



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB III

### METODOLOGI DAN PERANCANGAN APLIKASI

#### 3.1 Metodologi Penelitian

Adapun tahap-tahap metodologi penelitian aplikasi repositori tutorial pengembangan piranti lunak berbahasa Indonesia adalah sebagai berikut.

##### 1. Studi literatur

Pada tahap studi literatur akan dilakukan pembelajaran pada teori-teori terkait penelitian dari aplikasi yang akan dirancang dan dibangun dengan mencari literatur, jurnal, *paper*, dan referensi lain terkait topik yang dipilih.

##### 2. Analisis Kebutuhan

Pada tahap ini akan dijabarkan usaha terkait pengumpulan data, menjelaskan data yang diperlukan, serta teknik pengumpulan data yang digunakan.

##### 3. Perancangan

Pada tahap perancangan akan dibuat *Data Flow Diagram*, *Flowchart*, *User Interface*, *Entity Relationship Diagram*, skema *database*, dan struktur tabel.

##### 4. Implementasi

Hasil perancangan yang telah dibuat selanjutnya akan diimplementasikan ke dalam kode dengan bahasa pemrograman yang dipilih.

##### 5. Uji Coba

Setelah aplikasi selesai dirancang dan dibangun, akan dilaksanakan pengujian terhadap aplikasi tersebut dengan melakukan pencarian tutorial pada repositori dan menghitung *precision*, *recall*, *f-measure*, MAP, serta pengujian penerimaan aplikasi.

## 6. Penulisan Laporan

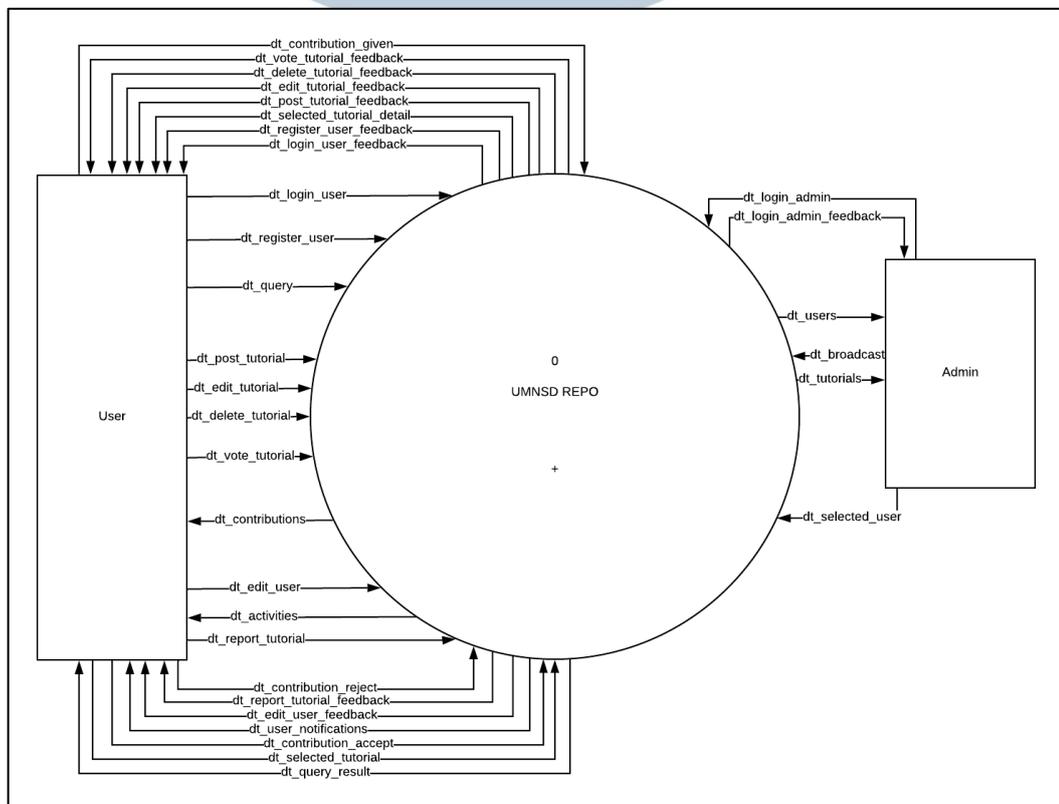
Pada tahap ini akan dilakukan pencatatan dari hasil uji coba. Setiap hasil yang didapat akan dihitung keberhasilannya untuk mengetahui dampak dari aplikasi yang telah dibangun

### 3.2 Perancangan Aplikasi

Proses perancangan aplikasi dimulai dengan membuat *Data Flow Diagram* (DFD), *Flowchart*, *Entity Relationship Diagram* (ERD), skema *database*, struktur tabel, serta tampilan antarmuka.

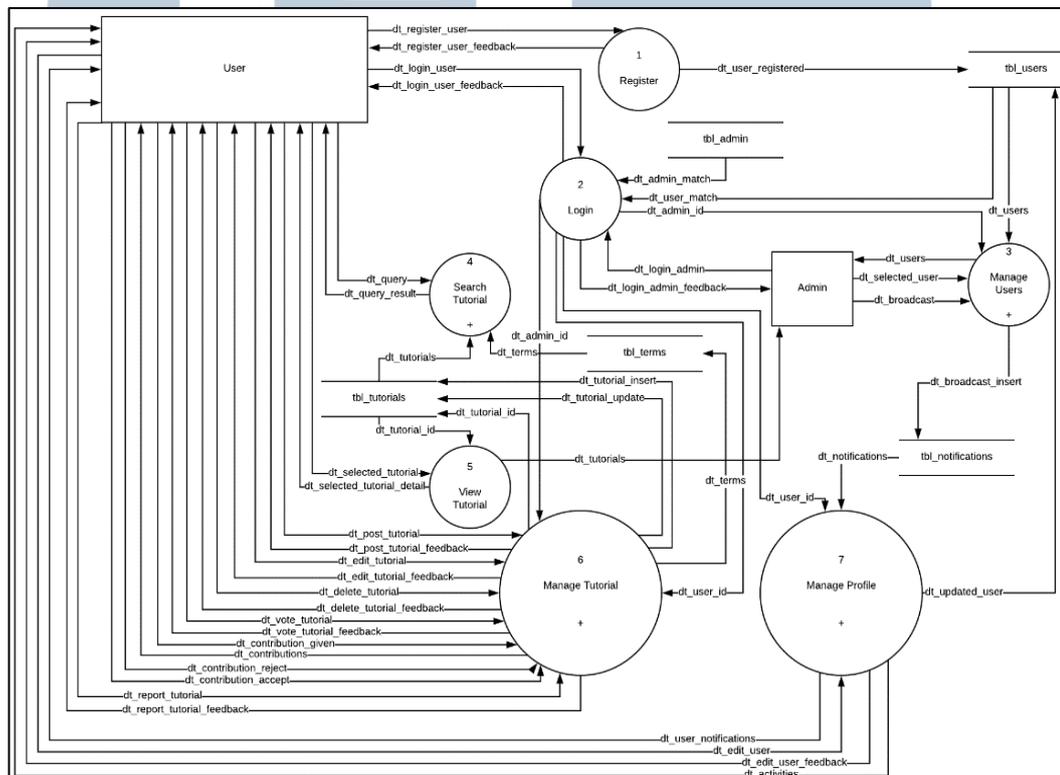
#### 3.2.1 Data Flow Diagram

*Data Flow Diagram* menggambarkan data yang mengalir pada sistem untuk membantu memahami logika sistem secara terstruktur dan jelas.



Gambar 3.1 Context Diagram

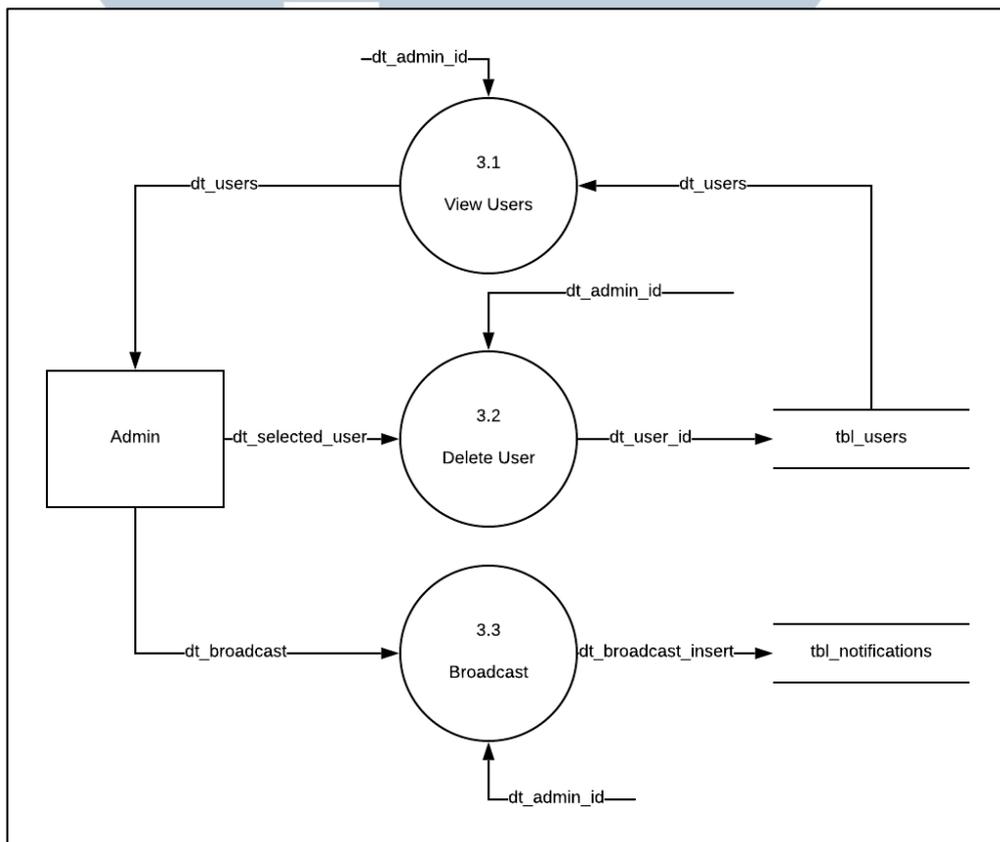
Aplikasi ini memiliki dua entitas, yaitu *user* dan *admin*. Entitas *user* merupakan pengguna yang melakukan pencarian tutorial maupun manajemen tutorial pada aplikasi, sedangkan entitas *admin* merupakan pengguna aplikasi yang melakukan manajemen tutorial dan manajemen pengguna secara umum. Gambar 3.1 merupakan DFD *Context Diagram* yang menggambarkan data yang mengalir pada aplikasi secara garis besar.



Gambar 3.2 DFD Level 0

Adapun DFD Level 0 dari aplikasi ini ditunjukkan pada Gambar 3.2. Pada level ini terdapat tujuh proses, yaitu proses Register, Login, Manage Users, Search Tutorial, View Tutorial, Manage Tutorial, dan Manage Profile. Pada Proses Register, *user* diminta memberikan data identitas sesuai dengan kebutuhan aplikasi, setelah data selesai dimasukkan, sistem akan memberikan *feedback* apakah proses registrasi tersebut berhasil atau gagal. Adapun Proses Login bisa dilakukan oleh

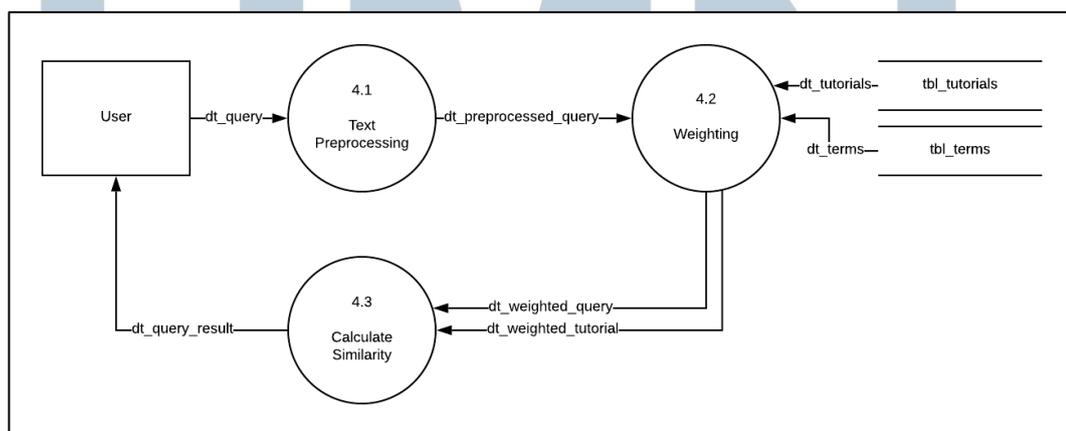
*user* ataupun *admin*. Pada proses ini *user* memasukkan data *login* sesuai permintaan sistem, setelah data di-*submit*, proses *login* akan memberikan *feedback* apakah proses *login* berhasil atau gagal. Proses Manage Users dilakukan oleh *admin*, manajemen *user* hanya bisa dilakukan jika *admin* berhasil melakukan proses *login*. Pada proses Search Tutorial, *query* dimasukkan oleh *user*, kemudian proses akan mengembalikan hasil pencarian sesuai dengan *query* yang telah dimasukkan oleh *user*. Proses pencarian tutorial dapat dilakukan tanpa melakukan proses *login*. Pada proses View Tutorial, *user* dan *admin* memilih tutorial dan sistem akan menampilkan detail dari tutorial yang dicari. *User* bisa melakukan manajemen profil dan manajemen tutorial terhadap tutorial.



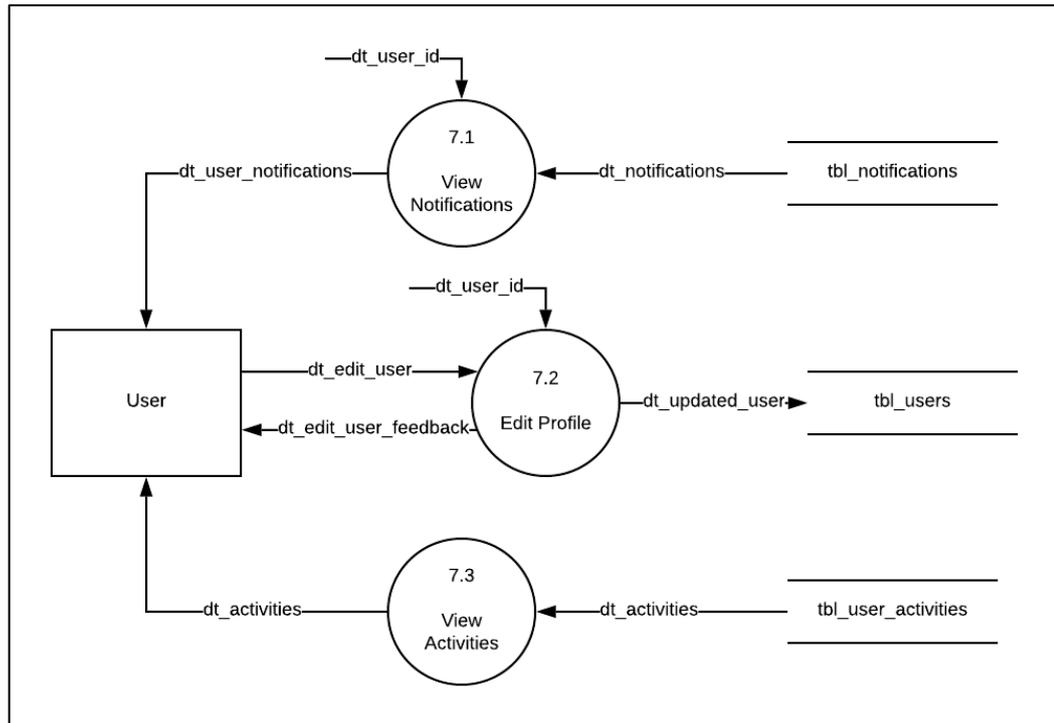
Gambar 3.3 DFD Level 1 Subproses Manage Users

Pada subproses Manage Users seperti yang ditunjukkan pada Gambar 3.3, *admin* bisa melihat *user* yang telah terdaftar pada aplikasi dan menghapus *user*. Data *user* diambil dari tabel *users*. Adapun pada proses Broadcast, *admin* bisa memberikan pengumuman terkait hal yang perlu diberitahukan kepada *user*. Data *broadcast* akan dimasukkan ke tabel notifikasi.

Gambar 3.4 menunjukkan subproses Search Tutorial. Pada subproses ini Vector Space Model diimplementasikan. Subproses ini terdiri dari tiga proses, yaitu proses Text Preprocessing, Weighting, dan Calculate Similarity. Proses pencarian dimulai dengan mengambil *query user*, kemudian *query* tersebut di-*preprocess* pada proses Text Preprocessing. Setelah *query* di-*preprocess*, dilakukan pembobotan terhadap masing-masing *query* dan data tutorial yang diambil dari tabel pada proses Weighting. Kemudian dilakukan perhitungan probabilitas kecocokan antara *query* yang dimasukkan oleh *user* dengan semua tutorial yang terdapat pada tabel pada proses Calculate Similarity dan memberikan hasil berdasarkan perhitungan pada proses tersebut.



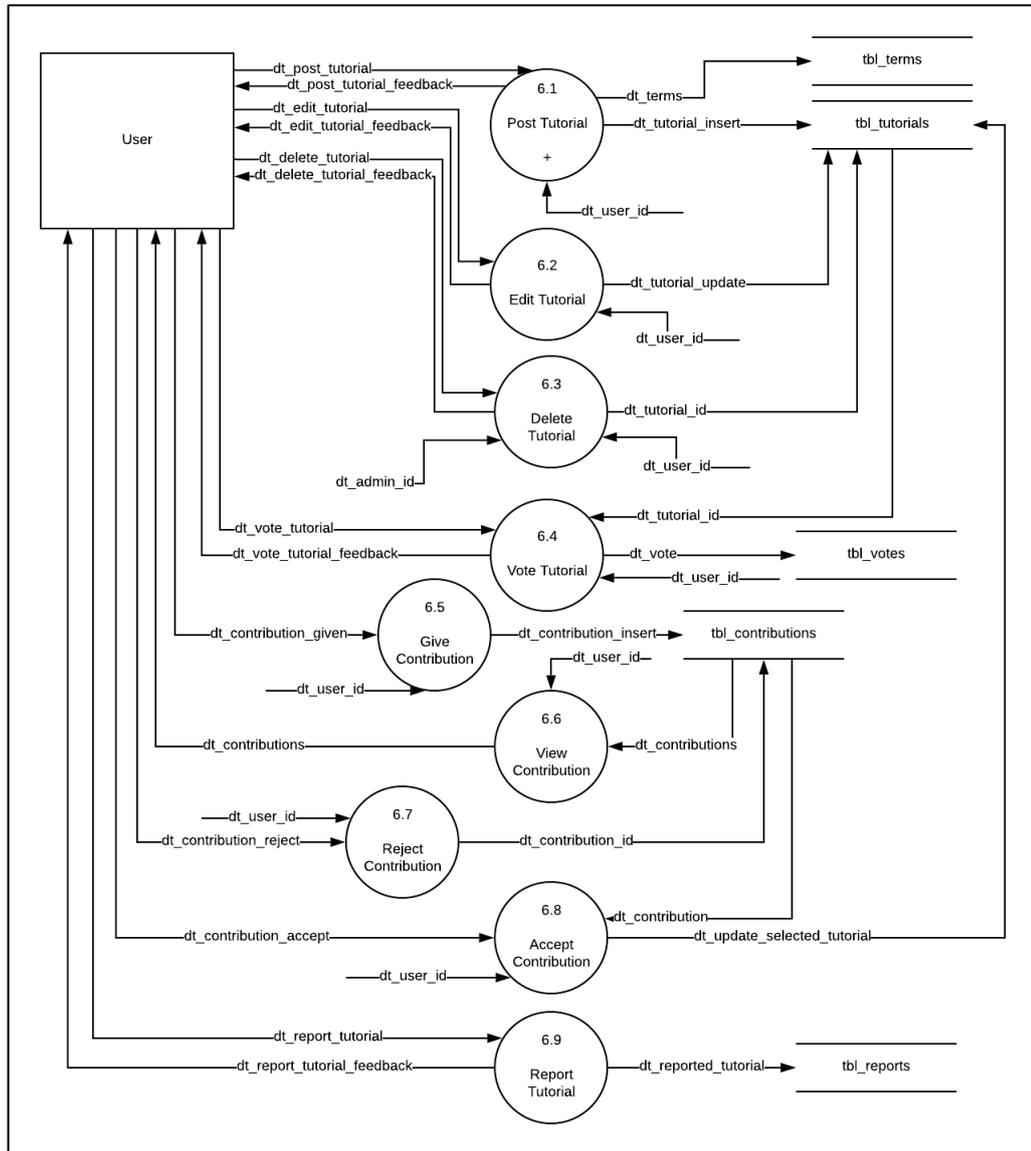
Gambar 3.4 DFD Level 1 Subproses Search Tutorial



Gambar 3.5 Subproses Manage Profile

Gambar 3.5 menunjukkan subproses dari Manage Profile. Pada subproses ini, terdapat tiga proses, yaitu proses View Notifications, Edit Profile, serta View Activities. Proses View Notifications berfungsi untuk menampilkan notifikasi seperti pemberitahuan dari *admin* atau *user* lain pada aplikasi. Proses Edit Profile berfungsi untuk melakukan perbaruan terhadap data diri yang telah dimasukkan setelah registrasi. Adapun proses View Activities berfungsi untuk menampilkan kegiatan *user* dimana data diambil dari tabel activities.

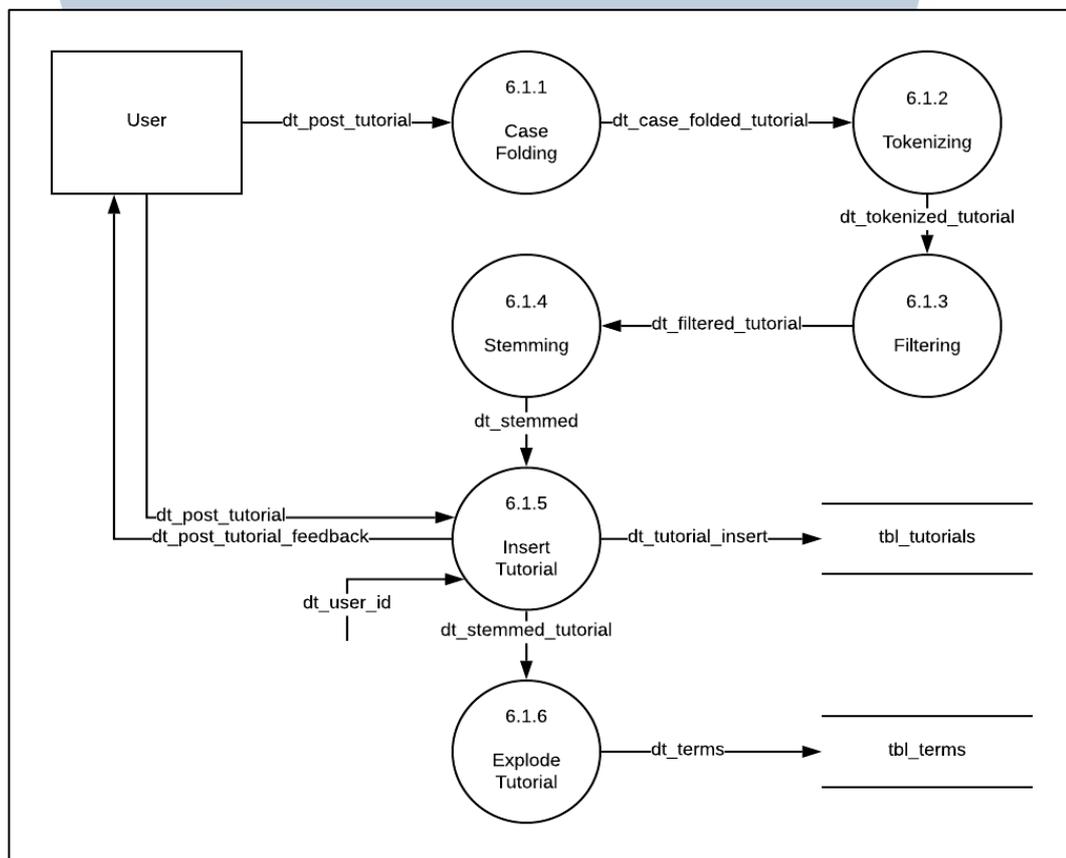
DFD untuk subproses manajemen tutorial ditunjukkan pada Gambar 3.6. Pada subproses ini terdapat sembilan proses, yaitu proses Post Tutorial, Edit Tutorial, Delete Tutorial, Vote Tutorial, Give Contribution, View Contribution, Reject Contribution, Accept Contribution, dan Report Tutorial.



Gambar 3.6 DFD Level 1 Subproses Manage Tutorial

Proses Post Tutorial berfungsi untuk menambahkan tutorial baru pada aplikasi dan memberikan *feedback* terhadap kegiatan yang dilakukan *user*. Proses Edit Tutorial berfungsi untuk melakukan pembaruan terhadap tutorial yang dibuat oleh *user* dan memberikan *feedback* terhadap pembaruan. Adapun proses Delete Tutorial berfungsi untuk menghapus tutorial yang dimiliki oleh *user* dan memberikan *feedback* bahwa tutorial telah dihapus. Proses Vote Tutorial berfungsi untuk memberikan *vote* terhadap tutorial milik *user* lain. Proses Give Contribution

berfungsi untuk memberikan saran pembaruan terhadap tutorial milik *user* lain. Saran pembaruan tersebut dimasukkan kedalam tabel *contributions*. Kontribusi berupa saran pembaruan terhadap tutorial tersebut bisa dilihat oleh pemilik tutorial pada proses View Contribution. Proses Reject Contribution berfungsi untuk menolak saran pembaruan dari *user*. Proses Accept Contribution berfungsi untuk menerima saran pembaruan *user*. Jika saran pembaruan tersebut diterima, tutorial yang dimaksud akan diperbarui. Adapun proses Report Tutorial berfungsi untuk melaporkan tutorial atas alasan tertentu dari *user* tersebut.



Gambar 3.7 DFD Level 2 Subproses Post Tutorial

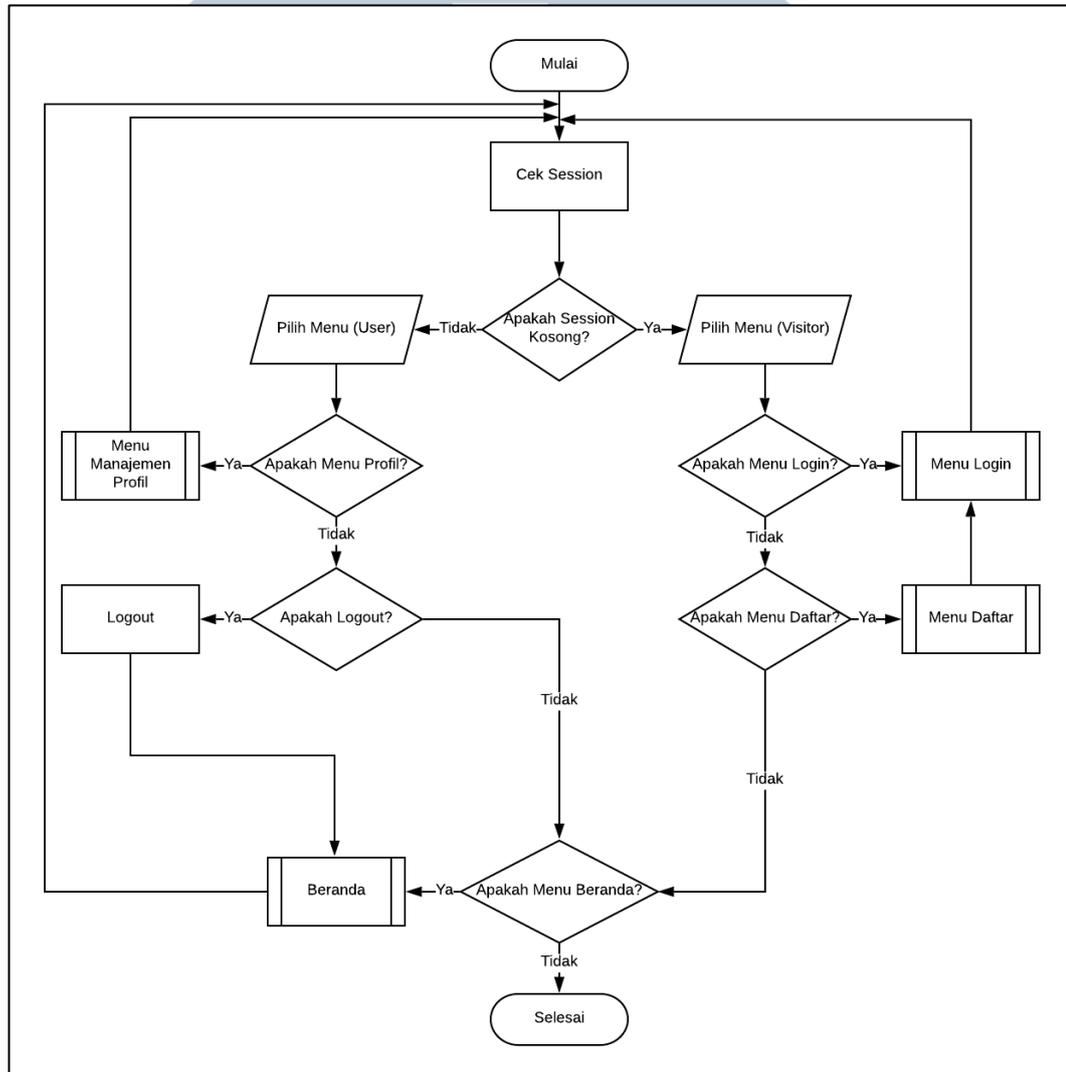
Gambar 3.7 menunjukkan subproses dari Post Tutorial. Subproses ini terdiri dari enam proses, yaitu proses Case Folding, Tokenizing, Filtering, Stemming,

Insert Tutorial, dan Explode Tutorial. Sebelum data tutorial yang telah dibuat *user* dimasukkan ke tabel tutorial, data tutorial tersebut melewati beberapa proses. Proses Case Folding berfungsi untuk membuat data tutorial menjadi *lowercase*. Setelah itu, pada proses Tokenizing dilakukan penghilangan tanda baca dan mengekstrak data tutorial sehingga menghasilkan kata-kata yang berdiri masing-masing. Data tutorial yang telah melewati proses Tokenizing akan difilter pada proses Filtering untuk menghilangkan kata-kata yang dianggap tidak layak untuk dijadikan sebagai pembeda berdasarkan daftar *stopword* yang ada. Kemudian, setiap kata yang ada pada tutorial tersebut diubah menjadi kata dasar pada proses Stemming. Pada proses Insert Tutorial akan disimpan data tutorial yang telah melewati proses Case Folding hingga Stemming maupun data yang belum melewati proses tersebut pada tabel tutorial. Adapun setiap term yang ada pada data tutorial yang telah di-*stem* pada proses Stemming disimpan dalam tabel *terms* melalui proses Explode Tutorial.

### 3.2.2 Flowchart

Setelah *Data Flow Diagram* dirancang, tahap selanjutnya adalah pembuatan *flowchart* untuk menjelaskan alur aplikasi TERESTRIAL ini. Gambar 3.8 menunjukkan menu utama dari aplikasi ini. Ketika aplikasi dijalankan, dilakukan pengecekan *session*. Jika *session* kosong maka menu yang disediakan yaitu menu umum Beranda, Login, dan Daftar. Jika Login dipilih maka akan ditampilkan halaman dari Menu Login, jika Daftar dipilih maka akan ditampilkan halaman dari Menu Daftar, jika Beranda dipilih maka akan ditampilkan Menu Beranda. Adapun jika *session* tidak kosong maka menu yang disediakan yaitu Menu Beranda, Menu Profil, serta Logout. Jika Menu Profil dipilih, maka ditampilkan halaman untuk

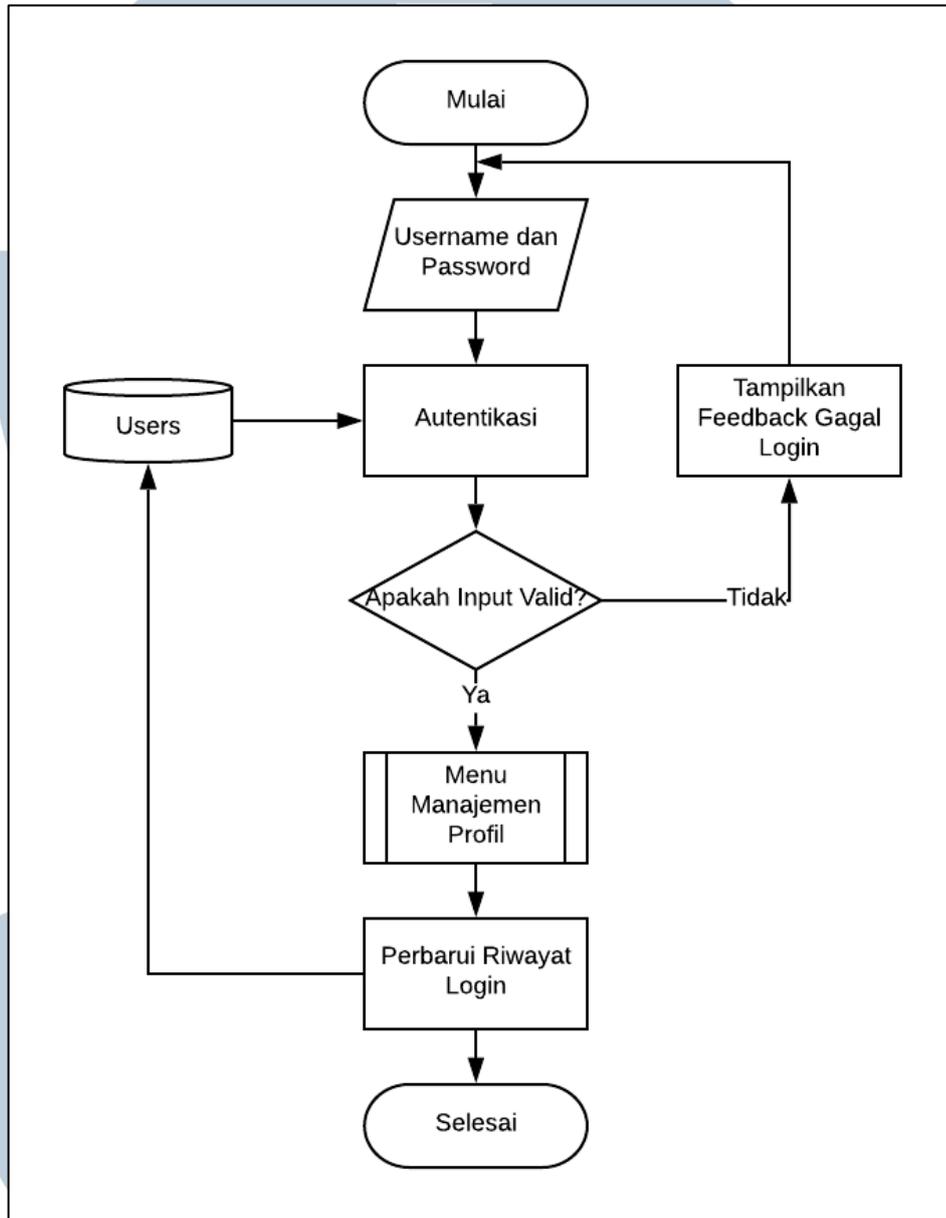
melakukan manajemen profil. Jika Logout dipilih maka session dihapus dan menu yang disediakan adalah menu umum kembali.



Gambar 3.8 Menu Utama

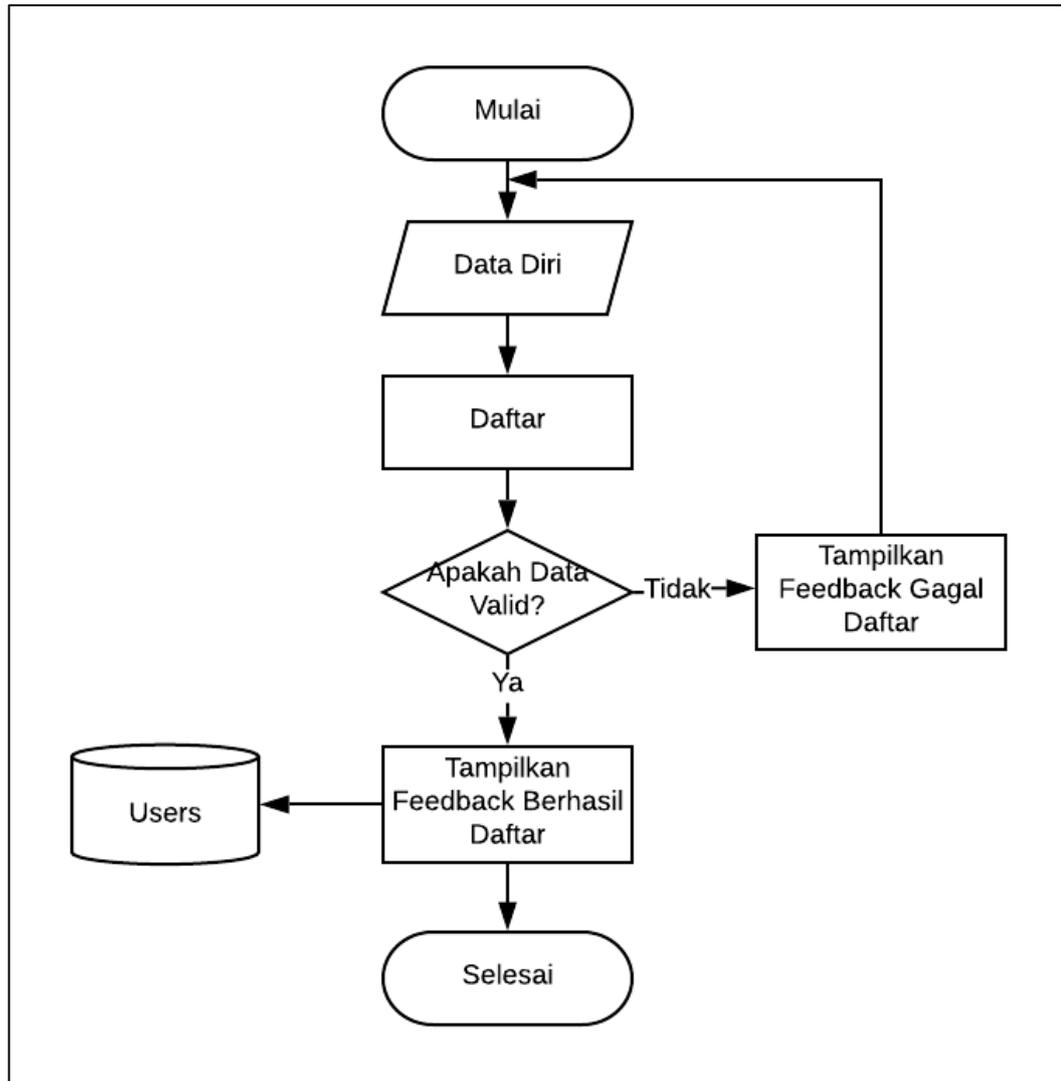
Flowchart untuk login ditunjukkan pada Gambar 3.9. Pada menu ini, data input berupa *username* dan *password* dimasukkan. Kemudian, dilakukan proses autentikasi dengan mencocokkan data *user* pada *database*. Jika data *login* tersebut valid, Menu Profil ditampilkan serta riwayat *login* diperbarui. Adapun jika data

login tidak valid, ditampilkan *feedback* kepada *user* bahwa data yang dimasukkan oleh *user* tersebut tidak valid.



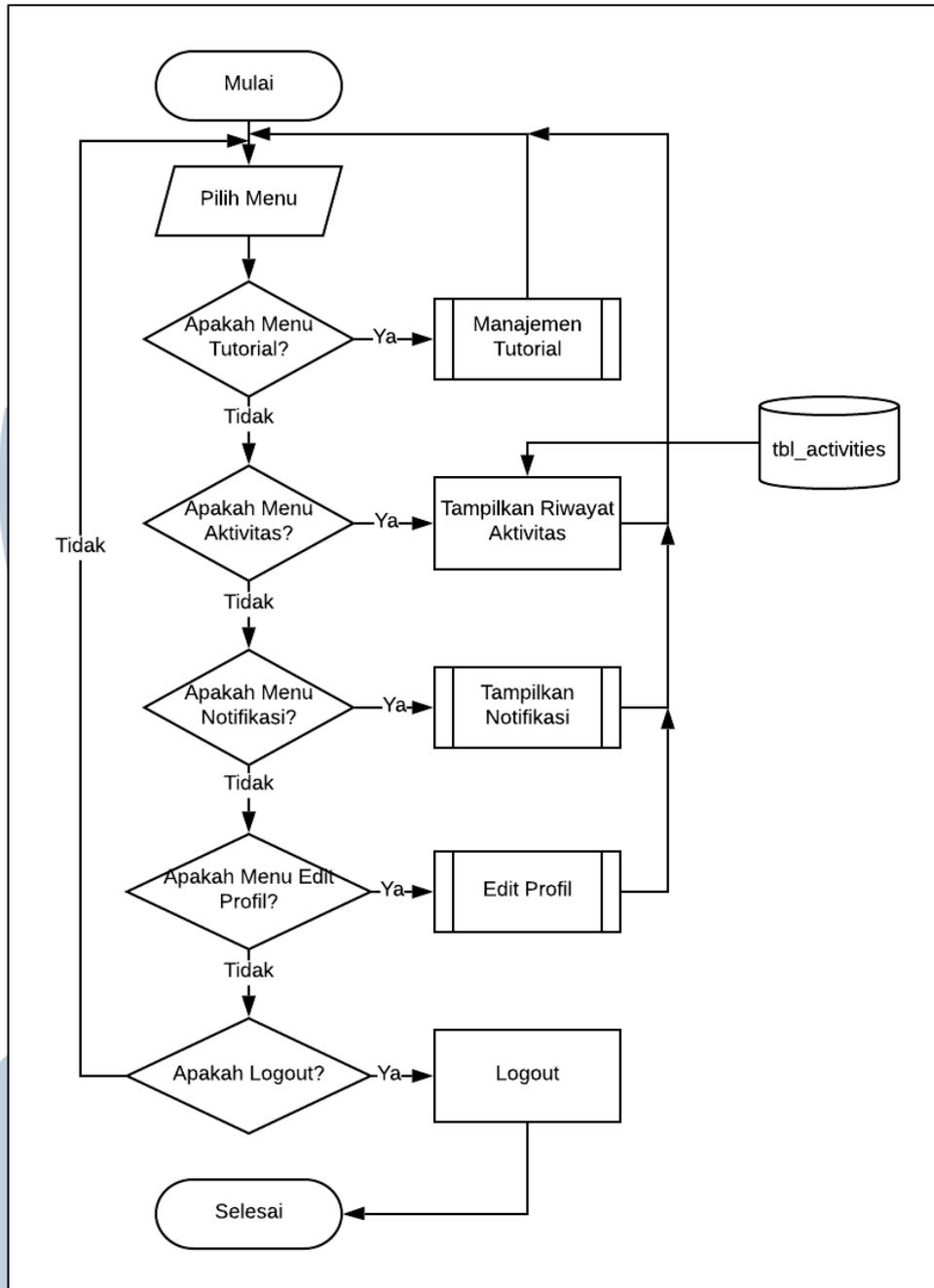
Gambar 3.9 Flowchart Login User

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.10 Flowchart Register User

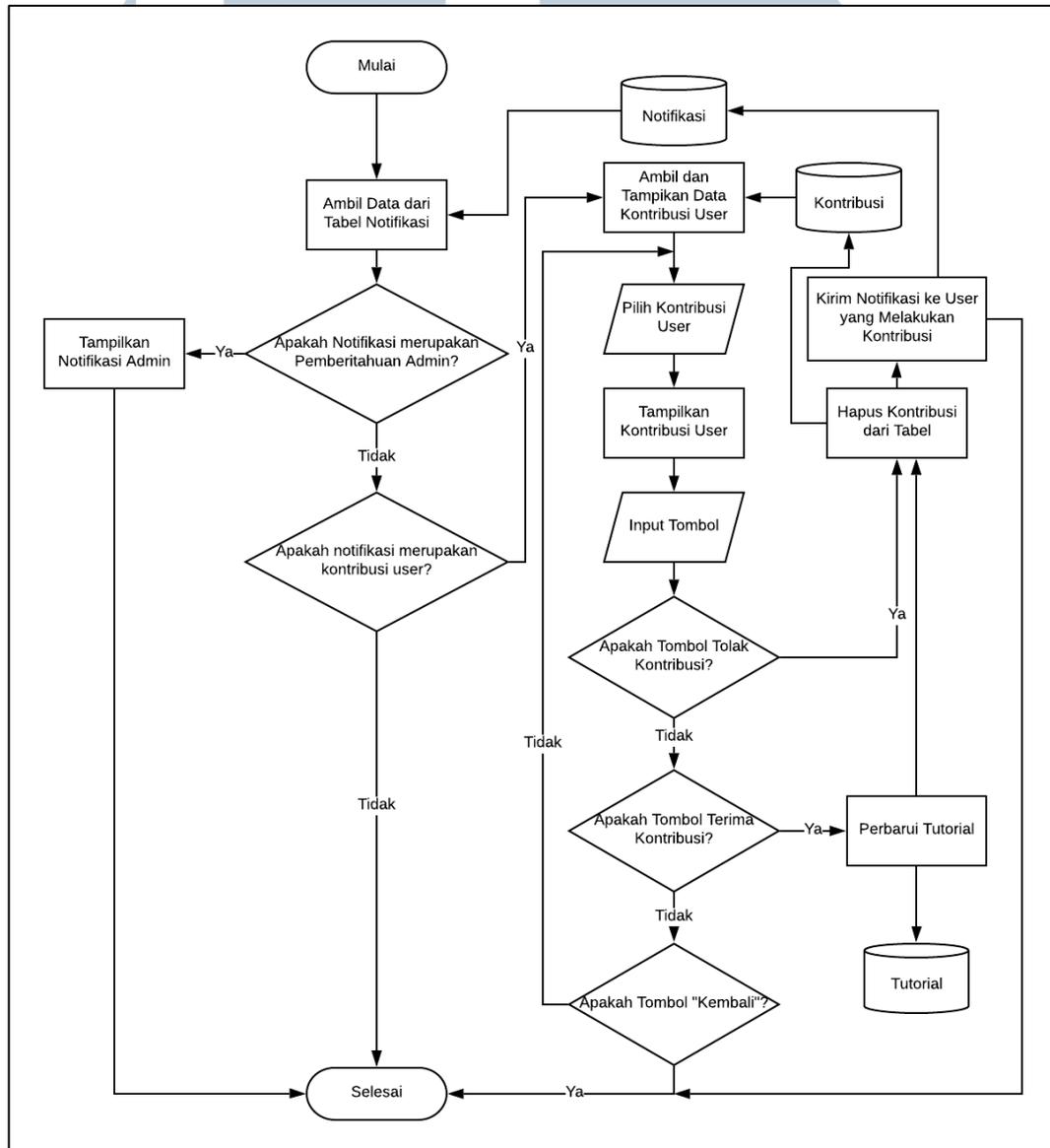
Jika *user* belum terdaftar, *user* bisa melakukan registrasi pada Menu Register. Pada menu ini, dilakukan input berupa data diri. Jika data registrasi valid, data *user* tersebut disimpan ke tabel. Jika data registrasi tidak valid, ditampilkan *feedback* bahwa registrasi gagal dilakukan. Adapun *flowchart* untuk melakukan pendaftaran ditunjukkan pada Gambar 3.10.



Gambar 3.11 Flowchart Manajemen Profil

Gambar 3.11 menunjukkan *flowchart* dari Menu Manajemen Profil. Jika Menu Tutorial dipilih, maka akan ditampilkan Halaman untuk melakukan manajemen tutorial. Jika Menu Aktivitas dipilih, maka akan ditampilkan riwayat aktivitas *user* yang diambil dari tabel *activities*. Jika Menu Notifikasi dipilih, maka sistem akan

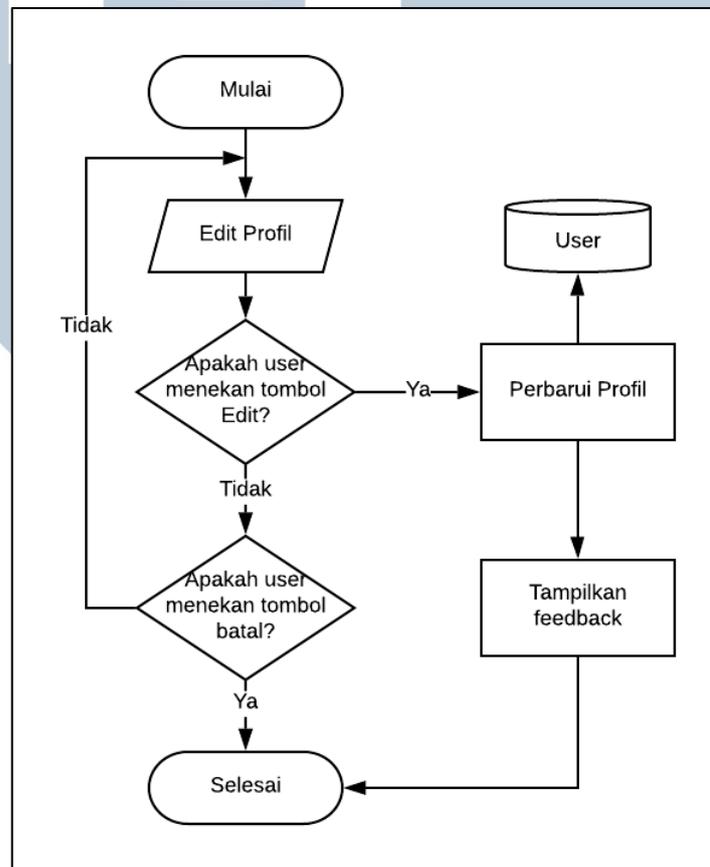
menampilkan Notifikasi yang ditujukan kepada *user*. Jika Menu Edit Profil dipilih, maka ditampilkan halaman untuk melakukan pembaruan profil. Adapun jika Logout dipilih, maka sistem akan kembali menampilkan menu umum.



Gambar 3.12 Flowchart Notifikasi

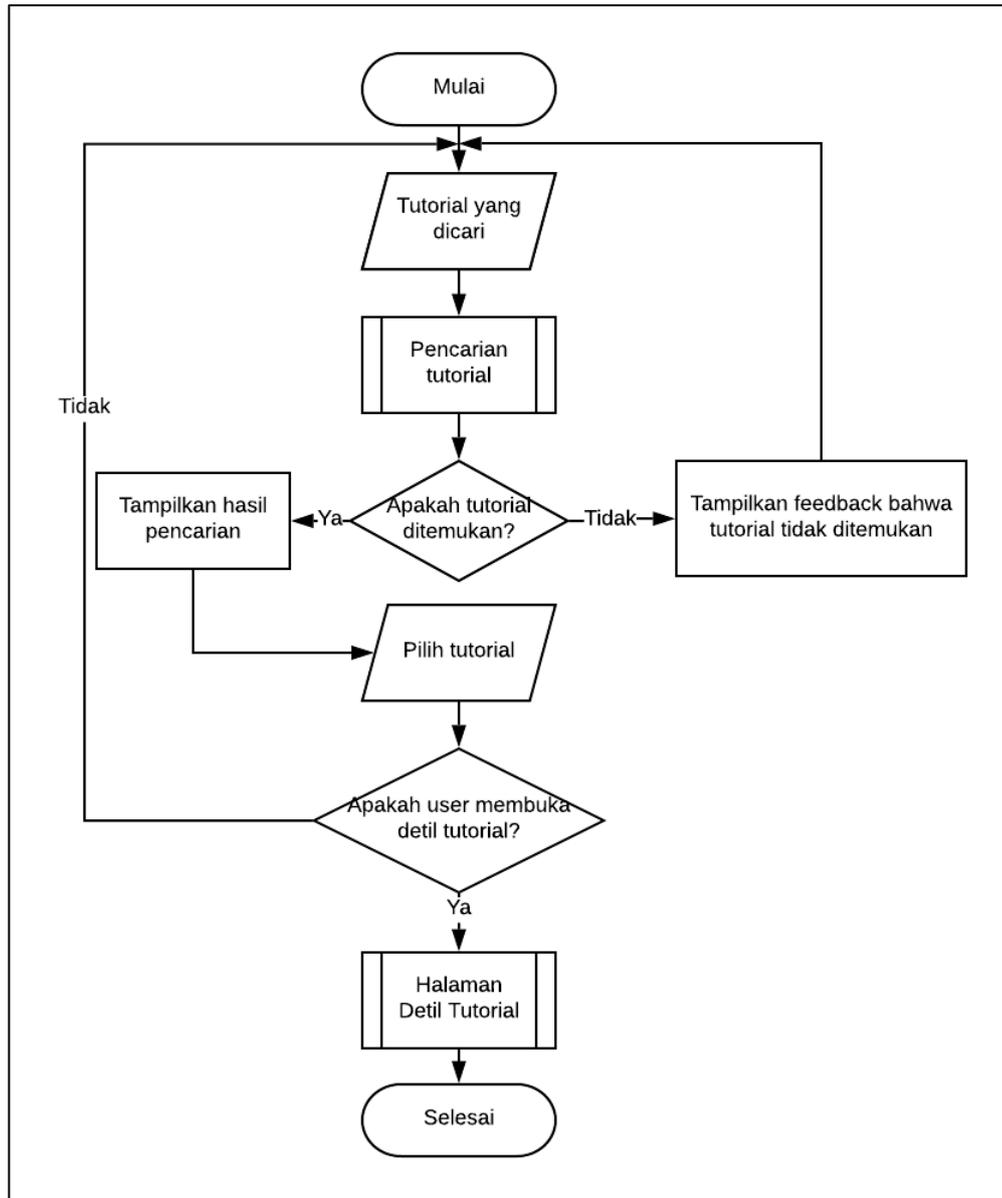
Flowchart notifikasi ditunjukkan pada Gambar 3.12. Data notifikasi diambil dari tabel notifikasi. Notifikasi bisa berupa informasi dari *admin* atau pemberitahuan kontribusi dari *user*. Jika jenis notifikasi tersebut merupakan

pemberitahuan dari *admin*, maka pemberitahuan hanya sekadar ditampilkan. Jika jenis notifikasi merupakan kontribusi dari *user* lain, maka ditampilkan data kontribusi dari *user* tersebut. Pemilik tutorial bisa melihat detail kontribusi dari *user* tersebut. Jika kontribusi ditolak, maka tidak ada perubahan pada tutorial. Adapun jika kontribusi diterima, maka tutorial akan diperbarui sesuai saran pembaruan dari *user* lain tersebut.



Gambar 3.13 Flowchart Edit Profil

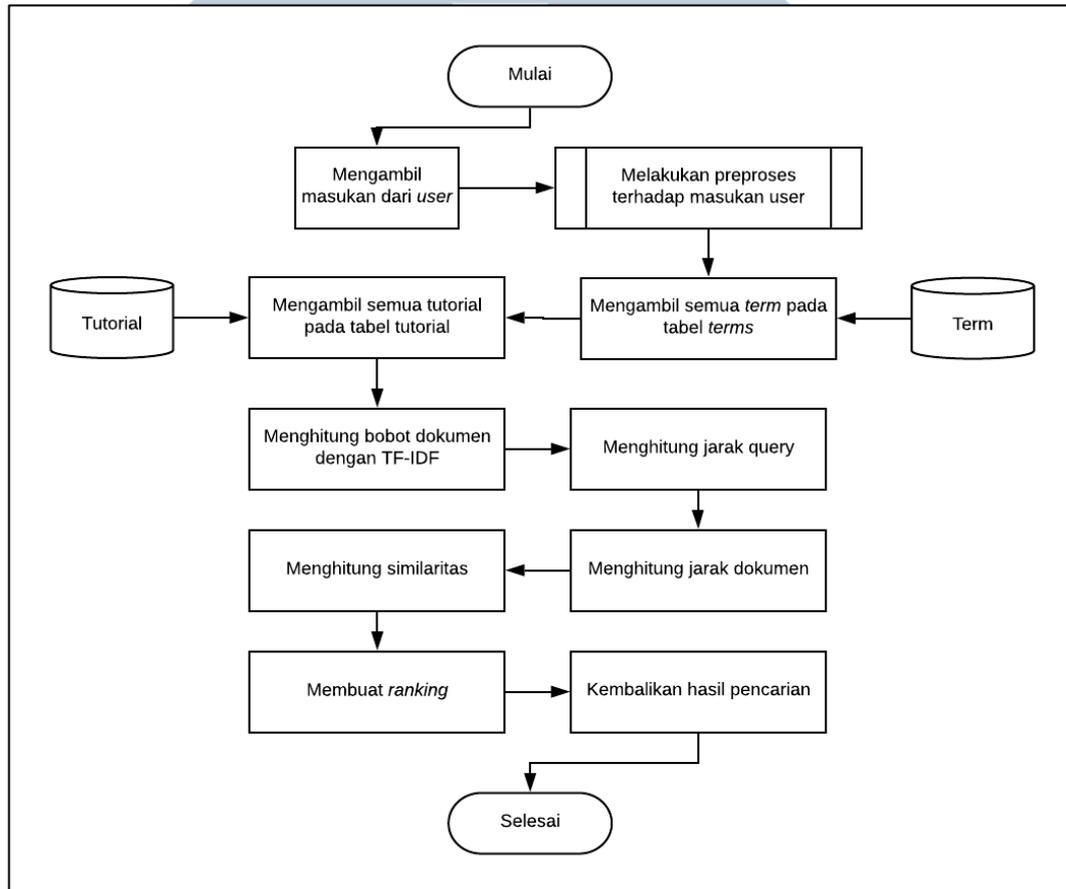
Gambar 3.13 menunjukkan *flowchart* dari Edit Profil. Perubahan pada data identitas *user* yang di-*submit* akan disimpan pada tabel *user*. Kemudian sistem akan menampilkan *feedback* bahwa pembaruan profil berhasil disimpan. Jika pembaruan batal, maka data *user* pada tabel tidak diperbarui.



Gambar 3.14 Flowchart Beranda

Halaman Beranda digunakan untuk melakukan pencarian tutorial. *Query* yang dimasukkan oleh *user* diproses pada proses pencarian tutorial. Jika tutorial ditemukan, maka sistem akan menampilkan daftar tutorial yang dianggap cocok dengan *query* yang dimasukkan dan *user* bisa melihat detil dari tutorial yang dipilih dengan membuka salah satu tutorial yang dikembalikan oleh sistem. Adapun jika tidak ada tutorial yang dianggap cocok dengan *query* yang dimasukkan *user*, maka

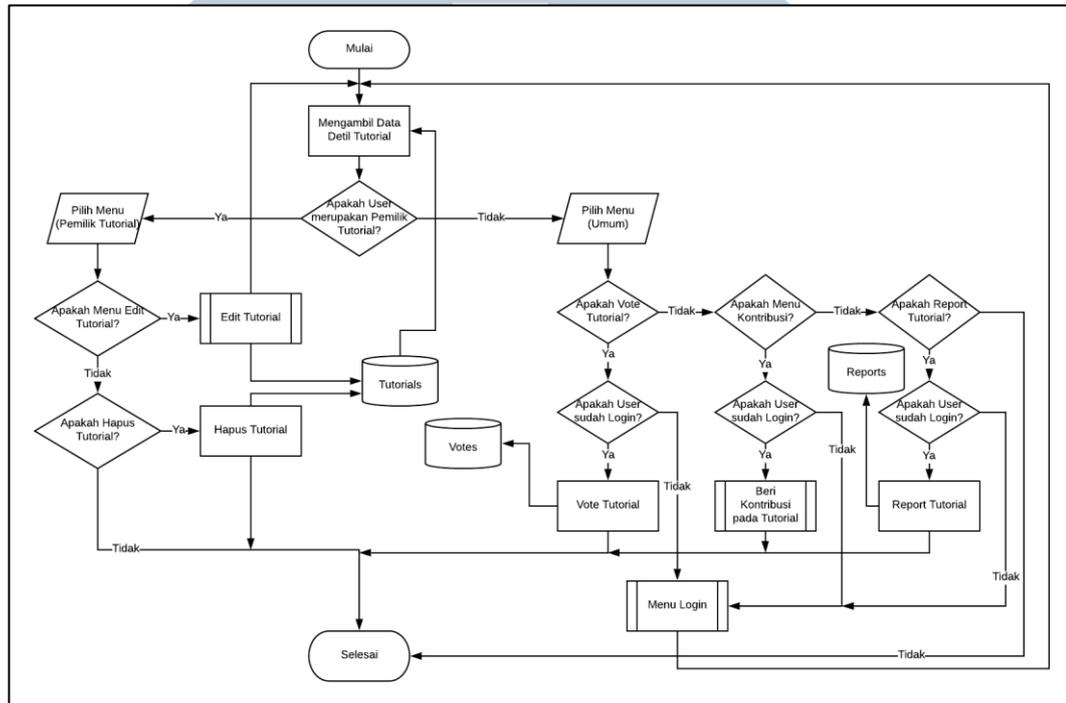
sistem akan menampilkan *feedback* bahwa tutorial tidak ditemukan. Flowchart dari Menu Beranda ditunjukkan pada Gambar 3.14.



Gambar 3.15 Flowchart Pencarian Tutorial

Gambar 3.15 menunjukkan *flowchart* dari proses pencarian tutorial. *Query* yang dimasukkan *user* diambil dan dilakukan proses *preprocessing*. Kemudian, sistem mengambil data *term* dan data tutorial yang terdapat pada masing-masing tabel. Setelah itu, dilakukan pembobotan dengan TF-IDF untuk setiap kata pada *query* serta tutorial dan dilakukan perhitungan jarak untuk *query* dan dokumen yang berupa tutorial tersebut. Kemudian, dilakukan perhitungan tingkat similaritas dengan *cosine similarity* untuk mengetahui tingkat kecocokan antara *query* dengan tutorial yang ada pada tabel dan dilakukan *pe-ranking-an* berdasarkan tutorial yang

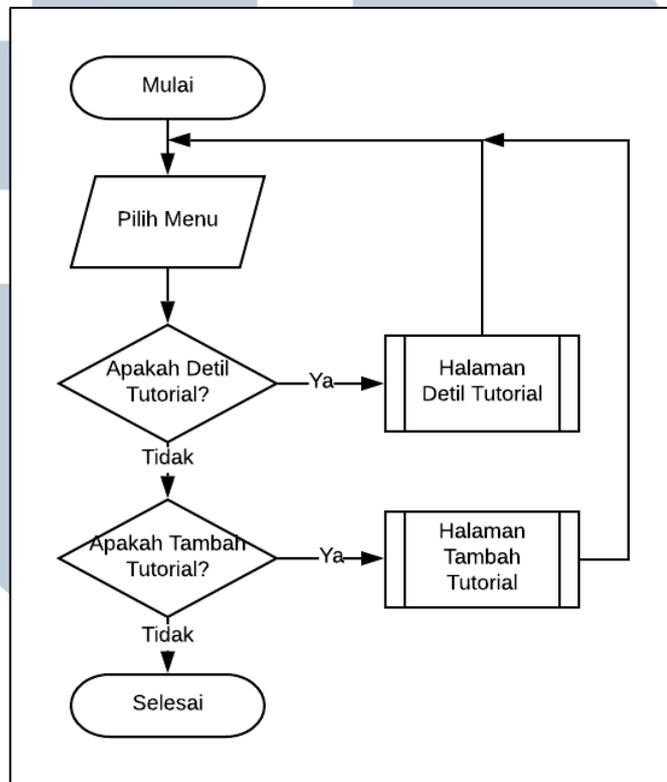
memiliki kecocokan. Tutorial-tutorial yang memiliki kecocokan dengan *query* akan ditampilkan pada halaman pencarian.



Gambar 3.16 Flowchart Detil Tutorial

*Flowchart* yang menjelaskan alur program yang terjadi pada halaman detil tutorial ditunjukkan pada Gambar 3.16. Halaman tersebut menampilkan detil berupa konten dari tutorial yang dipilih serta tindakan yang dapat dilakukan pada tutorial tersebut. Tindakan lebih jauh pada halaman detil tutorial hanya dapat dilakukan jika *user* sudah *login*. Jika detil tutorial tersebut dibuka oleh pemilik tutorial, maka tindakan yang bisa dilakukan yaitu Edit Tutorial untuk melakukan pembaruan terhadap tutorial dan Hapus Tutorial untuk menghapus tutorial dari tabel tutorial. Adapun jika detil tutorial tersebut dibuka oleh *user* lain yang telah melakukan *login*, maka tindakan yang bisa dilakukan yaitu melakukan *vote* pada tutorial jika *user* tersebut merasa terbantu dengan tutorial yang dibuka. *User* bisa

melakukan *report* pada tutorial jika konten dari tutorial tersebut tidak sesuai dengan ketentuan pada aplikasi. Jika tindakan yang dipilih adalah Kontribusi, maka akan ditampilkan *form* pada Halaman Kontribusi.

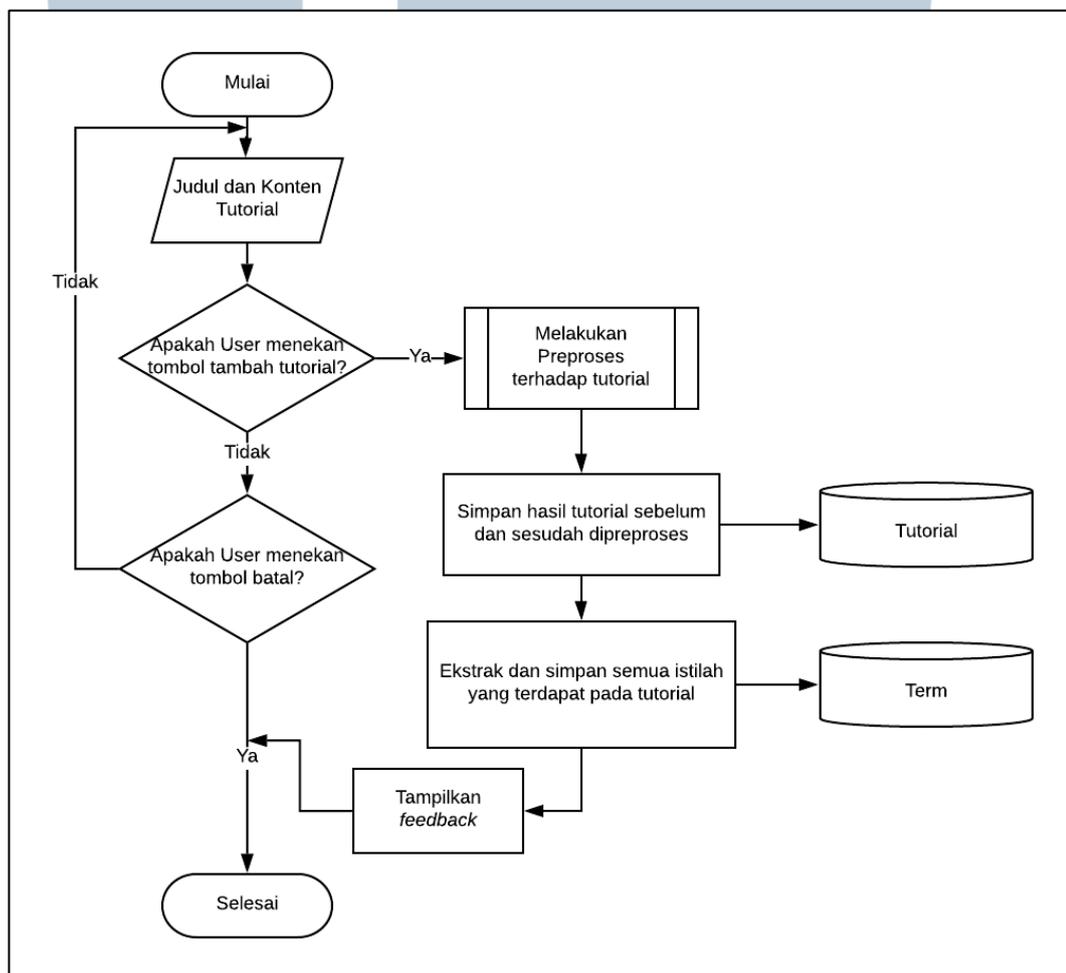


Gambar 3.17 Flowchart Manajemen Tutorial

Gambar 3.17 menunjukkan *flowchart* dari Manajemen Tutorial. Adapun manajemen tutorial yang ditunjukkan pada Gambar 3.17 menunjukkan tindakan yang dapat dilakukan oleh pemilik tutorial setelah melakukan *login*. Jika menu Detil Tutorial yang dipilih, maka sistem akan menampilkan halaman detil tutorial seperti yang ditunjukkan pada Gambar 3.16. Adapun jika menu Tambah Tutorial yang dipilih, maka akan ditampilkan *form* untuk melakukan pembuatan tutorial.

Pada *flowchart* Tambah Tutorial seperti yang dapat dilihat pada Gambar 3.18, dilakukan input terhadap data tutorial seperti judul dan konten tutorial oleh *user*.

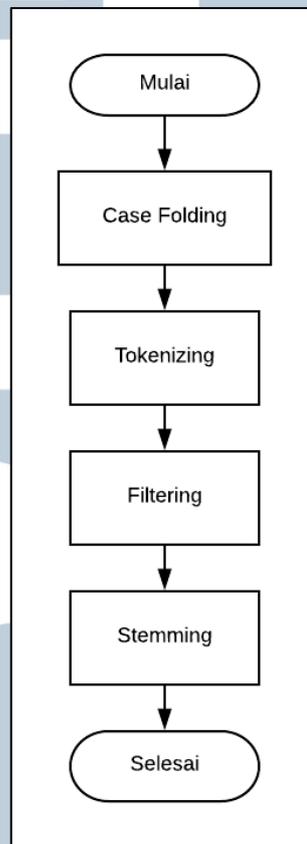
Jika data tutorial berupa judul dan konten tutorial telah terisi dan *user* menekan tombol Tambah Tutorial, maka data tutorial tersebut dipraproses terlebih dahulu. Data tutorial yang telah dipraproses dan belum dipraproses disimpan ke tabel Tutorial. Kemudian, data tutorial yang telah dipraproses diekstrak dan setiap kata pada data tutorial tersebut dimasukkan ke tabel Term. Setelah proses penyimpanan tutorial selesai, sistem menampilkan *feedback* untuk memberitahu *user* bahwa tutorial yang dibuat telah tersimpan.



Gambar 3.18 Flowchart Tambah Tutorial

Gambar 3.19 menunjukkan *flowchart* dari Praproses Tutorial. Sebelum data tutorial disimpan ke dalam tabel tutorial, data tersebut diubah menjadi huruf kecil

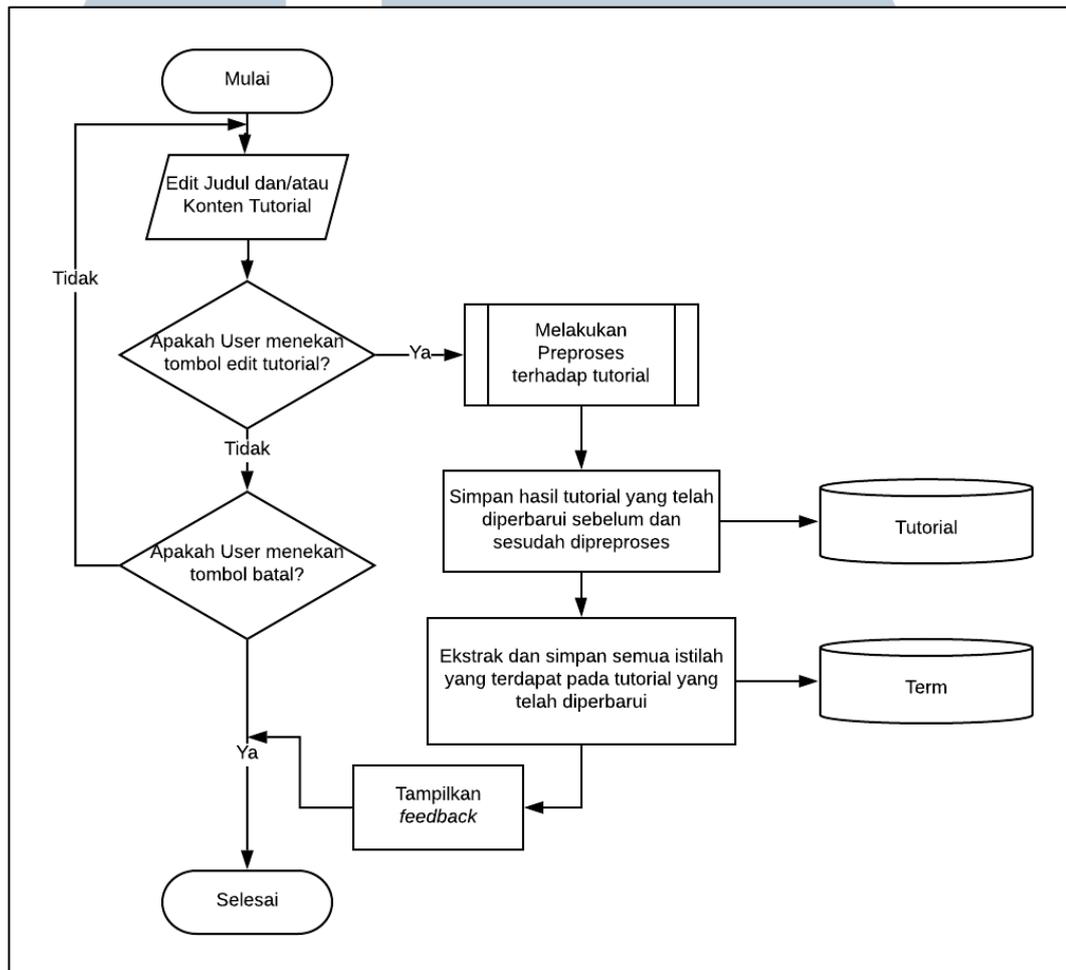
dengan melakukan proses *case folding*, setelah itu dilakukan proses tokenizing untuk menghilangkan tanda baca serta memecah atau meng-*explode* data tutorial menjadi kumpulan kata-kata. Tiap kata yang ada pada data tutorial difilter untuk menghilangkan kata-kata yang dianggap tidak penting. Setelah itu, kata-kata pada data tutorial tersebut dikembalikan menjadi kata dasar dengan proses *stemming*. Proses Filtering dan Stemming dilakukan oleh *library* Sastrawi.



Gambar 3.19 Flowchart Praproses Tutorial

Adapun Gambar 3.20 menunjukkan *flowchart* untuk Edit Tutorial. Proses ini diawali dengan mengambil data tutorial yang dipilih untuk diperbarui dari tabel tutorial. Kemudian, *user* memperbarui tutorial dengan melakukan perubahan pada judul atau konten tutorial. Setelah itu, *user* melakukan konfirmasi data tutorial yang diperbarui dengan menekan tombol Edit Tutorial. Sebelum tutorial tersebut

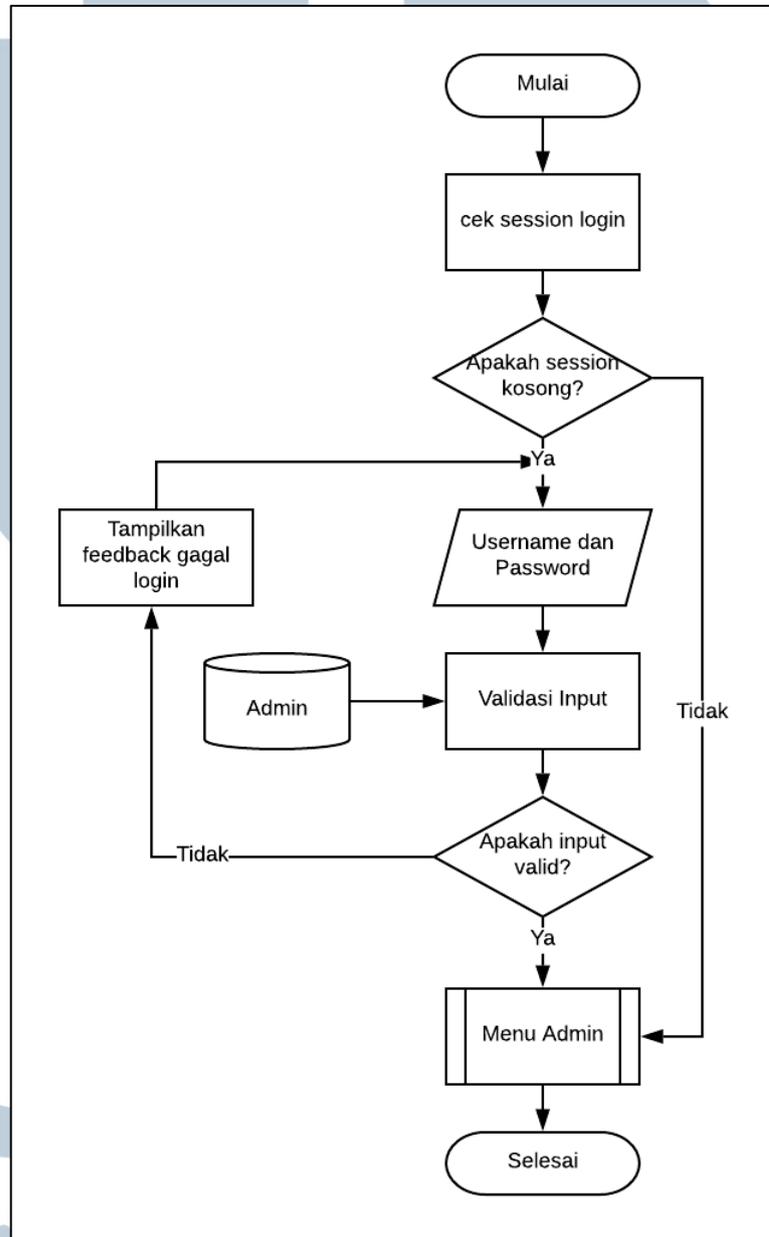
diperbarui, data tutorial tersebut melewati beberapa proses seperti melakukan praproses terhadap tutorial, memperbaiki tutorial dengan data sebelum dan sesudah dipraproses, mengekstrak data tutorial menjadi sekumpulan kata. Setelah itu, sistem kembali menampilkan *feedback* terhadap perbaruan data tutorial.



Gambar 3.20 Flowchart Edit Tutorial

Gambar 3.21 menunjukkan *flowchart* untuk masuk ke menu Admin. Dimulai dengan pengecekan *session*. Jika *session* kosong, maka akan ditampilkan halaman Login Admin. Pada halaman tersebut admin melakukan input data *login* seperti *username* dan *password*. Setelah itu, data *login admin* akan divalidasi dengan cara membandingkan data *login* dari *admin* dengan data *admin* yang ada pada tabel

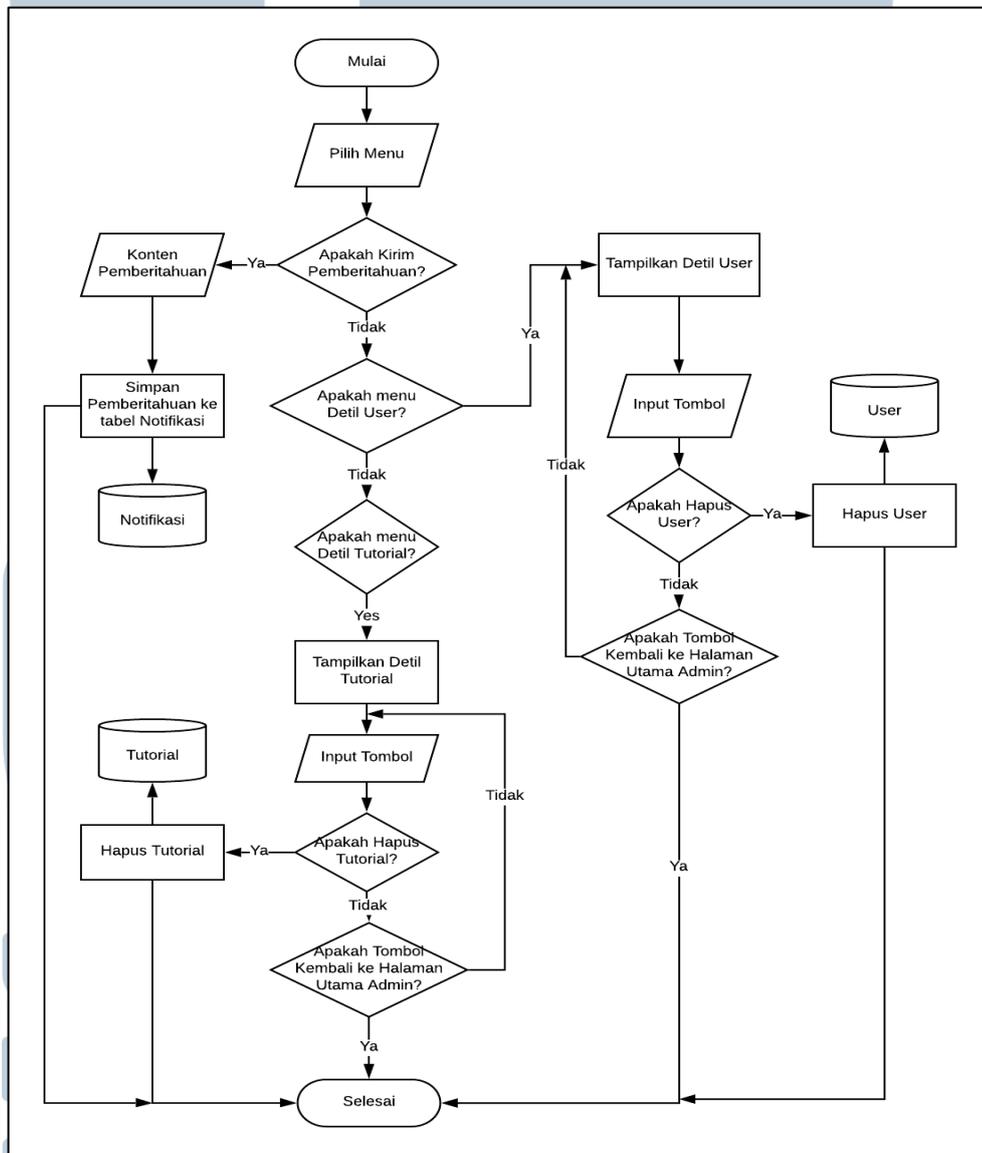
Admin di *database*. Jika data *login* tersebut valid, maka akan ditampilkan Menu Admin. Adapun jika input tidak valid, maka halaman yang ditampilkan tetap halaman Login.



Gambar 3.21 Flowchart Login Admin

Gambar 3.22 menunjukkan *flowchart* dari Menu Admin. Pada menu tersebut, *admin* memiliki beberapa pilihan menu seperti Kirim Pemberitahuan, Lihat Detil

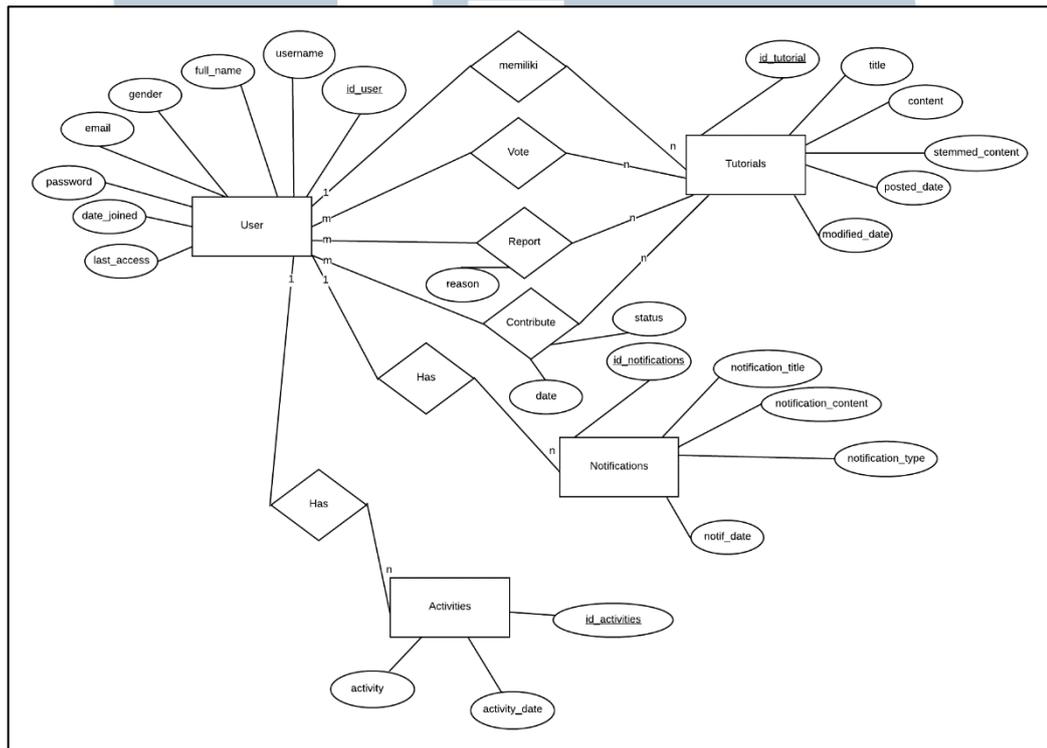
User, dan Lihat Detil Tutorial. Jika menu Kirim Pemberitahuan dipilih, maka *admin* diminta memasukkan input berupa konten pemberitahuan untuk disampaikan kepada *user* melalui notifikasi. Data pemberitahuan tersebut disimpan ke dalam tabel notifikasi yang terdapat pada *database*. *Admin* bisa melihat semua profil *user* yang telah terdaftar pada aplikasi dan bisa menghapus *user* yang dipilih oleh *admin*. Adapun pada halaman utama terdapat daftar tutorial dan *admin* bisa melihat detail tutorial serta menghapus tutorial tersebut.



Gambar 3.22 Flowchart Menu Admin

### 3.2.3 Entity Relationship Diagram

Dalam melakukan perancangan *database*, tahap pertama yang perlu dilakukan yaitu merancang *Entity Relationship Diagram (ERD)* dengan tujuan mengetahui hubungan antar entitas dan atribut-atribut yang dimiliki oleh setiap entitas. Gambar 3.23 menunjukkan ERD dari aplikasi repositori tutorial.



Gambar 3.23 Entity Relationship Diagram

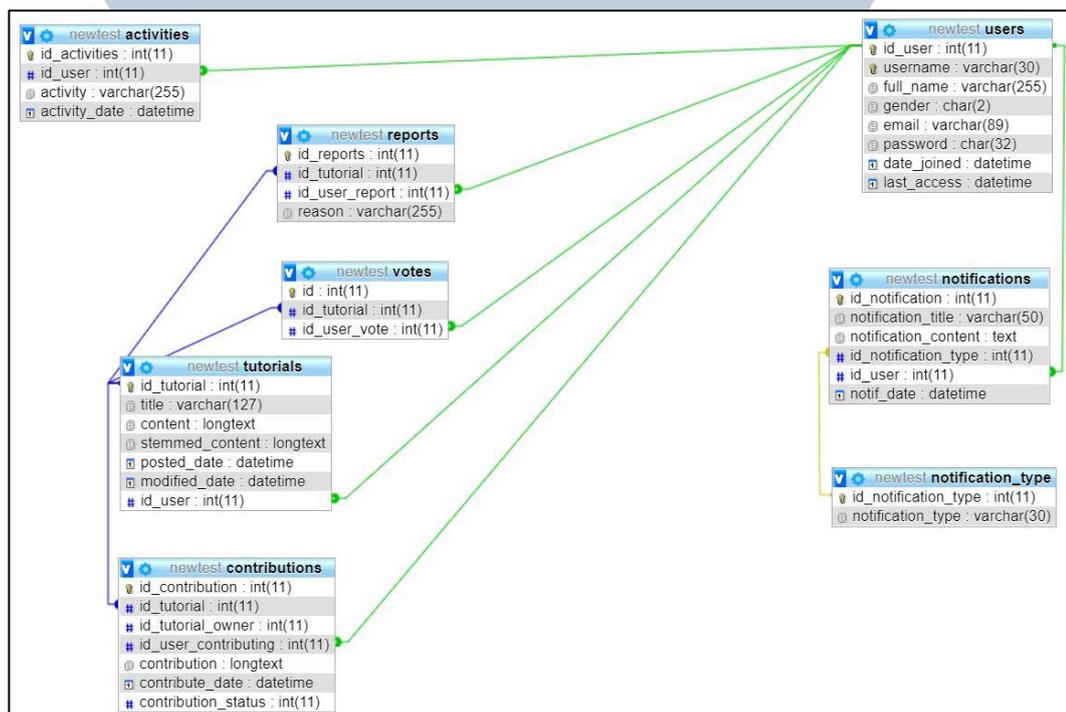
Pada Gambar 3.23 dapat diketahui bahwa entitas *user* memiliki atribut *id\_user*, *username*, *full\_name*, *gender*, *email*, *password*, *date\_joined*, dan *last\_access*.

Adapun atribut *id\_user* merupakan *primary key* pada entitas *User*. Pada Entitas *Activities* terdapat atribut *id\_activities* dimana atribut tersebut sebagai *primary key*, *activity*, dan *activity\_date*. Pada entitas *Notifications* terdapat atribut *id\_notifications*, *notification\_date*, *notification\_content*, *notification\_type*, serta *notification\_date*. *Primary key* pada entitas *Notifications* adalah *id\_notification*.

Pada entitas Tutorial terdapat atribut `id_tutorial` sebagai *primary key*. Atribut lain yang terdapat pada entitas Tutorial adalah `title`, `content`, `stemmed_content`, `posted_date`, dan `modified_date`. Ketika User melakukan report terhadap tutorial, akan dicatat alasan *user* melakukan *report*. Adapun ketika User melakukan kontribusi kepada tutorial, akan dicatat tanggal dan status dari kontribusi tersebut.

### 3.2.4 Database Schema

Setelah melakukan perancangan ERD, dilakukan perancangan skema *database* untuk mengetahui hubungan antar kolom pada suatu tabel dengan kolom pada tabel lain.



Gambar 3.24 Database Schema

Hubungan tersebut menunjukkan ketergantungan antara satu tabel dengan tabel lainnya. Tabel-tabel tersebut adalah tabel `users`, `activities`, `contributions`, `notifications`, `notifications_type`, `reports`, `tutorials`, dan `votes`. tabel `users` memiliki

hubungan dengan tabel activities, reports, votes, tutorials, contributions dan notifications. Tabel tutorials memiliki hubungan dengan tabel reports, votes, dan contributions. Tabel notifications memiliki hubungan dengan tabel notification\_type. Skema Database dapat dilihat pada Gambar 3.24.

### 3.2.5 Struktur Tabel

Database yang digunakan pada penelitian ini yaitu MySQL. Database yang dibuat memiliki sebelas tabel, diantaranya yaitu tabel activities, admin\_repo, broadcasts, contributions, notifications, notification\_type, reports, terms, tutorials, users, dan votes. Adapun nama tabel, fungsi serta deskripsi tabel adalah sebagai berikut.

1. Nama Tabel : activities  
 Fungsi : menyimpan riwayat aktivitas yang telah dilakukan oleh user pada aplikasi repositori. Deskripsi tabel ditunjukkan pada Tabel 3.1.

Tabel 3.1 Tabel activities

| Nama Field    | Type Data    | Keterangan                   |
|---------------|--------------|------------------------------|
| id_activities | int(11)      | Primary key                  |
| id_user       | int(11)      | Foreign key dari tabel users |
| activity      | varchar(255) | Keterangan aktivitas user    |
| activity_date | datetime     | Tanggal aktivitas dilakukan  |

2. Nama Tabel : admin\_repo  
 Fungsi : menyimpan data admin. Deskripsi tabel ditunjukkan pada Tabel 3.2.

Tabel 3.2 Tabel admin\_repo

| <b>Nama Field</b> | <b>Tipe Data</b> | <b>Keterangan</b> |
|-------------------|------------------|-------------------|
| id_admin          | int(11)          | Primary key       |
| username          | varchar(30)      | Username admin    |
| email             | Varchar(89)      | Email admin       |
| password          | Char(32)         | Password admin    |

3. Nama Tabel : broadcasts

Fungsi : menyimpan data broadcast yang dibuat oleh admin. Tabel broadcast ditunjukkan pada Tabel 3.3.

Tabel 3.3 Tabel broadcasts

| <b>Nama Field</b> | <b>Tipe Data</b> | <b>Keterangan</b>            |
|-------------------|------------------|------------------------------|
| Id_broadcast      | Int(11)          | Primary key                  |
| Broadcast_title   | Varchar(50)      | Judul broadcast              |
| Broadcast_content | Text             | Konten broadcast             |
| Broadcast_date    | datetime         | Tanggal pengiriman broadcast |

4. Nama Tabel : contributions

Fungsi : menyimpan kontribusi berupa pembaruan yang disarankan oleh *user* lain kepada pemilik tutorial. Tabel contributions ditunjukkan pada Tabel 3.4.

Tabel 3.4 Tabel contributions

| <b>Nama Field</b>    | <b>Tipe Data</b> | <b>Keterangan</b>                                    |
|----------------------|------------------|--|
| Id_contribution      | Int(11)          | Primary key  |
| Id_tutorial          | Int(11)          | Foreign key dari tabel users                         |
| Id_tutorial_owner    | Int(11)          | Foreign key (id_user yang memiliki tutorial)         |
| Id_user_contributing | Int(11)          | Foreign key (user yang menjadi kontributor)          |
| Contribution         | Longtext         | Data saran perbaikan yang dilakukan oleh contributor |

|                     |          |                              |
|---------------------|----------|------------------------------|
| Contribution_date   | Datetime | Tanggal melakukan kontribusi |
| Contribution_status | Int(11)  | Status kontribusi            |

5. Nama Tabel : notifications  
 Fungsi : menyimpan data notifikasi yang berasal dari *admin* maupun *user*. Tabel 3.5 menunjukkan tabel notifications.

Tabel 3.5 Tabel notifications

| Nama Field           | Tipe Data   | Keterangan                               |
|----------------------|-------------|--|
| Id_notification      | Int(11)     | Primary key                              |
| Notification_title   | Varchar(50) | Judul Notifikasi                         |
| Notification_content | Text        | Konten Notifikasi                        |
| Id_notification_type | Int(11)     | Foreign Key dari tabel notification_type |
| Id_user              | Int(11)     | Foreign key dari tabel users             |
| Notif_date           | Datetime    | Tanggal notifikasi dibuat                |

6. Nama Tabel : notification\_type  
 Fungsi : menyimpan jenis notifikasi yang diterima oleh *user*. Tabel notification\_type ditunjukkan pada Tabel 3.6

Tabel 3.6 Tabel notification\_type

| Nama Field           | Tipe Data   | Keterangan      |
|----------------------|-------------|-----------------|
| Id_notification_type | Int(11)     | Primary key     |
| Notification_type    | Varchar(30) | Tipe notifikasi |

7. Nama Tabel : reports  
 Fungsi : menyimpan informasi terkait tutorial yang di-*report* serta *user* yang melakukan report tersebut. Tabel reports ditunjukkan pada Tabel 3.7.

Tabel 3.7 Tabel reports

| Nama Field     | Tipe Data    | Keterangan                       |
|----------------|--------------|----------------------------------|
| Id_reports     | Int(11)      | Primary key                      |
| Id_tutorial    | Int(11)      | Foreign key dari tabel tutorials |
| Id_user_report | Int(11)      | Foreign key dari tabel users     |
| reason         | Varchar(255) | Alasan melakukan report tutorial |

8. Nama Tabel : terms

Fungsi : menyimpan istilah atau *term* yang terdapat pada data tutorial. Tabel 3.8 menunjukkan tabel untuk menyimpan data *terms*.

Tabel 3.8 Tabel terms

| Nama Field | Tipe Data    | Keterangan                       |
|------------|--------------|----------------------------------|
| Id         | Int(11)      | Primary Key                      |
| Term       | Varchar(100) | Term yang terdapat pada tutorial |

9. Nama Tabel : tutorials

Fungsi : menyimpan data tutorial yang telah dibuat oleh *user* dan digunakan sebagai bahan pencarian. Tabel 3.9 menunjukkan detail dari Tabel tutorials.

Tabel 3.9 Tabel tutorials

| Nama Field      | Tipe Data    | Keterangan                        |
|-----------------|--------------|-----------------------------------|
| Id_tutorial     | Int(11)      | Primary key                       |
| Title           | Varchar(127) | Judul Tutorial                    |
| Content         | Longtext     | Konten yang masih <i>raw</i>      |
| Stemmed_content | Longtext     | Konten yang telah di- <i>stem</i> |
| Posted_date     | Datetime     | Tanggal tutorial di- <i>post</i>  |
| Modified_date   | Datetime     | Tanggal tutorial diperbarui       |
| Id_user         | Int(11)      | Foreign Key dari tabel users      |

10. Nama Tabel : users

Fungsi : menyimpan informasi terkait *user* yang telah terdaftar pada aplikasi repositori tutorial. Tabel users ditunjukkan pada Tabel 3.10

Tabel 3.10 Tabel users

| Nama Field  | Type Data    | Keterangan   |
|-------------|--------------|--|
| Id_user     | Int(11)      | Primary key  |
| Username    | Varchar(30)  | Nama pengguna  |
| Full_name   | Varchar(255) | Nama lengkap pengguna  |
| Gender      | Char(2)      | <i>Gender user</i>   |
| Email       | Varchar(89)  | Email user   |
| Password    | Char(32)     | Password user  |
| Date_joined | datetime     | Tanggal user bergabung   |
| Last_access | datetime     | Tanggal terakhir kali <i>user</i> melakukan <i>login</i> pada aplikasi |

11. Nama Tabel :votes

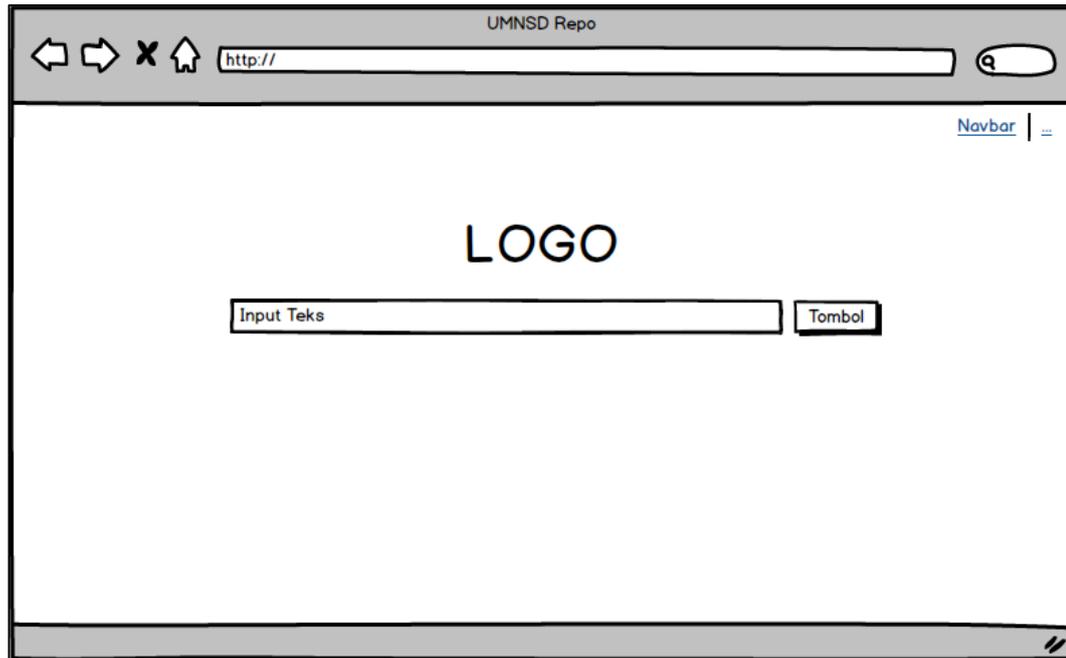
Fungsi : menyimpan informasi terkait tutorial yang telah di-*vote* serta *user* yang melakukan *vote* pada tutorial tersebut. Tabel votes ditunjukkan pada Tabel 3.11.

Tabel 3.11 Tabel votes

| Nama Field   | Type Data | Keterangan                       |
|--------------|-----------|----------------------------------|
| Id           | Int(11)   | Primary key                      |
| Id_tutorial  | Int(11)   | Foreign key dari tabel tutorials |
| Id_user_vote | Int(11)   | Foreign key dari tabel users     |

### 3.2.6 Rancangan Tampilan Antarmuka

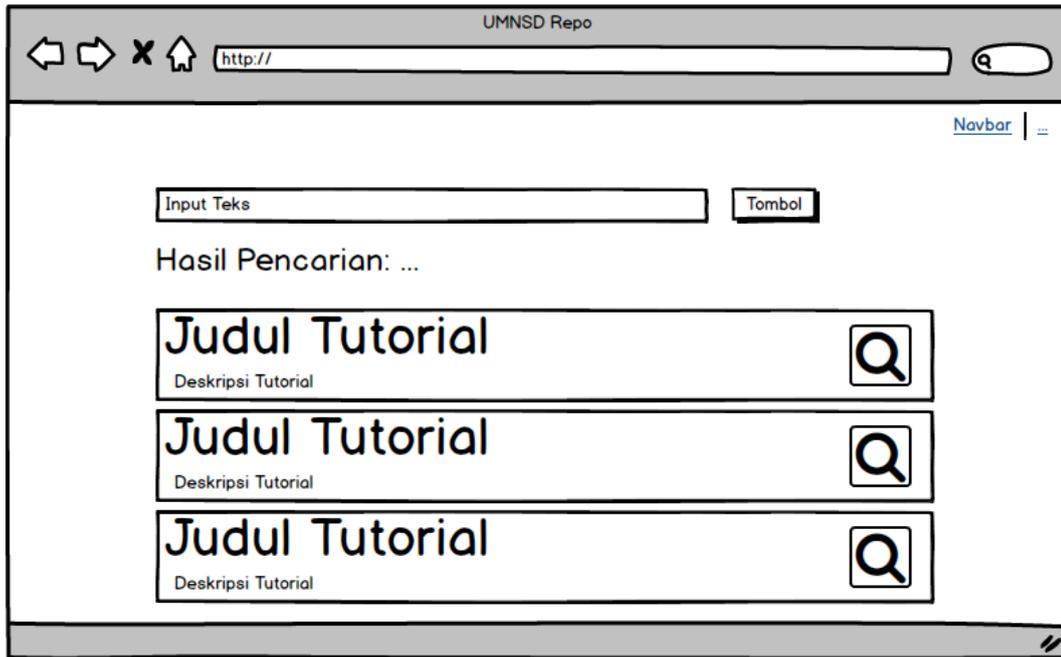
Perancangan tampilan antarmuka dilakukan dengan membuat sketsa yang menggambarkan seluruh tampilan aplikasi repositori tutorial ini. Berikut rancangan tampilan antarmuka pada aplikasi repositori.



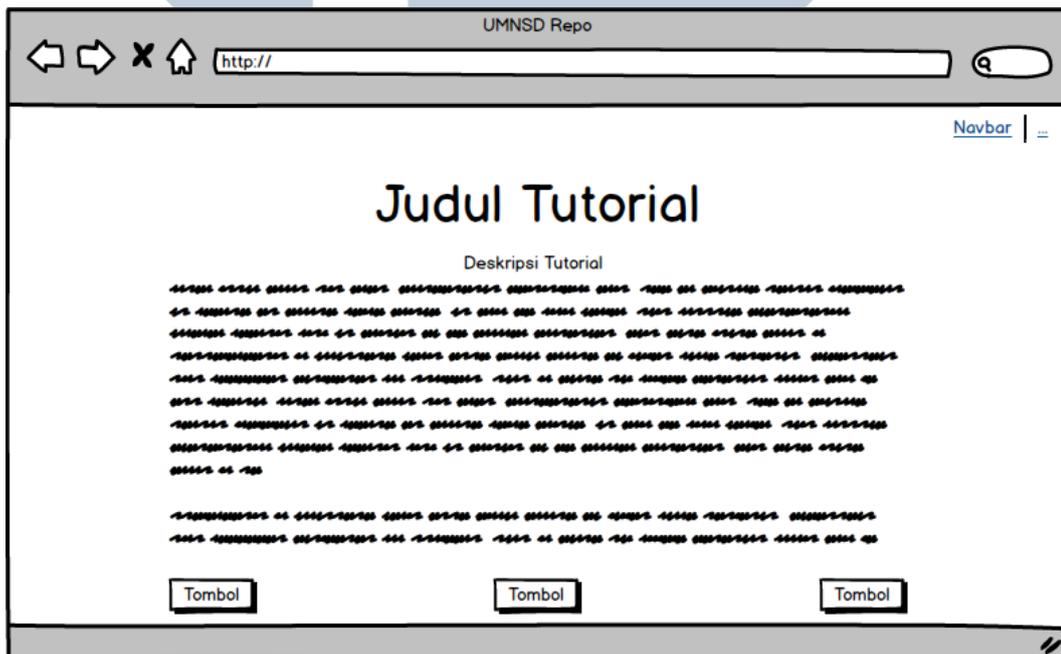
Gambar 3.25 Halaman Utama

Gambar 3.25 menunjukkan halaman utama dari aplikasi repositori tutorial. Pada halaman tersebut terdapat navbar pada bagian kanan atas. Navbar tersebut akan ada pada setiap halaman. Adapun pada bagian tengah terdapat sebuah logo aplikasi, input teks untuk memasukkan *query* terkait pencarian tutorial, serta tombol untuk memroses pencarian.

Gambar 3.26 menunjukkan halaman yang menampilkan hasil pencarian setelah *user* melakukan *query* pada halaman utama. Pada halaman hasil pencarian, disediakan kolom *input* serta tombol dengan tujuan memudahkan *user* untuk kembali melakukan pencarian tutorial tanpa harus kembali ke halaman utama. Di bawah kolom *input* terdapat daftar hasil tutorial yang direpresentasikan dalam bentuk *card*. Masing-masing *card* terdiri dari judul tutorial, deskripsi tutorial, serta tombol untuk membuka detail tutorial.



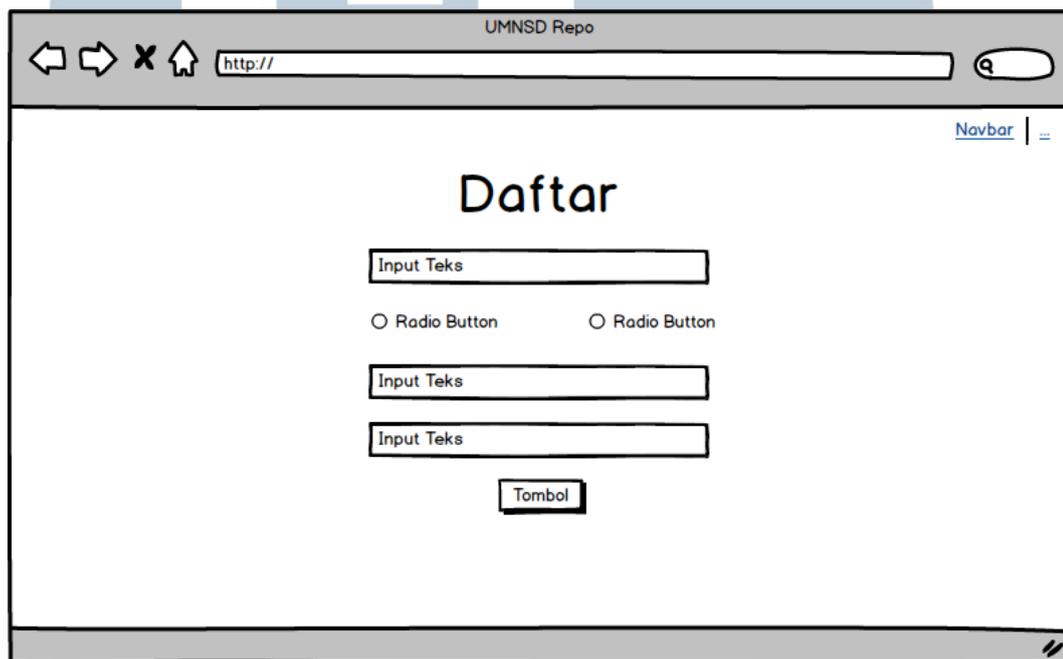
Gambar 3.26 Halaman Hasil Pencarian



Gambar 3.27 Halaman Detil Tutorial

Pada halaman detil tutorial seperti yang ditunjukkan pada Gambar 3.27, terdapat judul tutorial, deskripsi tutorial, konten tutorial, serta beberapa tombol

untuk melakukan tindakan pada tutorial tersebut. Pengguna aplikasi yang melihat tutorial akan dibedakan menjadi tiga untuk menentukan tombol yang ditampilkan serta tindakan yang dilakukan terhadap tutorial. Adapun pengguna tersebut yaitu *admin*, *user* sebagai pemilik tutorial, dan *user* general. Untuk melakukan tindakan terhadap tutorial, *user* harus melakukan *login*. Adapun jika *user* belum terdaftar, *user* diminta melakukan *register* atau daftar pada aplikasi TERESTRIAL ini.



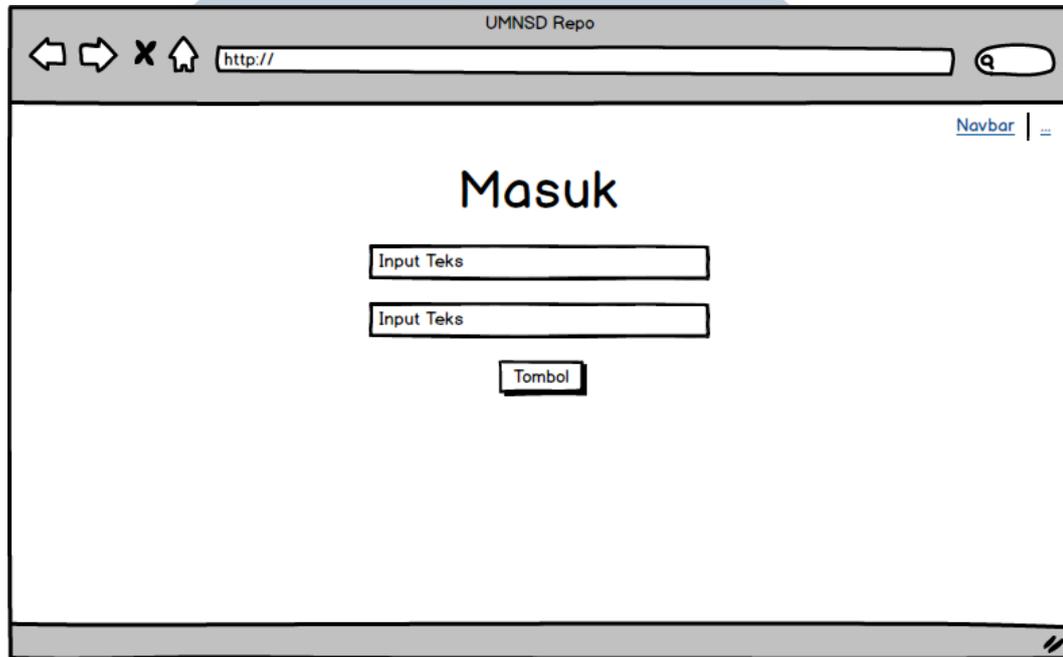
The image shows a web browser window with the title 'UMNSD Repo'. The address bar contains 'http://'. The main content area displays a registration form titled 'Daftar'. The form includes the following elements: a text input field labeled 'Input Teks', two radio buttons, another text input field labeled 'Input Teks', a third text input field labeled 'Input Teks', and a button labeled 'Tombol'. A 'Navbar' link is visible in the top right corner of the page content.

Gambar 3.28 Halaman Mendaftar sebagai User

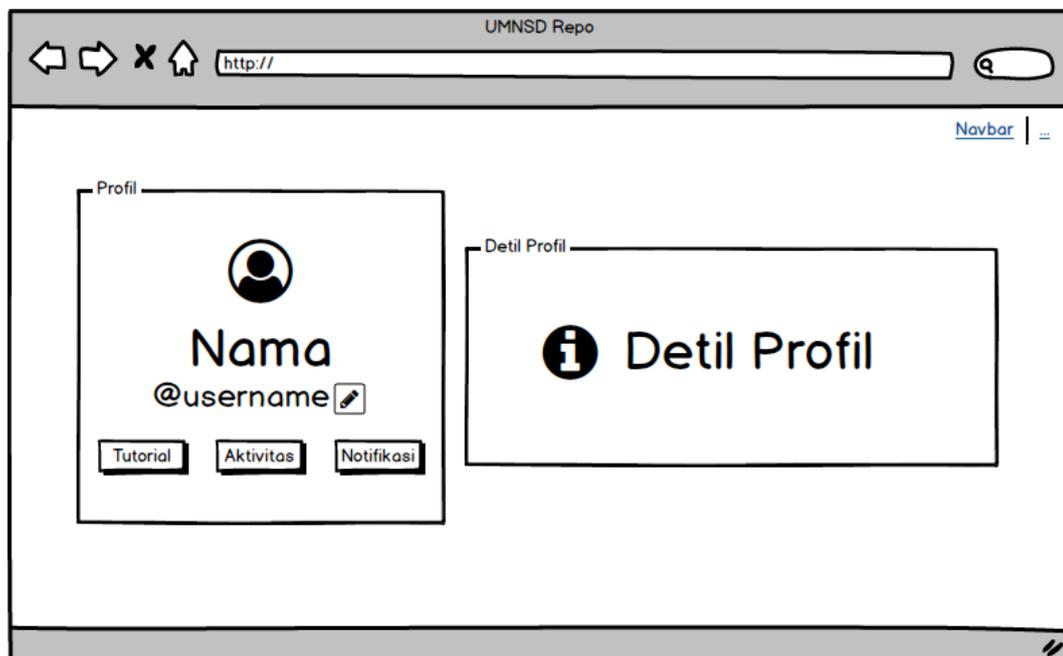
Gambar 3.28 menunjukkan halaman untuk mendaftar sebagai *user*. Pada halaman tersebut terdapat *form* untuk melakukan pendaftaran. *Form* tersebut terdiri dari beberapa kolom *input*, *radio button*, serta tombol untuk melakukan konfirmasi pendaftaran.

Adapun Gambar 3.29 menunjukkan halaman untuk melakukan *login* pada aplikasi TERESTRIAL. Pada halaman tersebut terdapat *form* untuk melakukan

login. Terdapat dua buah kolom input serta satu tombol untuk melakukan proses login.

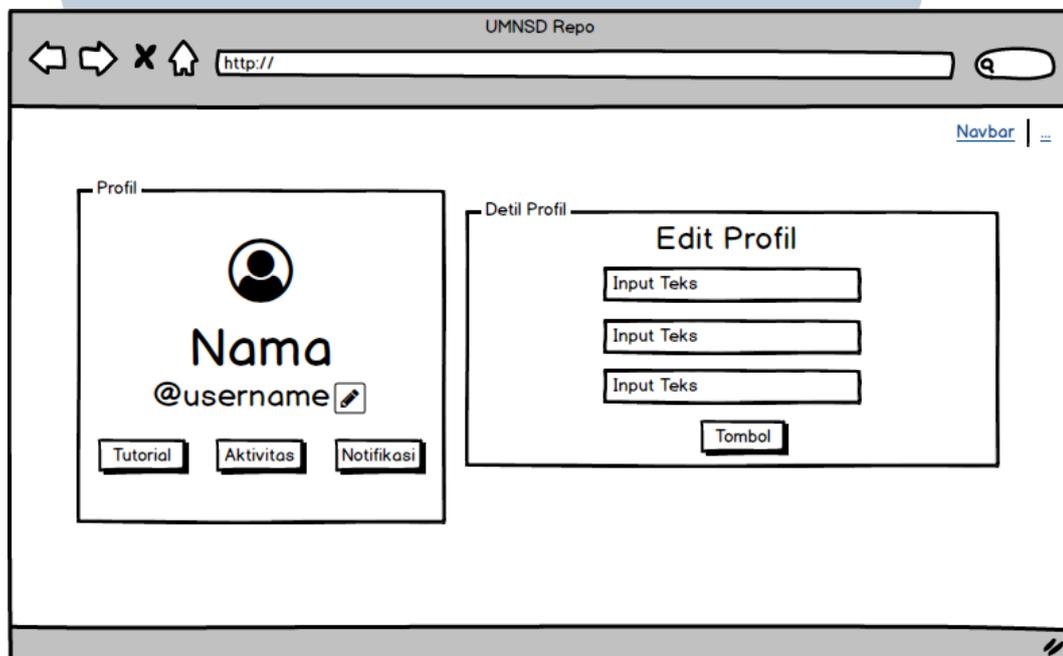


Gambar 3.29 Halaman Login



Gambar 3.30 Halaman Profil User

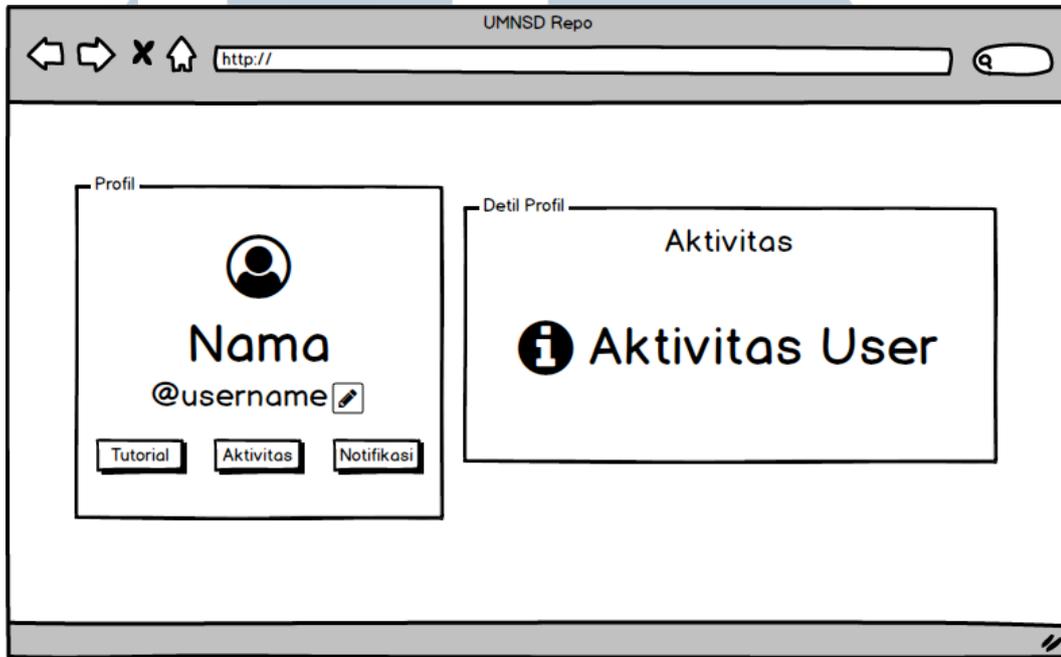
Gambar 3.30 menunjukkan halaman profil user. Pada halaman tersebut, terdapat dua buah *card* dimana *card* pada bagian kiri yaitu *card* untuk menampilkan identitas *user* seperti nama dan *username* dari *user* tersebut serta beberapa tombol untuk menampilkan detail dari *user* seperti tombol untuk menampilkan daftar tutorial, tombol untuk menampilkan riwayat aktivitas, tombol untuk melakukan *edit* profil, dan tombol untuk menampilkan notifikasi. *Feedback* dari tombol untuk menampilkan detail *user* tersebut ditampilkan pada *card* bagian kanan, sejajar dengan *card* untuk menampilkan identitas *user*.



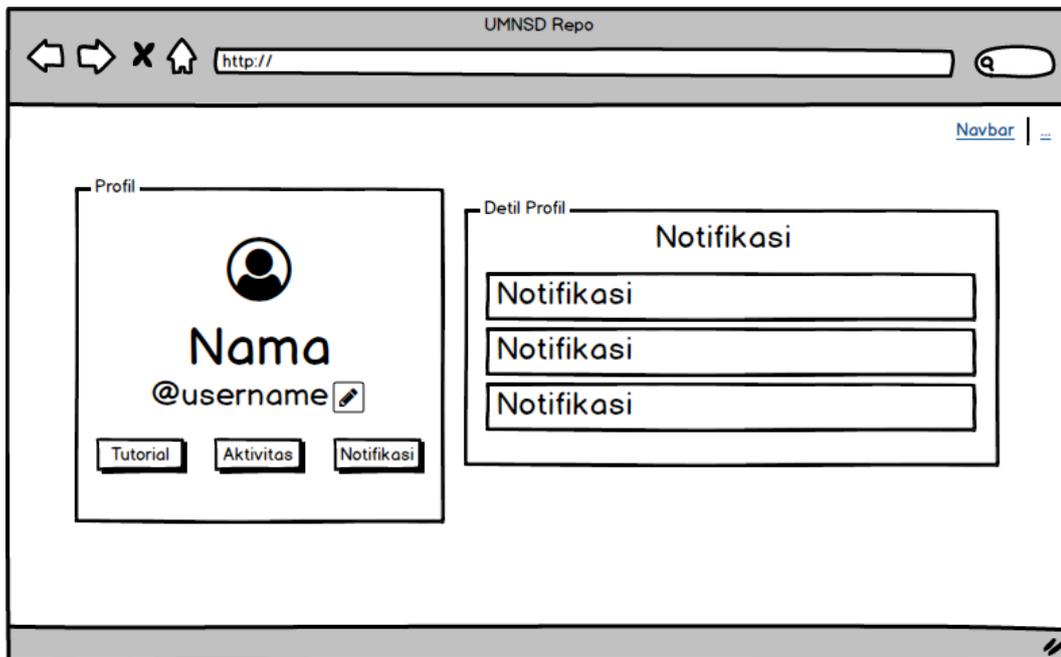
Gambar 3.31 Tampilan Edit Profil

Gambar 3.31 menunjukkan *card* detail *user* untuk melakukan edit profil. Tampilan Edit Profil merupakan *form* yang terdiri dari beberapa kolom input yang telah terisi dan terdapat tombol untuk melakukan konfirmasi terhadap *edit* yang dilakukan oleh *user* tersebut.

Gambar 3.32 menunjukkan *card* detail *user* untuk menampilkan riwayat aktivitas *user*. Tampilan pada *card* tersebut hanya berisi riwayat aktivitas yang telah dilakukan oleh *user*.

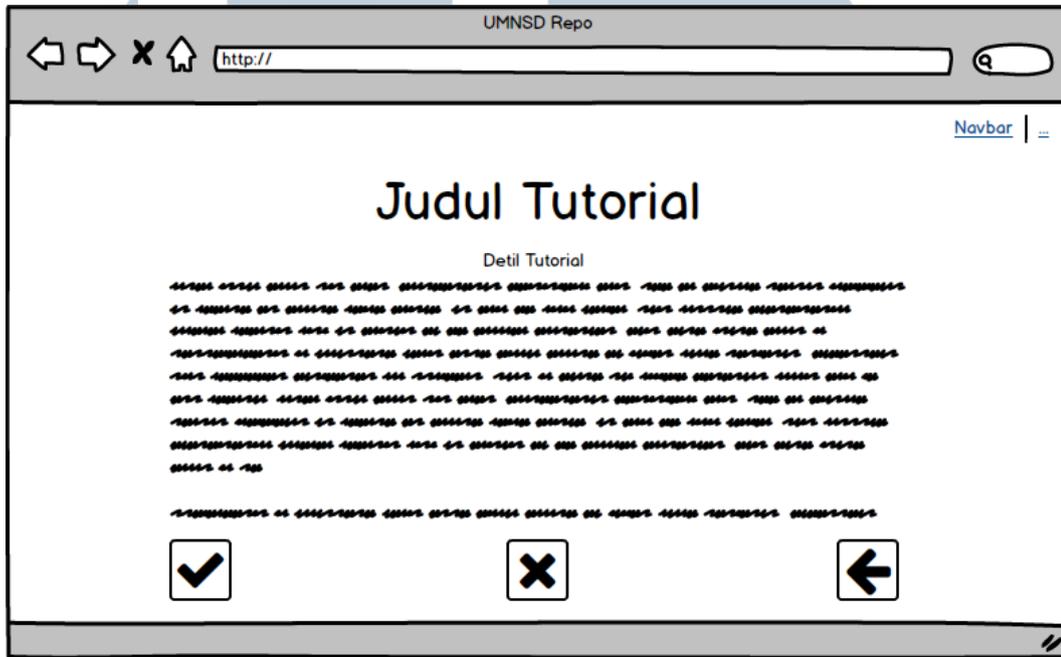


Gambar 3.32 Tampilan Aktivitas User



Gambar 3.33 Tampilan Notifikasi

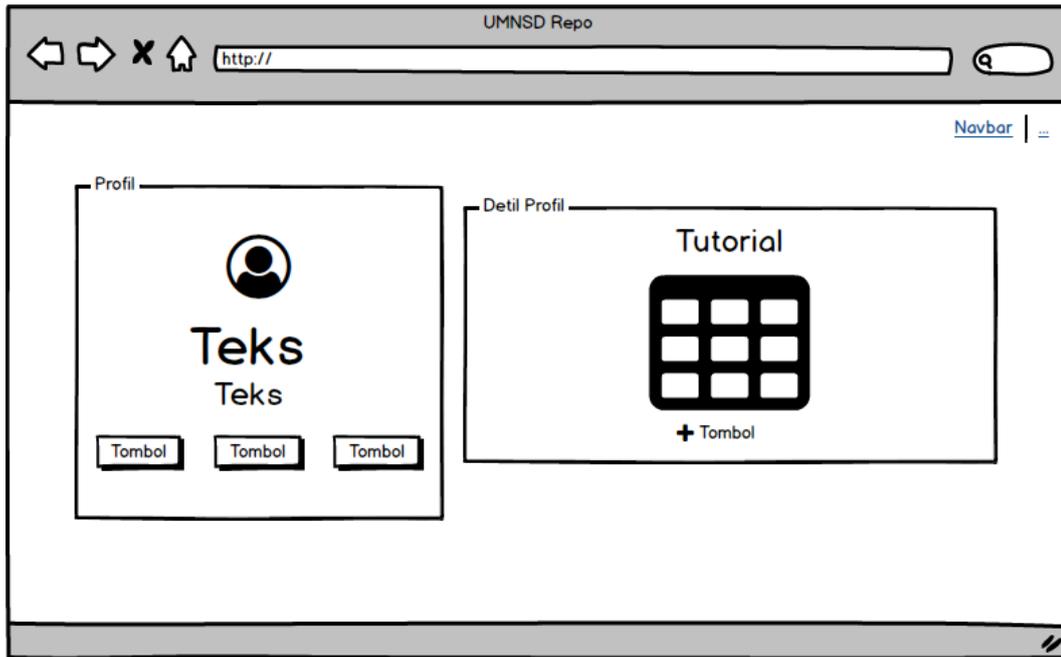
Gambar 3.33 menunjukkan *card* detail *user* untuk menampilkan notifikasi. Notifikasi bisa berupa informasi dari *admin*, dan juga informasi terkait kontribusi atau saran pembaruan dari *user*.



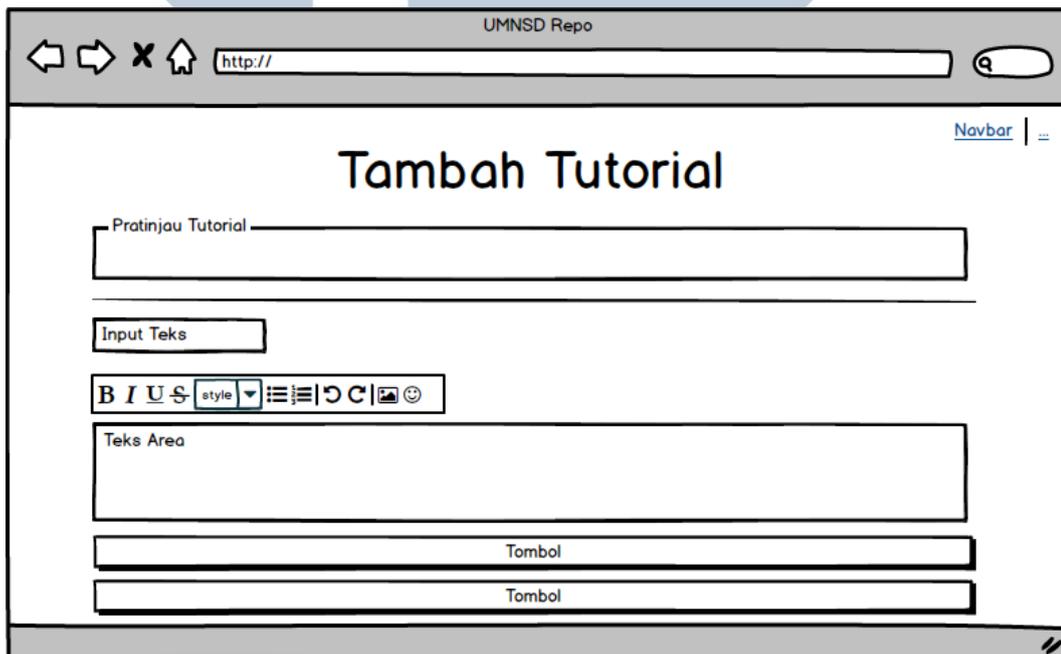
Gambar 3.34 Halaman Detil Kotribusi

Gambar 3.34 menunjukkan halaman detail kontribusi, halaman tersebut memuat detail dari saran perbaikan yang ditawarkan oleh *user* lain. Serta tombol untuk menerima kontribusi, menolak kontribusi, dan kembali ke halaman sebelumnya.

Tampilan *card* detail *user* untuk tindakan terhadap tombol Tutorial ditunjukkan pada Gambar 3.35. Pada *card* tersebut terdapat tabel daftar tutorial milik *user* tersebut. Jika profil tersebut milik *user* yang sedang login, maka terdapat tombol untuk menambahkan tutorial tepat dibawah tabel daftar tutorial.



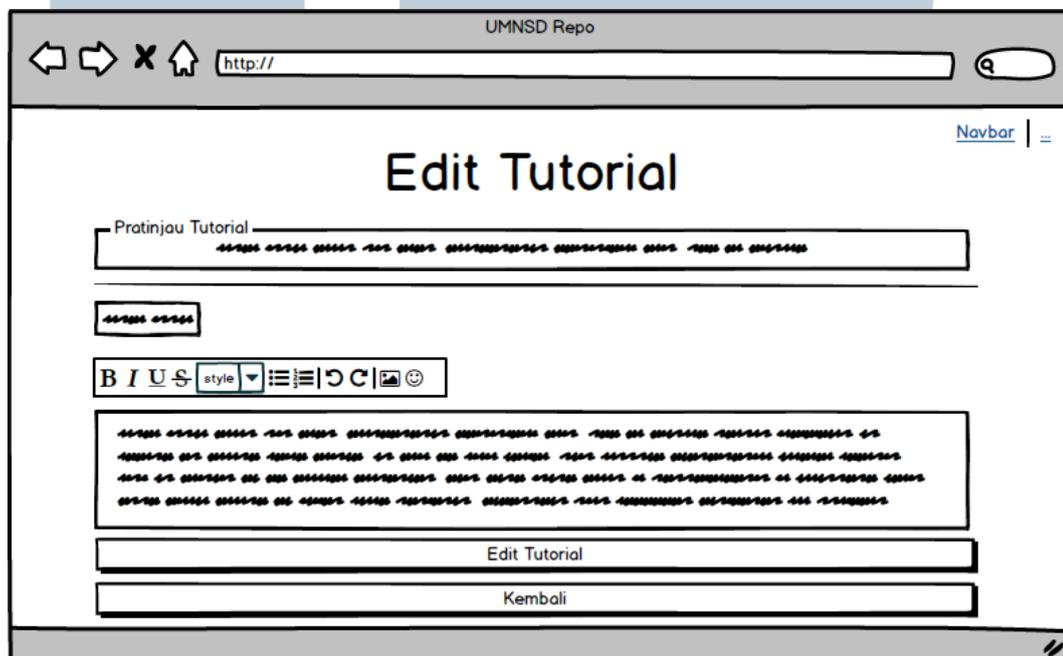
Gambar 3.35 Tampilan Daftar Tutorial



Gambar 3.36 Halaman Tambah Tutorial

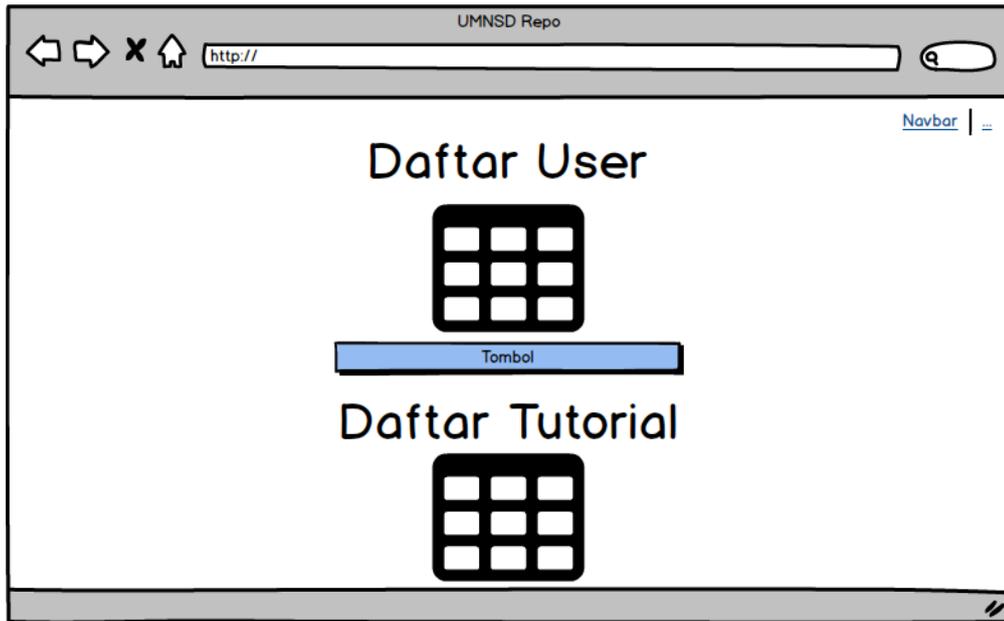
Gambar 3.36 menunjukkan halaman untuk melakukan penambahan tutorial. Menu Tambah Tutorial hanya dapat dilakukan oleh pemilik tutorial. Pada halaman

tersebut, terdapat suatu bagian untuk melakukan pratinjau pada tutorial yang sedang dibuat oleh *user* sebagai pemilik tutorial tersebut. Terdapat *form* untuk menambahkan data tutorial yang terletak dibawah bagian Pratinjau Tutorial. *Form* tersebut terdiri dari kolom input teks untuk memasukkan judul tutorial dan *textarea* untuk memasukkan konten dari tutorial. Terdapat *toolbar* yang terletak diatas *textarea* yang akan digunakan untuk membantu proses pembuatan tutorial yang dilakukan oleh user.



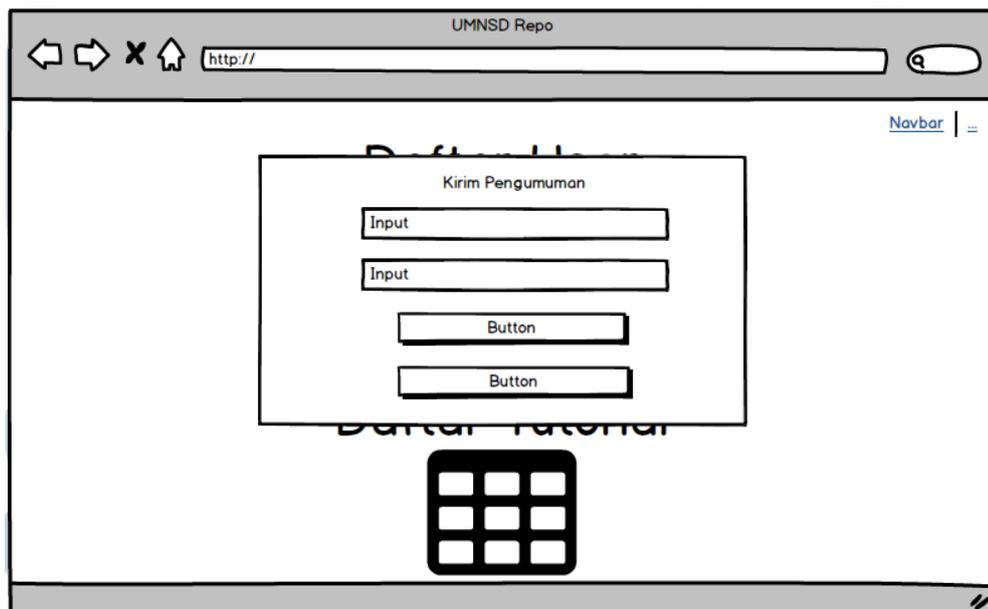
Gambar 3.37 Halaman Edit Tutorial

Gambar 3.37 menunjukkan halaman untuk melakukan pembaruan terhadap tutorial. Halaman tersebut memiliki *form* yang sama seperti Halaman Tambah Tutorial. Pada halaman Edit Tutorial, kolom input dan kolom *textarea* sudah dalam keadaan terisi oleh data tutorial yang sedang dipilih tersebut.



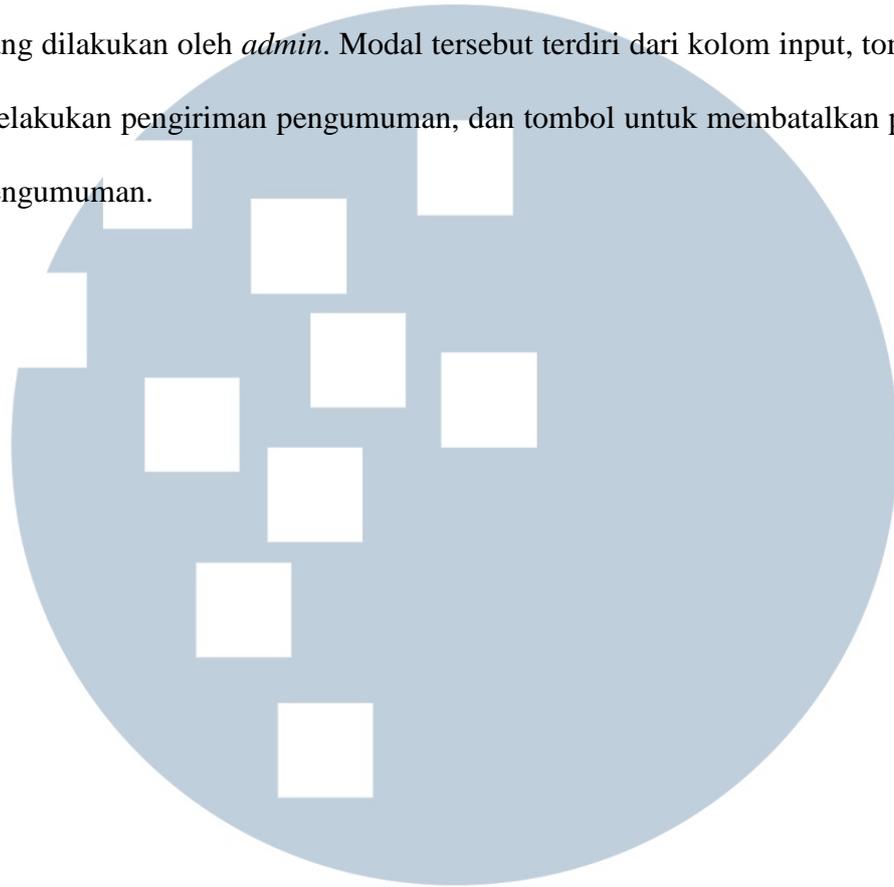
Gambar 3.38 Halaman Utama Admin

Halaman Utama Admin ditunjukkan pada Gambar 3.38. Pada halaman tersebut, terdapat dua tabel, yaitu Daftar User dan Daftar Tutorial. Diantara kedua tabel tersebut, terdapat satu tombol yang akan digunakan untuk melakukan pengiriman notifikasi kepada *user* yang terdaftar di aplikasi ini.



Gambar 3.39 Kirim Pengumuman

Gambar 3.39 menunjukkan modal untuk melakukan pengiriman pengumuman yang dilakukan oleh *admin*. Modal tersebut terdiri dari kolom input, tombol untuk melakukan pengiriman pengumuman, dan tombol untuk membatalkan pengiriman pengumuman.



# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA