

## PERTEMUAN 1

### PENGANTAR PEMROGRAMAN BERBASIS FRAMEWORK DAN REACTJS

#### 1. Tujuan

Setelah menyelesaikan pertemuan ini mahasiswa diharapkan mampu:

1. Mengetahui konsep pemrograman berbasis framework
2. Mengetahui konsep modern JavaScript
3. Melakukan *set up* lingkungan pengembangan pada perangkat masing-masing
4. Mengetahui konsep ReactJS

#### 2. Pengenalan Pemrograman Berbasis Framework

##### Apa itu *Framework*?

*Framework* secara bahasa dalam [kamus Oxford](#) memiliki beberapa pengertian, yaitu:

[1] *the parts of a building or an object that support its weight and give it shape.*

[2] *a set of beliefs, ideas or rules that is used as the basis for making judgements, decisions, etc.*

Dari pengertian nomor satu menjelaskan bahwa *framework* adalah bagian dari sebuah bangunan atau sebuah objek yang mendukung bobot dan memberikan suatu bentuk. Pengertian dari nomor dua menjelaskan bahwa *framework* adalah seperangkat panutan, ide atau aturan-aturan yang digunakan sebagai dasar untuk membuat hukum, keputusan, dsb.

*Framework* (kerangka kerja) adalah sebuah kerangka program yang digunakan untuk membantu *developer* dalam mengembangkan kode secara konsisten. Adanya *framework developer* bisa mengurangi jumlah bug pada aplikasi yang dibuat. Karena, fungsi dan variabel yang sudah tersedia di dalam komponen *framework* tersebut.

Sebelum belajar lebih jauh tentang *framework*, ada baiknya jika Anda memahami pengertian *framework* terlebih dahulu. Sesuai dengan namanya, *framework* adalah kerangka kerja yang digunakan untuk mengembangkan aplikasi berbasis desktop atau aplikasi berbasis website.

*Framework* sengaja diciptakan untuk membantu developer mengembangkan aplikasi lebih cepat serta tersusun dan terstruktur. Dengan menggunakan *framework* Anda akan lebih mudah untuk membuat aplikasi, karena Anda hanya perlu menyusun komponen-komponen pemrograman yang sudah jadi. Sehingga developer dan programmer tidak perlu melakukan koding program yang diulang-ulang.

## Model Pengembangan Web Berbasis Arsitektur Frontend – Backend

Dalam dunia pengembangan web modern, model Frontend dan Backend menjadi standar utama. Model ini memungkinkan pemisahan tanggung jawab antara antarmuka pengguna (*frontend*) dan logika bisnis serta manajemen data (*backend*). Frontend-Backend Architecture adalah pendekatan modular di mana frontend dan backend dikembangkan sebagai bagian terpisah. *Frontend* menangani tampilan UI/UX dan interaksi pengguna, sedangkan *backend* menangani pengolahan data, autentikasi, dan komunikasi dengan database.

Tabel 1. Perbedaan Model Frontend-Backend

Fitur	Frontend-Backend (Modern)	Monolitik (Lama)
Modularitas	Frontend & backend dapat dikembangkan secara terpisah	Semua dalam satu proyek, sulit dipisahkan
Skalabilitas	Backend bisa dioptimalkan untuk performa tinggi	Kinerja rendah pada skala besar
Fleksibilitas Teknologi	Frontend bisa React/Next.js, backend bisa Node.js/Python	Terbatas pada satu stack teknologi
Keamanan	Backend dapat diisolasi dari frontend	Rentan terhadap eksploitasi karena backend dan UI digabung

Tabel 2. Teknologi Frontend

Teknologi	Fungsi
HTML5	Struktur dasar halaman web
CSS3	Styling dan desain tampilan
JavaScript (ES6+)	Interaktivitas halaman
React.js / Next.js	Framework untuk membangun UI modern
Tailwind CSS	Framework CSS untuk styling cepat
Redux / Zustand	State management untuk menyimpan data di frontend

Tabel 3. Teknologi Backend

Teknologi	Fungsi
<b>Node.js / Express.js</b>	Menjalankan backend dengan JavaScript
<b>Django (Python)</b>	Framework backend untuk aplikasi berbasis data
<b>Laravel (PHP)</b>	Framework PHP yang populer
<b>PostgreSQL / MongoDB</b>	Database untuk menyimpan data
<b>JWT / OAuth</b>	Autentikasi pengguna
<b>REST API / GraphQL</b>	Menghubungkan frontend dengan backend

## Apa yang Perlu Disiapkan

Beberapa hal yang perlu disiapkan dalam melakukan praktikum matakuliah ini antara lain:

- PC atau Laptop dengan spesifikasi minimum Web Developer
- Koneksi internet
- Chrome Browser
- Git
- VS Code
- Node.js

## Pengetahuan yang Diperlukan

Beberapa pengetahuan sebelumnya yang akan membantu dalam mempelajari materi dalam matakuliah ini antara lain:

- Pemrograman Dasar
- Pengetahuan dasar tentang HTML, CSS, JavaScript, dan [Chrome DevTools](#).
- Standar instalasi dan konfigurasi perangkat lunak seperti Git, Visual Studio Code (VS Code), dan Node.js

### 3. Pengenalan Modern JavaScript

Modern JavaScript merujuk pada versi JavaScript yang diperbarui sejak ECMAScript 6 (ES6) pada tahun 2015, hingga versi terbaru yang dirilis setiap tahunnya (ES7, ES8, ES9, hingga ESNext).

Sebelum ES6, JavaScript memiliki banyak keterbatasan, seperti:

- Keterbatasan dalam variabel (hanya var) → Solusi: let dan const.
- Tidak ada modularisasi → Solusi: import/export.
- Callback hell (callback dalam callback) → Solusi: async/await.
- Kode lebih panjang dan sulit dibaca → Solusi: Arrow functions & Destructuring.

Dari penjelasan singkat ini, diketahui bahwa modern JavaScript lebih ringkas, lebih aman, dan lebih efisien dibandingkan versi lama. Beberapa diantara fitur utama di ES6 antara lain:

#### 1. Variabel `let` dan `const`

Sebelum ES6 (Versi Lama):

```
var name = "John"; // Masalah: Scope variabel tidak terbatas dalam blok
console.log(name);
```

Setelah ES6:

```
let name = "John"; // Hanya bisa diakses dalam blok {}
const age = 25; // Tidak bisa diubah
console.log(name, age)
```

#### 2. Template Literals:

Sebelum ES6 (Versi Lama):

```
let name = "John";
console.log("Hello, " + name + "!"); // Pakai tanda +
```

Setelah ES6:

```
console.log(`Hello, ${name}!`); // Pakai backtick ``
```

#### 3. Strict Mode:

Mode ini akan memastikan kode JavaScript berjalan dengan aturan yang lebih ketat. Mengaktifkan strict mode dapat mencegah kesalahan umum dan memperbaiki perilaku aneh pada kode.

```
"use strict";
```

#### 4. Arrow Function `(()=>{})`

Sebelum ES6 (Versi Lama):

```
function sayHello(name) {  
  return "Hello, " + name;  
}
```

Setelah ES6:

```
const sayHello = (name) => `Hello, ${name}`;  
console.log(sayHello("Alice")); // Output: Hello, Alice
```

#### 5. Destructuring untuk Mengambil Data dengan Mudah

Sebelum ES6 (Versi Lama):

```
const user = { name: "Alice", age: 22 };  
const name = user.name;  
const age = user.age;  
console.log(name, age);
```

Setelah ES6:

```
const user = { name: "Alice", age: 22 };  
const { name, age } = user; // Destructuring  
console.log(name, age);
```

## Framework JavaScript Modern

Framework seperti React, Angular, dan Vue.js menyediakan struktur, alat, dan komponen-komponen siap pakai untuk membangun aplikasi. Framework ini memanfaatkan fitur-fitur JavaScript modern untuk mempermudah pengembangan aplikasi.

```
// Contoh penggunaan React  
import React from 'react';  
// Komponen React berada di sini
```

## 4. Pengenalan ReactJS

### Pengenalan ReactJS Framework

React adalah sebuah library JavaScript yang dikembangkan dan dikelola oleh Facebook dan Instagram untuk menampilkan data dan membuat komponen-komponen User Interface.

Menurut layanan analitik Libscore, saat ini React digunakan oleh Netflix, Imgur, Bleacher Report, Feedly, Airbnb, SeatGeek, HelloSign, Walmart, dan lain-lain (Wikipedia).

## Konsep Atomic Design Methodology (ADM)

We're not designing pages, we're designing systems of components. —Stephen Hay

Dalam konsep ADM, komponen dapat diumpakan sebagai atom. Kita menggabungkan atom-atom menjadi molekul, molekul menjadi organisme, organisme menjadi template, template menjadi halaman-halaman. React sangat cocok diaplikasikan ke dalam konsep pengembangan ADM karena dalam React kita memulai dari mengembangkan komponen-komponen yang kita gabungkan untuk membentuk sebuah halaman. Keterangan lebih lanjut mengenai ADM dapat dibaca di <https://bradfrost.com/blog/post/atomic-web-design/>

Beberapa fitur-fitur yang dimiliki React adalah sebagai berikut:

1. Deklaratif

React membuat proses pembuatan antarmuka pengguna interaktif menjadi lebih mudah. Buat tampilan sederhana untuk setiap state di aplikasi Anda, dan React akan secara efisien memperbarui dan me-render hanya komponen yang diperlukan ketika data Anda berubah. Tampilan yang deklaratif membuat jalannya kode Anda menjadi lebih terprediksi dan lebih mudah untuk di-debug.

2. Berbasis Komponen

Bangun komponen-komponen terenkapsulasi yang mengatur state-nya sendiri, kemudian gabungkan mereka untuk membentuk antarmuka pengguna yang lebih kompleks. Karena logika pada komponen React ditulis langsung menggunakan JavaScript (bukan menggunakan templat), Anda dapat dengan mudah mengoper data melalui aplikasi Anda dan menempatkan state agar tetap berada di luar DOM.

3. Belajar sekali, tulis di mana saja

Kami tidak berasumsi mengenai teknologi yang Anda gunakan di dalam stak Anda, jadi Anda dapat mengembangkan fitur baru menggunakan React tanpa menulis ulang kode yang sudah ada. React juga dapat digunakan untuk me-render di server menggunakan Node dan membangun aplikasi mobile menggunakan React Native.

## Framework React

Jika Anda ingin membuat aplikasi baru atau situs web baru dengan menggunakan React, kami merekomendasikan Anda untuk memilih salah satu dari framework berbasis React yang populer di komunitas. framework menyediakan fitur-fitur yang sebagian besar aplikasi dan situs web butuhkan, termasuk routing, pengambilan data, dan pembuatan halaman HTML.

Beberapa Framework React antara lain: **Next.js, Remix, Gatsby, Expo**

## Apakah bisa menggunakan React tanpa menggunakan framework?

Anda dapat menggunakan React tanpa menggunakan framework—begitupun Anda dapat menggunakan React untuk sebagian dari halaman proyek Anda. Namun, jika Anda ingin membuat aplikasi atau situs baru sepenuhnya menggunakan React, kami merekomendasikan Anda untuk **menggunakan framework**.

Menggunakan framework dalam proyek React **sangat disarankan** karena seiring waktu, kebutuhan proyek akan semakin kompleks, terutama dalam hal manajemen kode, pemecahan bundel JavaScript, pengambilan data, serta optimasi performa untuk berbagai kondisi jaringan dan perangkat. Framework React membantu mengatasi tantangan ini dengan menyediakan integrasi yang baik antara bundler, router, dan library pengambilan data, serta memungkinkan rendering di server atau saat build untuk meningkatkan kecepatan tampilan awal.

Selain itu, framework menawarkan **struktur kode yang lebih baik**, dukungan komunitas yang aktif, serta peningkatan berkelanjutan. Tanpa framework, pengembang berisiko terjebak dalam dependensi usang atau bahkan tanpa sadar membangun framework mereka sendiri dengan desain yang kurang optimal. Oleh karena itu, framework React memberikan solusi yang lebih efisien dan berkelanjutan dibandingkan dengan pengaturan manual.

## 5. Format Laporan Praktikum

Setiap mahasiswa selesai melaksanakan praktikum, wajib membuat laporan tertulis berupa file `README.md` (disertai bukti screenshot hasil praktik) yang merupakan bagian dari project pada tiap pertemuan. Kemudian di push ke akun GitHub masing-masing dengan nama repository yang disepakati oleh dosen pengampu. Selanjutnya, link repository tersebut dikumpulkan ke dosen sesuai media yang disepakati. Anda dapat membuat repository baru melalui tautan ini: <https://github.com/new> Perhatikan gambar di bawah ini untuk membuat repository baru.

### Perhatian:

Bagian-bagian yang perlu diisi dan disesuaikan setiap praktikum tiap minggunya.

- **Repository name:** isi dengan format **#01-nama-materi** → 01 berarti pertemuan pekan ke-1. Ini akan menjadi tautan yang dikumpulkan ke dosen.
- **Description:** isi dengan **NIM - Nama - Pem. Berbasis Framework 2024**
- Pastikan pilih **Public** agar dosen dapat mengakses repo Anda.
- Tidak perlu menambahkan file README di sini, karena tiap project React nanti sudah ada.

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---

*Required fields are marked with an asterisk (\*).*


**Repository template**

No template ▾

Start your repository with a template repository's contents.

---

**Owner \*** **Repository name \***

 jti-polinema ▾ / #01-setup-environment-re


✓ Your new repository will be created as -01-setup-environment-react.  
The repository name can only contain ASCII letters, digits, and the characters -, ., and \_.


Great repository names are short and memorable. Need inspiration? How about [supreme-waddle](#) ?

**Description (optional)**

NIM - Nama - Pem. Berbasis Framework 2024

---

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

---

**Initialize this repository with:**

☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

.gitignore template: None ▾


Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

**Choose a license**

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

---

 You are creating a public repository in the jti-polinema organization.

---

[Create repository](#)

**Catatan:** contoh laporan praktikum dapat diakses melalui tautan ini <https://github.com/jti-polinema/-01-contoh-laporan-react> Atau Anda dapat melihat versi raw dari file README tersebut melalui: <https://raw.githubusercontent.com/jti-polinema/-01-contoh-laporan-react/master/README.md>



## 6. Praktikum 1: Menyiapkan Lingkungan Pengembangan

### Instalasi Git

1. Jika Git belum terpasang pada komputer Anda, silakan unduh di web resminya: **<https://git-scm.com/>** kemudian lakukan instalasi seperti biasa sesuai langkah-langkah pada proses install aplikasi.

**Catatan:** Untuk memahami lebih baik tentang Git, Anda dapat mempelajarinya di tautan ini: <https://www.petanikode.com/git-untuk-pemula/>

2. Jika telah berhasil melakukan install Git, silakan verifikasi dengan membuka Command Prompt (CMD) atau Git Bash atau Terminal sesuai dengan sistem operasi Anda. Lalu ketikkan perintah berikut:

```
git -version
```

**Perhatian:** Pastikan versi Git telah tampil, jika tidak tampil maka perlu dicek variabel environment pada sistem operasi Anda.

3. Jika telah berhasil melakukan instalasi Git, maka perlu diset akun secara global untuk mengidentifikasi bahwa akun yang ingin digunakan adalah milik Anda. Lakukan perintah berikut pada CMD, Git Bash atau Terminal.

Ketik perintah `git config --global user.name "username Anda"` untuk menset username Anda secara global pada sistem.

Lalu perintah `git config --global user.email "email@email.com"` isi sesuai email yang Anda gunakan di akun GitHub.

**Catatan:** Jika Anda hanya ingin menset username dan email secara lokal, maka dapat menghapus sintaks `--global`. Perintah lainnya tetap sama. Namun harus dilakukan pada folder root repository Anda.

### Instalasi VS Code

Silakan unduh Visual Studio Code jika belum terinstal di komputer Anda melalui **<https://code.visualstudio.com/Download>**. VS Code ini nanti akan digunakan setiap kali

praktikum untuk mengembangkan aplikasi. Jika pada komputer Anda sudah terpasang VS Code, Anda dapat melewati langkah ini.

**Perhatian:** Silakan install sesuai sistem operasi yang Anda miliki. Pastikan Anda telah menyelesaikan proses instalasi dengan sukses sebelum melanjutkan ke langkah berikutnya.

## Instalasi NodeJS

Selanjutnya, Anda perlu memasang NodeJS sudah termasuk NPM (Node Package Manager) melalui <https://nodejs.org/en/download/>. Tool ini akan selalu digunakan untuk praktikum dalam satu semester ke depan.

**Perhatian:** Silakan install sesuai sistem operasi yang Anda miliki. Pastikan Anda telah menyelesaikan proses instalasi dengan sukses sebelum melanjutkan ke langkah berikutnya.

## Apa itu Node.js?

Menurut tautan ini (<https://nodejs.org/en/about>) yaitu *"As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications"*.

Dalam pemrograman komputer, pemrograman *event-driven* adalah paradigma pemrograman di mana aliran program ditentukan oleh peristiwa eksternal (*external event*). *Event* umumnya dapat berupa UI (*user interface*) seperti dari *mouse*, *keyboard*, *touchpad*, *touchscreen*, sensor, dan sebagainya.

Nodejs ini menggunakan bahasa pemrograman JavaScript yang bersifat *event-driven* asinkronus dan dirancang untuk membangun aplikasi berbasis jaringan yang bersifat *scalable*.

## Apa itu NPM (Node Package Manager)?

NPM (*Node Package Manager*) adalah pengelola package JavaScript yang merupakan bawaan dari Node.js. Dengan NPM, kita dapat:

- **Menginstal dan mengunduh package JavaScript** yang diperlukan untuk membangun aplikasi dengan JavaScript.
- **Mengelola package secara otomatis**, sehingga Anda tidak perlu lagi mengunduh dependency package secara manual atau memastikan versi package \.

NPM memiliki dua fungsi utama:

- **Sebagai Repository:** NPM berfungsi sebagai platform tempat orang dapat berbagi package JavaScript buatan mereka di npmjs.org. Anda juga dapat berkontribusi dengan package buatan Anda sendiri. Dengan fungsi ini, Anda bisa mendownload package yang diperlukan untuk mendukung proyek Anda kapan saja, tanpa harus membangun fitur dari nol.
- **Sebagai Command Line Interface (CLI):** NPM menyediakan antarmuka baris perintah (mirip dengan Command Prompt di Windows atau Terminal di Linux) untuk mengelola, mengunduh, dan mengunggah package JavaScript.

Jadi, NPM memudahkan Anda dalam mengelola package dan mempercepat proses pembuatan aplikasi.

### Pertanyaan Praktikum 1

1. Jelaskan kegunaan masing-masing dari Git, VS Code dan NodeJS yang telah Anda install pada sesi praktikum ini!
2. Buktikan dengan screenshot yang menunjukkan bahwa masing-masing tools tersebut telah berhasil terinstall di perangkat Anda!

## 7. Praktikum 2: Membuat Proyek Pertama React Menggunakan Next.js

1. Membuat folder proyek baru dengan nama **belajar-react**. Melalui konsol/command prompt/CMD masuk ke dalam folder tersebut dan jalankan perintah ini:

```
npx create-next-app
```

2. Buat proyek baru dengan nama hello-world seperti di bawah ini. Nama proyek ini perlu dimasukkan pertama kali melalui konsol.

```
D:\BAHAN-AJAR\PBF\Project\belajar-react>npx create-next-app
Need to install the following packages:
create-next-app@15.1.6
Ok to proceed? (y) y

✔ What is your project named? ... hello-world
✔ 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:\BAHAN-AJAR\PBF\Project\belajar-react\hello-world.
```

3. Buka folder proyek hello-world menggunakan VS Code. Masuk ke dalam folder proyek hello-world dengan perintah:

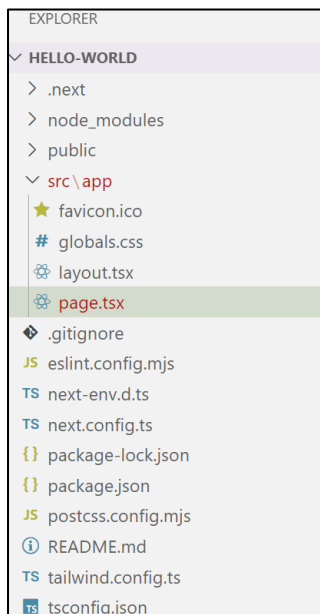
```
cd hello-world
```

Kemudian setelah masuk ke folder hello-world, masukkan perintah:

```
code .
```

Maka VS Code akan membuka project react Anda yang telah dibuat bernama **hello-world**.

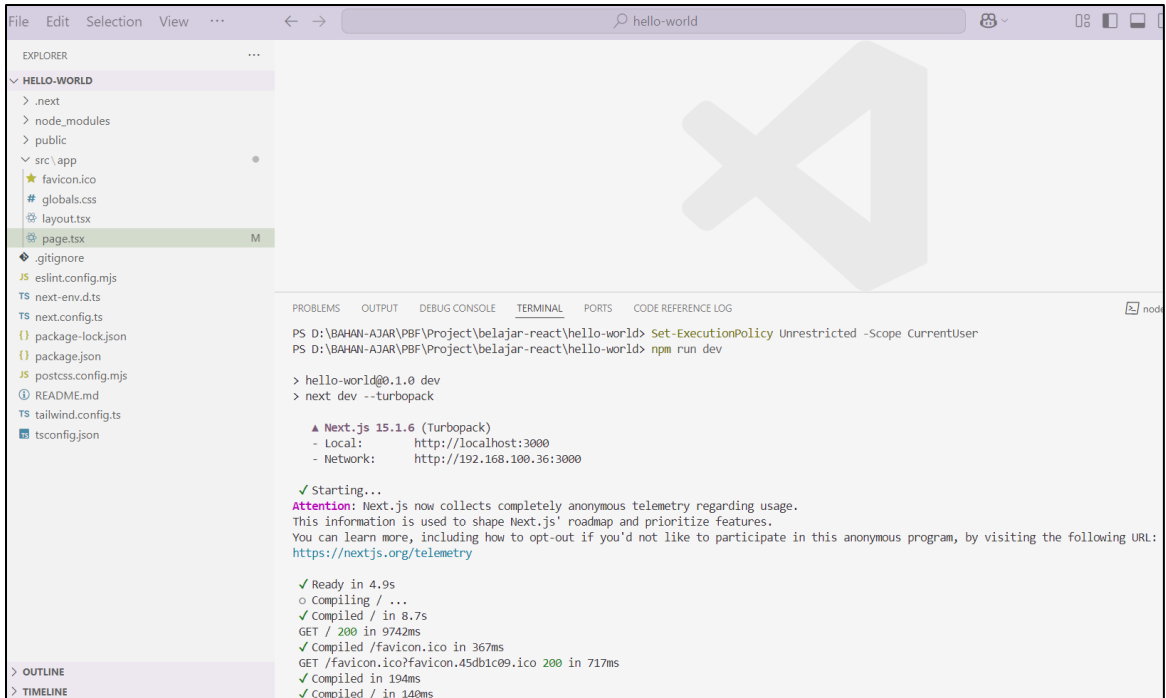
Dan akan menampilkan struktur folder proyek seperti di bawah ini.



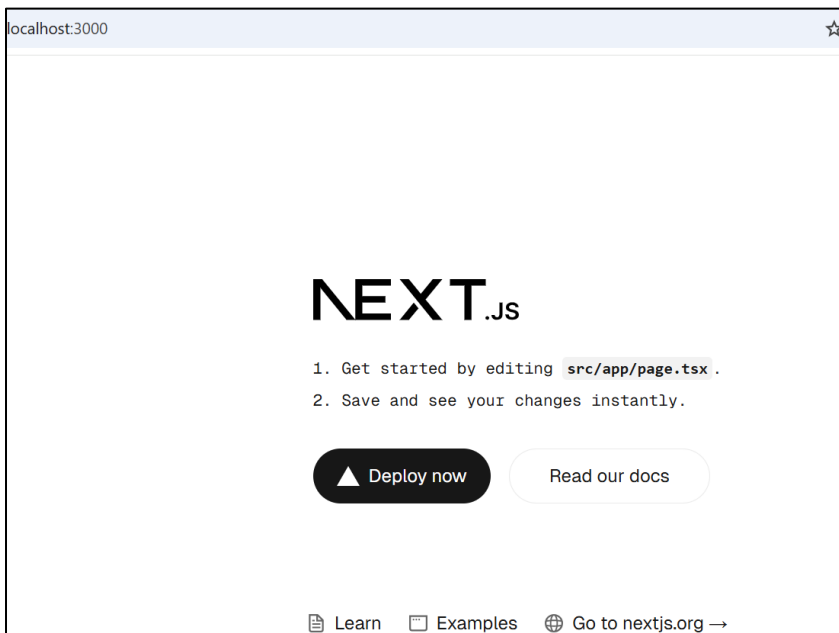
4. Running proyek hello-world dengan memasukkan perintah di bawah ini melalui konsol atau terminal di dalam VS Code.

```
npm run dev
```

Tunggu proses kompilasi hingga selesai. Lalu Anda dapat membuka alamat localhost di browser: <http://localhost:3000/>



Jika di browser telah tampil seperti gambar berikut ini, **Selamat!**



## Pertanyaan Praktikum 2

1. Pada Langkah **ke-2**, setelah membuat proyek baru menggunakan Next.js, terdapat beberapa istilah yang muncul. Jelaskan istilah tersebut, **TypeScript**, **ESLint**, **Tailwind CSS**, **App Router**, **Import alias**, **App router**, dan **Turbopack**!

2. Apa saja kegunaan folder dan file yang ada pada struktur proyek React yang tampil pada gambar pada tahap percobaan **ke-3**!
3. Buktikan dengan screenshot yang menunjukkan bahwa tahapan percobaan di atas telah berhasil Anda lakukan!

## 8. Praktikum 3: Menambahkan Komponen React (Button)

Aplikasi React dibuat dari **komponen**. Komponen adalah bagian dari UI (user interface, antarmuka pengguna) yang memiliki logika dan tampilan tersendiri. Sebuah komponen dapat berukuran sekecil tombol, atau sebesar seluruh halaman. **Komponen React adalah fungsi JavaScript yang mengembalikan markup.**

1. Di dalam folder proyek yang telah dibuka di VS Code, buka file `page.tsx`
2. Tambahkan fungsi `MyButton` yang mengembalikan markup komponen button yang akan ditambahkan ke dalam webpage

```
function MyButton() {  
  return (  
    <a  
      href="http://localhost:3000"  
      target="_blank"  
      rel="noopener noreferrer"  
      className="px-4 py-2 bg-blue-500 text-white rounded-lg hover:bg-blue-600 transition duration-300"  
    >  
      Buka Halaman  
    </a>  
  );  
}
```

3. Tambahkan komponen button tersebut di samping button Read Our Docs.

```
40 | Deploy now  
41 | </a>  
42 |  
43 | <a  
44 |   className="rounded-full b  
45 |   href="https://nextjs.org/  
46 |   target="_blank"  
47 |   rel="noopener noreferrer"  
48 | >  
49 |   Read our docs  
50 | </a>  
51 | <MyButton />  
   | </div>
```

Komponen button Read Our Docs

Penambahan komponen MyButton yang baru dibuat

Perhatikan bahwa komponen `MyButton` dimulai dengan huruf kapital. Dengan cara itulah Anda mengetahui bahwa itu adalah sebuah komponen React. Nama komponen React harus **selalu dimulai dengan huruf kapital**, sedangkan tag HTML harus menggunakan huruf kecil. Kata kunci `export default` menentukan komponen utama di dalam berkas (file).

4. Simpan perubahan dan coba lihat perubahan melalui web browser!

### Pertanyaan Praktikum 3

1. Buktikan dengan screenshot yang menunjukkan bahwa tahapan percobaan di atas telah berhasil Anda lakukan!

## 9. Praktikum 4: Menulis Markup dengan JSX

Sintaksis markup yang Anda lihat di atas disebut dengan JSX. JSX ini opsional, tetapi sebagian besar proyek React menggunakan JSX untuk kenyamanannya. Semua alat yang kami rekomendasikan untuk pengembangan lokal mendukung JSX secara langsung. JSX lebih ketat daripada HTML. Anda harus menutup tag seperti `<br />`. Komponen Anda juga tidak boleh mengembalikan beberapa tag JSX. Anda harus membungkusnya menjadi induk bersama (shared parent), seperti `<div>...</div>` atau sebuah pembungkus kosong `<>...</>`:

1. Tambahkan kode JSX di bawah ini ke dalam file `page.tsx`.

```
function Profile() {
  return (
    <>
      <div className="text-xs">Developed by: {user.name}</div>
      <img
        className="rounded-full"
        src={user.imageUrl}
        alt={'Foto ' + user.name}
        style={{
          width: user.imageSize,
          height: user.imageSize
        }}
      />
    </>
  );
}
const user = {
  name: 'Fulanah bin Fulan',
  imageUrl: 'https://i.imgur.com/yX0vdOSs.jpg',
```

```
    imageSize: 90,  
  };
```

2. Tambahkan komponen MyProfile setelah komponen MyButton.

```
47 | >  
48 |   Read our docs  
49 | </a>  
50 |   <MyButton />  
51 | </div>  
52 |   <Profile />  
53 | </main>
```

3. Simpan dan amati perubahan di halaman web yang dihasilkan!

#### Pertanyaan Praktikum 4

1. Untuk apakah kegunaan sintaks `user.imageUrl`?
2. Buktikan dengan screenshot yang menunjukkan bahwa tahapan percobaan di atas telah berhasil Anda lakukan!

#### Daftar Pustaka

1. Fielding, R. (2000). *RESTful API Design*.
2. Subramaniam, V. (2020). *GraphQL for Modern Web Applications*.
3. Keith, J. (2010). *HTML5 for Web Designers*.
4. Meyer, E. (2013). *CSS: The Definitive Guide, 4th Edition*.