



Website CRUD Cake with by:Celsy and Tiara


# Tugas Presentasi

Smk Astrindo  
29-10-2025



# latar belakang

Di era digital saat ini, banyak organisasi dan bisnis membutuhkan sistem aplikasi web yang mampu memudahkan pengelolaan data secara efisien — mulai dari penambahan, pembacaan, pembaruan, hingga penghapusan data (Create-Read-Update-Delete atau CRUD). Sistem manual atau semi-otomatis sering kali rentan terhadap kesalahan, duplikasi data, kurangnya integrasi, dan memakan waktu banyak.



# Tujuan Proyek

- Membangun aplikasi web berbasis CRUD (Create, Read, Update, Delete) untuk pengelolaan data produk (Cake) dari toko.
- Mempermudah proses pemesanan dan administrasi stok agar data selalu uptodate dan dapat diakses kapan saja.
- Mendukung usaha dalam bersaing dan adaptif di pasar Cake yang semakin kreatif dan digital.

# Alat dan Bahan

perangkat lunak

Canva AI dan chat gpt

- canva: untuk membuat desain

Chat gpt: untuk membuat kode website

perangkat keras

PC, HP dan koneksi internet



# Konsep CRUD

## 1. Create (Membuat / Menambah Data Kue)

Fungsi ini digunakan untuk menambahkan data kue baru ke dalam sistem

## 2. Read (Membaca / Menampilkan Data Kue)

Digunakan untuk menampilkan daftar kue yang ada di database kepada pengguna.

## 3. Update (Mengubah Data Kue)

Fungsi ini memungkinkan admin untuk memperbarui data kue yang sudah ada

## 4. Delete (Menghapus Data Kue)

Digunakan untuk menghapus data kue dari sistem. Contohnya:

Admin menghapus kue yang sudah tidak dijual lagi.



# Desain UI menggunakan Canva AI

Canva AI di gunakan untuk:

- membuat kode website
- desain

```
// 4. The app stores data in localStorage under the key 'cakes'.

export default function CakeCRUD() {
  const [cakes, setCakes] = useState([]);
  const [query, setQuery] = useState("");
  const [editing, setEditing] = useState(null); // cake being edited
  const [form, setForm] = useState({ name: "", flavor: "", price: "", image: "" });
  const [showForm, setShowForm] = useState(false);

  useEffect(() => {
    const saved = localStorage.getItem("cakes");
    if (saved) setCakes(JSON.parse(saved));
    else {
      // sample initial data
      const sample = [
        { id: 1, name: "Classic Chocolate", flavor: "Chocolate", price: 45, image: "" },
        { id: 2, name: "Vanilla Dream", flavor: "Vanilla", price: 40, image: "" },
      ];
      setCakes(sample);
      localStorage.setItem("cakes", JSON.stringify(sample));
    }
  }, []);

  useEffect(() => {
```



# Cuplikan kode

```
import React, { useEffect, useState } from "react";
```

```
// CakeCRUD.jsx // Single-file React component (Tailwind CSS classes used). // How to use: // 1. Create a React app (Vite / Create React App). // 2. Install and configure Tailwind CSS (or replace classes with your own CSS). // 3. Place this file as CakeCRUD.jsx and import it into your app (e.g., App.jsx). // 4. The app stores data in localStorage under the key 'cakes'.
```

```
export default function CakeCRUD() { const [cakes, setCakes] = useState([]); const [query, setQuery] = useState(""); const [editing, setEditing] = useState(null); // cake being edited const [form, setForm] = useState({ name: "", flavor: "", price: "", image: "" }); const [showForm, setShowForm] = useState(false);
```

```
useEffect(() => { const saved = localStorage.getItem("cakes"); if (saved) setCakes(JSON.parse(saved)); else { // sample initial data const sample = [ { id: 1, name: "Classic Chocolate", flavor: "Chocolate", price: 45, image: "" }, { id: 2, name: "Vanilla Dream", flavor: "Vanilla", price: 40, image: "" }, ]; setCakes(sample); localStorage.setItem("cakes", JSON.stringify(sample)); }, [ ]);
```

```
useEffect(() => { localStorage.setItem("cakes", JSON.stringify(cakes)); }, [cakes]);
```

```
function resetForm() { setForm({ name: "", flavor: "", price: "", image: "" }); setEditing(null); }
```

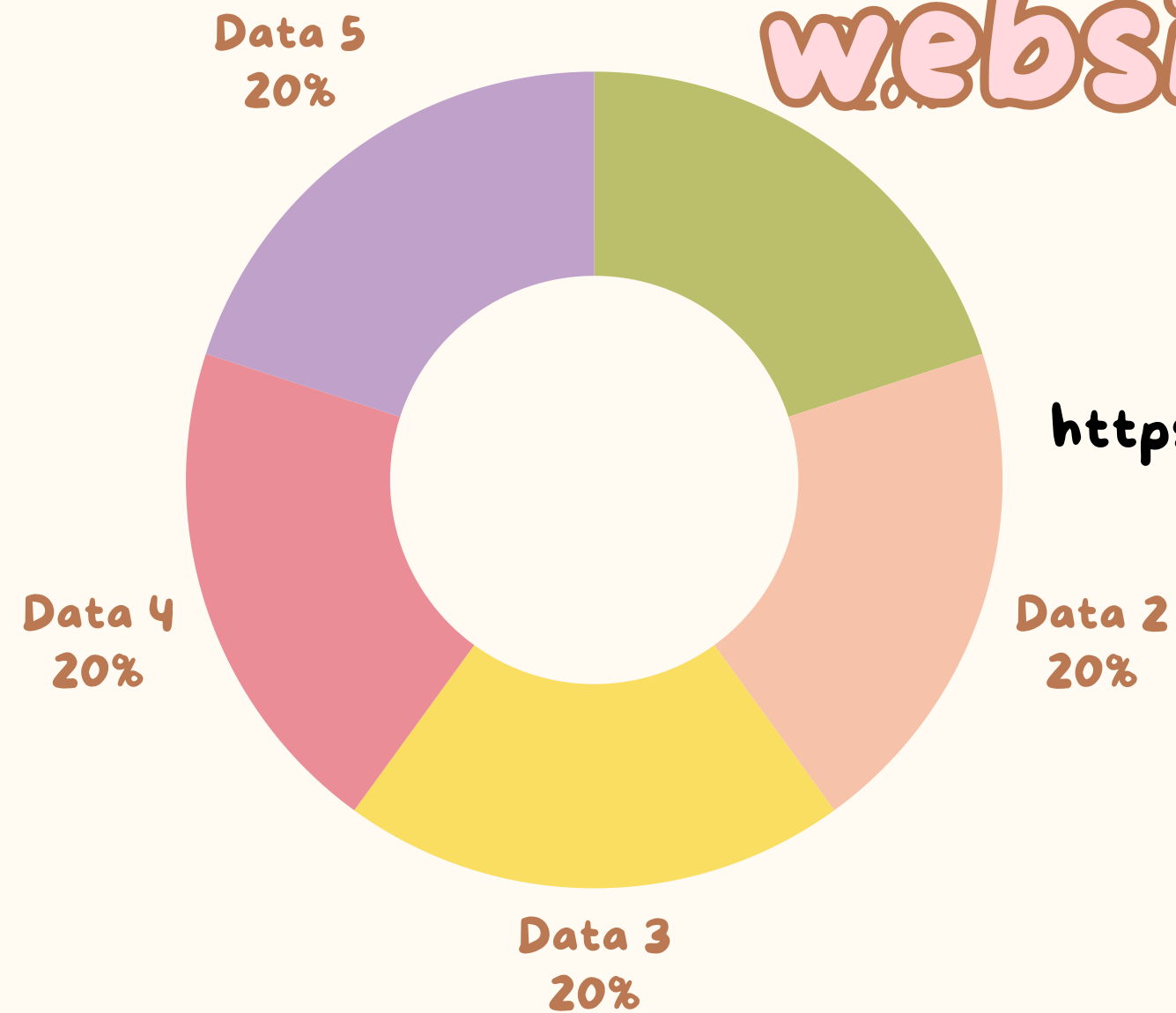
```
function handleChange(e) { const { name, value } = e.target; setForm((s) => ({ ...s, [name]: value })); }
```

```
function validateForm() { if (!form.name.trim()) return "Nama kue wajib diisi."; if (!form.flavor.trim()) return "Rasa kue wajib diisi."; if (!form.price || isNaN(Number(form.price))) return "Harga harus berupa angka."; return null; }
```

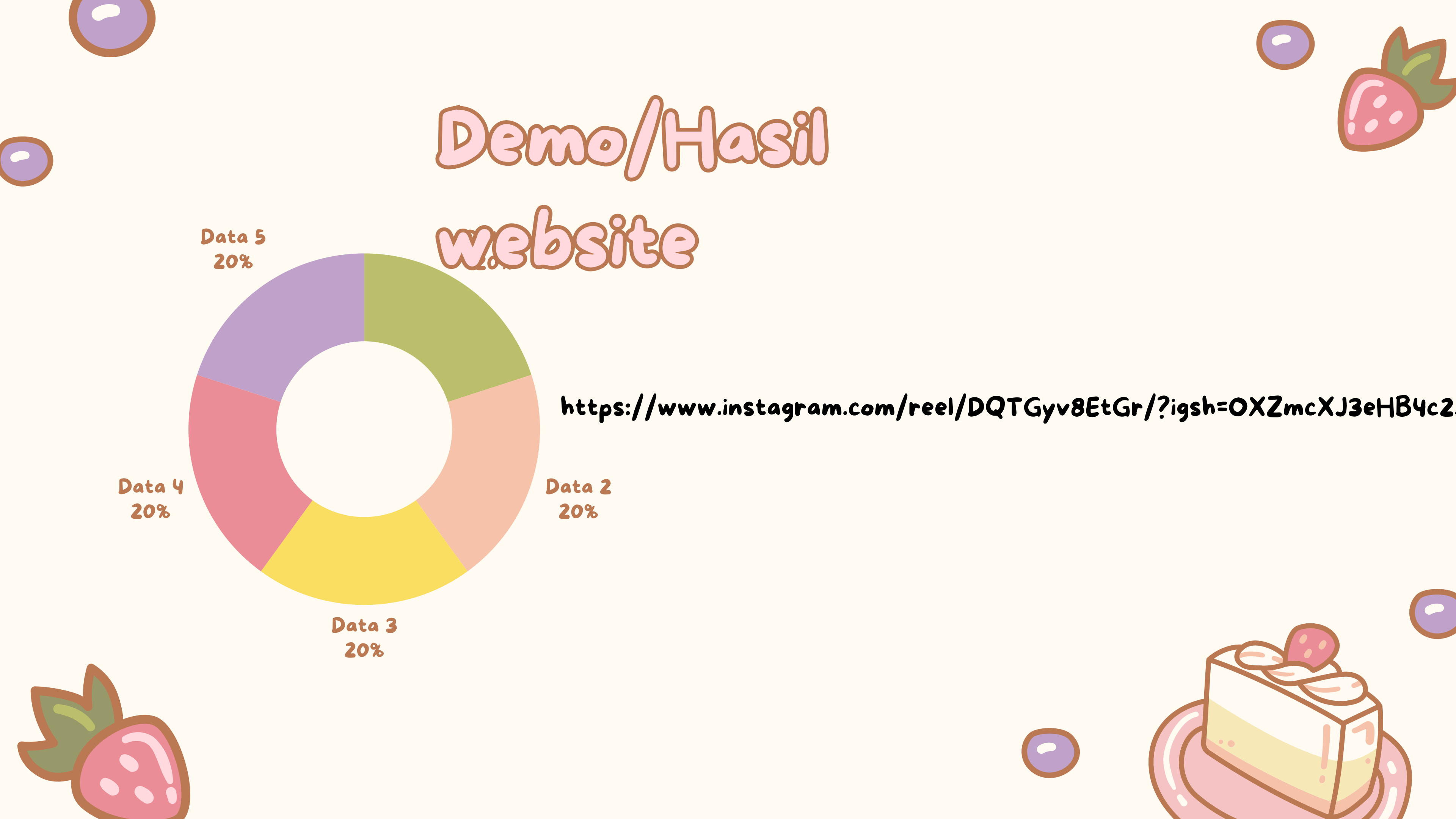
```
function handleAdd() { const err = validateForm(); if (err) return alert(err); const nextId = cakes.length ? Math.max(...cakes.map((c) => c.id)) + 1 : 1; const newCake = { id: nextId, name: form.name.trim(), flavor: form.flavor.trim(), price: Number(form.price), image: form.image.trim() }; setCakes((s) => [newCake, ...s]); resetForm(); setShowForm(false); }
```



# Demo/Hasil website



<https://www.instagram.com/reel/DQTGyv8EtGr/?igsh=OXZmcXJ3eHB4c2>







# Hasil pembelajaran

Mendesain tampilan menarik dengan Canva AI.

Bekerja sama dalam tim dan membagi tugas.

Menyelesaikan error kecil dalam kode (debugging).



# kendala dan solusi

- Koneksi database error Periksa file koneksi dan nama database
- Data kue tidak tersimpan Cek form input dan query INSERT
- Data tidak tampil Periksa query SELECT dan loop data
- Edit data gagal Pastikan id\_kue dikirim dan query UPDATE benar
- Hapus data gagal Cek id\_kue dan query DELETE
- Upload gambar error Pastikan folder upload benar dan izin tulis aktif
- Tampilan berantakan Gunakan CSS/Bootstrap untuk mempercantik tampilan
- Website tidak aman Tambahkan login dan validasi input



# kesimpulan

Pembuatan website CRUD Cake bertujuan untuk mempermudah pengelolaan data kue seperti menambah, menampilkan, mengedit, dan menghapus data secara efisien. Melalui sistem ini, proses administrasi menjadi lebih cepat, rapi, dan terkomputerisasi. Dengan desain database yang baik dan tampilan yang menarik, website CRUD Cake dapat membantu pengguna mengelola informasi kue secara mudah, aman, dan praktis.



Terima Kasih

