

BAB III

PERMASALAHAN PEKON PAREREJO

3.1 Analisa Permasalahan yang Dihadapi Pekon

Berikut ini adalah temuan dan perumusan masalah yang terdapat di Pekon Parerejo.

3.1.1 Temuan Masalah

Banjir merupakan salah satu permasalahan yang sering terjadi di Pekon Parerejo, terutama pada area persawahan. Banyak petani mengalami kerugian karena banjir datang secara tiba-tiba tanpa adanya informasi atau peringatan sebelumnya. Misalnya, saat musim panen, padi yang sudah dipotong atau hampir siap panen bisa terendam banjir akibat hujan lebat di malam hari. Kondisi ini membuat hasil panen rusak dan menurunkan kualitas padi.

Permasalahan semakin kompleks karena masyarakat, khususnya petani, tidak memiliki akses terhadap sistem informasi yang mampu memberikan perkiraan potensi banjir secara cepat dan akurat. Selama ini, informasi yang tersedia hanya bersumber dari berita atau laporan setelah banjir terjadi, sehingga tidak bermanfaat untuk pencegahan. Akibatnya, masyarakat tidak dapat mengambil langkah antisipasi, seperti mempercepat panen atau melindungi hasil panen yang sudah dikumpulkan.

3.1.2 Perumusan Masalah

Berdasarkan temuan masalah tersebut, maka rumusan masalah dalam kerja praktek ini dapat disusun dalam bentuk pertanyaan sebagai berikut:

- a. Bagaimana cara memperoleh informasi kondisi cuaca yang relevan untuk wilayah Pekon Parerejo?
- b. Bagaimana memanfaatkan informasi tersebut untuk memprediksi kemungkinan terjadinya banjir?
- c. Bagaimana merancang sebuah sistem yang mampu menyajikan informasi prediksi banjir secara *real-time*?
- d. Bagaimana menyediakan informasi tersebut dalam bentuk media yang mudah diakses masyarakat?

3.1.3 Kerangka Pemecahan Masalah

Untuk menjawab perumusan masalah di atas, langkah-langkah pemecahan masalah yang dilakukan adalah:

- a. Mengumpulkan data cuaca harian historis dari NASA POWER, meliputi parameter curah hujan, suhu, kelembapan, dan kecepatan angin.
- b. Mengidentifikasi kejadian banjir di Pekon Parerejo berdasarkan berita, kemudian menganalisis kondisi cuaca pada saat itu.
- c. Membuat aturan *fuzzy* (*Fuzzy Rule-Based*) untuk memberi label pada data banjir (aman, siaga, banjir).
- d. Melatih model prediksi menggunakan algoritma *decision tree* agar dapat melakukan analisis secara *real-time* dan mengurangi bias dari aturan *fuzzy*.
- e. Mengimplementasikan model prediksi ke dalam *website* yang menampilkan informasi banjir harian secara *real-time*, sehingga mudah diakses dan dipahami masyarakat.

3.2 Landasan Teori

Landasan teori digunakan sebagai dasar acuan dalam penelitian ini, meliputi data cuaca harian, metode *fuzzy rule based*, algoritma *decision tree*, serta pemanfaatan *website* sebagai media informasi untuk prediksi banjir secara *real-time*.

3.2.1 Data Cuaca NASA POWER

NASA POWER (Prediction Of Worldwide Energy Resources) adalah layanan data yang menyediakan informasi cuaca dan iklim berbasis satelit. Data ini mencakup parameter harian seperti suhu, kelembapan, curah hujan, kecepatan angin, dan titik embun.

Data ini sering digunakan untuk riset pertanian, energi, hingga mitigasi bencana karena sifatnya yang *open access* (terbuka) dan cakupan global.

3.2.2 Logika Fuzzy (Fuzzy Rule-Based)

Fuzzy set atau himpunan *fuzzy* adalah suatu himpunan yang keanggotaannya tidak hanya bersifat biner (0 = tidak masuk, 1 = masuk) seperti pada himpunan klasik, tetapi memiliki derajat keanggotaan dengan nilai kontinu antara 0 sampai 1.

Derajat keanggotaan ini dinyatakan melalui fungsi keanggotaan (*membership function*) yang memetakan setiap elemen ke dalam interval [0,1]. Semakin mendekati 1, maka semakin tinggi tingkat keanggotaan elemen tersebut dalam himpunan fuzzy (Zadeh, 1965).

Rule-based fuzzy logic systems (sistem logika *fuzzy* berbasis aturan) adalah sistem yang menggunakan aturan IF–THEN *fuzzy* untuk mengekspresikan

pengetahuan dalam bentuk himpunan *fuzzy*, memungkinkan sistem merepresentasikan dan menangani ketidakpastian, ketidaktepatan (*imprecision*), dan hubungan kompleks antar variabel.

3.2.3 Machine Learning

Machine learning adalah suatu bidang ilmu yang mempelajari algoritma yang memungkinkan komputer untuk belajar dari data. Program komputer dikatakan belajar jika kinerjanya dalam suatu tugas meningkat seiring pengalaman yang diperolehnya (Mitchell, 1997).

3.2.4 Decision Tree

Decision tree adalah suatu model prediktif berbentuk struktur pohon yang digunakan untuk melakukan klasifikasi maupun regresi dengan cara membagi data ke dalam cabang-cabang (*nodes*) berdasarkan variabel prediktor yang paling diskriminatif (de Ville, 2013). Proses pembagian dilakukan secara rekursif hingga mencapai kondisi tertentu, dan hasil akhirnya berupa *node* terminal (*leaf*) yang merepresentasikan keputusan atau prediksi.

Keunggulan utama *decision tree* adalah kemampuannya menghasilkan model yang mudah diinterpretasikan, dapat menangani data kategorikal maupun numerik, serta mampu memperlihatkan hubungan interaksi antar variabel secara jelas.

3.2.5 Python

Python adalah bahasa pemrograman yang mudah dipelajari, kuat, dan mendukung pemrograman berorientasi objek. *Sintaks* yang sederhana dan tipe data dinamis membuat *Python* ideal untuk pembuatan skrip maupun pengembangan

aplikasi cepat di berbagai platform. *Interpreter* dan pustaka standar *Python* tersedia gratis, serta dapat diperluas dengan modul tambahan atau fungsi dari bahasa lain seperti C/C++. *Python* juga dapat digunakan sebagai bahasa ekstensi untuk aplikasi yang dapat dikustomisasi. Menurut Guido van Rossum, pencipta *Python*, *Python* dirancang untuk kemudahan penggunaan, fleksibilitas, dan kemampuan mempercepat pengembangan aplikasi. Dengan fitur-fitur ini, *Python* banyak digunakan dalam pengembangan program, skrip, dan modul yang efisien (Van Rossum & Drake, 2006).

3.2.6 Teknologi WEB (HTML, CSS, JavaScript)

Berikut landasan teori mengenai HTML, CSS, dan *JavaScript*.

3.2.6.1 HTML (HyperText Markup Language)

HTML adalah bahasa *markup* yang digunakan untuk membangun struktur atau kerangka dasar halaman web. HTML berfungsi sebagai “tulang punggung” atau “dasar” sebuah situs web karena menentukan elemen-elemen dasar yang membentuk konten halaman.

3.2.6.2 CSS (Cascading Style Sheets)

CSS adalah bahasa yang digunakan untuk menata tampilan halaman web agar lebih menarik dan rapi. Melalui CSS, pengembang dapat mengatur warna, *font*, tata letak, dan elemen visual lainnya sehingga situs web lebih estetis dan mudah dibaca.

3.2.6.3 Javascript

JavaScript adalah bahasa *scripting* yang memberikan interaktivitas dan perilaku dinamis pada situs web. Melalui *JavaScript*, elemen halaman dapat

merespons tindakan pengguna, menampilkan animasi, dan membuat pengalaman pengguna lebih interaktif.

3.3 Metode yang Digunakan

Metode penelitian dalam kerja praktek ini dilakukan melalui beberapa tahapan, antara lain:

3.3.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data cuaca harian historis dari Pekon Parerejo yang diambil dari NASA POWER dengan rentang waktu Januari 2022 hingga Juni 2025. Data ini dipilih karena NASA POWER menyediakan informasi meteorologi yang cukup lengkap, akurat, dan dapat diakses secara bebas.

DATE	Humidity (%)	Temp max (°C)	Temp min (°C)	Temp mean (°C)	Dew point (°C)	Wind max (m/s)	Rain (mm/day)
1/1/2022	86.4	29.01	24.06	26.07	23.54	1.95	0.61
1/2/2022	87.57	28.52	24.13	25.99	23.71	2.12	0.41
1/3/2022	88.71	27.49	24.2	25.57	23.55	2.24	0.58
1/4/2022	81.74	29.64	23.73	26.28	22.8	1.9	0.23
1/5/2022	83.13	29.69	24.25	26.33	23.15	2.03	2.74
.
.
.
6/25/2025	85.32	28.37	24.53	26.32	23.64	1.91	0.28
6/26/2025	86.12	28.48	25.14	26.4	23.87	1.84	0.29
6/27/2025	86.85	28.62	24.95	26.43	24.04	1.5	0.6

6/28/2025	90.42	28.06	24.56	25.93	24.23	2.56	28.9
6/29/2025	87.31	28.04	23.35	25.2	22.88	3.37	14.85
6/30/2025	90.18	28.22	23.65	25.71	23.96	2.26	10.88

Tabel 3.1 Data Historis Cuaca

Adapun parameter cuaca yang digunakan meliputi *DATE* (tanggal) sebagai penanda waktu agar analisis dapat dilakukan secara kronologis, *humidity* (kelembapan) yang berpengaruh terhadap potensi curah hujan dan kondisi udara yang dapat memicu banjir, *temp_max* (suhu maksimum) untuk menunjukkan suhu tertinggi harian yang relevan dengan pengaruh panas ekstrem terhadap siklus air, *temp_min* (suhu minimum) untuk menggambarkan suhu terendah harian yang dapat memengaruhi tingkat penguapan dan kelembapan, serta *temp_mean* (suhu rata-rata) yang memberikan gambaran umum kondisi suhu harian sebagai indikator iklim. Selain itu, digunakan juga *dew_point* (titik embun) untuk mengindikasikan kelembapan udara dan potensi kondensasi yang terkait dengan pembentukan hujan, *wind_max* (kecepatan angin maksimum) untuk menganalisis pergerakan awan hujan serta dampak angin terhadap potensi bencana, dan *rain* (curah hujan) sebagai parameter utama yang secara langsung berkaitan dengan kemungkinan terjadinya banjir.

3.3.2 Pelabelan Data

Tahap pelabelan data dilakukan menggunakan metode *Fuzzy Rule-Based* dengan tiga kategori numerik, yaitu 0 (aman), 1 (siaga), dan 2 (banjir). Aturan ditentukan berdasarkan kombinasi parameter cuaca (curah hujan, kelembapan, suhu rata-rata, dan kecepatan angin) serta disesuaikan dengan catatan kejadian banjir dari berita lokal.

Kondisi Cuaca	Aturan Fuzzy	Kategori
Curah hujan < 20 mm dan kelembapan < 75%	Hujan kecil, kelembapan rendah	0
20–35 mm hujan dan kelembapan $\geq 80\%$ atau hujan > 40 mm dengan angin ≥ 15 km/jam	Hujan sedang/deras disertai kelembapan/angin tinggi	1
Curah hujan > 35 mm, kelembapan $\geq 85\%$, suhu < 28 °C	Hujan deras, kelembapan tinggi, suhu rendah	2
Kondisi lain	Default	0

Tabel 3.2 Aturan Fuzzy Rule-Based

3.3.3 Pembangunan Model Prediksi

Tahap pembangunan model prediksi dilakukan menggunakan algoritma *Decision Tree Classifier*. Algoritma ini dipilih karena mampu menghasilkan model yang mudah dipahami melalui representasi pohon keputusan serta dapat menangani data dengan atribut numerik maupun kategorikal. Dataset yang digunakan pada tahap ini merupakan data cuaca historis yang telah melalui proses pelabelan dengan metode *Fuzzy Rule-Based*.

Proses pelatihan model dilakukan dengan membagi dataset menjadi dua bagian, yaitu data latih sebesar 80% dan data uji sebesar 20% menggunakan metode *train-test split*. Model *Decision Tree* dibangun dengan parameter *criterion* menggunakan *entropy* untuk menentukan pemilihan atribut terbaik, serta max depth dibatasi hingga kedalaman 5 guna mengurangi risiko *overfitting*. Setelah proses pelatihan selesai, model diuji menggunakan data uji untuk mengukur tingkat kinerja dan keandalannya.

Evaluasi model dilakukan menggunakan metrik *accuracy* untuk melihat persentase ketepatan klasifikasi, serta *classification report* yang mencakup nilai *precision*, *recall*, dan *f1-score*. Dengan kombinasi metrik ini, performa model dapat dinilai secara menyeluruh, baik dari sisi ketepatan keseluruhan maupun keseimbangan dalam mendeteksi setiap kategori, yaitu aman, siaga, dan banjir. Model yang telah terlatih kemudian disimpan dalam bentuk file agar dapat digunakan kembali pada tahap implementasi sistem prediksi berbasis web.

3.3.4 Implementasi Sistem

Tahap implementasi sistem dilakukan dengan mengintegrasikan model prediksi banjir ke dalam aplikasi berbasis web. Pada sisi *backend*, sistem dikembangkan menggunakan bahasa pemrograman *Python* dengan *framework Flask*. *Flask* digunakan untuk membangun *Application Programming Interface (API)* yang berfungsi sebagai penghubung antara model prediksi dengan antarmuka aplikasi. Model *Decision Tree Classifier* yang telah dilatih sebelumnya dipanggil melalui API ini untuk memberikan hasil prediksi berdasarkan input data cuaca.

Sementara itu, pada sisi *frontend*, tampilan antarmuka dirancang menggunakan kombinasi *HTML*, *CSS*, dan *JavaScript*. *HTML* digunakan untuk membangun struktur halaman web, *CSS* untuk memperindah tampilan agar lebih interaktif dan mudah dipahami pengguna, serta *JavaScript* untuk menambahkan fungsionalitas dinamis seperti pengambilan data secara *real-time* dari *backend*. Dengan integrasi ini, sistem mampu menampilkan hasil prediksi kondisi banjir (aman, siaga, banjir) secara langsung kepada pengguna melalui *browser*.

3.3.5 Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan data cuaca *real-time* yang diambil langsung sebagai input ke dalam model *Decision Tree Classifier* yang telah dilatih sebelumnya. Data tersebut diproses oleh sistem tanpa perlu pelabelan ulang, sehingga hasil prediksi (aman, siaga, atau banjir) dapat diperoleh secara otomatis.

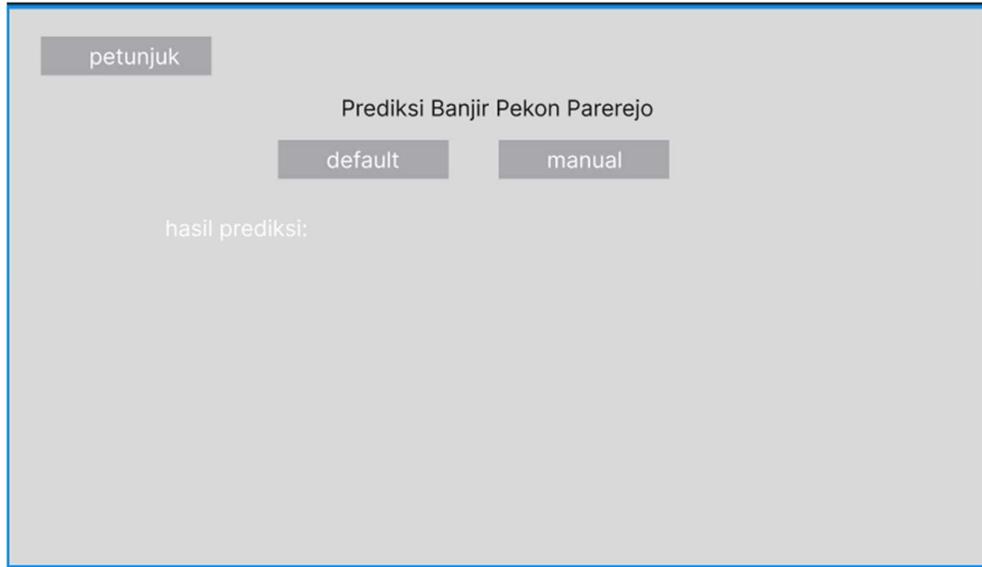
Hasil prediksi ditampilkan secara langsung melalui antarmuka web sehingga pengguna dapat segera mengetahui kondisi terkini. Pengujian ini memastikan bahwa sistem mampu memberikan informasi banjir secara cepat, akurat, dan sesuai dengan tujuan implementasi.

3.4 Rancangan Program

Dalam merancang program terdapat rancangan halaman utama, halaman prediksi *default* berdasarkan data cuaca, halaman prediksi manual untuk memasukkan parameter secara mandiri, serta halaman bantuan yang berisi petunjuk penggunaan aplikasi.

3.4.1 Rancangan Halaman Utama

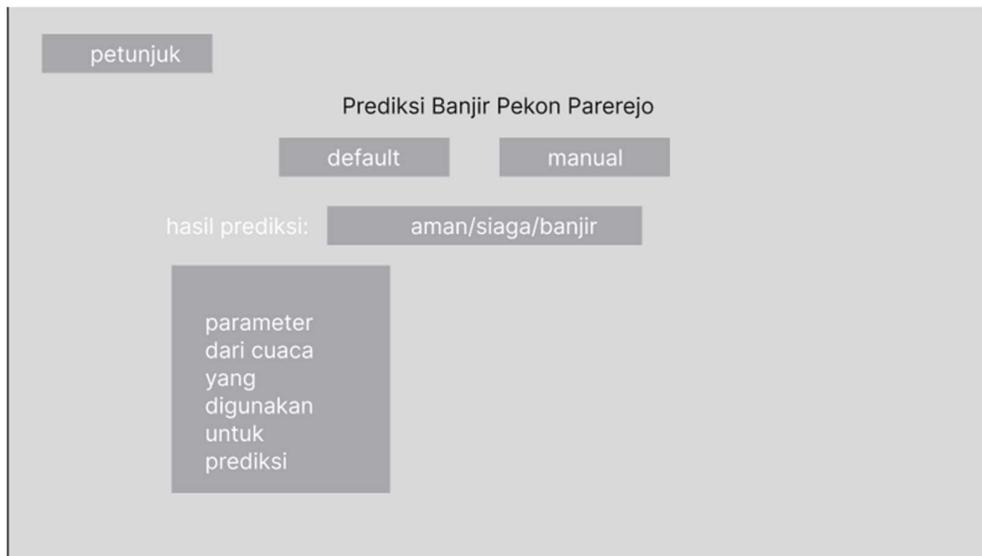
Berikut ini merupakan rancangan untuk halaman utama website prediksi.



Gambar 3.1 Rancangan Halaman Utama

3.4.2 Rancangan Halaman Prediksi *Default*

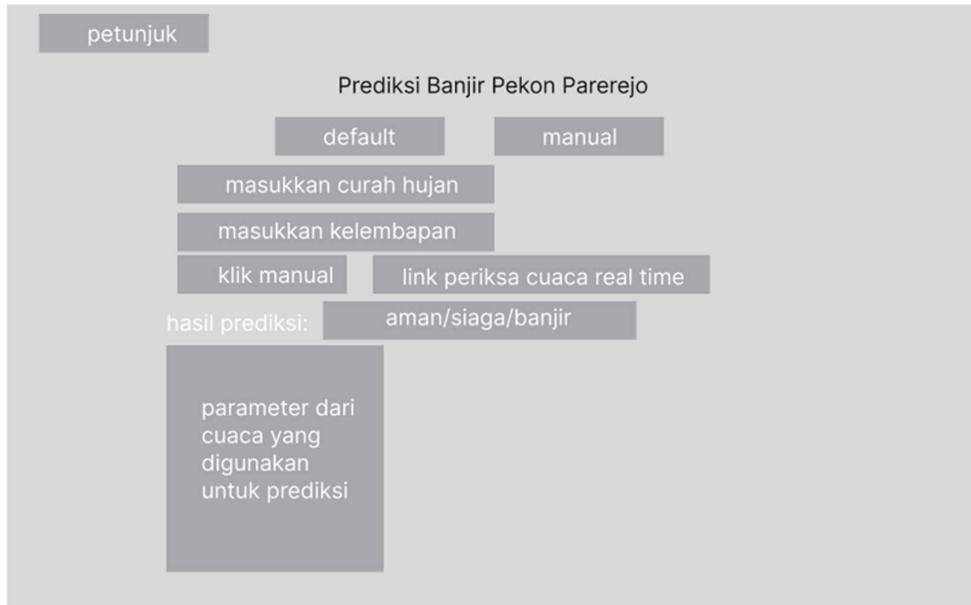
Berikut ini adalah rancangan untuk halaman prediksi *default*.



Gambar 3.2 Rancangan Halaman Prediksi *Default*

3.4.3 Rancangan Halaman Prediksi Manual

Berikut ini adalah rancangan untuk halaman prediksi manual.



Gambar 3.3 Rancangan Halaman Prediksi Manual