KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)
Terakreditasi Nomor 204/E/KPT/2022 | Vol. 6, No. 2, April (2025), pp. 403-410

Penerapan Metode *Rapid Application Development* Untuk Perancangan Arsitektur Presensi Berbasis Pengenalan Wajah dengan *Amazon Web Service* (AWS) Pada STMIK Widya Cipta Dharma

Michael Steeven[1], *Heny Pratiwi*[2*], *Siti Lailiyah*[3]
[1]*Prodi Teknik Informatika, STMIK Widya Cipta Dharma, Indonesia*
[2]*Prodi Sistem Informasi, STMIK Widya Cipta Dharma, Indonesia*
E-mail: 2143093@wicida.ac.id[1], *henypratiwi@wicida.ac.id*[2*],
lailiyah@wicida.ac.id[3]

Abstract

The Rapid Application Development (RAD) method is a software development approach that emphasizes speed and flexibility in building systems through iterations that actively involve users. In the increasingly advancing digital era, the RAD method has become an effective solution for creating software that meets user needs in a shorter time compared to traditional methods. One relevant implementation of RAD is in the development of a face recognition-based attendance system, which requires accuracy, efficiency, and speed in processing biometric data. This research aims to design a face recognition-based attendance system using Amazon Web Service (AWS) Rekognition with the RAD method. With the RAD approach, this system is developed iteratively through rapid design, prototyping, and continuous user feedback. The results of this study show that the application of RAD in developing a face recognition-based attendance system improves the efficiency of attendance recording and minimizes the risk of fraud in the attendance process.

Keywords: *Rapid Application Development, Attendance, Face Recognition, Amazon Rekognition.*

Abstrak

Metode Rapid Application Development (RAD) merupakan pendekatan pengembangan perangkat lunak yang menekankan kecepatan dan fleksibilitas dalam membangun sistem melalui iterasi yang melibatkan pengguna secara aktif. Dalam era digitalisasi yang semakin berkembang, metode RAD menjadi solusi yang efektif dalam menciptakan perangkat lunak yang sesuai dengan kebutuhan pengguna dalam waktu yang lebih singkat dibandingkan metode tradisional. Salah satu implementasi RAD yang relevan adalah dalam pengembangan sistem presensi berbasis pengenalan wajah, yang membutuhkan ketepatan, efisiensi, dan kecepatan dalam pemrosesan data biometrik. Penelitian ini bertujuan untuk merancang sistem presensi berbasis pengenalan wajah menggunakan layanan Amazon Web Service (AWS) Rekognition dengan metode RAD. Dengan pendekatan RAD, sistem ini dikembangkan secara iteratif melalui perancangan cepat, prototipe, serta umpan balik pengguna yang berkelanjutan. Hasil dari penelitian ini menunjukkan bahwa penerapan RAD mampu meningkatkan efisiensi pencatatan kehadiran dan meminimalkan risiko kecurangan dalam proses absensi.

Kata kunci: *Rapid Application Development, Presensi, Pengenalan Wajah, Amazon Rekognition.*

1. Pendahuluan

Metode *Rapid Application Development* (RAD) adalah salah satu pendekatan dalam pengembangan perangkat lunak yang menekankan kecepatan dan fleksibilitas

ISSN: 2720-992X
Copyright © KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)

dalam membangun sistem. Berbeda dengan metode pengembangan tradisional seperti Waterfall, yang bersifat linear dan membutuhkan waktu lama dalam setiap tahapannya, RAD memungkinkan pengembang untuk bekerja dalam siklus iteratif yang lebih cepat dan efisien. Metode ini pertama kali diperkenalkan oleh James Martin pada tahun 1991 sebagai solusi terhadap kendala dalam pengembangan perangkat lunak yang sering kali memakan waktu lama dan tidak fleksibel terhadap perubahan kebutuhan pengguna[1].

Keunggulan utama RAD terletak pada kemampuannya untuk menghasilkan prototipe dalam waktu singkat dan mempercepat pengujian serta perbaikan sistem berdasarkan umpan balik pengguna. Dengan demikian, metode ini sangat cocok untuk proyek yang memiliki persyaratan yang dapat berubah seiring waktu atau memerlukan keterlibatan langsung dari pengguna akhir selama proses pengembangannya[1].

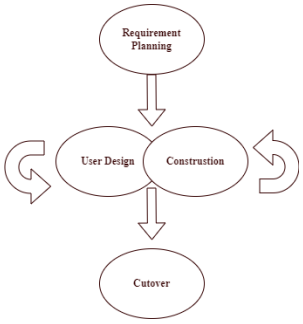
Dalam dunia industri dan akademik, penerapan RAD semakin berkembang, terutama dalam membangun sistem yang memerlukan perubahan cepat dan adaptasi terhadap kebutuhan pengguna. Salah satu contoh implementasi RAD yang memiliki dampak signifikan adalah sistem presensi berbasis pengenalan wajah. Sistem ini mengandalkan teknologi biometrik untuk mendeteksi dan mengenali wajah seseorang guna mencatat kehadiran secara otomatis. Dalam pengembangannya, penggunaan metode RAD memungkinkan perancangan sistem yang lebih fleksibel dan dapat diperbaiki secara berkelanjutan berdasarkan hasil uji coba yang dilakukan bersama pengguna.

Amazon Web Services (AWS) menyediakan layanan Amazon Rekognition yang memungkinkan pengolahan gambar dan analisis wajah secara otomatis menggunakan kecerdasan buatan. Layanan ini dapat digunakan untuk membangun sistem presensi yang lebih aman, akurat, dan efisien. Dengan mengintegrasikan metode RAD dalam pengembangan sistem ini, diharapkan proses implementasi dapat dilakukan dengan lebih cepat, sementara keandalan sistem tetap terjaga. Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem presensi berbasis pengenalan wajah menggunakan layanan AWS Rekognition dengan metode RAD. Dengan pendekatan ini, diharapkan sistem dapat lebih adaptif terhadap kebutuhan pengguna dan mampu mengatasi berbagai tantangan yang sering muncul dalam pencatatan kehadiran secara manual[2].

2. Metodologi Penelitian

2.1. *Rapid Application Development* (RAD)

Rapid Application Development (RAD) adalah pendekatan pengembangan perangkat lunak yang menekankan pada siklus pengembangan yang cepat dengan melibatkan pengguna dalam setiap tahapan. RAD merupakan model proses perangkat lunak yang menekankan pada daur pengembangan hidup yang singkat, dan versi adaptasi cepat dari metode Waterfall dengan menggunakan kontruksi komponen[3].



Gambar 1. Tahapan Metode RAD

RAD terbagi menjadi empat tahapan yang terstruktur dan saling bergantungan satu dengan yang lainnya, yaitu:

a) *Reqruitments Planning*
Pada tahap ini, user dan analyst melakukan semacam pertemuan untuk melakukan identifikasi tujuan dari aplikasi atau sistem dan melakukan identifikasi kebutuhan informasi untuk mencapai tujuan. Pada tahap ini hal terpenting adalah adanya keterlibatan dari kedua belah pihak, bukan hanya sekedar persetujuan akan proposal yang sudah dibuat. Untuk lebih jauh lagi, keterlibatan user bukan hanya dari satu tingkatan pada suatu organisasi, melainkan beberapa tingkatan organisasi sehingga informasi yang dibutuhkan untuk masing-masing user dapat terpenuhi dengan baik[1].

b) *User Design*
Pada tahap ini dilakukan proses perancangan desain, sesuai dengan hasil dari tahap sebelumnya. Proses pembuatan desain sistem yang akan dibuat mencakup *Use Case Diagram* dan *activity diagram*[1].

c) *Construction*
Tahapan ini mencakup pengembangan sistem berdasarkan desain yang telah dirancang sebelumnya. Pengembangan fitur pengenalan wajah menggunakan layanan Amazon Rekognition untuk meningkatkan akurasi dan efisiensi proses pencatatan kehadiran[1].

d) *Cutover*
Pada tahap ini, *Black Box Testing* digunakan untuk pengujian fungsionalitas sistem. Selain itu, pengujian dilakukan juga menggunakan *User Experience Questionnaire* (UEQ) untuk mengevaluasi tingkat kepuasan dan pengalaman pengguna saat menggunakan perangkat lunak[1].

2.2. *Black Box*

Blackbox Testing adalah teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional tanpa pengetahuan internal atau struktur kode sumber dari sistem yang diuji. Fokus utama adalah pada input dan output sistem untuk mengevaluasi apakah perangkat lunak berperilaku sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini melibatkan analisis fungsionalitas *eksterna*l perangkat lunak dan mencoba mencari *bug* atau kesalahan dengan memasukkan berbagai jenis *input* dan memeriksa apakah *output* yang dihasilkan sesuai dengan harapan. Dengan pendekatan ini, pengujian *Black Box* memastikan bahwa perangkat lunak berkinerja dengan benar dari perspektif pengguna tanpa memerhatikan detail internal implementasinya[4].

2.3. *User Experience Questionnaire* (UEQ)

Experience Questionnaire (UEQ) digunakan untuk mengukur bagaimana pengguna berinteraksi dengan produk atau sistem tertentu. Tiga aspek utama terdiri dari *attractiveness*, *pragmatic quality*, dan *hedonic quality*. *Pragmatic quality* mengacu pada bagaimana layanan berfungsi atau melakukan tugasnya, seperti kegunaannya. *Hedonic quality* mengacu pada ekspresi diri dan nilai-nilai pribadi pengguna, seperti yang berkaitan dengan perasaan atau emosi mereka. Dalam UEQ, tiga aspek terdiri dari 6 skala ini: daya tarik (*attractiveness*), kejelasan (*perspicuity*), efisiensi (*efficiency*), ketepatan (*dependability*), stimulasi *(stimulation*), dan kebaruan[1].

2.4. Pengenalan Wajah

Pengenalan wajah adalah teknik biometrik yang sangat berguna untuk mengidentifikasi wajah seseorang dengan memisahkan area wajah dari latar belakang dalam citra masukan. Proses ini memeriksa apakah citra memiliki ciri wajah, dan jika ya, citra wajah dipisahkan dari latar belakang. Menurut US *Government Accountability Office*

(2015), empat komponen yang dibutuhkan untuk pengenalan wajah adalah kamera, *faceprint*, dan *database*[2].

2.5. *Amazon Web Services* (AWS)

Amazon Web Services (AWS) adalah platform komputasi awan yang menyediakan berbagai layanan infrastruktur teknologi informasi berbasis cloud. AWS memungkinkan pengembang untuk membangun, mengelola, dan menskalakan aplikasi dengan cepat tanpa harus mengelola infrastruktur fisik[2]. Layanan AWS yang digunakan dalam penelitian ini meliputi:

a) Amazon Rekognition: Layanan ini memungkinkan deteksi, analisis, dan perbandingan wajah menggunakan pembelajaran mesin. Amazon Rekognition dapat mengenali wajah secara real-time dan membandingkannya dengan database gambar yang telah disimpan di Amazon S3.
b) Amazon S3 (Simple Storage Service): Penyimpanan objek yang digunakan untuk menyimpan dataset gambar wajah yang akan dibandingkan oleh Amazon Rekognition. Amazon S3 menawarkan skalabilitas tinggi, keamanan, dan durabilitas untuk menyimpan data dengan kapasitas besar.
c) AWS Lambda: Layanan komputasi tanpa server yang memungkinkan eksekusi kode secara otomatis sebagai respons terhadap peristiwa tertentu, seperti pengunggahan gambar ke Amazon S3. AWS Lambda mengurangi kebutuhan akan infrastruktur server konvensional, sehingga sistem menjadi lebih efisien dan hemat biaya.
d) Amazon DynamoDB: Database NoSQL yang digunakan untuk menyimpan hasil deteksi wajah dan metadata pengguna. DynamoDB menawarkan kinerja tinggi dengan latensi rendah, sehingga cocok untuk aplikasi real-time seperti sistem presensi berbasis pengenalan wajah.
e) Amazon API Gateway: Layanan yang memungkinkan pembuatan, pengelolaan, dan pengamanan API yang digunakan oleh frontend aplikasi untuk berkomunikasi dengan layanan backend AWS lainnya.

Dengan menggunakan kombinasi layanan di atas, sistem presensi berbasis pengenalan wajah dapat berjalan dengan optimal dan memiliki skalabilitas tinggi.

2.6. *Unified Modeling Language* (UML)

Unified Modeling Language adalah notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendekskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek[5]. *Unified Modelling Language* (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industry untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak[6]. UML yang digunakan diantaranya adalah sebagai berikut:

1) *Use Case Diagram*
   Use case merupakan diagram aktivitas yang digunakan untuk membuat model proses bisnis berdasarkan perspektif pengguna dan gambaran tanggapan atas permintaan pengguna. *Use Case Diagram* digunakan untuk mengidentifikasi dan merinci requirements yang untuk memperjelas proses developer pada saat mengembangkan sistem[5].
2) *Activity Diagram*
   Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan aktor[5].

3. Hasil dan Pembahasan

3.1. *Requirments planning*

Sistem presensi ini menggunakan AWS Rekognition untuk:

a) Mendeteksi wajah dari kamera.

b) Membandingkan wajah dengan database yang tersimpan di Amazon S3

c) Menyimpan hasil verifikasi kehadiran di DynamoDB.

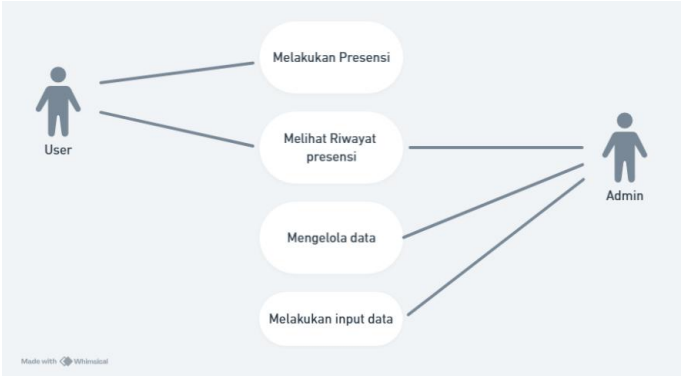
Arsitektur sistem terdiri dari frontend berbasis web yang memungkinkan karyawan untuk mengakses layanan presensi, backend menggunakan AWS Lambda yang bertindak sebagai pemroses utama untuk menangani permintaan presensi, serta penyimpanan data di Amazon S3 untuk menyimpan gambar wajah dan DynamoDB untuk mencatat data kehadiran. AWS API Gateway digunakan sebagai perantara komunikasi antara frontend dan backend, sedangkan layanan AWS Rekognition bertugas mengenali dan mencocokkan wajah pengguna dengan data yang tersimpan di sistem.

3.2. *User Design*

Pada tahap ini menghasilkan desain sistem berupa *Use Case Diagram, activity diagram.*

a) *Use Case Diagram*

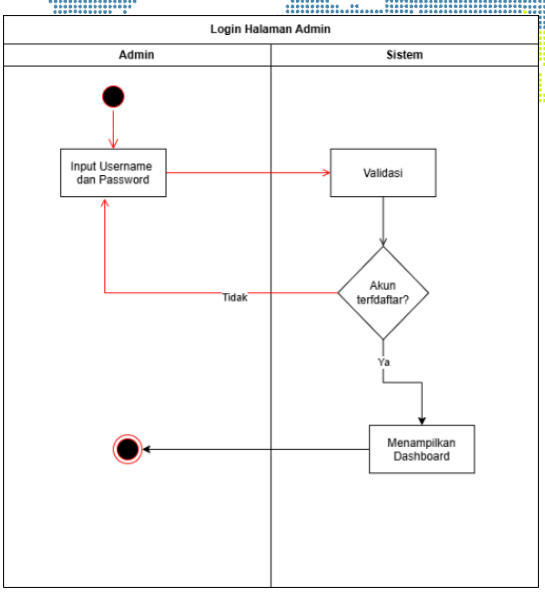
User dapat melakukan presensi dan melihat riwayatnya, sedangkan Admin memiliki akses lebih luas, termasuk mengelola data, memasukkan data, dan melihat laporan presensi. Diagram ini menunjukkan bahwa User hanya berinteraksi dengan fitur dasar, sementara Admin bertanggung jawab atas pengelolaan dan pemantauan data presensi. Dapat dilihat pada Gambar 2.



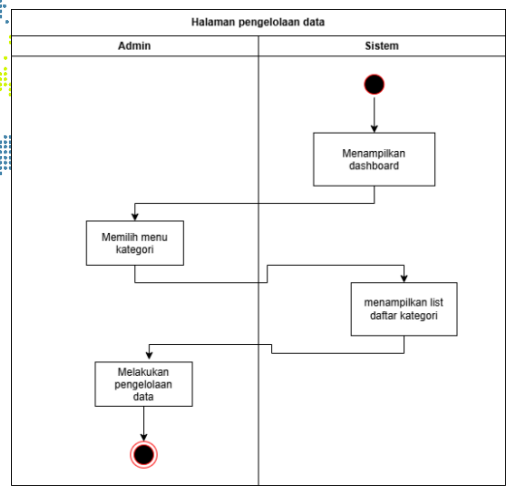
Gambar 2. *Use Case Diagram*

b) *Activity diagram*

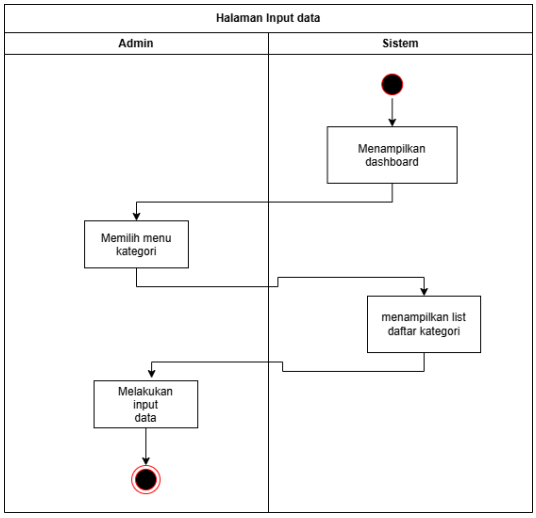
Activity Diagram menunjukkan aliran aktivitas dalam perancangan sistem. Aliran aktivitas dijelaskan mulai dari bagaimana sistem dimulai sampai proses akhir dari aktivitas sistem berakhir.



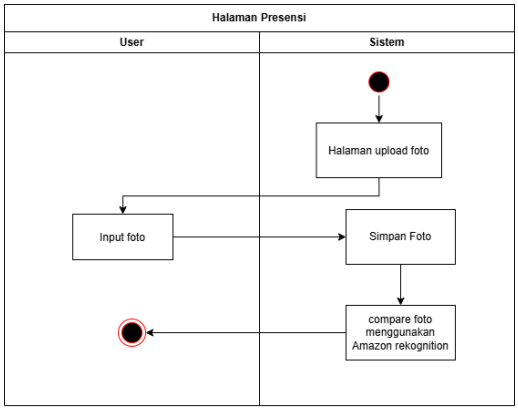
Gambar 3. Login halaman admin



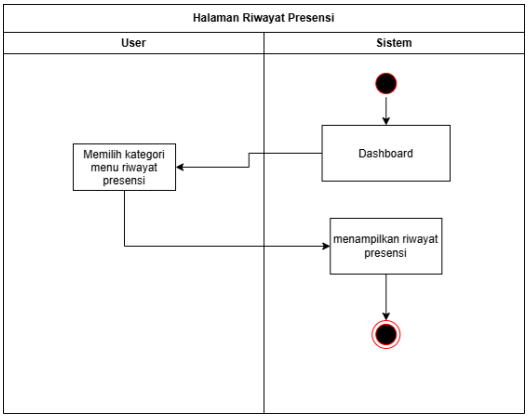
Gambar 4. Halaman Kelola data



Gambar 5. Halaman input data



Gambar 6. Halaman Presensi

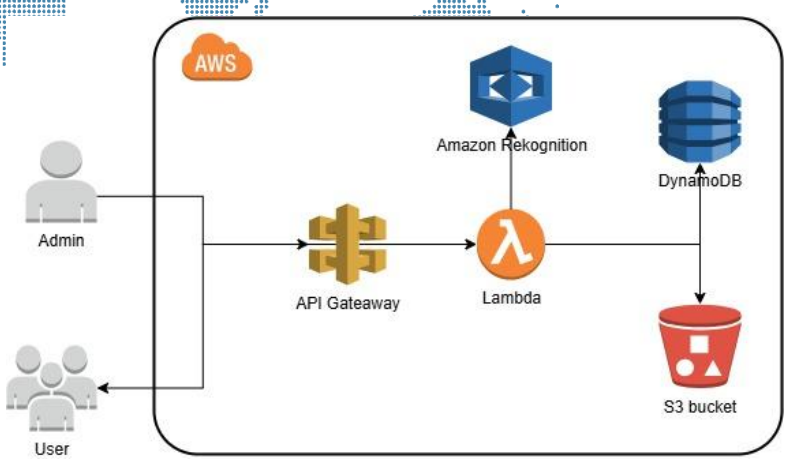


Gambar 7. Halaman Riwayat Presensi

3.3. *Construction*

Contruction merupakan tahap pada RAD dimana pengembang bekerja secara langsung dengan user, membuat rancangan akhir, membangun dan menguji prototype. Hasil

keluaran tahapan ini adalah dokumentasi dan instruksi yang penting untuk mengoperasikan aplikasi baru dan prosedur yang diperlukan agar sistem dapat beroperasi[7].



Gambar 8. Arsitektur menggunakan AWS

1. User mengakses aplikasi web dan mengunggah gambar wajah untuk presensi.
2. API Gateway menerima permintaan dan meneruskannya ke AWS Lambda.
3. AWS Lambda mengambil gambar tersebut dan mengunggahnya ke Amazon S3.
4. AWS Rekognition menganalisis gambar dan membandingkannya dengan data yang sudah tersimpan.
5. Jika wajah cocok, informasi kehadiran dicatat di DynamoDB.
6. Hasil verifikasi dikembalikan ke user melalui API Gateway.

3.4. *Cutover*

Tahap ini adalah tahap pengujian keseluruhan sistem yang dibangun. Semua komponen perlu diuji secara menyeluruh. Pada penelitian ini, pengujian dilakukan dengan menggunakan *black box testing*, dimana pengujian ini lebih menguji ke Tampilan Luar (*interface*) dari suatu aplikasi agar mudah digunakan oleh Customer. *Black box testing* bekerja dengan mengabaikan struktur control sehingga perhatianya hanya terfokus pada informasi domain[7].

4. Kesimpulan

Penerapan metode *Rapid Application Development* dalam pengembangan sistem presensi berbasis pengenalan wajah menggunakan AWS Rekognition memberikan hasil yang efektif dalam meningkatkan efisiensi pencatatan kehadiran. Sistem ini memungkinkan verifikasi wajah secara otomatis, mengurangi risiko kecurangan, serta mempermudah proses absensi dalam lingkungan akademik maupun bisnis. Implementasi layanan AWS, termasuk Amazon Rekognition, S3, dan DynamoDB, memberikan keandalan dan skalabilitas tinggi dalam operasional sistem.

Daftar Pustaka

[1] N. W. Rhomadhona, M. A. Muhammad, P. B. Wintoro, And Y. Mulyani, "Penerapan Metode *Rapid Application Development* Untuk Sistem Informasi Event Berbasis Web Pada Universitas Lampung," *Jurnal Informatika Dan Teknik Elektro Terapan*, Vol. 13, No. 1, Jan. 2025, Doi: 10.23960/Jitet.V13i1.5597.

[2] "Optimalisasi Pencarian Dan Pengelompokan Foto Berbasis Ai Pengenalan Wajah Dengan Aws Recognition".

[3] D. Hariyanto *Et Al.*, "Implementasi Metode," *Jurnal Al-Ilmi*, Vol. 13, No. 1, Pp. 110–117, 2021.

[4] W. N. Cholifah, Y. Yulianingsih, And S. M. Sagita, "Pengujian Black Box Testing Pada Aplikasi Action & Strategy Berbasis Android Dengan Teknologi Phonegap," *String (Satuan Tulisan Riset Dan Inovasi Teknologi)*, Vol. 3, No. 2, P. 206, 2018, Doi: 10.30998/String.V3i2.3048.

[5] H. Pratiwi, N. Fitriani, P. Studi, T. Informatika, Q. Plan, And M. Q. Design, "Membangun Aplikasi Monitoring Kehadiran Pegawai Berbasis Android Pada Disnakertrans".

[6] S. Lailiyah, "Penerapan Algoritma Tabu Search Pada Sistem Penjadwalan Ujian Tugas Akhir Mahasiswa".

[7] Nurman Hidayat And Kusuma Hati, "Penerapan Metode *Rapid Application Development* (Rad) Dalam Rancang Bangun Sistem Informasi Rapor Online (Siraline)," *Jurnal Sistem Informasi*, Vol. 10, No. 1, Pp. 8–17, 2021, Doi: 10.51998/Jsi.V10i1.352.