**Laporan *Progress* Sistem Informasi Manajemen Organisasi menggunakan *HTML5, CSS3, Javascript*, dan *Firebase Firestore***

**PEMROGRAMAN WEB E081**



**Disusun Oleh :**

**Kelompok 1**

|  |  |  |
| --- | --- | --- |
| 1 | Triyono Rifan | 20081010003 |
| 2 | Novandi Kevin Pratama | 20081010005 |
| 3 | Alief Indy Millani | 20081010013 |
| 4 | Galan Ahmad Defanka | 20081010031 |

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

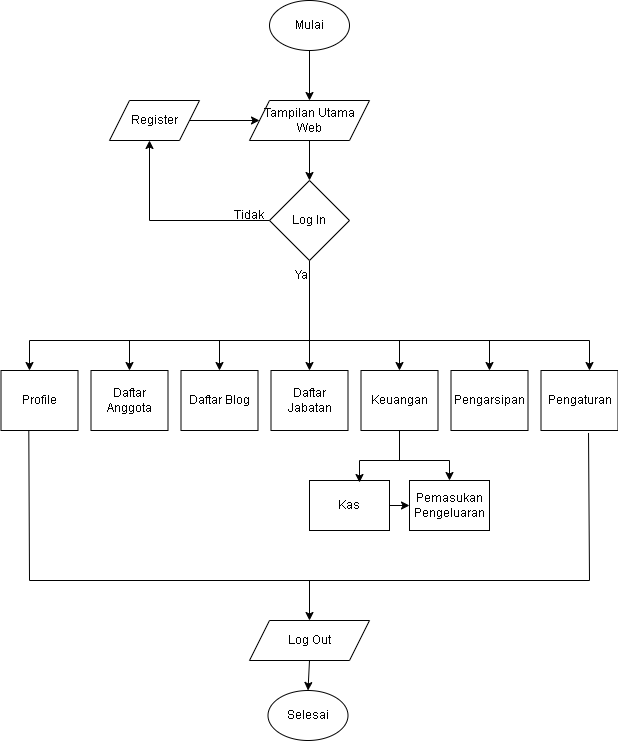
**UPN “VETERAN” JAWA TIMUR**

**2022**

# Perancangan Sistem Manajemen Organisasi

Pada tugas *final project* kali ini, kami membuat SIMANSI (Sistem Manajemen Organisasi). SIMASI adalah sebuah sistem yang akan mengelola dan menata sebuah organisasi secara umum untuk mempermudah pendataan data yang berada dalam sebuah organisasi. Sistem yang dibuat akan mengolah beberapa data dalam sebuah organisasi antara lain adalah data anggota, data kas, data keuangan, data blog, dan data arsip. Data tersebut nantinya akan disimpan dalam sebuah basis data. Sistem basis data yang dipakai dalam SIMANSI adalah *Firebase*.

Sebelum itu, kami membuat rancangan sistem terlebih dahulu berbentuk *flowchart*. Untuk gambarnya adalah sebagai berikut.

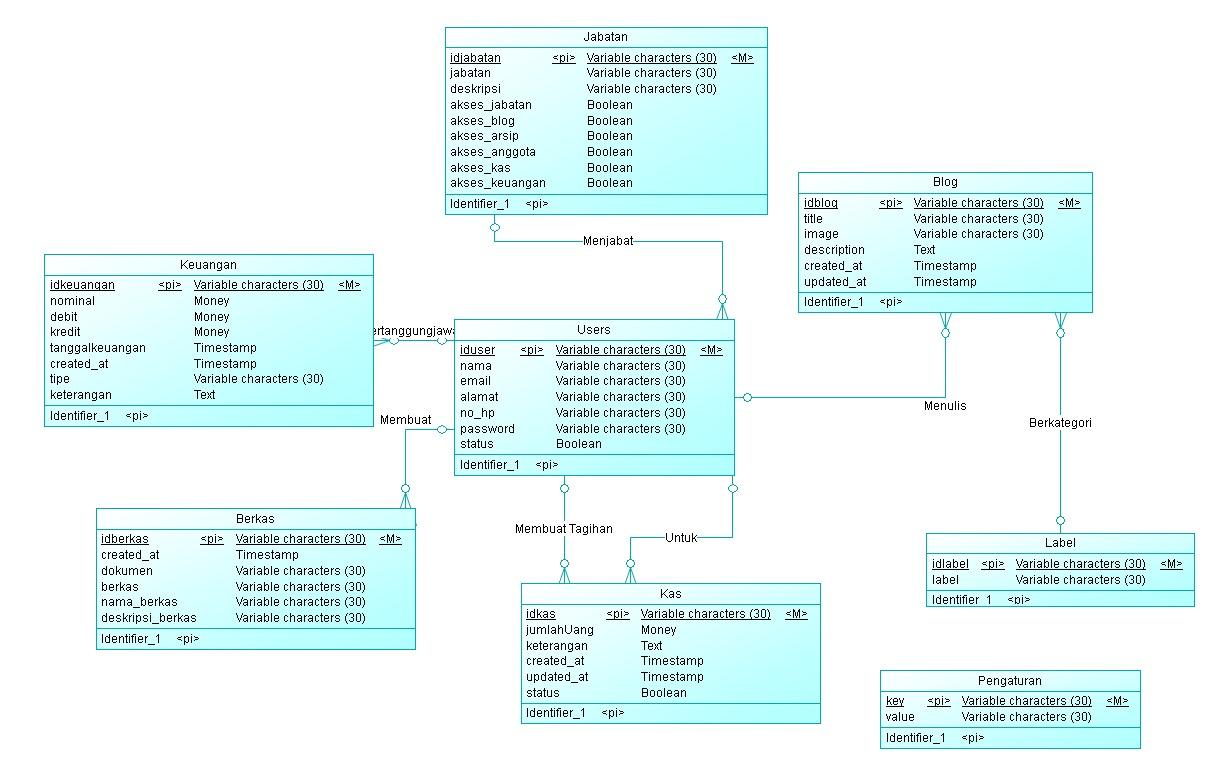


Gambar 1. *Flowchart* SIMANSI

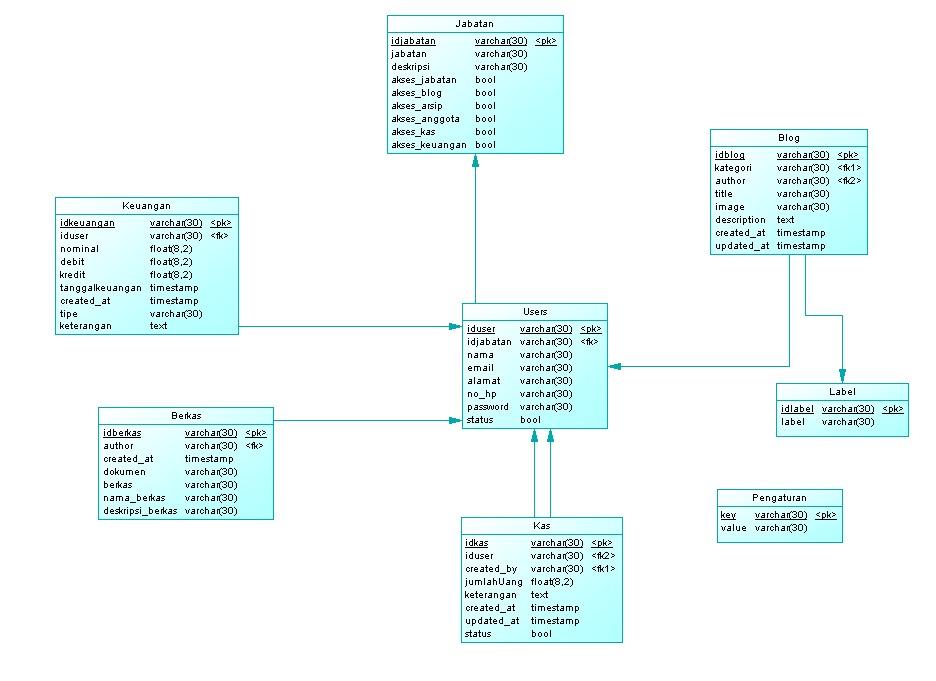
Lalu ada penjelasan alur *flowchart* sebagai berikut

1. Mulai
2. Masuk ke dalam *link* web
3. Tekan tombol log in
4. Jika belum mendaftar maka register terlebih dahulu, jika sudah isi dengan email dan *password*
5. Setelah itu masuk ke dalam *dashboard*, di dalam *dashboard* bisa memilih fitur yang akan digunakan
6. Setelah fitur dipilih, maka bisa langsung menggunakannya.
7. Jangan lupa *save* jika telah menggunakan fitur.
8. Tekan *sign out* jika sudah menggunakan web.
9. Selesai

Setelah itu kami membuat CDM dan PDM untuk rancangan basis data kami.



Gambar 2. CDM SIMANSI



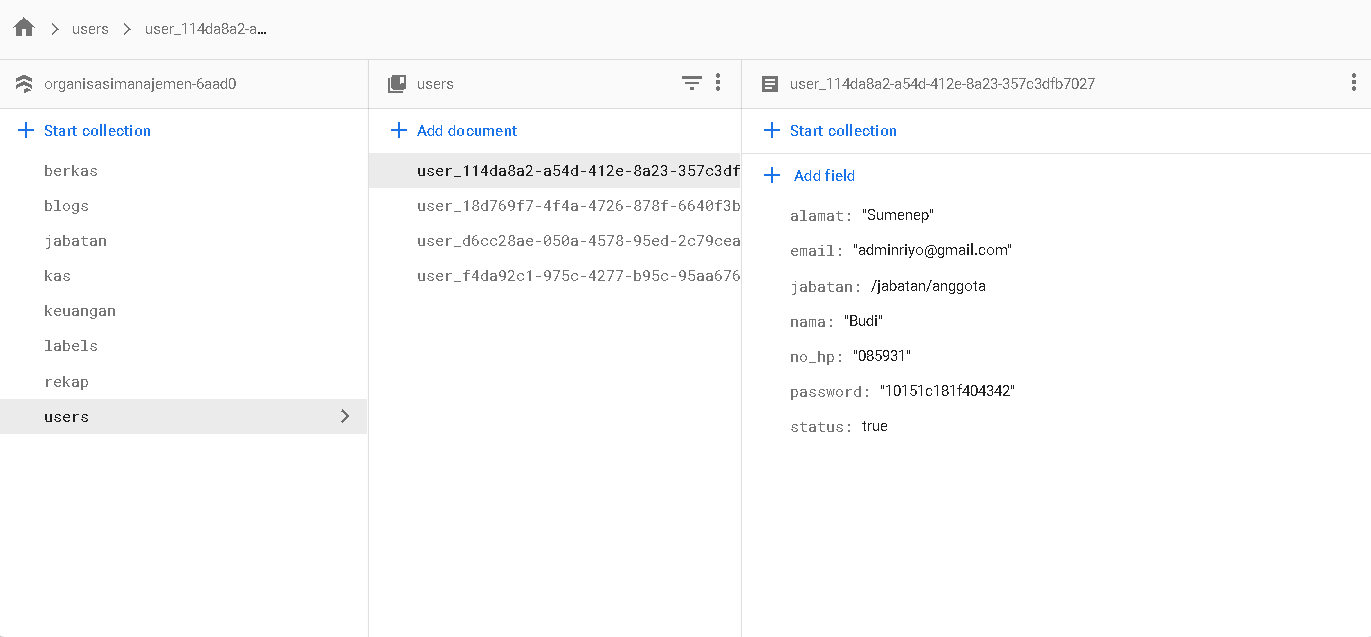
Gambar 3. PDM SIMANSI

Pada basis data tersebut memiliki tujuh entitas yang digunakan dalam sistem kami. Ketujuh tabel tersebut adalah *user*, jabatan, blog, label, keuangan, berkas, kas, dan pengaturan. Untuk masing-masing tabel memiliki beberapa atribut sebagai berikut.

* User : iduser, nama, email, alamat, no hp, password, status
* Jabatan : idjabatan, jabatan, deskripsi, akses\_jabatan, akses\_blog, akses\_arsip, akses\_anggota, akses\_kas, akses\_keuangan
* Blog : idblog, tittle, image, description, create\_at, update\_at
* Label : idlabel, label
* Keuangan : idkeuangan, nomina, debit, kredit, tanggal keuangan, created\_at, tipe, keterangan
* Berkas : idberkas, created\_at, dokumen, nama\_berkas, deskripsi\_berkas
* Kas : idkas, jumlahUang, keterangan, created\_at, updated\_at, status
* Pengaturan : key, value

**Pengaplikasian Basis Data pada *Firebase Firestore***

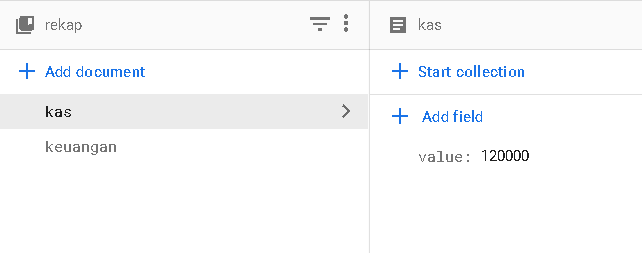
*Firebase* adalah layanan cloud yang disediakan Google untuk para developer. Layanan *cloud* yang paling sering digunakan adalah layanan *firestore*. Layanan ini menyediakan layanan basis data berbentuk *NoSQL*. Dalam pengimplementasian basis data pada *firestore* mengalami sedikit perbedaan. Struktur pada *firestore* hampir sama strukturnya dengan JSON (*Javascript Notation Object*). Berikut hasil implementasi basis data yang dirancang pada *Firebase Firestore*:



Gambar 4. *Collection* pada *Firebase Firestore*

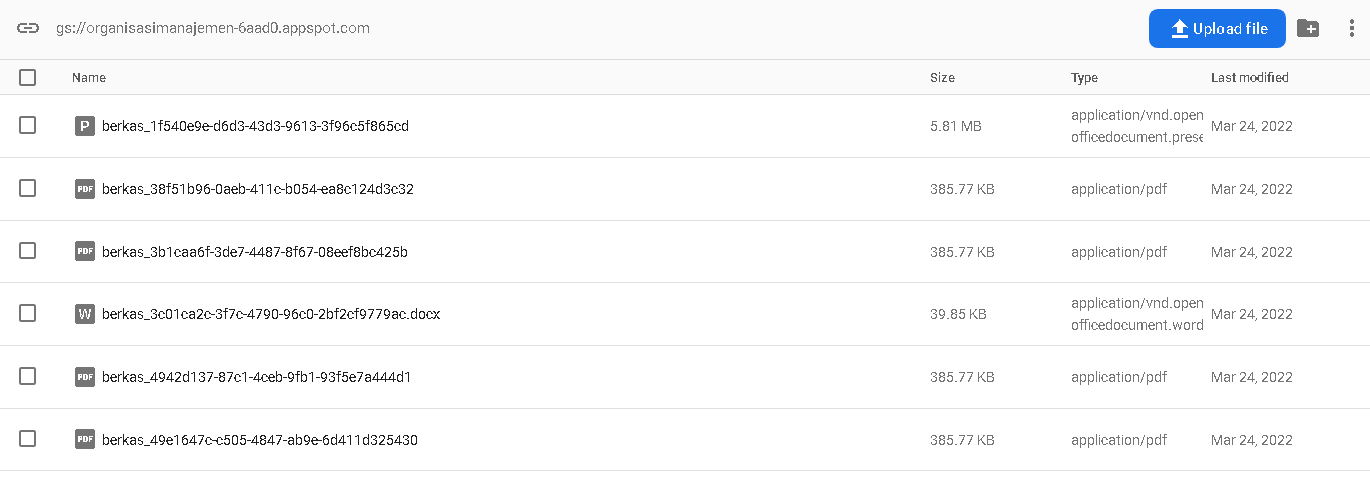
Struktur basis data *Firebase Firestore* terdiri dari *collection*, *document*, dan *field*. Untuk melakukan *foreign* dapat menggunakan *referrer* sehingga akan memanggil *document* lain sebagai *referrer field*. Hal ini akan mendukung ketika suatu saat aplikasi ingin dikembangkan dikarenakan sifat dari basis data *NoSQL* sendiri adalah fleksibel dan antar *document* dapat memiliki *field* yang berbeda satu sama lain.

Namun, untuk basis data *NoSQL* memiliki beberapa kekurangan dibandingkan basis data *SQL*. Salah satunya adalah tidak tersedia *aggregate function* seperti *sum* untuk menjumlahkan *value*. Dalam mengatasi kekurangan ini, dibuat *collection* baru untuk melakukan rekap jumlah saldo keuangan dan kas. Ketika melakukan penginputan data pada *collection* keuangan ataupun kas akan menambahkan *value field* dari rekap. Berikut struktur *collection* rekap:



Gambar 5. *Collection* Rekap pada *Firebase Firestore*

Selain menggunakan layanan *Firebase Firestore* untuk menyimpan data, kami juga menggunakan *Firebase Cloud Storage* sebagai media penyimpanan foto dan arsip berkas. Pada *value* akan menyimpan *link download* dari berkas sehingga dapat digunakan untuk selanjutnya.

 Gambar 6. *Cloud Storage* pada *Firebase*

Untuk mengakses data dari *firebase* menggunakan *javascript*. Dalam pendefinisian *javascript* harus bertipe module. Dan melakukan konfigurasi yang telah ada pada *firebase console*. Data yang dipanggil akan bersifat *asynchronous* lalu akan dilakukan pengulangan dan manipulasi elemen pada *HTML*. Dalam manipulasi ini dapat menggunakan *JQuery* untuk mempermudah manipulasi elemen.

**Tampilan Badan SIMANSI**

Pembuatan SIMANSI (Sistem Manajemen Organisasi) secara umum memiliki dua ranah pengaplikasian yakni secara pelanggan dan secara admin. Tampilan yang disuguhkan secara umum juga sama dari kedua ranah pengaplikasian namun, secara admin akan memiliki beberapa fitur lebih yang tidak bisa didapatkan pelanggan. Istilahnya, pelanggan ini adalah seorang pengunjung dan admin adalah para anggota organisasi yang akan menata tampilan aplikasi beserta isinya.

Badan SIMANSI yang dirancang memiliki beberapa bagian yakni index.html sebagai halaman atau badan utama, blog.html sebagai badan dari tampilan blog yang dapat digunakan untuk menaruh informasi atau kegiatan-kegiatan seputar organisasi, dan ada tampilan *login* serta registrasi untuk bagian menentukan bahwa pengunjung hanyalah pengunjung biasa atau salah satu anggota organisasi (admin).

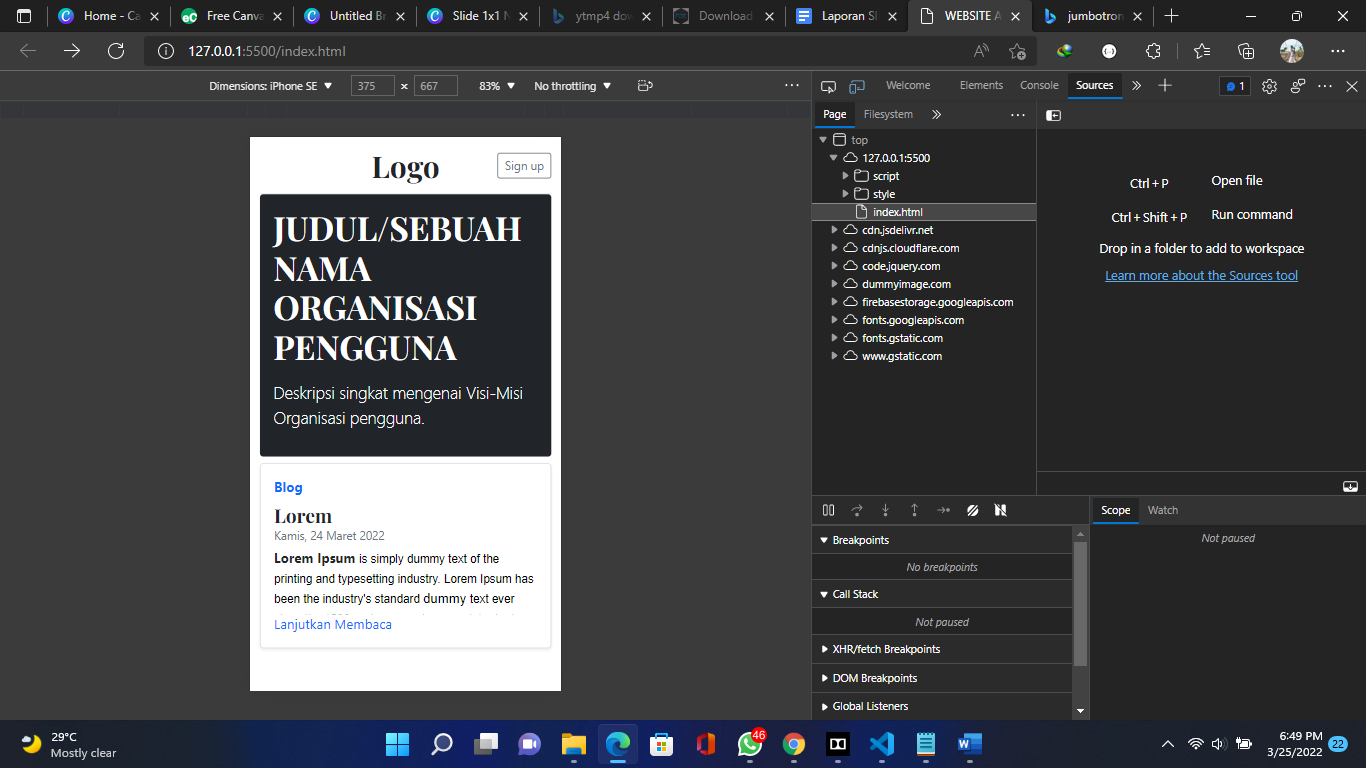
Berikut adalah beberapa tampilan sementara SIMANSI.



Gambar 7. Tampilan Sementara Halaman Utama

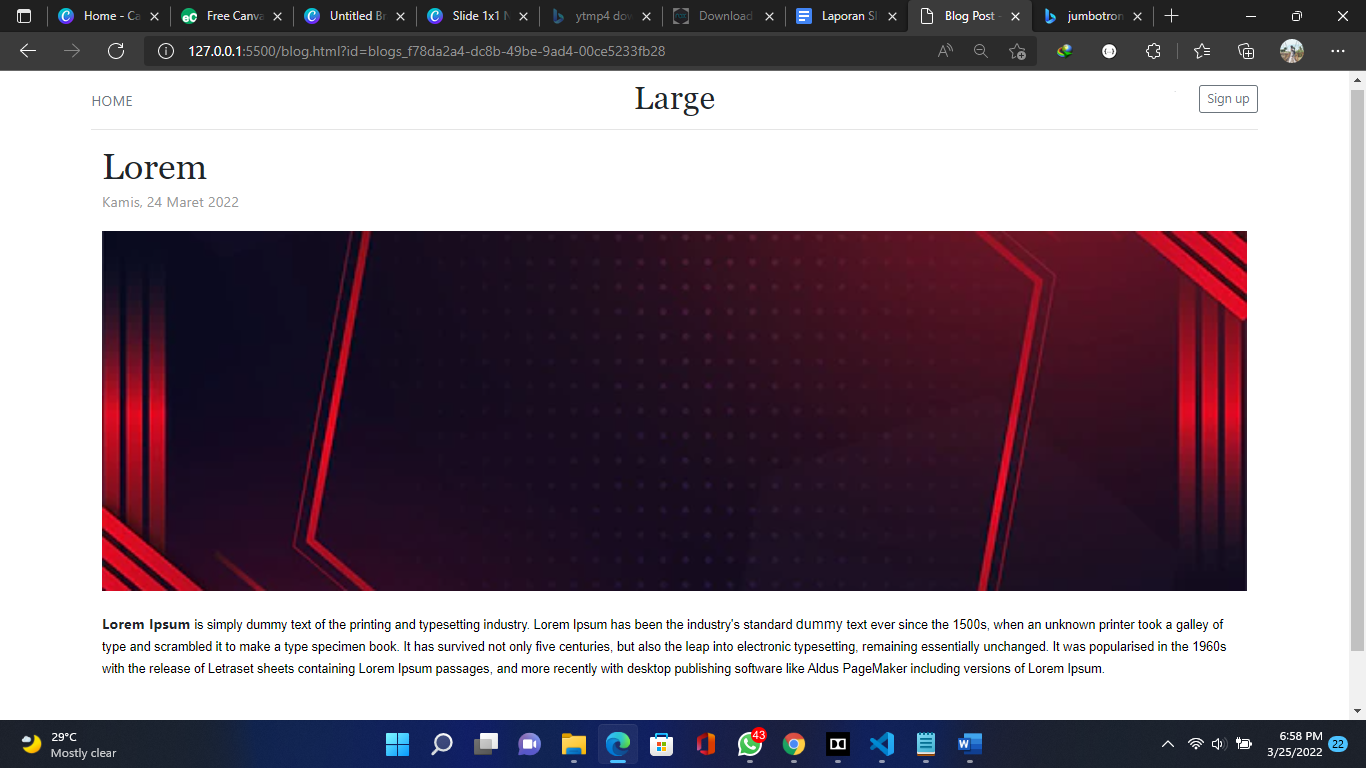
Pada *header* terdapat dua atribut yakni ada tulisan “Logo” yang nantinya akan diubah menjadi logo organisasi pemakai dan tombol “*Sign up*” adalah tombol yang akan mengarahkan ke proses *login*. Terdapat blog yang dapat diklik akan menuju tampilan Blog atau konten yang disuguhkan.

Tampilan sementara pada halaman utama juga sudah *responsive* atau dalam artian akan menyesuaikan ukuran perangkat baik itu laptop maupun ponsel. Gambarnya bisa dilihat berikut.



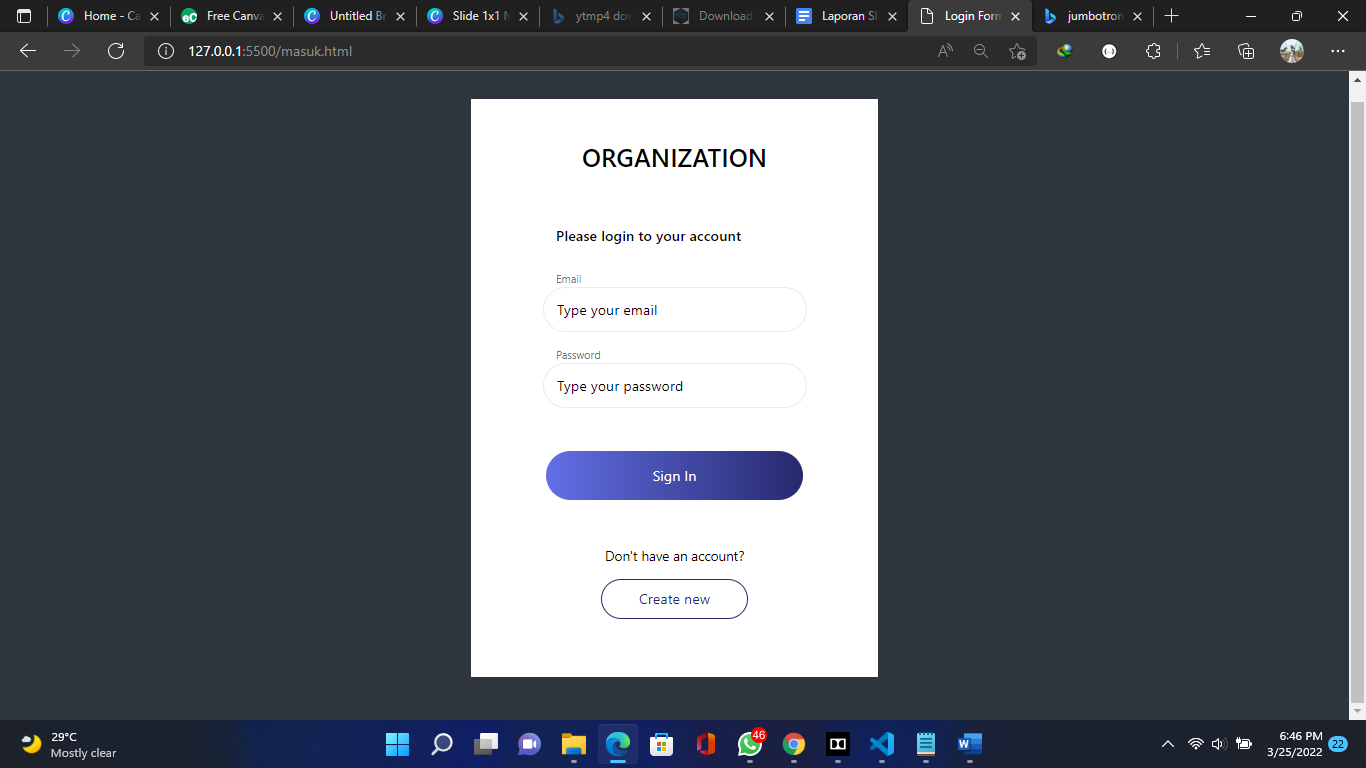
Gambar 8. Tampilan Sementara Halaman Utama *Responsive*

Seluruh elemen yang termasuk tadi akan secara otomatis menyesuaikan dengan rasio ukuran layar perangkat yang digunakan karena dalam *codingan* bagian *class element* terdapat *text-center* dan dalam *css* juga diatur *width* 100%.



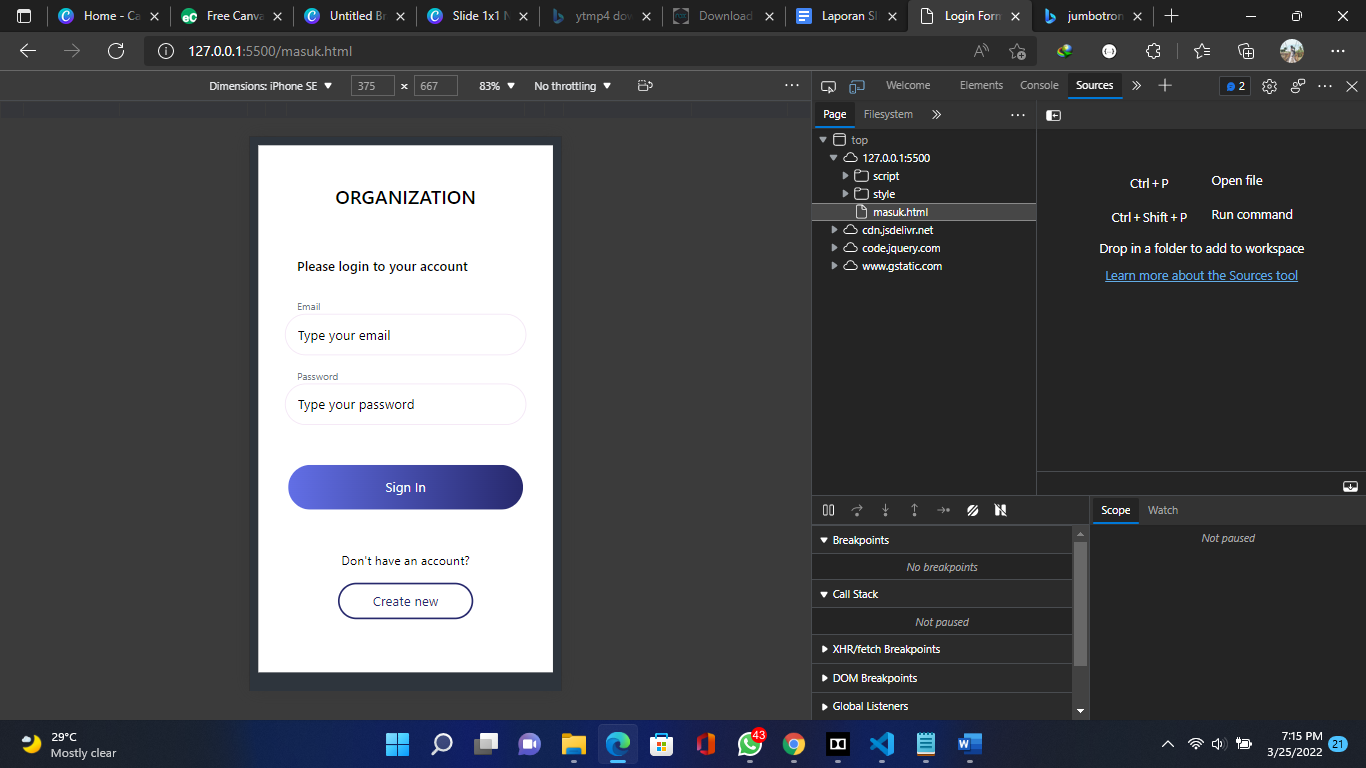
Gambar 9. Tampilan Sementara Blog

*Header* pada halaman Blog sama seperti halaman utama namun, ada tambahan “*Home*” yakni jika diklik akan kembali ke halaman utama. Pada tampilan yang terlihat di atas adalah tampilan yang bisa dinikmati oleh pelanggan dan admin.

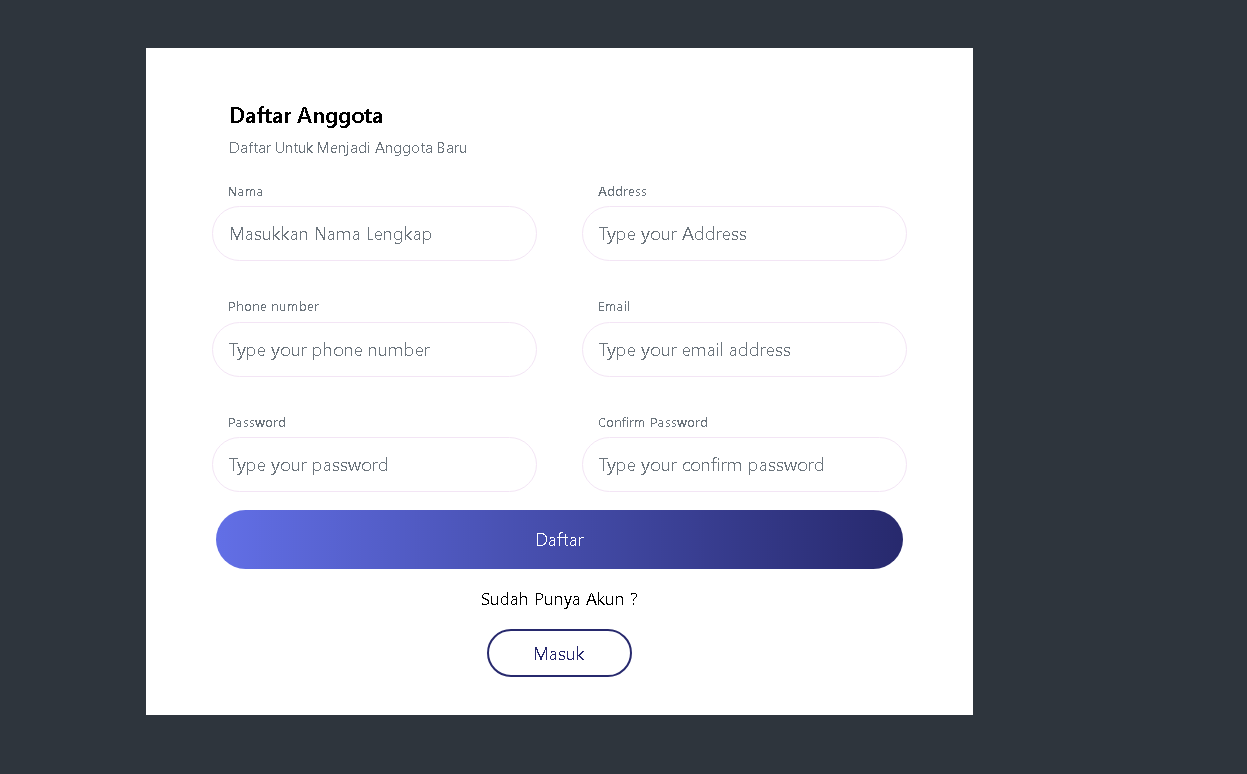


Gambar 10. Tampilan Sementara *Login*

Setelah klik tombol “*Sign up*” maka akan beralih ke tampilan di atas atau proses *login*. Dalam tampilan *login* terdapat beberapa elemen yang bisa dilihat terutama pada kolom pengisian. Terdapat keterangan “*Type your email*” itu menggunakan *placeholder* yakni sebagai keterangan selain sub bab seperti “Email” dan “*Password*”. Pada tampilan Tombol *Sign In* dihias menggunakan gradasi warna pada *css* antara warna biru dan ungu gelap dengan pola *linear-gradient*, sama halnya seperti tombol “*Create New*”.



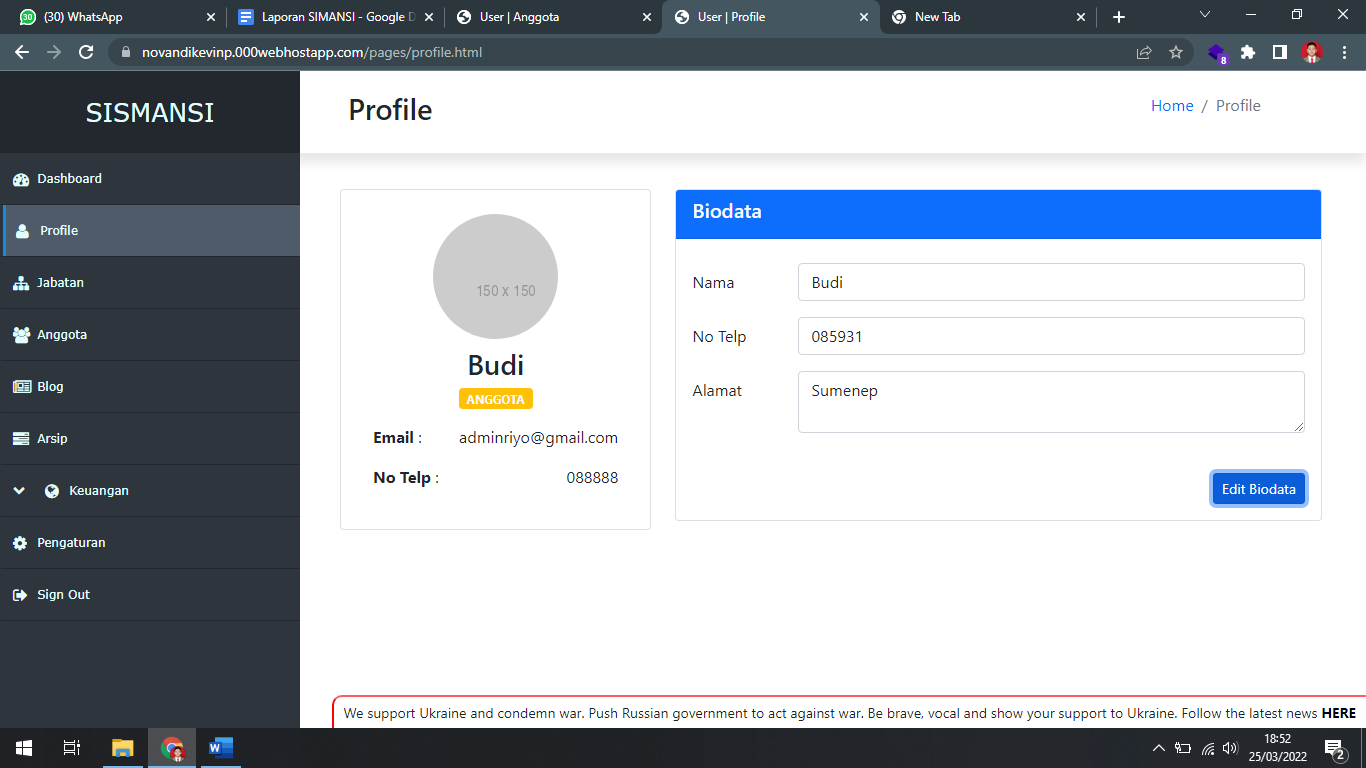
Gambar 11. Tampilan Sementara *Login Responsive*





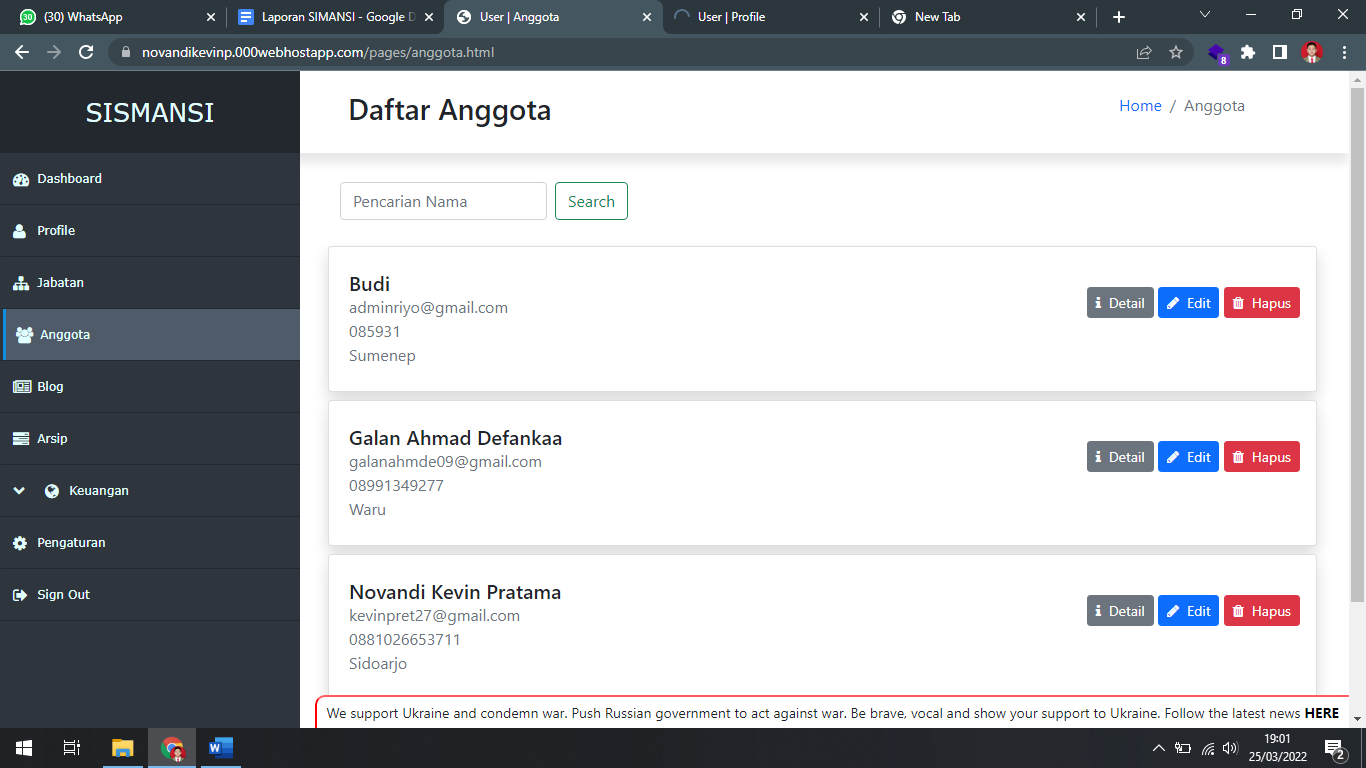
Gambar 12. Tampilan Sementara Registrasi

Pada bagian registrasi pada dasarnya komponen elemen-elemennya sama seperti proses *login*. Tampilan registrasi ini juga secara umum akan ada dua, yang pertama yakni saat menggunakan laptop atau tablet dan yang kedua saat menggunakan ponsel. Tampilan gambar di atas adalah tampilan saat menggunakan laptop sedangkan untuk ponsel tadi akan menjadi satu kolom.



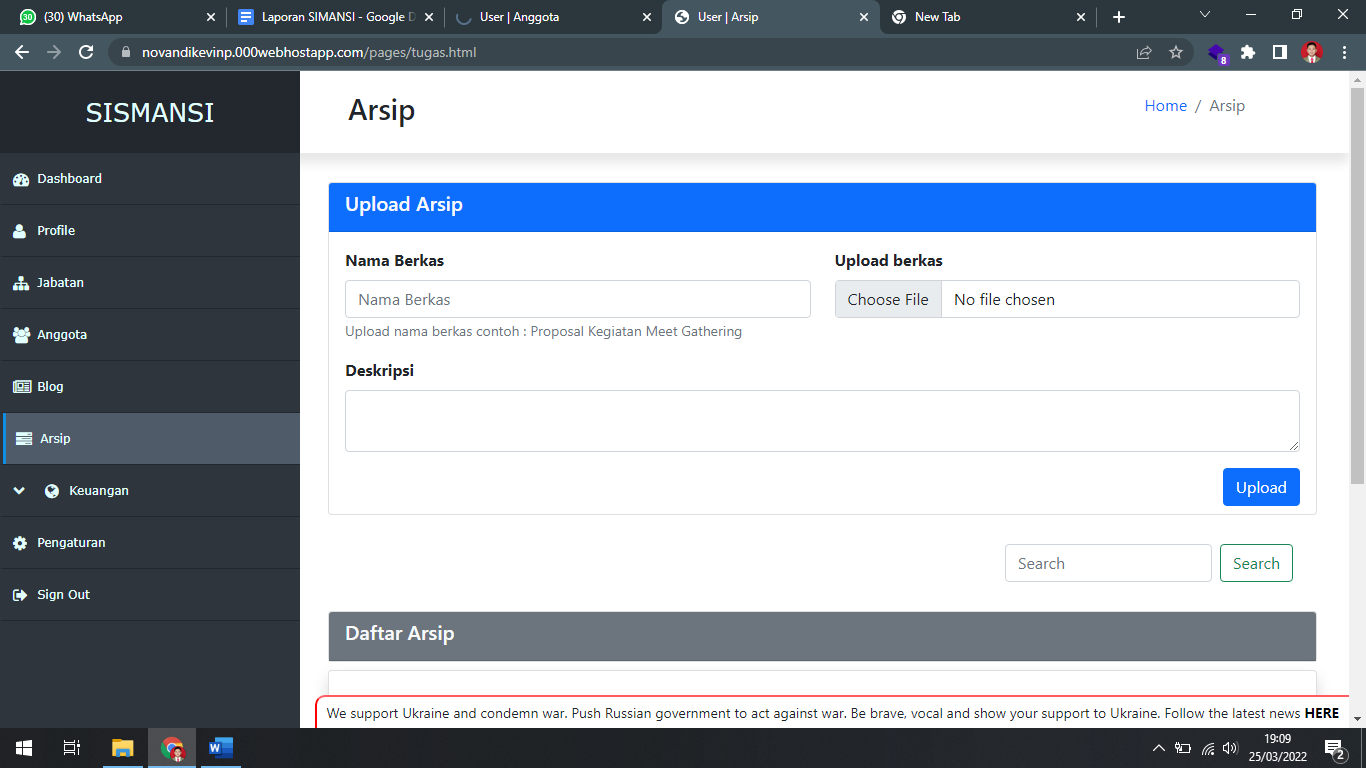
Gambar 14. Tampilan Sementara Profil

Ketika *user* sudah berhasil registrasi pada halaman blog, maka *user* dapat mengakses *dashboard* seperti tampilan di atas. terdapat *navbar* pada bagian atas menunjukkan bagan informasi halaman dan judul aplikasi yakni SISMANSI. Terdapat *sidebar* pada bagian kanan untuk memungkinkan pengguna dalam mengakses setiap halaman pada bagian ini. Pada bagian profil ini merupakan bagian pribadi *user* yang dapat memodifikasi foto profil, nama, nomor telepon dan alamat pada bagian tersebut.



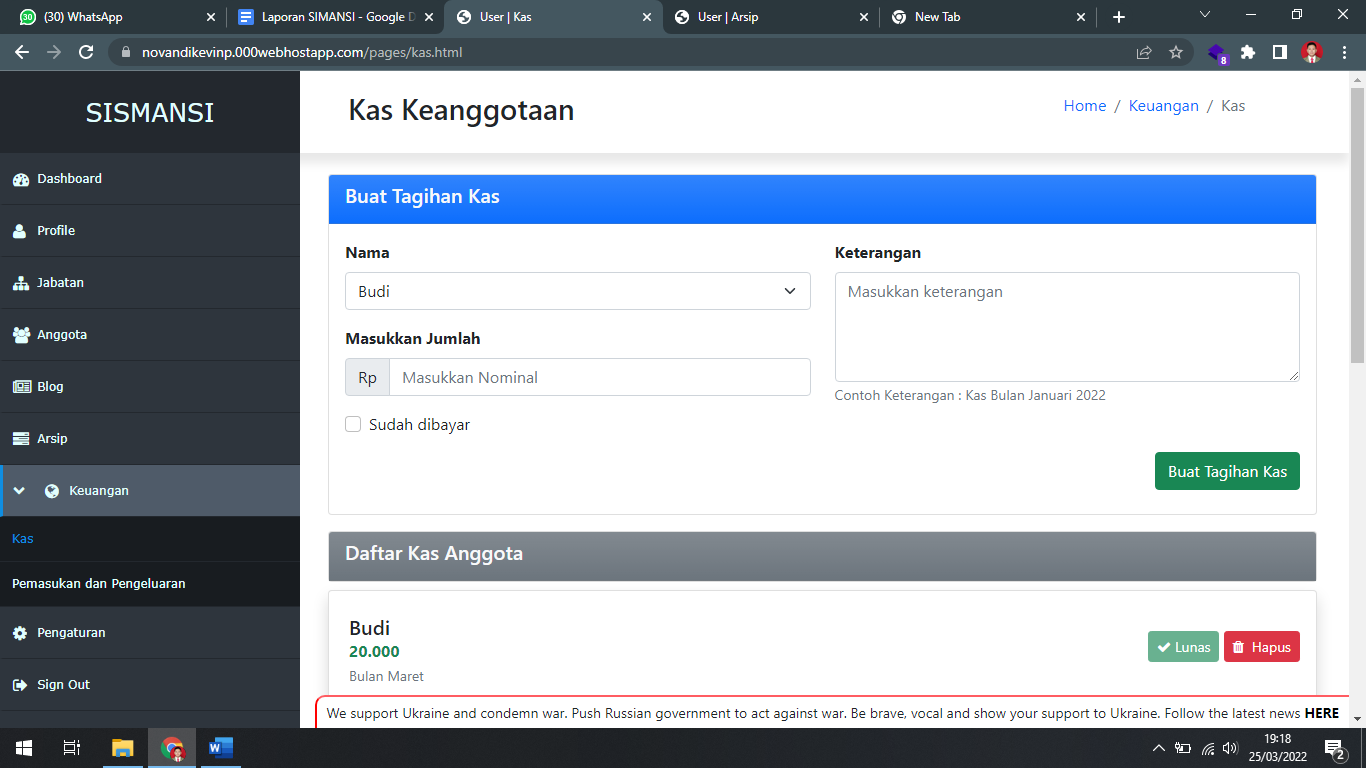
Gambar 15. Tampilan Sementara Daftar Anggota

Halaman daftar Anggota menampilkan semua daftar anggota yang terdaftar pada aplikasi. pada daftar tersebut kita dapat melihat detail *user* memodifikasi data, dan menghapus data.



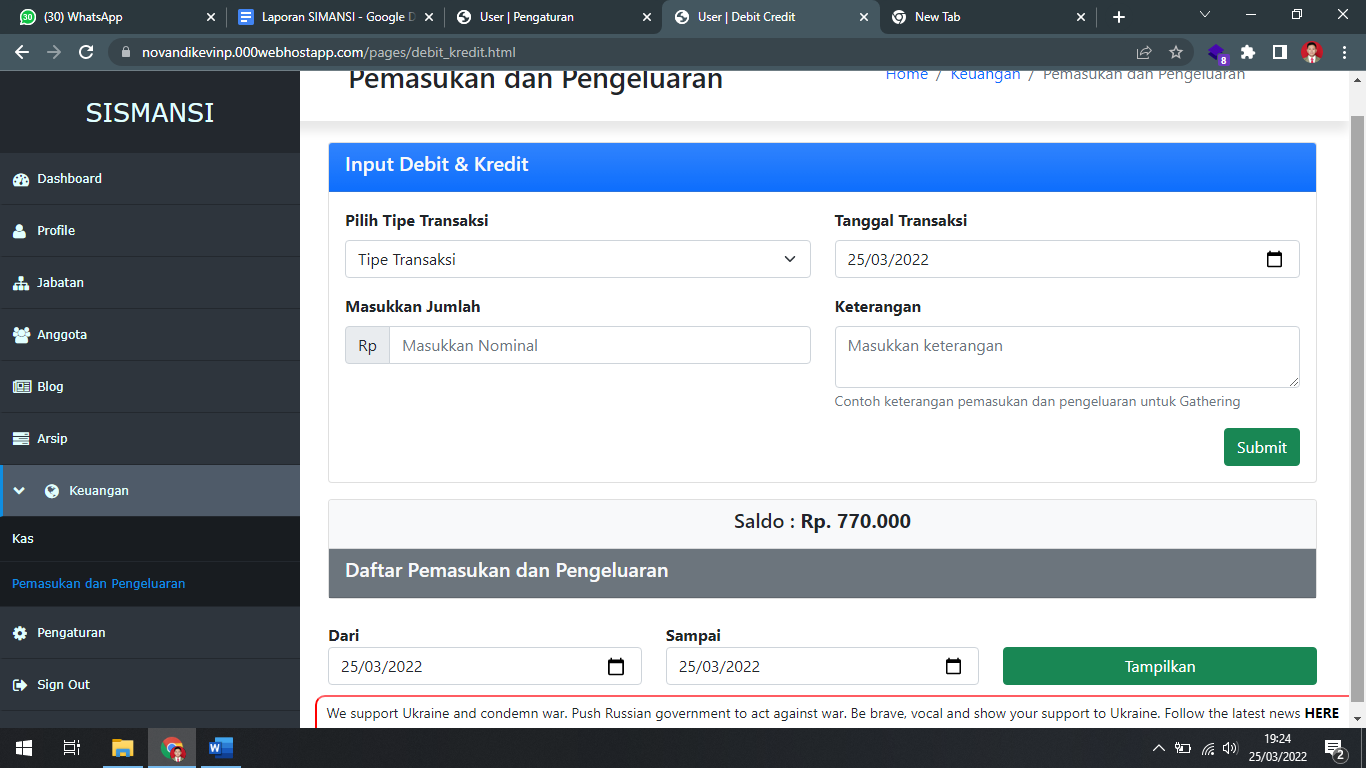
Gambar 16. Tampilan Sementara Daftar Berkas

Pada halaman ini *user* dapat *mengupload* hasil berkas untuk arsip manajemen. User dapat menyimpan berbagai macam *file* secara daring agar aman jika kelak dibutuhkan kembali dan dapat melihat daftar dokumen yang telah *diupload* pada halaman ini.



Gambar 17. Tampilan Sementara Daftar Kas

Halaman ini untuk mengelola keuangan pada setiap anggota. pada umumnya setiap organisasi pasti mengumpulkan dana untuk berjalannya sebuah organisasi. dan halaman ini membantu hal tersebut dan juga dapat menampilkan daftar tanggungan Kas anggota telah lunas atau belum lunas.



Gambar 18. Tampilan Sementara Daftar Keuangan

Halaman Debit dan Kredit digunakan untuk mengelola proses transaksi keuangan yang digunakan. biasanya proses organisasi ini berlangsung pada perbendaharaan sebuah manajemen mahasiswa baik organisasi atau lainnya. Serta pada halaman ini dapat menampilkan sisa saldo terbaru.