

Spesifikasi Kebutuhan Perangkat Lunak Aplikasi FitLife

Dosen Pengampu:
Hayatunnufus, S.Kom, M.Cs



Oleh :

Michael Valent Satrio Munthe	231401103
Susena Yudha Wijaya	231401109

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

TAHUN 2025

DAFTAR ISI

1. Pendahuluan	2
1.1 Tujuan Penulisan Dokumen	2
1.2 Lingkup Produk	2
2. Deskripsi Umum.....	3
2.1 Prespektif Produk.....	3
2.2 Fungsi Produk	3
2.3 Karakteristik Pengguna.....	4
2.4 Batasan.....	4
2.5 Asumsi dan Ketergantungan	4
3. Analisis Kebutuhan Aplikasi.....	5
3.1 Analisis Kebutuhan Fungsional dan Non-Fungsional.....	5
4. Scenario Model Aplikasi.....	7
4.1 Use Case Scenario.....	7
4.2 Use Case Diagram.....	12
4.3 Activity Diagram.....	13
5. Analisis Sistem Aplikasi.....	15
5.1 Data Dictionary	15
6. Prediksi Biaya Aplikasi FitLife.....	19
6.1 Estimasi Biaya Pembuatan Awal.....	19
6.2 Estimasi Biaya Maintenance Bulanan	20
7. Rancangan Tampilan Sistem.....	21

1.Pendahuluan

1.1 Tujuan Penulisan Dokumen

Dokumen ini ditulis dalam rangka menjabarkan spesifikasi kebutuhan perangkat lunak yang akan digunakan untuk aplikasi **FitLife** yang sedang dikembangkan. Penjabaran spesifikasi perangkat lunak meliputi deskripsi umum sistem perangkat lunak, analisis kebutuhan aplikasi, skenario model penggunaan, analisis sistem dan desain, serta rancangan tampilan sistem. Perangkat lunak yang akan dibangun adalah aplikasi mobile untuk kebugaran dan gaya hidup sehat bernama **FitLife**. Dalam membangun aplikasi FitLife, kami memerlukan spesifikasi serta rincian kebutuhan pengguna yang tepat agar aplikasi yang dibangun tepat sasaran, inovatif, dan sesuai dengan kebutuhan pengguna untuk mencapai tujuan kebugaran mereka.

1.2 Lingkup Produk

FitLife adalah sebuah aplikasi kebugaran personal berbasis mobile yang didesain untuk membantu pengguna dalam memandu latihan, memantau progres untuk mencapai tujuan gaya hidup sehat mereka. Aplikasi FitLife memungkinkan pengguna untuk sign up, log in, mengelola profil biometrik (tinggi, berat, usia) yang terintegrasi dengan kalkulasi BMI, mencatat aktivitas latihan, mengelola asupan kalori harian, serta menyediakan platform sosial untuk berinteraksi (chat) dengan sesama teman. Fitur-fitur ini dirancang untuk membantu pengguna dalam memahami dan mengoptimalkan perjalanan kebugaran mereka.

Aplikasi ini dibangun menggunakan framework Flutter dengan bahasa pemrograman Dart, yang memungkinkan pengembangan lintas platform (Android dan iOS) dari satu basis kode. Desain antarmuka (UI/UX) diaplikasikan secara intuitif, navigasi yang mudah diikuti, dan didominasi oleh gradasi warna merah ke hitam yang menarik dan energik. Sistem diawasi oleh kami yang bertanggung jawab untuk mengembangkan fitur-fitur yang diperlukan pengguna dalam mencapai target kebugaran. Kemudian, data-data yang dimasukkan oleh pengguna akan dilindungi kerahasiaan dan integritasnya, serta performa aplikasi akan selalu dijaga agar stabil dan responsif.

2. Deskripsi Umum

2.1 Prespektif Produk

FitLife adalah produk aplikasi mobile standalone yang berfungsi sebagai asisten kebugaran pribadi di perangkat pengguna. Aplikasi ini akan berinteraksi secara aktif dengan layanan backend (API dan database PostgreSQL/Firebase) untuk mengelola data pengguna, riwayat aktivitas, dan informasi latihan. FitLife juga akan memanfaatkan fungsionalitas seperti penyimpanan lokal untuk data cache dan preferensi.

2.2 Fungsi Produk

Aplikasi FitLife menyediakan serangkaian fungsi inti yang terbagi menjadi beberapa area utama:

- **Autentikasi & Manajemen Profil:** Memungkinkan pengguna untuk membuat dan mengelola akun pribadi, termasuk pendaftaran/login via email dan kata sandi, manajemen *username*, serta kemampuan untuk mengubah kata sandi dan mengunggah/mengganti foto profil.
- **Data Biometrik & Kesehatan:** Pengguna dapat memasukkan data tinggi, berat, dan usia, yang kemudian digunakan sistem untuk secara otomatis menghitung dan mengklasifikasikan Indeks Massa Tubuh (BMI) mereka. Aplikasi juga memungkinkan input kalori yang dikonsumsi harian dan menampilkan ringkasan kalori terbakar.
- **Panduan & Pelacakan Latihan:**
 - **Daftar Gerakan Latihan:** Menyediakan katalog gerakan latihan yang dibagi menjadi dua kategori utama: **Kardio (tanpa alat)** dan **Olahraga dengan Alat** (misalnya, *push-up*, *tricep extension*, *squat*). Setiap gerakan dilengkapi deskripsi dan panduan.
 - **Fitur "Free Run":** Mengestimasi kalori yang terbakar dari inputan user selama aktivitas lari.
 - **Pelacakan Latihan Umum:** Pengguna dapat memilih gerakan dari daftar, mencatat set, dan menyimpan aktivitas mereka ke riwayat.
- **Ringkasan Progres Harian:** Menawarkan "Recap" harian yang merangkum semua aktivitas latihan dan kalori, dengan opsi untuk menyimpan sebagai foto dan membagikannya.
- **Jaringan Sosial :** Membangun fitur sosial yang memungkinkan pengguna untuk berkomunikasi antar pengguna dan mengelola daftar pertemanan (menambah teman, menerima/menolak permintaan pertemanan).
- **Pengaturan Aplikasi:** Menyediakan kontrol bagi pengguna untuk mengelola informasi profil dan preferensi aplikasi.

2.3 Karakteristik Pengguna

- **Nama Peran:** Pengguna FitLife (FitLife User)
- **Deskripsi:** Individu dari berbagai tingkat kebugaran yang mencari alat bantu untuk memandu latihan, memantau progres, dan mencari motivasi sosial. Mereka mungkin adalah pemula yang membutuhkan panduan dasar gerakan, atau pengguna menengah yang ingin melacak aktivitas mereka secara detail.
- **Keahlian:** Memiliki pemahaman dasar tentang penggunaan aplikasi *smartphone* dan navigasi.
- **Tanggung Jawab:** Menyediakan data personal yang akurat, mencatat aktivitas latihan secara konsisten, dan berinteraksi secara etis dalam komunitas.

2.4 Batasan

- Aplikasi ini memerlukan koneksi internet stabil untuk fitur-fitur seperti login, sinkronisasi data, komunikasi API, dan fungsi sosial.
- Aplikasi tidak menyediakan layanan pelatih kebugaran profesional secara langsung, melainkan berfungsi sebagai panduan dan pencatat aktivitas.
- Aplikasi tidak akan mendukung penggunaan offline penuh untuk semua fitur, terutama yang membutuhkan data real-time dari server.

2.5 Asumsi dan Ketergantungan

- Pengguna memiliki *smartphone* dengan versi sistem operasi Android (min. [versi 6]).
- *Backend* server, API, dan *database* (PostgreSQL, Firebase) akan berfungsi secara stabil dan tersedia.
- Pengguna memiliki pemahaman dasar tentang terminologi kebugaran dan cara melakukan gerakan latihan.
- Ketersediaan *library* dan *tools* pengembangan (Flutter/Dart, Provider/Riverpod, Dio, go_router, shared_preferences) yang digunakan.

3. Analisis Kebutuhan Aplikasi

3.1 Analisis Kebutuhan Fungsional dan Non-Fungsional

Kebutuhan Fungsional :

No	Kebutuhan Fungsional	Deskripsi
1	Autentikasi & Profil Pengguna	Pengguna dapat mendaftar (email, sandi, username), login, mengubah kata sandi, mengunggah/mengganti foto profil, serta mengisi data biometrik (tinggi, berat, usia) untuk perhitungan BMI otomatis.
2	Manajemen Kalori & Aktivitas Harian	Pengguna dapat melihat ringkasan kalori harian (terbakar vs. konsumsi), menginput kalori yang dikonsumsi, melihat <i>My Plan</i> dan <i>Recent Activity</i> , serta menggunakan fitur "Recap" harian untuk menyimpan/berbagi rangkuman aktivitas.
3	Pelacakan Latihan & Aktivitas	Pengguna dapat menggunakan fitur "Free Run", memilih dan mencatat penyelesaian berbagai jenis gerakan latihan (kardio, dengan alat), serta melacak set, repetisi, dan durasi yang dilakukan.
4	Fitur Sosial	Pengguna dapat menerima permintaan pertemanan, dan berkomunikasi dengan teman.
5	Pengaturan Aplikasi	Pengguna dapat mengakses dan mengubah pengaturan terkait profil, akun (password), dan keluar dari akun (<i>logout</i>).

Kebutuhan Non-Fungsional :

No	Kebutuhan Non-Fungsional	Deskripsi
1	Antarmuka Aplikasi (UI/UX)	Aplikasi memiliki antarmuka yang intuitif dan mudah digunakan dengan desain konsisten bergradasi merah ke hitam. Navigasi sederhana dan elemen visual yang jelas mempermudah pengguna.
2	Kinerja	Aplikasi memiliki kinerja yang responsif dan cepat terhadap tindakan pengguna, dengan waktu muat halaman dan respons API yang minimal.
3	Keamanan Data	Data pribadi pengguna (password, biometrik) dan aktivitas harus dilindungi dengan enkripsi saat penyimpanan dan transmisi. Aplikasi akan meminta izin yang relevan dan menjaga kerahasiaan data.
4	Keandalan	Aplikasi harus stabil, memiliki <i>uptime</i> tinggi, dan mampu memulihkan diri dari <i>error</i> tanpa kehilangan data penting. Data pengguna akan dicadangkan secara teratur.
5	Kompatibilitas	Aplikasi harus dapat berjalan dengan lancar di berbagai perangkat <i>smartphone</i> dengan sistem operasi Android versi yang kompatibel, serta beradaptasi dengan berbagai ukuran layar.

4. Scenario Model Aplikasi

4.1 Use Case Scenario

Skenario 1: Registrasi Akun dan Onboarding Profil Pengguna Baru

- **Aktor:** Pengguna Baru
- **Deskripsi:** Pengguna yang belum memiliki akun ingin mendaftar dan menyelesaikan pengaturan profil awal untuk dapat menggunakan aplikasi FitLife.
- **Proses:**
 1. Pengguna membuka aplikasi FitLife.
 2. Pengguna memilih opsi "Sign Up" atau "Daftar".
 3. Aplikasi menampilkan formulir pendaftaran akun awal.
 4. Pengguna memasukkan **alamat email** yang valid dan **membuat kata sandi**.
 5. Pengguna menekan tombol "Daftar" atau "Lanjutkan".
 6. Sistem melakukan validasi data (format email, kekuatan sandi).
 7. Jika data valid, aplikasi mengarahkan ke halaman input data biometrik.
 8. Pengguna memasukkan **berat badan, tinggi badan, gender, dan tanggal lahir**.
 9. Aplikasi secara otomatis menghitung dan mengklasifikasikan BMI berdasarkan input tersebut.
 10. Pengguna menekan tombol "Lanjutkan".
 11. Aplikasi mengarahkan ke halaman pembuatan Bio dan pemasangan Foto Profil.
 12. Pengguna dapat **menulis bio singkat** (opsional) dan/atau **mengunggah foto profil** dari galeri (opsional).
 13. Pengguna menekan tombol "Selesai" atau "Lanjutkan".
 14. Jika semua langkah selesai, sistem membuat akun dan menyimpan data profil.
 15. Aplikasi menampilkan pesan konfirmasi dan mengarahkan pengguna ke **Home Page**.
 16. *Skenario Alternatif:* Jika data yang diinput tidak valid, aplikasi menampilkan pesan *error* dan meminta koreksi.

Skenario 2: Mencatat Aktivitas Lari Manual di "Free Run"

- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin mencatat aktivitas lari mereka secara manual dengan memasukkan jarak dan waktu tempuh, lalu sistem akan mengkalkulasi kalori terbakar.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menavigasi ke menu "Free Run".
 3. Aplikasi menampilkan formulir untuk input lari manual.
 4. Pengguna memasukkan **jarak tempuh** (misal: 5 km) dan **waktu tempuh** (misal: 30 menit).
 5. Pengguna menekan tombol "Kalkulasi" atau "Simpan".
 6. Sistem secara otomatis **mengkalkulasi estimasi kalori yang terbakar** berdasarkan jarak, waktu, dan data biometrik pengguna (berat badan, gender, usia).
 7. Aplikasi menampilkan hasil kalkulasi kalori terbakar.
 8. Pengguna mengkonfirmasi penyimpanan aktivitas.
 9. Sistem menyimpan aktivitas lari ke riwayat pengguna dan memperbarui total kalori terbakar di *Home Page*.
 10. Aplikasi mengarahkan pengguna kembali ke *Home Page*.

Skenario 3: Memilih Latihan dan Menambahkannya ke Plan

- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin memilih jenis latihan dari daftar yang tersedia, melihat detailnya, dan menambahkannya ke rencana latihan mereka.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menavigasi ke menu "Start/Activity".
 3. Aplikasi menampilkan dua kategori latihan: "Cardio" dan "Weight Training".
 4. Pengguna memilih salah satu kategori, misal "Weight Training".
 5. Aplikasi menampilkan daftar gerakan di bawah kategori tersebut (misal: Bench Press, Squat, Deadlift, dll.).
 6. Pengguna menekan salah satu gerakan, misal "Bench Press".

7. Aplikasi menampilkan halaman detail gerakan "Bench Press" yang berisi:
 - **Deskripsi** gerakan.
 - **Manfaat** gerakan.
 - **Estimasi kalori yang terbakar** per set.
8. Pengguna menekan tombol "Add to Plan".
9. Sistem menambahkan latihan "Bench Press" ke "My Plan" pengguna yang akan terlihat di *Home Page*.
10. Aplikasi menampilkan konfirmasi bahwa latihan telah ditambahkan.
11. Aplikasi mengarahkan pengguna kembali ke daftar latihan atau *Home Page*.

Skenario 4: Menambah Teman dan Chatting di Social Page

- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin menambah teman baru menggunakan kode unik dan kemudian berkomunikasi melalui fitur *chat*.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menavigasi ke "Social Page".
 3. Aplikasi menampilkan daftar teman dan tombol "Add a Friend".
 4. Pengguna menekan tombol "Add a Friend".
 5. Aplikasi menampilkan input field untuk memasukkan kode unik teman.
 6. Pengguna memasukkan **kode unik** teman yang ingin ditambahkan.
 7. Pengguna menekan tombol "Kirim Permintaan".
 8. Sistem mengirim permintaan pertemanan ke pengguna dengan kode unik tersebut.
 9. **Pengguna Target (teman):** Menerima notifikasi permintaan pertemanan.
 10. **Pengguna Target:** Masuk ke Social Page, melihat permintaan, dan memilih "Terima" atau "Tolak".
 11. Jika permintaan diterima, kedua pengguna ditambahkan ke *Friend List* masing-masing.
 12. Pengguna kemudian mengklik nama teman di *Friend List* mereka.
 13. Aplikasi membuka antarmuka *chat* dengan teman tersebut.
 14. Pengguna dapat mengetik pesan dan mengirimkannya.

15. **Skenario Alternatif:** Jika kode unik tidak ditemukan, aplikasi menampilkan pesan *error*. Jika permintaan ditolak, status pertemanan tidak berubah.

Skenario 5: Mengedit Profil Melalui Settings Page

- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin mengubah informasi profil dasar mereka, seperti foto profil, bio, dan tanggal lahir.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menavigasi ke "Settings Page".
 3. Aplikasi menampilkan dua kategori pengaturan: "Edit Profile" dan "Account".
 4. Pengguna memilih "Edit Profile".
 5. Aplikasi menampilkan halaman edit profil dengan opsi untuk:
 - Mengubah **Foto Profil** (unggah dari galeri).
 - Mengubah **Bio**.
 - Mengubah **Tanggal Lahir**.
 6. Pengguna melakukan perubahan yang diinginkan.
 7. Pengguna menekan tombol "Simpan Perubahan".
 8. Sistem menyimpan pembaruan profil ke *database*.
 9. Aplikasi menampilkan konfirmasi bahwa profil telah diperbarui dan mengarahkan kembali ke halaman pengaturan atau *Home Page*.

Skenario 6: Mengubah Kata Sandi Melalui Settings Page

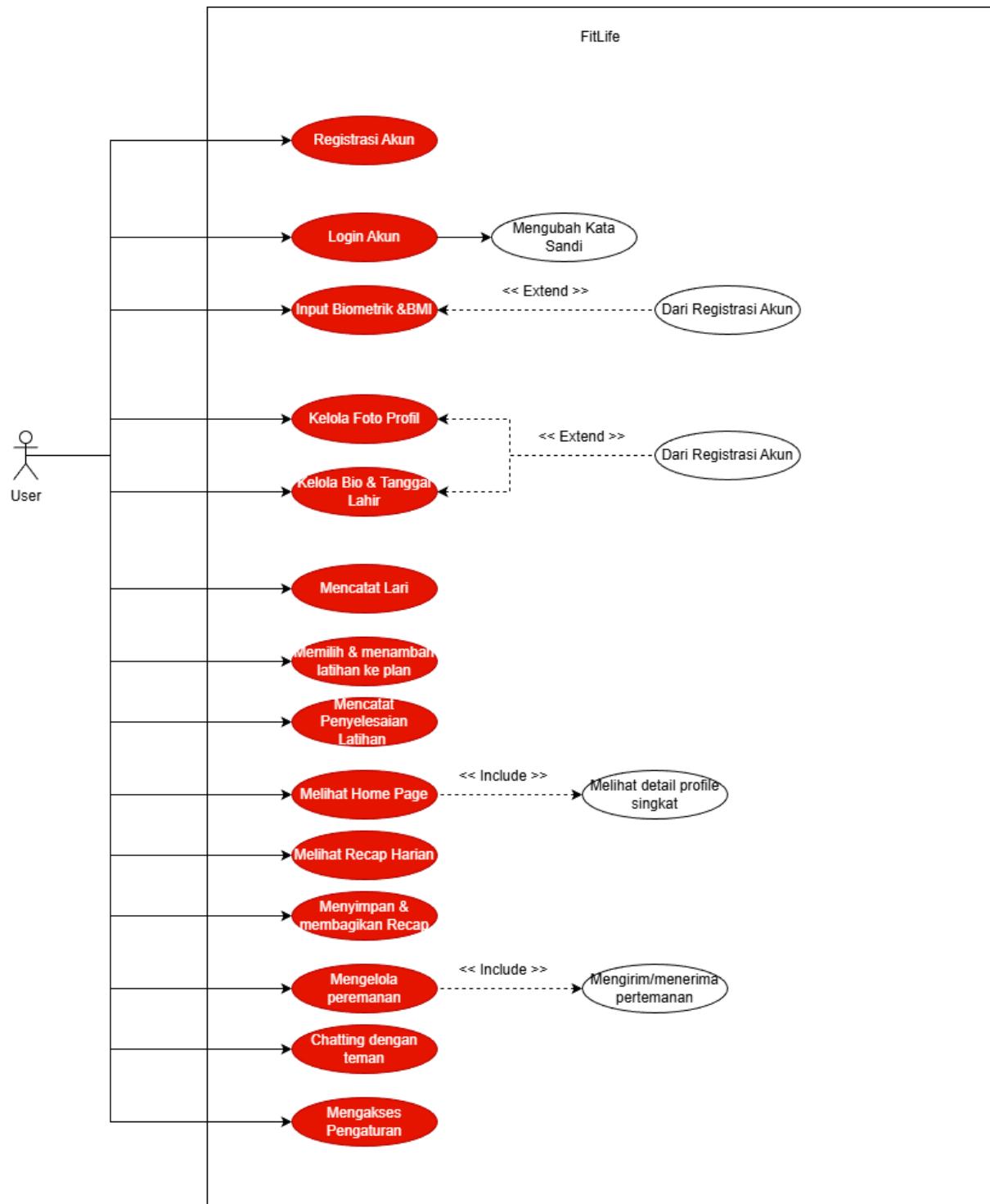
- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin mengubah kata sandi akun mereka.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menavigasi ke "Settings Page".
 3. Aplikasi menampilkan dua kategori pengaturan: "Edit Profile" dan "Account".
 4. Pengguna memilih "Account".
 5. Aplikasi menampilkan opsi "Ubah Kata Sandi".
 6. Pengguna menekan "Ubah Kata Sandi".

7. Aplikasi menampilkan formulir untuk mengubah kata sandi (input Kata Sandi Lama, Kata Sandi Baru, Konfirmasi Kata Sandi Baru).
8. Pengguna memasukkan kata sandi yang diperlukan.
9. Pengguna menekan tombol "Simpan Kata Sandi".
10. Sistem memvalidasi Kata Sandi Lama dan menyimpan Kata Sandi Baru yang terenkripsi.
11. Aplikasi menampilkan konfirmasi bahwa kata sandi telah diubah.
12. *Skenario Alternatif:* Jika Kata Sandi Lama salah atau Kata Sandi Baru tidak memenuhi kriteria, aplikasi menampilkan pesan *error*.

Skenario 7: Melihat dan Membagikan "Recap" Harian

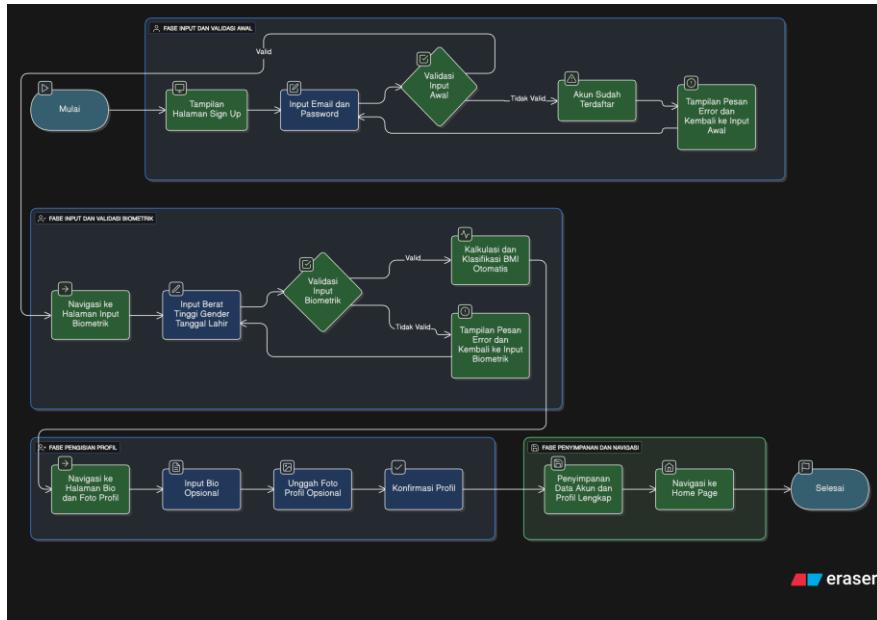
- **Aktor:** Pengguna Terdaftar
- **Deskripsi:** Pengguna ingin melihat rangkuman komprehensif dari semua aktivitas yang telah dilakukan pada hari tersebut, termasuk latihan dari plan, dan total kalori yang terbakar, kemudian membagikannya.
- **Proses:**
 1. Pengguna sudah *login* dan berada di *Home Page*.
 2. Pengguna menekan tombol "Recap" yang ada di *Home Page*.
 3. Sistem secara otomatis **mengumpulkan semua data aktivitas** yang tercatat untuk hari tersebut, termasuk aktivitas "Free Run" dan semua latihan yang diselesaikan yang merupakan bagian dari "My Plan" pengguna.
 4. Sistem kemudian **menghitung total akumulasi kalori yang terbakar** dari seluruh aktivitas tersebut.
 5. Aplikasi menampilkan halaman "Recap Harian" yang menyajikan rangkuman visual dari aktivitas yang dilakukan dan total kalori terbakar.
 6. Pengguna memiliki opsi untuk menekan tombol "Simpan sebagai Foto".
 7. Aplikasi menyimpan gambar Recap ke galeri perangkat pengguna.
 8. Pengguna juga memiliki opsi untuk menekan tombol "Bagikan".
 9. Aplikasi membuka dialog berbagi perangkat, memungkinkan pengguna memilih platform (misalnya, WhatsApp, Instagram, dll.) untuk berbagi foto Recap tersebut.
 10. *Skenario Alternatif:* Jika tidak ada aktivitas yang tercatat untuk hari itu, Recap akan menampilkan pesan bahwa "Tidak ada aktivitas yang dilakukan hari ini."

4.2 Use Case Diagram

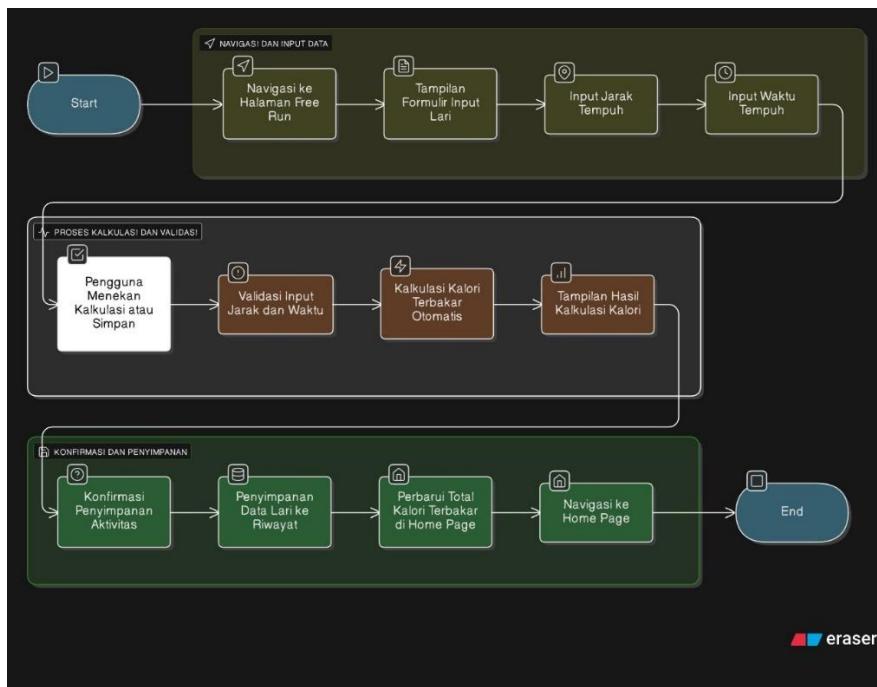


4.3 Activity Diagram

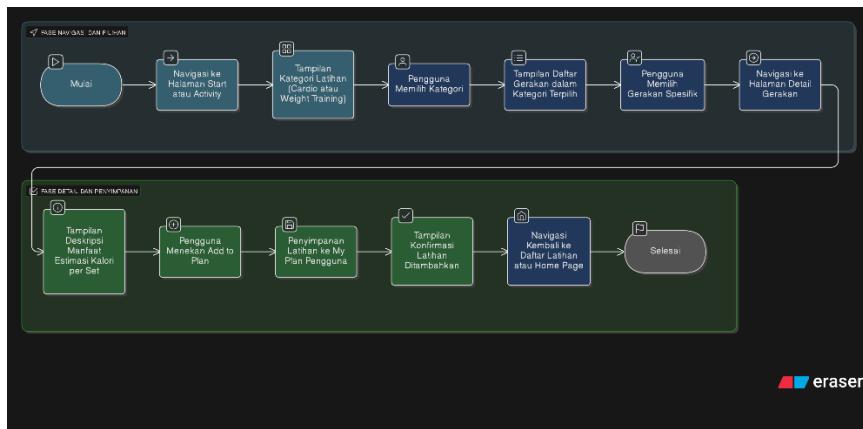
Alur Registrasi Akun :



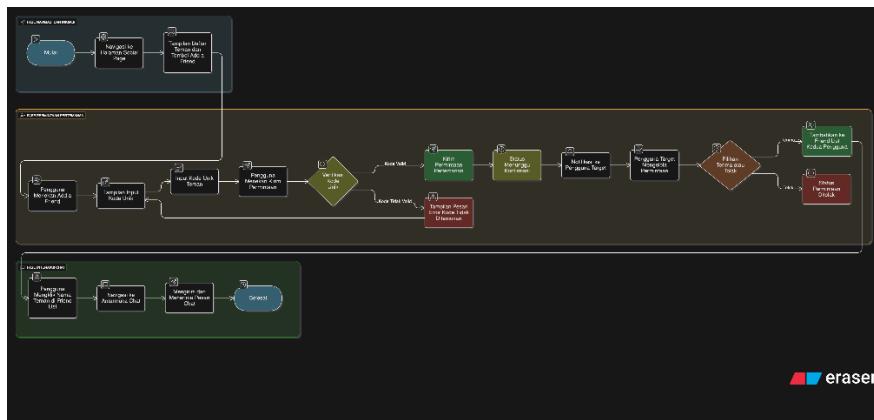
Alur Pencatatan FreeRun :



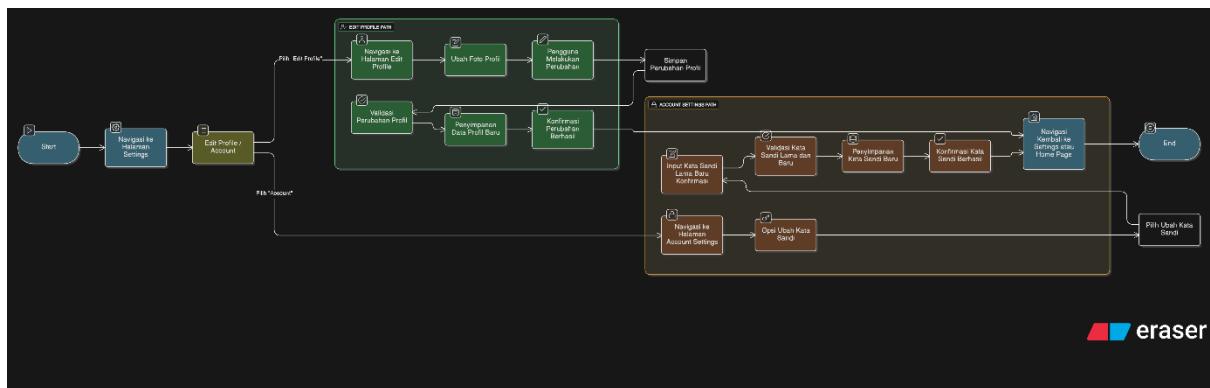
Alur Pemilihan Latihan dan Penambahan ke Plan :



Alur Menambah Teman dan Chatting :



Alur Pengaturan Profil & Akun :



5. Analisis Sistem Aplikasi

5.1 Data Dictionary

1. Tabel Users (Informasi Akun & Profil Pengguna)

Primary Key: userId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	userId	VARCHAR(50)	ID unik pengguna dari Firebase Authentication	NOT NULL, PRIMARY KEY
2	email	VARCHAR(100)	Email pengguna untuk login	NOT NULL, UNIQUE
3	username	VARCHAR(50)	Nama pengguna untuk identitas sosial	NOT NULL
4	shortId	VARCHAR(20)	Kode unik singkat untuk fitur pertemanan	UNIQUE
5	weight	FLOAT	Berat badan pengguna (kg)	NULLABLE
6	height	FLOAT	Tinggi badan pengguna (cm)	NULLABLE
7	gender	VARCHAR(10)	Jenis kelamin pengguna	NULLABLE
8	dateOfBirth	DATE	Tanggal lahir pengguna	NULLABLE
9	bio	TEXT	Deskripsi singkat profil pengguna	NULLABLE
10	profilePicturePath	VARCHAR(255)	Path foto profil di Firebase Storage	NULLABLE
11	friends	ARRAY	Daftar userId teman	NULLABLE
12	lastActivityDate	TIMESTAMP	Tanggal aktivitas terakhir pengguna	NULLABLE
13	netDailyCalorieGoal	INT	Target bersih kalori harian pengguna	NULLABLE

2. Tabel Workout Plan (Rencana Latihan Pengguna)

Primary Key: planId Foreign Key: userId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	planId	VARCHAR(50)	ID unik latihan dalam plan	NOT NULL, PRIMARY KEY
2	userId	VARCHAR(50)	ID pengguna pemilik plan	NOT NULL, FOREIGN KEY
3	workoutName	VARCHAR(100)	Nama gerakan latihan	NOT NULL
4	sets	INT	Jumlah set latihan	NOT NULL
5	reps	VARCHAR(20)	Jumlah repetisi per set	NOT NULL
6	estimatedCalories	INT	Estimasi kalori terbakar	NULLABLE
7	createdAt	TIMESTAMP	Waktu latihan ditambahkan	NOT NULL

3. Tabel Riwayat Aktivitas (Recent Activity & Free Run)

Primary Key: activityId Foreign Key: userId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	activityId	VARCHAR(50)	ID unik aktivitas	NOT NULL, PRIMARY KEY
2	userId	VARCHAR(50)	ID pengguna	NOT NULL, FOREIGN KEY
3	activityType	VARCHAR(50)	Jenis aktivitas (Free Run / Workout)	NOT NULL
4	durationMinutes	FLOAT	Durasi aktivitas (menit)	NULLABLE

No	Nama Field	Tipe Data	Keterangan	Constraint
5	distanceKm	FLOAT	Jarak lari (km)	NULLABLE
6	caloriesBurned	INT	Kalori terbakar	NULLABLE
7	timestamp	TIMESTAMP	Waktu aktivitas dicatat	NOT NULL

4. Tabel Friend Request (Permintaan Pertemanan)

Primary Key: requestId

Foreign Key: senderId, recipientId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	requestId	VARCHAR(50)	ID permintaan pertemanan	NOT NULL, PRIMARY KEY
2	senderId	VARCHAR(50)	ID pengirim permintaan	NOT NULL
3	senderUsername	VARCHAR(50)	Username pengirim	NOT NULL
4	status	VARCHAR(20)	Status (pending / accepted / rejected)	NOT NULL
5	timestamp	TIMESTAMP	Waktu permintaan dikirim	NOT NULL

5. Tabel Chat Room

Primary Key: chatRoomId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	chatRoomId	VARCHAR(100)	ID chat (gabungan 2 userId)	NOT NULL, PRIMARY KEY
2	userId1	VARCHAR(50)	ID pengguna pertama	NOT NULL
3	userId2	VARCHAR(50)	ID pengguna kedua	NOT NULL

6. Tabel Messages (Pesanan Chat)

Primary Key: messageId

Foreign Key: chatRoomId

No	Nama Field	Tipe Data	Keterangan	Constraint
1	messageId	VARCHAR(50)	ID unik pesan	NOT NULL, PRIMARY KEY
2	chatRoomId	VARCHAR(100)	ID ruang chat	NOT NULL, FOREIGN KEY
3	senderId	VARCHAR(50)	ID pengirim pesan	NOT NULL
4	content	TEXT	Isi pesan chat	NOT NULL
5	timestamp	TIMESTAMP	Waktu pesan dikirim	NOT NULL

6. Prediksi Biaya Aplikasi FitLife

6.1 Estimasi Biaya Pembuatan Awal

Biaya ini mencakup semua proses dari perencanaan hingga peluncuran versi pertama aplikasi (*Minimum Viable Product - MVP*).

Komponen Biaya	Deskripsi Kebutuhan FitLife	Estimasi Waktu	Estimasi Biaya (IDR)
1. Riset & Perencanaan	Analisis kebutuhan, pembuatan <i>wireframe</i> dan <i>user flow</i> .	2 - 4 Minggu	Rp 10.000.000 - Rp 25.000.000
2. Desain UI/UX	Desain antarmuka 4 halaman utama, <i>onboarding</i> , dan aset visual yang menarik.	4 - 6 Minggu	Rp 20.000.000 - Rp 45.000.000
3. Pengembangan Frontend	Pembuatan aplikasi mobile (<i>cross-platform</i> seperti Flutter/React Native) untuk implementasi semua fitur <i>user-facing</i> (UI, logika <i>client-side</i>).	12 - 16 Minggu	Rp 60.000.000 - Rp 180.000.000
4. Pengembangan Backend	Pembuatan API, <i>database</i> untuk menyimpan data pengguna, riwayat latihan, postingan sosial, dan fitur <i>real-time</i> (chat/notifikasi).	10 - 14 Minggu	Rp 50.000.000 - Rp 150.000.000
5. Testing & QA	Pengujian fungsional, performa, dan keamanan, terutama pada fitur BMI, kalkulasi kalori, dan sosial.	4 - 6 Minggu	Rp 15.000.000 - Rp 40.000.000
6. Manajemen Proyek	Koordinasi tim dan memastikan proyek berjalan sesuai jadwal.	Sepanjang Proyek	Rp 10.000.000 - Rp 25.000.000
Total Estimasi Pembuatan Awal	(Total Durasi: 5-7 bulan)		Rp 165.000.000 - Rp 465.000.000

6.2 Estimasi Biaya Maintenance Bulanan

Biaya ini diperlukan untuk menjaga aplikasi tetap berjalan, aman, dan *up-to-date*.

Komponen Biaya	Deskripsi	Estimasi Biaya Minimal (IDR)/Bulan	Estimasi Biaya Ideal (IDR)/Bulan
1. Server / Hosting	Biaya infrastruktur <i>backend</i> (API, Database) di layanan <i>cloud</i> (AWS/GCP/Firebase). Diperlukan untuk menyimpan data pengguna, <i>feed</i> sosial, dan <i>chat</i> .	Rp 500.000	Rp 5.000.000
2. Lisensi Platform	Biaya Google Play Developer (\$25 sekali bayar), dirata-ratakan.	Rp 150.000	Rp 300.000
3. API Pihak Ketiga	Biaya layanan <i>push notification</i> (jika menggunakan layanan berbayar) atau <i>library</i> tertentu.	Rp 0	Rp 1.000.000
4. Biaya Tenaga Kerja (<i>Maintenance & Bug Fix</i>)	Alokasi waktu <i>developer</i> paruh waktu/kontrak untuk perbaikan <i>bug</i> , pembaruan OS, dan keamanan. <i>Rata-rata 15-20% dari total biaya pengembangan</i> .	Rp 15.000.000	Rp 35.000.000
Total Estimasi Maintenance Bulanan		Rp 15.650.000	Rp 41.300.000

Kesimpulan Prediksi Biaya

Kategori Biaya	Estimasi Rendah (IDR)	Estimasi Tinggi (IDR)
Total Biaya Pembuatan Awal	Rp 165.000.000	Rp 465.000.000
Total Biaya Maintenance Bulanan	Rp 15.650.000	Rp 41.300.000

7. Rancangan Tampilan Sistem

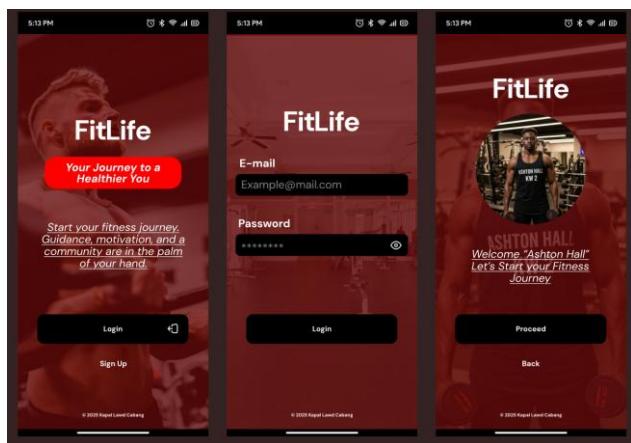
Link Figma Rancangan Interface :

[Figma FitLife](#)

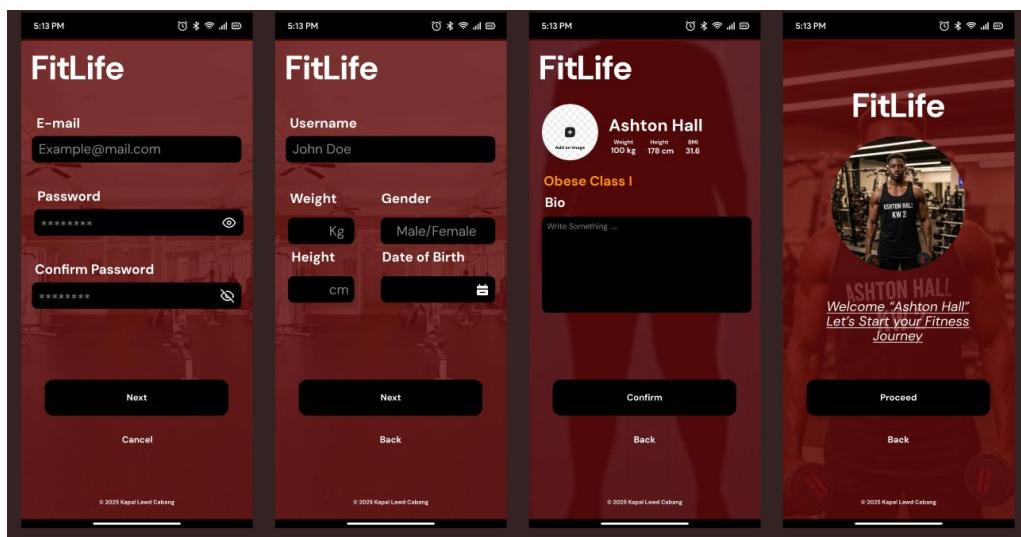
Splash Screen :



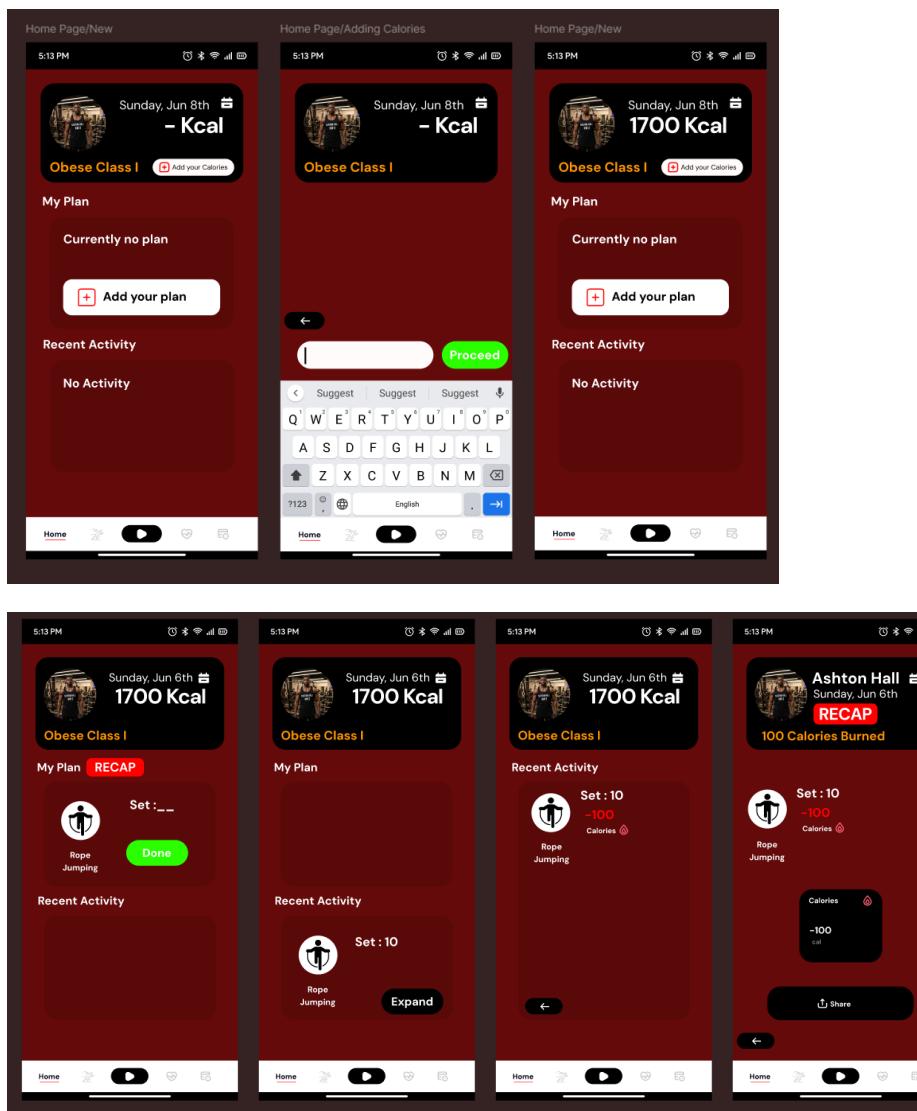
Halaman Login dan Register :



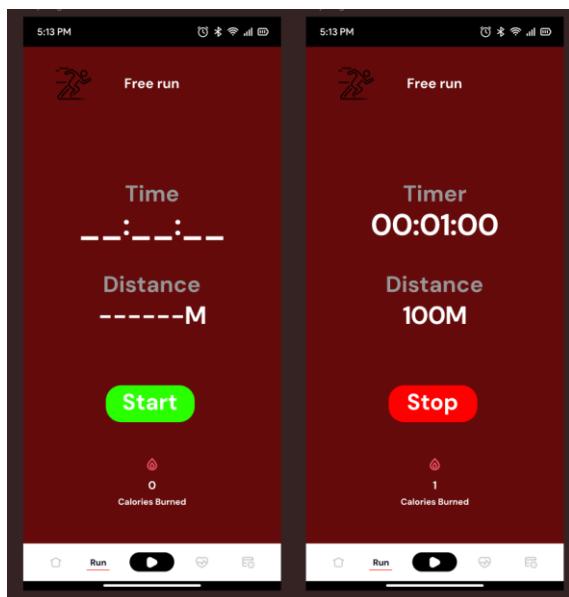
Halaman Register :



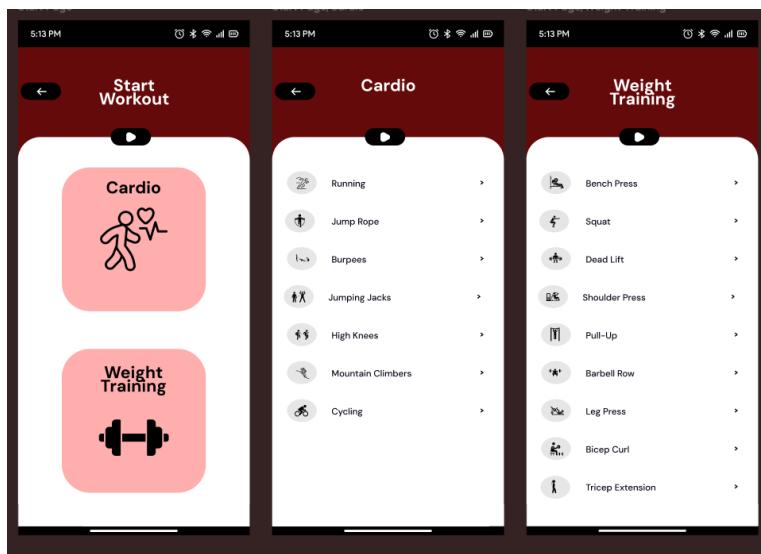
Halaman Home :



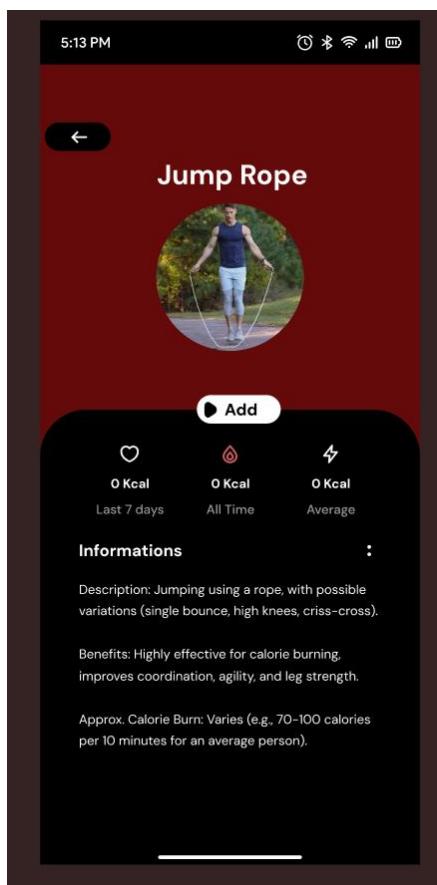
Halaman Free Run :



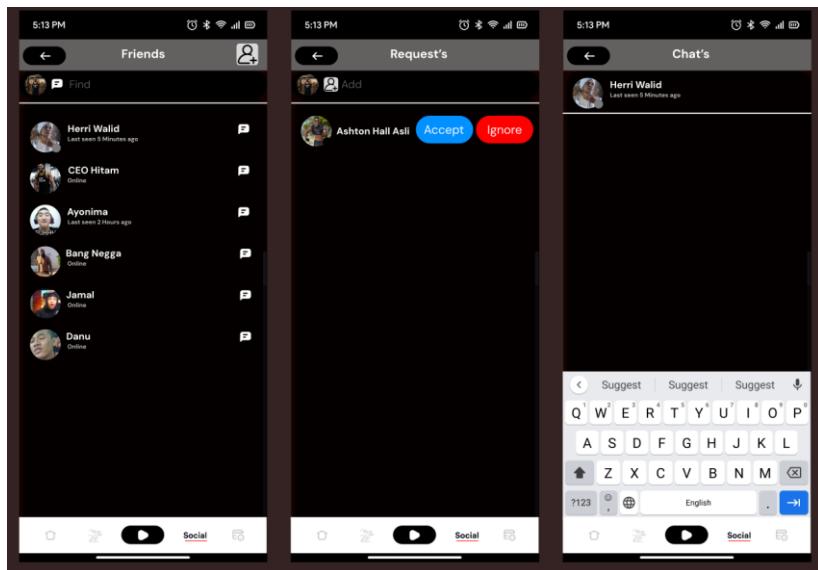
Halaman Start/Activity :



Halaman Sub Menu Start/Activity :



Halaman Social :



Halaman Settings :

