Pengembangan Aplikasi Pesan Instan Terenkripsi Menggunakan Algoritma Kriptografi AES (Advanced Encryption Standard)

Lutfa Ibtihaji Ilham[a], Adhika Pramita Widyassari[b*]
dikasari9@gmail.com

[ab] Sekolah Tinggi Teknologi Ronggolawe Cepu, Jurusan Teknik Elektro

Intisari

Komunikasi melalui aplikasi pesan instan kini menjadi aspek vital, pengembang aplikasi saling bersaing untuk menawarkan aplikasi pesan instan dengan berbagai kemudahan bagi pengguna, namun mengesampingkan aspek keselamatan pengguna. Penggunaan aplikasi pesan instan secara masif dan ketidaktahuan pengguna terhadap data menjadikan ini sasaran empuk bagi orang-orang yang tidak berhak mengambil atau mencuri data dari pengguna aplikasi pesan instan dengan berbagai teknik penyadapan. Penelitian ini bertujuan untuk mengembangkan aplikasi pesan instan berbasis Android yang menawarkan keamanan data pesan, pengiriman pesan dengan penundaan yang minimal dan mendukung berbagai perangkat Android. Sistem aplikasi perpesanan instan dibuat menggunakan Android Studio, bahasa pemrograman Kotlin, metode enkripsi data AES (Advanced Encryption Standard), model siklus hidup pengembangan sistem prototipe, dan terintegrasi dengan berbagai layanan cloud Firebase termasuk: Firebase Authentication, Firebase Cloud Storage, dan Basis data waktu nyata Firebase. Aplikasi pesan instan yang dikembangkan dapat berjalan di perangkat Android versi 4 hingga versi terbaru dan mampu mendukung aktivitas pengiriman pesan antar pengguna dengan cepat dan aman. Melalui hasil pengujian black box, aplikasi pesan instan yang dikembangkan memperoleh tingkat kelayakan hingga 96% yang artinya aplikasi pesan instan yang dikembangkan sudah layak untuk digunakan secara umum.

Kata kunci: Android Studio, Firebase, AES (Advanced Encryption Standard), Aplikasi Pesan Instan, Keamanan Data.

Abstract

Communication through instant messaging applications is now a vital aspect, application developers compete with each other to offer instant messaging applications with a variety of convenience for users, but override the user safety aspects. The massive use of instant messaging applications and the user's ignorance of the data make this an easy target for people who are not entitled to retrieve or steal data from instant messaging application users with various tapping techniques. This research aims to develop an Android-based instant messaging application that offers message data security, sending messages with minimal delay and supports a variety of Android devices. The instant messaging application system was built using Android Studio, Kotlin programming language, AES (Advanced Encryption Standard) data encryption method, Prototyping system development life cycle model and integrated with a variety of Firebase cloud services including: Firebase Authentication, Firebase Cloud Storage, and Firebase Real- time database. Instant messaging applications that are developed can run on Android devices version 4 to the latest version and are able to support messaging activities between users quickly and safely. Through the results of black box testing, the instant messaging application that was developed gained a level of eligibility up to 96%, which means the instant messaging application that was developed was suitable for general use.

Keywords**:** *Android Studio, Firebase, AES (Advanced Encryption Standard), Instant Messaging Application, Data Security.*

1. Pendahuluan

Komunikasi merupakan aspek yang vital dalam kehidupan keseharian di dunia, sejak pada zaman dahulu orang melakukan komunikasi jarak jauh melalui pesan ke sesama dengan berbagai macam media, mulai dari melalui surat, SMS, hingga e-mail yang sekarang sudah ditinggalkan dan cenderung digunakan untuk kepentingan bisnis karena kebiasaan pengguna yang menginginkan pesan dibalas secara cepat. Oleh karena itu, mulai bermunculan berbagai aplikasi pesan berbasis desktop yang menawarkan kecepatan pengiriman dengan jeda yang minimal, jenis aplikasi pesan tersebut berhasil mengikat banyak pengguna dan memiliki peranan yang penting untuk menunjang aktivitas perpesanan keseharian. Namun setelah berjalannya waktu dan banyaknya penggunaan gawai bersistem operasi Android, mulailah bermunculan aplikasi pesan instan berbasis Android yang sekarang ini menjadi bagian dari kehidupan setiap orang mulai dari remaja hingga dewasa.

Beberapa penelitian tentang pengembangan aplikasi pesan instan berbasis Android yang dilakukan oleh (Sandy, Akbar, & Hariadi, 2017) telah mampu menghasilkan aplikasi pesan instan berbasis Android yang dibangun dengan bahasa pemrograman Java dan layanan cloud Firebase, aplikasi pesan instan yang dihasilkam memiliki masukan berupa kanvas yang dapat dibagikan antar pengguna untuk melakukan percakapan secara online, namun belum memiliki proses enkripsi isi pesan. Penelitan kedua oleh (Chatterjee, Chakraborty, Decosta, & Nath, 2018) telah mampu menghasilkan aplikasi pesan instan berbasis Android yang dibangun dengan bahasa pemrograman Java dan layanan cloud Firebase, aplikasi yang dihasilkan mampu mengirimkan pesan dengan jeda yang minimal, namun belum memiliki proses enkripsi isi

pesan. Penelitian ketiga oleh S. S. R. Emmadi & S. Potluri (Emmadi & Potluri, 2019) telah mampu menghasilkan aplikasi pesan instan berbasis Android yang dibangun dengan bahasa pemrograman Java dan layanan cloud Firebase, aplikasi yang dihasilkan memiliki fitur antarmuka dengan berbagai kemudahan untuk pengguna, namun belum memiliki proses enkripsi isi pesan. Penelitian keempat oleh A. R. Nugroho & W. Pramusinto (Nugroho & Pramusinto, 2018) telah mampu menghasilkan aplikasi pesan instan berbasis Android yang dibangun dengan bahasa pemrograman Java dan layanan cloud Firebase, aplikasi yang dihasilkan mampu mengirimkan pesan secara aman, namun belum memiliki akses pintas cepat pesan untuk mengakses pesan terbaru dan belum memiliki autentikasi berupa email dan kata sandi.

Berdasarkan literatur *review* yang ada, maka muncul ide untuk merancang, membangun dan mengimplementasikan aplikasi pesan instan berbasis Android yang menerapkan metode keamanan data pada pesan menggunakan algoritma kriptografi AES (Advanced Encryption Standard) dan memanfaatkan layanan *cloud* Firebase untuk menunjang dukungan penyimpanan berkas, autentikasi pengguna dan basis data *real-time*.

2. Kerangka Teori

2.1. Android Studio

Android Studio adalah IDE (Integrated Development Environment) resmi untuk pengembangan aplikasi Android dan bersifat open source atau bebas. Pelucuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada saat acara konferensi Google I/O untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android.

Android Studio sendiri dikembangkan berdasarkan Intellij IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools). Android Studi memiliki fitur sebagai berikut:

a. Projek berbasis pada Gradle Build.
b. Refactory dan pembenahan bug yang cepat.
c. Tools baru yang bernama Lint yang diklaim dapat memonitor kecepatan, kegunaan, serta kompatibelitas aplikasi dengan cepat.
d. Mendukung Proguard dan App-signing untuk keamanan.
e. Memiliki GUI aplikasi android lebih mudah.
f. Didukung oleh Google Cloud Platform untuk setiap aplikasi yang dikembangkan.(Suryana, 2018)

2.2. Kotlin

Kotlin merupakan bahasa pemrograman modern yang terinspirasi dari bahasa pemrograman Swift, Scala, Groovy, C# dan lain-lain yang kompatibel dengan Android dan mampu memberikan solusi pada masalah bahasa pemrograman java seperti *null pointer exceptions*.

Kotlin didesain oleh *JetBrains Professionals,* berdasarkan analisis dari pengalaman-pengalaman para pengembang tentang bagaimana membuat kode java menjadi bersih dan efektif.

Dukungan Kotlin untuk Platform Android dirilis pertama kali diluncurkan pada tahun 2017 oleh Google dan memiliki banyak komunitas aktif yang membuat bahasa pemrograman ini berkembang sangat pesat (Moskala & Wojda, 2017).

2.3. Firebase

Firebase adalah alat Backend-as-a-Service (BaaS) yang sangat populer yang diluncurkan pada April 2012. Karena kemampuan dan teknologinya yang canggih seperti *real-time database*, alat ini menarik minat Google dan diakuisisi oleh Google pada tahun 2014. Firebase telah berubah menjadi alat yang ampuh, berisi dukungan untuk sebagian besar kasus penggunaan pengembangan perangkat lunak pada dunia nyata. Firebase membagi toolchain menjadi tiga kategori berbeda yaitu: membangun aplikasi, meningkatkan kualitas aplikasi, dan mengembangkan bisnis aplikasi.

Penggunaan Firebase dapat memudahkan pengembang dalam membangun aplikasi seluler yang hebat, baik itu untuk iOS atau Android. Firebase memiliki banyak hal yang ditawarkan, mulai dari analisis hingga akuisisi pengguna dan pengujian A/B, dan daftar layanannya akan terus dikembangkan dan bertambah seiring berjalannya waktu (Kumar S, 2018).

2.4. Andvanced Encryption Standard (AES)

Advanced Encryption Standard (AES) adalah salah satu algoritma enkripsi blok cipher yang diterbitkan oleh National InstituteStandar and Technology (NIST) pada tahun 2000. Tujuan utama dari algoritma ini adalah untuk menggantikan algoritma DES setelah muncul beberapa aspek yang rentan pada algoritma enkripsi tersebut. NIST mengundang para ahli yang bekerja pada enkripsi dan keamanan data di seluruh dunia untuk memperkenalkan algoritma blok cipher inovatif untuk mengenkripsi dan mendekripsi data dengan struktur yang kuat dan kompleks.

Dari seluruh dunia banyak kelompok mengajukan algoritma mereka. NIST menerima lima algoritma untuk dievaluasi. Setelah melakukan berbagai kriteria dan parameter keamanan, mereka memilih satu dari lima algoritma enkripsi yang diusulkan oleh dua *cryptographers* Belgia Joan Daeman dan Vincent Rijmen. Nama asli algoritma AES adalah algoritma Rijndel.

Algoritma kriptografi AES memiliki beberapa proses perulangan untuk menciptakan teks *cipher*, yaitu:

a. 10 kali perulangan, untuk menciptakan teks *cipher* dengan kekuatan enkripsi sebesar 128bit.
b. 12 kali perulangan, untuk menciptakan teks *cipher* dengan kekuatan enkripsi sebesar 192bit.
c. 14 kali perulangan, untuk menciptakan teks *cipher* dengan kekuatan enkripsi sebesar 256bit.(Daemen & Rijmen, 2013)

2.5. System Development Life Cycle (SDLC)

SDLC atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik). Seperti halnya proses metamorfosis pada kupu-kupu, untuk menjadi kupu-kupu yang indah maka dibutuhkan beberapa tahap untuk dilalui, sama halnya dengan membuat perangkat

lunak, memiliki daur tahapan yang dilalui agar menghasilkan perangkat lunak yang berkualitas.

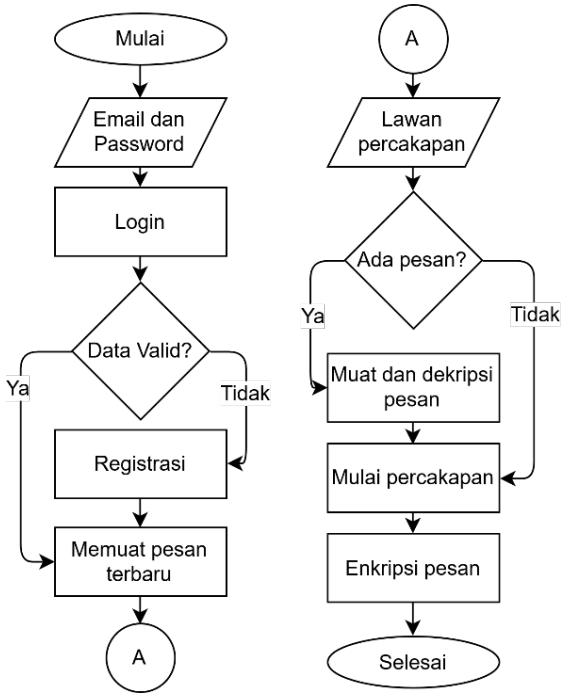
Metode *Prototyping* sangat cocok digunakan untuk kebutuhan pengembangan aplikasi yang dapat digunakan oleh klien atau pengguna secara cepat. Hal positif dari model *Prototyping* ini adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan di setiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap) (Kurniawan, 2013).

3. Metodologi

3.1. Skema Sistem Aplikasi

Untuk membangun aplikasi berdasarkan permasalahan yang diangkat, maka diuraikanlah skema proses keseluruhan aplikasi meliputi:

1. Pengguna melakukan *login* pada aplikasi menggunakan email dan kata sandi, apabila pengguna belum memiliki akun, maka pengguna diharuskan melakukan pendaftaran terlebih dahulu.
2. Pengguna yang berhasil melakukan *login* akan disuguhkan pada antarmuka pesan terbaru yang berisikan pesan-pesan yang diterima oleh pengguna.
3. Pengguna dapat melakukan percakapan dengan pengguna lain melalui tombol "Pesan Baru" kemudian memilih lawan percakapan melalui antarmuka daftar kontak, pengguna juga dapat melakukan percakapan melalui antarmuka pesan terbaru dengan cara memilih salah satu pesan.
4. Sistem akan memproses enkripsi dan dekripsi data pesan menggunakan algoritma AES (*Advanced Encryption Standard*) secara otomatis untuk setiap perubahan yang terjadi pada aplikasi oleh setiap tindakan yang dilakukan oleh pengguna.



Gambar 3.1 Diagram Alir Sistem Aplikasi

3.2. Skema Basis Data

Sistem basis data yang digunakan adalah Firebase Real-time Database yang merupakan sistem basis data NoSQL, oleh karena itu tidak ada aturan khusus terkait kardinalitas atau relasi pada basis data sehingga dalam pengembangan aplikasi lebih terfokus pada algoritma yang digunakan. Berikut merupakan struktur *child* atau tabel basis data yang digunakan:

a. users

Pada *child* ini, memiliki *children* sebagai berikut:

- uid, merupakan ID unik pengguna.
- username, merupakan nama lengkap pengguna.
- profileimageurl merupkan *url* gambar profile pengguna.

Tabel 3.1 *Child* users

Field	Type
userkey	*String (child)*
uid	String
username	String
profileimageurl	String

b. user-messages

Pada *child* ini memiliki *children* sebagai berikut:

- id, merupakan ID unik pesan.
- toId, merupakan ID unik penerima pesan.
- fromId, merupakan ID unik pengirim pesan.
- text, merupakan isi pesan.
- secretkey, merupakan kunci enkripsi dan dekripsi pesan.
- timestamp, merupakan waktu pengiriman pesan.

Tabel 3.2 *Child* user-messages

Field	Type
messageskey	*String (child)*
id	String
toId	String
fromId	String
text	String
secretkey	String
timestamp	Long

c. latest-messages

Child ini merupakan cerminan dari *child* user-messages yang memiliki struktur tabel dan nilai yang sama.

Tabel 3.3 *Child* latest-messages

Field	Type
messageskey	*String (child)*
id	String

toId	String
fromId	String
Text	String
Secretkey	String
Timestamp	Long

4. Hasil dan Pembahasan

4.1. Implementasi Antarmuka Masuk

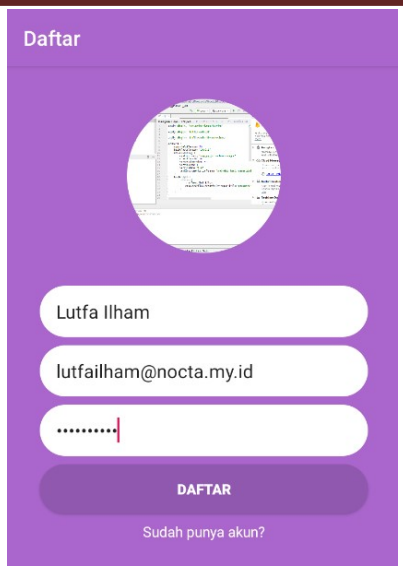
Pada awal aplikasi dibuka, pengguna akan disuguhkan dengan antarmuka untuk masuk. Pengguna yang sudah terdaftar dapat memasukkan email dan kata sandi kemudian melakukan *login,* sistem aplikasi akan memeriksa dan memastikan bahwa setiap masukan sudah benar, apabila pengguna belum terdaftar, maka pengguna dapat melakukan pendaftaran dengan mengetuk tombol "Belum punya akun?".



Gambar 4.1 Antarmuka Masuk

4.2. Implementasi Antarmuka Daftar

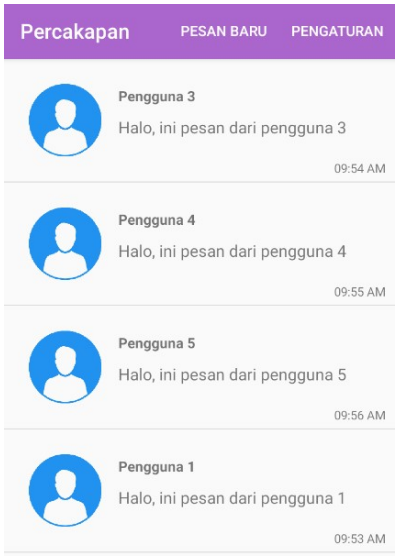
Pada antarmuka daftar, pengguna dapat melakukan pendaftaran ke dalam sistem aplikasi dengan memasukkan masukan berupa: gambar profil pengguna, nama lengkap, email dan kata sandi. Ketika pengguna melakukan pendaftaran, sistem aplikasi akan memeriksa dan memastikan setiap masukan sudah benar, apabila setiap masukan benar dan pengguna melakukan pendaftaran, maka sistem aplikasi akan memberikan informasi yang memberitahukan bahwa pendaftaran berhasil dan mengalihkan ke antarmuka pesan terbaru.



Gambar 4.2 Antarmuka Daftar

4.3. Implementasi Antarmuka Pesan Terbaru

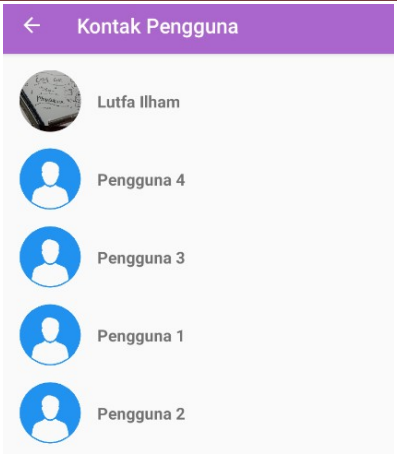
Pada saat pengguna berhasil melakukan *login* ke dalam aplikasi, maka sistem aplikasi akan menampilkan antarmuka pesan terbaru. Pada antarmuka ini, sistem akan memuat sekaligus mendekripsi pesan yang dikirmkan dari pengguna lain ke pengguna saat ini secara *real-time* dan otomatis, antarmuka ini menawarkan kemudahan untuk melakukan percakapan dengan pengguna lain yaitu dengan cara mengetuk salah satu pesan terbaru, maka sistem akan mengalihkan pada antarmuka percakapan dengan pengguna yang terpilih sebelumnya.



Gambar 4.3 Antarmuka Pesan Terbaru

4.4. Implementasi Antarmuka Daftar Kontak Pengguna

Antarmuka daftar kontak pengguna berisi informasi atau data mengenai semua pengguna yang terdaftar dalam sistem aplikasi yang tersimpan pada layanan *cloud* Firebase Real-time Database.



Gambar 4.4 Antarmuka Daftar Kontak Pengguna

4.5. Implementasi Antarmuka Percakapan

Pada antarmuka percakapan, pengguna dapat melakukan aktivitas perpesanan seperti pada umumnya. Sistem aplikasi akan melakukan proses enkripsi dan dekripsi data pesan menggunakan algoritma kriptografi AES (*Advanced Encryption Standard*), kemudian pesan yang terkirim akan disimpan ke dalam basis data.

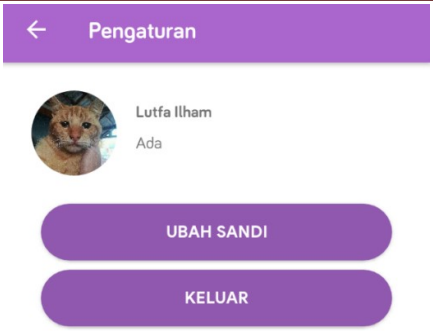


Gambar 4.5 Antarmuka Percakapan

4.6. Implementasi Antarmuka Pengaturan

Pada antarmuka pengaturan menyediakan informasi tentang pengguna login saat ini, antarmuka ini memiliki beberapa tombol sebagai berikut:

a. Keluar, yang berfungsi untuk melakukan logout pengguna.
b. Ubah sandi, yang berfungsi untuk melakukan perubahan pada kata sandi pengguna saat ini.
c. Gambar profil, yang berfungsi untuk mengganti gambar profil pengguna saat ini.



Gambar 4.6 Antarmuka Pengaturan

4.6. Pengujian Sistem

Pengujian sistem dilakukan dengan teknik *black box*, pengujian dilakukan untuk memastikan kelayakan penggunaan dari setiap komponen-komponen pada sistem aplikasi yang dikembangkan, berikut tabel hasil pengujian *black box* berdasarkan rekap pengujian dari 5 penguji:

Tabel 4.1 Hasil Pengujian *black box*

Komponen	Presentase Kelayakan
Masuk	96%
Daftar	94%
Pesan Terbaru	100%
Kontak Pengguna	100%
Percakapan	96%
Pengaturan	92%
Rata-rata	**96%**

5. Kesimpulan dan Saran

Kesimpulan yang dapat ditarik dari hasil dan pembahasan adalah sebagai berikut:

1. Aplikasi dapat mengirim dan menerima pesan antar pengguna secara instan atau *real-time*.
2. Metode dan proses enkripsi data pesan dapat menjaga keamanan data pesan oleh pihak yang tidak berhak atau tidak bertanggung jawab.
3. Aplikasi dapat dijalankan pada berbagai platform Android, mulai dari versi 4.1 hingga terkini.
4. Aplikasi memiliki autentikasi berupa email dan password sehingga memungkinkan penggunaan secara global atau umum.
5. Aplikasi memiliki dashboard pesan terbaru untuk mengakses pesan kepada pengguna lain sehingga akses pesan menjadi lebih mudah dan cepat.
6. Berdasarkan hasil pengujian dengan teknik *black box*, sistem aplikasi yang dikembangkan mampu memperoleh tingkat kelayakan hingga mencapai 96%.

Saran dari peneliti, sistem yang dibuat saat ini belum seutuhnya sempurna, oleh karena itu diharapkan agar sistem yang dibuat ini dapat dikembangkan kembali hingga dapat semakin efektif dalam penggunaannya. Saran-saran yang dapat diberikan berdasarkan hasil dan pembahasan adalah sebagai berikut:

1. Menambahkan fitur pilih dan hapus pesan.
2. Menambahkan proses kompresi gambar agar pengguna tidak perlu melakukan kompresi gambar secara manual.
3. Menambahkan proses caching agar konten-konten aplikasi dapat dimuat lebih cepat pada saat aplikasi dimuat ulang atau restart meski tanpa menggunakan koneksi internet.
4. Tata ulang layout antarmuka agar aplikasi lebih nyaman digunakan.

Daftar Pustaka

Artikel Jurnal:

Chatterjee, N., Chakraborty, S., Decosta, A., & Nath, A. (2018). Real-time Communication Application Based on Android Using Google Firebase. *IJARCSMS*, *6*(4), 74–79.

Daemen, J., & Rijmen, V. (2013). *The Design of Rijndael: AES - The Advanced Encryption Standard Information Security and Cryptography*. Leuven: Springer Science & Business Media.

Emmadi, S. S. R., & Potluri, S. (2019). Android Based Instant Messaging Application Using Firebase. *IJRTE*, *7*(5), 352–355.

Kumar S, A. (2018). *Mastering Firebase for Android Development: Build real-time, scalable, and cloud-enabled Android apps with Firebase*. Birmingham: Packt Publishing.

Moskala, M., & Wojda, I. (2017). *Android development with Kotlin: Learn Android application development with the extensive features of Kotlin*. Birmingham: Packt Publishing.

Nugroho, A. R., & Pramusinto, W. (2018). Implementasi Kriptografi dengan Algoritma Caesar Cipher, AES 192 dan DES untuk Aplikasi Pesan Instan Berbasis Android. *SKANIKA*, *1*(1), 14–19.

Sandy, L. A., Akbar, R. J., & Hariadi, R. R. (2017). Rancang Bangun Aplikasi Chat pada Platform Android dengan Media Input Berupa Canvas dan Shareable Canvas untuk Bekerja dalam Satu Canvas Secara Online. *JTITS*, *6*(2), A454–A457. https://doi.org/10.12962/j23373539.v6i2.23782

Suryana, D. (2018). *Android Studio: Belajar Android Studio* (Vol. 1). Jakarta: Dayat Suryana Independent.

Buku:

Kurniawan, D. (2013). *Metodologi Pengembangan Software*. Semarang: Fasilkom.