

LAMPIRAN

Lampiran 1. Surat Kesiediaan Pembimbing Tugas Akhir

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Faisal Ali Muhamad
NIM : 20090153
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Dwi Intan Af'idah, S.T., M.Kom.
Status : Dosen Tetap
NIDN : 0620089303
Jabatan Fungsional : Asisten Ahli
Pangkat/Golongan : Penata Muda Tk.I/III

Pada hari ini, Kamis tanggal 14 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat adanya kemajuan pengerjaan skripsi yang dipresentasikan setiap dua minggu. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

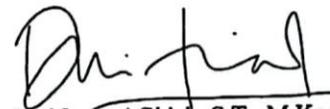
Tegal, 14 Maret 2024

Pihak Pertama



Faisal Ali Muhamad

Pihak Kedua



Dwi Intan Af'idah, S.T., M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M. Kom.
NIPY. 09.015.225

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Faisal Ali Muhamad
NIM : 20090153
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Rosid Mustofa, S.Kom, M.Kom
Status : Dosen Praktisi
NIDN : -
Jabatan Fungsional : -
Pangkat/Golongan : -

Pada hari ini, Kamis tanggal 14 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat adanya kemajuan pengerjaan skripsi yang dipresentasikan setiap satu minggu. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

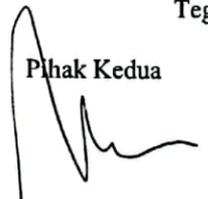
Tegal, 14 Maret 2024

Pihak Pertama



Faisal Ali Muhamad

Pihak Kedua



Rosid Mustofa, S.Kom, M.Kom

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M. Kom.
NIPY. 09.015.225

Lampiran 2. Surat Pernyataan Pengajuan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Faisal Ali Muhamad
Kewarganegaraan : Indonesia
Alamat : Jalan Cendrawasih, Lontrong 12, No.8 Kelurahan Randugunting RT 06/RW 04, Kecamatan Tegal Selatan, Kota Tegal, 52131.
2. Nama : Dwi Intan Afidah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT 003/RW 001, Kecamatan Bulakamba, Kabupaten Brebes, 52253.
2. Nama : Rosid Mustofa, S.Kom., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Kedungbanteng, RT 25/RW 12, Kecamatan Kedungbanteng, Kabupaten Tegal, 52472.

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan :
Berupa : Program Komputer
Berjudul : Aplikasi Mobile untuk Deteksi Jenis Tindak Kejahatan menggunakan Metode YOLOv5 melalui Kamera CCTV
 - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
 - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
 - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
 - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
 - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
 - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami langgar, maka saya / kami bersedia secara sukarela bahwa:
 - a. permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau

- b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
- c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, Juli 2024



(Faisal Ali Muhamad)

(Dwi Intan Afidah, S.T., M.Kom.)

(Rosid Mustofa, S.Kom., M.Kom.)

Lampiran 3. Surat Pengalihan Hak Cipta

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini:

1. Nama : Faisal Ali Muhamad
Kewarganegaraan : Indonesia
Alamat : Jalan Cendrawasih, Lontrong 12, No.8 Kelurahan Randugunting RT 06/RW 04, Kecamatan Tegal Selatan, Kota Tegal, 52131.
2. Nama : Dwi Intan Afidah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT 003/RW 001, Kecamatan Bulakamba, Kabupaten Brebes, 52253.
2. Nama : Rosid Mustofa, S.Kom., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Kedungbanteng, RT 25/RW 12, Kecamatan Kedungbanteng, Kabupaten Tegal, 52472.

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada:

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Aplikasi Mobile untuk Deteksi Jenis Tindak Kejahatan menggunakan Metode YOLOv5 melalui Kamera CCTV" untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta
Ketua P3M



(Dr. Afdi Budi Riyanta, S.Si., M.T)

Tegal, Juli 2024
Pencipta



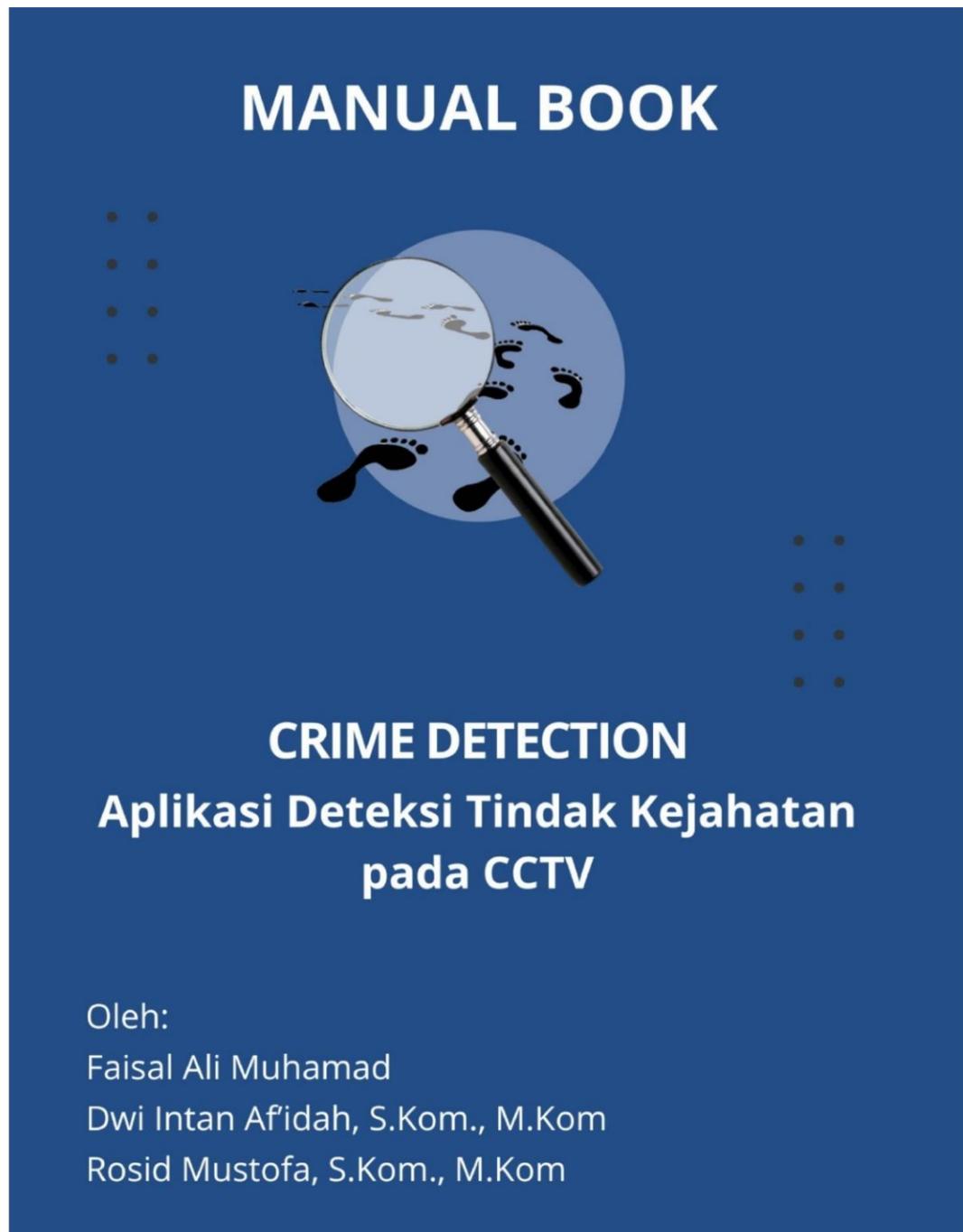
(Faisal Ali Muhamad)



(Dwi Intan Afidah, S.T., M.Kom.)



Rosid Mustofa, S.Kom, M.Kom



DAFTAR ISI

DAFTAR ISI.....	2
DAFTAR GAMBAR.....	3
1. PENDAHULUAN.....	4
1.1. Tujuan Pembuatan Dokumen.....	4
1.2. Deskripsi Umum Aplikasi.....	4
1.3. Deskripsi Dokumen.....	4
2. PERANGKAT YANG DIBUTUHKAN.....	4
2.1. Perangkat Lunak.....	4
2.2. Perangkat Keras.....	5
3. MENU DAN CARA PENGGUNAAN.....	5
3.1. Struktur Menu.....	5
3.2. Cara Penggunaan.....	7
1. <i>Splash Screen</i>	7
2. <i>Onboarding</i>	7
3. Halaman Login.....	9
4. Halaman Registrasi.....	10
5. Halaman Verifikasi Email.....	11
6. Halaman Utama.....	11
7. Halaman History dan Detail History.....	12
8. Halaman Profil Pengguna.....	13

DAFTAR GAMBAR

Gambar 1. Splash Screen	7
Gambar 2. Tampilan Onboarding	8
Gambar 3. Tampilan Halaman Login	9
Gambar 4. Tampilan Halaman Registrasi	10
Gambar 5. Tampilan Halaman Verifikasi Email	11
Gambar 6. Tampilan Halaman Utama	12
Gambar 7. Halaman History	13
Gambar 8. Halaman Profile	13

1. PENDAHULUAN

1.1. Tujuan Pembuatan Dokumen

Dokumen *user manual* ini dibuat untuk memberikan penjelasan dan gambaran atau sebagai panduan kepada pengguna mengenai penggunaan Aplikasi Mobile Untuk Deteksi Deteksi Jenis Tindak Kejahatan menggunakan Metode YOLOv5 melalui CCTV.

1.2. Deskripsi Umum Aplikasi

Aplikasi ini dirancang untuk melakukan pemantauan pada area perumahan dan juga bisa mendeteksi jenis tindak kejahatan secara *real-time* melalui kamera CCTV.

1.3. Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan Aplikasi Mobile Untuk Deteksi Jenis Tindak Kesehatan

1. Bagian 1

Bagian ini berisi informasi umum yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem, dan deskripsi dokumen.

2. Bagian 2

Bagian ini berisi perangkat yang dibutuhkan untuk penggunaan aplikasi meliputi perangkat keras dan perangkat lunak.

3. Bagian 3

Bagian ini berisi panduan atau cara penggunaan.

2. PERANGKAT YANG DIBUTUHKAN

2.1. Perangkat Lunak

Untuk menjalankan aplikasi ini, berikut beberapa perangkat lunak yang dibutuhkan yaitu:

1. Sistem Operasi Android. Aplikasi ini harus dijalankan pada perangkat yang menggunakan sistem operasi Android.
2. Koneksi Internet. Untuk menjalankan aplikasi dengan lancar, diperlukan koneksi internet yang stabil.
3. Penyimpanan Data.

2.2. Perangkat Keras

Selain perangkat lunak, aplikasi ini juga memerlukan perangkat keras, terutama CCTV dan smartphone Android. CCTV harus memiliki resolusi yang cukup tinggi untuk menangkap gambar dengan jelas dan detail, sementara smartphone Android digunakan untuk menerima notifikasi dan mengelola pengaturan aplikasi.

3. MENU DAN CARA PENGGUNAAN

3.1. Struktur Menu

Berikut struktur menu yang ada pada aplikasi SideFish adalah sebagai berikut:

1. *Splash Screen*
Tampilan awal saat aplikasi dibuka. Menampilkan logo aplikasi sebelum pengguna diarahkan ke halaman *onboarding*.
2. *Onboarding*
Tampilan untuk memperkenalkan fitur-fitur pada aplikasi.
3. Register
Menu ini digunakan untuk mendaftarkan akun baru. Pengguna harus mengisi informasi seperti nama, email, *password* dan nomor telpon untuk membuat akun.
4. Login
Menu ini digunakan untuk masuk ke aplikasi dengan akun yang sudah terdaftar. Pengguna harus memasukkan email dan kata sandi yang telah didaftarkan.
5. Input CCTV
Menu ini digunakan untuk menginputkan CCTV yang akan digunakan.
6. Riwayat
Menu ini menampilkan riwayat hasil deteksi yang ada pada CCTV. History tersebut mencakup informasi mengenai jenis kejahatan, tanggal deteksi, dan lokasi CCTV.
7. Detail History
Menu ini menampilkan detail dari history yang tercatat. Detail tersebut mencakup gambar, ip camera, jenis kejahatan, tanggal, waktu dan lokasi CCTV.

8. Profile

Menu ini digunakan untuk melihat dan mengedit informasi profil mereka, seperti nama, email, dan kata sandi.

3.2. Cara Penggunaan.

1. *Splash Screen*

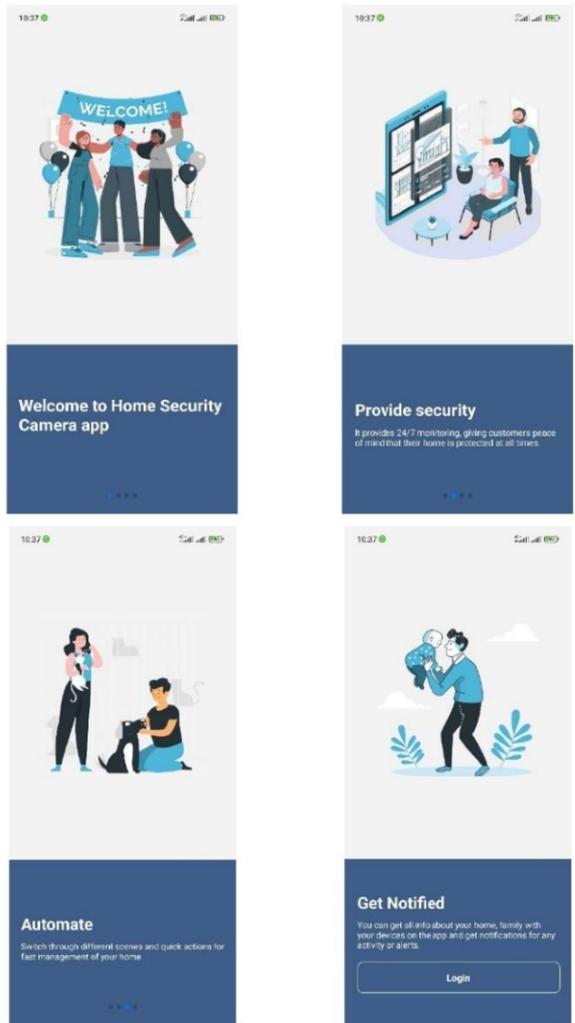
Buka aplikasi dan pastikan *smartphone* terhubung ke internet. Tunggu beberapa saat. Tampilan tersebut dapat dilihat pada Gambar 1.



Gambar 1. *Splash Screen*

2. *Onboarding*

Setelah itu aplikasi akan menampilkan halaman *onboarding* dimana terdapat empat layar. Anda harus geser ke kiri untuk melanjutkan dan memulai aplikasi. Klik button "Login" untuk masuk ke aplikasi. Tampilan tersebut dapat dilihat pada Gambar 2.



Gambar 2. Tampilan *Onboarding*

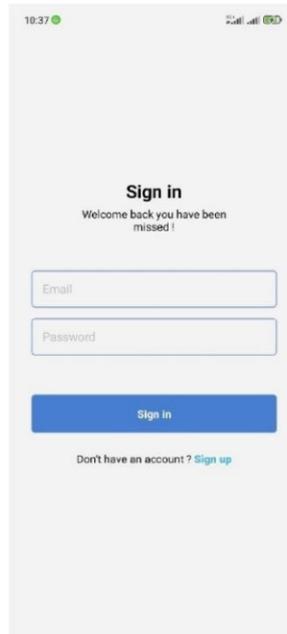
3. Halaman Login

Pada halaman ini, Anda dapat masuk ke aplikasi dengan menggunakan email dan *password* yang telah didaftarkan sebelumnya. Tampilan tersebut dapat dilihat pada Gambar 3. Berikut langkah-langkah nya:

- 1) Masukkan Email dan Password. Pastikan sudah terdaftar dan terverifikasi.
- 2) Klik button Sign In.
- 3) Aplikasi akan menampilkan halaman *home*.

Catatan:

- 1) Jika belum melakukan pendaftaran, Anda bisa klik text “Don’t have an account?” dan akan menampilkan halaman registrasi.



Gambar 3. Tampilan Halaman Login

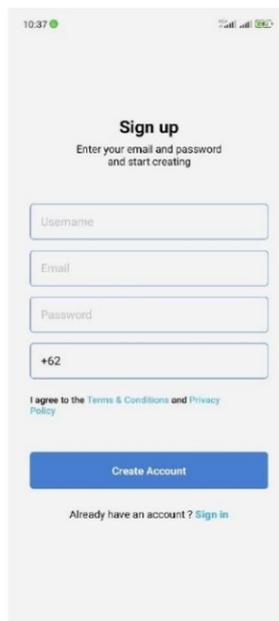
4. Halaman Registrasi

Pada halaman ini, Anda akan diarahkan untuk membuat akun baru supaya dapat mengakses semua fitur pada aplikasi. Tampilan tersebut dapat dilihat pada Gambar 4. Berikut langkah-langkah nya:

- 1) Terdapat empat formular untuk proses pendaftaran, masukkan informasi yang diperlukan seperti *Username*, Email, Password dan No. Telepon.
- 2) Klik button “Create Account”
- 3) Aplikasi akan menampilkan halaman otp.

Catatan:

- 1) Jika Anda sudah memiliki akun, Anda bisa klik text “Already have an account? Sign In”



Gambar 4. Tampilan Halaman Registrasi

5. Halaman Verifikasi Email

Halaman ini digunakan untuk memastikan bahwa alamat email yang Anda daftarkan valid dan dapat diakses oleh Anda. Tampilan tersebut dapat dilihat pada Gambar 5. Berikut langkah-langkahnya:

- 1) Masukkan kode OTP yang sudah terkirim ke Email yang Anda gunakan.
- 2) Klik button "Verify OTP"
- 3) Aplikasi akan menampilkan halaman login.

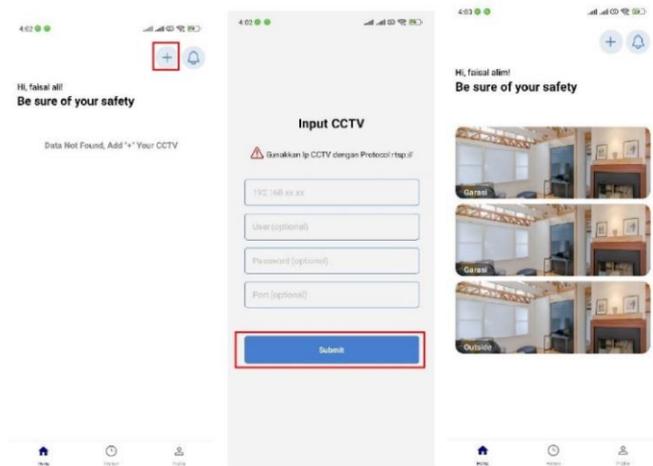


Gambar 5. Tampilan Halaman Verifikasi Email

6. Halaman Utama

Pada halaman utama merupakan halaman yang dilihat setelah Anda berhasil login ke aplikasi. Tampilan tersebut dapat dilihat pada Gambar 6. Berikut langkah-langkah yang harus dilakukan:

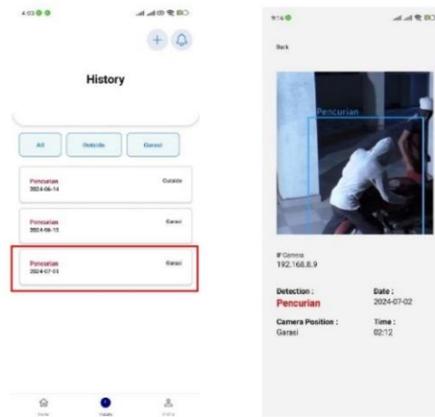
- 1) Klik tombol plus.
- 2) Isi formulir yang dibutuhkan yaitu ip camera, user, *password*, dan port.
- 3) Klik tombol submit untuk menyimpan.
- 4) CCTV akan ditampilkan di aplikasi.



Gambar 6. Tampilan Halaman Utama

7. Halaman History dan Detail History

Halaman ini digunakan untuk mencatat deteksi yang sudah dilakukan. Ketika Anda klik salah satu history, aplikasi akan menampilkan detail yang menampilkan informasi yaitu Hasil Deteksi, Tanggal, dan Lokasi CCTV. Berikut tampilan pada halaman ini dapat dilihat pada Gambar 7.



Gambar 7. Halaman History

8. Halaman Profil Pengguna

Halaman ini menampilkan detail dari profil Anda. Terdapat *username* dan email yang Anda gunakan. Berikut tampilan halaman profile dapat dilihat pada Gambar 15.



Gambar 8. Halaman Profile

DOKUMEN TEKNIS



CRIME DETECTION

Aplikasi Deteksi Tindak Kejahatan pada CCTV

Oleh:

Faisal Ali Muhamad

Dwi Intan Afidah, S.Kom., M.Kom

Rosid Mustofa, S.Kom., M.Kom

DAFTAR ISI

DAFTAR ISI	2
DAFTAR GAMBAR	3
Profile.....	4
Latar Belakang	4
Spesifikasi Teknis	6

DAFTAR GAMBAR

Gambar 1.1. Source Code Tampilan Registrasi	7
Gambar 1.2. Source Code OTP.....	8
Gambar 1.3. Source Code Login.....	8
Gambar 1.4. Source Code Tampilan Home	9
Gambar 1.5. Source Code Tampilan Edit Username	10
Gambar 1.6. Source Code Tampilan History	11
Gambar 1.7. Source Code Tampilan Detail History	12
Gambar 1.8. Source Code Tampilan Kesimpulan Profile	13
Gambar 2.1. Route Endpoint API	14
Gambar 2.2. Route Registrasi	15
Gambar 2.3. Route Login.....	15
Gambar 2.4. Route Login.....	16
Gambar 2.5. Upload Image ke Cloudinary.....	17
Gambar 2.6. Input Data CCTV.....	18
Gambar 2.7. Route List History	19
Gambar 2.8. Route Detail History	19

Profile

Crime Detection adalah aplikasi untuk mendeteksi tindak kejahatan menggunakan algoritma YOLOv5 melalui CCTV. Aplikasi ini digunakan untuk menilai dan mengidentifikasi aktivitas mencurigakan atau tindak kejahatan dengan cepat dan tepat hanya dengan menangkap rekaman CCTV. Hasil deteksi akan langsung ditampilkan dengan memunculkan bounding box serta label pada objek yang terdeteksi dalam rekaman dan menampilkannya pada histori laporan pengguna, sehingga pengguna dapat segera mengambil tindakan tanpa perlu khawatir terdapat kesalahan dalam pemantauan keamanan.

Latar Belakang

Tindak kejahatan di Indonesia merupakan insiden yang membahayakan ketertiban masyarakat dan melanggar peraturan hukum. Beberapa contoh kejahatan yang sering terjadi di Indonesia termasuk pencurian, penculikan, perampokan, penipuan, dan bahkan pembunuhan. Badan Pusat Statistik (BPS) Kriminal 2023 memberikan gambaran besar tentang situasi keamanan di Indonesia. Data dari Polri menunjukkan adanya fluktuasi dalam jumlah kejahatan, terutama dengan peningkatan yang signifikan pada tahun 2022, di mana total kejahatan mencapai 276.507, dengan crime rate sebesar 137. Sementara itu sepanjang tahun 2023 mengalami kenaikan sebesar 4,3% jika dibandingkan dengan tahun 2022 lalu. Total jumlah kejahatan sepanjang tahun 2023 sebanyak 288.472 perkara, naik 11.965 perkara jika dibandingkan dengan 2022 yang sebanyak 276.507 perkara. Meskipun ada fluktuasi dalam persentase penduduk yang menjadi korban kejahatan, tingkat pelaporan ke polisi tetap rendah, hanya sekitar 22,98% dari total korban.

Pada era digital saat ini, perkembangan teknologi telah memberikan dampak yang signifikan terhadap berbagai aspek kehidupan manusia, termasuk dalam bidang keamanan dan pengawasan. Perkembangan sistem deteksi kriminalitas merupakan salah satu aspek yang terus berkembang seiring dengan perkembangan teknologi informasi dan komunikasi. Dalam konteks ini, penggunaan aplikasi mobile untuk deteksi jenis tindak kejahatan melalui kamera CCTV menjadi semakin relevan karena dapat mendeteksi dan mencegah tindak kejahatan. Sistem deteksi kejahatan berbasis kamera CCTV telah menjadi salah satu solusi yang efektif dalam memantau keamanan suatu wilayah. Namun, kebanyakan sistem

yang ada masih bergantung pada manusia untuk menganalisis data yang diperoleh dari kamera CCTV, yang sering kali tidak efisien dan rentan terhadap human error.

Penelitian sebelumnya telah dilakukan untuk mengevaluasi hasil dan kinerja model YOLOv8 dalam mendeteksi serta mengidentifikasi senjata berbahaya dalam citra yang telah dimanipulasi sebelumnya. Hasil penelitian menunjukkan bahwa model tersebut mencapai presisi sebesar 84%, recall sebesar 77%, mAP sebesar 84%, dan F1-Score sebesar 88%, dengan waktu pelatihan selama 4 jam 6 menit. Pengujian dilakukan pada gambar grayscale, gambar yang diputar, pengaturan tingkat cahaya yang berbeda, gambar yang kabur, serta pengujian deteksi berdasarkan jarak, dan menunjukkan hasil yang memuaskan, meskipun masih membutuhkan peningkatan. Namun, perlu dicatat bahwa penelitian ini belum dilakukan secara real-time.

Berdasarkan permasalahan utama yang ingin diselesaikan adalah kurangnya efisiensi dalam mendeteksi jenis tindak kejahatan menggunakan sistem berbasis kamera CCTV. Oleh karena itu penelitian dilakukan untuk merancang dan mengimplementasikan sebuah aplikasi mobile yang dapat mendeteksi jenis tindak kejahatan secara real-time melalui kamera CCTV menggunakan metode You Only Look Once version 5 (YOLOv5).

Manfaat

1. Efektivitas pengawasan keamanan dengan memanfaatkan teknologi terkini semakin efisien.
2. Memudahkan masyarakat dalam memperoleh dan memasang kamera CCTV, yang diharapkan dapat meningkatkan rasa aman dan mengurangi tingkat kejahatan secara keseluruhan.
3. Mempermudah pencegahan dan respons terhadap kejahatan dengan mendeteksinya secara *real-time*.

Spesifikasi Teknis

Spesifikasi Teknis, meliputi:

1. Modul Pengguna
2. *Source Code*

Berikut uraian spesifikasi untuk pembangunan aplikasi:

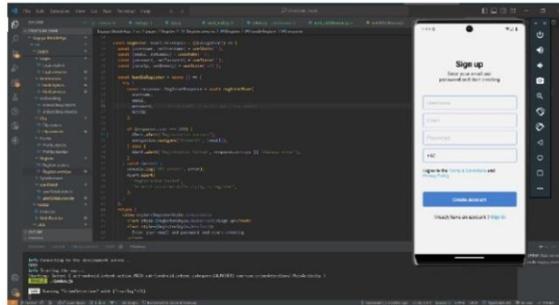
1. Flask Python
2. Visual Studio Code
3. Postman
4. Google Collaboration
5. Web Browser (Chrome, Edge)
6. React Native
7. Cloudinary

Berikut uraian spesifikasi modul:

1. Modul untuk pengguna
2. *Source Code*
 - 1) *User Interface*.

1. Tampilan Register

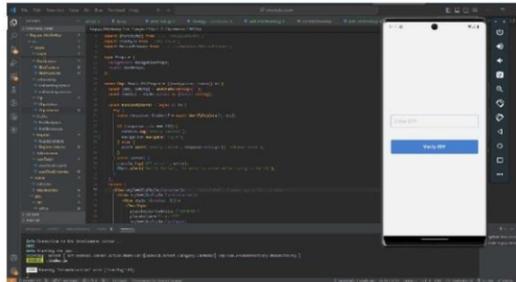
Tampilan registrasi ini menampilkan formulir yang terdiri dari empat input yaitu username, email, password, dan no telpon. Masing-masing dikelola oleh 'TextInput' untuk mengontrol teks yang diinput pengguna, serta divalidasi menggunakan 'value' dan 'onChange' untuk menampilkan inputan yang diisikan ke dalam form register untuk memastikan data diisi dengan benar. Halaman juga menyediakan tombol *Create Account* yang menggunakan fungsi 'handleRegister' untuk mengirim data ke server melalui API. Dan juga tersedia teks link 'Sign In' jika pengguna sudah memiliki akun. *Source Code* tersebut dapat dilihat pada Gambar 1.1.



Gambar 1.1. Source Code Tampilan Registrasi

2. Tampilan OTP

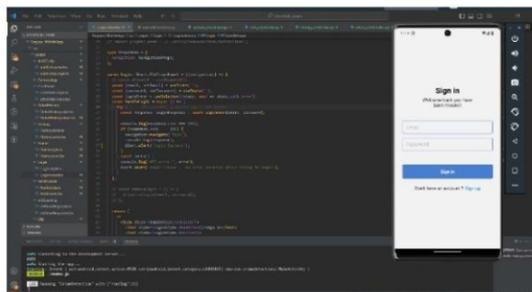
Halaman ini menggunakan berbagai komponen hook dari React Native seperti View, Alert, TextInput, serta mengimpor tipe, fungsi API. Pada awalnya, komponen ini mengharapkan props navigation dan route, serta mendefinisikan sebuah variabel state otp untuk menyimpan input OTP yang diberikan oleh pengguna. Email pengguna diambil dari route.params. Fungsi handleOtpVerif adalah fungsi asinkron yang digunakan untuk memverifikasi OTP dengan memanggil fungsi API VerifyOtp menggunakan email dan OTP yang diberikan. Jika verifikasi berhasil (kode respons 200), komponen ini akan menavigasi ke tampilan 'Login'. Jika verifikasi gagal, atau terjadi kesalahan dalam proses verifikasi, komponen ini akan menampilkan alert dengan pesan kesalahan yang sesuai. Pada bagian rendering, komponen ini merender sebuah View utama yang berisi TextInput untuk menampilkan input OTP dan sebuah tombol kustom ButtonPrimary yang akan memicu fungsi handleOtpVerif saat ditekan. Fungsi handleOtpVerif ini bertujuan untuk memverifikasi OTP dari pengguna, serta memberikan navigasi dan umpan balik yang sesuai berdasarkan hasil verifikasi. *Source Code* tersebut dapat dilihat pada Gambar 1.2.



Gambar 1.2. *Source Code* OTP

3. Tampilan Login

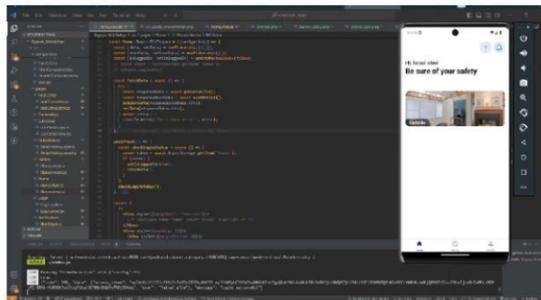
Tampilan login ini menyajikan sebuah formulir dengan dua input utama, yaitu email dan password, Masing-masing dikelola oleh 'TextInput' untuk mengontrol teks yang diinput pengguna, serta divalidasi menggunakan 'value' dan 'onChange' untuk menampilkan inputan yang diisikan ke dalam form login untuk memastikan data diisi dengan benar. Halaman juga menyediakan tombol *Sign In* yang menggunakan fungsi 'handleLogin' untuk mengirim data ke server melalui API. *Source Code* tersebut dapat dilihat pada Gambar 1.3.



Gambar 1.3. *Source Code* Login

4. Tampilan Home

Tampilan Home menampilkan list data CCTV dan nama pengguna. Beberapa hook dan komponen dari React Native serta utilitas seperti AsyncStorage dan fungsi API (`getDataCctv`, `userDetail`) diimpor. Komponen Home mendefinisikan beberapa state: data untuk menyimpan data CCTV, `userData` untuk menampilkan nama pengguna, dan `isLoggedIn` untuk mengecek status login. Fungsi `fetchData` digunakan untuk mengambil data CCTV dan username pengguna dari API, kemudian menyimpannya dalam state yang sesuai. Pada saat komponen dirender, `useEffect` dijalankan untuk memeriksa token login yang disimpan di AsyncStorage, dan jika token ada, menandakan pengguna sudah login dan memanggil `fetchData`. Tampilan komponen menampilkan salam untuk pengguna dengan username pengguna diikuti oleh daftar data CCTV yang tersedia. Jika tidak ada data CCTV, pesan "Data Not Found, Add '+' Your CCTV" akan ditampilkan. Komponen ini juga memiliki logika navigasi untuk berpindah ke tampilan kamera saat salah satu item CCTV disentuh. *Source Code* tersebut dapat dilihat pada Gambar 1.4.

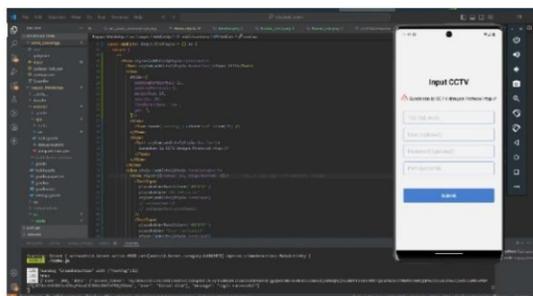


Gambar 1.4. *Source Code* Tampilan Home

5. Tampilan Add CCTV

Halaman input data CCTV. Pada tampilan ini mengimpor berbagai modul yang diperlukan termasuk `View`, `Text`, `TextInput`, serta beberapa utilitas khusus dan komponen kustom seperti `ButtonPrimary`, `Icon`, dan fungsi API `inputCctv`.

Komponen AddCctv mendefinisikan beberapa inputan untuk menyimpan data input seperti ip, lokasiCamera, userIp, passwordUser, path, port, dan image. Fungsi handleSubmit digunakan untuk mengirimkan data input ke API dengan memanggil fungsi inputCctv dan menangani responnya. Pada bagian rendering, komponen ini menampilkan form input dengan beberapa TextInput untuk menerima data IP, user, password, dan port CCTV, serta sebuah tombol kustom ButtonPrimary untuk mengirimkan data tersebut. Form ini juga menampilkan pesan peringatan menggunakan ikon dan teks yang mengingatkan pengguna untuk menggunakan protocol `rtsp://` pada IP CCTV. Semua elemen tersebut dirender dalam sebuah View. *Source Code* tersebut dapat dilihat pada Gambar 1.5.



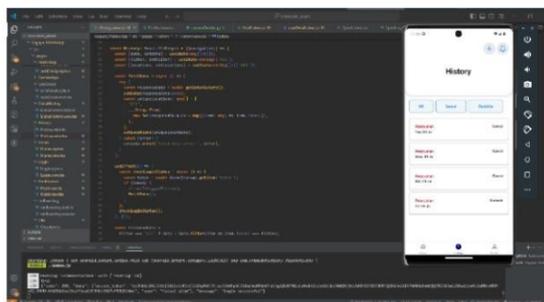
Gambar 1.5. *Source Code* Tampilan Edit Username

6. Tampilan History

Halaman history yang menampilkan data list riwayat CCTV berdasarkan filter lokasi. Pada awalnya, komponen ini mengimpor berbagai modul yang diperlukan seperti View, Text, ScrollView, TouchableOpacity, serta beberapa utilitas dan komponen kustom seperti CardList, FilterComponent, dan fungsi API `getDataHistory`. Komponen History mendefinisikan beberapa *state* untuk menyimpan data riwayat (*data*), filter lokasi (*filter*), dan daftar lokasi untuk (*locations*). Fungsi `fetchData` digunakan untuk mengambil data riwayat dari

API, kemudian disimpan dalam state data dan juga menyimpan daftar lokasi untuk dalam state locations.

Pada saat komponen dirender, `useEffect` dijalankan untuk memeriksa token login yang disimpan di `AsyncStorage`. Jika token ditemukan, maka fungsi `fetchData` akan dipanggil. Data riwayat yang ditampilkan disaring berdasarkan lokasi yang dipilih dalam filter. Jika filter adalah "All", semua data akan ditampilkan, sedangkan jika filter sesuai dengan lokasi tertentu, hanya data dengan lokasi tersebut yang akan ditampilkan. *Source Code* tersebut dapat dilihat pada Gambar 1.6.



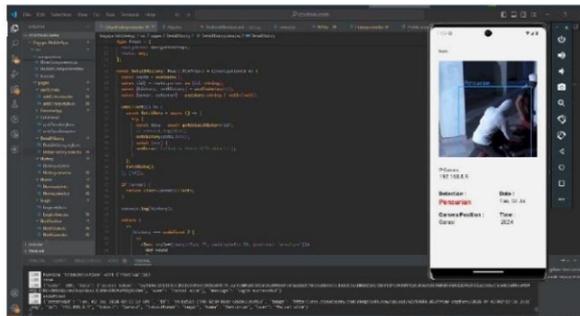
Gambar 1.6. *Source Code* Tampilan History

7. Tampilan Detail History

Tampilan detail history menampilkan detail dari riwayat CCTV yang dipilih. Komponen ini mengimpor berbagai modul yang diperlukan seperti `View`, `Text`, `ScrollView`, `Image`, serta beberapa utilitas dan komponen kustom seperti `useRoute` dari `'@react-navigation/native'` dan fungsi API `'getDetailHistory'`. Pada saat komponen dirender, `useEffect` dijalankan untuk memanggil fungsi `fetchData`. Jika terjadi error saat mengambil data, pesan error akan disimpan dalam state `error` dan ditampilkan pada layar.

Jika data history belum ada, komponen akan menampilkan pesan "Not Found". Jika data history ada, komponen akan menampilkan detail riwayat

CCTV seperti gambar, alamat IP kamera, nama deteksi, tanggal, lokasi kamera, dan waktu deteksi. *Source Code* tersebut dapat dilihat pada Gambar 1.7.



Gambar 1.7. *Source Code* Tampilan Detail History

8. Tampilan Kesimpulan Profile

Halaman profile pada aplikasi ini menampilkan dan mengelola informasi profil pengguna dalam aplikasi. beberapa state untuk menyimpan data pengguna yang diambil dari API melalui fungsi `userDetail`, dan memanfaatkan `useEffect` untuk mengecek status login saat komponen pertama kali dirender. Jika token ditemukan di `AsyncStorage`, fungsi `fetchData` akan dipanggil untuk mengambil data pengguna dan menyimpannya dalam state. Tampilan profil terdiri dari beberapa elemen, termasuk header "Profile", profil yang menampilkan username dan email pengguna, serta beberapa opsi lainnya seperti "Notification", "Support", "About us", dan "Logout". fungsi `handleLogout` digunakan untuk menghapus token dari `AsyncStorage` dan mengarahkan pengguna kembali ke halaman login. *Source Code* tersebut dapat dilihat pada Gambar 1.8.

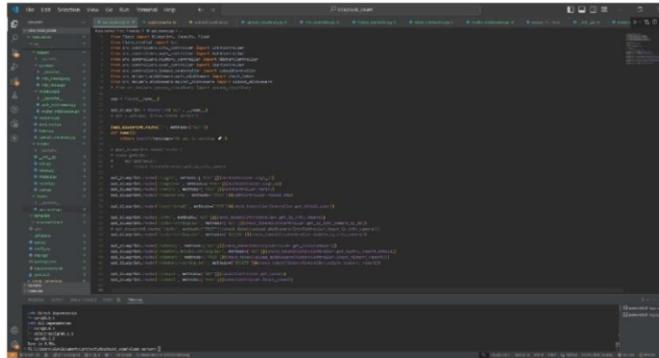


Gambar 1.8. *Source Code* Tampilan Kesimpulan Profile

2) Logic

1. Route Endpoint API

Bagian Kode ini mendefinisikan konfigurasi rute API menggunakan Flask untuk aplikasi pendeteksi kejahatan. Di dalam aplikasi Flask, sebuah Blueprint bernama `api` didefinisikan untuk mengelompokkan semua rute API. Ada berbagai rute yang diatur untuk menangani operasi otentikasi, pengelolaan pengguna, CCTV, riwayat, dan lokasi. Rute ini mencakup metode HTTP seperti GET, POST, dan DELETE. Beberapa rute, seperti rute untuk mendapatkan detail pengguna dan pengelolaan CCTV serta riwayat, dilindungi oleh middleware `check_token` untuk memastikan hanya pengguna yang terotentikasi yang dapat mengaksesnya. Selain itu, middleware `upload_middleware` digunakan untuk mengelola unggahan file pada rute tertentu. Fungsi dari setiap rute diarahkan ke metode di dalam berbagai controller yang didefinisikan di dalam direktori `controllers`. Semua rute ini digabungkan ke dalam blueprint dan diatur untuk menangani berbagai permintaan yang berhubungan dengan otentikasi, manajemen CCTV, riwayat kejadian, dan lokasi. *Source Code* tersebut dapat dilihat pada Gambar 2.1.



Gambar 2.1. Route Endpoint API

2. Route Registrasi

Bagian ini merupakan fungsi `sign_up()` yang digunakan untuk mendaftarkan pengguna baru dalam aplikasi. Fungsi ini pertama-tama mengambil data JSON dari permintaan dan mengekstrak username, email, notelp, dan password. Password dienkripsi menggunakan `bcrypt` sebelum disimpan. Fungsi ini juga menghasilkan ID unik untuk pengguna menggunakan `UUID` dan mengirim email verifikasi yang mengandung OTP ke email pengguna. Kemudian, fungsi mencoba untuk membuat entri baru dalam basis data dengan menggunakan model `User` dan menyimpan perubahan tersebut. Jika pendaftaran berhasil, fungsi mengembalikan pesan sukses. Jika terjadi kesalahan, seperti email yang sudah terdaftar, perubahan basis data dibatalkan dan pesan kesalahan yang sesuai dikembalikan. Jika ada kesalahan lainnya, fungsi mengembalikan pesan kesalahan server internal. *Source Code* tersebut dapat dilihat pada Gambar 2.2.

```

class AuthController:
    def __init__(self):
        self.db = db

    def register(self):
        data = request.get_json()
        username = data.get('username')
        email = data.get('email')
        password = data.get('password')
        if not password:
            return jsonify({'message': 'password is required'})

        user = User.query.filter_by(username=username, email=email).first()
        if user:
            return jsonify({'message': 'user already exists', 'code': 400})

        user = User(username=username, email=email, password=password)
        self.db.add(user)
        self.db.commit()

        return jsonify({'message': 'user registered successfully', 'code': 200, 'data': user.username})

    def login(self):
        data = request.get_json()
        username = data.get('username')
        password = data.get('password')
        if not password:
            return jsonify({'message': 'password is required'})

        user = User.query.filter_by(username=username).first()
        if not user:
            return jsonify({'message': 'user not found', 'code': 400})

        if not user.check_password(password):
            return jsonify({'message': 'password is incorrect', 'code': 400})

        return jsonify({'message': 'login successful', 'code': 200, 'data': user.username})

    def verify_email(self, email):
        user = User.query.filter_by(email=email).first()
        if not user:
            return jsonify({'message': 'user not found', 'code': 400})

        otp = generate_otp()
        user.verify_otp = otp
        self.db.commit()

        send_email(email, otp)

        return jsonify({'message': 'otp sent successfully', 'code': 200, 'data': otp})

```

Gambar 2.2. Route Registrasi

3. Verifikasi Email

Bagian ini merupakan fungsi `send_verification_email()` yang digunakan untuk mengirim email verifikasi kepada pengguna baru yang mendaftar. Fungsi ini menghasilkan kode OTP (One-Time Password) acak enam digit. Kemudian, fungsi mencoba untuk mengirim email menggunakan ekstensi mail dari aplikasi Flask. Email yang dikirim berisi kode OTP dan ditujukan kepada alamat email yang diberikan, dengan subjek "Email Verification". Jika pengiriman email mengalami kesalahan, pesan kesalahan dicetak di konsol. Fungsi ini akhirnya mengembalikan kode OTP yang telah dihasilkan. *Source Code* tersebut dapat dilihat pada Gambar 2.3.

```

def send_verification_email(email):
    user = User.query.filter_by(email=email).first()
    if not user:
        return jsonify({'message': 'user not found', 'code': 400})

    otp = generate_otp()
    user.verify_otp = otp
    db.commit()

    msg = Message('Email Verification', sender='no-reply@localhost', recipients=[email])
    msg.body = f'Your verification code is: {otp}'
    mail = Mail(app)
    mail.send(msg)

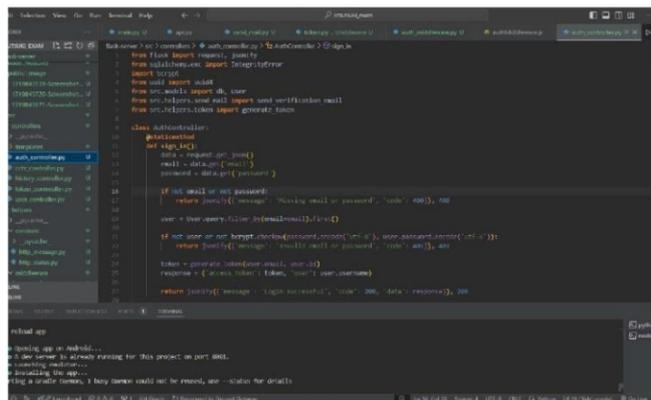
    return jsonify({'message': 'otp sent successfully', 'code': 200, 'data': otp})

```

Gambar 2.3. Route Login

4. Route Login

Pada bagian ini merupakan fungsi `sign_in()` yang digunakan untuk menangani proses login pengguna. Pertama, fungsi ini mengambil data JSON dari permintaan, kemudian mengekstrak email dan password dari data tersebut. Jika email atau password tidak disertakan, fungsi akan mengembalikan respons dengan pesan kesalahan dan kode status 400. Selanjutnya, fungsi mencari pengguna di basis data berdasarkan email yang diberikan. Jika pengguna tidak ditemukan atau password tidak cocok, fungsi akan mengembalikan respons dengan pesan kesalahan dan kode status 401. Jika validasi berhasil, fungsi menghasilkan token akses menggunakan fungsi `generate_token()` dengan email dan ID pengguna, kemudian mengembalikan respons yang berisi token akses dan nama pengguna dengan pesan sukses dan kode status 200. *Source Code* tersebut dapat dilihat pada Gambar 2.4.



```
def sign_in
  data = request.raw_post_data
  result = data.get('email')
  password = data.get('password')

  if not email or not password
    return jsonify(message: "Missing email or password", code: 400), 400
  end

  user = User.query.filter_by(email=email).first()

  if not user or not bcrypt.checkpw(password.encode('utf-8'), user.password.encode('utf-8'))
    return jsonify(message: "Invalid email or password", code: 401), 401
  end

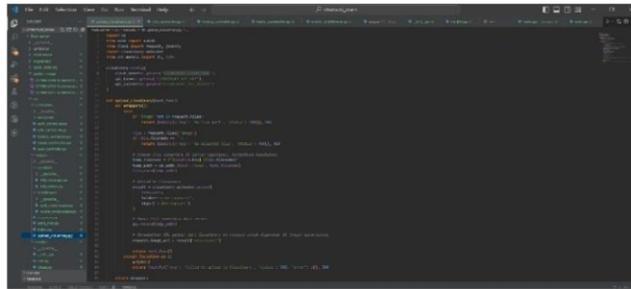
  token = generate_token(email=email, user_id=user.id)
  response = { access_token: token, user: user.username }

  return jsonify(message: "Login successful", code: 200, data: response), 200
end
```

Gambar 2.4. Route Login

5. Upload Image ke Cloudinary

Pada bagian ini mendefinisikan fungsi `upload_cloudinary()` yang bertindak sebagai middleware untuk mengunggah gambar ke Cloudinary sebelum melanjutkan ke fungsi berikutnya. Fungsi ini pertama-tama memeriksa apakah ada file gambar dalam permintaan. Jika tidak ada, atau jika file tidak dipilih, fungsi akan mengembalikan pesan kesalahan. File yang diterima disimpan sementara di server dengan nama file unik. Selanjutnya, file diunggah ke Cloudinary di folder 'crime-capture/' dan diberi tag 'crime-capture'. Setelah berhasil diunggah, file sementara dihapus dari server, dan URL gambar yang dihasilkan dari Cloudinary ditambahkan ke permintaan untuk digunakan di fungsi selanjutnya. Jika terjadi kesalahan selama proses ini, pesan kesalahan akan dikembalikan. *Source Code* tersebut dapat dilihat pada Gambar 2.5.

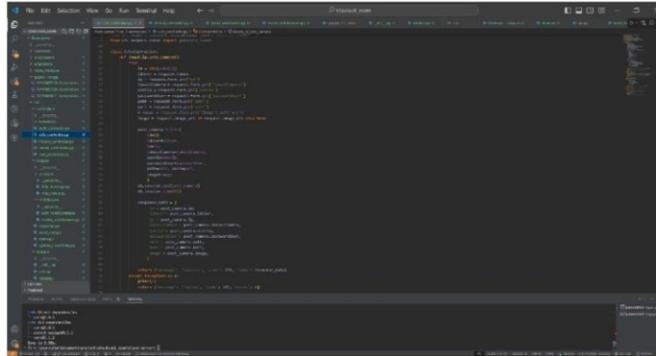


Gambar 2.5. Upload Image ke Cloudinary

6. Input Data CCTV

Pada bagian ini adalah fungsi `input_ip_cctv_camera()` yang digunakan untuk menyimpan data kamera CCTV ke dalam basis data. Fungsi ini pertama-tama menghasilkan ID unik untuk kamera CCTV dan mengambil ID pengguna dari token permintaan. Selanjutnya, fungsi ini mengambil berbagai informasi dari formulir permintaan seperti alamat IP kamera, lokasi kamera, nama pengguna dan kata sandi untuk IP, path, port, dan URL gambar. Data ini kemudian disimpan dalam objek `Cctv` yang baru dan ditambahkan ke sesi basis

data untuk disimpan. Jika operasi berhasil, fungsi ini mengembalikan respons dengan pesan sukses dan data kamera yang disimpan. Namun, jika terjadi kesalahan, fungsi ini akan mencetak pesan kesalahan dan mengembalikan respons gagal dengan kode kesalahan 500. *Source Code* tersebut dapat dilihat pada Gambar 2.6.



Gambar 2.6. Input Data CCTV

7. Route List History

Pada bagian ini, terdapat Fungsi `get_historyReport()` digunakan untuk mendapatkan daftar riwayat laporan pengguna dari basis data berdasarkan `userId` yang diperoleh dari token permintaan. Fungsi ini mencoba untuk mengambil semua laporan yang sesuai dengan `userId`, mengurutkannya berdasarkan waktu pembuatan laporan secara menurun. Setiap laporan kemudian diubah menjadi bentuk kamus (dictionary) menggunakan metode `to_dict()`. Jika berhasil, fungsi ini mengembalikan respons dengan pesan sukses, kode status 200, dan data laporan. Jika terjadi kesalahan, fungsi mencetak pesan kesalahan di konsol dan mengembalikan respons dengan pesan gagal, kode status 400, dan rincian kesalahan. *Source Code* tersebut dapat dilihat pada Gambar 2.7.

```

def get_historyReport():
    try:
        idUser = request.token
        get_history = Report.query.filter_by(userId=idUser).order_by(Report.createdAt.desc()).all()
        response_data = [report.to_dict() for report in get_history]
        return {'message': 'success', 'code': 200, 'data': response_data}
    except Exception as e:
        print(e)
        return {'message': 'failed', 'code': 400, 'error': e}

```

Gambar 2.7. Route List History

8. Route Detail History

Pada bagian ini, terdapat Fungsi `get_histy_report_detail(id)` digunakan untuk mendapatkan detail laporan riwayat berdasarkan ID laporan yang diberikan. Fungsi ini mencoba mengambil laporan dari basis data yang sesuai dengan ID tersebut menggunakan metode `filter_by` dan `first`. Jika laporan ditemukan, data laporan akan diubah menjadi format kamus (dictionary) yang berisi informasi seperti ID, gambar, IP CCTV, nama, lokasi, lokasi rumah, nama pengguna, dan waktu pembuatan laporan. Jika berhasil, fungsi ini mengembalikan respons dengan pesan sukses, kode status 200, dan data laporan. Jika terjadi kesalahan, fungsi akan mencetak pesan kesalahan di konsol dan mengembalikan respons dengan pesan gagal, kode status 400, dan rincian kesalahan. *Source Code* tersebut dapat dilihat pada Gambar 2.8.

```

def get_histy_report_detail(id):
    try:
        item = Report.query.filter_by(id=id).first()
        report = {
            "id": item.id,
            "image": item.image,
            "ip": item.ip_cctv,
            "name": item.name,
            "location": item.location,
            "location_rumah": item.location_rumah,
            "user": item.user.username,
            "createdAt": item.createdAt
        }
        return {'message': 'success', 'code': 200, 'data': report}
    except Exception as e:
        print(e)
        return {'message': 'failed', 'code': 400, 'error': e}

```

Gambar 2.8. Route Detail History

Lampiran 5. Sertiifikat HKI Yang Terbit

 <p>REPUBLIC INDONESIA KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA</p> <h1 style="text-align: center;">SURAT PENCATATAN CIPTAAN</h1>	
<p>Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:</p>	
Nomor dan tanggal permohonan	: EC00202463715, 11 Juli 2024
Pencipta	
Nama	: Faisal Ali Muhamad, Dwi Intan Afidah, S.T., M.Kom. dkk
Alamat	: Jalan Cendrawasih Lontrong 12, RT 06/RW 04, Kelurahan Randugunting, Kecamatan Tegal Selatan, Kota Tegal, Propinsi Jawa Tengah 52131, Tegal Selatan, Tegal, Jawa Tengah, 52131
Kewarganegaraan	: Indonesia
Pemegang Hak Cipta	
Nama	: Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Alamat	: Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Tegal, Jawa Tengah 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Aplikasi Mobile Untuk Deteksi Jenis Tindak Kejahatan Menggunakan Metode YOLOv5 Melalui Kamera CCTV
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 11 Juli 2024, di Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	: 000639070
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL u.b Direktur Hak Cipta dan Desain Industri
	 IGNATIUS M.T. SILALAHI NIP. 196812301996031001
<p>Disclaimer: Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.</p>	

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Faisal Ali Muhamad	Jalan Cendrawasih Lontrong 12, RT 06/RW 04, Kelurahan Randugunting, Kecamatan Tegal Selatan, Kota Tegal, Propinsi Jawa Tengah 52131, Tegal Selatan, Tegal
2	Dwi Intan Afidah, S.T., M.Kom.	Desa Grinting RT 003/RW 001, Kecamatan Bulakamba, Kabupaten Brebes, Propinsi Jawa Tengah 52253, Bulakamba, Brebes
3	Rosid Mustofa, S.Kom, M.Kom.	Desa Kedungbanteng, RT 25/RW 12, Kecamatan Kedung Banteng, Kabupaten Tegal, Propinsi Jawa Tengah 52472, Kedung Banteng, Tegal



Lampiran 6. Lembar Bimbingan



**SARJANA TERAPAN TEKNIK INFORMATIKA
POKITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Faisal Ali Muhamad
Nim : 20090153
No. Ponsel : 0895358958540
Judul TA : Aplikasi Mobile untuk Deteksi Jenis Tindak Kejahatan menggunakan Metode YOLOv5 melalui Kamera CCTV

Dosen Pembimbing I : Dwi Intan Af'idah, S.T., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	19/03/2024	Konsep Produk	Dihistory dapat dimunculkan time, ip camera, nama camera	<i>Dwi Intan Af'idah</i>
2.	04/04/2024	Konsep Produk	melakukan modeling untuk mendeteksi gerakan mencurigakan / tindak kejahatan	<i>Dwi Intan Af'idah</i>
3.	25/04/2024	Konsep Produk	Melanjutkan mengerjakan integrasi aplikasi dan memperbaiki akurasi model	<i>Dwi Intan Af'idah</i>
4.	17/05/2024	Konsep Produk	Implementasi model dan integrasi dengan cctv	<i>Dwi Intan Af'idah</i>
5.	31/05/2024	Konsep Produk	Memperbaiki konfigurasi dengan kamera cctv	<i>Dwi Intan Af'idah</i>
6.	28/06/2024	Konsep Produk	Melanjutkan manual book dan implementasi dengan ip kamera laptop untuk sementara	<i>Dwi Intan Af'idah</i>



**SARJANA TERAPAN TEKNIK INFORMATIKA
POKITEKNIK HARAPAN BERSAMA**

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
7.	8/7/2024	Manual Book	Pengajuan HKI dan melanjutkan menyusun laporan	
8.	12/7/2024	Laporan	- Revisi Bab 1 - Implementasi upload data history	
9.	16/7/2024	Laporan	perbaiki sesuai Catatan	
10.	17/7/2024	Laporan	ACC	

Tegal, 2024
Dosen Pembimbing I

Dwi Intan Af'idah, S.T., M.Kom.
NIPY. 11.020.470



SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN SKRIPSI

Nama : Faisal Ali Muhamad
Nim : 20090153
No. Ponsel : 0895358958540
Judul TA : Rancang Bangun Aplikasi Mobile untuk Deteksi Jenis Tindak Kejahatan menggunakan Metode YOLOv5 melalui Kamera CCTV
Dosen Pembimbing II : Rosid Mustofa, S.Kom, M.Kom

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1	21/3/24	Konsep awal	roba di tuliskan berjalan langu prota sgr.	
2	22/5/24	Prudu awal	lagu implementasi model aga bua cetera otomat	
3	12/24/6	Aplikasi	diraphen yah dan roba krat use orang.	
4	27/24/6	Tulisan bab 1-2	aku tyloyy yah.	



SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
5	3/7 ²¹	tulisa	Bab 3 cek lagi	Az.
6	5/7 ²¹	T.U.K	Bab 1 - 3 OK	Az.
7	10/7 ²¹	produk	cek dan uji	Az.
8	16/7 ²¹	produk + Tulisan	Assidang	Az.

Tegal, 16/7 - 2024
Dosen Pembimbing II

Rosid Mustofa, S.Kom, M.Kom.