

Rancang Bangun Digitalisasi Formulir Perintah Kerja Berbasis *Website* dan Aplikasi *Android* dengan Metode *Unified Modelling Language* (UML)

Zefanya Ardhiyanti Purna Murti¹⁾
Teknik Mesin, Politeknik Gajah Tunggal
aguerack@gmail.com

Anis Choirunnisa²⁾
Teknologi Informasi, Politeknik Gajah Tunggal
anis@poltek-gt.ac.id

Donny Muda Priyangan³⁾
Teknologi Informasi, Politeknik Gajah Tunggal
donny@poltek-gt.ac.id

ABSTRAK

Penelitian ini bertujuan untuk merancang dan membangun sistem digitalisasi FPK berbasis *website* dan aplikasi *android* pada *Section Engineering*. Sistem ini dirancang sebagai solusi untuk mengatasi kendala pada sistem manual yang selama ini digunakan, seperti keterlambatan proses, risiko kehilangan data, dan inefisiensi biaya serta waktu. Metodologi yang digunakan dalam penelitian ini adalah metode *Agile* untuk pengembangan *software* dan *Unified Modelling Language* (UML) untuk perancangan sistem, meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram* untuk menggambarkan struktur, alur kerja, dan interaksi antar pengguna secara komprehensif. Pengumpulan data dilakukan melalui observasi langsung, studi literatur, dan analisa data historis seperti jumlah penggunaan FPK dan waktu input manual. Hasil dari penelitian ini adalah sistem digitalisasi FPK yang melibatkan empat aktor utama, yaitu Admin, *Section Head*, Engineer, dan Tenant. Sistem ini memiliki fitur utama yang mencakup *Form Complain*, *Work Order*, *Approval*, dan *Survey Satisfaction*. Setiap pengguna memiliki akses dan fungsi yang berbeda sesuai dengan perannya masing-masing. Kesimpulan dari penelitian ini menunjukkan bahwa sistem digitalisasi FPK mampu meningkatkan efisiensi kerja, mengurangi *human error*, menekan biaya operasional, dan menyediakan arsip digital yang lebih aman serta mudah diakses.

Kata Kunci: FPK, UML, Aplikasi, *Website*, *Android*

ABSTRACT

This study aims to design and build a website-based FPK digitalization system and android application in the Engineering Section. This system is designed as a solution to overcome the constraints of the manual system that has been used so far, such as process delays, risk of data loss, and cost and time inefficiencies. The methodology used in this study is the Agile method for software development and the Unified Modeling Language (UML) for system design, including Use Case Diagrams, Activity Diagrams, Sequence Diagrams, and Class Diagrams to describe the structure, workflow, and interactions between users comprehensively. Data collection was carried out through direct observation, literature studies, and historical data analysis such as the number of FPK usage and manual input time. The result of this study is an FPK digitalization system that involves four main actors, namely Admin, Section Head, Engineer, and Tenant. This system has main features that include Complain Form, Work Order, Approval, and Satisfaction Survey. Each user has different access and functions according to their respective roles. The conclusion of this study shows that the FPK digitalization system is able to increase work efficiency, reduce human error, reduce operational costs, and provide digital archives that are safer and easier to access.

Key Words: FPK, UML, Application, Website, Android

I. PENDAHULUAN

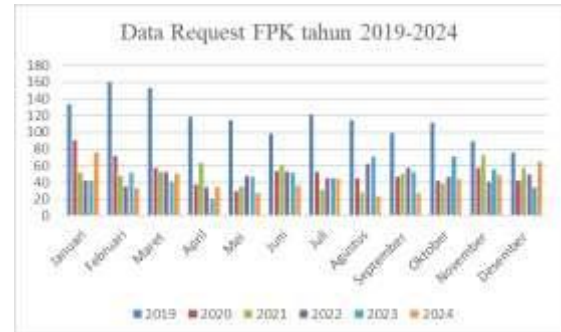
Semakin berkembangnya zaman, inovasi teknologi termasuk dalam bidang manajemen properti dan fasilitas, terus mengalami kemajuan yang pesat. Transformasi digital telah menjadi faktor penting dalam meningkatkan efisiensi dan kualitas layanan. Hubungan antara manajemen gedung dan penyewa menjadi kunci utama dalam operasionalisasi dan kelangsungan kerja sama dalam model bisnis penyewaan gedung. Hal ini melibatkan komunikasi yang efektif, penanganan keluhan secara tepat waktu, dan pemeliharaan lingkungan yang nyaman[1]. Namun hingga kini, masih banyak proses pelaporan dan penanganan keluhan (*service request*) yang mencakup perbaikan fasilitas, kebersihan, maupun keluhan operasional lainnya yang berjalan secara manual, salah satunya di departement *Building Management* divisi *Engineering*. Salah satu sistem yang masih berjalan secara manual yakni Formulir Perintah Kerja (FPK) yang terdapat pada proses kerja perawatan dan perbaikan alat serta gedung. Ketergantungan pada sistem manual tersebut mengakibatkan sejumlah kendala seperti keterlambatan, kehilangan informasi, serta kurangnya transparansi dalam proses penanganan keluhan. Selain itu, proses manual tersebut dinilai kurang efektif karena tidak efisien untuk proses perbaikan karena bersifat *urgent*. Kekurangan lain dari sistem manual antara lain yakni kurangnya efisien karena sistem manual seringkali memakan banyak waktu dibandingkan sistem yang telah berjalan secara otomatis. Rekapitulasi data secara manual rentan terhadap kesalahan (*human error*) yang berdampak pada tingkat akurasi informasi, juga mempersulit pekerja dalam proses pengolahan data.



Gambar 1. Kondisi Penyimpanan Berkas

Seperti yang terlihat pada Gambar 1, penyimpanan data fisik juga memiliki banyak kekurangan lain yakni diantaranya membutuhkan ruang penyimpanan yang relatif besar, resiko berkas

hilang bahkan berkas rusak. Banyaknya data yang disimpan secara manual juga mempersulit pekerja dalam mengakses berkas, bahkan terkadang sulit ditemukan saat data sedang dibutuhkan. Terdapat juga data dari penelitian yang telah kami lakukan, sebagai berikut.



Gambar 2. Data Kebutuhan Kertas FPK per Tahun

Data diatas menunjukkan banyaknya penggunaan kertas dalam 6 tahun (2019-2024) yang berjumlah total sebanyak 16.393 FPK, dimana pada 1 FPK memiliki 3 rangkap kertas yakni berwarna putih yang ditujukan untuk arsip data *Engineering*, berwarna biru ditujukan untuk TRO/Sec/BM, dan kertas berwarna pink yang ditujukan untuk *Tenant* atau pelapor. Data diatas menghasilkan rata-rata sebanyak 2.732 FPK per tahun. Pada satu buku FPK terdapat 50 rangkap FPK berjumlah 150 lembar kertas. Harga 1 buku FPK bernilai 25.000 rupiah.

Tabel 1. Harga Kertas per FPK

Komponen	Satuan	QTY	Biaya	
			Satuan	Total
Harga Buku				
FPK (150 lembar)	Buku	19	Rp 25,000	Rp 475,000
Kertas per FPK (3 rangkap)	Request	2732	Rp 500	Rp 1,366,000
Total Biaya Manual per Tahun				Rp 1,841,000

Tabel 2. Data Waktu Input FPK Manual

No	Waktu Proses	No	Waktu Proses
1	504	16	503
2	525	17	543
3	470	18	504
4	518	19	482
5	472	20	500
6	554	21	490
7	480	22	468
8	484	23	475
9	535	24	512
10	511	25	517

Tabel 2. Lanjutan

No	Waktu Proses	No	Waktu Proses
11	524	26	519
12	474	27	512
13	495	28	474
14	471	29	503
15	465	30	550
Rata-rata	501.13	Rata-rata	501.13

Dari 30 sampel yang telah didapat, menghasilkan rata rata waktu dalam satu kali penginputan FPK secara manual, yakni selama 501,13 detik. Sehingga kami juga mencoba untuk menganalisa permasalahan di atas menggunakan metode *Fishbone Diagram*:



Gambar 3. *Fishbone Diagram*

Berdasarkan analisa yang didapat dari *Fishbone Diagram*, penelitian ini merujuk pada permasalahan yakni tidak adanya sistem digital untuk formulir perintah kerja sehingga seluruh proses kerja masih dilakukan secara manual.

Dengan kemajuan teknologi, pembuatan sistem digital dalam bidang industri telah menjadi solusi inovatif untuk masalah yang kita temui di kantor. Salah satu solusi yang digunakan adalah sistem *digital* untuk mempermudah proses kerja menjadi lebih efisien. Sistem ini dapat mengurangi waktu kerja menjadi lebih singkat dan mengubah sistem manual menjadi otomatis sehingga mempermudah para pekerja untuk berkomunikasi sehingga proses menjadi lebih efisien, dari segi waktu ataupun tenaga.

Berdasarkan hal tersebut, penelitian yang kami jalankan bertujuan untuk merancang dan membangun sistem digital untuk Formulir Perintah Kerja yang dapat berfungsi secara otomatis agar dapat mempermudah para pekerja dalam menjalankan proses kerja, meningkatkan efisiensi

serta mengurangi resiko kesalahan yang mungkin muncul.

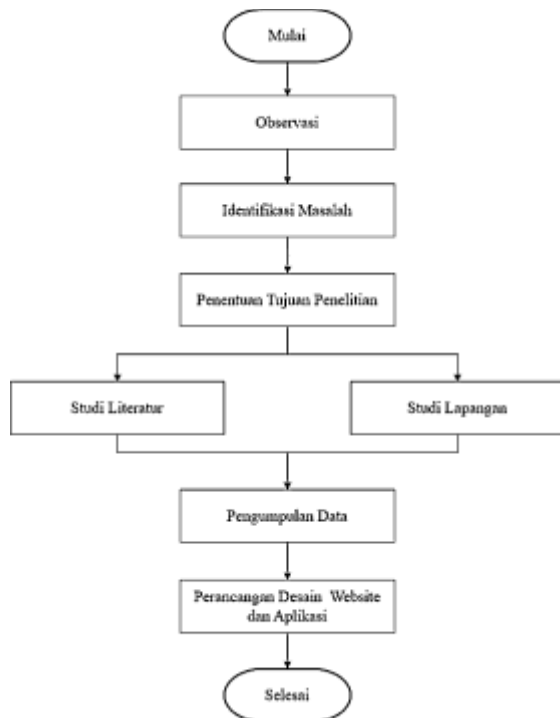
Perancangan sistem digital menggunakan metode UML dapat membantu dalam menggambarkan dan merancang struktur serta fungsi sistem secara menyeluruh. UML adalah bahasa yang digunakan untuk menspesifikasikan, memvisualisasikan, dan membuat struktur dasar sistem perangkat lunak. Ini juga memungkinkan pemodelan aturan bisnis [2]. UML terdiri dari beberapa diagram yang digunakan untuk merancang atau mengembangkan perangkat lunak berbasis objek, antara lain *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

Use Case Diagram merupakan visualisasi dari suatu sistem yang terdiri dari beberapa komponen, seperti *Actor*, *use case*, dan hubungan antar komponen [3]. *Activity Diagram* merupakan diagram yang digunakan untuk memvisualisasikan proses paralel yang terjadi ketika sistem dijalankan [4]. *Sequence Diagram* membantu dalam memahami persyaratan sistem baru, mendokumentasikan proses, dan memvisualisasikan skenario teknis saat sistem berjalan, memungkinkan pengguna memprediksi perilaku sistem [5]. Sedangkan *Class Diagram* menggambarkan struktur dan hubungan antar *class*, *package*, dan *objek*, termasuk pewarisan dan asosiasi [6].

II. METODE PENELITIAN

A. Alur Penelitian

Alur penelitian ini dilakukan untuk merancang *Website* dan Aplikasi berbasis *Android* Formulir Perintah Kerja (FPK) yang dapat dilihat pada gambar berikut :



Gambar 4. Flowchart Alur Penelitian

1) Observasi

Observasi merupakan langkah awal dalam melakukan penelitian dengan cara mengamati proses secara langsung kondisi lapangan untuk mencari informasi terkait fenomena yang ingin diteliti dan dianalisa. pada tahapan ini, observasi dilakukan pada *Section Engineering* dengan mengamati proses penerimaan komplain dari *Tenant* / pelapor, pengisian permintaan pada *form* perintah kerja yang dilanjutkan dengan penanganan komplain pelapor yang dilakukan oleh teknisi, proses pengisian dan penyerahan FPK oleh *Engineer* kepada Admin.

2) Identifikasi Masalah

Peneliti mengidentifikasi masalah lapangan terlebih dahulu sebelum melakukan penelitian. Hal ini dilakukan dengan mengamati bagaimana proses perintah kerja berlangsung, mulai dari penerimaan keluhan dari *Tenant*, pengisian FPK pada bagian permintaan oleh Admin, penanganan keluhan oleh *Engineer*, pengisian FPK oleh *Engineer*, dan kembalinya FPK ke Admin sebagai arsip data. Setelah mengidentifikasi masalah ditemukan bahwa masih terdapat banyak sistem yang berjalan secara manual dan menggunakan kertas, salah satunya Formulir Perintah Kerja (FPK). Hal tersebut menyebabkan penumpukan *form* pada ruang penyimpanan. Penumpukan ini juga menimbulkan

resiko hilangnya berkas atau bahkan rusak serta sulitnya proses pencarian data.

3) Penentuan Tujuan Penelitian

Berdasarkan observasi dan identifikasi masalah yang dilakukan oleh penulis, penelitian ini bertujuan untuk mempermudah proses perintah kerja dengan merancang bangun digitalisasi Formulir Perintah Kerja (FPK) berbasis *Website* dan aplikasi.

4) Studi Literatur dan Studi Lapangan

Studi literatur adalah metode pengumpulan data dan informasi serta sumber pendukungnya, seperti buku, catatan, hasil penelitian sebelumnya, dan artikel-artikel ilmiah [7]. Studi literatur dilakukan oleh penulis untuk memperkuat kajian serta memperkaya wawasan sehingga memudahkan penulis dalam melakukan penelitian. Sedangkan studi lapangan merupakan metode untuk mendapatkan data dan informasi dengan melakukan observasi dan wawancara secara langsung [8]. Studi lapangan bertujuan untuk mengetahui sistem dan cara kerja formulir perintah kerja, serta memahami kondisi yang sebenarnya terjadi.

5) Pengumpulan Data

Pengumpulan data merupakan teknik atau cara yang digunakan oleh penulis untuk memperoleh informasi yang dibutuhkan dalam penelitian [9]. Pada tahap ini, penulis melakukan pengumpulan data yang diperlukan dalam penelitian. Adapun data yang dikumpulkan mencakup data waktu penginputan FPK ke dalam *excel* oleh Admin, data jumlah FPK setiap bulan dalam kurun waktu 5 tahun ke belakang, data karyawan pada *Section Engineering*, data daftar *Tenant* di *Head Office*, dan data *cost* yang dikeluarkan untuk penggunaan kertas FPK.

6) Perancangan Desain Website dan Aplikasi

Penulis membuat perancangan desain *Website* dan aplikasi untuk menentukan *layout* tombol, *interface*, dan fitur yang akan dibuat. Perancangan dilakukan dengan menggunakan metode *Unified Modelling Language* (UML). UML digunakan untuk memvisualisasikan suatu sistem agar mempermudah dalam proses perancangan. Jenis UML yang digunakan pada perancangan desain ini diantaranya *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

III. HASIL DAN PEMBAHASAN

A. Perancangan UML Aplikasi Android dan Website

Ada empat aktor utama yang bertanggung jawab atas *Website* dan aplikasi Formulir Perintah Kerja (FPK) yaitu Admin, *Section Head*, *Engineer*, dan *Tenant*. Sesuai dengan peran dan tugas mereka dalam sistem, masing-masing aktor memiliki menu dan fungsi yang berbeda. Admin memiliki kontrol penuh atas sistem, termasuk sistem dan pengelolaan data, serta pengaturan pengguna. *Tenant* bertanggung jawab untuk membuat dan mengajukan *Form Complain*, memantau status pengajuan mereka, serta memberikan umpan balik terhadap pekerjaan yang telah selesai dikerjakan. Setelah *Form Complain* disetujui oleh Admin, *Engineer* bertanggung jawab untuk menyelesaikan pekerjaan yang tercantum dalam komplain. Sementara itu, *Section Head* dan Admin memiliki peran penting dalam proses persetujuan perintah kerja. Masing-masing harus memberikan persetujuan sesuai dengan wewenang dan tanggung jawab mereka.

Dengan demikian, setiap aktor dapat mengakses menu dan fungsi yang sesuai dengan peran mereka, yang memungkinkan mereka untuk mengelola perintah kerja secara efisien dan efektif. Diagram-diagram berikut ini akan memberikan gambaran lebih lanjut mengenai menu dan fungsi yang tersedia untuk setiap aktor dalam *website* dan aplikasi FPK.

1) Use Case Diagram



Gambar 5. Use Case Diagram All User

Use Case Diagram diatas merupakan gambaran mengenai sistem Formulir Perintah Kerja (FPK) yang dirancang untuk memfasilitasi interaksi

antara *Tenant*, Admin, *Section Head*, dan *Engineer* dalam menangani keluhan, perbaikan, dan evaluasi layanan. Diagram tersebut menunjukkan bahwa sistem yang dibuat memiliki beberapa fitur utama, seperti *Form Complain*, *Work Order*, *Approval*, dan *Survey Satisfaction*. Secara keseluruhan, *Use Case Diagram* ini digunakan sebagai dasar perancangan *website* dan aplikasi yang dibuat.

Tabel 3. Hak Akses Sistem

	HAK AKSES ALL USER			
	ADMIN	SECTION HEAD	ENGINEER	TENANT
LOGIN	Read data	Read data	Read data	Read data
DASHBOARD	Read data	Read data	Read data	Read data
PROFILE USER	Read data, Insert Data, Delete data	Read data	-	-
FORM COMPLAIN	Insert data, Update data, Read data	Read data	-	Insert data, Read data
WORK ORDER	Insert data, Update data, Read data	Read data	Insert data, Read data	-
APPROVAL	Insert data, Read data	Insert data, Read data	-	-
SURVEY SATISFACTION	Insert data, Read data	Read data	-	Insert data, Read data
INSTRUCTION	-	-	-	-
LOGOUT	-	-	-	-

Tabel 3 merupakan tabel hak akses sistem yang menunjukkan fitur yang diizinkan untuk diakses oleh setiap pengguna (*Admin*, *Section Head*, *Engineer*, dan *Tenant*). Beberapa jenis akses meliputi *Insert* (menambahkan data), *Update* (memperbarui data), *Read* (membaca data), dan *Delete* (menghapus data). Setiap pengguna hanya dapat mengakses fitur sesuai peran dan tanggung jawabnya masing-masing, oleh karena itu dirancang tabel ini untuk mendukung pengendalian akses berbasis peran (*Role-Based Access Control/RBAC*) [10].

Tabel 4. Izin Akses Pengguna

No	Pengguna	Akses	
		Website	Aplikasi Android
1	Admin	√	
2	Section Head	√	
3	Engineer	√	
4	Tenant		√

Aplikasi yang dikembangkan terdiri dari dua platform, yaitu Website dan Aplikasi Android, yang digunakan sesuai dengan peran masing-masing pengguna. Tabel 4 merupakan izin akses pengguna terhadap media sistem. Admin, Section Head, dan Engineer dapat mengakses Website untuk mengelola, menyetujui, dan menindaklanjuti Formulir Perintah Kerja (FPK). Sedangkan Tenant menggunakan aplikasi Android untuk mengajukan permintaan kerja (komplain) dan memantau status penanganan.

Berikut merupakan penjelasan mengenai pihak-pihak yang berinteraksi dalam sistem beserta fungsi dari masing-masing fitur yang tersedia.

a) *Login*

Semua pengguna melakukan *Login* ke sistem dengan memasukkan kredensial yang valid (seperti *Username* dan *Password*). Setelah berhasil *Login*, pengguna diarahkan ke *Dashboard*.

b) *Dashboard*

Setelah *Login*, pengguna diarahkan ke *Dashboard* yang menampilkan informasi utama dan menu navigasi. *Dashboard* ini menampilkan jumlah *form* dari masing-masing fitur.

c) *Profile User*

Halaman ini menampilkan semua *User* yang terdaftar baik melalui *Website* ataupun Aplikasi *Android*. Halaman ini memungkinkan user untuk melihat detail informasi pribadi, seperti nama, departemen, *tenancy*, dan lantai. Admin dapat melihat, menambahkan, dan menghapus data pribadi semua user. *Section Head* hanya memiliki akses untuk melihat detail informasi pribadi semua user. Sedangkan *Engineer* dan *Tenant* hanya dapat melihat data pribadinya saja.

d) *Form Complain*

Selain *Tenant* yang dapat melakukan penginputan pada halaman *Form Complain*, Admin juga mendapatkan akses ke *Form Complain* dengan tujuan untuk melakukan penginputan bagi *Tenant* yang mengajukan keluhan melalui telepon, pesan, maupun datang langsung ke ruang *Engineering*. Selain itu, *Section* hanya dapat melihat *Form Complain* yang diajukan oleh *Tenant* maupun Admin.

e) *Survey Satisfaction*

Admin dan *Section Head* dapat melihat hasil rating atau penilaian dari *Tenant* berupa *survey* kepuasan setelah pekerjaan telah di selesaikan oleh *Engineer*. *Survey Satisfaction* merupakan akhir dari proses FPK itu sendiri yang berarti status komplain tersebut sudah *closed* atau tutup.

f) *Work Order*

Halaman ini merupakan halaman penginputan dari hasil pekerjaan yang dilakukan oleh *Engineer* yang berisi nama yang mengerjakan, tindakan penanggulangan, bukti foto ketika proses pengerjaan, dan bukti foto ketika telah selesai mengerjakan, dengan tujuan untuk melanjutkan FPK ke proses *approve* dari *Section Head*.

g) *Approval*

Approval merupakan proses persetujuan yang diajukan oleh *Engineer* kepada Admin dan

Section Head dan sekaligus melaporkan bahwa pekerjaan dari komplain yang dilakukan *Tenant* telah selesai di kerjakan dan membutuhkan *approve* untuk menyatakan bahwa komplain tersebut telah selesai dikerjakan.

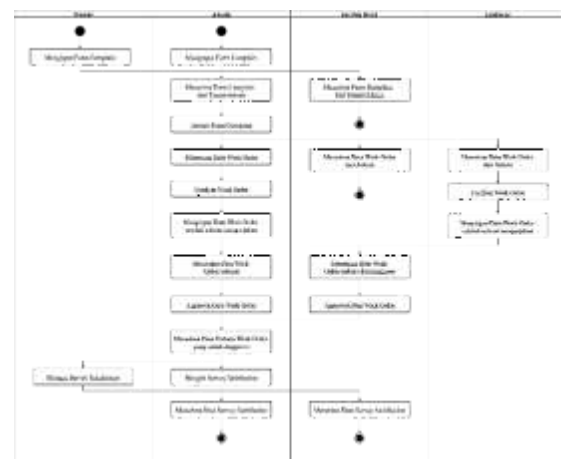
h) *Instruction*

Halaman ini merupakan panduan bagi para pengguna yang dirancang untuk membantu pengguna memahami cara mengoperasikan website maupun aplikasi *Android*. Halaman ini memuat informasi langkah-langkah pengajuan komplain, pengisian *Form Complain*, pengecekan status pekerjaan, serta cara memberikan feedback melalui *Survey Satisfaction*.

i) *Logout*

Demi menjaga keamanan akun dan tidak disalahgunakan oleh orang yang tidak berwenang, pengguna dapat mengeluarkan akun dengan cara *Logout* dari aplikasi.

2) Activity Diagram



Gambar 6. Activity Diagram All User

Proses dimulai dengan pengguna mengajukan keluhan dengan cara menginput *Form Complain* yang berisi tanggal, status, nama, *tenancy*, *departement*, lantai, jenis keluhan, deskripsi keluhan, dan foto keluhan. Setelah *Form Complain* terkirim, data *Form Complain* akan otomatis terkirim ke Admin dan *Section Head*.

Setelah menerima *Form Complain* dari *Tenant*, Admin bertanggung jawab untuk menyetujui *Form Complain* yang diajukan. Setelah *Form Complain* disetujui oleh Admin, *Engineer* akan menerima *Form Complain* berupa *Work Order* yang kemudian perlu dikonfirmasi sebelum *Work Order* tersebut dikerjakan. Setelah komplain selesai dikerjakan, *Engineer* diwajibkan untuk melengkapi data *Work Order*. *Work Order* yang sudah dikirim

Proses *Approval* melibatkan Admin dan *Section Head*. Sebelum melakukan *Approval*, Admin maupun *Section Head* akan melakukan pengecekan data *Work Order* yang sudah dikerjakan oleh *Engineer*. Admin maupun *Section Head* wajib menyetujui *Work Order* yang sudah dilakukan pengecekan. *Work Order* yang sudah di-approve akan terkirim ke *Tenant* sebagai report bahwa *Work Order* sudah selesai dikerjakan.

3) *Sequence Diagram*

Sequence Diagram Admin menggambarkan interaksi antara sistem dan Admin dalam mengelola *Website* yang mencakup berbagai fitur seperti *Login*, *Dashboard*, *Profile User*, *Form Complain*, *Work Order*, *Approval*, *Survey Satisfaction*, dan *Logout*.

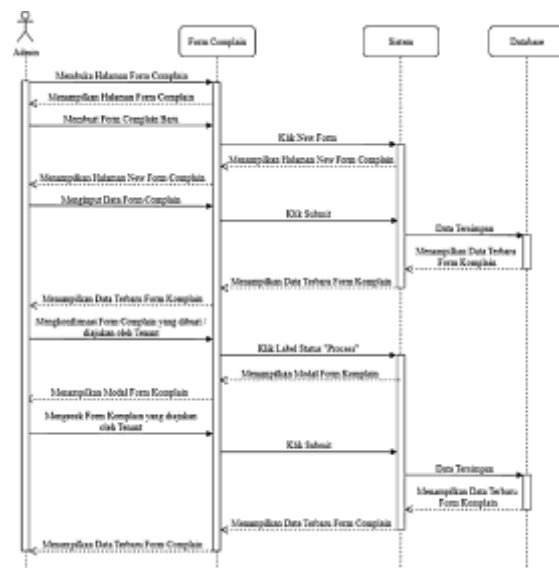


```

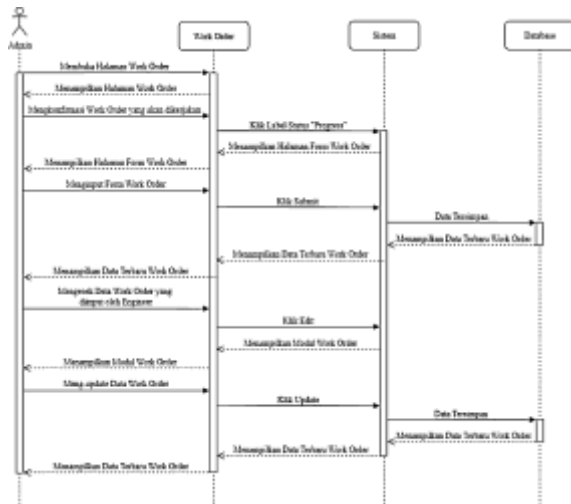
sequenceDiagram
    actor Admin
    participant Profile User
    Admin->>Profile User: Memasukkan Halaman Profile User
    Profile User-->>Admin: Menampilkan Profile User
  
```

The diagram shows an actor labeled 'Admin' and a participant labeled 'Profile User'. A solid arrow points from 'Admin' to 'Profile User' with the label 'Memasukkan Halaman Profile User'. A dashed arrow points from 'Profile User' back to 'Admin' with the label 'Menampilkan Profile User'.

Profile User menampilkan semua *User* yang terdaftar baik melalui *Website* ataupun Aplikasi. Selain *Profile User*, Admin juga dapat membuka halaman *Form Complain*.

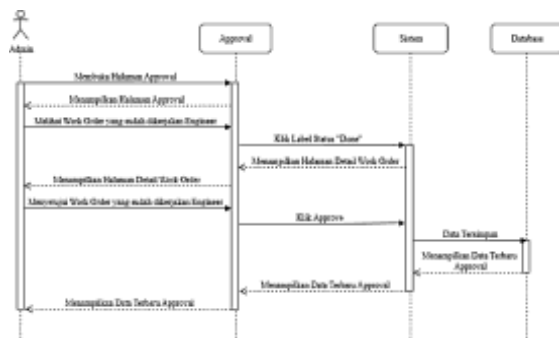


Pada halaman *Form Complain*, Admin dapat melihat tabel yang berisi riwayat komplain. Selain dapat melihat, Admin juga mendapatkan akses untuk menginput *Form Complain*, menerima dan mengedit *Form Complain* dari *Tenant*. *Form Complain* yang sudah diterima oleh Admin akan berubah menjadi *Work Order*.



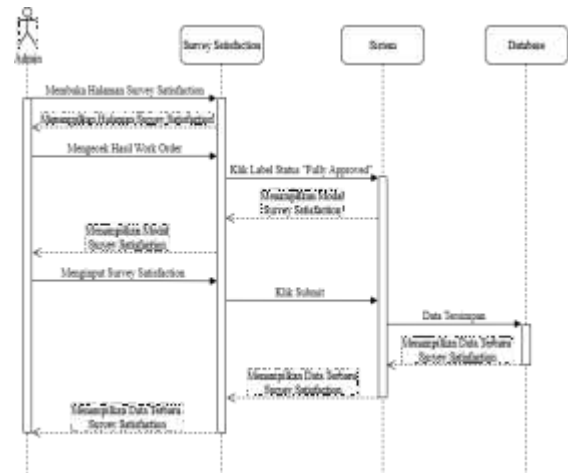
Gambar 10. Sequence Diagram Work Order Admin

Pada halaman *Work Order*, Admin dapat melihat, menerima, dan menginput *Work Order*. Proses penginputan *Work Order* dilakukan setelah komplain selesai dikerjakan oleh *Engineer*.



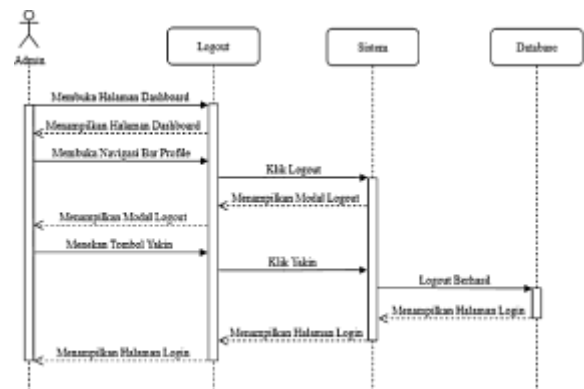
Gambar 11. Sequence Diagram Approval Admin

Setelah *Work Order* di-submit, Admin dapat melihat detail *Work Order* tersebut dan dapat memberikan persetujuan (*Approval*) terhadap *Work Order* yang telah dikerjakan. *Work Order* dapat berlanjut ke tahapan selanjutnya, jika Admin dan *Section Head* sudah menyetujui *Work Order* tersebut.



Gambar 12. Sequence Diagram Survey Satisfaction Admin

Selanjutnya, Admin dapat mengakses halaman *Survey Satisfaction* untuk mengisi umpan balik terhadap *Work Order* yang sudah dikerjakan oleh *Engineer*. Selain itu, Admin juga dapat melihat *Survey Satisfaction* yang telah selesai diisi oleh *Tenant*. Setiap data yang masuk melalui pengisian *Form Complain*, *Work Order*, *Approval*, dan *Survey Satisfaction* akan otomatis tersimpan ke dalam database.



Gambar 13. Sequence Diagram Logout Admin

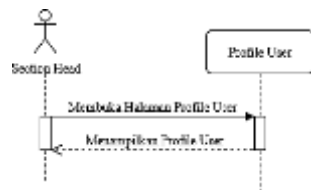
Setelah semua proses selesai dilakukan, Admin dapat keluar dari *website* dengan memilih menu *Logout*. Setelah melakukan *Logout*, Admin akan diarahkan kembali ke halaman *Login*.

b) Section Head



Gambar 14. Sequence Diagram Login Section Head

Gambar 14 menunjukkan *Sequence Diagram Login Section Head*. Sama halnya dengan Admin, *Section Head* perlu melakukan *Login* terlebih dahulu sebelum mengakses *Website*.



Gambar 15. *Sequence Diagram Profile User Section Head*

Pada fitur *Profile User*, *Section Head* hanya dapat melihat semua *user* yang terdaftar di dalam *website*.



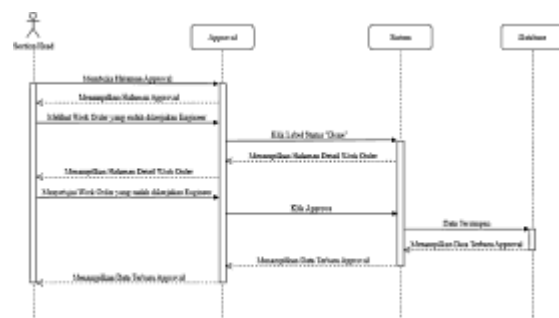
Gambar 16. *Sequence Diagram Form Complain Section Head*

Pada fitur *Form Complain*, *Section Head* hanya dapat menerima dan melihat *Form Complain* yang diajukan oleh Admin maupun *Tenant*.



Gambar 17. *Sequence Diagram Work Order Section Head*

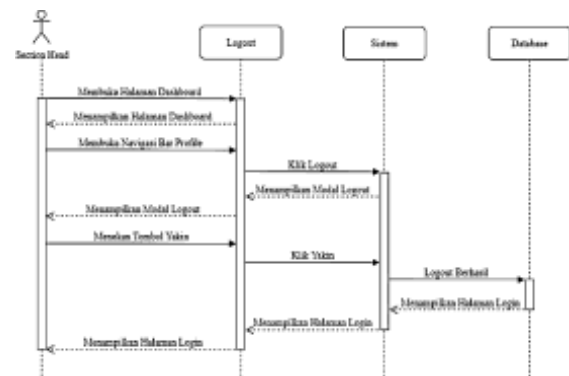
Pada fitur *Work Order*, *Section Head* juga hanya bisa melihat *Work Order* yang akan dikerjakan maupun yang sudah dikerjakan oleh *Engineer*.



Gambar 18. *Sequence Diagram Approval Section Head*

Pada halaman *Approval*, *Section Head* dapat melakukan persetujuan (*Approval*) terhadap *Work Order* yang telah dikerjakan oleh *Engineer*. *Work Order* yang sudah disetujui oleh *Section Head*, harus

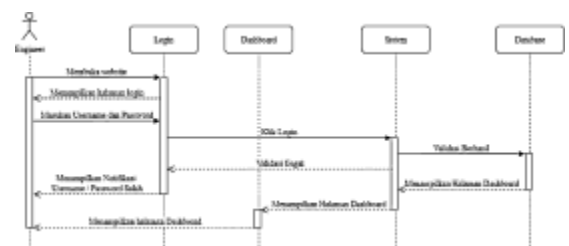
disetujui juga oleh Admin sebelum *Work Order* tersebut dinyatakan selesai.



Gambar 19. *Sequence Diagram Logout Section Head*

Setelah semua proses selesai, *Section Head* dapat keluar dari *website* dengan memilih menu *Logout*, yang akan mengarahkan kembali ke halaman *Login*.

c) Engineer



Gambar 20. *Sequence Diagram Login Engineer Sequence Diagram Engineer*

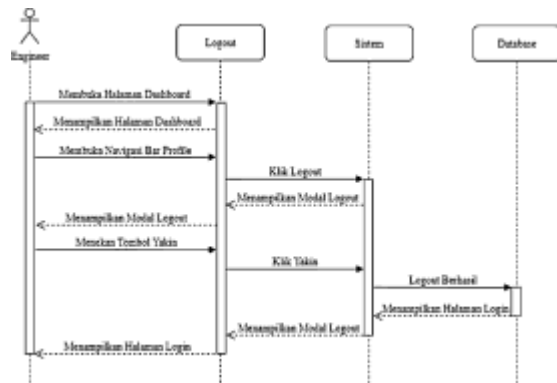
memperlihatkan alur yang dilakukan oleh *Engineer*. Sama seperti Admin dan *Section Head*, *Engineer* perlu melakukan *Login* terlebih dahulu sebelum mengakses *website*. Jika validasi berhasil, *Engineer* akan diarahkan ke halaman *Dashboard*. Pada halaman ini, *Engineer* dapat melihat jumlah *Work Order* dan jumlah *Approval*.



Gambar 21. *Sequence Diagram Work Order Engineer*

Pada *Website Engineer*, hanya terdapat 1 fitur yaitu *Work Order*. *Form Complain* yang sudah diterima oleh Admin, akan masuk ke halaman *Work*

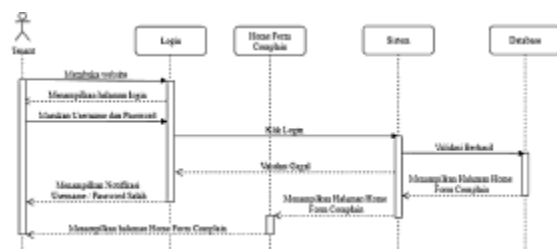
Order Engineer. Setelah pekerjaan yang dilakukan selesai, *Engineer* wajib melengkapi *Work Order*. *Work Order* yang sudah selesai diinput perlu disetujui oleh Admin dan *Section Head*. Selama *Work Order* belum disetujui oleh Admin dan *Section Head*, maka *Work Order* tersebut dinyatakan belum selesai.



Gambar 22. Sequence Diagram Logout Engineer

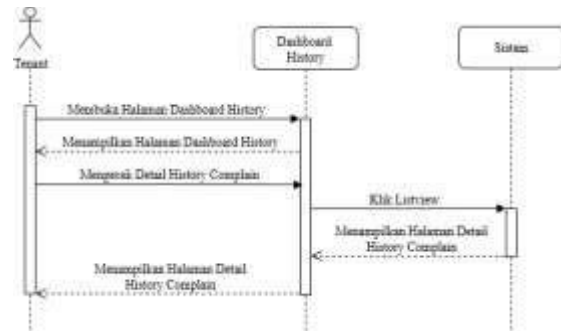
Setelah semua proses selesai, *Engineer* dapat keluar dari *website* dengan memilih menu *Logout*, yang akan mengarahkan kembali ke halaman *Login*.

d) Tenant



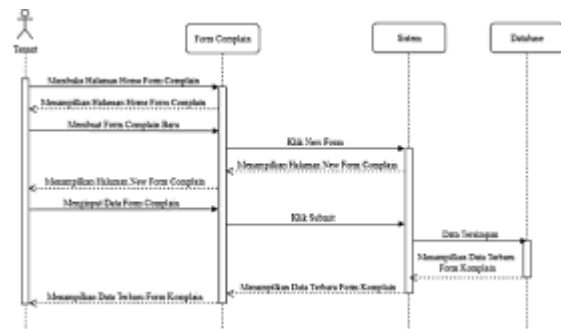
Gambar 23. Sequence Diagram Login Tenant

Gambar diatas merupakan *Sequence Diagram Tenant*, dimana *Tenant* hanya memiliki akses ke *Form Complain* dan *Survey Satisfaction*. Sebelum mengakses kedua fitur tersebut, *Tenant* perlu melakukan *Login* terlebih dahulu. Jika validasi berhasil, *Tenant* akan diarahkan ke halaman *Dashboard*. Pada halaman ini, *Tenant* dapat melihat riwayat komplain yang sudah selesai.



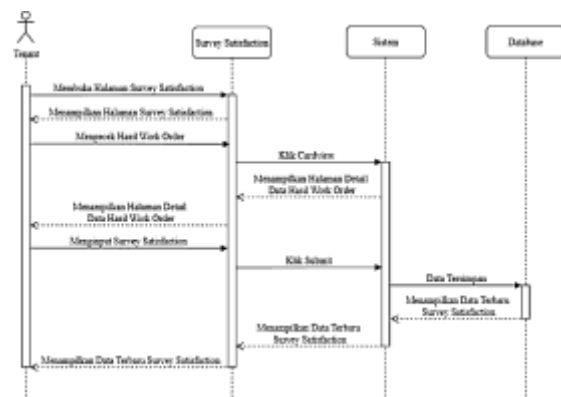
Gambar 24. Sequence Diagram Dashboard History Tenant

Sequence diagram diatas menggambarkan alur sistem dalam proses melihat riwayat komplain dan memeriksa detail komplain yang diajukan oleh *Tenant*. *Tenant* dapat melihat riwayat komplain dengan memilih salah satu item (*listview*) untuk melihat detailnya.



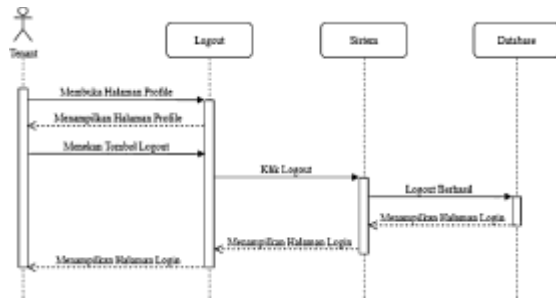
Gambar 25. Sequence Diagram Form Complain Tenant

Untuk mengajukan komplain selanjutnya, *Tenant* dapat beralih ke halaman *Form Complain* dengan mengisi *Form Complain* baru. *Form Complain* yang diajukan akan terkirim ke Admin untuk diproses lebih lanjut.



Gambar 26. Sequence Diagram Survey Satisfaction Tenant

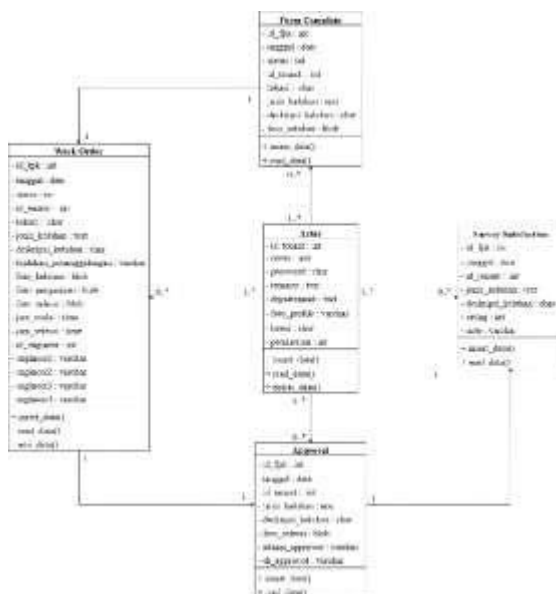
Setelah komplain selesai dikerjakan, *Tenant* wajib mengisi *Survey Satisfaction*. Hal tersebut dilakukan sebagai umpan balik terhadap pekerjaan yang sudah dilakukan.



Gambar 27. Sequence Diagram Logout Tenant

Setelah semua proses selesai, *Section Head* dapat keluar dari *website* dengan memilih menu *Logout*, yang akan mengarahkan kembali ke halaman *Login*.

4) Class Diagram



Gambar 28. Class Diagram

Dalam sistem manajemen Formulir Perintah Kerja, *Class Diagram* menunjukkan struktur basis data dan hubungan antar entitas. Ini termasuk entitas utama seperti *Form Complain*, *Work Order*, *Approval*, *Survey Satisfaction*, dan *Actor*. Entitas aktor mewakili pengguna sistem dengan atribut seperti *id_User*, *Username*, dan *Password*, serta informasi tambahan seperti *tenancy*, *departement*, *lantai*, dan *permission*. Ini menunjukkan bahwa satu pengguna dapat melakukan banyak pengaduan, pekerjaan, dan *Survey Satisfaction* karena entitas ini memiliki hubungan satu ke banyak (1..*) dengan entitas *Form Complain*, *Work Order*, *Approval*, dan *Survey Satisfaction*.

Informasi tentang pengaduan yang dibuat oleh *Tenant* berisi tanggal, lokasi, jenis keluhan, deksripsi keluhan, dan foto keluhan. Setiap jenis pengaduan memiliki hubungan satu ke satu dengan

Work Order, yang berarti satu pengaduan menghasilkan satu *Work Order*. Sebagai tanggapan terhadap pengaduan, entitas *Work Order* menyimpan rincian pekerjaan yang dilakukan, termasuk waktu pelaksanaan, identitas *Engineer* yang ditugaskan, dan foto dokumentasi pekerjaan.

Proses persetujuan pekerjaan oleh Admin dan *Section Head*, yang ditunjukkan dengan atribut *Admin_approved* dan *sh_approved*, dicatat dengan entitas *Approval*. Ini terkait langsung dengan *Form Complain* dan *Tenant*. Untuk memberikan umpan balik terhadap pekerjaan yang dikerjakan, *Tenant* dapat menggunakan entitas *Survey Satisfaction*. Survey ini memiliki fitur seperti rating dan note. Entitas *Form Complain*, *Approval*, dan *Survey Satisfaction* memiliki metode operasi dasar seperti *insert_data()* dan *read_data()*. Entitas *Actor* memiliki metode operasi dasar seperti *insert_data()*, *read_data()*, dan *delete_data()*. Sedangkan entitas *Work Order* memiliki metode operasi dasar seperti *insert_data()*, *read_data()*, dan *edit_data()*. Metode-metode ini menggambarkan fungsi CRUD (*Create*, *Read*, *Update*, *Delete*) dalam pengelolaan data. Diagram ini memberikan gambaran lengkap tentang struktur sistem yang mendukung proses pelaporan, penanganan, persetujuan, hingga evaluasi layanan melalui pengawasan kepuasan *Tenant*.

B. Struktur Database

Website dan aplikasi *Android* Formulir Perintah Kerja (FPK) dibuat dengan menggunakan MySQL sebagai sistem manajemen basis data yang dikelola menggunakan *phpMyAdmin*. Adapun beberapa tabel *Database* yang digunakan dalam pembuatan *Website* dan aplikasi Formulir Perintah Kerja adalah sebagai berikut:

1) Tabel users

Tabel 3. Struktur Tabel *users*

Nama	Tipe	Panjang	Default
<i>id_tenant</i>	<i>int</i>	11	<i>auto increment</i>
<i>nama</i>	<i>text</i>	-	-
<i>password</i>	<i>varchar</i>	255	-
<i>tenancy</i>	<i>text</i>	-	-
<i>departement</i>	<i>text</i>	-	-
<i>lantai</i>	<i>varchar</i>	3	-
<i>permission</i>	<i>int</i>	11	-

2) Tabel *work_order*

Tabel 4. Struktur Tabel *work_order*

Nama	Tipe	Panjang	Default
id_FPK	int	11	auto_increment
tanggal	date	-	-
status	int	11	-
nama	text	3	-
tenancy	text	-	-
departement	text	-	-
lokasi	varchar	20	-
lantai	varchar	3	-
jenis_keluhan	text	-	-
deskripsi_keluhan	varchar	255	-
tindakan_pena nggulangan	varchar	255	-
foto_keluhan	varchar	100	-
foto_pengerja an	varchar	100	-
foto_selesai	varchar	100	-
jam_mulai	time	-	-

Tabel 5. Lanjutan

Nama	Tipe	Panjang	Default
jam_selesai	time	-	-
id_engineer	int	11	-
engineer1	varchar	255	-
engineer2	varchar	255	-
engineer3	varchar	255	-
engineer4	varchar	255	-
admin_approved	varchar	255	-
sh_approved	varchar	255	-
rating	int	5	-
note	varchar	255	-

IV. KESIMPULAN

Berdasarkan hasil perancangan dan analisis yang dilakukan, dapat disimpulkan bahwa sistem digitalisasi Formulir Perintah Kerja (FPK) berbasis *website* dan aplikasi *Android* berhasil dirancang untuk mengatasi berbagai kendala yang ditemukan pada sistem manual. Sistem manual masih bergantung pada penggunaan kertas, terbukti tidak efisien, memakan banyak waktu, dan menimbulkan kesulitan dalam pengarsipan data serta saat proses pencarian data. Penelitian ini memodelkan sistem baru dengan menggunakan *Unified Modelling Language* (UML) yang mencakup *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. Pemodelan ini secara jelas memvisualisasikan alur kerja dan interaksi antar aktor yaitu Admin, *Section Head*, *Engineer*, dan *Tenant*, dalam menggunakan fitur-fitur inti sistem

yaitu *Form Complain*, *Work Order*, *Approval*, dan *Survey Satisfaction*. Dengan demikian, implementasi sistem digital FPK ini diharapkan dapat meningkatkan efisiensi dan efektivitas proses kerja secara signifikan dengan mengotomatisasi alur kerja, mengurangi potensi human error, menekan biaya operasional, dan menyediakan arsip data yang aman serta mudah diakses.

DAFTAR PUSTAKA

- [1] N. Elyasi, A. Bellini, and N. J. Klungseth, "Digital transformation in facility management: An analysis of the challenges and benefits of implementing digital twins in the use phase of a building," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 1176, no. 1, 2023, doi: 10.1088/1755-1315/1176/1/012001.
- [2] S. Eko, "Perancangan Aplikasi Pengenalan Budaya Nusantara Berbasis Android Dengan Metode Rad," *J. Ilmu Komput. JIK*, vol. 5, no. 01, pp. 30–39, 2022.
- [3] F. N. Hasanah, *Buku Ajar Rekayasa Perangkat Lunak*. 2020. doi: 10.21070/2020/978-623-6833-89-6.
- [4] Siska Narulita, Ahmad Nugroho, and M. Zakki Abdillah, "Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS)," *Bridg. J. Publ. Sist. Inf. dan Telekomun.*, vol. 2, no. 3, pp. 244–256, 2024, doi: 10.62951/bridge.v2i3.174.
- [5] R. Rohmanto and T. Setiawan, "Perbandingan Efektivitas Sistem Pembelajaran Luring dan Daring Menggunakan Metode Use case dan Sequence Diagram," *Intern. (Information Syst. Journal)*, vol. 5, no. 1, pp. 53–62, 2022, doi: 10.32627/internal.v5i1.506.
- [6] S. Ramdany, "Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web," *J. Ind. Eng. Syst.*, vol. 5, no. 1, 2024, doi: 10.31599/2e9afp31.
- [7] L. M. Rohmah, "Studi Literatur: Media Pembelajaran Melalui Model Pembelajaran Discovery Learning dalam meningkatkan hasil Belajar Siswa pada pembelajaran Biologi Tahun 2022/2023," *Skripsi Univ. Islam Negeri Kiai Achmad Siddiq Jember*, 2023.
- [8] O. Eko Anggraini, A. Prio Agus Santoso, E. Chotidjah Suhatmi, and U. Hanifah, "Analisis Yuridis Pengupahan Tenaga Kesehatan Kontrak Di Rumah Sakit," *J. Bisnis Manaj. dan Akunt.*, vol. 1, no. 2, pp.

- 78–84, 2021, doi:
10.47701/bismak.v1i2.1207.
- [9] P. Utomo, N. Asvio, and F. Prayogi,
“Metode Penelitian Tindakan Kelas (PTK):
Panduan Praktis untuk Guru dan Mahasiswa
di Institusi Pendidikan,” *Pubmedia J.
Penelit. Tindakan Kelas Indones.*, vol. 1, no.
4, p. 19, 2024, doi: 10.47134/ptk.v1i4.821.
- [10] B. Systems, “User and role management
module,” no. March, pp. 1685–1689, 2025.