

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan dari pengembangan backend, pengujian, dan perbandingan yang telah dilakukan, dapat disimpulkan menjadi beberapa point-point :

1. Implementasi API menggunakan Adonis.js dengan Node.js versi 20.18.0 dan NPM versi 10.8.2 dilakukan secara incremental dalam tiga tahap: API Prediksi Cuaca, API Fitting Kalori, dan API Segmentasi. API Prediksi Cuaca menampilkan data cuaca dari BMKG, API Fitting Kalori menghitung data terkait kalori seperti jarak dan energi kumulatif, sedangkan API Segmentasi menampilkan segmentasi jalur berdasarkan kriteria seperti elevasi dan waktu tempuh. Ketiga API diakses melalui server lokal dengan method GET dan berhasil diuji menggunakan Postman, menunjukkan status 200 OK (200 OK merupakan keterangan jika data sudah berhasil diambil) dengan keluaran format JSON sesuai harapan.
2. *Framework* Adonis js menunjukkan performa yang lebih baik dibandingkan Laravel dalam pengembangan *backend* untuk *website* Muncak.id, terutama pada aspek kecepatan dan efisiensi. Adonis js yang menggunakan metode *asynchronous processing* terbukti lebih unggul dalam menangani permintaan API, seperti prediksi cuaca dan fitting kalori, dengan waktu respons lebih cepat yaitu 3,704 ms dan 3,696 ms serta *throughput* lebih tinggi yaitu 4,08 *req/sec*. Namun, untuk API segmentasi perbedaan performa antara kedua *framework* hampir sama dengan respons time Laravel 10,138 ms dan Adonis Js 10,130 ms. Kedua *framework* juga menunjukkan stabilitas yang baik karena tidak ada error selama pengujian.
3. Hasil dari pengujian manual yang dilakukan, seperti membuat *flowgraph*, menghitung *cyclomatic complexity*, dan menyusun *test case* berhasil memvalidasi bahwa alur program pada *website* sudah berjalan dengan baik.

4. Evaluasi kualitas berdasarkan metode McCall menunjukkan bahwa aspek *Correctness* dan *Integrity* perlu ditingkatkan, meskipun aspek *Usability* sudah berada di kategori baik. Sementara itu, *automation tools* seperti Sonarqube dan Codacy juga membantu dalam mengidentifikasi *bug* atau *Reliability*, *code smells*, dan potensi masalah keamanan pada kode atau *Security*.

Secara keseluruhan, Adonis js lebih cocok digunakan untuk aplikasi dengan kebutuhan performa tinggi, sementara Laravel tetap menjadi pilihan yang baik untuk aplikasi dengan kompleksitas lebih rendah. Hasil penelitian ini memberikan acuan bagi pengembang dalam memilih *framework* yang sesuai dengan kebutuhan dan tujuan pengembangan aplikasi berbasis web.

5.2. Saran

Berdasarkan dari penelitian ini menghasilkan beberapa saran yaitu :

1. Memilih *framework* sesuai kebutuhan. Gunakan Adonis js untuk aplikasi yang membutuhkan performa tinggi dan skalabilitas dalam menangani banyak permintaan secara bersamaan dan gunakan Laravel untuk aplikasi dengan kompleksitas lebih rendah dan berfokus pada kemudahan pengembangan.
2. Meningkatkan aspek *Correctness* dan *Integrity* dengan melakukan revisi dan validasi terhadap kode serta alur program menggunakan *code review* yang lebih mendalam.
3. Memperbaiki kode dengan menggunakan rekomendasi dari Sonarqube dan Codacy secara konsisten untuk memperbaiki *code smells*, *bug*, dan potensi resiko keamanan.

Saran kode perbaikan akan dicantumkan pada lampiran.