

**APLIKASI CHATTING DENGAN FITUR THREADING
BERBASIS SISTEM OPERASI ANDROID**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika**



Disusun Oleh :

Nama : Achmad Naufal Syafiq

No. Mahasiswa : 12523270

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2016

**APLIKASI CHATTING DENGAN FITUR THREADING
BERBASIS SISTEM OPERASI ANDROID**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika**



Disusun Oleh :

Nama : Achmad Naufal Syafiq

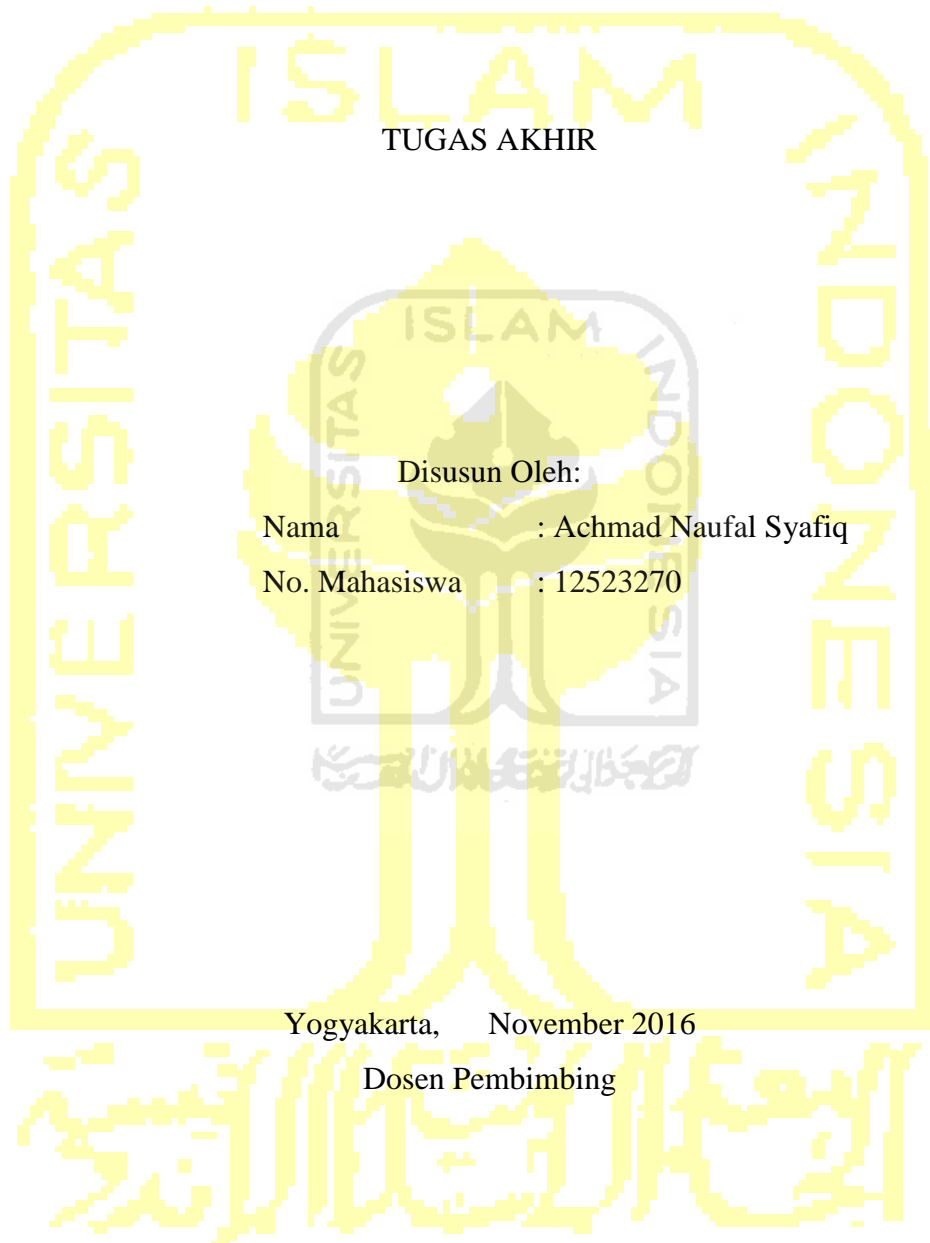
No. Mahasiswa : 12523270

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2016

LEMBAR PENGESAHAN PEMBIMBING

**APLIKASI CHATTING DENGAN FITUR THREADING
BERBASIS SISTEM OPERASI ANDROID**



Dr. Raden Teduh Dirgahayu, ST., MSc.

LEMBAR PENGESAHAN PENGUJI**APLIKASI CHATTING DENGAN FITUR THREADING BERBASIS
SISTEM OPERASI ANDROID**

TUGAS AKHIR

Disusun Oleh:

Nama : Achmad Naufal Syafiq

No. Mahasiswa : 12523270

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat untuk
Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas Teknologi
Industri Universitas Islam Indonesia
Yogyakarta, Desember 2016

Tim Penguji,

Dr. Raden Teduh Dirgahayu, ST., MSc.

Ketua

Beni Suranto S.T., M.Soft.Eng.

Anggota I

Almed Hamzah S.T., M.Eng.

Anggota II

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia

(Hendrik, S.T., M. Eng.)

LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertanda tangan dibawah ini:

Nama : Achmad Naufal Syafiq

No. Mahasiswa : 12523270

Tugas Akhir dengan judul:

APLIKASI CHATTING DENGAN FITUR THREADING BERBASIS SISTEM OPERASI ANDROID

Menyatakan dengan sesungguhnya bahwa tugas akhir ini adalah karya saya sendiri dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi manapun.

Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, November 2016



Achmad Naufal Syafiq

HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Sembah sujud serta syukur kehadiran Allah SWT. Taburan cinta dan rahmat-Mu yang telah memberikan kekuatan dan membekali dengan ilmu. Sholawat beserta salam senantiasa kami panjatkan kepada junjungan kita Nabi Muhammad SAW sebagai pemberi syafaat bagi seluruh umat manusia.

Kepada orang tua tercinta, Bapak Amin Musa serta Ibunda tersayang Ai Kurniasari, yang senantiasa memberikan semangat, nasihat serta doa dan kasih sayang yang tiada henti penulis persembahkan tugas akhir ini. Penulis menyadari bahwa hal ini saja tidak cukup untuk membalas budi dan kasih sayang yang orang tua tercinta yang sudah diberikan sejak lahir. Semoga dengan pencapaian dari penulis ini dapat membuat bangga dan bahagia kepada kedua orang tua tercinta.

Kepada adik – adik tersayang, A Faris Akbar, M Hafizh Azar, M Hanif Ismail, Haifa Nurul I, dan Hasna Nur F yang selalu menjadi motivasi bagi kaka untuk menjadi individu yang lebih baik. Semoga adik – adik dapat melampaui apa yang sudah kaka capai sehingga bisa menjadi kebanggaan bagi kedua orang tua. Amin

Ya Rabbal Alamin.

MOTTO

“Dan tolong – menolonglah kamu dalam (mengerjakan) kebaikan dan takwa, dan jangan tolong menolong dalam perbuatan dosa dan pelanggaran. Dan bertakwalah kamu kepada Allah, sesungguhnya Allah amat berat siksa-Nya”

(QS. Al-Maidah’5:2)

“Dua hal apabila dimiliki oleh seseorang dia dicatat oleh Allah sebagai orang yang bersyukur dan sabar. Dalam urusan agama (ilmu dan ibadah) dia melihat kepada yang lebih tinggi lalu meniru dan mencontohnya. Dalam urusan dunia dia melihat kepada yang lebih bawah, lalu bersyukur kepada Allah bahwa dia masih diberi kelebihan”

(HR. Thirmizi)

“Gantungkan cita citamu setinggi langit! Bermimpilah setinggi langit. Jika engkau jatuh, engkau akan jatuh diantara bintang bintang”

(Ir. Soekarno)

“Hiduplah seperti engkau akan mati besok. Belajarlah seakan kau hidup selamanya”

(Mahatma Gandhi)

“If you can’t fly, then run, if you can’t run, then walk, if you can’t walk then crawl, but whatever you do, you have to keep moving forward”

(Martin Luther King Jr)

KATA PENGANTAR



Assalamu'alaikum warahmatullahi wabarakatuh

Alhamdulillah Robbil ‘Alamin, puji syukur kehadiran Allah SWT yang senantiasa melimpahkan rahmat dan karunia-Nya. Shalawat serta salam senantiasa kita junjungkan kepada Nabi Muhammad SAW beserta keluarga dan para sahabat sehingga penulis dapat menyelesaikan laporan tugas akhir yang berjudul “Aplikasi Chatting Dengan Fitur Threading”

Laporan ini disusun sebagai salah satu syarat yang harus ditempuh dalam rangka menyelesaikan pendidikan pada jenjang Strata 1 di jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia. Laporan ini dapat terselesaikan berkat bantuan, dukungan dan bimbingan yang diberikan dari berbagai pihak, maka dari itu izinkan penulis mengucapkan terimakasih kepada :

1. Kedua orang tua penulis, Bapak Amin Musa dan Ibu Ai Kurniasari serta segenap keluarga yang senantiasa mendoakan serta memberikan dukungan baik secara moral maupun material selama masa studi di Universitas Islam Indonesia.
2. Kepada Adik – adik penulis yang selalu menjadi motivasi untuk dapat menyelesaikan tugas akhir ini dengan sebaik baiknya.
3. Bapak Dr. Ir. Harsoyo, M.Sc. Selaku Rektor Universitas Islam Indonesia.
4. Bapak Hendrik, S.T, M. Eng. Selaku ketua jurusan yang telah memberikan kesempatan kepada penulis dalam menyelesaikan tugas akhir.
5. Bapak Dr. Raden Teduh Dirgahayu, ST., M.Sc., Ph.D. selaku dosen pembimbing tugas akhir yang telah memberikan ilmu serta wawasan dan memberikan waktunya kepada penulis demi membimbing menyelesaikan tugas akhir.

6. Bapak Hanson Prihantoro Putro S.T., M.T., selaku dosen pembimbing akademik yang telah membimbing penulis sepanjang perjalanan menuntut ilmu di Jurusan Teknik Informatika, Universitas Islam Indonesia.
7. Bapak dan ibu dosen Jurusan Teknik Informatika yang telah memberikan ilmunya kepada penulis.
8. Teman – teman Gravity, Jurusan Teknik Informatika sebagai teman seperjuangan selama menempuh masa pendidikan Strata 1 di Universitas Islam Indonesia.
9. Hashfi Adib Ghazian serta Syaerul Alim B Z yang telah menjadi teman sejati disaat susah maupun senang.
10. Teman – teman kos Wisma Putra Pangestu yang telah memberikan semangat dan doanya kepada penulis.
11. Para developers di *stackoverflow* yang selalu memberikan solusi pada setiap permasalahan yang penulis hadapi.
12. Semua pihak yang tidak dapat penulis sebutkan satu persatu, terimakasih atas seluruh bantuan dan doanya.

Tugas akhir ini tidak lepas dari kekurangan dan ketidak sempurnaan oleh karena terbatasnya kemampuan dan pengetahuan penulis, oleh karena itu kritik dan saran sangat penulis harapkan untuk kemudian dijadikan perbaikan dimasa mendatang. Semoga laporan ini dapat memberikan manfaat bagi pembacanya. Amin Ya Rabbal Alamin.

Wassalamu'alaikum warahmatullahi wabarokatuh.

Yogyakarta, November 2016

Achmad Naufal Syafiq



SARI

Kebutuhan individu akan informasi semakin meningkat sejalan dengan cepatnya perkembangan alat komunikasi yang belakangan ini menjadi trend bagi segala kalangan. Pertukaran informasi yang kian cepat membuat pengguna sering mendapat informasi yang tidak mereka inginkan. Banyak jenis aplikasi yang mencoba mengelompokkan penyebaran informasi seperti forum internet dan sebagainya, namun belum mengakomodasi beberapa hal seperti kemudahan akses data dan penyampaian informasi secara real – time, kebutuhan akan fitur untuk mengelompokkan pembahasan berdasarkan sebuah permasalahan (threading) sangat dibutuhkan juga belum banyak tersedia pada aplikasi chatting. Dari beberapa kekurangan tersebut terpikirlah ide untuk membuat aplikasi yang dapat mengakomodasi pertukaran informasi melalui chatting secara real – time dan juga memberikan batasan dari pembahasannya.

Aplikasi Chatting Dengan Fitur Threading berbasis sistem operasi android, menerapkan Usecase Diagram dan Activity Diagram sebagai alat untuk merancang kebutuhan fungsionalitas yang diimplementasikan pada saat pembuatan aplikasi. Android Studio digunakan sebagai Integrated Development Environment (IDE) dengan bahasa pemrograman Java dan menggunakan Firebase Realtime Database Sebagai database sebagai basis datanya.

Perancangan aplikasi menggunakan beberapa metode, mulai dari perancangan antarmuka, perancangan basis data hingga perancangan fungsionalitas yang menentukan fitur – fitur apa saja yang terdapat didalam aplikasi. Proses implementasi dilakukan sekaligus pengujian baik dari sisi fungsionalitas maupun non-fungsionalitas untuk memastikan apakah aplikasi sudah dibuat sesuai dengan perancangan. Dari proses perancangan, implementasi aplikasi hingga proses pengujian didapatkan bahwa aplikasi sudah dibuat sesuai dengan perancangan dan dapat mengakomodasi percakapan sesuai dengan pembahasan.

Kata kunci : Aplikasi Chatting, Forum, Android Studio, Firebase, Real – Time, Database, Usecase Diagram, Activity Diagram.

TAKARIR

<i>Username</i>	Nama Pengguna
<i>Profile Photo</i>	Foto Profil
<i>Chat</i>	Pesan
<i>Real – time</i>	Waktu Nyata
<i>Device</i>	Perangkat
<i>Software</i>	Perangkat Lunak
<i>Online</i>	Dalam Jaringan
<i>Offline</i>	Luar Jaringan
<i>Capability</i>	Kemampuan
<i>Password</i>	Sandi
<i>Email</i>	Surel
<i>Database</i>	Basis Data
<i>Login</i>	Masuk
<i>Logout</i>	Keluar
<i>Join</i>	Bergabung
<i>Activity Diagram</i>	Diagram aktifitas
<i>Usecase Diagram</i>	Diagram Usecase
<i>Alert</i>	Peringatan

DAFTAR ISI

APLIKASI CHATTING DENGAN FITUR THREADING BERBASIS SISTEM	
OPERASI ANDROID.....	i
LEMBAR PENGESAHAN PEMBIMBING.....	ii
LEMBAR PENGESAHAN PENGUJI.....	iii
LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN.....	v
MOTTO.....	vi
KATA PENGANTAR.....	vii
SARI.....	ix
TAKARIR.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR.....	xv
BAB I LATAR BELAKANG.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
1.6 Metodologi Penelitian.....	2
1.7 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI.....	5
2.1 Forum.....	5
1.1.1 Forum Internet.....	5
2.1.2 Threading.....	5
2.1.3 Komunikasi dalam forum.....	6

2.1.4	Contoh Forum.....	6
2.2	Chatting	9
2.2.1	Aplikasi <i>chatting</i> dengan fitur <i>threading</i>	9
2.2.2	Contoh Aplikasi Chatting	10
2.2.3	Komparasi Aplikasi Serupa.....	12
2.3	Android.....	12
2.3.1	Aplikasi Berbasis Android.....	13
2.4	Android Studio	13
2.5	Picasso Image Loader.....	13
2.5.1	Picasso Circle Transform.....	13
2.6	Firestore	14
2.6.1	Firestore Authentication	14
2.6.2	Firestore real – time database.....	14
2.6.3	Firestore Storage.....	15
BAB III ANALISIS DAN PERANCANGAN		16
3.1	Konsep Sistem.....	16
3.2	Analisis Sistem.....	17
3.3	Kebutuhan Sistem	17
3.3.1	Aktor.....	17
3.3.2	Kebutuhan Sistem Fungsional.....	17
3.3.3	Kebutuhan Sistem Non Fungsional	19
3.4	Use Case Diagram	19
3.5	Activity Diagram	20
3.6	ERD (<i>Entity Relationship Diagram</i>).....	32
3.6.1	Relasi Antar Entitas.....	34

3.7 Perancangan Antar Muka.....	35
BAB IV HASIL DAN PEMBAHASAN	48
4.1 Batasan Implementasi	48
4.2 Hasil Implementasi	48
4.2.1 Hasil Implementasi Kebutuhan Registrasi <i>User</i>	48
4.2.2 Hasil Implementasi Kebutuhan <i>Login</i>	50
4.2.3 Hasil Implementasi Kebutuhan Buat Thread	51
4.2.4 Hasil Implementasi Kebutuhan Lihat Thread	53
4.2.5 Hasil Implementasi Kebutuhan Lihat <i>Chat</i>	54
4.2.6 Hasil Implementasi Kebutuhan Lihat Deskripsi Thread	55
4.2.7 Hasil Implementasi Kebutuhan <i>Join</i> Thread.....	56
4.2.8 Hasil Implementasi Kebutuhan <i>Leave</i> Thread.....	57
4.2.9 Hasil Implementasi Kebutuhan Kirim <i>Chat</i>	57
4.2.10 Hasil Implementasi Kebutuhan <i>Reset Password</i>	58
4.2.11 Hasil Implementasi Kebutuhan Kelola Akun	59
4.2.12 Hasil Implementasi Kebutuhan <i>Sign – Out</i>	61
4.3 Pengujian Perangkat Lunak	61
4.3.1 Pengujian Menggunakan <i>Black Box Testing</i>	61
4.3.2 <i>User Acceptance Test</i> (UAT)	66
BAB V KESIMPULAN DAN SARAN.....	68
5.1 Kesimpulan	68
5.2 Saran	68
DAFTAR PUSTAKA	69

DAFTAR TABEL

Tabel 1.1 Tabel Komparasi Aplikasi Serupa	12
Tabel 4.1 Tabel Pengujian <i>Blackbox – Testing</i>	61
Tabel 4.2 Tabel Pengujian Kompatibilitas.....	67



DAFTAR GAMBAR

Gambar 2.1 Halaman Muka kaskus.com	7
Gambar 2.2 Fitur Thread Kaskus	7
Gambar 2.3 Fitur <i>Comment</i> Kaskus	8
Gambar 2.4 Survey Fitur <i>Grouping</i> Pengguna Aplikasi Chatting	9
Gambar 2.5 <i>Chat room</i> dan <i>Timeline</i> Line Chatting App	10
Gambar 2.6 Fitur <i>Grouping</i> Aplikasi Discord	11
Gambar 3.1 <i>Use Case Diagram</i> Aplikasi 4Chat	20
Gambar 3.2 <i>Activity Diagram Login</i>	21
Gambar 3.3 <i>Activity Diagram</i> Buat Thread	22
Gambar 3.4 <i>Activity Diagram Join Thread</i>	23
Gambar 3.5 <i>Acitivity Diagram Leave Thread</i>	24
Gambar 3.6 <i>Acitivity Diagram</i> Kirim Chat	25
Gambar 3.7 <i>Acitivity Diagram</i> Lihat Chat	26
Gambar 3.8 <i>Acitivity Diagram Registrasi User</i>	27
Gambar 3.9 <i>Acitivity Diagram</i> Lihat Thread	28
Gambar 3.10 <i>Acitivity Diagram</i> Reset Password	29
Gambar 3.11 <i>Acitivity Diagram</i> Kelola Akun	30
Gambar 3.12 <i>Acitivity Diagram</i> Melihat Deskripsi Thread	31
Gambar 3.13 <i>Acitivity Diagram Sign-out</i>	32
Gambar 3.14 <i>Entity Relationship Diagram</i> Aplikasi 4chat	34
Gambar 3.15 Rancangan Antarmuka <i>Splash Screen</i>	36
Gambar 3.16 Rancangan Antarmuka <i>Login</i>	37
Gambar 3.17 Rancangan Antarmuka <i>Add Thread</i>	38
Gambar 3.18 Rancangan Antarmuka <i>Hisoty Chat</i>	39
Gambar 3.19 Rancangan Antarmuka <i>Browse</i>	40
Gambar 3.20 Rancangan Antarmuka <i>Register User</i>	41
Gambar 3.21 Rancangan Antarmuka <i>Change Password</i>	42
Gambar 3.22 Rancangan Antarmuka <i>Chatting</i>	43
Gambar 3.23 Rancangan Antarmuka <i>Expand Title And Description</i>	44
Gambar 3.24 Rancangan Antarmuka <i>Expand Title And Description 2</i>	45

Gambar 3.25 Rancangan Antarmuka <i>Update Profile</i>	46
Gambar 3.26 Rancangan Antarmuka <i>Profile Page</i>	47
Gambar 4.1 <i>Button “No account yet? Create one”</i>	49
Gambar 4.2 Tampilan Registrasi User	49
Gambar 4.3 <i>Error handling</i> pada halaman Registrasi User.....	50
Gambar 4.4 Tampilan <i>Login</i>	51
Gambar 4.5 Tombol <i>Add Thread</i>	52
Gambar 4.6 Tampilan <i>Add Thread</i>	52
Gambar 4.7 Tampilan <i>Browse Title</i>	53
Gambar 4.8 <i>Alert Dialog</i> pada <i>Browse Title</i>	54
Gambar 4.9 Tampilan <i>Chat</i> didalam Thread.....	54
Gambar 4.10 Tombol <i>Expand Description</i>	55
Gambar 4.11 <i>Dialog Alert</i> pada halaman <i>Chat</i>	555
Gambar 4.12 Tampilan <i>Tab Joined Thread</i>	56
Gambar 4.13 Contoh Data <i>Chat</i>	57
Gambar 4.14 Tampilan <i>Menu Option</i>	58
Gambar 4.15 Form Reset Password	558
Gambar 4.16 <i>Error Alert</i> ketika password tidak sama.....	59
Gambar 4.17 Halaman <i>Edit Profile</i>	60
Gambar 4.18 Halaman <i>Profile Page</i>	60

BAB I

LATAR BELAKANG

1.1 Latar Belakang

Kebutuhan akan komunikasi yang cepat, murah, dan aman sudah ditunjukkan mulai dari awal manusia menemukan alat komunikasi pertama yang berupa lukisan berisi pesan. Bagaimanapun sejarah tentang komunikasi mulai dari belum ditemukannya alfabet hingga sekarang, menuntun manusia untuk terus berinovasi menemukan cara mempermudah penyampaian pesan, baik berbentuk teks, suara, gambar, maupun media lain.

Threading mulai banyak dikenal oleh para pengguna internet, khususnya mereka yang aktif pada forum – forum sebagai bentuk dari ke-akurasi-an informasi dan fitur yang digunakan untuk menspesifikasikan masalah. Keunggulan informasi yang didapat dari penggunaan fasilitas *Threading* pada forum juga mudahnya pengguna untuk mencari permasalahan yang sesuai dengan kebutuhan informasinya. Adanya fitur *reply* pada forum memungkinkan pengguna untuk memberikan pemikirannya sesuai dengan topik permasalahan yang dijelaskan oleh pembuat thread dengan harapan pembahasan tidak berpindah di luar topik.

Penggabungan beberapa fitur pada satu aplikasi dapat dilakukan untuk mencapai aspek aspek yang dibutuhkan dalam penyampaian informasi (komunikasi). *Communication Software* dengan tujuan penyampaian pesan yang cepat (*real-time chatting*), dan pengelompokan informasi yang dapat dilakukan dengan fitur Thread dapat menjadi solusi permasalahan yang dapat terjadi pada sebuah aplikasi *messaging*.

1.2 Rumusan Masalah

Bagaimana merancang dan membuat aplikasi *chatting* dengan fitur threading di sistem operasi Android?

1.3 Batasan Masalah

Untuk menjaga fokus pembahasan selama proses pembangunan aplikasi maka ditentukan beberapa batasan :

1. Aplikasi tidak menyediakan fitur percakapan personal (*one-on-one*).
2. Aplikasi hanya dapat menyediakan media penyampaian pesan berupa teks.

1.4 Tujuan Penelitian

Yang ingin dicapai dari penelitian ini adalah :

Membangun aplikasi chatting yang memiliki fitur threading berbasis sistem operasi Android.

1.5 Manfaat Penelitian

Diharapkan penelitian dapat menghasilkan aplikasi yang meningkatkan tingkat efisiensi dari percakapan atau pembahasan dari suatu topik atau permasalahan.

1.6 Metodologi Penelitian

1. Pengumpulan Data

Pengumpulan data bertujuan untuk mencari informasi terkait dengan bahasan penelitian yang dapat membantu dalam menentukan atau mendefinisikan fitur yang akan diimplementasikan didalam aplikasi. adapun cara pengumpulannya dilakukan dengan metode sebagai berikut :

- a. Observasi

Mengumpulkan informasi dengan cara melihat sistem sistem serupa yang memiliki fasilitas threading dan juga sistem yang memiliki fasilitas *messaging* sebagai acuan untuk implementasi sistem.

- b. Pustaka

Pengumpulan informasi yang dilakukan dengan cara mencari dari buku baik mengenai perancangan *User Interface*, implementasi teknologi yang dibutuhkan maupun pembuatan aplikasi berbasis Android.

2. Analisa Kebutuhan

Pada tahap ini akan dijelaskan mengenai fitur-fitur untuk membangun aplikasi chatting dengan fitur threading berbasis sistem operasi android, mulai dari tools untuk membangun aplikasi yang berbasis sistem operasi android dan teknologi real-time database yang digunakan untuk membuat fitur chatting didalam aplikasi. Tahap ini juga yang menentukan alur pengerjaan pembuatan aplikasi (*programming*).

3. Perancangan

Proses lanjutan dari analisa yang dilakukan dengan mendefinisikan apa saja yang dibutuhkan pada saat pengimplementasian perangkat lunak. Perancangan aplikasi chatting dengan fitur threading menggunakan beberapa metode yaitu perancangan database, perancangan kebutuhan fungsional, dan juga perancangan antarmuka yang menggunakan *mock-up* sebagai gambaran bagai acuan untuk mengimplementasi tampilan aplikasi.

4. Implementasi

Tahap implementasi dilakukan sesuai dengan perancangan, pembuatan aplikasi dimulai dengan perancangan basis data, kemudian pembuatan aplikasi Android yang meliputi implementasi kebutuhan fungsional maupun antarmuka.

5. Pengujian

Pengujian dilakukan dengan beberapa metode yang ditunjukkan untuk melihat apakah aplikasi sudah sesuai dengan kebutuhan fungsional dan apakah aplikasi sudah mengakomodasi kebutuhan pengguna sesuai dengan tujuannya.

1.7 Sistematika Penulisan

Tugas akhir berikut disusun dengan sistem penulisan sebagai berikut:

BAB I LATAR BELAKANG :

Dalam bab ini akan dibahas mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan metode penelitian.

BAB II LANDASAN TEORI

Pembahasan dibab ini meliputi landasan teori yang digunakan untuk penyusunan tugas akhir pembuatan aplikasi *messaging* dengan fitur multi threading

BAB III ANALISIS DAN PERANCANGAN

Di dalam bab ini berisi pembahasan mengenai perancangan aplikasi mulai dari perancangan basis data, kebutuhan fungsionalitas dari aplikasi hingga perancangan antarmuka. Perancangan ditujukan untuk memberikan gambaran awal dari aplikasi sebelum dilakukan implementasi.

BAB IV HASIL DAN PEMBAHASAN

Pembahasan dan hasil yang dimaksud dalam bab ini adalah menguji aplikasi dengan beberapa metode pengujian untuk mendapatkan *feedback* sehingga dapat disimpulkan hal hal yang berkaitan baik dengan sengan sisi fungsionalitas maupun kompatibilitas aplikasi.

BAB V SIMPULAN DAN SARAN

Berisi kesimpulan dari seluruh proses pengerjaan perangkat lunak juga saran dan kesimpulan yang dapat dijadikan sebagai saran untuk pengembangan aplikasi *messaging* dengan fitur threading ini.

BAB II

LANDASAN TEORI

2.1 Forum

Menurut Kamus Besar Bahasa Indonesia (KBBI) Forum adalah lembaga atau badan, bisa disebut juga wadah namun dalam pengertian lain sebuah forum bisa didefinisikan sebuah tempat pertemuan untuk bertukar pikiran secara bebas.

1.1.1 Forum Internet

Sebuah forum internet dapat didefinisikan sebagai tempat berdiskusi pada website. Ditujukan untuk memfokuskan topic diskusi mengenai segala hal. Kemudahan dalam mengakses dan memberikan pernyataan melalui *online* dengan berbagai sarana seperti media sosial , maupun website yang didedikasikan untuk sebuah forum seperti Kaskus.co.id menjadi salah satu faktor kesuksesan forum internet untuk menggaet penggunanya.

2.1.2 Threading

Threading umumnya fitur utama yang terdapat pada forum – forum internet digunakan untuk memisahkan suatu pembahasan dengan pembahasan lainnya, biasanya threading terdiri dari beberapa komponen yaitu, judul, deskripsi dan terdapat juga fitur *comment* yang digunakan baik oleh pembuat thread sendiri maupun pengguna forum lainnya, threading sendiri ditujukan untuk meningkatkan ke-efektifan dari pembahasan sebuah masalah. Dengan mempersempit topik pembahasan pada sebuah group discussion diharapkan pembahasan dapat menjadi lebih produktif sehingga menghasilkan kesimpulan sesuai dengan permasalahan, Seperti dinyatakan oleh David Gram (<http://smallbusiness.chron.com/effective-group-communication-processes-3187.html>) :

“Groups, or work teams, can accomplish great things in small and large businesses alike. A group's overall effectiveness, however, hinges to a great extent on the effectiveness of the participants' communication abilities”.

Thread biasanya dikelompokkan kedalam masing masing kategori sesuai dengan objek yang menjadi pembahasan.

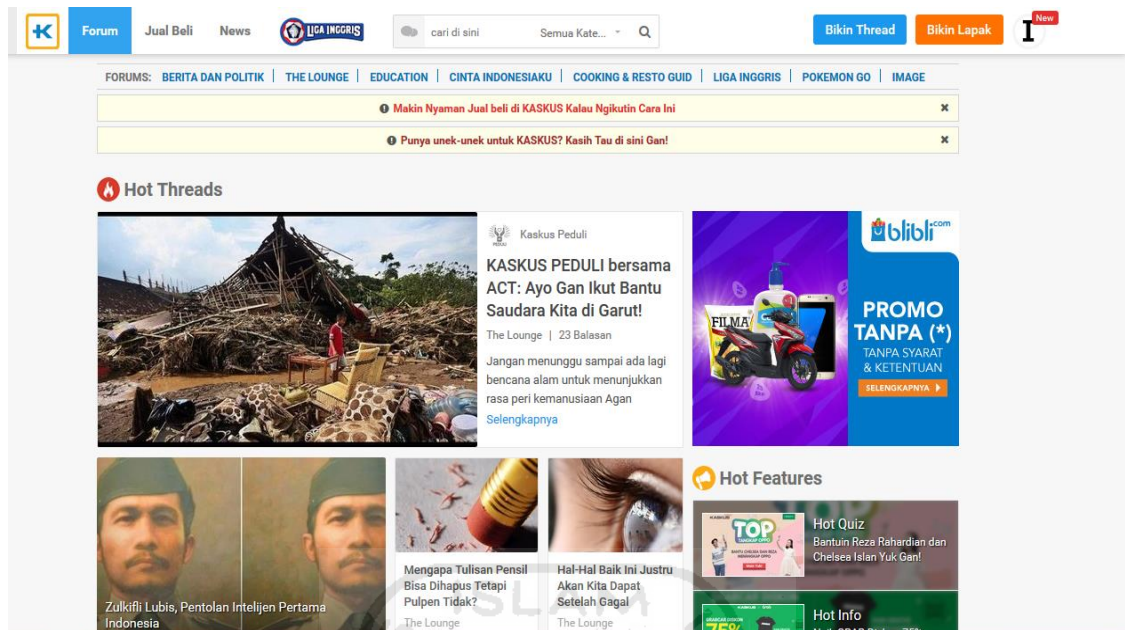
2.1.3 Komunikasi dalam forum

Threading merupakan salah satu istilah dalam forum yang digunakan untuk menggambarkan sebuah permasalahan sebuah thread biasanya terdiri dengan judul dan deksripsi dari masalah yang akan dibahas. Untuk memeberikan tanggapan pada sebuah permasalahan para pengguna forum biasanya menggunakan fitur comment. Sebuah forum juga biasanya memberikan fasilitas pengiriman pesan sehingga antar sesama user dapat saling berkomunikasi secara langsung

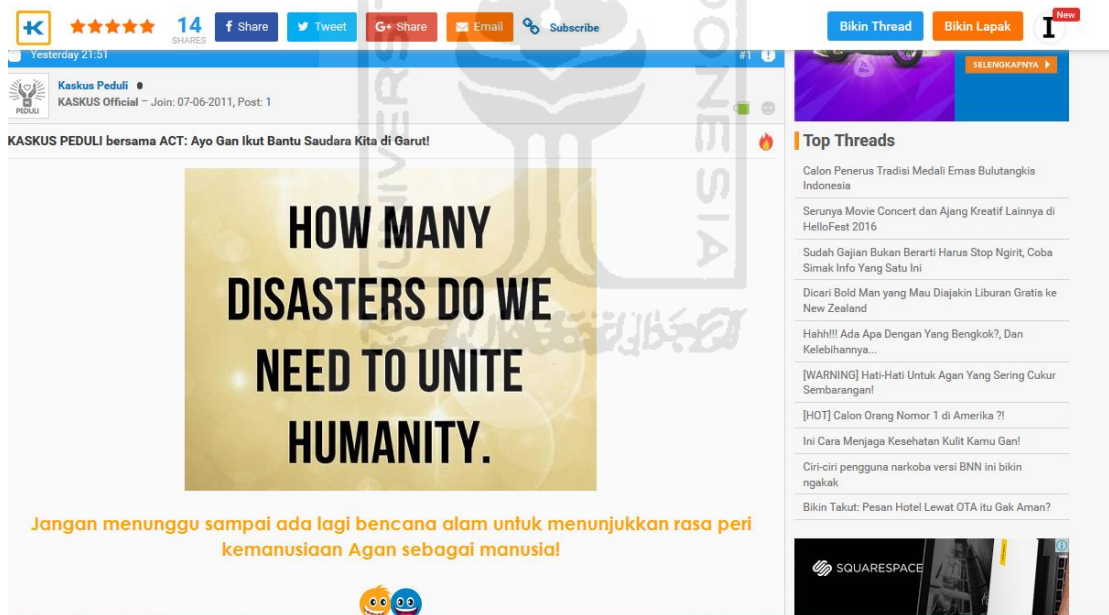
2.1.4 Contoh Forum

Dengan kemajuan teknologi yang pesat, termasuk juga didalamnya internet, menimbulkan banyak forum - forum dengan kategori dan tujuan masing masing berikut beberapa contoh forum yang terdapat pada internet :

Kaskus merupakan satu dari banyak forum di Indonesia yang berhasil untuk terus berkembang dan juga merupakan salah satu forum terbesar di Indonesia. Fitur utama dari kaskus adalah threading seperti dapat pada gambar 2.2, dimana terdapat menu *HOT THREAD* seperti pada gambar 2.1, merupakan kumpulan artikel artikel yang mendapat banyak *rate* (fitur yang digunakan oleh user untuk menilai suatu thread) dan juga banyaknya komentar dari para pengguna forum lain. Selain itu masing masing user dilengkapi dengan reputasi sebagai acuan baik buruknya aktifitas setiap user yang diberikan oleh pengguna lainnya yang dinotasikan sebagai CENDOL (untuk reputasi baik) dan BATA (untuk reputasi buruk). Komentar sendiri merupakan fitur dalam forum yang digunakan untuk memberikan tanggapan pada suatu thread, contohnya dapat dilihat di gambar 2.3.



Gambar 2.1 Halaman Muka kaskus.com



Gambar 2.2 Fitur Thread Kaskus

The screenshot displays a forum interface with a navigation bar at the top containing 'Forum', 'Jual Beli', 'News', and 'LIGA INGGRIS'. A search bar is located on the right. The main content area shows a thread from 'Yesterday 21:52' with a post by user 'SadarAljaber' (Kaskus Maniac, joined 04-12-2011, 4,783 posts) containing the text 'pertamex ka?' and 'ane bantu doa aja gan, belum gajian' accompanied by a 'LEH UGA' meme. Below this, a comment by user 'manggor18' (Kaskus Maniac, joined 15-10-2013, 6,904 posts) is shown, featuring a quote of the original post and the text 'apaan tuh gan? :v'. A large watermark for 'UNIVERSITAS ISLAM INDONESIA' is overlaid on the bottom half of the image.

Gambar 2.3 Fitur *Comment* Kaskus

2.2 Chatting

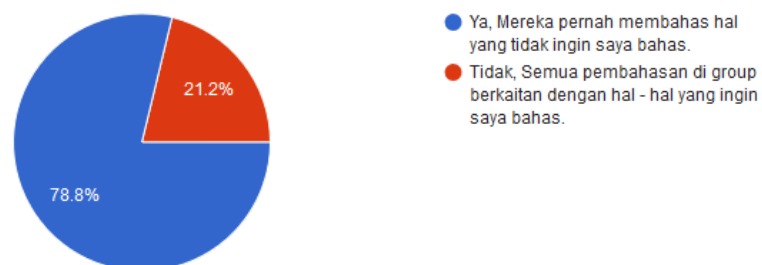
Chatting merupakan sebuah bentuk komunikasi yang dilakukan melalui internet dalam bentuk tulisan secara *real time*. Memiliki berbagai bentuk mulai dari aplikasi berbasis website, desktop, maupun aplikasi berbasis mobile. Di era yang serba instan dan komunikasi merupakan salah satu kebutuhan pada saat ini, *chatting* melalui media internet dinilai memberikan berbagai keuntungan seperti mudahnya pengiriman dan cepatnya penyampaian pesan. Biaya penggunaan internet yang semakin murah juga menjadi faktor pendukung banyaknya pengguna fasilitas *chatting* pada internet.

2.2.1 Aplikasi *chatting* dengan fitur *threading*

Aplikasi *chatting* dengan fitur *threading* merupakan aplikasi yang memberikan sarana bagi pengguna forum untuk berkomunikasi secara *real-time*. Dengan pemberian fitur *real-time* pada sebuah pembahasan permasalahan user dapat memberikan pendapatnya melalui *chatting* dengan cepat dan relatif lebih mudah karena aplikasi berbasis perangkat bergerak. Hasil survey (gambar 2.4) menyatakan bahwa dari lebih dari 75% dari 33 responden yang menggunakan aplikasi *chatting* dengan fitur *grouping* menyatakan bahwa ketika melakukan pembahasan terdapat hal hal yang tidak relevant dengan tujuan diskusi.

Apakah ada member pada group yang anda ikuti membahas hal - hal yang menurut anda tidak relevan?

(33 responses)



Gambar 2.4 Survey Fitur *Grouping* Pengguna Aplikasi *Chatting*

Sehingga diharapkan dengan fitur *threading* dapat mengurangi pembahasan yang tidak relevan dengan masalah yang sedang dibahas.

2.2.2 Contoh Aplikasi Chatting

Aplikasi chatting banyak terdapat diberbagai platform baik dekstop maupun mobile, dengan semakin majunya teknologi mobile phone, banyak developer yang mengkhususkan untuk membuat aplikasi yang berbasiskan perangkat bergerak baik didalam sistem operasi Android, iOS, Windows Phone, maupun lainnya.

1. Line

Line merupakan salah satu aplikasi chatting dengan pengguna terbanyak didunia dengan hampir 218 juta user yang aktif. Fitur utama dari Line adalah percakapan personal namun juga terdapat fitur *Timeline* yang memunculkan status yang *diposting* baik user itu sendiri maupun lainnya seperti pada gambar 2.4.



Gambar 2.5 Chat room dan Timeline Line Chatting App

2. Discord

Discord merupakan salah satu aplikasi chatting yang memerlukan *key* untuk masuk dalam sebuah percakapan tertentu, apabila user tidak memiliki *key* tersebut maka pengguna tidak diperbolehkan masuk, cara lain untuk bergabung dalam percakapan adalah mendapatkan *invite* dari user lain yang sudah bergabung didalam suatu percakapan. Contoh grouping dalam aplikasi discord dapat dilihat pada gambar 2.5.



Gambar 2.6 Fitur *Grouping* Aplikasi Discord

2.2.3 Komparasi Aplikasi Serupa

Tabel 1.1 Tabel Komparasi Aplikasi Serupa

	Threading	Realtime Discussion	Mobile	Public
Kaskus	✓		✓	✓
Discord		✓	✓	
Line		✓	✓	

Selain fitur threading, realtime dan dapat diakses melalui perangkat bergerak untuk memudahkan pengguna dalam penggunaan aplikasi, pengaksesan data harus dapat dilakukan secara menyeluruh, sehingga pengguna dapat mencari thread atau pembahasan yang ingin mungkin sudah dibuat sebelumnya oleh pengguna lain. Dengan data yang ditunjukkan secara *public*, diharapkan dapat mengurangi duplikasi pembahasan, sehingga dapat meningkatkan keefektifan dari fitur threading.

2.3 Android

Android adalah sebuah sistem operasi terbuka gratis, yang dikembangkan oleh Google. Android versi 1.x diluncurkan pertamakali pada tahun 2008 meskipun tergolong baru android kabarkan sudah memiliki fitur yang lebih baik dibandingkan iPhone maupun Windows mobile, hal tersebut dimungkinkan karena Android sistem terhubung dengan Google *Services* yang memungkinkan banyak fitur terdapat didalamnya, meskipun pada awalnya terdapat banyak *bugs*, dan kecepatan proses yang lamban. Android 2.x khususnya 2.3 Gingerbread merupakan pengembangan dari versi sebelumnya dengan beberapa kemajuan seperti *user experience* yang lebih baik android dapat bersaing dengan sistem operasi lainnya dipasaran. Hingga saat ini Android sudah meluncurkan banyak versi sampai yang terbaru yaitu Android 7.0 Nougat, diklaim Android memiliki 1.4 miliar user aktif diseluruh dunia. Menurut data dari statcounter.com, bulan Juni 2016 Android menduduki peringkat utama mengalahkan kepopuleran sistem operasi yang lainnya.

2.3.1 Aplikasi Berbasis Android

Aplikasi native merupakan aplikasi yang dibuat menggunakan bahasa pemrograman spesifik untuk sebuah platform tertentu. Aplikasi android menggunakan bahasa pemrograman Java sebagai *native programming language*. Aplikasi native pada umumnya memberikan performa yang lebih baik dibandingkan dengan aplikasi berbasis lain dengan waktu akses yang lebih cepat dan dapat dibuka dalam keadaan *offline* maupun *online*. Penggunaan layar sentuh pada aplikasi yang berbasis android juga sangat membantu kemudahan dalam pengoprasiannya, dengan fasilitas touch-screen memudahkan user untuk mengakses fitur dengan satu sentuhan jari. Berbeda dengan aplikasi web yang membutuhkan aplikasi lain berupa browser dan harus mengetikkan alamat web, *native android application* dirancang khusus untuk tugas tertentu setiap aplikasinya.

2.4 Android Studio

Merupakan aplikasi IDE (Integrated Development Environment) resmi yang dirancang khusus untuk pengembangan *software* berbasis android. Diumumkan pada 16 Mei, 2013 pada acara Google I/O *conference*. Merupakan perangkat lunak gratis yang tersedia dibawah lisensi Apache 2.0. Android studio dibangun untuk memudahkan para pembembang perangkat Android dengan berbagai fitur seperti support untuk pengembangan aplikasi *Android Wear*.

2.5 Picasso Image Loader

Picasso image loader adalah sebuah *library* yang dikembangkan untuk *platform* android dengan tujuan utama untuk mengunduh dan menyimpan sementara file image didalam aplikasi android.

2.5.1 Picasso Circle Transform

Merupakan sebuah *library* dibawah picasso yang ditujukan untuk menampilkan gambar dalam bentuk lingkaran dimasing - masing *device*.

2.6 Firebase

Adalah layanan DbaaS (Database as a Service) dengan konsep realtime. Firebase merupakan penyedia layanan cloud dengan backend sebagai servis yang berbasis di San Fransisco, California. Merupakan pengembangan dari teknologi sebelumnya yaitu GCM (Google Chat Message).

2.6.1 Firebase Authentication

Dalam pengerjaan aplikasi ini firebase authentication berperan untuk *handle* aktifitas autentikasi pengguna, mulai dari pendaftaran, penyimpanan data *profile* pengguna dan terdapat fungsi fungsi yang digunakan untuk manajemen user, seperti mengubah password, dan mereset password user.

2.6.2 Firebase real – time database

Real time database yang disediakan oleh firebase merupakan sebuah database yang dalam pengimplementasiannya pengembang tidak perlu menambahkan fungsi real-time secara terpisah seperti pada database yang berbasis SQL (*Structuraed Query Language*) lainnya, seperti MySql, Oracle, PostgreSQL dan lain lainnya karena Firebase sudah memberikan fitur *real time* ketika pengembang mengimplementasikannya kesetiap project yang sudah terdaftar library Firebase didalamnya.

Tidak seperti database pada umumnya Firebase merupakan *database NoSql* dimana dalam pemanggilan data Firebase hanya menyediakan Sql – like query seperti *orderByChild*, *startAt*, *startEnd* dan beberapa fungsi penyaringan lainnya. Tidak hanya memberikan fitur *real – time* didalam pengimplementasiannya, Firebase juga memberikan fitur *offline capability* dimana pengembang dapat mengembangkan aplikasi dengan kemampuan apabila perangkat pengguna tidak terhubung dengan internet maka user masih dapat melihat data yang sebelumnya sudah disimpan secara otomatis ketika mengakses database melalui internet.

Didalam aplikasi ini Firebase realtime database merupakan database utama dan hanya satu satunya database yang digunakan, baik untuk menyimpan thread yang dibuat oleh user, hingga percakapan yang dilakukan oleh masing masing pengguna. Pemilihan database ini didasarkan pada fungsi *real – time* yang sangat

efisien apabila digunakan untuk aplikasi yang menyediakan fitur *chatting*, karena proses *chatting* pada umumnya dilakukan dengan *real – time*, ketika satu user memasukkan data kedalam database, otomatis user lain yang memiliki *previledge* untuk dapat melihatnya tanpa harus melakukan proses *refresh*.

2.6.3 Firebase Storage

Firebase storage dikembangkan oleh google untuk memenuhi kebutuhan pengembang akan media penyimpanan *online* atau *cloud storage* yang terintegrasi langsung dengan Firebase cloud service. Firebase cloud storage pada aplikasi ini digunakan untuk menyimpan file berupa gambar yang diunggah oleh user untuk nantinya dipakai sebagai foto profil masing masing user. Foto foto yang diunggah oleh pengguna akan diterima oleh aplikasi sebagai URL (*uniform resource locator*) dan dimasukkan kedalam masing – masing akun pengguna.



BAB III

ANALISIS DAN PERANCANGAN

3.1 Konsep Sistem

Aplikasi chatting dengan fitur threading (4chat) merupakan jenis aplikasi dengan kategori komunikasi yang memiliki tujuan untuk memudahkan orang membahas sebuah topik permasalahan dengan mempersempit topik pembicaraan sehingga para pengguna dapat lebih fokus membicarakan topik tertentu yang ingin diselesaikan.

Aplikasi sejenis dengan kategori komunikasi menyediakan fasilitas *real – time chatting* baik satu user ke user lain, maupun fitur *group* yang memungkinkan para pengguna untuk membuat sebuah wadah bagi beberapa user untuk saling berkomunikasi. Aplikasi chatting dengan fitur threading memiliki perbedaan dalam membentuk sebuah percakapan. Para user diharuskan memiliki topik pembicaraan yang dapat dibahas oleh pengguna lain sehingga pembahasan yang dilakukan mengerucut pada sebuah masalah.

Pengguna memiliki kebebasan untuk memilih topik mana yang ingin dia bahas. Didalam aplikasi ini para user dapat melihat topik mana yang baru dibuat oleh user dengan melihat *timeline* topik. Selain itu fitur lain memungkinkan user untuk melihat topik mana saja yang user sudah pernah bahas sebelumnya. Untuk menggunakan aplikasi pengguna diharuskan mendaftar dengan *e-mail*, *password*, dan *username* dan user lain dapat melihat username kita. Fitur notifikasi memungkinkan untuk melihat siapa saja yang memberikan balasan pada topik yang sedang kita ikuti. Aplikasi ini juga menyediakan fitur *reset password* yang digunakan untuk mengganti *password* lama dengan yang baru sesuai kehendak pengguna.

Backend aplikasi ini ditangani oleh teknologi Firebase yang dirancang oleh google untuk membuat aplikasi dengan *real time* database. Sehingga proses management data banyak dilakukan di Firebase *console*, seperti melihat

siapa saja yang sedang melakukan aktifitas chatting dan topik apa saja yang sudah dibuat oleh pengguna.

3.2 Analisis Sistem

Analisis sistem diharapkan dapat meningkatkan optimalisasi untuk pengerjaan aplikasi. Dalam tahap ini akan dijelaskan kebutuhan apa saja yang diperlukan oleh user berdasarkan hasil observasi dengan mengacu pada fitur dari aplikasi sejenis maupun fitur baru yang digunakan untuk mencapai tujuan dari pengerjaan aplikasi ini. Kebutuhan aplikasi dibuat dengan Notebook Hp yang memiliki spesifikasi Processor AMD A8 – 7410, Graphic Card AMD Radeon R5 dan 4GB RAM.

3.3 Kebutuhan Sistem

3.3.1 Aktor

Aplikasi ini nantinya akan digunakan oleh pengguna secara umum apabila sudah memasang pada perangkat Android mereka, pengguna tidak dibatasi selama perangkat yang digunakan memenuhi persyaratan perangkat yaitu *device* dengan sistem operasi Android minimal 4.0.3 (Ice Cream Sandwich). Adapun untuk menggunakan aplikasi ini pengguna diharuskan memiliki koneksi internet untuk dapat mengakses data yang terdapat pada database.

3.3.2 Kebutuhan Sistem Fungsional

1. Register User

Pendaftaran dilakukan untuk autentikasi setiap user yang dilakukan oleh pengguna, dalam proses ini user diharuskan untuk mengisi *username*, *e-mail*, dan *password*.

2. Login

Login merupakan prosedur yang harus dilakukan oleh para *end users* apabila ingin menggunakan aplikasi ini. Adapun data yang diisi berdasarkan pada yang sebelumnya dimasukkan pada proses *register*.

3. Buat Thread

Fitur ini memungkinkan par pengguna membuat topik pembicaraan baru dengan dua buah isian data, yaitu judul dan isi dari pembasahan.

4. Lihat Thread

Aplikasi ini menyediakan fitur untuk melihat topik apa saja yang sudah dibuat oleh pengguna lain pada *timeline* lihat topik.

5. Lihat *Chat*

Lihat Chat dapat dilakukan oleh setiap *user* dengan mengakses topik pembahasan yang sebelumnya sudah diikuti (*join*).

6. Lihat Deskripsi Thread

Setiap thread memiliki deskripsi tentang apa yang ingin sipembuat bahas didalam thread, baik si pembuat maupun pengguna lain dapat melihat deksripsi tersebut dengan fitur lihat deskripsi thread.

7. *Join* Thread.

Join thread merupakan fitur yang harus digunakan oleh user apabila ingin bergabung didalam sebuah thread, apabila sudah join pada suatu thread, otomatis akan masuk kedalam chat room, dan judul thread akan masuk kedalam daftar thread yang user tersebut ikuti.

8. *Leave* Thread

Merupakan fitur yang digunakan untuk menghapus keberlangganan user pada sebuah thread yang sebelumnya sudah melakukan join kethread tersebut.

9. Kirim Chat

Kirim Chat merupakan fungsi yang digunakan oleh masing masing pengguna untuk memberikan komentar pada bahasan thread secara *real-time*.

10. Reset Password

Fitur ini memungkinkan user untuk mengganti password yang lama dengan yang baru, user harus terlebih dahulu berada dalam posisi *logged-in* sebelum bisa menggunakan fitur ini.

11. Kelola akun

Kelola akun merupakan sebuah fitur yang memungkinkan user untuk mengganti identitas personal yang digunakan untuk mengidentifikasi data

didalam aplikasi. Ada dua data yang dapat dikelola yaitu, *username* dan foto profil.

12. *Sign – out*

Merupakan sebuah fitur yang digunakan user untuk keluar dari posisi login yang sebelumnya digunakan untuk menggunakan aplikasi.

3.3.3 Kebutuhan Sistem Non Fungsional

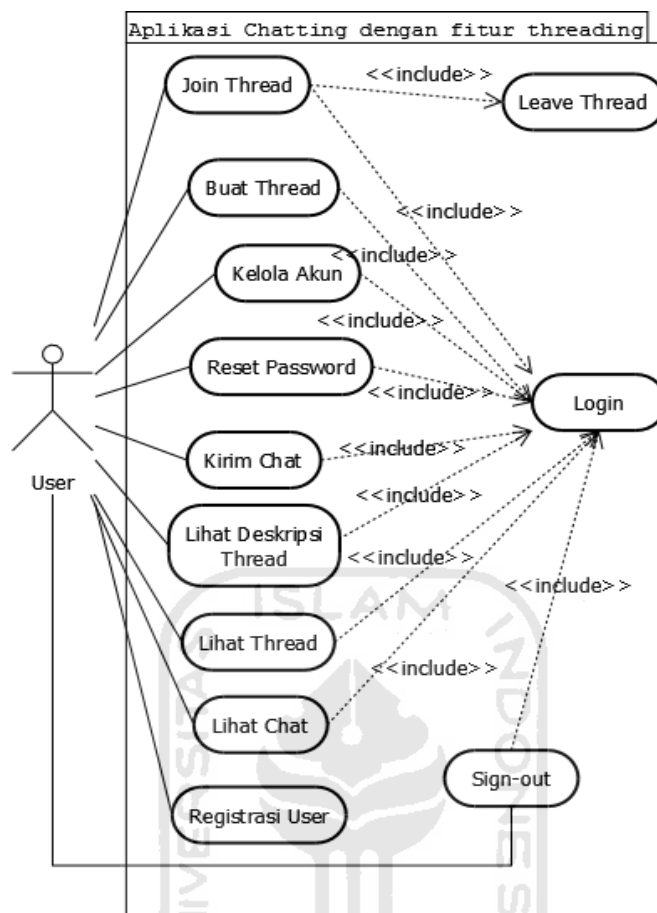
1. *Availability*

Perangkat lunak dapat digunakan oleh pengguna 24 jam dalam 7 hari selama pengguna memiliki akses Internet.

3.4 Use Case Diagram

Use Case Diagram (UCD) digunakan untuk mendeskripsikan aktifitas apa saja yang dapat dilakukan oleh sistem dan siapa yang dapat melakukan aktifitas tersebut. Secara sederhana dapat dijelaskan bahwa UCD adalah diagram yang menjelaskan interaksi yang dilakukan oleh sistem dengan aplikasi. Terdapat dua komponen utama pada *Use Case Diagram* yaitu Aktor dan *Use Case* itu sendiri.

Hanya terdapat 1 aktor pada UCD Aplikasi Chatting dengan Fitur Threading, yaitu pengguna secara umum seperti yang terlihat pada gambar 3.1.



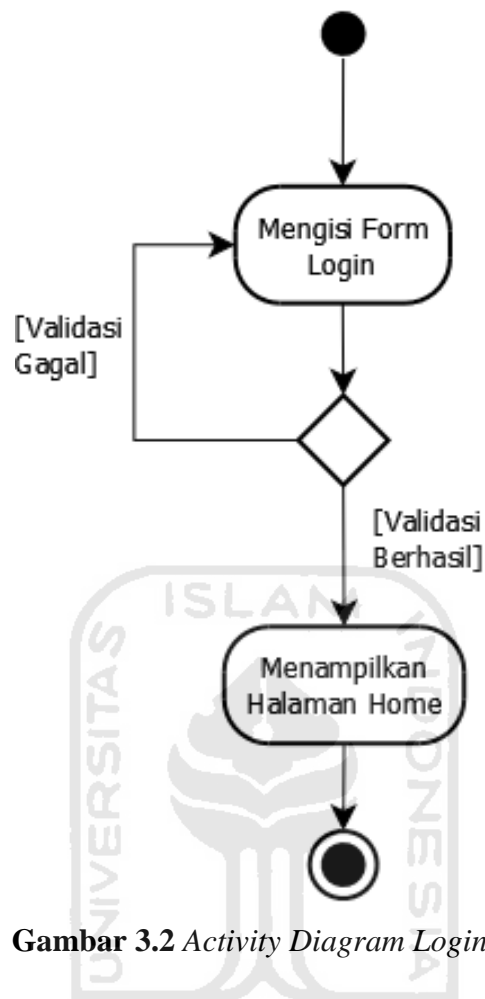
Gambar 3.1 Use Case Diagram Aplikasi 4Chat

3.5 Activity Diagram

Activity diagram merupakan representasi proses sistem yang digambarkan dalam bentuk alur diagram dari masing-masing use-case. Diagram ini menjelaskan bagaimana alur dari sebuah proses dalam sistem dimulai. Diagram dimulai dari aktivitas itu dijalankan hingga aktivitas tersebut berakhir.

1. Activity Diagram Login

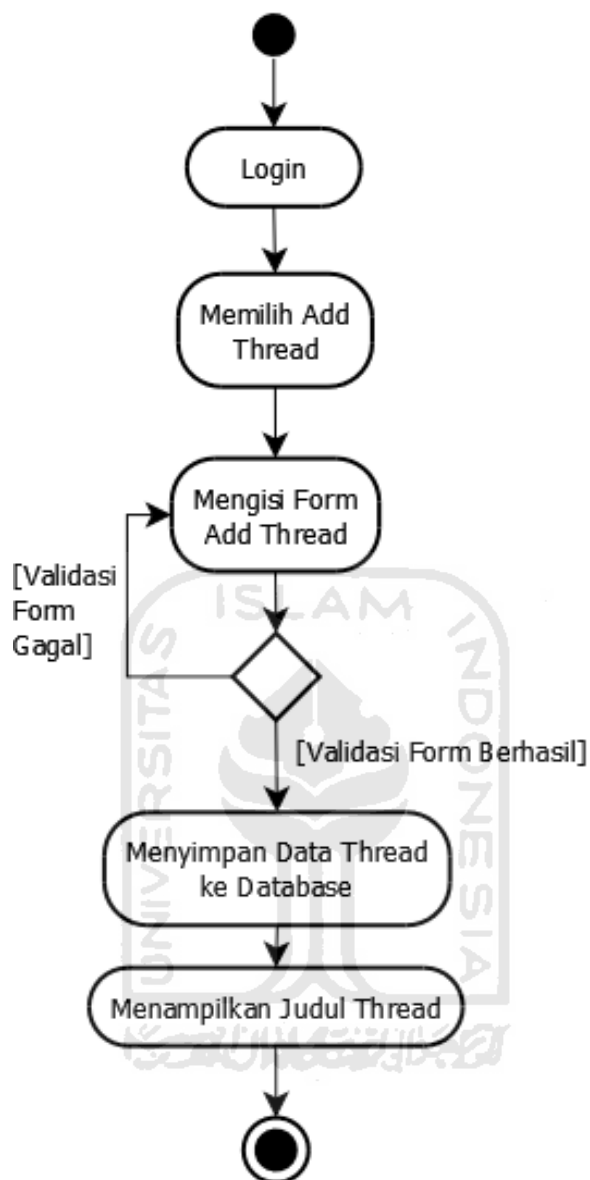
Diagram ini menggambarkan proses awal user ketika ingin menggunakan beberapa fitur lainnya, seperti lihat thread, join thread, chat dan sebagainya. Diagram dapat dilihat pada gambar 3.2.



Gambar 3.2 Activity Diagram Login

2. Activity Diagram Buat Thread

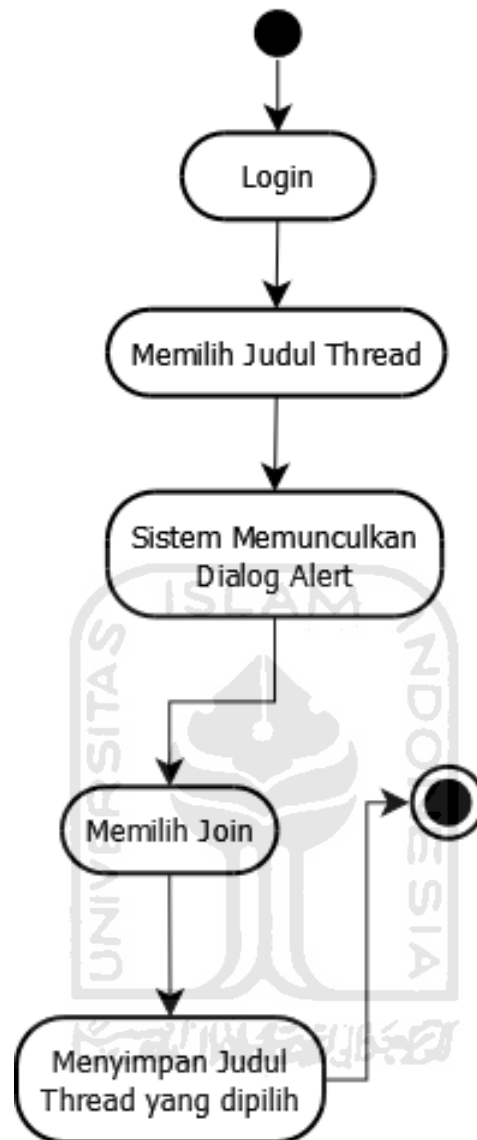
Pada Activity Diagram ini akan digambarkan mengenai proses pembuatan thread, untuk menjalankan fungsi membuat thread, user diharuskan untuk ada pada keadaan login terlebih dahulu. Activity diagram buat thread dapat dilihat pada gambar 3.3.



Gambar 3.3 Activity Diagram Buat Thread

3. Activity Diagram Join Thread

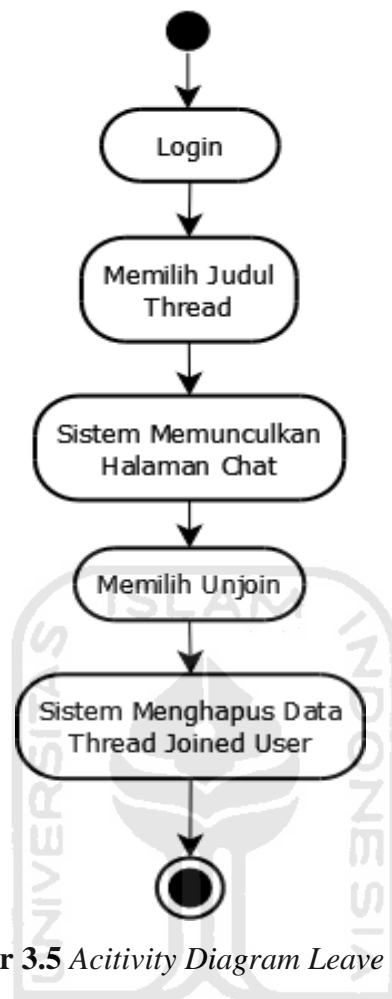
Join Thread merupakan fitur yang digunakan *user* apabila ingin mengikuti percakapan yang ada didalamnya, pengguna diharuskan untuk bergabung. Sama seperti membuat thread, fitur ini juga mengharuskan pengguna untuk login terlebih dahulu. Diagram Aktifitasnya dapat dilihat pada gambar 3.4.



Gambar 3.4 Activity Diagram Join Thread

4. Activity Diagram Leave Thread

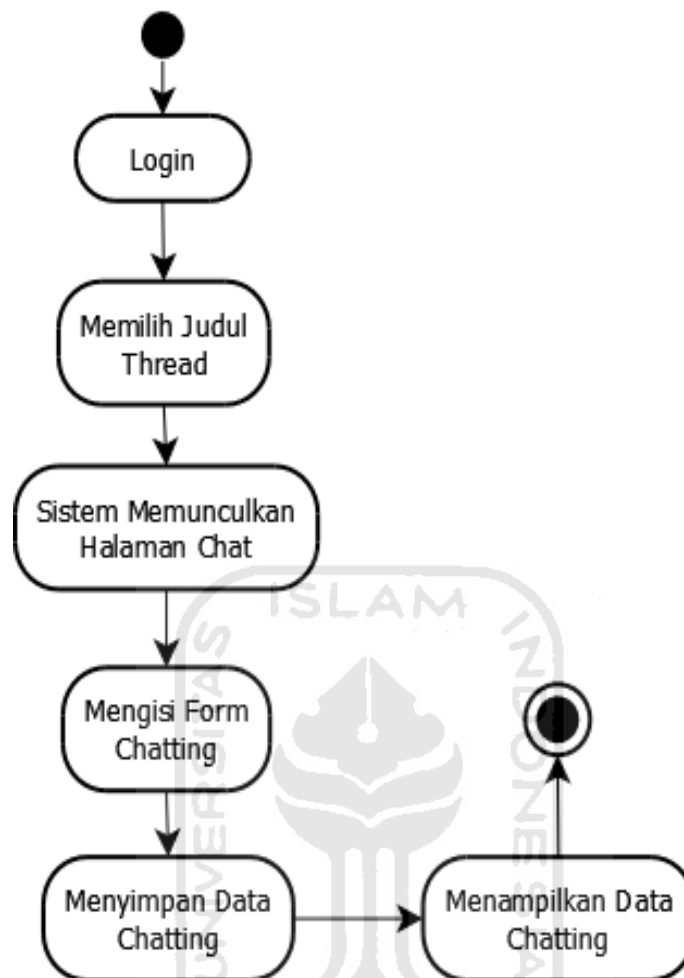
Leave Thread merupakan sebuah aktifitas dimana apabila *user* yang awalnya sudah bergabung pada sebuah percakapan didalam thread memutuskan untuk meninggalkan thread tersebut. Fitur ini hanya dapat digunakan pada thread yang sudah diikuti atau *join* sebelumnya. Activity diagramnya dapat dilihat pada gambar 3.5.



Gambar 3.5 *Acitivity Diagram Leave Thread*

5. *Acitivity Diagram Kirim Chat*

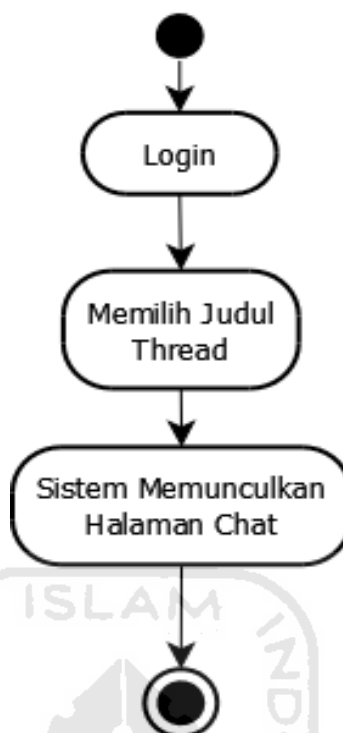
Mengirim chat adalah fitur mengirim pesan didalam masing masing thread, pembahasan masing masing thread berbeda sesuai dengan judul yang ditentukan oleh pembuat thread. Diagram dari aktifitas ini dapat dilihat pada gambar 3.6.



Gambar 3.6 *Acitivity Diagram* Kirim Chat

6. *Acitivity Diagram* Lihat Chat

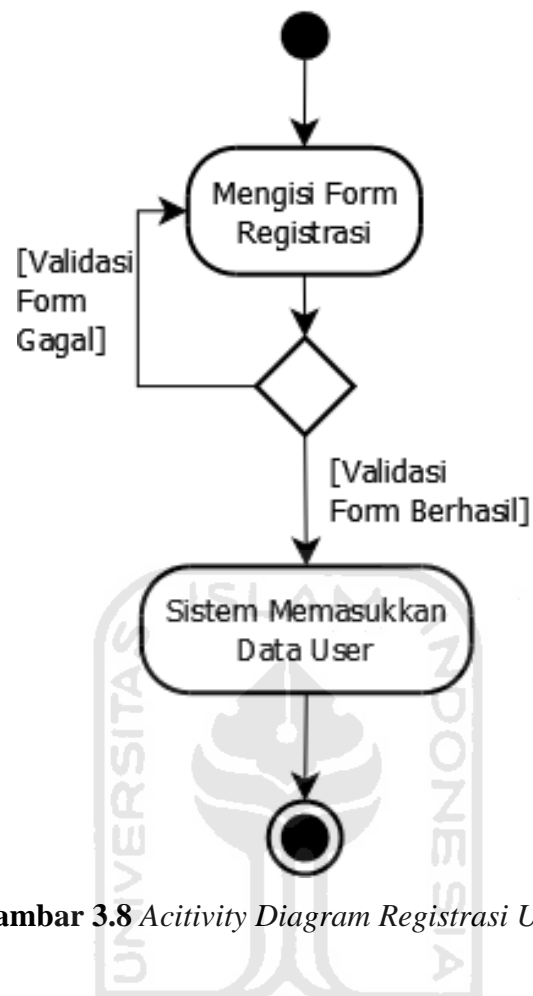
Sama seperti aktifitas *Kirim Chat*, aktifitas melihat *chat* juga dilakukan pada masing masing thread yang sebelumnya user sudah melakukan proses Join terlebih dahulu, aktifitas ini digunakan user untuk melihat pembahasan dimasing masing thread. *Activity diagram*-nya dapat dilihat pada gambar 3.7.



Gambar 3.7 *Activity Diagram* Lihat Chat

7. *Activity Diagram* Registrasi User

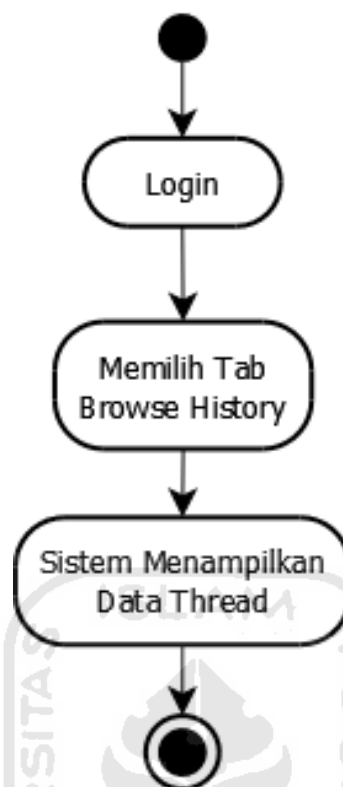
Registrasi user adalah fitur yang digunakan bagi para pengguna aplikasi yang ingin mengikuti percakapan didalam thread, namun belum memiliki akun. Didalamnya terdapat form yang berisi data untuk mengautentikasi user. Gambar diagram dari aktifitas ini dapat dilihat pada gambar 3.8.



Gambar 3.8 *Activity Diagram Registrasi User*

8. *Activity Diagram* Lihat Thread

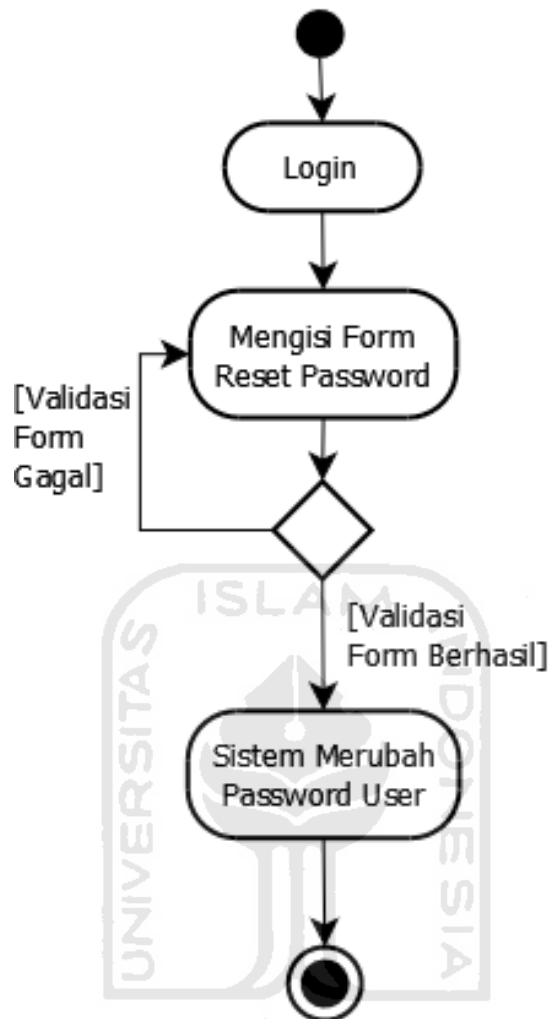
Activity Diagram Lihat Thread merupakan diagram yang menggambarkan aktifitas user untuk melihat thread apa saja yang sudah dibuat baik oleh user itu sendiri maupun user lain. Sama seperti beberapa aktifitas sebelumnya, proses melihat thread hanya dapat dilakukan apabila user sudah login sebelumnya. Gambar diagram dari aktifitas ini dapat dilihat pada gambar 3.9.



Gambar 3.9 *Activity Diagram* Lihat Thread

9. *Activity Diagram* Reset Password

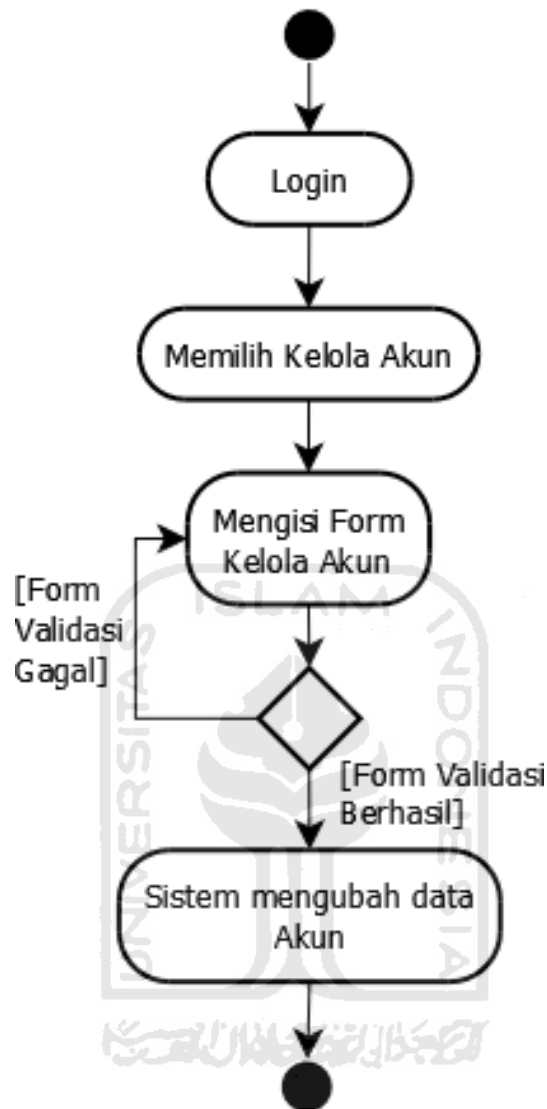
Diagram aktifitas Reset *Password* adalah diagram yang menggambarkan aktifitas apabila user memilih untuk mengganti *password* lamanya dengan *password* yang baru. Untuk melakukan aktifitas ini user diharuskan login terlebih dahulu. Diagramnya dapat dilihat pada gambar 3.10.



Gambar 3.10 *Activity Diagram* Reset Password

10. *Activity Diagram* Kelola Akun

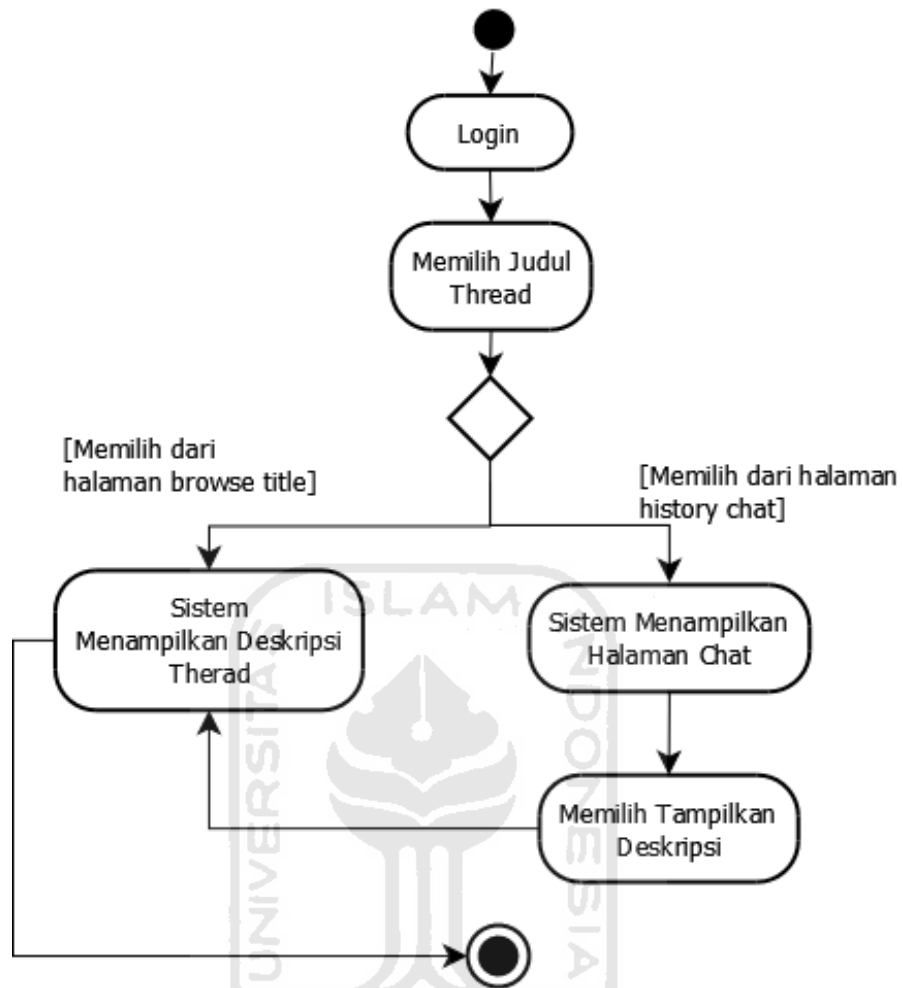
Mengelola akun merupakan sebuah aktifitas yang digunakan user untuk merubah identitasnya yang akan ditampilkan apabila user melakukan *chatting* didalam thread dengan pengguna lain. Untuk melakukan aktifitas ini user diharuskan berada dalam posisi sudah *login*. Diagram aktifitas Kelola Akun dapat dilihat pada gambar 3.11.



Gambar 3.11 *Acitivity Diagram* Kelola Akun

11. *Acitivity Diagram* Melihat Deskripsi Thread

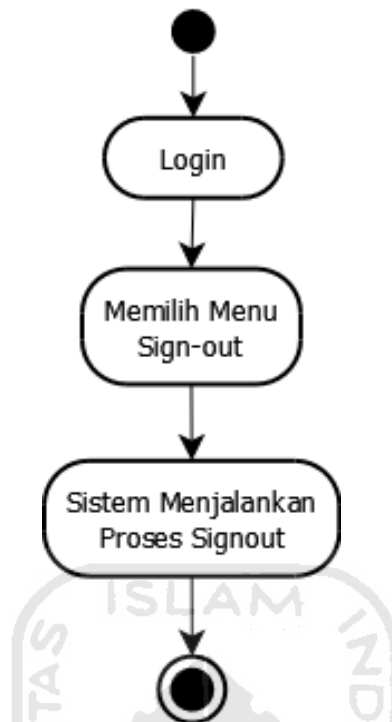
Aktifitas melihat deskripsi thread ini menjelaskan bagaimana proses user untuk melihat deskripsi dari sebuah thread yang sebelumnya sudah dibuat baik oleh user tersebut maupun user lain. Ada dua cara yang bisa dilakukan oleh user untuk melihat deskripsi dari sebuah thread. Gambar diagram dari aktifitas ini dapat dilihat pada gambar 3.12.



Gambar 3.12 *Acitivity Diagram* Melihat Deskripsi Thread

12. *Acitivity Diagram Sign Out*

Sign- out merupakan sebuah aktifitas dimana apabila user ingin keluar dari akun yang sebelumnya sudah digunakan untuk mengautentifikasi pengguna tersebut. Dengan aktifitas ini user sudah tidak dapat mengakses beberapa fitur yang berada didalam aplikasi hingga user kembali melakukan proses login. Diagram dari aktifitas ini dapat dilihat pada Gambar 3.13.



Gambar 3.13 *Acitivity Diagram Sign-out*

3.6 ERD (*Entitiy Relationship Diagram*)

ERD merupakan sebuah teknik pemodelan data yang direpresentasikan dalam bentuk diagram berisikan entitas dari sistem dan hubungan antara entitas tersebut. Dalam pengembangan aplikasi chatting yang memiliki fitur threading ERD yang dibuat akan diimplementasikan kedalam Firebase Database.

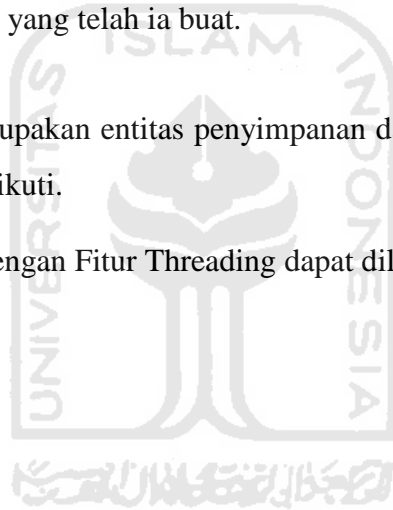
Ada lima entitas yang terdapat didalam ERD Aplikasi Chatting Dengan Fitur Threading, diantaranya adalah :

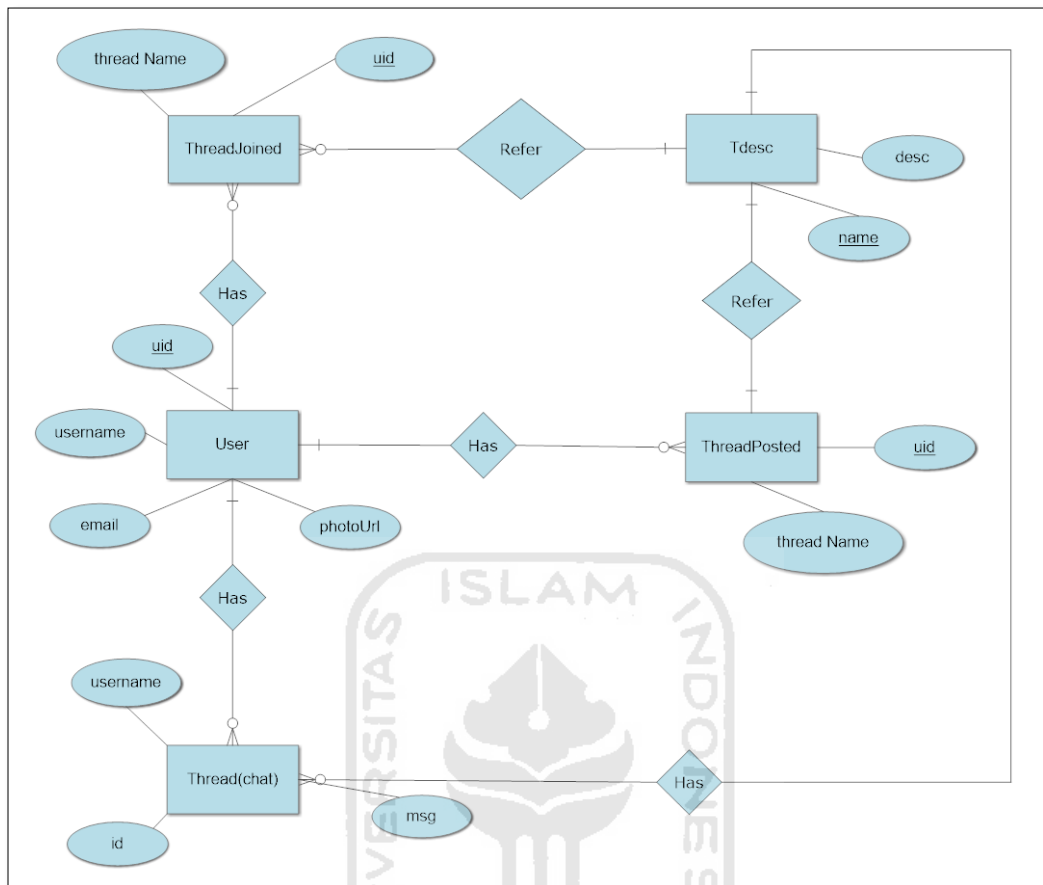
1. *User*

User merupakan sebuah entitas sekaligus berperan sebagai aktor atau end user dari aplikasi ini, user memiliki data personal yang nantinya akan digunakan untuk memberikan identitas pada data yang dimunculkan pada aplikasi.

2. *Tdesc*
Sebuah entitas yang merepresentasikan penyimpanan data thread yang berisikan judul dan deskripsi dari sebuah thread.
3. *Thread (chat)*
Entitas Thread (*chat*) adalah sebuah entitas penyimpanan data yang didalamnya terdapat pesan yang dikirimkan oleh satu user didalam sebuah thread, pesan yang dikirimkan nantinya akan disimpan sesuai dengan threadnya, sehingga tidak muncul pada thread lain.
4. *Thread Posted*
Thread poster adalah sebuah entitas dimana didalamnya terdapat data user dan thread apa saja yang telah ia buat.
5. *Thread Joined*
Thread Joined merupakan entitas penyimpanan data yang berisikan thread apa saja yang user ikuti.

ERD Aplikasi Chatting Dengan Fitur Threading dapat dilihat pada gambar 3.14.





Gambar 3.14 Entity Relationship Diagram Aplikasi 4chat

3.6.1 Relasi Antar Entitas

Relasi antar entitas menggambarkan keterkaitan antara suatu data dengan data lainnya pada struktur database.

1. Relasi User dengan Tdesc(thread description)

User yang merupakan sebuah entitas pada basis data dan juga berperan sebagai aktor memiliki keterkaitan dengan thread, dimana pembuat thread yang akan digunakan sebagai ruang diskusi adalah user, data yang terdapat pada thread adalah nama, dan deskripsi yang nantinya diakses menggunakan nama dari thread tersebut.

2. Relasi User dengan Thread (chat)

Thread(chat) yang merupakan entitas dengan 3 atribut yang berhubungan dengan user, tiga atribut tersebut adalah id, msg (message), dan username.

Atribut yang terdapat pada chat merupakan data yang didapat dari user setiap kali mengirimkan pesan atau chat baru. *Primary key* dari *relationship* ini adalah id yang terdapat pada entitas user.

3. Relasi Tdesc dengan Thread Poster

Thread poster adalah entitas penyimpanan yang beratributkan dua data yaitu uid dan thread name, uid yang terdapat didalam thread poster merupakan data yang didapatkan dari entitas Tdesc.

4. Relasi User dengan Thread Joined

User dalam hal ini sebagai aktor dapat bergabung dengan sebuah thread, data yang berisikan siapa saja yang mengikuti thread apa, disimpan dalam sebuah entitas yaitu Thread Joined, didalam thread joined terdapat dua atribut yaitu uid dan nama thread, uid sama seperti pada entitas Thread Poster didapatkan dari entitas user yang memiliki atribut uid yang merupakan *primary key*.

5. Relasi Tdesc dengan Tjoined

Tdesc adalah sebuah entitas yang menampung deskripsi masing masing thread, relasi dengan Tjoined adalah ketika pengguna bergabung didalam sebuah *thread*. Nama thread yang terdapat pada Tdesc dikirim ke Tjoined.

6. Relasi User dengan Thread Poster

Thread Posted merupakan sebuah entitas yang didalamnya terdapat uid User yang nantinya akan dikirimkan apabila pengguna membuat *thread* baru.

3.7 Perancangan Antar Muka

Perancangan antar muka yang dibuat berdasarkan platform yang akan digunakan yaitu mobile dan sistem operasi Android. Terdapat 12 rancangan antar muka.

1. Rancangan Antarmuka *Splash Screen*

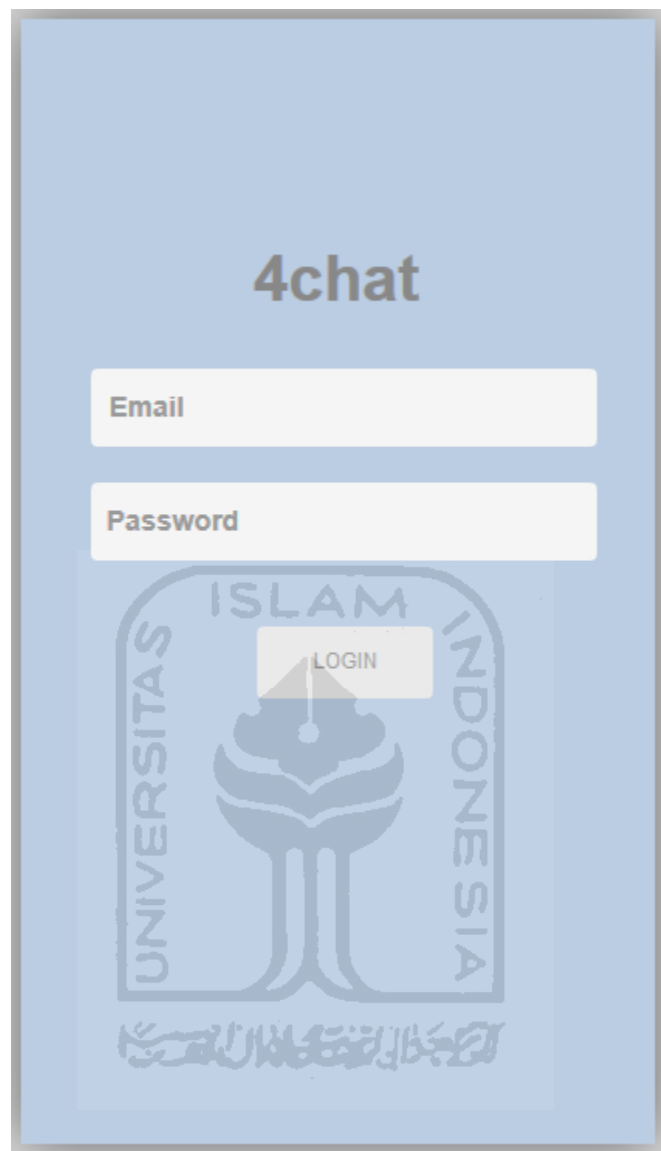
Rancangan antar muka ini akan digunakan ketika user masuk kedalam aplikasi, berisikan logo antarmuka ini akan berlangsung selama dua detik. Rancangan Antarmuka *Splash Screen* dapat dilihat pada gambar 3.15.



Gambar 3.15 Rancangan Antarmuka *Splash Screen*

2. Rancangan Antarmuka *Login*

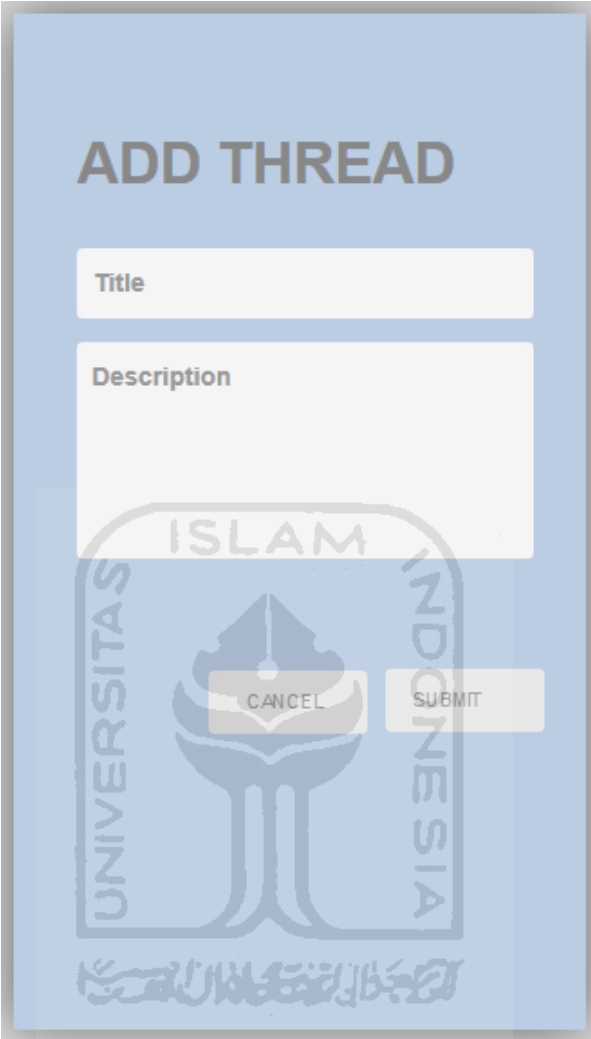
Rancangan antarmuka ini digunakan apabila pengguna ingin menggunakan fitur yang ada aplikasi chatting dengan threading ini. Didalamnya terdapat sebuah form yang mengharuskan user memasukkan email dan password yang sebelumnya sudah didaftarkan melalui form registrasi. Rancangan Antarmuka *Login* dapat dilihat pada gambar 3.16.



Gambar 3.16 Rancangan Antarmuka *Login*

3. Rancangan Antarmuka *Add Thread*

Rancangan antarmuka *add thread* berisikan sebuah form yaitu *Title*, dan *Description*. Didalam perancangan ini ada persyaratan apabila ingin berhasil menambahkan thread kedalam database yaitu, title tidak boleh kurang dari 10 karakter, dan deskripsi tidak kurang dari 15 karakter. *Alert* akan muncul apabila persyaratan tersebut belum dipenuhi dan user memencet tombol *submit*. Rancangan Antarmuka *Add Thread* dapat dilihat pada gambar 3.17.

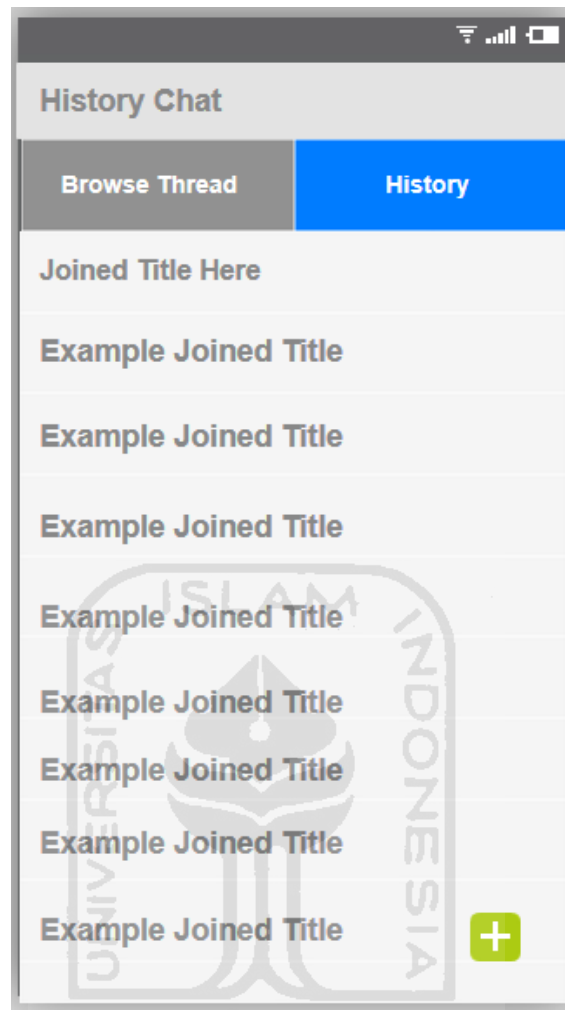


The image shows a mobile application interface for adding a new thread. The title 'ADD THREAD' is displayed at the top in a bold, dark font. Below the title, there is a form with two main sections: a 'Title' input field and a 'Description' text area. At the bottom of the form, there are two buttons: 'CANCEL' and 'SUBMIT'. A large, semi-transparent watermark of the Universitas Indonesia logo is overlaid on the form, featuring the text 'UNIVERSITAS INDONESIA' and 'ISLAM' around a central emblem.

Gambar 3.17 Rancangan Antarmuka *Add Thread*

4. Rancangan Antarmuka *History Chat*

Rancangan antarmuka history chat ada didalam halaman homepage, yang merupakan sebuah *Activity Tab* dimana ada dua *fragment* dalam satu halaman dimana *History Chat* merupakan salah satunya, didalam rancangan antarmuka ini akan terdapat thread apa saja yang sudah user ikuti. Rancangan Antarmuka *History Chat* dapat dilihat pada gambar 3.18.

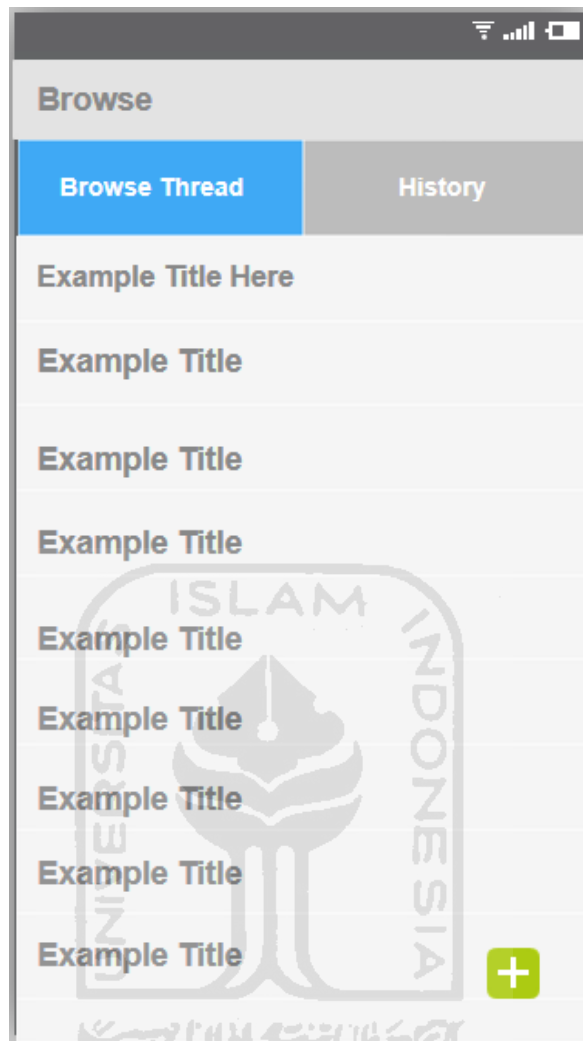


Gambar 3.18 Rancangan Antarmuka *Hisoty Chat*

5. Rancangan Antarmuka *Browse*

Didalam antarmuka *Browse* akan terdapat thread – thread yang sudah dibuat oleh para pengguna yang sudah bergabung didalam aplikasi ini. Sama dengan *History Chat*, *Browse Chat* juga berada dalam *tab activity* yang ada didalam halaman awal aplikasi.

Ketika user ingin bergabung kedalam thread maka pengguna harus memilih salah satu judul thread yang tersedia di tampilan *Browse History* lalu memilih join. Apabila berhasil maka pengguna akan diarahkan kehalaman *Chat* dan judul yang dipilih akan masuk kehalaman *History Chat*. Rancangan Antarmuka *Browse* dapat dilihat pada gambar 3.19.



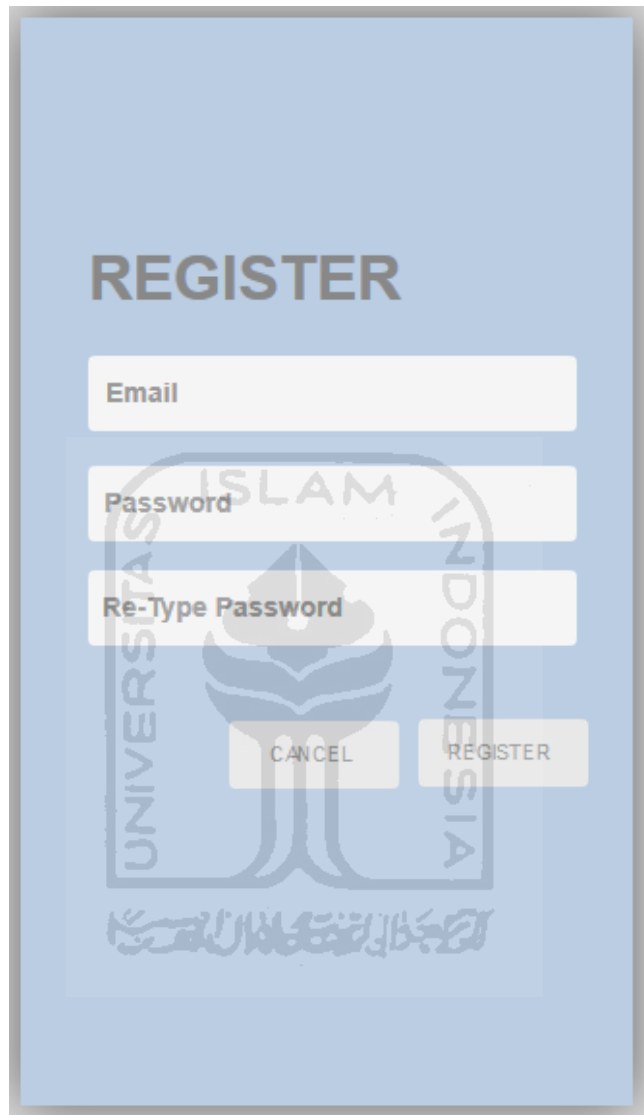
Gambar 3.19 Rancangan Antarmuka *Browse*

6. Rancangan Antarmuka *Register User*

Register User merupakan aktifitas yang digunakan oleh user apabila ingin mendaftarkan emailnya kedalam aplikasi chatting dengan fitur threading ini. Tampilannya berupa form yang berisikan email, password, dan re-type password yang harus diisi oleh user berdasarkan dari password yang sudah diketik sebelumnya.

Adapun persyaratan user apabila ingin melakukan registrasi adalah email yang pengguna masukkan belum pernah didaftarkan sebelumnya, apabila sudah maka didalam antarmuka akan memunculkan pesan bahwa proses *registrasi* gagal. Selain itu password sendiri tidak boleh kurang dari

6 karakter. Rancangan Antarmuka *Register User* dapat dilihat pada gambar 3.20.

The image shows a mobile application registration screen with a light blue background. At the top, the word "REGISTER" is displayed in a large, bold, dark grey font. Below the title, there are three white input fields with rounded corners. The first field is labeled "Email", the second "Password", and the third "Re-Type Password". At the bottom of the form, there are two buttons: "CANCEL" on the left and "REGISTER" on the right. A large, semi-transparent watermark of the logo for Universitas Islam Indonesia is centered over the form. The logo features a stylized green and blue emblem with the text "UNIVERSITAS ISLAM INDONESIA" and Arabic calligraphy at the bottom.

Gambar 3.20 Rancangan Antarmuka *Register User*

7. Rancangan Antarmuka *Change Password*

Rancangan Antarmuka *Change Password* merupakan antarmuka yang digunakan oleh pengguna apabila ingin merubah passwordnya. Rancangan antarmuka ini terdiri dari dua buah *EditText* yang berisikan *New Password* atau merupakan password baru yang dikehendaki oleh pengguna dan *Re-Type Password* yang merupakan konfirmasi dari sandi yang

sebelumnya sudah ketik oleh pengguna. Rancangan Antarmuka *Change Password* dapat dilihat pada gambar 3.21.

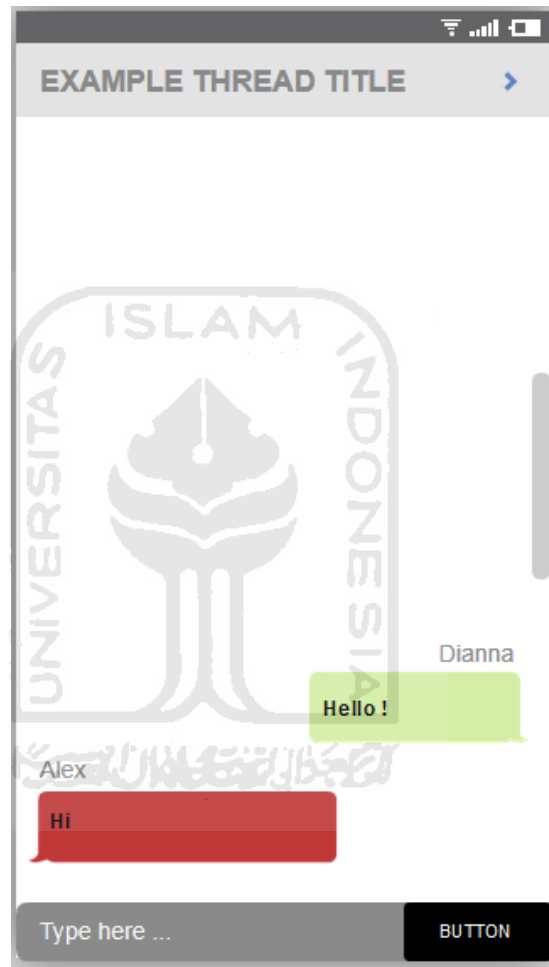
Gambar 3.21 Rancangan Antarmuka *Change Password*

8. Rancangan Antarmuka *Chatting*

Rancangan antarmuka *chatting* merupakan antarmuka yang digunakan oleh pengguna untuk melakukan aktifitas mengirimkan pesan didalam sebuah thread kepada pengguna lain yang berlangganan didalam thread tersebut.

Antarmuka terdiri dari judul yang merupakan *title* dari thread dimana pengguna sedang melakukan chat, *chat bubble* yang merupakan isi pesan baik dari pengguna itu sendiri maupun pengguna lain, perbedaan

tampilan *chat bubble* terletak dari siapa yang memiliki pesan tersebut, apabila pesan itu dikirim oleh pengguna itu sendiri maka akan muncul disebelah kanan, dan apabila pesan itu dikirim oleh pengguna lain maka akan muncul di sebelah kiri. Rancangan Antarmuka *Chatting* dapat dilihat pada gambar 3.22.

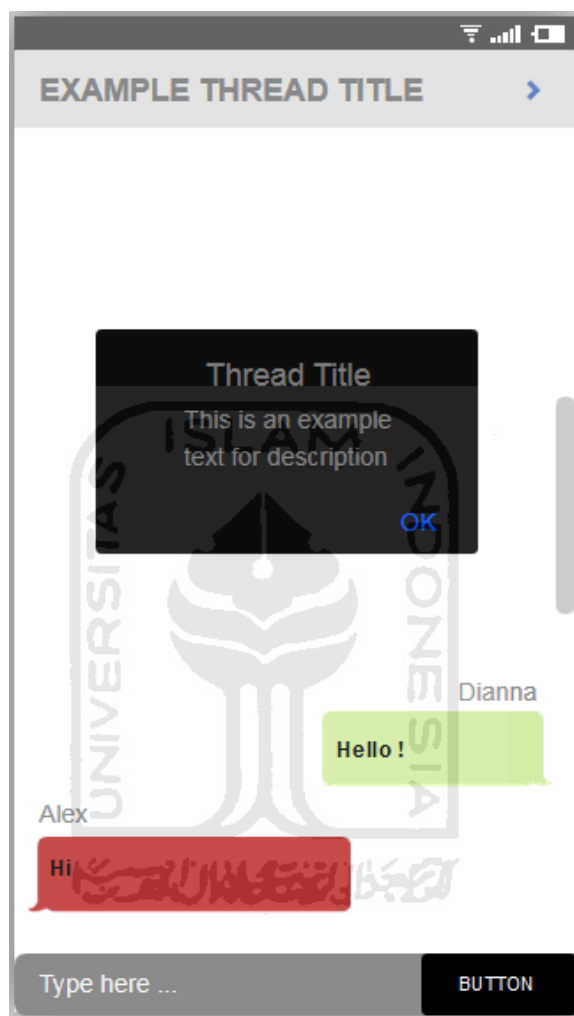


Gambar 3.22 Rancangan Antarmuka *Chatting*

9. Rancangan Antarmuka *Expand Title And Description*

Rancangan antarmuka *expand title and description* merupakan rancangan yang muncul apabila pengguna ingin melihat judul dan deskripsi dari sebuah thread, ada dua cara yang dapat pengguna pergunakan apabila ingin mengakses antarmuka ini, yaitu melalui halaman *browse title* yang berada pada *tab activity* dihalaman muka, yang kedua dengan menekan

tombol expand yang berbentuk (>) didalam halaman chatting masing masing thread.



Gambar 3.23 Rancangan Antarmuka *Expand Title And Description*

Gambar 3.23 merupakan rancangan antarmuka *expand title description* yang merupakan sebuah fitur yang ada didalam thread, pengguna dapat melihat judul dan deskripsi dari thread tersebut. Untuk mengakses tampilan ini dari dalam thread terlebih dahulu pengguna harus bergabung dengan sebuah thread.

Disisi lain apabila pengguna ingin melihat deskripsi tanpa harus bergabung dengan thread melalui antarmuka *browse title* yang terdapat dalam *tab pages*. Apabila pengguna ingin melihat deskripsi melalui

tampilan *browse title* user hanya perlu memilih judul mana yang ingin dilihat deskripsinya. Rancangan antarmukanya dapat dilihat pada gambar 3.24.

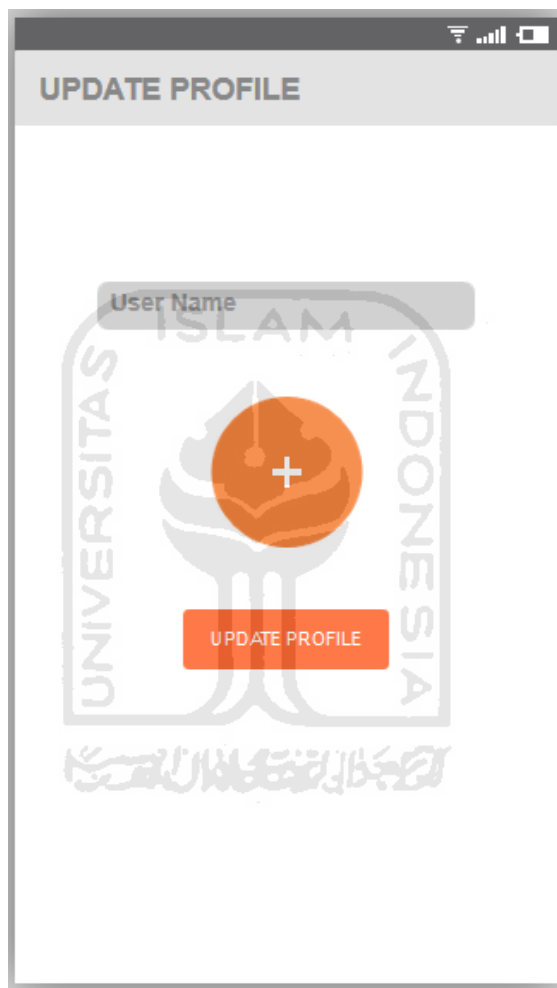


Gambar 3.24 Rancangan Antarmuka *Expand Title And Description 2*

10. Rancangan Antarmuka *Update Profile*

Rancangan antarmuka *update profile* berisikan form yang terdiri dari username yang merupakan sebuah *EditText* atau *text field* yang nantinya akan digunakan oleh pengguna sebagai identitas pada aktifitas Chat didalam thread, *ImageButton* sebuah tombol berbentuk lingkaran yang

merupakan *image picker* dimana apabila pengguna menggunakan tombol tersebut akan diarahkan ke sebuah jendela gallery untuk memilih gambar, dan apabila user memilih sebuah gambar, maka *ImageButton* tersebut menampilkan gambar yang pengguna pilih. Rancangan Antarmuka *Update Profile* dapat dilihat pada gambar 3.25.

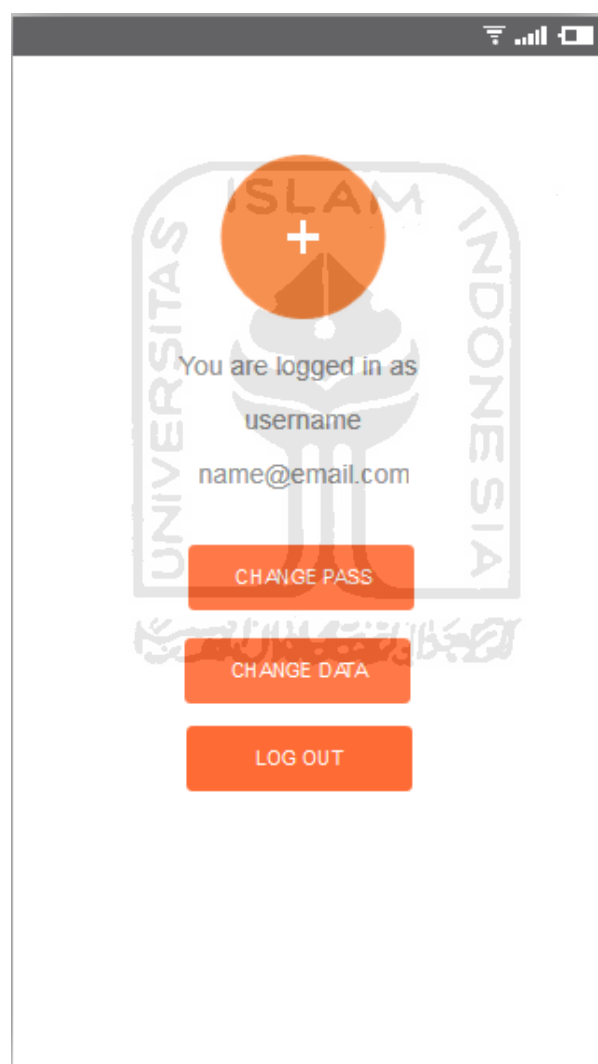


Gambar 3.25 Rancangan Antarmuka *Update Profile*

11. Rancangan Antarmuka *Profile Page*

Dalam rancangan antarmuka *profile page* terdapat sebuah *imageView* yang merupakan gambar dari photo profil yang sebelumnya sudah pengguna unggah kemudia diunduh oleh aplikasi dan ditampilkan dalam bentuk lingkaran. Selain itu ada beberapa data seperti *username* dan email pengguna sebagai data identitas pengguna.

Didalam rancangan antarmuka ini juga terdapat 3 tombol yang memiliki fungsi asing masing, yaitu tombol *change pass* yang merupakan tombol untuk berpindah ke halaman *Change Password* lalu *change data* yang akan berpindah kehalaman *Update Profile* untuk memperbaharui data pengguna, dan kemudia *log out* yang merupakan tombol supaya pengguna dapat keluar dari akun yang sedang digunakan. Rancangan Antarmuka *Profile Page* dapat dilihat pada gambar 3.26.



Gambar 3.26 Rancangan Antarmuka *Profile Page*

BAB IV

HASIL DAN PEMBAHASAN

4.1 Batasan Implementasi

Dalam implementasinya, Aplikasi Chatting Dengan Fitur Threading, terdapat beberapa batasan, berikut merupakan batasan – batasannya:

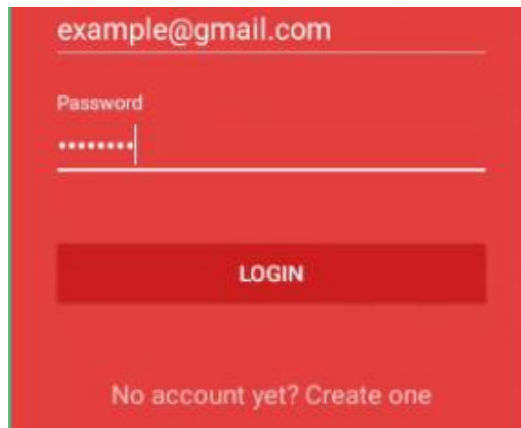
1. Pengguna tidak dapat bergabung didalam thread apabila belum mengisikan data berupa *username* dan *profile picture*. Karena didalamnya perlu identitas untuk mengetahui siapa pengirim pesan.
2. Aplikasi Chatting Dengan Fitur Threading hanya dapat digunakan diperangkat dengan sistem operasi Android berversi 4.0.3 (*Icecream Sandwich*) atau lebih.
3. Pengguna perangkat lunak tidak dapat menghapus thread yang sudah dibuat, dan hanya dapat menghapus thread yang sudah diikuti.
4. Pengguna harus me *restart* aplikasi apabila setelah login apabila ingin mendapatkan uid baru.

4.2 Hasil Implementasi

Hasil Implementasi pada bab ini merupakan realisasi dari perancangan Aplikasi yang sudah dibahas di bab sebelumnya. Implementasi dilakukan untuk memastikan bahwa perangkat lunak dirancang secara baik. Berikut hasil implementasi dari Aplikasi Chatting Dengan Fitur Threading:


4.2.1 Hasil Implementasi Kebutuhan Registrasi User

Kebutuhan registrasi user diimplementasikan dengan tujuan supaya pengguna yang baru memasang aplikasi chatting dengan fitur threading dapat mengakses fungsi fungsi lain yang membutuhkan autentikasi terlebih dahulu. Pada pengimplementasinya menggunakan salah satu fitur yang disediakan oleh Firebase Cloud Service yaitu Firebase Authentication dimana Firebase menyediakan library untuk autentikasi *user* melalui *email* dan *password*.

A screenshot of a login form on a red background. At the top, there is a text input field containing 'example@gmail.com'. Below it is a 'Password' label and a password input field with seven dots. A dark red button labeled 'LOGIN' is centered below the password field. At the bottom, there is a link that says 'No account yet? Create one'.

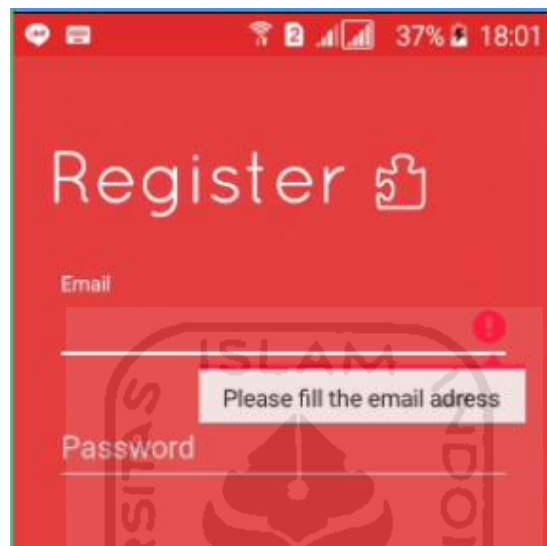
Gambar 4.1 Button “No account yet? Create one”

Halaman Registrasi *User* diakses melalui Halaman Login menggunakan tombol ““No account yet? Create one”” seperti pada gambar 4.1. Didalam tampilan Registrasi User terdapat 3 *EditText* yang masing masing merupakan identitas untuk kemudian digunakan bagi pengguna untuk melakukan *Login* seperti pada gambar 4.2.

A screenshot of a user registration form on a red background. The word 'Register' is displayed at the top in white with a small icon. Below it is a watermark logo for 'UNIVERSITAS ISLAM INDONESIA'. The form contains three text input fields: 'Email', 'Password', and 'Re-Type Password'. A dark red button labeled 'REGISTER' is centered at the bottom.

Gambar 4.2 Tampilan Registrasi User

Dalam proses pengisian identitas yang berupa email, password, dan *re-type password* ada beberapa persyaratan seperti pengguna tidak boleh mengosongkan *field* username, aplikasi akan memberikan peringatan berupa *error text* pada *field* yang tidak sesuai dengan persyaratan perangkat lunak (gambar 4.3).

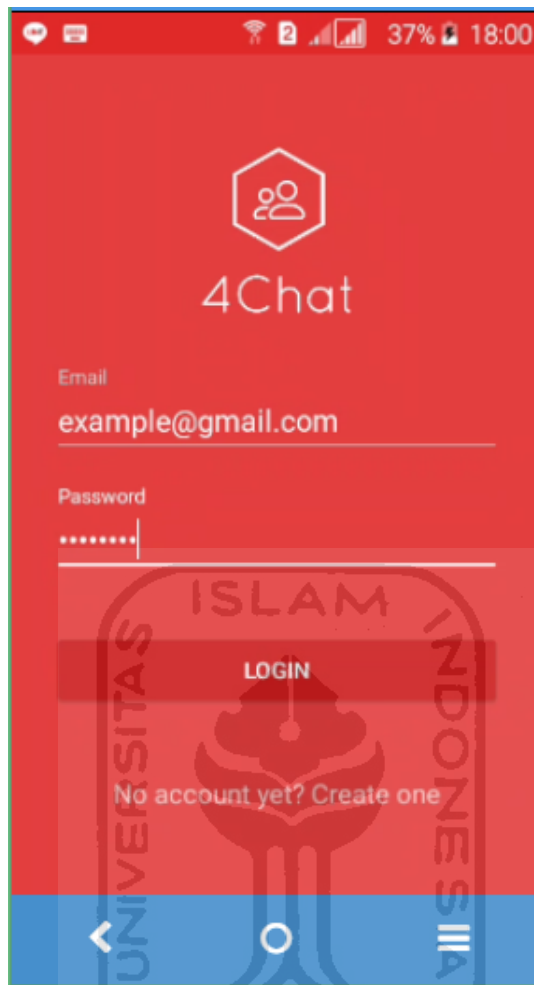


Gambar 4.3 *Error handling* pada halaman Registrasi User

Apabila pengguna berhasil melakukan proses registrasi berikutnya aplikasi akan mengarahkan ke tampilan *Login*.

4.2.2 Hasil Implementasi Kebutuhan Login

Login dalam Aplikasi Chatting Dengan Fitur Threading digunakan untuk mengautentifikasi user dengan, fungsi autentifikasi disini mengaplikasikan fitur dari Firebase Autentifikasi Login, sesuai dengan proses registrasi. Dengan pemanggilan method *signInWithEmailAndPassword()* pada variabel *FirebaseAuth* aplikasi akan mengirimkan data yang pengguna masukkan dan diautentikasi.



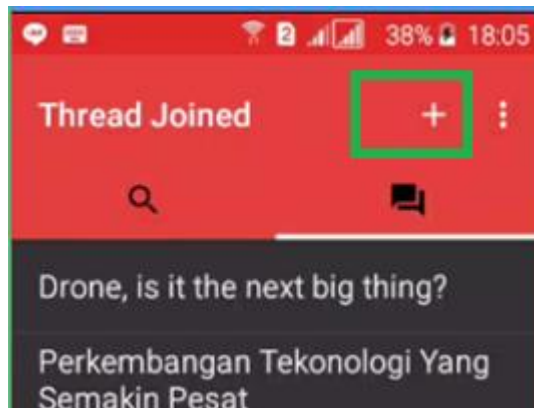
Gambar 4.4 Tampilan *Login*

State Login merupakan posisi dimana apakah user sudah melakukan login atau belum. Aplikasi dapat mengetahui *State Login* tersebut melalui method *AuthStateListener()* yang dipanggil melalui object *FirebaseAuth.AuthListener*. apabila state dari user tidak sama dengan *null* maka aplikasi akan mengarahkan tampilan ke tabpages yang merupakan halaman awal aplikasi setelah pengguna login, sedangkan ketika user sama dengan *null* pada saat aplikasi dijalankan akan memunculkan tampilan login seperti pada gambar 4.4.

4.2.3 Hasil Implementasi Kebutuhan Buat Thread

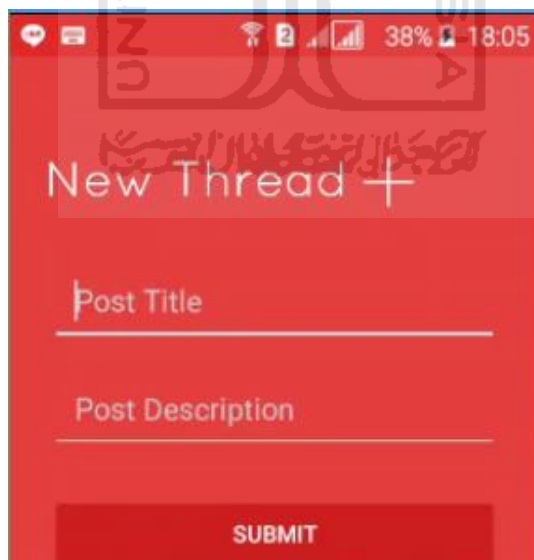
Membuat Thread merupakan sebuah fungsi yang dapat dilakukan apabila pengguna sudah berada dalam posisi *Login*. Untuk mengakses tampilan ini pengguna

dapat menggunakan tombol (+) yang berada pada *Action Bar* ditampilkan awal ketika user berhasil login kedalam aplikasi seperti pada gambar 4.5.



Gambar 4.5 Tombol *Add Thread*

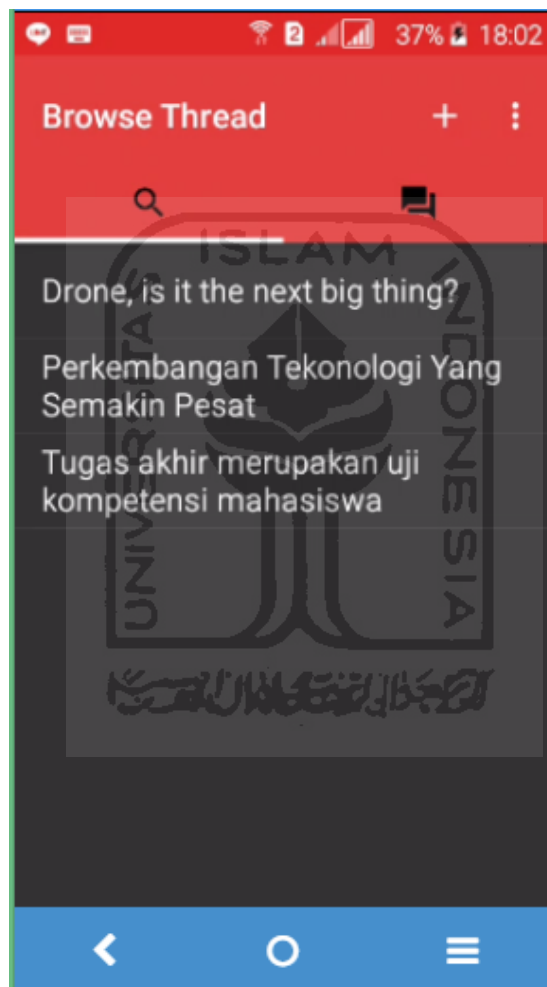
Didalam halaman *add thread* atau membuat thread, terdapat sebuah form yang berisikan dua buah field yaitu *Title* dan *Post Description* yang merupakan judul dan deskripsi dari thread tersebut. halaman *add thread* dapat dilihat pada gambar 4.6.



Gambar 4.6 Tampilan *Add Thread*

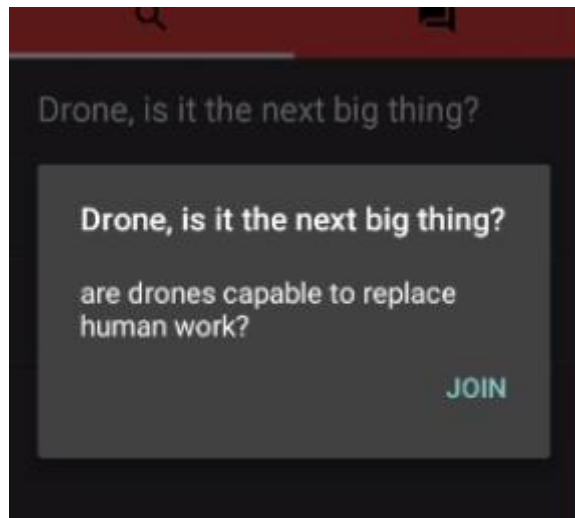
4.2.4 Hasil Implementasi Kebutuhan Lihat Thread

Kebutuhan ini meliputi proses menampilkan semua thread – thread yang tersedia didalam *database* yang sudah dibuat oleh pengguna lain. Tampilan *List* dari thread tersebut akan ditampilkan dalam bentuk *listView* dengan Judul Thread sebagai isinya pada *fragment Browse Thread* yang merupakan tampilan tab *BrowseTitle* pada halaman *tabpages* seperti pada gambar 4.7.



Gambar 4.7 Tampilan *Browse Title*

List yang terdapat pada tampilan ini digunakan untuk menampilkan *Alert Dialog* yang berisikan Judul, Deskripsi Thread, dan Tombol Join yang digunakan untuk bergabung dalam sebuah percakapan, seperti pada Gambar 4.8. Pengguna dapat mengaksesnya dengan menyentuh salah satu judul.



Gambar 4.8 Alert Dialog pada Browse Title

4.2.5 Hasil Implementasi Kebutuhan Lihat Chat

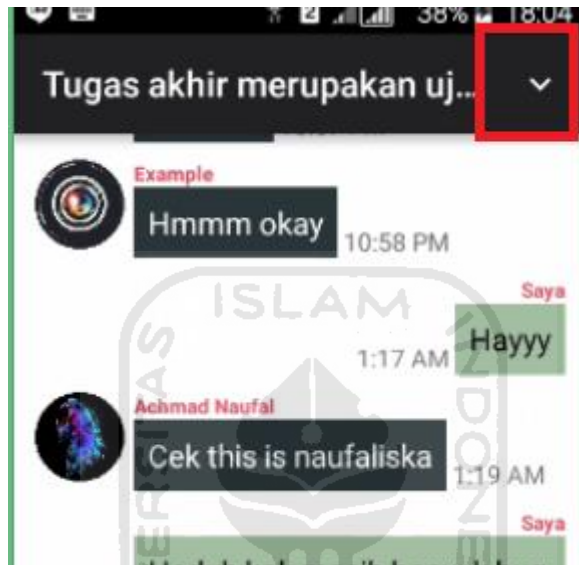
Kebutuhan melihat *chat* (gambar 4.9) meliputi fitur dimana pengguna dapat melihat *history* atau kumpulan pesan yang sudah dikirim oleh pengguna lain sebelumnya apabila ada. Tampilan ini dapat diakses langsung ketika pengguna menggunakan fitur *join* pada *Dialog Alert* yang berada pada tampilan *Browse Title* seperti pada gambar 4.8.



Gambar 4.9 Tampilan Chat didalam Thread

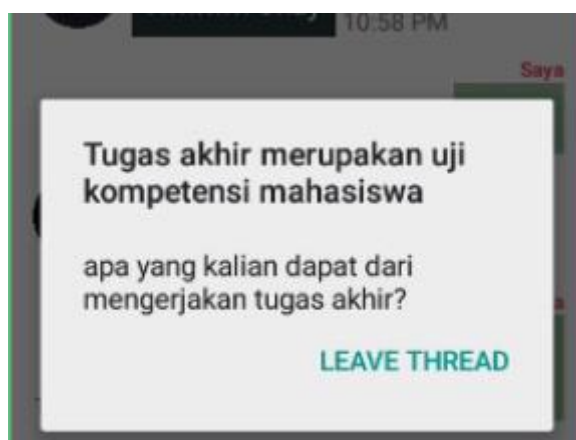
4.2.6 Hasil Implementasi Kebutuhan Lihat Deskripsi Thread

Melihat deskripsi thread dapat dilakukan dengan dua cara, yaitu dengan melihat melalui tampilan Browse History seperti pada gambar 4.8. namun juga dapat dilihat didalam Thread itu sendiri pada tampilan *Chatting* dengan menggunakan tombol (∇) pada Action Bar seperti pada Gambar 4.10.



Gambar 4.10 Tombol *Expand Description*

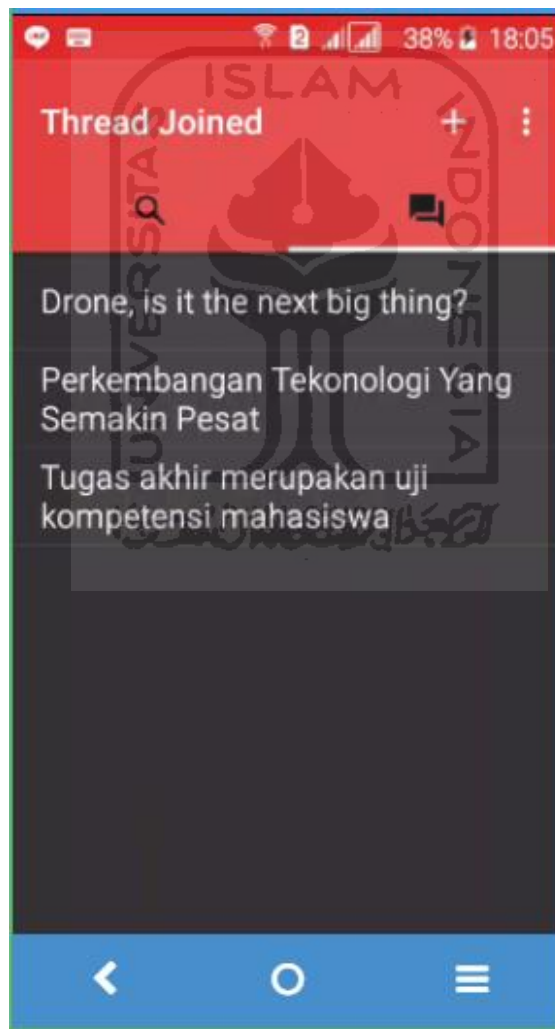
Tombol *Expand Description* akan memunculkan *Dialog Alert* yang berisikan Judul, Deskripsi Thread dan juga sebuah tombol "Leave Thread" yang merupakan fungsi untuk menghapus thread yang diikuti seperti pada gambar 4.11.



Gambar 4.11 *Dialog Alert* pada halaman *Chat*

4.2.7 Hasil Implementasi Kebutuhan *Join Thread*

Kebutuhan *Join Thread* merupakan meliputi proses bagaimana pengguna bergabung dalam sebuah thread, seperti pada gambar 4.8, pengguna dapat menambahkan thread kedalam daftar thread yang sudah diikuti dengan menggunakan tombol “*Join*” yang ada pada *Dialog Alert* seperti pada gambar 4.8 kemudian aplikasi memasukkan kedalam database dan memunculkannya pada fragment *Chat History* yang muncul di tab pages apabila pengguna memilih tab *Thread Joined*. Hasil Implementasi Kebutuhan *Join Thread* dapat dilihat pada gambar 4.12.



Gambar 4.12 Tampilan *Tab Joined Thread*

4.2.8 Hasil Implementasi Kebutuhan *Leave Thread*

Leave Thread merupakan sebuah fungsi untuk menghapus sebuah thread pada daftar thread yang sudah diikuti. Pengguna dapat menggunakan fitur ini didalam thread yang ingin dihapus dan menekan tombol *expand description* (▾), lalu aplikasi akan memunculkan *Dialog Alert* seperti pada gambar 4.11 kemudian pengguna dapat menekan tombol “*Leave Thread*” untuk menghapus thread tersebut dari daftar thread yang sudah diikuti.

4.2.9 Hasil Implementasi Kebutuhan *Kirim Chat*

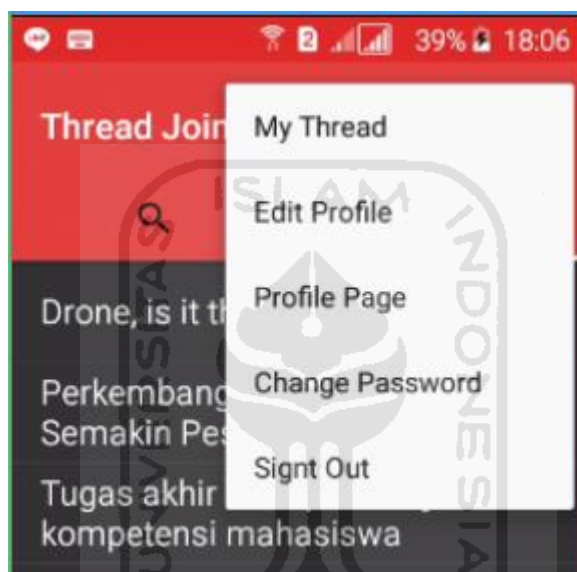
Apabila user telah bergabung didalam Thread pengguna dapat mengakses thread tersebut langsung setelah menekan tombol “*Join*” didalam *Dialog Alert* seperti pada gambar 4.8 maupun melalui tab *Thread Joined* seperti pada gambar 4.12. Didalam setiap chat yang dikirim terdapat *Key* yang di *generate* setiap pesan dikirim yang didapatkan dengan method *getKey()*. Didalam *Key* akan terdapat data dari chat berupa *msg* yang merupakan isi pesan, *username* yang merupakan identitas nama pengguna, *zUid_chat* yang merupakan *id* dari setiap pengguna, *zzPhoto* yang berisikan URL(*Uniform Resource Locator*) dari gambar yang disimpan pada *Firebase Storage*, dan *zzzTimestamp* yang merupakan waktu chat dimasukkan berbentuk *unix-timestamp*, gambar dari databasenya dapat dilihat pada gambar 4.13.



Gambar 4.13 Contoh Data *Chat*

4.2.10 Hasil Implementasi Kebutuhan *Reset Password*

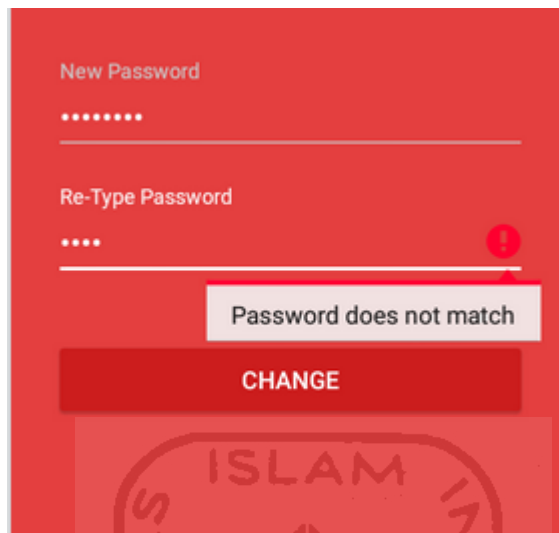
Reset Password (gambar 4.15) merupakan fitur dimana pengguna dapat mengganti password lamanya dengan password yang baru, autentikasi dilakukan dengan menyocokkan kedua field yang terdapat pada form reset password. Untuk mengakses halaman *Reset Password* pengguna harus membuka *menu option* yang terdapat pada halaman *tabpages* pada halaman awal dan memilih *Change Password*.



Gambar 4.14 Tampilan *Menu Option*

Gambar 4.15 Form Reset Password

Apabila isi kedua field tersebut tidak sama akan muncul pesan error “*password doesn't match*” seperti pada gambar 4.16.

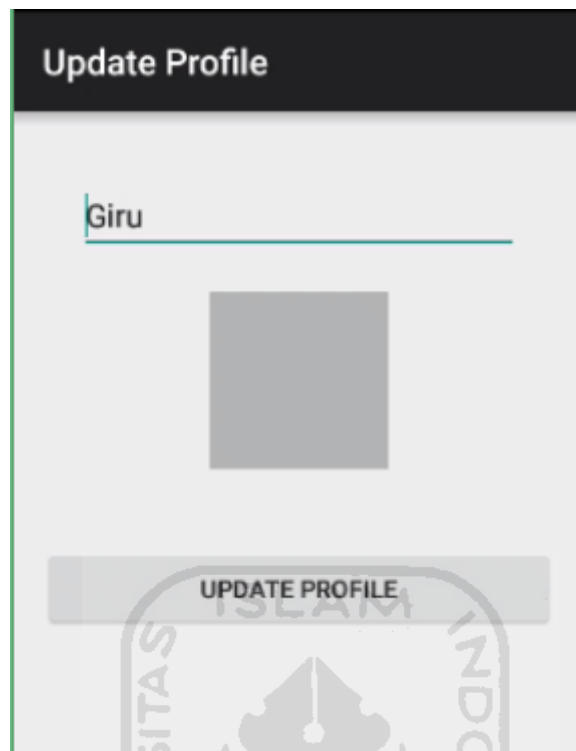


Gambar 4.16 *Error Alert* ketika password tidak sama

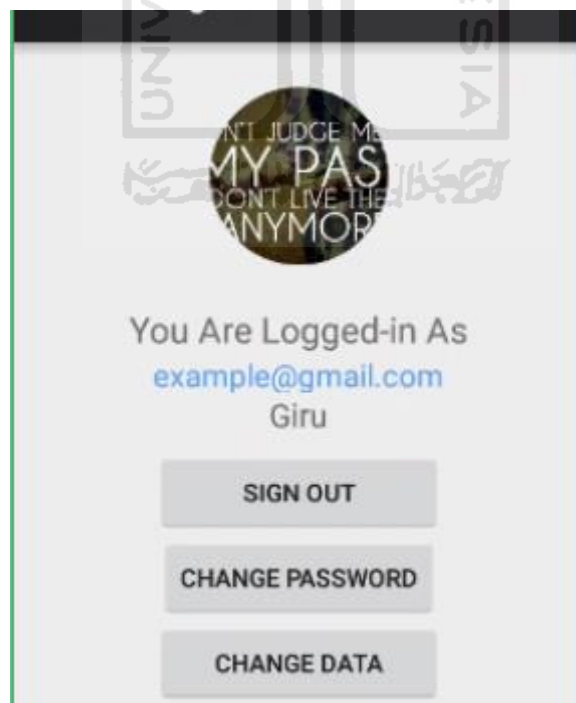
4.2.11 Hasil Implementasi Kebutuhan Kelola Akun

Kebutuhan kelola akun merupakan kebutuhan sistem untuk mengatur atau mengganti identitas pengguna baik berupa username maupun foto profil. Untuk mengubah identitas pengguna dapat masuk kehalaman *Edit Profile* yang dapat diakses melalui *Option Menu* seperti pada gambar 4.14 dan memilih *Edit Profile* yang akan memunculkan halaman seperti pada gambar 4.17.

Pengguna juga dapat melihat profile yang sudah diatur sebelumnya pada halaman *Profile Page* yang dapat diakses sama seperti *Edit Profile* melalui *Option Menu* seperti pada gambar 4.14, yang akan memunculkan halaman *Profile Page* seperti pada gambar 4.18.



Gambar 4.17 Halaman *Edit Profile*



Gambar 4.18 Halaman *Profile Page*

4.2.12 Hasil Implementasi Kebutuhan *Sign – Out*

Sign – Out merupakan fitur dari aplikasi yang memungkinkan pengguna untuk keluar dari akun yang sudah melakukan login sebelumnya pada sebuah *mobile device*. Untuk mengakses fitur ini pengguna dapat membuka *Menut Option* seperti pada gambar 4.14 lalu memilih *Sign Out* ataupun melalui halaman *Profile Page* dengan menekan tombol *Sign Out* seperti pada gambar 4.18.

4.3 Pengujian Perangkat Lunak

Pengujian perangkat lunak ditujukan untuk menentukan apakah aplikasi yang telah diimplementasikan sesuai dengan hasil perancangan. Pengujian juga ditujukan untuk menentukan kualitas dari *software* dengan cara mengujinya dengan metode sesuai dengan batasan implementasinya.

4.3.1 Pengujian Menggunakan *Black Box Testing*

Tahap pengujian Aplikasi Chatting Dengan Fitur Threading dilakukan dengan menggunakan metode *Black Box Testing*, *Black Box Testing* merupakan metode pengujian yang memfokuskan pada keperluan fungsional dari sebuah perangkat lunak. Pengujian dilakukan oleh seorang yang sudah biasa menggunakan aplikasi serupa seperti aplikasi *messaging* pada perangkat mobile. Hasil pengujian menggunakan metode *Black Box Testing* dapat dilihat pada tabel 4.1.

Tabel 4.1 Tabel Pengujian *Blackbox – Testing*

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	<i>Register User</i>	Pengguna berhasil mendaftarkan emailnya kedalam sistem dan aplikasi memunculkan halaman <i>Edit</i>	Tampil halaman <i>Edit Profile</i>	Valid

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
		<i>Profile</i> . Pengguna terlebih dahulu memasukkan data berupa email, username, password dan re-type password.		
2	<i>Login</i>	Pengguna berhasil masuk kehalaman <i>tabpages</i> atau halaman <i>home</i> . Pengguna terlebih dahulu memasukkan data login berupa email dan password	Pengguna berhasil masuk dan aplikasi memunculkan halaman <i>tabpages</i>	Valid
3	Buat Thread	Thread berhasil dibuat dan dimunculkan pada halaman <i>tabpages</i> di tab <i>Browse Title</i> . Terlebih dahulu pengguna diharapkan untuk memasukkan data yaitu judul dan deskripsi thread	Judul yang baru ditambahkan terdapat pada tab <i>Browse Title</i>	Valid

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
4	Lihat Thread	Aplikasi berhasil memunculkan list thread yang sudah dibuat oleh pengguna. User harus ada dalam keadaan sudah login dan berada dalam tab <i>Browse Title</i>	Muncul semua daftar judul yang tersedia didatabase	Valid
5	Lihat <i>Chat</i>	Aplikasi data menampilkan <i>chat</i> yang sudah dikirim oleh pengguna sesuai dengan thread yang sedang dimasuki. Pengguna terlebih dahulu harus bergabung dithread tersebut dan masuk kedalam threadnya,	Berhasil memunculkan chat yang tersedia didalam tab tersebut	Valid
6	Lihat Deskripsi Thread	Aplikasi berhasil memunculkan deskripsi thread sesuai dengan thread yang dipilih. Pengguna terlebih dahulu memilih	Aplikasi memunculkan deskripsi sesuai dengan judul thread yang dipilih	Valid

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
		thread mana yang ingin dilihat deskripsinya		
7	<i>Join Thread</i>	Pengguna berhasil memasukkan thread yang ingin diikuti dan aplikasi menampilkannya pada tab <i>Thread Joined</i> . Pengguna terlebih dahulu harus bergabung pada sebuah thread dengan menggunakan fitur <i>join</i> .	Aplikasi menampilkan halaman chat dan menambahkan judul thread ke halaman <i>Thread Joined</i>	Valid
8	<i>Leave Thread</i>	Pengguna berhasil keluar ke halaman <i>tabpages</i> dan menghapus judul thread yang diinginkan pada tab <i>Thread Joined</i> , pengguna terlebih dahulu harus bergabung thread dan menggunakan fitur <i>Leave Thread</i>	Tampil halaman awal atau <i>tabpages</i> dan menghapus thread yang diinginkan pada tab <i>Thread Joined</i>	Valid

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
		yang ada pada <i>Alert Dialog</i> didalam Chat		
9	Kirim <i>Chat</i>	Pengguna berhasil mengirimkan pesan, dan aplikasi memunculkannya pada daftar chat yang ada didalam percakapan.	Pesan terkirim dan aplikasi memunculkan data pesan.	Valid
10	<i>Reset Password</i>	User berhasil mengubah <i>password</i> , dan berhasil <i>login</i> mengguna <i>password</i> baru. Pengguna terlebih dahulu mengisikan form <i>reset password</i> secara benar	Pengguna berhasil merubah <i>password</i> dan login dengan <i>password</i> baru	Valid
11	Kelola Akun	Pengguna dapat mengganti data akun yang berupa <i>username</i> dan foto profil Kemudia aplikasi dapat menampilkan data terbaru pada halaman <i>Profile</i>	Pada halaman profile page muncul data yang diperbaharui baik <i>username</i> dan foto profil	Valid

No	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
		<i>Page</i> . Pengguna harus mengisikan <i>username</i> dan foto profil pada halaman <i>Edit Profile</i>		
12	<i>Sign - out</i>	Pengguna berhasil melakukan sign – out dan pindah ke halaman <i>Login</i> .	Pengguna berhasil keluar dan aplikasi memunculkan halaman login	Valid

4.3.2 User Acceptance Test (UAT)

Pengujian dengan UAT merupakan tahap akhir yang ditujukan untuk menentukan apakah aplikasi sudah sesuai dengan kebutuhan pengguna secara keseluruhan baik dari fungsi maupun tampilan aplikasi (Matthew Setter, usersnap.com/blog/user-acceptance-testing-right/). Pengujian ini juga dimaksudkan untuk mendapatkan umpan balik dari beberapa pengguna Aplikasi Chatting Dengan Fitur Threading sekaligus mengetahui komptaibilitas dari masing masing perangkat dan versi Android yang digunakan. Berikut kesimpulan yang didapat dari hasil pengujian yang dilakukan terhadap 4 *user* :

1. Fungsionalitas

Sisi fungsionalitas merangkup masalah kebutuhan yang tersedia pada aplikasi, seperti login, menambah *thread* dan sebagainya. Dari 4 pengguna yang dijadikan sebagai responden dalam pengujian menggunakan metode UAT yang sifatnya terbuka ini dapat disimpulkan bahwa semua fitur berjalan dengan baik pada masing masing perangkat bergerak yang digunakan untuk mencoba aplikasi ini. Hasil dari kuisisioner pengujian menggunakan metode UAT dapat dilihat pada lembar lampiran.

2. Kompatibilitas

Dari sisi kompatibilitas pengguna mencoba memasang dan menggunakan aplikasi pada perangkat yang berbeda beda. Hasil dari pengujian pada perangkat dapat dilihat pada tabel 4.2.

Tabel 4.2 Tabel Pengujian Kompatibilitas

No	Perangkat	Status
1	Xiami Redmi Note Pro 3	Kompatibel
2	Samsung Galaxy S5	Kompatibel
3	Oppo A37	Kompatibel
4	Sony Xperia J	Kompatibel
5	Asus Zenfon Go	Kompatibel

Namun beberapa pengguna mengalami isu pada tampilan, seperti diperangkat Xiaomi Redmi Note 3 dengan layar 5.5 Inch pada fitur *chat* peletakan pesan dikirim dan diterima tidak sesuai seperti pada perangkat lain, namun secara keseluruhan tidak mengganggu fungsionalitas penggunaan aplikasi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan perancangan, implementasi dan pengujian sistem yang telah dilakukan, maka penulis dapat disimpulkan bahwa:

1. Aplikasi Chatting Dengan Fitur Threading telah dikembangkan dan diuji dengan beberapa metode pengujian, dan mendapatkan hasil bahwa aplikasi yang dibangun sudah sesuai dengan perancangan yang dilakukan baik dari sisi fungsionalitas maupun tampilan.
2. Dari hasil pengujian yang dilakukan menghasilkan beberapa hal yang menjadi pertimbangan untuk pengembangan selanjutnya, baik dari pengembangan aplikasi disisi fungsionalitas, maupun non fungsionalitas.

5.2 Saran

Dari sistem yang telah dibuat masih terdapat banyak kekurangan dan kelemahan yang masih dapat dikembangkan lebih lanjut sehingga dapat meningkatkan kualitas dan kinerja dari perangkat lunak agar menjadi lebih baik dalam penggunaannya. Jadi penulis menyarankan:

1. Pada pengembangan berikutnya aplikasi dapat memiliki fungsi pencarian yang sehingga memudahkan pengguna baik dalam pencarian thread maupun *chat*.
2. Terdapat fungsi notifikasi yang dapat diatur sehingga pengguna dapat memilih thread mana yang ingin diaktifkan fungsi notifikasinya.
3. Penyampaian pesan tidak hanya berupa teks namun juga dapat melalui media lainnya, seperti gambar, maupun *attachment* yang berupa file.
4. Adanya fungsi kategorisasi sehingga pengguna dapat mencari berdasarkan kategori pembahasan yang diinginkan.

DAFTAR PUSTAKA

Gram, David. Effective Group Communication Process, Diakses 7 Desember, 2016 dari <http://smallbusiness.chron.com/effective-group-communication-processes-3187.html>

Setter, Matthew. User Acceptance Test – How To Do It Right!, Diakses 7 Desember, 2016 dari usersnap.com/blog/user-acceptance-testing-right/



LAMPIRAN




Rekap Kuisioner User Acceptance Test


No	Umur	Pertanyaan	Jawaban
1	Fakhrus Syakirin	Apa jenis handphone anda (merk dan tipe) dan versi berpada android yang terpasang pada android anda? (bisa lebih dari satu)	Xiaomi redmi note 3 pro, Samsung Galaxy S5
		Dari pengalaman menggunakan Aplikasi Chatting Dengan Fitur Threading di perangkat mobile anda, adakah isu pada tampilan (peletakan tampilan yang tidak sesuai) yang mengganggu penggunaan aplikasi?	Ada, mungkin tidak terlalu mengganggu seperti peletakan chat bubble yg kurang rapi dan pewarnaan aplikasi yg kurang sesuai saja, mungkin bisa diperbaiki supaya nyaman dan menarik bagi pengguna. Tapi secara fungsionalitas semua sdh bagus
		Apa pendapat anda mengenai fitur untuk menambah Thread?	Mempermudah dalam menemukan solusi dan bahan pembicaraan karena dari thread yg dibuat, pengguna bisa saling berdiskusi
		Bagaimana penilaian anda terhadap fitur 'Browse Thread' dan 'Join Thread'	Sudah cukup baik krn dapat mencari topik pembicaraan dan dapat menjadi filter agar suatu topik pembicaraan tidak bercampur dengan topik pembicaraan lainnya

		Kritik dan saran mengenai Aplikasi Chatting Dengan Fitur Threading	Sudah cukup baik tetapi perlu adanya notifikasi pemberitahuan bila ada pesan terbaru dan perlu adanya simpulan maupun komentar yg bagus dari setiap topik pembicaraan agar topik tidak dibuka selamanya tetapi ada saatnya topik ditutup jika sudah mendapatkan solusi maupun kesimpulan
2	Rani Sri Yulianti	Apa jenis handphone anda (merk dan tipe) dan versi berpada android yang terpasang pada android anda? (bisa lebih dari satu)	OPPO A37/Lollipop 5.1
		Dari pengalaman menggunakan Aplikasi Chatting Dengan Fitur Threading di perangkat mobile anda, adakah isu pada tampilan (peletakan tampilan yang tidak sesuai) yang mengganggu penggunaan aplikasi?	Tidak
		Apa pendapat anda mengenai fitur untuk menambah Thread?	Sudah bagus sesuai dengan tujuan dari aplikasi dan sederhana, serta kemudahan ikon + yang sesuai yang

			umumnya orang sudah tau fungsinya
		Bagaimana penilaian anda terhadap fitur 'Browse Thread' dan 'Join Thread'	Tampak sederhana dan mudah dibaca serta pengalihan dari browse thread ke join thread mudah tidak ribet,
		Kritik dan saran mengenai Aplikasi Chatting Dengan Fitur Threading	adanya penambahan fitur komentar dari pernyataan seperti di WA, adanya pemberitahuan, adanya fitur pencarian sehingga dapat memudahkan pencarian isu yang kita inginkan dan pengecerakan thread yang sudah ada / lama (tenggelam) dengan mudah supaya menghindari adanya topik thread yang serupa, pada bagian my thread tidak ditambah dengan fitur hapus atau edit karena jika ada kesalahan penulisan maka thread tidak dapat diganti atau dihapus dan itu menjadikan sampah untuk memenuhi tampilan thread, adanya fitur invite atau berbagi link atau apa gitu yang dapat memudahkan kita untuk mengajak orang

			menggunakan aplikasi tersebut dan gabung pada thread tertentu. Cukup sekian dan terimakasih sajaah
3	Sherly Angelica Rosa	Apa jenis handphone anda (merk dan tipe) dan versi berpada android yang terpasang pada android anda? (bisa lebih dari satu)	Sony Xperia J, Android 4.1.2
		Dari pengalaman menggunakan Aplikasi Chatting Dengan Fitur Threading di perangkat mobile anda, adakah isu pada tampilan (peletakan tampilan yang tidak sesuai) yang mengganggu penggunaan aplikasi?	Sepertinya tidak ada.
		Apa pendapat anda mengenai fitur untuk menambah Thread?	Saya suka ide ini, karena memberikan akses ke pengguna untuk membuat Thread sesuai keinginan mereka.
		Bagaimana penilaian anda terhadap fitur 'Browse Thread' dan 'Join Thread'	Sepertinya lebih baik di fitur Browse Thread ditambahkan juga fitur 'search' jadi ketika thread sudah banyak nantinya, dan pengguna hanya ingin join thread yang sudah ada dan

			<p>tidak mau menambah thread, mereka bisa lebih mudah mencari thread yang mereka mau. Daripada mereka harus scroll ke bawah terus (kecuali memang ada batasan nantinya di jumlah Thread). Plus, you might put hashtag as one of the requirement to make a new thread in the description in order to ease other users in their browsing by only type a keyword. Lalu, di fitur Join Thread, menurut saya mungkin orderingnya akan lebih baik kalau itu sesuai dengan waktu. Jadi bukan abjad. Karena itu dapat memudahkan pengguna untuk kembali ke thread mereka yang paling baru. (kaya app Line gitu, bisa arranged by time order).</p>
		<p>Kritik dan saran mengenai Aplikasi Chatting Dengan Fitur Threading</p>	<p>- Ketika Update Profile, dan loading dengan memunculkan "updating user data" mungkin kalau bisa pakai indikator persentase saja, jadi pengguna tau sudah sampai mana proses updating. Karena, di kasus</p>

			<p>saya, updatingnya takes time dan ternyata gagal.</p> <p>- Ada indikator read apa tidaknya kah? atau memang belum ada yang read ya? - Di dalam percakapannya, it would be better if ada keterangan tanggal juga selain waktu, jadi pengguna yang baru saja masuk ke thread bisa tau kapan terakhir thread itu aktif. - Lebih baik nanti di app nya diberikan semacam tutorial atau introduction about the app gitu setelah pengguna baru join dan buka appnya. Atau at least, nanti bentuknya seperti welcoming message untuk menyambut pengguna baru ini. Jadi ga bingung2 bgt pas masuk (buat mereka yang ga tau sama sekali soal app ini).</p>
4	M Syaerul Alim	Apa jenis handphone anda (merk dan tipe) dan versi berpada android yang terpasang pada android anda? (bisa lebih dari satu)	Asus Zenfone GO Android OS, v5.1 (Lollipop)

		Dari pengalaman menggunakan Aplikasi Chatting Dengan Fitur Threading di perangkat mobile anda, adakah isu pada tampilan (peletakan tampilan yang tidak sesuai) yang mengganggu penggunaan aplikasi?	Saya rasa sudah cukup bagus
		Apa pendapat anda mengenai fitur untuk menambah Thread?	Tidak ada kendala berarti saat menggunakan aplikais ini
		Bagaimana penilaian anda terhadap fitur 'Browse Thread' dan 'Join Thread'	Menarik, tampilan simpel dan ringkas, tidak bertele-tele
		Kritik dan saran mengenai Aplikasi Chatting Dengan Fitur Threading	Kedepannya agar ditambahkan fitur kirim gambar, jaid tidak sekedar teks saja.

APLIKASI CHATTING DENGAN FITUR THREADING BERBASIS SISTEM OPERASI ANDROID

Achmad Naufal Syafiq

Jurusan Teknik Informatika

Universitas Islam Indonesia

achmadnaufalsyafiq@gmail.com

Abstrak – Aplikasi Chatting Dengan Fitur Threading dibangun untuk membantu para pengguna perangkat bergerak khususnya yang menjalankan sistem operasi Android untuk dapat membahas sebuah permasalahan dengan pengguna lain yang memiliki ketertarikan dengan topik yang sama. Banyak jenis aplikasi yang mencoba mengelompokkan penyebaran informasi seperti forum internet dan sebagainya, namun belum mengakomodasi beberapa hal seperti portabilitas dan real – time data transfer. Dari beberapa kekurangan tersebut terpikirlah ide untuk membuat aplikasi yang dapat mengakomodasi pertukaran informasi melalui chatting secara real – time dan juga memberikan batasan dari pembahasannya. Aplikasi Chatting Dengan Fitur Threading berbasis sistem operasi android, menerapkan Usecase Diagram dan Activity Diagram sebagai alat untuk merancang kebutuhan fungsionalitas yang akan di implementasikan pada saat pembuatan aplikasi. Android Studio digunakan sebagai Integrated Development Environment (IDE) dengan bahasa pemrograman Java dan menggunakan Firebase Realtime Database Sebagai database sebagai basis datanya. Aplikasi ini dirancang bagi para pengguna smartphone khususnya perangkat yang memiliki sistem operasi Android, sehingga diharapkan dapat memudahkan dalam proses pengaksesannya dan juga dapat membantu mengakomodasi pembahasan yang berbeda beda.

Kata kunci : Aplikasi Chatting, Forum, Android Studio, Firebase, Real – Time, Database, Usecase Diagram, Activity Diagram.

I. PENDAHULUAN

Kebutuhan akan komunikasi yang cepat, murah, dan aman sudah ditunjukkan mulai dari awal manusia menemukan alat komunikasi pertama yang berupa lukisan berisi pesan. Bagaimanapun sejarah tentang komunikasi mulai dari belum ditemukannya *alphabet* hingga sekarang, menuntun manusia untuk terus berinovasi menemukan cara mempermudah penyampaian pesan, baik berbentuk teks, suara, gambar, maupun media lain.

Threading mulai banyak dikenal oleh para pengguna internet, khususnya mereka yang aktif pada forum – forum sebagai bentuk dari ke-akurasi-an informasi dan fitur yang digunakan untuk menspesifikasikan masalah. Keunggulan informasi yang didapat dari penggunaan fasilitas *Threading* pada forum juga mudahnya pengguna untuk mencari permasalahan yang sesuai dengan kebutuhan informasinya. Adanya fitur *reply* pada forum memungkinkan pengguna untuk memberikan pemikirannya sesuai dengan topik permasalahan yang dijelaskan oleh pembuat thread dengan harapan pembahasan tidak berpindah diluar topik.

Penggabungan beberapa fitur pada satu aplikasi dapat dilakukan untuk mencapai aspek aspek yang dibutuhkan dalam penyampaian informasi (komunikasi). *Communication Software* dengan tujuan penyampaian pesan yang cepat (*real-time chatting*), dan pengelompokan informasi yang dapat dilakukan dengan fitur Thread dapat menjadi solusi permasalahan yang dapat terjadi pada sebuah aplikasi *messaging*.

II. LANDASAN TEORI

A. Pengertian Forum

Menurut Kamus Besar Bahasa Indonesia (KBBI) Forum adalah lembaga atau badan, bisa disebut juga wadah namun dalam pengertian lain sebuah forum bisa didefinisikan sebuah tempat pertemuan untuk bertukar pikiran secara bebas. Sedangkan menurut Cambridge Dictionary forum merupakan suatu situasi atau pertemuan dimana orang dapat berbicara tentang masalah terutama yang menjadi minat atau perhatian masyarakat.

A. Forum Internet

Sebuah forum internet dapat didefinisikan sebagai tempat berdiskusi pada website. Ditujukan untuk memfokuskan topic diskusi mengenai 26actor segala hal. Kemudahan dalam mengakses dan memberikan pernyataan melalui *online* dengan berbagai sarana seperti media, maupun website yang didedikasikan untuk sebuah forum seperti Kaskus.co.id menjadi salah satu 26actor kesuksesan forum internet untuk menggaet penggunanya.

B. Aplikasi Chatting Dengan Fitur Threading

Aplikasi *chatting* dengan fitur *threading* merupakan aplikasi yang memberikan sarana bagi pengguna forum untuk berkomunikasi secara *real-time*. Dengan pemberian fitur *real-time* pada sebuah pembahasan permasalahan user dapat memberikan pendapatnya melalui *chatting* dengan cepat dan relatif lebih mudah karena aplikasi berbasis perangkat bergerak.

C. Firebase

Adalah layanan DbaaS (Database as a Service) dengan konsep *realtime*. Firebase merupakan penyedia layanan cloud dengan backend sebagai servis yang berbasis di San Fransisco, California. Merupakan pengembangan dari teknologi sebelumnya yaitu GCM (Google Chat Message).

D. Firebase *Real – Time Database*

Real time database yang disediakan oleh firebase merupakan sebuah database yang dalam pengimplementasiannya pengembang tidak perlu menambahkan fungsi *real-time*

secara terpisah seperti pada database yang berbasis SQL (*Structuraed Query Language*) lainnya, seperti MySQL, Oracle, PostgreSQL dan lain lainnya karena Firebase sudah memberikan fitur *real time* ketika pengembang mengimplementasikannya kesetiap project yang sudah terdaftar library Firebase didalamnya.

Tidak seperti database pada umumnya Firebase merupakan *database NoSql* dimana dalam pemanggilan data Firebase hanya menyediakan Sql – like query seperti *orderByChild*, *startAt*, *startEnd* dan beberapa fungsi penyaringan lainnya. Tidak hanya memberikan fitur *real – time* didalam pengimplementasiannya, Firebase juga memberikan fitur *offline capability* dimana pengembang dapat mengembangkan aplikasi dengan kemampuan apabila perangkat pengguna tidak terhubung dengan internet maka user masih dapat melihat data yang sebelumnya sudah disimpan secara otomatis ketika mengakses database melalui internet.

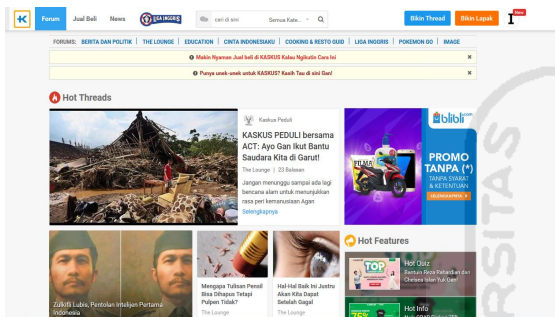
Didalam aplikasi ini Firebase *realtime database* merupakan database utama dan hanya satu satunya database yang digunakan, baik untuk menyimpan *thread* yang dibuat oleh user, hingga percakapan yang dilakukan oleh masing masing pengguna. Pemilihan database ini didasarkan pada fungsi *real – time* yang sangat efisien apabila digunakan untuk aplikasi yang menyediakan fitur *chatting*, karena proses *chatting* pada umumnya dilakukan dengan *real – time*, ketika satu user memasukkan data kedalam database, otomatis user lain yang memiliki *previledge* untuk dapat melihatnya tanpa harus melakukan proses *refresh*.

E. Firebase Storage

Firestore storage dikembangkan oleh google untuk memenuhi kebutuhan pengembang akan media penyimpanan *online* atau *cloud storage* yang terintegrasi langsung dengan Firebase cloud service. Firebase cloud storage pada aplikasi ini digunakan untuk menyimpan file berupa gambar yang diunggah oleh user untuk nantinya dipakai sebagai foto profil masing masing user. Foto foto yang diunggah oleh pengguna akan diterima oleh aplikasi sebagai URL (*uniform resource locator*) dan dimasukkan kedalam masing masing akun pengguna.

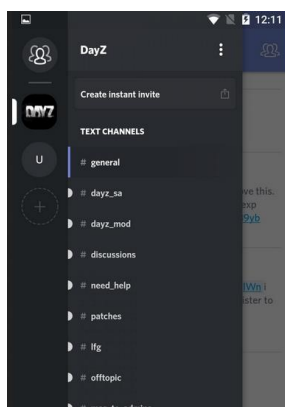
F. Aplikasi Serupa

Kaskus merupakan satu dari banyak forum di Indoonesia yang berhasil untuk terus berkembang dan juga merupakan salah satu forum terbesar di Indonesia. Fitur utama dari kaskus adalah threading dimana terdapat menu *HOT THREAD* yang merupakan kumpulan artikel-artikel yang mendapat banyak *rate* (fitur yang digunakan oleh user untuk menilai suatu thread) dan juga banyaknya komentar dari para pengguna forum lain. Selain itu masing-masing user dilengkapi dengan reputasi sebagai acuan baik-buruknya aktifitas setiap user yang diberikan oleh pengguna lainnya yang dinotasikan sebagai CENDOL (untuk reputasi baik) dan BATA (untuk reputasi buruk).



Gambar 1. Halaman Muka Kaskus

Discord merupakan salah satu aplikasi chatting yang memerlukan *key* untuk masuk dalam sebuah percakapan tertentu, apabila user tidak memiliki *key* tersebut maka pengguna tidak diperbolehkan masuk, cara lain untuk bergabung dalam percakapan adalah mendapatkan *invite* dari user lain.



Gambar 2. Fitur Grouping Discord

III. ANALISIS DAN PERANCANGAN

Aplikasi chatting dengan fitur threading merupakan jenis aplikasi dengan kategori komunikasi yang memiliki tujuan untuk memudahkan orang membahas sebuah topik permasalahan dengan mempersempit topik pembicaraan sehingga para pengguna dapat lebih fokus membicarakan topik tertentu yang ingin diselesaikan.

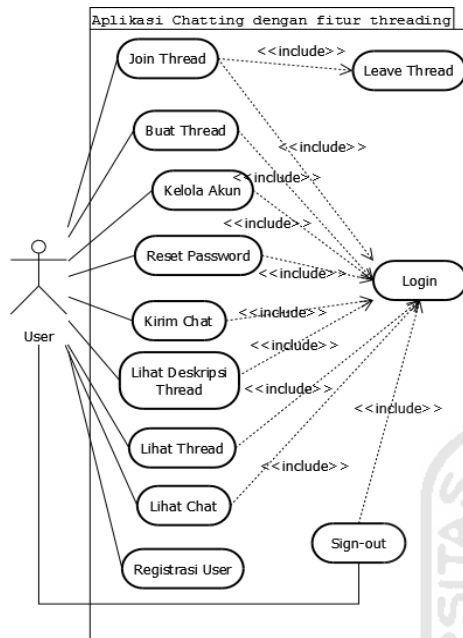
Aplikasi sejenis dengan kategori komunikasi menyediakan fasilitas *real-time chatting* baik satu user ke user lain, maupun fitur *group* yang memungkinkan para pengguna untuk membuat sebuah wadah bagi beberapa user untuk saling berkomunikasi. Aplikasi chatting dengan fitur threading memiliki perbedaan dalam membentuk sebuah percakapan. Para user diharuskan memiliki topik pembicaraan yang dapat dibahas oleh pengguna lain sehingga pembahasan yang dilakukan mengerucut pada sebuah masalah.

Pengguna memiliki kebebasan untuk memilih topik mana yang ingin dia bahas. Didalam aplikasi ini para user dapat melihat topik mana yang baru dibuat oleh user dengan melihat *timeline* topik. Selain itu fitur lain memungkinkan user untuk melihat topik mana saja yang user sudah pernah bahas sebelumnya. Untuk menggunakan aplikasi pengguna diharuskan mendaftar dengan *e-mail*, *password*, dan *username* dan user lain dapat melihat *username* kita. Fitur notifikasi memungkinkan untuk melihat siapa saja yang memberikan balasan pada topik yang sedang kita ikuti. Aplikasi ini juga menyediakan fitur *reset password* yang digunakan untuk mengganti *password* lama dengan yang baru sesuai kehendak pengguna.

Backend aplikasi ini sebagian besar ditangani oleh teknologi Firebase yang dirancang oleh google untuk membuat aplikasi dengan *real-time* database. Sehingga proses management data banyak dilakukan di Firebase *console*, seperti melihat siapa saja yang sedang melakukan aktifitas chatting dan topik apa saja yang sudah dibuat oleh pengguna.

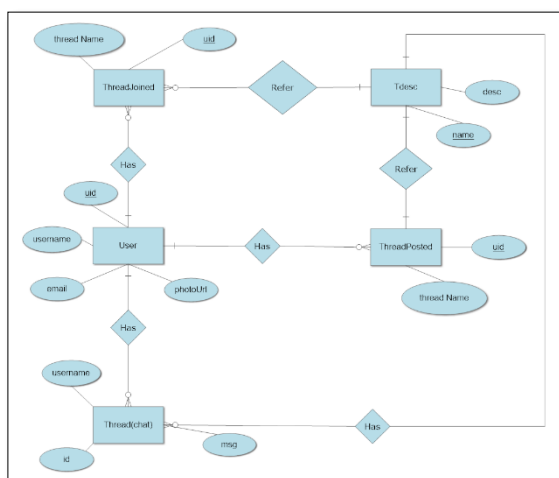
Perancangan direpresentasikan dengan beberapa diagram yang merupakan hasil dari komparasi dengan aplikasi sejenis dan mengadopsi beberapa fitur yang dapat diterapkan pada aplikasi ini. Salah satunya

adalah *Usecase Diagram* (UCD), digunakan untuk mendeskripsikan aktifitas apa saja yang dapat dilakukan oleh sistem dan siapa yang dapat melakukan aktifitas tersebut. Secara sederhana dapat dijelaskan bahwa UCD adalah diagram yang menjelaskan interaksi yang dilakukan oleh sistem dengan aplikasi.



Gambar 3 Diagram *Usecase* Aplikasi Chatting Dengan Fitur Threading

ERD merupakan sebuah teknik memodelkan data yang direpresentasikan dalam bentuk diagram berisikan entitas dari sistem dan hubungan antara entitas tersebut. Dalam pengembangan aplikasi chatting yang memiliki fitur threading ERD yang dibuat akan diimplementasikan kedalam Firebase Database.



Gambar 4. *Entity Relationship Diagram*

Relasi antar entitas menggambarkan keterkaitan antara suatu data dengan data lainnya pada struktur database.

7. Relasi User dengan Tdesc(thread description)

User yang merupakan sebuah entitas pada basis data dan juga berperan sebagai aktor memiliki keterkaitan dengan thread, dimana pembuat thread yang akan digunakan sebagai ruang diskusi adalah user, data yang terdapat pada thread adalah nama, dan deskripsi yang nantinya diakses menggunakan nama dari thread tersebut.
8. Relasi User dengan Thread (chat)

Thread(chat) yang merupakan entitas dengan 3 atribut yang berhubungan dengan user, tiga atribut tersebut adalah id, msg (message), dan username. Atribut yang terdapat pada chat merupakan data yang didapat dari user setiap kali mengirimkan pesan atau chat baru. *Primary key* dari *relationship* ini adalah id yang terdapat pada entitas user.
9. Relasi Tdesc dengan Thread Poster

Thread poster adalah entitas penyimpanan yang beratributkan dua data yaitu uid dan thread name, uid yang terdapat didalam thread poster merupakan data yang didapatkan dari entitas Tdesc.
10. Relasi User dengan Thread Joined

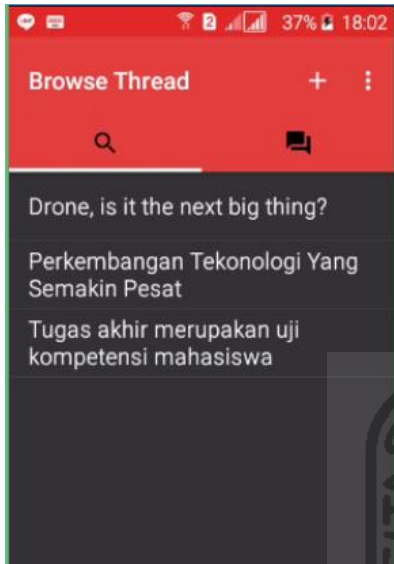
User dalam hal ini sebagai aktor dapat bergabung dengan sebuah thread, data yang berisikan siapa saja yang mengikuti thread apa, disimpan dalam sebuah entitas yaitu Thread Joined, didalam thread joined terdapat dua atribut yaitu uid dan nama thread, uid sama seperti pada entitas Thread Poster didapatkan dari entitas user yang memiliki atribut uid yang merupakan *primary key*.
11. Relasi Tdesc dengan Tjoined

Tdesc adalah sebuah entitas yang menampung deskripsi masing masing thread, relasi dengan Tjoined adalah ketika pengguna bergabung didalam sebuah *thread*. Nama thread yang terdapat pada Tdesc dikirim ke Tjoined.
12. Relasi User dengan Thread Poster

Thread Posted merupakan sebuah entitas yang didalamnya terdapat uid User yang nantinya akan dikirimkan apabila pengguna membuat *thread* baru.

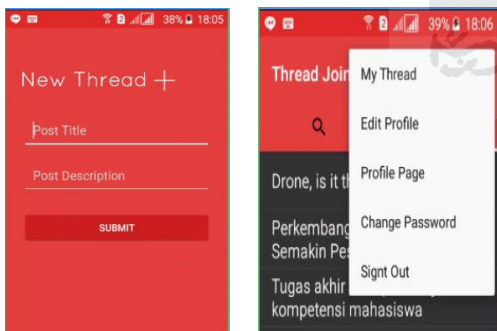
IV. HASIL DAN PEMBAHASAN

a. Hasil Implementasi merupakan realisasi dari perancangan Aplikasi yang sudah dibahas di bab sebelumnya. Implementasi dilakukan untuk memastikan bahwa perangkat lunak dirancang secara baik.



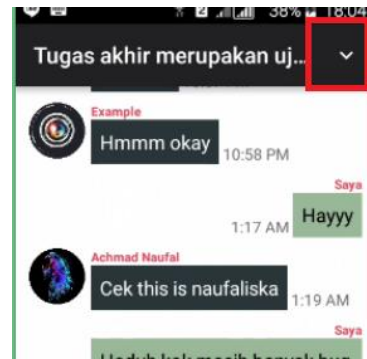
Gambar 5. Tampilan Muka

Pada Aplikasi ini terdapat fitur tambah thread, ubah password, ubah foto profil dan *username*, terdapat juga fitur untuk melihat thread yang sudah dibuat.



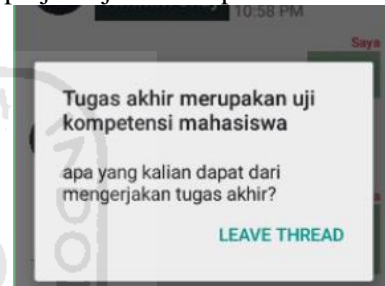
Gambar 6. Hasil Implementasi Tambah Thread dan fitur aplikasi

Chatting merupakan fitur penting dalam aplikasi ini, fitur ini memungkinkan pengguna untuk dapat berkomunikasi dengan pengguna lain didalam sebuah thread, sesuai dengan thread yang dimasuki.



Gambar 7. Fitur Chatting Dalam Aplikasi

Dalam setiap thread terdapat deskripsi yaitu penjelasan dari judul yang dimasukkan oleh pengguna apabila ingin membuat sebuah thread, deskripsi sendiri berfungsi untuk memperjelas judul dan pembahasan.



Gambar 8. Hasil Implementasi tampilan deskripsi

b. Hasil Pengujian

Pengujian perangkat lunak ditujukan untuk menentukan apakah aplikasi yang telah diimplementasikan sesuai dengan hasil perancangan. Pengujian juga ditunjukkan untuk menentukan kualitas dari *software* dengan cara mengujinya dengan metode sesuai dengan batasan implementasinya.

Pengujian menggunakan 2 metode yaitu *black - box* testing yang merupakan metode pengujian yang memfokuskan pada keperluan fungsional dari sebuah perangkat lunak. Pengujian dilakukan oleh seorang yang sudah biasa menggunakan aplikasi serupa seperti aplikasi messaging pada perangkat mobile.

Metode kedua yaitu *User Acceptance Test* (UAT) yang dimaksudkan untuk mendapatkan umpan balik dari beberapa pengguna Aplikasi Chatting Dengan Fitur Threading sekaligus mengetahui komptaibilitas dari masing masing perangkat dan versi Android yang digunakan. Berikut kesimpulan yang didapat dari hasil pengujian yang dilakukan terhadap 4 *user*.

Sisi fungsionalitas merangkul masalah kebutuhan yang tersedia pada aplikasi, seperti login, menambah *thread* dan sebagainya. Dari 4 pengguna yang dijadikan sebagai responden dalam pengujian menggunakan metode UAT yang sifatnya terbuka ini dapat disimpulkan bahwa semua fitur berjalan dengan baik pada masing masing perangkat bergerak yang digunakan untuk mencoba aplikasi ini.

	Xiamo Redmi Note Pro 3	Sams ung Gala xy S5	Oppo A37	Son y Xp eria J	Asus Zenfon Go
Komp atibel	✓	✓	✓	✓	✓
Tidak Komp atibel					

Tabel 1 Tabel Pengujian Kompatibilitas

Namun beberapa pengguna mengalami isu pada tampilan, seperti diperangkat Xiaomi Redmi Note 3 dengan layar 5.5 Inch pada fitur *chat* peletakan pesan dikirim dan diterima tidak sesuai seperti pada perangkat lain, namun secara keseluruhan tidak mengganggu fungsionalitas penggunaan Aplikasi.

V. KESIMPULAN DAN SARAN

Berdasarkan perancangan, implementasi dan pengujian sistem yang telah dilakukan, maka penulis dapat disimpulkan bahwa:

3. Aplikasi Chatting Dengan Fitur Threading telah dikembangkan dan diuji dengan beberapa metode pengujian.
4. Aplikasi berhasil mengakomodasi percakapan berdasarkan thread. Sehingga pengguna dapat berdiskusi topik yang sesuai dengan bahasan thread tersebut secara *real – time*.

Saran yang diberikan oleh pengguna sistem untuk perbaikan aplikasi dikemudian hari antara lain aplikasi dapat memiliki fungsi pencarian yang sehingga memudahkan pengguna baik dalam pencarian thread maupun *chat*, Terdapat fungsi notifikasi yang dapat diatur sehingga pengguna dapat memilih thread mana yang ingin diaktifkan fungsi notifikasinya dan juga perbaikan pada segi tampilan.

DAFTAR PUSTAKA

Setiawan, Wira. Membuat Aplikasi Chat Sederhana Dengan Firebase di Android, Diakses 15 Agustus, 2016 dari www.wirasetiawan29.wordpress.com/2016/05/20/membuat-aplikasi-chat-sederhana-dengan-firebase-di-android

Tjandra, Suhatati., Pickerling, C. (2015) Aplikasi Metode – Metode Testing Pada Configuration, Compatibility Dan Usability Perangkat Lunak. Teknik Informatika Sekolah Tinggi Teknik Surabaya

Khan, Belal. (2016). Create Chat App Using GCM, Diakses 10 Agustus, 2016 dari <https://www.simplifiedcoding.net/create-chat-app-for-android-using-gcm-1/>



SARAN/USULAN PRESENTASI KEMAJUAN TUGAS AKHIR

Nama Mhs. : Achmed Nurfal Syafiq

No. Mhs. : _____

Judul TA : Aplikasi Messaging Android dan Threading

- Perlu diperjelas apa itu fitur Threading.
- Jelaskan keunggulan app ini dibanding dengan aplikasi sejenis.



Nilai kemajuan Tugas Akhir: _____ (0 - 100)
(studi pustaka, perancangan, penguasaan materi, ketepatan)

Yogyakarta, 20-September-2016

Dosen,

Chadur Kusum Den
(nama terang)

Dilampirkan pada Laporan TA yang diajukan untuk pendadaran