DAFTAR ISTILAH

- Activity* : Diagram UML yang memodelkan alur kerja (*workflow*) atau rangkaian aktivitas dalam sebuah sistem.
- Diagram*
- Agile* : Paradigma pengembangan perangkat lunak yang menekankan fleksibilitas, kolaborasi, dan pengembangan iteratif.
- Android* : Sistem operasi open-source dari *Google* untuk perangkat bergerak yang menjadi platform target aplikasi ini.
- Black Box Testing* : Metode pengujian fungsionalitas perangkat lunak dari perspektif *user*, tanpa melihat struktur kode internalnya.
- Bottom-up* : Pendekatan pengumpulan data yang dimulai dari level individu atau unit terkecil.
- Carbon Footprint* : Jumlah total emisi gas rumah kaca (dinyatakan dalam CO₂e) yang dihasilkan secara langsung maupun tidak langsung dari aktivitas individu atau organisasi.
- Class Diagram* : Diagram struktur UML yang menggambarkan kelas-kelas dalam sistem, beserta atribut, metode, dan hubungannya.
- Cloud Firestore* : Layanan basis data NoSQL dari *Firebase* yang digunakan untuk menyimpan data aplikasi dalam bentuk dokumen dan koleksi.
- Dart* : Bahasa pemrograman yang digunakan untuk membangun aplikasi dengan kerangka kerja Flutter.
- Dokumen (Firebase)* : Unit penyimpanan data utama dalam *Cloud Firestore* yang berisi kumpulan pasangan field dan nilai.
- Faktor Emisi* : Nilai yang merepresentasikan jumlah emisi CO₂e yang dihasilkan per satuan aktivitas (misal: per liter bahan bakar).
- Firebase* : Platform *Backend as a Service* (BaaS) dari *Google* yang menyediakan layanan seperti autentikasi, basis data, dan storage.

- Firestore* : Layanan dari *Firestore* yang digunakan untuk mengelola proses pendaftaran dan login pengguna secara aman.
- Figma* : Perangkat lunak desain berbasis web yang digunakan untuk merancang dan membuat prototipe antarmuka (UI) dan pengalaman pengguna (UX).
- Flutter* : Kerangka kerja (*framework*) *open-source* dari *Google* untuk membangun aplikasi multi-platform dari satu basis kode.
- Gamifikasi* : Penggunaan elemen dan mekanisme permainan dalam konteks non-permainan untuk meningkatkan keterlibatan *user*.
- Global Warming Potential* : Metrik yang membandingkan dampak pemanasan dari berbagai gas rumah kaca terhadap CO₂.
- Layered Architecture* : Pola arsitektur perangkat lunak yang memisahkan aplikasi menjadi beberapa lapisan logis, seperti presentasi, bisnis, dan data.
- Prototype Model* : Metode pengembangan perangkat lunak (SDLC) yang berfokus pada pembuatan model kerja awal (prototipe) untuk divalidasi oleh *user* sebelum pengembangan penuh.
- Nudging* : Teknik intervensi dalam desain perilaku yang bertujuan untuk memengaruhi pilihan *user* ke arah yang positif tanpa paksaan.
- Sequence Diagram* : Diagram interaksi UML yang menunjukkan bagaimana objek-objek saling berkomunikasi dan bertukar pesan secara berurutan dalam suatu waktu.
- Sub-Collection (Firestore)* : Sebuah koleksi yang terikat pada sebuah dokumen di *Cloud Firestore*, digunakan untuk mengorganisir data secara hierarkis.
- Scope* : Klasifikasi sumber emisi (*Scope 1, 2, 3*) berdasarkan standar GHG Protocol untuk mengkategorikan emisi langsung dan tidak langsung.

- Tier* : Tingkatan metodologi (*Tier* 1, 2, 3) dalam perhitungan emisi yang merepresentasikan tingkat kerincian dan akurasi data yang digunakan.
- Unified Modeling Language (UML)* : Bahasa pemodelan visual standar yang digunakan untuk merancang, mendokumentasikan, dan memvisualisasikan sistem perangkat lunak.
- Usability* : Aspek dari pengalaman *user* yang mengukur tingkat kemudahan, efisiensi, dan kepuasan *user* saat memakai suatu aplikasi.
- Use Case Diagram* : Diagram UML yang memodelkan fungsionalitas sistem dan interaksinya dengan *user* (aktor) dari sudut pandang eksternal.
- User Experience (UX)* : Keseluruhan persepsi dan respons emosional *user* yang timbul dari interaksi dengan suatu produk atau sistem.
- User Interface (UI)* : Tampilan visual dan komponen interaktif dari sebuah aplikasi yang memungkinkan *user* berinteraksi dengan sistem.
- Waterfall* : Model pengembangan perangkat lunak (SDLC) yang bersifat sekuensial dan linear, di mana setiap fase harus selesai sebelum fase berikutnya dimulai.