

STRUKTUR CSS YANG DIGUNAKAN PADA FRONTEND

Studi Kasus: Student Management UI — Layout 16:9

Student Management

Home

Students

About

Form Student

NIM

Masukkan NIM

Nama Lengkap

Masukkan nama lengkap

Jurusan

Pilih Jurusan

Email

Masukkan email

Simpan

Batal

Reset

Data Mahasiswa

Cari mahasiswa...

No	NIM	Nama	Jurusan	Email	Aksi
1	231001	Andi Pratama	Informatika	andi@mail.com	
2	231002	Siti Aisyah	Sistem Informasi	siti@mail.com	
3	231003	Budi Santoso	Teknik Komputer	budi@mail.com	
4	231004	Nina Marlina	Manajemen	nina@mail.com	
5	231005	Rizky Pratama	Informatika	rizky@mail.com	

Menampilkan 1 - 5 dari 50 data

«

1

2

3

»

Dokumen ini menjelaskan struktur CSS yang digunakan untuk membangun tampilan Student Management seperti desain UI di atas. Fokus pembelajaran adalah hubungan antara HTML sebagai struktur dan CSS sebagai pengatur layout, visual, komponen, serta responsive design.

1. Gambaran Arsitektur Frontend

FRONTEND

HTML

Struktur dan konten

CSS

Global / Reset

Variables

Navbar

Layout

Form

Table

Button

Search

Pagination

Responsive

1

```
└─ JavaScript
  └─ Interaksi dan data
```

2. Struktur File CSS

```
student-management/
├── index.html
├── css/
│   └── style.css
└── js/
    └── app.js
```

Untuk studi kasus HTML + CSS, file utama yang digunakan adalah index.html dan style.css. JavaScript dapat ditambahkan kemudian untuk membuat fungsi CRUD dan Fetch API.

3. CSS Reset dan Global Style

Bagian pertama stylesheet menghilangkan margin/padding bawaan browser dan menetapkan box-sizing serta style global.

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  min-height: 100vh;
  font-family: "Segoe UI", Arial, sans-serif;
  color: #182235;
  background: #eef3f8;
}
```

Konsep utama: universal selector, box model, font-family, color, background, dan viewport height.

4. CSS Variables

```
:root {
  --primary: #1469e8;
  --primary-dark: #0f4fae;
  --danger: #f52f3d;
  --secondary: #6c7b8b;
  --border: #d9e0e8;
  --text: #182235;
  --surface: #ffffff;
```

```
--radius: 14px;
}
```

CSS Variables membuat warna, radius, dan nilai desain dapat digunakan kembali oleh berbagai komponen.

5. Struktur Navbar

```
.navbar {
  height: 76px;
  padding: 0 34px;
  display: flex;
  align-items: center;
  justify-content: space-between;
  color: white;
  background: #123f70;
}

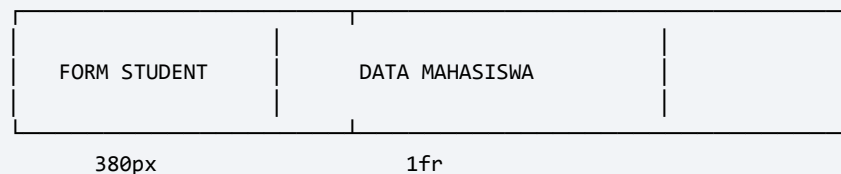
.nav-menu {
  display: flex;
  align-items: center;
  gap: 36px;
}
```

Navbar menggunakan Flexbox. `justify-content: space-between` memisahkan brand dan menu navigasi.

6. Layout Utama dengan CSS Grid

```
.container {
  width: min(1150px, calc(100% - 32px));
  margin: 18px auto;
  display: grid;
  grid-template-columns: 380px 1fr;
  gap: 16px;
}
```

Layout utama terdiri dari dua kolom: kolom kiri untuk Form Student dan kolom kanan untuk Data Mahasiswa.



7. Card Component

```
.card {
  background: var(--surface);
}
```

```
border: 1px solid #e3e8ee;
border-radius: var(--radius);
padding: 28px;
box-shadow: 0 8px 22px rgba(33, 54, 79, .08);
}
```

Card menggunakan kombinasi background, border, border-radius, padding, dan box-shadow untuk menghasilkan panel UI seperti pada desain.

8. Struktur Form Student

```
.form-group {
  margin-bottom: 16px;
}

.form-group label {
  display: block;
  margin-bottom: 7px;
  font-weight: 600;
}

.form-group input,
.form-group select {
  width: 100%;
  height: 44px;
  padding: 0 14px;
  border: 2px solid var(--border);
  border-radius: 7px;
}
```

Form memanfaatkan width 100%, padding, border, dan border-radius. Setiap input dikelompokkan dalam .form-group agar struktur CSS mudah dipelihara.

9. Focus State pada Form

```
.form-group input:focus,
.form-group select:focus {
  border-color: var(--primary);
  box-shadow: 0 0 0 3px rgba(20, 105, 232, .10);
}
```

Pseudo-class :focus memberikan feedback visual ketika pengguna sedang mengisi form.

10. Button Component

```
.btn {
  min-height: 42px;
  padding: 0 17px;
  border: 0;
}
```

```

    border-radius: 8px;
    color: white;
    font-weight: 600;
}

.btn-primary {
    background: var(--primary);
}

.btn-secondary {
    background: var(--secondary);
}

.btn-danger {
    background: var(--danger);
}

```

Struktur ini menggunakan konsep reusable component: `.btn` sebagai style dasar, kemudian `.btn-primary`, `.btn-secondary`, dan `.btn-danger` sebagai variasi.

11. Search Box dengan Flexbox

```

.search-box {
    display: flex;
    height: 42px;
}

.search-box input {
    width: 210px;
    padding: 0 13px;
}

.search-box button {
    width: 48px;
}

```

Input pencarian dan tombol pencarian diletakkan berdampingan menggunakan `display: flex`.

12. Struktur Table

```

table {
    width: 100%;
    border-collapse: separate;
    border-spacing: 0;
}

th,
td {
    padding: 13px 12px;
    border-bottom: 1px solid var(--border);
}

th {

```

```

        color: white;
        background: var(--primary);
    }

tbody tr:hover {
    background: #f5f9ff;
}

```

Table menggunakan width 100%, padding pada cell, warna header, dan pseudo-class :hover untuk memberikan feedback saat pointer berada di atas baris.

13. Action Button: Edit dan Delete

```

.actions {
    display: flex;
    gap: 7px;
}

.action {
    width: 40px;
    height: 40px;
    border: 0;
    border-radius: 6px;
    color: white;
}

.edit {
    background: var(--primary);
}

.delete {
    background: var(--danger);
}

```

14. Pagination

```

.pagination {
    display: flex;
    justify-content: flex-end;
    margin-top: 25px;
}

.pagination button {
    min-width: 43px;
    height: 42px;
    border: 1px solid var(--border);
    background: white;
}

.pagination .active {
    color: white;
}

```

```
background: var(--primary);
}
```

15. Responsive Design

Pada desktop, Form Student dan Data Mahasiswa berada dalam dua kolom. Pada layar lebih kecil, layout diubah menjadi satu kolom.

```
@media (max-width: 950px) {
  .container {
    grid-template-columns: 1fr;
  }
}

@media (max-width: 650px) {
  .nav-menu a {
    display: none;
  }

  .table-header {
    flex-direction: column;
  }

  .form-actions {
    flex-wrap: wrap;
  }
}
```

Desktop

Form	Data Mahasiswa
------	----------------

Mobile

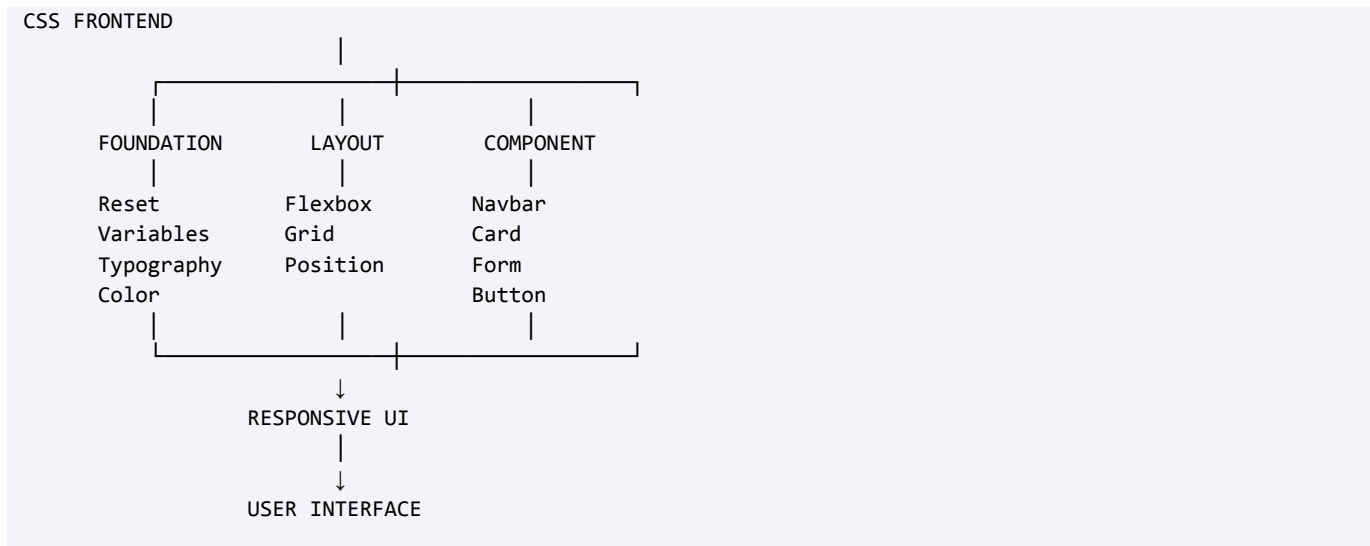
Form Student
Data Mahasiswa

16. Ringkasan Struktur CSS

Bagian CSS	Teknologi/Konsep	Fungsi
Reset & Global	Selector, Box Model	Menetapkan aturan dasar
Variables	CSS Custom Properties	Warna dan nilai reusable
Navbar	Flexbox	Navigasi horizontal
Main Layout	CSS Grid	Dua kolom utama
Card	Box Model	Panel UI
Form	Form styling	Input data mahasiswa
Button	Reusable class	Aksi pengguna

Search	Flexbox	Pencarian data
Table	Table CSS	Daftar mahasiswa
Action	Flexbox	Edit/Hapus
Pagination	Flexbox	Navigasi halaman
Responsive	Media Query	Desktop, tablet, mobile

17. Peta Konsep CSS Frontend

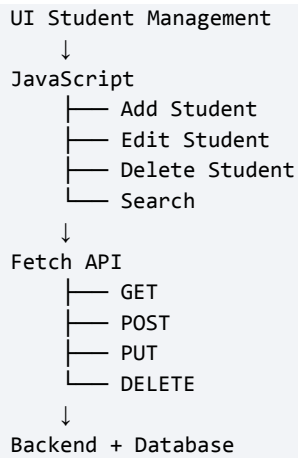


18. Urutan Pembelajaran untuk Mahasiswa

1. HTML: buat struktur semantic halaman.
2. CSS Reset dan Box Model.
3. CSS Variables dan typography.
4. Flexbox untuk navbar, search, dan action.
5. CSS Grid untuk layout utama.
6. Styling form dan table.
7. Reusable button dan card component.
8. Pseudo-class :hover dan :focus.
9. Media Query untuk responsive design.
10. JavaScript pada tahap berikutnya untuk CRUD dan Fetch API.

19. Pengembangan ke JavaScript





Dengan pendekatan bertahap ini, studi kasus yang sama dapat digunakan untuk materi HTML, CSS, JavaScript, REST API, backend, hingga database.

20. Kesimpulan

Desain Student Management menggunakan beberapa konsep inti CSS modern: CSS Variables untuk konsistensi desain, Flexbox untuk susunan satu dimensi, CSS Grid untuk layout dua dimensi, Box Model untuk ukuran dan jarak, pseudo-class untuk interaksi visual, serta Media Query untuk responsive design. Struktur tersebut memisahkan tanggung jawab styling berdasarkan komponen sehingga lebih mudah dipahami, dikembangkan, dan dipelihara oleh mahasiswa.