

Aplikasi *Mobile* Periklanan Digiprom Berbasis Android

¹Muhammad Hafiz Taruligani Sihite, ²Ardiansyah, ³Rizky Prabowo, dan ⁴Febi Eka Febriansyah

^{1,2,3,4} Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung
Jalan Soemantri Brojonegoro No.1 Gedung Meneng, Bandar Lampung
e-mail: hafiztaruligani@gmail.com, ardiansyah@fmipa.unila.ac.id, rizky.prabowo@fmipa.unila.ac.id,
febi.febriansyah02@fmipa.unila.ac.id

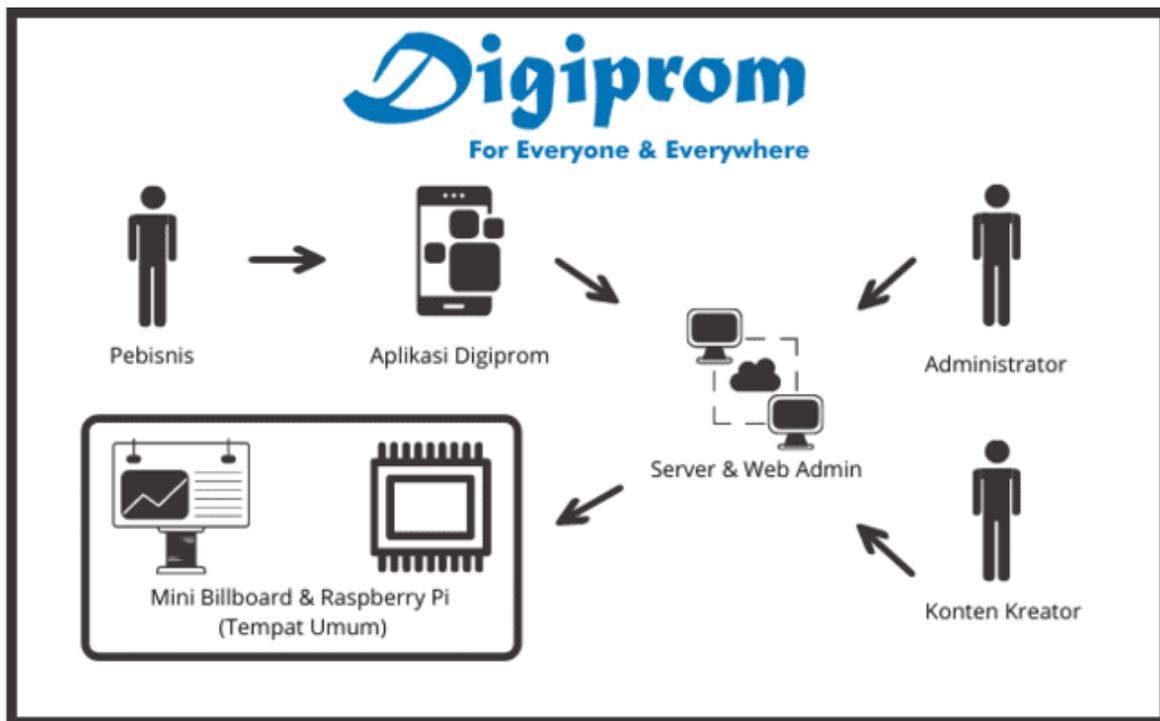
Abstract — *Digiprom is a startup that provides ad serving services. These advertisements will later be shown in public places, such as supermarkets, cafes, malls, and others. The problem faced by Digiprom is that there is no information technology to help interact between Digiprom and its clients (businessman). The interactions in question are such as registration as a client, the process of making a contract between the client and Digiprom, the process of monitoring the contract that has been made and the delivery of other information about Digiprom. To solve this problem, an Android-based mobile application will be developed to help Digiprom interact with its clients. This research focuses on the development side of the Android application, while the server side is not included in this study. The development of this application will use the Waterfall method and tested using black box testing and user acceptance testing. The system design is described using three UML diagrams, namely use case diagrams, activity diagrams and class diagrams. Program code written using Kotlin language with MVVM architecture (Model, View, ViewModel). There are three main objects in this system, namely users (businessman/clients), ads (advertisements) and location (advertising locations). Each object will have its own repository. The repository has three data sources, namely network (server), Room Database, and Data Store. The use of MVVM architecture facilitates the development and maintenance process, because if there is a change in the data structure of an object, it will not affect the data structure of other objects. The results of this research (Android application) will be uploaded to the Google Play Store under the name Digiprom.*

Keywords: *Advertisement; Android; Kotlin.*

1. PENDAHULUAN

Bisnis merupakan suatu kegiatan individu atau kelompok yang melakukan aktivitas produksi, pembelian atau penjualan dalam bentuk barang maupun jasa untuk meraih suatu tujuan tertentu [1]. Dalam bidang bisnis, para pebisnis membutuhkan suatu bentuk strategi untuk mengembangkan bisnisnya. Salah satu bentuk strategi itu adalah menggunakan iklan untuk mempromosikan produk-produk bisnis yang mereka jalani. Iklan merupakan suatu bentuk komunikasi yang bersifat komersial dan non-personal tentang ide-ide, barang atau jasa melalui suatu media tertentu [2]. Media iklan di era sekarang ini juga bervariasi, dari media siaran, cetak, maupun secara *online*.

Digiprom merupakan sebuah *startup* baru yang bergerak dalam bidang periklanan yang beroperasi di wilayah Bandar Lampung. *Storyboard* dari Digiprom digambarkan pada Gambar 1. Gambar tersebut menggambarkan Digiprom menampung iklan dari para klien dalam bentuk video berdurasi 10 detik. Kumpulan iklan tersebut nantinya akan dikelola oleh *administrator* dan konten kreator untuk ditampilkan melalui beberapa perangkat media layar dengan ukuran 40 *inch*. Media layar direncanakan akan terpasang di berbagai tempat-tempat umum seperti rumah makan, bengkel, kafe, swalayan dan lain-lain. Iklan para klien nantinya akan ditampilkan pada tempat-tempat sesuai dengan kontrak yang sudah ditetapkan dan disepakati antara Digiprom dengan kliennya.



Gambar 1. *Storyboard* Digiprom

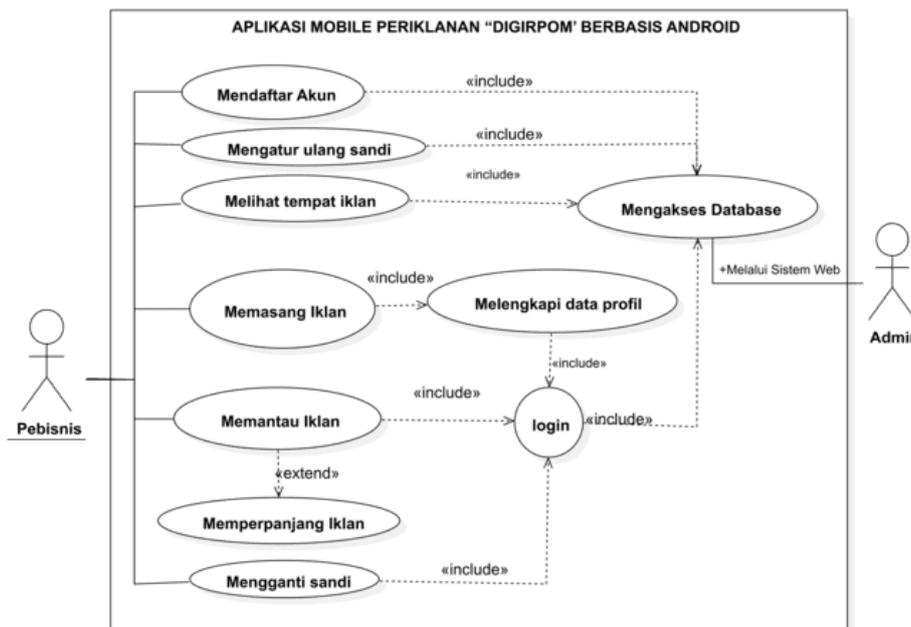
Permasalahan yang dihadapi oleh Digiprom adalah belum adanya teknologi informasi [3] untuk membantu interaksi antara Digiprom dengan kliennya. Interaksi yang dimaksud adalah seperti pendaftaran menjadi klien, proses pembuatan kontrak antara klien dengan Digiprom, proses pemantauan kontrak yang sudah dibuat dan penyampaian informasi lainnya mengenai Digiprom. Untuk menyelesaikan masalah ini, maka akan dibangun sebuah aplikasi *mobile* yang berbasis Android [4]. Aplikasi tersebut diharapkan akan dapat memudahkan interaksi antara Digiprom dengan kliennya, sehingga membantu bisnis yang dijalankan oleh Digiprom.

2. METODOLOGI PENELITIAN

Pengembangan aplikasi Digiprom akan menggunakan metode *Waterfall*, sehingga tahapannya adalah analisis kebutuhan, desain sistem, desain *interface*, penulisan kode program dan pengujian [5]. Sumber data penelitian ini didapatkan dengan cara melakukan wawancara dengan pihak Digiprom. Beberapa poin dari hasil wawancara dijadikan sebagai kebutuhan sistem, di antaranya manajemen akun *user* (pebisnis), memasang iklan, memantau iklan, dan penyampaian informasi mengenai Digiprom.

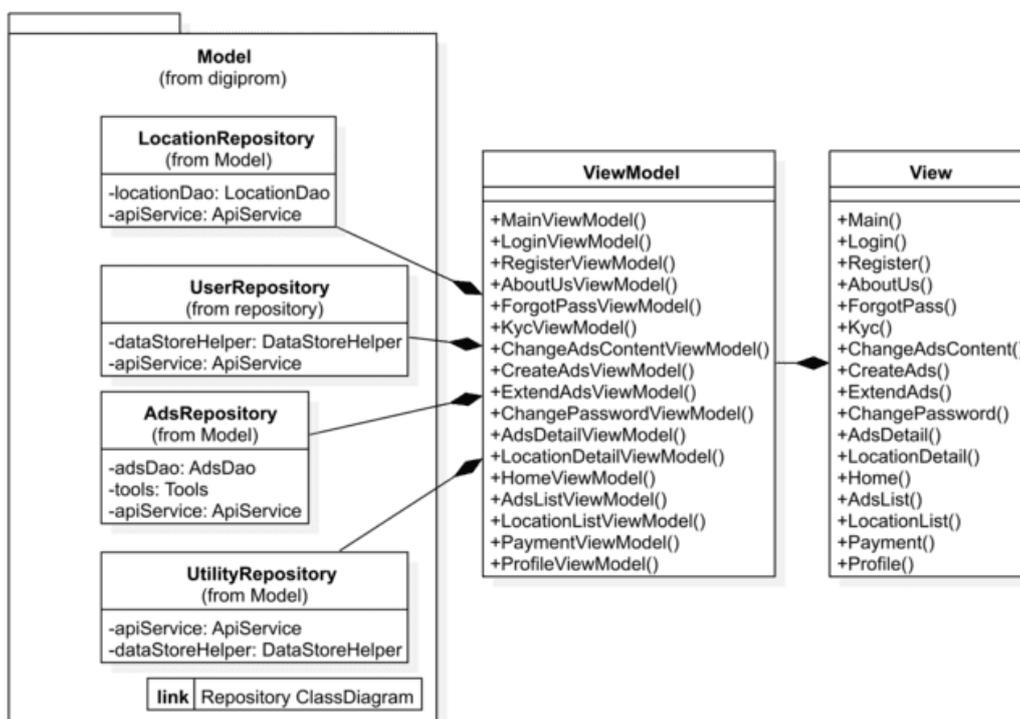
2.1. Desain Sistem

Desain sistem dari aplikasi ini menggunakan tiga buah diagram *unified modelling language* (UML) [6], yaitu *use case diagram* [7], *activity diagram* [8], dan *class diagram* [9]. Sistem ini memiliki dua aktor atau pengguna, yaitu *user* dan admin. Akan tetapi admin hanya dapat mengakses *database* melalui *website*, sehingga desain sistem ini akan lebih berfokus pada aktor *user*. Gambar 2 menggambarkan *use case diagram* Digiprom secara keseluruhan pada beberapa aktivitas, *user* harus melakukan *login* terlebih dahulu sebelum melakukan aktivitas tersebut. Hal tersebut dilakukan agar *user* dapat melihat aktivitas lainnya secara sekilas terlebih dahulu sebelum melakukan pendaftaran akun. Setiap aktivitas yang ada pada *use case* ini memiliki *activity diagram* masing-masing, akan tetapi tidak dituliskan dalam jurnal ini.



Gambar 2. Use case diagram Digiprom.

Sistem ini dikembangkan menggunakan paradigma arsitektur MVVM (*Model, View, ViewModel*) dan memiliki tiga objek utama, yaitu *user, location, dan ads* (iklan). Objek-objek tersebut akan dikelola dengan kelas repositorinya masing-masing, akan tetapi tidak menutup kemungkinan satu repositori membutuhkan repositori lainnya. Repositori memiliki data dari tiga sumber, yaitu *network* (server), penyimpanan lokal berbasis SQL, dan penyimpanan lokal berbasis *key-value pair*. Terlihat pada Gambar 3 bahwa *repository* nantinya akan terhubung dengan *ViewModel*, dan *ViewModel* terhubung dengan *View*.



Gambar 3. Main class diagram Digiprom.

2.2. Penulisan dan Pengujian Kode Program

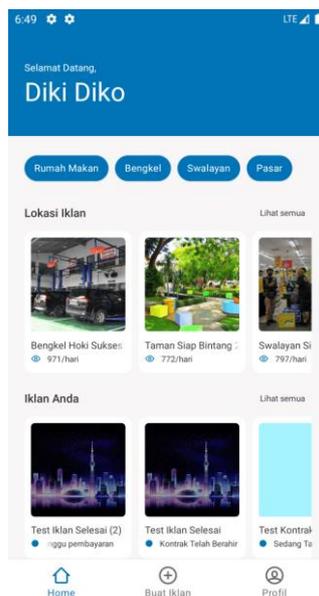
Bahasa yang digunakan dalam penulisan kode program adalah Kotlin [10] dengan SQL *database* lokal menggunakan *library* Room [11] dan *Structured Query Language* (SQL) *database* server. Penulisan kode program dilakukan dengan menggunakan arsitektur *model-view-viewmodel* (MVVM) sehingga bentuk data lebih terstruktur dan kode program tersusun rapi. Hal ini akan memudahkan *developer* dalam melakukan *maintenance* pada aplikasi. Aplikasi berjalan baik apabila tidak terjadi *bottleneck*, sehingga digunakan *Coroutine* [12] pada beberapa bagian untuk mengontrol jalannya kode program. Pengujian sistem ini dilakukan dengan menggunakan dua metode, yaitu metode *Black Box Testing* [13] dan metode *User Acceptance Testing* (UAT) [14]. Pengujian UAT dilakukan dalam bentuk pertanyaan langsung kepada pihak Digiprom. Setiap jawaban dari pertanyaan tersebut memiliki bobot penilaian masing-masing [15].

3. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini adalah berupa aplikasi *mobile* berbasis Android beserta hasil pengujiannya. Hal tersebut akan dijabarkan dalam bentuk *screenshots* dari beberapa halaman aplikasi, setelahnya akan dijabarkan hasil pengujiannya. Selanjutnya Digiprom akan disebut sebagai admin dan pebisnis akan disebut sebagai *user*.

3.1. Halaman Beranda

Halaman beranda memiliki tampilan seperti pada Gambar 4. Terdapat dua bagian tampilan utama, yaitu bagian tampilan lokasi iklan dan bagian tampilan iklan yang dimiliki oleh *user*. Pada bagian tampilan lokasi iklan menampilkan *list* lokasi yang dapat digunakan oleh *user* untuk memasang iklan. Setiap item *list* tempat menampilkan informasi di antaranya foto, nama dan jumlah pengunjung harian dari tempat tersebut. Terdapat tombol “lihat semua”, *user* akan diarahkan ke halaman *list* iklan yang lebih detail. Jika salah satu item *list* di klik maka aplikasi akan mengarahkan ke halaman detail dari lokasi tersebut. Pada bagian tampilan kedua (tampilan “Iklan Anda”), terdapat *list* iklan yang dimiliki oleh *user*. Item *list* iklan tersebut menampilkan informasi nama, status dan foto iklan.

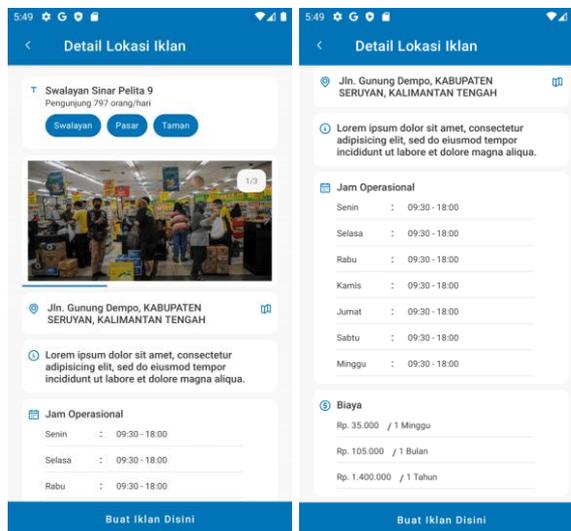


Gambar 4. Halaman beranda.

3.2. Halaman Detail Lokasi Iklan

Gambar 5 (a) merupakan bagian atas tampilan terdapat nama dari tempat iklan, jumlah pengunjung harian, serta kategori dari lokasi tersebut. Selanjutnya terdapat beberapa foto dari tempat iklan tersebut. Pada bagian

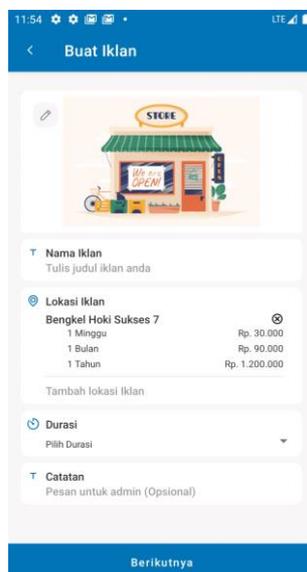
alamat lokasi iklan, terdapat sebuah tombol *map* yang apabila *user* menekan tombol tersebut, maka akan diarahkan ke aplikasi *map* yang menunjukkan lokasi tersebut pada *map*. Kemudian pada Gambar 5 (b) merupakan bagian bawah tampilan, terdapat deskripsi dari tempat tersebut beserta jam operasional dan biaya pemasangan iklan di tempat tersebut. Di bagian paling bawah dari tampilan ini terdapat tombol “Buat iklan disini” yang apabila ditekan, maka *user* akan diarahkan ke halaman pembuatan iklan dengan mencantumkan lokasi tersebut pada formulir pembuatan iklannya.



Gambar 5. Halaman detail lokasi (a) bagian atas; (b) bagian bawah.

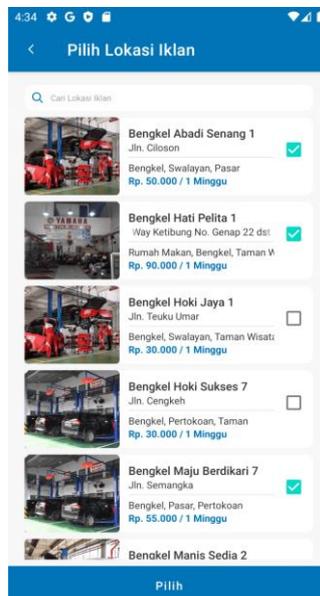
3.3. Halaman Buat Iklan

Halaman ini memiliki tampilan seperti pada Gambar 6. Pembuatan iklan hanya bisa dilakukan oleh *user* apabila *user* telah memasukkan akun ke dalam aplikasi dan sudah melengkapi data profil. Apabila *user* belum melengkapi data profil, maka aplikasi akan memunculkan pemberitahuan untuk melengkapi data profil terlebih dahulu dan diarahkan ke halaman untuk melengkapi data profil. *User* dapat memasukkan nama iklan dan menambahkan *file* yang akan dijadikan konten iklan dengan menekan tombol bergambar pensil. Jenis *file* yang dapat digunakan adalah *file* video atau *file* gambar.



Gambar 6. Halaman buat iklan.

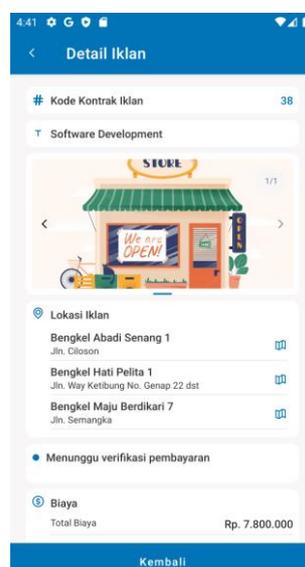
Untuk menambahkan lokasi iklan, *user* dapat menekan tombol tambah tempat, kemudian akan diarahkan ke halaman *list* tempat dengan *checkbox* yang menandakan lokasi yang dipilih. Hal tersebut memiliki tampilan seperti pada Gambar 7. Setelah lokasi iklan dipilih, *user* dapat memilih durasi dari pemasangan dengan harga yang sudah tertera. Durasi yang dipilih hanya bisa satu jenis untuk semua tempat. Terdapat juga kolom catatan untuk admin yang bersifat opsional. Setelah formulir terisi lengkap, *user* dapat menekan tombol “Berikutnya” agar formulir tersebut diunggah ke *server* Digiprom dan mengarahkan ke halaman detail pembayaran.



Gambar 7. Halaman memilih lokasi pemasangan iklan.

3.4. Halaman Detail Iklan

Berdasarkan Gambar 8, pada bagian atas halaman terdapat informasi berupa kode kontrak iklan yang dimiliki oleh iklan tersebut, diikuti oleh nama iklan dan konten iklan yang sedang ditayangkan. Tampilan “Lokasi Iklan” menunjukkan informasi di mana saja iklan tersebut ditayangkan, yang apabila ditekan maka akan mengarah ke aplikasi Google Maps dengan menunjukkan lokasi tersebut. Terdapat juga informasi yang menunjukkan status dari iklan tersebut seperti menunggu pembayaran, menunggu verifikasi pembayaran oleh admin, iklan sedang tayang atau status yang lainnya.



Gambar 8. Halaman detail iklan.

3.5. Hasil Pengujian

Pengujian yang pertama dilakukan dengan menggunakan metode *black-box testing*. Pengujian ini dilakukan oleh penguji independen dengan menggunakan kelas uji dari setiap halaman atau aktivitas pada hasil implementasi. Total skenario yang ada pada pengujian ini adalah 47 skenario dengan tingkat keberhasilan 100%, sehingga secara fungsionalitas sistem ini berjalan dengan baik. Pengujian yang kedua dilakukan dengan menggunakan metode *User Acceptance Testing* (UAT), yang hasilnya dapat dilihat pada Tabel 1.

Tabel 1. Hasil perhitungan UAT.

<i>User Acceptance Testing</i>	STS	TS	C	S	SS
Bobot penilaian (BP)	1	2	3	4	5
Hasil kuesioner (HP)	0	0	8	0	0
BP x HP	0	0	24	0	0
Total Skor			24		

Pengujian ini dilakukan dengan cara memberikan pertanyaan langsung dalam bentuk kuesioner kepada pemilik bisnis Digiprom. Kuesioner berisi delapan pertanyaan dengan lima macam jawaban, yaitu STS (Sangat Tidak Setuju), TS (Tidak Setuju), C (Cukup), S (Setuju) dan SS (Sangat Setuju). Berdasarkan data pada Tabel 1, jumlah total skor yang didapatkan adalah 24 dari total maksimal skor 40. Apabila nilai tersebut dipersentasekan, maka akan bernilai 60%, sehingga dapat diambil kesimpulan bahwa Aplikasi Digiprom cukup diterima oleh pemilik Digiprom. Adapun saran dari *user* adalah untuk meningkatkan UI/UX dari aplikasi Digiprom.

4. KESIMPULAN

Penelitian ini berhasil diselesaikan dengan terbangunnya sebuah aplikasi *mobile* berbasis Android dengan nama Digiprom. Aplikasi tersebut dibangun menggunakan bahasa Kotlin dengan fitur-fitur yang membantu interaksi pebisnis dengan Digiprom. Hasil *black box testing* menunjukkan fungsionalitas aplikasi berjalan sepenuhnya dengan tingkat keberhasilan dari semua skenario adalah 100%, sedangkan hasil *User Acceptance Testing* menunjukkan bahwa aplikasi cukup diterima atau cukup bermanfaat bagi Digiprom. Terdapat beberapa kekurangan dari aplikasi ini, di antaranya belum adanya *dark mode user interface*. Selain itu, jika terjadi kesalahan jaringan (*network exception*), maka pesan kesalahan yang ditampilkan oleh aplikasi masih belum spesifik mengenai penyebab dari kesalahan tersebut, sehingga diperlukan penyempurnaan *result and exception handling* pada beberapa fungsi *networking* di dalam kode program.

DAFTAR PUSTAKA

- [1] R. Sembiring, *Pengantar Bisnis*. La Goods Publishing, 2014.
- [2] R. Kriyantono, *Manajemen Periklanan: Teori dan Praktik*. Jawa Timur: Universitas Brawijaya Press, 2013.
- [3] A. Kadir & T. Ch. Triwahyuni, *Pengantar Teknologi Informasi Edisi Revisi*, Yogyakarta: Andi Yogyakarta, 2014. doi: 10.13140/2.1.4734.7840.
- [4] C. Google, *Android Developer Fundamental Course: Practical Workbook*. 2016.
- [5] A. M. Dima & M. A. Maassen, "From waterfall to agile software: Development models in the IT sector, 2006 to 2018. impacts on company management," *Journal of International Studies*, 11(2), 315–326, 2018, doi: 10.14254/2071-8330.2018/11-2/21.
- [6] R. S. Wazlawick, *Object-Oriented Analysis and Design for Information Systems*. Elsevier, 2014.

- [7] A. Hendini, Pemodelan UML Sistem Informasi Monitoring Penjualan dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak), *Jurnal Khatulistiwa Informatika*, 4(2), 107-116, 2016.
- [8] F. N. Hasanah & R. S. Untari, *Buku Ajar Rekayasa Perangkat Lunak*. UMSIDA PRESS, 2020. doi: 10.21070/2018/978-602-5914-09-6.
- [9] B. Mallick & N. Das, "An Approach to Extended Class Diagram Model of UML for Object Oriented Software Design," *International Journal of Innovative Technology & Adaptive Management (IJITAM)*, 1(2), 2013.
- [10] N. S. Sibarani, G. Munawar, & B. Wisnuadhi, Analisis Performa Aplikasi Native Android Menggunakan Bahasa Pemrograman Java dan Kotlin, *Prosiding Industrial Research Workshop and National Seminar*, Vol. 9, 2018.
- [11] (2022), C. Google, "Room | Android Developers," [Online] <https://developer.android.com/training/data-storage/room> (accessed Jul. 06, 2022).
- [12] (2022), C. Google, "Kotlin coroutines on Android | Android Developers," <https://developer.android.com/kotlin/coroutines> (accessed Jul. 06, 2022).
- [13] T. J. Snadhika, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)," *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, vol. 03, no. 02, pp. 45–48, 2018, doi: 10.30591/jpit.v3i1.647.
- [14] B. Hambling & P. van Goethem, *User Acceptance Testing: A step-by-step guide*. BCS Learning and Development Ltd, 2013.
- [15] J. Abraham & I. E. Ismail, *Unit Testing dan User Acceptance Testing pada Sistem Informasi Pelayan Kategorial Pelayanan Anak*, n.d.