

PENERAPAN POLA ARSITEKTUR MVVM PADA PERANCANGAN APLIKASI PENGADUAN MASYARAKAT BERBASIS ANDROID

Irya Muhammad Riyadhi¹, Intan Purnamasari, M.Kom.², Kamal Prihandani, M.Kom.³

^{1,2,3}Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

Email: iryamuhammadr@gmail.com

ABSTRACT

The Model-View-ViewModel (MVVM) architectural pattern can be applied in designing a mobile complaint application. MVVM separates the graphical interface and business processes, making it easier to develop and maintain applications. In an MVVM implementation, event and observer data are separated, observing calls from event listeners are avoided, and state is used for data. The MVVM architectural pattern can also reduce dependencies so that there is no close relationship between the view and the viewmodel. In addition, MVVM can also be used to manage the concentration of Android application development that functions as a public complaint. In this community complaint application it is intended that the public can notify their complaints without having to come to the location of the complaint. In addition, it can make it easier for the authorities to follow up on these complaints, so that the public government can provide better service and be responsive to complaints from the public.

Keywords: MVVM, Community Complaint, Android Application

ABSTRAK

Pola arsitektur Model-View-ViewModel (MVVM) dapat diterapkan dalam perancangan aplikasi mobile pengaduan masyarakat. MVVM memisahkan antara grafis antarmuka dan proses bisnis, sehingga memudahkan pengembangan dan perawatan aplikasi. Dalam penerapan MVVM, event dan observer data dipisahkan, pemanggilan observe dari event listener dihindari, dan state dimanfaatkan untuk data. Pola arsitektur MVVM juga dapat mengurangi ketergantungan sehingga tidak ada hubungan erat antara view dan viewmodel. Selain itu, MVVM juga dapat digunakan untuk mengelola konsentrasi pengembangan aplikasi Android yang berfungsi sebagai pengaduan masyarakat. Dalam aplikasi pengaduan masyarakat ini bertujuan agar masyarakat dapat memberi tahu keluhannya tanpa harus datang ke lokasi pengaduan. Selain itu, dapat memudahkan pihak berwenang untuk menindaklanjuti pengaduan tersebut, sehingga pemerintah public dapat memberikan pelayanan yang lebih baik dan responsive terhadap keluhan yang diberikan masyarakat.

Kata Kunci: MVVM, Pengaduan Masyarakat, Aplikasi Android

Riwayat Artikel :

Tanggal diterima : 10-05-2023

Tanggal revisi : 11-05-2023

Tanggal terbit : 12-05-2023

DOI :

<https://doi.org/10.31949/infotech.v9i1.5246>

INFOTECH journal by Informatika UNMA is licensed under CC BY-SA 4.0

Copyright © 2023 By Author



1. PENDAHULUAN

1.1. Latar Belakang

Desa merupakan salah satu bagian k dari pemerintah yang memiliki peran penting dalam proses pembangunan suatu daerah. Selain itu, Desa juga merupakan bagian di mana pemerintah berinteraksi dengan masyarakat secara langsung terutama dalam menerima pengaduan dan keluhan masyarakat. Banyaknya pengaduan dan tuntutan masyarakat kepada pemerintah menjadikan pemerintah harus mampu menangani pengaduan dan keluhan dari masyarakat secara cepat dan efektif. Pengaduan atau keinginan yang disampaikan dan pengaduan dari masyarakat merupakan hal yang penting bagi sebuah institusi. Karena sesuai keinginan masyarakat, lembaga pemerintahan akan sangat terbantu dikarenakan sudah mengetahui apa kebutuhan yang dibutuhkan masyarakat.

Desa Bayur Kidul adalah sebuah desa yang berlokasi di Kecamatan Cilamaya Kulon, Kabupaten Karawang, Provinsi Jawa Barat, Indonesia. Sampai saat ini desa Bayur Kidul masih melakukan penyampaian pengaduan secara tatap muka. Masyarakat harus mendatangi kantor desa secara langsung untuk menyampaikan pengaduannya. Namun pada kenyataannya terdapat banyak kendala seperti terbatasnya jam kerja aparat Desa, kesalahan peninputan data (*Human Error*) maupun ketidaktahuan masyarakat tentang cara menyampaikan pengaduannya, Imbasnya pemerintah desa tidak dapat melayani serta menampung seluruh keluhan yang diadukan oleh masyarakat.

Berdasarkan hasil wawancara dengan pemerintah Desa Bayur Kidul, didapati beberapa permasalahan yang dialami oleh masyarakat di Desa Bayur Kidul terutama adalah masalah kebersihan lingkungan yang kerap menjadi masalah utama di desa tersebut. Kurangnya kesadaran masyarakat akan kebersihan lingkungan serta koordinasi antara masyarakat dengan pemerintah desa menjadi sumber dari masalah di desa ini. Hal tersebut mendorong untuk dibuatnya sebuah aplikasi berbasis Android sehingga dapat memudahkan koordinasi antara masyarakat dengan pemerintah Desa Bayur Kidul dalam hal pengaduan masyarakat. Pada aplikasi pengaduan ini masyarakat dan pemerintah desa dapat dengan mudah melakukan koordinasi tentang permasalahan kebersihan ini. Masyarakat dapat membuat laporan tentang kebersihan lingkungan pada aplikasi ini. Kemudian pemerintah desa akan menindaklanjuti laporan yang diterima dari masyarakat.

Perancangan sebuah aplikasi perlu memperhatikan arsitektur yang akan digunakan agar penulisan kode menjadi terstruktur dan mudah untuk dirawat pada saat pengembangan, pengujian, dan pemeliharaan aplikasi (Martin, 2018). Salah satu pola arsitektur yang dapat diterapkan pada perancangan aplikasi ini adalah pola arsitektur MVVM. Penerapan pola arsitektur MVVM ini memungkinkan aplikasi untuk

mempertahankan keadaan dari perubahan konfigurasi yang terjadi pada *smartphone* pengguna, serta mempermudah akses data. Pola arsitektur MVVM juga memecah kode menjadi beberapa komponen, yaitu *Model*, *View*, dan *ViewModel*. Oleh karena itu, penelitian yang dilakukan ini akan difokuskan pada bagaimana merancang sebuah aplikasi pengaduan masyarakat dengan menerapkan pola arsitektur MVVM yang diharapkan dapat dimanfaatkan oleh masyarakat Desa Bayur Kidul dan dapat dikembangkan serta dipelihara dengan baik.

1.2. Tinjauan Pustaka

Aplikasi pengaduan masyarakat berbasis Android dapat membantu masyarakat untuk melaporkan masalah atau permasalahan yang ada di lingkungan sekitarnya. Namun, pengembangan aplikasi ini harus mempertimbangkan arsitektur yang tepat supaya aplikasi dapat berjalan dengan baik, efisien, dan dapat diuji dengan baik. Salah satu arsitektur yang dapat digunakan dalam pengembangan aplikasi ini adalah pola arsitektur Model-View-ViewModel (MVVM).

Pola arsitektur MVVM memisahkan antara tampilan (*View*), logika bisnis (*ViewModel*), dan model data (*Model*) pada aplikasi. Hal ini dapat memudahkan pengembangan dan perawatan aplikasi karena setiap bagian aplikasi memiliki tanggung jawab yang jelas dan terpisah. Selain itu, pola arsitektur MVVM juga dapat meningkatkan pengalaman pengguna, kinerja aplikasi, dan efisiensi pengembangan.

Dalam tinjauan pustaka ini, akan dijelaskan beberapa referensi jurnal yang membahas tentang penerapan pola arsitektur MVVM pada perancangan aplikasi pengaduan masyarakat berbasis Android. Referensi pustaka ini dapat membantu dalam memahami bagaimana menerapkan pola arsitektur MVVM pada aplikasi pengaduan masyarakat dan meningkatkan kualitas serta efisiensi pengembangan aplikasi.

Ali Ikhwan, dkk (2023) dalam penelitiannya yang berjudul "Perancangan Sistem Informasi Laporan Pengaduan Masyarakat Berbasis WEB pada Dinas ESDM SUMUT" menuliskan bahwa aplikasi pengaduan masyarakat dapat dirancang dan dikembangkan dengan menggunakan *platform android*. User dapat dengan mudah meng-unggah keluhan yang dirasakannya dengan cara mengisi form yang sudah disediakan. User diharuskan meng-upload bukti kejadian dalam bentuk foto dan deskripsi kejadian. Tetapi, aplikasi ini tidak memiliki umpan balik sebagai apresiasi kepada peng-unggah laporan.

Penelitian selanjutnya yang dilakukan oleh Galang Bayu Dwianto (2021) pada jurnalnya yang berjudul "Rancang Bangun Aplikasi Pengaduan Berbasis Android" juga menunjukkan bahwa keluhan yang dilakukan masyarakat tentang fasilitas umum kepada pemerintah dapat dikembangkan ke dalam aplikasi mobile. Pengguna dapat melakukan pengaduan

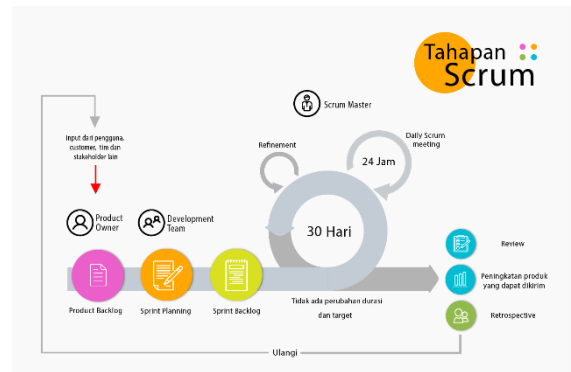
dengan mudah dengan cara mengisi form yang telah disediakan. Namun, aplikasi yang dikembangkan ini masih belum dapat memberikan umpan balik ke pengguna yang mengirimkan keluhannya.

Selain itu, Bambang Yuwono, dkk (2017) dalam penelitiannya yang berjudul “Pengembangan Aplikasi Mobile pada Pelayanan Pemerintah Kota Yogyakarta dalam Rangka Tanggap Respon Informasi dan Keluhan dari Masyarakat” membahas dan meneliti tentang aplikasi pelayanan masyarakat yang berupa aplikasi pengaduan masyarakat untuk Koya Yogyakarta. Aplikasi tersebut dapat digunakan untuk mengirimkan keluhan dan aspirasi masyarakat.

Dari tinjauan pustaka tersebut, dapat disimpulkan bahwa penerapan pola arsitektur MVVM pada aplikasi mobile pengaduan masyarakat berbasis Android dapat mempermudah masyarakat dalam menyampaikan keluhan dan aspirasinya. Aplikasi pengaduan masyarakat yang berbasis Android ini dapat dirancang dan dikembangkan dengan menggunakan platform Android, dan pengguna atau user dapat melakukan pengaduan dengan cara mengirimkan foto beserta deskripsinya pada form yang telah disediakan. Namun, perlu ditambahkan fitur untuk mengirim video dan suara agar lebih memudahkan petugas dalam menangani pengaduan. Oleh karena itu, penerapan MVVM pada aplikasi mobile pengaduan masyarakat berbasis Android dapat menjadi solusi yang tepat untuk meningkatkan kualitas pelayanan publik dan mempermudah masyarakat dalam menyampaikan keluhan dan aspirasinya.

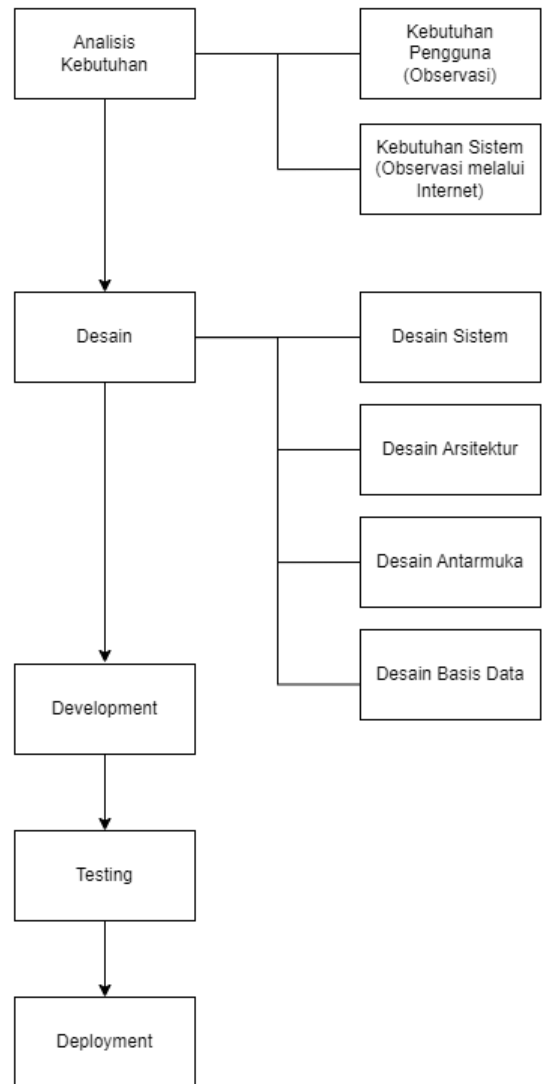
1.3. Metodologi Penelitian

Metode pengembangan perangkat lunak dengan model agile adalah metodologi yang dipilih sebagai metodologi yang dilakukan pada penelitian ini. Model *agile* merupakan salah satu metode pengembangan perangkat lunak yang dilakukan secara berulang dan bertahap. Model *agile* ini sangat cocok digunakan untuk proyek kecil yang persyaratannya cepat berubah, namun tidak untuk yang mempunyai proyek besar karena akan sulit untuk memprediksi waktunya. Model *agile* memiliki beberapa macam *framework*. Dalam penelitian ini penulis menggunakan *framework scrum*. Tahapan yang pertama dilakukan pada *framework scrum* adalah melakukan analisis kebutuhan pengguna dan kebutuhan sistem, menentukan *product backlog*, merancang *design*, menentukan *sprint*, mengeksekusi *sprint* yang telah dibuat, melakukan pengujian serta melakukan *deployment*.



Gambar 1. Tahapan Scrum

Rancangan awal yang akan dibuat pada penelitian ini yaitu, mengumpulkan data dengan cara melakukan observasi, pengamatan dan wawancara secara langsung. Langkah selanjutnya yaitu pengembangan aplikasi dengan menggunakan metode agile yang terdiri dari analysis, design, development, testing, deployment. Pada saat tahap deployment ini aplikasi akan dirilis ke internet agar bisa diakses dengan menggunakan smartphone.



Gambar 2. Alur Penelitian dengan Model Agile

a) Requirements

Ini merupakan tahap perancangan apa saja yang akan dibutuhkan seperti, pengumpulan data dan kebutuhan pengguna yang diperlukan untuk membuat suatu aplikasi yang akan dibuat agar pengembang dapat menentukan data dan sistem yang pengguna butuhkan sehingga proses pembuatan aplikasi dapat diimplementasikan dengan baik.

b) Design

Langkah selanjutnya yaitu membuat desain, dimulai dengan membuat desain arsitektur seperti *use case diagram*, *activity diagram*, *class diagram*, serta *sequence diagram*, yang kemudian dilanjutkan dengan membuat desain *database* dan yang terakhir membuat desain tampilan *interface* aplikasi.

c) Development

Selanjutnya melakukan tahap implementasi hasil dari analisis perancangan (*Requirements*) dan hasil dari desain sistem, desain arsitektur, desain *database* dan desain tampilan *interface*. Aplikasi ini akan dibuat dengan menggunakan bahasa pemrograman Kotlin dan Android Studio sebagai IDE.

d) Testing

Tahap ini merupakan tahap pengujian dengan melakukan *Unit Test* dengan menggunakan *JUnit* dan *Mockito* untuk menguji semua fitur-fitur pada aplikasi yang telah penulis buat secara komprehensif sehingga diharapkan aplikasi dapat dinyatakan aman dari *bug* dan *error* sehingga dapat didistribusikan serta mudah untuk *maintenance* pada kemudian hari.

e) Deployment

Setelah tahap testing aplikasi selesai dan dinyatakan aman dari *error* dan *bug*, langkah terakhir yaitu melakukan tahap pendistribusian aplikasi ke internet dengan format *Android Package Kit (APK)* sehingga aplikasi dapat diinstal dan dijalankan pada *smartphone* yang kompatibel dengan format APK. Atau dengan menggunakan format *Android App Bundle (AAB)* yang merupakan format file android terbaru yang memiliki keunggulan yaitu ukuran file yang lebih kecil.

2. PEMBAHASAN

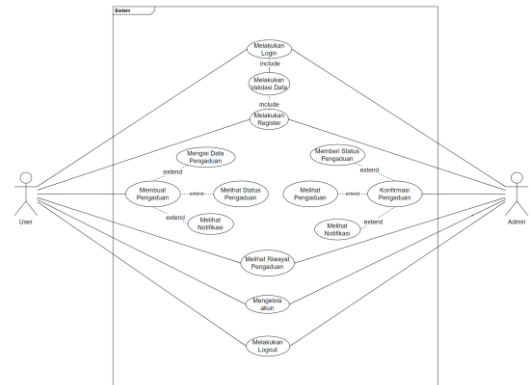
Setelah melakukan tahapan requirements, di mana pada tahap ini dibutuhkan pengumpulan data dan kebutuhan pengguna untuk mengetahui data dan sistem apa saja aplikasi agar pembuatannya dapat diimplementasikan dengan baik. Selanjutnya dilakukanlah tahapan sebagai berikut:

2.1. Desain Sistem

a) Usecase

Use case diagram merupakan jenis diagram yang digunakan di pemodelan sistem untuk menggambarkan interaksi yang dilakukan oleh pengguna (aktor) dengan sistem. Diagram ini dapat digunakan untuk mengidentifikasi kebutuhan fungsional sistem dan memvisualisasikan skenario penggunaan sistem. Penggunaan use case diagram dalam pemodelan sistem sangat penting karena dengan adanya usecase diagram dapat memastikan

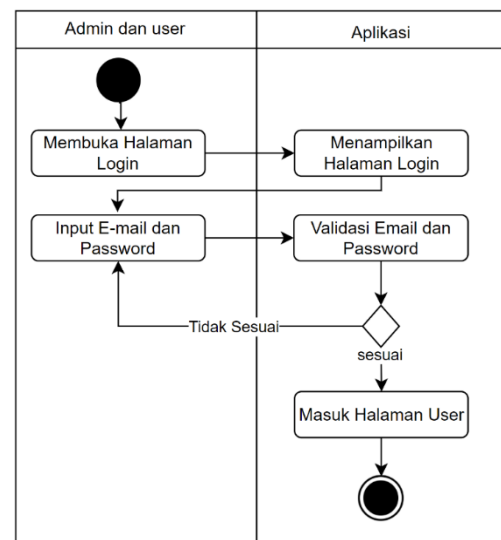
bahwa sistem yang dibangun dapat memenuhi kebutuhan pengguna dan memudahkan pengembangan sistem secara sistematis dan terstruktur.



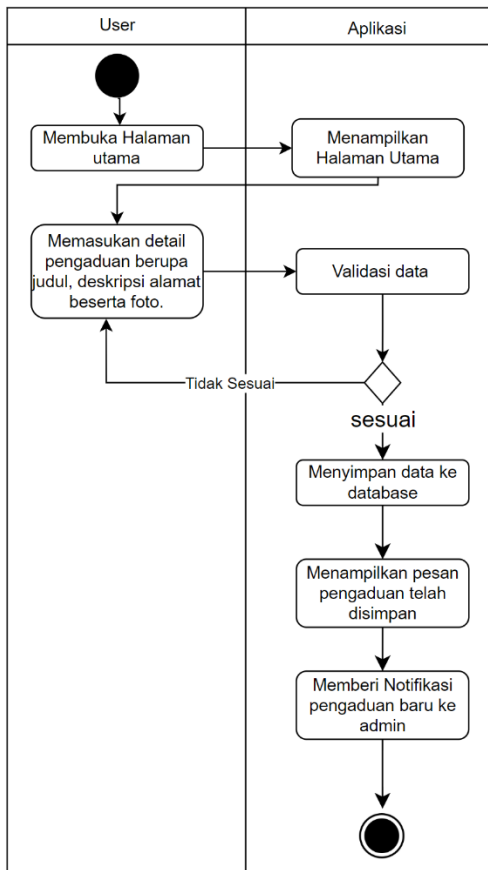
Gambar 3. Usecase Diagram

b) Activity Diagram

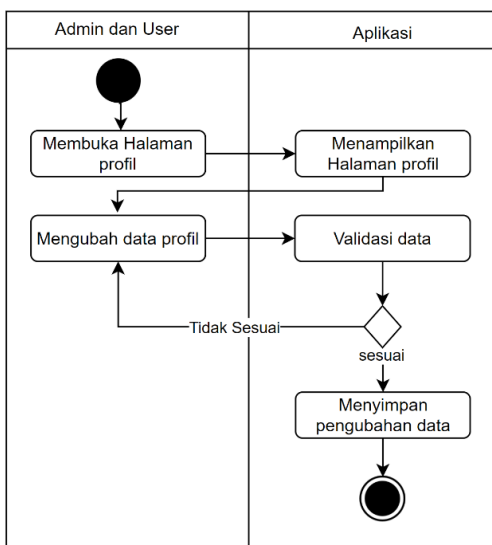
Activity Diagram merupakan jenis diagram yang digunakan dalam pemodelan sistem untuk membentuk visual dari alur kerja atau aktivitas yang terjadi pada sebuah sistem atau proses bisnis. Diagram ini digunakan untuk memodelkan proses yang terjadi pada sebuah sistem dan menggambarkan urutan proses yang terjadi dalam sistem atau proses bisnis. Activity Diagram dapat berisi aktivitas yang dilakukan maupun tindakannya. Selain itu, dapat berisi pilihan atau pengulangan. Diagram ini sangat berguna dalam pemodelan sistem karena dapat membantu pengembang sistem untuk memahami alur kerja sistem secara visual dan memudahkan dalam mengidentifikasi masalah atau kekurangan dalam sistem.



Gambar 4. Activity Diagram Login



Gambar 5. Activity Diagram Buat Pengaduan

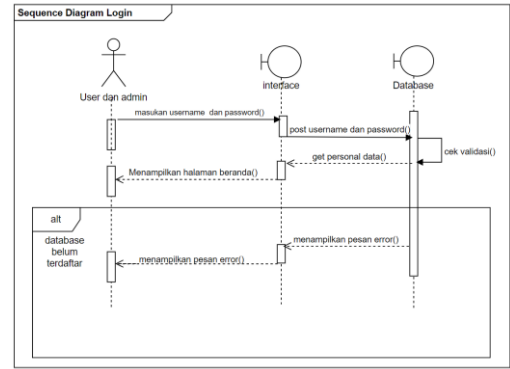


Gambar 6. Activity Diagram Kelola Akun

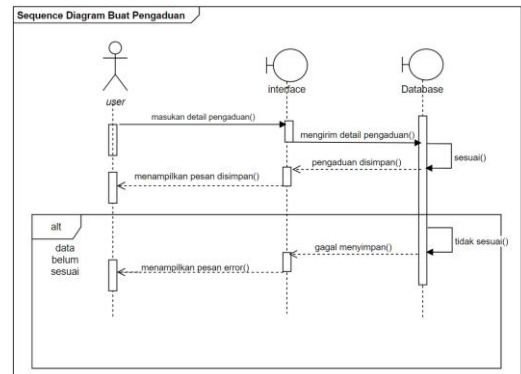
c) Sequence Diagram

Sequence Diagram adalah jenis diagram yang digunakan dalam pemodelan sistem untuk menggambarkan interaksi yang dilakukan oleh objek didalam sistem. Diagram ini menunjukkan urutan pesan yang dikirim antara objek dan bagaimana objek tersebut berinteraksi satu sama lain dalam sistem. Sequence Diagram sangat berguna dalam pemodelan sistem karena dapat membantu pengembang sistem untuk memahami interaksi antara objek dalam sistem secara visual dan memudahkan dalam

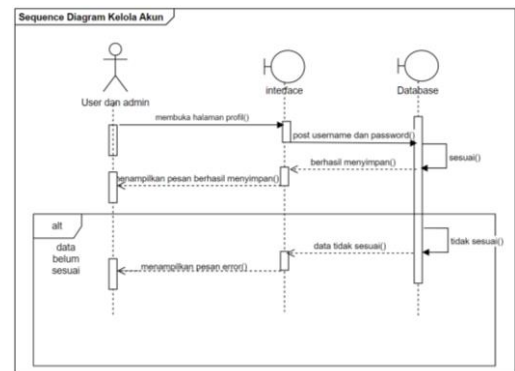
mengidentifikasi masalah atau kekurangan dalam sistem.



Gambar 7. Sequence Diagram Login



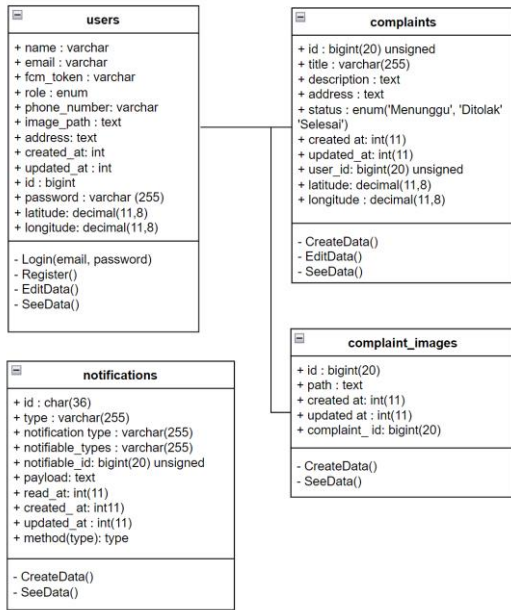
Gambar 8. Sequence Diagram Buat Pengaduan



Gambar 9. Sequence Diagram Kelola Akun

d) Class Diagram

Class Diagram adalah jenis diagram yang ditujukan dalam pemodelan sistem untuk menggambarkan struktur sistem atau perangkat lunak. Diagram ini menunjukkan kelas-kelas dalam sistem, atribut-atribut yang dimiliki oleh kelas, serta hubungan antara kelas-kelas tersebut. Class Diagram sangat berguna dalam pemodelan sistem karena dapat membantu pengembang sistem untuk memahami struktur sistem secara visual dan memudahkan dalam mengidentifikasi masalah atau kekurangan dalam sistem.

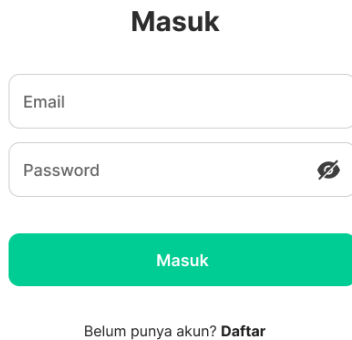


Gambar 10. Class Diagram

2.2. Implementasi pada Aplikasi

Pada tahap ini, diimplementasikan User Interface yang dirancang untuk memiliki fitur sebagai berikut:

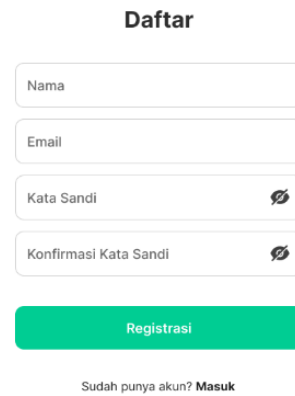
- a) Halaman Login



Gambar 11. Halaman Login

Halaman ini adalah halaman login untuk admin dan user yang menampilkan form login yang berisi email dan password sebagai persyaratan login bagi User dan Admin. Halaman login juga memiliki tombol untuk mengarah ke halaman registrasi jika pengguna belum memiliki akun.

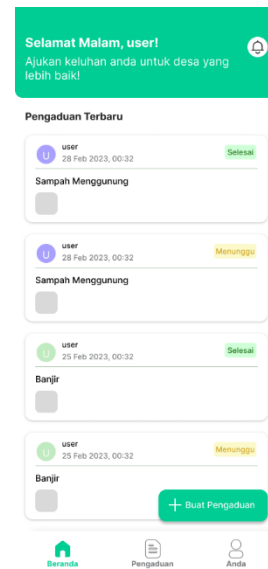
- b) Halaman Register



Gambar 12. Halaman Register

Pada gambar diatas menampilkan form registrasi untuk admin dan halaman Register user dan admin. Di mana admin dan user diminta untuk mengisi form registrasi yang berisi Nama, email, kata sandi dan konfirmasi kata sandi. Setelah itu user dan admin dapat menekan tombol registrasi. Jika data yang dimasukkan sesuai maka akan menampilkan pesan berhasil menyimpan, tetapi jika data yang dimasukkan tidak sesuai maka sistem akan menolaknya, dan menampilkan pesan error.

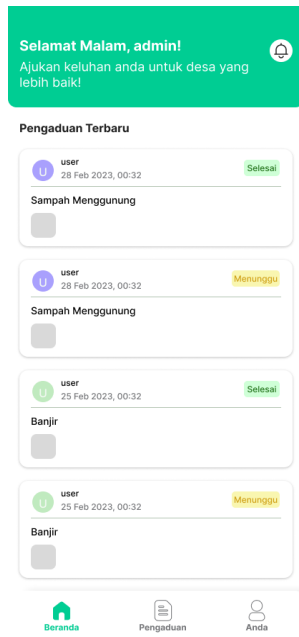
- c) Halaman Home untuk User



Gambar 13. Halaman Home User

Pada gambar diatas menampilkan halaman beranda untuk user. Halaman ini menampilkan beberapa fitur seperti buat pengaduan, notifikasi, daftar pengaduan terbaru. tersedia juga menu untuk melihat riwayat pengaduan. dan yang terakhir tersedia menu profil untuk kelola akun user beserta logout di dalamnya.

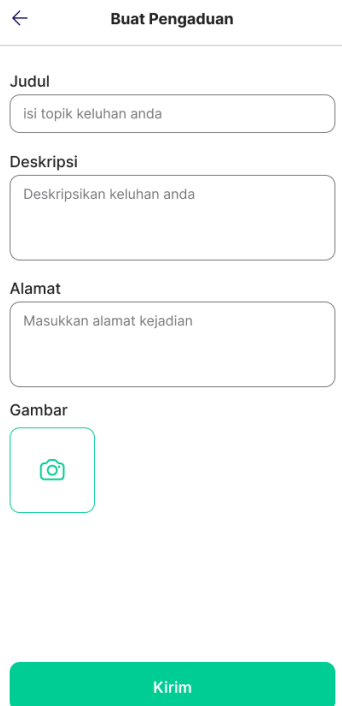
- d) Halaman Home Admin



Gambar 14. Halaman Home Admin

Pada gambar diatas menampilkan halaman beranda untuk admin. Halaman ini mirip seperti halaman beranda untuk *user*, namun perbedaannya admin tidak memiliki fitur untuk membuat aduan. Halaman ini menampilkan beberapa fitur seperti notifikasi, daftar pengaduan terbaru. tersedia juga menu untuk melihat riwayat pengaduan. dan yang terakhir tersedia menu profil untuk kelola akun admin beserta *logout* di dalamnya

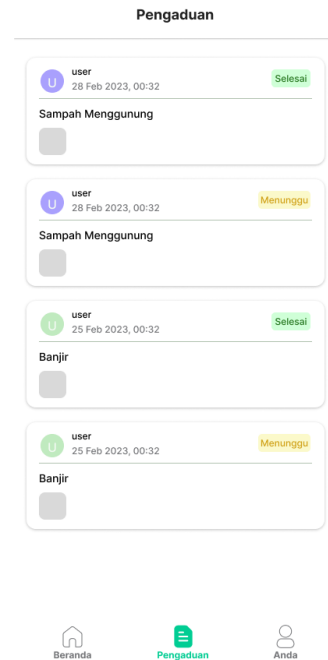
e) Halaman Buat Pengaduan



Gambar 15. Halaman Buat Pengaduan

Pada gambar diatas menampilkan halaman buat pengaduan yang berisikan form yang harus diisi sebagai syarat buat pengaduan, agar pengaduan yang diajukan memenuhi kriteria yang sudah ditentukan. form berisi judul pengaduan, Deskripsi dari pengaduan, alamat terjadinya masalah yang ingin diajukan beserta gambar dari masalahnya.

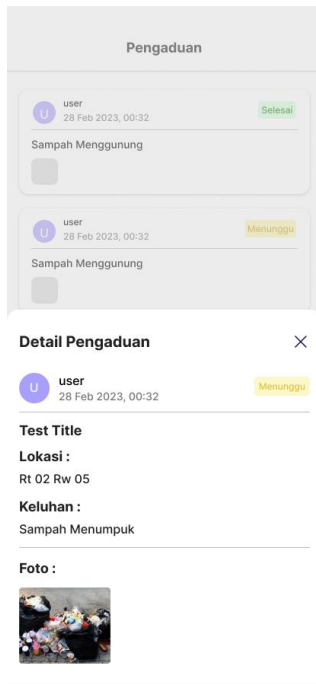
f) Halaman Riwayat Pengaduan



Gambar 16. Halaman Riwayat Pengaduan

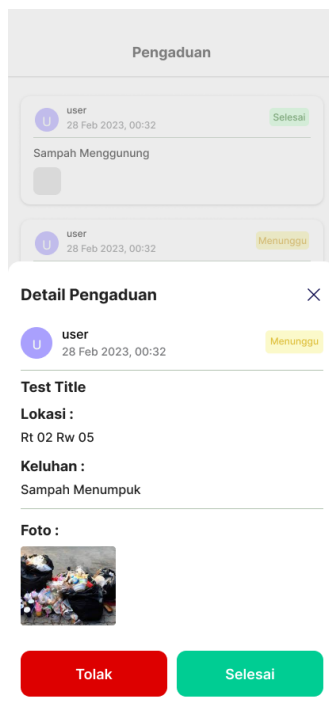
Pada gambar diatas menampilkan halaman dari riwayat pengaduan untuk admin dan *user*. Di dalam halaman riwayat pengaduan ini terdapat list pengaduan yang telah diajukan *user*, beserta status dari aduannya. Halaman riwayat pengaduan ini sama pada *user* maupun admin.

g) Halaman Detail Pengaduan User



Gambar 17. Halaman Detail Pengaduan User
Berikut pada gambar diatas merupakan tampilan dari halaman detail pada pengaduan untuk *user*. setelah memilih pengaduan yang ingin dilihat, pada tampilan ini memperlihatkan informasi yang diinginkan *user* seperti, judul pengaduan, deskripsi pengaduan, gambar yang dimaksud dalam pengaduan beserta status pengaduannya.

h) Halaman Detail Pengaduan Admin



Gambar 17. Halaman Detail Pengaduan Admin

Pada gambar diatas menampilkan halaman detail pengaduan untuk admin. Di mana pada halaman ini terdapat detail masalah yang diadakan *user*. Selain itu tugas admin juga memilah pengaduan mana saja

yang dapat ditangani dengan cara menekan tombol 'selesai' atau 'tolak'.

(1)

2.3. Pengujian

Pada tahap ini, peneliti menggunakan 2 bentuk testing yaitu, Alpha testing dan Beta testing, Berikut Penjabaran dan hasil dari kedua testing:

a) Alpha Testing

Pengujian alpha merupakan pengujian perangkat lunak yang dilaksanakan dengan tujuan mengetahui kinerja dari aplikasi yang telah dibuat. Pengujian alpha saat pengujian aplikasi ini menggunakan teknik black box. Berikut ini hasil pengujian alpha testing pada aplikasi pengaduan.

Tabel 1. Black Box Register User

Input	Planning	Response System	Result
Hasil uji normal			
Memasukkan semua data dengan benar	Data akan disimpan lalu user akan dialihkan ke halaman home	Data akan disimpan lalu user akan dialihkan ke halaman home	[✓] Terima [] Tolak
Hasil uji tidak normal			
Hanya memasukkan beberapa data dan ada data yang tidak sesuai	Menampilkan pesan terdapat data yang tidak sesuai	Menampilkan pesan terdapat data yang tidak sesuai	[✓] Terima [] Tolak

Tabel 2. Black Box Login User

Input	Planning	Response System	Result
Hasil uji normal			
Email dan password yang dimasukkan sesuai	Masuk ke halaman home	Masuk ke halaman home	[✓] Terima [] Tolak
Hasil uji tidak normal			
Email yang dimasukkan	Menampilkan peringatan	Menampilkan peringatan	[✓] Terima

an tidak sesuai	email tidak sesuai	email tidak sesuai	[] Tolak
Password yang dimasukkan tidak sesuai	Menampilkan peringatan password tidak sesuai	Menampilkan peringatan password tidak sesuai	[√] Terima [] Tolak

Tabel 3. Black Box Login Admin

Input	Planning	Response System	Result
Hasil uji normal			
Email dan password yang dimasukkan sesuai	Masuk ke halaman home	Masuk ke halaman home	[√] Terima [] Tolak
Hasil uji tidak normal			
Email yang dimasukkan tidak sesuai	Menampilkan peringatan email tidak sesuai	Menampilkan peringatan email tidak sesuai	[√] Terima [] Tolak

Tabel 4 Black Box Membuat Pengaduan

Input	Planning	Response System	Result
Hasil uji normal			
Mengisi judul, deskripsi, alamat, dan foto bukti	Muncul notifikasi bahwa pengaduan berhasil dibuat dan dialihkan ke halaman home	Muncul notifikasi bahwa pengaduan berhasil dibuat dan dialihkan ke halaman home	[√] Terima [] Tolak
Hasil uji tidak normal			
Tidak mengisi judul, deskripsi, alamat, dan foto bukti	Muncul peringatan bahwa ada data yang masih kosong	Muncul peringatan bahwa ada data yang masih kosong	[√] Terima [] Tolak
Terjadi masalah koneksi saat	Muncul peringatan bahwa pengaduan	Muncul peringatan bahwa pengaduan	[√] Terima [] Tolak

request ke database.	gagal dibuat	gagal dibuat	
----------------------	--------------	--------------	--

Tabel 5. Black Box Melihat Pengaduan Terbaru (User)

Input	Planning	Response System	Result
Hasil uji normal			
Membuka halaman home	Menampilkan 5 data pengaduan berdasarkan pengaduan terbaru	Menampilkan 5 data pengaduan berdasarkan pengaduan terbaru	[√] Terima [] Tolak
Hasil uji tidak normal			
Membuka halaman home	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	[√] Terima [] Tolak
Terjadi masalah koneksi saat request ke database.	Muncul peringatan bahwa pengaduan gagal dimuat	Muncul peringatan bahwa pengaduan gagal dimuat	[√] Terima [] Tolak

Tabel 6. Black Box Melihat Seluruh Pengaduan (User)

Input	Planning	Response System	Result
Hasil uji normal			
Membuka halaman pengaduan	Menampilkan seluruh data pengaduan berdasarkan pengaduan terbaru	Menampilkan seluruh data pengaduan berdasarkan pengaduan terbaru	[√] Terima [] Tolak
Hasil uji tidak normal			

Membuka halaman pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	[✓] Terima [] Tolak
Terjadi masalah koneksi saat request ke database.	Muncul peringatan bahwa pengaduan gagal dimuat	Muncul peringatan bahwa pengaduan gagal dimuat	[✓] Terima [] Tolak

Tabel 7. Black Box Melihat Notifikasi Berdasarkan User

Input	Planning	Response System	Result
Hasil uji normal			
Membuka halaman notifikasi	Menampilkan seluruh notifikasi berdasarkan notifikasi terbaru	Menampilkan seluruh notifikasi berdasarkan notifikasi terbaru	[✓] Terima [] Tolak
Hasil uji tidak normal			
Membuka halaman notifikasi	Muncul peringatan bahwa data tidak ditemukan karena belum ada notifikasi yang masuk	Muncul peringatan bahwa data tidak ditemukan karena belum ada notifikasi yang masuk	[✓] Terima [] Tolak
Terjadi masalah koneksi saat request ke database.	Muncul peringatan bahwa notifikasi gagal dimuat	Muncul peringatan bahwa notifikasi gagal dimuat	[✓] Terima [] Tolak

Tabel 8. Black Box Melihat Pengaduan Terbaru (Admin)

Input	Planning	Response System	Result
Hasil uji normal			

Membuka halaman home	Menampilkan 5 data pengaduan berdasarkan pengaduan terbaru	Menampilkan 5 data pengaduan berdasarkan pengaduan terbaru	[✓] Terima [] Tolak
Hasil uji tidak normal			
Membuka halaman home	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	[✓] Terima [] Tolak
Terjadi masalah koneksi saat request ke database.	Muncul peringatan bahwa pengaduan gagal dimuat	Muncul peringatan bahwa pengaduan gagal dimuat	[✓] Terima [] Tolak

Tabel 9. Black Box Melihat Seluruh Pengaduan (Admin)

Input	Planning	Response System	Result
Hasil uji normal			
Membuka halaman pengaduan	Menampilkan seluruh data pengaduan berdasarkan pengaduan terbaru	Menampilkan seluruh data pengaduan berdasarkan pengaduan terbaru	[✓] Terima [] Tolak
Hasil uji tidak normal			
Membuka halaman pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	Muncul peringatan bahwa data tidak ditemukan karena belum ada yang membuat pengaduan	[✓] Terima [] Tolak
Terjadi masalah koneksi	Muncul peringatan bahwa	Muncul peringatan bahwa	[✓] Terima [] Tolak

saat request ke database.	pengaduan gagal dimuat	pengaduan gagal dimuat	
---------------------------	------------------------	------------------------	--

Tabel 10. Black Box Mengubah Status Pengaduan (Admin)

Input	Planning	Response System	Result
Hasil uji normal			
Membuka halaman detail pengaduan	Mengubah status pengaduan menjadi tolak atau selesai	Mengubah status pengaduan menjadi tolak atau selesai	[√] Terima [] Tolak
Hasil uji tidak normal			
Terjadi masalah koneksi saat request ke database.	Muncul peringatan bahwa gagal mengubah status pengaduan	Muncul peringatan bahwa gagal mengubah status pengaduan	[√] Terima [] Tolak

b) Beta Testing

Pengujian beta testing pada aplikasi pengaduan masyarakat akan dilakukan oleh pengguna secara langsung. Adapun media yang digunakan untuk memberikan penilaian yaitu kuesioner. Hasil dari kuesioner tersebut nantinya akan dihitung untuk mengetahui kesimpulan dari kinerja yang telah dilakukan aplikasi ini. Kuesioner ini terdapat 6 pertanyaan. Jawaban dari pertanyaan tersebut berupa skala likert yang berskala 1 sampai 5. Berikut ini adalah skor penilaian yang diberikan masyarakat dengan menggunakan skala likert untuk setiap pertanyaan kuesioner.

Tabel 11. Skala Likert

Tingkat Kepuasan	Skala
Sangat Setuju	5
Setuju	4
Cukup Setuju	3
Kurang Setuju	2
Sangat Tidak Setuju	1

Tabel 12. Bobot Nilai

No.	Persentase Penilaian	Interpentasi
1.	81% - 100%	Sangat Setuju
2.	61% - 80%	Setuju
3.	41% - 60%	Cukup Setuju
4.	21% - 40%	Kurang Setuju
5.	0% - 20%	Sangat Tidak Setuju

Persentasi dari hasil data kuisisioner akan dihitung dengan menggunakan rumus:

$$Y = \frac{\Sigma(N.R)}{Skor\ ideal} \times 100\%$$

Responden terdiri dari 30 orang yang berasal dari berbagai golongan untuk mendapatkan review yang baik dari responden. Berdasarkan perhitungan pada tabel 4.37, tiap responden memiliki nilai skor maksimal 30 dan jika di total keseluruhan menjadi 900. Selanjutnya, jumlah nilai kategori jawaban mendapatkan skor 823. Skor tersebut lalu dimasukkan ke dalam rumus. Berikut adalah hasil perhitungan dari skor yang sudah didapat.

Tabel 13. Hasil Survei

R	Nomor Pertanyaan						Ttl	Maks
	1	2	3	4	5	6		
R1	5	5	5	4	5	5	29	30
R2	5	5	5	5	5	5	30	30
R3	4	5	4	4	4	5	26	30
R4	4	5	4	4	4	4	25	30
R5	5	5	5	5	5	5	30	30
R6	5	5	5	5	5	4	29	30
R7	4	5	5	5	5	4	28	30
R8	4	4	4	4	4	5	25	30
R9	4	5	4	5	4	5	27	30
R10	5	5	5	5	5	5	30	30
R11	4	5	4	5	4	5	27	30
R12	4	5	5	4	5	4	27	30
R13	4	5	5	4	5	4	27	30
R14	5	5	5	5	5	5	30	30
R15	4	4	4	5	5	4	26	30
R16	4	5	5	4	4	5	27	30
R17	4	4	4	4	4	4	24	30
R18	4	4	5	5	4	4	26	30
R19	4	4	5	5	4	5	27	30
R20	4	5	5	5	5	5	29	30
R21	4	4	5	5	4	4	26	30
R22	5	4	4	4	4	5	26	30
R23	4	4	4	4	4	5	25	30
R24	4	5	5	5	5	5	29	30
R25	5	5	5	4	4	4	27	30
R26	4	5	5	5	4	5	28	30
R27	5	4	4	5	5	5	28	30
R28	4	5	5	5	4	5	28	30
R29	5	5	4	5	5	5	29	30
R30	4	5	5	5	4	5	28	30
Jumlah							823	900

Berdasarkan perhitungan pada tabel 4.37, tiap responden memiliki nilai skor maksimal 30 dan jika di total keseluruhan menjadi 900. Selanjutnya, jumlah nilai kategori jawaban mendapatkan skor 823. Skor tersebut lalu dimasukkan ke dalam rumus. Berikut adalah hasil perhitungan dari skor yang sudah didapat:

$$Y = \frac{\Sigma(N.R)}{Skor\ ideal} \times 100\%$$

$$Y = \frac{823}{900} \times 100\% = 91.44\%$$

Hasil perhitungan persentase adalah 91.44%. Hasil tersebut dapat disimpulkan bahwa aplikasi pengaduan masyarakat dapat melakukan tugasnya dengan baik dan dapat memenuhi kebutuhan penggunaannya.

3. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan yaitu Penerapan Pola Arsitektur MVVM pada Perancangan Aplikasi Pengaduan Masyarakat Berbasis Android, maka didapatkan kesimpulan sebagai berikut:

1. Penelitian ini menghasilkan aplikasi pengaduan masyarakat berbasis Android dengan menggunakan pola arsitektur MVVM. Dengan menerapkan pola arsitektur ini dapat mempermudah untuk melakukan pemeliharaan kode yang telah ada.
2. Pengujian *beta testing* melibatkan 30 responden dan menghasilkan nilai 91,44% yang dapat disimpulkan bahwa aplikasi ini cukup membantu masyarakat dalam melakukan pengaduan.

PUSTAKA

- Akbar Ritonga, Ibnu Rasyid Munthe, Masrizal, A. (2019). Jurnal Mantik Jurnal Mantik. *Mobile-Based National University Online Library Application Design*, 3(2), 10–19. <http://iocscience.org/ejournal/index.php/mantik/article/view/882/595>
- Bhavsar, K., Shah, D. V., & Gopalan, D. S. (2020). Scrum: An Agile Process Reengineering In Software Engineering. *International Journal of Innovative Technology and Exploring Engineering*, 9(3), 840–848. <https://doi.org/10.35940/ijitee.c8545.019320>
- Heryandi, A. (2018). Developing Application ProgramMing Interface (API) for Student Academic Activity Monitoring using Firebase Cloud Messaging (FCM). *IOP Conference Series: Materials Science and Engineering*, 407(1). <https://doi.org/10.1088/1757-899X/407/1/012149>
- Ibrahim, A. F., Musa, N., & Jamaludin, M. N. F. (2018). Internship Application System (IAS) for University Students using Laravel. *Journal of Computing Research and Innovation*, 3(4), 12–18. <https://doi.org/10.24191/jcrinn.v3i4.85>
- Joni Karman, A. T. M. (2018). Sistem Informasi Geografis Lokasi Pemetaan Masjid Berbasis Android Kota Lubuklinggau. *LIS Liga Ilmu Serantau*, 42–50. <https://doi.org/https://doi.org/10.31227/osf.io/tmk4g>
- Khawas, C., & Shah, P. (2018). Application of Firebase in Android App Development-A Study. *International Journal of Computer Applications*, 179(46), 49–53. <https://doi.org/10.5120/ijca2018917200>
- Lou, T. (2016). *A comparison of Android Native App Architecture Master ' s Programme in ICT Innovation A Comparison of Android Native App Architecture – MVC , MVP and MVVM*.
- Manuputty, A. D., Hendrawan, S., & Haryanto, B. (2020). Design of Information Systems for Research Permit Application with Agile Method and Website Based Laravel Framework. *Journal of Information Systems and Informatics*, 2(1), 60–78. <https://doi.org/10.33557/journalisi.v2i1.45>
- Martin, R. C. (2017). Clean Architecture: A Craftsman's Guide to Software Structure and Design. In *Prentice Hall*. Prentice Hall, 2018. <https://www.amazon.com/Clean-Architecture-Craftsmans-Software-Structure/dp/0134494164%0Ahttps://www.safaribooksonline.com/library/view/clean-architecture-a/9780134494272/>
- Rais, M. (2019). Penerapan Konsep Object Oriented Programming Untuk Aplikasi Pembuat Surat. *PROtek : Jurnal Ilmiah Teknik Elektro*, 6(2), 96–101. <https://doi.org/10.33387/protk.v6i2.1242>
- Ricu Sidiq, & Najuah. (2020). Pengembangan E-Modul Interaktif Berbasis Android pada Mata Kuliah Strategi Belajar Mengajar. *Jurnal Pendidikan Sejarah*, 9(1), 1–14. <https://doi.org/10.21009/jps.091.01>
- Schwaber, K., & Sutherland, J. (2011). The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game. In *Scrum. org, October* (Vol. 2, Nomor October). <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>
- Sibarani, N. S., Munawar, G., & Wisnuadhi, B. (2018). Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Kotlin. In *Prosiding Industrial Research Workshop and National Seminar. Industrial Research Workshop and National Seminar*, 9(July). <https://doi.org/https://doi.org/10.35313/irwns.v9i0.1116>
- Siregar, H. F., Siregar, Y. H., & Melani, M. (2018). (2018). Perancangan Aplikasi Komik Hadist Berbasis Multimedia. *JurTI (Jurnal Teknologi Informasi)*, 2(2), 113-121. *JurTI (Jurnal Teknologi Informasi)*, 2(2), 113–121. <http://www.jurnal.una.ac.id/index.php/jurtti/article/view/425>
- Sutabri. (2020). Bab II Tinjauan Pustaka Dan Dasar Teori. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699. <https://eprints.utdi.ac.id/8897/>
- Suzanti, I. O., Fitriani, N., Jauhari, A., & Khozaimi, A. (2020). REST API Implementation on Android Based Monitoring Application. *Journal of Physics: Conference Series*, 1569(2). <https://doi.org/10.1088/1742-6596/1569/2/022088>
- Widharma, I. G. S. (2017). Perancangan Simulasi Sistem Pendaftaran Kursus Berbasis Web

- Dengan Metode Sdlc. *Matrix : Jurnal Manajemen Teknologi dan Informatika*, 7(2), 38. <https://doi.org/10.31940/matrix.v7i2.527>
- Wobkenberg, R. D., & Llp, E. (2020). (12) *United States Patent*. 2.
- Wulandari, D., & Gusti, K. (2020). Perguruan Tinggi Berbasis Web Dengan Metode Agile Scrum Studi Kasus Upn “ Veteran ” Jakarta Design-Build of Web Based Higher Education Quality Assurance System Application With Agile Scrum Method Study At Upn “ Veteran ” Jakarta. *Sigma-Mu*, 12, 14–29.