



UNIVERSITAS SEMARANG
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
TEKNIK INFORMATIKA

Mobile Application

Modul Praktikum Mahasiswa

Oleh:

Alauddin Maulana Hirzan, S. Kom., M. Kom
NIDN. 0607069401

Daftar Isi

Pendahuluan	4
0.1 Mengenal Android	4
0.2 Mengenal Firestore dan Realtime Database	4
0.3 Mengenal JSON-Tree	5
0.4 Mengenal Flutter Framework	6
Persiapan Praktikum	7
0.5 Perangkat Keras	7
0.6 Perangkat Lunak	7
1 Praktikum 1	8
1.1 Mengenal Pengembangan Antarmuka Thunkable	8
1.2 Tutorial	8
2 Praktikum 2	16
2.1 Navigasi Halaman	16
2.2 Tutorial	16
3 Praktikum 3	23
3.1 Konten Tambahan	23
3.2 Tutorial	23
4 Praktikum 4	30
4.1 Konektivitas Database	30
4.2 Tutorial	30
5 Praktikum 5	41
5.1 Input Database	41
5.2 Tutorial	41
6 Praktikum 6	51
6.1 Read Database	51
6.2 Tutorial	51
7 Praktikum 7	59
7.1 Multi Data Entry dan Display	59
7.2 Tutorial	59

8 Praktikum 8 **72**

8.1 Proyek Aplikasi 72

8.2 Deskripsi Proyek 72

8.3 Daftar Jenis Aplikasi 72

Daftar Gambar

- 1 Perangkat Android 4
- 2 Realtime Database 5
- 3 Contoh JSON Tree 5
- 4 Flutter 6

- 1.1 Tampilan Laman Thunkable 8
- 1.2 Tampilan Laman Registrasi Akun 9
- 1.3 Tampilan Laman Registrasi Akun 9
- 1.4 Tampilan Laman Pembuatan Aplikasi 9
- 1.5 Tampilan Laman Utama Editor 10
- 1.6 Tampilan Laman Utama Blok 10
- 1.7 Tampilan Laman Mode **Design** dan Fitur Lainnya 11
- 1.8 Konfigurasi Aplikasi 11
- 1.9 Konfigurasi Aplikasi 11
- 1.10 Kembali ke Mode Designer 12
- 1.11 Memasukkan Label 12
- 1.12 Atur Ulang Komponen 13
- 1.13 Kustomisasi Label 13
- 1.14 Menambahkan komponen **Image** 13
- 1.15 Mengganti Gambar Komponen **Image** 14
- 1.16 Menambahkan Komponen **Button** 14
- 1.17 Mengubah Komponen **Button** 15
- 1.18 **Footer Aplikasi** dengan **Label** 15

- 2.1 Membuka Proyek Sebelumnya 16
- 2.2 Mengubah Layar Awal 17
- 2.3 Menambahkan Layar Baru 17
- 2.4 Mengubah Nama Layar 17
- 2.5 Tampilan Layar Kedua 18
- 2.6 Memodifikasi Label 18
- 2.7 Meletakkan 2 Tombol 19
- 2.8 Mengisi Deskripsi Fakultas 19
- 2.9 Membuka Mode Blocks 20
- 2.10 Menarik Blok Klik btnNext 20
- 2.11 Menarik Blok Navigasi 21
- 2.12 Menyatukan Potongan Blok 21
- 2.13 Mengubah Arah Navigasi 21
- 2.14 Menguji Aplikasi 22

2.15	Kembali ke Editing	22
2.16	Navigasi Mundur	22
3.1	Buka Projek	23
3.2	Buat Halaman Baru	24
3.3	Komponen Halaman P_SI	24
3.4	Komponen Halaman P_TI	25
3.5	Komponen Halaman P_IK	26
3.6	Komponen Halaman P_Par	26
3.7	Blok Aksi btnNext Tentang Kami	27
3.8	Blok Aksi btnKembali dan ProdiTI	27
3.9	Blok Aksi btnKembali dan ProdiIK	27
3.10	Blok Aksi btnKembali dan ProdiPar	28
3.11	Blok Aksi btnKembali	28
3.12	Pengujian Desain	29
4.1	Buka Web Firebase	30
4.2	Login ke Firebase	31
4.3	Membuka Console	31
4.4	Menambah proyek baru	32
4.5	Memberi nama proyek	32
4.6	Mematikan opsi analytic	33
4.7	Proyek selesai dibuat	33
4.8	Navigasi ke Realtime Database	34
4.9	Membuat Realtime Database	34
4.10	Memilih lokasi database	35
4.11	Mengatur keamanan database	35
4.12	Tampilan Realtime Database	36
4.13	Mengatur akses database	36
4.14	Mengubah akses database	37
4.15	Membuka Project Settings	37
4.16	Navigasi ke Your Apps	38
4.17	Memberi nama aplikasi	38
4.18	39
4.19	Membuka konfigurasi Thunkable	39
4.20	Menempel Data APIKey dan databaeURL	40
5.1	Halaman Baru	41
5.2	Mengubah Identitas Halaman	42
5.3	Menghubungkan Akses Halaman	42
5.4	Komponen Halaman Registrasi	43
5.5	Inisialisasi variable data	44
5.6	Aksi Tombol Batalkan	44
5.7	Aksi Tombol Bersihkan	45
5.8	Blok Awal Tombol Kirimkan	45
5.9	Inisialisasi RTDB	46
5.10	Aksi Pengiriman Data Tanpa Data	46
5.11	Blok Awal Untuk Data JSON	47
5.12	Konfigurasi Create Objek 1	47

5.13	Membuat JSON Bertingkat	48
5.14	Hasil Akhir JSON Bertingkat	48
5.15	Aksi Terakhir Kembali ke Home	49
5.16	Hasil Sementara	49
5.17	Hasil Akhir	50
6.1	Tambah Layar Baru	51
6.2	Tambah Komponen ke Layar	52
6.3	Masuk mode Block	52
6.4	Inisialisasi Variabel Clouds	53
6.5	Inisialisasi Variabel App	53
6.6	Mengatur isi dari itemlist	54
6.7	Eksekusi Kondisi Peminat	54
6.8	Komponen Set Variable	55
6.9	Komponen Set Variable	55
6.10	Mengambil Data Nama	56
6.11	Megambil Data Email dan Kontak	56
6.12	Komponen set Item	57
6.13	Menarik itemlist ke list	57
6.14	Tes Tampilan	58
6.15	Tes Tampilan	58
7.1	Buka Praktikum 6 dan Bagian Block - Registrasi	59
7.2	Hapus bagian dalam btnKirimkan	60
7.3	Memasukkan fungsi tambah data	61
7.4	Memasukkan Nama Variabel	62
7.5	Memasukkan Nama Variabel Dinamis	62
7.6	Memasukkan Blok List	63
7.7	Memasukkan Data ke List	63
7.8	Memasukkan Blok Pembersih	64
7.9	Hasil Register	64
7.10	Pindah Halaman Peminat	64
7.11	Hapus Blok di dalam Peminat	65
7.12	Variable data dengan Text kosong	65
7.13	Variable itemlist-v2 dengan empty list	66

Pendahuluan

0.1 Mengetahui Android

Sistem operasi Android adalah sistem operasi seluler untuk digunakan terutama untuk perangkat layar sentuh, ponsel, dan tablet. Desainnya memungkinkan pengguna memanipulasi perangkat seluler secara intuitif, dengan gerakan jari yang mencerminkan gerakan umum, seperti mencubit, menggesek, dan mengetuk.



Gambar 1: Perangkat Android

0.2 Mengetahui Firebase dan Realtime Database

Firebase adalah platform yang dikembangkan oleh Google untuk membuat aplikasi seluler dan web. Salah satunya produk yang sering digunakan di Firebase adalah Realtime Database. Firebase Realtime Database adalah database yang dihosting di cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung.



Gambar 2: Realtime Database

0.3 Mengenal JSON-Tree

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan. Mudah bagi manusia untuk membaca dan menulis. Sangat mudah bagi mesin untuk menguraikan dan menghasilkan. Ini didasarkan pada subset dari Standar Bahasa Pemrograman JavaScript ECMA-262 Edisi ke-3 - Desember 1999. JSON adalah format teks yang sepenuhnya bebas bahasa tetapi menggunakan konvensi yang akrab bagi pemrogram keluarga bahasa C, termasuk C, C++, C#, Java, JavaScript, Perl, Python, dan banyak lainnya. Properti ini menjadikan JSON sebagai bahasa pertukaran data yang ideal.

JSON dibangun di atas dua struktur:

- Kumpulan pasangan nama/nilai. Dalam berbagai bahasa, ini diwujudkan sebagai objek, catatan, struct, kamus, tabel hash, daftar kunci, atau array asosiatif.
- Daftar nilai yang diurutkan. Dalam kebanyakan bahasa, ini diwujudkan sebagai array, vektor, daftar, atau urutan.

Ini adalah struktur data universal. Hampir semua bahasa pemrograman modern mendukungnya dalam satu atau lain bentuk. Masuk akal bahwa format data yang dapat dipertukarkan dengan bahasa pemrograman juga didasarkan pada struktur ini.

```
Response Body  Select Body
{
  "results": [
    {
      "bioguide_id": "P000606",
      "birthday": "1948-08-15",
      "chamber": "house",
      "contact_form": "https://pittenger.house.gov/contact/email-me",
      "crp_id": "N00034416",
      "district": 9,
      "facebook_id": "376142742468386",
      "fax": "202-225-3389",
    }
  ]
}
```

Gambar 3: Contoh JSON Tree

0.4 Mengenal Flutter Framework

Flutter adalah kerangka kerja sumber terbuka oleh Google untuk membangun aplikasi multi-platform yang indah, dikompilasi secara asli, dari satu basis kode. Flutter mengubah proses pengembangan aplikasi. Buat, uji, dan terapkan aplikasi seluler, web, desktop, dan tersemat yang cantik dari satu basis kode.



Gambar 4: Flutter

Persiapan Praktikum

Agar praktikum dapat berjalan dengan lancar, mahasiswa diwajibkan memenuhi persyaratan berikut baik dalam bentuk perangkat keras maupun lunak:

0.5 Perangkat Keras

- Prosesor dengan 4 inti
- RAM minimal 4GB, rekomendasi 8GB
- HDD 10GB

0.6 Perangkat Lunak

Perangkat lunak berikut ini wajib diinstall oleh mahasiswa demi lancarnya praktikum:

- Browser

Bab 1

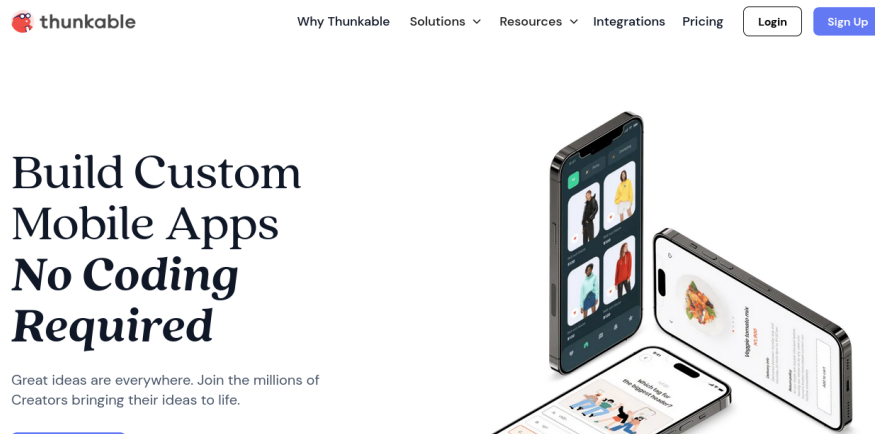
Praktikum 1

1.1 Mengetahui Pengembangan Antarmuka Thunkable

Di bagian ini mahasiswa diajarkan bagaimana membuat antarmuka aplikasi seluler dengan menggunakan **Thunkable**. Mahasiswa diwajibkan mengikuti tahapan yang dijelaskan di modul ini.

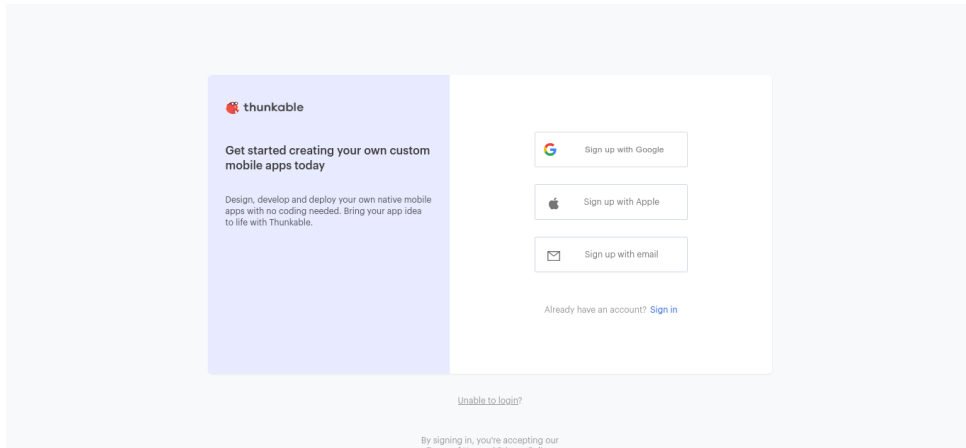
1.2 Tutorial

1. Mahasiswa membuka browser dan buka website <https://thinkable.com/>.



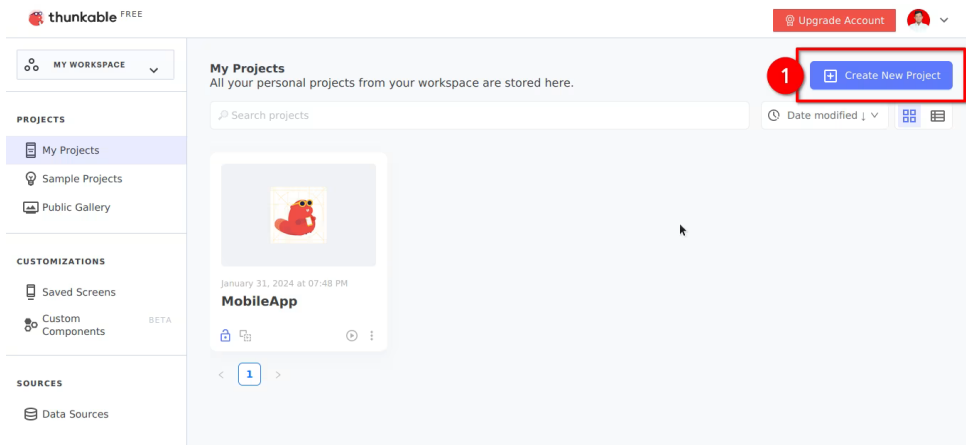
Gambar 1.1: Tampilan Laman Thunkable

2. Setelah masuk ke halaman **Thunkable**, daftar akun dengan menggunakan **GMail** dengan melakukan klik tombol yang ada.



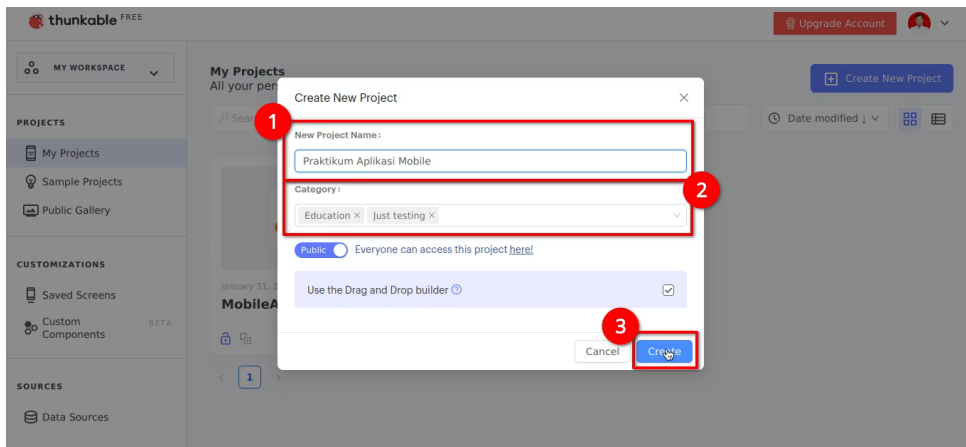
Gambar 1.2: Tampilan Laman Registrasi Akun

3. Sesudah membuat akun, maka mahasiswa akan bisa melihat halaman dasbor dari **Thinkable**. Buatlah **Create New Project** untuk memulai membuat projek baru.



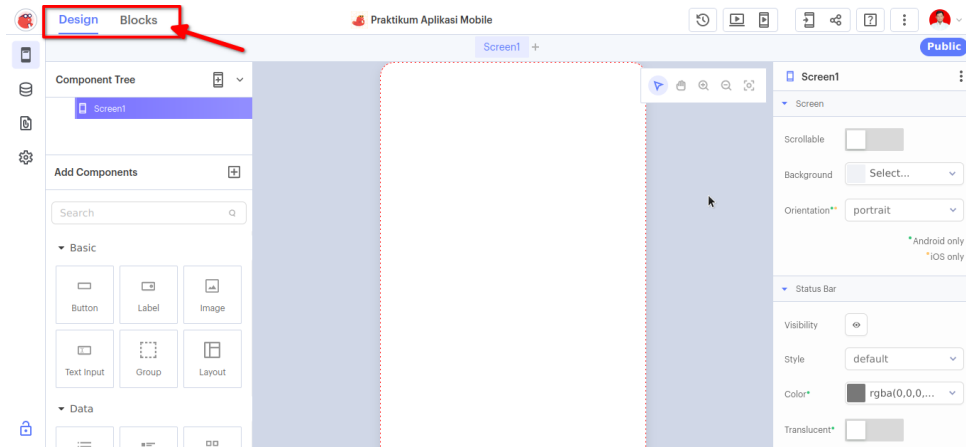
Gambar 1.3: Tampilan Laman Registrasi Akun

4. **Thinkable** akan menampilkan window kecil untuk informasi apa yang akan dibuat. Isi dengan contoh sebagai berikut lalu klik **Create**:



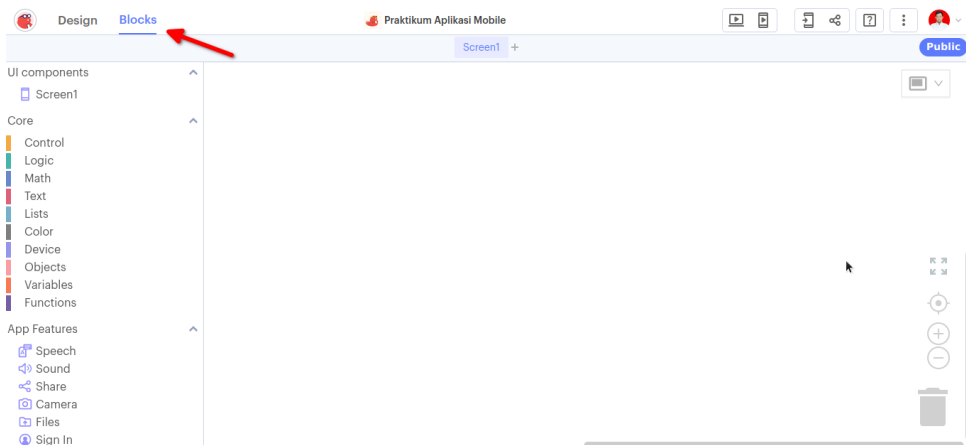
Gambar 1.4: Tampilan Laman Pembuatan Aplikasi

5. **Thunkable** lalu akan mengalihkan secara otomatis ke kanvas antarmuka aplikasi. Di halaman ini memiliki dua jendela utama yaitu **Design** dan **Block**



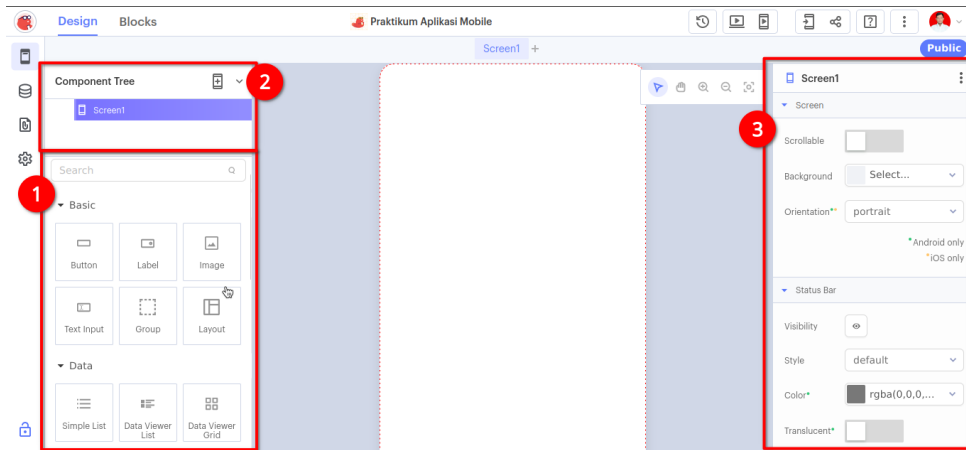
Gambar 1.5: Tampilan Laman Utama Editor

6. Jendela saat ini **Design** digunakan untuk membuat antarmuka. Sedangkan **Block** digunakan untuk mengatur perilaku antarmuka



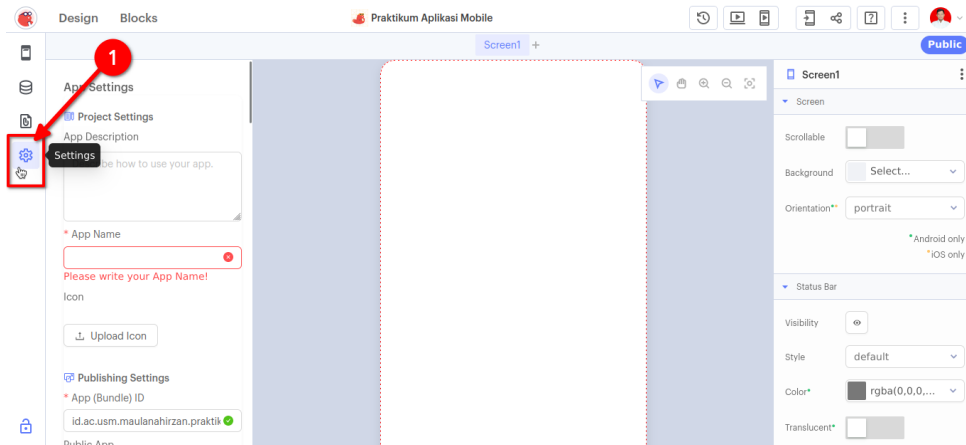
Gambar 1.6: Tampilan Laman Utama Blok

7. Kembali ke mode **Design**, dan dibagian inilah mahasiswa dapat menggunakan **Add Components** yang ada di sebelah kiri untuk **menambah komponen**, **Component Tree** untuk **Fokus dan Hapus Komponen** dan **Kustomisasi Objek** di sebelah kanan.



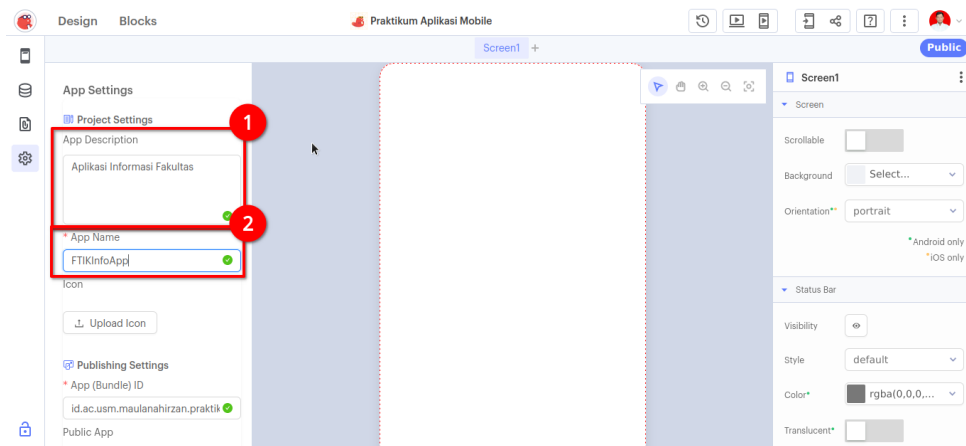
Gambar 1.7: Tampilan Laman Mode **Design** dan Fitur Lainnya

8. Sebelum memasukkan komponen UI ke Kanvas, klik **Settings** yang ada di panel sebelah kiri. Klik **Icon Roda Gigi**



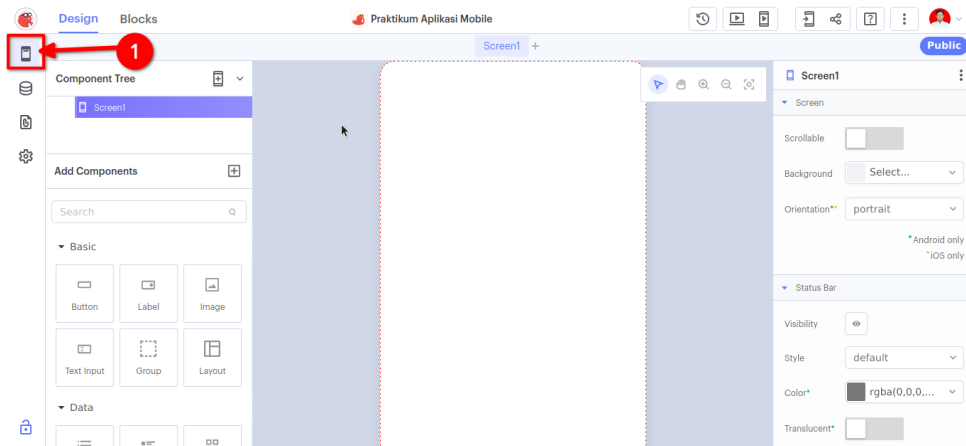
Gambar 1.8: Konfigurasi Aplikasi

9. Di bagian **App Description** dan **App Name**, isi sesuai dengan contoh:



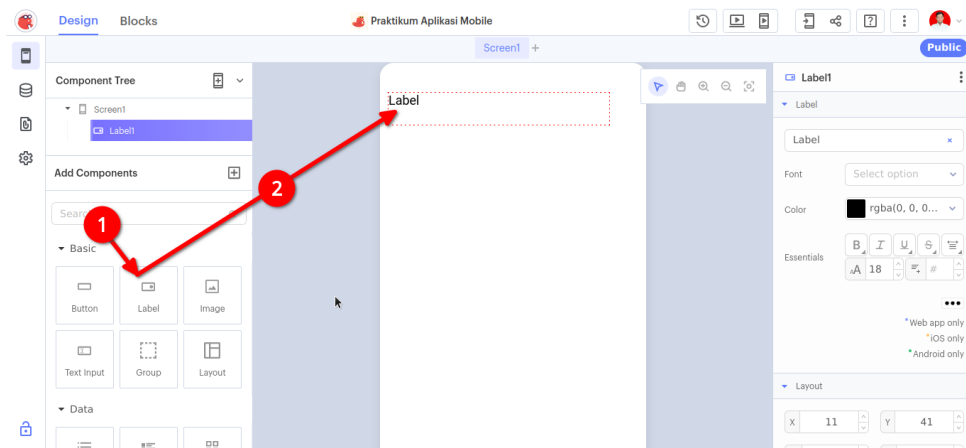
Gambar 1.9: Konfigurasi Aplikasi

10. Kembali ke mode **Designer** dengan klik **Icon HP** di atas **Icon Database** di panel sebelah kiri



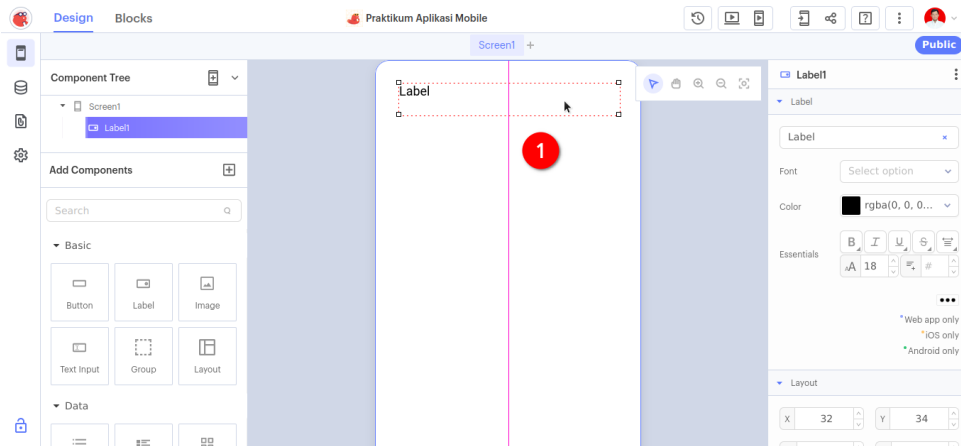
Gambar 1.10: Kembali ke Mode Designer

11. Lalu buatlah antarmuka sederhana. **Tarik** komponen **Label** ke **Kanvas**



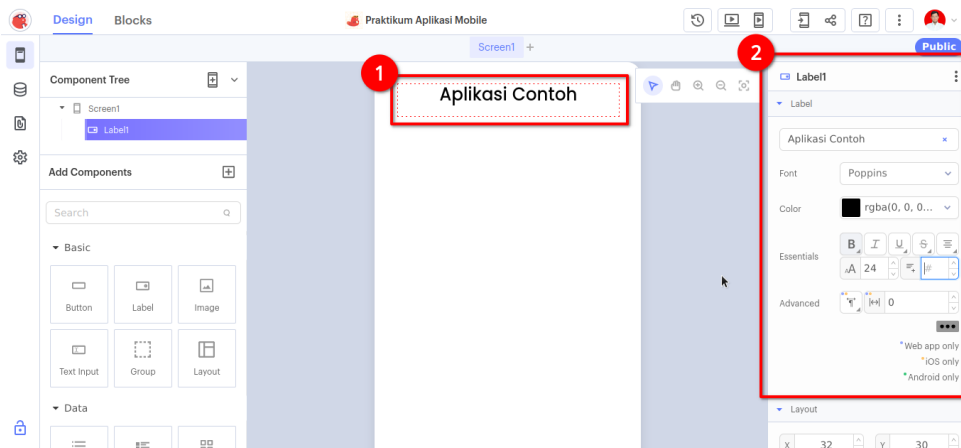
Gambar 1.11: Memasukkan Label

12. Atur ulang tatak letak **Label** agar tepat ada di tengah. Akan muncul garis warna ungu seperti di Gambar:



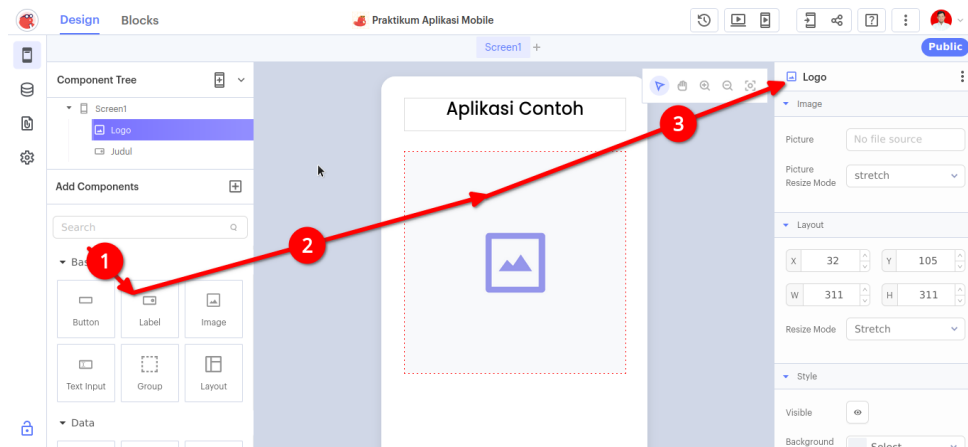
Gambar 1.12: Atur Ulang Komponen

- Isi dengan judul aplikasi dengan menggunakan **Panel Kanan**. Atur juga elemen lainnya agar terlihat menarik



Gambar 1.13: Kustomisasi Label

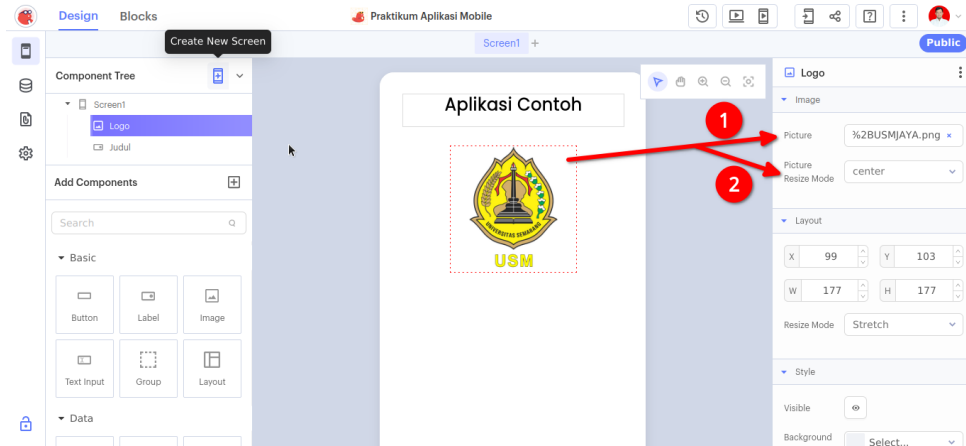
- Berikutnya tambahkan komponen **Image** ke **Kanvas**. Pastikan **ID Komponen** diganti dengan **Logo** dan berada di tengah aplikasi. Lihat Gambar:



Gambar 1.14: Menambahkan komponen **Image**

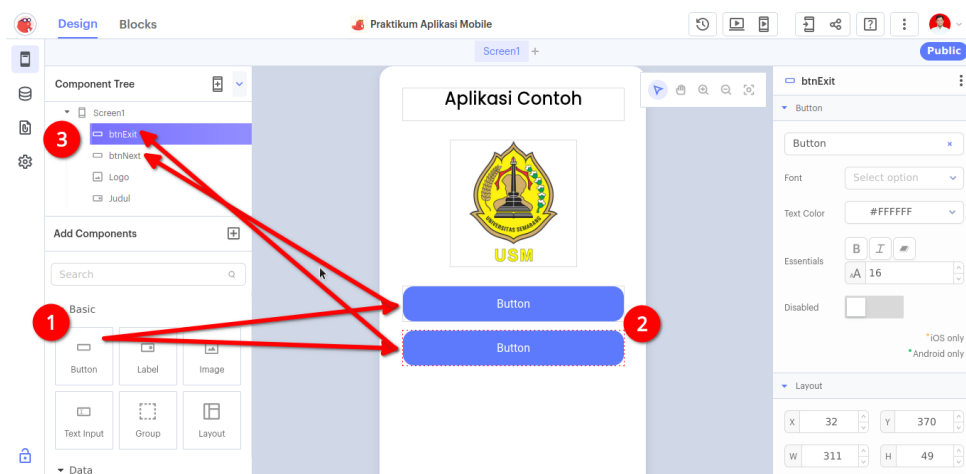
15. Untuk mengganti gambar dengan logo cukup memberikan link url berikut ke **Panel Kanan**, dan atur agar gambar terlihat bagus.

- URL : https://1.bp.blogspot.com/-iPhCdFL8S_c/XohGexgk4jI/AAAAAAAAAGaA/F7Csvkx0FRs0naEn_f19ZNNQ_vaV_TZDgCLcBGAsYHQ/s1600/LOGO%2BUSMJAYA.png



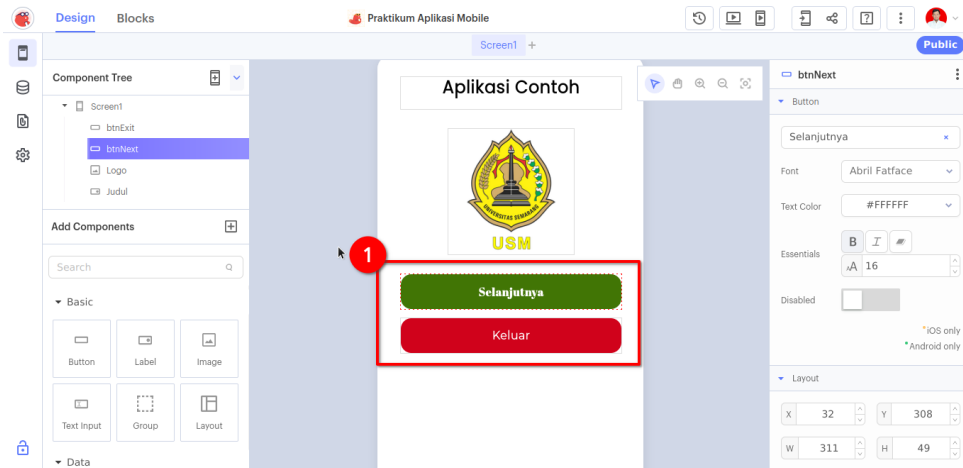
Gambar 1.15: Mengganti Gambar Komponen **Image**

16. Berikutnya adalah memberikan komponen lagi berupa dua (2) Tombol. Beri ID sebagai **btnNext** dan **btnExit** seperti gambar berikut:



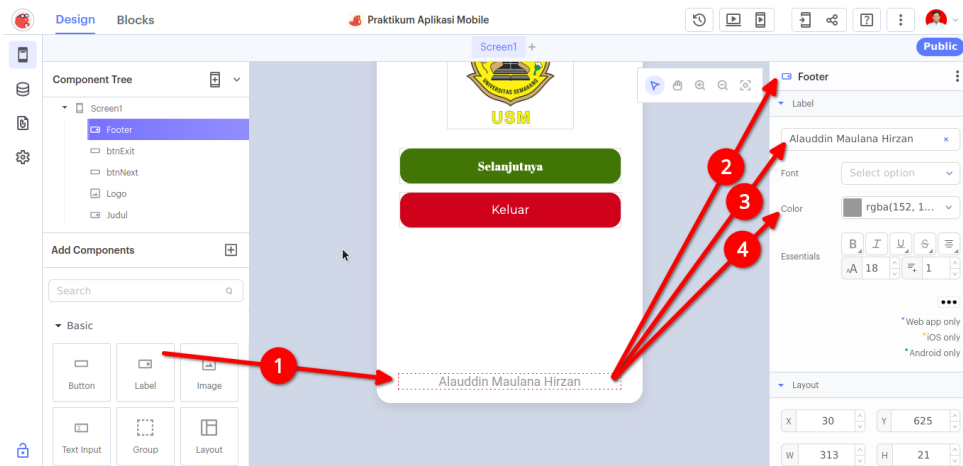
Gambar 1.16: Menambahkan Komponen **Button**

17. Berikan teks kepada **Button** dan hias secara **bebas**. Perhatikan contoh Gambar:



Gambar 1.17: Mengubah Komponen **Button**

18. Untuk penghias akhir, berikan **footer** di bawahnya dengan **Label** dengan font berukuran kecil dan warna abu-abu. Isi teks lihat contoh:



Gambar 1.18: **Footer** Aplikasi dengan **Label**

Bab 2

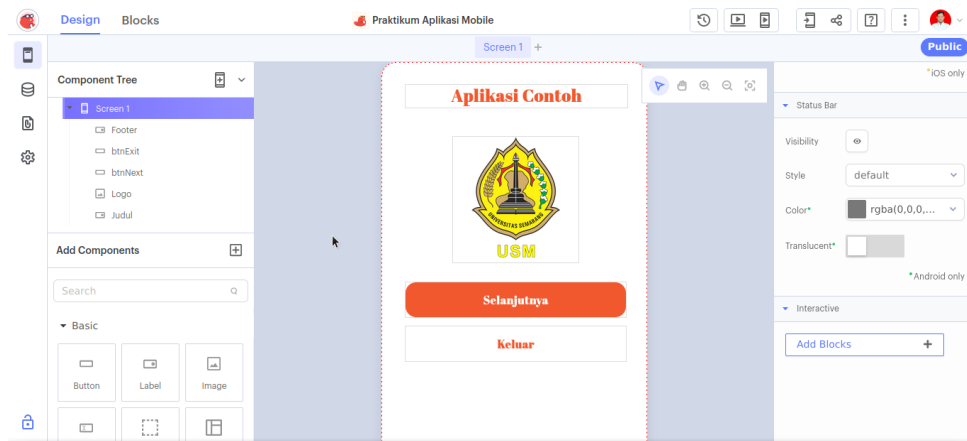
Praktikum 2

2.1 Navigasi Halaman

Di bagian ini mahasiswa diajarkan bagaimana membuat navigasi antar layar dengan menggunakan **Thunkable**. Mahasiswa diwajibkan menyelesaikan **Praktikum 1**, sebelum melanjutkan **Praktikum 2**.

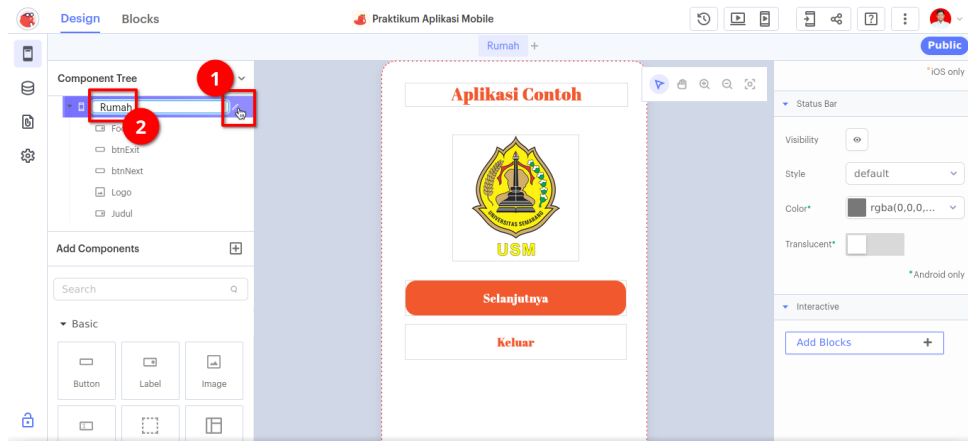
2.2 Tutorial

1. Buka kembali proyek yang sudah dibuat sebelumnya.



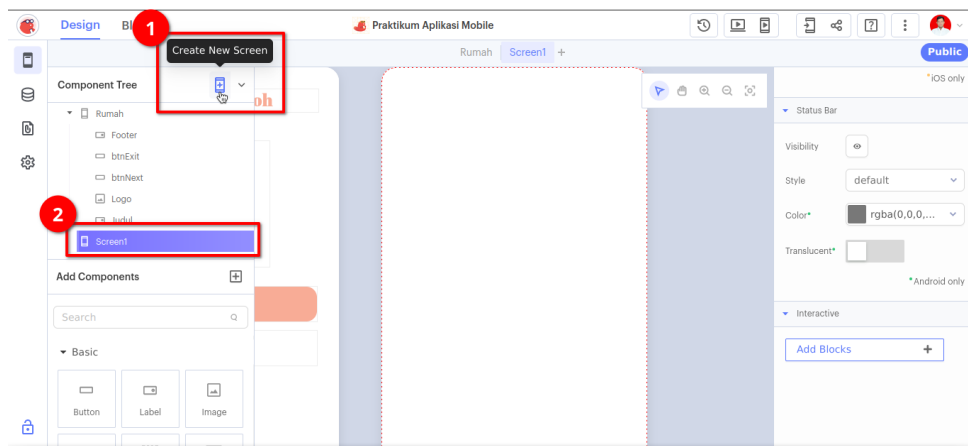
Gambar 2.1: Membuka Proyek Sebelumnya

2. Ubah **Komponen Screen 1** dengan nama baru **Home** atau **Rumah** dengan menekan tombol kecil di samping komponen. Lalu tekan **Enter**



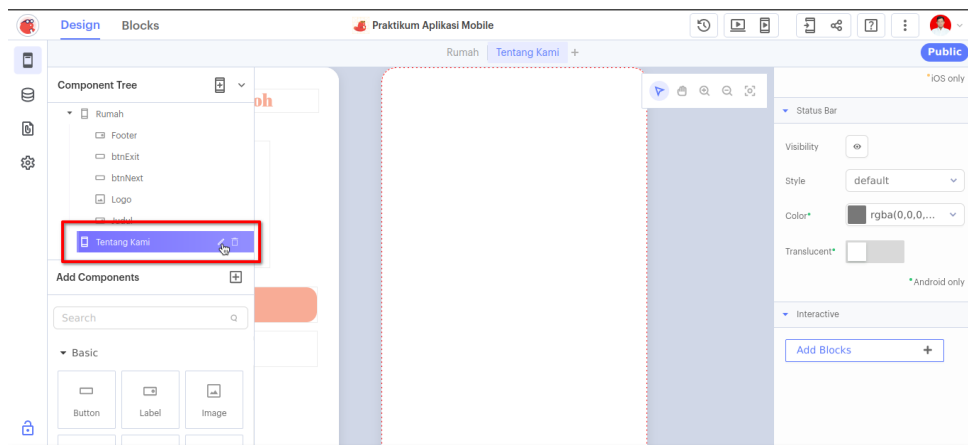
Gambar 2.2: Mengubah Layar Awal

3. Tambahkan **Layar Baru** dengan melakukan klik **Icon** yang ada di sebelah kanan tulisan **Component Tree**.



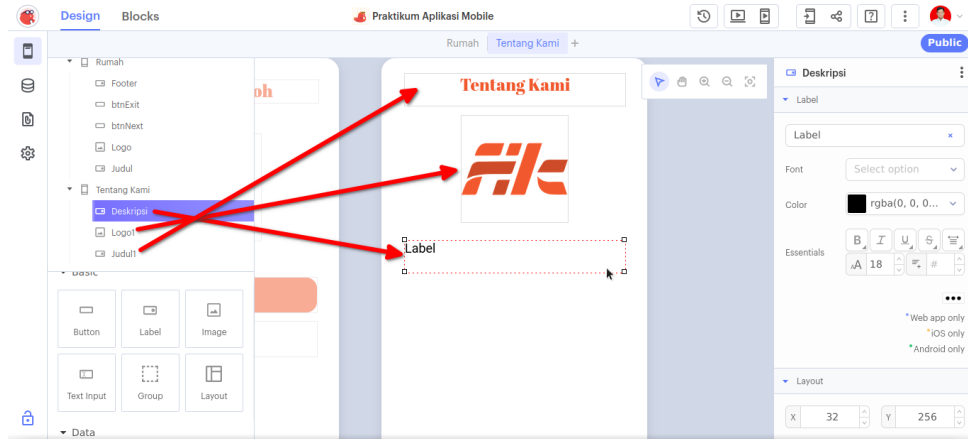
Gambar 2.3: Menambahkan Layar Baru

4. Ubah **Layar Baru** tersebut menjadi **Tentang Kami**



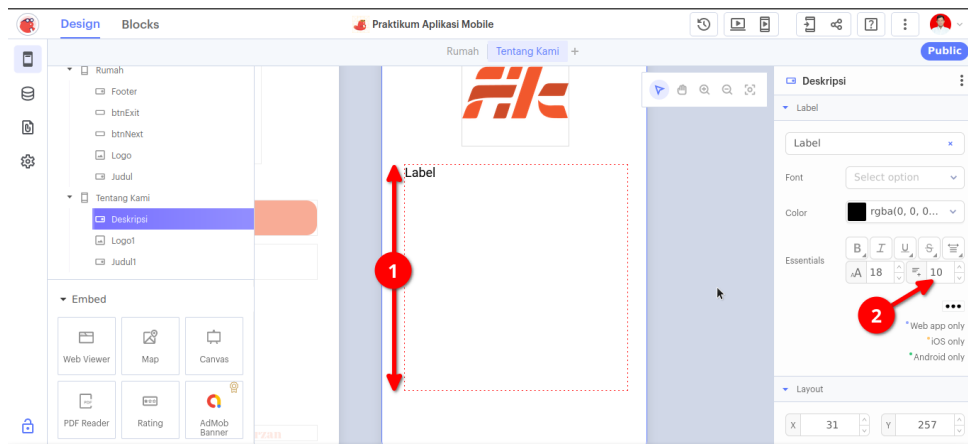
Gambar 2.4: Mengubah Nama Layar

5. Masukkan komponen-komponen seperti 1 **Label** untuk Judul, 1 **Image** untuk Logo FTIK, dan 1 **Label** untuk **Deskripsi**. Lihat Contoh:
Gunakan Link : <https://ftik.usm.ac.id/wp-content/uploads/2022/12/Favicon-FTIK-USM.png>



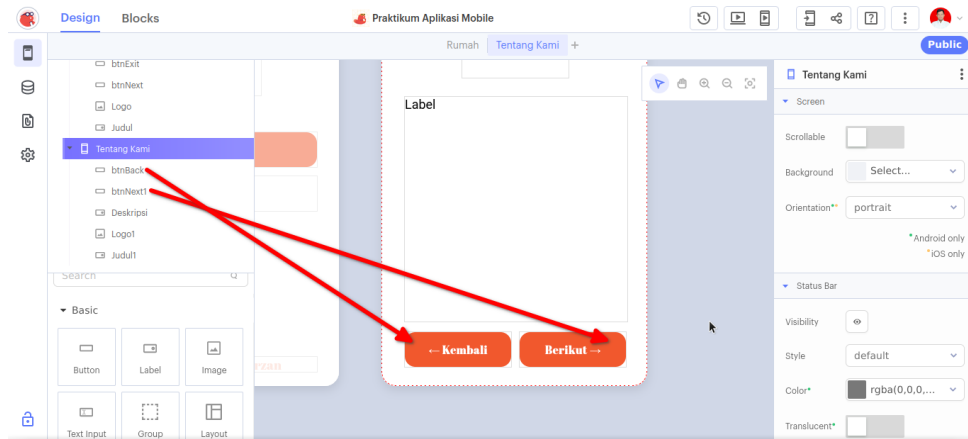
Gambar 2.5: Tampilan Layar Kedua

6. Agar komponen **Label** terakhir dapat digunakan untuk **Teks Banyak**, pertinggi **Komponen Label**, dan berikan jumlah baris di sisi kanan. Lihat Gambar:



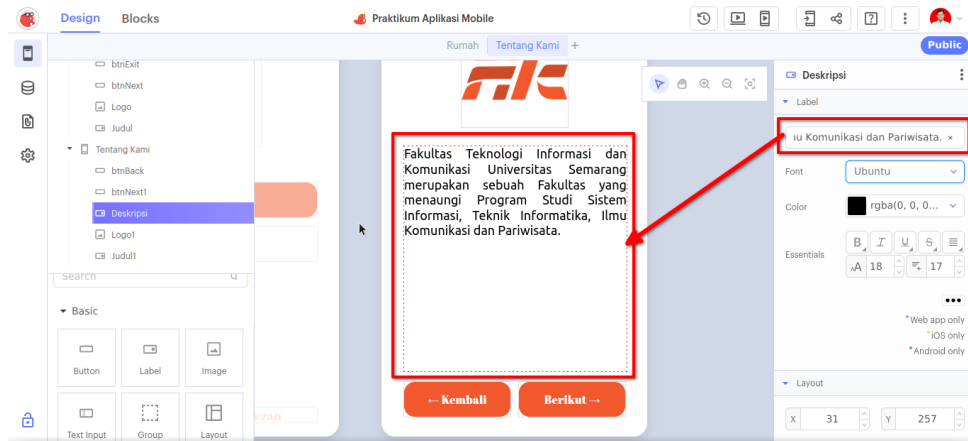
Gambar 2.6: Memodifikasi Label

7. Pastikan di bawah **Label** ada sisa untuk meletakkan dua tombol. Masukkan 2 **Button** tepat di bawah **Label Lebar** tersebut. Perhatikan Gambar dan ID masing-masing Tombol:



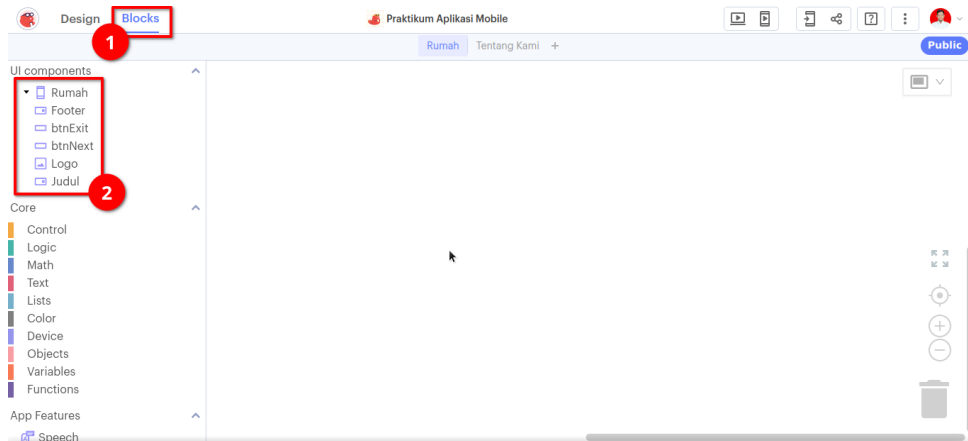
Gambar 2.7: Meletakkan 2 Tombol

8. Kemudian isi label dengan **Deskripsi Fakultas** dari FTIK. Label tidak menerima Enter dalam tulisan. Gunakan **Notepad** untuk merapikan tulisan. Lihat Gambar:



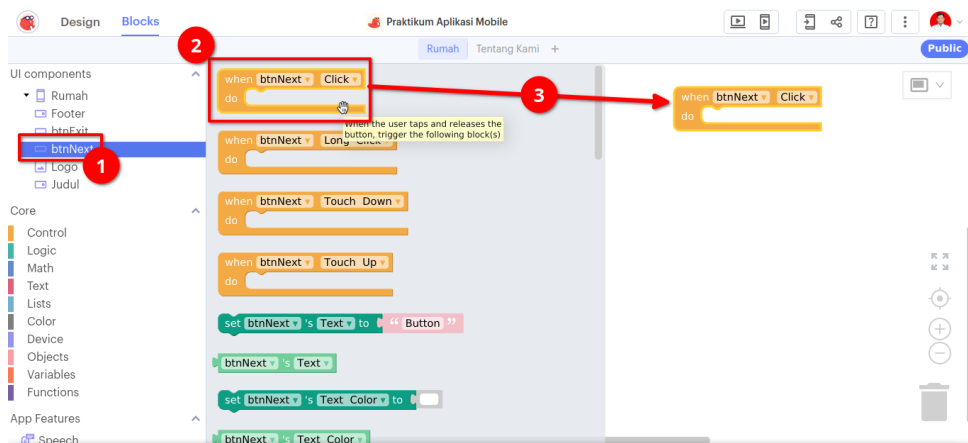
Gambar 2.8: Mengisi Deskripsi Fakultas

9. Tampilan sudah selesai. Maka langkah berikutnya adalah memberikan alur navigasi. Pastikan **Layar Pertama** sudah di klik atau dipilih terlebih dahulu. Di tahap ini mahasiswa masuk ke mode **Blocks**



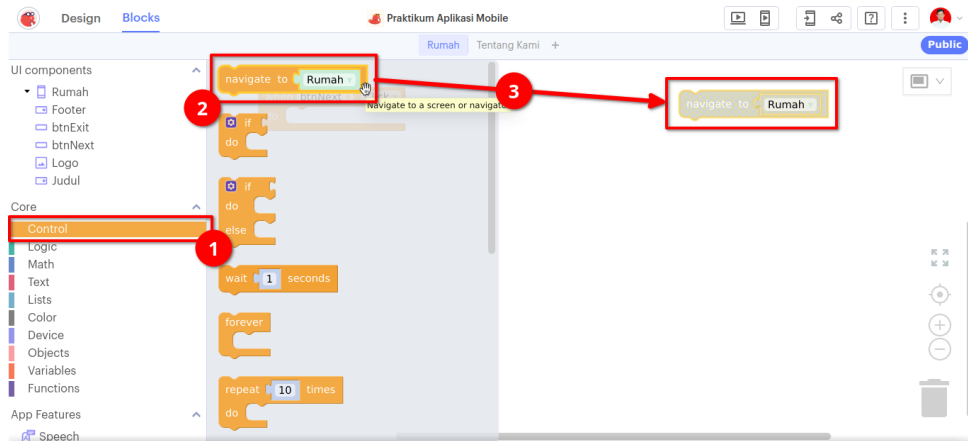
Gambar 2.9: Membuka Mode Blocks

10. Untuk memberikan aksi ke **btnNext** milik **Home** atau **Rumah**. Klik-dan-Tarik **btnNext**, pilih **Blok Kuning** dengan tulisan **When btnNext Click do** ke **Kanvas**



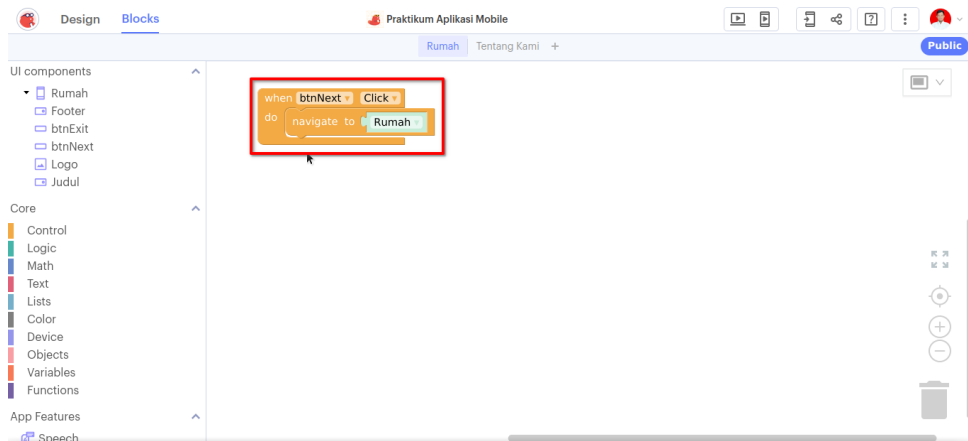
Gambar 2.10: Menarik Blok Klik btnNext

11. Untuk bisa berpindah ke **Laman Tentang Kami**, gunakan **Kategori Control** dengan **Blok navigate to**. Lihat Gambar:



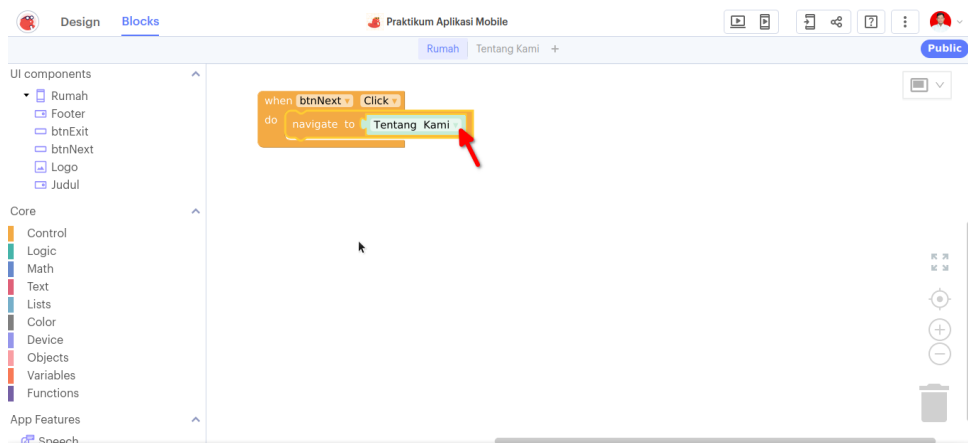
Gambar 2.11: Menarik Blok Navigasi

12. Langkah terakhir adalah mencocokkan kedua potongan tersebut menjadi satu bagian. Lihat Gambar:



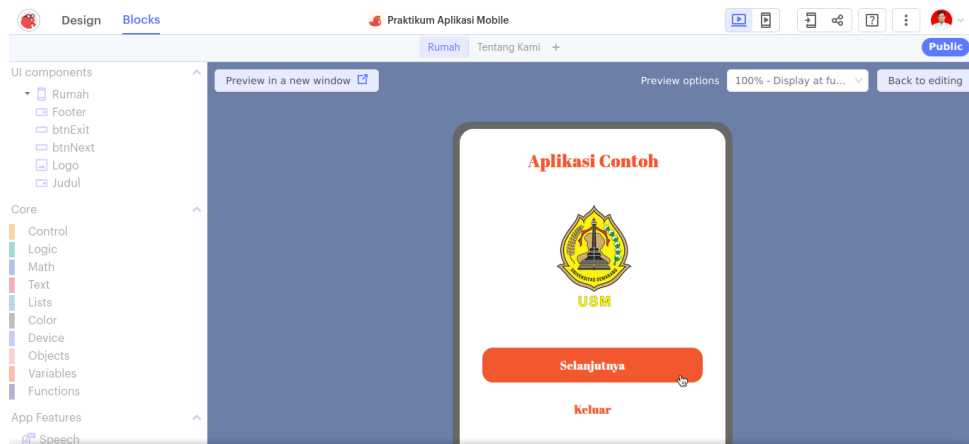
Gambar 2.12: Menyatukan Potongan Blok

13. Ubah arah navigasi dari **Rumah** ke **Tentang Kami**



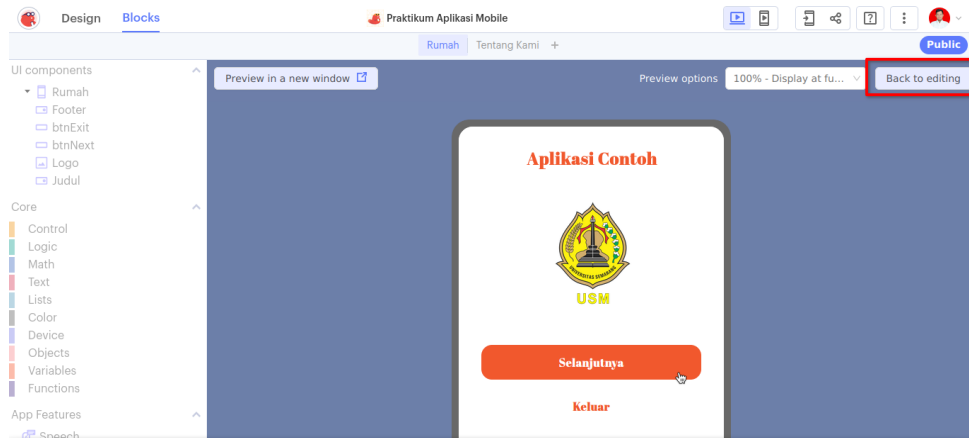
Gambar 2.13: Mengubah Arah Navigasi

14. Tes aplikasi dengan menekan **Computer Play** di bagian atas. Bukan **Phone Play**



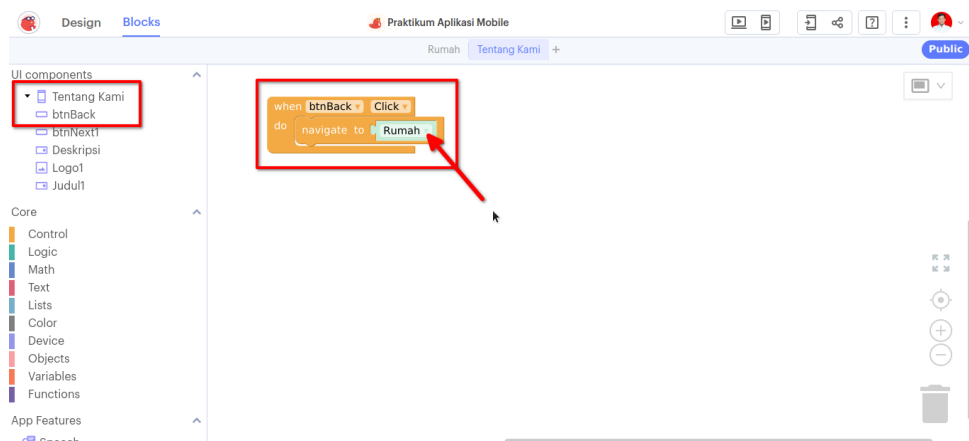
Gambar 2.14: Menguji Aplikasi

15. Jika sudah klik **Back to Editing** untuk kembali ke mode semula.



Gambar 2.15: Kembali ke Editing

16. Lakukan hal yang sama dengan tombol **btnBack** yang ada di **Tentang Kami**. Pastikan **Tentang Kami** sudah dipilih sebelum masuk mode **Blocks**



Gambar 2.16: Navigasi Mundur

Bab 3

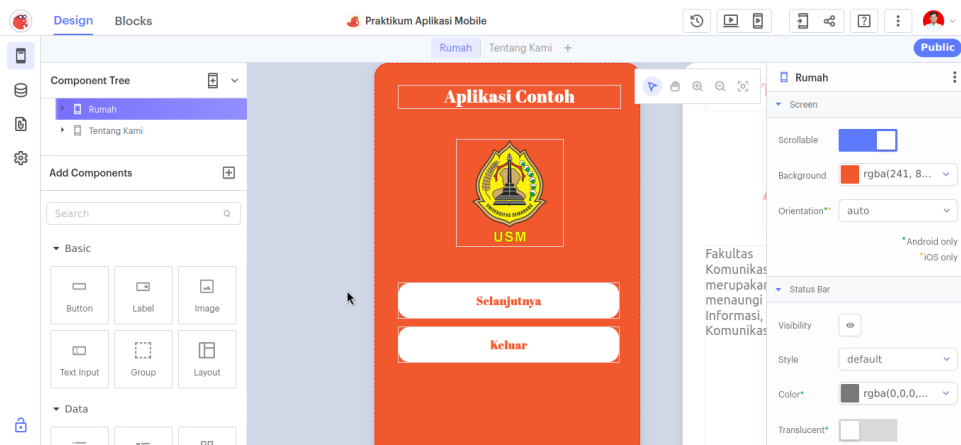
Praktikum 3

3.1 Konten Tambahan

Di bagian ini mahasiswa diajarkan menambahkan konten tambahan beserta navigasinya menggunakan **Thunkable**. Mahasiswa diwajibkan menyelesaikan **Praktikum 2**, sebelum melanjutkan **Praktikum 3**.

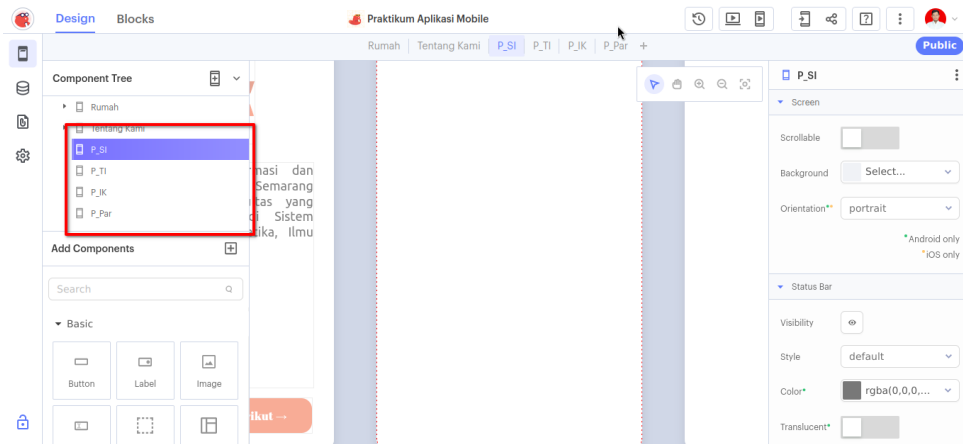
3.2 Tutorial

1. Buka kembali **Thunkable**, lalu buka projek sebelumnya.



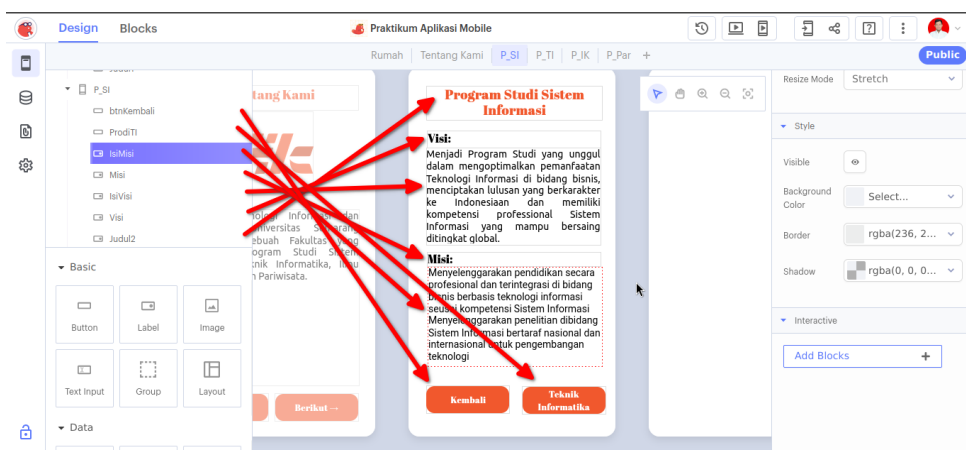
Gambar 3.1: Buka Projek

2. Buat **Empat** halaman berikutnya dengan nama **P_SI**, **P_TI**, **P_IK**, dan **P_Par**



Gambar 3.2: Buat Halaman Baru

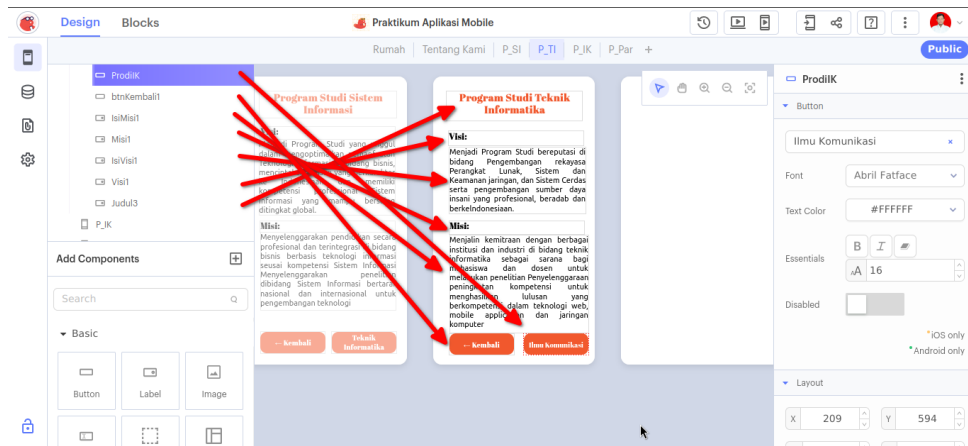
3. Dimulai dari halaman **P_SI**, masukkan komponen berikut: (Isi dapat dicek di ftik.usm.ac.id)
 - (a) Label (Judul)
 - (b) Label (Visi)
 - (c) Label (Isi Visi)
 - (d) Label (Misi)
 - (e) Label (Isi Misi)
 - (f) Button (Teknik Informatika) → ID: **ProdiTI**
 - (g) Button (Kembali) → ID: **btnKembali**



Gambar 3.3: Komponen Halaman P_SI

4. Lanjutkan dengan halaman **P_TI** dengan komponen sebagai berikut (Sama dengan sebelum):
 - (a) Label (Judul)
 - (b) Label (Visi)

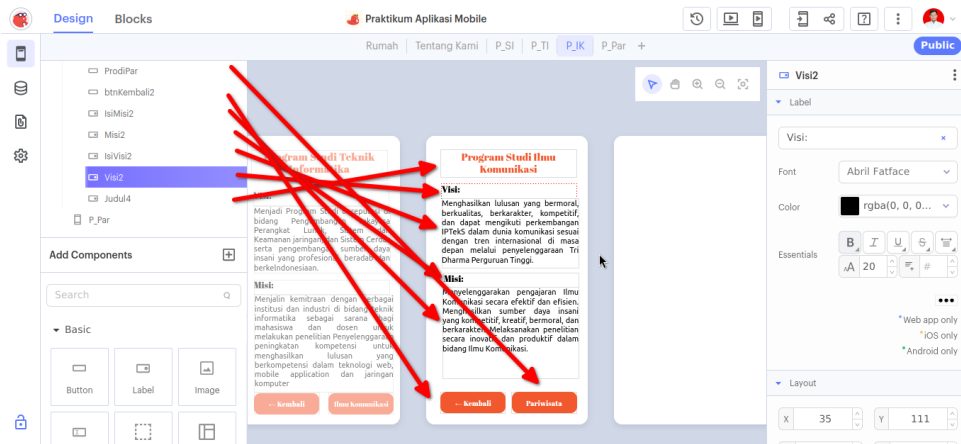
- (c) Label (Isi Visi)
- (d) Label (Misi)
- (e) Label (Isi Misi)
- (f) Button (Ilmu Komunikasi) → ID: **ProdiIK**
- (g) Button (Kembali) → ID: **btnKembali**



Gambar 3.4: Komponen Halaman P_TI

5. Lanjutkan dengan halaman **P_ IK** dengan komponen sebagai berikut (Sama dengan sebelum):

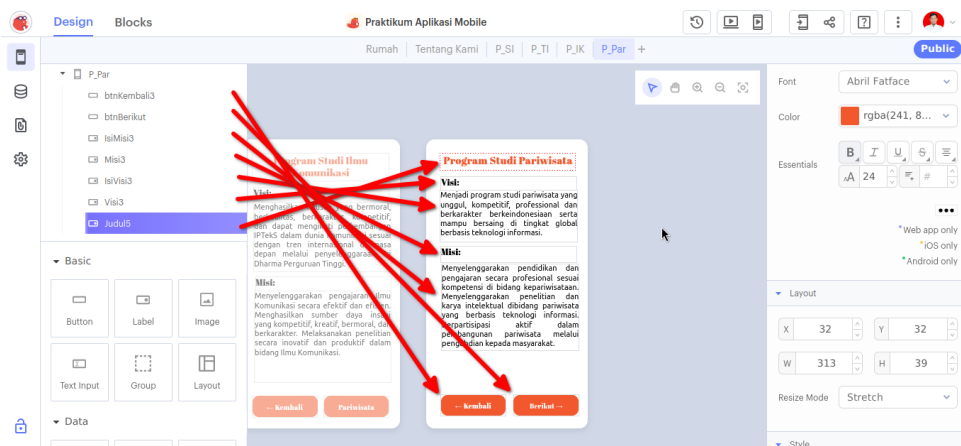
- (a) Label (Judul)
- (b) Label (Visi)
- (c) Label (Isi Visi)
- (d) Label (Misi)
- (e) Label (Isi Misi)
- (f) Button (Pariwisata) → ID: **ProdiPar**
- (g) Button (Kembali) → ID: **btnKembali**



Gambar 3.5: Komponen Halaman P_IK

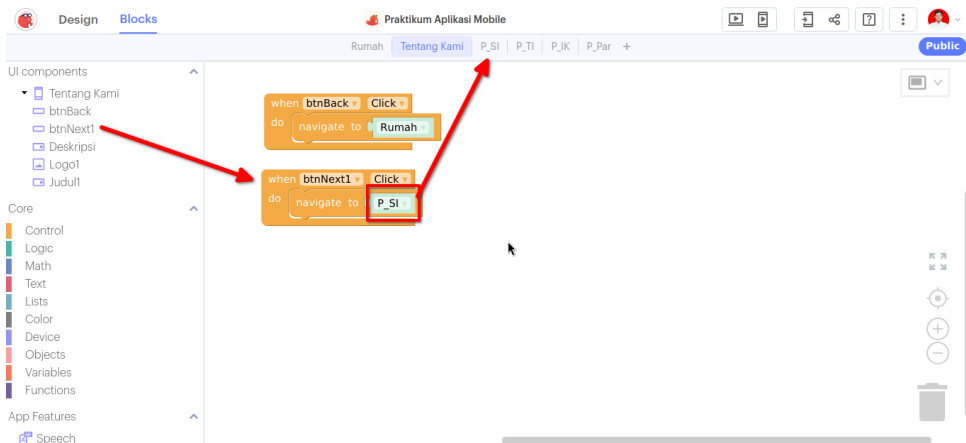
6. Terakhir, lanjutkan dengan halaman **P_Par** dengan komponen sebagai berikut (Sama dengan sebelum):

- (a) Label (Judul)
- (b) Label (Visi)
- (c) Label (Isi Visi)
- (d) Label (Misi)
- (e) Label (Isi Misi)
- (f) Button (Berikut) → ID: **btnBerikut**
- (g) Button (Kembali) → ID: **btnKembali**



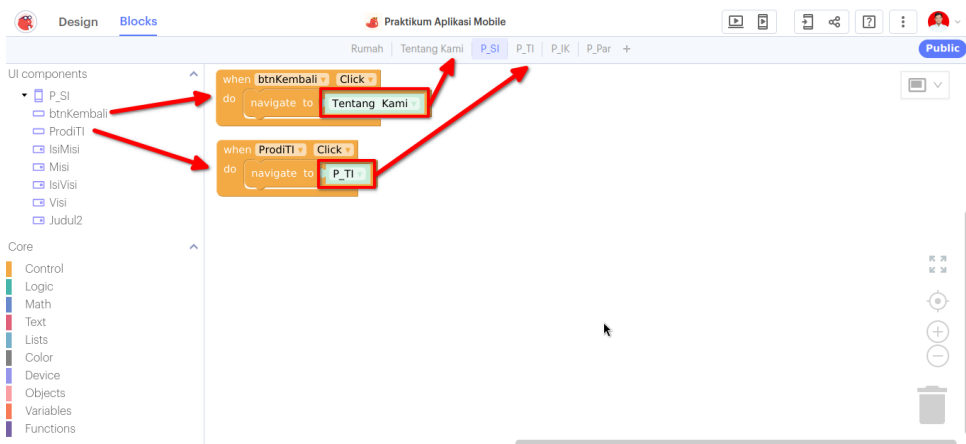
Gambar 3.6: Komponen Halaman P_Par

7. Buka halaman **Tentang Kami**, dan masuk ke mode **Blocks**. Buatlah blok untuk tombol **btnNext** seperti berikut:



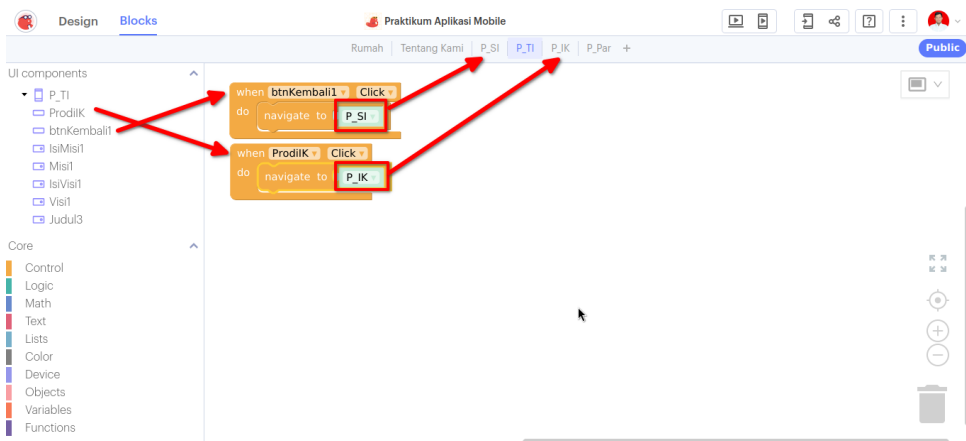
Gambar 3.7: Blok Aksi btnNext Tentang Kami

8. Buka halaman **P_SI**, dan masuk ke mode **Blocks**. Buatlah blok untuk tombol **btnKembali** dan **ProdiTI** seperti berikut:



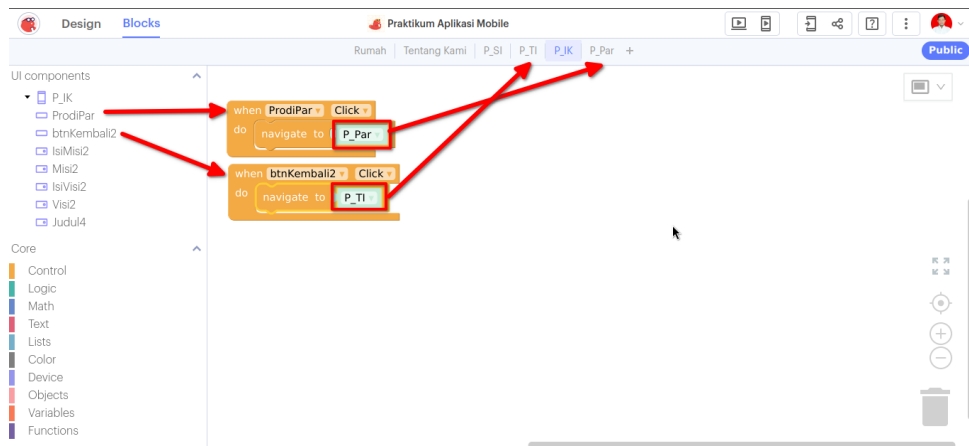
Gambar 3.8: Blok Aksi btnKembali dan ProdiTI

9. Buka halaman **P_TI**, dan masuk ke mode **Blocks**. Buatlah blok untuk tombol **btnKembali** dan **ProdiIK** seperti berikut:



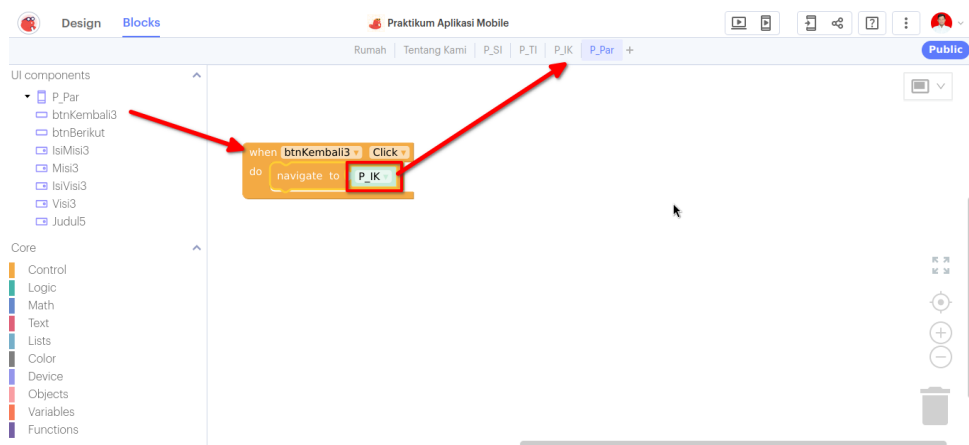
Gambar 3.9: Blok Aksi btnKembali dan ProdiIK

10. Buka halaman **P_IK**, dan masuk ke mode **Blocks**. Buatlah blok untuk tombol **btnKembali** dan **ProdiPar** seperti berikut:



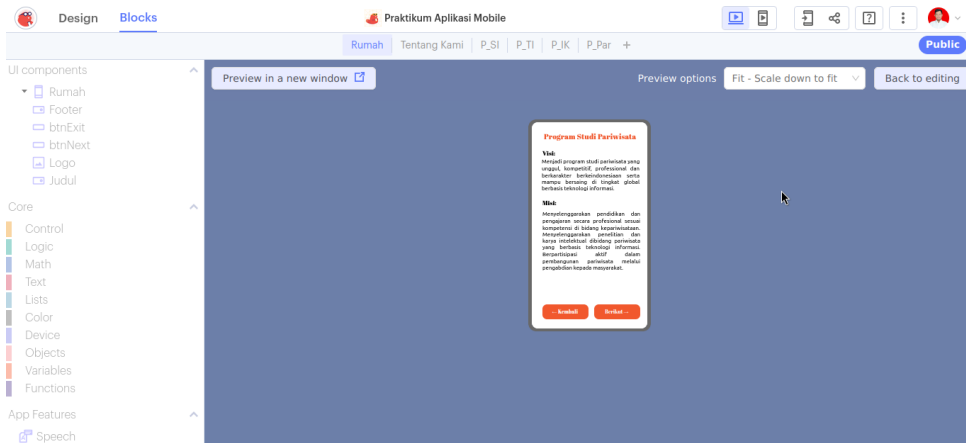
Gambar 3.10: Blok Aksi btnKembali dan ProdiPar

11. Buka halaman **P_Par**, dan masuk ke mode **Blocks**. Buatlah blok untuk tombol **btnKembali** seperti berikut:



Gambar 3.11: Blok Aksi btnKembali

12. Uji Aplikasi untuk memastikan **Flow Halaman** sesuai



Gambar 3.12: Pengujian Desain

Bab 4

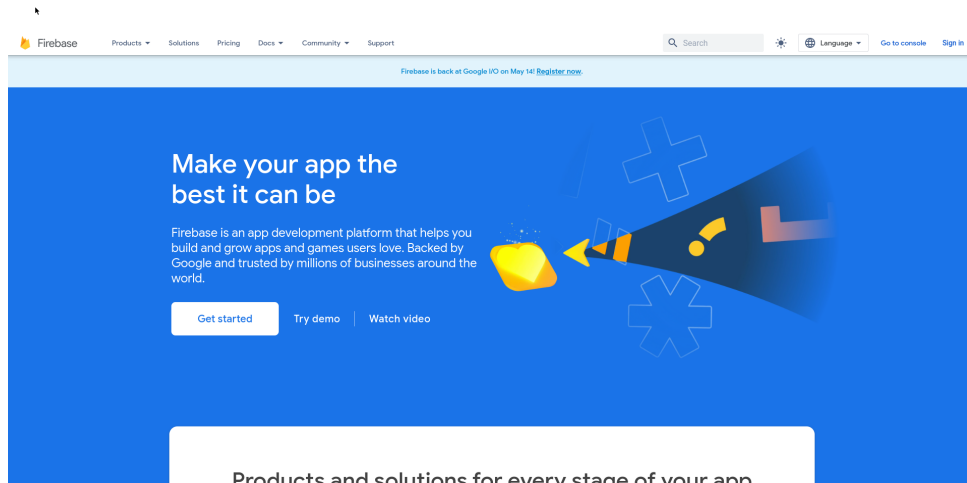
Praktikum 4

4.1 Konektivitas Database

Di bagian ini mahasiswa diajarkan Bagaimana membuat database di Google Firebase dan menghubungkannya ke projek aplikasinya. Mahasiswa diwajibkan menyelesaikan **Praktikum 3**, sebelum melanjutkan **Praktikum 4**.

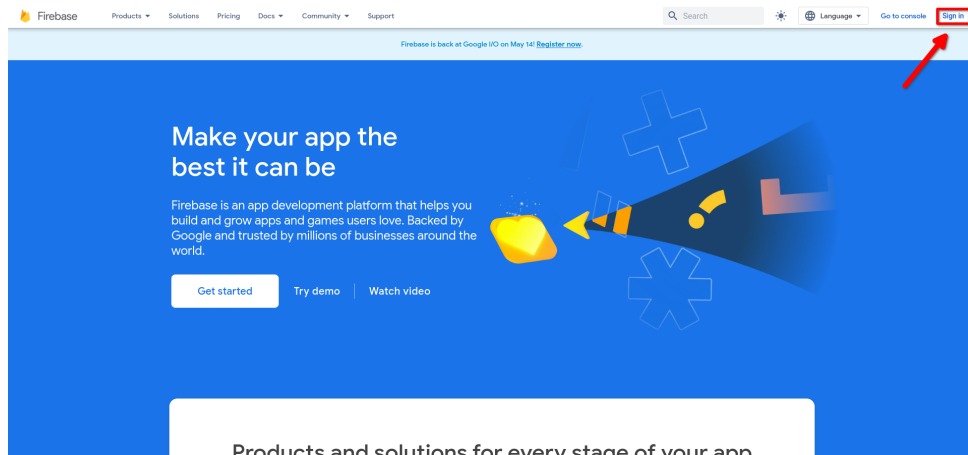
4.2 Tutorial

1. Buka website **Thunkable** dan **Google Firebase** melalui link berikut: <https://firebase.google.com>



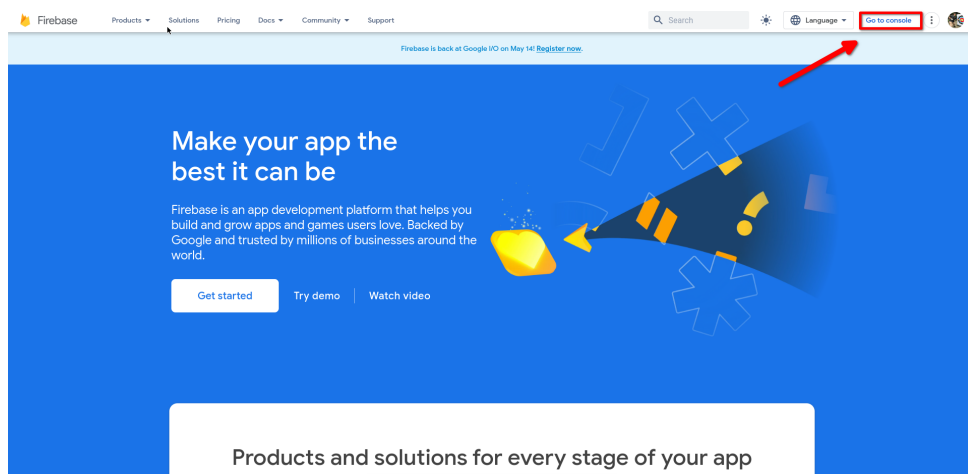
Gambar 4.1: Buka Web Firebase

2. Klik **Sign-In** untuk masuk ke dalam sistem **Firebase**



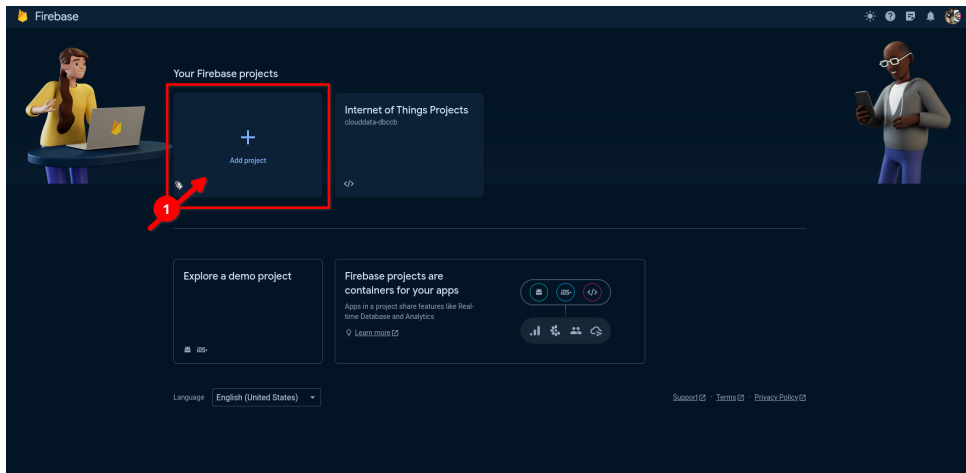
Gambar 4.2: Login ke Firebase

3. Jika kembali ke halaman depan atau sudah pernah mengakses **Firebase**, cukup Klik **Go to console**.



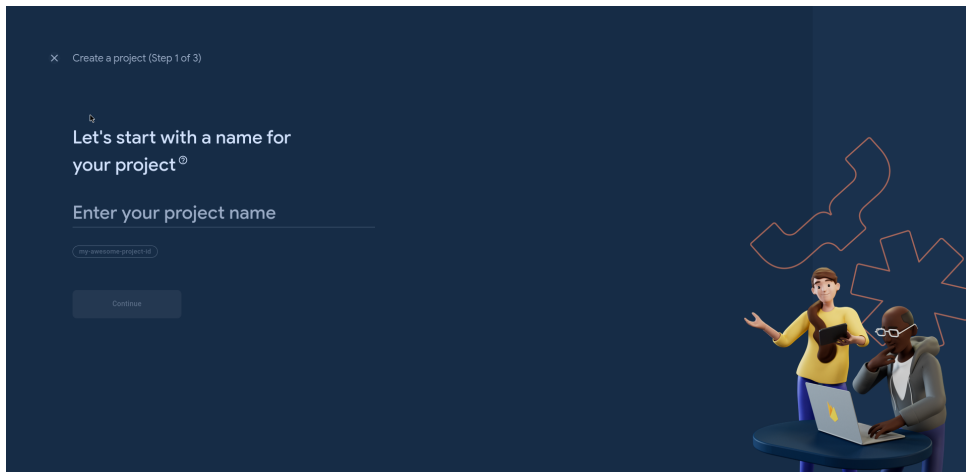
Gambar 4.3: Membuka Console

4. **Firebase** akan memperlihatkan halaman proyek kan klik tanda Plus + untuk menambahkan proyek baru



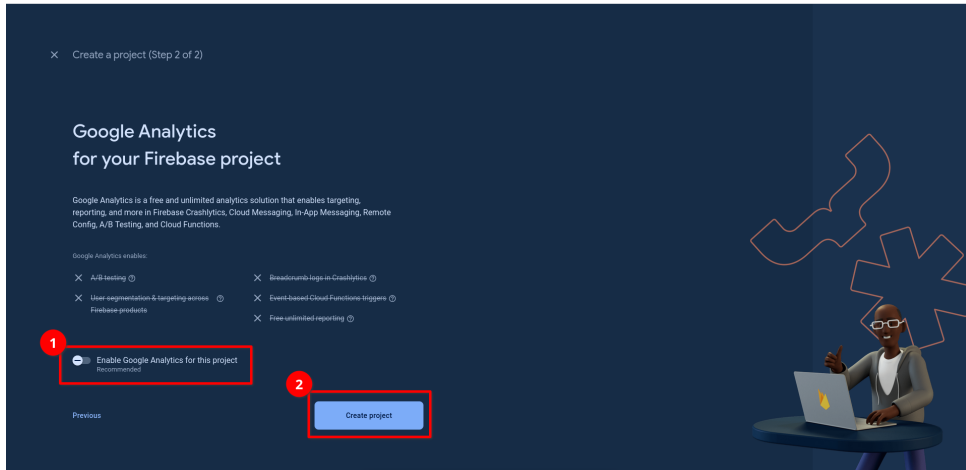
Gambar 4.4: Menambah proyek baru

5. Masukkan nama proyek dengan nama yang mudah diingat. Lalu klik **Continue**



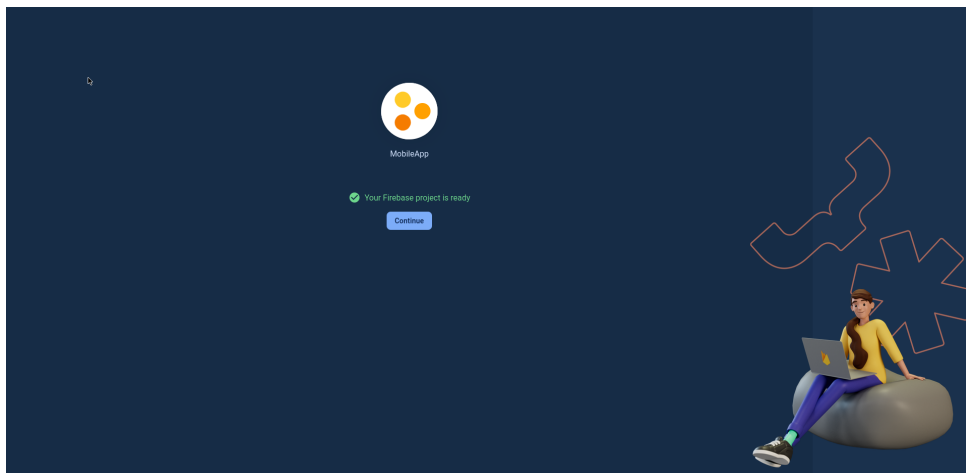
Gambar 4.5: Memberi nama proyek

6. Di halaman berikutnya akan menanyakan untuk **Google Analytic**. Cukup matikan opsi **Enable Google Analytic**. Lalu pilih **Create Project**



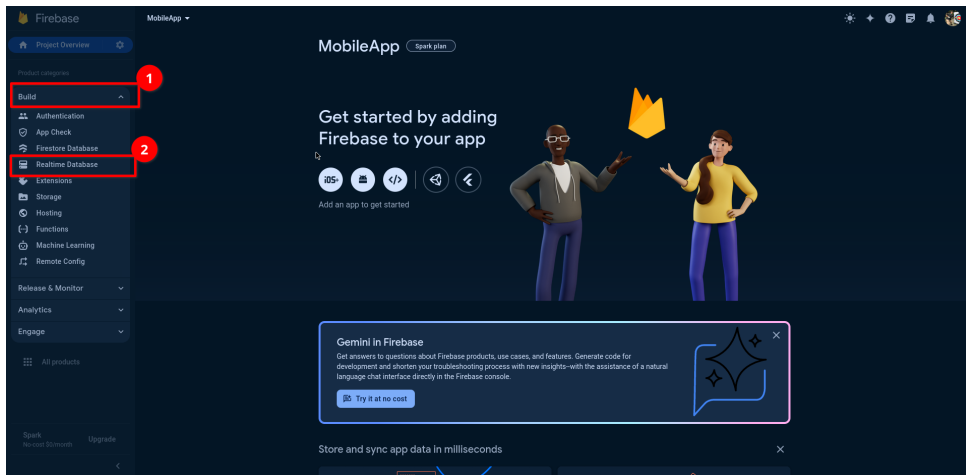
Gambar 4.6: Mematikan opsi analytic

7. Tunggu **Firestore** membuat proyek tersebut
8. Jika sudah selesai, klik **Continue**



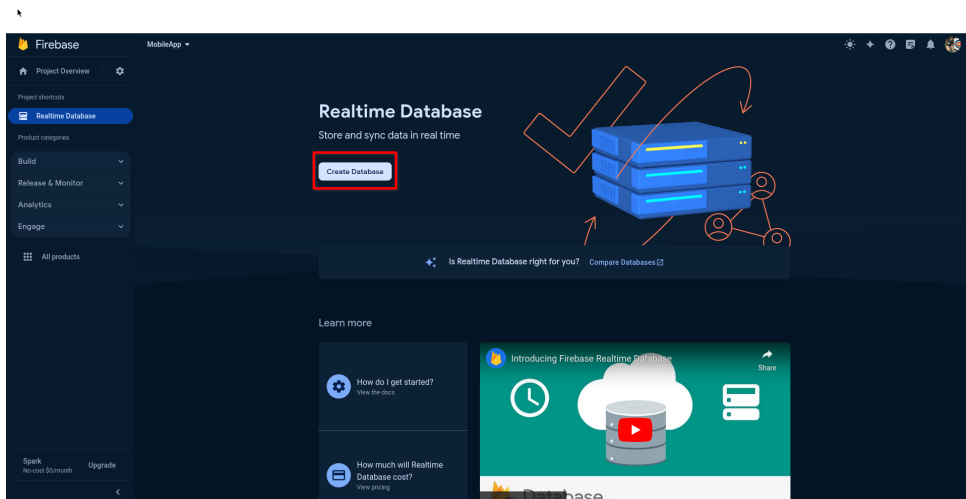
Gambar 4.7: Proyek selesai dibuat

9. Kemudian akan diarahkan ke halaman **Dashboard Firebase**. Dari sini mahasiswa akan membuat database dengan klik **Build** di samping kanan dan pilih **Realtime Database**



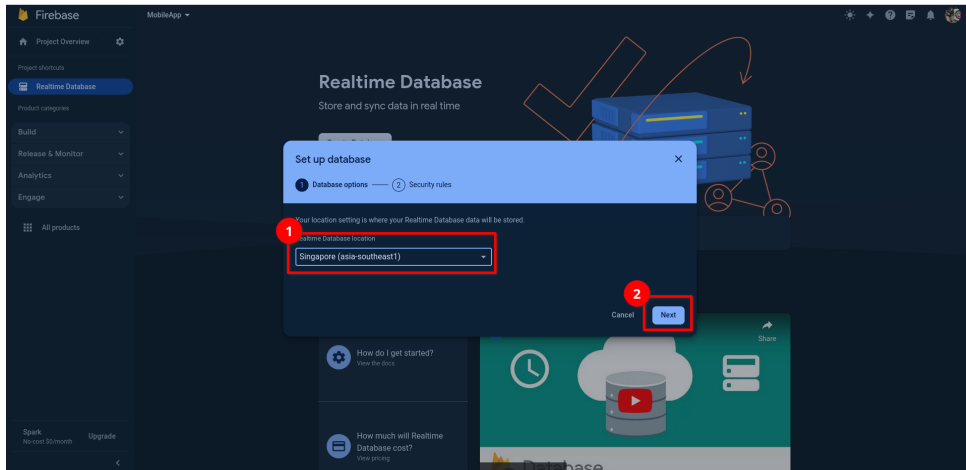
Gambar 4.8: Navigasi ke Realtime Database

10. Untuk membuat database, cukup klik **Create Database**



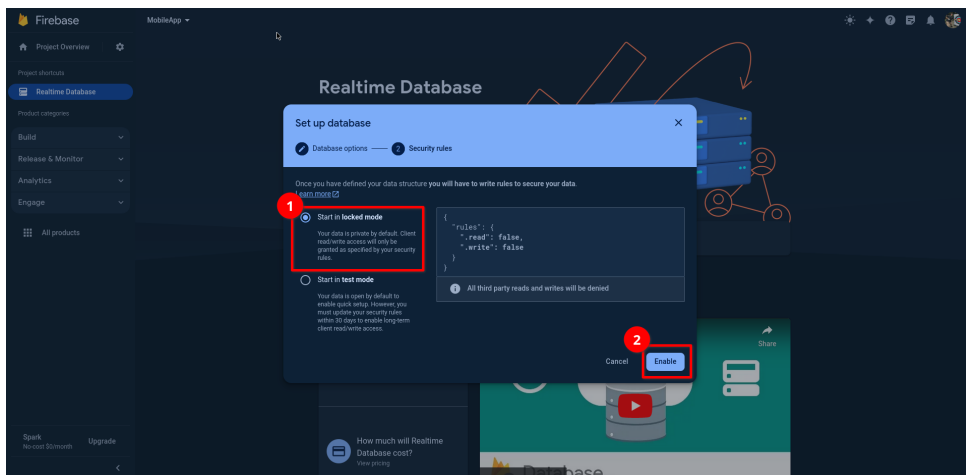
Gambar 4.9: Membuat Realtime Database

11. Berikutnya adalah memilih lokasi database akan dibuat, pilih **Singapore** dan **Next**



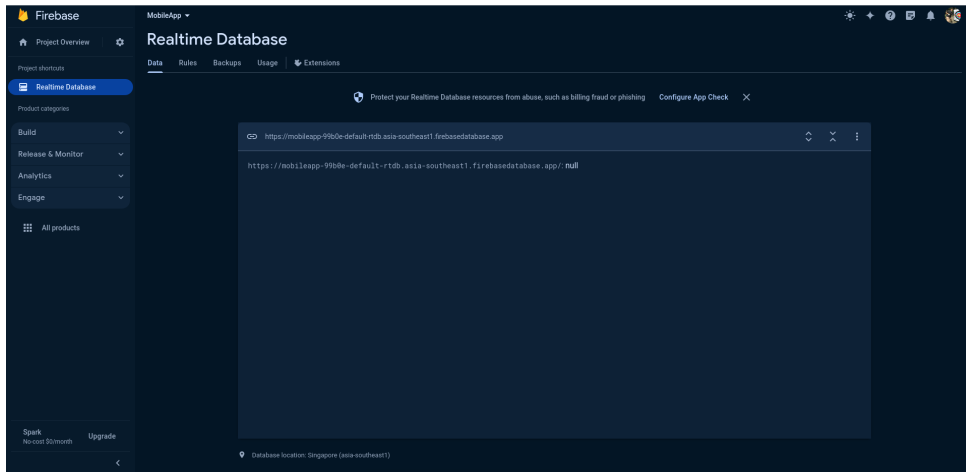
Gambar 4.10: Memilih lokasi database

12. Lalu dikonfigurasi berikutnya memilih **rules** untuk melindungi akses dari pihak luar. Pilih **Locked Mode** dan **Enable**



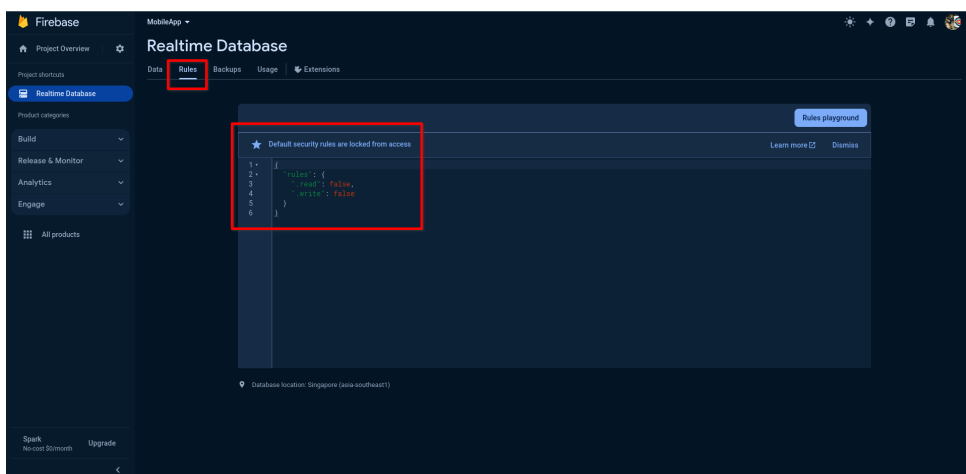
Gambar 4.11: Mengatur keamanan database

13. Tunggu **Firestore** melakukan provisi database
14. Jika sukses akan terlihat seperti berikut



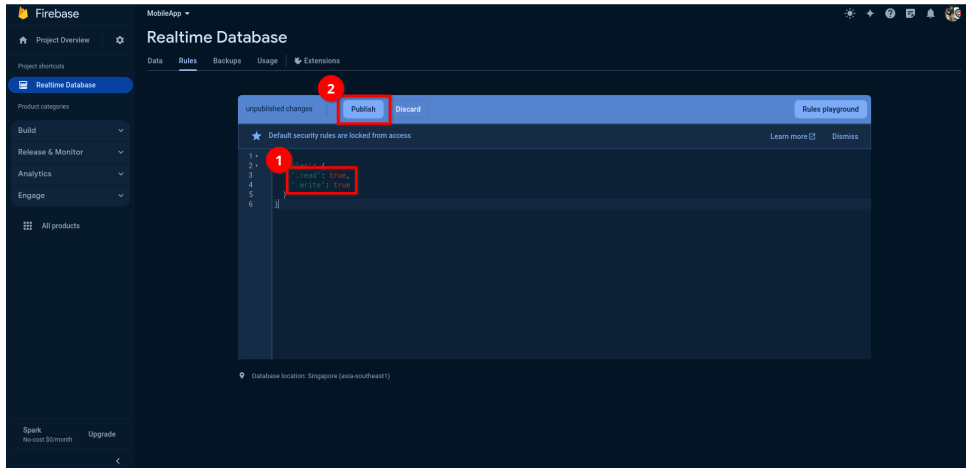
Gambar 4.12: Tampilan Realtime Database

15. Berikutnya mengatur keamanan database, klik **Rules** di bagian atas database.



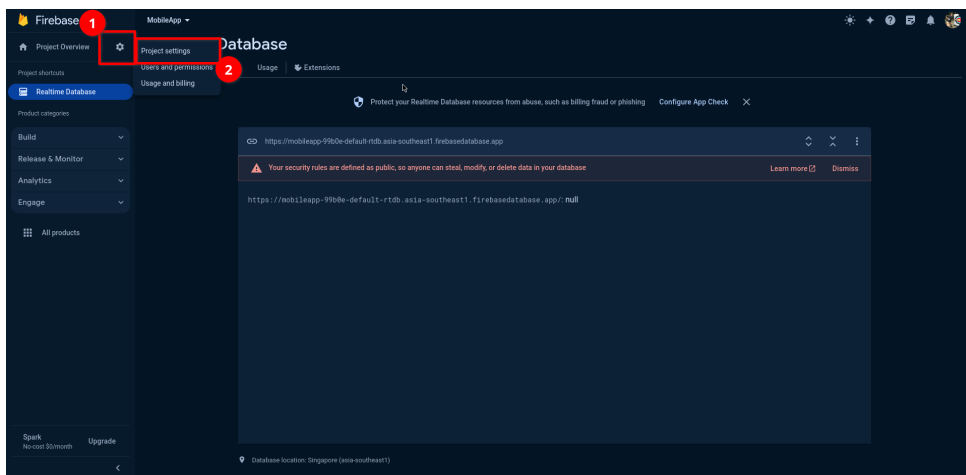
Gambar 4.13: Mengatur akses database

16. Untuk mengubah akses database cukup mudah, dengan mengubah nilai **false** menjadi **true**. Lalu klik **Publish**



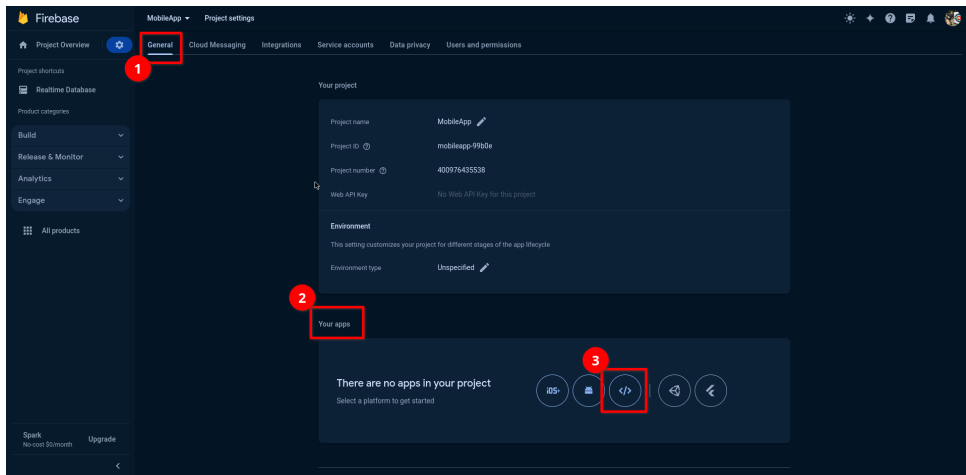
Gambar 4.14: Mengubah akses database

17. Jika sudah, maka langkah berikutnya adalah membuat akses aplikasi dengan melakukan Klik **Roda Gigi** yang bisa dilihat di sebelah kanan **Project Overview**. Kemudian pilih **Project Settings**



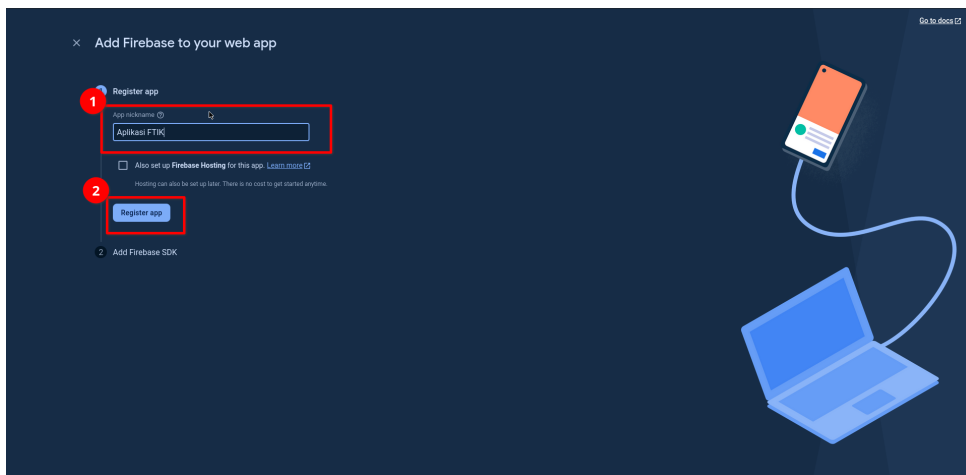
Gambar 4.15: Membuka Project Settings

18. Pastikan sudah memilih tab **General**, kemudian turun ke bawah di bagian **Your apps**. Pilih Logo Web (Sebelah Kanan Android)



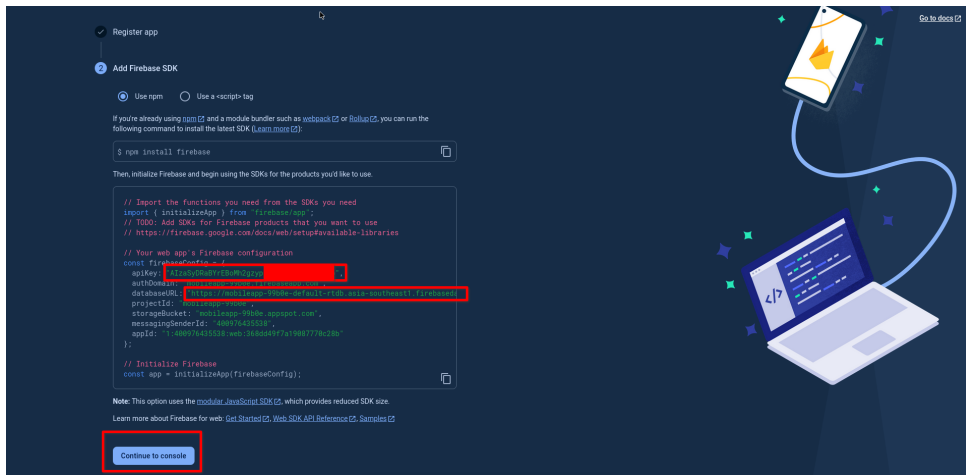
Gambar 4.16: Navigasi ke Your Apps

19. Berikan nama untuk aplikasi yang didaftarkan (bebas tapi mudah dipahami)



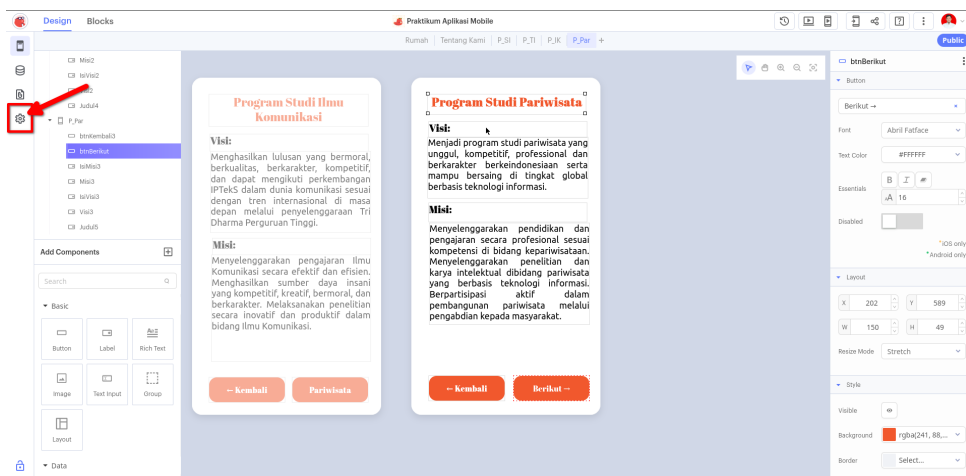
Gambar 4.17: Memberi nama aplikasi

20. Selanjutnya Kopi **APIKey** dan **databaseURL** dan simpan baik-baik di Notepad dan klik **Continue to Console** untuk kembali



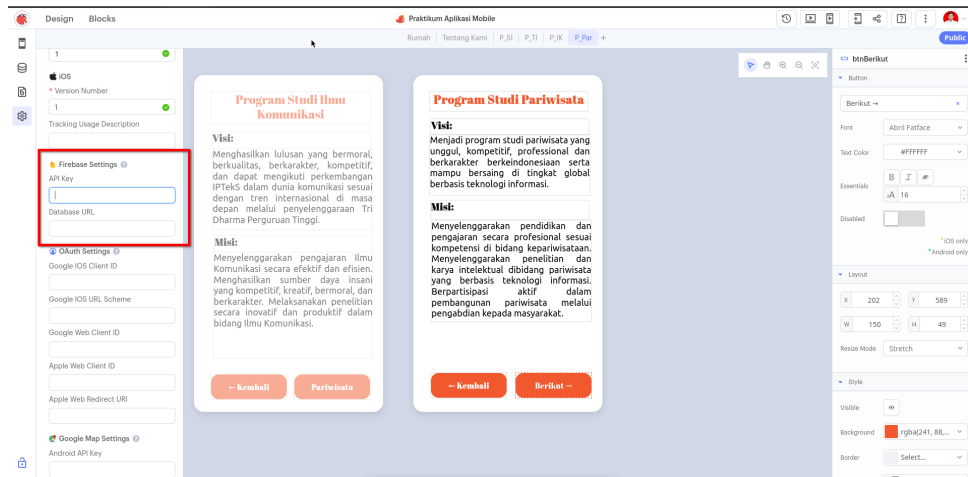
Gambar 4.18

21. **Token APIKey** sudah didapatkan, berikutnya kembali ke **Thunkable** dan buka proyek masing-masing. Setelah terbuka klik **Settings**



Gambar 4.19: Membuka konfigurasi Thunkable

22. Turun ke bawah untuk menemukan konfigurasi **Firestore** dan tempel data yang sudah dikopi tadi



Gambar 4.20: Menempel Data APIKey dan databaeURL

23. Jika sudah, simpan projek dengan klik mode **Design** dan konfigurasi Database Selesai

Bab 5

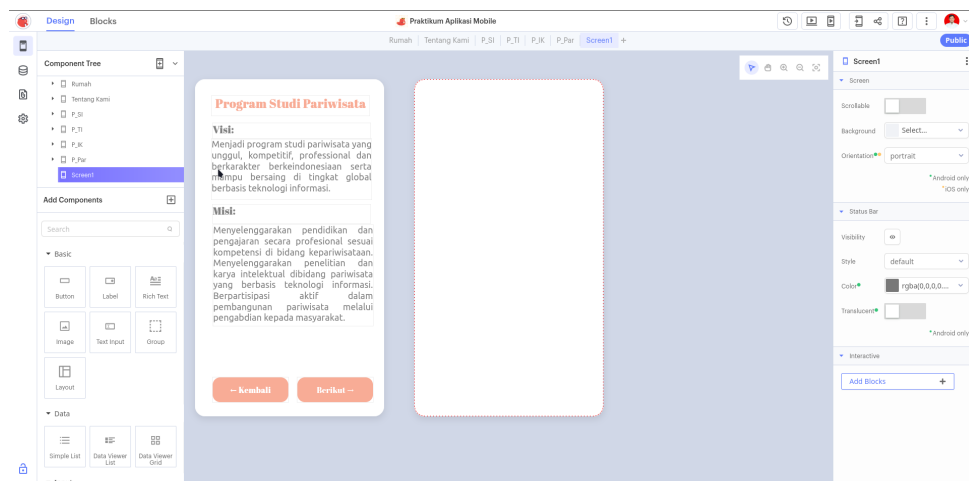
Praktikum 5

5.1 Input Database

Di bagian ini mahasiswa diajarkan Bagaimana melakukan input data ke Firebase **Real-time Database** menggunakan **Thunkable**. Mahasiswa diwajibkan menyelesaikan Praktikum 4 sebelum melanjutkan ke bagian ini

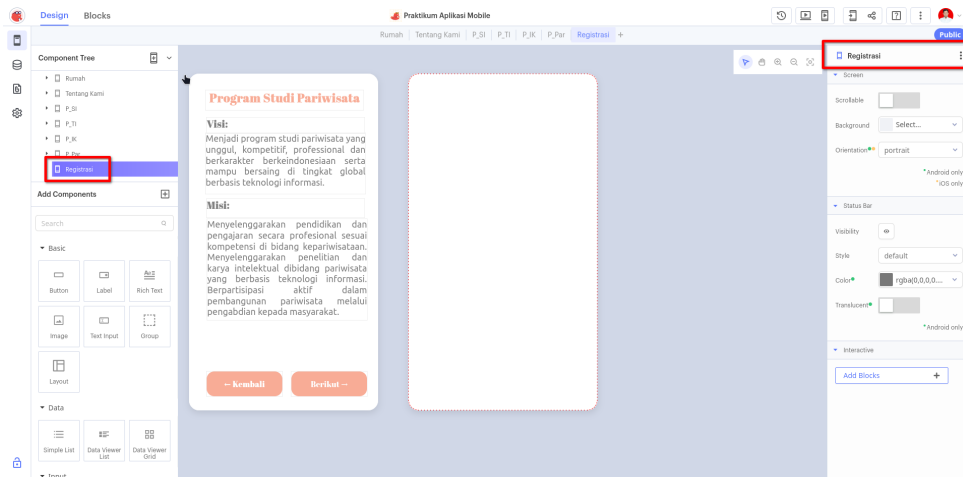
5.2 Tutorial

1. Buka **Thunkable**, dan buatlah **Satu Halaman Baru** setelah **Halaman Program Studi Pariwisata**



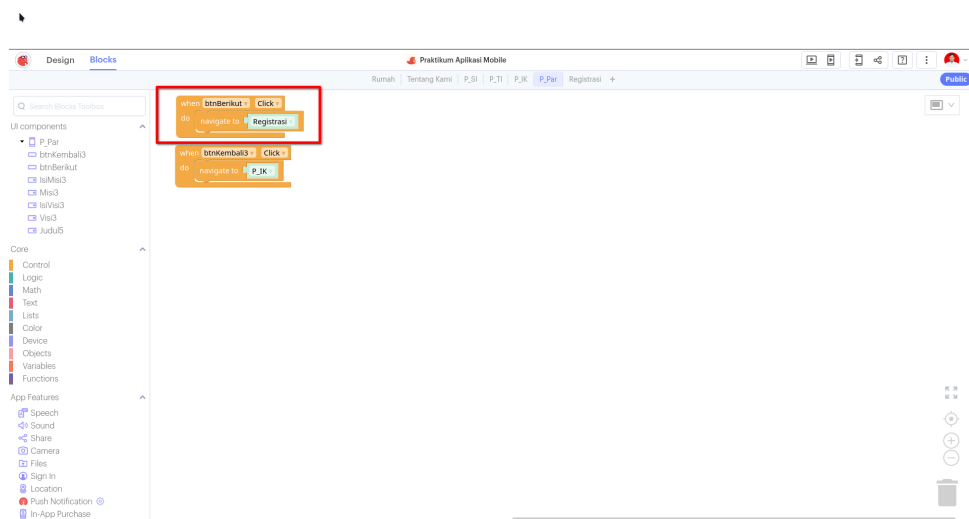
Gambar 5.1: Halaman Baru

2. Ubah **Nama / ID** halaman tersebut menjadi **Registrasi**



Gambar 5.2: Mengubah Identitas Halaman

3. Setelah itu, hubungkan **Halaman Registrasi** dengan **Halaman Program Studi Pariwisata** agar bisa diakses

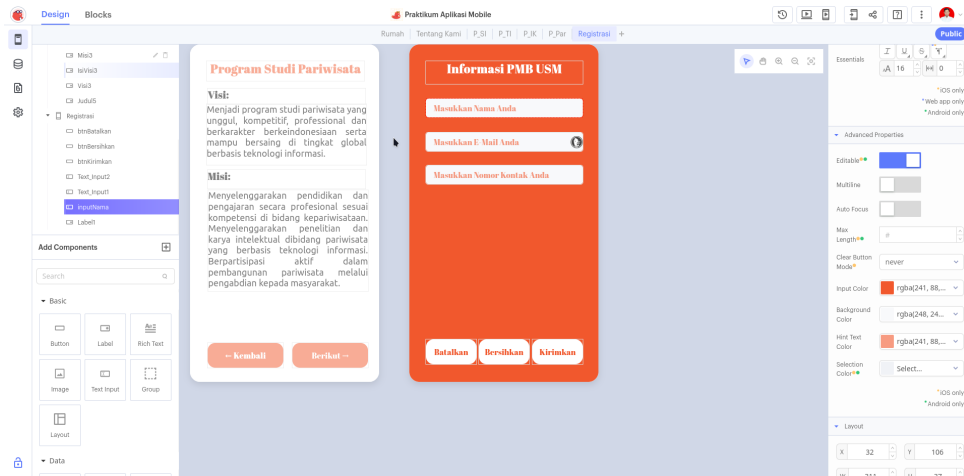


Gambar 5.3: Menghubungkan Akses Halaman

4. Di halaman baru **Registrasi** tambahkan komponen berikut secara berurutan:

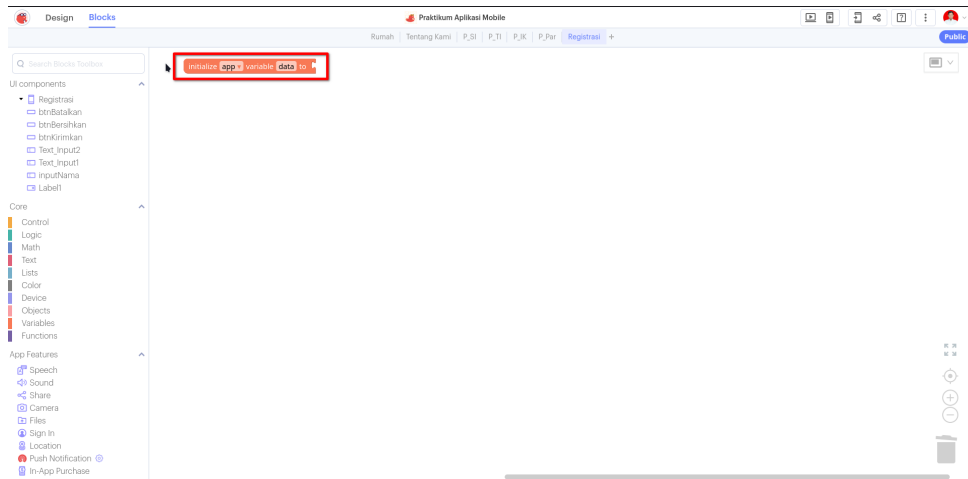
- **Label :**
 - **ID :** Judul6
 - **Text :** Informasi PMB USM
- **Text Input :**
 - **ID :** inputNama
 - **Hint :** Masukkan Nama Anda
 - **Keyboard :** default
- **Text Input :**

- ID : inputEmail
- Hint : Masukkan E-Mail Anda
- Keyboard : email address
- **Text Input :**
 - ID : inputPhone
 - Hint : Masukkan Nomor Kontak Anda
 - Keyboard : phone pad
- **Button :**
 - ID : btnKirim
 - Text : Kirimkan
- **Button :**
 - ID : btnBersihkakan
 - Text : Bersihkan
- **Button :**
 - ID : btnBatal
 - Text : Batalkan



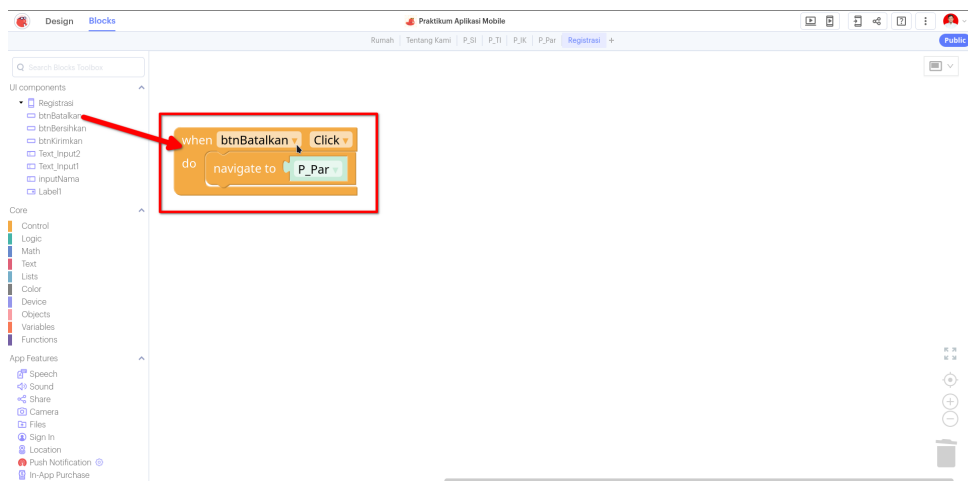
Gambar 5.4: Komponen Halaman Registrasi

5. Berikutnya adalah menyusun block **Program** dari halaman **Registrasi**
6. Pertama masukkan Komponen **Variable**, pilih **initialize app variable name to**. Ganti nama variabel menjadi **data**



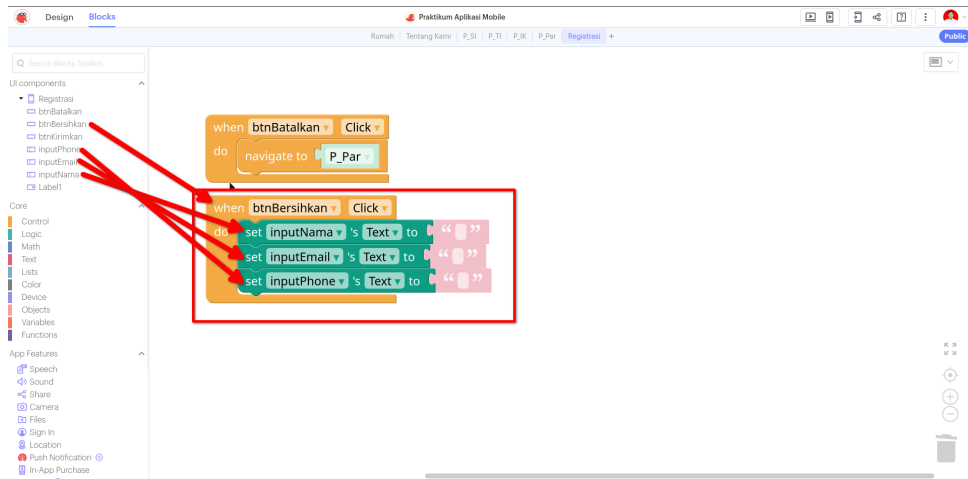
Gambar 5.5: Inisialisasi variable data

7. Setelah itu, buat aksi ketika Tombol **Batal** ditekan.



Gambar 5.6: Aksi Tombol Batalan

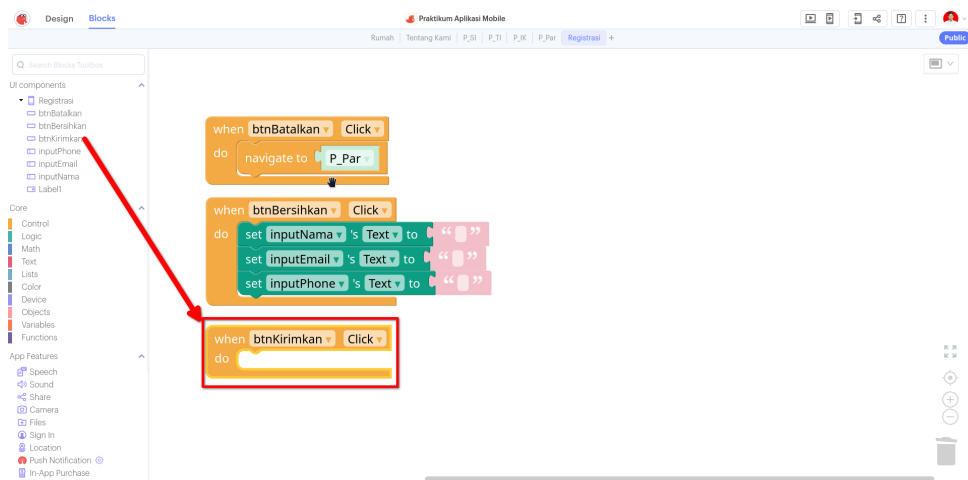
8. Lanjutkan dengan Tombol **Bersihkan** yang di mana melakukan **Set TEXT** menjadi Kosong ke **Input**



Gambar 5.7: Aksi Tombol Bersihkan

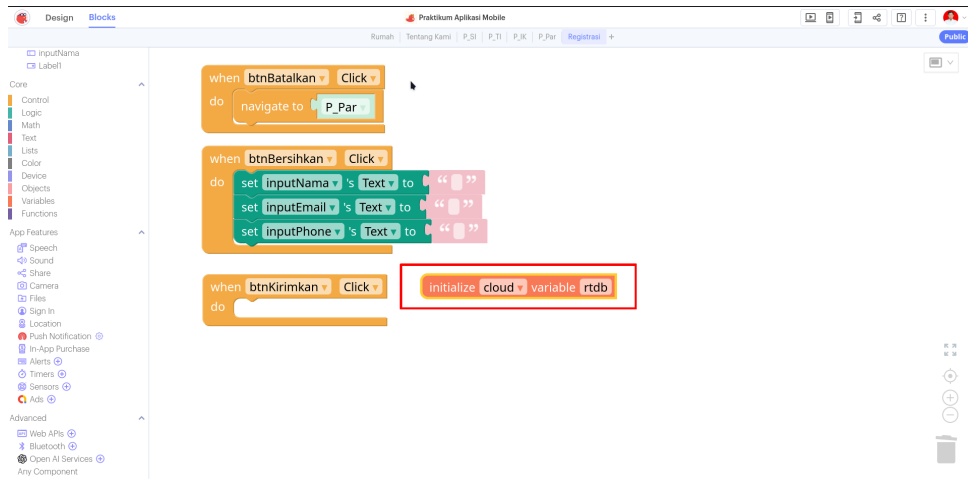
9. Komponen terakhir adalah Tombol **Kirimkan** yang berguna mengirimkan Data ke RTDB

(a) Dimulai dari Blok **Tombol btnKirimkan**



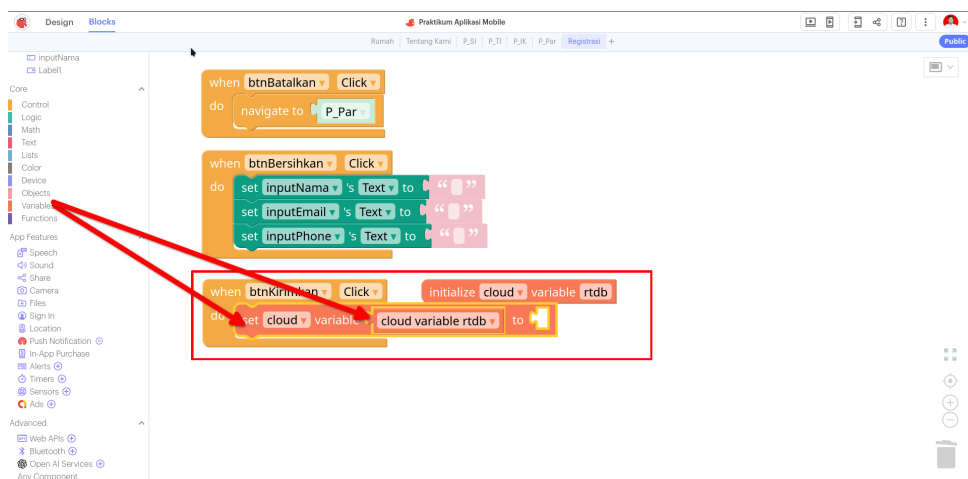
Gambar 5.8: Blok Awal Tombol Kirimkan

(b) Agar bisa mengakses database yang sudah dikonfigurasi sebelumnya. Buka Komponen **Variables** dan pilih **initialize ... variable**. Lalu ubah menjadi **cloud** dengan nama **rtddb**



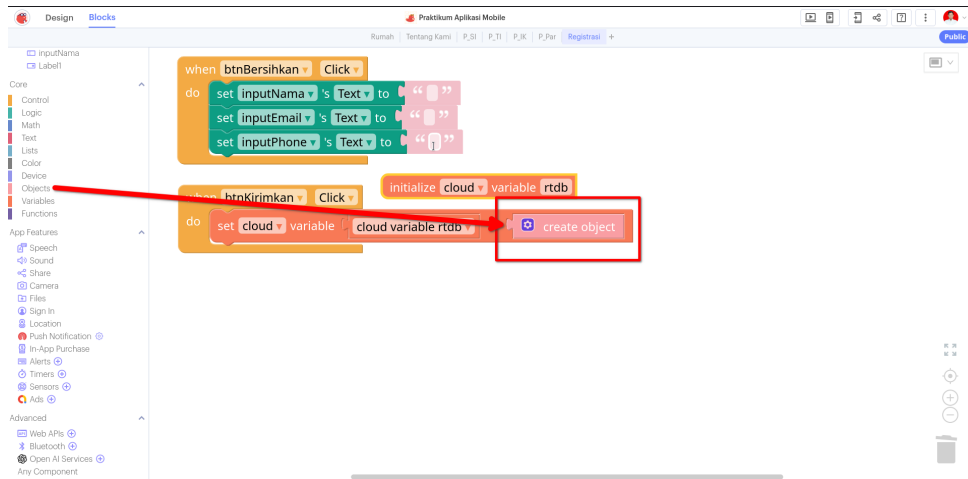
Gambar 5.9: Inisialisasi RTDB

- (c) Berikutnya adalah memberikan Komponen untuk mengirimkan data ke database. Buka Komponen **Variables** lagi dan pilih **set ... variable ... to**. Konfigurasikan ke mode **cloud**. Sedangkan **variabel** yang digunakan adalah Komponen **Variables** pilih **cloud variable rtdb**.



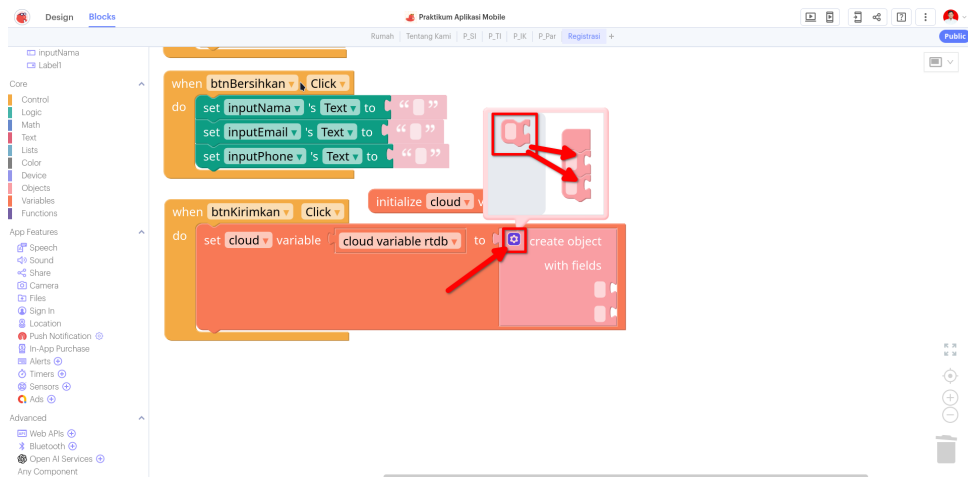
Gambar 5.10: Aksi Pengiriman Data Tanpa Data

- (d) Setelah konfigurasi dasar selesai, maka berikutnya adalah membentuk data agar bisa dikirimkan ke database. Pilih **Objects** dan pilih **create object**.



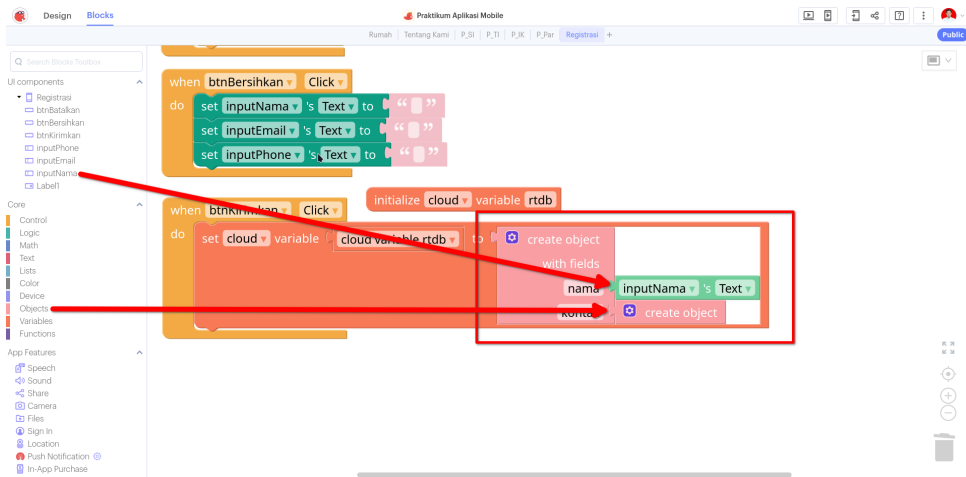
Gambar 5.11: Blok Awal Untuk Data JSON

- (e) Untuk lapisan pertama dari data JSON adalah nama. Tambahkan 2 **Property** di blok tersebut dengan menekan **Roda Gigi Biru** dan menarik **Potongan Blok Kiri ke Kanan**



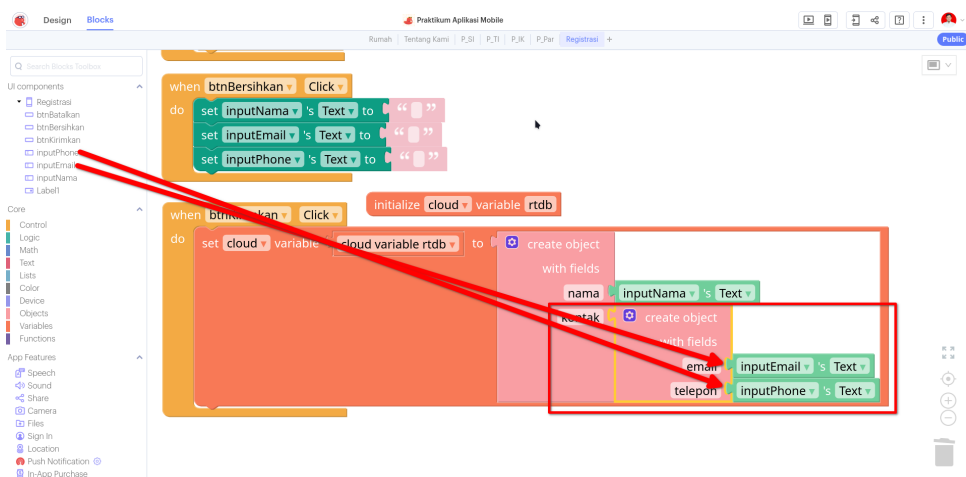
Gambar 5.12: Konfigurasi Create Objek 1

- (f) Setelah itu masukkan ke dalam kolom nama **Property** dengan **nama** dan **kontak**. Kolom **nama** disandingkan dengan **inputNama TEXT**, sedangkan **kontak** disandingkan dengan **create object** baru



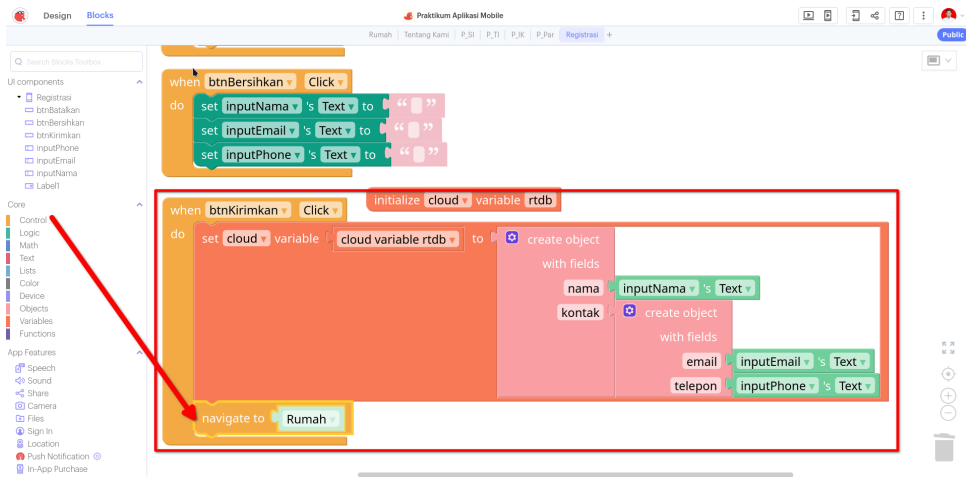
Gambar 5.13: Membuat JSON Bertingkat

- (g) Ulangi langkah (e) dengan membuat 2 **Property** untuk **email** dan **telepon**. Sandingkan masing-masing dengan **inputEmail** dan **inputPhone**



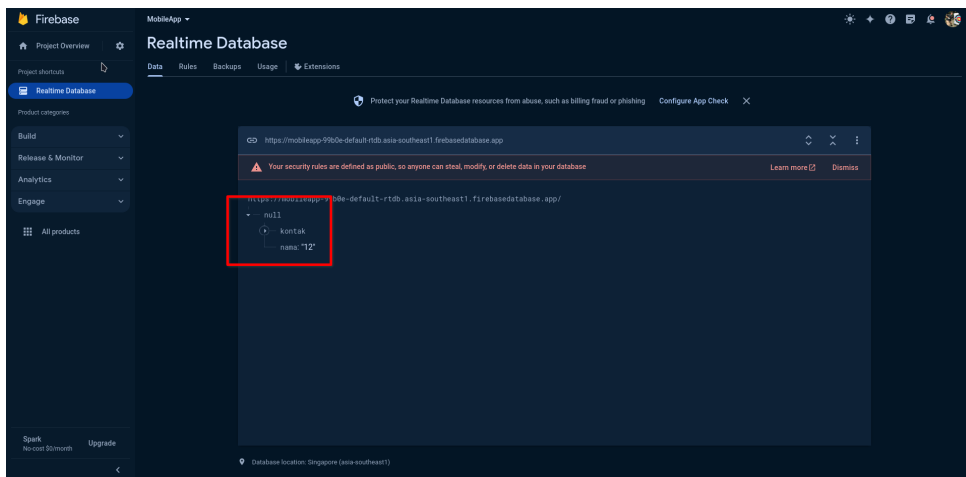
Gambar 5.14: Hasil Akhir JSON Bertingkat

- (h) Setelah tambahkan Komponen **Control** untuk navigasi kembali ke halaman awal



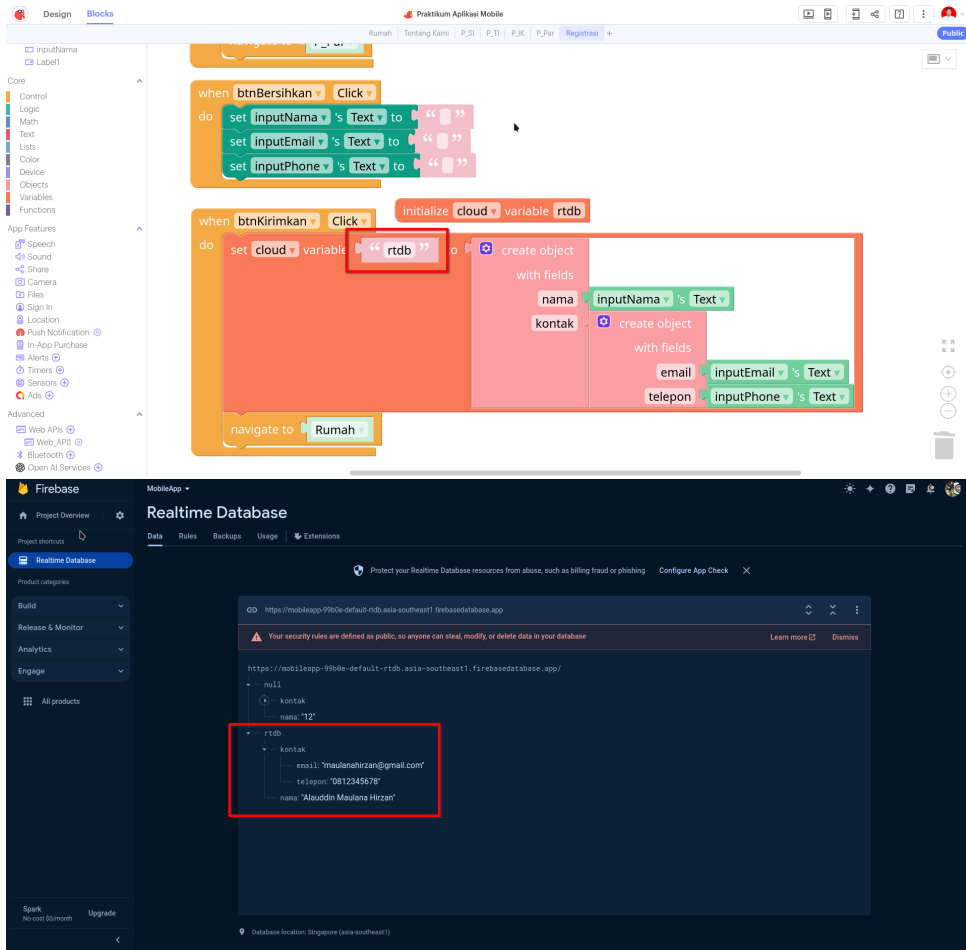
Gambar 5.15: Aksi Terakhir Kembali ke Home

10. Hasil Kueri adalah **null** dengan isi Nama dan Kontak.



Gambar 5.16: Hasil Sementara

11. Untuk bisa mencegah ini bisa dengan mengganti komponen **cloud variable rtdb** dengan teks **"rtdb"**



Gambar 5.17: Hasil Akhir

Bab 6

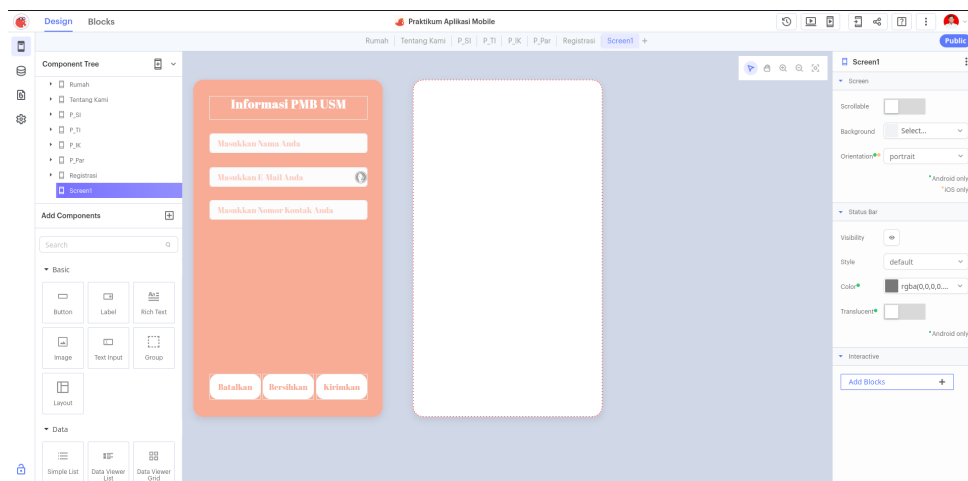
Praktikum 6

6.1 Read Database

Di bagian ini mahasiswa diajarkan Bagaimana membaca data dari Firebase **Realtime Database** menggunakan **Thunkable**. Mahasiswa diwajibkan menyelesaikan Praktikum 5 sebelum melanjutkan ke bagian ini

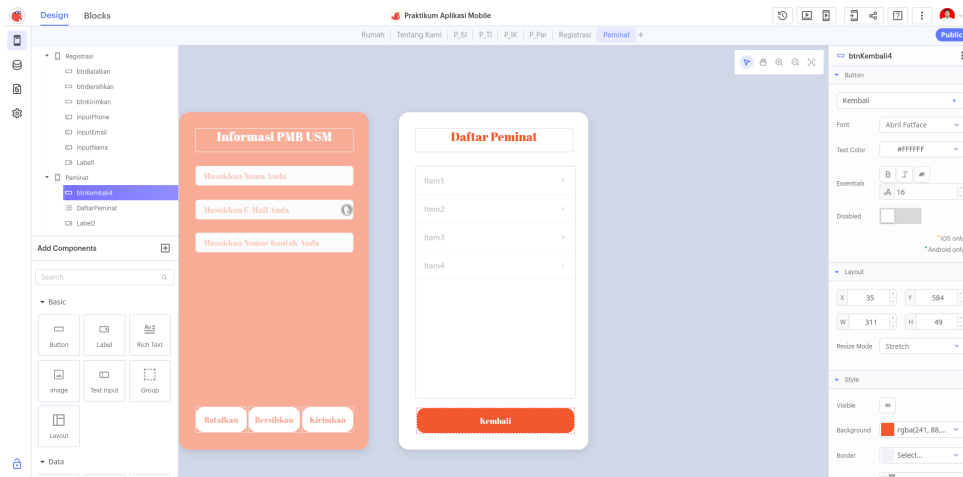
6.2 Tutorial

1. Buka kembali projek yang telah dibuat, lalu buat satu layar baru dengan nama **Peminat**.



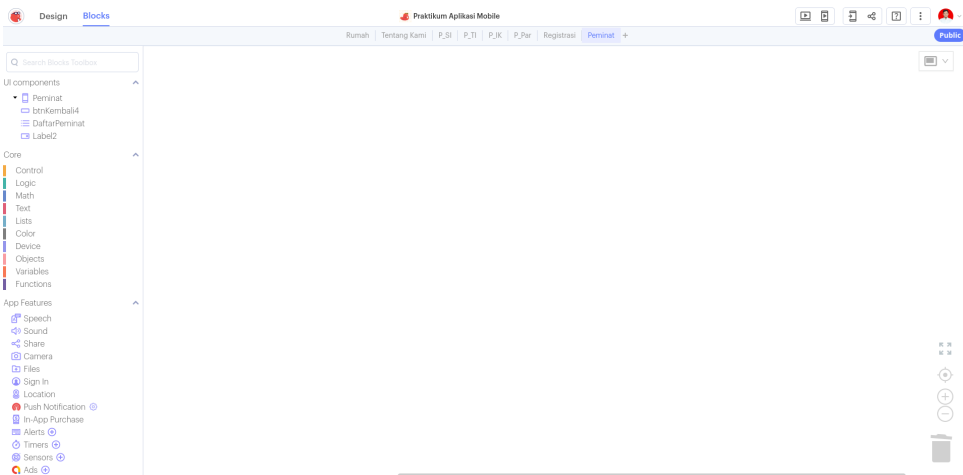
Gambar 6.1: Tambah Layar Baru

2. Lalu tambahkan komponen seperti berikut:
 - **Label** : text → Daftar Peminat
 - **Data** → **Simple List** : id → Daftar Peminat
 - **Button** : **Kembali**



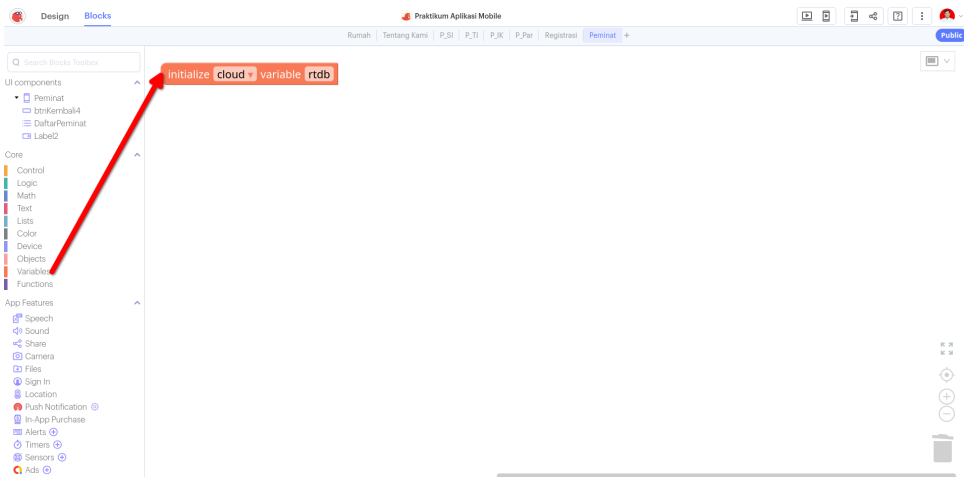
Gambar 6.2: Tambah Komponen ke Layar

3. Masuk ke mode **Block** untuk mengatur alur program.



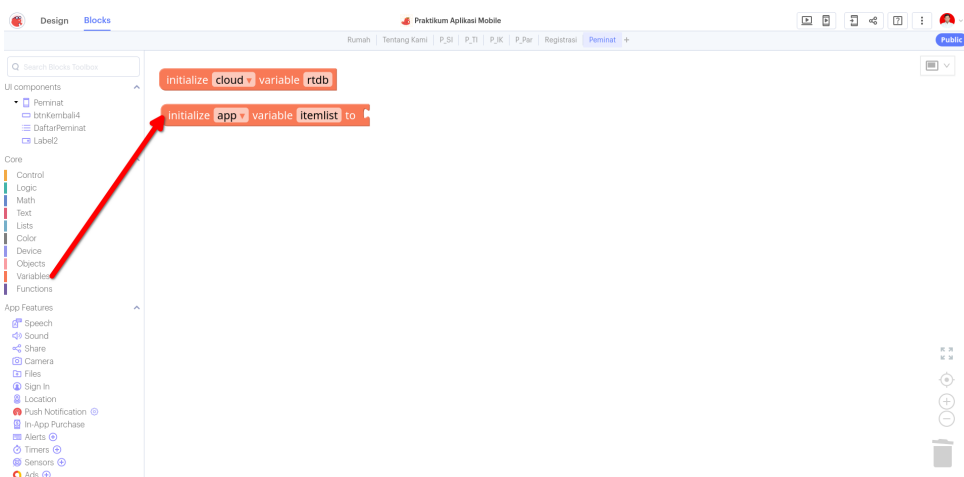
Gambar 6.3: Masuk mode Block

4. Masukkan **Variables** → **initialize ... variable ...**. Lalu konfigurasi ke mode **cloud** dengan variabel **rtdb**



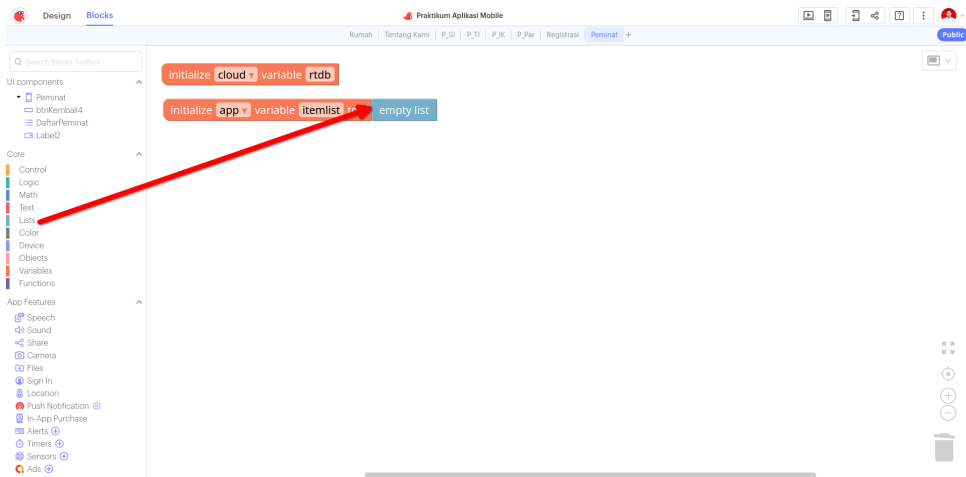
Gambar 6.4: Inisialisasi Variabel Clouds

5. Tambahkan satu **initialize ... variable ...** untuk membuat variabel lokal yang menampung data dari **rtdb**. Beri nama **itemlist**:



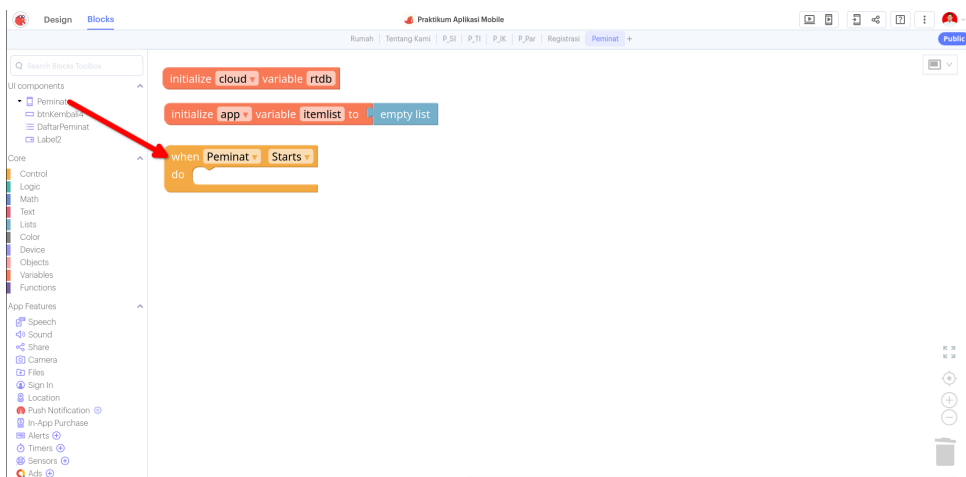
Gambar 6.5: Inisialisasi Variabel App

6. Dari komponen **Lists** → **empty list**, sambungkan ke komponen sebelumnya



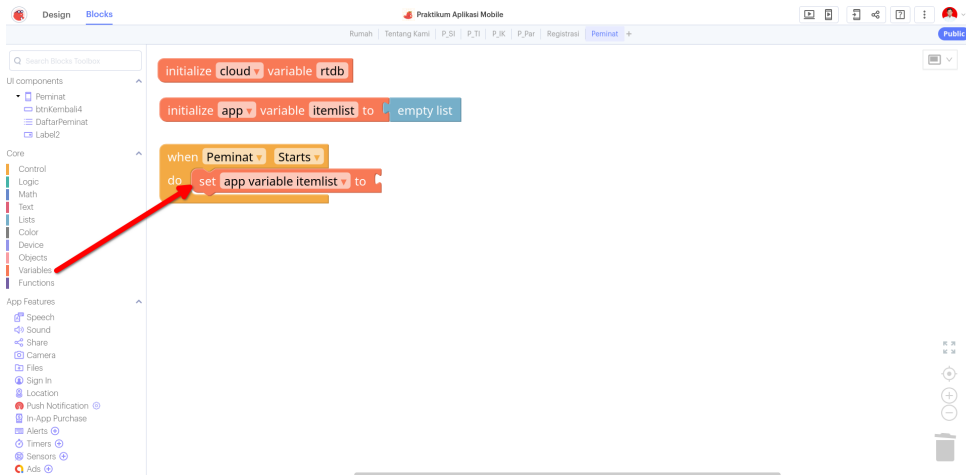
Gambar 6.6: Mengatur isi dari **itemlist**

- Agar data dapat diambil ketika **Halaman Daftar Peminat** dibuka, gunakan **Halaman Peminat** dan pilih **when Peminat starts do ...**



Gambar 6.7: Eksekusi Kondisi Peminat

- Tambahkan komponen **Variables** → **set app variable itemlist to** di dalam Komponen sebelumnya



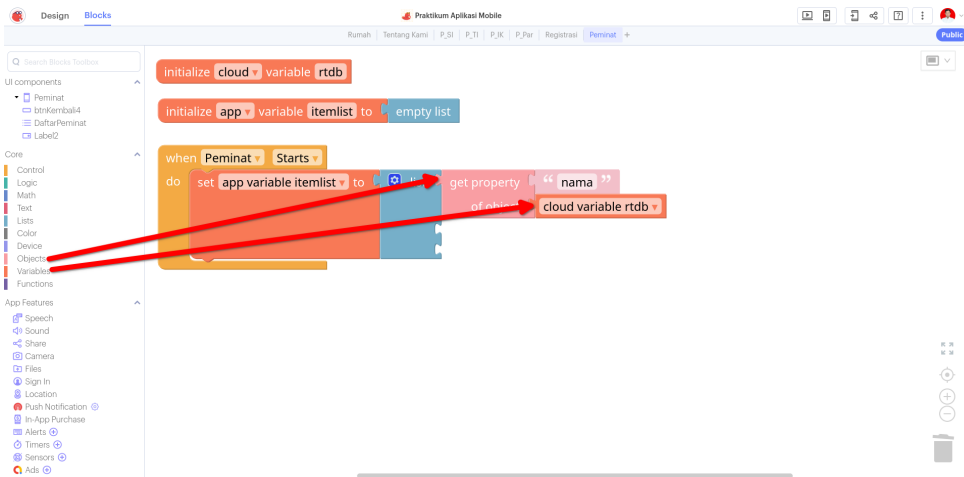
Gambar 6.8: Komponen Set Variable

9. Tambahkan Komponen **Lists** → **List 1 2 3** dengan **roda gigi biru**. Hapus komponen teks 1 2 3.



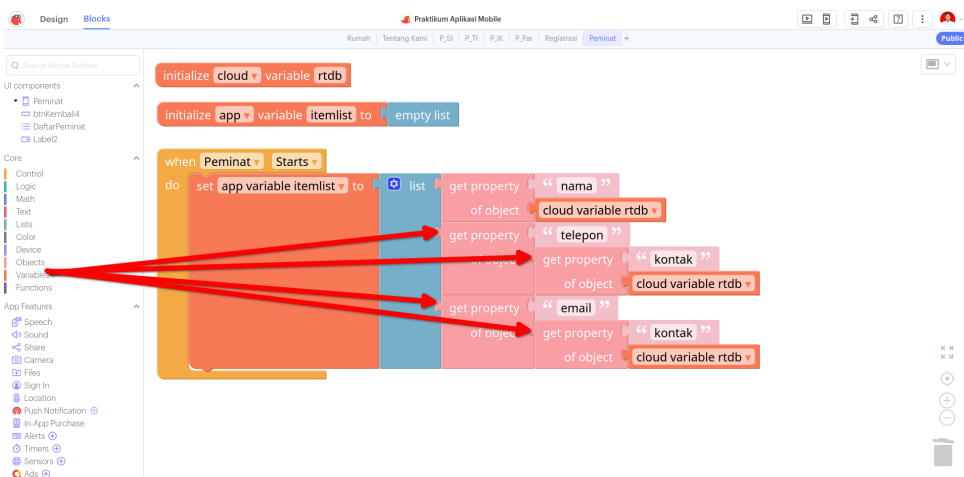
Gambar 6.9: Komponen Set Variable

10. Tambahkan Komponen **Objects** → **get property ... of object ...**. Untuk bagian atas diisi **nama**, sedangkan yang bawa diisi komponen **Variables** → **cloud variable rtdb**



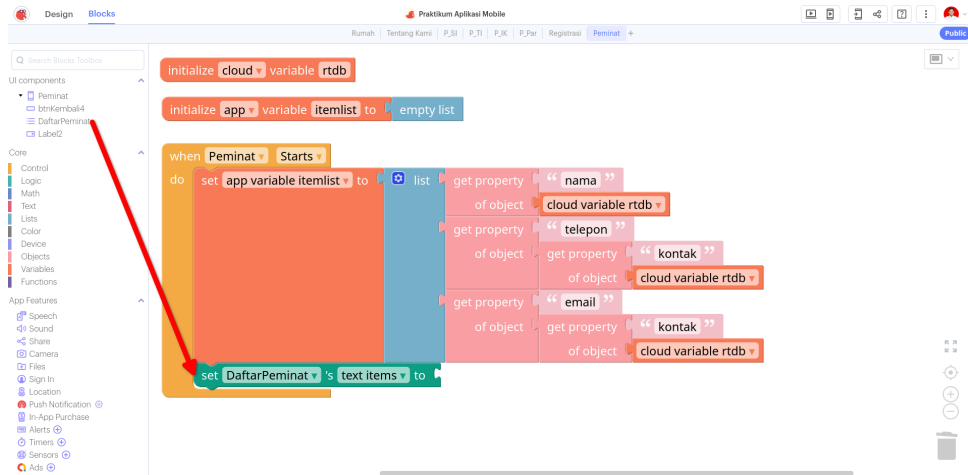
Gambar 6.10: Mengambil Data Nama

11. Untuk mengambil data **Email** maupun **Telepon** membutuhkan lapisan ganda dari **get property .. of object ...**. Perhatikan gambar berikut:



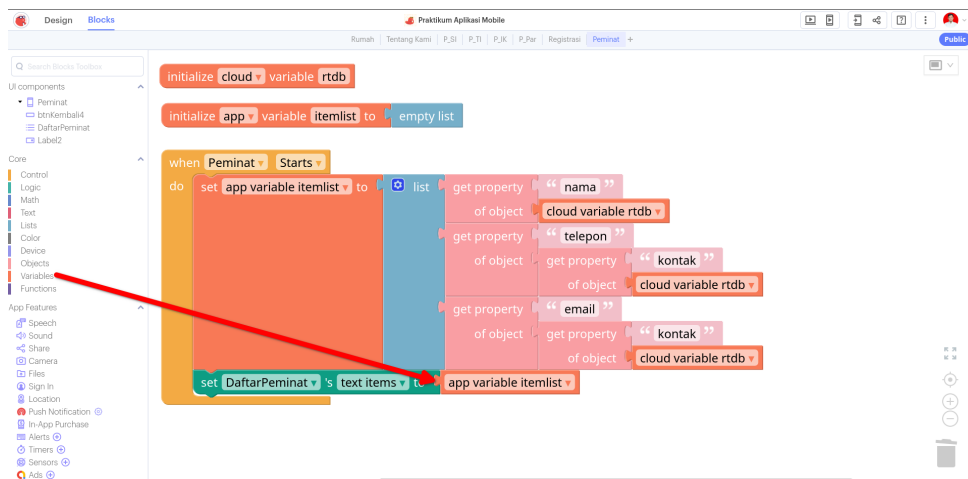
Gambar 6.11: Mengambil Data Email dan Kontak

12. Variable **itemlist** sudah diisi. Berikutnya adalah mengatur Komponen **Lists** → **DaftarPeminat** untuk mengambil data tersebut. Pilih **DaftarPeminat** → **set DaftarPeminat's text items to ...** di bawah komponen sebelumnya



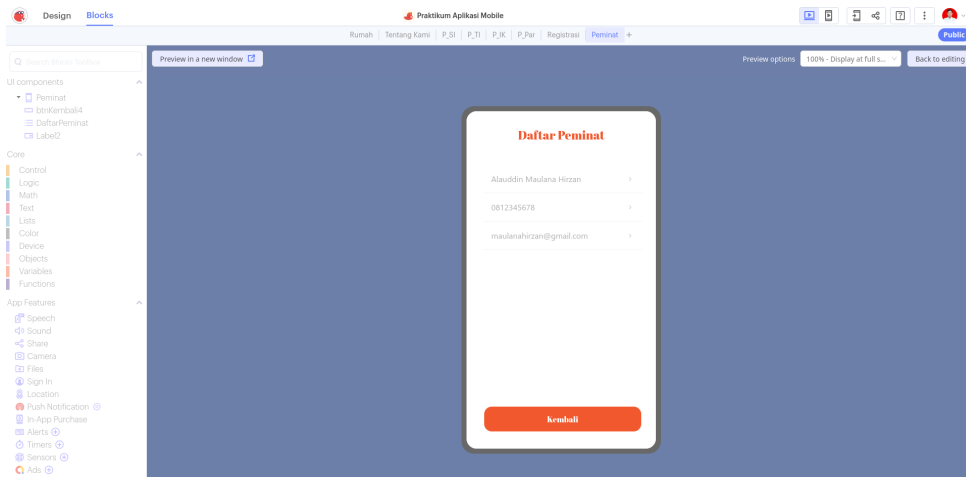
Gambar 6.12: Komponen set Item

13. Terakhir cukup menarik variable **itemlist** ke **set item** tersebut



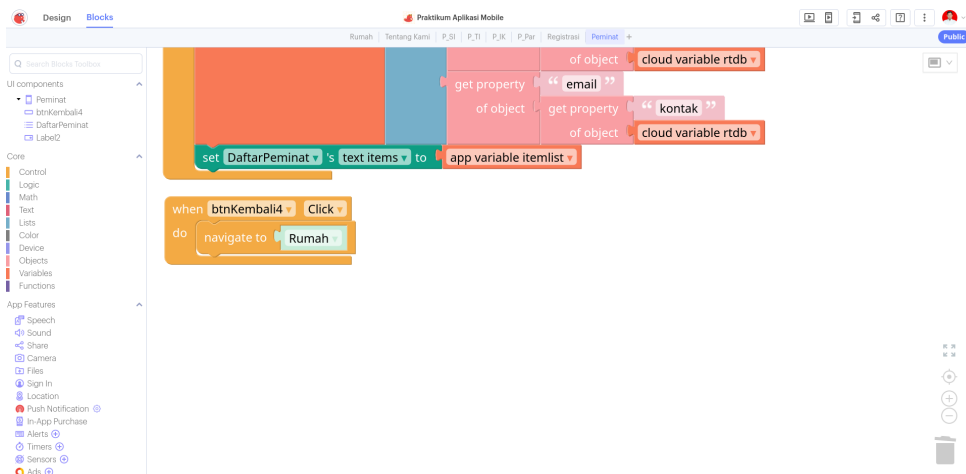
Gambar 6.13: Menarik itemlist ke list

14. Tes aplikasi agar tampil seperti berikut:



Gambar 6.14: Tes Tampilan

15. Sentuhan terakhir adalah menambahkan aksi ke **btnKembali** untuk navigasi ke **Halaman Rumah**



Gambar 6.15: Tes Tampilan

Bab 7

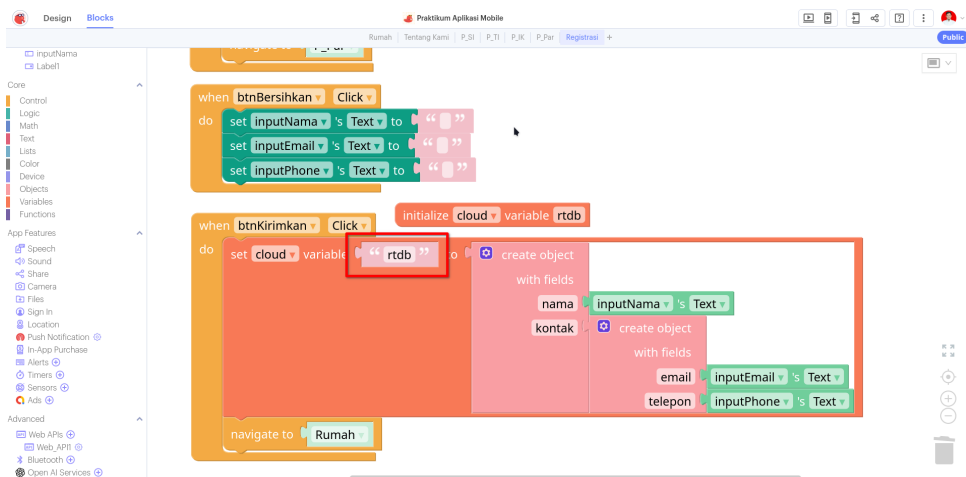
Praktikum 7

7.1 Multi Data Entry dan Display

Di bagian ini mahasiswa diajarkan Bagaimana melakukan input banyak data ke Fire-base **Realtime Database** menggunakan **Thunkable** dan cara membacanya. Mahasiswa diwajibkan menyelesaikan Praktikum 6 sebelum melanjutkan ke bagian ini

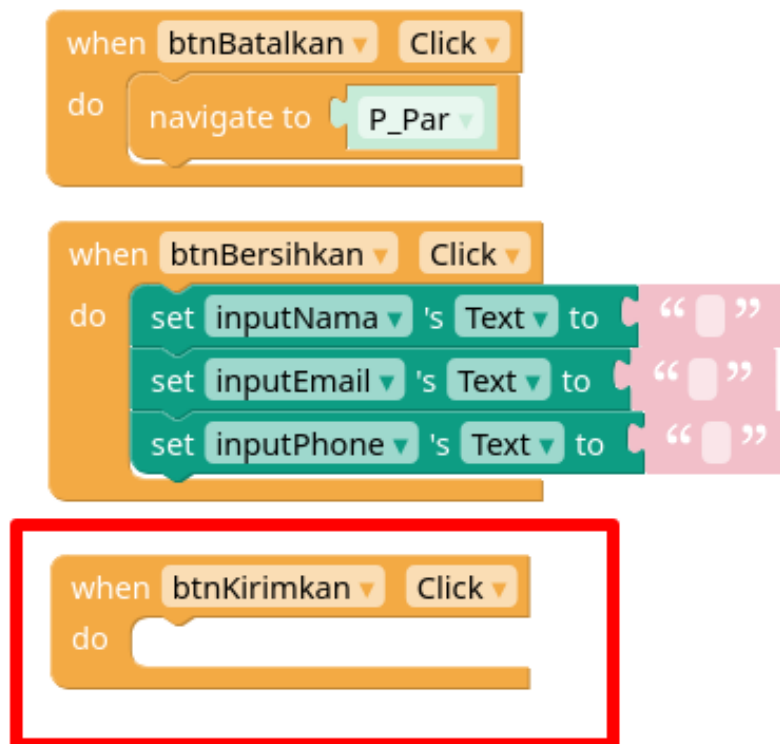
7.2 Tutorial

1. Buka kembali proyek yang sudah dibuat sebelumnya. Buka bagian **Block - Registrasi**



Gambar 7.1: Buka Praktikum 6 dan Bagian Block - Registrasi

2. Hapus blok yang ada di dalam **btnKirimkan** sehingga kosong seperti di gambar berikut:



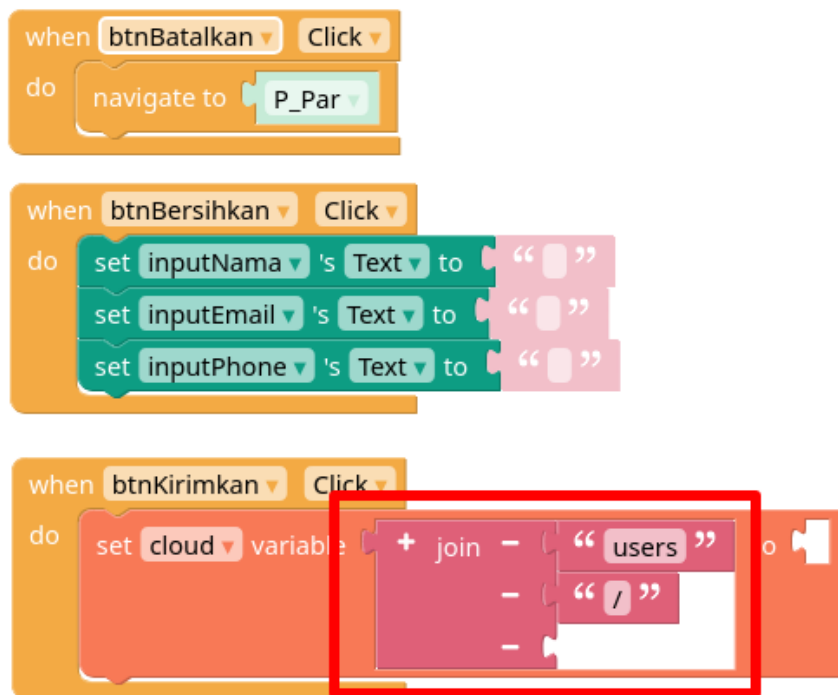
Gambar 7.2: Hapus bagian dalam **btnKirimkan**

3. Sesuai dihapus, masukkan blok **set ... variable ... to** ke dalam blok **btnKirimkan**



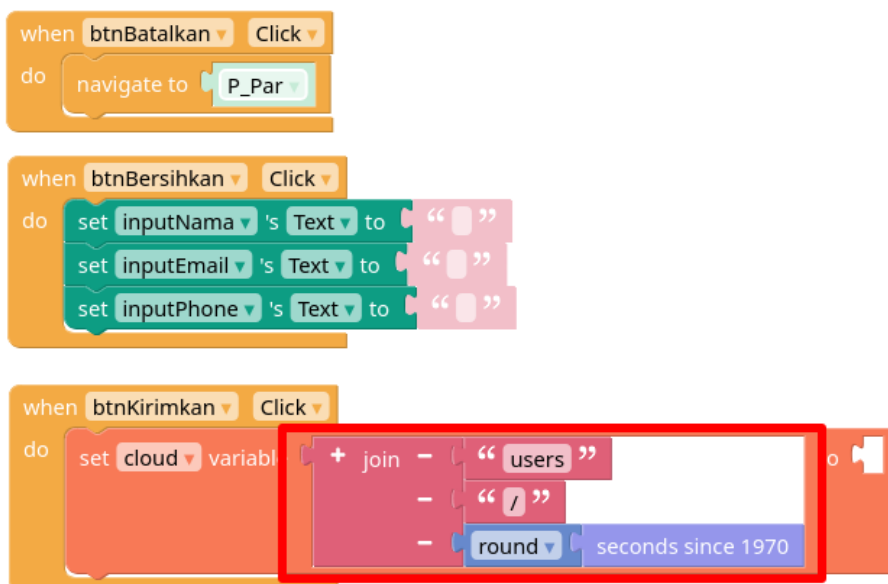
Gambar 7.3: Memasukkan fungsi tambah data

4. Untuk mengeset nama variable secara dinamis, masukkan blok **Text** dengan jenis **join ...**. Di dalamnya tambahkan 1 blok ekstra agar muat **3 Blok Teks**, Blok **Text** pertama diisi **users** dan kedua adalah **/**.



Gambar 7.4: Memasukkan Nama Variabel

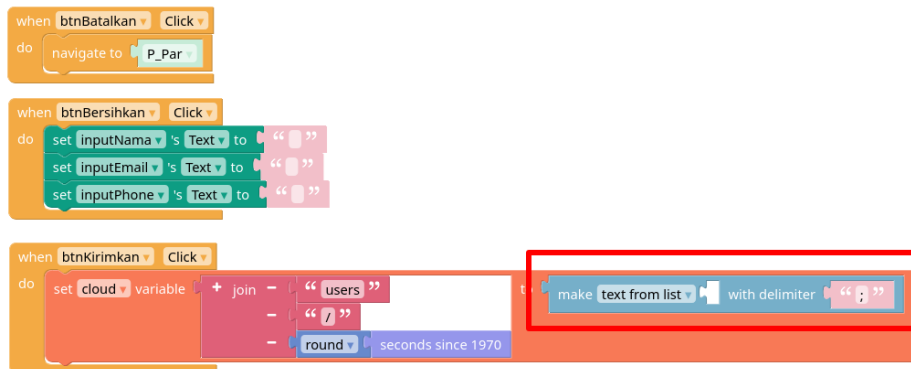
5. Berikutnya mengisikan nama dinamis untuk variable. Di sini menggunakan dua jenis blok: Blok **Devices** jenis **second since 1970** disambungkan ke Blok **Math** jenis **round**. Lalu sambungkan ke colokkan ketiga bloknnya



Gambar 7.5: Memasukkan Nama Variabel Dinamis

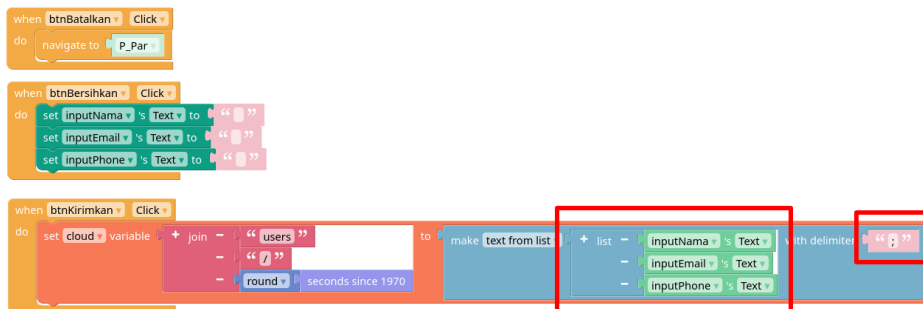
6. Sesudah lengkap, berikutnya adalah membangun datanya. Data dirubah dalam

bentuk list. Dari Blok **List** ambil **make list from text ... with delimiter ...**. Ubah mode blok ini ke **make text from list**



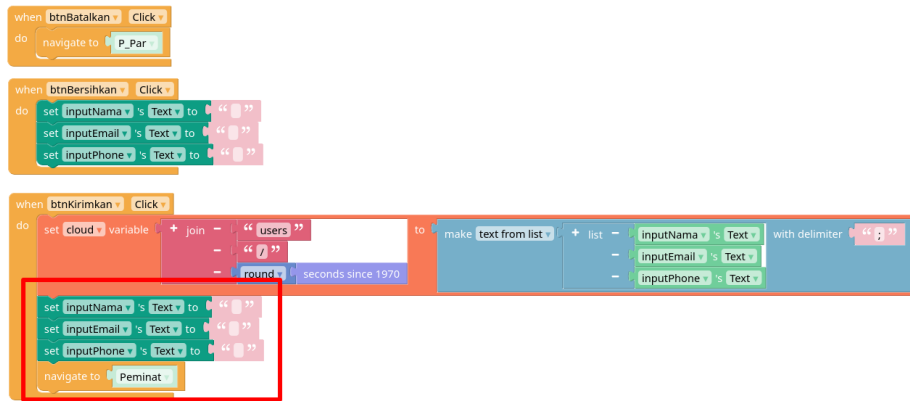
Gambar 7.6: Memasukkan Blok List

7. Di bagian ini adalah membangun data, dari blok **Lists** ambil **+ list -**. Blok ini kemudian dimasukkan input dari masing-masing **inputName**, **inputEmail**, dan **inputPhone**. Ubah **Delimiter** menjadi **Titik Koma** ;



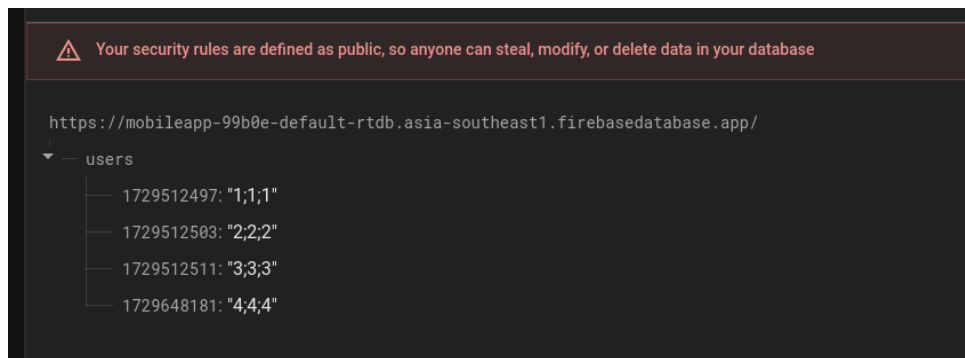
Gambar 7.7: Memasukkan Data ke List

8. Setelah itu tambahkan blok dari **btnBersihkan** dengan cara di duplikat. Lalu letakkan di bawah blok **set cloud variable ...** tadi. Sekaligus blok navigasi otomatis ke halaman **Peminat**



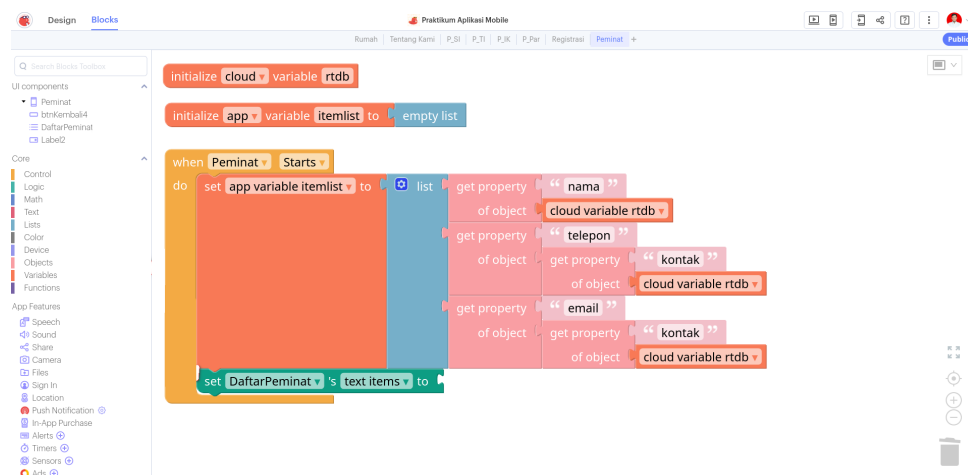
Gambar 7.8: Memasukkan Blok Pembersih

9. Tes aplikasi untuk entri data



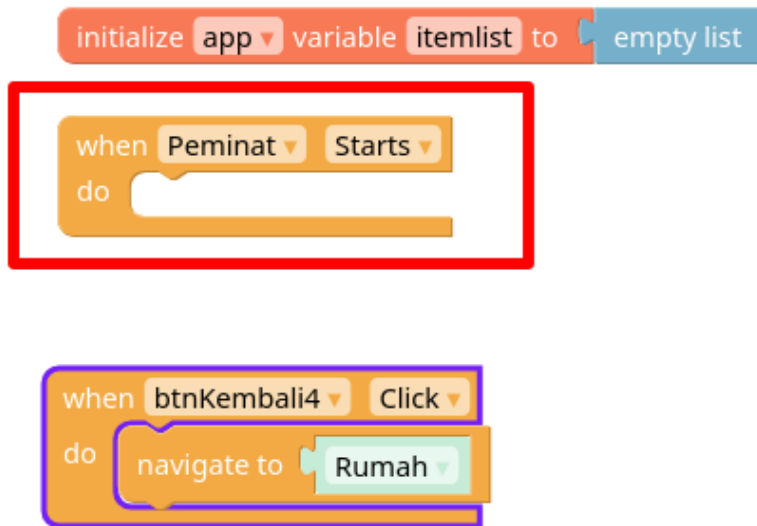
Gambar 7.9: Hasil Register

10. Pindah ke halaman Peminat dan buka Blocks



Gambar 7.10: Pindah Halaman Peminat

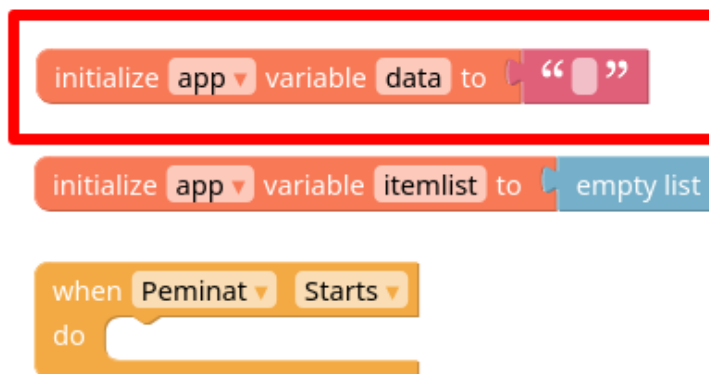
11. Hapus apa yang ada di dalam when Peminat Starts ...



Gambar 7.11: Hapus Blok di dalam **Peminat**

12. Tambahkan dua variabel baru dari blok **Variable**:

- (a) Blok **initialize app variable data to ...** disambungkan dengan blok **Text** kosong



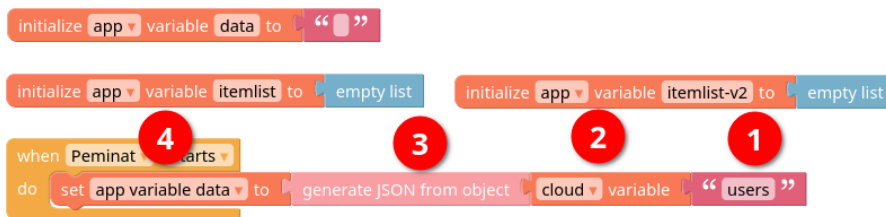
Gambar 7.12: Variable **data** dengan **Text** kosong

- (b) Blok **initialize app variable itemlist-v2 to ...** disambungkan dengan blok **Lists** jenis **Empty List**

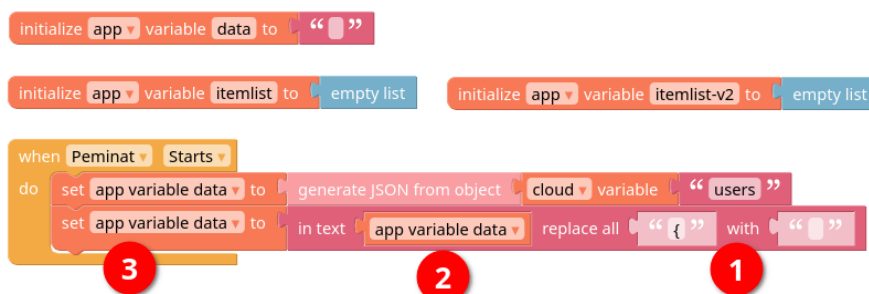


Gambar 7.13: Variable **itemlist-v2** dengan empty list

13. Berikutnya adalah proses pengambilan dan pemrosesan data. Di bagian ini ada beberapa blok yang akan digunakan dimulai dari lapisan 1 dan seterusnya.
14. Lapisan 1 : Pengambilan data dari Cloud dan disimpan ke variable **data**.
 - (a) Blok **Text** dengan isi **users**
 - (b) Blok **Variable** dengan jenis **cloud variable ...**
 - (c) Blok **Object** dengan jenis **generate JSON from object ...**
 - (d) Blok **Variable** dengan jenis **set app variable data to ...**



15. Lapisan 2 : Menghapus simbol { menjadi kosong
 - (a) Blok **Text** jenis **in text ... replace all ... with ...**
 - (b) Isi **replace all** dengan { dan kosongkan sebelumnya
 - (c) Blok **Variable** jenis **app variable data**
 - (d) Blok **Variable** jenis **set app variable data to ...**



16. Lapisan 3 : Menghapus simbol } menjadi kosong

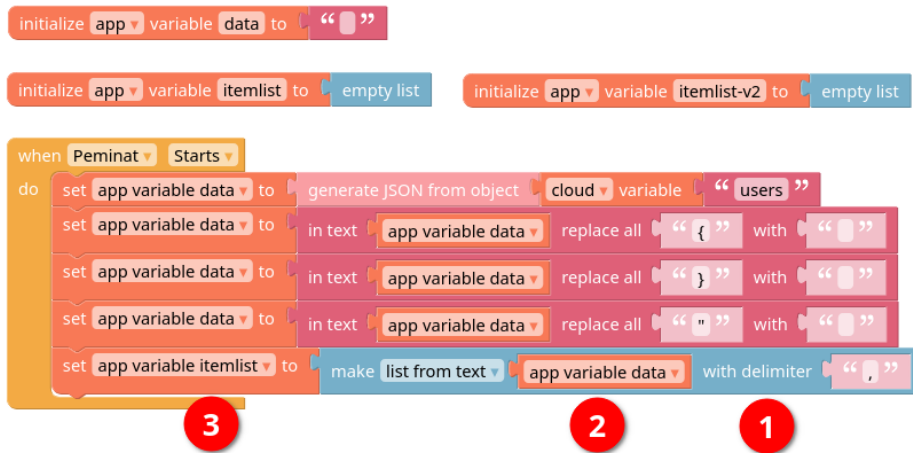
- (a) Blok **Text** jenis **in text ... replace all ... with ...**
- (b) Isi **replace all** dengan **}**
- (c) Blok **Variable** jenis **app variable data**
- (d) Blok **Variable** jenis **set app variable data to ...**

17. Lapisan 4 : Menghapus simbol " menjadi kosong

- (a) Blok **Text** jenis **in text ... replace all ... with ...**
- (b) Isi **replace all** dengan **"**
- (c) Blok **Variable** jenis **app variable data**
- (d) Blok **Variable** jenis **set app variable data to ...**

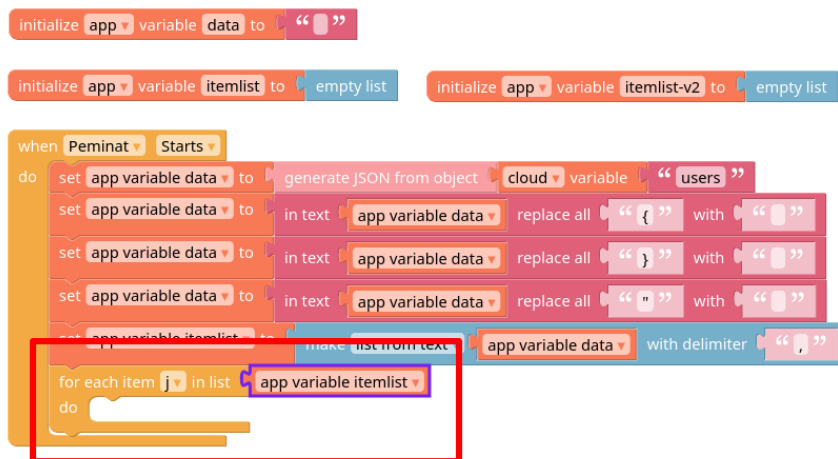
18. Lapisan 5 : Memecah data teks tadi menjadi list.

- (a) Blok **Lists** dengan jenis **make list from text ... with delimiter ...**
- (b) Blok **Variable** dengan jenis **app variable data** dan delimiter **Koma ,**
- (c) Blok **Variable** dengan jenis **set app variable itemlist to ...**



19. Lapisan 5 : Iterasi data list untuk diproses lebih lanjut

- (a) Blok **Control** jenis **for each item ... in list ... do** (blok ini akan membuat variable baru secara otomatis, contoh : j)
- (b) Blok **Variable** jenis **app variable itemlist**



20. Lapisan 6 : Pemrosesan data list

- (a) Blok **Text** ambil jenis **in text ... replace all ... with ...**
- (b) Blok **Text** ambil jenis **newline** untuk kotak kosong sisi kanan, dan sisi kiri dengan simbol :
- (c) Blok **Variable** dengan jenis **j** (jika variabel otomatis adalah j)
- (d) Blok **Variable** ambil jenis **set ... to ...**

```

initialize app variable data to ""
initialize app variable itemlist to empty list
initialize app variable itemlist-v2 to empty list

when Peminat Starts
do
  set app variable data to generate JSON from object cloud variable "users"
  set app variable data to in text app variable data replace all "{" with "\""
  set app variable data to in text app variable data replace all "}" with "\""
  set app variable data to in text app variable data replace all "\"" with "\""
  set app variable itemlist to make list from text app variable data with delimiter ","
  for each item j in list app variable itemlist
  do
    set j to in text j replace all ":" with newline
    set j to in text j replace all ":" with newline
    set j to in text j replace all ":" with newline
    set j to in text j replace all ":" with newline
  
```

21. Lapisan 7 : Duplikat Lapisan 6 dan ganti : dengan **Titik Koma ;** ;

```

initialize app variable data to ""
initialize app variable itemlist to empty list
initialize app variable itemlist-v2 to empty list

when Peminat Starts
do
  set app variable data to generate JSON from object cloud variable "users"
  set app variable data to in text app variable data replace all "{" with "\""
  set app variable data to in text app variable data replace all "}" with "\""
  set app variable data to in text app variable data replace all "\"" with "\""
  set app variable itemlist to make list from text app variable data with delimiter ","
  for each item j in list app variable itemlist
  do
    set j to in text j replace all ":" with newline
    set j to in text j replace all ";" with newline
  
```

22. Lapisan 8 : Memasukkan data dari variable **j** (jika variabel otomatis adalah **j**) ke list **itemlist-v2**

- (a) Blok **Lists** pilih jenis **in list ... insert at last as ...**
- (b) Blok **Variable** pilih **j** (jika variabel otomatis adalah **j**)
- (c) Blok **Variable** pilih **app variable itemlist-v2**


```

initialize app variable data to ""
initialize app variable itemlist to empty list
initialize app variable itemlist-v2 to empty list

when Peminat Starts
do
  set app variable data to generate JSON from object cloud variable "users"
  set app variable data to in text app variable data replace all "{" with "\"{"
  set app variable data to in text app variable data replace all "}" with "\"}"
  set app variable data to in text app variable data replace all "\"" with "\"\""
  set app variable itemlist to make list from text app variable data with delimiter ","
  for each item j in list app variable itemlist
  do
    set j to in text j replace all ":" with "\n"
    set j to in text j replace all ";" with "\n"
    in list app variable itemlist-v2 insert at last as j
  
```

23. Terakhir adalah menampilkan data dari variabel **itemlist-v2**. Gunakan blok **DaftarPeminat's text items to ...** disambukan dengan blok **Variable** jenis **app variable itemlist-v2**. Letakkan blok ini tepat di bawah **for each item ...**

```

initialize app variable data to ""
initialize app variable itemlist to empty list
initialize app variable itemlist-v2 to empty list

when Peminat Starts
do
  set app variable data to generate JSON from object cloud variable "users"
  set app variable data to in text app variable data replace all "{" with "\"{"
  set app variable data to in text app variable data replace all "}" with "\"}"
  set app variable data to in text app variable data replace all "\"" with "\"\""
  set app variable itemlist to make list from text app variable data with delimiter ","
  for each item j in list app variable itemlist
  do
    set j to in text j replace all ":" with "\n"
    set j to in text j replace all ";" with "\n"
    in list app variable itemlist-v2 insert at last as j
  set DaftarPeminat's text items to app variable itemlist-v2
  
```

24. Tes Aplikasi. Hasil akan terlihat lebih rapi jika menggunakan data asli

Daftar Peminat

1729512497

1

1

1

>

1729512503

2

2

2

>

1729512511

3

3

3

>

1729648181

4

4

4

>

Kembali

Bab 8

Praktikum 8

8.1 Proyek Aplikasi

Di bagian ini mahasiswa diminta untuk membuat aplikasi dengan menggunakan **Thunkable** berdasarkan konsep-konsep yang telah diajarkan di teori. Mahasiswa diwajibkan menyelesaikan Praktikum 7 sebelum melanjutkan ke bagian ini

8.2 Deskripsi Proyek

- Buatlah prototipe antarmuka aplikasi
- Mahasiswa memilih 1 Jenis Aplikasi yang sudah di tentukan di sini

8.3 Daftar Jenis Aplikasi

1. Aplikasi Jejaring Sosial:
 - (a) Halaman Masuk/Pendaftaran
 - (b) Umpan Berita
 - (c) Halaman Profil
 - (d) Halaman Pesan
 - (e) Halaman Pemberitahuan
 - (f) Halaman Teman/Koneksi
 - (g) Halaman Pengaturan
 - (e) Halaman Pembayaran
 - (f) Riwayat Pesanan
 - (g) Pengaturan Akun
2. Aplikasi E-commerce:
 - (a) Layar Pembuka
 - (b) Daftar Produk
 - (c) Halaman Detail Produk
 - (d) Keranjang Belanja
 3. Aplikasi Pelacakan Kebugaran:
 - (a) Halaman Perkenalan / Selamat Datang
 - (b) Pelacak Aktivitas Harian
 - (c) Rencana Latihan
 - (d) Dasbor Kemajuan
 - (e) Pelacak Kalori
 - (f) Lencana Pencapaian
 - (g) Pengaturan/Preferensi
 4. Aplikasi Pemesanan Perjalanan:

- (a) Pencarian Destinasi
 - (b) Daftar Hotel
 - (c) Pemesanan Penerbangan
 - (d) Perencana Rencana Perjalanan
 - (e) Peta dan Navigasi
 - (f) Ulasan dan Peringkat
 - (g) Halaman Profil/Rekening
5. Aplikasi Perbankan:
- (a) Ikhtisar Rekening
 - (b) Riwayat Transaksi
 - (c) Transfer Dana
 - (d) Pembayaran Tagihan
 - (e) Pencari Lokasi ATM/Cabang
 - (f) Pelacak Anggaran
 - (g) Halaman Bantuan/Dukungan
6. Aplikasi Berita:
- (a) Berita Utama/Berita Terpopuler
 - (b) Kategori/Bagian
 - (c) Tampilan Artikel
 - (d) Penandaan/Favorit
 - (e) Fungsi Pencarian
 - (f) Pemberitahuan untuk Berita Terbaru
 - (g) Preferensi/Pengaturan
7. Aplikasi Pendidikan:
- (a) Katalog Kursus
 - (b) Modul Pelajaran
 - (c) Halaman Kuis/Penilaian
 - (d) Pelacak Kemajuan
 - (e) Forum Diskusi
 - (f) Perpustakaan Sumber Daya
 - (g) Manajemen Profil/Akun
8. Aplikasi Cuaca:
- (a) Kondisi Cuaca Saat Ini
 - (b) Prakiraan Cuaca Per Jam
 - (c) Prakiraan Harian
 - (d) Tampilan Radar/Peta
 - (e) Peringatan/Peringatan
 - (f) Manajemen Lokasi
 - (g) Preferensi/Pengaturan
9. Aplikasi Pembelajaran Bahasa:
- (a) Pemilihan Pelajaran
 - (b) Latihan Kosakata
 - (c) Latihan Tata Bahasa
 - (d) Pelacakan Kemajuan
 - (e) Latihan Pengucapan
 - (f) Forum Komunitas
 - (g) Manajemen Akun
10. Aplikasi Pencarian Pekerjaan:
- (a) Daftar Pekerjaan
 - (b) Pencarian Tersimpan
 - (c) Pembuat Lanjutkan
 - (d) Pelacak Aplikasi
 - (e) Profil Perusahaan
 - (f) Persiapan Wawancara
 - (g) Pengaturan Akun
11. Aplikasi Manajemen Tugas:
- (a) Daftar Tugas
 - (b) Detail Tugas/Halaman Edit
 - (c) Tampilan Kalender
 - (d) Pengingat/Pemberitahuan
 - (e) Fitur Kolaborasi/Berbagi
 - (f) Tag/Label
 - (g) Pengaturan/Preferensi Akun
12. Aplikasi Resep:

- (a) Koleksi Resep
- (b) Detail Resep
- (c) Daftar Bahan
- (d) Petunjuk Memasak
- (e) Perencanaan Makanan
- (f) Daftar Belanja
- (g) Manajemen Akun