

## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Studi Literatur

Studi literatur merupakan serangkaian kegiatan untuk melakukan penelusuran dengan metode membaca referensi dari sumber ilmiah, mencatat atau merangkum hal penting dari sumber ilmiah, pengumpulan data pustaka, dan pengolahan bahan penelitian. Studi literatur dilakukan untuk mencari referensi yang ada pada penelitian sebelumnya, mempelajari teori yang akan diimplementasikan pada proses perancangan. Hal itu bertujuan supaya penelitian memiliki landasan teori yang jelas.

#### 3.2 Pengumpulan Data

Pada tahapan ini, hal yang dilakukan adalah mengumpulkan data-data yang diperlukan. Pada penelitian ini, maka data yang diperlukan adalah data-data mengenai *hero* yang terdapat pada *Mobile Legends*. Data yang akan diambil adalah data terbaru atau *update* terakhir dari *Mobile Legends*.

Data *hero* tidak diambil dari data statistik (seperti *win rate* atau *pick rate*), melainkan data kualitatif mengenai mekanik *skill* setiap *hero*. Pengumpulan data dilakukan melalui proses *feature engineering* dengan langkah-langkah sebagai berikut:

1. Studi Literatur Fandom Wiki: Setiap *hero* (120+ *hero*) dianalisis satu per satu melalui *database Mobile Legends Fandom Wiki*.
2. Ekstraksi Fitur: Deskripsi *skill* setiap *hero* diterjemahkan menjadi fitur biner (0 atau 1).
3. Pembuatan Vektor: Fitur-fitur tersebut dikategorikan ke dalam dua jenis vektor:
  - (a) Vektor Kemampuan ( $V_{has}$ ): Fitur yang dimiliki *hero*, seperti *has\_burst*, *has\_hard\_cc*, *has\_anti\_heal*, dll.
  - (b) Vektor Kelemahan ( $V_{weak}$ ): Fitur yang membuat *hero* rentan, seperti *weak\_to\_burst*, *weak\_to\_hard\_cc*, *weak\_to\_anti\_heal*, dll.

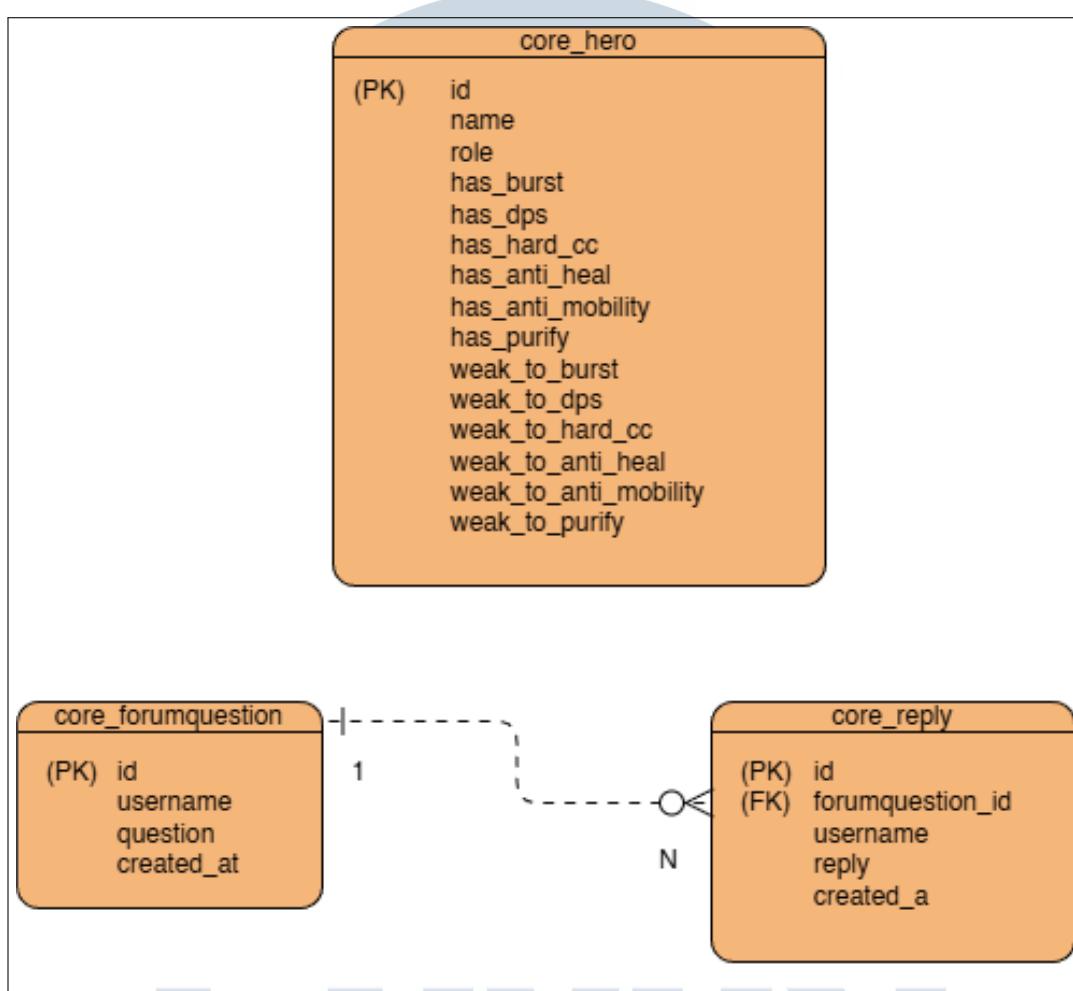
Proses ini mengubah data kualitatif ("Alucard lemah terhadap *anti-heal*") menjadi data kuantitatif biner (*weak\_to\_anti\_heal*: 1) yang dapat diproses oleh algoritma CBF.

### 3.3 Perancangan Sistem

Perancangan sistem merupakan tahap perencanaan struktur dan alur kerja sistem rekomendasi counter pick hero Mobile Legends sebelum dilakukan proses implementasi. Pada tahap ini, perancangan sistem dilakukan untuk menggambarkan bagaimana komponen-komponen sistem saling berinteraksi, mulai dari pengguna memasukkan data hero musuh hingga sistem menghasilkan rekomendasi hero counter pick. Perancangan sistem bertujuan untuk memastikan bahwa sistem yang dibangun dapat berjalan sesuai dengan kebutuhan fungsional dan mudah diimplementasikan.



### 3.3.1 Entity Relationship Diagram (ERD)



Gambar 3.1. Entity Relationship Diagram

Entity Relationship Diagram (ERD) pada sistem ini menggambarkan hubungan antar tabel inti dalam database *ml\_counterpick*. Tabel *core\_hero* berdiri secara independen sebagai penyimpan data hero beserta atribut-atributnya. Sementara itu, tabel *core\_forumquestion* memiliki relasi satu-ke-banyak dengan tabel *core\_reply*, di mana satu pertanyaan forum dapat memiliki lebih dari satu balasan. ERD ini digunakan sebagai acuan dalam implementasi struktur tabel pada database MySQL guna memastikan keterkaitan data tersusun secara konsisten dan sesuai dengan prinsip basis data relasional.

### 3.3.2 Perancangan Database

Database *ml\_counterpick* terdiri dari beberapa tabel bawaan framework Django serta tabel inti yang digunakan secara langsung dalam sistem rekomendasi. Pada penelitian ini, pembahasan database difokuskan pada tabel inti (*core tables*) yang berperan langsung dalam proses rekomendasi *counter-pick* hero Mobile Legends dan pengelolaan forum komunitas. Sementara itu, tabel bawaan Django seperti *auth\_\**, *django\_\**, dan *sessions* digunakan untuk mendukung kebutuhan sistem secara internal, sehingga tidak dibahas secara rinci dalam penelitian ini.

Tabel 3.1. Daftar Inti Database Sistem

No	Nama Tabel	Fungsi
1	core_hero	Menyimpan data utama hero Mobile Legends, termasuk nama hero, role, serta atribut kemampuan dan kelemahan hero yang direpresentasikan dalam bentuk data biner (0 dan 1) yang digunakan dalam perhitungan Content-Based Filtering.
2	core_forumquestion	Menyimpan data pertanyaan atau topik diskusi yang dibuat oleh pengguna pada forum komunitas.
3	core_reply	Menyimpan data balasan atau komentar pengguna terhadap pertanyaan pada forum diskusi.

Tabel 3.2 digunakan untuk menyimpan data hero Mobile Legends yang menjadi objek utama dalam sistem rekomendasi counter pick. Tabel ini memuat informasi dasar hero seperti nama dan role, serta atribut kemampuan dan kelemahan hero yang direpresentasikan dalam bentuk data biner, yaitu nilai 1 untuk menunjukkan atribut dimiliki dan nilai 0 untuk menunjukkan atribut tidak dimiliki. Data pada tabel *core\_hero* digunakan secara langsung dalam proses perhitungan algoritma Content-Based Filtering (CBF), sehingga tabel ini tidak memiliki relasi langsung dengan tabel lainnya.

Tabel 3.2. Struktur Tabel *core\_hero*

No	Nama Atribut	Tipe Data	Keterangan
1	id	INT (PK)	Primary Key
2	name	VARCHAR(50)	Nama hero
3	role	VARCHAR(20)	Role hero
4	has_burst	TINYINT(1)	Atribut burst (0/1)
5	has_dps	TINYINT(1)	Atribut DPS (0/1)
6	has_hard_cc	TINYINT(1)	Atribut hard CC (0/1)
7	has_anti_heal	TINYINT(1)	Atribut anti heal (0/1)
8	has_anti_mobility	TINYINT(1)	Atribut anti mobility (0/1)
9	has_purify	TINYINT(1)	Atribut purify (0/1)
10	weak_to_burst	TINYINT(1)	Lemah terhadap burst
11	weak_to_dps	TINYINT(1)	Lemah terhadap DPS
12	weak_to_hard_cc	TINYINT(1)	Lemah terhadap hard CC
13	weak_to_anti_heal	TINYINT(1)	Lemah terhadap anti heal
14	weak_to_anti_mobility	TINYINT(1)	Lemah terhadap anti mobility
15	weak_to_purify	TINYINT(1)	Lemah terhadap purify

Tabel 3.3 *core\_forumquestion* digunakan untuk menyimpan data pertanyaan yang diajukan oleh pengguna pada forum komunitas sistem. Tabel ini memuat informasi mengenai identitas pengguna, isi pertanyaan, serta waktu pembuatan pertanyaan. Data pada tabel 3.3 *core\_forumquestion* berfungsi sebagai media interaksi pengguna dalam forum diskusi dan menjadi induk (parent) dalam relasi dengan tabel balasan forum.

Tabel 3.3. Struktur Tabel *core\_forumquestion*

No	Nama Atribut	Tipe Data	Keterangan
1	id	INT (PK)	Primary Key
2	username	VARCHAR(50)	Nama pengguna
3	question	TEXT	Isi pertanyaan forum
4	created_at	DATETIME	Waktu dibuat

Tabel 3.4 *core\_reply* digunakan untuk menyimpan data balasan atau komentar terhadap pertanyaan yang terdapat pada forum komunitas. Tabel ini memiliki relasi one-to-many dengan tabel 3.4 *core\_forumquestion*, di mana satu pertanyaan

forum dapat memiliki lebih dari satu balasan. Setiap data balasan mencakup informasi pengguna, isi balasan, serta waktu pembuatan balasan. Relasi ini diimplementasikan melalui penggunaan atribut foreign key yang menghubungkan tabel *core\_reply* dengan tabel *core\_forumquestion*.

Tabel 3.4. Struktur Tabel *core\_reply*

No	Nama Atribut	Tipe Data	Keterangan
1	id	INT (PK)	Primary Key
2	forumquestion_id	INT (FK)	Foreign Key ke core_forumquestion
3	username	VARCHAR(50)	Nama pengguna
4	reply	TEXT	Isi balasan
5	created_at	DATETIME	Waktu dibuat

### 3.3.3 Normalisasi Database

Proses normalisasi database dilakukan untuk menghindari redundansi data dan menjaga konsistensi penyimpanan data. Pada sistem ini, tabel *core\_hero*, *core\_forumquestion*, dan *core\_reply* telah memenuhi bentuk normal ketiga (Third Normal Form / 3NF).

Setiap tabel memiliki karakteristik sebagai berikut:

1. 1NF (First Normal Form): Setiap atribut bersifat atomik, artinya setiap kolom hanya menyimpan satu nilai.
2. 2NF (Second Normal Form): Tidak terdapat ketergantungan parsial, sehingga setiap atribut non-kunci bergantung sepenuhnya pada primary key.
3. 3NF (Third Normal Form): Tidak terdapat ketergantungan transitif antar atribut non-kunci, sehingga setiap atribut non-kunci hanya bergantung pada primary key.

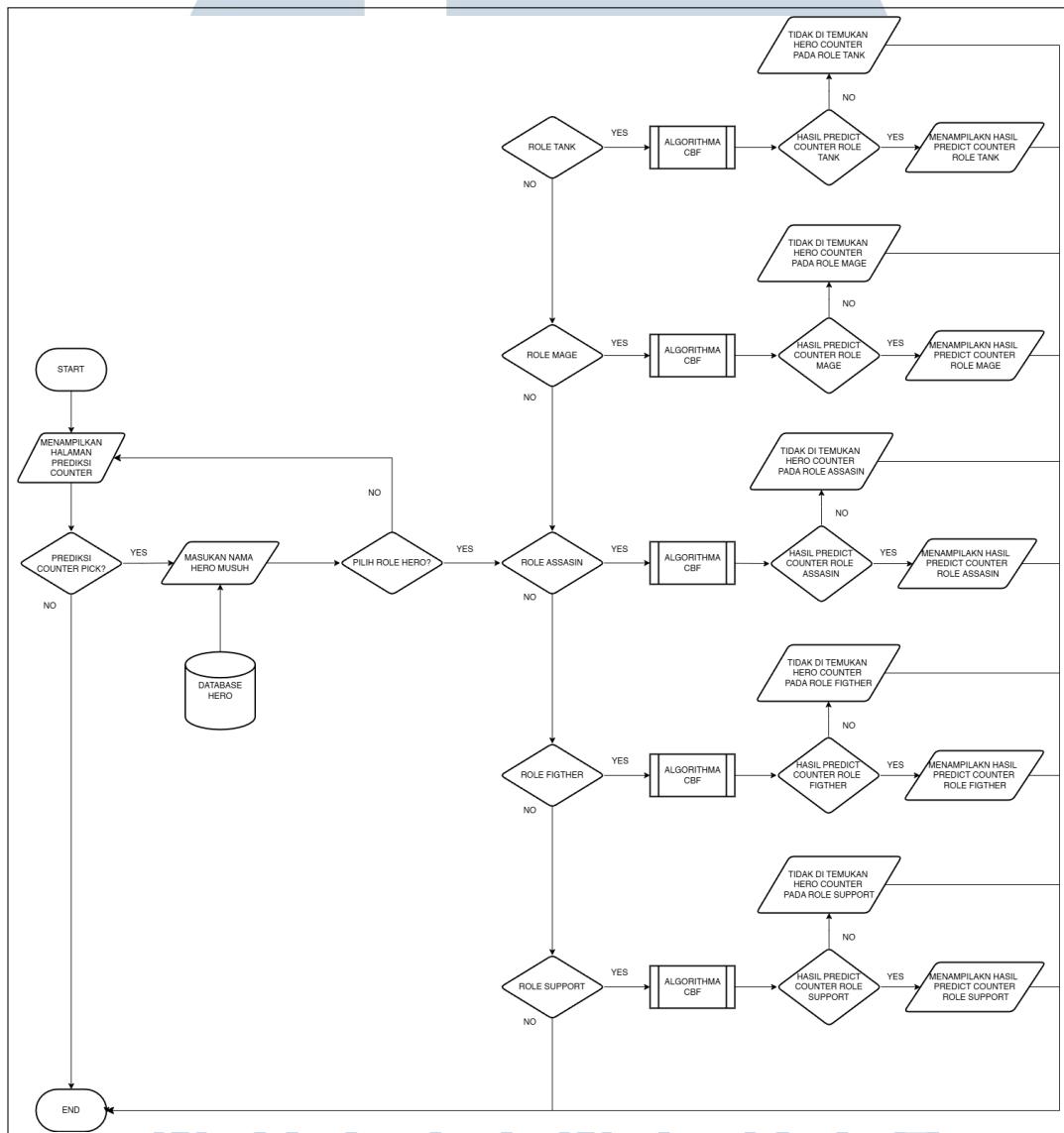
Dengan demikian, struktur database pada sistem ini telah terorganisir dengan baik, meminimalkan duplikasi data, dan mempermudah proses pemeliharaan serta pengelolaan data.

### 3.3.4 Flowchart

Perancangan *flowchart* pada *website* ini mencakup lima alur utama, yaitu untuk Sistem Prediksi *Counter Hero*, Sistem Forum Komunitas, Sistem Data

*Hero*, Sistem Menu Utama, dan Algoritma *Content-Based Filtering* (CBF). Setiap *flowchart* dirancang untuk menjelaskan proses dan alur interaksi pengguna dengan sistem secara rinci agar memudahkan navigasi dan penggunaan fitur.

## A Flowchart Sistem Prediksi Counter Hero



Gambar 3.2. Flowchart Sistem Prediksi Counter Hero

*Flowchart* yang ditunjukkan pada Gambar 3.2 menggambarkan sistem prediksi *counter hero* yang dimulai dengan menampilkan halaman prediksi *counter*. Sistem kemudian mewajibkan pengguna untuk memilih *role hero* yang ingin dicari

(*Tank, Mage, Assassin, Fighter, Support, Marksman*) dan memasukkan nama *hero* musuh. Setiap kali *input role* dan *hero* musuh diterima, sistem akan menjalankan algoritma CBF (*Content-Based Filtering*) untuk menganalisis dan memprediksi *counter* yang sesuai. Jika hasil prediksi ditemukan, sistem akan menampilkan 5 hasil prediksi *counter* teratas untuk *role* tersebut. Namun jika tidak ditemukan *hero counter* (misalnya tidak ada *Tank* yang cocok), sistem akan menampilkan pesan bahwa *hero counter* tidak ditemukan pada *role* tersebut. Alur ini memastikan bahwa setiap pencarian mendapatkan analisis yang spesifik berdasarkan *role* yang diminta pengguna.

## B Flowchart Sistem Forum Komunitas

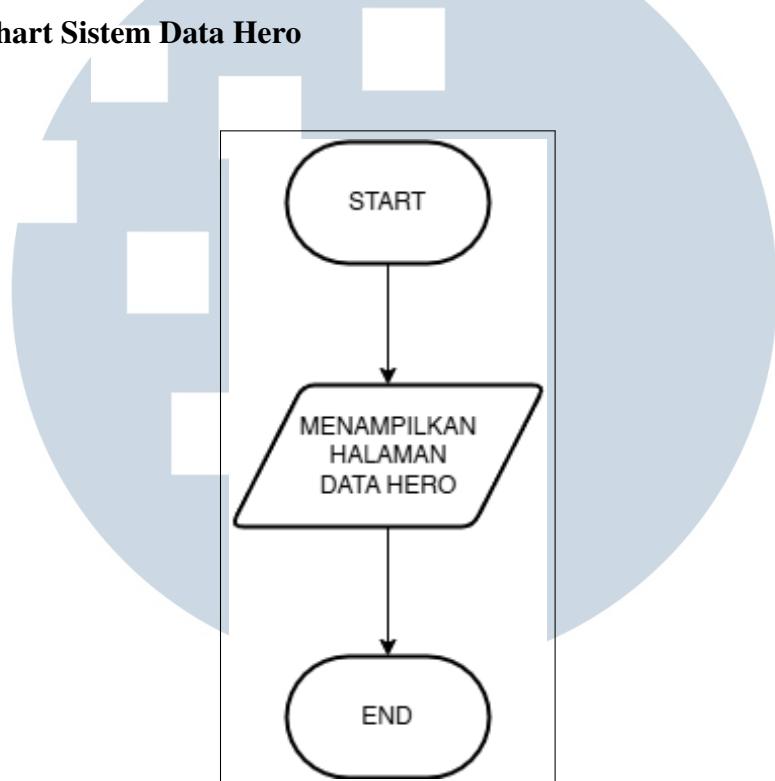


Gambar 3.3. Flowchart Sistem Forum Komunitas

Flowchart pada Gambar 3.3 menunjukkan sistem forum komunitas yang berfungsi sebagai *platform* interaksi antar pengguna. Proses dimulai dengan menampilkan halaman forum komunitas kepada pengguna. Sistem kemudian memberikan dua pilihan utama: apakah pengguna ingin membuat pertanyaan baru atau memberikan komentar pada topik yang sudah ada. Jika pengguna memilih untuk bertanya, sistem akan menampilkan *form input* nama dan pertanyaan yang memungkinkan pengguna untuk mengajukan pertanyaan baru. Jika pengguna memilih untuk berkomentar, sistem akan menampilkan *form input* nama dan

komentar untuk merespons diskusi yang telah ada. Setelah salah satu aksi dipilih dan dilakukan, sistem akan berakhir. Alur ini dirancang untuk memfasilitasi komunikasi dua arah antara pengguna dalam komunitas *Mobile Legends*.

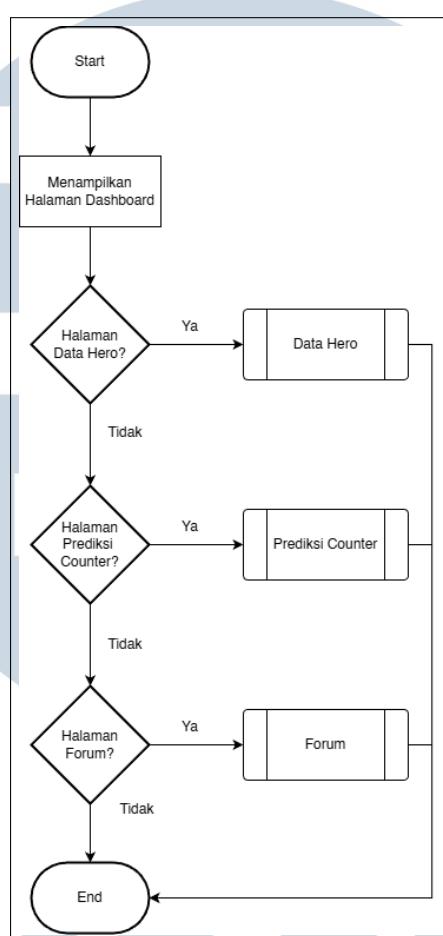
### C Flowchart Sistem Data Hero



Gambar 3.4. Flowchart Sistem Data Hero

Flowchart yang ditampilkan pada Gambar 3.4 merupakan sistem yang paling sederhana, hanya terdiri dari tiga tahap linear. Sistem dimulai dengan menampilkan halaman data *hero* kepada pengguna yang berisi informasi lengkap mengenai semua *hero* yang tersedia dalam game *Mobile Legends*. Halaman ini menampilkan data *hero* seperti nama dan *role*. Setelah halaman berhasil ditampilkan, sistem langsung berakhir tanpa ada interaksi atau pemrosesan data tambahan. Flowchart ini menggambarkan fungsi dasar untuk menampilkan informasi statis yang dapat digunakan pengguna sebagai referensi dalam memahami karakteristik setiap *hero*.

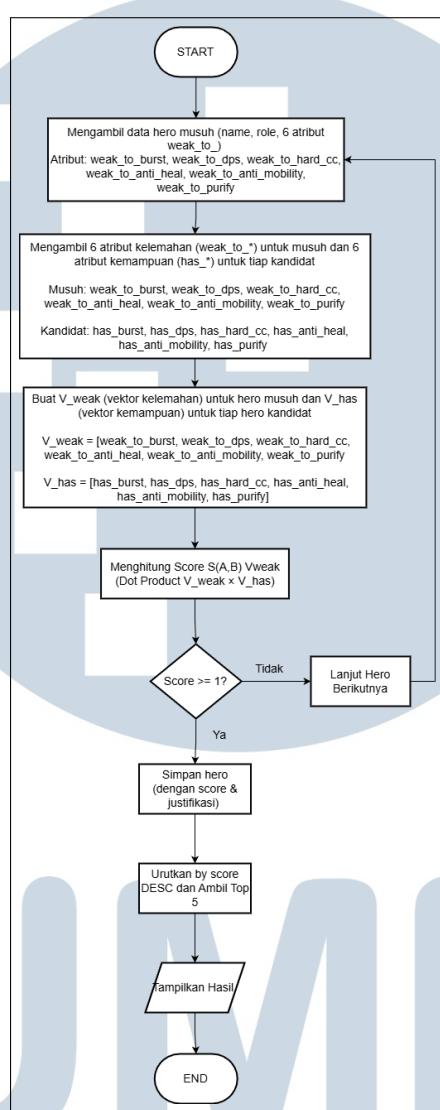
## D Flowchart Sistem Menu Utama



Gambar 3.5. Flowchart Sistem Menu Utama

Flowchart yang ditunjukkan pada Gambar 3.5 menampilkan struktur navigasi menu utama aplikasi secara linear dan terorganisir. Sistem dimulai dan kemudian menampilkan halaman beranda sebagai *landing page* yang menyambut pengguna. Selanjutnya sistem menampilkan halaman data *hero* yang berisi informasi komprehensif tentang *hero-hero* yang tersedia dalam *game*. Kemudian sistem menampilkan halaman prediksi *counter* untuk mengakses fungsi prediksi utama aplikasi. Setelah itu sistem menampilkan halaman forum komunitas untuk memfasilitasi interaksi dan diskusi antar pengguna. Semua halaman ditampilkan secara berurutan tanpa ada percabangan atau pilihan khusus, dan sistem berakhir setelah semua komponen utama aplikasi telah ditampilkan. Struktur ini memberikan gambaran menyeluruh tentang semua fitur yang tersedia dalam aplikasi.

## E Flowchart Algoritma Content-Based Filtering (CBF)



Gambar 3.6. Flowchart Algoritma Content-Based Filtering (CBF)

Flowchart pada Gambar 3.6 menjelaskan detail algoritma *Content-Based Filtering* yang menjadi inti dari sistem prediksi *counter hero*. Proses dimulai dengan mengambil *input hero* musuh dan *role* pilihan dari pengguna. Sistem kemudian mengakses *database* untuk mengambil data *hero* musuh. Selanjutnya, sistem membuat Vektor Kelemahan ( $V_{weak}$ ) untuk *hero* musuh tersebut, berdasarkan fitur-fitur biner (0/1) seperti *weak\_to\_burst*, *weak\_to\_hard\_cc*, dan *weak\_to\_anti\_heal*.

Setelah Vektor Kelemahan musuh dibuat, sistem menyaring semua *hero* lain di *database* berdasarkan *role* yang dipilih pengguna. Sistem kemudian melakukan

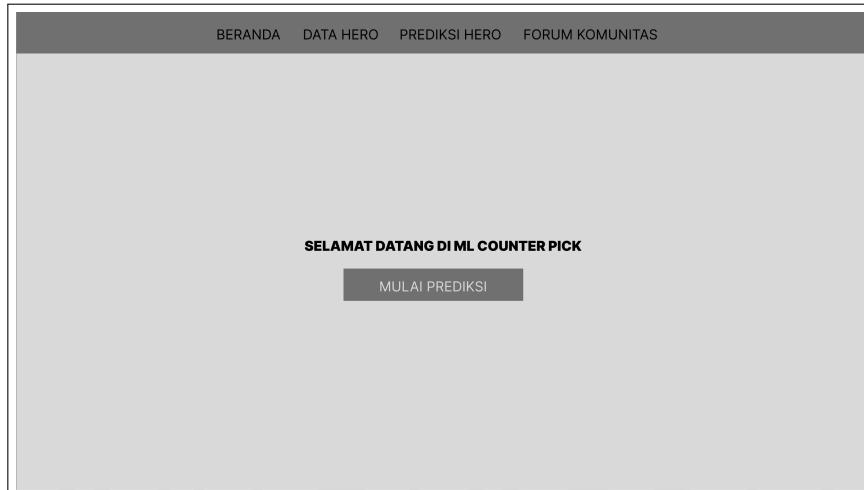
*looping* pada setiap *hero* yang lolos filter *role* (calon *counter*). Di dalam *loop*, sistem membuat Vektor Kemampuan ( $V_{has}$ ) untuk calon *counter* (e.g., *has\_burst*, *has\_hard\_cc*, *has\_anti\_heal*).

Sistem menghitung *Cosine Similarity* untuk mengukur tingkat kecocokan antara  $V_{weak}$  musuh dan  $V_{has}$  calon *counter*. Skor *similarity* (antara 0 hingga 1) ini disimpan. Setelah *loop* selesai, sistem mengurutkan semua calon *counter* berdasarkan skor *Cosine Similarity* mereka secara *descending* (dari tertinggi ke terendah). Lima *hero* dengan skor tertinggi (paling mirip) akan diambil dan ditampilkan sebagai hasil prediksi *counter*. Algoritma ini memastikan rekomendasi *counter* yang akurat berdasarkan pencocokan mekanik *skill*.

### 3.3.5 Wireframe

*Wireframe* atau antarmuka *website* terbagi menjadi beberapa halaman yaitu halaman utama (Beranda), halaman Data Hero, halaman Prediksi Hero, dan halaman Forum Komunitas.

#### A Halaman Beranda



Gambar 3.7. Halaman Beranda

Gambar 3.7 menampilkan halaman Beranda yang merupakan halaman awal ketika pengguna mengakses *website* ML Counter Pick. Halaman ini memiliki navigasi utama di bagian atas yang terdiri dari menu BERANDA, DATA HERO, PREDIKSI HERO, dan FORUM KOMUNITAS. Di tengah halaman, terdapat teks

sambutan "SELAMAT DATANG DI ML COUNTER PICK" yang ditampilkan dengan *font* besar dan tebal untuk memberikan kesan pertama yang kuat kepada pengguna. Pada bagian tengah bawah teks sambutan, terdapat tombol *call-to-action* bertuliskan MULAI PREDIKSI yang berfungsi sebagai pintu masuk utama bagi pengguna untuk memulai proses pencarian *hero counter*. Tombol ini dirancang dengan ukuran yang cukup besar dan ditempatkan secara strategis untuk memudahkan pengguna dalam mengakses fitur utama *website*. Desain halaman beranda ini menggunakan pendekatan minimalis dengan fokus pada kemudahan navigasi, sehingga pengguna dapat langsung memahami tujuan utama *website* dan mengakses fitur yang dibutuhkan tanpa kebingungan.

## B Halaman Data Hero

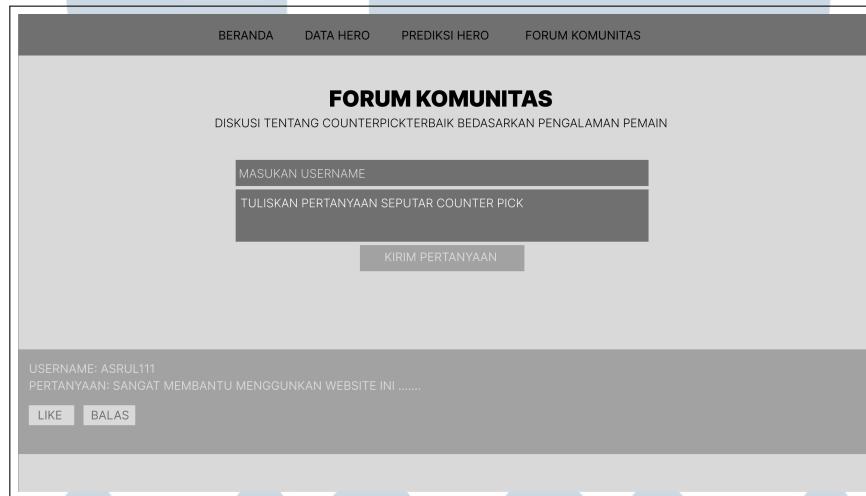


Gambar 3.8. Halaman Data Hero

Gambar 3.8 menunjukkan Halaman Data Hero yang berfungsi untuk menampilkan daftar lengkap *hero-hero* yang tersedia dalam *database* sistem. Halaman ini memiliki struktur navigasi yang sama di bagian atas dengan menu BERANDA, DATA HERO, PREDIKSI HERO, dan FORUM KOMUNITAS. Konten utama halaman ini menampilkan galeri *hero* dalam bentuk *grid layout* dengan delapan kolom *hero* yang tersusun rapi. Setiap *card hero* terdiri dari kotak gambar bertuliskan "FOTO HERO" yang berfungsi sebagai *placeholder* untuk menampilkan gambar atau ikon *hero*. Di bawah setiap gambar, terdapat informasi "NAMA HERO" yang menampilkan nama *hero* tersebut, dan di baris berikutnya terdapat keterangan "ROLE HERO" dalam huruf miring yang menunjukkan peran

atau kelas *hero* dalam permainan (seperti *Tank*, *Marksman*, *Mage*, *Fighter*, *Assassin*, atau *Support*). Tata letak *grid* ini dirancang responsif dengan jarak yang proporsional antar *card*, memudahkan pengguna untuk melihat dan memilih *hero* yang ingin mereka ketahui informasinya. Desain yang konsisten untuk setiap *card hero* memberikan pengalaman visual yang teratur dan profesional, sehingga pengguna dapat dengan mudah *browsing* dan mengeksplorasi berbagai *hero* yang tersedia dalam game *Mobile Legends*.

### C Halaman Forum Komunitas

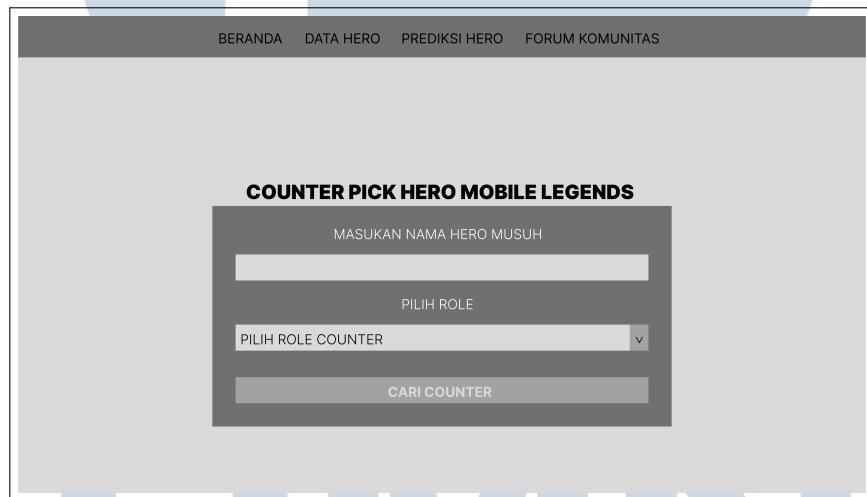


Gambar 3.9. Halaman Forum Komunitas

Gambar 3.9 menampilkan Halaman Forum Komunitas yang dirancang sebagai wadah interaksi dan diskusi antar pengguna *website*. Halaman ini memiliki *header* dengan judul besar FORUM KOMUNITAS dan subjudul yang menjelaskan tujuan halaman: "DISKUSI TENTANG COUNTERPICK TERBAIK BERDASARKAN PENGALAMAN PEMAIN". Pada bagian utama halaman, terdapat *form input* yang memungkinkan pengguna untuk berpartisipasi dalam diskusi. *Form* ini terdiri dari dua *field input* utama: *field* pertama bertuliskan "MASUKAN USERNAME" untuk pengguna memasukkan nama atau identitas mereka, dan *field* kedua yang lebih besar bertuliskan "TULISKAN PERTANYAAN SEPUTAR COUNTER PICK" yang berfungsi sebagai area teks untuk menuliskan pertanyaan, komentar, atau pengalaman mereka terkait *counter pick hero*. Di bawah kedua *field* tersebut, terdapat tombol KIRIM PERTANYAAN yang digunakan untuk mengirimkan *posting* atau komentar ke

forum. Bagian bawah halaman menampilkan contoh *postingan* dari pengguna dengan format yang terstruktur. Setiap *postingan* menampilkan "USERNAME:" diikuti nama pengguna (contoh: ASRUL111), kemudian "PERTANYAAN:" diikuti isi komentar atau pertanyaan pengguna. Pada contoh yang ditampilkan, terdapat komentar "SANGAT MEMBANTU MENGGUNAKAN WEBSITE INI ....." yang menunjukkan testimoni positif dari pengguna. Di bawah setiap *postingan*, tersedia tombol interaksi LIKE dan BALAS yang memungkinkan pengguna lain untuk memberikan respons atau terlibat dalam diskusi. Desain halaman ini mendorong interaksi komunitas dan berbagi pengalaman antar pemain *Mobile Legends*.

#### D Halaman Prediksi Hero



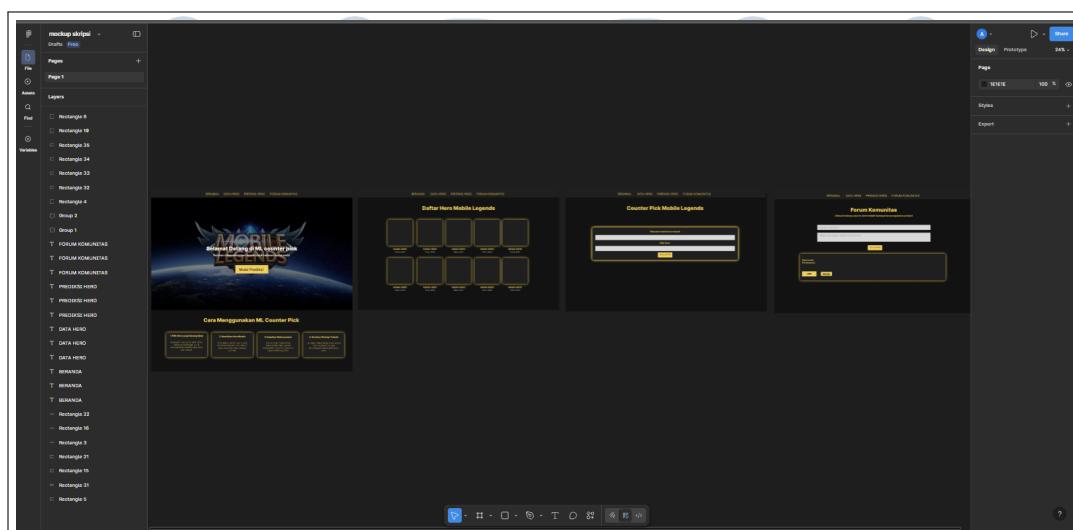
Gambar 3.10. Halaman Prediksi Hero

Gambar 3.10 menampilkan Halaman Prediksi Hero yang merupakan fitur inti dari *website* ini. Halaman ini memiliki judul besar di bagian atas COUNTER PICK HERO MOBILE LEGENDS yang menjelaskan fungsi utama halaman tersebut. Bagian utama halaman terdiri dari *form input* yang memungkinkan pengguna untuk mencari *hero counter* dengan cara yang lebih terstruktur. *Form* ini memiliki tiga komponen utama: pertama, *field input* teks dengan label "MASUKAN NAMA HERO MUSUH" yang berfungsi untuk pengguna memasukkan nama *hero* lawan yang ingin di-*counter*; kedua, *dropdown menu* dengan label "PILIH ROLE" yang menampilkan opsi "PILIH ROLE COUNTER" untuk memfilter hasil *counter* berdasarkan *role* atau kelas *hero* tertentu yang diinginkan; dan ketiga, tombol CARI COUNTER yang ditempatkan di bagian bawah *form* untuk menjalankan proses

pencarian. Desain halaman ini menggunakan kotak besar berwarna abu-abu gelap sebagai *container* untuk *form input*, menciptakan fokus visual yang jelas pada area interaksi utama. Pendekatan ini membuat halaman lebih *user-friendly* karena pengguna dapat dengan mudah memahami langkah-langkah yang perlu dilakukan: memasukkan nama *hero* musuh, memilih *role counter* yang diinginkan (opsional), dan menekan tombol pencarian untuk mendapatkan rekomendasi *hero counter* yang sesuai. Setelah tombol CARI COUNTER ditekan, sistem akan memproses *input* dan menampilkan hasil rekomendasi *hero* yang efektif untuk melawan *hero* musuh yang telah diinput.

### 3.3.6 Perancangan Antarmuka Pengguna (UI/UX)

Perancangan antarmuka pengguna (*User Interface/User Experience*) pada sistem rekomendasi *counter-pick* hero *Mobile Legends* dilakukan menggunakan *Figma* sebagai alat bantu pembuatan *wireframe* dan *prototype*. Perancangan ini menerapkan prinsip *UI/UX* berupa *consistency*, *ease of use*, dan *simplicity* guna memastikan keseragaman tampilan, kemudahan penggunaan, serta desain yang sederhana dan berfokus pada fungsi utama sistem. Hasil rancangan antarmuka kemudian diimplementasikan ke dalam sistem berbasis web menggunakan framework *Django* sehingga menghasilkan antarmuka yang interaktif.



Gambar 3.11. Perancangan Figma

Pada Gambar 3.11 menunjukkan rancangan antarmuka pengguna sistem rekomendasi *counter-pick* hero *Mobile Legends* yang dibuat menggunakan *Figma*.

Rancangan ini digunakan sebagai *wireframe* dan *prototype* awal sebelum tahap implementasi ke dalam sistem berbasis web.

### 3.4 Pembuatan Sistem

Pada tahapan ini, hal yang dilakukan selanjutnya adalah membuat sistem berdasarkan tahapan pada perancangan sistem yang sudah dibuat. Pada tahap ini, pembuatan sistem akan menerapkan metode yang digunakan yaitu *Content-Based Filtering* (CBF) pada pembuatan sistemnya.

### 3.5 Pengujian dan Evaluasi

Setelah melakukan pembuatan sistem, hal selanjutnya adalah melakukan evaluasi pada sistem. Jika terdapat kekurangan atau *bug* pada sistem, maka akan dilakukan perbaikan pada sistem tersebut. Evaluasi juga akan dilakukan dengan melakukan pengecekan pada perhitungan yang telah sistem lakukan dengan perhitungan manual untuk memverifikasi hasil perhitungan yang dihasilkan.

Perhitungan manual dilakukan dengan mengambil Vektor Kelemahan ( $V_{weak}$ ) musuh dan Vektor Kemampuan ( $V_{has}$ ) calon *counter*, kemudian menghitung *Cosine Similarity* di antara keduanya untuk dibandingkan dengan hasil sistem. Evaluasi mengenai tingkat kepuasan juga akan dilakukan dengan menggunakan metode *End User Computing Satisfaction*, dengan cara memberikan pertanyaan dalam *Google Form* kepada kalangan mahasiswa khususnya yang bermain *Mobile Legends*.

### 3.6 Penulisan Laporan

Pada tahap ini, dilakukan penyusunan laporan penelitian yang mendokumentasikan secara rinci setiap langkah dalam proses perancangan, pengembangan, dan evaluasi sistem. Laporan ini dirancang untuk memberikan gambaran menyeluruh mengenai penelitian, dimulai dari latar belakang, tujuan, dan metodologi hingga hasil pengujian dan analisis. Setiap tahapan dijelaskan secara kronologis dan sistematis agar mudah dipahami oleh pembaca. Selain itu, berbagai elemen pendukung seperti diagram, tabel, dan visualisasi disertakan untuk memperjelas informasi dan memperkuat penyampaian hasil. Penulisan laporan ini mengutamakan kejelasan dan ketepatan dalam penyajian, dengan mengikuti panduan akademik yang berlaku. Proses revisi dan validasi dilakukan secara berkala untuk memastikan laporan memenuhi standar ilmiah dan relevansi

penelitian. Laporan ini tidak hanya berfungsi sebagai dokumentasi akhir, tetapi juga diharapkan menjadi referensi yang bermanfaat bagi penelitian di masa mendatang.

### 3.7 Perhitungan Tingkat Kepuasan Pengguna

Untuk menghitung tingkat kepuasan pengguna, *End User Computing Satisfaction* (EUCS) digunakan, yang membutuhkan survei pengguna untuk mengetahui seberapa puas mereka dengan aspek penggunaan. Pertanyaan survei melibatkan elemen seperti desain antarmuka pengguna, penjelasan informasi, kemudahan pengguna, dan kepuasan umum pengguna terhadap *website*. Hasil survei akan dihitung untuk mengetahui seberapa efektif aplikasi dalam memenuhi kebutuhan dan memberikan kemudahan bagi pengguna dan admin.

