

Tanggal Asistensi :	
Tanda Tangan	Nilai

MODUL 7

Deploy With Netlify

Nama : Imam Agil Aiman

Nim : F55123066

Kelas : B

2.1.7 *Deploy With Netlify*

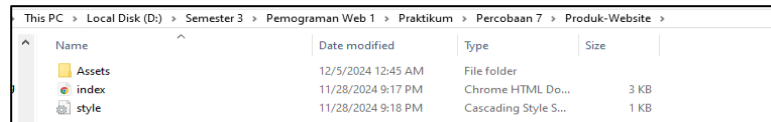
Netlify adalah platform *cloud* yang menyediakan layanan untuk meng-*host* dan men-*deploy* situs *web* dan aplikasi. Berikut adalah beberapa fitur dan fungsi utama dari Netlify:

1. *Hosting Statis*: Netlify terutama dikenal sebagai platform *hosting* statis. Ini berarti Netlify sangat cocok untuk meng-*host* situs *web* yang terdiri dari berkas HTML, CSS, *JavaScript*, dan aset statis lainnya. Netlify secara otomatis menyajikan situs *web* dari *cache* di lokasi geografis yang dekat dengan pengguna, memberikan kinerja yang lebih baik.
2. Integrasi Git: Netlify menyediakan integrasi yang kuat dengan sistem kontrol versi Git seperti GitHub, GitLab, dan Bitbucket. Pengguna dapat menghubungkan repositori Git pengguna ke Netlify, dan setiap kali ada perubahan pada repositori, Netlify akan secara otomatis membangun dan men-*deploy* situs baru.
3. Pengelolaan Alur Kerja (*Workflow*): Netlify memfasilitasi penerapan alur kerja (*workflow*) yang lancar. Dapat mengkonfigurasi berbagai tahap dalam alur kerja pengembangan, pengujian, dan penerbitan.
4. Pembaruan Berbasis Pengguliran (*Rolling Releases*): Ketika ada pembaruan situs, Netlify mendukung pembaruan berbasis pengguliran, yang berarti pembaruan diterapkan secara bertahap ke *server*.
5. SSL Otomatis dan HTTPS Kuat: Netlify menyediakan SSL otomatis dan HTTPS gratis untuk semua situs yang di-*host*, bahkan untuk situs dengan domain kustom.
6. Fungsi *Serverless* (*Serverless Functions*): Netlify mendukung fungsi *serverless*, yang memungkinkan pengguna menulis dan menjalankan kode *backend* tanpa perlu mengelola *server*. Fungsi *serverless* ini dapat digunakan untuk menangani logika *server-side*.
7. Pembaruan Otomatis dari *Branch* atau *Pull Request*: Pengguna dapat mengaktifkan penerapan otomatis untuk setiap perubahan pada *branch* tertentu atau *pull request*. Ini mempermudah uji coba dan pengujian sebelum penerbitan ke lingkungan produksi.

3.1.7 Deploy With Netlify

1. Project

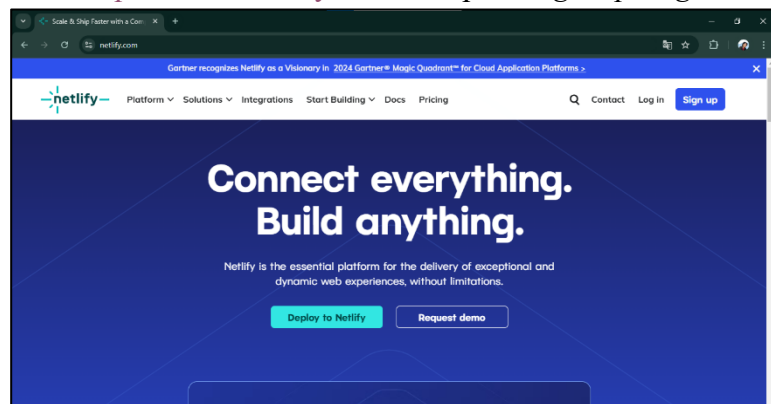
- a. Duplicate project pertemuan ke 5 dan buatkan dalam folder baru “Pertemuan-7”



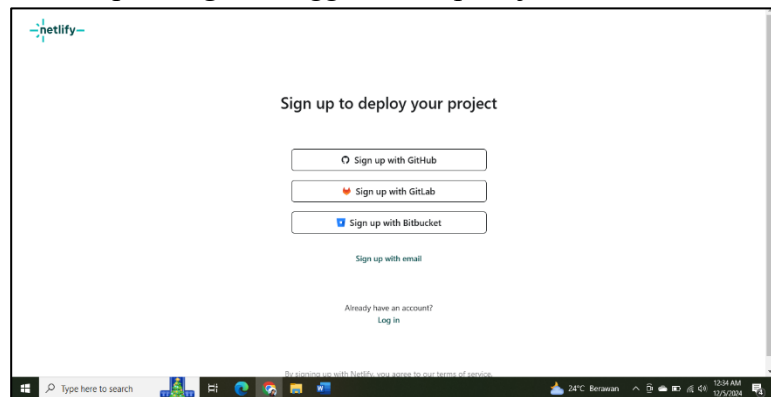
Name	Date modified	Type	Size
Assets	12/5/2024 12:45 AM	File folder	
index	11/28/2024 9:17 PM	Chrome HTML Do...	3 KB
style	11/28/2024 9:18 PM	Cascading Style S...	1 KB

2. Deploy

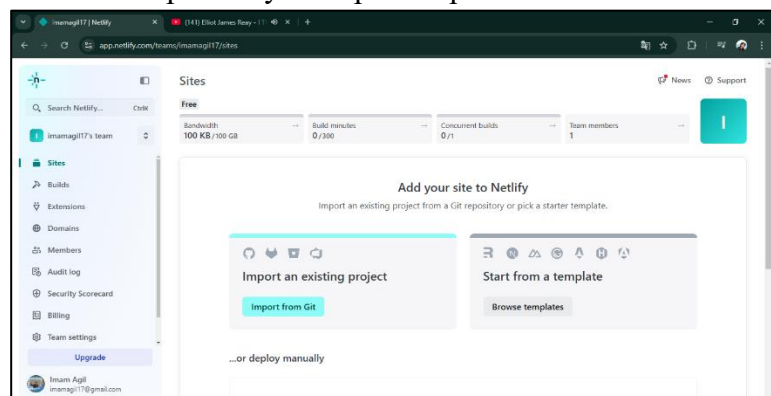
- a. Buka <https://www.netlify.com/> dan pilih *Sign Up/Log In*



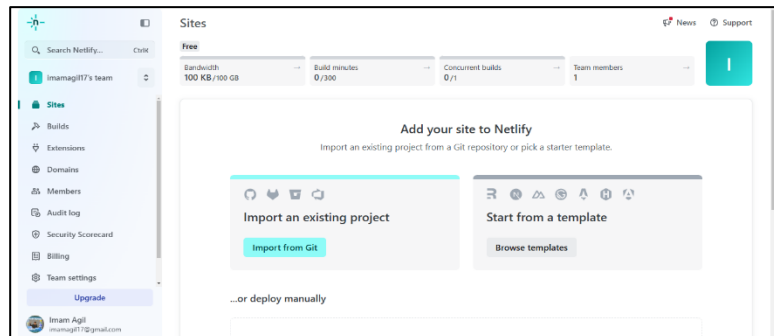
- b. Silakan pilih *login* menggunakan apa saja



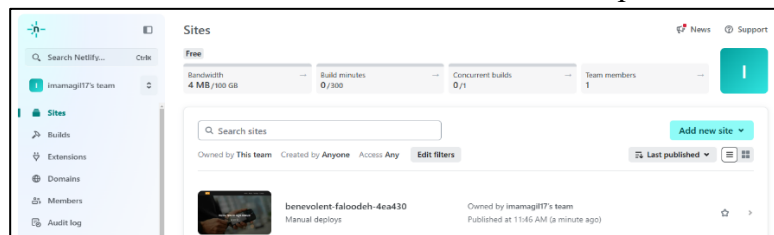
- c. Ikuti semua prosesnya sampai tampil halaman ini



- d. Pilih *site*, kemudian *drag & drop* folder “Pertemuan-6”



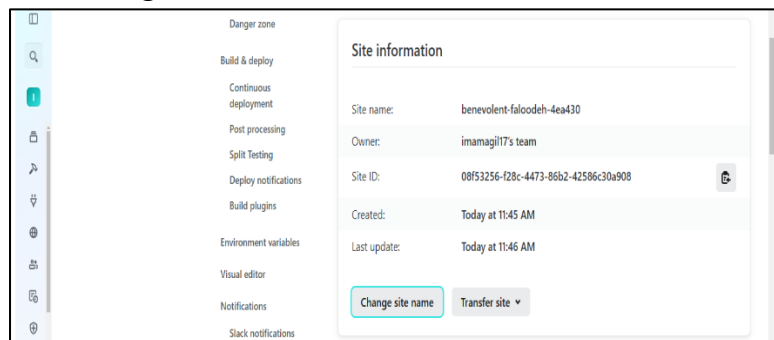
- e. Pada menu utama “*Team Overview*” *scroll* dan pilih *site* tadi



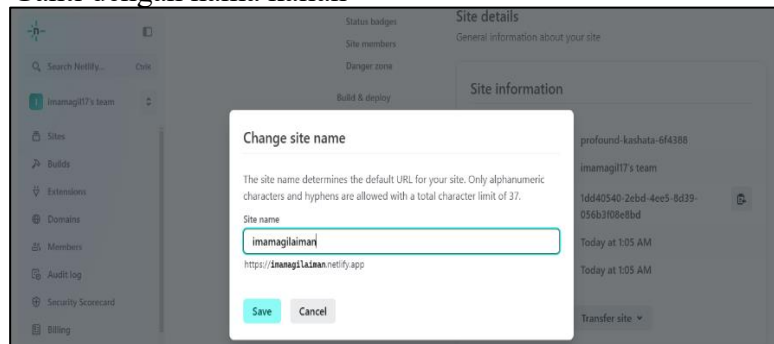
- f. Kemudian pilih *site configuration*



- g. Pilih *change site name*

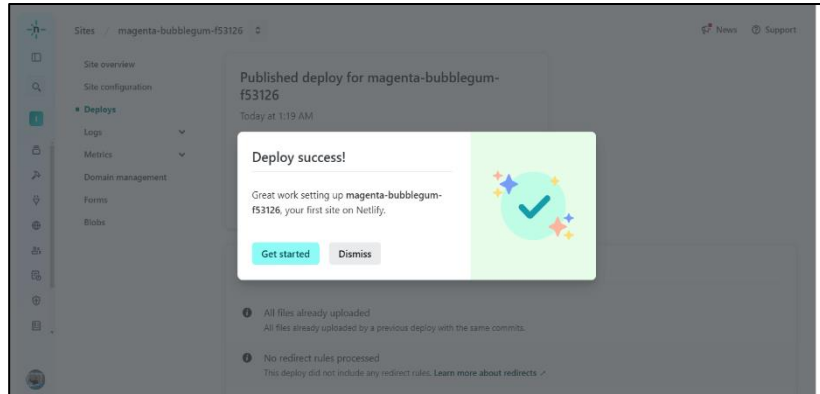


- h. Ganti dengan nama kalian



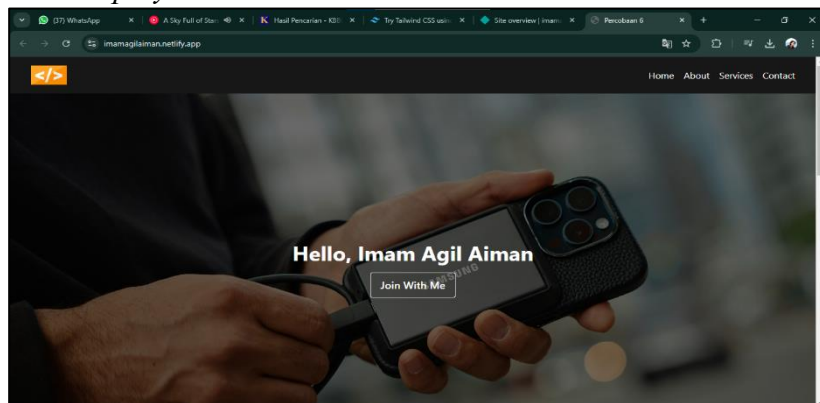
4.1.7 *Deploy With Netlify*

1. *Success Deploy*



Gambar diatas menunjukkan antarmuka dari Netlify, yang diamana sebuah platform untuk meng-*hosting* dan mengelola aplikasi *web*. "*Deploy success!*" menunjukkan bahwa proses *deploy* (pengunggahan dan penerapan) situs *web* bernama *magenta-bubblegum-f53126* telah berhasil dilakukan. "*Get started*" untuk melanjutkan pengaturan dari situs tersebut, sementara "*Dismiss*" menutup pemberitahuan ini.

2. *Web Deploy*



Gambar diatas merupakan hasil *deploy* yang menunjukkan sebuah situs *web* pribadi yang berhasil diunggah dan di-*hosting* menggunakan Netlify. Alamat *website*-nya tertulis ” <https://imamagilaiman.netlify.app/>”, ini adalah *subdomain default* yang disediakan oleh Netlify. Situsnya menampilkan *header*, *background*, pesan utama, dan tombol CAT yang mengarahkan pengunjung ke langkah berikutnya

BAB V

PENUTUP

5.1 Kesimpulan

5.1.1 *Basic tag HTML Part 1*

Kesimpulan dari modul 1 adalah untuk mengenalkan *tag-tag* dasar pada HTML yang digunakan untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di internet melalui layanan *web* dengan menggunakan berbagai macam *tag-tag* yang sesuai dengan kebutuhan seperti teks, gambar, *link*, *header*, *image*, dan lain-lain

5.1.2 *Basic Tag HTML Part 2*

Kesimpulan mengenai konsep dasar HTML *part 2* yang mencakup *bullet list*, tabel, dan *form*. Dalam HTML, membuat daftar dengan menggunakan daftar tak berurutan (*unordered list*) dan daftar berurutan (*ordered list*). Tabel dalam HTML menggunakan beberapa elemen penting, yaitu : *table row*, *table header*, dan *table data*, dengan kombinasi elemen tersebut dapat membuat tabel yang rapi. Formulir yang memungkinkan pengguna memasukkan data, elemen-elemen dalam *form*, seperti : “<input>”, “<textarea>”, dan “<select>”.

5.1.3 *HTML Layout*

Penggunaan *semantic layout* dalam HTML, seperti “<nav>”, “<main>”, “<article>”, dan “<footer>”, memberikan konteks dan makna yang lebih jelas pada struktur halaman, hal ini membantu pembaca layar untuk memahami konten dan fungsi setiap bagian, meningkatkan *Search Engine Optimization* (SEO) halaman. Sedangkan *layout* seperti “<div>” tidak memberikan arti atau tujuan khusus. Artinya *layout* lebih sulit diinterpretasikan oleh mesin pencari. *Division* merupakan kontainer blok yang sering digunakan untuk mengelompokkan elemen-elemen dalam sebuah bagian. Biasanya “<div>” dipakai dengan CSS untuk mengatur tata letak atau gaya dari sekelompok elemen, tapi “<div>” tidak memberikan informasi atau makna yang spesifik.

5.1.4 CSS (*Cascading Style Sheets*)

Ketika menggunakan CSS untuk mendesain dan mengatur tata letak elemen-elemen dalam halaman *web*, dapat memberikan gaya visual pada HTML, dan membuat halaman *web* lebih menarik, mudah dibaca, dan responsif terhadap berbagai perangkat. CSS memungkinkan pengaturan berbagai elemen visual, seperti warna, ukuran, *font*, *margin*, *padding*, dan *border*. CSS sangat penting dalam pengembangan *web modern*, memungkinkan pengembang untuk menciptakan desain yang estetik, fungsional, dan responsif.

5.1.5 CSS *Part 2 Flex & Grid*

CSS *Flexbox* dan *Grid* adalah dua teknik tata letak yang sangat kuat dalam pengembangan *web* moderen. Dengan memahami kapan dan bagaimana menggunakan masing-masing alat ini akan memungkinkan untuk menciptakan desain yang responsif, fleksibel, dan estetik. *Flexbox* menawarkan kesederhanaan dan fleksibilitas untuk tata letak satu dimensi, sedangkan *Grid* memberikan kontrol yang lebih besar untuk tata letak dua dimensi yang kompleks. Kombinasi keduanya digunakan untuk mendapatkan tata letak yang optimal dan efisien.

5.1.6 CSS *Framework TailwindCSS*

Tailwind CSS adalah salah satu *framework* CSS moderen yang berbasis utilitas. *Framework* ini memberikan kumpulan kelas siap pakai untuk mempermudah pengembangan antarmuka *web* yang responsif dan konsisten. *Framework* ini juga sangat cocok bagi pengembang yang ingin menciptakan desain yang unik tanpa bergantung pada gaya bawaan (*predefined components*) dan dengan pengendalian penuh atas tampilan.

5.1.7 *Deploy With Netlify*

Netlify adalah platform yang memudahkan proses *deploy* (pengunggahan dan penerapan) aplikasi *web* statis atau berbasis *serverless*. Dengan Netlify, pengguna dapat mengunggah proyek mereka, mengatur domain, dan mengelola *file* secara otomatis tanpa memerlukan pengaturan *server* yang rumit.