

Development Of A Mobile-Based Digital Student Id Card Application Using The Flutter Framework

Wahit Fitriyanto¹, Muhammad Abdul Aziz², Yusuf Eko Rohmad³, Taman Ginting⁴

Universitas Boyolali

wahitfitriyanto@gmail.com¹, dotacome@gmail.com², fita.ver@gmail.com³, ginting79@gmail.com⁴

Abstract

The management of physical Student ID Cards at Boyolali University has faced printing inefficiencies and the risk of forgery due to the rapidly growing number of students over the past five years. This research aims to develop a mobile-based Digital KTM application using the Flutter framework and Appwrite backend to provide a secure and efficient identity solution. The software development method applied is the Waterfall model, which includes the stages of requirements analysis, UML diagram, ERD, code construction, and deployment. System security is enhanced through the integration of local fingerprint biometric authentication based on the BiometricPrompt API connected to a Trusted Execution Environment, as well as the implementation of a dynamic QR code based on SHA-256 cryptography with a five-minute expiration limit to prevent visual duplication. Functional testing results using Black Box Testing on 13 test case scenarios showed a high success rate. Usability evaluation via a Likert scale questionnaire administered to 30 respondents yielded an average usability score of 83.70%, indicating that this application is highly suitable for full implementation. This research successfully eliminated the weaknesses of physical cards, reduced issuance time from weeks to minutes, and supports the realization of the Smart Campus vision at Boyolali University.

Keywords: Digital KTM; Flutter Framework; Biometric Authentication; Smart Campus; Waterfall Method

1. Introduction

Higher education is currently experiencing major changes. Campuses can no longer just be places to learn theory, but must become digital ecosystems that serve students quickly and efficiently. This is driven by the demands of technological modernization [1]. In addition, in a regional context, the Boyolali Regency Government, which has proclaimed itself a Smart City pioneer in Central Java for the past decade, is actively building digital infrastructure [2]. Digitalization in Boyolali has now become a mainstream strategy in regional development, where the regency government aggressively transforms public service infrastructure towards an integrated Smart City ecosystem.

This massive wave of technological adoption in the campus's external environment creates a stark gap in service standards if educational institutions like Boyolali University do not immediately align themselves. When the people of Boyolali are accustomed to a responsive digital ecosystem, student expectations of academic services also rise, demanding the transformation of conventional identity systems into digital equivalents of the city's progress [3].

However, the reality on the ground often differs from this digital vision. Boyolali University (UBY) still relies heavily on physical PVC Student Identity Cards (KTM). The use of these physical cards causes classic problems such as expensive printing costs, slow distribution, fragility, and a high risk of loss which burdens campus administration [4]. From a user perspective, all students possess smart devices (smartphones) that they carry daily. This high ownership of smart devices represents a massive infrastructure asset that has not been optimally utilized by the campus administration.

The urgency for digital transformation becomes even more pressing when considering the significant growth in student numbers. Data indicates that student enrollment jumped from 509 in 2020 to 1,564 in 2025, an increase of over 300% in five years. If the card issuance process remains physical, the administrative workload will become unmanageable [5].

Identity management at Boyolali University currently faces four main issues: weeks-long issuance times, restricted campus access for students without cards, high replacement costs for lost cards, and the inability to update data in real-time, making verification difficult. Moreover, lost physical cards are highly susceptible to misuse by unauthorized parties [6].

To overcome these challenges, this study proposes the development of a mobile-based Digital KTM application. The technology choice is crucial, and this study selected Google's Flutter framework. Flutter was chosen due to its superior performance, UI efficiency, and

ability to compile directly to native machine code [7]. Because the transition to digital inherently demands high security, ordinary passwords are no longer deemed sufficient. Thus, this application integrates biometric fingerprint security [8].

To prevent spoofing attacks, the system design uses the Trusted Execution Environment (TEE) native to Android, ensuring biometric data remains strictly isolated within the hardware and is not loosely stored by the application [9]. This research utilizes the Waterfall development methodology because the requirements for the KTM system are definitive, stable, and require strict data validation standards [10]. The success of this digital transformation is expected to shift the paradigm from ownership-based security to biometric-based security and bolster Boyolali University's image as an adaptive Smart Campus ready for national competitiveness by 2028.

2. Literature Review

A comprehensive and critical exploration of current scientific literature relevant to mobile application development, digital identity systems, Flutter framework implementation, and biometric security protocols maps the position of this research. Research by Pualinggi et al. [11] designed an E-KTM system linked directly to the Higher Education Database (PDDikti) via QR Codes. Their findings confirm that the primary value of a Digital KTM lies in backend data interoperability. Ramadhan et al. [12] explored functional value-add through parking ticketing systems using Android-based KTM integrated via RESTful APIs. Meanwhile, Fahru Rozi et al. [13] highlighted the challenge of device fragmentation, demonstrating that Flutter's single codebase approach drastically reduces development complexity while maintaining responsive performance.

Jannah et al. [14] concluded via a systematic literature review that biometric authentication is statistically and practically superior to traditional methods like passwords. However, they warned about biometric data privacy. In response, Saputri [15] recommended utilizing standard OS-level biometric APIs to balance security and usability. Amirullah et al. [16] combined AES-256 encryption with fingerprint authentication, showing that biometrics can serve as a trigger to unlock encrypted data in local memory. Table 1 presents the State of the Art comparison matrix, highlighting the research gap filled by this study.

Table 1: State of the Art Literature Comparison

No	Researcher & Year	Research Topic	Main Focus	Limitations / Difference with this Research
1	Rozi et al. (2025)	Mobile Academic Management System	Single codebase efficiency	Focuses on academic data; lacks specific biometric security for identity.
2	Pualinggi et al. (2021)	E-KTM PDDIKTI Integration	National data integration & QR	Uses Android Native; device access security is not the primary focus.
3	Jannah et al. (2024)	Digital Wallet Biometrics	Biometrics vs Password	Theoretical literature review, not specific product development.
4	Ferdiansyah & Sofian (2023)	Mobile Presence via QR & LBS	Location-based presence	Focuses on presence without biometrics as an authentication layer.
5	Anugraha et al. (2025)	Biometric Blockchain E-Voting	Vote integrity security	Blockchain implementation is too complex for KTM needs.
6	Saputra (2023)	Mobile Banking Biometrics	Security vs Convenience	This research adapts banking security principles to academia.
7	Kasliono et al. (2024)	IoT Plant Monitoring	Waterfall hardware sync	Serves as methodological reference for API Biometric integration in Flutter.
8	Sri Rahayu et al. (2024)	Archive Info System	SOP transformation via Waterfall	Ensures KTM issuance flow adheres to University regulations.
9	Ramadhan et al. (2023)	KTM Parking Ticketing	Parking gate integration	Focuses on parking, while this study focuses on application access security.
10	Lendo et al. (2023)	Mobile Digital Library	Library UI/UX	Focuses on library content, whereas this study focuses on personal identity.
11	Zakaria (2023)	Employee Presence via GPS	Flutter GPS accuracy	Focuses on GPS, whereas this study focuses on Fingerprint sensors.
12	Riswanto et al. (2025)	Biometric E-Voting	Non-repudiation	Uses overly complex Blockchain; this study uses local TEE biometrics.
13	Amirullah et al. (2024)	AES & Biometric File Security	Local data encryption	Focuses on generic files; this study encrypts student profile data.
14	Faqih et al. (2025)	Tourist Ticket Booking	Transactions & Dynamic QR	Focuses on tourism. This study adapts the dynamic QR mechanism for ID.

3. Methodology

3.1 Research Location and Time

This research was conducted at the Informatics Engineering Study Program, Faculty of Communication and Informatics Engineering, Universitas Boyolali. The research duration spanned from December 2025 to June 2026. The required resources included administrative documents, physical KTM samples, the university logo, student photo samples, the Flutter framework, and Appwrite BaaS components.

3.2 Waterfall Method

The operational framework adopted is the Waterfall software development model. This highly systematic and sequential approach ensures that each development stage is fully completed before progressing to the next. The phases are Requirements Analysis, Planning, Modeling (using UML and ERD), Construction (coding using Dart), and Deployment [10].

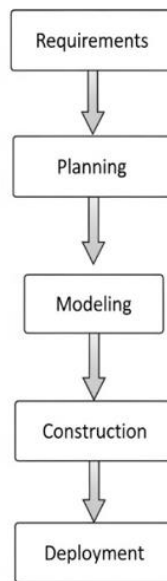


Fig. 1: Waterfall Methodology Model

3.3 System Analysis and Design

The analysis phase identified the shortcomings of the current manual system: high printing costs, slow issuance, lack of real-time validation, and physical vulnerabilities. Consequently, the new system requirements were defined into functional and non-functional categories. Non-functional requirements emphasize security (SSL/TLS, TEE Keystore), high performance, and usability. The system architecture utilizes the Unified Modeling Language (UML) to visualize workflows. The Use Case diagram establishes interaction boundaries for Students and Administrators.

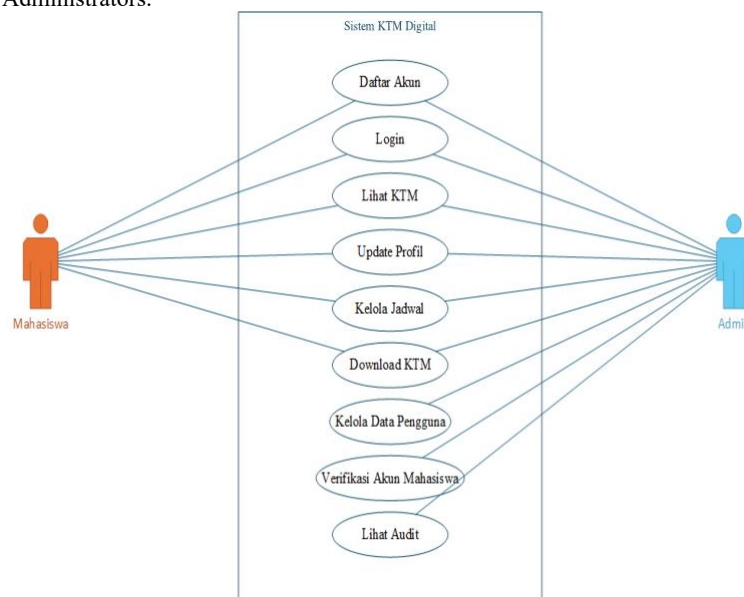


Fig. 2: System Use Case Diagram

3.4 Database Design (Appwrite TablesDB)

The database architecture uses Appwrite's TablesDB, replacing traditional NoSQL document structures with robust relational schemas. This structural integrity is crucial for academic data. The database comprises five primary tables: Account, KTM, Schedule, Audit, and Account History. Table 2 displays the Account Table structure which consolidates all user profiles, enforcing unique constraints to maintain data integrity.

Table 2: Account Table Structure

No	Field Name	Data Type	Description
1	NIM	Varchar(50)	Student ID Number (Primary Key)
2	Email	Varchar(255)	User email address
3	Nama	Varchar(255)	Full name
4	Program_studi	Varchar(100)	Study program
5	Peran	Varchar(50)	Access role level
6	Status_akun	Varchar(50)	Operational account status
7	Diperbarui_pada	Datetime	Last modification timestamp

The Schedule table (Table 3) maps dynamic academic activities, utilizing composite unique indices to prevent class scheduling conflicts and duplicate enrollments.

Table 3: Schedule Table Structure

No	Field Name	Data Type	Description
1	Id_pengguna	Varchar(255)	User ID linked to the schedule
2	Kode_mata_kuliah	Varchar(50)	Course Code
3	Hari	Integer(1)	Day of the week (1-7)
4	Jam_mulai	Varchar(10)	Class start time
5	Lokasi	Varchar(100)	Classroom/Lab location
6	Dosen	Varchar(255)	Lecturer name

Table 4 defines the KTM entity structure in the database.

Table 4: KTM Entity Structure

No	Field Name	Data Type	Description
1	Nim	Varchar(50)	Student ID Number (Foreign Key)
2	Nama	Varchar(255)	Student Name
3	Program_studi	Varchar(100)	Study Program
4	Fakultas	Varchar(100)	Faculty
5	Id_foto	Varchar(255)	Profile Picture ID in Storage
6	Diperbarui_pada	Datetime	Last updated timestamp

The Audit Trail and History tables ensure transparency.

Table 5: Audit Trail Structure

No	Field Name	Data Type	Description
1	Waktu	Datetime	Timestamp of action
2	Aksi	Varchar(255)	Action type (Login, Update, etc.)
3	Uid_pelaku	Varchar(255)	Initiator User ID
4	Uid_target	Varchar(255)	Target User ID
5	Pesan	Varchar(255)	System response message

6

Metadata

Varchar(500)

Browser and IP metadata

3.5. System Security Implementation

The application relies heavily on two primary security infrastructures: Local Biometric Authentication via the Android BiometricPrompt API, and Cryptographic QR Code Generation using the SHA-256 algorithm. The system delegates all fingerprint processing to the hardware Trusted Execution Environment (TEE). If the biometric signature matches, the Android Keystore decrypts the local application state.

4. Results And Discussion

4.1. Interface Implementation

The mobile-based Digital Student ID Card application was successfully built. The user interface prioritizes usability and modern aesthetics based on Material Design guidelines. The application separates user authorization boundaries to direct users to appropriate dashboard.



Fig. 3: Login Authentication Page



Fig. 4: Student Dashboard Interface

Figure 5 showcases the core feature: the Digital KTM. This virtual card dynamically displays academic data alongside an automatically

generated cryptographic QR Code. This QR code acts as an anti-spoofing mechanism, refreshing periodically to invalidate screenshots.

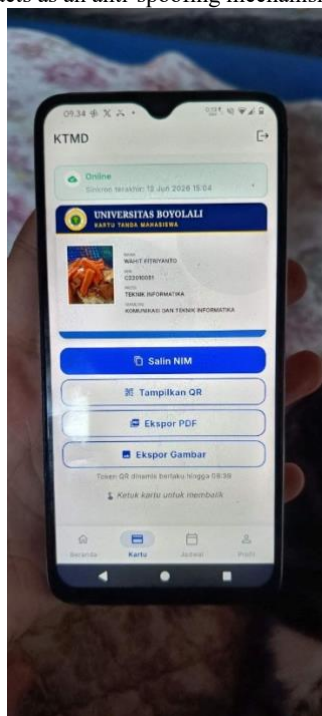


Fig. 5: Digital Student Identity Card (KTM)

4.2. Functional Testing (Black Box Testing)

Functional testing was conducted via Black Box Testing to validate that all software components behaved as per specifications without examining internal code logic. The test involved 30 active respondents evaluating 13 scenarios covering registration, biometric security, card rendering, and real-time updates.

Table 6: Functional Testing Result

No	Test Case Scenario	Expected Result	Status
1	Input complete biodata and upload photo	System processes registration and saves to Appwrite	Pass
2	Omit required fields during registration	System rejects registration and shows warnings	Pass
3	Input valid credentials	System validates via Appwrite and redirects to Dashboard	Pass
4	Initiate fingerprint login with valid finger	App detects sensor and verifies biometrics successfully	Pass
5	Attempt fingerprint login with unregistered finger	System triggers rejection and keeps app locked	Pass
6	Monitor Digital KTM screen over 5 minutes	App automatically refreshes dynamic SHA-256 QR code	Pass
7	Execute "Download Card" utility	App converts view to PDF/Image and saves locally	Pass
8	Update personal contact info in profile	System instantly synchronizes new data to Appwrite DB	Pass

The results confirmed high system reliability with zero functional failures or backend access errors. Table 7 demonstrates the usability scores derived from the Likert scale questionnaire, validating user satisfaction across multiple interface aspects.

Table 7: Usability Evaluation Scores

No	Evaluation Indicator	Average Score (%)	Feasibility Category
1	Ease of learning the application	82.00	Highly Feasible
2	Efficiency of the digital KTM feature	82.67	Highly Feasible
3	Clarity of the interface design	82.67	Highly Feasible
4	Speed of biometric authentication	87.33	Highly Feasible
5	Reliability of the dynamic QR code	80.67	Highly Feasible
6	Completeness of academic schedule	84.00	Feasible

7	Clarity of error messages	84.67	Highly Feasible
8	Ease of profile data modification	84.67	Highly Feasible
9	Overall application satisfaction	84.67	Highly Feasible

4.3. Usability Evaluation

Beyond functional testing, system usability was quantified using a Likert-scale questionnaire. The cumulative average usability score reached 83.70%, categorizing the application as "Highly Feasible" for full-scale implementation at the university.

4.4. Discussion

The choice of the Flutter framework proved highly strategic in bridging the OS fragmentation (Android and iOS) among the student body. Flutter's custom rendering engine delivered smooth UI transitions at 60 FPS regardless of device specifications. Furthermore, utilizing Appwrite as a Backend-as-a-Service bypassed the massive overhead of managing traditional servers while providing rigorous TablesDB relational validations and SSL/TLS encrypted data streams.

The implementation of Trusted Execution Environment (TEE) hardware isolation fundamentally shifts the campus identity paradigm. Biometric templates are never exposed to the application layer. When the user initiates a login, the BiometricPrompt API queries the TEE. Upon success, the Android Keystore unlocks the AES-256 symmetric key required to decrypt the cached student profile. If the device is stolen and a new fingerprint is registered, the Keystore automatically revokes the key, permanently securing the data.

Visual forgery via screenshots is effectively mitigated by the dynamic QR code mechanism. The app generates a security token using the SHA-256 hashing algorithm, combining the NIM, a timestamp, and a random server salt. Campus verification systems enforce a strict 5-minute expiration window on these tokens, instantly rendering shared screenshots invalid.

5. Conclusions

The mobile-based Digital Student ID Card application at Universitas Boyolali was successfully developed utilizing the Flutter framework and Appwrite backend. The system incorporates military-grade access security through local biometric fingerprint authentication isolated within the device's Trusted Execution Environment (TEE). Usability evaluations yielded a strong 83.70% satisfaction score among students, proving high end-user acceptance. The integration of SHA-256 cryptography to generate 5-minute dynamic QR codes successfully eradicates the visual forgery vulnerabilities associated with traditional PVC physical cards.

References

- [1] Nurhayati & Mulyanti, "Modernization of Technology in Higher Education Ecosystems," 2025.
- [2] D. I. Lukmana and A. R. Pratama, "Analisis Faktor Penentu Kesuksesan Aplikasi Smartcity Di Kabupaten Boyolali," *Incomtech : Jurnal Telekomunikasi Dan Komputer*, vol. 13, no. 2, p. 155, 2023.
- [3] T. Afif, N. Hidayat, S. Nasiroh, and H. A. Prakosa, "Sistem Manajemen Kartu Tanda Mahasiswa Digital Berbasis Mobile Website Universitas Perwira Purbalingga," *Perwira Journal of Science & Engineering*, vol. 4, no. 2, pp. 98–104, 2024.
- [4] Kustiawati et al., "Flutter Framework Performance in Mobile Application Development," 2025.
- [5] M. F. Rozi, H. Siregar, and Y. A. Hambali, "Rancang Bangun Sistem Manajemen Akademik Mahasiswa Berbasis Mobile Multiplatform Menggunakan Flutter," *Jurnal Komputer Teknologi Informasi Sistem Informasi (Juktisi)*, vol. 4, no. 2, pp. 459–468, 2025.
- [6] P. Pualinggi, M. Sarjan, and A. Kahpi, "Integrasi E-Ktm (Kartu Tanda Mahasiswa Elektronik) Dengan Pddikti Menggunakan Qr-Code Berbasis Android," *Journal Peqguruang*, vol. 3, no. 1, pp. 396–400, 2021.
- [7] B. R. Ramadhan, A. Faisol, and D. Rudhistiar, "Rancang Bangun Sistem Ticketing Parkir Menggunakan Ktm Berbasis Android Menggunakan Restful Api," *Jurnal Mahasiswa Teknik Informatika*, vol. 7, no. 5, 2023.
- [8] M. Jannah, M. F. Hidayat, M. Agustiyani, P. W. Buana, and F. Purwani, "Implementasi Autentikasi Biometrik Untuk Meningkatkan Keamanan Dan Privasi Penggunaan Dompot Digital," *Journal Of Sciencetech Research And Development*, vol. 6, no. 2, pp. 531–539, 2024.
- [9] H. H. Amirullah, A. Eviyanti, and S. Sumarno, "Aplikasi Keamanan Berkas Dengan Enkripsi Aes Dan Biometrik Sidik Jari Berbasis Android," *Smatika Jurnal*, vol. 14, no. 01, pp. 23–32, 2024.
- [10] K. Kasliono, I. Ruslianto, and Y. Erniajan, "Penerapan Model Waterfall Dalam Pengembangan Perangkat Lunak Pemantauan Tanaman Anggur Berbasis Mobile Menggunakan Iot," *Journal Of Computer System And Informatics (Josyc)*, vol. 5, no. 3, pp. 526–534, 2024.
- [11] M. Riswanto and N. Anugraha, "Sistem E-Voting Berbasis Blockchain Dengan Autentikasi Biometrik Sidik Jari Dan Protokol Zero-Knowledge Proof Untuk Pengaman Privasi," *Jurnal Minfo Polgan*, vol. 14, no. 2, pp. 2756–2767, 2025.
- [12] F. Reza Ferdiansyah and R. Sofian, "Implementasi Teknologi Qr Code Dan Location Based Service Pada Presensi Mobile Flutter," *Jurnal Ilmiah Intech*, vol. 5, no. 2, pp. 85–97, 2023.
- [13] A. Faqih et al., "Rancang Bangun Aplikasi Mobile Untuk Pemesanan Tiket Wisata Berbasis Flutter," *Jurnal Informatika Dan Teknik Elektro Terapan*, vol. 13, no. 2, 2025.
- [14] M. Al Haqqi, I. Arwani, and S. A. Wicaksono, "Pengembangan Aplikasi Monitoring Program Kerja Berbasis Mobile Menggunakan Framework Flutter," *J-Ptiik*, vol. 7, no. 1, 2023.
- [15] F. Hudaya, "Rancang Bangun Sistem Holter Monitor Dengan Konektivitas Bluetooth Low Energy Dan Aplikasi Berbasis Appwrite," *Jakarta: Jurusan Teknik Elektromedik*, 2024.