

BAB 3 METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Alur penelitian yang digunakan dalam rancang bangun aplikasi deteksi berita hoaks menggunakan algoritma Naive Bayes berbasis Android dari awal sampai akhir dapat dilihat pada gambar 3.1.



Gambar 3.1. *Flowchart* alur penelitian

1. Studi Literatur

Tahap pertama diawali dengan melakukan pencarian dan mempelajari hal terkait penelitian. Proses ini melibatkan pembelajaran artikel dan jurnal dari internet yang relevan dengan sumber-sumber yang digunakan dalam penelitian. Tujuan dari tahap ini sebagai dasar teori untuk menyelesaikan masalah penelitian.

2. Pengumpulan Data

Pada tahap kedua, melakukan pengumpulan data untuk membuat dataset berita dengan topik politik di Indonesia melalui *website scraping* pada media *online* yakni Kompas, CNN, Detik untuk berita valid dan Turnbackhoax untuk berita hoaks.

3. Perancangan Aplikasi

Tahap ini melibatkan perancangan atau konsep awal aplikasi yang mencakup pembuatan *flowchart*, perancangan *user interface*, dan struktur database.

4. Pemrograman dan pengimplementasian algoritma pada aplikasi Android

Tahapan ini diawali dengan proses pembuatan aplikasi berdasarkan rancangan beserta fitur-fitur yang sudah ditetapkan. Pengimplementasian algoritma kedalam aplikasi juga termasuk ke dalam tahapan ini.

5. Pengujian Aplikasi

Tahap berikutnya setelah menyelesaikan pembuatan aplikasi dan model algoritma dengan melakukan pengujian aplikasi untuk memastikan fitur-fitur beserta implementasi algoritma dapat bekerja dengan baik. Metode *black box testing* digunakan untuk pengujian sistem. Sedangkan, pengujian akurasi model dilakukan dengan perhitungan confusion matrix.

6. Penulisan Laporan

Tahap ini mendokumentasikan seluruh proses perancangan dan pengembangan aplikasi dari awal sampai akhir kedalam laporan. Dalam proses penulisan laporan diikuti dengan konsultasi bersama dosen pembimbing untuk mendapat masukan serta memastikan penelitian telah dikerjakan dengan benar dengan hasil yang baik.

3.2 Analisis Kebutuhan

Hasil analisis kebutuhan untuk aplikasi deteksi berita hoaks berbasis Android ini dibagi menjadi 2 bagian yakni kebutuhan *input* dan *output*. Kebutuhan *input* mencakup informasi yang harus dimasukkan pengguna ke dalam aplikasi serta teks judul berita politik yang akan diperiksa. Informasi yang perlu dimasukkan untuk registrasi akun meliputi *username*, *email*, dan *password* pengguna. Kebutuhan *output* yang mencakup hasil yang dihasilkan oleh aplikasi yakni hasil pengecekan berita yang menunjukkan apakah judul berita tersebut tergolong hoaks atau tidak, dengan kategori *Fake News*(Hoaks) atau *Real News*(Bukan hoaks).

Pada aplikasi deteksi berita hoaks menyediakan fitur *login* dan *register* untuk meningkatkan fungsionalitas dan interaksi pengguna. Untuk mengakses fitur-fitur yang terdapat pada aplikasi, pengguna harus melakukan registrasi akun (untuk pengguna baru) ataupun login (untuk pengguna lama). Setelah login, pengguna dapat memanfaatkan fitur deteksi berita hoaks dan juga dapat melihat riwayat pengecekan berita yang telah dilakukan sebelumnya. Agar fitur-fitur tersebut berjalan dengan baik, maka aplikasi deteksi berita hoaks dirancang dengan dua sistem utama, yaitu *front-end* dan *back-end*. Proses otentikasi dan otorisasi pengguna seperti pada proses login dan registrasi beserta proses pengecekan berita hoaks dengan menggunakan algoritma untuk menganalisis dan mendeteksi konten berita yang dimasukkan pengguna akan dikerjakan oleh sistem *back-end*. Sistem ini juga menyimpan dan mengelola data riwayat pengecekan berita. Sedangkan, sistem *front-end* akan menyediakan antarmuka pengguna yang intuitif dan responsif. Selain itu, *front-end* juga menampilkan tampilan untuk pengecekan berita hoaks, menampilkan riwayat pengecekan sebelumnya dan juga melihat detail profil akun pengguna.

Pembuatan aplikasi ini menggunakan algoritma Multinomial Naive Bayes sebagai algoritma utama, namun untuk menerapkan algoritma ini agar dapat berjalan pada aplikasi Android, maka perlu menggunakan *Application Programming Interface*(API) pada sistem *back-end* yang nantinya akan dijalankan pada *web server*. API ini dibangun dengan bahasa pemrograman Python dengan menggunakan *framework* Flask dan SpringBoot dengan bahasa pemrograman Java. Seluruh hal yang berkaitan dengan *database*, seperti data pengguna maupun hasil pengecekan berita hoaks akan disimpan dengan menggunakan *database* pada PostgreSQL. Dengan demikian, algoritma Multinomial Naive Bayes akan diintegrasikan dengan sistem *back-end* melalui API yang diimplementasikan dengan Flask dan SpringBoot. Dengan menggunakan kombinasi teknologi ini, aplikasi akan dapat melakukan pengecekan berita hoaks menggunakan algoritma Multinomial Naive Bayes melalui API yang berjalan pada *web server*. Pada penelitian ini, aplikasi mobile berbasis Android dikembangkan menggunakan bahasa pemrograman Kotlin.

3.3 Perancangan Aplikasi

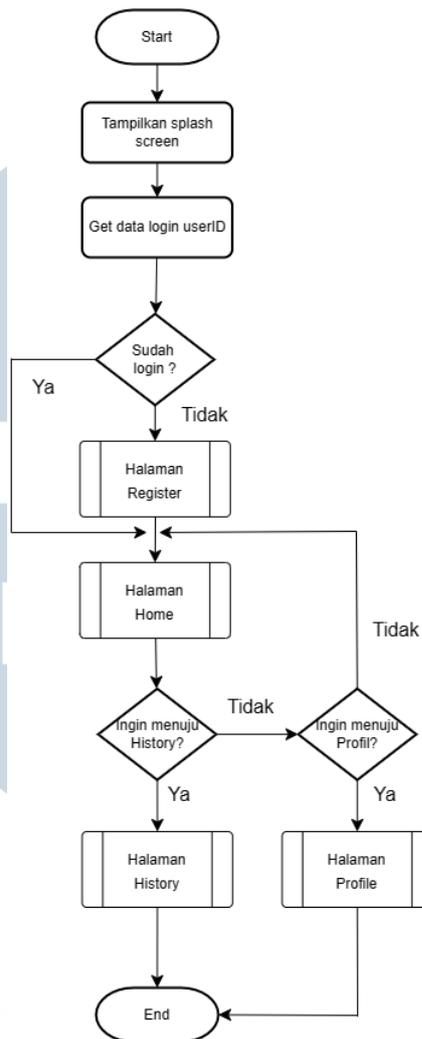
Perancangan aplikasi deteksi berita hoaks mencakup beberapa komponen, yaitu *flowchart*, *Entity Relationship Diagram* (ERD), struktur tabel beserta

rancangan antarmuka (*user interface*). Penggambaran hubungan antar tabel-tabel dalam basis data pada aplikasi digambarkan dengan pembuatan *Entity Relationship Diagram*. Selanjutnya, dibuat struktur tabel juga digunakan untuk memberikan detail mengenai struktur setiap tabel yang digunakan pada basis data seperti nama *column*, tipe data, beserta *identity*. Pada perancangan aplikasi terutama aplikasi Android, tentunya dibutuhkan rancangan antarmuka (*user interface*) untuk memberikan gambaran umum mengenai tata letak beserta desain aplikasi secara keseluruhan.

3.3.1 Flowchart

Berikut merupakan flowchart utama dari aplikasi HoaxGY dengan algoritma Multinomial Naive Bayes berbasis Android. Saat aplikasi HoaxGY dijalankan, akan muncul splash screen. Sebelum masuk ke dalam halaman Login, aplikasi akan mengecek data login userID yang disimpan pada perangkat Android menggunakan DataStore. Jika pengguna belum pernah login / pertama kali menggunakan aplikasi maka akan diarahkan ke halaman Register terlebih dahulu. Namun sebaliknya, apabila pengguna sudah pernah *login* kedalam aplikasi, maka pengguna akan langsung ditunjukkan ke halaman Home. Setelah berhasil masuk ke halaman Home, maka pengguna memiliki opsi untuk masuk ke halaman History atau halaman Profil sesuai dengan kebutuhan pengguna. Gambar 3.2 merupakan *flowchart* utama yang menjelaskan proses aplikasi HoaxGY secara umum.



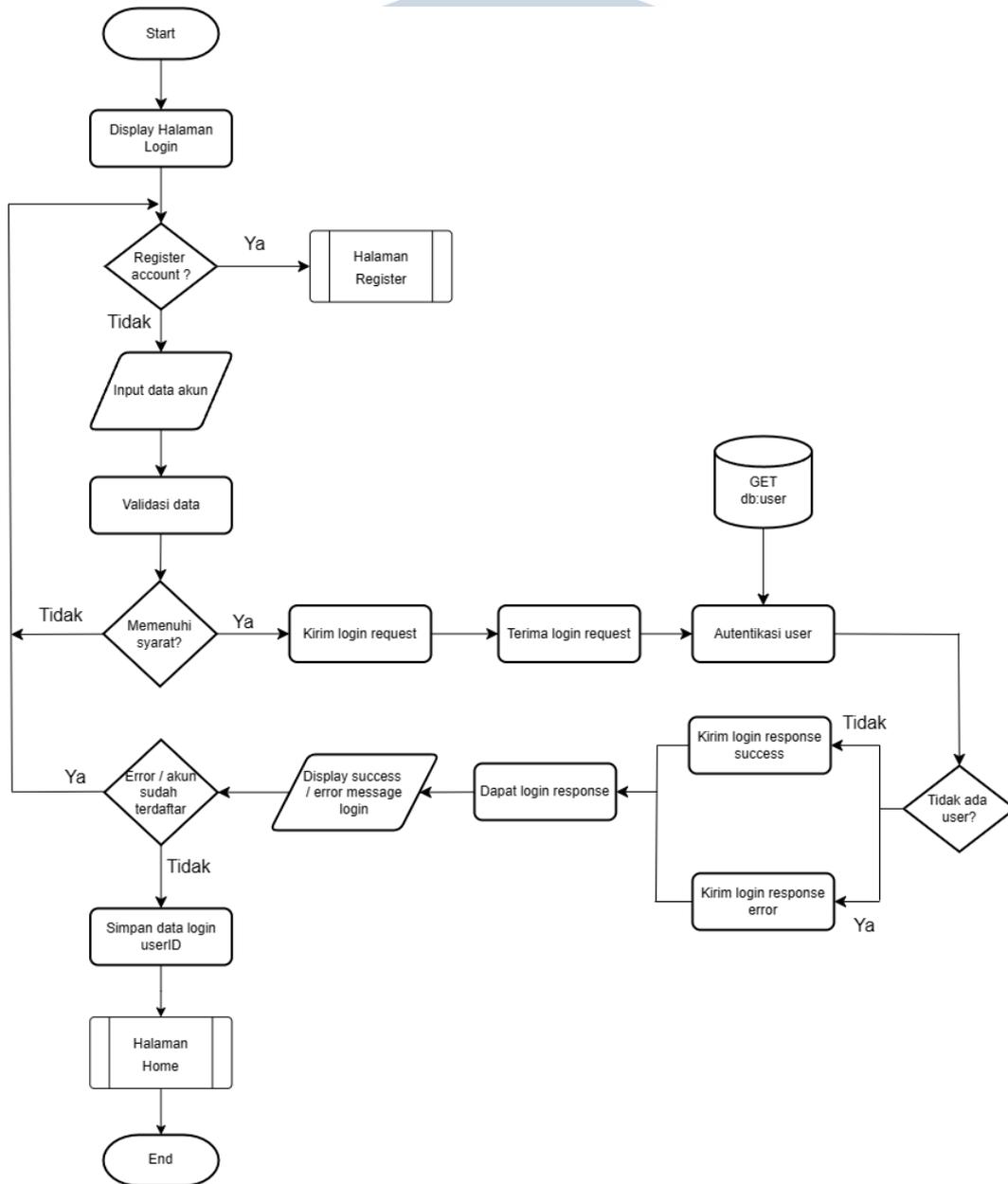


Gambar 3.2. *Flowchart* Utama aplikasi HoaxGY

A Halaman Login

Pada halaman Login, pengguna mempunyai opsi untuk langsung memasukkan data akun atau membuat akun baru terlebih dahulu. Jika ingin membuat akun terlebih dahulu, maka pengguna dapat masuk ke halaman Register. Namun, jika pengguna ingin langsung melakukan login, pengguna dapat langsung mengisi form login pada aplikasi dengan memasukkan username dan password. Setelah itu, sistem akan mendapatkan data dari pengguna berupa username dan password yang selanjutnya *request* tersebut divalidasi dan dicek pada *database*. Jika terdapat data pengguna pada *database* user, maka akan dikirim response success dan pengguna berhasil login. Namun sebaliknya, jika tidak terdapat pada *database* user maka akan dikirim response error. Apabila akun memenuhi syarat dan terverifikasi

maka aplikasi Android akan menyimpan data userID pengguna dan diarahkan langsung ke halaman Home. Gambar 3.3 merupakan alur dari halaman Login.



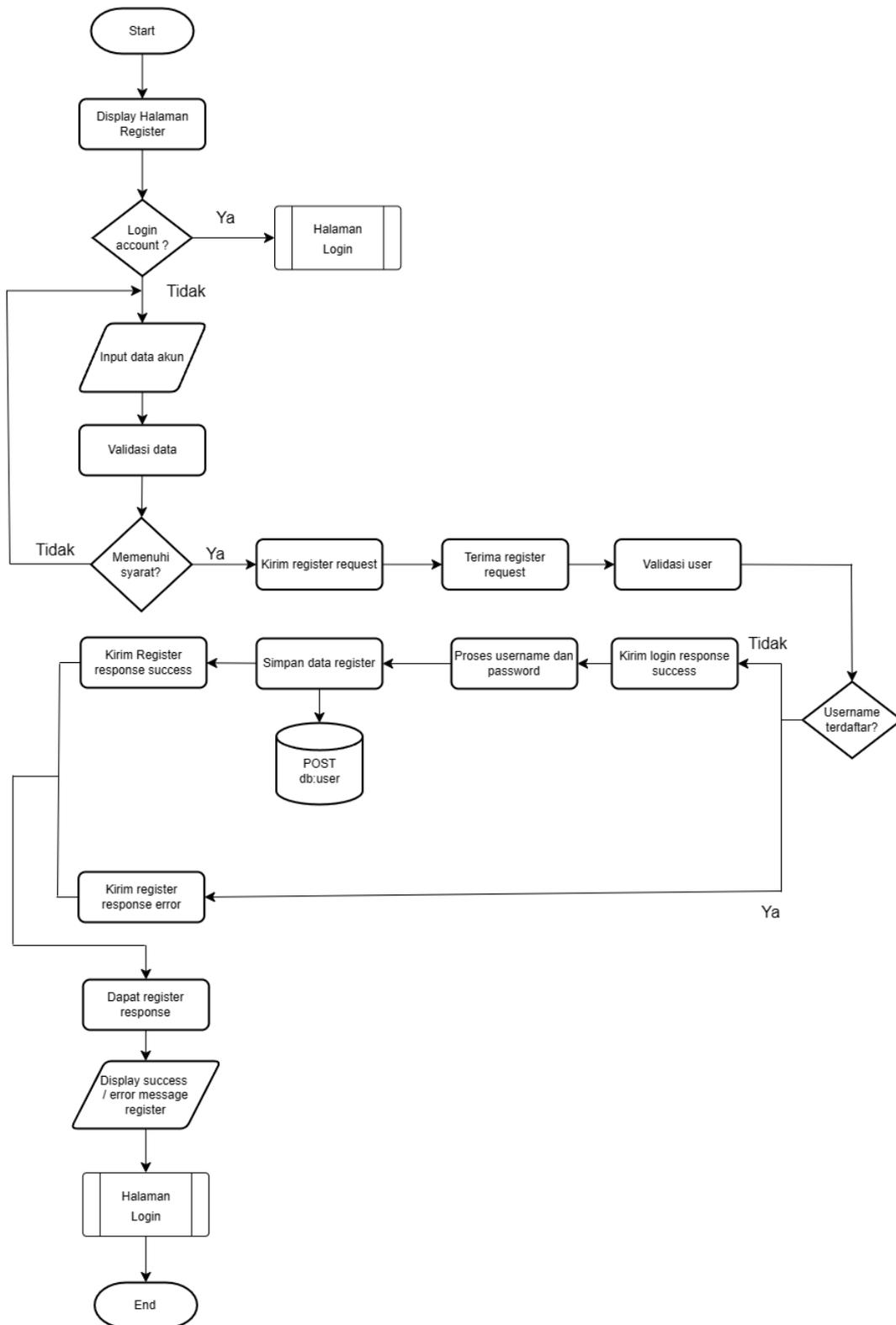
Gambar 3.3. Flowchart Login aplikasi HoaxGY

B Halaman Register

Pada halaman Register, terdapat pilihan untuk membuat akun atau langsung melakukan *login* ke akun yang sudah dimiliki. Jika pengguna ingin langsung *login*, maka akan dialihkan ke halaman Login. Namun sebaliknya, jika pengguna ingin membuat akun baru maka pengguna dapat langsung mengisi form register yang

terdapat pada aplikasi. Setelah itu, sistem akan menerima input request data dari pengguna dan dicek validasi apakah username sudah terdaftar sebelumnya. Jika username telah terdaftar maka akan dikembalikan *response error* pada pengguna. Jika tidak terdapat pada *database* user, maka username akan didaftarkan pada *database* dan akan mengirimkan response success kepada pengguna dan pengguna akan langsung diarahkan ke halaman Login. Gambar 3.4 merupakan alur dari halaman Register.



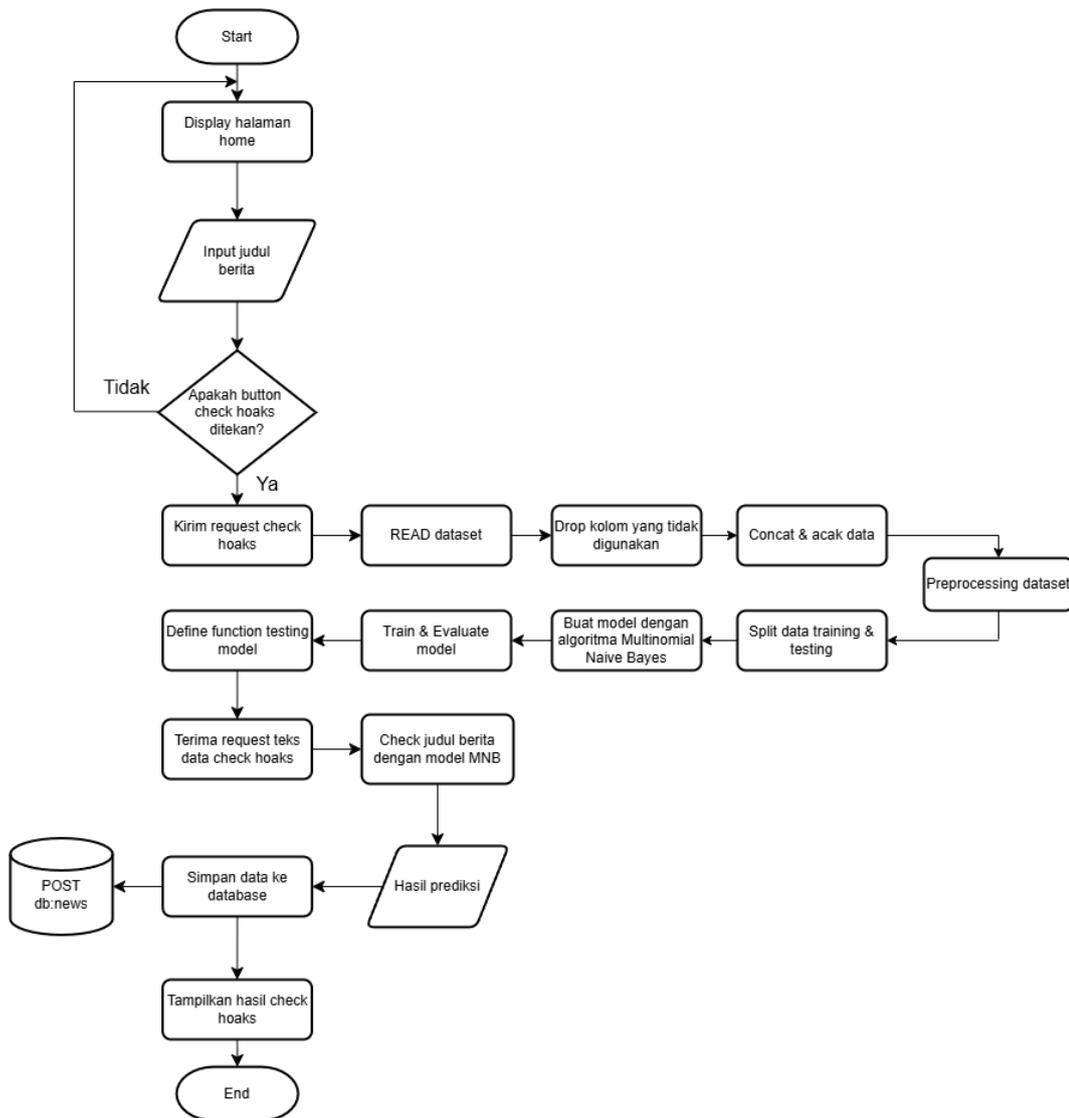


Gambar 3.4. Flowchart Register aplikasi HoaxGY

C Halaman Home

Pada halaman Home, pengguna dapat memasukkan judul berita politik kedalam *text field* yang disediakan untuk dapat dilakukan pengecekan berita hoaks dan dimasukkan kedalam *database news*. Terdapat juga *button* untuk melakukan pengecekan berita hoaks dengan mengirim request check hoaks ke back-end check hoaks. Pertama-tama, sistem back-end akan membaca 2 dataset (*real news* dan *fake news*) terlebih dahulu. Langkah kedua, melakukan *drop* pada kolom yang tidak digunakan dalam pembuatan model yang dilanjutkan dengan *concat* dan acak data pada dataset. Selanjutnya, melakukan *preprocessing* pada dataset. Berikutnya melakukan split data dimana data dibagi menjadi *training data* dan *testing data* dan dilanjutkan dengan pembuatan model dengan algoritma Multinomial Naive Bayes. Setelah model berhasil dibuat, maka dilakukan train model menggunakan *training data* dan juga evaluasi model dengan confusion matrix. Berikutnya, melakukan define function testing model sebelum menerima request data teks untuk dideteksi. Proses pengecekan berita hoaks dengan menggunakan algoritma Multinomial naive bayes. Hasil output berupa hasil prediksi yang datanya akan disimpan kedalam database news dengan metode POST. Setelah itu, hasil pengecekan akan ditampilkan kepada pengguna. Gambar 3.5 merupakan alur dari halaman Home.

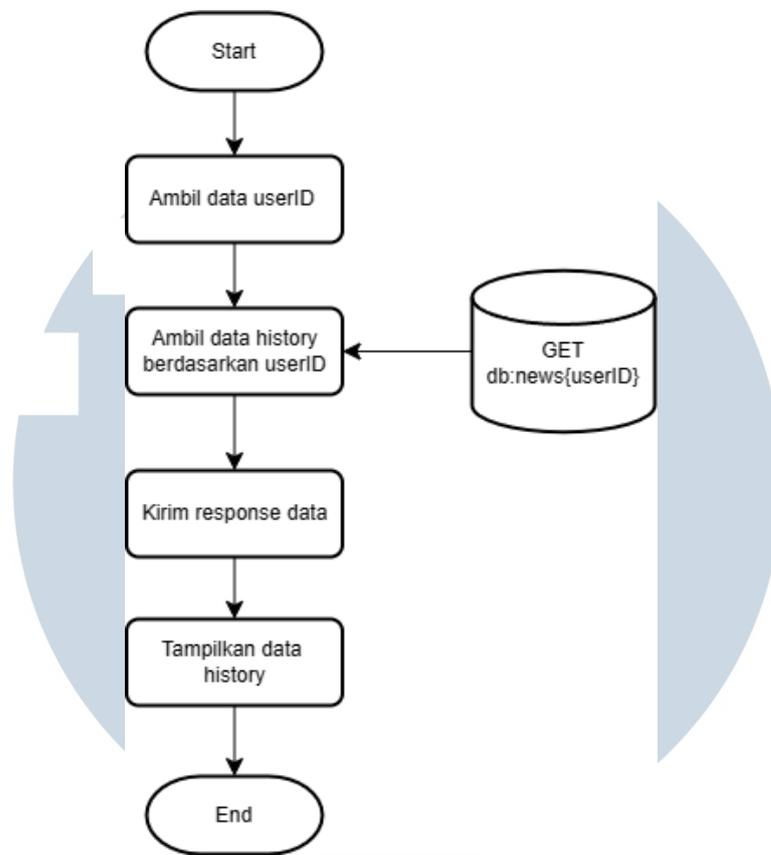




Gambar 3.5. Flowchart Home aplikasi HoaxGY

D Halaman History

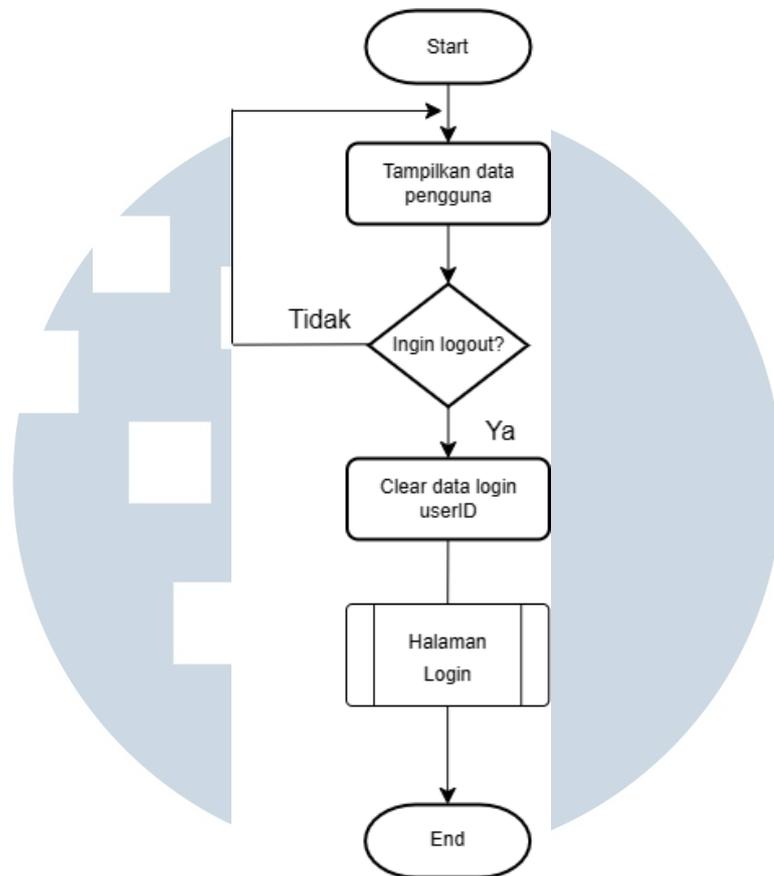
Halaman History terdapat daftar riwayat pengecekan berita hoaks yang diambil dari *database* user sesuai dengan akun yang berhasil login. Data history akan diambil dari *database* news berdasarkan *userID* pengguna dan nantinya dikirim response data untuk ditampilkan ke *front-end*. Gambar 3.6 merupakan alur dari halaman History.



Gambar 3.6. *Flowchart* History aplikasi HoaxGY

E Halaman Profile

Halaman Profile akan menampilkan data pengguna berupa username dan email pengguna yang datanya diambil pada saat pengguna login. Jika pengguna ingin *logout* dari akun, maka terdapat *button* logout yang akan mengarahkan pengguna ke halaman login kembali. Pada saat pengguna logout, maka data login userID akan dihapus dari perangkat, sehingga ketika pengguna masuk kembali kedalam aplikasi maka akan diarahkan ke halaman Login terlebih dahulu. Gambar 3.7 merupakan alur dari halaman History.

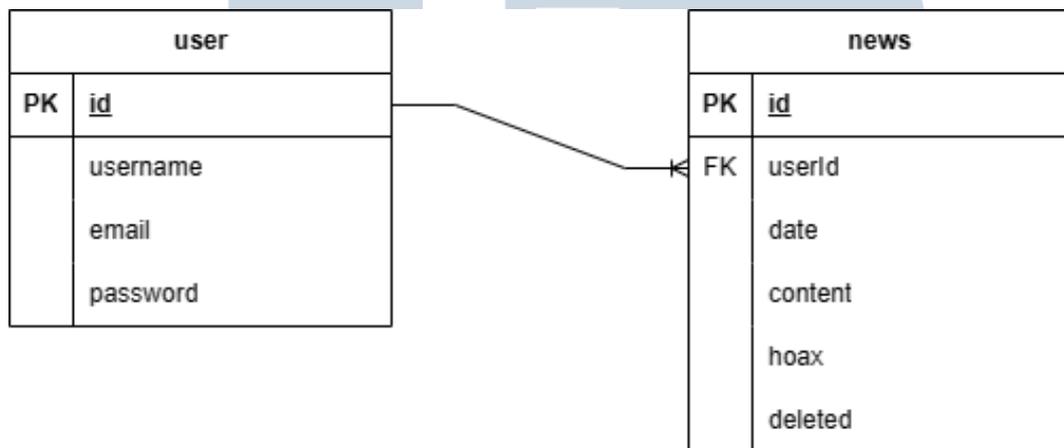


Gambar 3.7. *Flowchart* Profile aplikasi HoaxGY

UMMN
 UNIVERSITAS
 MULTIMEDIA
 NUSANTARA

3.3.2 Entity Relationship Diagram(ERD)

Diagram dibawah menjelaskan hubungan antar entitas pada aplikasi. Pada aplikasi deteksi berita hoaks, dibutuhkan 2 entitas yang digunakan, yaitu entitas "user" dengan entitas "news". Konsep relasi yang diterapkan antar entitas yaitu relasi *One To Many*, dimana *userId* pada entitas *news* merupakan *foreign key* dari *id* pada entitas *user* sebagai *primary key*. Gambar 3.8 adalah ERD dari *database* aplikasi HoaxGY.



Gambar 3.8. Entity Relationship Diagram Aplikasi

3.3.3 Struktur Tabel

Berikut ini adalah desain *database* yang akan digunakan dalam aplikasi HoaxGY. Struktur tabel berisi nama kolom, tipe data, beserta identitas kolom.

1. Tabel user

Tabel ini merupakan tabel data akun pengguna. Dimana pengguna memiliki id yang unik (*primary key*, *username*, *email*, dan *password*. Tabel 3.1 adalah struktur dari "Tabel User".

Tabel 3.1. Tabel user

Column	Data Type	Identity
id	integer	Primary Key
username	varchar(255)	
email	varchar(255)	
password	varchar(255)	

2. Tabel news

Tabel berikut berisikan data check berita hoaks yang telah dilakukan. Data check berita hoaks berisikan id sebagai *primary key*, userId sebagai *foreign key* dari "Tabel User". Terdapat kolom content yaitu teks yang ingin dideteksi. Selanjutnya, kolom hoax untuk menandai konten berita yang tergolong berita hoaks yang menggunakan tipe data boolean. Tabel 3.2 adalah struktur dari "Tabel News".

Tabel 3.2. Tabel news

Column	Data Type	Identity
id	integer	Primary Key
userId	integer	Foreign Key
date	date	
content	varchar(255)	
hoax	boolean	
deleted	boolean	



3.3.4 Rancangan Antarmuka

Bagian ini memuat hasil rancangan *user interface* yang akan diimplementasikan. Saat pertama kali membuka aplikasi maka akan menampilkan "Splash Screen". Gambar 3.9 merupakan hasil rancangan antarmuka dari "Splash Screen".



Gambar 3.9. Rancangan Splash Screen Aplikasi HoaxGY

Selanjutnya merupakan rancangan antarmuka untuk halaman Login. Terdapat logo aplikasi Hoaxgy dan diikuti dengan *input text box* username dan password untuk mengisi data akun dari pengguna. Jika pengguna belum mempunyai akun, maka terdapat *text button* "Daftar" yang akan mengarahkan pengguna ke halaman Register. Tombol Login akan men-*trigger* fungsi login pada aplikasi. Gambar 3.10 merupakan hasil rancangan antarmuka halaman Login.

MULTIMEDIA
NUSANTARA



Gambar 3.10. Rancangan antarmuka Login Aplikasi HoaxGY

Hasil rancangan antarmuka pada halaman Register hampir mirip dengan halaman Login. Hanya saja terdapat tambahan *text box email* dan juga terdapat *button* untuk men-*trigger* fungsi register. Apabila pengguna sudah mempunyai akun, maka terdapat *text button* "Login" yang akan mengarahkan pengguna ke halaman Login. Gambar 3.11 merupakan hasil rancangan antarmuka dari halaman Register.

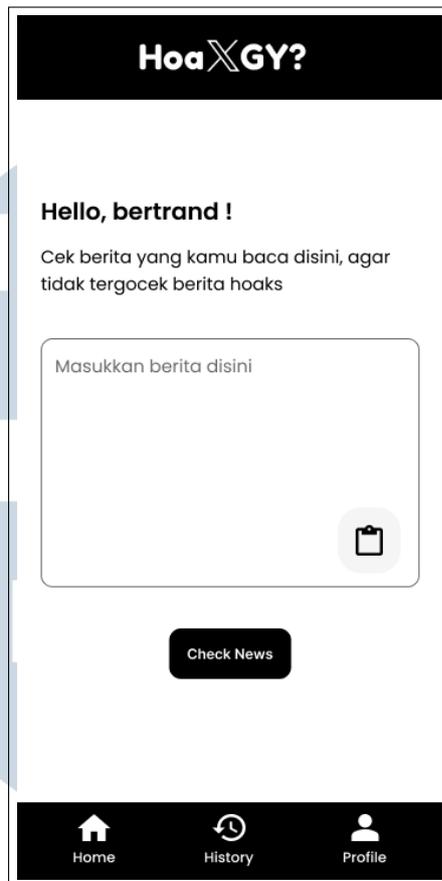
U M I N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.11. Rancangan antarmuka Register Aplikasi HoaxGY

Pada halaman Home, terdapat *text box* untuk mengisi judul berita yang akan dideteksi. Terdapat juga *button paste* untuk memudahkan pengguna dalam menempel teks judul berita. Setelah memasukkan teks judul berita, maka pengguna tinggal menekan *button Check News* untuk menjalankan fungsi deteksi berita. Gambar 3.12 adalah rancangan antarmuka dari halaman Home.

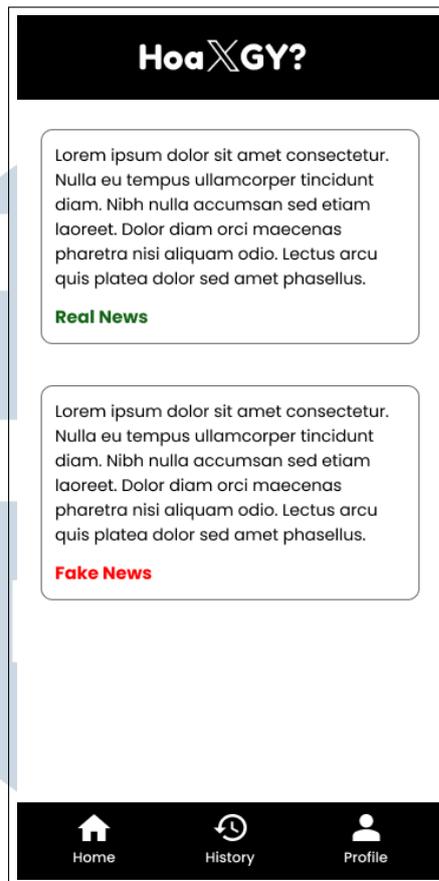
UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.12. Rancangan antarmuka Home Aplikasi HoaxGY

Halaman History berisi mengenai judul berita yang telah dideteksi. Pada halaman ini akan menampilkan judul berita dan hasil pengecekan hoaks. Gambar 3.13 adalah rancangan antarmuka dari halaman History.

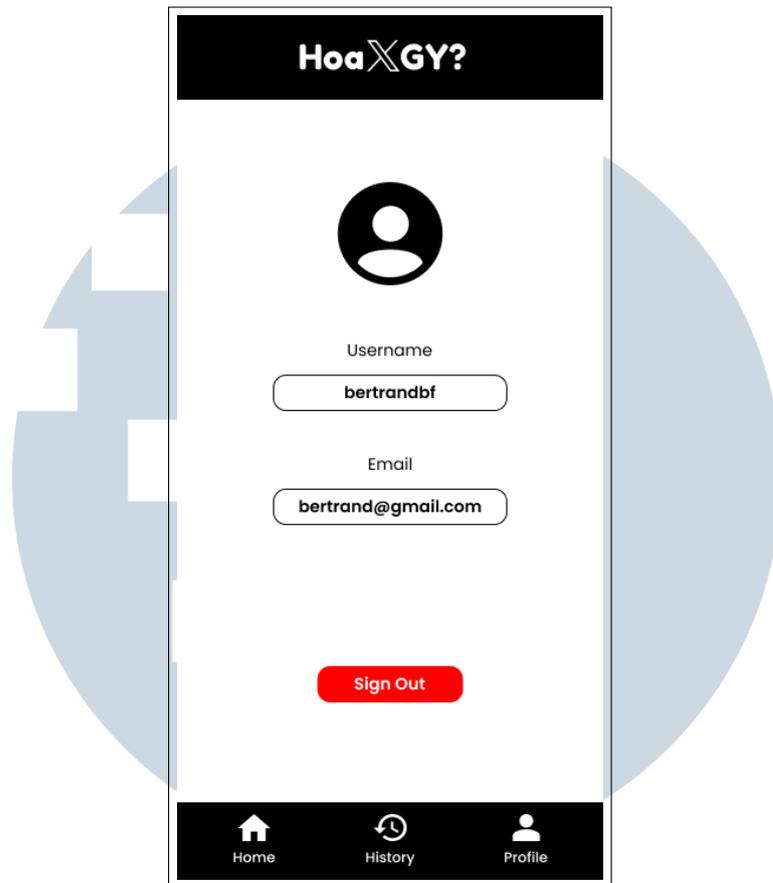
UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.13. Rancangan antarmuka History Aplikasi HoaxGY

Halaman Profile berisi mengenai informasi akun pengguna berupa email dan username yang terdaftar. Selain itu, juga terdapat *button* untuk *signout* dari akun tersebut. Gambar 3.14 adalah rancangan antarmuka dari halaman Profile.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.14. Rancangan antarmuka Profile Aplikasi HoaxGY

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA