

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi web memungkinkan aplikasi berbasis browser memiliki kapabilitas yang semakin mendekati aplikasi native. Salah satu pendekatan yang banyak digunakan adalah Progressive Web App (PWA), yaitu aplikasi web yang dapat berjalan secara lintas platform, memiliki performa tinggi, serta mendukung akses terhadap fitur perangkat seperti kamera, dan penyimpanan lokal[1]. Kemampuan ini membuka peluang pengembangan aplikasi web dengan fungsi yang kompleks namun tetap ringan dan mudah diakses melalui berbagai perangkat[2].

Salah satu implementasi pemanfaatan teknologi tersebut diwujudkan dalam proyek Retina Telkomsel, yaitu sebuah PWA yang ditujukan untuk mendukung sistem dokumentasi dan identifikasi visual terhadap produk-produk yang beredar di pasaran yang didukung dengan kecerdasan buatan (AI) untuk proses identifikasinya. Modul ini memungkinkan pengguna mengambil gambar produk seperti poster promosi, tampilan etalase toko, tampak depan toko (*storefront*), hingga kartu perdana dari berbagai merek. Untuk fitur poster, *storefront*, dan etalase, pengambilan gambar dilakukan satu kali. Namun, pada fitur kartu perdana, pengguna dapat melakukan pengambilan gambar secara berulang untuk mencakup berbagai jenis dan variasi kartu yang tersedia di pasaran.

Pada tahap awal, pengembangan modul kamera dilakukan dengan menggunakan *library* react-webcam yang dikombinasikan dengan elemen canvas untuk proses manipulasi gambar[3]. Namun, pendekatan ini dinilai kurang optimal karena menghasilkan gambar dengan kualitas rendah, berukuran sekitar 400KB, dan sering kali tampak buram (*blur*). Selain itu, penggunaan *library* ini memiliki keterbatasan dalam hal fleksibilitas dan kontrol terhadap kualitas gambar. Hal ini menghambat proses identifikasi oleh sistem AI serta menurunkan kualitas pengalaman pengguna (*user experience*).

Sebagai solusi, dilakukan pengembangan ulang menggunakan elemen HTML video dan canvas yang dikombinasikan langsung dengan API navigator.mediaDevices.getUserMedia()[4], tanpa manipulasi tambahan terhadap gambar yang ditangkap. Pendekatan ini memungkinkan pengambilan gambar

dengan kontrol resolusi yang lebih tinggi, menghasilkan ukuran gambar sekitar 3MB, tergantung pada kemampuan kamera perangkat pengguna. Fokus utama dari pengembangan ini adalah memaksimalkan kualitas gambar yang ditangkap sesuai dengan kemampuan perangkat tanpa proses kompresi atau manipulasi yang dapat menurunkan kualitas gambar.

Selanjutnya, ditambahkan fitur interaktif berupa *zoom* dan *crop* untuk meningkatkan fleksibilitas pengguna dalam mengambil gambar. Fitur *zoom* memberikan kendali tambahan terhadap objek yang ditangkap tanpa perlu mengubah posisi fisik perangkat. Sementara itu, fitur *crop* dikembangkan menggunakan *library* react-image-crop[5]. Pemilihan *library* ini didasarkan pada keunggulannya dalam hal presisi pemotongan gambar, efisiensi performa di perangkat seluler, serta kemudahan integrasi ke dalam aplikasi berbasis React. Dibandingkan dengan *library cropping* lainnya, react-image-crop lebih unggul dari segi kesederhanaan antarmuka dan kestabilan performa[6]. Dengan adanya fitur ini, pengguna dapat memotong bagian gambar sesuai kebutuhan secara lebih presisi dan efisien.

Peningkatan kualitas gambar dari pengembangan sistem ini memunculkan tantangan baru terkait penyimpanan lokal pada browser. LocalStorage yang umum digunakan dalam pengembangan web memiliki kapasitas penyimpanan yang sangat terbatas (sekitar 5MB) dan tidak dirancang untuk menyimpan data berukuran besar seperti gambar resolusi tinggi. Untuk mengatasi keterbatasan tersebut, dipilih IndexedDB sebagai solusi penyimpanan lokal. IndexedDB merupakan penyimpanan berbasis objek yang dirancang untuk menangani data dalam jumlah besar secara asinkron[7]. Keunggulan IndexedDB dibandingkan LocalStorage, SessionStorage, atau Cookies antara lain adalah kapasitas penyimpanan yang jauh lebih besar, dukungan terhadap penyimpanan objek kompleks seperti string base64 atau Blob[8].

Dalam implementasinya, gambar yang ditangkap dari elemen canvas disimpan dalam format base64 menggunakan metode `canvas.toDataURL()`[9]. Format base64 dipilih karena bersifat teks, mudah disimpan di IndexedDB, dapat di-embed langsung ke dalam elemen HTML atau CSS tanpa konversi tambahan, serta mendukung efisiensi pengiriman data dalam kondisi jaringan yang tidak stabil[10]. Pemanfaatan IndexedDB dalam konteks ini tidak hanya mengoptimalkan penyimpanan gambar beresolusi tinggi di sisi klien, tetapi juga mendukung strategi efisiensi jaringan dengan melakukan pengiriman data ke server secara batch saat koneksi tersedia, yang relevan untuk penggunaan aplikasi di daerah dengan akses

internet terbatas[11].

Dengan serangkaian pengembangan tersebut, modul kamera dan penyimpanan pada proyek Retina Telkomsel diharapkan tidak hanya mendukung proses dokumentasi secara lebih akurat dan efisien, tetapi juga meningkatkan pengalaman pengguna pada platform berbasis web. Keberhasilan ini juga berkontribusi terhadap proses identifikasi berbasis AI yang menjadi tujuan utama dari proyek tersebut.

1.2 Maksud dan Tujuan Kerja Magang

Maksud dan tujuan dari pelaksanaan kerja magang ini antara lain sebagai berikut:

1.2.1 Maksud

Maksud dari pelaksanaan program kerja magang ini adalah untuk mendapatkan pengalaman langsung dalam lingkungan kerja profesional, khususnya di bidang pengembangan Progressive Web App (PWA). Melalui program ini, diharapkan dapat memahami proses dan alur kerja pengembangan perangkat lunak, serta menerapkan ilmu dan keterampilan yang telah diperoleh selama perkuliahan pada proyek nyata.

1.2.2 Tujuan

- Meningkatkan pemahaman dan kemampuan teknis dalam pengembangan Progressive Web App (PWA), termasuk pemanfaatan fitur perangkat seperti kamera dan penyimpanan lokal.
- Melatih keterampilan dalam mengidentifikasi dan menyelesaikan permasalahan teknis agar fitur yang dikembangkan dapat berjalan secara optimal.
- Mengasah kemampuan kerja sama tim dan komunikasi profesional di lingkungan kerja.

1.3 Waktu dan Prosedur Pelaksanaan Kerja Magang

Program kerja magang sebagai Fullstack App Developer Internship di PT Devoteam Cloud Services dilaksanakan dalam skema kerja hybrid dengan total durasi 640 jam yang dialokasikan khusus untuk pelaksanaan magang, serta 207 jam tambahan yang diperuntukkan bagi penyusunan laporan akhir. Kegiatan magang berlangsung selama enam bulan, dimulai pada tanggal 3 Februari 2025 dan berakhir pada 2 Agustus 2025. Waktu kerja ditetapkan lima hari dalam satu minggu, yaitu dari hari Senin hingga Jumat, dengan jam kerja mulai pukul 09.00 hingga 19.00 WIB, sesuai dengan ketentuan perusahaan. Skema kerja hybrid terdiri atas kerja jarak jauh (online) pada hari Senin dan Rabu, serta kerja langsung di kantor (onsite) pada hari Selasa, Kamis, dan Jumat.

