



UNIVERSITAS
GADJAH MADA

TRASHTRACK

APLIKASI MANAJEMEN SAMPAH RUMAH TANGGA



Proyek Akhir Mata Kuliah Pemrograman Berorientasi Objek

TIM GASS:

- Anggito Muhammad Amien (22/498140/TK/54627)
- Reihan Athar Zildaniar (22/497191/TK/54482)
- Syaifullah Hilmi Ma'arij (22/497775/TK/54568)

Project Repository:

<https://github.com/saaip7/TrashTrackProjectV2>



Tentang Aplikasi

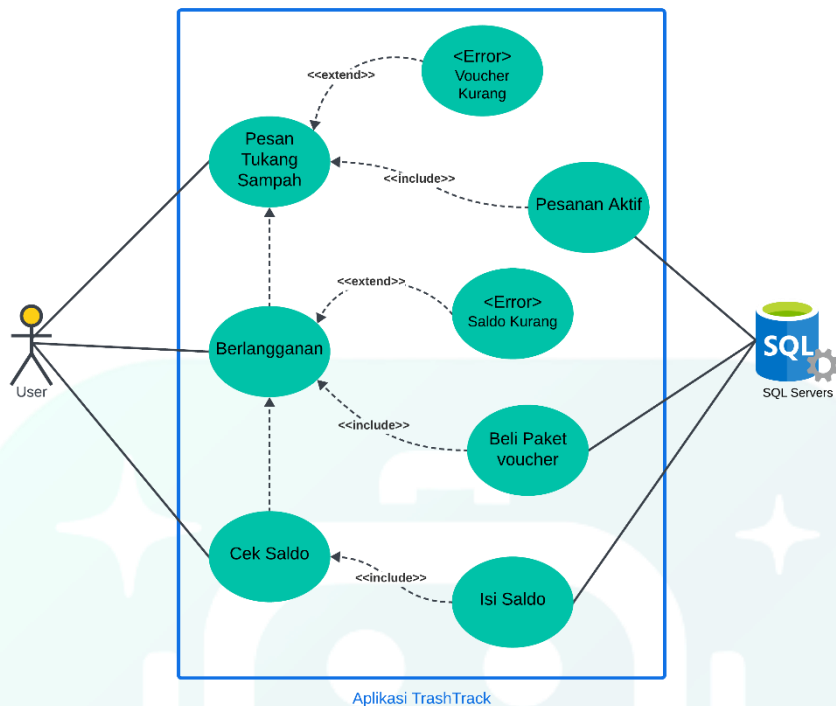
TrashTrack merupakan aplikasi desktop untuk mendukung tujuan global yakni memenuhi Sustainable Development Goals (SDGs) kategori **Sustainable Cities and Communities** dan **Responsible Consumption and Production** yang mengatasi permasalahan sampah untuk mencapai tujuan pembangunan berkelanjutan serta menciptakan lingkungan yang bersih dan nyaman.

TrashTrack merupakan aplikasi berbasis langganan yang dapat memanggil tukang sampah ke alamat tertentu. Sehingga, TrashTrack memberikan manfaat untuk memudahkan terutama ibu rumah tangga untuk mengatur sampah rumahnya.

Fitur Aplikasi

1. **Pesan Tukang Sampah**, digunakan untuk memanggil tukang sampah ke alamat tujuan yang diinput user.
2. **Berlangganan**, merupakan fitur untuk menambah voucher (seperti kuota) untuk menggunakan fitur pesan tukang sampah.
3. **Isi Saldo**, merupakan fitur untuk mengisi saldo akun yang digunakan untuk membeli paket berlangganan sehingga user dapat melakukan pemesanan tukang sampah,

Main Use Case



Use Case Diagram diatas merupakan diagram yang menggambarkan hubungan user dengan aplikasi TrashTrack dan bagaimana alurnya. User sebagai pengguna aplikasi mempunyai 3 akses, yakni melakukan pemesanan tukang sampah, berlangganan, dan cek saldo. Pemesanan tukang sampah hanya bisa dilakukan jika mempunyai voucher yang dibeli dengan saldo akun user tersebut. Terdapat fungsi Isi Saldo dengan hubungan include terhadap Cek Saldo yang menandakan bahwa fungsi Isi Saldo merupakan fungsionalitas use case Cek Saldo. Terdapat juga error message dengan hubungan extend yang berarti suatu kondisi dari use case tersebut terpenuhi, seperti error message Voucher Kurang yang terjadi Ketika user melakukan pemesanan tukang sampah, tetapi voucher akun user tersebut habis sehingga kondisi error message tersebut terpenuhi.

Implementasi

Database

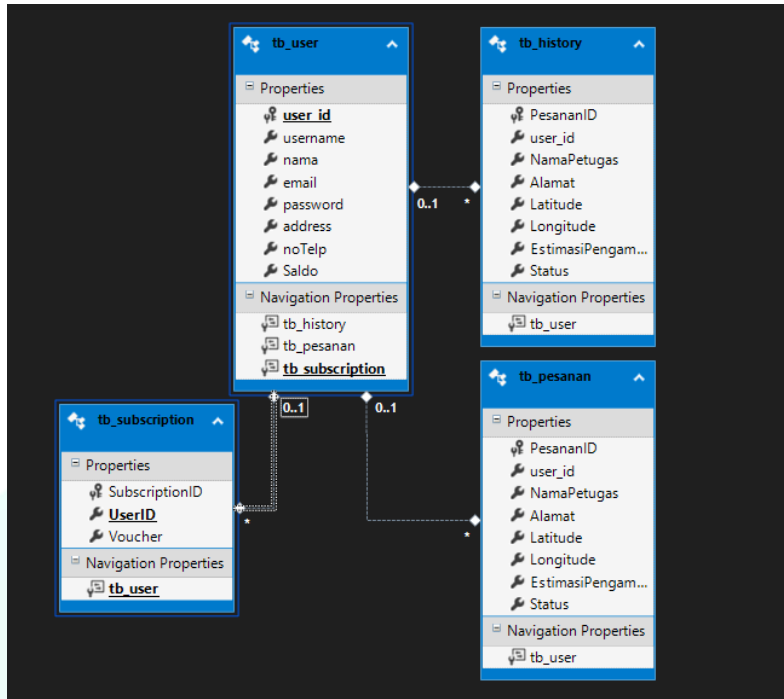
Aplikasi ini bertujuan untuk memudahkan pengguna dalam pengelolaan sampah. Dalam aplikasi ini diperlukan tempat untuk menyimpan profil user, voucher yang dimiliki setiap user, dan pesanan user beserta historynya. Untuk memenuhi kebutuhan tersebut, dirancang suatu Database yang memanfaatkan Data Interface menggunakan ADO.Net. Database dibuat



dengan menggunakan aplikasi Microsoft SQL Server dan diberi nama trashTrackProject. Database ini terdiri atas 4 tabel. Keempat tabel tersebut yakni `tb_user`, `tb_subscription`, `tb_pesanan`, dan `tb_history`. Setiap tabel memiliki kegunaan dan struktur kolom masing-masing. Berikut rincian struktur tabel berikut.

1. **Tabel Pengguna (`tb_user`):** Menyimpan informasi lengkap mengenai pengguna, termasuk nama, alamat, email, username, password, dan informasi kontak. Kunci utama `user_id` dijamin berbeda untuk setiap entitas pengguna.
2. **Tabel Berlangganan (`tb_subscription`):** Merupakan tabel yang menyimpan data voucher yang dimiliki oleh setiap entitas pengguna. Dalam tabel ini terdapat 3 kolom yakni `SubscriptionID` sebagai primary key, `UserID` sebagai foreign key dengan tabel pengguna, dan `Voucher` yang menyimpan data voucher pengguna.
3. **Tabel Pesanan (`tb_pesanan`):** Menyimpan informasi transaksi pesanan aktif. Didalamnya terdapat beberapa kolom diantaranya nama petugas, estimasi pengambilan, alamat, korrdinat latitude, koordinat longitude, dan status pesanan. Hubungan antara Tabel Pesanan dan Tabel Pengguna diatur melalui foreign key `user_id`.
4. **Tabel Histori (`tb_history`):** Menyimpan informasi transaksi pesanan yang selesai/tidak aktif. Didalamnya terdapat beberapa kolom yang sama dengan Tabel Pesanan diantaranya nama petugas, estimasi pengambilan, alamat, korrdinat latitude, koordinat longitude, dan status pesanan. Hubungan antara Tabel Histori dan Tabel Pengguna diatur melalui foreign key `user_id`.

Kemudian hubungan antara tabel dalam database ini dapat kita buat dalam suatu model. Berikut model dari database yang digunakan.



Component-Based

Mapsui.Wpf by Mapsui Developers 4.1.2

WPF map components based on the Mapsui library

Package Mapsui.Wpf diimplementasikan pada halaman pesan untuk memunculkan map. Map yang muncul pada halaman pesan memiliki fitur pan dan zoom secara default. Implementasi map pada page pesan dengan cara berikut.

```

//Create Map
MapControl.Map?.Layers.Add(Mapsui.Tiling.OpenStreetMap.CreateTileLayer());
//Map Borders
MapControl.Map.Navigator.OverridePanBounds = new MRect(12250759.8997, -838054.2427, 12339231.2208, -924691.7367);
MapControl.Map.Navigator.OverrideZoomBounds = new MMinMax(0, 200);
  
```

Map yang muncul akan memiliki batasan sesuai segiempat dengan koordinat diatas dan memiliki zoom maksimal 200, sehingga daerah yang muncul pada map hanya DIY dan sekitarnya.

Ketika user meletakkan pin dengan klik kanan, lokasi kursor relatif terhadap map akan disimpan dan dapat diubah menjadi koordinat pin. Informasi koordinat pin ini kemudian dikirim menjadi HTTP Request dengan API dan didapat HTTP Response untuk kemudian difilter dan didapat informasi alamat sesuai koordinat pin.

System.Data.SqlClient by Microsoft 4.8.5

Provides the data provider for SQL Server. These classes provide access to versions of SQL Server and encapsulate database-specific protocols, including tabular data stream (TDS)



Package ini menyediakan kelas-kelas yang mendukung terhubung ke dan berinteraksi dengan server SQL menggunakan .NET Framework. Ini adalah implementasi khusus untuk SQL Server dan menyediakan fungsionalitas seperti koneksi database, eksekusi pernyataan SQL, dan pengelolaan transaksi. Fungsionalitas Utama yang disediakan oleh Package ini sebagai berikut.

- **SqlConnection:** Kelas ini digunakan untuk membuka koneksi ke database SQL Server.
- **SqlCommand:** Kelas ini digunakan untuk mengeksekusi pernyataan SQL terhadap database.
- **SqlDataReader:** Kelas ini memungkinkan membaca baris-baris data dari database SQL Server.
- **SqlParameter:** Kelas ini digunakan untuk mewakili parameter yang akan digunakan dalam pernyataan SQL.

```
public bool UpdateUserProfile()
{
    SqlConnection connection = new SqlConnection(connectionString);
    bool isSuccessful = false;
    try
    {
        string query = "UPDATE tb_user SET nama = @Nama, address = @Alamat, noTelp = @NomorTelp, email=@Email WHERE user_id = @UserId";

        using (SqlCommand command = new SqlCommand(query, connection))
        {
            // Parameterized query untuk menghindari SQL Injection
            command.Parameters.AddWithValue("@Nama", Nama);
            command.Parameters.AddWithValue("@Alamat", Alamat);
            command.Parameters.AddWithValue("@NomorTelp", NoTelp);
            command.Parameters.AddWithValue("@Email", Email);
            command.Parameters.AddWithValue("@UserId", userID);

            connection.Open();

            // Eksekusi pernyataan UPDATE
            int rows = command.ExecuteNonQuery();
        }
    }
}
```

Package ini digunakan salah satunya pada fungsi UpdateUserProfile() pada class user. Dalam fungsi tersebut, digunakan fungsionalitas SqlConnection dan SqlCommand. Selain fungsi tersebut, berbagai fungsi yang memerlukan koneksi dengan database menggunakan fungsionalitas dari package ini.

Application Programming Interface (API)

TrashTrack merupakan aplikasi berbasis langganan yang dapat memanggil tukang sampah ke alamat tertentu sehingga perlu adanya informasi geografis dari pemesan. TrashTrack menggunakan API dari geocode yang digunakan untuk mendapatkan informasi lengkap mengenai sebuah lokasi berdasarkan input berupa alamat atau koordinat. Untuk menggunakan API, perlu inisiasi HttpClient seperti berikut.



```
using (var client = new HttpClient())
{
    if (input != null)
    {
        string formattedInput = input.Replace(' ', '+');
        string url = "https://geocode.maps.co/search?q=" + formattedInput;
        var response = await client.GetAsync(url);
        var result = await response.Content.ReadAsStringAsync();

        string jsonString = JsonConvert.SerializeObject(result);

        return jsonString;
    }
    else return null;
}
```

HttpClient memungkinkan adanya HTTP Request dan Response. Ada dua jenis API dari geocode yang digunakan pada program ini.

1. API Forward Geocoding

API Forward Geocoding dalam program ini berfungsi untuk mendapatkan lokasi-lokasi pada map yang sesuai dengan input user yang berupa string. Link HTTP Request nya adalah <https://geocode.maps.co/search?q=address> , dengan address adalah masukan dari user pada search bar. Berikut adalah contoh HTTP Response untuk forward geocoding dari geocode.

```
[{"place_id":189067004,"licence":"Data © OpenStreetMap contributors, ODbL 1.0. https://osm.org/copyright","powered_by":"Map Maker: https://maps.co","osm_type":"way","osm_id":368550475,"boundingbox":["-6.1758164","-6.1749759","106.8267719","106.8276001"],"lat":"-6.1753936","lon":"106.82718601871409","display_name":"National Monument, Jalan Medan Merdeka Utara, RW 02, Gambir, Central Jakarta, Jakarta Special Capital Region, 10110, Indonesia","class":"building","type":"yes","importance":0.6739616198583311}]
```

Bisa dilihat bahwa HTTP Response memiliki format json sehingga perlu dilakukan ekstraksi untuk mendapatkan string alamat yang lengkap yaitu pada bagian display_name. Untuk forward geocoding, hasil pencarian dapat memberikan lebih dari satu lokasi sehingga perlu dilakukan ekstraksi untuk menyimpan beberapa informasi hasil pencarian ini. Lokasi-lokasi ini disimpan dalam array dengan jumlah maksimal dari hasil pencarian adalah enam. User dapat memilih salah satu hasil pencarian untuk menampilkan pin pada map. Oleh karena itu, ekstraksi juga dilakukan untuk mendapatkan informasi koordinatnya yaitu longitude dan latitude. Setelah informasi koordinat diketahui, pin dapat dimunculkan pada map dengan akurat.

2. API Reverse Geocoding

Selain melalui forward geocoding, user juga dapat meletakkan pin secara langsung dengan klik kanan pada map. Ketika klik kanan, informasi koordinat



pada peta disimpan dan HTTP Request dikirim melalui <https://geocode.maps.co/reverse?lat=latitude&lon=longitude> dengan mengganti latitude dan longitude dengan informasi koordinat pada map. HTTP Response yang didapat identik dengan forward geocoding tetapi reverse geocoding hanya menghasilkan satu lokasi yang spesifik. Reverse geocoding digunakan untuk menampilkan nama lokasi yang diletakkan pin oleh user.

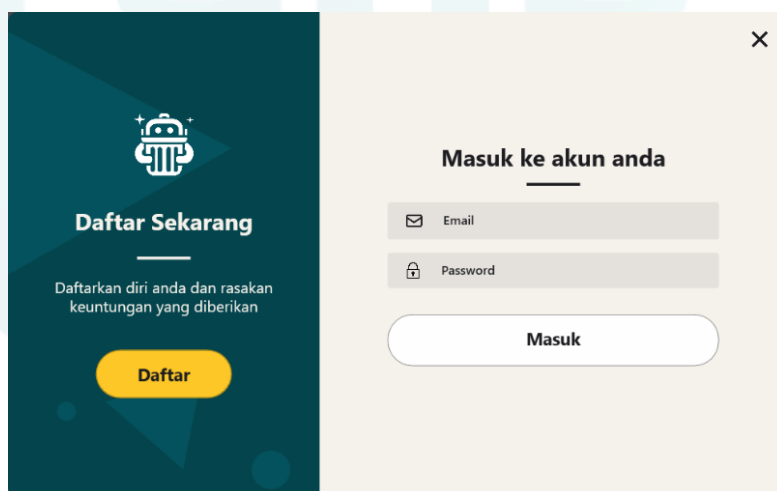
Refactoring

Refactoring dipakai pada seluruh kelas dengan melakukan extract method sehingga method dapat dibaca dan diketahui dengan jelas fungsionalitasnya. Kode dalam aplikasi mayoritas telah diberi komentar penjelas dan pemberian nama yang jelas sehingga fungsionalitas fungsi dapat diketahui dengan jelas.

Aplikasi ini juga sudah dibuat dengan struktur MVVM (Model-View-ViewModel) yang merupakan salah satu bentuk dari hasil refactoring sehingga kejelasan struktur pada aplikasi dapat diketahui dan memudahkan dalam pembacaan code dan pengembangan oleh developer lain (anggota yang lain).

Screenshot Execution

1. Login Page



Bagian ini memanfaatkan TextBlock, TextBox, Button, Image untuk logo dan icon-icon yang digunakan, elemen Canvas berupa Polygon dan Rectangle. Apabila user belum memasukkan Email atau Password dan atau salah memasukkan Email atau Password, akan muncul MessageBox error. Namun, jika user sudah terdaftar dalam database dan memasukkan Email atau Password yang benar, maka MainWindow akan



terbuka dan LoginPage ini akan tertutup. Tombol daftar pada sisi kiri juga akan mengarahkan user ke halaman Sign Up.

2. Sign Up Page

The screenshot shows a sign-up form titled "Daftarkan diri anda". It features a dark teal sidebar on the left with a white robot icon and the text "Sudah punya akun?" and a yellow "Masuk" button. The main form area is light beige and contains six input fields: Username, Nama, Email, Password, Nomor Telepon, and Alamat, followed by a white "Daftar" button.

Secara struktur, bagian ini sama seperti dengan Login Page sebelumnya, yang membedakan hanyalah jumlah dari TextBox yang fungsinya juga berbeda. TextBox di Login Page digunakan untuk masuk ke dalam aplikasi dengan memeriksa apakah akun sudah terdaftar di database atau belum. Sedangkan pada Sign Up Page ini, TextBox digunakan untuk melakukan registrasi ke dalam database.

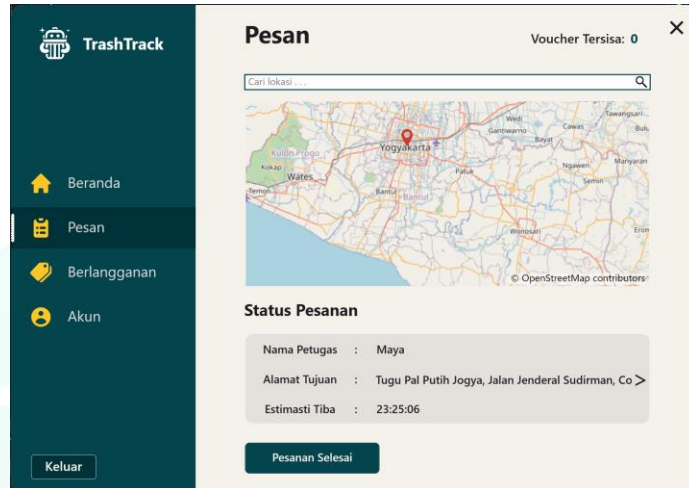
3. Main Page (Main Window) – Beranda

The screenshot shows the main page titled "Beranda". It features a dark teal sidebar with the TrashTrack logo and navigation items: Beranda, Pesan, Berlangganan, and Akun, plus a "Keluar" button. The main content area is light beige and contains a banner for "Tambah Hari Tambah Bersih" with an image of a boat. Below the banner are two sections: "Pesanan Aktif" with fields for "Petugas" and "Estimasi", and "Voucher Tersisa" showing "28 Voucher".

User yang berhasil mendaftar dan berhasil login akan dihadapkan dengan Main Page yang terdapat navigation bar disebelah kiri untuk berpindah ke sub-page yang terdiri dari beranda, pesan, berlangganan, dan akun. Tampilan awal setelah berhasil

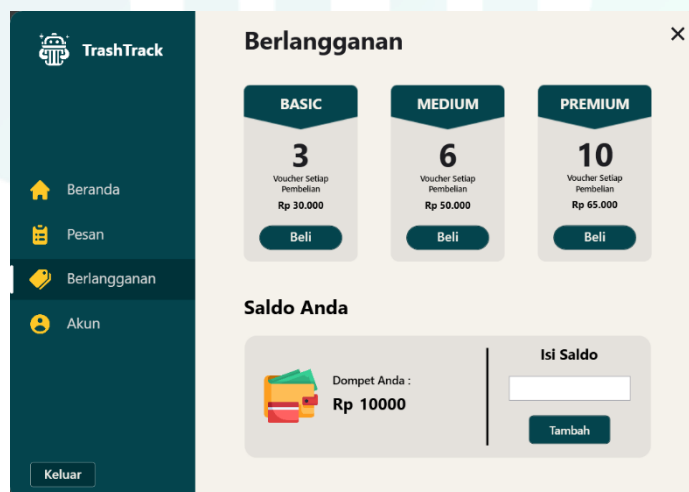
login adalah beranda yang berisikan ringkasan singkat mengenai voucher yang dimiliki dan status pesanan yang aktif.

4. Main Page – Pesan



Bagian ini merupakan bagian utama dari aplikasi ini yang digunakan untuk memesan tukang sampah. Pertama user harus memasukkan alamat tujuan sampah diambil, kemudian barulah memesan tukang sampah untuk datang dengan menekan tombol pesan. Apabila user tidak memasukkan alamat terlebih dahulu, MessageBox akan muncul dengan peringatan untuk memasukkan alamat terlebih dahulu. Map yang ada dalam page ini merupakan API yang difungsikan user untuk memasukkan alamat tersebut.

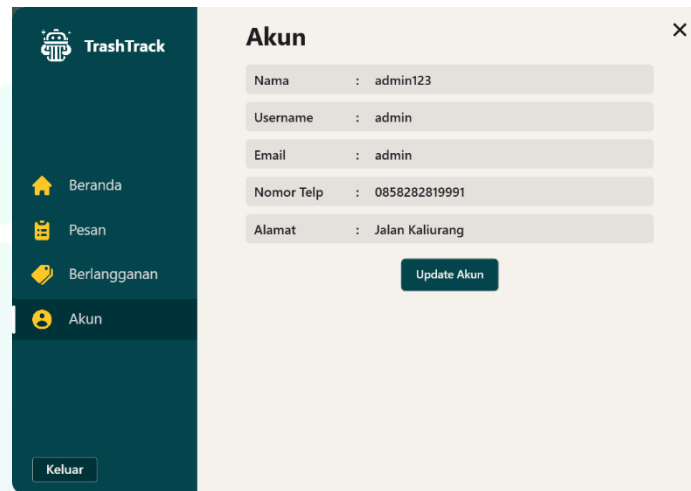
5. Main Page – Berlangganan





Bagian ini adalah page yang ditujukan untuk membeli voucher (semacam kupon) untuk memesan tukang sampah. Aplikasi ini juga dilengkapi dengan saldo akun sehingga untuk membeli voucher, diperlukan saldo yang dapat diisi oleh user. Setelah membeli salah satu dari ketiga voucher tersebut, akun user yang terdapat di dalam database akan terupdate. Begitu juga dengan saldo akun.

6. Main Page – Akun



Bagian ini merupakan page terakhir yang difungsi untuk melihat profil dari user dan juga dapat melakukan perubahan data dari page ini. Data yang berasal dari database dipanggil ke dalam page ini. Namun, merubah password dari sebuah user akun masih belum bisa dilakukan dipage ini.

7. SQL Server Database

user_id	username	nama	email	password	address	noTelp	Saldo
3	admin	admin123	admin	admin	Jalan Kaliurang	0858282819991	10000
6	aku	aku	aku@mail	aku	085712974454	babakan	0
7	dia	dia	dia@mail	dia	085712974454	babakan	0
8	suharti	txtNama	aku@mail.com	12345	bbkn	087	0
9	bagas	bagas31	bagas@mail.com	1234	bbk	1234	0
10	jkw	jkw	jkw	jkw	solo	087	0
11	jk	jk	jk	jk	jk	jk	0
12	anggitto	anggitto	anggitto	anggitto	anggitto	anggitto	0

SubscriptionID	UserID	Voucher
1	3	28
2	9	8
3	3	29
4	10	11

PesananID	user_id	NamaPetugas	Alamat	Latitude	Longitude	EstimasiPengambilan	Status
1	3	Supardi	Poncosari, Srandakan, Bantul Regency, Special Re...	110.224719388597308	-7.976897391595650	01/01/01 12:00:00 AM	Belum diambil
2	3	Supardi	Poncosari, Srandakan, Bantul Regency, Special Re...	110.238763188427882	-7.970575563873043	01/01/01 12:00:00 AM	Belum diambil
3	3	Supardi	Poncosari, Srandakan, Bantul Regency, Special Re...	110.236209770276872	-7.969311188605979	01/01/01 12:00:00 AM	Belum diambil
4	3	Siti	Poncosari, Srandakan, Bantul Regency, Special Re...	110.236209770276872	-7.969311188605979	21/12/23 9:24:24 AM	Belum diambil
5	3	Supardi	Trimurti, Srandakan, Bantul Regency, Special Regi...	110.254083697333982	-7.949080619513736	01/01/01 12:00:00 AM	Belum diambil



Gambar diatas adalah potong layar dari Microsoft SQL Server Management Studio yang menampilkan database dari aplikasi TrashTrack. Terdapat table user, table subscription, table pesanan aktif, dan table riwayat pesanan.

Jobdesc

Semua anggota melakukan pekerjaan kelompok secara adil. Semua anggota kelompok turut serta dalam perencanaan aplikasi dan pengembangan bersama seperti testing aplikasi untuk melihat bug, melihat kekurangan elemen, dan sebagainya. Namun, dalam pengembangan aplikasi untuk menjadi satu kesatuan utuh, setiap anggota mempunyai peran yang berbeda.

- Anggito Muhammad Amien

Pertama, membuat desain database yang digunakan untuk menyimpan data user, berlangganan, pesanan, dan histori pesanan pada aplikasi ini. Hal tersebut juga mencakup pembuatan fungsi-fungsi untuk melakukan CRUD (*Create, Read, Update, Delete*) ke Database. Kedua, juga berperan dalam mendesain Class user, pesan, dan subscription beserta dengan method yang dimilikinya. Ketiga, berperan dalam mengintegrasikan kode agar data pada database yang dimiliki oleh setiap entitas pengguna dapat ditampilkan dalam GUI.

- Reihan Athar Zildaniar

Berperan dalam pembuatan MapControl untuk memungkinkan kendali map pada umumnya dan pemberian pin pada map yang informasi koordinatnya tersimpan dan dikirim menjadi API request untuk mendapatkan detail lokasi. Berperan dalam pemberian batasan pada map. Apabila lokasi terlalu jauh dari DIY dan sekitarnya, pesanan tidak dapat dibuat. Berperan dalam pembuatan search bar dengan API communication dan filtering API response untuk menampilkan hasil pencarian serta memunculkan pin yang akurat pada map apabila salah satu hasil pencarian diklik.

- Syaifullah Hilmi Ma'arij

Berperan dalam pembuatan desain awal di Figma untuk memudahkan pengembangan aplikasi, berperan dalam merancang struktur aplikasi dengan metode MVVM (Model-View-ViewModel) agar aplikasi mudah dipahami dan dikembangkan



oleh developer lain (anggota yang lain), serta berperan dalam mengaplikasikan desain Figma ke dalam code WPF (frontend).

Source Figma:

<https://www.figma.com/file/ILfhqROKkRjJ8fkPMWIFyC/TrashTrack-Project?type=design&node-id=0%3A1&mode=design&t=E0gLYsZMD5Ru8EUV-1>

