

Aplikasi Management Dan Rekomendasi Atlet Pada Pengurus Besar Ju-Jitsu Kota Surabaya Berbasis Website Dengan Metode Naïve Bayes

Bara Laily Mubarak¹, Bonda Sisephaputra²

^{1,2}Jurusan Teknik Informatika, Universitas Negeri Surabaya

¹baralaily.19045@mhs.unesa.ac.id

²bondasisephaputra@unesa.ac.id

Abstrak— Perkembangan teknologi informasi telah membawa perubahan signifikan dalam berbagai bidang, termasuk olahraga. Pendataan atlet yang dilakukan secara manual pada Pengurus Besar Ju-Jitsu Indonesia (PBJI) Kota Surabaya sering menimbulkan masalah seperti data yang tidak valid, duplikasi, dan memakan waktu lama. Untuk mengatasi permasalahan ini, penelitian ini mengembangkan aplikasi manajemen dan rekomendasi atlet berbasis website menggunakan metode Naïve Bayes untuk mendukung proses seleksi atlet berprestasi. Sistem rekomendasi menggunakan metode Naïve Bayes yang memanfaatkan data prestasi atlet untuk menghasilkan rekomendasi yang akurat. Selain itu, sistem ini dibandingkan dengan pendekatan berbasis query filter untuk menilai kelebihan masing-masing metode berdasarkan akurasi, kecepatan, dan kemudahan implementasi.

Hasil penelitian menunjukkan bahwa metode Naïve Bayes memiliki tingkat akurasi yang sama dengan query filter, dan memerlukan waktu pemrosesan yang lebih lama. Aplikasi ini telah diuji menggunakan metode Blackbox Testing dan User Acceptance Testing (UAT) dengan hasil bahwa semua fitur berfungsi sesuai kebutuhan pengguna. Aplikasi ini diharapkan dapat meningkatkan efisiensi dan akurasi dalam manajemen dan seleksi atlet berprestasi di PBJI Kota Surabaya.

Kata Kunci: Naïve Bayes, Query Filter, Manajemen Atlet, RAD, Rekomendasi, PBJI Kota Surabaya.

I. PENDAHULUAN

Perkembangan teknologi yang sangat pesat telah membantu manusia dalam membuat, menyimpan, mengubah, dan menyebarkan informasi. Pemanfaatan teknologi informasi dapat diterapkan di berbagai bidang contohnya pada bidang olahraga, teknologi informasi memudahkan dalam mendukung manajemen dan kegiatan operasional. Pada setiap kegiatan olahraga dapat memanfaatkan sistem informasi, salah satunya adalah kegiatan pendataan atlet. Pendataan atlet yang dirancang secara online akan sangat memudahkan karena penggunaannya yang praktis dan efisien, serta mengurangi kesalahan pada saat memasukkan data atlet.

Ju-Jitsu merupakan sebuah bela diri yang bersifat defensif dengan memanfaatkan kekuatan lawan untuk melakukan

teknik membanting maupun mengunci musuh [1]. Pengurus Besar Ju-Jitsu Indonesia atau biasa dipanggil dengan PBJI Kota Surabaya merupakan organisasi beladiri Ju-Jitsu pada tingkat kabupaten dan kota yang bertugas mengatur dan mengelola perguruan beladiri dibawah naungannya. PBJI Kota Surabaya menaungi banyak perguruan beladiri dengan berbagai aliran di dalamnya.

Proses pendataan atlet pada PBJI Kota Surabaya selama ini dilakukan secara manual, yaitu pengurus perguruan yang berada dibawah naungan PBJI Kota Surabaya menyerahkan berkas atlet mereka, kemudian admin dari PBJI Kota Surabaya akan melakukan pendataan menggunakan aplikasi Ms Excel. Seiring naiknya anggota dari perguruan maka data yang harus dimasukkan semakin banyak, sehingga memerlukan banyak waktu dan tenaga, sering juga data yang dimasukkan salah, dobel, atau tertukar dengan atlet lain bahkan ada juga yang datanya tidak terinput dan hilang. Sehingga data yang diperoleh menjadi tidak valid. Data atlet ini berfungsi sebagai filter untuk mencegah atlet yang bukan berasal dari bela diri Ju-Jitsu mengikuti kegiatan kejuaraan, yang dimaksudkan untuk menyaring atlet-atlet berprestasi. Data juga berfungsi untuk mencari atlet yang berprestasi guna dilakukan seleksi untuk mengikuti kejuaraan tingkat provinsi, nasional, bahkan internasional. Karena data yang tidak valid, sehingga pengurus PBJI Kota Surabaya tidak memiliki rekomendasi atlet berdasarkan prestasi mereka yang didapatkan, sehingga pengurus perlu melakukan seleksi masal setiap kali akan diadakan kejuaraan tingkat provinsi dan nasional. Oleh karena itu, pendataan atlet secara konvensional dinilai tidak efisien. Pendataan dan management data atlet yang baik dapat membantu efisiensi waktu dalam pencarian data penentuan atlet yang akan ditandingkan dalam kejuaraan. Untuk itu PBJI Kota Surabaya membutuhkan sebuah aplikasi manajemen dan rekomendasi atlet yang memudahkan pihak perguruan, atlet maupun pihak PBJI sendiri. Aplikasi manajemen berbasis online memiliki banyak keuntungan yaitu efisiensi pembiayaan dan tenaga dalam melaksanakan pendataan atlet, memudahkan dalam seleksi data atlet,

memudahkan dalam mencari atlet berprestasi. memudahkan atlet untuk mendaftar sebuah kejuaraan, menjadi sumber informasi tentang Ju-Jitsu khususnya di Kota Surabaya.

Fitur yang akan dibangun dalam sistem informasi ini meliputi penginputan data atlet, penginputan data perguruan, penginputan data prestasi, upload sertifikat, detail sertifikat, cetak sertifikat perguruan, detail data atlet, detail data perguruan, penginputan berita, penginputan kegiatan, pendaftaran kegiatan, penjurian kegiatan, pembayaran kegiatan, rekomendasi atlet. Sistem informasi pendataan atlet berbasis website menjadi pilihan terbaik dalam mengatasi permasalahan pada Organisasi PBJI Kota Surabaya.

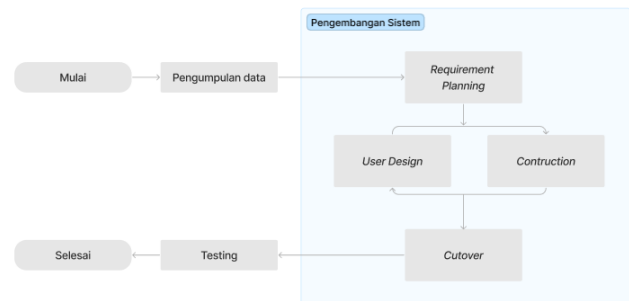
Aplikasi berbasis website memungkinkan mempermudah pengguna dalam pengeoperasiannya, karena website mudah diakses berbagai macam perangkat, sehingga semua informasi dapat mudah dan cepat diterima oleh pengguna. Sistem informasi berbasis website juga dapat membantu dalam pengelolaan dan pengolahan data, sehingga data yang didapatkan lebih akurat dibandingkan dengan data yang diolah secara manual.

Aplikasi management dan rekomendasi atlet Ju-Jitsu pada PBJI Kota Surabaya akan dikembangkan menggunakan metode Naïve Bayes dalam pembuatan rekomendasi atlet Ju-Jitsu. Metode Naïve Bayes merupakan metode klasifikasi sederhana menggunakan probabilitas dan statistik untuk memprediksi peluang di masa depan menggunakan data lampau dimana suatu objek didefinisikan bersifat bebas [2][3]. Metode ini dapat berfungsi untuk pembuatan rekomendasi atlet pada PBJI Kota Surabaya. Metode ini dipilih karena mudah dalam penggunaannya dan juga memiliki akurasi yang tinggi dibandingkan dengan metode yang lainnya [2]. Metode ini dapat berjalan dengan dataset yang kecil dan dapat berkembang dengan bertambahnya data yang dimasukkan, sehingga metode ini sangat cocok jika digabungkan dengan aplikasi management atlet, dimana data atlet dan data prestasi yang bertambah terus menerus.

Berdasarkan permasalahan diatas mendorong penulis untuk merancang dan membangun aplikasi berbasis website pada PBJI Kota Surabaya dengan menggunakan metode pengembangan Rapid Application Development (RAD), yang mana metode ini berfokus pada siklus waktu pengembangan dan fleksibilitas mengikuti kebutuhan pengguna serta membuat usulan penelitian “Aplikasi Management dan Rekomendasi Atlet pada Pengurus Besar Ju-Jitsu Kota Surabaya Berbasis Website Dengan Metode Naïve Bayes” dengan harapan aplikasi tersebut dapat membantu organisasi untuk meningkatkan mutu pelayanan [4][5].

II. METODE PENELITIAN

Penelitian dilakukan dengan beberapa tahap. Tahap pertama berupa pengumpulan data, kemudian dilakukan analisis kebutuhan. Setelah itu dilanjutkan dengan pengembangan system dan diakhiri dengan pengujian. Berikut gambar alur dari tahapan penelitian:



Gbr. 1 Tahapan Penelitian

A. Pengumpulan Data

Pengumpulan data dilakukan dengan cara yaitu studi lapangan dan studi literatur. Dari kedua data tersebut akan diolah dan dianalisa menjadi kebutuhan fungsional sebuah aplikasi.

1. Studi Lapangan

Studi lapang dilakukan dengan wawancara langsung kepada Ketua PBJI Kota Surabaya. Wawancara dilakukan guna mendapatkan informasi terkait kebutuhan dalam membangun sistem informasi pendataan atlet Ju-Jitsu di PBJI Kota Surabaya. Dari hasil wawancara akan disusun fitur yang sesuai dengan permasalahan dan kebutuhan organisasi, sehingga aplikasi dapat berjalan sesuai dengan kebutuhan.

2. Studi Literatur

Studi literatur dilakukan dengan pengumpulan data dan teori pada buku, makalah, jurnal, artikel ilmiah, sebagai pendukung serta mencari referensi untuk membangun system informasi berbasis website menggunakan metode pengembangan perangkat lunak RAD. Studi literatur juga difungsikan untuk mengetahui bagaimana cara penerapan metode RAD untuk pengembangan website dan metode Naïve Bayes dalam pembuatan sistem rekomendasi atlet.

Berikut hasil wawancara mengenai ketentuan aplikasi yang terjadi.

1. Sistem dapat mengelola 4 tingkatan user (Anggota Perguruan, Admin Perguruan, Admin Aliran Perguruan, Admin PBJI Kota Surabaya).
2. Sistem dapat mengelola dan menyimpan data atlet yang dimasukkan oleh admin perguruan.

3. Sistem dapat mengelola dan menyimpan data prestasi atlet.
4. Sistem dapat merekomendasikan atlet yang berprestasi.
5. Sistem dapat mengelola kegiatan yang diadakan oleh PBJI Kota Surabaya dan Admin aliran perguruan.
6. Sistem dapat mengelola berita yang dimasukkan oleh PBJI Kota Surabaya dan Admin aliran perguruan.
7. Sistem berbasis website

B. Pengembangan Sistem

Pada pengembangan sistem informasi pendataan atlet Ju-Jitsu di PBJI Kota Surabaya akan menerapkan metode Rapid Application Development (RAD). RAD sendiri merupakan bagian dari System Development Life Cycle (SDLC).

1. Rapid Application Development (RAD)

RAD atau Rapid Application Development merupakan metode pengembangan perangkat lunak yang fokus pada siklus waktu pengembangan dengan mengadaptasi dari model waterfall dimana pengembangan dilakukan dengan pendekatan konstruksi berbasis komponen [4][5].

Aplikasi yang digunakan untuk membantu mempercepat proses development adalah QuickAdminPanel. QuickAdminPanel adalah sebuah alat berbasis web yang memungkinkan pengembang untuk secara cepat membangun aplikasi Laravel dengan antarmuka berbasis Bootstrap. Alat ini dirancang untuk mempermudah pembuatan panel administrasi dan aplikasi CRUD (Create, Read, Update, Delete) tanpa perlu menulis kode secara manual untuk setiap fungsionalitas. ini.

a. Requirement Planning

Pada tahap ini akan dilakukan analisis berdasarkan data yang diperoleh dari studi literatur dan studi lapangan. Setelah melakukan studi literatur dan lapangan, masalah-masalah yang muncul akan dianalisis sehingga system akan dibuat sesuai dengan permasalahan yang dihadapi user. Fase ini berakhir ketika sudah disetujui oleh pihak user dan pengembang [6].

Dari pengumpulan data ketentuan aplikasi, dapat dirumuskan fitur-fitur dalam aplikasi sebagai berikut:

1) Anggota Perguruan

Berikut list fitur dari user akses anggota perguruan

- Manajemen profile meliputi melihat dan mengubah data profile atlet. Data yang berubah perlu divalidasi oleh admin aliran perguruan
- Manajemen data prestasi meliputi melihat, menambah, mengubah dan menghapus data prestasi. Data prestasi divalidasi oleh admin aliran perguruan. Data prestasi yang valid digunakan untuk menentukan rekomendasi atlet

- Manajemen data ijazah meliputi melihat, menambah, mengubah dan menghapus data ijazah. Anggota perguruan dapat mengubah tingkatan sabuk jika data ijazah sudah diverifikasi oleh admin aliran perguruan
- Melihat data kegiatan meliputi anggota perguruan hanya dapat melihat riwayat kegiatan yang sudah diikuti.

2) Admin Perguruan

Berikut list fitur dari user akses admin perguruan

- Manajemen profile meliputi melihat dan mengubah data profile perguruan. Data yang berubah perlu divalidasi oleh admin aliran perguruan
- Manajemen anggota meliputi melihat, menambah, mengubah, menghapus, dan melakukan transfer data anggota. Admin perguruan juga dapat menambah data ijazah anggotanya. Data anggota divalidasi oleh admin aliran perguruan. Data anggota yang valid digunakan untuk menentukan rekomendasi atlet
- Manajemen kegiatan meliputi melihat, dan mengikuti kegiatan. Perguruan yang datanya sudah divalidasi oleh Admin aliran perguruan dan sertifikat perguruannya sudah disahkan oleh Admin PBJI Kota Surabaya, dapat mendaftarkan anggotanya untuk mengikuti kegiatan.

3) Admin Aliran Perguruan

Berikut list fitur dari user akses admin aliran perguruan

- Manajemen perguruan meliputi melihat, memvalidasi data perguruan yang terdaftar, dan memblokir perguruan yang sudah tidak aktif.
- Manajemen anggota perguruan meliputi melihat, dan memvalidasi data anggota perguruan. Anggota yang sudah divalidasi dapat mengikuti kegiatan yang dibuat oleh admin PBJI Kota Surabaya maupun Admin Aliran Perguruan
- Manajemen data ijazah meliputi melihat, dan memvalidasi ijazah kenaikan sabuk anggota perguruan. Data ijazah yang sudah divalidasi dapat digunakan oleh anggota perguruan untuk mengubah tingkatan sabuk.
- Manajemen data prestasi meliputi melihat, memvalidasi, memblokir data prestasi anggota perguruan. Data prestasi yang valid akan digunakan untuk proses rekomendasi atlet.

- Manajemen kegiatan Melihat meliputi membuat, Mengubah, dan Menghapus kegiatan. Kegiatan yang sudah memiliki pendaftar tidak dapat dihapus.
- Manajemen berita meliputi melihat, Membuat, Mengubah, dan Menghapus berita.

4) Admin PBJI Kota Surabaya

Berikut list fitur dari user akses admin PBJI Kota Surabaya

- Manajemen perguruan dan anggota perguruan meliputi melihat data perguruan, dan anggotanya
- Manajemen sertifikat perguruan meliputi melihat, memverifikasi, dan memblokir sertifikat perguruan.
- Manajemen kegiatan meliputi membuat membuat, Mengubah, dan Menghapus kegiatan. Kegiatan yang sudah memiliki pendaftar tidak dapat dihapus.
- Manajemen berita Melihat meliputi membuat, Mengubah, dan Menghapus berita.

b. Design

Pada tahap ini akan dibangun prototipe dengan menggabungkan teknik join application design (J.A.D) untuk menerjemahkan keinginan user yang mudah untuk dipahami oleh user, sehingga user dapat memberikan saran dan memodifikasi sesuai kebutuhan [6].

Pada tahap Design user dapat terjun langsung dan berkomunikasi dengan developer untuk menyocokkan alur aplikasi dan tampilan aplikasi, sehingga mengurangi perubahan dan kesalahan waktu tahap berikutnya.

c. Construction

Fase ini merupakan fase pembuatan program berdasarkan design yang telah sesuai dengan kebutuhan dan disepakati oleh user [6].

Pada fase ini apabila developer memiliki dalam pengembangan dapat langsung bernegosiasi dengan user begitu sebaliknya, ketika user ingin melakukan perubahan baik tampilan maupun alur dapat langsung bernegosiasi dengan developer.

d. Cutover

Pada tahap ini user dapat melakukan pengujian fitur aplikasi, kemudian memberikan masukan terkait fitur tersebut, jika fitur sudah sesuai keinginan maka akan dilanjutkan ke proses production [4][6].

Apabila pada tahap ini aplikasi belum berjalan sempurna maka akan mundur ke fase contruction. Tahap ini juga mencakup aktivitas seperti memindahkan sistem ke lingkungan produksi serta pengujian akhir untuk memastikan bahwa aplikasi siap digunakan dalam skenario nyata. Tujuan utamanya adalah menjamin sistem sesuai kebutuhan pengguna, baik dari segi desain maupun fungsionalitas.

2. Naïve Bayes

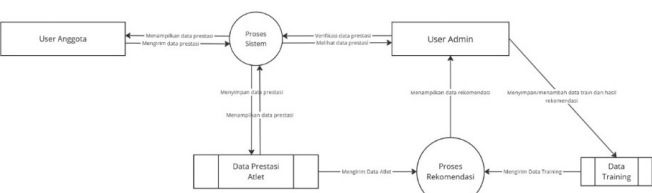
Metode Naïve Bayes merupakan metode klasifikasi sederhana menggunakan probabilitas dan statistik untuk memprediksi peluang di masa depan menggunakan data lampau dimana suatu objek didefinisikan bersifat bebas [2][3]. Maksud bersifat bebas adalah bahwa sebuah data tidak berkaitan dengan ada atau tidaknya data yang sama. Penggunaan metode ini digunakan untuk pendukung keputusan yang dapat membantu menentukan atlet berprestasi, yang layak diusulkan mewakili Surabaya dalam kompetisi PORPROV (Pekan Olahraga Provinsi). Sistem ini bertujuan untuk mempermudah seleksi atlet, mempercepat proses, dan meningkatkan keakuratan rekomendasi.

Proses pengolahan data dimulai dengan tahap data cleaning, yaitu memastikan data prestasi atlet yang dimasukkan telah diverifikasi oleh Admin Aliran Perguruan. Tahap ini bertujuan untuk memastikan validitas dan konsistensi data sehingga model Naïve Bayes dapat beroperasi dengan optimal. Setelah data bersih, sistem melanjutkan ke tahap training, di mana pola dari data prestasi atlet dipelajari oleh model untuk menghasilkan prediksi. Terakhir, sistem melakukan proses predict untuk memberikan rekomendasi berdasarkan data masukan terbaru.

Sebagai pembanding, sistem rekomendasi berbasis query filter menggunakan aturan logis tetap yang diterjemahkan dalam SQL untuk menyaring data. Meskipun lebih sederhana dalam implementasi, query filter tidak memiliki kemampuan adaptasi terhadap pola data baru seperti yang dimiliki Naïve Bayes. Dalam skripsi ini, kedua pendekatan akan dibandingkan berdasarkan tiga parameter utama: kecepatan proses, akurasi hasil, dan relevansi rekomendasi dengan kebutuhan pengguna.

Parameter yang akan digunakan dalam menentukan rekomendasi anatara lain:

1. Usia
2. Tingkatan Sabuk
3. Jumlah Partisipasi Kejurkab/Kejurkot
4. Jumlah Juara 1 Kejurkab/Kejurkot
5. Jumlah Juara 2 Kejurkab/Kejurkot

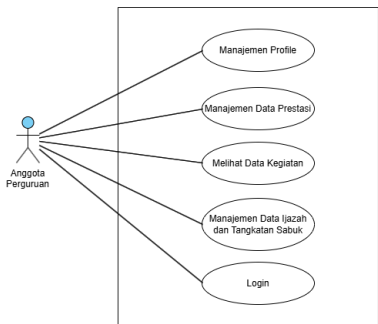


Gbr. 3 Data Flow Diagram Rekomendasi Atlet

3. Use Case Diagram

Dari rumusan fitur-fitur diatas dapat digambarkan Use Case Diagram seperti dibawah ini:

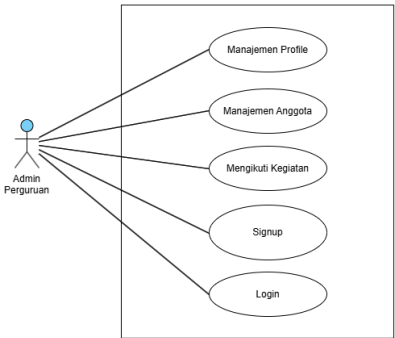
a. Anggota Perguruan



Gbr. 4 Use Case Diagram Anggota Perguruan

Anggota perguruan dapat melakukan login, yang kemudian bisa untuk mengakses fitur manajemen profile, manajemen data prestasi, manajemen data ijazah, dan melihat riwayat kegiatan.

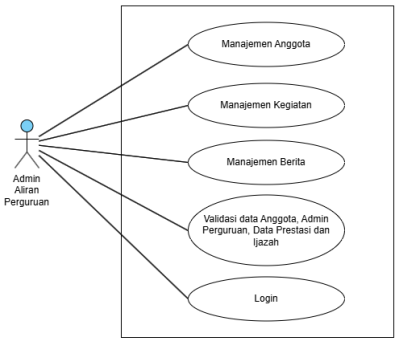
b. Admin Perguruan



Gbr. 5 Use Case Diagram Admin Perguruan

Admin Perguruan dapat memiliki akses untuk fitur manajemen profile, manajemen anggota, manajemen profile, mengikuti kegiatan login, dan signup. Fitur signup hanya terdapat pada Admin Perguruan.

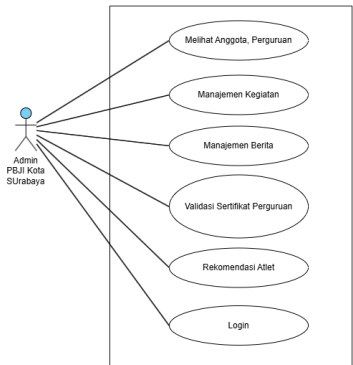
c. Admin Aliran Perguruan



Gbr. 6 Use Case Diagram Admin Aliran Perguruan

Admin aliran perguruan mendapatkan akses untuk fitur manajemen anggota, manajemen kegiatan, manajemen berita, validasi semua data yang dimasukkan oleh admin perguruan.

d. Admin PBJI Kota Surabaya



Gbr. 7 Use Case Diagram Admin PBJI Kota Surabaya

Admin PBJI Kota Surabaya memiliki akses ke fitur melihat data anggota dan admin perguruan, manajemen kegiatan, manajemen berita, memvalidasi sertifikat perguruan, dan rekomendasi atlet.

B. Pengembangan Website

Setelah selesai pada tahap perumusan fitur-fitur aplikasi dan user desain, langkah berikutnya yaitu *construction*.

1. Penggunaan *QuickAdminPanel* dalam Pembuatan Fitur

Penggunaan aplikasi *QuickAdminPanel* dimaksudkan untuk mempercepat proses pengembangan website. Berikut langkah-langkah dalam membuat project di aplikasi *QuickAdminPanel* dan contoh dalam pembuatan salah satu fitur pada aplikasi:

a. Pembuatan Project

Pembuatan project dilakukan dengan mengunjungi website quickadminpanel.com kemudian login dengan user yang telah didaftarkan. Pada halaman dashboard pilih New Project, Pada pop up menu yang muncul pilih Laravel 10 + jQuery. Kemudian akan diarahkan pada halaman pembuatan project. Isi data project sesuai kebutuhan. Setelah diisi semua data tekan tombol; create, tunggu hingga project berhasil dibuat.

b. Pembuatan Halaman

Untuk pembuatan halaman pertama-tama tekan tombol Create CRUD Menu pada dashboard project yang telah dibuat. Pada Halaman yang muncul isi semua form sesuai kebutuhan dari tampilan web yang ingin di buat. Pada bagian setting digunakan untuk mengatur nama model yang akan dikonversi menjadi nama tabel padadatabase, dan juga terdapat pengaturan untuk menu nama menu di sidebar. Pada bagian ini juga terdapat pengaturan untuk mengaktifkan atau menonaktifkan tampilan CRUD-nya. Selanjutnya Pada bagian field digunakan untuk mengatur kolom pada tabel sekaligus akan dibuat untuk tampilan list, dan juga form create dan update.

2. Desain dan Pengembangan Halaman-Halaman Web

Setelah pembuatan menggunakan *QuickAdminPanel*project bisa disempurnakan manual dengan mengunduh kode kemudian kode diubah secara langsung agar hasil yang didapatkan sesuai dengan desain yang sudah ditentukan. Berikut merupakan hasil dari antarmuka website yang telah dibuat:

a. Homepage

Homepage bertujuan untuk memberikan informasi singkat mengenai website. Homepage menyajikan informasi singkat berita atau kegiatan yang sedang terbaru, dan juga berisi penjelasan singkat mengenai PBJI Kota Surabaya.

b. List Kegiatan

Halaman ini dibuat untuk memberikan daftar semua kegiatan baik yang sudah berlangsung atau sudah selesai. Semua kegiatan yang dibuat oleh admin aliran perguruan maupun admin PBJI Kota Surabaya secara otomatis akan di tampilkan pada halaman ini.

c. Detail Kegiatan

Halaman ini dibuat dengan tujuan untuk menampilkan seluruh informasi event/kegiatan yang dibuka oleh pengguna. Halaman ini berisi banner/foto kegiatan, deskripsi kegiatan, kategori peserta dalam mengikuti kegiatan, perguruan yang sudah terdaftar, dan informasi mengenai tanggal dan status kegiatan. Apabila perguruan sudah terverifikasi dan kegiatan masih berlangsung, akan muncul tombol daftar untuk mendaftarkan anggota perguruan.

d. List Berita

Halaman ini digunakan untuk menampilkan semua berita yang telah dibuat oleh admin aliran perguruan maupun admin PBJI Kota Surabaya. Pengguna dapat melihat detail dari berita dengan menekan pada salah satu berita yang tersedia.

e. Detail Berita

Halaman ini merupakan detail dari berita. Informasi yang disediakan berupa banner/foto berita, judul, deskripsi, dan 5 berita terbaru.

f. Cari Ijazah

Halaman ini dibuat untuk mencari ijazah sabuk dari anggota perguruan. Halaman ini bersifat publik jadi semua pengguna dapat mencari ijazah yang sudah dimasukkan ke dalam aplikasi. Data yang ditampilkan berupa foto ijazah, nomor ijazah, dan detail dari pemilik ijazah.

g. Detail Aliran Perguruan

Halaman ini berisi tentang aliran perguruan yang terdaftar resmi di PBJI Kota Surabaya. Aliran yang terdaftar ada 2 yaitu IJI (Institut Jujitsu Indonesia) dan IJUKAI (Indonesian Ju-jitsu Kai). Pada halaman ini juga terdapat informasi jumlah anggota perguruan, jumlah

perguruan, dan jumlah kegiatan yang sudah dilaksanakan oleh aliran perguruan tersebut.

h. Login

Halaman ini digunakan pengguna yang terdaftar untuk masuk ke menu dashboard. Pengguna diwajibkan untuk mengisi email dan juga password, yang kemudian akan diverifikasi dengan data yang sudah tersimpan pada database. Jika data tidak ada atau tidak sesuai dengan yang ada di database akan memunculkan notifikasi error. Setelah pengguna masuk, pengguna akan mendapatkan akses berdasarkan role yang sudah didaftarkan.

i. Register

Halaman daftar digunakan untuk admin perguruan mendaftarkan perguruanannya. Admin perguruan diwajibkan untuk mengisi email, aliran perguruan, dan juga password. Setelah pengguna terdaftar, pengguna akan diarahkan ke halaman dashboard dengan status belum aktif. Pengguna wajib melengkapi data perguruan untuk proses validasi oleh admin perguruan.

j. Dashboard (Anggota Perguruan)

Halaman ini berisi profile dari anggota perguruan. Pada halaman ini juga terdapat status dari akun anggota, dimana jika status belum aktif maka anggota tidak dapat mengakses fitur-fitur yang lainnya. Pada halaman ini juga terdapat fitur untuk mengubah password. Setelah anggota login untuk pertama kali disarankan untuk mengubah password default yang terbuat otomatis dari system.

k. Prestasi (Anggota Perguruan)

Halaman ini digunakan para anggota untuk menambahkan prestasi yang telah dibuat. Anggota bisa menambahkan prestasi dengan menekan tombol plus yang ada pada pojok atas kanan pada bagian data prestasi.

Pada form yang muncul, anggota perguruan wajib mengisi semua data dengan benar, kemudian tekan tombol submit. Setelah data prestasi ditambah, akan muncul pada list data prestasi dengan status verifikasi. Data yang sudah ditambah bisa dihapus atau diedit sebelum terverifikasi oleh Admin Aliran Perguruan.

l. Riwayat Kegiatan (Anggota Perguruan)

Halaman ini menampilkan kegiatan yang sudah pernah diikuti oleh anggota perguruan. Anggota perguruan dapat melihat detail kegiatan dengan menekan nama kegiatannya.

m. Riwayat Ujian (Anggota Perguruan)

Halaman ini digunakan untuk menambahkan data ijazah sabuk anggota perguruan. Ijazah bisa ditambah dengan menekan tombol plus pada pojok kanan atas bagian Riwayat Ujian. Setelah menekan tombol plus maka akan muncul form yang wajib diisi.

Pada form yang muncul anggota wajib mengisi semua data dan menekan tombol submit untuk menyimpan data. Data yang sudah disimpan akan muncul pada list riwayat ujian dengan status verifikasi. Setelah Ijazah diverifikasi oleh Admin Perguruan, ijazah dapat digunakan untuk memperbarui data sabuk anggota. Untuk memperbarui data sabuk, bisa menekan tombol panah atas pada bagian Riwayat Ujian. Pada form yang muncul peserta bisa memilih tingkatan sabuk yang telah diverifikasi dan menekan tombol submit.

n. Dashboard (Admin Perguruan)

Halaman ini berisi profile dari Perguruan dan juga terdapat informasi status dari akun dan sertifikat perguruan. Jika akun tidak terverifikasi maka tidak dapat akses untuk semua fitur Admin Perguruan yang ada pada aplikasi. Sedangkan jika sertifikat perguruan tidak terverifikasi oleh Admin PBJI Kota Surabaya, Admin Perguruan tidak dapat mendaftarkan anggotanya pada kegiatan yang tersedia. Pada halaman ini juga terdapat fitur untuk mengubah password. Ketika pertama kali masuk halaman ini setelah register, admin perguruan wajib untuk melengkapi data perguruan yang nanti akan diverifikasi oleh admin aliran perguruan. Setelah akun terverifikasi, admin perguruan diwajibkan untuk mencetak sertifikat perguruan. Setelah dicetak sertifikat, perguruan diminta untuk pergi ke kantor PBJI Kota Surabaya guna mengesahkan sertifikat tersebut. Ketika sertifikat disahkan, Admin dari PBJI Kota Surabaya juga akan memverifikasi sertifikat yang ada pada aplikasi.

o. Riwayat Kegiatan (Admin Perguruan)

Halaman ini berisi daftar riwayat kegiatan yang telah diikuti perguruan. Pada riwayat ini juga terdapat jumlah peserta di setiap kegiatan yang diikuti.

p. Riwayat Pembayaran (Admin Perguruan)

Halaman ini berisi Riwayat Pembayaran dari sebuah kegiatan. Verifikasi pembayaran dilakukan secara manual oleh penyelenggara kegiatan. Setelah melakukan pembayaran admin dapat mengupload bukti pembayaran pada tombol mata. Jika sudah terupload maka akan muncul tombol lihat bukti untuk melihat bukti yang telah diupload. Pembayaran yang belum dikonfirmasi bisa dibatalkan dan

secara otomatis juga membatalkan keikutsertaan pada kegiatan. Peserta yang dibatalkan pendaftarannya dapat didaftarkan ulang oleh admin. Pembayaran yang sudah dikonfirmasi tidak dapat dibatalkan.

q. Keanggotaan (Admin Perguruan)

Pada halaman ini berisi daftar anggota dari perguruan. Perguruan dapat menambah, mengubah, menonaktifkan, dan memblokir anggota. Perguruan juga bisa menghapus data dari atlet jika belum pernah mengikuti kegiatan apapun. Jika sudah pernah mengikuti kegiatan maka anggota hanya bisa diblokir untuk menonaktifkan akun. Perguruan juga memiliki fitur transfer anggota, dimana anggota perguruan dapat di pindah ke perguruan lain. Pada proses penambahan anggota, system akan otomatis membuat email dan password untuk anggota perguruan. Password dapat diubah oleh masing-masing anggota melalui menu dashboard. Pada proses mengubah data anggota, akan ada proses verifikasi ulang oleh admin aliran perguruan. Pada halaman ini juga terdapat informasi profile tentang perguruan. Informasi ini dapat diedit oleh admin perguruan tanpa perlu verifikasi ulang.

r. Dashboard (Admin Aliran Perguruan)

Halaman ini berisi kegiatan dan berita yang telah dibuat. Admin Aliran Perguruan bisa mengelola kegiatan dan berita pada halaman ini. Kegiatan yang sudah lama bisa diarsipkan agar tidak tampil pada halaman list kegiatan. Admin juga dapat membuat, mendownload data peserta kegiatan, memverifikasi pembayaran, mengedit dan juga menghapus data kegiatan. Hal yang sama juga berlaku untuk data berita, admin aliran perguruan dapat menambah, mengedit, dan menghapus berita.

s. Ijazah (Admin Aliran Perguruan)

Halaman ini berisi daftar ijazah yang telah diunggah oleh anggota perguruan. Admin bisa memverifikasi dan menolak data ijazah. Ijazah yang terverifikasi bisa digunakan oleh anggota untuk menaikkan level sabuk.

t. Data Prestasi (Admin Aliran Perguruan)

Halaman ini berisi daftar prestasi yang telah diunggah oleh anggota perguruan. Admin bisa memverifikasi dan menolak data prestasi, Data Prestasi yang terverifikasi secara otomatis akan digunakan pada system rekomendasi atlet.

u. Dojo/Perguruan (Admin Aliran Perguruan)

Halaman ini berfungsi untuk mengelola akun perguruan. Admin Aliran dapat memverifikasi, memblokir,

menonaktifkan sementara akun perguruan. Admin Aliran Perguruan juga dapat merest password akun perguruan.

v. Anggota Perguruan (Admin Aliran Perguruan)

Halaman ini berfungsi untuk mengelola akun anggota perguruan. Fitur-fitur pengelolaan mencakup memverifikasi, memblokir, menonaktifkan, dan menghapus data anggota. Anggota yang terverifikasi dapat mengikuti kegiatan dan dapat tampil pada menu rekomendasi.

w. Dashboard (Admin PBJI Kota Surabaya)

Halaman ini berisi statistik singkat dari total anggota perguruan, total perguruan, total ijazah, total kegiatan yang telah dibuat. Pada halaman ini juga terdapat fitur yang sama dengan admin aliran perguruan, dimana Admin PBJI Kota Surabaya juga mendapatkan akses untuk mengelola kegiatan dan berita.

x. Sertifikat Dojo/Perguruan (Admin PBJI Kota Surabaya)

Halaman ini berfungsi untuk memverifikasi atau memblokir sertifikat dari perguruan. Sertifikat Perguruan yang sudah di blokir dapat diverifikasi kembali.

y. Administrasi (Admin PBJI Kota Surabaya)

Halaman ini hanya berisi daftar anggota perguruan, perguruan, dan juga admin aliran perguruan. Karena akun dengan role Admin PBJI dibuat dengan tujuan utama untuk mengawasi apakah Aliran Perguruan di bawah kewenangannya dapat berjalan dengan baik.

C. Implementasi Sistem Rekomendasi

Pada tahap ini dilakukan implementasi sistem rekomendasi atlet menggunakan metode naïve bayes dan query filter.

1. Naïve Bayes

Sistem rekomendasi ini dibangun dengan bantuan *library* PHP Rubix/ML, yang merupakan alat pembelajaran mesin untuk pemrosesan data berbasis PHP. Proses pengembangan meliputi langkah-langkah berikut:

a. Persiapan Data/*Cleaning*

Data diambil dari prestasi peserta yang sudah divalidasi oleh Admin Aliran Perguruan. Kemudian menormalisasi data dengan mengisi data yang kosong dengan 0 dan menyamakan format yang dipakai dalam data train. Berikut kode yang dibuat dalam aplikasi.

```
$allAwards = UserAward::select(
    'user_id',
    'championship_type',
    'award',
    DB::raw('COUNT(*) as count'))
    ->where('date', '>=', Carbon::now()->subYears(2))
    ->where('status', 1);

$userAwards = UserAward::get();

$dataUser = [];
foreach ($userAwards as $user) {
    $userId = $user->user_id;
    $counts = [
        'region' => [0, 0, 0, 0],
        'province' => [0, 0, 0, 0],
        'porkab' => [0, 0, 0, 0],
        'open' => [0, 0, 0, 0],
    ];
    if (isset($allAwards[$userId])) {
        foreach ($allAwards[$userId] as $award) {
            $type = $award->championship_type;
            $awardValue = $award->award;
            $count = $award->count;
            switch ($type) {
                case 1: $category = 'region'; break;
                case 2: $category = 'province'; break;
                case 4: $category = 'porkab'; break;
                case 7: $category = 'open'; break;
                default: continue 2;
            }
            $counts[$category][$awardValue] = $count;
        }
    }
    $ageFormatted = Carbon::parse($user->date_of_birth)
        ->diffInYears(Carbon::now());
    $dataUser[] = [$ageFormatted, $user->belt_level];
    $dataUser[] = array_merge(
        $counts['region'],
        $counts['province'],
        $counts['porkab'],
        $counts['open']
    );
}
```

Gbr. 8 Kode Mengambil & memproses data prestasi user

b. Implementasi Naïve Bayes/Training

Pada proses ini pertama mengambil semua data yang ada pada table data train. Kemudian inisialisasi model dengan *classifier* yang ada pada *library* Rubix/ML. Masukkan data train yang sudah diambil diawal ke model untuk mempelajari pola klasifikasi. Model ini akan digunakan untuk menentukan rekomendasi.

```
// Mengambil data dari tabel data train
$dataTrain = RecommendationTrain::get();
$formattedTrain = [];
$formattedLabel = [];

// Cleaning data dan menormalisasi data training
foreach ($dataTrain as $value) {
    $x = [
        $value->age,
        $value->level,
        $value->total_win_1_region_competition,
        $value->total_win_2_region_competition,
        $value->total_win_3_region_competition,
        $value->total_participate_province_competition,
        $value->total_win_1_province_competition,
        $value->total_win_2_province_competition,
        $value->total_win_3_province_competition,
        $value->total_participate_porkab_competition,
        $value->total_win_1_porkab_competition,
        $value->total_win_2_porkab_competition,
        $value->total_win_3_porkab_competition,
        $value->total_participate_open_competition,
        $value->total_win_1_open_competition,
        $value->total_win_2_open_competition,
        $value->total_win_3_open_competition,
    ];
    $y = $value->is_recommended;
    switch ($value->is_recommended) {
        case 0: $y = 'nr'; break;
        case 1: $y = 'lr'; break;
        case 2: $y = 'r'; break;
        case 3: $y = 'mr'; break;
    }
    $formattedTrain[] = $x;
    $formattedLabel[] = $y;
}

// Memasangkan Label dan juga data train
$dataset= new Labeled($formattedTrain, $formattedLabel);

// Membuat model Naïve Bayes
$estimator = new GaussianNB();
$estimator->train($dataset);
```

Gbr. 9 Kode Pembuatan model dan implementasi Naive Bayes

c. Uji Model/Predict

Data yang sudah dibersihkan pada proses cleaning akan dilakukan uji dengan model yang telah dibuat dari training. Hasilnya akan berupa rekomendasi berdasarkan dengan tingkat kecocokan pada pola yang telah diajarkan pada tahap training. Hasil ini akan ditampilkan pada daftar rekomendasi atlet yang terdapat pada Dashboard Admin PBJI Kota Surabaya. Berikut kode dalam aplikasi

```

foreach ($dataUser as $key => $data) {
    $datasetSample = new Unlabeled([$data]);
    $predictions = $estimator->predict($datasetSample);
    $probs = $estimator->proba($datasetSample);

    $predictedNumber = 0;
    switch ($predictions[0]) {
        case 'nr': $predictedNumber = 0; break;
        case 'lr': $predictedNumber = 1; break;
        case 'r': $predictedNumber = 2; break;
        case 'mr': $predictedNumber = 3; break;
    }
    $normalized = normalizeData([
        $probs[0]['nr'],
        $probs[0]['lr'],
        $probs[0]['r'],
        $probs[0]['mr']
    ]);
    $userAwards[$key]->recomendation = [
        'sample' => implode(',', $data),
        'predictedNumber' => $predictedNumber,
        'normalized' => $normalized,
    ];

    $dataRecomendation = $userAwards->toArray();

    usort($dataRecomendation, function ($a, $b) {
        $proba = $a['recomendation']['predictedNumber'];
        $probb = $b['recomendation']['predictedNumber'];

        return $probb <=> $proba;
    });
}

```

Gbr. 10 Kode Uji coba model Naive Bayes

2. Query Filter

Query filter dilakukan dengan menyaring data secara langsung dari database menggunakan *syntax* SQL berdasarkan atribut yang telah ditentukan oleh user.

Tahap implementasinya dimulai dari pengambilan data prestasi user 2 tahun terakhir yang sudah diverifikasi oleh admin aliran perguruan. Kemudian data dinormalisasi dan diformat agar mudah pada proses labeling :

```

$allAwards = UserAward::select(
    'user_id',
    'championship_type',
    'award',
    DB::raw('COUNT(*) as count'))
    ->where('date', '>=', Carbon::now()->subYears(2))
    ->where('status', 1);

$userAwards = UserAward::get();

$dataUser = [];
foreach ($userAwards as $user) {
    $userId = $user->user_id;
    $counts = [
        'region' => [0, 0, 0, 0],
        'province' => [0, 0, 0, 0],
        'porkab' => [0, 0, 0, 0],
        'open' => [0, 0, 0, 0],
    ];
    if (isset($allAwards[$userId])) {
        foreach ($allAwards[$userId] as $award) {
            $type = $award->championship_type;
            $awardValue = $award->award;
            $count = $award->count;
            switch ($type) {
                case 1: $category = 'region'; break;
                case 2: $category = 'province'; break;
                case 4: $category = 'porkab'; break;
                case 7: $category = 'open'; break;
                default: continue 2;
            }
            $counts[$category][$awardValue] = $count;
        }
    }
    $ageFormatted = Carbon::parse($user->date_of_birth)
        ->diffInYears(Carbon::now());
    $dataUser[] = [$ageFormatted, $user->belt_level];
    $dataUser[] = array_merge(
        $counts['region'],
        $counts['province'],
        $counts['porkab'],
        $counts['open']
    );
}

```

Gbr. 11 Kode Mengambil Data User

Setelah data selesai diformat kemudian dilakukan labeling pada data user satu-persatu.

```

foreach ($dataUser as $key => $data) {
    $label = $this->getRecommendationLabel($data);
    $userAwards[$key]->recomendation = [
        'predictedNumber' => $label,
    ];
}

```

Gbr. 12 Kode Implementasi metode filter query

Berikut fungsi yang digunakan untuk memberikan label pada data user dengan ketentuan sebagai berikut:

- Usia 15-17 Tahun dan Tingkatan Sabuk Kuning – Coklat (Level 2-6)
- Mempunyai Prestasi Juara 1/2/3 pada kejurprov atau
- Mempunyai Prestasi Juara 1 pada porkab atau
- Mempunyai Prestasi Juara 2 pada porkab ditambah dengan Juara 1 pada Kejurkab/kejurkot atau

- Mempunyai Prestasi Juara 2 pada porkab ditambah dengan Juara 2/3 pada Kejurkab/kejurkot sebanyak 2 kali atau
- Mempunyai Prestasi Juara 2 pada porkab ditambah dengan Juara 1 pada Kejuaraan Open min 2 kali atau
- Mempunyai Prestasi Juara 2 pada porkab ditambah dengan Juara 2/3 pada Kejuaraan Open min 4 kali atau
- Mempunyai Prestasi Juara 3 pada porkab ditambah dengan Juara 1/2/3 pada Kejuaraan Open min 2 kali dan Juara 1/2/3 pada Kejurkot/kejurprov

```
private function getRecommendationLabel($data)
{
    [$age,$beltLevel, $kejurkotP, $kejurkot1, $kejurkot2,
    $kejurkot3, $kejurprovP, $kejurprov1, $kejurprov2,
    $kejurprov3, $porkabP, $porkab1, $porkab2, $porkab3,
    $openP, $open1, $open2, $open3] = $data;

    // Inisialisasi variabel untuk prestasi
    $kejurkotWins = $kejurkot1 + $kejurkot2;
    $porkabWins = $porkab1 + $porkab2;
    $openWins = $open1 + $open2 + $open3;
    $open_wins = $open1 + $open2 + $open3;
    $open_wins_2 = $open1 + $open2;

    $very_recomend = $porkab1 ||
    ($kejurprov1 + $kejurprov2 + $kejurprov3) > 0 ||
    $porkab2 > 1 || ($porkab2 > 0 || $open_wins_2 > 2)
    || $kejurkot1 > 1 || ($kejurkot1 > 0 && $open_wins_2 > 2)
    || ($kejurkot2 > 0 && $open_wins_2 > 3) || $open1 >= 4
    || ($open_wins_2 >= 4 || ($open_wins_2 >= 2 && $open3 > 3);

    $recomend = $porkab2 > 0 || $kejurkot1 > 0 ||
    ($kejurkot2 > 0 && $open_wins > 2) ||
    $open_wins_2 > 2 || ($open_wins_2 > 0 && $open3 > 3);

    // Aturan sabuk
    if ($beltLevel < 2 || $beltLevel > 6) {
        return 0;
    }
    // Aturan usia dan kejuaraan
    elseif ($age >= 17 && $age <= 20 &&
    $beltLevel >= 2 && $beltLevel <= 6) {
        return ($very_recomend || $recomend ? 2 : 0;
    } elseif (($age >= 21 && $age <= 22) ||
    ($age >= 15 && $age <= 16)) {
        return $very_recomend ? 2 : 0;
    } elseif ($age < 15 || $age > 23) {
        return 0;
    }

    // Jika tidak ada juara sama sekali
    if ($kejurkot1 + $kejurkot2 + $kejurkot3 +
    $kejurprov1 + $kejurprov2 + $kejurprov3 +
    $porkab1 + $porkab2 + $porkab3 +
    $open1 + $open2 + $open3 == 0) {
        return 0;
    }
    return 0;
}
```

Gbr. 13 Kode labeling pada data

D. Pengujian Sistem Rekomendasi dan Sistem Validasi

Setelah selesai pembuatan pada fitur rekomendasi menggunakan fitur naïve bayes dan query filter, kemudian dilakukan perbandingan antara kedua metode tersebut dengan 20 data sample yang telah dibuat. Data sample dapat dilihat pada halaman lampiran.

Dari 20 data sample yang digunakan menampilkan hasil sebagai berikut:

1. Metode Naïve Bayes

Berikut hasil dari rekomendasi atlet menggunakan metode naïve bayes yang ditampilkan dengan format tabel.

Akurasi		75.19%
Akurasi Model		73.68%
Waktu Pemrosesan		0.47 Detik

NO	NAMA	DOJO	TINGKAT	UMUR	STATUS	TINGKAT REKOMENDASI
1	atlet5	Pengurnan Ju-Jitsu	Kyu V (Kuning)	19 th 3 bth	Dirkomendasikan	99.15%
2	atlet9	Pengurnan Ju-Jitsu	Kyu II (Biru)	16 th 3 bth	Dirkomendasikan	90.62%
3	atlet14	Pengurnan Ju-Jitsu	Kyu III (Oranye)	15 th 3 bth	Dirkomendasikan	86.38%
4	atlet17	Pengurnan Ju-Jitsu	Kyu V (Kuning)	16 th 3 bth	Dirkomendasikan	81%
5	atlet11	Pengurnan Ju-Jitsu	Kyu V (Kuning)	18 th 3 bth	Dirkomendasikan	79.53%
6	atlet5	Pengurnan Ju-Jitsu	Kyu I (Coklat)	20 th 3 bth	Dirkomendasikan	74.66%
7	atlet12	Pengurnan Ju-Jitsu	Kyu V (Kuning)	17 th 3 bth	Dirkomendasikan	73.39%
8	atlet10	Pengurnan Ju-Jitsu	Kyu I (Coklat)	21 th 3 bth	Dirkomendasikan	70.84%
9	Atlet Baru	Pengurnan Ju-Jitsu	Kyu V (Kuning)	18 th 1 bth	Dirkomendasikan	66.26%
10	atlet20	Pengurnan Ju-Jitsu	Kyu IV (Hijau)	20 th 3 bth	Dirkomendasikan	60.03%
11	atlet19	Pengurnan Ju-Jitsu	Kyu IV (Hijau)	19 th 3 bth	Dirkomendasikan	53.01%
12	atlet8	Pengurnan Ju-Jitsu	Kyu III (Oranye)	17 th 3 bth	Dirkomendasikan	51.58%
13	atlet3	Pengurnan Ju-Jitsu	Kyu III (Oranye)	22 th 3 bth	Tidak Dirkomendasikan	49.01%
14	atlet18	Pengurnan Ju-Jitsu	Kyu V (Kuning)	17 th 3 bth	Tidak Dirkomendasikan	46.17%
15	atlet15	Pengurnan Ju-Jitsu	Kyu II (Biru)	17 th 3 bth	Tidak Dirkomendasikan	44.42%
16	atlet4	Pengurnan Ju-Jitsu	Kyu II (Biru)	21 th 3 bth	Tidak Dirkomendasikan	34.33%
17	atlet13	Pengurnan Ju-Jitsu	Kyu IV (Hijau)	16 th 3 bth	Tidak Dirkomendasikan	26.72%
18	atlet16	Pengurnan Ju-Jitsu	Kyu I (Coklat)	13 th 3 bth	Tidak Dirkomendasikan	13.6%
19	atlet7	Pengurnan Ju-Jitsu	Kyu IV (Hijau)	18 th 3 bth	Tidak Dirkomendasikan	6.63%
20	atlet2	Pengurnan Ju-Jitsu	Kyu IV (Hijau)	23 th 3 bth	Tidak Dirkomendasikan	6.43%
21	atlet11	Pengurnan Ju-Jitsu	Dan I	24 th 3 bth	Tidak Dirkomendasikan	5.45%

Gbr. 14 Hasil Rekomendasi Methode Naive Bayes

2. Metode Query Filter

Berikut hasil dari rekomendasi atlet menggunakan metode naïve bayes yang ditampilkan dengan format tabel.

Akurasi : 76.19%
Akurasi Model : 73.98%
Waktu Pemrosesan : 0.07 Detik

Show 25 entries

NO	NAMA	DOJO	TINGKAT	UMUR	STATUS	TINGKAT REKOMENDASI
1	atlet5	Penggunaan Ju-Utsu	Kyu V (Kuning)	19 th 3 bn	Direkomendasikan	99.15%
2	atlet9	Penggunaan Ju-Utsu	Kyu II (Biru)	16 th 3 bn	Direkomendasikan	90.82%
3	atlet14	Penggunaan Ju-Utsu	Kyu III (Oranye)	15 th 3 bn	Direkomendasikan	86.38%
4	atlet17	Penggunaan Ju-Utsu	Kyu V (Kuning)	16 th 3 bn	Direkomendasikan	81%
5	atlet1	Penggunaan Ju-Utsu	Kyu V (Kuning)	18 th 3 bn	Direkomendasikan	79.53%
6	atlet5	Penggunaan Ju-Utsu	Kyu I (Coklat)	20 th 3 bn	Direkomendasikan	74.66%
7	atlet12	Penggunaan Ju-Utsu	Kyu V (Kuning)	17 th 3 bn	Direkomendasikan	73.39%
8	atlet10	Penggunaan Ju-Utsu	Kyu I (Coklat)	21 th 3 bn	Direkomendasikan	70.84%
9	Atlet Baru	Penggunaan Ju-Utsu	Kyu V (Kuning)	18 th 1 bn	Direkomendasikan	66.26%
10	atlet20	Penggunaan Ju-Utsu	Kyu IV (Hijau)	20 th 3 bn	Direkomendasikan	60.03%
11	atlet19	Penggunaan Ju-Utsu	Kyu IV (Hijau)	19 th 3 bn	Direkomendasikan	53.01%
12	atlet8	Penggunaan Ju-Utsu	Kyu III (Oranye)	17 th 3 bn	Direkomendasikan	51.58%
13	atlet3	Penggunaan Ju-Utsu	Kyu III (Oranye)	22 th 3 bn	Tidak Direkomendasikan	49.01%
14	atlet18	Penggunaan Ju-Utsu	Kyu V (Kuning)	17 th 3 bn	Tidak Direkomendasikan	46.17%
15	atlet15	Penggunaan Ju-Utsu	Kyu II (Biru)	17 th 3 bn	Tidak Direkomendasikan	44.42%
16	atlet4	Penggunaan Ju-Utsu	Kyu II (Biru)	21 th 3 bn	Tidak Direkomendasikan	34.33%
17	atlet13	Penggunaan Ju-Utsu	Kyu IV (Hijau)	16 th 3 bn	Tidak Direkomendasikan	26.72%
18	atlet16	Penggunaan Ju-Utsu	Kyu I (Coklat)	13 th 3 bn	Tidak Direkomendasikan	13.6%
19	atlet7	Penggunaan Ju-Utsu	Kyu IV (Hijau)	18 th 3 bn	Tidak Direkomendasikan	6.53%
20	atlet2	Penggunaan Ju-Utsu	Kyu IV (Hijau)	23 th 3 bn	Tidak Direkomendasikan	6.43%
21	atlet11	Penggunaan Ju-Utsu	Dan I	24 th 3 bn	Tidak Direkomendasikan	5.45%

Gbr. 15 Hasil Rekomendasi Methode Query Filter

Dari hasil diatas dapat disimpulkan bahwa metode Naïve bayes memiliki keunggulan dibanding menggunakan query filter.

a. Akurasi

Akurasi hasil yang didapatkan metode query filter yaitu 75%, hal ini bisa terjadi karena saat pengembangan klasifikasi dibuat *hardcode* dengan mengklasifikasi secara manual untuk setiap datanya. Dimana klasifikasi manual ini rentan terhadap bug pada perubahan data secara dinamis. Apabila pengklasifikasian dilakukan pada data yang berjumlah besar, akurasi akan menurun secara drastis dan perlu mengubah code untuk menyesuaikan data yang ada.

Akurasi penggunaan metode naïve bayes sama dengan query filter yaitu pada 75%, meskipun data train yang dipakai hanya berjumlah 200 data, yang mana termasuk sedikit untuk digunakan sebagai data train. Data train ini akan terus berkembang kedepannya dengan cara user Admin PBJI menambahkan data train dari data atlet yang terpilih. Semakin banyak data train maka akan semakin meningkatkan akurasi dari rekomendasinya.

b. Waktu Pemrosesan

Pengujian waktu pemrosesan menggunakan *function* *microtime* yang sudah ada pada Laravel. Cara untuk implementasinya seperti berikut:

```
// Mengambil waktu sebelum mulai proses rekomendasi
$startTime = microtime(true);

...Proses Rekomendasi

// Mengambil waktu setelah proses rekomendasi
$endTime = microtime(true);

// Mengambil selisih waktu mulai dan akhir
$endTime - $startTime
```

Gbr. 16 Menghitung waktu pemrosesan

Hasil yang didapatkan, metode naïve bayes memiliki waktu pemrosesan yang lebih lambat yakni 0.1 – 0.2 detik, sedangkan pada metode query mendapatkan 0.04 – 0.1 detik.

c. Kemudahan dalam implementasi

Metode query lebih mudah dalam implementasinya, dimana pengembang cukup mengambil data setiap user, kemudian mengklasifikasi secara langsung dengan menggunakan perulangan dan percabangan. Data yang sudah diklasifikasi ditampilkan dalam tabel.

Berbeda dengan metode naïve bayes, dimana pengembang harus membuat data train terlebih dahulu, kemudian melakukan cleaning pada data tersebut sebelum disimpan pada tabel training. Data train kemudian di gunakan untuk membuat model klasifikasi yang kemudian model digunakan untuk membuat rekomendasi berdasarkan pola pada data train yang sudah disimpan.

d. Data yang ditampilkan

Pada metode naïve bayes dapat ditampilkan dan diurutkan berdasarkan tingkat kecocokan pada pola dengan skala 0%-100%, dimana dari persentase tersebut dapat dikelompokkan menjadi, 0%-49% tidak direkomendasikan, 50%-69% kurang direkomendasikan, 70%-89% direkomendasikan, dan 90%-100% sangat direkomendasikan

Sedangkan pada metode query filter tidak dapat diurutkan berdasarkan persentase kecocokan dan hanya menampilkan data yang direkomendasikan saja.

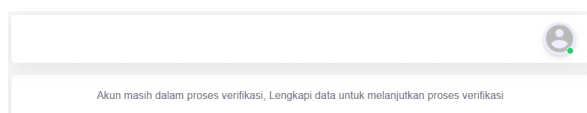
G. Pengujian Sistem Validasi Perguruan dan Atlet

Sistem validasi perguruan dan atlet disini berguna untuk memilah perguruan dan atlet yang benar-benar dari

beladiri Ju-Jitsu, agar atlet yang berasal dari perguruan beladiri Ju-Jitsu dapat berkembang dan mendapatkan prestasi tanpa khawatir ada persaingan dari atlet beladiri lain yang hanya ingin mendapatkan keuntungan di turnamen beladiri Ju-Jitsu.

Pengujian dilakukan dengan cara :

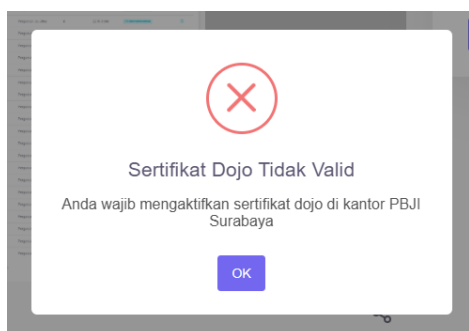
1. Memaksa masuk ke fitur-fitur selain dashboard untuk akun yang belum divalidasi oleh admin aliran perguruan.



Gambar 17 Fitur terkunci pada akun belum terverifikasi

Dari gambar 17 terlihat bahwa akun yang belum terverifikasi tidak dapat mengakses fitur selain dashboard, dan muncul keterangan akun masih dalam proses verifikasi.

2. Memaksa mengikuti kegiatan dengan akun yang sertifikat perguruan belum divalidasi oleh system



Gbr. 18 Tidak dapat mendaftar kegiatan pada akun belum terverifikasi

Dari gambar 18 terlihat bahwa akun yang belum terverifikasi tidak dapat mendaftar pada kegiatan, dan muncul keterangan untuk mengaktifkan sertifikat di kantor PBJI Surabaya.

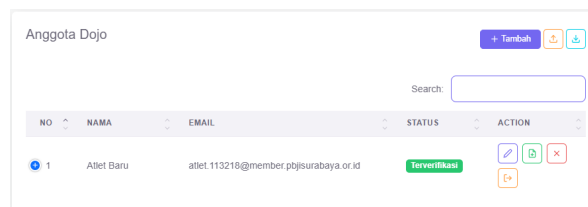
3. Memaksa mengikuti kegiatan untuk atlet yang belum divalidasi oleh admin aliran perguruan



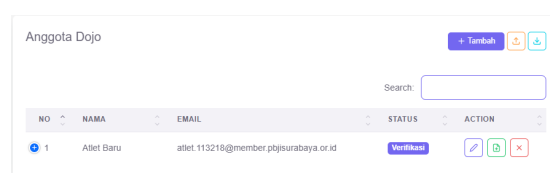
Gbr. 19 Peserta tidak muncul pada form pendaftaran kegiatan

Dari gambar 19 terlihat bahwa akun atlet yang belum terverifikasi tidak muncul pada form pendaftaran kegiatan.

4. Mengubah data atlet sebelum mendaftarkan kegiatan



Gbr. 20 Data atlet sebelum diubah



Gbr. 21 Data atlet setelah diubah

Dari gambar 20 dan 21 terlihat bahwa akun atlet yang sudah terverifikasi kemudian diubah data atlet tersebut, otomatis akun dari atlet tersebut menjadi proses verifikasi kembali.

E. Deployment dan Pengujian Blackbox

Pada tahap ini akan dilakukan pengujian dengan metode blackbox testing oleh pengembang, guna memastikan aplikasi berjalan lancar. Jika pada proses blackbox tidak ditemukan masalah akan dilanjutkan proses deployment ke production yaitu dengan mengunggah pada hosting.

1. Blackbox Testing

Pada tahap ini, dilakukan pengujian terhadap fitur-fitur utama sistem menggunakan metode Blackbox Testing. Pengujian dilakukan dengan memberikan input tertentu dan memeriksa apakah output sesuai dengan yang diharapkan. Pengujian dilakukan pada semua fitur pada website, diantaranya yaitu fitur login dan register, fitur pengelolaan anggota, fitur pengelolaan dan pendaftaran, fitur pengelolaan berita kegiatan, fitur pengelolaan perguruan, fitur pengelolaan ijazah, fitur pengelolaan data prestasi, fitur rekomendasi. Dari hasil testing menunjukkan bahwa semua fitur berjalan dengan baik tanpa ada kendala. Dokumentasi lengkap mengenai hasil pengujian dapat ditemukan pada lampiran.

2. Deploy Production

Pada tahap ini, proses deployment dilakukan untuk memindahkan sistem ke lingkungan production menggunakan layanan hosting dari Niagahoster. Proses ini bertujuan agar aplikasi dapat diakses secara luas oleh pengguna melalui domain resmi. Adapun langkah-langkah yang dilakukan adalah sebagai berikut:

a) Menyiapkan Server

Sebelum memulai deployment, pengembang memastikan semua kebutuhan server telah dipenuhi, yaitu:

- Domain: Mengonfigurasi domain resmi yang akan digunakan untuk aplikasi, domain yang digunakan pada aplikasi ini adalah <https://pbjisurabaya.or.id>.
- Database MySQL: Membuat dan mengatur database pada hosting untuk menyimpan data yang dibutuhkan aplikasi.
- Sertifikat SSL: Mengaktifkan SSL untuk memastikan koneksi aman (https). Sertifikat SSL meningkatkan kepercayaan pengguna dan keamanan data.

b) Mengunggah File Project ke Server

Setelah server siap, file proyek diunggah ke server. Pengembang menggunakan Git sebagai alat bantu pengelolaan versi:

- File proyek di-*push* ke repository Git.
- Server hanya perlu menarik data dari repository Git (dengan *git pull*) untuk menyinkronkan file.
- Untuk pembaruan selanjutnya, pengembang cukup memperbarui file di Git dan menjalankan perintah *git-pull* pada server untuk mendapatkan versi terbaru.

c) Konfigurasi file environment

Setelah file proyek berhasil diunggah, pengembang mengonfigurasi file `.env` agar sesuai dengan kebutuhan production environment.

d) Migrasi database

Setelah konfigurasi environment selesai, pengembang menjalankan migrasi database untuk memastikan struktur tabel sesuai dengan desain aplikasi. Pada Laravel, proses migrasi dilakukan dengan perintah `php artisan migrate`

e) Pengujian

Setelah deployment selesai, dilakukan pengujian langsung pada aplikasi di lingkungan production untuk memastikan:

- Semua fitur berjalan dengan baik.
- Aplikasi dapat diakses melalui URL yang ditentukan.
- Tidak ada error pada koneksi ke database atau layanan lainnya.

F. Testing

Pada tahap ini, dilakukan pengujian akhir pada aplikasi yang telah di-deploy ke lingkungan production. Tujuan dari UAT adalah untuk memastikan bahwa aplikasi memenuhi kebutuhan dan ekspektasi pengguna berdasarkan skenario penggunaan di dunia nyata. Testing ini melibatkan semua role pengguna sesuai dengan fitur yang telah dirancang untuk masing-masing role.

Pengujian dilakukan dengan meminta feedback langsung dari pengguna terkait pengalaman penggunaan aplikasi, kemudahan navigasi, serta fungsionalitas setiap fitur. Berdasarkan hasil UAT, semua fitur aplikasi berhasil berjalan dengan baik dan dinyatakan memenuhi kebutuhan pengguna di semua role. Dokumentasi lengkap mengenai hasil pengujian dan feedback pengguna dapat ditemukan pada lampiran.

IV. KESIMPULAN

Berdasarkan penelitian dan implementasi yang dilakukan, beberapa kesimpulan yang dapat diambil adalah sebagai berikut:

1. Pembangunan Aplikasi Manajemen dan Rekomendasi Atlet

Aplikasi manajemen dan rekomendasi atlet Ju-Jitsu berbasis website berhasil dikembangkan sebagai salah satu fasilitas di PBJI Kota Surabaya. Aplikasi ini mencakup fitur-fitur seperti pengelolaan data atlet, prestasi, dan kegiatan, serta sistem rekomendasi atlet berbasis metode Naïve Bayes, yang memudahkan proses seleksi atlet berprestasi.

Aplikasi yang dibuat secara efektif dapat membantu PBJI Kota Surabaya dalam mendata atlet dan menentukan atlet untuk mengikuti kejuaraan berdasarkan fitur rekomendasi. Hal ini dibuktikan dengan hasil UAT dari beberapa pengguna dengan bermacam-macam role akun mendapatkan tanggapan yang positif.

Pendekatan Rapid Application Development (RAD) memfasilitasi pengembangan aplikasi secara iteratif dan cepat. Proses mulai dari perencanaan kebutuhan, desain, hingga implementasi berhasil

diselesaikan dengan baik menggunakan alat bantu seperti *QuickAdminPanel*.

2. Implementasi Sistem Rekomendasi dengan Naïve Bayes

Sistem rekomendasi atlet dibangun menggunakan metode Naïve Bayes dengan memanfaatkan library Rubix/ML pada bahasa pemrograman PHP. Implementasi ini memungkinkan sistem untuk mempelajari pola data historis prestasi atlet dan memberikan rekomendasi dengan tingkat akurasi yang baik. Sedangkan query filter diimplementasi dengan mengambil data dari database yang kemudian dikelompokkan menggunakan perulangan dan percabangan.

Berdasarkan pengujian, metode Query Filter lebih unggul dalam hal implementasi karena pengembang hanya perlu menambahkan klasifikasi melalui perulangan dan percabangan, akan tetapi data menjadi tidak dinamis karena tidak ada analisis probabilistic dimana akan mengakibatkan turunnya tingkat akurasi. Dalam hal akurasi metode naïve bayes memiliki tingkat akurat yang sama dengan query filter meskipun data train yang digunakan relatif kecil. Jika data train lebih besar dan variatif, maka tingkat akurasi naïve bayes akan jauh Lebih tinggi dibandingkan dengan query filter. Pada kecepatan query filter terbilang Lebih cepat karena data yang diproses Lebih sedikit dibandingkan naïve bayes. Akan tetapi tidak memungkinkan ketika data user Lebih besar metode query akan menurun tingkat kecepatannya.

3. Sistem Validasi Data

Sistem validasi data berhasil diterapkan untuk memastikan bahwa data yang dimasukkan oleh pengguna valid, tidak doble, dan lengkap. User dengan data yang belum terverifikasi tidak dapat menjalankan fitur-fitur yang ada pada akun yang telah terverifikasi. Pada system juga terdapat pengecekan ketika user mengubah data, maka otomatis akan terkunci dan menunggu proses validasi ulang, untuk mengecek apakah data yang dimasukkan valid.

V. SARAN

Berdasarkan proses penelitian, terdapat beberapa saran yang dapat bermanfaat pada penelitian selanjutnya, antara lain:

1. Peningkatan Data Training

Untuk meningkatkan akurasi metode Naïve Bayes, disarankan untuk menambah jumlah data training, terutama data historis prestasi atlet dari berbagai kejuaraan. Data yang lebih bervariasi dapat membantu algoritma dalam mengenali pola yang lebih kompleks.

3. Pengembangan Lebih Lanjut

Sistem dapat dikembangkan lebih lanjut dengan menambahkan fitur evaluasi performa atlet secara berkala berdasarkan hasil pertandingan terbaru. Hal ini akan membantu pembaruan data rekomendasi agar lebih relevan.

4. Optimasi Query Filter

Untuk meningkatkan kecepatan pengambilan data, perlu dilakukan optimasi pada query SQL, seperti penggunaan indeks pada kolom yang sering diakses dan pemanfaatan teknik caching. Juga perlu penambahan kode untuk mengambil data sesuai kriteria rekomendasi secara langsung, agar mempercepat waktu pemrosesannya.

5. Integrasi dengan Platform Lain

Disarankan untuk mengintegrasikan sistem dengan platform lain, seperti sistem manajemen kompetisi atau aplikasi pelaporan prestasi atlet, guna menciptakan ekosistem yang terhubung dan lebih efisien.

REFERENSI

- [1] M. F. Amin and A. Sudijandoko, "ANALISIS KEMAMPUAN KONDISI FISIK DOMINAN ATLET PUTRI JU-JITSU DOJO UNIVERSITAS NEGERI SURABAYA," *J. Kesehat. Olahraga*, vol. 7, no. 5, pp. 430–435, 2019, [Online]. Available: <https://ejournal.unesa.ac.id/index.php/jurnal-kesehatan-olahraga/article/view/29512>
- [2] N. R. S. Huaturuk, R. D. Rahmadani, and D. J. AK, "Komparasi Akurasi Naïve Bayes dan Support Vector Machine (SVM) untuk Rekomendasi Produk in Fashion Dress," *Centive 2018*, pp. 168–173, 2018.
- [3] R. Y. Hayuningtyas, "Penerapan Algoritma Naïve Bayes untuk Rekomendasi Pakaian Wanita," *J. Inform.*, vol. 6, no. 1, pp. 18–22, 2019, doi: 10.31311/ji.v6i1.4685.
- [4] T. Pricillia and Zulfachmi, "Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD)," *J. Bangkit Indones.*, vol. 10, no. 1, pp. 6–12, 2021, doi: 10.52771/bangkitindonesia.v10i1.153.
- [5] Subianto, "Penerapan Metode Rapid Application Development dalam Perancangan Sistem Informasi Pendataan," *J. Infokam*, vol. 16, no. 1, pp. 46–55, 2020, [Online]. Available: <http://amikjtc.com/jurnal/index.php/jurnal/article/view/218/164#>
- [6] Deni Murdiani and Muhamad Sobirin, "Perbandingan Metodologi Waterfall Dan Rad (Rapid Application Development) Dalam Pengembangan Sistem Informasi," *J. Inform. Teknol. dan Sains*, vol. 4, no. 4, pp. 302–306, 2022, doi: 10.51401/jinteks.v4i4.2008.
- [7] F. C. Ningrum, D. Suherman, S. Aryanti, H. A. Prasetya, and A. Saifudin, "Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 125, 2019, doi: 10.32493/informatika.v4i4.3782.
- [8] M. A. Chamida, A. Susanto, and A. Latubessy, "Analisa User Acceptance Testing Terhadap Sistem Informasi Pengelolaan Bedah

Rumah Di Dinas Perumahan Rakyat Dan Kawasan Permukiman
Kabupaten Jepara,” *Indones. J. Technol. Informatics Sci.*, vol. 3, no. 1,

pp. 36–41, 2021, doi: 10.24176/ijtis.v3i1.7531.