 className="text-2xl font-bold mb-4">Daftar Artikel</h2>
      <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-4">
        {articles.map((article) => (
          <div key={article.id} className="border p-3 rounded">
            <div className="font-size-16">{article.title}</div>
            <img src={article.urlImage} alt={article.title} className="w-40 h-20" />
            <p>{article.description}</p>
            <p>{article.author}</p>
          </div>
        ))}
      </div>
    </div>
  );
}
```

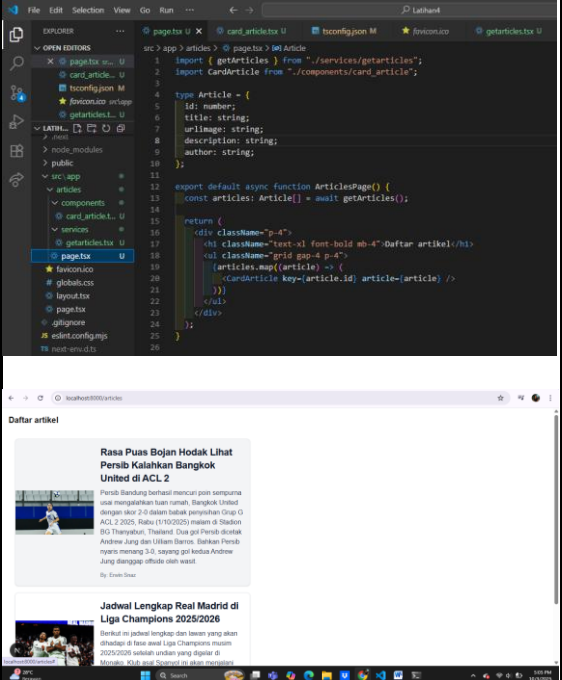
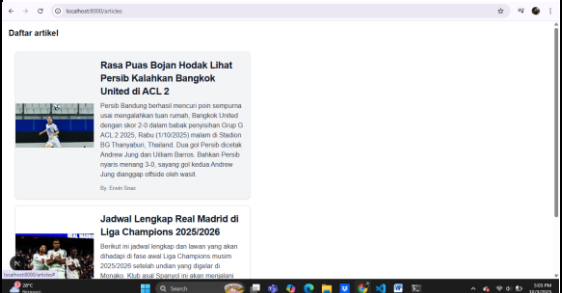
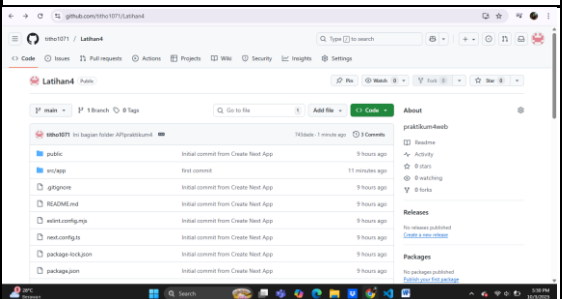
3. Modifikasi src/app/articles/page.tsx dengan mengganti baris kode dengan flobite horizontal components.



```
import { useState } from 'react';
import { useRouter } from 'next/router';
import { getArticles } from '../services/getarticles';

export default function ArticlePage() {
  const [articles, setArticles] = useState<Article[]>();

  return (
    <div className="p-4">
      <h2 className="text-2xl font-bold mb-4">Daftar Artikel</h2>
      <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-4">
        {articles.map((article) => (
          <div key={article.id} className="border p-3 rounded">
            <div className="font-size-16">{article.title}</div>
            <img src={article.urlImage} alt={article.title} className="w-40 h-20" />
            <p>{article.description}</p>
            <p>{article.author}</p>
          </div>
        ))}
      </div>
    </div>
  );
}
```


<p>1. Gunakan Dynamic components seperti pada Praktikum 2. Dengan menempatkan flowbite ke folder components di dalam /src/app/articles/components</p> <p>2. Kemudian pada page.tsx cukup dipanggil saja dengan di dalam panggil nama komponennya dan mengirim parameter seperti pada gambar dibawah ini</p> <p>3. Pastikan dynamic component tersimpan di folder components dibawah articles sehingga strukturnya seperti ini:</p>		
<p>B.</p>	<p>Github dan Viscode</p>	
<p>1.</p>	<p>Tidak ada</p>	<p>https://github.com/titho1071/Latihan4.git</p>
<p>2.</p>		
<p>...</p>		