BAB III
ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan menjelaskan analisis dan perancangan dari aplikasi media sosial mikroblog (*twitter client)* dengan menerapkan metode *user centered desig*n untuk meningkatkan *usability.*

3.1. Analisis Sistem

Dalam analisis sistem dilakukan penguraian dari media sosial mikroblog yaitu *twitter* dengan menerapkan metode UCD untuk meningkatkan *usability.* Hal itu dilakukan untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan sehingga ditemukan kelemahan, kesempatan, hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya. Melakukan analisis terhadap sistem yang sedang berjalan bertujuan sebagai dasar perancangan atau perbaikan sistem yang lama. Dan hasil analisis tersebut dapat diketahui kelemahan dan kekurangan sistem yang lama dan dapat dirancang atau diperbaiki menjadi sebuah sistem yang lebih efektif dan efisien.

3.1.1. Analisis Masalah

Twitter sebagai salah satu mikroblog yang masih minim fitur karena hanya memiliki fitur seperti *reply tweet, original retweet, favorite, direct message* dan *list*. *Twitter* juga memiliki keterbatasan karena hanya dapat melakukan pembaharuan

dalam format teks, tidak dapat menampilkan informasi dalam format gambar dan melakukan unggah gambar, serta memperpendek sebuah tautan. Hal ini menjadi salah satu batasan dalam berbagi informasi karena pengguna hanya dapat berbagi informasi dalam format teks saja.

3.1.2 Analisis Mikroblog *Twitter*

Dasar aplikasi media sosial mikroblog yang dibangun disini adalah *twitter* , dipilih karena merupakan salah satu mikroblog yang memiliki pengguna terbesar di dunia, bahkan Indonesia merupakan negara dengan pengguna terbesar no 2 di dunia.[11]

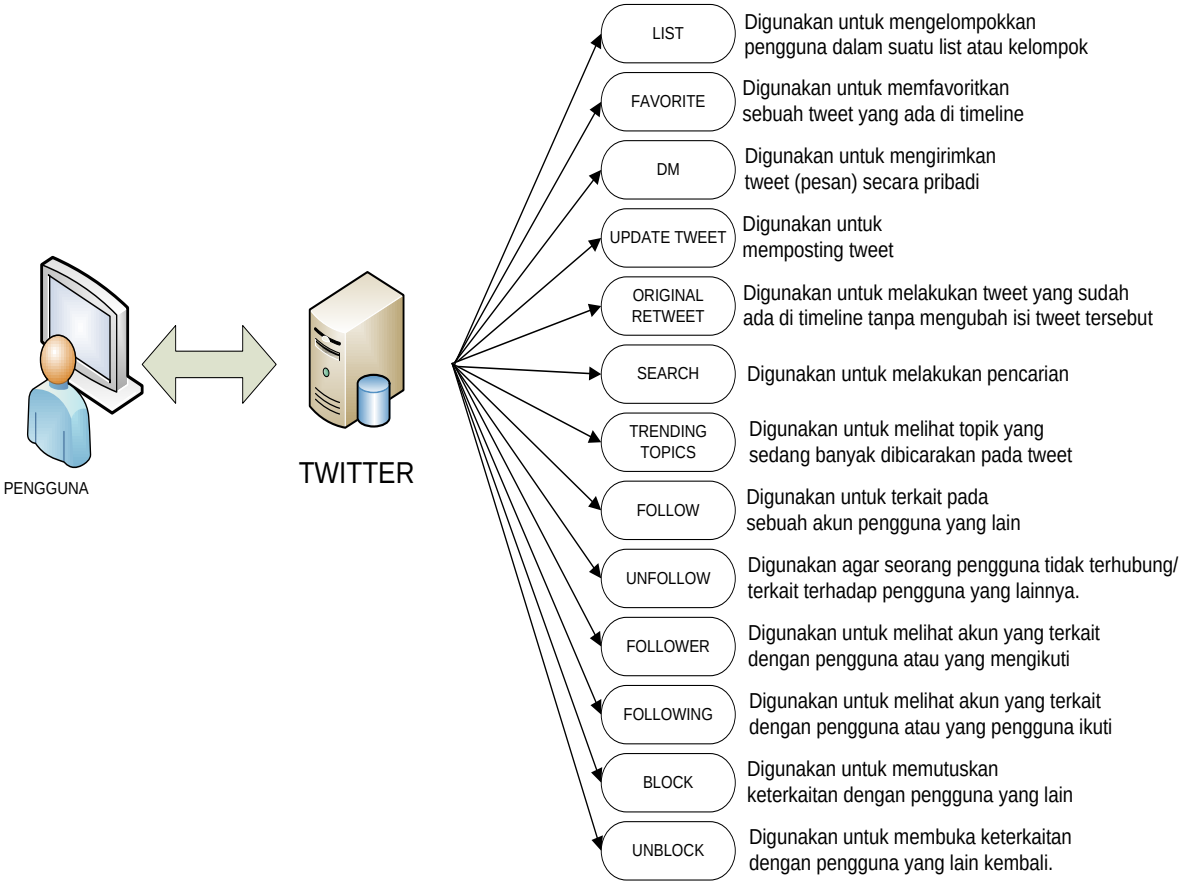
Beberapa istilah yang dipergunakan di Twitter [8]

1. Tweet - sesuatu yang ditulis di status Twitter.
2. @ - penanda yang ditulis didepan nama pengguna lain agar *tweet* yang dimasukkan diterima oleh pengguna tersebut.
3. Reply – balasan terhadap tweet.
4. Following - pengguna lain yang *tweet* nya akan selalu diterima.
5. Follower - pengguna lain yang akan selalu menerima *tweet.*
6. ReTweet (RT) adalah melakukan posting kembali tweet dari user lain.
7. Direct Messages - pesan personal.
8. Favorites - menyimpan *tweet* tertentu yang diinginkan.

9. Hash Tags (#) - penanda yang ditulis di depan topik tertentu agar pengguna lain bisa mencari topik yang sejenis yang ditulis oleh orang lain juga.

10. Trending Topics - topik yang sedang banyak dibicarakan banyak pengguna dalam suatu waktu yang bersamaan.

Pada gambar 3.1 dapat dilihat gambaran fitur-fitur pada *twitter* .



Gambar 3.1 Fitur-Fitur Twitter

3.1.2.1 Kelebihan dan Kelemahan *Twitter*

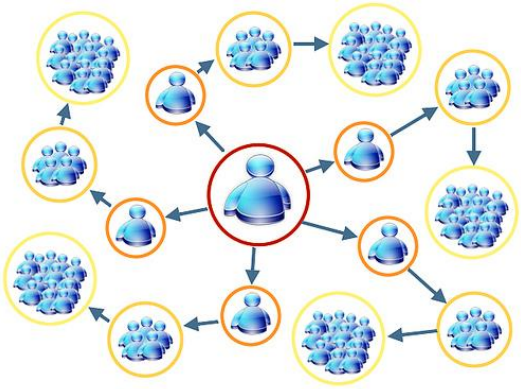
Twitter banyak diminati karena menghadirkan hal-hal berikut yang dirasa sangat berguna bagi pengguna antara lain :

1. Hubungan Asimetrik

   Menjalin hubungan pertemanan di dalam *twitter* itu sangat sederhana , namun revolusioner . Sebelum *twitter* muncul , media sosial seperti *facebook* dan *friendster* membutuhkan *mutual relationship* yang artinya disaat salah satu pengguna mengajukan permohonan hubungan pertemanan dan diterima oleh yang diminta , maka untuk selanjutnya mereka saling mengikuti *update* masing – masing . Di *Twitter*, secara *default* kita bisa melihat *update status* seseorang tanpa harus meminta persetujuan dulu.

   Hubungan asimetrik itu mewakili kehidupan manusia sehari-hari , seseorang bisa dikenal oleh banyak orang namun tidak harus mengenal keseluruhan orang yang dikenalnya. Keuntungan dari sistem ini , pengguna tidak dipusingkan oleh melihat status *update* dari orang yang tidak dikenal pengguna. Tetapi, jika pengguna memiliki pengikut *(follower)* yang banyak, setiap apa yang kita tuliskan sebagai status update , akan terlihat di semua pengikut . Contohnya kita menuliskan : *"Ada yang tau lokasi factory outlet di Bandung?"* , jika pengguna memiliki 10.000 *follower* , maka status *update* itu akan ditampilkan di 10.000 *follower*. Pada saat pertama kali melakukan status *update* atau dengan melakukan

retweet seperti *"RT @user_pertama : "Ada yang tau lokasi factory outlet di Bandung?"* sehingga bisa dilihat pula oleh pengikutnya , misalnya pengguna tadi memiliki 1200 pengikut , maka pesan tadi akan diperlihatkan kepada pengikutnya yang sebanyak 1200 tersebut. Dan jika satu pengikut yang melakukan RT , bagaimana jika 10.000 orang yang mengikuti penulis status pertama dan di *retweet* ke pengikutnya lagi dan seterusnya. Jika dianalogikan seperti gambar pada 3.2 .



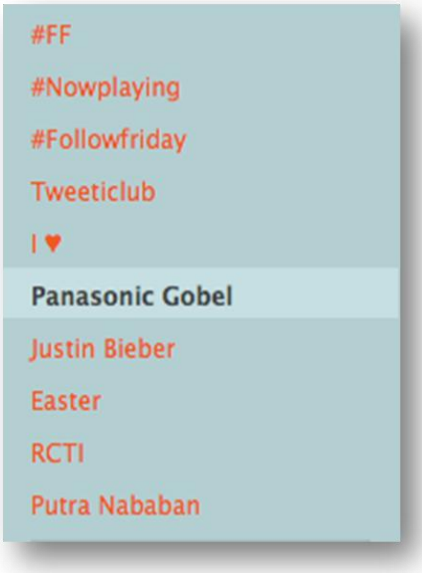
Gambar 3.2 Hubungan Pertemanan di *Twitter*

2. *Follow* dan *update* melalui SMS

   Dengan mengaktifkan layanan SMS , pengguna dapat menerima pesan *(tweet)* melalui SMS dan juga melakukan status update melalui SMS , jadi internet bukanlah suatu kewajiban dalam penggunaan *twitter* karena pengguna diberi keleluasaan dalam memilih akses sesuai dengan kebutuhan dan kemampuannya.

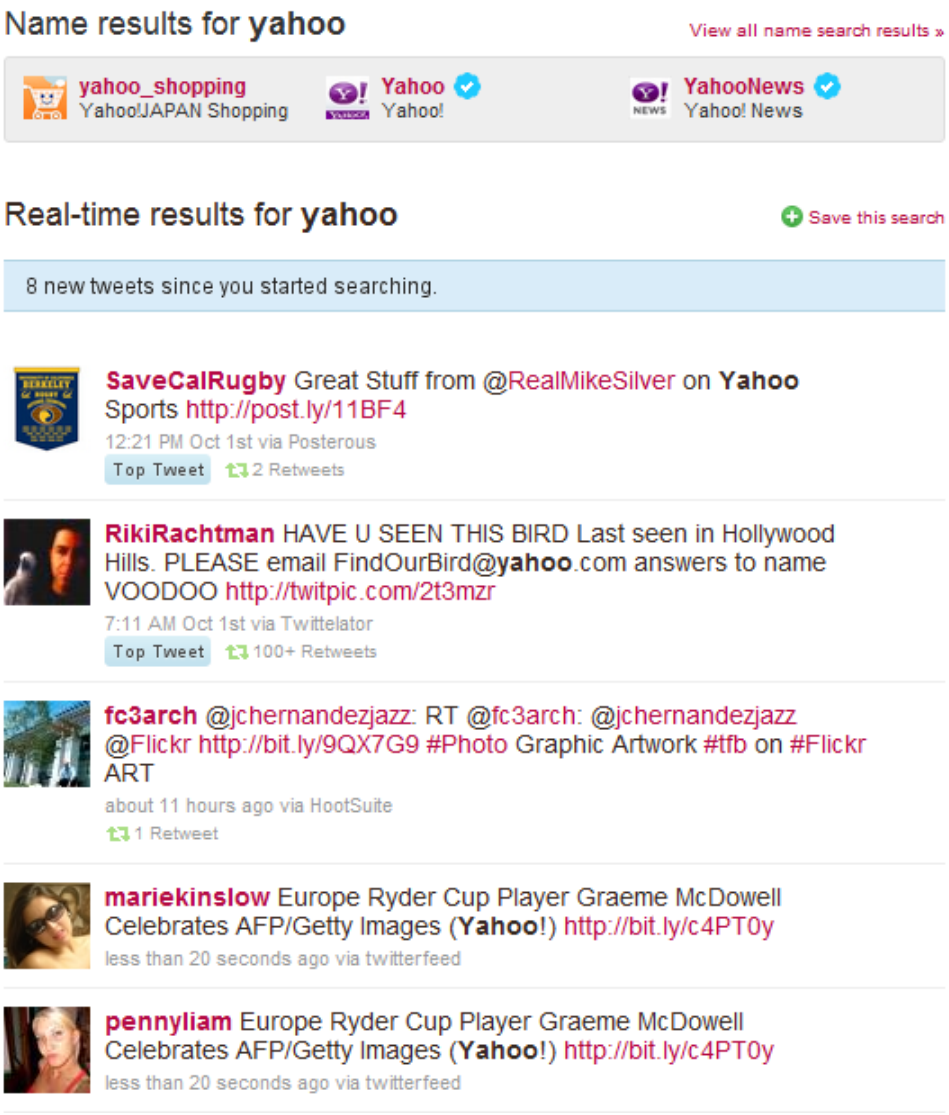
3. *Trends* dan *search*

Trends atau yang biasa juga disebut "*Trending Topics"*adalah 10 topik yang paling banyak dibicarakan pengguna *twitter* secara *real time* , dengan mengetahui apa yang paling banyak dibicarakan , kita akan selalu mengetahui *update* terbaru, misalnya tanpa harus membuka media pemberitaan seperti televisi dan surat kabar. *Trending topics* pada *twitter* dapat dilihat pada gambar 3.3 .



Gambar 3.3 *Trending Topic*

Search juga merupakan fitur yang sangat penting , pencarian di *twitter* adalah *real-time* berupa data dari status update pengguna *twitter* , bukan hasil dari penyimpanan *(indexing)* contohnya jika melakukan pencarian terhadap kata kunci "YAHOO" maka akan keluar semua yang berhubungan dengan kata kunci tersebut seperti pada gambar 3.4 .



Gambar 3.4 Hasil Pencarian Kata Kunci "YAHOO"

4. API terbuka

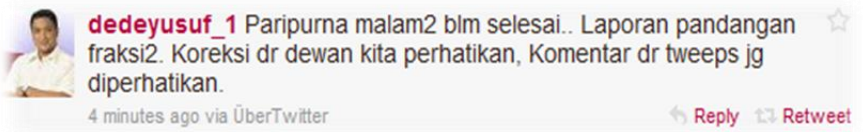
   Alasan mengapa *twitter* sangat terkenal diantara para *developer* , *inventor* , *hacker* dan *entrepreneur* adalah karena API terbuka. Dengan API

terbuka , *twitter* memberikan izin untuk menggunakan data dan fungsionalitas kepada siapapun yang ingin mengolahnya.

Dari kelebihan *twitter* seperti yang diungkapkan di atas, berikut adalah fakta mengenai kegunaan *twitter* :

1. sebagai media komunikasi, misalnya antara seorang selebritis dengan penggemarnya dan seorang presiden dengan warganya.
2. sebagai salah satu media *collaborative learning* (bersifat umum) karena interaksi antara sesama pengguna yang tidak terbatas dalam segala bidang.
3. sebagai media pengiriman berita, perusahaan media seperti BBC, *The New York Times* dan Al Jazeera sedang mencoba mikroblog khususnya *Twitter* sebagai cara untuk mengirim dan menerima berita.
4. sebagai media protes sosial, karena pengguna dapat mengangkat sebuah topik yang dapat tersebar ke seluruh dunia dan memungkinkan untuk menjadi *trending topic.*

   Gambar 3.5 adalah salah satu contoh petikan *tweet* yang merupakan pemanfaatan *twitter* sebagai alat komunikasu antara pejabat dan masyarakat.



Gambar 3.5 Kegunaan Twitter sebagai Komunikasi Antara Pejabat Pemerintah dan Masyarakat.

Kelemahan *twitter* :

1. Kemampuan terbatas yaitu, menemukan orang-orang, mengirim pesan singkat, balasan langsung terhadap '*tweet*' yang diposting.
2. Dibatasi sampai 140 karakter per *update*.
3. Belum ada strategi keuangan yang mudah dan jelas.
4. Lebih menekankan pada perhitungan follower.

3.1.3 Analisis Aplikasi Terhadap Metode UCD

Kehadiran aplikasi media sosial khususnya mikroblog melahirkan satu pilihan baru untuk berkomunikasi dengan masyarakat secara lebih luas. Salah satunya adalah mikroblog *twitter*. Walaupun *Twitter* yang hadir dengan hanya dibatasi 140 karakter setiap kali melakukan pembaharuan tidak membuat *twitter* sulit diterima di masyarakat, hal tersebut dibuktikan dengan meningkatnya pengguna *twitter* setiap tahunnya dan telah menjadi salah satu *trend* komunikasi saat ini. Dengan membangun *twitter client* dan menambahkan beberapa fitur diharapkan dapat mengurangi batasan yang dialami pengguna selama ini. Pembangunan *twitter client* dibangun dengan menerapkan metode UCD yang berguna untuk melakukan analisis kebutuhan pengguna sehingga dapat meningkatkan *usability* (kegunaan) . Metode UCD yang akan dilakukan yaitu : kuisioner, wawancara, *task modelling* ,dan *prototyping*.

3.1.3.1 Metode UCD

1. Kuisioner

   Kuisioner merupakan salah satu teknik pengumpulan data yang dilakukan dengan cara memberikan seperangkat pertanyaan atau pernyataan tertulis kepada pengguna. Kuisioner akan dilakukan pada pengguna yang memiliki account di *twitter* dengan pertanyaan mengenai fitur yang ada pada web *twitter,* kegunaan yang dirasakan oleh pengguna dan kekurangan atau keterbatasan yang dialami oleh pengguna selama ini. Dengan hasil yang didapat dari kuisioner, maka peneliti dapat mengetahui kebutuhan pengguna secara langsung.

2. *Interview*

   Interview dilakukan untuk berinteraksi dengan pengguna dengan maksud untuk mencari tahu apa yang dibutuhkan pengguna dari sistem yang akan dibangun. Pertanyaan yang diajukan pada wawancara secara garis besar hampir sama dengan kuisioner, berikut beberapa pertanyaan untuk wawancara:

   1. Berapa lama waktu yang anda butuhkan untuk mengerti cara kerja *twitter*?
   2. Untuk apa saja biasanya anda menggunakan *twitter*?
   3. Menurut anda apakah fitur pada *twitter* saat ini sudah memenuhi kebutuhan anda sebagai pengguna?Alasan anda?
   4. Menurut anda, apa kelebihan dan kekurangan *twitter?*

5. Fitur apa yang paling sering anda gunakan pada *twitter*, dan mengapa?
6. Apakah anda melakukan *follow* terhadap pengguna dari luar Indonesia atau yang menggunakan bahasa asing?
7. Fitur tambahan apa yang anda inginkan ada di *twitter?*

3. *Task Modelling*

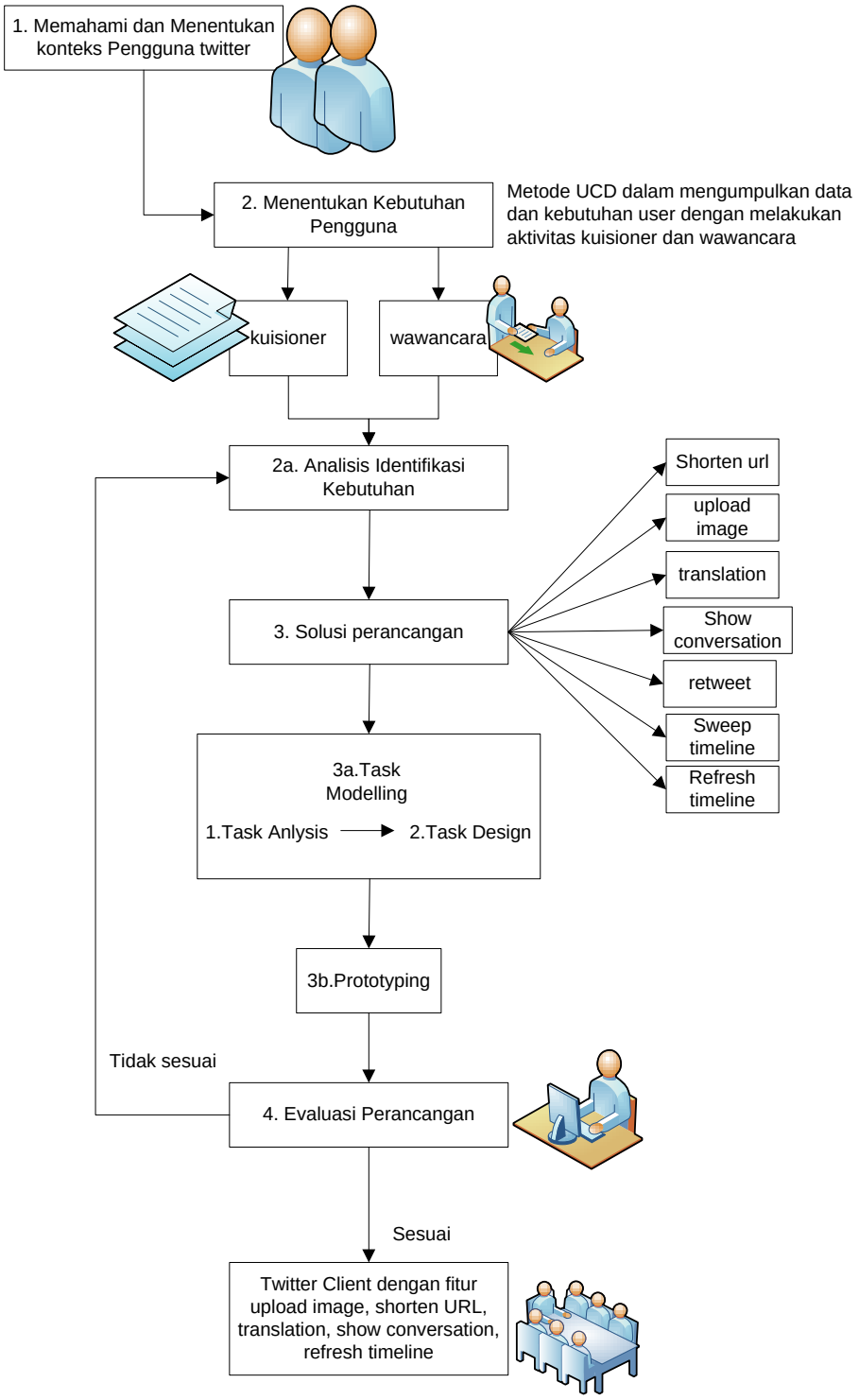
   Proses menganalisis dan memaparkan apa saja dapat yang dilakukan pengguna pada sistem yang akan dibangun. Pada aktivitas ini akan dijelaskan dengan menggunakan *task* analisis sistem yang dan kemudian untuk fitur-fitur yang ditambahkan akan digambarkan *task design* interaksi dengan sistem yang dibangun.

4. *Prototyping*

   Prototipe merupakan proses untuk membangun solusi perancangan yang akan dibangun pada *twitter client.* Pada perancangan aplikasi *client* ini menggunakan *Low-fidelity Prototyping* yaitu berupa sedikit gambaran dari aplikasi yang dibangun, yaitu menggambarkan halaman depan dan penjelasan singkat mengenai fungsionalitas yang ada. Penggambarannya akan menggunakan sketsa.

3.1.3.2 Proses UCD

Proses UCD yang akan dilakukan dapat dilihat pada gambar 3.6 .



Gambar 3.6 Proses UCD

1. Memahami dan Menentukan Konteks Pengguna

Pengguna pada aplikasi *twitter client* adalah bersifat umum dan hanya dapat digunakan oleh pengguna yang memiliki *account twitter*. Pengguna mengerti cara kerja *twitter* sehingga dapat menggunakan *twitter client* yang dibangun.

2. Menentukan Kebutuhan Pengguna

Pada umumnya *twitter* digunakan untuk berbagi informasi, namun saat ini web *twitter h*anya menampilkan fitur yang memungkinkan pengguna berbagi informasi hanya dalam format teks saja. Di awal penelitian akan digunakan metode UCD untuk mengumpulkan data. Karena itu dilakukan aktivitas kuisioner dan wawancara yang berguna untuk mengetahui kebutuhan dan batasan yang ditemui pengguna saat ini dengan pertanyaan dibawah ini.

Berikut analisis kuisioner yang telah dilakukan :

1. Pertanyaan no 1

Berapa lama waktu yang anda butuhkan untuk mengerti menggunakan *twitter*?

Hasil jawaban pertanyaan kuisioner no 1 dapat dilihat pada tabel 3.1 .

3.1 Tabel Hasil Jawaban Pertanyaan No. 1

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
1-7 hari	24	40	60%
1-2 minggu	8	40	20%

3-4 minggu	5	40	12,5%
>1 bulan	3	40	7,5%

2. Pertanyaan no 2

Seberapa sering anda melakukan *tweet*?

Hasil jawaban pertanyaan kuisioner no 2 dapat dilihat pada tabel 3.2 .

3.2 Tabel Hasil Jawaban Pertanyaan No. 2

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
Setiap hari	30	40	100%
Sekali seminggu	0	40	0%
Sekali sebulan	0	40	0%
Hampir tidak pernah	0	40	0%

3. Pertanyaan no 3

Apakah anda sering melakukan *reply tweet* (membalas *tweet*)?

Hasil jawaban pertanyaan kuisioner no 3 dapat dilihat pada tabel 3.3 .

3.3 Tabel Hasil Jawaban Pertanyaan No. 3

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase

Ya	34	40	85%
Kadang-kadang	6	40	15%
Hampir tidak pernah	0	40	0%

4. Pertanyaan no 4

Informasi dalam format apa saja yang sering anda gunakan untuk melakukan *tweet?*

Hasil jawaban pertanyaan kuisioner no 4 dapat dilihat pada tabel 3.4 .

3.4 Tabel Hasil Jawaban Pertanyaan No. 4

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
Text	20	40	50%
Image	7	40	17,5%
url (link)	13	40	32,5%

5. Pertanyaan no 5

Informasi apa saja yang sering anda *tweet?*

Hasil jawaban pertanyaan kuisioner no 5 dapat dilihat pada tabel 3.5 .

3.5 Tabel Hasil Jawaban Pertanyaan No. 5

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
Informasi	30	40	100%

umum (apa saja)			
Informasi teknologi	0	40	0%
Informasi kesehatan	0	40	0%
Update news	0	40	0%

6. Fitur apa yang paling sering anda gunakan di *twitter* saat ini?

Hasil jawaban pertanyaan kuisioner no 6 dapat dilihat pada tabel 3.6 .

3.6 Tabel Hasil Jawaban Pertanyaan No. 6

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
List	0	40	2,5%
Original Retweet	7	40	17,5%
Favorite	1	40	%
reply	32	40	80%

7.Apakah anda sering menggunakan bahasa asing dalam melakukan *tweet*?

Hasil jawaban pertanyaan kuisioner no 7 dapat dilihat pada tabel 3.7 .

3.7 Tabel Hasil Jawaban Pertanyaan No. 7

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase

Ya	19	40	47,5%
Kadang-kadang	12	40	30%
Tidak	9	40	22,5%

8. Fitur apa yang anda inginkan ada pada *twitter client?*

Hasil jawaban pertanyaan kuisioner no 8 dapat dilihat pada tabel 3.8 .

3.8 Tabel Hasil Jawaban Pertanyaan No. 8

Kategori Jawaban	Frekuensi Jawaban	Jumlah Populasi Sample	Jumlah Persentase
Upload image	7	40	17,5%
url shortener	13	40	32,5%
Retweet	9	40	22,5%
Translation	7	40	17,5%
Show conversation	4	40	10%

Dari hasil kuisioner dan wawancara yang telah dilakukan maka dapat ditarik beberapa kesimpulan yaitu sebagai berikut:

1) *Twitter* sudah cukup mudah untuk dimengerti karena (>60%) pengguna hanya membutuhkan waktu kurang dari 7 hari untuk mengerti cara kerja *twitter*.

2) Pengguna aktif melakukan *tweet.*

3) Populasi pengguna adalah pengguna yang umumnya menggunakan *tweet* untuk berbagi informasi apa saja, tidak ada kepentingan atau kebutuhan secara khusus.

4) Pengguna sering menggunakan atau menambahkan *link* didalam *tweet*nya.

5) Pengguna aktif dalam melakukan *reply tweet.*

6) Pengguna juga melakukan *follow* terhadap pengguna di luar Indonesia.

7) Pengguna sering menggunakan bahasa asing dalam *tweet*nya.

8) Pengguna tidak dapat melakukan unggah gambar dan memperpendek sebuha tautan (url).

9) Pengguna memberikan respon yang baik dengan adanya beberapa fitur tambahan seperti isi kuisioner no 8. Fitur *shorten url*, *retweet* dan *upload image* merupakan fitur yang paling banyak dipilih untuk ditambahkan pada aplikasi *twitter client.*

2a. Analisis Identifikasi Kebutuhan

Dari hasil wawancara dan kuisioner yang telah dilakukan, maka dapat dianalisis mengenai kebutuhan pengguna. Secara umum berikut hasil analisis kebutuhan pengguna pada *web twitter* :

1. Karena tidak dapat melakukan unggah gambar, pengguna tidak dapat berbagi informasi yang disertai format gambar. Pengguna juga tidak dapat melihat informasi (gambar) yang ada di *timeline*, karena informasi yang ditampilkan hanya sebuah *url* saja.
2. Setiap kali melakukan pembaharuan yang hanya 140 karakter membuat pengguna merasa terbatas jika ingin menambahkan *link* karena biasanya sebuah tautan atau *link* akan panjang atau menggunakan banyak karakter.
3. Tidak adanya fitur ”*refresh timeline”* sehingga pengguna harus menunggu waktu interval *refresh* dari *twitter*. Hal tersebut mengakibatkan pengguna tidak dapat melihat *update tweet* yang masuk di *timeline* dengan leluasa.
4. Keterbatasan penguasaan bahasa, kebutuhan ini muncul jika pengguna yang ingin menggunakan bahasa asing untuk melakukan *update tweet*. Hal ini membuat pengguna merasa terbatasa karena harus melihat kamus atau membuka situs *google translate* terlebih dahulu.
5. Keterbatasan jika pengguna ingin melakukan *update* terhadap *tweet* yang ada di *timeline (retweet),* karena *twitter* hanya menyediakan fitur *new retweet*, sehingga pengguna tidak dapat menambahkan komentar pada *tweet* tersebut.

6. Tidak adanya fitur yang memungkinkan pengguna untuk melihat kembali pembicaraan terhadapa sebuah *tweet* yang mendapat balasan atau *reply*.

   Gambar 3.7 adalah gambar kelemahan dari *twitter* yang diambil langsung dari *tweet* pengguna.



Gambar 3.7 Keterbatasan Fitur *Twitter*

3. Solusi Perancangan

Dari keterbatasan fitur tersebut maka dapat diambil kesimpulan mengenai kebutuhan pengguna. Solusi dari keterbatasan yang telah dijelaskan di atas adalah dengan menghadirkan fitur-fitur yang dapat mengatasi kebutuhan tersebut pada *twitter client* yang dibangun. Pada tahapan ini akan dilakukan aktivitas UCD yaitu *task modelling* dan *prototyping* sehingga dapat dijelaskan mengenai fitur-fitur yang

akan ditambahkan pada aplikasi *client* yang merupakan hasil dari analisis kebutuhan pengguna. Fitur- fitur tersebut adalah *upload image, shorten URL, retweet, sweep timeline, show conversation, translation,* dan *refresh timeline* yang diharapkan dapat mengatasi batasan yang dialami pengguna saat ini.

3a. Task Modelling

Pada aktivitas ini *task modelling* akan dijelaskan dengan menggunakan *task analysis* dan *task design*.

1. *Task Analysis*

Task analysis akan dikelompokkan menjadi 2 yaitu berdasarkan *task* yang sudah ada pada *twitter* dan beberapa task yang merupakan fitur tambahan yang dihasilkan dari proses UCD yang telah dilakukan.

Task pada *twitter* :

1. Task 1 Melakukan *Login* ke aplikasi *twitter client*

   Sub_task 1.1 : Pengguna meginputkan *user name* dan password akun *twitter.*

2. Task 2 Melakukan *update tweet*

   Sub_task 2.1 : Pengguna menginputkan *tweet* yang akan diposting ke dalam kolom *tweet* kemudian pilih menu *tweet.*

3. Task 3 Melakukan *reply*

   Sub_task 3.1 : Pengguna memilih menu *reply* pada salah satu *twee*t yang ada di halaman *timeline.*

Sub_task 3.2 : Pengguna menginputkan *tweet* yang merupakan balasan terhadap *tweet* yang dipilih tadi pada kolom *tweet* kemudian klik menu *tweet.*

4. Task 4 Melakukan *favorite*

Sub_task 4.1 : Pengguna memilih menu *favorite* pada salah satu *tweet* yang ada di halaman *timeline.*

5. Task 5 Melakukan *original retweet*

Sub_task 5.1 : Pengguna memilih menu *original retweet* pada salah satu *tweet* yang ada di halaman *timeline.*

6. Task 6 Melakukan *delete tweet*

Sub_task 6.1 : Pengguna memilih menu *delete* pada *tweet* yang akan dihapus dari *timeline.*

7. Task 7 Melakukan *folow*

Sub_task 7.1 : Pengguna memilih akun yang akan di *follow*, dan kemudian klik menu *follow*.

8. Task 8 Melakukan *unfollow*

Sub_task 8.1 : Pengguna memilih akun yang akan di unfollow, dan kemudian klik menu *unfollow*.

9. Task 9 Melakukan *block*

Sub_task 9.1 : Pengguna memilih akun yang akan di *block* dan kemudian klik menu *block.*

10. Task 10 Melakukan *unblock*

Sub_task 10.1 : Pengguna memilih akun yang akan di *unblock* dan kemudian klik menu *unblock* .

11. Task 11 Melihat *followers*

Sub_task 11.1 : Pengguna memilih menu *followers* dan akan ditampilkan akun yang terkait atau yang mengikuti akun tersebut.

12. Task 12 Melihat *following*

Sub_task 12.1 : Pengguna memilih menu *following* dan akan ditampilkan akun yang diikuti akun tersebut.

13. Task 13 Melihat *mentions*

Sub_task 13.1 : Pengguna memilih menu *mentions* dan akan ditampilkan *tweet* yang mengaitkan nama akun tersebut pada *tweet* yang telah di*posting*.

14. Task 14 Melihat DM (*direct messages*)

Sub_task 14.1 : Pengguna memilih menu *direct messages* dan akan ditampilkan pesan yang masuk.

15. Task 15 Mengirimkan DM

Sub_task 15.1 : Pengguna memilih menu *direct messages*

Sub_task 15.2 : Pengguna menginputkan nama akun yang akan dikirimi pesan lalu menginputkan pesan pada kolom *tweet* kemudian klik *send.*

16. Task 16 Melihat *lists*

Sub_task 16.1 : Pengguna memilih menu *lists* dan kemudian akan ditampilkan *list* yang ada.

17. Task 17 Membuat *list*

Sub_task 17.1 : Pengguna memilih menu *lists* dan kemudian akan ditampilkan *list* yang ada.

Sub_task 17.2 : Pengguna memilih menu *create a new list*

Sub_task 17.3 : Pengguna menginputkan nama *list* yang akan dibuat kemudian klik *create.*

18. Task 18.1 Menghapus *list*

Sub_task 18.1 : Pengguna memilih menu *lists* dan kemudian akan ditampilkan *list* yang ada.

Sub_task 18.2 : Pengguna memilih menu *delete* pada *list* yang akan dihapus.

19. Task 19 Melakukan *add members*

Sub_task 19.1 : Pengguna memilih menu *lists* dan kemudian akan ditampilkan *list* yang ada.

Sub_task 19.2 : Pengguna memilih menu *add members* pada *list* yang akan ditambah anggotanya.

20. Task 20 Melihat *favorites*

Sub_task 20.1 : Pengguna memilih menu *favorites* dan kemudian akan ditampilkan *tweet* yang telah difavoritkan.

21. Task 21 Melihat *retweets*

Sub_task 21.1 : Pengguna memilih menu *retweets* dan kemudian akan ditampilkan *retweet* perah dilakukan.

22. Task 22 Melakukan Search

Sub_task 22.1 : Pengguna menginputkan keyword pada kolom search kemudian akan ditampilkan hasil pencarian.

Berikut adalah task untuk fitur-fitur yang ditambahkan pada aplikasi *twitter client* yang dibangun :

1. Task 1 Melakukan *shorten url*

Sub_task 1.1 : Pengguna menginputkan url yang akan dipendekkan ke dalam kolom tweet.

Sub_task 1.2 : Pengguna memilih menu shorten url, kemudian akan ditampilkan hasil url yang sudah dipendekkan pada kolom tweet.

2. Task 2 Melakukan *translation*

Sub_task 2.1 : Pengguna menginputkan *tweet* yang akan diterjemahkan pada kolom *tweet.*

Sub_task 2.2 : Pengguna memilih menu *translation* kemudia memilih bahasa terjemahan yang diinginkan kemudin klik *translate* lalu akan ditampilkan hasil terjemahan pada kolom *tweet.*

3. Task 3 Melakukan *Upload image*

Sub_task 3.1 : Pengguna memilih menu *upload image*

Sub_task 3.2 : Pengguna memilih *image*(gambar) yang akan di *upload* dengan klik *choose file*, lalu klik *upload.*

4. Task 4 Melakukan *restore previous tweet*

Sub-task 4.1 : Pengguna memilih menu *restore previous tweet*, kemudian akan ditampilkan *tweet* yang terakhir kali diposting pada kolom *tweet.*

5. Task 5 Melakukan *sweep timeline*

Sub_task 5.1 : Pengguna memilih menu *sweep timeline*, kemudian *tweet* yang ada di bagian bawah akan dihapus (tidak dtampilkan).

6. Task 6 Melakukan *refresh timeline*

Sub_task 6.1 : Pengguna memilih menu *refresh timeline*, kemudian akan ditampilkan *tweet* terbaru pada halaman *timeline.*

7. Task 7 Melakukan *retweet*

Sub_task 7.1 : Pengguna memilih menu *retweet* pada salah satu *tweet* yang ada di *timeline*.

8. Task 8 Melakukan *show conversation*

Sub_task 8.1 : Pengguna memilih menu *show conversation*, kemudian akan ditampilkan balasan terhadap sebuah *tweet* secara berurutan.

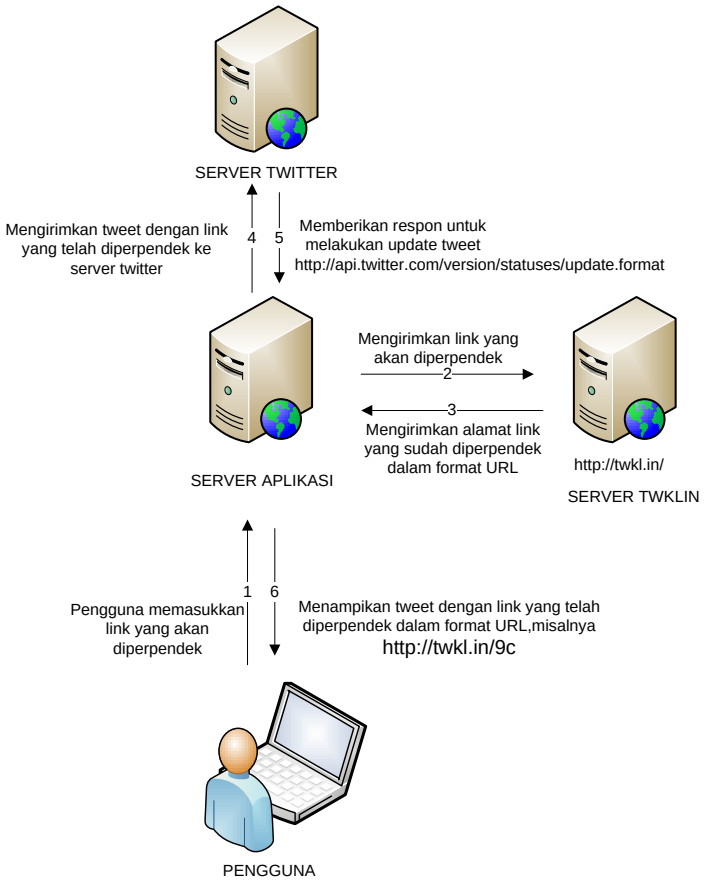
2. *Task Design*

Task design akan dijelaskan dengan menggambarkan komunikasi antara fitur - fitur ditambahkan dengan *twitter* dan pengguna. Untuk *task design* hanya akan

menggambarkan fitur-fitur yang ditambahkan saja. Berikut gambaran *task design* penambahan fitur pada *twitter client* yang akan dibangun :

1. *shorten URL*

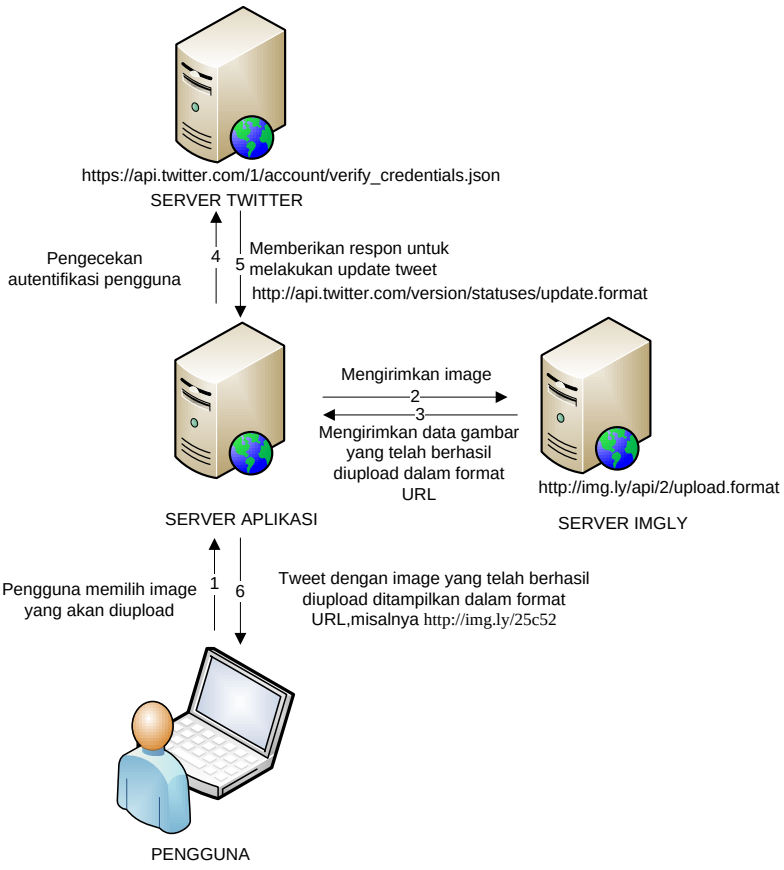
   Fitur ini akan membantu pengguna untuk memperpendek suatu *link (url).* Layanan ini menggunakan server twkl.in untuk memperpendek url. Hasil dari url yang telah dipeperpendek misalnya http://twkl.in/9c . Komunikasi fitur *shorten URL* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.8 .



Gambar 3.8 *Fitur URL Shortener*

2. *Upload image* (unggah gambar)

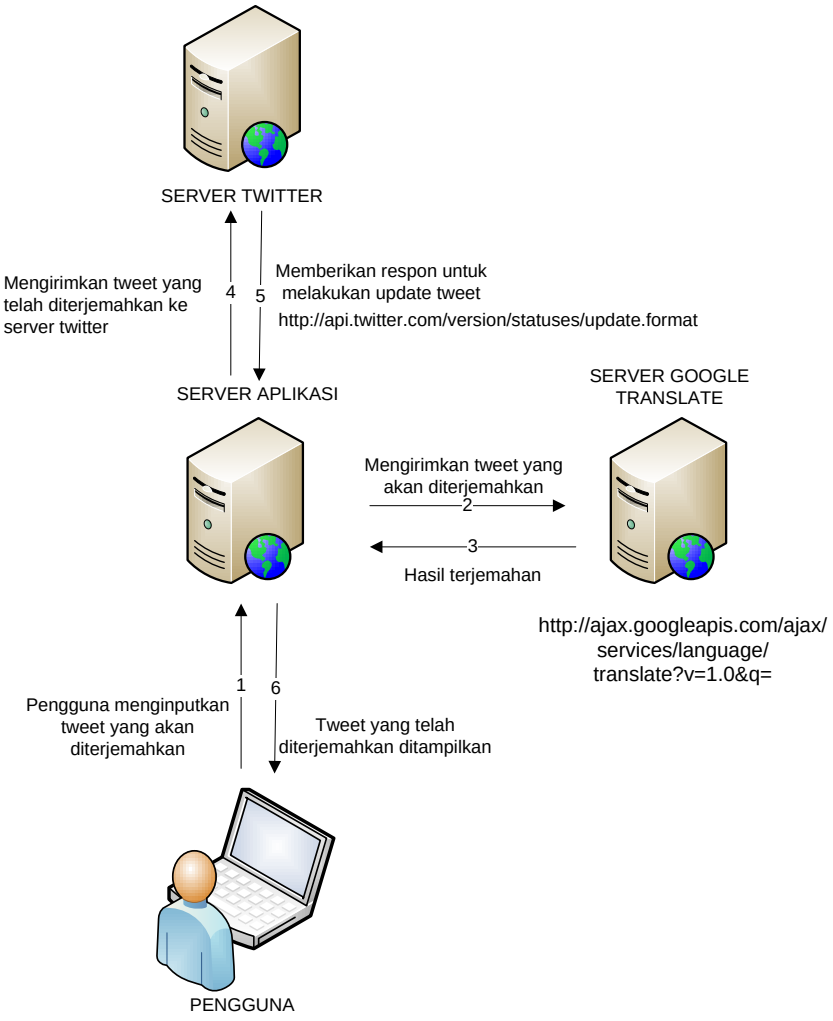
Fitur ini akan memungkinkan pengguna dalam melakukan unggah gambar. Layanan ini menggunakan server img.ly untuk melakukan unggah gambar, dan setiap file gambar yang diupload akan disimpan oleh img.ly. Hasil *upload image* (gambar) dalam format url, misalnya http://img.ly/25c52. Maksimal ukuran gambar yang diunggah adalah 1 MB. Komunikasi fitur *upload image* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.9 .



Gambar 3.9 *Fitur Upload Image*

3. *Translation*

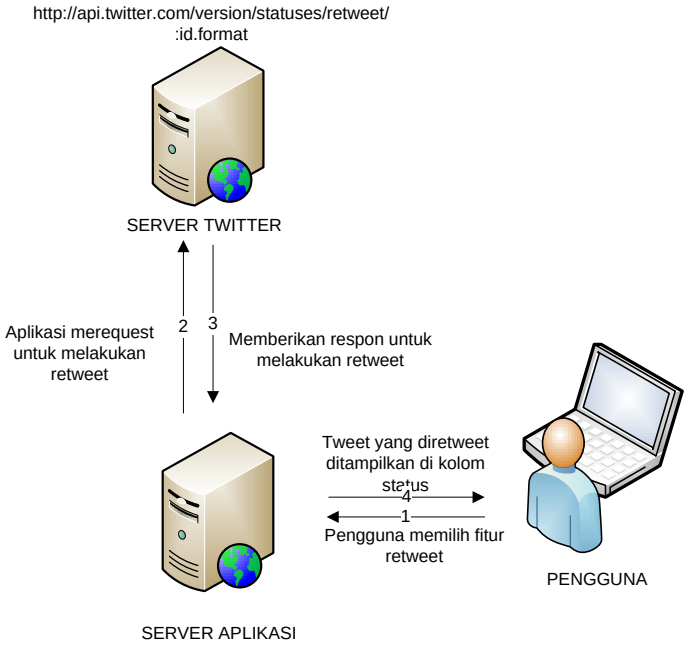
Fitur ini akan membantu pengguna dalam menerjemahkan *tweet* yang akan di posting. Layanan ini akan memberikan pilihan dalam 48 bahasa dengan menggunakan API *google translate*. Komunikasi fitur *translation* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.10 .



Gambar 3.10 *Fitur Translation*

4. *Retweet*

Fitur ini akan memberikan kemudahan bagi pengguna untuk melakukan *update* terhadap *tweet* yang ada di *timeline* dan menambahkan komentar pada *tweet* tersebut namun tetap dibatasi 140 karakter per *update*. Komunikasi fitur *retweet* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.11 .

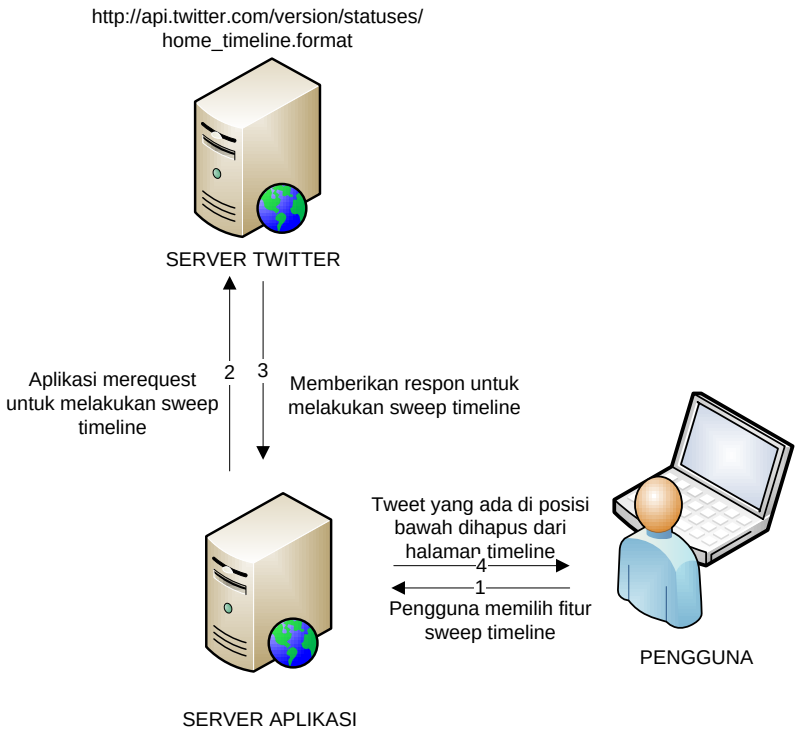


Gambar 3.11 *Fitur Retweet*

5. *Sweep Timeline*

Fitur ini akan memungkinkan pengguna untuk membersihan halaman *timeline*nya (bukan menghapus), agar tidak terlalu memanjang sehingga *tweet* yang telah masuk dan berada di posisi bawah (akhir) tidak muncul lagi di

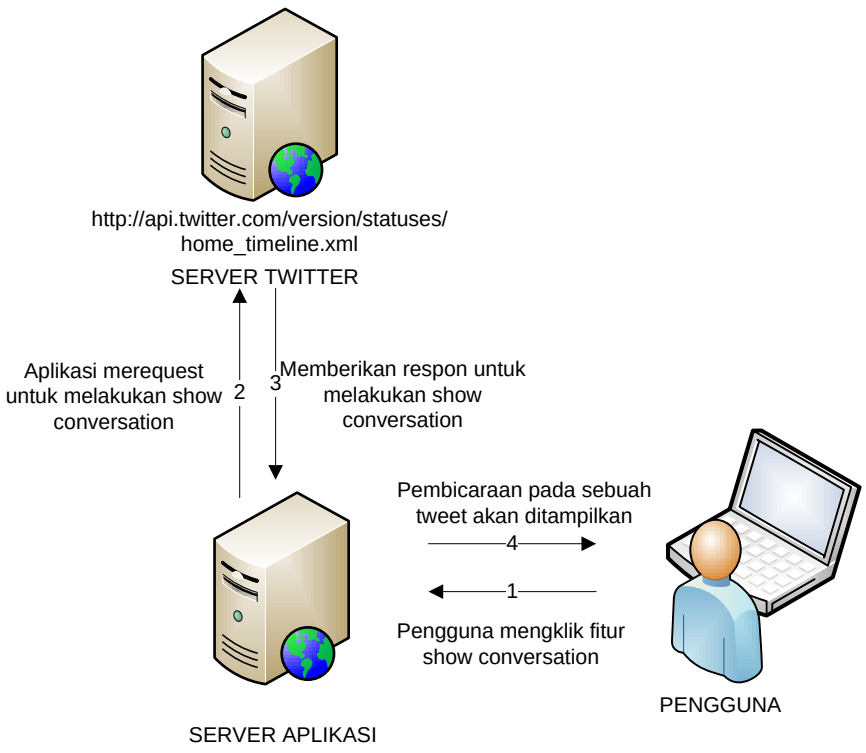
halaman awal *timeline* lagi. Komunikasi fitur *sweep timeline* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.12 .



Gambar 3.12 Fitur Sweep Timeline

6. *Show conversation*

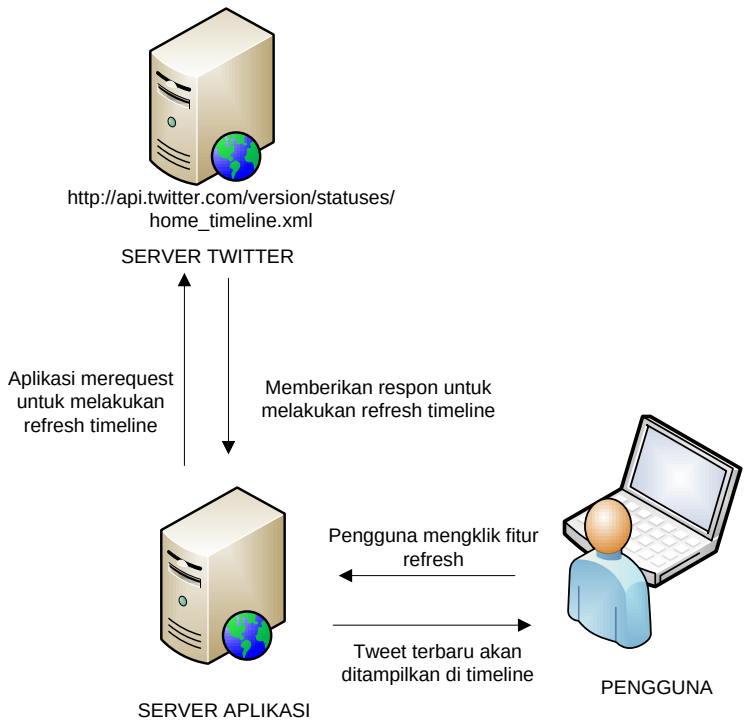
Fitur ini akan menampilkan *tweet* pengguna yang mendapat *'reply'* dari pengguna yang lainnya, dan ditampilkan secara berurutan pada sebuah *tweet* . Komunikasi fitur *show conversation* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.13 .



Gambar 3.13 *Fitur Show Conversation*

7. *Refresh timeline*

Fitur ini akan memberi keleluasaan pada pengguna untuk melakukan *refresh timeline* pada halaman *timeline* tanpa harus menunggu interval waktu *refresh* dari *twitter*. (rentang waktu interval *refresh twitter* sekitar 90 detik). Komunikasi fitur *refresh timeline* pada aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.14 .



Gambar 3.14 *Fitur Refresh Timeline*

3b . *Prototype*

Pada aktivitas ini akan digambarkan secara umum mengenai gambaran aplikasi *twitter client* yang dibangun dengan menggunakan *Low-fidelity Prototyping* yaitu :

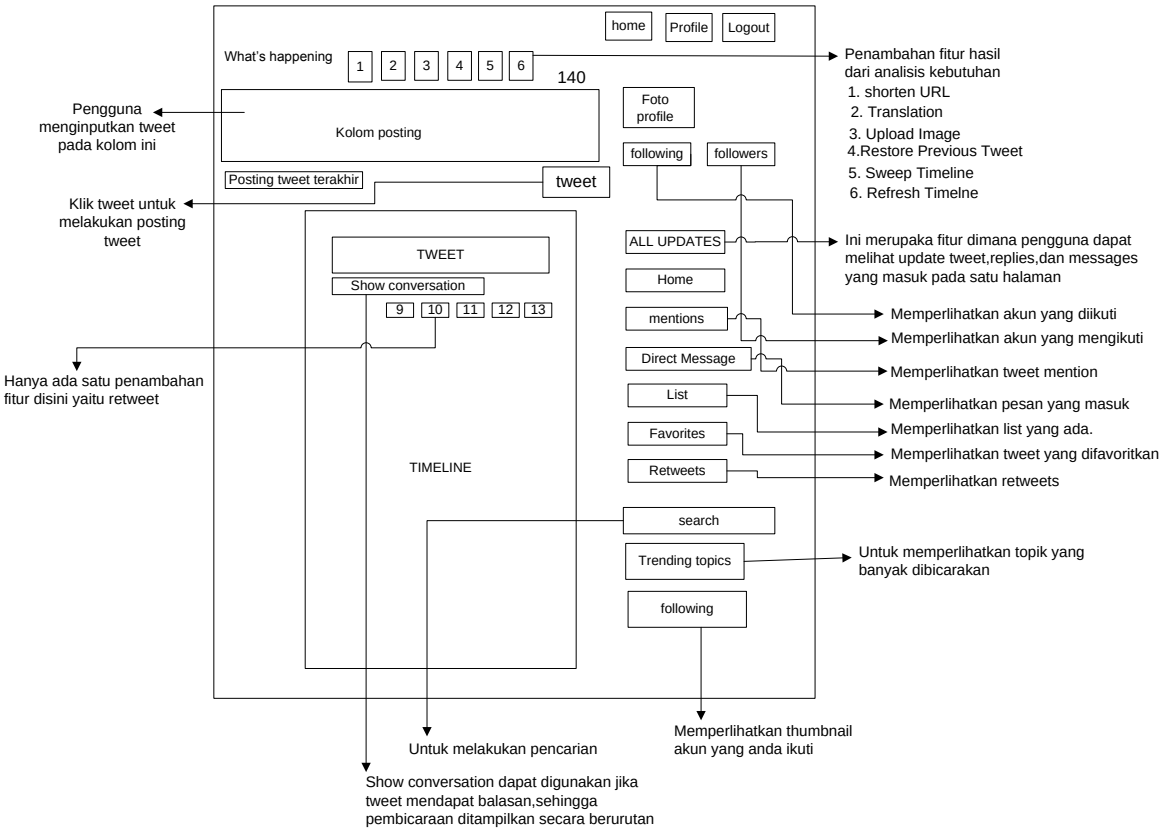
1. Gambaran cepat dari sistem final.
2. Mempunyai fungsi atau interaksi yang terbatas.
3. Lebih menggambarkan konsep, dan layout layar dibanding model interaksi pengguna dengan sistem.
4. Tidak untuk memperlihatkan secara rinci bagaimana operasi sistem aplikasi.

5. Memperlihatkan konsep pendekatan secara umum tanpa harus membuang banyak tenaga, biaya dan waktu.

Low-fidelity Prototyping akan menggambarkan dengan sketsa yang menjelaskan secara singkat mengenai fitur yang ada pada *twitter* dan beberapa fitur yang ditambahkan, sehingga akan memudahkan untuk melihat di bagian mana fitur-fitur yang ditambahkan akan ditempatkan.

1. Sketsa Aplikasi *Twitter Client (*halaman utama)

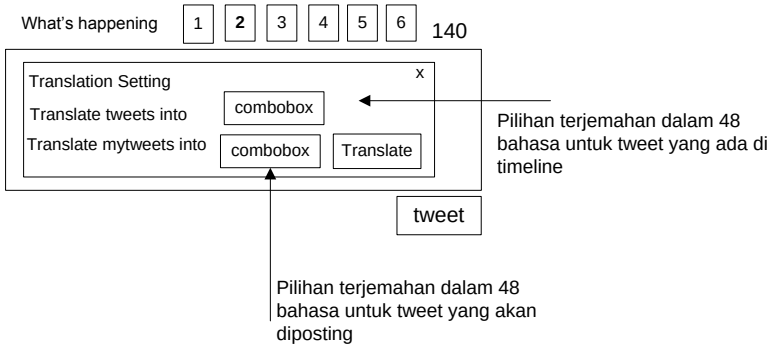
Dibawah ini merupakan sketsa dari halaman utama dari aplikasi *twitter client* yang dibangun. Pada sketsa juga akan dijelaskan secara singkat mengenai fitur-fitur yang digunakan. Sketsa dari halaman utama aplikasi dapat dilihat pada gambar 3.15 .



Gambar 3.15 Sketsa Aplikasi *Twitter Client (*halaman utama)

2. Sketsa *Translation Setting*

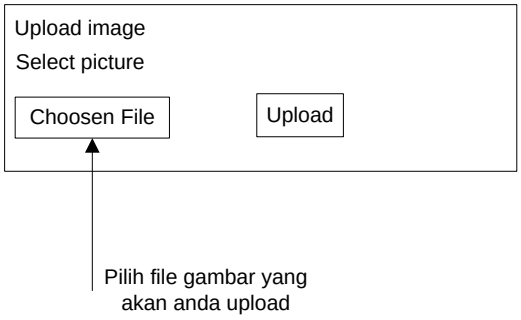
Sketsa dari fitur *translation* dapat dilihat pada gambar 3.16 .



Gambar 3.16 Sketsa *Translation Setting*

3. Sketsa *Upload Image*

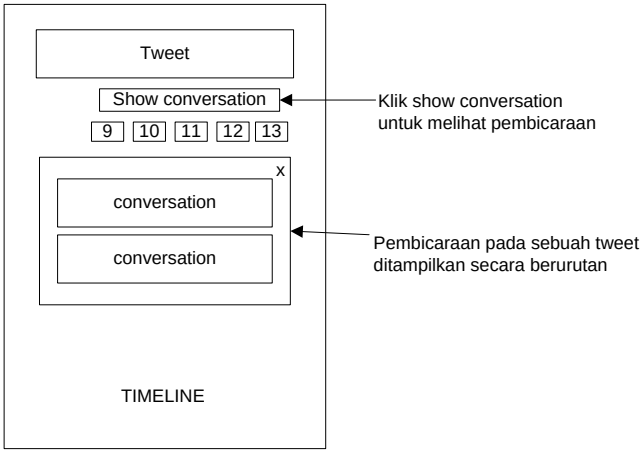
Sketsa dari fitur *upload image* dapat dilihat pada gambar 3.17 .



Gambar 3.17 Sketsa *Upload Image*

4. Sketsa *Show Conversation*

Sketsa dari fitur *show conversation* dapat dilihat pada gambar 3.18 .



Gambar 3.18 Sketsa *Show Conversation*

4. Evaluasi Perancangan

Tahapan ini merupakan kelanjutan dari solusi perancangan dimana hasil solusi yang didapat dari tahap solusi perancangan dapat dievaluasi dan diambil kesimpulan tentang kebutuhan pengguna terhadap fitur yang akan ditambahkan pada *twitter client* yang dibangun. Fitur-fitur tersebut adalah *shorten url, upload image, translation, sweep timeline, refresh, show conversation* dan *retweet.* Berikut evaluasi kegunaan fitur yang dikembangkan pada aplikasi *client* terhadap kebutuhan pengguna :

1. Untuk memberikan keleluasaan pengguna dalam berbagi informasi, aplikasi *client* menghadirkan fitur *upload image* dan *shorten url* sehingga pengguna dapat berbagi informasi dalam format gambar dan berbagi *link* namun tetap menghemat karakter dengan memperpendek *link* yang akan di *tweet.* Pengguna juga dimungkinkan untuk melihat *tweet* dari pengguna yang lain dalam file gambar yang menggunakan layanan *twitpic.com, plixi.com*, dan *flickr.com* karena akan ditampilkan di *timeline.*
2. Untuk pengguna yang mengalami keterbatasan dalam penggunaan bahasa asing, aplikasi *client* memungkinkan pengguna untuk melakukan terjemahan terhadap *tweet* yang akan diposting maupun *tweet* yang ada di *timeline* dengan menghadirkan fitur *translation.* Fitur ini dilengkapi dengan terjemahan ke dalam 48 bahasa.

3. Untuk melihat *update tweet* tanpa harus menunggu *refresh* dari *twitter,* aplikasi *client* menghadirkan fitur *refresh* sehingga pengguna dapat dengan leluasa melihat *update tweet.*
4. Untuk menambahkan komentar pada *tweet* yang ada di *timeline,* aplikasi *client* menghadirkan fitur *new retweet* dan *retweet* sehingga pengguna lebih leluasa melakukan *tweet* baik secara *original* maupun jika ingin menambahkan komentar.
5. Untuk mempermudah pengguna melihat pembicaraan pada *tweet* yang di '*reply*', aplikasi *client* menghadirkan fitur *show conversation.* Fitur ini akan sangat membantu pengguna karena pada *web twitter* sendiri tidak menyediakan fitur yang memungkinkan pengguna melihat pembicaraan yang mendapatkan beberapa kali balasan dari pengguna lainnya.

3.1.3.3 Analisis Metode UCD Terhadap *Usability* (Kegunaan)

Twitter adalah salah satu media sosial mikroblog yang mulai banyak digunakan oleh masyarakat di dunia saat ini. Saat ini *twitter* tidak hanya digunakan sebagai jejaring sosial saja, tapi juga sebagai salah satu media komunikasi yang cukup interaktif dan merupakan salah satu media promosi. Gambar 3.19 adalah gambar petikan yang berisi *tweet* mengenai kegunaan *twitter.*



Gambar 3.19 Petikan Kegunaan *Twitter*

Dari kegunaan *twitter* tersebut, pengguna masih mengalami keterbatasan, salah satunya adalah hanya memberikan kesempatan pada pengguna untuk melakukan *tweet* dalam format *text* saja. Dari proses UCD maka didapat solusi perancangan yang kemudian dilanjutkan dengan evaluasi mengenai fitur yang akan ditambahkan pada aplikasi *client* yang diharapkan dapat meningkatkan kegunaan (*usability) twitter*. Setelah melakukan aktivitas UCD maka dapat ditarik kesimpulan mengenai fitur yang dapat meningkatkan *usability* (kegunaan) *twitter* sebagai salah satu media komunikasi mikroblog. Berikut beberapa kegunaan dari fitur tersebut :

1. Melakukan unggah gambar (*upload image)* dan menampilkannya di *timeline*
2. Menampilkan *tweet* dalam format *image* yang menggunakan layanan server untuk *upload image* yaitu : *plixi, twitpic, flickr,* dan *img.ly* .
3. Memperpendek *url (link)*
4. Melakukan terjemahan(*translation)*

5. Melakukan *retweet*
6. Melihat pembicaraan pada satu *tweet (show conversation)*

Usability pada aplikasi akan diukur dengan beberapa karateristik yaitu: *learnibility*, *efficiency*, dan *satisfaction*. Karateristik tersebut akan diuji pada tahap pengujian sehingga dapat dilakukan pengujian untuk melihat respon pengguna secara langsung. Pengujian yang akan dilakukan adalah dengan melakukan kuisioner terhadap pengguna yang telah menggunakan aplikasi *twitter client* yang dibangun.

3.1.4 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

Pada analisis kebutuhan sistem non fungsional ini dijelaskan analisis mengenai perangkat keras (*hardware),* perangkat lunak (*software),* dan pengguna (*user*) sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan.

3.1.4.1 Analisis Kebutuhan Perangkat Keras (*Hardware*)

Agar aplikasi dapat berjalan dengan baik, maka dibutuhkan perangkat keras yang sesuai dengan kebutuhan aplikasi. Berikut ini adalah spesifikasi minimun perangkat keras yang dibutuhkan oleh *client* agar dapat menjalankan aplikasi secara optimal, yaitu :

1. *Processor* dengan kecepatan 2,8 GHz atau lebih
2. Hard Disk 120 GB.
3. RAM 512 MB
4. Perangkat masukan standar seperti *keyboard* dan *mouse.*

3.1.4.2 Analisis Kebutuhan Perangkat Lunak (*Software)*

Analisis kebutuhan perangkat lunak merupakan analisis kemampuan perangkat lunak untuk menjalankan sistem yang akan dibangun. Standar perangkat lunak yang dibutuhkan untuk mendapatkan hasil yang maksimal adalah sebagai berikut :

1. koneksi internet yang baik
2. browser yang mendukung seperti : *firefox* >3.5 dan *google chrome > 4*
3. resolusi monitor minimal 1024 x 768.

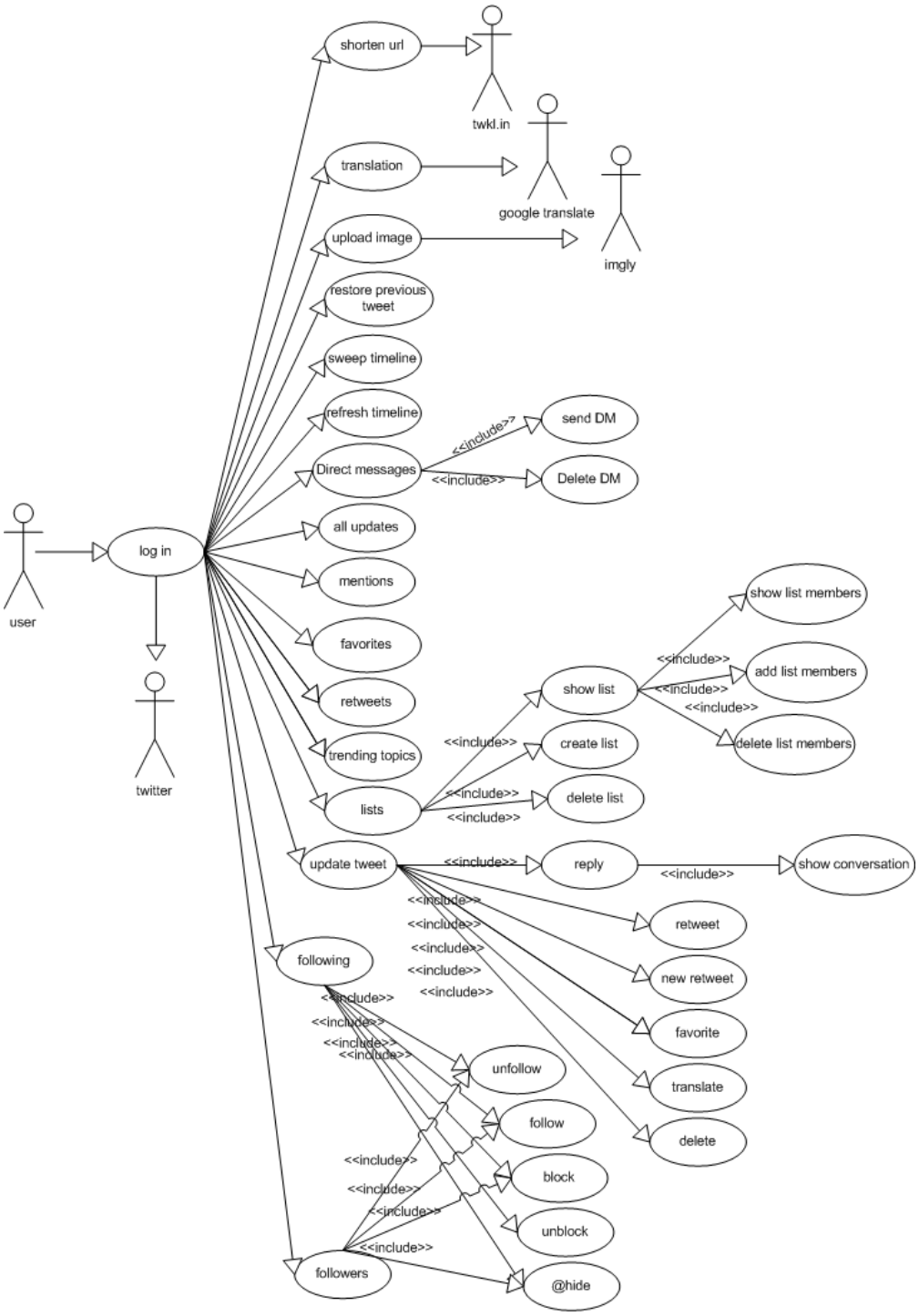
3.1.4.3 Analisis Pengguna

Aplikasi ini tidak membedakan level pengguna , sehingga semua pengguna dapat menggunakan fasilitas yang sama, dengan asumsi telah memiliki *account* di

twitter. Pengalaman pengguna terhadap *web twitter* juga diperlukan yaitu mengerti cara kerja *twitter* sehingga dapat menjalankan sistem yang ada pada *twitter client*. Aplikasi *client* ini bersifat umum dan tidak ditujukan untuk kebutuhan pengguna secara khusus.

3.1.5 Analisis Kebutuhan Fungsional

Kebutuhan fungsional akan digambarkan dengan beberapa diagram yaitu *use case diagram*, *sequence diagram*, dan *class diagram*. *Use case diagram* pada aplikasi yang akan dibangun dapat dilihat pada gambar 3.20 :



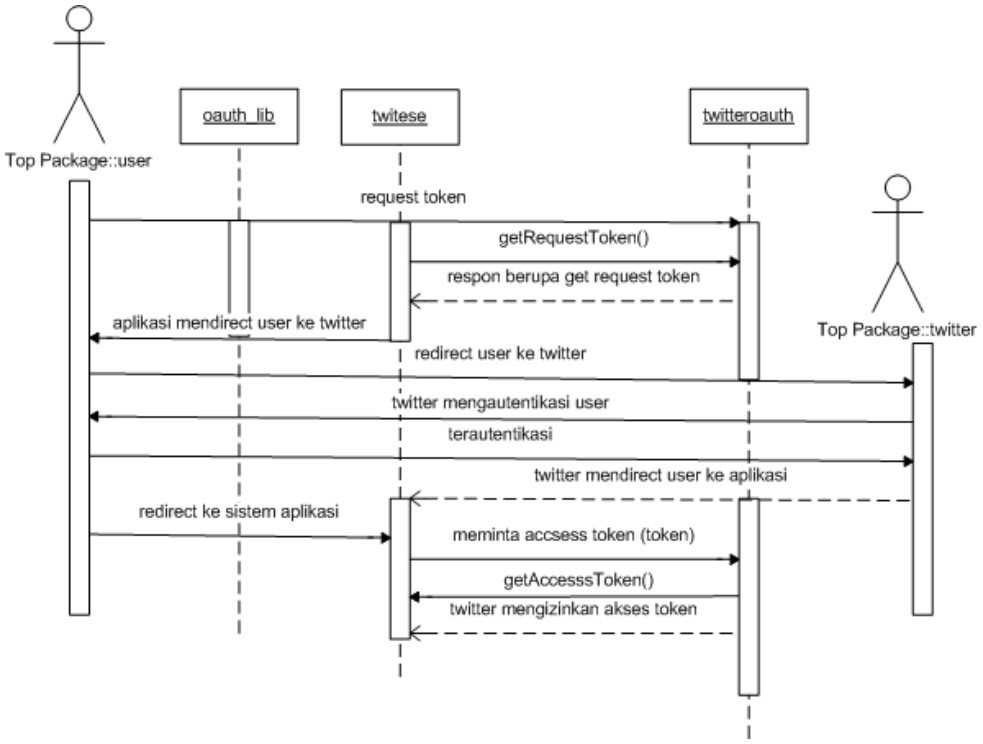
Gambar 3.20 Use Case Diagram

3.1.5.1 Login

Skenario dari proses login dapat dilihat pada tabel 3.9 dan diagram sequence login pada gambar 3.21.

3.9 Tabel Skenario Login

Identifikasi		
Nomor	1	
Nama	Login	
Tujuan	Autentikasi pengguna	
Deskripsi	User memilih log in, sistem menerima data identifikasi pengguna untuk masuk ke aplikasi	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	Pengguna belum terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. user klik log in		2.Melakukan request token
3.Twitter memberikan request token		4.Me-*redirect* pengguna ke twitter
5.Pengguna menginputkan username dan password		
6.Twitter mengautentikasi pengguna		7.Sistem mengirimkan data pengguna ke twitter
8.Pengguna terautentikasi		
9.Twitter me-*redirect* pengguna ke sistem aplikasi		
10.Pengguna te-*redirect* ke sistem aplikasi		11.Sistem Aplikasi meminta *access token*
12.Twitter mengizinkan access token		
Kondisi akhir	user berhasil log in	



Gambar 3.21 Diagram Sequence Login

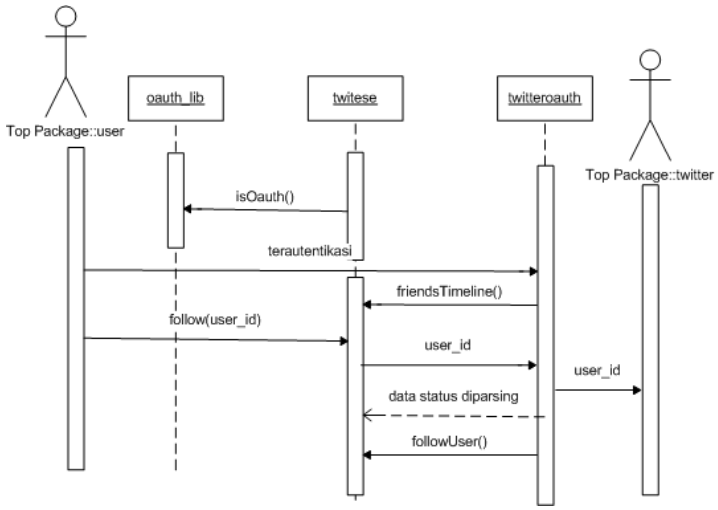
3.1.5.2 Follow

Skenario follow dapat dilihat pada tabel 3.10 dan diagram sequence follow pada gambar 3.22 .

3.10 Tabel Skenario Follow

Identifikasi		
Nomor	2	
Nama	Follow	
Tujuan	Melakukan Follow user	
Deskripsi	Twitter mengautentikasi pengguna , lalu pengguna melakukan follow dan sistem aplikasi mengirimkan ke twitter menampilkan ke tampilan	
Tipe		
Aktor	User, Twitter	
Skenario Utama		
Kondisi awal	User telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**

1.twitter mengautentikasi user	2. Sistem Aplikasi mendapat *access token*
3.user memilih menu follow	4. sistem mengirimkan user_id yang akan difollow ke twitter
5.Twitter menerima user_id yang akan difollow	6.Sistem meminta status dari twitter
	7.Data status mentah diparsing
	8.Status ditampilkan ke user
Kondisi akhir	Status ditampilkan dengan kondisi user bersangkutan ditambahkan sebagai orang yang difollow



Gambar 3.22 Diagram Sequence Follow

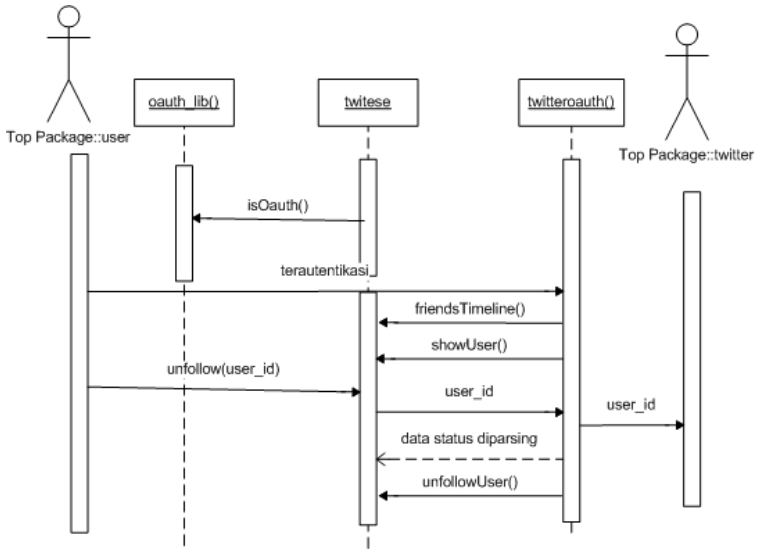
3.1.5.3 Unfollow

Skenario unfollow dapat dilihat pada tabel 3.11 dan diagram sequence unfollow pada gambar 3.23 .

3.11 Tabel Skenario Unfollow

Identifikasi	
Nomor	3
Nama	Unfollow
Tujuan	Melakukan Unfollow user
Deskripsi	Twitter mengautentikasi pengguna , lalu user melakukan unfollow dan sistem aplikasi mengirimkan ke twitter menampilkan ke tampilan

Tipe		
Aktor	User, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. Twitter melakukan autentikasi		2. Sistem Aplikasi meminta *access token*
3.user memilih menu unfollow		4.user_id yang akan diunfollow dikirimkan ke twitter
5.Twitter menerima user_id		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi user bersangkutan bukan lagi sebagai orang yang diunfollow	



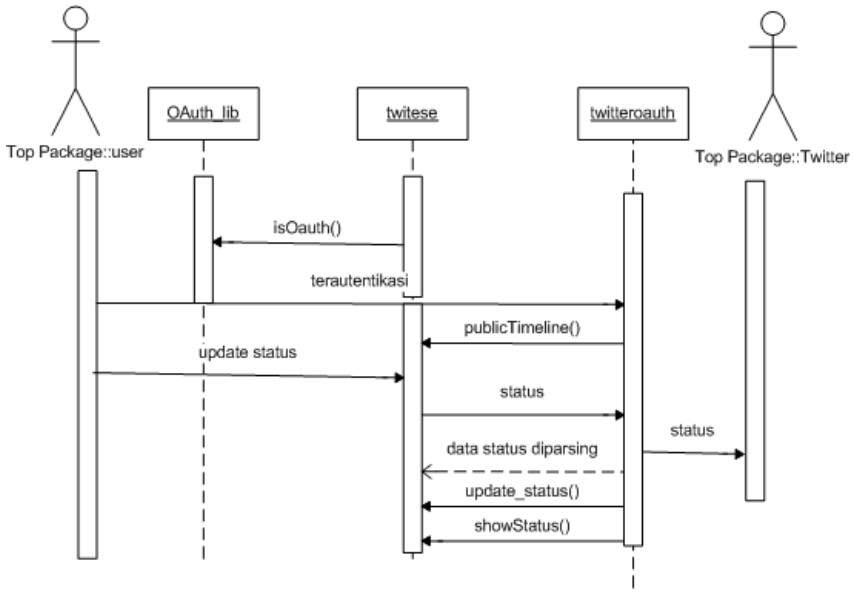
Gambar 3.23 Diagram Sequence Unfollow

1.1.5.4 Update Tweet

Skenario update tweet dapat dilihat pada tabel 3.12 dan diagram sequence update tweet pada gambar 3.24 .

3.12 Tabel Skenario Update Tweet

Identifikasi		
Nomor	4	
Nama	Update tweet	
Tujuan	Melakukan Update tweet	
Deskripsi	Twitter mengautentikasi user, lalu user melakukan update tweet dan sistem aplikasi mengirimkan ke twitter menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi pengguna		2. Sistem Aplikasi meminta *access token*
3. Pengguna menginput tweet dan memilih menu update status		4.Status (tweet) dikirimkan ke twitter
5.Twitter menerima status		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status (tweet) update ditampilkan	



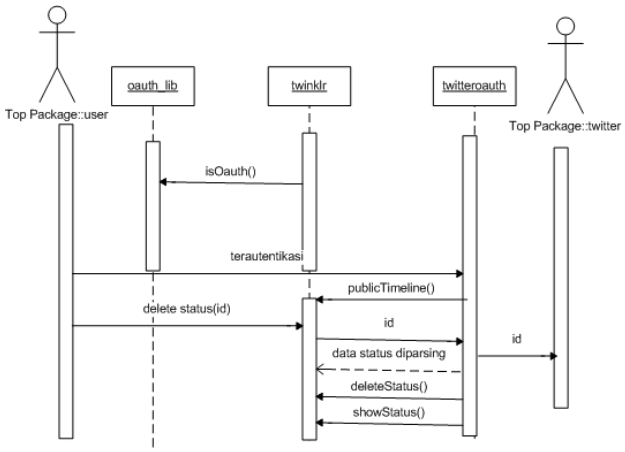
Gambar 3.24 Diagram Sequence Update Tweet

3.1.5.5 Delete Tweet

Skenario delete tweet dapat dilihat pada tabel 3.13 dan diagram sequence delete tweet pada gambar 3.25 .

3.13 Tabel Skenario Delete Tweet

Identifikasi		
Nomor	5	
Nama	Delete Tweet	
Tujuan	Melakukan Delete tweet	
Deskripsi	Twitter mengautentikasi user , lalu user melakukan delete tweet dan sistem aplikasi mengirimkan ke twitter menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih menu delete		4.id status yang akan didelete dikirimkan ke twitter
5.Twitter menerima id status		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan status yang telah terhapus.	



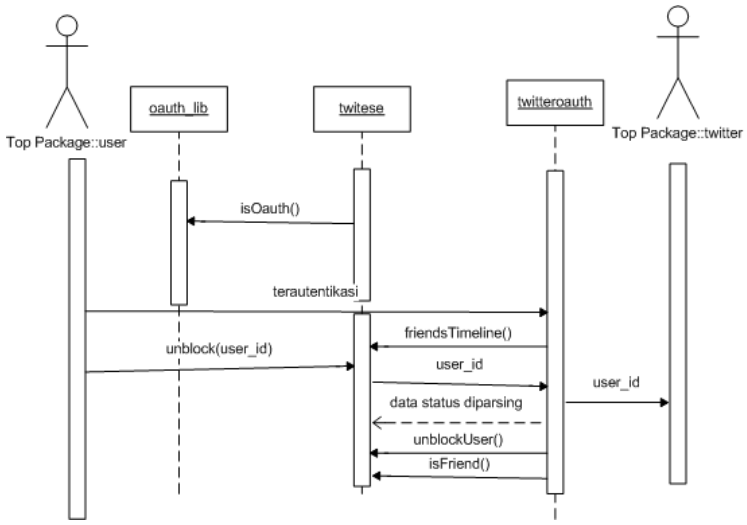
Gambar 3.25 Diagram Sequence Delete Tweet

3.1.5.6 Unblock

Skenario unblock dapat dilihat pada tabel 3.14 dan diagram sequence unblock pada gambar 3.26 .

3.14 Tabel Skenario Unblock

Identifikasi		
Nomor	6	
Nama	Unblock	
Tujuan	Melakukan Unblock user	
Deskripsi	Twitter mengautentikasi user , lalu user melakukan unblock dan sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
2.user memilih menu unblock		4. user_id yang akan diunblock dikirimkan ke twitter
4.Twitter menerima user_id		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi user bersangkutan bukan lagi sebagai user yang di block	



Gambar 3.26 Diagram Sequence unblock

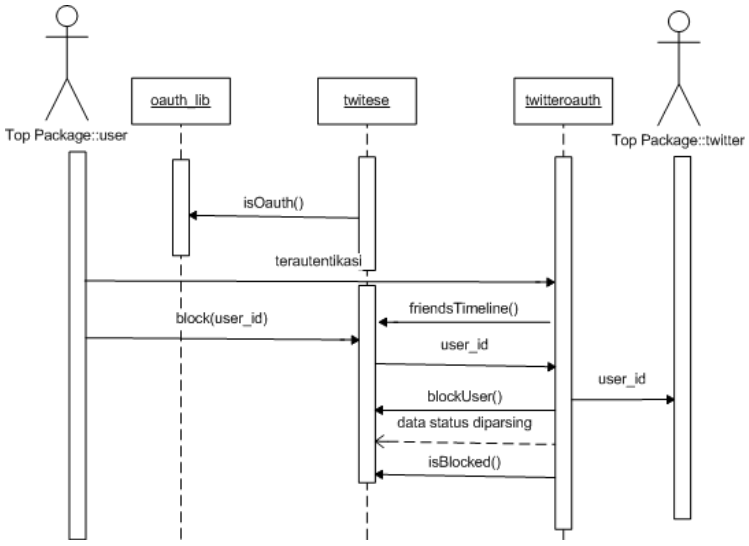
3.1.5.7 Block

Skenario block dapat dilihat pada tabel 3.15 dan diagram sequence block pada gambar 3.27 .

3.15 Tabel Skenario Block

Identifikasi		
Nomor	7	
Nama	Block	
Tujuan	Melakukan Block user	
Deskripsi	Twitter mengautentikasi user , lalu user melakukan block user dan sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. Twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3. User memilih menu block		4. id user yang akan diblock dikirimkan ke twitter
5.Twitter menerima id		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing

	8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi user bersangkutan bukan lagi sebagai follower atau status diblock.



Gambar 3.27 Diagram Sequence Block

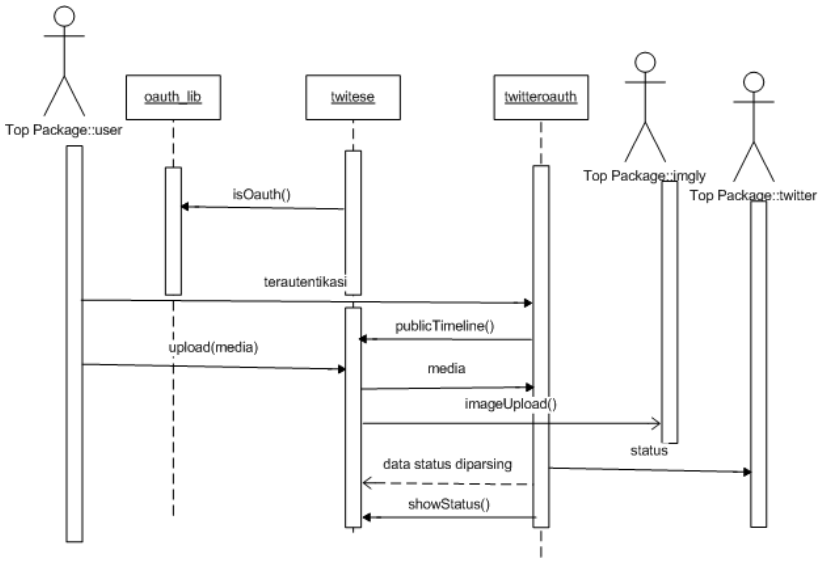
3.1.5.8 Upload Image

Skenario dari upload image dapat dilihat pada tabel 3.16 dan diagram sequence upload image pada gambar 3.28 .

3.16 Tabel Skenario Upload Image

Identifikasi	
Nomor	8
Nama	Upload Image
Tujuan	Melakukan unggah gambar
Deskripsi	Twitter mengautentikasi user, lalu user melakukan unggah gambar ke imgly lalu update status dan sistem aplikasi menampilkan ke tampilan.
Tipe	
Aktor	user, Twitter, img.ly
Skenario Utama	
Kondisi awal	user telah terautentikasi

Aksi Aktor	Reaksi Sistem
1. twitter mengautentikasi user	2. Sistem Aplikasi meminta *access token*
3.user memilih menu upload image	4.media dikirimkan ke imgly
5.Imgly menerima media	6.status dikirimkan ke twitter
7.Twitter menerima status	8.Sistem meminta status dari twitter
	9.Data status mentah diparsing
	10.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan menampilkan gambar



Gambar 3.28 Diagram Sequence Upload Image

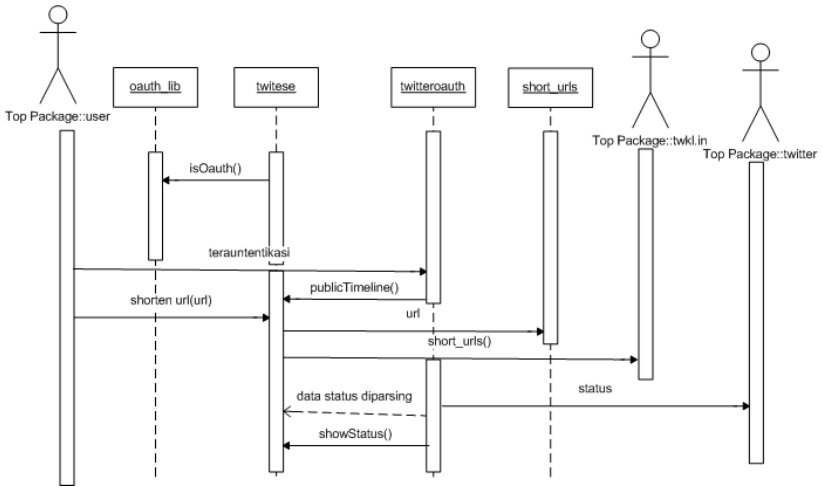
3.1.5.9 Shorten URL

Skenario dari shorten URL dapat dilihat pada tabel 3.17 dan diagram sequence shorten URL pada gambar 3.28 .

3.17 Tabel Skenario Shorten url

Identifikasi	
Nomor	9
Nama	Shorten URL
Tujuan	Memperpendek link
Deskripsi	Twitter mengautentikasi user , lalu user memperpendek sebuah link dan mengirimkan ke twkl.in lalu update status

	dan sistem aplikasi mengirimkan ke twitter lalu menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter, twkl.in	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user menginputkan link yang akan dipendekkan		4. link dikirimkan ke twkl.in
5.Twkl.in menerima link		6.status dan link dikirimkan ke twitter
7.Twitter menerima status		8.Sistem meminta status dari twitter
		9.Data status mentah diparsing
		10.Status ditampilkan ke pengguna dengan link yang sudah dipendekkan
Kondisi akhir	Status ditampilkan dengan menampilkan link yang sudah dipendekkan.	



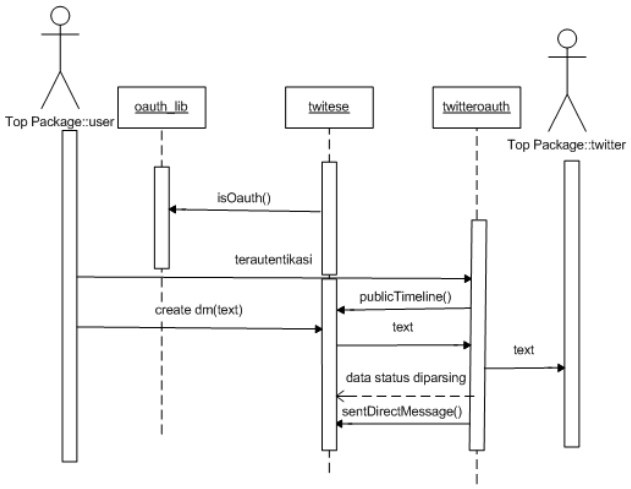
Gambar 3.29 Diagram Sequence Shorten URL

3.1.5.10 Create DM

Skenario create DM dapat dilihat pada tabel 3.18 dan diagram sequence create DM pada gambar 3.20 .

3.18 Tabel Skenario Create DM

Identifikasi		
Nomor	10	
Nama	Create DM	
Tujuan	Mengirim pesan	
Deskripsi	Twitter mengautentikasi user , lalu user menginputkan pesan dalam format text lalu sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih menu create DM dan menginputkan pesan (text)		4.text dikirimkan ke twitter
5.Twitter menerima text		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi DM telah terkirim	



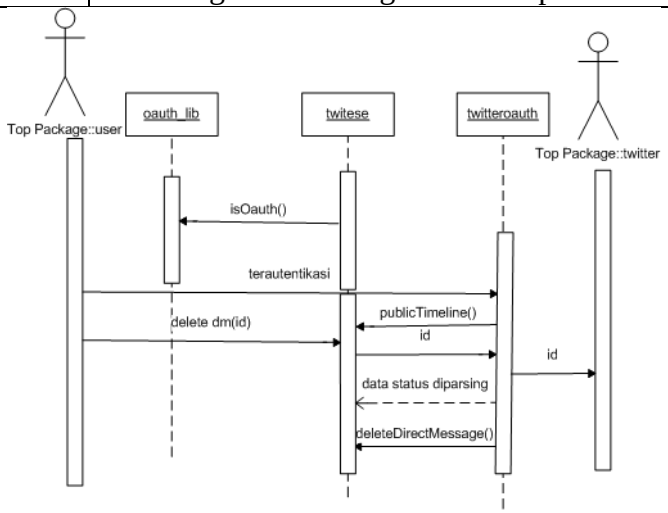
Gambar 3.30 Diagram Sequence Create DM

3.1.5.11 Delete DM

Skenario delete DM dapat dilihat pada tabel 3.19 dan diagram sequence delete DM pada gambar 3.31.

3.19 Tabel Skenario Delete DM

Identifikasi		
Nomor	11	
Nama	Delete DM	
Tujuan	Melakukan penghapusan pesan	
Deskripsi	Twitter mengautentikasi user , lalu user melakukan penghapusan pesan dan sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih menu delete DM		4.Id DM (pesan) yang akan dihapus dikirimkan ke twitter
5.Twitter menerima id DM		6.Sistem meminta status dari twitter
		7.Data mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	DM dengan id bersangkutan terhapus	



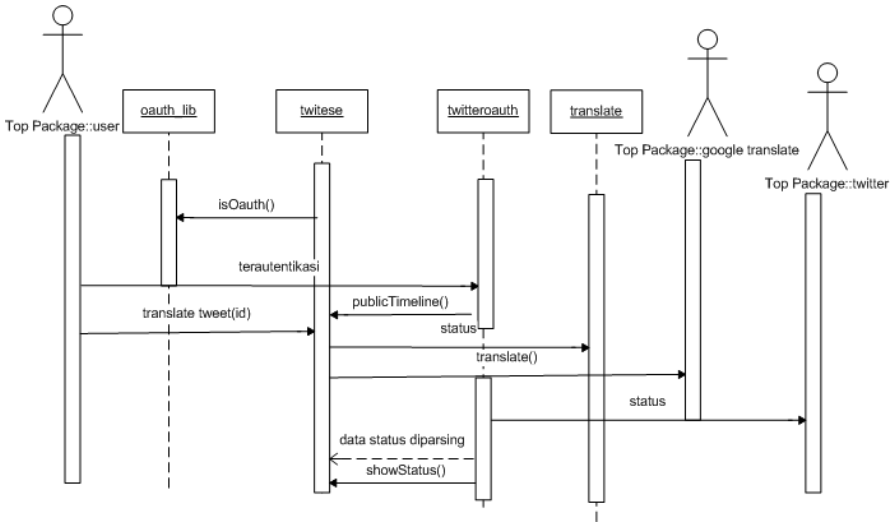
Gambar 3.31 Diagram Sequence Delete DM

3.1.5.12 Translation

Skenario dari translation dapat dilihat pada tabel 3.20 dan diagram sequence translation pada gambar 3.32 .

3.20 Tabel Skenario Translation

Identifikasi		
Nomor	12	
Nama	Translation	
Tujuan	Menterjemahkan status yang ada di timeline	
Deskripsi	Twitter mengautentikasi user , lalu user mengirim status yang dipilih pengguna ke google translate dan sistem aplikasi menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter,google translate	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih menu translate pada salah satu tweet yang ada ditimeline		4.status (id) dikirim ke google translate
5.google translate menerima status		6.status (id) dikirimkan ke twitter
7.Twitter menerima status		8.Sistem meminta status dari twitter
		9.Data status mentah diparsing
		10.Status ditampilkan ke pengguna dengan hasil terjemahan
Kondisi akhir	Status ditampilkan dengan menampilkan status yang telah diterjemahkan.	



Gambar 3.32 Diagram Sequence Translation

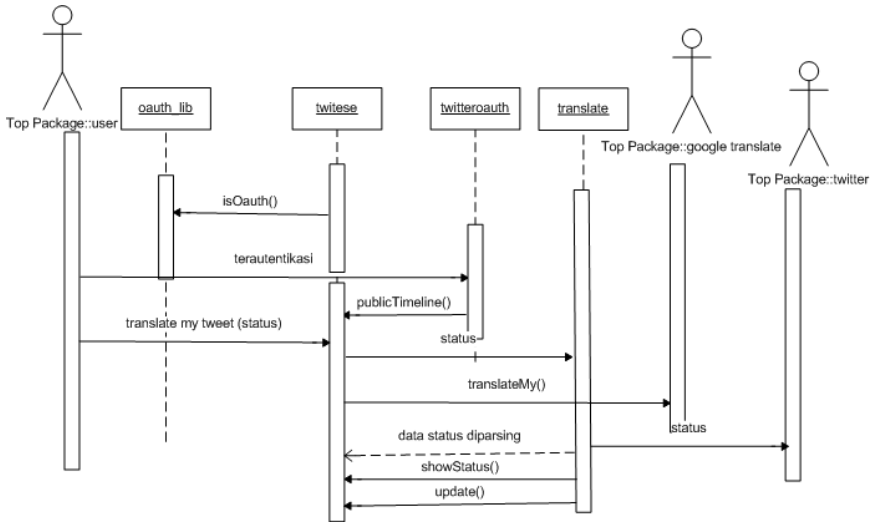
3.1.5.13 Translation My Tweet

Skenario dari translation my tweet dapat dilihat pada tabel 3.21 dan diagram sequence translate my tweet pada gambar 3.33 .

3.21 Tabel Skenario Translation My Tweet

Identifikasi		
Nomor	13	
Nama	Translate my tweet	
Tujuan	Menterjemahkan status yang akan di posting	
Deskripsi	Twitter mengautentikasi user, lalu user menginputkan tweet yang akan diterjemahkan,sistem mengirim status ke google translate dan sistem aplikasi menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter,google translate	
Skenario Utama		
Kondisi awal	User telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user menginputkan status yang akan diterjemahkan dan memilih menu translate		4.status dikirim ke google translate

5.google translate menerima status	6.status dikirimkan ke twitter
7.Twitter menerima status	8.Sistem meminta status dari twitter
	9.Data status mentah diparsing
	10.Status ditampilkan ke pengguna dengan hasil terjemahan
Kondisi akhir	Status ditampilkan dengan menampilkan status yang telah diterjemahkan.



Gambar 3.33 Diagram Sequence Translation My Tweet

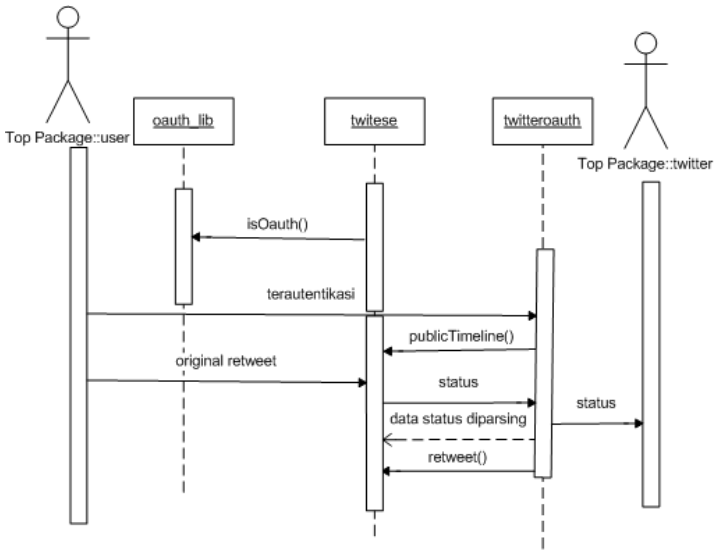
3.1.5.14 Original Retweet

Skenario dari original retweet dapat dilihat pada tabel 3.22 dan diagram sequence retweet pada gambar 3.34 .

3.22 Tabel Skenario Original Retweet

Identifikasi	
Nomor	14
Nama	Original Retweet
Tujuan	Melakukan Original Retweet
Deskripsi	Twitter mengautentikasi user , lalu user memilih status yang akan diretweet (original), kemudian sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan
Tipe	
Aktor	user, Twitter

Skenario Utama		
Kondisi awal	user sudah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih status yang diretweet		4.id status dikirimkan ke twitter
4.Twitter menerima nama user		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan original retweet yang dipilh oleh pengguna.	



Gambar 3.34 Diagram Sequence Original Retweet

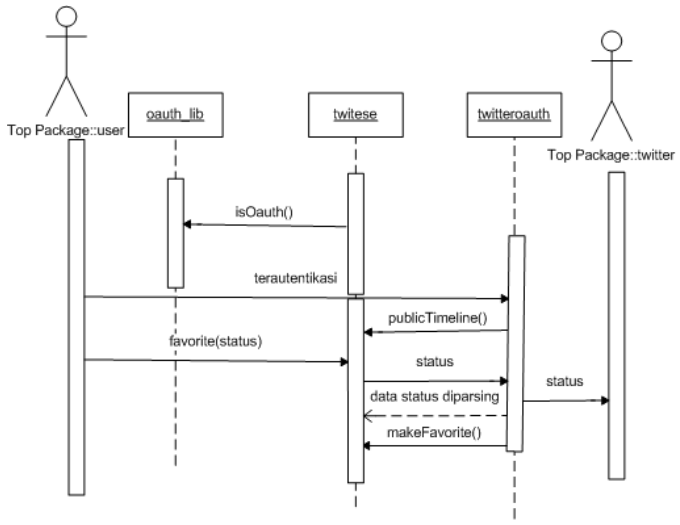
3.1.5.15 Favorite

Skenario favorite dapat dilihat pada tabel 3.23 dan diagram sequence favorite pada gambar 3.35 .

3.23 Tabel Skenario Favorite

Identifikasi	
Nomor	15
Nama	Favorite

Tujuan	Menambahkan status sebagai favorite	
Deskripsi	Twitter mengautentikasi user , lalu user memilih favorite tweet dan sistem aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2 Sistem Aplikasi meminta *access token*
3.user memilih salah satu tweet untuk difavoritkan		4. status yang difavoritkan dikirimkan ke twitter
5.Twitter menerima status		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi status tersebut telah difavoritkan	



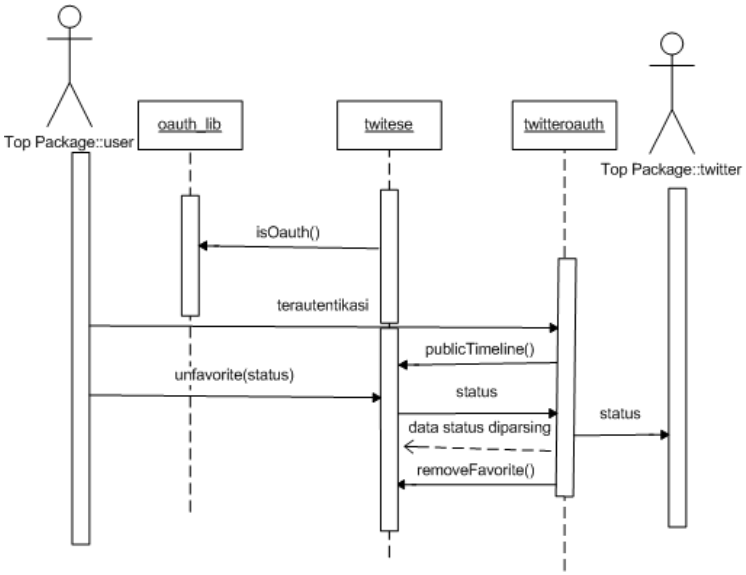
Gambar 3.35 Diagram Sequence Favorite

1.5.5.16 Unfavorite

Skenario unfavorite dapat dilihat pada tabel 3.24 dan diagram sequence unfavorite pada gambar 3.36 .

3.24 Tabel Skenario Unfavorite

Identifikasi		
Nomor	16	
Nama	Unfavorite Status	
Tujuan	Melakukan unfavorite terhadap status	
Deskripsi	Twitter mengautentikasi user , lalu user melakukan unfavorite user dan sistema aplikasi mengirimkan ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih status yang diunfavoritkan		4. status yang diunfavoritkan dikirimkan ke twitter
5.Twitter menerima id status		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan kondisi status dengan id bersangkutan tidak terfavoritkan lagi	



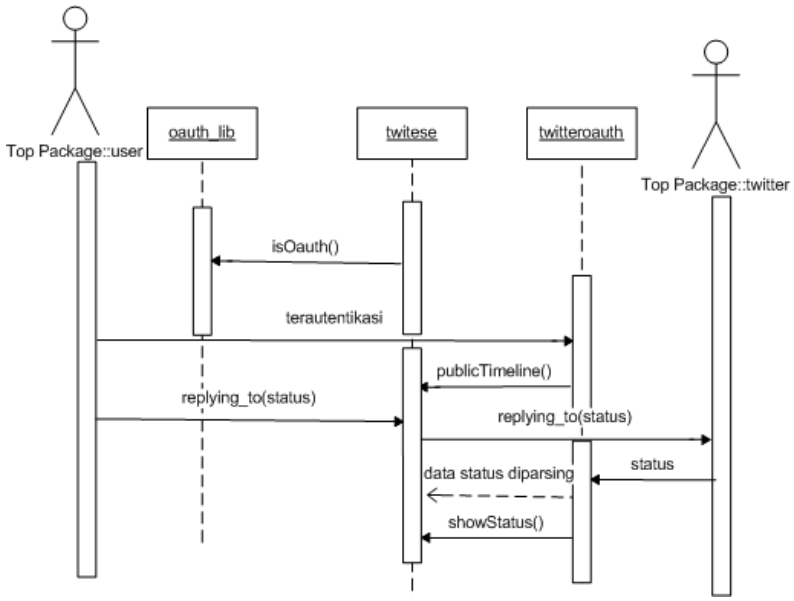
Gambar 3.36 Diagram Sequence Unfavorite

3.1.5.17 Show Conversation

Skenario show conversation dapat dilihat pada tabel 3.25 dan diagram sequence show conversation pada gambar 3.37 .

3.25 Tabel Skenario Show Conversation

Identifikasi		
Nomor	17	
Nama	Show Conversation	
Tujuan	Menampilkan pembicaraan pada sebuah tweet yang mendapat balasan dari pengguna yang lain.	
Deskripsi	Twitter mengautentikasi user , user memilih menu show conversation pada status yang dipilih lalu sistem aplikasi meminta data status ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3. pengguna memilih menu show conversation pada sebuah status		
		4.status dikirimkan ke twitter
5.Twitter mengirim status		
		6.Data status mentah diparsing oleh sistem
		7.Status ditampilkan ke pengguna dengan hasil terjemahan
Kondisi akhir	Pembicaraan pada status yang mendapat balasan dari pengguna yang lain ditampilkan di tampilan.	



Gambar 3.37 Diagram Sequence Show Conversation

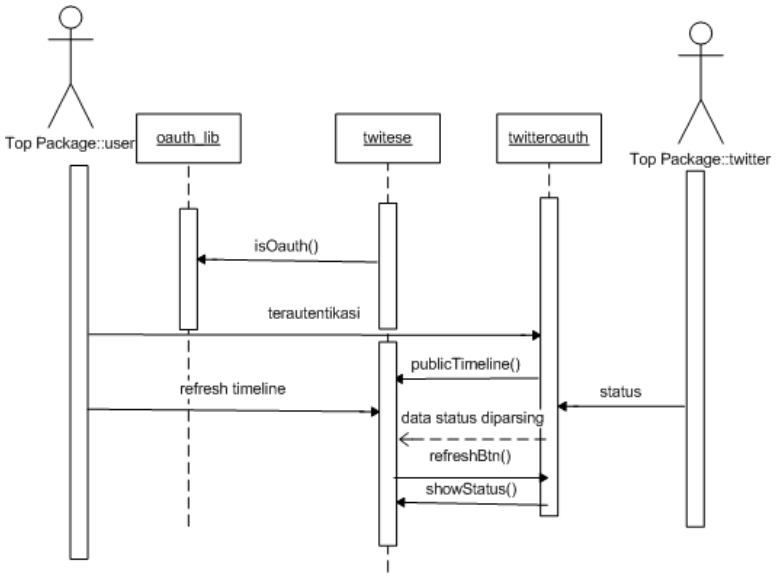
3.1.5.18 Refresh Timeline

Skenario refresh timeline dapat dilihat pada tabel 3.26 dan diagram sequence refresh timeline pada gambar 3.38.

3.26 Tabel Skenario Refresh Timeline

Identifikasi		
Nomor	18	
Nama	Refresh Timeline	
Tujuan	Menampilkan timeline dengan status terbaru	
Deskripsi	Twitter mengautentikasi user , pengguna memilih menu refresh timeline pada status yang dipilih lalu twitter mengirim status dan sistem aplikasi menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	User telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*

3. pengguna memilih menu refresh timeline	
	4.status timeline dikirimkan ke twitter
5.Twitter mengirim status ke sistem aplikasi	
	6.Data status mentah diparsing oleh sistem aplikasi
	7.Status ditampilkan ke pengguna dengan timeline tweet terupdate
Kondisi akhir	Timeline ditampilkan dengan status terbaru



Gambar 3.38 Diagaram Sequence Refresh Timeline

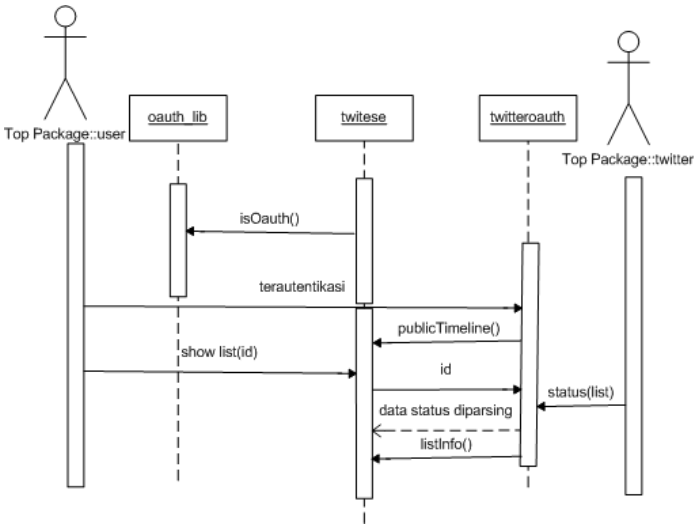
3.1.5.19 Lists

Skenario lists dapat dilihat pada tabel 3.27 dan diagram sequence lists pada gambar 3.39.

3.27 Tabel Skenario Lists

Identifikasi	
Nomor	19
Nama	Lists

Tujuan	Menampilkan Lists	
Deskripsi	Twitter mengautentikasi user , lalu user memilih list,sistem aplikasi meminta list ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
4.user memilih menu list		4.id list dikirimkan ke twitter
5.Twitter mengirimkan id list ke sistem aplikasi		
		6.Data status mentah diparsing oleh sistem aplikasi
		7.Status ditampilkan ke pengguna
Kondisi akhir	List ditampilkan	



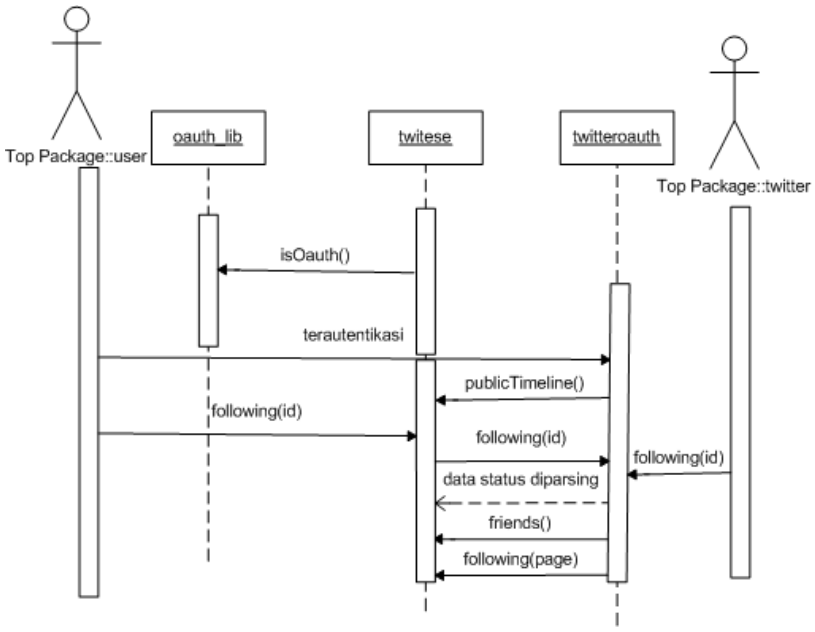
Gambar 3.39 Diagram Sequence Lists

3.1.5.20 Following

Skenario following dapat dilihat pada tabel 3.28 dan diagram sequence following pada gambar 3.40 .

3.28 Tabel Skenario Following

Identifikasi		
Nomor	20	
Nama	Following	
Tujuan	Melihat user yang difollow	
Deskripsi	Twitter mengautentikasi user , user memilih menu following lalu sistem aplikasi meminta data user yang difollow ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3. user mengirimkan id following		4.id user following dikirimkan ke twitter
5.Twitter mengirimkan id yang follow ke sistem aplikasi		
		6.Data status mentah diparsing sistem aplikasi
		7.Status ditampilkan ke pengguna
Kondisi akhir	User yang difollow ditampilkan	



Gambar 3.40 Diagram Sequence Following

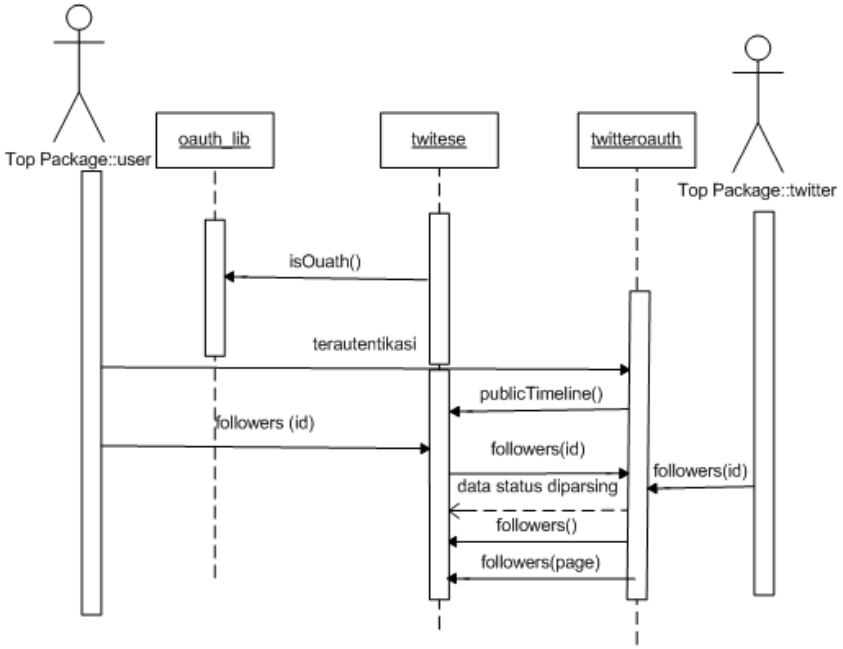
3.1.5.21 Followers

Skenario followers dapat dilihat pada tabel 3.29 diagram sequence followers pada dan gambar 3.41 .

3.29 Tabel Skenario Followers

Identifikasi		
Nomor	21	
Nama	Followers	
Tujuan	Melihat user yang mengikuti pengguna	
Deskripsi	Twitter mengautentikasi user,lalu user memilih menu follower dan sistem aplikasi meminta data follower ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*

3.user mengirimkan id follower	4.id follower pengguna dikirimkan ke twitter
5.Twitter mengirimkan id follower user ke sistem aplikasi	
	6.Data status mentah diparsing sistem aplikasi
	7.Status ditampilkan ke pengguna
Kondisi akhir	Follower ditampilkan



Gambar 3.41 Diagram Sequence Followers

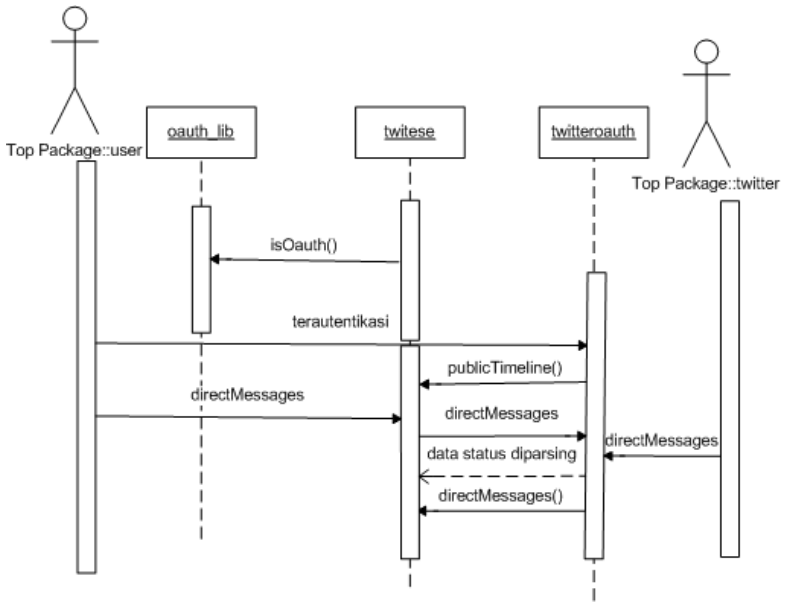
3.1.5.22 DM

Skenario DM dapat dilihat pada tabel 3.30 dan diagram sequence DM pada gambar 3.42 .

3.30 Tabel Skenario DM

Identifikasi	
Nomor	22
Nama	DM
Tujuan	Melihat isi DM (pesan)
Deskripsi	Twitter mengautentikasi user , lalu user memilih menu DM,sistem aplikasi meminta data DM ke twitter dan

	menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3. user mengirimkan id DM		4.id pengguna dikirimkan ke twitter
5.Twitter mengirimkan direct messages ke sistem aplikasi		
		6.Data status mentah diparsing sistem aplikasi
		7.Status ditampilkan ke pengguna
Kondisi akhir	DM ditampilkan	



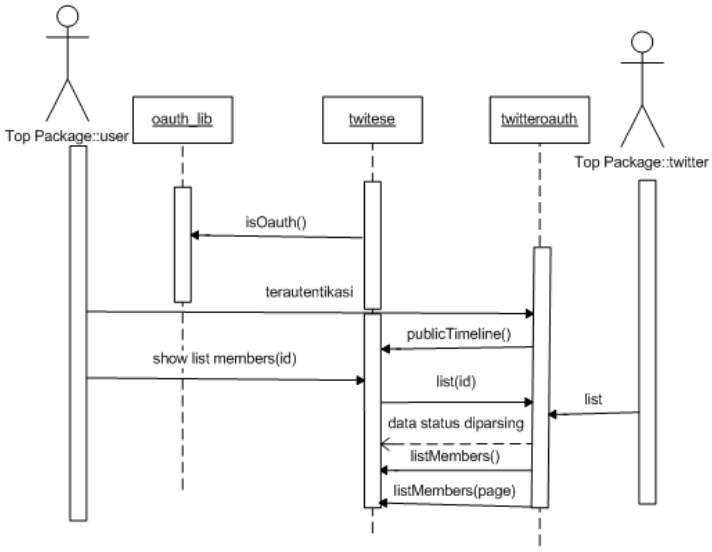
Gambar 3.42 Diagram Sequence DM

3.1.5.23 List Members

Skenario list members dapat dilihat pada tabel 3.31 dan diagram sequence list members pada gambar 3.43 .

3.31 Tabel Skenario List members

Identifikasi		
Nomor	23	
Nama	List Members	
Tujuan	Melihat Anggota List	
Deskripsi	Twitter mengautentikasi user , lalu user memilih menu show list members dan sistem aplikasi meminta data anggota list ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	User sudah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user mengirimkan nama list		4.Nama list dikirimkan ke twitter
5.Twitter mengirimkan nama user pada sebuah list ke sistem aplikasi		
		6.Data data mentah diparsing sistem aplikasi
		7.Data ditampilkan ke pengguna
Kondisi akhir	Data anggota list ditampilkan	



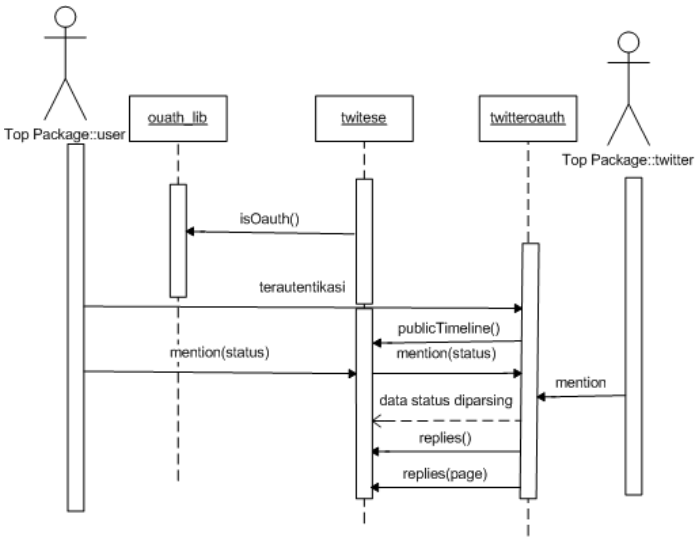
Gambar 3.43 Diagram Sequence List Members

3.1.5.24 Mentions

Skenario mentions dapat dilihat pada tabel 3.23 dan diagram sequence mentions pada gambar 3.32 .

3.32 Tabel Skenario Mention

Identifikasi		
Nomor	24	
Nama	Mentions	
Tujuan	Melihat Mentions	
Deskripsi	Twitter mengautentikasi user , user memilih menu show mention, sistem aplikasi meminta data mentions ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user sudah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user mengirimkan mention (status)		4.id mention dikirimkan ke twitter
5.Twitter mengirimkan mention(status) ke sistem aplikasi		
		6.Data status mentah diparsing sistem aplikasi
		7.Status ditampilkan ke pengguna
Kondisi akhir	Daftar mentions ditampilkan	



Gambar 3.44 Diagram Sequence Mentions

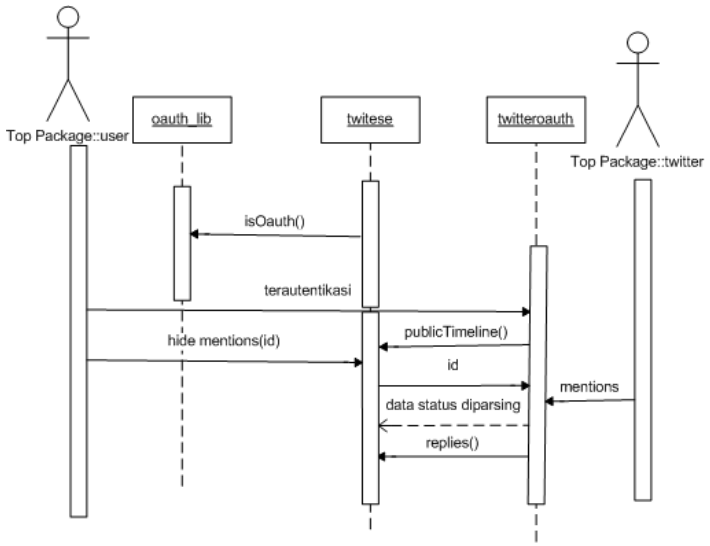
3.1.5.25 Hide Mention

Skenario hide mention dapat dilihat pada tabel 3.33 dan diagram sequence hide mention pada gambar 3.45 .

3.33 Tabel Skenario Hide Mention

Identifikasi		
Nomor	25	
Nama	Hide Mentions	
Tujuan	Menyembunyikan Mentions	
Deskripsi	Twitter mengautentikasi user , user memilih menu hide mention,sistem aplikasi meminta data mentions ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user sudah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user mengirimkan mention (id)		4.id hide mention dikirimkan ke twitter

5.Twitter mengirimkan mention(status) ke sistem aplikasi	
	6.Data status mentah diparsing sistem aplikasi
	7.Status ditampilkan ke pengguna
Kondisi akhir	Mentions tidak ditampilkan



Gambar 3.45 Diagram Sequence Hide Mention

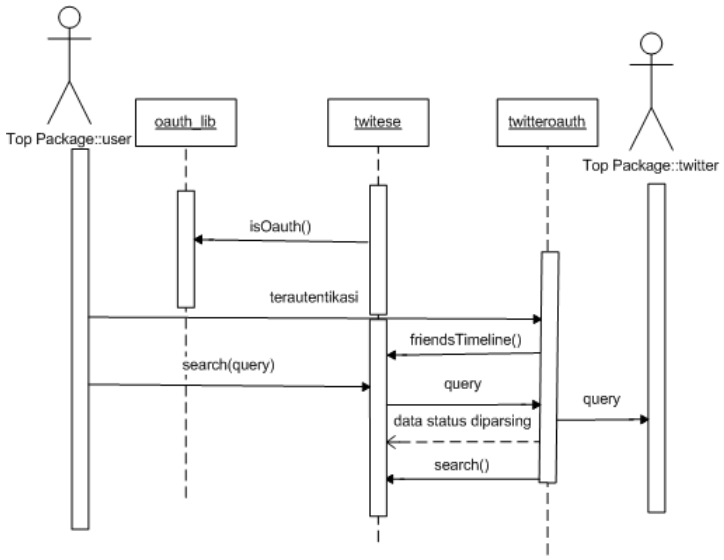
3.1.5.26 Search

Skenario search dapat dilihat pada tabel 3.34 dan diagram sequence search pada gambar 3.46 .

3.34 Tabel Skenario Search

Identifikasi	
Nomor	26
Nama	Search
Tujuan	Melakukan pencarian dengan kata kunci
Deskripsi	Twitter mengautentikasi user, user memilih menu search lalu sistem aplikasi melakukan pencarian ke twitter dan menampilkan ke tampilan
Tipe	
Aktor	user, Twitter

Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user mengirimkan query (kata kunci pencarian)		4.query dikirimkan ke twitter
5.Twitter menerima query		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.hasil pencarian ditampilkan ke pengguna
Kondisi akhir	Hasil percarian ditampilkan	



Gambar 3.46 Diagram Sequence Search

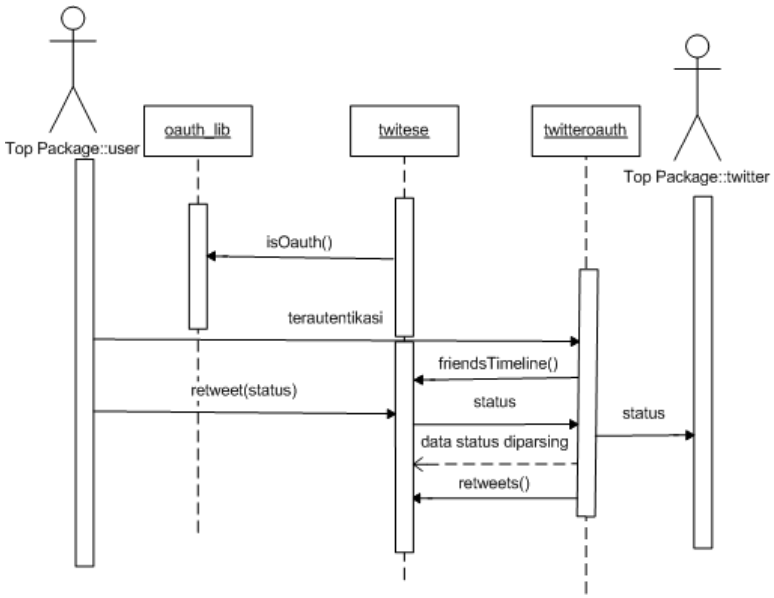
3.1.5.27 Retweet

Skenario retweet dapat dilihat pada tabel 3.35 dan diagram sequence retweet pada gambar 3.47 .

3.35 Tabel Skenario Retweet

Identifikasi	
Nomor	27
Nama	Retweet

Tujuan	Melakukan Retweet	
Deskripsi	Twitter mengautentikasi user , lalu user memilih status yang akan diretweet, kemudian sistem aplikasi mengirimkan id status ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user mengirimkan status retweet		4. status retweet dikirimkan ke twitter
5.Twitter menerima status retweet		6.Sistem meminta status retweet dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan dengan original retweet yang dipilh oleh pengguna.	



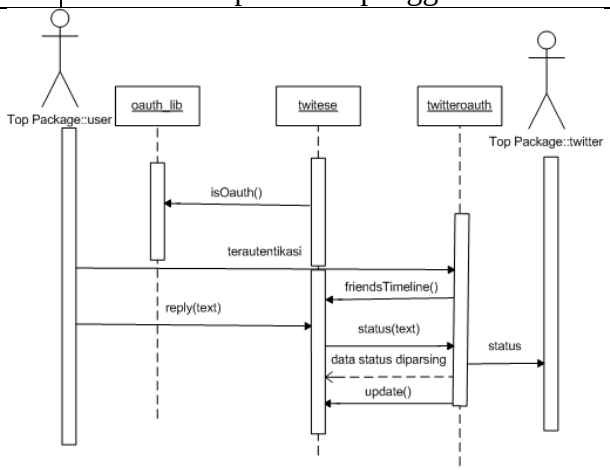
Gambar 3.47 Diagram Sequence Retweet

3.1.5.28 Reply

Skenario reply dapat dilihat pada tabel 3.36 dan diagram sequence reply pada gambar 3.48 .

3.36 Tabel Skenario Reply

Identifikasi		
Nomor	28	
Nama	Reply	
Tujuan	Melakukan Reply pada status	
Deskripsi	Twittre mengautentikasi user, lalu sistem aplikasi melakukan reply status ke twitter menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terauntentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3. Pengguna memilih menu reply pada status dan menginputkan status (text)		4.Status dikirimkan ke twitter
5.Twitter menerima status		6.Sistem meminta status dari twitter
		7.Data status mentah diparsing
		8.Status ditampilkan ke pengguna
Kondisi akhir	Status ditampilkan ke pengguna	



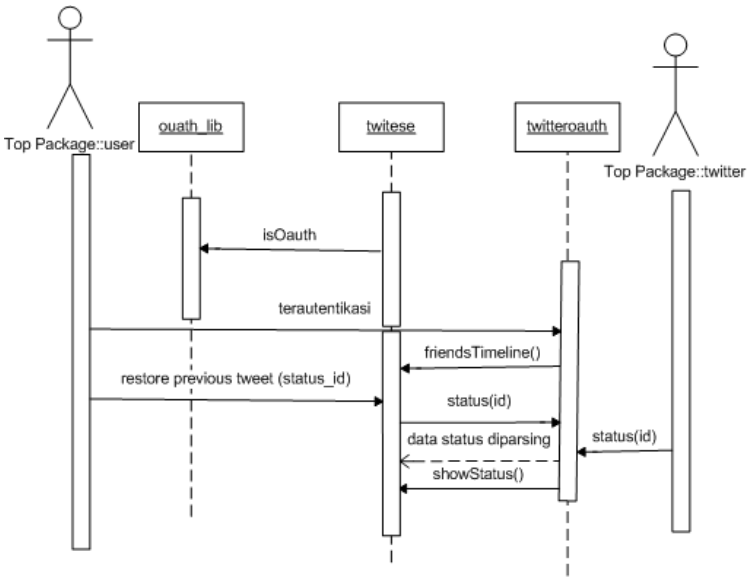
Gambar 3.48 Diagram Sequence Reply

3.1.5.29 Restore Previous Tweet

Skenario restore previous tweet dapat dilihat pada tabel 3.37 dan diagram sequence restore previous tweet pada gambar 3.49 .

3.37 Tabel Skenario Restore Previous tweet

Identifikasi		
Nomor	29	
Nama	Restore Previous	
Tujuan	Melihat Status terakhir	
Deskripsi	Twitter mengautentikasi user , user memilih menu restore previous tweet,lalu sistem aplikasi meminta data status terakhir ke twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1.twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
2.user memilih menu restore previous tweet		3.id status terakhir dikirimkan ke twitter
4.Twitter mengirim id status yag terkahir kali diposting		5.Sistem menerima status dari twitter
		6.Data status mentah diparsing
		7.Status ditampilkan ke pengguna
Kondisi akhir	Tweet yang terkahir kali diposting ditampilkan	



Gambar 3.49 Diagram Sequence Restore Previous

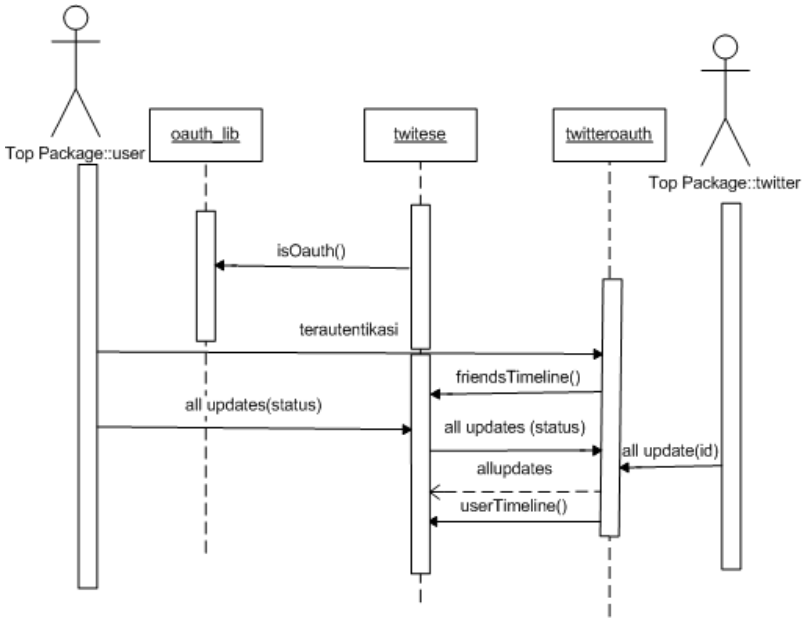
3.1.5.30 All Updates

Skenario all updates dapat dilihat pada tabel 3.38 dan diagram sequence all updates pada gambar 3.50 .

3.38 Tabel Skenario All Updates

Identifikasi		
Nomor	30	
Nama	All Updates	
Tujuan	Melihat semua updates (update,mention,message)	
Deskripsi	Twitter mengautentikasi user, user memilih menu all upates lalu sistem aplikasi meminta data updates dari twitter dan menampilkan ke tampilan	
Tipe		
Aktor	user, Twitter	
Skenario Utama		
Kondisi awal	user telah terautentikasi	
Aksi Aktor		**Reaksi Sistem**
1. twitter mengautentikasi user		2. Sistem Aplikasi meminta *access token*
3.user memilih menu all upadates		4.all update dikirimkan ke twitter

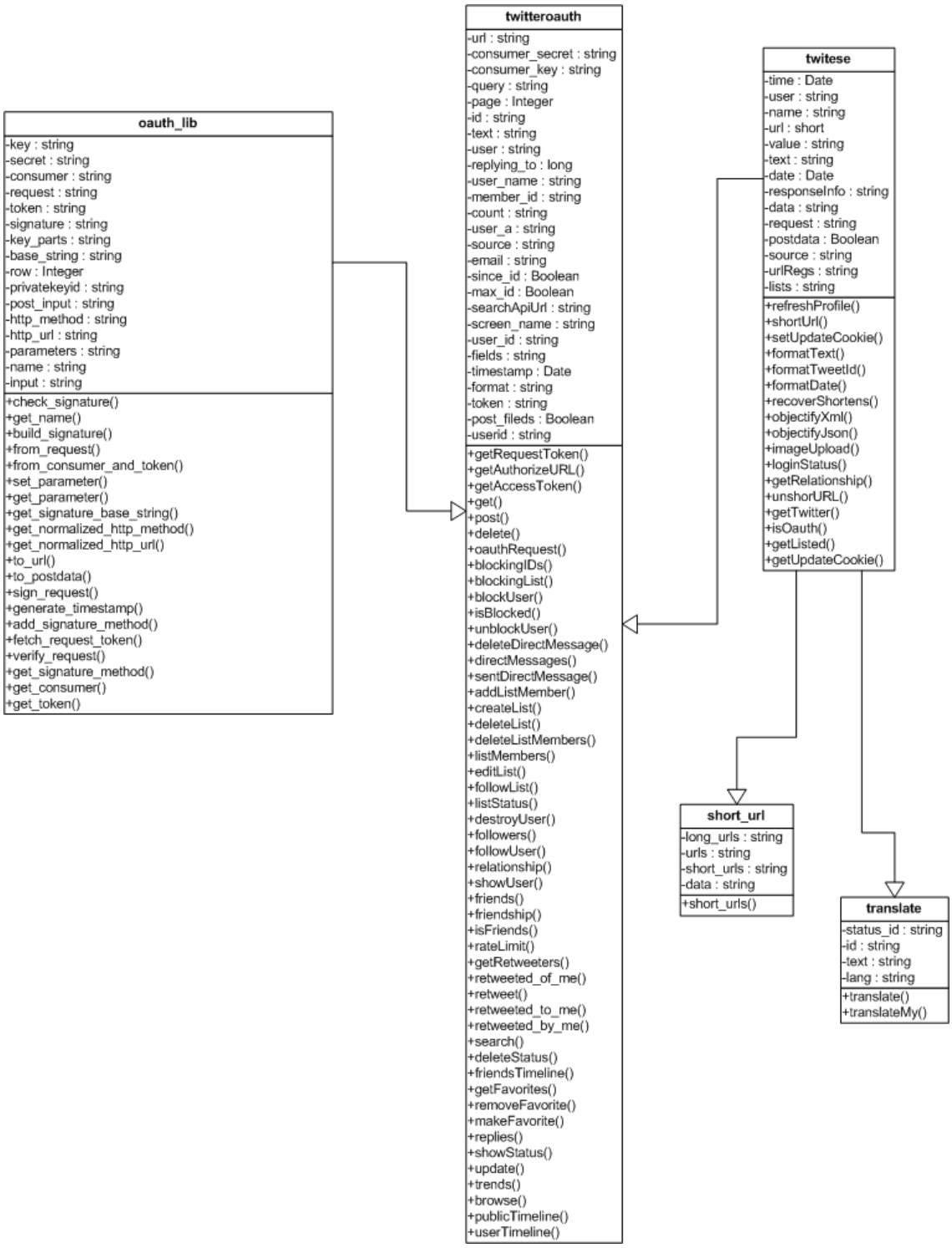
5.Twitter mengirim data all updates	6.Sistem menerima data updates dari twitter
	7.Data status mentah diparsing
	8.Status ditampilkan ke pengguna
Kondisi akhir	Status all updates ditampilkan



Gambar 3.50 Diagram Sequence All Updates

3.2 Class Diagram

Kelas diagram dari aplikasi *twitter client* yang dibangun dapat dilihat pada gambar 3.51 .



Gambar 3.51 Class Diagram

3.2.1 Spesifikasi Kelas

Berikut adalah tabel 3.39 spesifikasi kelas yang digunakan.

Tabel 3.39 Spesifikasi Kelas

No.	Nama Kelas	Jenis
1	Twitteroauth	Kontrol
2	Twitese	Kontrol
3	Oauth_lib	Kontrol
4	Translate	Kontrol
5	Short_url	Kontrol

3.2.1.1 Kelas *Twitteroauth*

Kelas *twitteroauth* adalah kelas utama yang digunakan untuk melakukan fungsi-fungsi pada *twitter* dan berkomunikasi dengan *twitter*.

3.2.1.1.1 Deskripsi Atribut

Deskripsi atribut kelas twitteroauth dapat dilihat pada tabel 3.40

Tabel 3.40 Deskripsi Atribut Kelas *Twitteroauth*

Nama	Tipe	Panjang	Range	Nilai Tetap
shortURL	String	20	'a'..'Z','0'..'9',-,/,spasi	-
user_a	String	30	'a'..'Z','0'..'9',-,/,spasi	-
user_b	string	30	'a'..'Z','0'..'9',-,/,spasi	-
query	String	-	'a'..'Z','0'..'9',-,/,spasi	-
text	String	140	'a'..'Z','0'..'9',-,/,spasi	140
page	integer	-	0-9	1
post_fields	Boolean	-	-	true
consumer_key	String	21	'a'..'Z','0'..'9',-,/,spasi	uOSHDDfQjip4lBJhArQrw

consumer_secret	String	43	'a'..'Z','0'..'9',-,/,spasi	pcWVcmKj9ljGxi6nSQFzspzgBZuGHV6rp30xC8ACmrA
id	String	-	'a'..'Z','0'..'9',-,/,spasi	-
timestamp	Date	-	-	-
searchApiUrl	string	50	'a'..'Z','0'..'9',-,/,spasi	-
max_id	string	-	'a'..'Z','0'..'9',-,/,spasi	-
since_id	string	-	'a'..'Z','0'..'9',-,/,spasi	-
replying_to	Long	-	-	-
status	String	140	'a'..'Z','0'..'9',-,/,spasi	-
user	String	30	'a'..'Z','0'..'9',-,/,spasi	-
token	String	-	'a'..'Z','0'..'9',-,/,spasi	-
user_id	String	-	'a'..'Z','0'..'9',-,/,spasi	-
format	String	-	'a'..'Z','0'..'9',-,/,spasi	-
username	String	30	'a'..'Z','0'..'9',-,/,spasi	-
fields	string	-	'a'..'Z','0'..'9',-,/,spasi	-
email	String	30	'a'..'Z','0'..'9',-,/,spasi	-

3.2.1.1.2 Deskripsi Layanan

1) Method getRequestToken() digunakan untuk meminta akeses *token* dari *twitter.*
2) Method deleteStatus() digunakan untuk menghapus *tweet.*
3) Method blockingIDs() digunakan untuk memblock *user.*
4) Method blockingList() digunakan untuk memblok *list.*
5) Method unblockUser() digunakan untuk mengunblock *user* yang sebelumnya telah diblock.
6) Method deleteDirectMessage() digunakan untuk menghapus pesan

yang masuk.

7) Method directMessages() digunkaan untuk melihat pesan yang masuk.
8) Method sendMessage() digunakan untuk mengirimkan pesan.
9) Method createList() digunakan untuk membuat *list* baru.
10) Method addListMembers() digunakan untuk menambah *user* pada sebuah *list.*
11) Method deleteListMembers() digunakan untuk menghapus *user* pada sebuah *list.*
12) Method deleteList() digunakan untuk menghapus *list.*
13) Method listMembers() digunakan untuk memperlihatkan anggota list tersebut.
14) Method followers() digunakan untuk melihat *user* yang terkait dengan *account* yang dimiliki.
15) Method followUser() digunakan untuk mengikuti salah satu *user* agar saling terkait.
16) Method getRetweeters() digunakan untuk melihat *retweet.*
17) Method retweet() digunakan untuk melakukan *retweet*.
18) Method retweeted_of_me() digunakan untuk melihat *retweet* dan *tweet* yang dilakukan.
19) Method retweeted_by_me() digunakan untuk melihat *retweet* yang dilakukan terhadap sebuah *tweet.*
20) Method search() digunakan untuk melakukan pencarian.

21) Method getFavorites() digunakan untuk melihat halaman *favorites.*

22) Method removeFavorite() digunakan untuk menghapus *tweet* yang sebelumnya difavoritkan.

23) Method replies() digunakan untuk melihat halaman *all updates*

24) Method friendsTimeline() digunakan untuk melihat halaman *timeline.*

25) Method showStatus() digunakan untuk melihat *tweet update*.

26) Method deleteStatus() digunakan untuk menghapus *tweet.*

27) Method trends() digunakan untuk melihat *trending topic*.

28) Method userTimeline(0 digunakan untuk melihat halaman *timeline* sendiri yang berisi *tweet* yang dilakukan.

29) Method PublicTimeline() digunakan untuk melihat *updates* pengguna yang saling terkait di *timeline.*

3.2.1.2 Kelas *Twitese*

Kelas *twitese* merupakan kelas yang berhubungan dengan kelas *twitteroauth* untuk berkomunikasi dengan *twitter.*

3.2.1.2.1 Deskripsi Atribut

Deskripsi atribut kelas twitese dapat dilihat pada tabel 3.41 .

Tabel 3.41 Deskripsi Atribut Kelas *Twitese*

Nama	Tipe	Panjang	Range	Nilai Tetap
url	String	100	’a’..’Z’,’0’..’9’,-,/,spasi	-

User	String	30	’a’..’Z’,’0’..’9’,-,/,spasi	-
name	String	30	’a’..’Z’,’0’..’9’,-,/,spasi	-
time	date	-	-	y-m-d
value	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
text	String	140	’a’..’Z’,’0’..’9’,-,/,spasi	-
response_info	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
id	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
data	string	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
request	string	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
post_data	Boolean	-	-	false
source	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-

3.2.1.2.2 Deskripsi Layanan

1) Method isOauth() digunakan untuk melakukan *request oauth*
2) Method verify() digunakan untuk melakukan identifikasi terhadap pengguna yang *login.*
3) Method loginStatus() digunakan untuk *login.*
4) Method setUpdateCookie() digunakan untuk membuat cookie pada saat melakukan update tweet.
5) Method formatText() digunakan untuk mengirimkan status dalam format *text.*
6) Method refreshProfile() digunakan untuk melakukan *refresh* dan menampilkan *update tweet.*
7) Method formatDate() digunakan untuk membuat keterangan waktu.
8) Method recoverShortens() digunakan untuk memperbaiki format *url* yang salah.
9) Method shortUrl() digunakan untuk memperpendek *url (link*).
10) Method unshortURL() digunakan untuk memperlihatkan url asli yang

telah diperpendek.

11) Method imageUpload() digunakan untuk melakukan unggah gambar.
12) Method getRelationship() digunakan untuk melihat status pertemanan dengan pengguna lainnya.
13) Method getTwitter() digunakan untuk meminta data ke *twitter*.
14) Method objectifyXml() digunakan untuk mentransfer data dalam format xml.
15) Method objectifyJason() digunakan untuk mentransfer data dalam format jason.

3.2.1.3 Kelas *Oauth_lib*

Kelas *oauth_lib* digunakan untuk melakukan proses autentikasi *oauth* yang berhubungan dengan *twitter* sehingga pengguna tetap dapat melakukan *login* pada *twitter* dan aplikasi *client* tidak menyimpan data pengguna *(username* dan *password)*

3.2.1.3.1 Deskripsi Atribut

Deskripsi atribut kelas oauth_lib dapat dilihat pada tabel 3.42 .

Tabel 3.42 Deskripsi Atribut Kelas Oauth_lib

Nama	Tipe	Panjang	Range	Nilai Tetap
key	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
secret	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
signature	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
key_parts	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
base_string	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
privatekeyId	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
Post_input	string	-	’a’..’Z’,’0’..’9’,-,/,spasi	-

Request	string	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
http_method	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
http_url	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
parameters	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
name	String	30	’a’..’Z’,’0’..’9’,-,/,spasi	-
input	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
token	String	-	-	-

3.2.1.3.2 Deskripsi Layanan

1) Method Check_signature() digunakan untuk mengecek *consumer key client* .

2) Method get_name() digunakan untuk mengirimkan nama pengguna ke *twitter.*

3) Method from_request_token() digunakan untuk meminta akses *token*

4) Method from_consumer_and_token() digunakan untuk memeriksa *secret key consumer.*

5) Method set_parameter() digunakan untuk membuat parameter yang akan dikirim ke twitter.

6) Method get_parameter() digunakan untuk meminta parameter dari *twitter.*

7) Method get_signature_base_string() digunakan untuk mengubah data ke dalam string tunggal untuk dikirim ke *twitter.*

8) Method fromRequest() digunakan untuk merespon *request oauth.*

9) Method to_postdata() digunakan untuk mengirimkan data ke *twitter.*

10) Method sign_request() digunakan untuk mengirimkan request login.

11) Method get_consumer() digunakan untuk mengirimkan *consumer key*

ke *twitter*.

12) Method loginStatus() digunakan untuk login.

13) Method get_normalized_http_method() digunakan untuk melakukan normalisasi data agar dapat dibaca oleh *twitter*.

14) Method get_normalized_http_url() digunakan untuk melakukan normalisasi url agar dapat dibaca oleh *twitter*.

15) Method url() digunakan untuk mengirimkan url ke *twitter*.

16) Method add_signature_method() digunakan untuk melakukan *signature request.*

17) Method fetch_request_token() digunakan untuk merespon *access token* yang diterima.

18) Method verify_request() digunakan untuk mengecek *request token* yang diterima apakah benar atau tidak.

19) Method generate_timestamp() digunakan untuk melakukan *generate* waktu agar menjadi bagian yang unik.

20) Method to_postdata() digunakan untuk mengirimkan data ke *twitter*.

21) Method get_consumer() digunakan untuk mengirimkan *request token*

22) Method get_token() digunakan untuk mengirimkan akses *token.*

3.2.1.4 Kelas *Translate*

Kelas *translate* merupakan kelas yang digunakan untuk melakukan terjemahan.

3.2.1.4.1 Deskripsi Atribut

Deskripsi atribut kelas translate dapat dilihat pada tabel 3.43 .

Tabel 3.43 Deskripsi Atribut Kelas *Translate*

Nama	Tipe	Panjang	Range	Nilai Tetap
status	String	140	’a’..’Z’,’0’..’9’,-,/,spasi	140
id	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
id_status	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
lang	String	30	’a’..’Z’,’0’..’9’,-,/,spasi	-

3.2.1.4.2 Deskripsi Layanan

1) Method translate() digunakan untuk menterjemahkan *tweet* yang ada di *timeline.*
2) Method translateMy() digunakan untuk menterjemahkan *tweet* yang akan *diupdate.*

3.2.1.5 Kelas *Short_url*

Kelas *short_url* merupakan kelas yang digunakan untuk memperpendek url .

3.2.1.5.1 Deskripsi Atribut

Deskripsi atribut kelas shorten_url dapat dilihat pada tabel 3.44 .

Tabel 3.44 Deskripsi Atribut Kelas Short_url

Nama	Tipe	Panjang	Range	Nilai Tetap
long_urls	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
urls	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-
short_urls	String	20	’a’..’Z’,’0’..’9’,-,/,spasi	-
data	String	-	’a’..’Z’,’0’..’9’,-,/,spasi	-

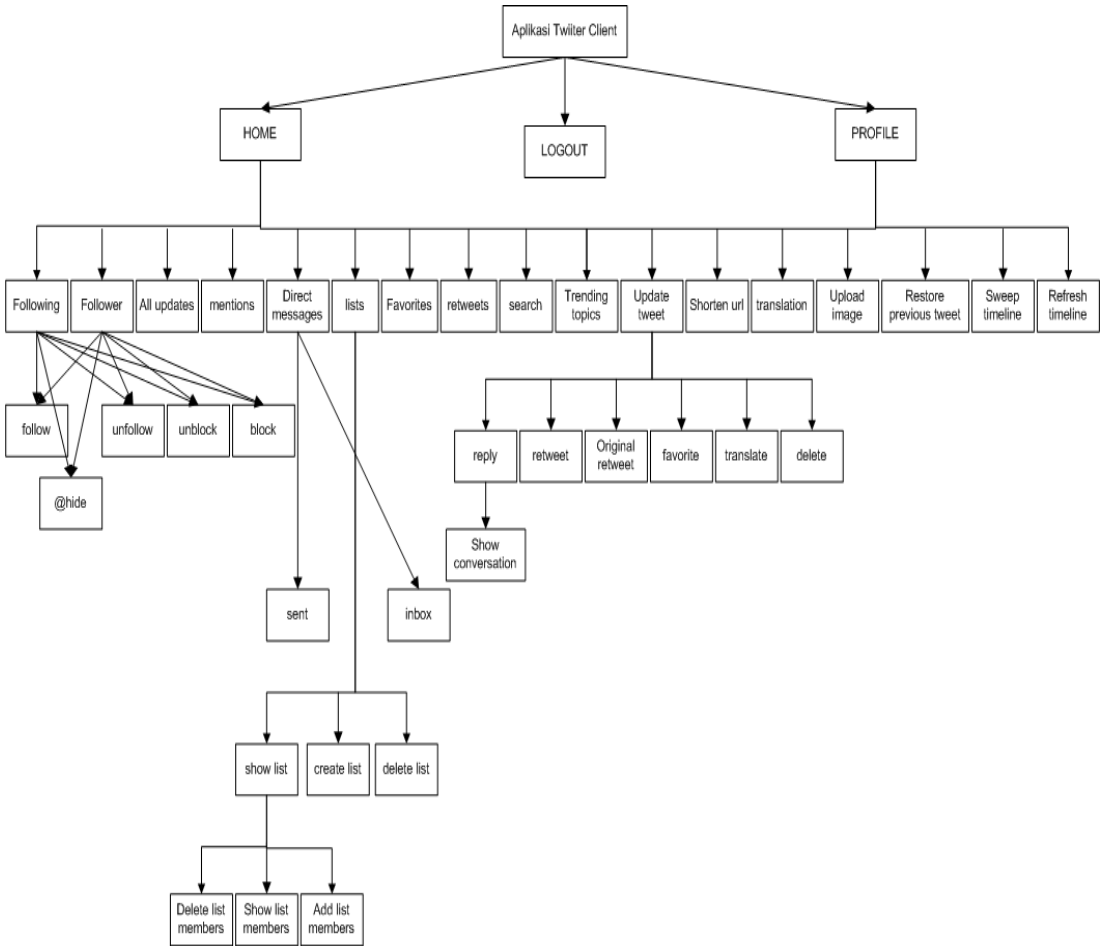
3.2.1.5.2 Deskripsi Layanan

Method Short_ulrs() digunakan untuk memperpendek url.

3.3 Perancangan Arsitektural

3.3.1 Perancangan Menu

Perancangan menu pada aplikasi *twitter client* seperti yang digambarkan pada gambar 3.52 .

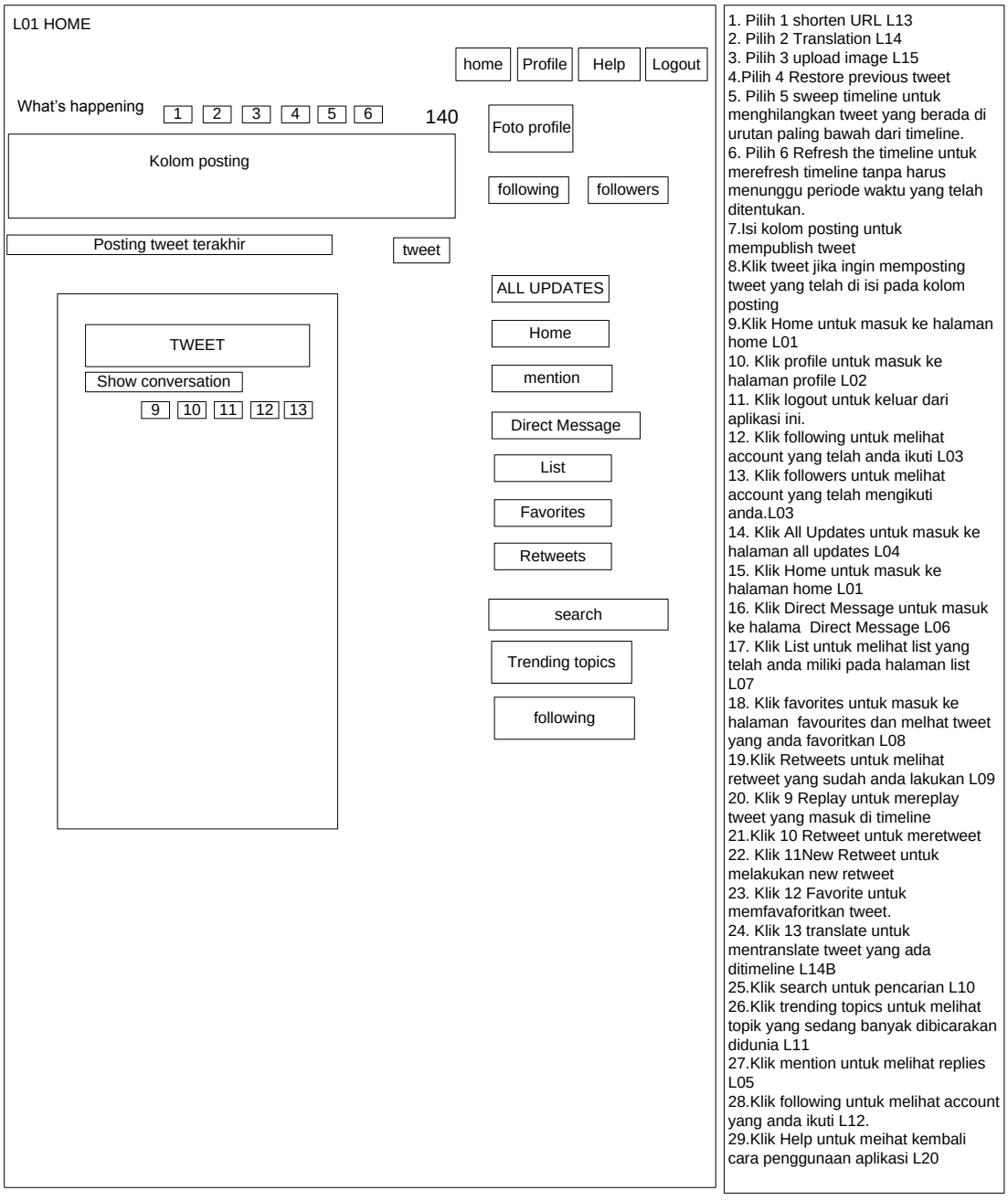


Gambar 3.52 Perancangan Menu

3.3.2 Perancangan Antarmuka

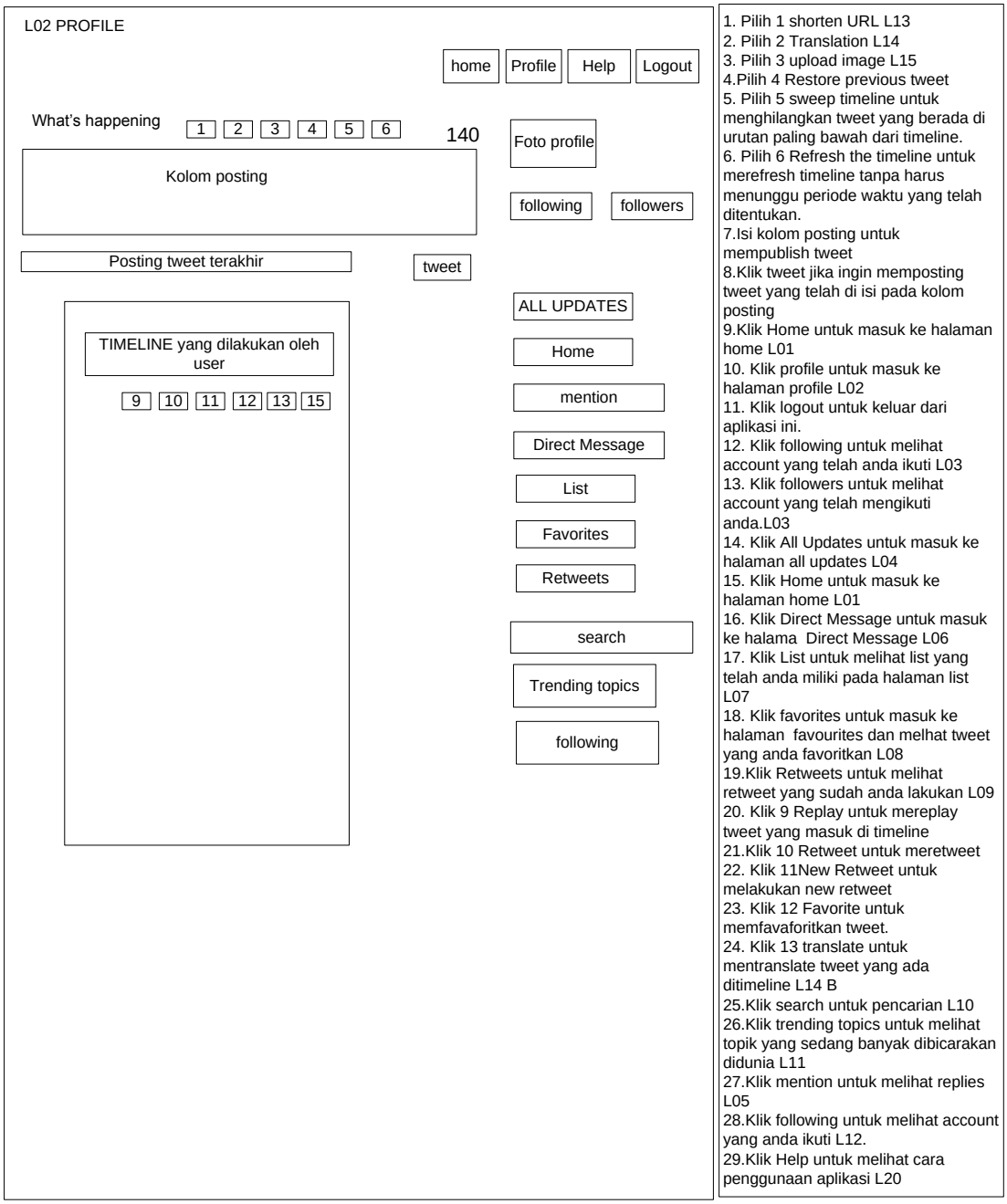
Antarmuka pemakai merupakan media komunikasi antara pemakai dengan sistem komputer. Hasil perancangan antarmuka pengguna akan memudahkan proses implementasi. Adapun antarmuka pemakai perangkat lunak yang akan dirancang ditampilkan sebagai berikut :

Pada gambar 3.53 dapat dilihat perancangan antarmuka halaman utama L01.



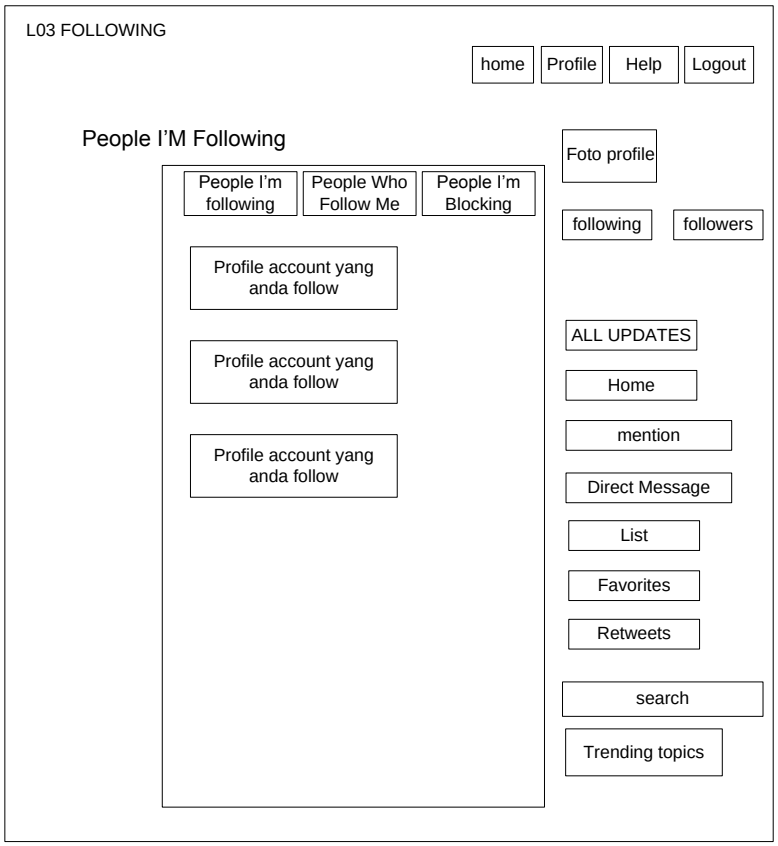
Gambar 3.53 Perancangan Antarmuka Halaman Utama (*home)*

Pada gambar 3.54 dapat dilihat perancangan antarmuka prfile L02.



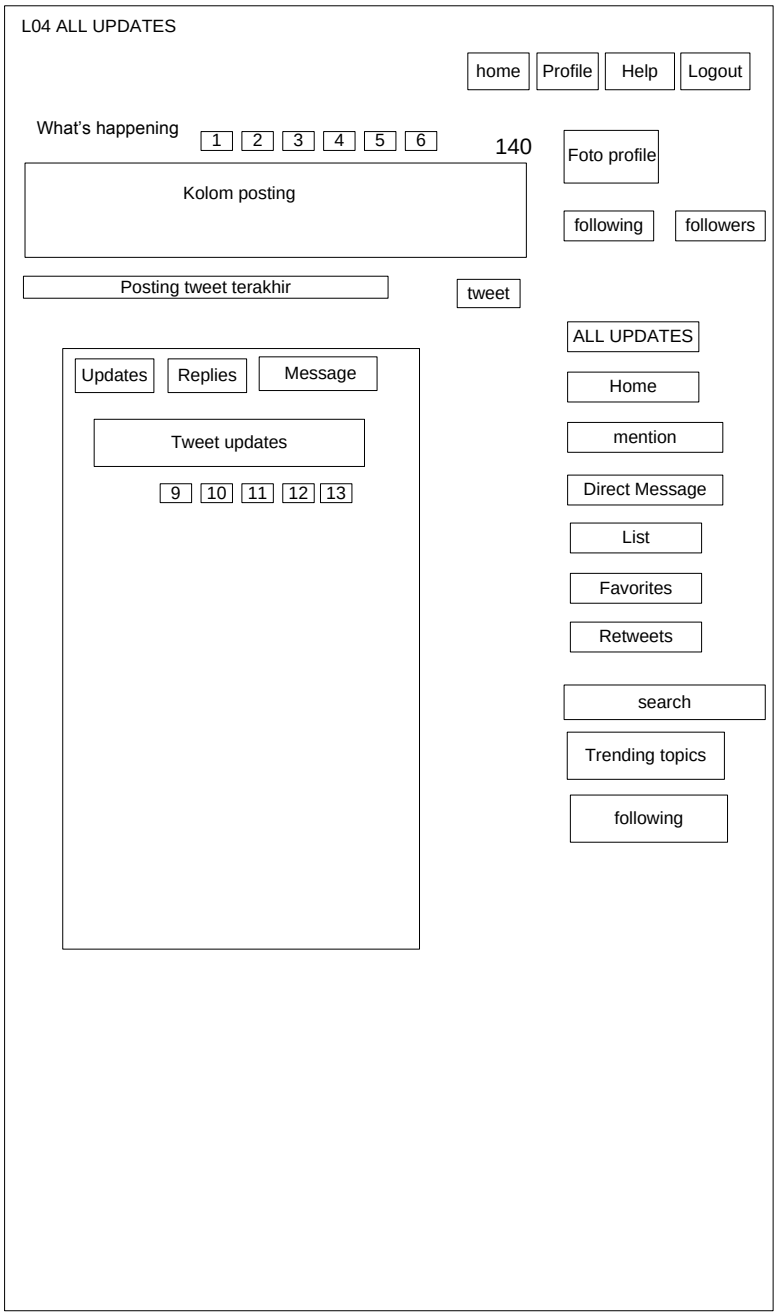
Gambar 3.54 Perancangan Antarmuka *Profile*

Pada gambar 3.55 dapat dilihat perancangan antarmuka following L03.



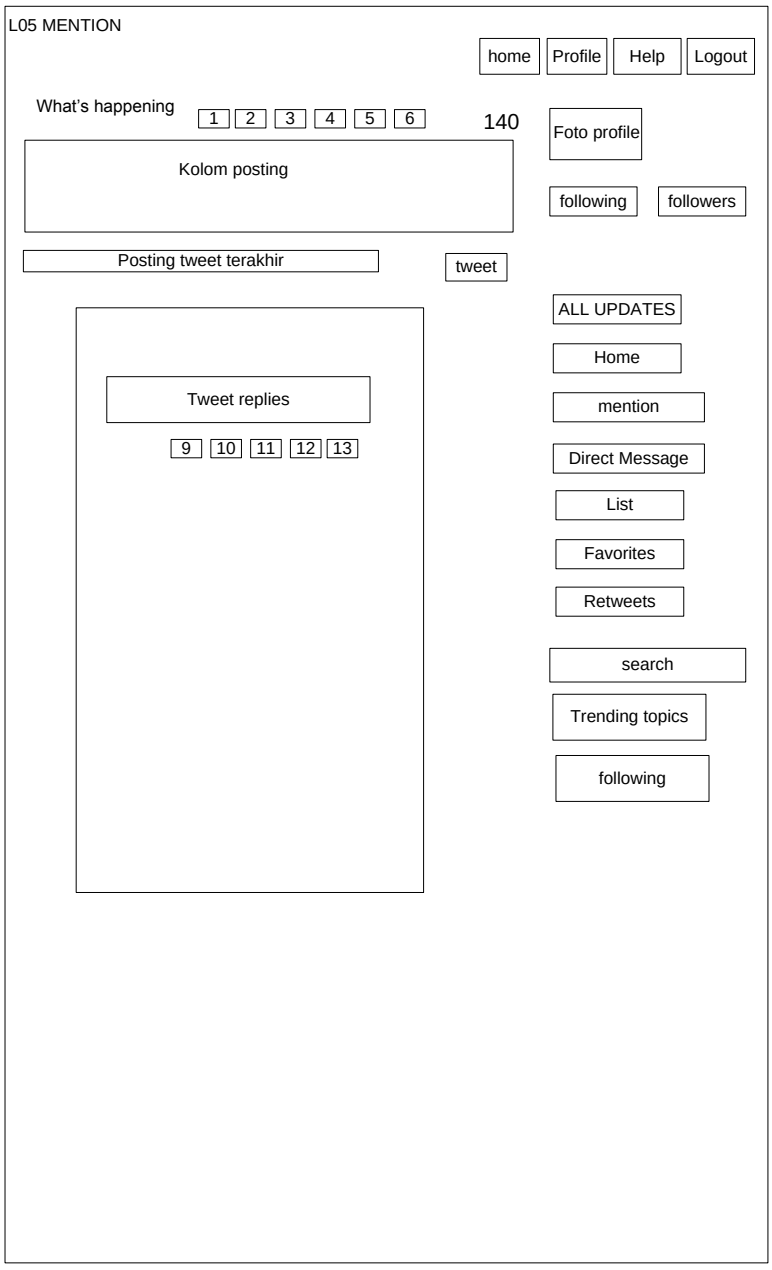
Gambar 3.55 Perancangan Antarmuka *Following*

Pada gambar 3.56 dapat dilihat perancangan antarmuka all updates L04.



Gambar 3.56 Perancangan Antarmuka *All Updates*

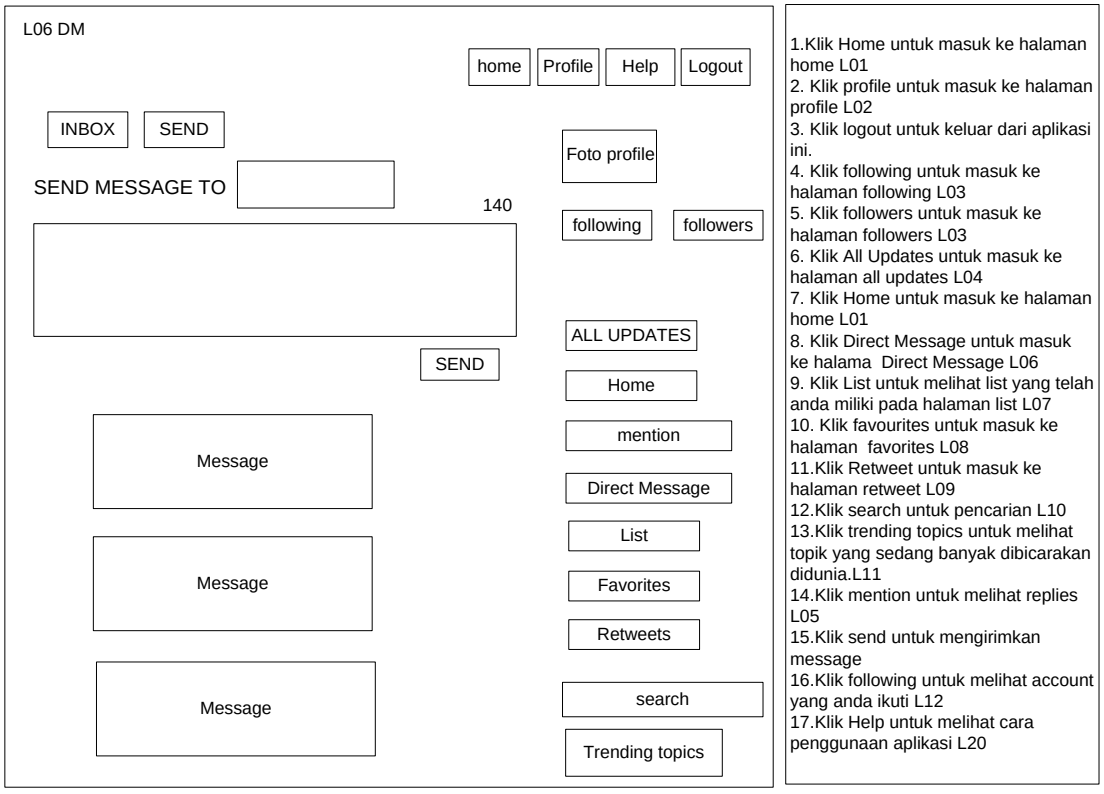
Pada gambar 3.57 dapat dilihat perancangan antarmuka mention L05.



1. Pilih 1 shorten URL L13
2. Pilih 2 Translation L14
3. Pilih 3 upload image L15
4.Pilih 4 Restore previous tweet
5. Pilih 5 sweep timeline untuk menghilangkan tweet yang berada di urutan paling bawah dari timeline.
6. Pilih 6 Refresh the timeline untuk merefresh timeline tanpa harus menunggu periode waktu yang telah ditentukan.
7.Isi kolom posting untuk mempublish tweet
8.Klik tweet jika ingin memposting tweet yang telah di isi pada kolom posting
9.Klik Home untuk masuk ke halaman home L01
10. Klik profile untuk masuk ke halaman profile L02
11. Klik logout untuk keluar dari aplikasi ini.
12. Klik following untuk melihat account yang telah anda ikuti L03
13. Klik followers untuk melihat account yang telah mengikuti anda.L03
14. Klik All Updates untuk masuk ke halaman all updates L04
15. Klik Home untuk masuk ke halaman home L01
16. Klik Direct Message untuk masuk ke halama Direct Message L06
17. Klik List untuk melihat list yang telah anda miliki pada halaman list L07
18. Klik favorites untuk masuk ke halaman favourites dan melhat tweet yang anda favoritkan L08
19.Klik Retweets untuk melihat retweet yang sudah anda lakukan L09
20. Klik 9 Replay untuk mereplay tweet yang masuk di timeline
21.Klik 10 Retweet untuk meretweet
22. Klik 11New Retweet untuk melakukan new retweet
23. Klik 12 Favorite untuk memfavaforitkan tweet.
24. Klik 13 translate untuk mentranslate tweet yang ada ditimeline L14 B
25.Klik search untuk pencarian L10
26.Klik trending topics untuk melihat topik yang sedang banyak dibicarakan didunia L11
27.Klik mention untuk melihat replies L05
28.Klik following untuk melihat account yang anda ikuti L12.
29.Klik Help untuk melihat cara penggunaan aplikasi L20

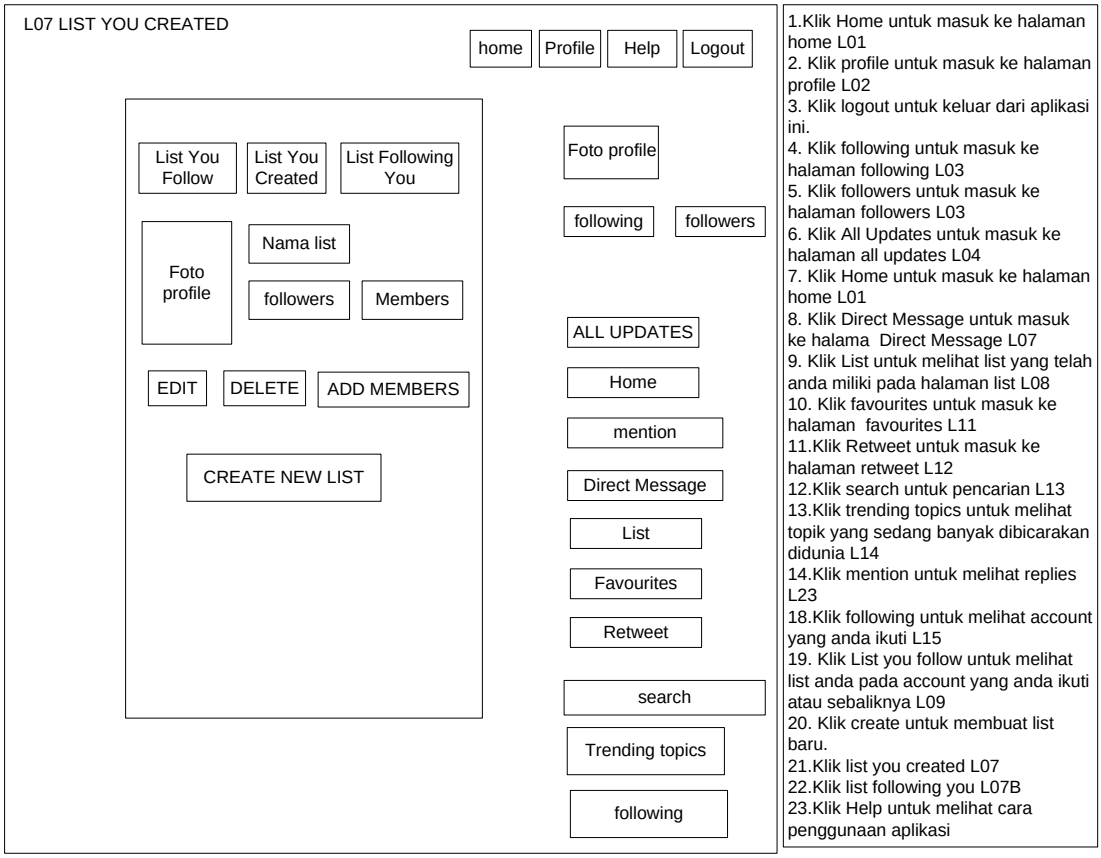
Gambar 3.57 Perancangan Antarmuka *Mention*

Pada gambar 3.58 dapat dilihat perancangan antarmuka DM L06.



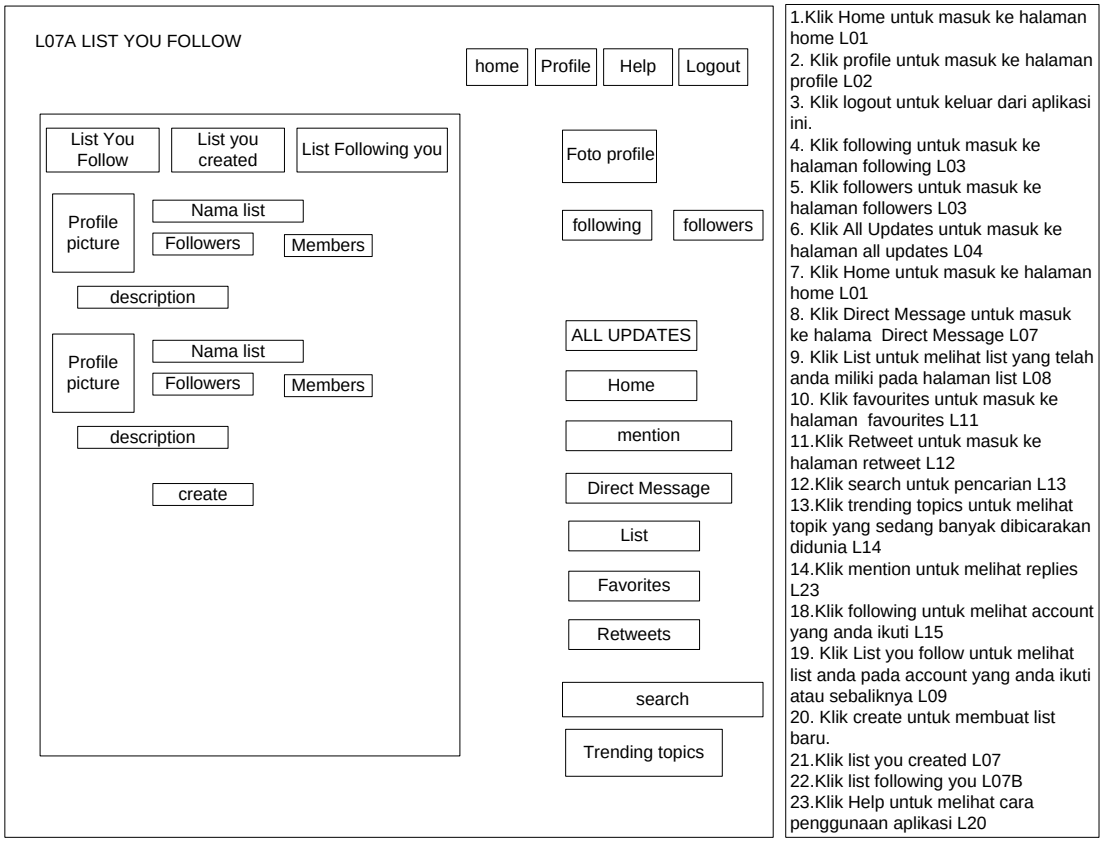
Gambar 3.58 Perancangan Antarmuka DM

Pada gambar 3.59 dapat dilihat perancangan antarmuka list you created L07.



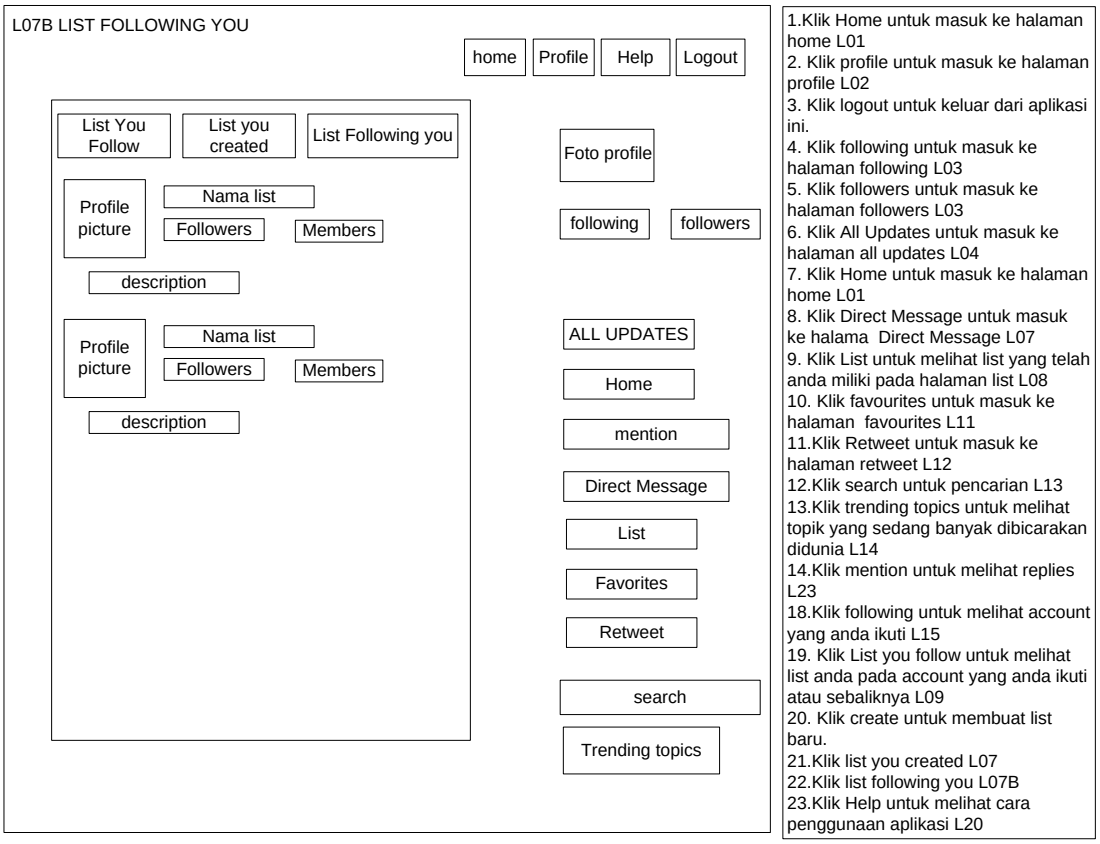
Gambar 3.59 Perancangan Antarmuka *List You Created*

Pada gambar 3.60 dapat dilihat perancangan antarmuka list you follow L07A.



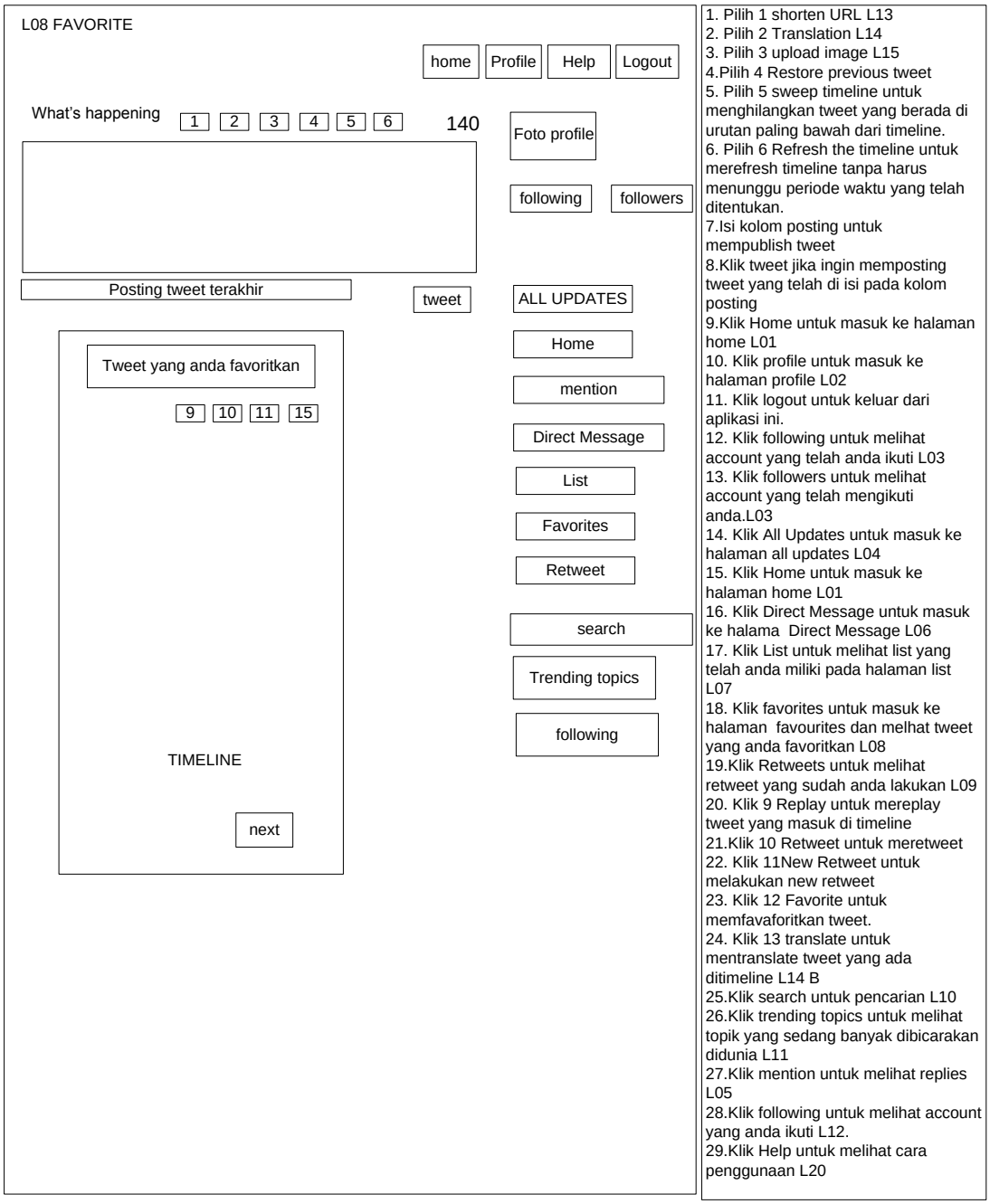
Gambar 3.60 Perancangan Antarmuka *List You Follow*

Pada gambar 3.61 dapat dilihat perancangan antarmuka list you following L07B.



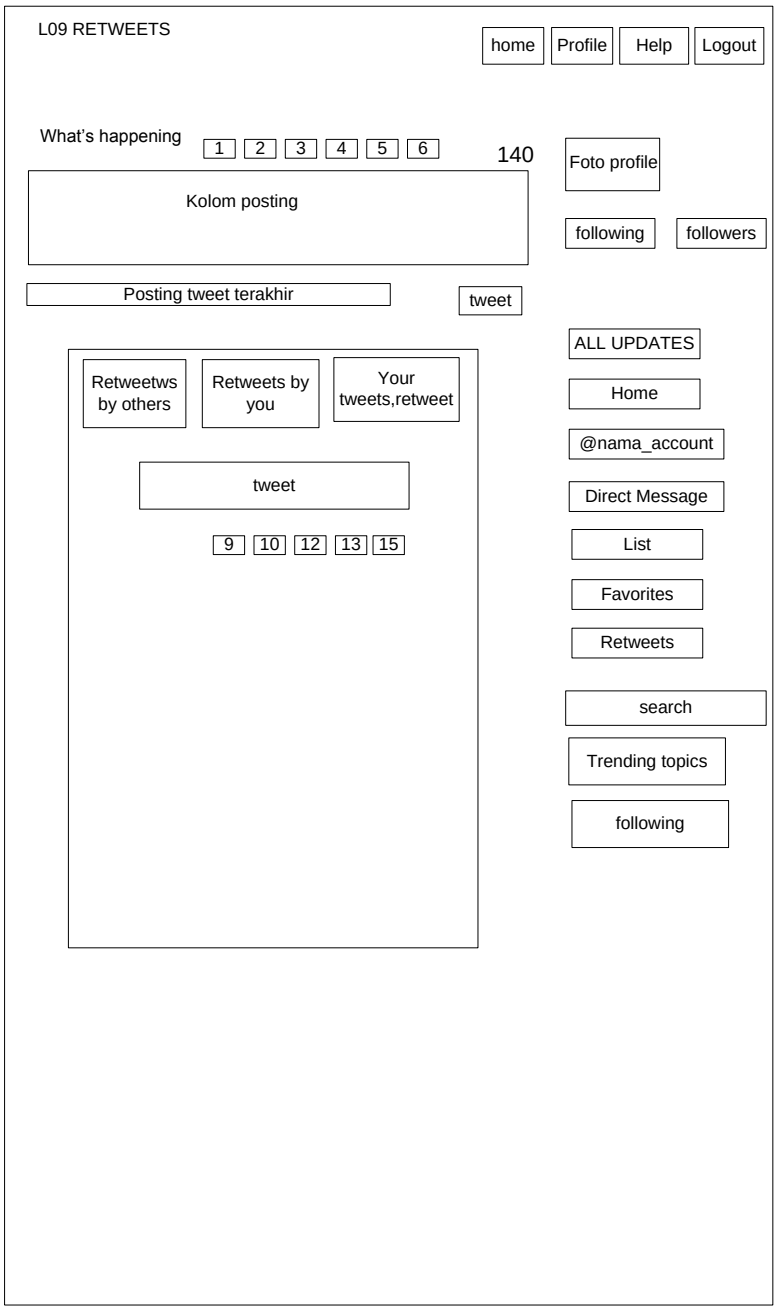
Gambar 3.61 Perancangan Antarmuka *List Following you*

Pada gambar 3.62 dapat dilihat perancangan antarmuka favorite L08.



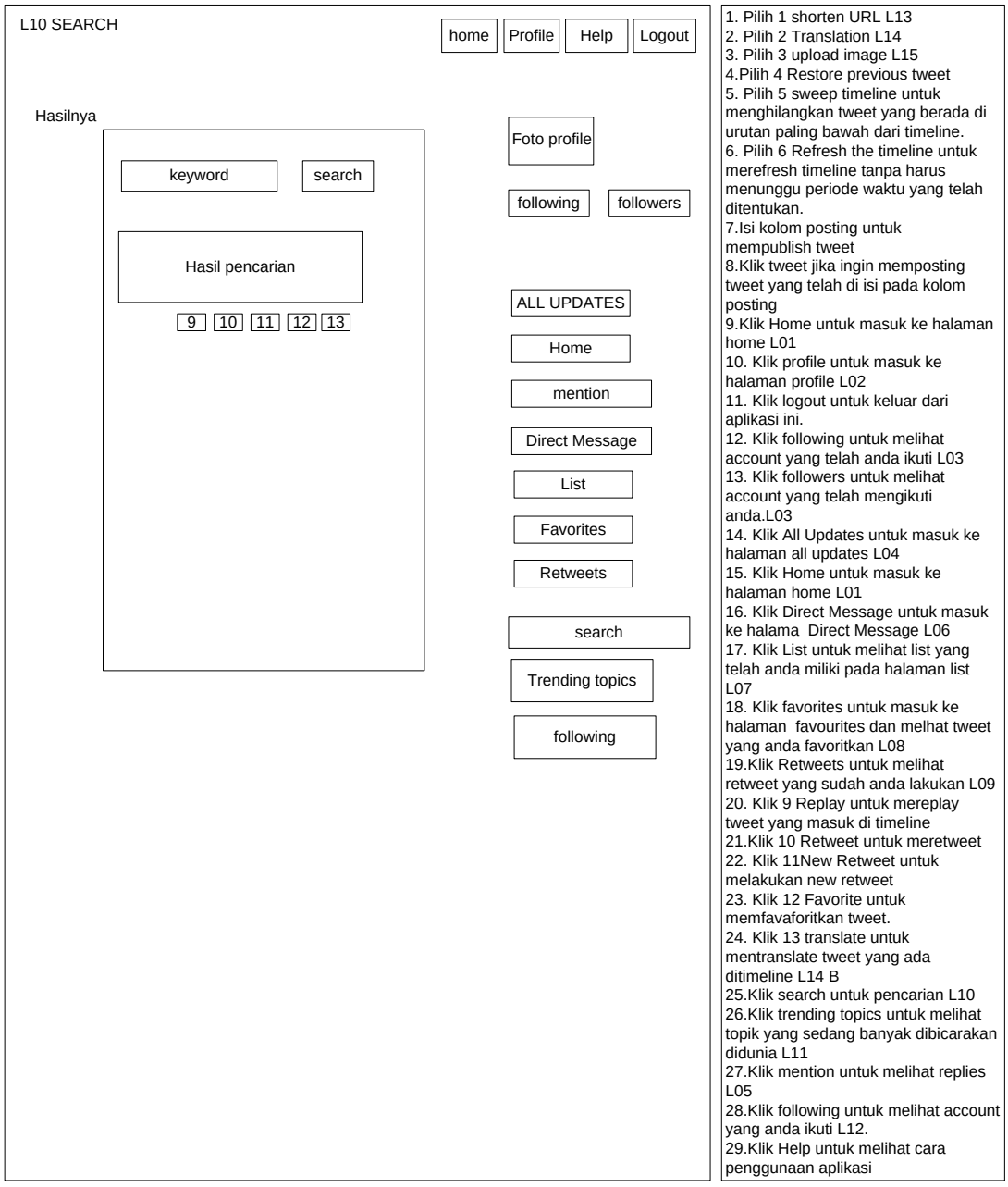
Gambar 3.62 Perancangan Antarmuka *Favorite*

Pada gambar 3.63 dapat dilihat perancangan antarmuka retweets L09.



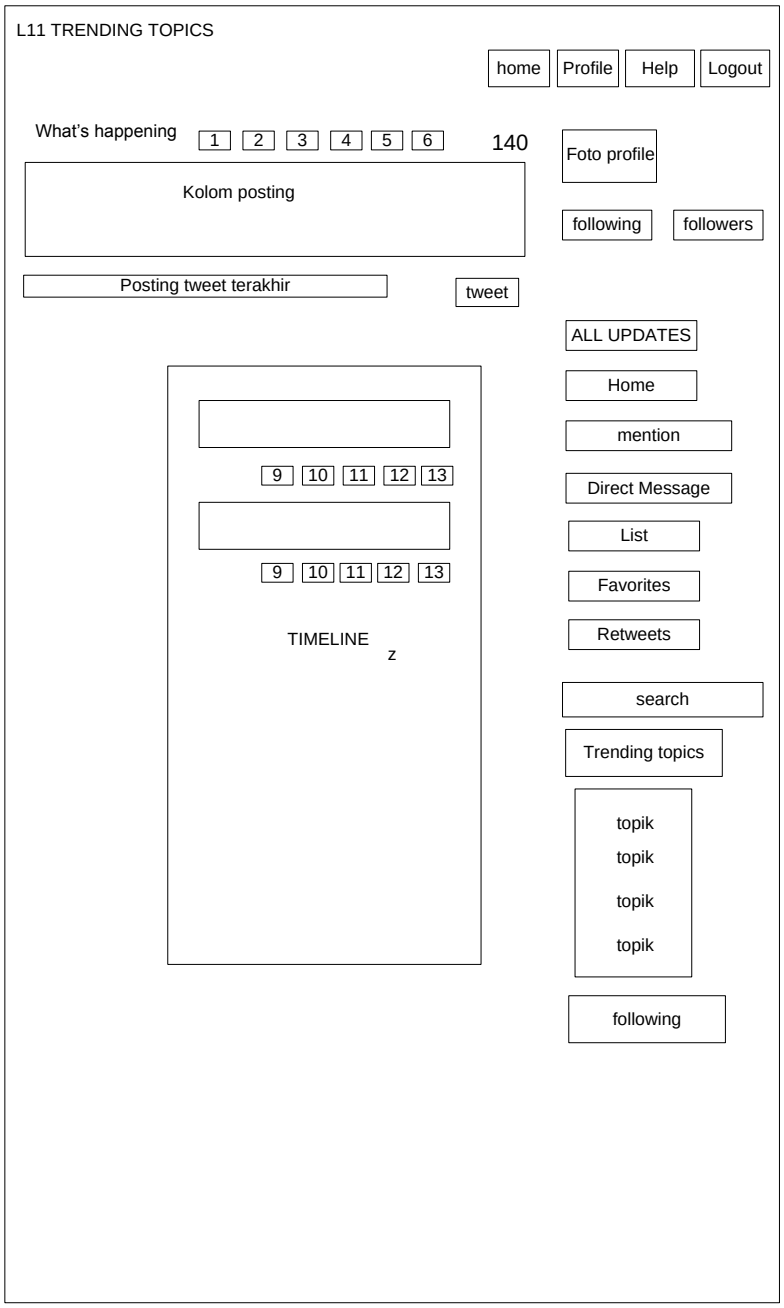
Gambar 3.63 Perancangan Antarmuka *Retweets*

Pada gambar 3.64 dapat dilihat perancangan antarmuka search L10.



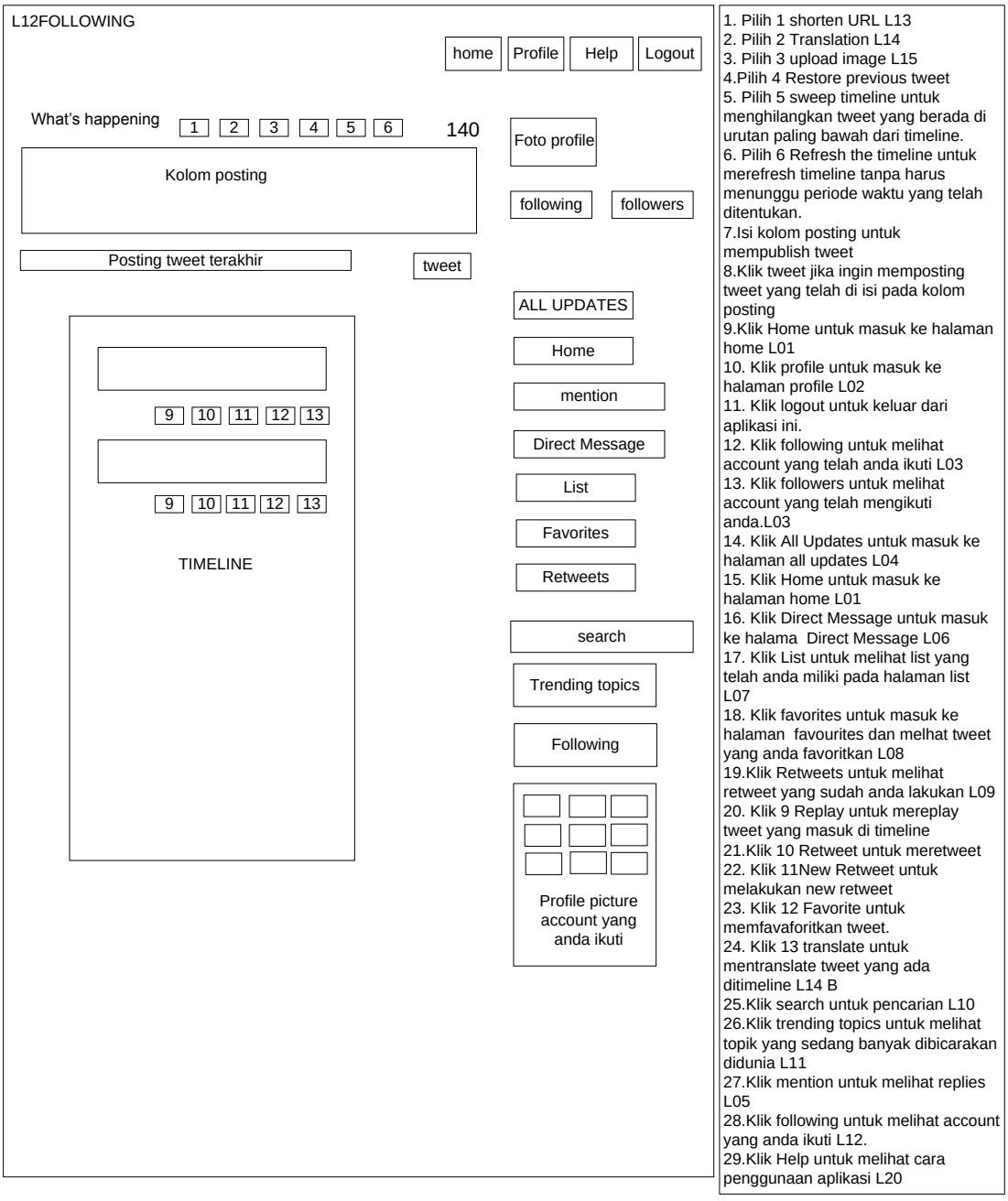
Gambar 3.64 Perancangan Antarmuka *Search*

Pada gambar 4.65 dapat dilihat perancangan antarmuka trending topics L11.



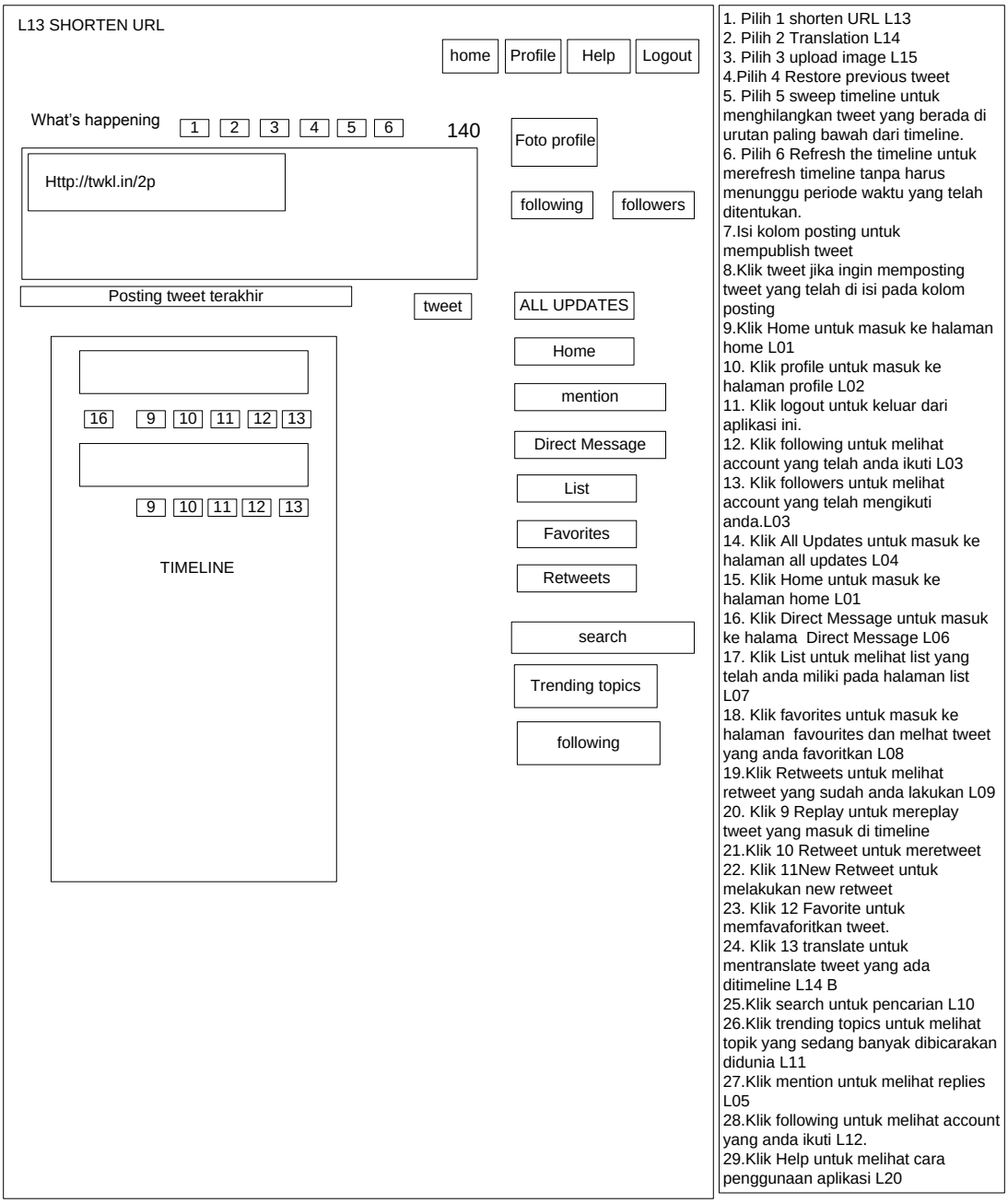
Gambar 3.65 Perancangan Antarmuka *Trending Topics*

Pada gambar 3.66 dapat dilihat perancangan antarmuka following L12.



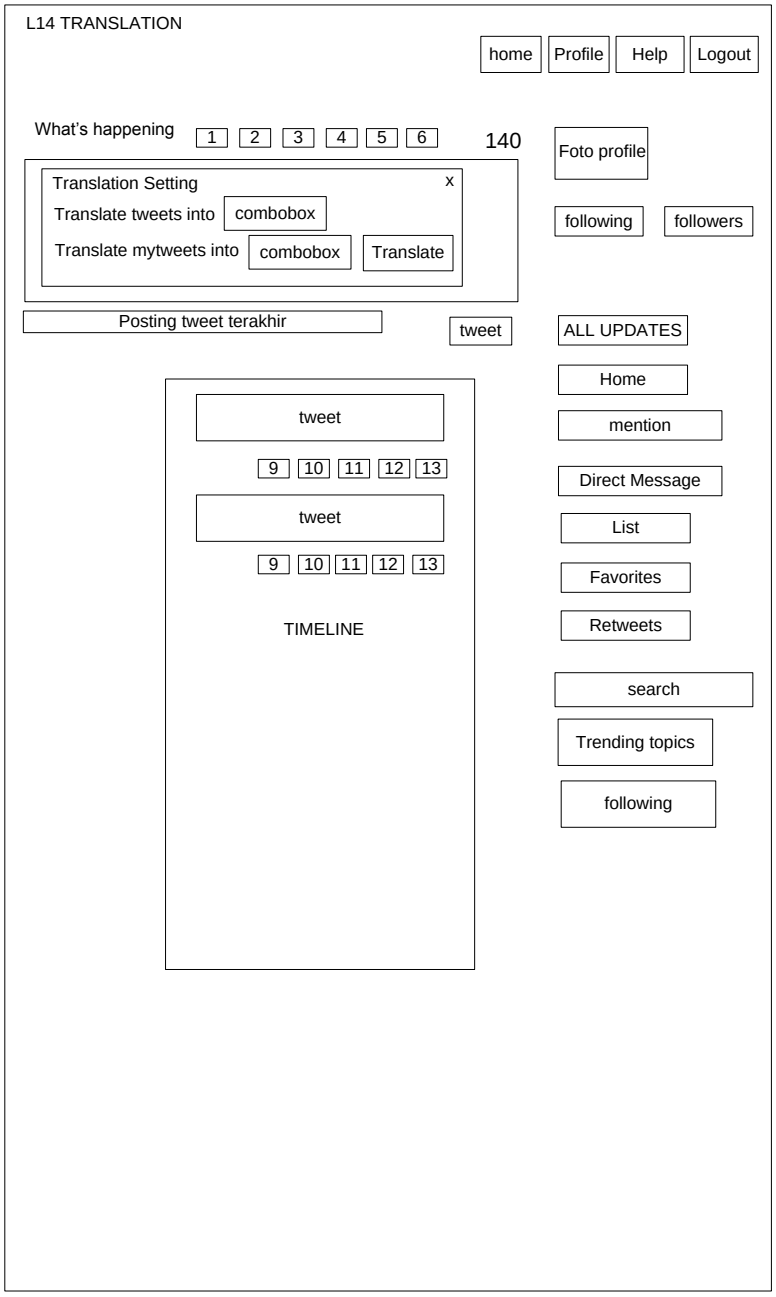
Gambar 3.66 Perancangan Antarmuka *Following*

Pada gambar 3.67 dapat dilihat perancangan antarmuka shorten url L13.



Gambar 3.67 Perancangan Antarmuka *Shorten Url*

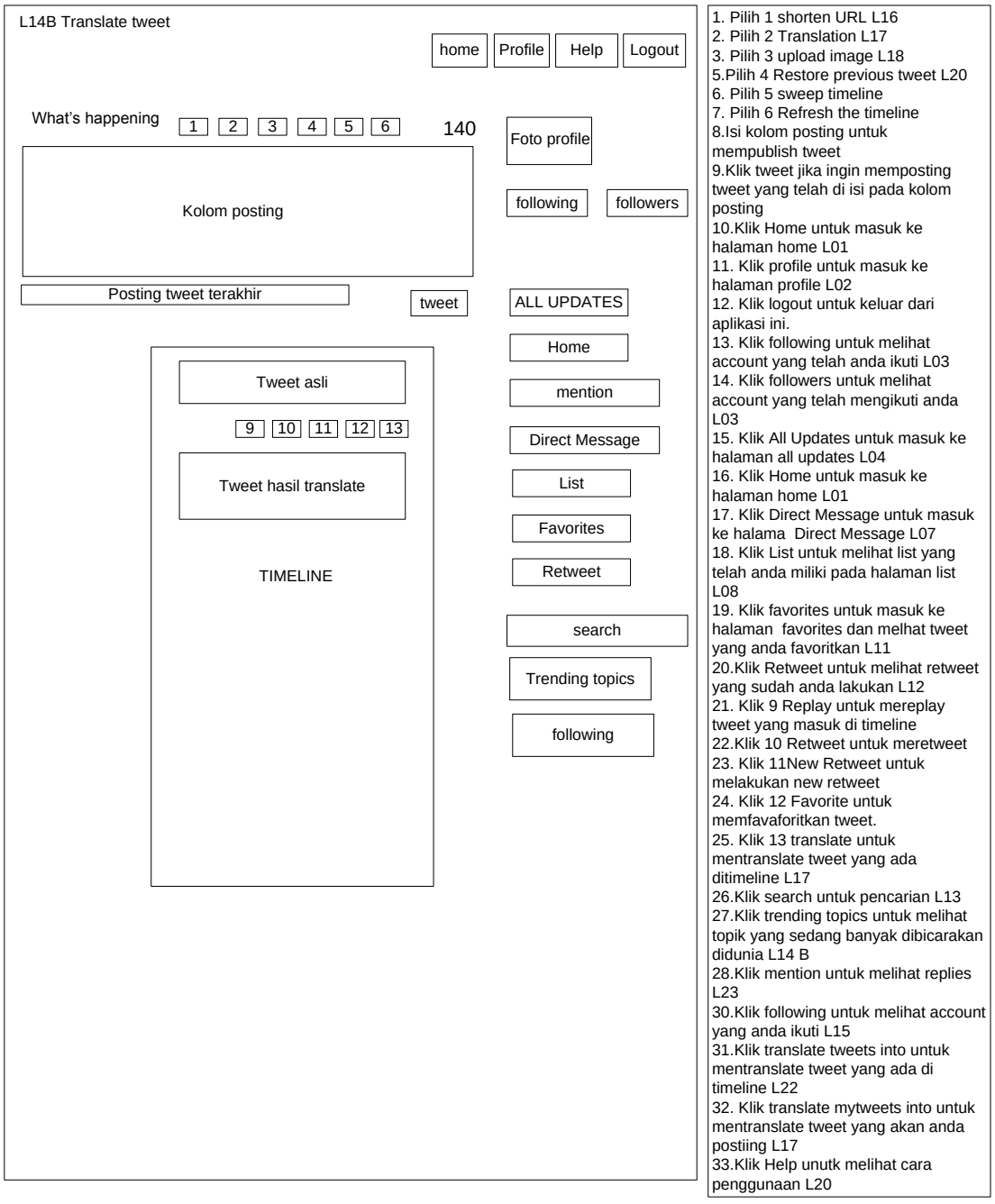
Pada gambar 3.68 dapat dilihat perancangan antarmuka translation L14.



1. Pilih 1 shorten URL L13
2. Pilih 2 Translation L14
3. Pilih 3 upload image L15
4.Pilih 4 Restore previous tweet
5. Pilih 5 sweep timeline untuk menghilangkan tweet yang berada di urutan paling bawah dari timeline.
6. Pilih 6 Refresh the timeline untuk merefresh timeline tanpa harus menunggu periode waktu yang telah ditentukan.
7.Isi kolom posting untuk mempublish tweet
8.Klik tweet jika ingin memposting tweet yang telah di isi pada kolom posting
9.Klik Home untuk masuk ke halaman home L01
10. Klik profile untuk masuk ke halaman profile L02
11. Klik logout untuk keluar dari aplikasi ini.
12. Klik following untuk melihat account yang telah anda ikuti L03
13. Klik followers untuk melihat account yang telah mengikuti anda.L03
14. Klik All Updates untuk masuk ke halaman all updates L04
15. Klik Home untuk masuk ke halaman home L01
16. Klik Direct Message untuk masuk ke halama Direct Message L06
17. Klik List untuk melihat list yang telah anda miliki pada halaman list L07
18. Klik favorites untuk masuk ke halaman favourites dan melhat tweet yang anda favoritkan L08
19.Klik Retweets untuk melihat retweet yang sudah anda lakukan L09
20. Klik 9 Replay untuk mereplay tweet yang masuk di timeline
21.Klik 10 Retweet untuk meretweet
22. Klik 11New Retweet untuk melakukan new retweet
23. Klik 12 Favorite untuk memfavaforitkan tweet.
24. Klik 13 translate untuk mentranslate tweet yang ada ditimeline L14 B
25.Klik search untuk pencarian L10
26.Klik trending topics untuk melihat topik yang sedang banyak dibicarakan didunia L11
27.Klik mention untuk melihat replies L05
28.Klik following untuk melihat account yang anda ikuti L12.
29.Klik Help untuk melihat cara penggunaan aplikasi L20

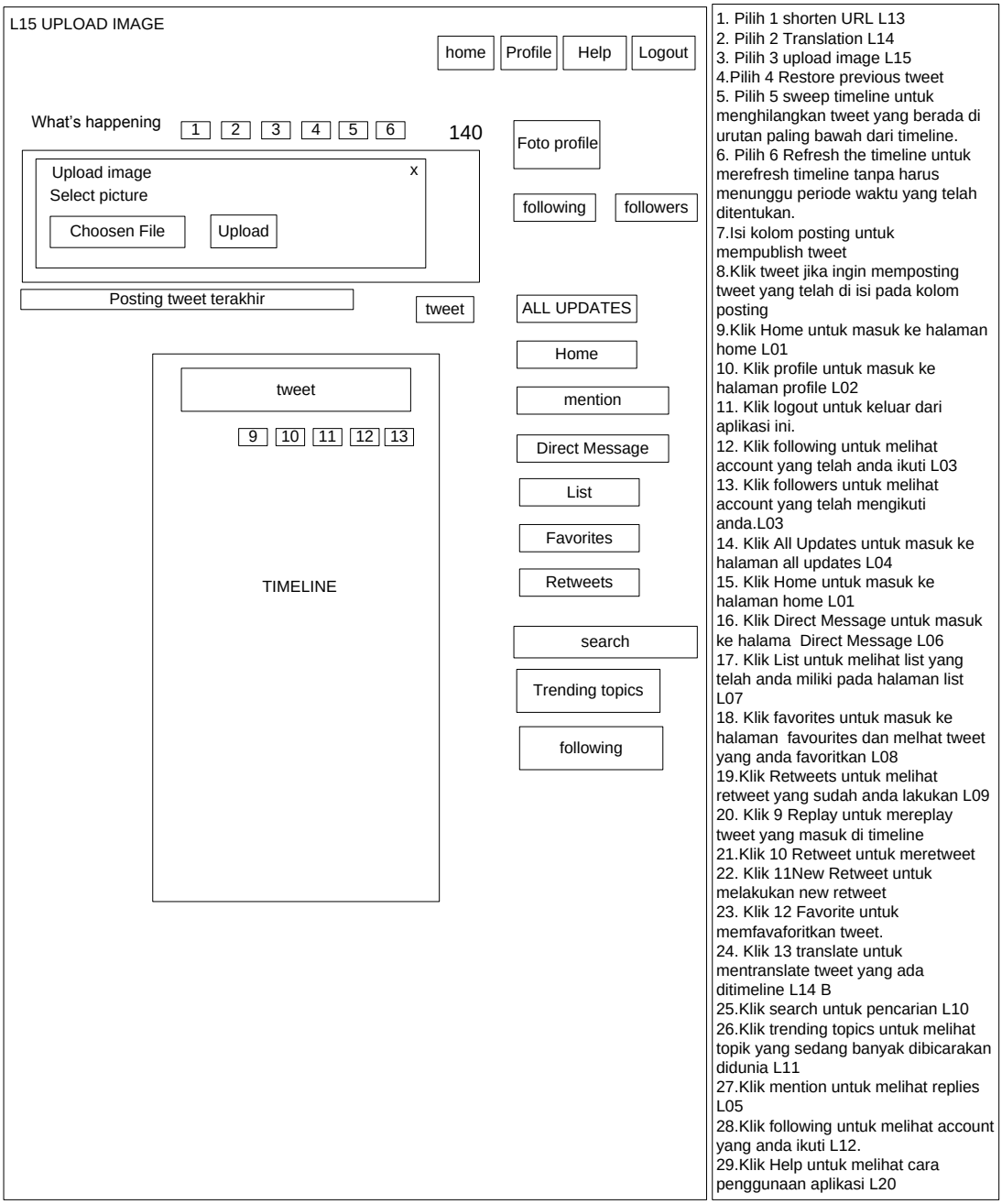
Gambar 3.68 Perancangan Antarmuka *Translation*

Pada gambar 3.69 dapat dilihat perancangan antarmuka translate tweet L14B.



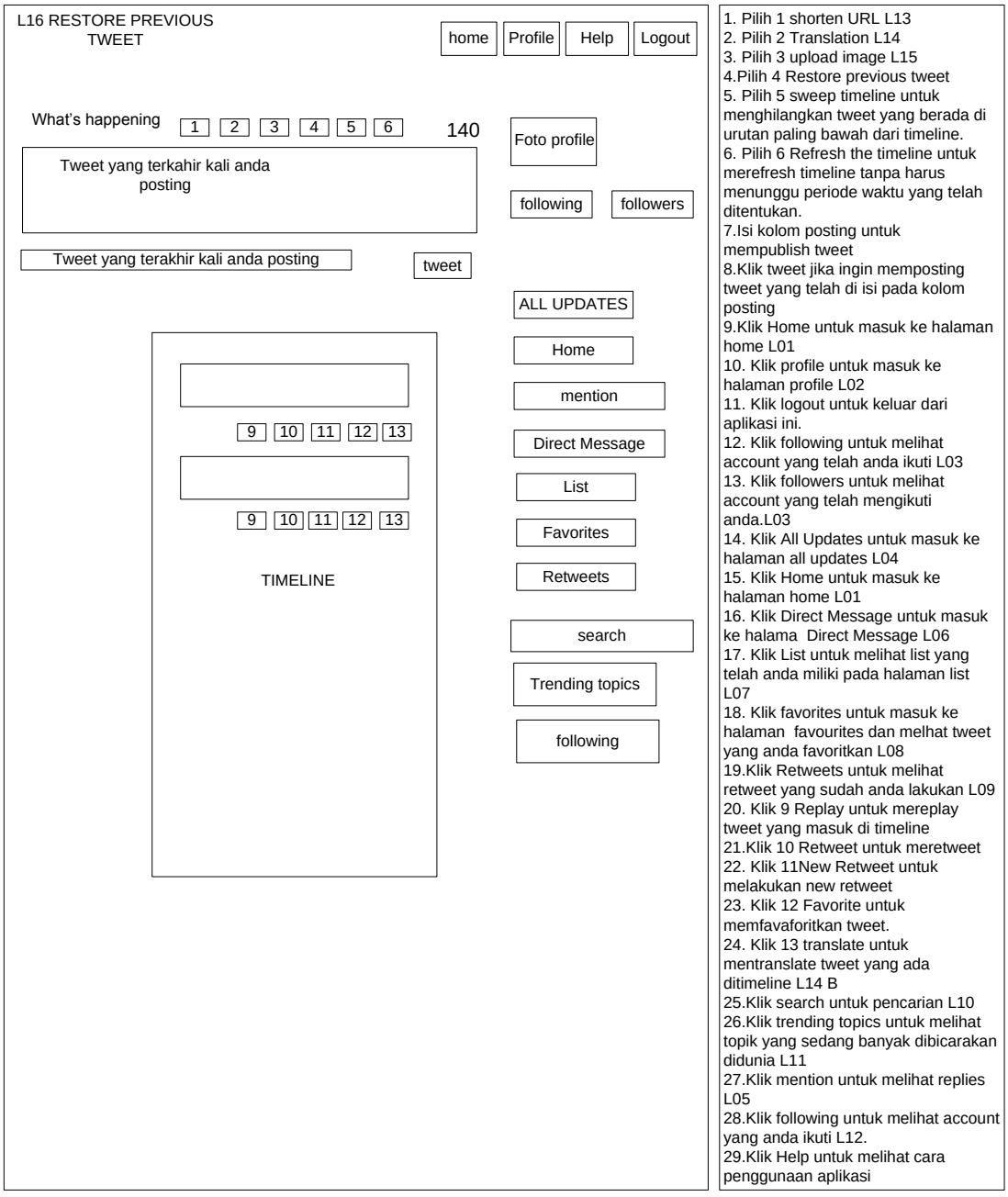
Gambar 3.69 Perancangan Antarmuka *Translate my Tweet*

Pada gambar 3.70 dapat dilihat perancangan antarmuka upload image L15.



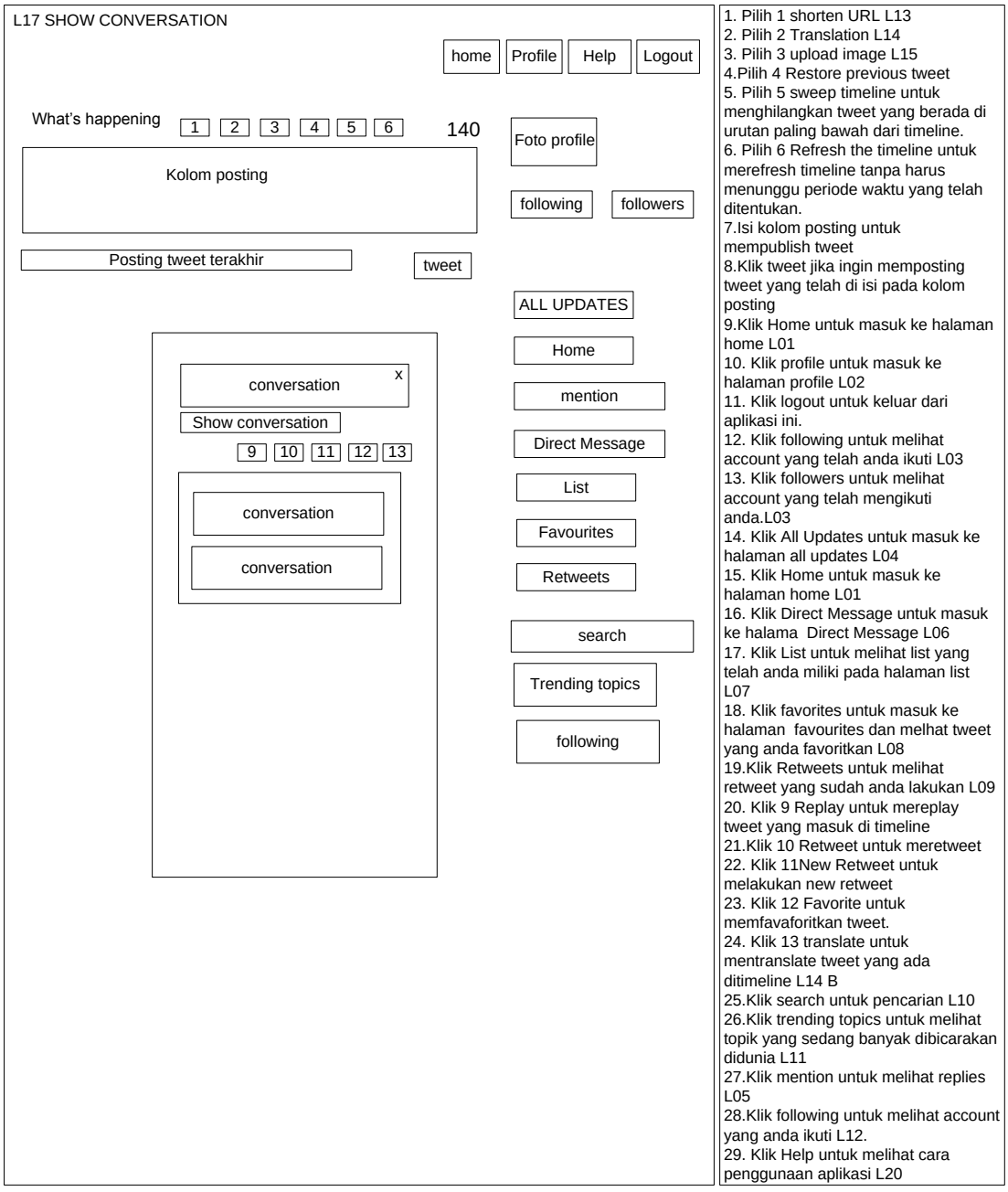
Gambar 3.70 Perancangan Antarmuka *Upload Image*

Pada gambar 3.71 dapat dilihat perancangan antarmuka restore previous tweet L16.



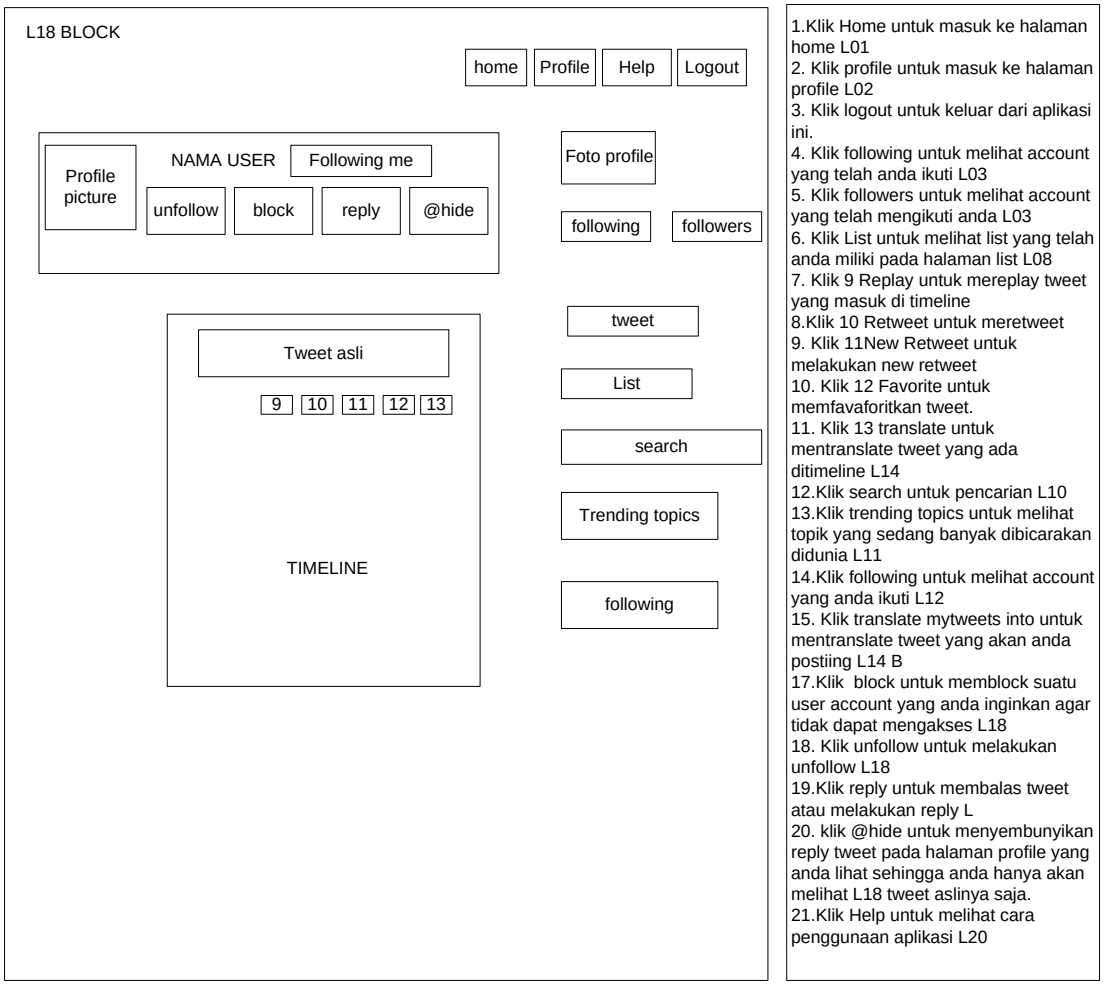
Gambar 3.71 Perancangan Antarmuka *Restore Previous Tweet*

Pada gambar 3.72 dapat dilihat perancangan antarmuka show conversation L17.



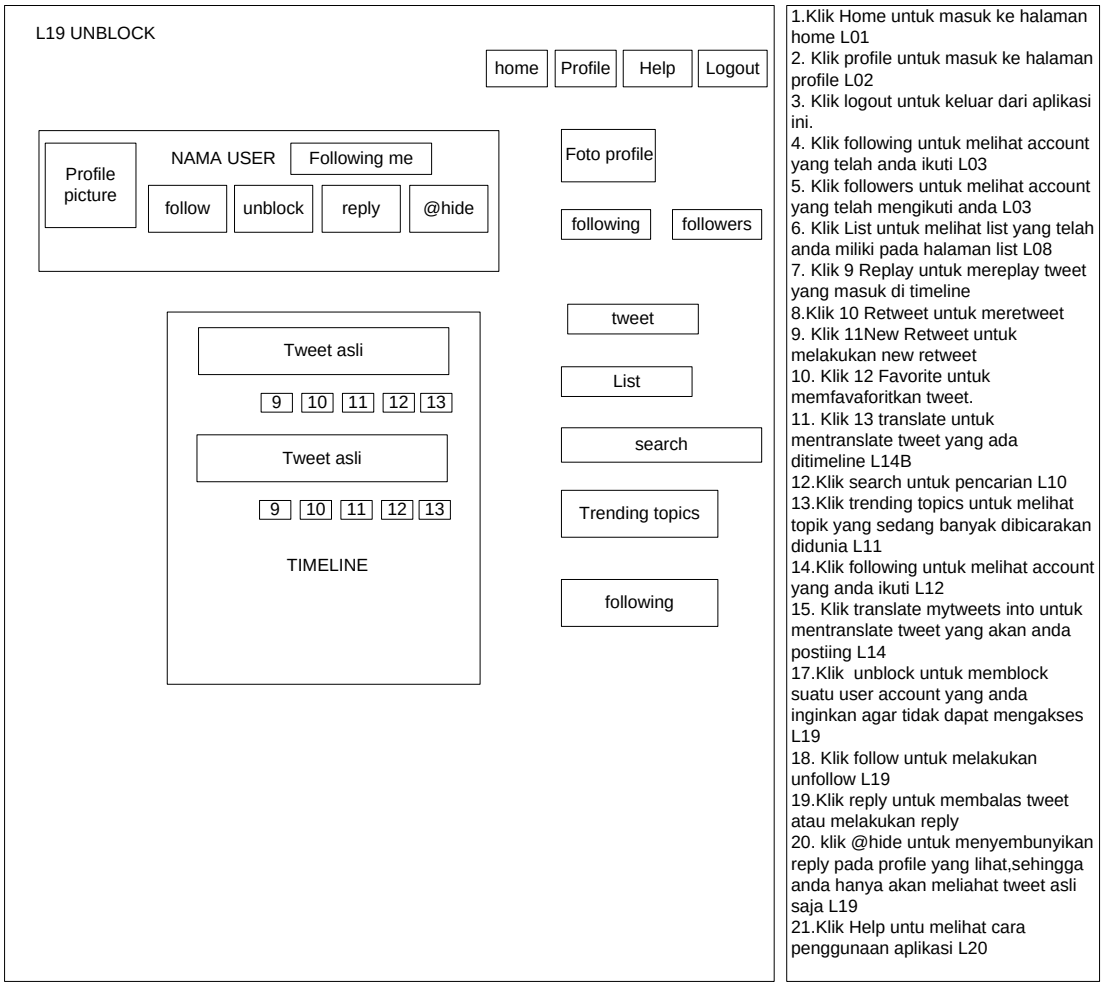
Gambar 3.72 Perancangan Antarmuka *show conversation*

Pada gambar 3.71 dapat dilihat perancangan antarmuka block L18.



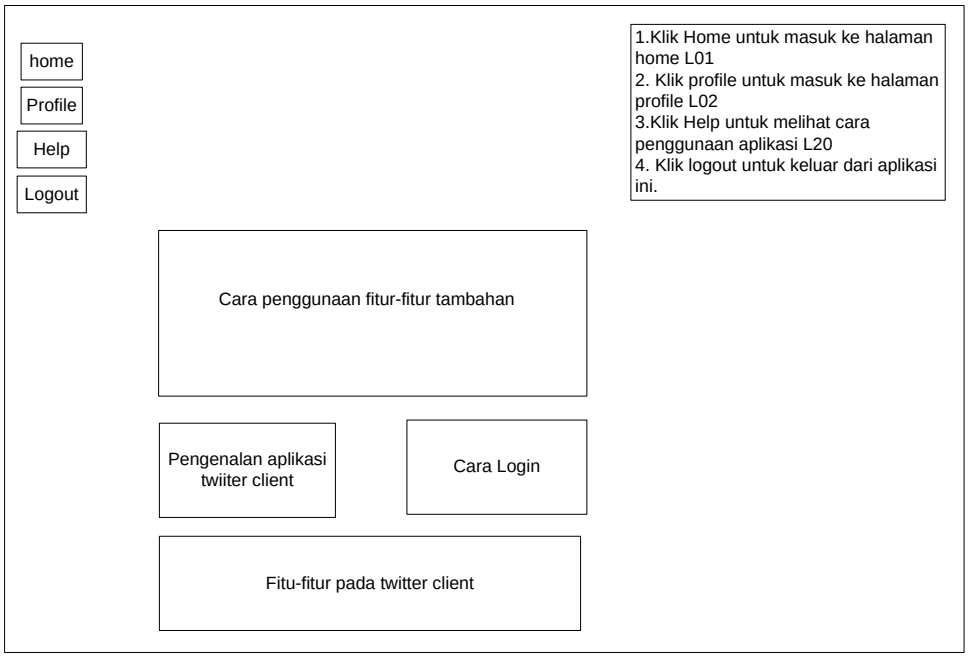
Gambar 3.73 Perancangan Antarmuka *Block*

Pada gambar 3.74 dapat dilihat perancangan antarmuka unblock L19



Gambar 3.74 Perancangan Antarmuka *Unblock*

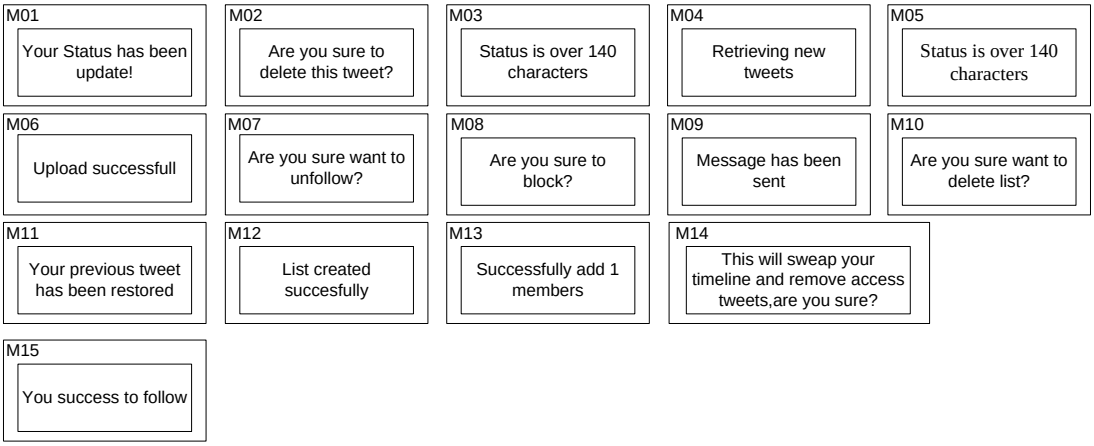
Pada gambar 3.75 dapat dilihat perancangan antarmuka help L20



Gambar 3.75 Perancangan Antarmuka *Help*

3.3.3 Perancangan Pesan

Perancangan pesan yang terdapat pada sistem ini dapat dilihat pada gambar 3.76 :



Gambar 3.76 Perancangan Pesan