



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PERANCANGAN APLIKASI LAYANAN PENGADUAN
CUSTOMER BERBASIS *WEB* MENGGUNAKAN *REACT JS* :
STUDI KASUS PADA PT INOVATIF 78**

TUGAS AKHIR

RANGGA ADI PUTRA

0110220047

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

Agustus 2024



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PERANCANGAN APLIKASI LAYANAN PENGADUAN
CUSTOMER BERBASIS *WEB* MENGGUNAKAN *REACT JS* :
STUDI KASUS PADA PT INOVATI F 78**

TUGAS AKHIR

RANGGA ADI PUTRA

0110220047

STT - NF

PROGRAM STUDI TEKNIK INFORMATIKA

DEPOK

Agustus 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Rangga Adi Putra

NIM : 0110220047

Depok, 27 Juli 2024

Tanda Tangan



Rangga Adi Putra

STT - NF

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Rangga Adi Putra

NIM : 0110220047

Program Studi : Teknik Informatika

Judul Skripsi : PERANCANGAN APLIKASI LAYANAN PENGADUAN
CUSTOMER BERBASIS *WEB* MENGGUNAKAN *REACT JS* :
STUDI KASUS PADA PT INOVATI F 78

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing



(Reza Maulana, S.Kom., M.Kom)

Penguji



(Nasrul. S.Pd.I, S.Kom, M.Kom)

Ditetapkan di : Depok

Tanggal : Senin, 12 Agustus 2024

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah Subhanahu wa ta'ala, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah Subhanahu wa ta'ala.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tifani Nabarian. S.Kom., M.T.I selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Ibu Nurul Janah, S.IIP., M.Hum selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama perkuliahan di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Bapak Reza Maulana, S.Kom., M.Kom selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
7. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
8. PT Inovatif 78 bapak Caesario Steve dan Ibu Wyda Ningsih beserta karyawan yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang

penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah Subhanahu wa ta'ala berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 27 Juli 2024



Rangga Adi Putra



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Rangga Adi Putra

NIM : 0110220047

Program Studi : Teknik Informatika

Jenis karya : Skripsi / Tugas Akhir

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty - Free Right*)** atas karya ilmiah saya yang berjudul :

Perancangan Aplikasi Layanan Pengaduan *Customer* Berbasis *Web* Menggunakan *React JS* : Studi Kasus Pada PT Inovatif 78

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 27 Juli 2024

STT - NF

Yang Menyatakan



(Rangga Adi Putra)

ABSTRAK

Nama : Rangga Adi Putra
NIM : 0110220047
Program Studi : Teknik Informatika
Judul : Perancangan Aplikasi Layanan Pengaduan *Customer* Berbasis Web Menggunakan *React JS* : Studi Kasus Pada PT Inovatif 78

Penelitian ini membahas perancangan aplikasi layanan pengaduan pelanggan berbasis web dengan menggunakan *React JS*, yang diterapkan pada PT Inovatif 78. Tujuan utama dari penelitian ini adalah untuk mengembangkan sistem yang efisien dan *user-friendly* untuk menangani pengaduan pelanggan secara *online*. Metodologi yang digunakan dalam pengembangan aplikasi ini mencakup analisis kebutuhan, desain antarmuka, implementasi, dan pengujian sistem. *React JS* dipilih sebagai teknologi utama untuk *frontend* karena kemampuannya dalam membangun antarmuka yang responsif dan interaktif. Studi kasus di PT Inovatif 78 menunjukkan bahwa aplikasi ini berhasil meningkatkan efektivitas pengelolaan pengaduan, mempermudah pelanggan dalam menyampaikan keluhan, dan mempercepat proses penyelesaian masalah. Hasil penelitian menunjukkan peningkatan kepuasan pelanggan dan efisiensi operasional perusahaan. Aplikasi ini diharapkan dapat menjadi model untuk pengembangan sistem pengaduan pelanggan berbasis web di perusahaan lain.

Kata kunci : Aplikasi pengaduan, *Blackbox testing*, *Firebase*, Layanan pengaduan, *React JS*.

ABSTRACT

Name : Rangga Adi Putra

NIM : 0110220047

Study Program : Informatics Engineering

Title : *Designing a Web-based Customer Complaint Service Application Using React JS: Case Study at PT Inovatif 78*

This research discusses the design of a web-based customer complaint service application using React JS, which is applied to PT Inovatif 78. The main objective of this research is to develop an efficient and user-friendly system for handling customer complaints online. The methodology used in the development of this application includes requirements analysis, interface design, implementation, and system testing. React JS was chosen as the main technology for the frontend due to its ability to build responsive and interactive interfaces. The case study at PT Inovatif 78 shows that this application successfully improves the effectiveness of complaint management, makes it easier for customers to submit complaints, and accelerates the problem resolution process. The results showed an increase in customer satisfaction and the company's operational efficiency. This application is expected to be a model for the development of web-based customer complaint systems in other companies.

Key words : *Complaint application, Blackbox testing, Firebase, Complaint service, React JS.*

STT - NF

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vi
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
BAB I	1
PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan dan Manfaat Penelitian	2
1.4 Batasan Masalah	2
1.5 Sistematika Penulisan	3
BAB II	5
KAJIAN LITERATUR	5
2.1 Perancangan	5
2.2 Aplikasi	5
2.3 Pengaduan <i>Customer</i>	5
2.4 <i>Website</i>	6
2.5 <i>ReactJS</i>	6
2.6 <i>Firebase</i>	7
2.7 <i>Activity Diagram</i>	7
2.8 <i>Use Case Diagram</i>	8

2.11	Tinjauan Pustaka	12
BAB III.....		19
METODOLOGI PENELITIAN		19
3.1	Tahapan Penelitian	19
3.2	Rancangan Penelitian	21
1.	Jenis Penelitian	21
3.	Metode Pengumpulan Data	22
4.	Metode Pengujian.....	23
5.	Metode Implementasi dan Evaluasi.....	23
6.	Lingkungan Pengembangan	23
BAB IV		25
IMPLEMENTASI DAN EVALUASI		25
4.1	Analisis Hasil Wawancara.....	25
4.2	Perancangan Use case.....	25
4.3	Perancangan Activity Diagram	27
4.4	<i>Database Firebase</i>	30
4.5	<i>Prototype Aplikasi</i>	34
4.6	Hasil Penelitian.....	38
4.9	Evaluasi	45
BAB V.....		47
KESIMPULAN DAN SARAN.....		47
5.1	Kesimpulan.....	47
5.2	Saran.....	48
DAFTAR PUSTAKA		50
LAMPIRAN		53

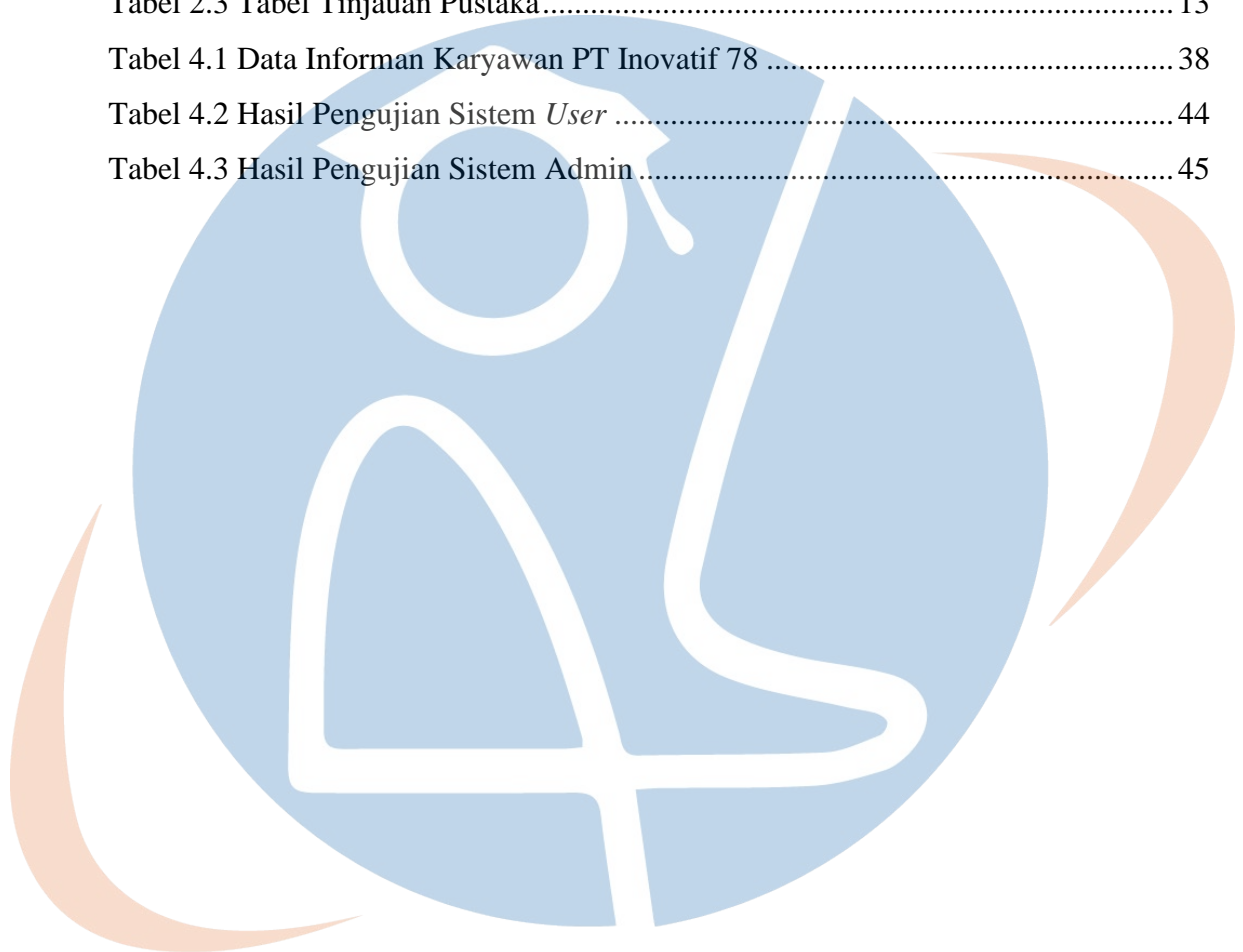
DAFTAR GAMBAR

Gambar 3.1 Diagram Tahapan Penelitian	16
Gambar 4.1 <i>Use Case Diagram</i>	26
Gambar 4.2 <i>Activity Diagram Admin</i>	27
Gambar 4.3 <i>Activity Diagram User</i>	29
Gambar 4.4 <i>Authentication User</i>	31
Gambar 4.5 <i>Firebase Storage</i>	32
Gambar 4.6 <i>Firestore Database</i>	33
Gambar 4.7 <i>Prototipe Halaman Awal User</i>	35
Gambar 4.8 <i>Prototipe Halaman Login User</i>	36
Gambar 4.9 <i>Prototipe Halaman Register User</i>	36
Gambar 4.10 <i>Prototipe Halaman Pengaduan User</i>	37

STT - NF

DAFTAR TABEL

<u>8</u> Tabel 2.1 Daftar Simbol <i>Activity Diagram</i>	7
Tabel 2.2 Daftar Simbol <i>Use Case</i>	9
Tabel 2.3 Tabel Tinjauan Pustaka.....	13
Tabel 4.1 Data Informan Karyawan PT Inovatif 78	38
Tabel 4.2 Hasil Pengujian Sistem <i>User</i>	44
Tabel 4.3 Hasil Pengujian Sistem Admin.....	45



STT - NF

BAB I

PENDAHULUAN

1.1 Latar belakang

PT Inovatif 78 adalah sebuah perusahaan yang bergerak di bidang jasa yang membantu pengelolaan permasalahan, memfasilitasi penyampaian usulan serta solusi teknologi dengan harapan memenuhi tujuan bisnis, menaikkan kinerja bisnis, serta mengatasi masalah *customer*. Namun, seperti banyak organisasi lainnya, PT Inovatif 78 juga menghadapi tantangan dalam pengelolaan pengaduan layanan sistem .

Pengelolaan pengaduan layanan sistem di PT Inovatif 78 masih dilakukan secara konvensional, di mana pelanggan harus menghubungi nomor layanan pelanggan atau mengirim email untuk melaporkan masalah atau kendala yang mereka alami. Metode ini sering kali memakan waktu dan tidak efisien, menyebabkan penundaan dalam penanganan pengaduan dan potensial mengakibatkan ketidakpuasan pelanggan [1].

Dalam zaman digitalisasi yang terus berkembang, saat ini, upaya untuk mendorong digitalisasi sedang dilakukan dengan sungguh-sungguh karena dianggap sebagai salah satu inovasi untuk memajukan suatu sektor[2]. Penggunaan teknologi bisa menjadi solusi yang efektif untuk meningkatkan efisiensi operasional dan meningkatkan pengalaman pelanggan menjadi lebih baik. [3]. Oleh karena itu, Penulis tertarik untuk menerapkan aplikasi pengaduan layanan sistem berbasis web yang mampu memberikan solusi yang lebih efisien dan responsif bagi pelanggan.

Dalam usaha untuk memenuhi kebutuhan tersebut, penelitian ini dengan judul perancangan aplikasi layanan pengaduan *customer* berbasis *web* menggunakan *React JS* : studi kasus pada PT Inovatif 78. Aplikasi ini akan dibangun menggunakan teknologi *React JS* untuk *frontend* dan *Firebase* sebagai *backend*.

1.2 Rumusan Masalah

Dari gambaran latar belakang tersebut, dapat diformulasikan permasalahan yang akan dibahas dalam penulisan ini, yakni:

1. Bagaimana rancang bangun aplikasi pengaduan layanan sistem berbasis web menggunakan *React Js*?
2. Apakah aplikasi yang telah dikembangkan sesuai dengan kebutuhan di pengguna?

1.3 Tujuan dan Manfaat Penelitian

1. Tujuan dari penelitian ini adalah:
 - a. Tujuan ini berfokus pada proses merancang dan membangun aplikasi yang efektif dan efisien untuk memfasilitasi pengaduan layanan melalui platform web. Dalam penulisan ini, aplikasi akan dikembangkan menggunakan *React JS*, yang merupakan salah satu *library JavaScript* terkemuka untuk pembangunan antarmuka pengguna yang interaktif.
 - b. Tujuan ini bertujuan untuk mengukur apakah aplikasi yang telah dikembangkan mampu untuk memenuhi kebutuhan dan ekspektasi pengguna untuk memastikan bahwa aplikasi bermanfaat dan fungsional sesuai dengan kebutuhan pengguna.
2. Manfaat dari penelitian ini meliputi:
 - a. Peningkatan efisiensi operasional perusahaan
 - b. Peningkatan kepuasan pelanggan melalui penanganan pengaduan yang lebih cepat dan efektif
 - c. Kontribusi terhadap pengembangan teknologi aplikasi berbasis web menggunakan *React JS* dan *Firebase*

1.4 Batasan Masalah

1. Dalam penelitian ini terdapat batasan masalah yang ditetapkan oleh penulis yaitu terdapat dari fungsional sistem yang dibuat di antara lain keamanan pada aplikasi

karena setiap fungsi yang digunakan untuk membuat fitur hanya memanfaatkan *tools backend* yang digunakan *Firebase*.

2. Dalam batasan masalah yang ke dua penulis dalam penelitian ini hanya membuat fitur berupa *form* pengaduan dan *admin* mengubah *status* pengaduan tidak ada fitur *chatbot*.

1.5 Sistematika Penulisan

Dalam penyusunan tugas akhir ini, struktur penulisan akan mengikuti urutan bab yang mencakup berbagai aspek penelitian. Berikut adalah tata cara penyusunan yang digunakan dalam laporan ini:

BAB I PENDAHULUAN

Bab pendahuluan ini bertujuan untuk memberikan konteks latar belakang penelitian, perumusan masalah, tujuan studi, manfaat hasil penelitian, serta batasan ruang lingkup yang menjadi fokus penelitian.

BAB II KAJIAN LITERATUR

Bab ini akan membahas teori-teori yang mendukung dalam pengembangan fitur, metode pengembangan perangkat lunak yang relevan, tinjauan literatur yang terkait, serta posisi penelitian ini dalam konteks konsep yang lebih luas.

BAB III METODOLOGI PENELITIAN

Bab ini akan menjelaskan tentang rancangan penelitian yang digunakan, alat dan teknik yang diterapkan, serta langkah-langkah yang diambil dalam rancangan solusi untuk informasi.

BAB IV IMPLEMENTASI DAN EVALUASI

Bab ini menjelaskan mengenai perencanaan penelitian yang digunakan, alat dan teknik yang digunakan, serta langkah-langkah yang diambil dalam merancang solusi, untuk mengekstraksi informasi dari resume.

BAB V KESIMPULAN DAN SARAN

Pada bagian ini akan merangkum dari penelitian, menjelaskan kesimpulan yang diperoleh, dan memberikan saran-saran yang relevan untuk penelitian yang mungkin dilakukan di masa mendatang.



STT - NF

BAB II KAJIAN LITERATUR

2.1 Perancangan

Perancangan adalah proses untuk menciptakan dan merancang sistem yang baru. Dari definisi di atas adalah tahapan yang mengikuti analisis dalam proses pengembangan sistem untuk merancang sistem yang diinginkan. Perancangan merupakan proses di mana data dan informasi disusun, yang sebelumnya telah melalui tahap perancangan atau desain, lalu disesuaikan dengan kebutuhan hingga mencapai hasil yang diharapkan[4].

2.2 Aplikasi

Aplikasi ialah *software* yang dipakai berdasarkan tujuan tertentu, seperti memproses *document*, mengarahkan *Windows &*, permainan (*game*), dan lain-lain. berdasarkan peneliti aplikasi merupakan suatu *software* yang bisa dipakai guna mempermudah juga mempercepat proses pekerjaan manusia yaitu seperti memproses *document*, bermain *game*, pengajuan masyarakat, pemberitahuan dan lain-lain [5].

2.3 Pengaduan *Customer*

Pengaduan *customer* yaitu sebuah akar informasi yang cukup penting untuk usaha-usaha bagian pengadaan pelayanan guna *problem solving* yang dapat terjadi, juga secara terus-menerus menjaga serta menaikkan pelayanan yang didapatkan agar tetap sejalan dengan keinginan dari *customer* [6].

Dalam penelitian ini pengaduan *customer* yaitu penjelasan yang dibagikan *customer* terkait seputaran masalah yang bisa dipakai untuk dasar acuan guna bagian penyelenggara layanan dalam hal ini adalah PT Inovatif 78 dapat menaikkan kualitas pelayanannya. Pengaduan *customer* juga bisa dipakai sebagai instrumen guna mempererat relasi antara *customer* dan penyelenggara layanan[7].

2.4 Website

Situs web merupakan sekelompok halaman yang saling terhubung, yang mengandung berbagai komponen seperti dokumen dan gambar disimpan di *server* web. Aplikasi web adalah aplikasi yang beroperasi di *server* web dan bisa diakses oleh pengguna melalui peramban web. Umumnya, aplikasi web menampilkan data pengguna dan informasi dari *server*[8].

Menurut Dillon, Schonhaler, dan Vossen, sejak awal tahun 1990-an, *World Wide Web* atau *website* telah mengubah kehidupan pribadi dan profesional, web telah berkembang menjadi platform yang terus berkembang dan menjadi pusat informasi yang luas, Dapat dijangkau melalui mesin pencari dan portal, web menyediakan ruang penyimpanan media yang menawarkan layanan *hosting* dan berbagai sumber daya yang sering kali gratis, serta mendukung layanan *do-it-yourself*. Selain itu, web juga membentuk *platform* perdagangan yang membantu individu dan perusahaan dalam meningkatkan kegiatan bisnis mereka[4].

2.5 ReactJS

ReactJS, atau lebih dikenal sebagai *React*, adalah perpustakaan *JavaScript* sumber terbuka yang digunakan untuk membuat antarmuka pengguna yang dinamis dan mempermudah pengembang dalam merancang aplikasi. *ReactJS* digunakan untuk menangani lapisan tampilan pada aplikasi satu halaman serta aplikasi mobile. *ReactJS* dikembangkan oleh *Facebook*, *Instagram*, dan komunitas pengembang lainnya.[9].

ReactJS bertujuan untuk menyediakan kemudahan, kejelasan, dan kemampuan untuk mengatasi skalabilitas. Salah satu fitur yang populer dari *ReactJS* adalah *JSX*, yang merupakan ekstensi dari sintaksis *ECMAScript*. *JSX* memudahkan pengembang dalam mengembangkan antarmuka pengguna pada *JavaScript*, serta membantu dalam proses debugging kesalahan pengembangan. Konsep *MVC (Model View Control)* dalam *ReactJS* menekankan penampilan elemen *View* saja, yang merupakan bagian yang ditekankan dalam pengembangan *ReactJS* karena pendekatannya yang lebih sederhana.[9].

2.6 *Firestore*






Menurut sumber dari *Dicoding* (2020), *Firestore* merupakan sebuah layanan dari *Google* yang dirancang untuk menyediakan kemudahan bagi pengembang aplikasi selama proses pengembangan. *Firestore*, atau yang dikenal sebagai BaaS (*Backend as a Service*), merupakan solusi yang diberikan oleh *Google* untuk memperlancar proses pengembangan aplikasi. Dengan menggunakan *Firestore*, pengembang aplikasi dapat fokus pada peningkatan aplikasi tanpa harus terlalu memperhatikan bagian *backend* secara detail. *Firestore* menawarkan layanan berupa basis data waktu nyata dan bagian belakang (*backend*) yang dapat dimanfaatkan di berbagai *platform*. Bagian belakang (*backend*) merujuk pada bagian dari kode aplikasi yang terhubung langsung dengan konten *database*. Dengan adanya *Firestore*, Pembuat aplikasi tidak perlu menciptakan *backend* mereka sendiri, melainkan dapat memanfaatkan antarmuka pemrograman aplikasi (*API*) yang telah disediakan oleh *Firestore*, sehingga proses pengembangan aplikasi menjadi lebih efisien[10]. *Firestore* dikembangkan dengan menggunakan basis data *MongoDB* sehingga *Firestore* menggunakan jenis basis data *NoSQL*. Berikut beberapa fitur yang disediakan oleh *Firestore*:

1. Fitur Analitik, digunakan untuk memantau pola Interaksi pengguna saat menggunakan aplikasi, yang dipresentasikan dalam satu *dashboard*.
2. Fitur Pengembangan, mencakup layanan pesan awan, otentikasi, *database realtime*, penyimpanan, *hosting*, laboratorium uji, dan pelaporan kegagalan sistem.

2.7 *Activity Diagram*

Diagram aktivitas mengilustrasikan urutan kerja atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam diagram aktivitas didasarkan pada penelitian yang tercatat dalam Jurnal Teknologi dan Sistem Informasi (2019) adalah :

Tabel 2.1 Daftar Simbol *Activity Diagram*


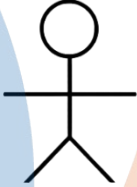

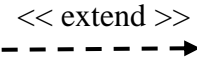
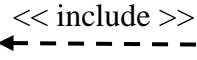
Simbol	Nama	Keterangan
	<i>Start point</i>	Digunakan pada pojok kiri atas sebagai tanda awal aktifitas
	<i>End point</i>	Akhir aktifitas
	<i>Activities</i>	Menampilkan sebuah <i>process</i> / aktifitas bisnis.
	<i>Decision Point</i>	Menunjukkan opsi untuk pengambilan keputusan, dengan kemungkinan benar atau salah.
	<i>Swimlane</i>	Pembagian aktifitas dalam diagram bertujuan untuk menunjukkan pelaku dan tugas yang dilakukan oleh mereka.

2.8 Use Case Diagram

Diagram *usecase* merupakan alat visual yang terstruktur untuk menggambarkan interaksi antara aktor dan sistem dalam bentuk skema yang terorganisir[11]. Diagram ini menyajikan berbagai elemen yang saling berhubungan, seperti aktor, *usecase*, dan relasi di antara mereka, untuk memperjelas bagaimana sistem berfungsi dan bagaimana berbagai fungsi atau layanan saling berinteraksi. Dengan menggunakan diagram *usecase*, kita dapat mendefinisikan dan menganalisis perilaku objek atau komponen dalam sistem, serta bagaimana kolaborasi antar berbagai elemen terwujud dalam implementasinya. Diagram ini membantu dalam memahami skenario pengguna dan kebutuhan sistem, serta memudahkan komunikasi dan dokumentasi selama proses pengembangan sistem.

Simbol-simbol yang digunakan dalam diagram *usecase* adalah sebagai berikut:

Tabel 2.2 Daftar Simbol *Usecase*

Nama Komponen	Keterangan	Simbol
<i>Usecase</i>	<i>Usecase</i> menggambarkan fungsionalitas yang ditawarkan oleh sistem dalam bentuk unit-unit yang saling berinteraksi melalui pertukaran pesan dengan aktor, dan biasanya dinyatakan menggunakan kata kerja.	
<i>Aktor</i>	Aktor adalah pengguna sistem yang dapat berupa bukan hanya manusia, tetapi juga aplikasi lain. Jika sebuah sistem berinteraksi dengan aplikasi lain, baik dengan menerima input maupun memberikan output, aplikasi tersebut juga dapat dianggap sebagai aktor.	
Asosiasi	Asosiasi digunakan untuk menghubungkan aktor dengan usecase, menggambarkan komunikasi antara aktor dan usecase.	
Ektensi	Relasi usecase tambahan adalah hubungan yang menghubungkan usecase utama dengan usecase tambahan, di mana usecase tambahan tersebut tetap dapat berfungsi secara mandiri meskipun tanpa adanya usecase tambahan tersebut.	
<i>Include</i>	Relasi <i>usecase</i> tambahan adalah hubungan yang menghubungkan usecase utama dengan <i>usecase</i> tambahan, di mana <i>usecase</i> tambahan memerlukan <i>usecase</i> ini untuk menjalankan fungsinya atau sebagai syarat agar fungsinya dapat dilakukan.	

2.9 Waterfall

Waterfall adalah model pengembangan sistem yang bersifat sistematis dan terstruktur, yang mengikuti urutan tahapan yang jelas dan linier. Model ini mencakup beberapa fase penting, termasuk analisis kebutuhan sistem, perencanaan, analisis dan desain, serta implementasi [12].

Dalam model *waterfall*, fase pertama, yaitu analisis dan definisi kebutuhan (*Requirements Analysis and Definition*), memainkan peran krusial. Pada tahap ini, layanan sistem yang akan dikembangkan, kendala yang mungkin dihadapi, dan tujuan yang ingin dicapai ditetapkan melalui serangkaian konsultasi dengan pengguna akhir dan pemangku kepentingan lainnya. Informasi yang dikumpulkan selama konsultasi ini kemudian didefinisikan secara rinci, berfungsi sebagai dasar spesifikasi sistem [13]. Spesifikasi ini menjadi panduan untuk semua fase selanjutnya dalam model *waterfall*, memastikan bahwa setiap tahapan pengembangan berfokus pada memenuhi kebutuhan yang telah ditetapkan secara tepat dan terperinci. Dengan demikian, model *waterfall* menawarkan pendekatan yang terorganisir dan terukur untuk pengembangan sistem, meskipun prosesnya tidak fleksibel dan sulit untuk diubah setelah fase-fase awal selesai.

Metode pengembangan sistem yang dipilih adalah metode *waterfall*, yang dikenal karena pendekatannya yang sistematis dan terstruktur. Model ini mengikuti urutan tahapan yang jelas, sebagaimana dijelaskan oleh *Sommerville* dan diuraikan oleh Hidayat (2021). Berikut adalah tahapan-tahapan dalam *Waterfall Model*:

1. Analisis

Tahap ini melibatkan pengumpulan dan analisis kebutuhan perangkat lunak dari pengguna untuk memastikan bahwa sistem yang dikembangkan benar-benar memenuhi kebutuhan mereka. Proses ini mencakup identifikasi dan dokumentasi persyaratan fungsional dan non-fungsional yang akan menjadi dasar pengembangan selanjutnya.

2. Desain Perangkat Lunak

Pada fase ini, desain aplikasi dirancang berdasarkan kebutuhan yang telah dikumpulkan. Ini mencakup pembuatan desain antarmuka pengguna yang intuitif, arsitektur perangkat lunak yang efektif, dan prosedur pengkodean yang akan diikuti selama pengembangan. Tujuan dari tahap ini adalah untuk merancang solusi teknis yang akan diimplementasikan.

3. Implementasi dan Pengujian Unit

Dalam tahapan ini, perangkat lunak dan basis data dikembangkan sesuai dengan desain yang telah dibuat sebelumnya. Setelah modul-modul program dikembangkan, masing-masing modul diuji secara individual melalui pengujian unit untuk memastikan bahwa mereka berfungsi dengan baik dan sesuai dengan spesifikasi yang ditetapkan.

4. Integrasi dan Pengujian Sistem

Pada fase ini, seluruh program diintegrasikan menjadi sistem yang lengkap. Setelah integrasi, sistem diuji secara menyeluruh untuk memastikan bahwa semua komponen bekerja secara harmonis dan bahwa sistem secara keseluruhan memenuhi persyaratan yang telah ditetapkan.

5. Operasi dan Pemeliharaan

Tahap terakhir melibatkan pemeliharaan sistem yang telah diluncurkan. Ini mencakup perbaikan *bug* yang ditemukan setelah peluncuran serta penyesuaian dan pembaruan untuk memenuhi kebutuhan pengguna yang mungkin berkembang seiring waktu. Pengembang bertanggung jawab untuk memastikan sistem tetap berfungsi dengan baik dan relevan dengan kebutuhan pengguna.

2.10 *BlackBox Testing*

Metode *Black Box Testing* adalah teknik pengujian yang digunakan untuk mengidentifikasi kesalahan dalam sistem aplikasi, seperti kesalahan pada fungsi sistem atau menu yang hilang [14]. Metode ini fokus pada pengujian fungsionalitas sistem aplikasi. Pengujian dilakukan dengan menggunakan data *input* acak untuk memastikan hasil yang akurat. Jika data *input* tidak valid,

sistem informasi akan menolaknya atau tidak menyimpannya dalam basis data. Sebaliknya, jika data *input* valid, maka data tersebut akan diterima dan disimpan dalam sistem informasi.

Pengujian *black box testing*, sering dikenal sebagai pengujian perilaku, adalah metode di mana struktur internal dan logika dari perangkat lunak yang diuji tidak diketahui oleh penguji [15]. Dalam pendekatan ini, pengujian dilakukan berdasarkan spesifikasi kebutuhan tanpa memerlukan analisis kode sumber. Hal ini memungkinkan pengujian dilakukan dari perspektif pengguna akhir, fokus pada bagaimana perangkat lunak berfungsi dalam praktik sesuai dengan spesifikasi yang telah ditetapkan.

Ada berbagai jenis pengujian dalam metode *black box*, termasuk partisi *ekivalen*, analisis nilai batas, grafik penyebab-efek, pengujian *array orthogonal*, pengujian transisi status, dan *fuzzing*. Setiap metode ini menawarkan pendekatan berbeda dalam mengevaluasi fungsionalitas sistem berdasarkan skenario dan data *input* yang dirancang untuk menguji batas dan kekuatan perangkat lunak.

Kelebihan utama dari *black box testing* adalah kemampuannya untuk mendeteksi kekurangan dalam spesifikasi kebutuhan yang mungkin tidak terlihat selama fase pengembangan perangkat lunak. Metode ini efektif dalam menemukan *bug* atau masalah yang berhubungan dengan fungsionalitas perangkat lunak yang sesuai dengan ekspektasi pengguna. Namun, kekurangan dari pendekatan ini adalah bahwa pengujian tidak dapat dilakukan secara menyeluruh. Pengetahuan penguji yang terbatas mengenai struktur internal perangkat lunak dapat menyebabkan beberapa masalah tidak terdeteksi, karena penguji tidak memiliki akses atau wawasan tentang kode sumber dan logika internal sistem yang diuji.

2.11 Tinjauan Pustaka

Dalam tinjauan literatur, peneliti akan menelaah Beberapa penelitian sebelumnya digunakan sebagai landasan referensi yang relevan serta sebagai

perbandingan antara penelitian yang dilakukan oleh peneliti dengan penelitian yang dilakukan oleh pihak lain.

Tabel 2.3 Tabel Tinjauan Pustaka

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
1.	Egi Bahari Dwi Fraska, 2023	<i>Design Of Customer Complaint Ticket Management Information System PT. Website-Based Indonesian Jinde Group Using The Waterfall Method</i>	Sistem informasi layanan pelaporan keluhan masalah jaringan <i>online</i>	Customer pada PT. Jinde Grup Indonesia	Website Sistem Informasi Manajemen Tiket Keluhan Pelanggan
2.	Nirma Dwi Wulansari, 2023	<i>Design Of A Complaint Information System At RSUD Dr. Soedarsono Pasuruan Website Based Using The Waterfall Method</i>	Layanan informasi dan kritik serta saran secara <i>online</i>	Pasien RSUD Dr. Soedarsono Pasuruan	Website layanan informasi pasien
3.	Sri Hanifah, 2021	<i>Design And Development Of A Web-Based Customer Complaints Application At Perumda Air Minum Tirta Muare Ulakan,</i>	Sistem layanan pengaduan pelanggan <i>online</i>	Pelanggan Perumda Air Minum Tirta Muare Ulakan Kabupaten Sambas	Website layanan pengaduan pelanggan

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
		Sambas <i>Regency</i>			
4.	Adelia Martika, 2024	<i>Designing A Customer Relationship Management System For Web-Based Customer Complaints Using The Extreme Programming Model</i>	Sistem manajemen dan layanan pengaduan nasabah online	Nasabah Bank Capital Indonesia	Website manajemen dan layanan pengaduan nasabah
5.	Iwan Setiawan, 2022	<i>Analysis And Design Of A Customer Complaints System At PT Customs Pos Indonesia (Persero) Mail Processing Center (MPC)</i>	Implementasi CRM pada Sistem pengaduan customer online	Customer PT Pos Indonesia (PERSERO)	Website dengan implementasi CRM pada sistem pengaduan customer
6.	Dwi Ismiyana Putri, 2022	<i>Penelitian Web-Based Customer Complaint Data Processing System Using Extreme Programming Method</i>	Layanan sistem pengolahan data keluhan pelanggan online	Perusahaan telekomunikasi	Website pengolahan data keluhan pelanggan
7.	Rangga Adi Putra, 2024	Perancangan Aplikasi Pelayanan Pengaduan Customer	Sistem layanan pengaduan customer secara online	Customer PT Inovatif 78	Website layanan pengaduan customer

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
		Berbasis Web Menggunakan <i>React JS</i> : Studi Kasus Pada PT Inovatif 78			

Perbandingan hal-hal di atas dengan penelitian-penelitian sebelumnya ditunjukkan pada penjelasan di bawah ini :

1. Tinjauan pustaka yang pertama adalah penelitian yang dilaksanakan Egi Bahari Dwi Fraska, Umi Chotijah pada tahun 2023 yang berjudul *Design of Customer Complaint Ticket Management Information System PT. Website-Based Indonesian Jinde Group Using the Waterfall Method*. Penelitian ini menerapkan model *waterfall* dengan penerapan pengujian *black box* sesuai dengan metodologi yang telah ditetapkan. Hasil yang didapat adalah penggunaan kerangka data administrasi tiket pengaduan klien, yang memperluas kapasitas divisi konfirmasi kualitas untuk mengawasi pengorganisasian masalah dengan lebih efektif . Ini mengutamakan tugas berdasarkan tiket keluhan pelanggan, sehingga mempercepat identifikasi masalah dan penyelesaiannya. [4].
2. Tinjauan pustaka kedua didasarkan pada penelitian yang dilakukan oleh Nirma Dwi Wulansari, Gita Indah Marthasari, dan Briansyah Setio Wiyono pada tahun 2023 dengan judul "Perancangan Sistem Informasi Pengaduan di Situs Web RSUD Dr. Soedarsono Pasuruan Berbasis Metode *Waterfall*". Model yang dipakai dalam merancang aplikasi ini adalah model *waterfall* dari metodologi pengembangan perangkat lunak. Model tersebut terdiri dari beberapa tahapan, yakni analisis, desain, implementasi, pengujian, dan pemeliharaan. Diharapkan bahwa hasil penelitian yang diaplikasikan dalam pembuatan sistem informasi pengaduan di RSUD Dr. R. Soedarsono Kota Pasuruan akan memberikan informasi yang mudah

diakses di mana pun dan kapan pun serta mendukung pasien dalam memberikan kritik dan saran melalui *platform* daring [16].

3. Pada *review* pustaka yang ketiga yang dilakukan oleh Sri Hanifah dan Erifa Syahnaz, pada tahun 2021 pada judul *Design and Development of a Web-Based Customer Complaints Application at Perumda Air Minum Tirta Muare Ulakan, Sambas Regency*. *Customer* yang mau menyampaikan keluhannya wajib mengunjungi kantor pusat, Setelah itu, proses tersebut masih dicatat di buku besar oleh bagian transaksi dan layanan. Dibuat aplikasi pengaduan *customer* melalui *website* supaya bisa membantu *customer* ketika menyampaikan pengaduannya. Melakukan pengaduan *customer* PERUMDA dengan metode *prototype*. Metode *prototyping* menghasilkan *prototipe* sistem yang berfungsi sebagai penghubung antara pengembang dan pengguna dalam proses pengembangan sistem informasi.. Langkah-langkah dalam model *prototipe* meliputi pengumpulan data, pembangunan *prototipe*, evaluasi *prototipe*, pembuatan sistem, pengujian sistem, dan evaluasi sistem. Berdasarkan hasil penelitian, disimpulkan bahwa diharapkan aplikasi pengaduan pelanggan daring ini akan mempermudah pelanggan dalam menyampaikan keluhan mereka di mana pun dan kapan pun tanpa harus datang langsung ke kantor Tirta Muare Ulaka Perumda Air, Kabupaten Sambas [17].
4. Tinjauan pustaka yang keempat yang dilakukan oleh Adelia Martika, Deanna Durbin Hutagalung, pada tahun 2024 pada judul *Designing a Customer Relationship Management System for Web-Based Customer Complaints Using the Extreme Programming Model*. Kesalahan dan kelalaian dalam pemrosesan pengaduan melalui telepon seringkali menjadi hambatan bagi pegawai dalam memberikan pelayanan yang cepat. Untuk mengatasi masalah ini, khususnya dalam konteks penyampaian keluhan pelanggan Bank *Capital* Indonesia, Dibutuhkan sebuah sistem yang mempermudah pelanggan, yaitu sistem E-CRM. Perancangan sistem ini dilakukan menggunakan metode *extreme programming*. Selain itu,

observasi, wawancara, dan penelitian literatur digunakan sebagai metode untuk mengumpulkan data. Basis data yang dimanfaatkan adalah PHP dan *MySQL*. Temuan dari penelitian menunjukkan bahwa sistem ini berhasil menciptakan suatu sistem yang efisien dan efektif dalam menangani keluhan pelanggan, dengan tingkat kepuasan dari evaluasi kuesioner mencapai 93%. Dengan adanya sistem CRM *online* ini, proses pelayanan menjadi lebih cepat dan responsif, menghindari kebosanan pelanggan terhadap hasil survei sebesar 86%. Selain itu, sistem ini juga mampu mengolah dan menyajikan laporan berdasarkan keluhan pelanggan, dengan tingkat kepuasan dari evaluasi kuesioner mencapai 89%.[7].

5. Tinjauan pustaka yang kelima dilakukan oleh Iwan Setiawan, Muhammad Ruslan Maulani, pada tahun 2022 dengan judul *analysis and design of a customer complaints system at PT customs Pos Indonesia (Persero) Mail Processing Center (MPC)*. Dengan metode yang dipakai ialah CRM dan *Black Box Testing*. Hasil yang di peroleh adalah menggunakan sistem pengaduan *customer* agar pengaduan dapat didaftarkan secara *online*, membuat sistem pengelolaan pengaduan agar pengelola dapat mengakses pengaduan yang masuk dan melacaknya dengan baik, membuat sistem evaluasi pada departemen sasaran untuk melihat kinerja departemen.[18].
6. Pada tinjauan pustaka yang keenam, dilakukan oleh Dwi Ismiyana Putri, Jafar Shadiq, Muhammad Surya Apandi, dan Muhammad Ari Kuncoro pada tahun 2022. Penelitian tersebut berjudul "Sistem Pengolahan Data Keluhan Pelanggan Berbasis Web Menggunakan Metode *Extreme Programming*." Metode *Extreme Programming* yang berbasis web digunakan dalam pengembangan sistem informasi pengaduan pelanggan, dengan bantuan *framework Laravel 9*. Sistem informasi yang dikembangkan memastikan tidak terjadi antrian saat melakukan transaksi keluhan pelanggan, dan menggunakan teknisi yang diatur secara jelas[19].
7. Berdasarkan keenam jurnal di atas didapatkan persamaan penelitian saya dengan Judul "Perancangan Aplikasi Pelayanan Pengaduan *Customer*

Berbasis *Web* Menggunakan *React JS* : Studi Kasus Pada PT Inovatif 78" sejalan dengan berbagai penelitian tentang pengembangan sistem pengaduan pelanggan berbasis web. Penelitian sebelumnya menggunakan metodologi berbeda seperti *waterfall* (Egi Bahari Dwi Fraska & Umi Chotijah, 2023; Nirma Dwi Wulansari et al., 2023), *prototyping* (Sri Hanifah & Erifa Syahnaz, 2021), *extreme programming* (Adelia Martika & Deanna Durbin Hutagalung, 2024; Dwi Ismiyana Putri et al., 2022), dan CRM dengan *black box testing* (Iwan Setiawan & Muhammad Ruslan Maulani, 2022). Semua penelitian ini bertujuan untuk meningkatkan efisiensi dan aksesibilitas dalam pengelolaan pengaduan pelanggan berbasis web.

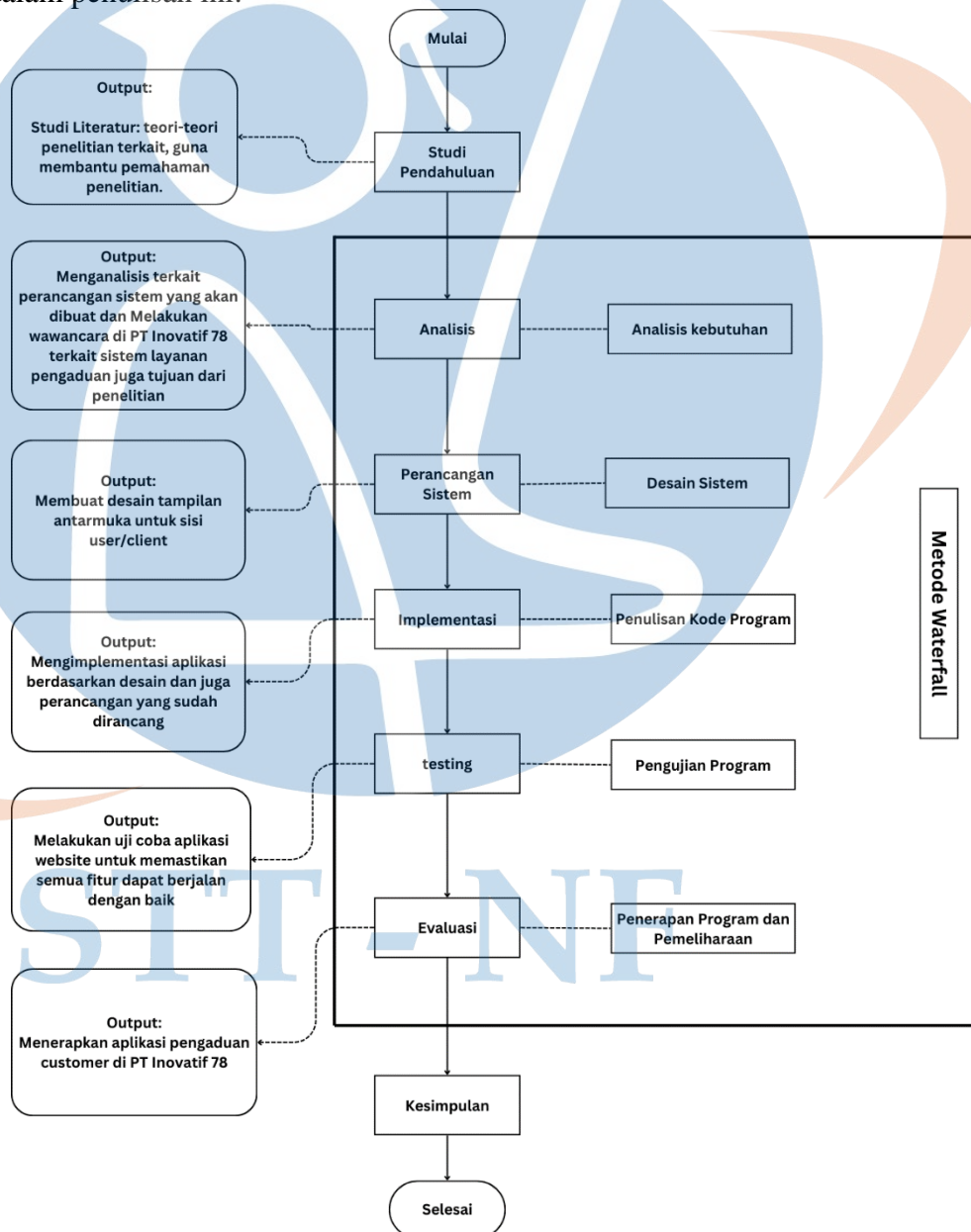


STT - NF

BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Dibawah ini adalah langkah-langkah yang akan dilakukan secara umum dalam penulisan ini:



Gambar 3.1 Diagram Tahapan Penelitian

Penjelasan:

Berikut adalah penjelasan dari setiap tahap dalam tahapan penelitian:

1. Studi Pendahuluan

Tahap pertama adalah studi pendahuluan, di mana dilakukan studi literatur untuk memahami teori-teori terkait penelitian. Tahap ini membantu memperoleh pemahaman mendalam tentang topik yang akan diteliti serta mengidentifikasi celah penelitian yang bisa diisi.

2. Analisis

Tahap ini melibatkan identifikasi dan analisis kebutuhan sistem yang akan dikembangkan. Informasi rinci mengenai kebutuhan sistem dikumpulkan dari berbagai pemangku kepentingan seperti pengguna akhir dan manajemen melalui wawancara, survei, dan observasi. Tahap ini memastikan bahwa semua kebutuhan dan harapan dari sistem yang akan dikembangkan dipahami dengan jelas.

3. Perancangan Sistem

Pada tahap ini, desain detail dari komponen-komponen sistem dibuat berdasarkan hasil analisis. Ini mencakup desain antarmuka pengguna, desain basis data, dan desain modul-modul program. Tujuan dari tahap ini adalah menghasilkan *blueprint* yang menjadi panduan bagi pengembang dalam membangun sistem.

4. Implementasi

Pada tahap ini, pengembang menulis kode program untuk membangun sistem sesuai dengan desain yang telah dibuat. Penggunaan bahasa pemrograman dan alat pengembangan yang tepat sangat penting untuk memastikan sistem berfungsi seperti yang diharapkan.

5. Pengujian

Sistem diuji secara menyeluruh untuk memastikan bahwa semua komponen berfungsi dengan baik dan memenuhi spesifikasi yang telah ditetapkan. Berbagai jenis tes dilakukan, termasuk uji unit, uji integrasi, uji sistem, dan uji

penerimaan, untuk menemukan dan memperbaiki bug atau kesalahan dalam sistem.

6. Evaluasi

Tahap ini melibatkan penerapan sistem ke lingkungan produksi dan mulai digunakan oleh pengguna akhir. Sistem juga dipelihara secara berkala untuk memastikan tetap berfungsi dengan baik dan dapat beradaptasi dengan perubahan kebutuhan atau lingkungan. Pada tahap ini, aplikasi pengaduan pelanggan diterapkan di PT Inovatif 78.

7. Kesimpulan

Semua hasil dari penelitian dan pengembangan sistem dirangkum dan didokumentasikan. Evaluasi terhadap proses dan hasil proyek dilakukan untuk menilai keberhasilan dan efektivitas metode yang digunakan. Pelajaran yang diperoleh selama proyek diidentifikasi untuk digunakan dalam proyek-proyek masa depan.

3.2 Rancangan Penelitian

Rancangan penelitian ini disusun sebagai langkah awal dalam proses penelitian yang mencakup jenis penelitian, metode penelitian, teknik analisis data, prosedur pengumpulan data, ruang lingkup pengembangan, durasi penelitian, metode pengujian, serta strategi implementasi dan evaluasi.

1. Jenis Penelitian

Metode penelitian ini menggunakan *Waterfall* adalah model proses pengembangan sistem yang terdiri dari beberapa tahapan yang dilakukan secara berurutan. Proses dimulai dengan Studi Pendahuluan, di mana dilakukan studi literatur terkait teori-teori penelitian untuk memperdalam pemahaman mengenai proyek yang akan dikerjakan. Tahap berikutnya adalah Analisis, yang melibatkan analisis kebutuhan sistem dan wawancara di PT Inovatif 78 untuk memahami kebutuhan dan tujuan sistem layanan pengaduan. Setelah itu, dilakukan Perancangan Sistem, di mana dibuat desain sistem dan tampilan antarmuka untuk pengguna atau klien.

Tahapan selanjutnya adalah Implementasi, di mana desain yang telah dibuat diterapkan ke dalam bentuk kode program. Setelah implementasi, dilakukan Testing atau pengujian untuk memastikan bahwa semua fitur aplikasi berjalan dengan baik dan sesuai dengan kebutuhan yang telah dianalisis. Setelah pengujian, dilakukan Evaluasi yang mencakup penerapan program dan pemeliharaan untuk memastikan bahwa aplikasi dapat diterapkan secara efektif di PT Inovatif 78. Proses ini diakhiri dengan Kesimpulan yang merangkum seluruh proses yang telah dilakukan, dan setelah itu proyek dinyatakan Selesai. Setiap tahap dalam metode *Waterfall* ini dilakukan secara berurutan, memastikan bahwa setiap bagian dari proyek telah dievaluasi secara menyeluruh sebelum melanjutkan ke tahap berikutnya.

Penelitian ini juga menerapkan metode penelitian pengembangan (R&D), yang bertujuan untuk menciptakan produk baru atau meningkatkan produk yang telah ada. Dalam konteks ini, pengembangan difokuskan pada aplikasi layanan pengaduan pelanggan di PT Inovatif 78 guna meningkatkan efisiensi.

2. Metode Analisis Data

Penelitian ini menggunakan pendekatan metode analisis kualitatif, metode ini merupakan proses untuk memahami fenomena-fenomena manusia atau sosial dengan menciptakan gambaran yang menyeluruh dan kompleks yang dapat disajikan dengan kata-kata, melaporkan pandangan terinci yang diperoleh dari sumber informan. Penelitian kualitatif adalah metode penelitian yang bertujuan untuk mendapatkan pemahaman tentang kenyataan melalui proses berpikir induktif [20]. Hasil dari analisis ini akan digunakan untuk mencapai tujuan penelitian.

3. Metode Pengumpulan Data

Untuk memperoleh data yang diperlukan guna mendukung pelaksanaan penelitian ini digunakan beberapa teknik pengumpulan data antara lain sebagai berikut :

a. Wawancara

Suatu teknik pengumpulan data dengan cara wawancara yang merupakan salah satu teknik pengumpulan data kualitatif, pada penelitian ini penulis akan melakukan wawancara yang digunakan untuk mengumpulkan informasi atau data baik secara lisan atau tertulis yang dilakukan pada PT Inovatif 78.

b. Studi Kepustakaan

Studi kepustakaan adalah usaha yang dilakukan peneliti untuk menghimpun informasi yang relevan dengan topik atau masalah yang akan atau sedang diteliti. Data terkait yang di dapat pada PT Inovatif 78 antara lain: Struktur Organisasi, Data masyarakat Metro dan laporan pengaduan yang sedang berjalan serta seperti jurnal yang terkait dengan penelitian sistem informasi pengaduan layanan *customer* berbasis *web*.

4. Metode Pengujian

Dalam penelitian ini, pengujian akan dilakukan dengan menggunakan metode *Blackbox Testing*, yaitu pengujian yang bertujuan untuk mengetahui hasil *input* dan *output* dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak tersebut. Metode ini adalah salah satu metode yang mudah digunakan untuk melakukan pengujian karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan[21]

5. Metode Implementasi dan Evaluasi

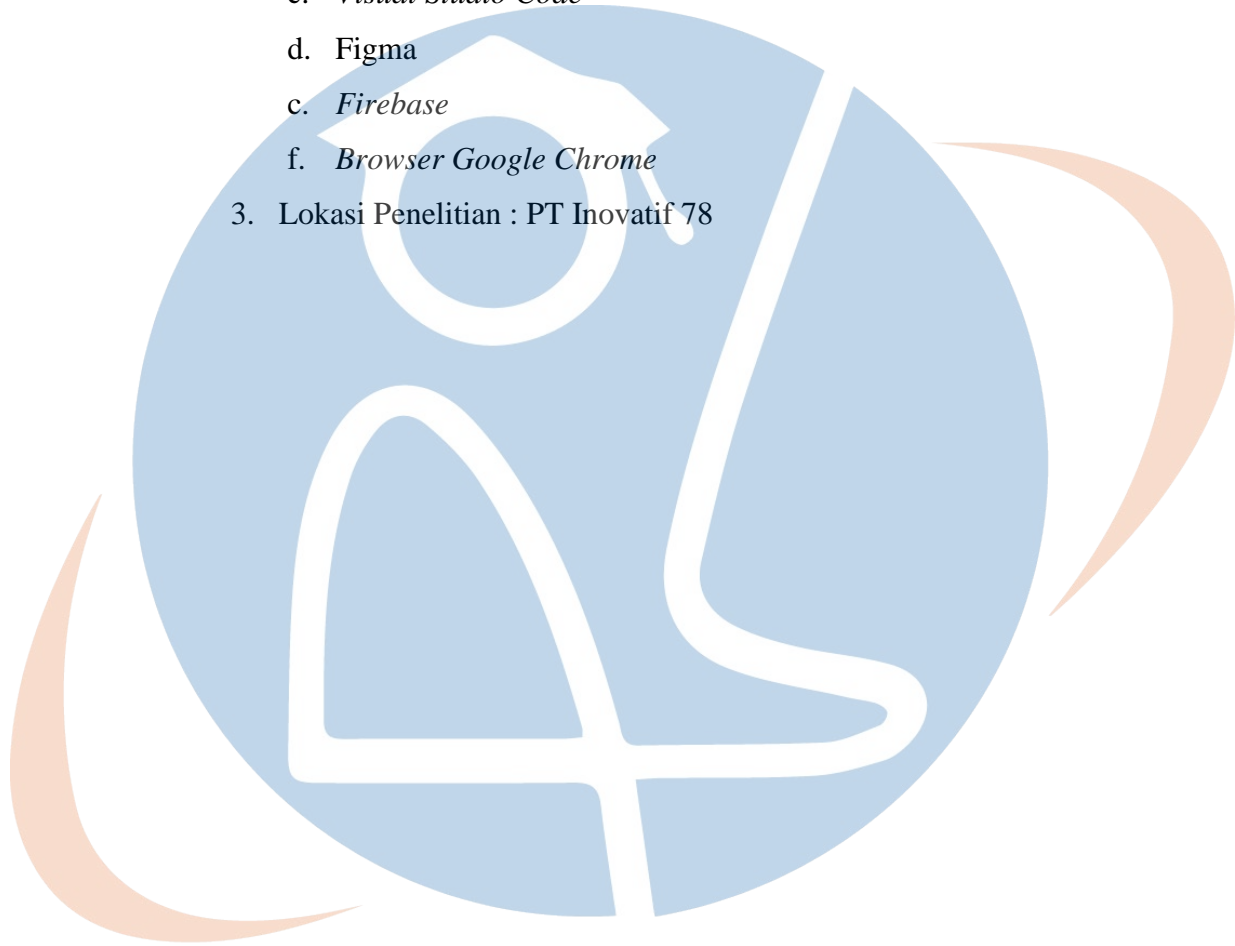
Pada tahapan implementasi dari penelitian ini adalah menerapkannya kepada pelayanan *customer* di PT Inovatif 78, dan untuk metode evaluasi dari penelitian ini melibatkan uji coba dari pengguna dan analisis kinerja sistem.

6. Lingkungan Pengembangan

1. Perangkat Keras yang digunakan:

- a. Laptop *HP Pavilion*
- b. Processor *AMD Ryzen 5 5500U with Radeon Graphics*
- a. Ram 16 GB

- b. SSD 512 GB
- 2. Perangkat Lunak yang digunakan:
 - a. Sistem Operasi *Windows 11* 64-bit
 - b. *React JS*
 - c. *Visual Studio Code*
 - d. Figma
 - c. *Firebase*
 - f. *Browser Google Chrome*
- 3. Lokasi Penelitian : PT Inovatif 78



STT - NF

BAB IV

IMPLEMENTASI DAN EVALUASI

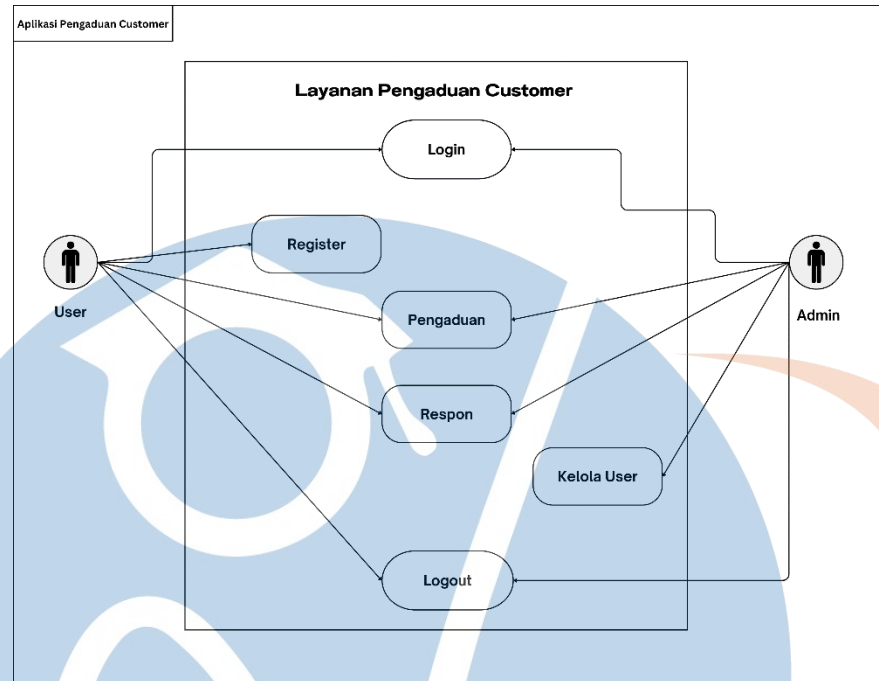
4.1 Analisis Hasil Wawancara

Berdasarkan hasil wawancara untuk penelitian yang telah dilakukan dengan beberapa tujuan yang menjadi dasar penelitian dalam perancangan sistem aplikasi layanan pengaduan *customer* pada PT Inovatif 78 berbasis *web* menggunakan *react js* dan *firebase*, setelah dilakukannya analisis berdasarkan hasil wawancara yang telah penulis lakukan dengan narasumber. Dalam bab ini penulis akan menyajikan hasil penelitian yang diperoleh, dengan menggunakan hasil wawancara dan dokumentasi. Disajikan dengan cara metode kualitatif dengan menjelaskan secara rinci data tersebut, hingga dapat ditarik suatu kesimpulan maka akan dilakukannya untuk perancangan desain sistem.

Dalam penyajian hasil penelitian, penulis akan memulai dengan menggambarkan konteks dan latar belakang dari setiap wawancara. Setiap wawancara akan dirinci dengan memberikan informasi tentang siapa yang diwawancarai, peran mereka dalam organisasi, dan poin-poin utama yang dibahas selama wawancara. Ini termasuk identifikasi masalah yang dihadapi, saran dan rekomendasi yang diberikan, serta harapan mereka terhadap sistem yang baru.

4.2 Perancangan *Use Case*

Pada desain perancangan sistem ini dilakukan dengan *Use case Diagram* adalah satu jenis dari Diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan aktor [22], *Use case Diagram* dapat mendeskripsikan tipe interaksi antara pengguna sistem dengan dengan sistemnya. Berikut adalah *use case diagram* pada penelitian ini.



Gambar 4. 1 *Use Case Diagram*

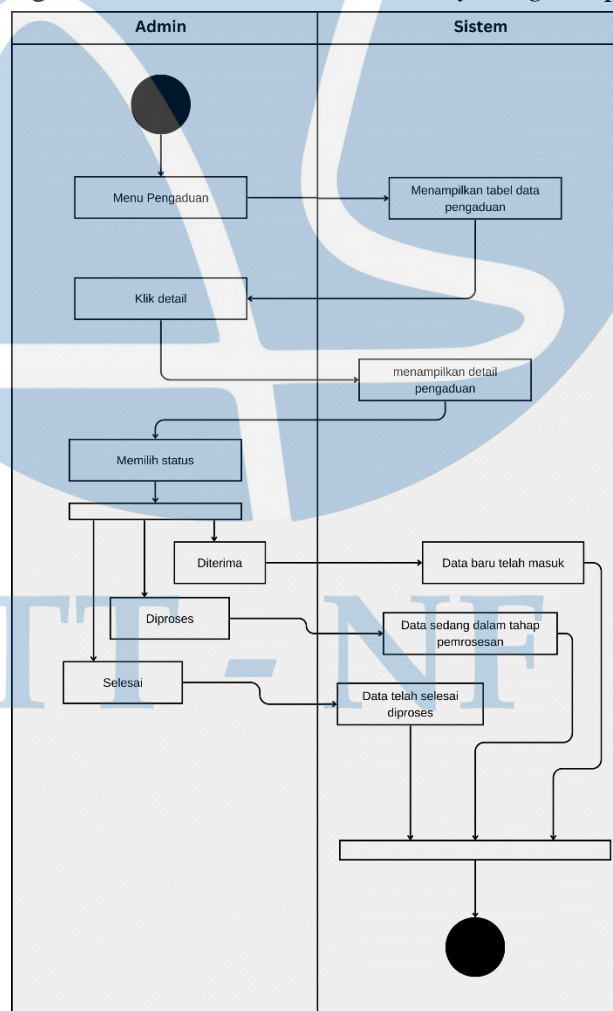
Diagram tersebut merupakan ilustrasi dari sistem *E-Complaint INV78* yang menunjukkan interaksi antara pengguna (*User*) dan administrator (*Admin*) dengan sistem. Pengguna dan administrator adalah aktor utama dalam sistem ini. Pengguna berperan dalam melakukan *login*, mendaftar, mengirim pengaduan, menerima respon, dan *logout*. Administrator bertanggung jawab untuk *login*, menerima dan merespon pengaduan, mengelola data pengguna, dan *logout*.

Pengguna dapat melakukan *login* untuk mengakses sistem, kemudian mendaftar jika belum memiliki akun. Setelah terdaftar dan masuk ke sistem, pengguna dapat membuat dan mengirim pengaduan yang akan diterima oleh administrator. Administrator kemudian memproses dan memberikan tanggapan atas pengaduan tersebut. Pengguna akan menerima respon dari administrator terkait pengaduan yang telah dikirimkan. Administrator juga memiliki kemampuan untuk mengelola data pengguna, termasuk menambahkan, mengubah, atau menghapus akun pengguna yang ada dalam sistem.

Selain itu, baik pengguna maupun administrator dapat keluar dari sistem dengan melakukan *logout* setelah menyelesaikan tugas atau interaksi mereka. Semua interaksi ini dilakukan melalui sistem yang berfungsi sebagai perantara antara pengguna dan administrator. Diagram ini memberikan gambaran menyeluruh mengenai alur kerja dan fungsi-fungsi yang ada dalam sistem layanan pengaduan *customer* memastikan setiap komponen dan aktor berperan sesuai dengan kebutuhan operasional sistem.

4.3 Perancangan Activity Diagram

Diagram aktivitas atau *Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak. Berikut adalah *Activity Diagram* pada penelitian ini:



Gambar 4. 2 *Activity Diagram Admin*

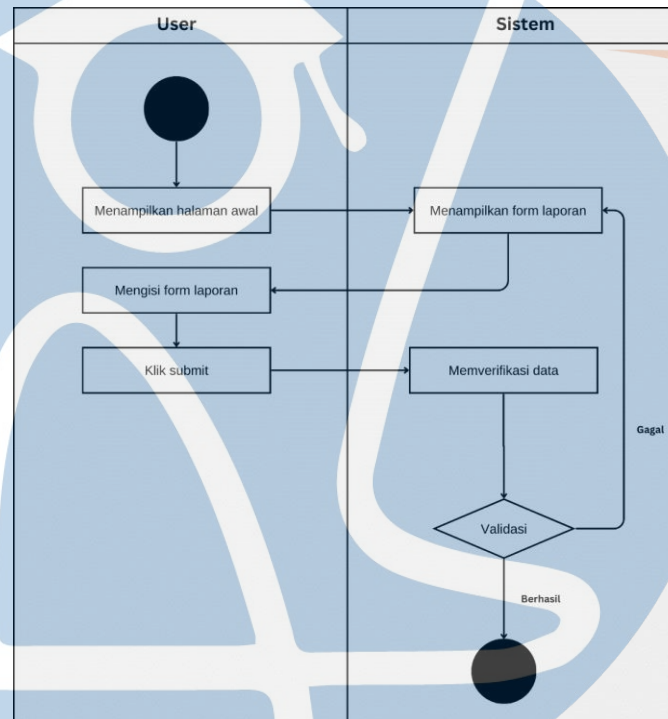
Diagram tersebut menggambarkan alur proses penanganan pengaduan dalam sistem E-Complaint INV78 yang melibatkan Administrator (Admin) dan sistem. Proses dimulai ketika Admin membuka Menu Pengaduan, di mana sistem kemudian menampilkan tabel data pengaduan yang berisi daftar pengaduan yang telah diajukan oleh pengguna. Admin kemudian memilih salah satu pengaduan dan mengklik untuk melihat detail pengaduan tersebut. Setelah itu, sistem menampilkan rincian lengkap dari pengaduan yang dipilih.

Setelah memeriksa detail pengaduan, Admin menentukan status pengaduan berdasarkan hasil pengecekan. Terdapat beberapa status yang dapat dipilih, yaitu "Diterima," "Diproses," dan "Selesai." Jika Admin memilih status "Diterima," sistem akan mencatat bahwa data baru telah masuk. Saat status "Diproses" dipilih, sistem mencatat bahwa pengaduan sedang dalam tahap pemrosesan. Akhirnya, jika status "Selesai" dipilih, sistem mencatat bahwa pengaduan telah selesai diproses.

The image features a large, semi-transparent watermark logo in the center. The logo consists of a blue circle containing a white stylized symbol that resembles a combination of the letters 'S' and 'T'. Below the circle, there are two orange curved lines that sweep upwards and outwards. At the bottom of the page, the text 'STT - NF' is displayed in a large, blue, serif font.

STT - NF

Alur ini memberikan gambaran yang jelas tentang tahapan yang harus dilalui oleh Admin dalam menangani pengaduan, mulai dari melihat daftar pengaduan, memeriksa rincian, menentukan status, hingga menyelesaikan pengaduan. Setiap langkah dalam proses ini dicatat oleh sistem untuk memastikan bahwa pengaduan ditangani secara efektif dan efisien, sesuai dengan prosedur yang telah ditetapkan.



Gambar 4. 3 Activity Diagram User

Dari diagram tersebut menggambarkan alur proses pelaporan pengaduan oleh pengguna (*User*) dalam sistem *E-Complaint INV78*, menggunakan istilah yang sesuai dengan Kamus Besar Bahasa Indonesia (KBBI). Proses dimulai dengan pengguna yang mengakses sistem dan menampilkan halaman awal. Sistem kemudian menampilkan formulir laporan yang harus diisi oleh pengguna. Pengguna mengisi formulir laporan dengan informasi yang diperlukan, lalu mengklik tombol "*submit*" untuk mengirim laporan tersebut.

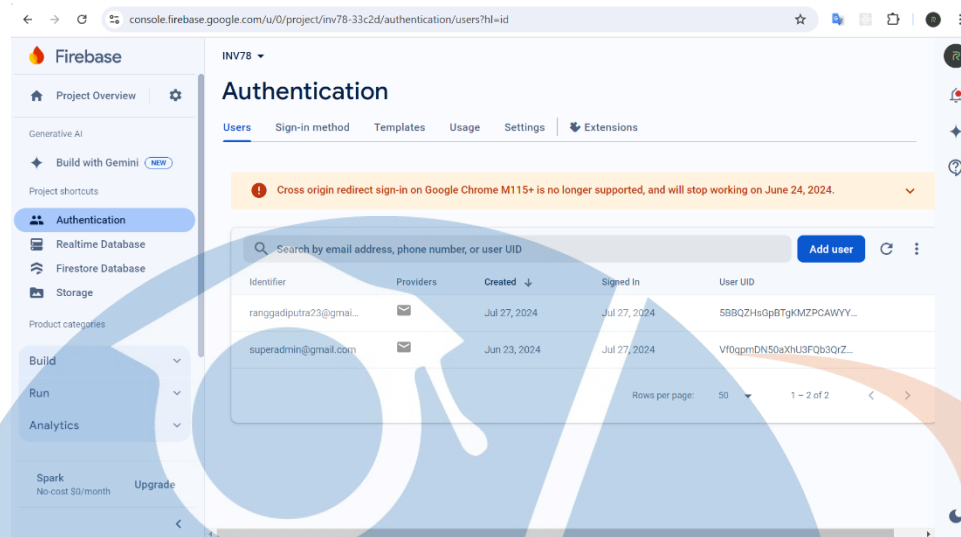
Setelah pengguna mengklik tombol "*submit*," sistem akan memverifikasi data yang dimasukkan oleh pengguna. Proses verifikasi ini melibatkan pengecekan kelengkapan dan keabsahan data yang di *input*. Jika data yang dimasukkan tidak valid atau terdapat kesalahan, sistem akan mengembalikan pengguna ke tahap pengisian formulir untuk memperbaiki kesalahan tersebut, menandakan proses telah gagal. Namun, jika data yang dimasukkan valid, sistem akan melanjutkan ke tahap validasi.

Pada tahap validasi, sistem memastikan bahwa semua data yang dimasukkan sudah benar dan sesuai dengan ketentuan yang berlaku. Jika validasi berhasil, proses pelaporan pengaduan dianggap selesai, dan laporan pengguna berhasil diterima oleh sistem. Diagram ini memberikan gambaran jelas mengenai langkah-langkah yang harus dilalui oleh pengguna dalam mengajukan laporan pengaduan, serta bagaimana sistem memproses dan memverifikasi data untuk memastikan keabsahan laporan tersebut.

4.4 Database Firebase

Pada perancangan *database* di sini penulis menggunakan *firebase* yang akan memudahkan penulis dalam pembuatan *table user* yang register, *table* pengaduan dan memudahkan dalam proses autentifikasi menggunakan email yang sudah didaftarkan pada halaman register. Berikut merupakan tampilan pada halaman *authentication user* pada *firebase*:

STT - NF

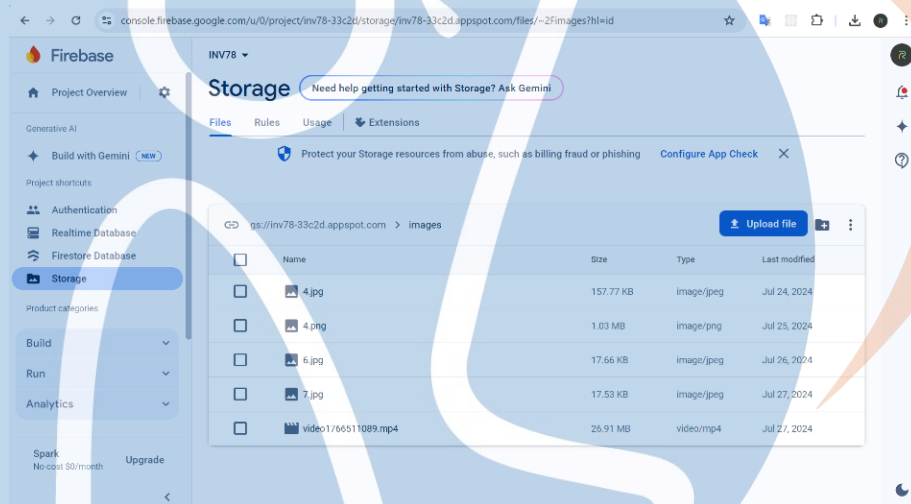


Gambar 4. 4 *Authentication User*

Firebase Authentication adalah layanan yang disediakan oleh *Google Firebase* untuk membantu pengembang aplikasi dalam mengelola proses autentikasi pengguna. Layanan ini memungkinkan pengguna untuk masuk ke aplikasi menggunakan berbagai metode autentikasi, termasuk *email* dan kata sandi, penyedia identitas eksternal seperti *Google*, *Facebook*, *Twitter*, dan *GitHub*, serta nomor telepon. *Firebase Authentication* juga mendukung autentikasi anonim, yang memungkinkan pengguna mengakses aplikasi tanpa perlu memberikan informasi identitas.

Dengan menggunakan *Firebase Authentication*, pengembang dapat dengan mudah mengintegrasikan fitur *login* dan pendaftaran ke dalam aplikasi mereka tanpa perlu membangun backend autentikasi dari awal. Layanan ini menyediakan API yang sederhana dan aman untuk menangani tugas-tugas seperti pendaftaran pengguna baru, otentikasi pengguna yang sudah ada, dan pengelolaan sesi pengguna. Selain itu, *Firebase Authentication* terintegrasi dengan baik dengan layanan *Firebase* lainnya, seperti *Firebase Firestore* dan *Firebase Realtime Database*, sehingga memudahkan pengembang untuk membangun aplikasi yang kaya fitur dan aman.

Layanan ini juga menyediakan fitur manajemen pengguna, yang memungkinkan pengembang untuk mengelola profil pengguna, memperbarui informasi pengguna, dan menangani proses *logout* dengan mudah. *Firebase Authentication* memastikan bahwa proses autentikasi dilakukan dengan aman, menggunakan standar enkripsi yang kuat untuk melindungi data pengguna. Oleh karena itu, *Firebase Authentication* adalah solusi yang kuat dan fleksibel untuk autentikasi pengguna, yang membantu pengembang aplikasi untuk menyediakan pengalaman login yang aman dan mudah bagi pengguna mereka.



Gambar 4. 5 *Firebase Storage*

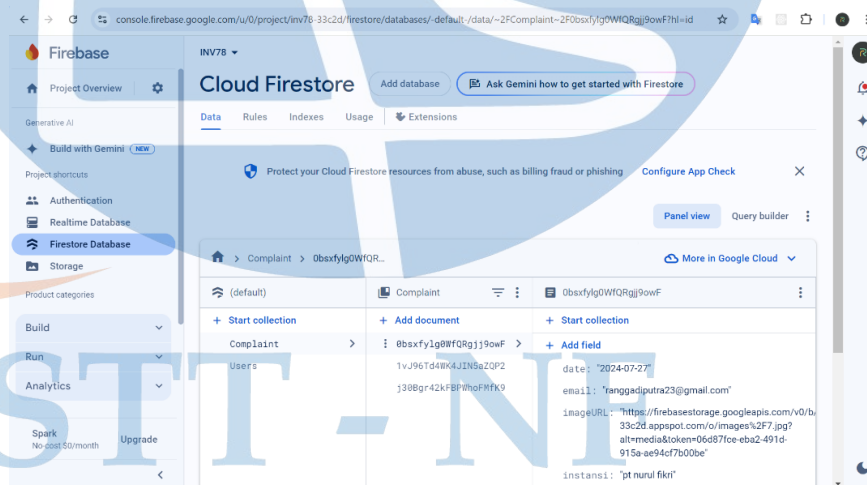
Firebase Storage adalah layanan penyimpanan cloud yang disediakan oleh Google Firebase untuk menyimpan dan mengelola konten buatan pengguna, seperti foto, video, dan file lainnya. Layanan ini dirancang untuk mengatasi kebutuhan aplikasi modern yang memerlukan penyimpanan dan akses data media secara aman dan efisien. Firebase Storage memanfaatkan Google Cloud Storage di backend, memberikan skalabilitas yang tinggi dan performa yang andal.

Dengan Firebase Storage, pengembang dapat dengan mudah mengunggah, mengunduh, dan berbagi file langsung dari aplikasi mereka. Layanan ini menyediakan SDK untuk berbagai platform, termasuk Android, iOS, dan Web, yang memudahkan integrasi penyimpanan file ke dalam aplikasi.

Firebase Storage juga mendukung fitur keamanan yang kuat, termasuk aturan keamanan berbasis *Firebase Authentication*, yang memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses atau memodifikasi *file* tertentu.

Firebase Storage juga menawarkan kemampuan untuk mengelola meta data file, seperti tipe file, ukuran, dan informasi lainnya, yang dapat digunakan untuk mengatur dan mengontrol akses ke file secara lebih efisien. Selain itu, *Firebase Storage* mendukung fitur *resumable uploads*, yang memungkinkan pengunggahan file yang besar secara bertahap, sehingga meminimalkan risiko kegagalan pengunggahan karena masalah konektivitas.

Secara keseluruhan, *Firebase Storage* menyediakan solusi penyimpanan yang kuat, fleksibel, dan mudah digunakan untuk aplikasi yang memerlukan pengelolaan konten media. Dengan integrasi yang mulus dengan layanan *Firebase* lainnya dan fitur keamanan yang kuat, *Firebase Storage* membantu pengembang membangun aplikasi yang dapat menangani file dan data media dengan aman dan efisien.



Gambar 4. 6 *Firestore Database*

Firebase Firestore adalah layanan basis data *NoSQL* yang disediakan oleh Google *Firebase*, dirancang untuk menyimpan, menyinkronkan, dan mengelola data aplikasi secara *real-time*. *Firestore* menawarkan struktur data

yang fleksibel dengan koleksi dan dokumen, memungkinkan pengembang untuk menyimpan data dalam format yang terorganisir dan mudah diakses. *Firestore* mendukung *query* yang kuat, yang memungkinkan pengambilan data berdasarkan berbagai kriteria tanpa mempengaruhi performa aplikasi.

Salah satu keunggulan utama *Firestore* adalah kemampuannya untuk menyinkronkan data secara *real-time*, yang berarti setiap perubahan data di server akan segera dipantulkan ke semua klien yang terhubung. Ini sangat berguna untuk aplikasi yang memerlukan pembaruan data secara langsung, seperti aplikasi obrolan, kolaborasi dokumen, dan *dashboard monitoring*. *Firestore* juga mendukung mode *offline*, memungkinkan aplikasi untuk terus berfungsi dan menyimpan data sementara bahkan ketika perangkat tidak memiliki koneksi internet. Data yang disimpan akan secara otomatis disinkronkan kembali ketika koneksi internet dipulihkan.

Firestore menawarkan integrasi yang erat dengan layanan *Firebase* lainnya, seperti *Firebase Authentication* untuk mengelola otentikasi pengguna dan *Firebase Storage* untuk penyimpanan file. Selain itu, *Firestore* menyediakan fitur keamanan yang canggih dengan aturan keamanan berbasis bahasa deklaratif, memungkinkan pengembang untuk menentukan kontrol akses data dengan granularitas tinggi.

Dengan *skalabilitas* yang tinggi dan performa yang andal, *Firestore* dapat menangani berbagai skenario penggunaan, dari aplikasi kecil hingga aplikasi besar dengan kebutuhan data yang kompleks. Secara keseluruhan, **Firebase Firestore** adalah solusi basis data yang kuat dan fleksibel untuk pengembangan aplikasi modern, memberikan kemampuan untuk menyimpan dan mengelola data dengan cara yang efisien, aman, dan mudah diakses.

4.5 *Prototype* Aplikasi

Prototype atau *prototipe* adalah sebuah metode dalam pengembangan produk dengan cara membuat rancangan, sampel, atau model dengan tujuan pengujian konsep atau proses kerja dari produk. *Prototype* sendiri bukanlah

produk final yang nantinya akan diedarkan. *Prototype* dibuat untuk kebutuhan awal *development software* dan untuk mengetahui apakah fitur dan fungsi dalam program berjalan sesuai dengan kebutuhan yang telah direncanakan. Sehingga pengembang produk dapat mengetahui kekurangan dan kesalahan lebih awal sebelum mengimplementasikan fitur lain ke dalam produk dan merilis produk. Berikut adalah *prototype* tampilan layanan pengaduan penulis



Gambar 4. 7 *Prototipe* Halaman Awal User


Pada halaman awal sistem layanan pengaduan *customer* pada PT Inovatif 78 sebelum mengisi *form* harus dilakukannya login terlebih dahulu jika sudah melakukan *login* akan langsung diarahkan ke halaman mengisi *form* pengaduan dan jika *user* belum mempunyai akun untuk *login* maka bisa dilakukan *register* terlebih dahulu.

Gambar 4. 8 *Prototipe Halaman Login User*

Pada gambar tersebut menunjukkan desain prototipe untuk halaman login user, pada halaman login user apabila user sudah mempunyai akun bisa langsung untuk mengisi inputan terkait email dan password untuk masuk kehalaman form pengaduan jika user belum memiliki akun bisa untuk registrasi untuk membuat akun, bisa untuk mengklik tulisan “Daftar sekarang” yang berwarna oren untuk.

Gambar 4.9 Prototipe Halaman Register User

Dari gambar prototipe di atas dapat dilihat untuk tampilan halaman registrasi user dalam membuat akun untuk bisa login, dari halaman registrasi tersebut user dapat memasukkan username, email, telepon, dan password, jika sudah mengisi form registrasi tersebut bisa langsung mengklik tombol submit.



The image shows a web form for reporting a complaint. At the top left is the logo for 'inovatif TUJUH DELAPAN' and a user profile icon. The main heading reads 'Hai, ada yang bisa kami bantu?'. Below this is a section titled 'Sampaikan Keluhan Anda'. The form contains several input fields: 'Ketik Judul Laporan Anda*', 'Pilih Tanggal Kejadian*' (with a calendar icon), 'Ketik Instansi Anda*', and 'Ketik Isi Laporan Anda*'. There is an 'UPLOAD LAMPIRAN' button and a 'Kirim' button at the bottom right. A large watermark 'STT - NF' is overlaid on the form. In the bottom right corner, there is a small text: 'Development @ inovatif 2024'.

Gambar 4. 10 *Prototipe* Halaman Pengaduan *User*

Disini merupakan tampilan untuk user mengisi *form* pengaduan jika user belum mengisi maka akan dilakukan validasi agar user tidak bisa

mengirimkan data kosong harus mengisi semua *form inputan* yang ada lalu dikirim jika data sudah terisi semua maka akan berhasil terkirim.

4.6 Hasil Penelitian

Berikut ini merupakan hasil analisis penelitian berdasarkan hasil wawancara yang telah penulis lakukan dengan narasumber. Penulis menyajikan hasil penelitian yang diperoleh dengan menggunakan hasil observasi, wawancara, dan dokumentasi. Dengan menjelaskan secara rinci hasil data tersebut, sehingga dapat ditarik suatu kesimpulan untuk pembuatan perancangan sistem aplikasi layanan pengaduan *customer* pada PT Inovatif 78 berbasis *web* menggunakan *React Js* dan *Firebase*.

Table 4. 1 Data Informan Karyawan Pt Inovatif 78

No	Nama	Umur	Jenis Kelamin
1	WS	26	Perempuan
2	RF	25	Laki-laki
3	RA	24	Laki-laki
Jumlah		3 informan	

Pada hasil wawancara ini penulis melakukan wawancara di sela-sela jam kerja dan jam istirahat yang di lakukan pada tanggal 06 Juni 2024 dan 07 Juni 2024, penulis mewawancarai 3 orang responden 1 perempuan karyawan PT Inovatif 78 dan 2 laki-laki tim teknis karyawan PT Inovatif 78.

Setelah dilakukannya wawancara dengan 3 narasumber, Berikut penulis dapat ambil kesimpulan dari wawancara yang telah dilakukan dari 3 narasumber tersebut sebagai berikut:

Layanan pengaduan atau *customer care* dianggap sangat penting oleh para narasumber untuk menangani masalah yang dialami oleh pengguna aplikasi atau situs *web*. Layanan ini memberikan umpan balik berharga bagi tim pengembang untuk memperbaiki kekurangan aplikasi dan meningkatkan

kepercayaan pengguna. Mereka mengharapkan fitur pengaduan yang lengkap, termasuk melalui email, chatbot, atau formulir web, serta sistem pelacakan untuk mengetahui status pengaduan. Perlindungan data pengguna dan notifikasi untuk memberitahu pengguna tentang tindak lanjut pengaduan juga dianggap esensial.

Untuk pengelolaan yang efisien, implementasi *chatbot* atau sistem respon otomatis dianjurkan agar layanan tersedia 24/7. Sistem tiket juga dianggap penting untuk melacak dan menyelesaikan pengaduan. Selain itu, data analitik sangat berguna untuk menganalisis masalah yang sering dikeluhkan dan mengurangi kesalahan yang sama di masa mendatang.

Penggunaan *React JS* untuk *frontend* dinilai memudahkan dan mempercepat pengembangan aplikasi, meskipun aplikasi pengaduan relatif sederhana. Penggunaan *React JS* juga bermanfaat untuk belajar dan mengikuti tren di dunia kerja. Sementara itu, *Firebase* cocok untuk pengembangan yang cepat dan efisien karena menyediakan berbagai layanan seperti *database*, *storage*, dan notifikasi secara *realtime*. *Firebase* dianggap sesuai untuk aplikasi skala kecil dan menengah, memudahkan proses *development* tanpa perlu membangun *query database* secara manual.

Para narasumber juga mengusulkan fitur tambahan seperti formulir pengaduan yang dilengkapi dengan *feedback* dari admin, fitur *chat real-time* (seperti *WhatsApp*) untuk respons cepat, fitur rating untuk memberikan *feedback* setelah pengaduan ditangani, dan kompresi gambar serta video untuk efisiensi penyimpanan dan pengiriman data. Secara keseluruhan, mereka menekankan pentingnya layanan pengaduan yang responsif, fitur yang *user-friendly*, dan pengelolaan yang efisien melalui penggunaan teknologi modern seperti *React JS* dan *Firebase*.

4.7 Implementasi Kode

Pada implementasi kode ini adalah proses di mana rancangan atau desain perangkat lunak diubah menjadi kode yang dapat dijalankan oleh komputer.

Pada tahapan ini penulis akan menjabarkan beberapa kode dari penulisan pemrograman yang telah penulis implementasikan di *React JS* :

```
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
import { getAuth } from "firebase/auth";
import { getFirestore } from "firebase/firestore";
import { getStorage } from "firebase/storage";

const firebaseConfig = {
  apiKey: "API_KEY_Firebase",
  authDomain: "inv78-33c2d.firebaseio.com",
  databaseURL: "https://inv78-33c2d-default-rtdb.firebaseio.com",
  projectId: "inv78-33c2d",
  storageBucket: "inv78-33c2d.appspot.com",
  messagingSenderId: "888268994871",
  appId: "1:888268994871:web:8ff4976ae15615271742b2",
  measurementId: "G-NXBLSSZ9B0"
};

const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
export const auth = getAuth();
export const db = getFirestore(app);
export const storage = getStorage(app);

export default app;
```

Pada penulisan kode pemrograman diatas merupakan kode untuk config atau terhubung ke firebase untuk bisa tersambung dengan firebase yang sudah

penulis daftarkan di firebase. Pada fungsi dari modul Firebase diimpor. `initializeApp` digunakan untuk menginisialisasi aplikasi Firebase, `getAnalytics` untuk layanan analitik, `getAuth` untuk autentikasi pengguna, `getFirestore` untuk database Firestore, dan `getStorage` untuk penyimpanan file. Sedangkan untuk Objek `firebaseConfig` berisi konfigurasi yang diperlukan untuk menghubungkan aplikasi dengan proyek Firebase yang spesifik. Ini termasuk `apiKey`, `authDomain`, `databaseURL`, `projectId`, dan lainnya, yang memastikan aplikasi Anda dapat berinteraksi dengan layanan Firebase yang tepat. Untuk Fungsi-fungsi yang export ini mengambil dan mengekspor instansi dari layanan Firebase yang berbeda: `auth` untuk autentikasi, `db` untuk database Firestore, dan `storage` untuk penyimpanan file. Ini memungkinkan komponen lain dalam aplikasi Anda untuk mengakses dan menggunakan layanan Firebase ini. Sedangkan “export default app” yang berfungsi untuk Aplikasi Firebase yang sudah diinisialisasi diekspor sebagai default. Ini memungkinkan bagian lain dari aplikasi untuk mengimpor dan menggunakan aplikasi Firebase yang telah dikonfigurasi.

```
const handleSubmit = async (e) => {
  e.preventDefault()
  try {
    const docRef = doc(db, 'Complaint', id)
    await updateDoc(docRef, formData)
    // alert('Data updated successfully!')
    Swal.fire({
      title: 'Success!',
      text: 'Data updated successfully! Redirecting...',
      icon: 'success',
      timer: 3000,
      timerProgressBar: true,
      showConfirmButton: false,
      willClose: () => {
```

```

        navigate('/pengaduan')
      }
    })
    console.log(sendEmailNotification(), 'email');
    navigate('/pengaduan')
  } catch (error) {
    console.error('Error updating document:', error)
  }
}
const sendEmailNotification = () => {
  const emailParams = {
    to_email: formData.email,
    complaint_title: formData.title,
    complaint_status: formData.status,
  }
  emailjs
    .send(
      import.meta.env.VITE_SERVICE_ID,
      import.meta.env.VITE_TEMPLATE_ID,
      emailParams,
      import.meta.env.VITE_USER_ID,
    )
    .then((response) => {
      console.log('Email successfully sent!', response.status,
response.text)
    })
    .catch((err) => {
      console.error('Failed to send email. Error: ', err)
    })
  }
}

```

Pada fungsi kode diatas tersebut penulis membuat 2 fungsi dalam pemrograman *React JS* dengan nama *variable* yang pertama *handleSubmit* yang berfungsi untuk Fungsi ini digunakan untuk menangani pengiriman formulir. Saat formulir dikirim (misalnya, setelah pengguna menekan tombol kirim), fungsi ini akan dipanggil. Fungsi ini mencegah perilaku default formulir dengan `e.preventDefault()`. Kemudian, menggunakan ID dokumen dan data formulir (*formData*), fungsi ini mengupdate dokumen dalam koleksi *Complaint* di *Firestore* dengan *updateDoc*. Setelah data diperbarui, notifikasi sukses ditampilkan menggunakan *SweetAlert2* (*Swal.fire*). Notifikasi ini memberikan umpan balik visual kepada pengguna dan secara otomatis mengarahkan mereka ke halaman `/pengaduan` setelah 3 detik. Selain itu, fungsi ini memanggil `sendEmailNotification()` untuk mengirim email pemberitahuan dan mencatat hasilnya di konsol. Akhirnya, jika semua langkah berhasil, pengguna diarahkan ke halaman `/pengaduan`. Sedangkan untuk penamaan *variable* *sendEmailNotification* adalah yang berfungsi untuk mengirim email notifikasi setelah data diperbarui. Fungsi ini menyiapkan parameter email menggunakan data dari formulir (*formData*) dan menggunakan `emailjs` untuk mengirim email. ID layanan, *template*, dan ID pengguna diambil dari variabel lingkungan. Jika email berhasil dikirim, pesan sukses dicatat di konsol; jika terjadi kesalahan, pesan kesalahan dicatat. Untuk mendapatkan `VITE_SERVICE_ID`, `VITE_TEMPLATE_ID` dan `VITE_USER_ID` penulis dapatkan dari *website* `emailjs` ketika sudah mendaftarkan akun.

4.8 Pengujian *BlackBox Testing*

Pada pembuatan atau merancang sebuah sistem jika sudah atau sedang di buat maka memerlukan pengujian untuk memastikan bahwa *software* atau aplikasi tersebut dapat berjalan sesuai dengan fungsional yang diharapkan. Pengujian *blackbox testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional

program[23]. Uji coba *blackbox* berusaha untuk menemukan kesalahan dalam beberapa kategori, di antaranya:

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan performa
5. Kesalahan *inisialisasi* dan terminasi

Untuk melakukan pengujian terhadap sistem, penulis melakukan pengujian dengan melakukan metode *blackbox testing* yang digunakan pada sistem yang telah dirancang. Sistem ini adalah metode pengujian yang digunakan untuk menguji fungsionalitas sistem yang telah dirancang[24]. Hasil dari pengujian dijelaskan pada tabel berikut:

Tabel 4. 2 Hasil Pengujian Sistem User

Sistem yang diuji	Luaran yang diharapkan	Hasil Pengujian	Persentase
<i>Home, button</i> tulisan <i>login</i>	Jika klik <i>button</i> maka akan mengarah ke halaman <i>login</i>	Berhasil	100%
<i>Register</i>	<i>User register</i> jika berhasil maka akan diarahkan ke halaman <i>login</i>	Berhasil	100%
<i>Login</i>	Jika otentikasi berhasil, maka akan diarahkan ke halaman pengaduan	Berhasil	100%
<i>Form</i> pengaduan	Jika sudah melakukan isi <i>form</i> pengaduan dan klik kirim maka akan muncul <i>pop up</i> berhasil	Berhasil	100%

Tabel 4. 3 Hasil Pengujian Sistem Admin

Sistem yang diuji	Luaran yang diharapkan	Hasil Pengujian	Persentase
<i>Login</i>	Jika otentikasi berhasil, maka akan diarahkan ke halaman pengaduan	Berhasil	100%
Menu pengaduan	Jika berhasil masuk ke <i>menu</i> maka akan tampil tabel pengaduan	Berhasil	100%
<i>Button</i> edit di tabel pengaduan	Ini akan bisa mengedit statusnya untuk diubah jadi diproses atau selesai	Berhasil	100%
<i>Button submit</i>	Jika berhasil maka akan muncul <i>popup</i> berhasil <i>update</i> data dan diarahkan kembali ke tabel pengaduan	Berhasil	100%
<i>Button logout</i>	Jika berhasil maka akan diarahkan ke halaman <i>login</i>	Berhasil	100%

4.9 Evaluasi

Berdasarkan hasil pengujian dari sistem yang telah dilakukan dengan metode *blackbox testing* yang di tunjukkan pada tabel di atas menunjukkan berhasil dari sistem yang telah dibuat. Evaluasi pengujian ini menunjukkan bahwa aplikasi bekerja dengan baik dan sesuai dengan spesifikasi yang telah ditetapkan oleh penulis, pada evaluasi ini penulis telah melakukan implementasi sistem menggunakan *React Js* dan *Firebase* telah berhasil dilakukan dan sistem yang telah dirancang telah berhasil dengan sesuai, akan

tetapi dari sistem yang telah dibuat perlu dilakukan tahap pengembangan lebih lanjut lagi.



STT - NF

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan terhadap perancangan sistem aplikasi layanan pengaduan *customer* pada PT Inovatif 78 berbasis web menggunakan *React Js* dan *Firebase* dapat disimpulkan bahwa:

1. Penulis telah berhasil melakukan pengembangan dan implementasi aplikasi layanan pengaduan customer berbasis website menggunakan React JS dan Firebase. Berikut adalah tahapan-tahapan dalam pengembangannya:
 - a. Analisis Kebutuhan: Mengidentifikasi kebutuhan pengguna dan menentukan fitur-fitur yang diperlukan dalam aplikasi layanan pengaduan.
 - b. Perancangan Sistem: Mendesain arsitektur aplikasi, termasuk antarmuka pengguna (UI) dan alur kerja sistem.
 - c. Pengembangan Frontend: Mengembangkan antarmuka pengguna menggunakan React JS untuk memastikan aplikasi responsif dan user-friendly.
 - d. Pengembangan Backend: Menggunakan Firebase untuk manajemen basis data, otentikasi, penyimpanan, dan notifikasi realtime.
 - e. Integrasi Sistem: Mengintegrasikan frontend dan backend untuk memastikan data dapat dikirim dan diterima dengan lancar antara antarmuka pengguna dan server.
 - f. Pengujian: Melakukan pengujian terhadap setiap fitur untuk memastikan aplikasi berjalan sesuai dengan yang diharapkan dan memperbaiki bug yang ditemukan.
 - g. Implementasi: Meluncurkan aplikasi yang sudah diuji dan memastikan semua fungsi bekerja dengan baik dalam lingkungan produksi.

- h. Pemeliharaan: Melakukan perawatan dan pembaruan aplikasi secara berkala untuk memastikan aplikasi tetap berfungsi optimal dan sesuai dengan kebutuhan pengguna.

Dengan melalui tahapan-tahapan ini, penulis berhasil mengembangkan dan mengimplementasikan aplikasi layanan pengaduan customer yang berbasis website menggunakan React JS dan Firebase.

2. Berdasarkan hasil pengujian sistem, seluruh fitur untuk pengguna dan admin menunjukkan hasil yang memuaskan dengan persentase keberhasilan 100%. Pada sistem pengguna, tombol login berhasil mengarahkan ke halaman login, proses registrasi mengarahkan ke halaman login setelah sukses, fitur login membawa pengguna ke halaman pengaduan setelah otentikasi, dan form pengaduan menampilkan pop-up konfirmasi setelah pengisian dan pengiriman form, semuanya dengan keberhasilan 100%. Untuk sistem admin, fitur login berhasil mengarahkan ke halaman pengaduan setelah otentikasi, menu pengaduan menampilkan tabel pengaduan, tombol edit di tabel memungkinkan perubahan status pengaduan, tombol submit menampilkan pop-up konfirmasi update data dan kembali ke tabel pengaduan, serta tombol logout mengarahkan kembali ke halaman login setelah logout, semuanya dengan keberhasilan 100%. Secara keseluruhan, sistem ini menunjukkan efektivitas dan reliabilitas yang tinggi dengan persentase keberhasilan 100%.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, saran penulis untuk penelitian selanjutnya adalah sebagai berikut:

1. Menggunakan *Framework* tertentu untuk *Backend*: Penelitian selanjutnya mungkin dapat mengimplementasikan sistem *backend* dengan menggunakan *framework* tertentu agar keamanan sistem dapat lebih terjaga.

2. Penambahan Fitur *Chat Real-Time*: Penelitian selanjutnya mungkin dapat menambahkan fitur *chat* secara *real-time* pada sistem layanan pengaduan untuk memfasilitasi interaksi antara pengguna dan admin.



STT - NF

DAFTAR PUSTAKA

- [1] N. * Melati and H. Noprisson², “Analisa Dan Perancangan Sistem Pengaduan Mahasiswa Berbasis Web (Studi Kasus: Universitas Mercu Buana Kranggan),” 2019. [Online]. Available: <https://jurnal.ikhafi.or.id/index.php/jusibi/185>
- [2] J. P. Tambusai *et al.*, “Digitalisasi Hubungan Masyarakat Berbasis Website di Sekolah.”
- [3] K. Komarudin and E. Nurmiati, “Perancangan Aplikasi Pengaduan Layanan Publik pada Kota Serang,” *Infomatek*, vol. 24, no. 2, pp. 103–112, Dec. 2022, doi: 10.23969/infomatek.v24i2.5533.
- [4] E. Bahari, D. Fraska, and U. Chotijah, “Perancangan Sistem Informasi Manajemen Tiket Keluhan Pelanggan PT. Jinde Grup Indonesia Berbasis Website Menggunakan Metode Waterfall,” *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 6, no. 5, p. 61121, 2023.
- [5] T. Rahmawati, A. Wirapraja, and D. F. Santoso, “PERANCANGAN SISTEM INFORMASI PENJUALAN SUKU CADANG MOTOR DAN FITUR KELUHAN PELANGGAN BERBASIS WEB PADA PT. FUBORU INDONESIA”, doi: 10.36595/misi.v5i2.
- [6] Adnan Buyung Nasution, B. F. E. Lubis, Nurul Amanda Khairani Lubis, and Friska Andriani, “Perancangan Sistem Pelaporan Keluhan Pelanggan Berbasis Website Menggunakan Metode Waterfall,” *Bulletin of Computer Science Research*, vol. 4, no. 1, pp. 40–49, Dec. 2023, doi: 10.47065/bulletincsr.v4i1.318.
- [7] A. Martika and D. Durbin Hutagalung, “Perancangan Sistem Customer Relationship Management Terhadap Pengaduan Nasabah Berbasis Web Menggunakan Model Extreme Programming”.
- [8] D. Mahdalena and L. Yulianti, “Perancangan Sistem Informasi Pelacakan Keluhan Pelanggan Berbasis Customer Relationship Management PDAM Bengkulu,” 2023.

- [9] A. Bhalla, S. Garg, and P. Singh, "PRESENT DAY WEB-DEVELOPMENT USING REACTJS," *International Research Journal of Engineering and Technology*, 2020, [Online]. Available: www.irjet.net
- [10] "PEMROGRAMAN ARTIKEL".
- [11] P. Marlina Ariansyah and K. Wijaya Sekolah Tinggi Manajemen Informatika dan Komputer Prabumulih, "Rancang Bangun Sistem Informasi Akademik Berbasis Web: Studi Kasus: SD Negeri 18 Tanah Abang," 2021.
- [12] S. Ramadan Siregar, "KLIK: Kajian Ilmiah Informatika dan Komputer Penerapan Metode Waterfall Dalam Pengembangan Sistem Informasi Masjid," *Media Online*, vol. 3, no. 1, pp. 26–32, 2022, [Online]. Available: <https://djournals.com/klik>
- [13] D. Supriadi, B. Susanto, U. Bina Sarana Informatika, and D. Direvisi Disetujui, "Perancangan Sistem Informasi Penggajian Karyawan Dengan metode Waterfall," *Journal Computer Science*, vol. 1, no. 1, 2022.
- [14] M. Nur Ichsanudin, M. Yusuf, S. Jurusan Rekayasa Sistem Komputer, J. Teknik Industri, I. AKPRIND Yogyakarta, and R. Artikel, "PENGUJIAN FUNGSIONAL PERANGKAT LUNAK SISTEM INFORMASI PERPUSTAKAAN DENGAN METODE BLACK BOX TESTING BAGI PEMULA INFO ARTIKEL ABSTRAK," vol. 1, no. 2, pp. 1–8, 2022, doi: 10.55123.
- [15] A. C. Praniffa, A. Syahri, F. Sandes, U. Fariha, Q. A. Giansyah, and M. L. Hamzah, "PENGUJIAN BLACK BOX DAN WHITE BOX SISTEM INFORMASI PARKIR BERBASIS WEB BLACK BOX AND WHITE BOX TESTING OF WEB-BASED PARKING INFORMATION SYSTEM," 2023.
- [16] N. Dwi Wulansari, G. I. Marthasari, and B. S. Wiyono, "Perancangan Sistem Informasi Pengaduan RSUD Dr. Soedarsono Pasuruan Berbasis Website Dengan Menggunakan Metode Waterfall," *REPOSITOR*, vol. 5, no. 3, pp. 747–758, 2023.
- [17] S. Hanifah, E. Syahnaz, P. Negeri Sambas, and K. Barat, "Rancang Bangun Aplikasi Pengaduan Pelanggan Berbasis Web Pada Perumda Air Minum Tirta

- Muare Ulakan Kabupaten Sambas,” vol. 3, 2021, [Online]. Available: <https://jurnal.ikhafi.or.id/index.php/jusibi>
- [18] “PENGADUAN CUSTOMER DI BEA CUKAI”.
- [19] D. Ismiyana Putri, J. Shadiq, M. Surya Apandi, and M. Ari Kuncoro, “SISTEM PENGOLAHAN DATA KELUHAN PELANGGAN BERBASIS WEB MENGGUNAKAN EXTREME PROGRAMMING METHOD”.
- [20] M. Nina Adlini, A. Hanifa Dinda, S. Yulinda, O. Chotimah, and S. Julia Merliyana, “METODE PENELITIAN KUALITATIF STUDI PUSTAKA,” 2022.
- [21] E. S. Nugraha, A. R. Padri, O. Nurdiawan, A. Faqih, and S. Anwar, “Implementasi Aplikasi Pengaduan Masyarakat Berbasis Android Pada Gedung DPRD,” *JURIKOM (Jurnal Riset Komputer)*, vol. 8, no. 6, p. 360, Dec. 2021, doi: 10.30865/jurikom.v8i6.3679.
- [22] A. Sri Wahyuningsih, S. Silaban, and K. Kunci, “HUMANTECH JURNAL ILMIAH MULTI DISIPLIN INDONESIA IMPLEMENTASI SISTEM INFORMASI LAYANAN PENGADUAN MASYARAKAT BERBASIS WEB PADA DESA KARYASARI MENGGUNAKAN METODE PROTOTYPE”.
- [23] A. Utomo, Y. Sutanto, E. Tiningrum, and E. M. Susilowati, “PENGUJIAN APLIKASI TRANSAKSI PERDAGANGAN MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS,” *Jurnal Bisnis Terapan*, vol. 4, no. 2, pp. 133–140, Dec. 2020, doi: 10.24123/jbt.v4i2.2170.
- [24] J. Panjaitan and A. F. Pakpahan, “Perancangan Sistem E-Reporting Menggunakan ReactJS dan Firebase,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, Apr. 2021, doi: 10.28932/jutisi.v7i1.3098.

LAMPIRAN

Narasumber: WS, Perempuan



Pertanyaan:

1. Apa tanggapan Anda terkait layanan pengaduan?

Jawaban:

“Kalo menurut saya, saya selaku tim prodak pada pt inovatif 78 ini ee.. layanan pengaduan atau customer care ini tu pada apps atau website itu sangat penting, karena adanya layanan customer care ini membuat user itu dapat melakukan laporan jika terjadi problem pada apps atau website sehingga hal tersebut juga menjadi feedback bagi kita ini sebagai pembuat apps atau si tim developnya ini dalam memahami kekurangan atau kelemahan pada apps tersebut dan ee.. hal ini juga itu dapat membuat tim develop untuk dapat menindak lanjuti hal kedepannya itu agar tidak terjadinya lagi ee.. bug atau kendala-kendala lainnya pada apps tersebut. Terus dengan adanya layanan customer care ini juga yang responsive dapat membuat user juga merasa keluhan mereka dapat didengar sehingga dapat meningkatkan kepercayaan user ini terhadap apps yang kita buat sehingga user bisa ajakan ee.. mengajak teman-temennya ini atau siapapun yang dia kenal untuk mendownload apps ini, ‘eh apps ini bagus loh ternyata bisa memudahkan kita contohnya ee.. untuk apa berbelanja kita memiliki ecommerce’ misalnya kita bikin apps ecommerce nih nah tapi dalam ecommerce itu tuh ternyata appsnya bagus jadi sehingga user A ini bisa

mengajak ke user-user lain untuk mendownload apps kita dan karena juga kalo misalnya ada kendala ee.. user itu merasa dirinya bisa apa namanya.. keluhannya didengar “

Pertanyaan:

2. Fitur apa yang diharapkan dalam sistem layanan pengaduan pada customer?

Jawaban:

“oh kalo fitur apa aja si yang biasanya yang harus ada pada layanan customer care itu ya, kalo menurut saya fitur-fitur atau item yang harusnya ada pada sistem customer care ini itu agar mudah dipahami atau digunakan oleh user itu tuh seperti satu ee.. misalkan contoh kita user bisa melakukan pengaduan melalui email, whatsapp, atau chatbot ya yang biasanya itu otomatis ya ataupun bisa melakukan pengaduan melalui formulir web selanjutnya ee.. jika pengaduan melalui formulir web harus ada tracking sistem jadi kita bisa tahu nih si user selaku user bisa tau pengaduan kita itu udah sudah diproses dari tahap sudah da di tahap ee.. sedang diriview niholeh pihak apa pembuat appsnya terus juga pada fitur customer care ini juga apa namanya si pembuat apps itu harus melakukan adanya suatu perlindungan dari data user data user itu tidak tersebar kemana-mana dan juga ee.. layanan customer care yang baik itu adanya notifikasi misalnya notifikasi by email atau pada apps tersebut sehingga jadi user tahu nih bahwa pengaduannya udah di tindak lanjuti”

Pertanyaan:

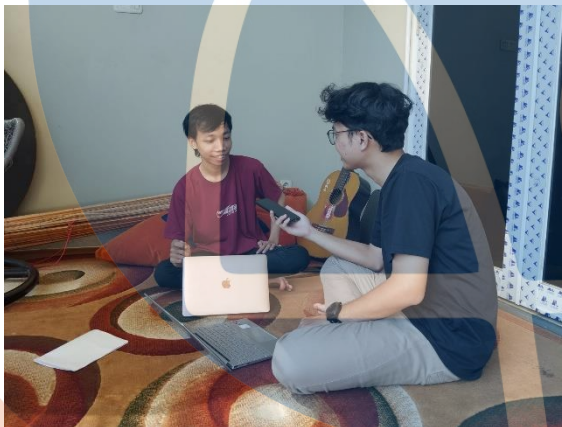
3. Menurut anda bagaimana pengelolaan sistem layanan customer secara efisien?

Jawaban:

“oh kalo menurut saya si ee.. layanan customer care yang efisien itu dia harus apa namanya dapat mengimplementasikan chatbot atau sistem respon otomatis untuk menangani pertanyaan-pertanyaan dasar ya agar jadi kalo ada user mau tanya hal-hal yang mendasar dengan adanya chatbot tersebut ee.. apa namanya bisa 1x24 jam sistemnya jadi user tidak harus tunggu apa namanya ee.. si customernya itu untuk

melakukan apa namanya menjawab pertanyaan itu dengan danya chatbotkan biasanya lebih cepet ya dan juga ee.. biar lebih efisien itu biasanya si menggunakan sistem tiket ya untuk melacak dan menyelesaikan pengaduan pelanggan jadi dengan adanya tiket itu dari phak apa namanya pembuat apps kita juga bisa tau nih pengaduan pelanggan itu juga udah sampai mana mungkin bisa ditrack dari tiket itu kalau misal ada yang belum dilayanikita bisa tahu tuh ohh ini belum dilayani terus juga harus ada kaya ini ya data analitik ini itu menganalisis masalah apa saja yang lagi tren atau yang biasa user keluhkan contohnya ohh ini ini gak bisa di klik buttonnya nih ohh berarti kedepannya lagi nanti untuk pembuat apps gua ini gak bisa di klik itu rata-rata karena apa jadi kedepannya lagi untuk meminimalisir lah kesalahan yang sama seperti itu”

Narasumber: RF, Laki-laki



Pertanyaan:

1. Apa tanggapan anda jika dalam pembuatan aplikasi layanan pengaduan ini menggunakan react js untuk frontend nya?

Jawaban:

“tanggapan saya tentang pembuatan ee.. menggunakan react js sebagai frontendnya ya nah disini kan react js itu adalah framwork yang mana ee.. itu adalah kerangka kerja ya yang membuat programmer sendiri menjadi ee.. mudah mengerjakannya dan juga cepet ya nah sebenarnya kalo aplikasi simpel si engga pake framework si juga

bisa si kalo dilihat dari pengaduan ini kan ee.. Cuma mengadukan aja kan cuman si bagus si ee.. kita bisa tau framework yang framework-framework karena di dunia kerja sendiri kan sudah banyak kan orang yang engga cari native ya jadi sekalian belajar juga disini nah keuntungan laennya menggunakan framework itu kita sudah mudah ya dalam pembuatan misalkan lifecycle nyakalau state manajemennya itu kan sudah dirancang yang kita gunakan nah mungkin si bagus-bagus saja si untuk menggunakan react js ini jadi gak ada masalah si”

Pertanyaan:

2. Apa tanggapan anda jika dalam pembuatan aplikasi layanan pengaduan ini menggunakan farebase untuk backendnya?

Jawaban:

“oke ee.. firebase itu sebenarnya hampir sama kaya framework dia dibangun sudah ada prodaknya kan semisal kaya database, storage, ee.. realtime notifikasi tinggal kita gunakan saja apa yang kita mau apa yang mau kita bangun kan nah mungkin disini yang digunakan dari firebasenya mungkin databasenya ya, sebenarnya si dengan menggunakan firebasejuga mempercepat dalam pengembangan development kalau misalkan kita menggunakan mysql kan itu kan termasuk sql ya kita harus bangun dulu querynya kalau firebase sendiri kan sudah No. sql kan jadi kita tinggal pikir saja alurnya tinggal diinsert doang kan ini juga membantu si dalam development kurasa si cocok saja menggunakan firebase apalagi firebase itu sudah didesain untuk ee.. database yang dari bawah dan menengah si cukup saja si oke mungkin itu saja si”

Pertanyaan:

3. Fitur apa yang diharapkan dalam sistem layanan pengaduan pada customer?

Jawaban:

“fitur unggulannya ya karena ini adalah aplikasi pengaduan customer ya berarti mungkin ee.. yang pertama si form dalam pengaduannya itu ada form untuk mengadukan ee.. website baru ada ada halaman ee.. customer service atau apa lah itu si yang hal ee.. yang biasa ya maksudnya simpelnya ya gitu doang si baru ada

feedbacknya dari admin atau pengurus cuman mungkin agak lebih bagus lagi ada ngechat begitu misal kaya wa jadi secara realtime dia langsung dapat begitu notifnya langsung merespon kurasa si kalau pengaduan ini gak terlalu banyak paling ya ada form pengaduan abis itu mungkin kalau bisa si feedbacknya ke email atau langsung ke nomor wa begitu itu saja si “

Narasumber: RA, Laki-laki



Pertanyaan:

1. Apa tanggapan anda jika dalam pembuatan aplikasi layanan pengaduan ini menggunakan react js untuk front end nya?

Jawaban:

“kalau dari teknis react js ini sudah cukup cepet si diakses ya karna dia basicnya javascript juga kan bagus kan react js mengikuti perkembangan juga”

Pertanyaan:

2. Apa tanggapan anda jika dalam pembuatan aplikasi layanan pengaduan ini menggunakan farebase untuk backendnya?

Jawaban:

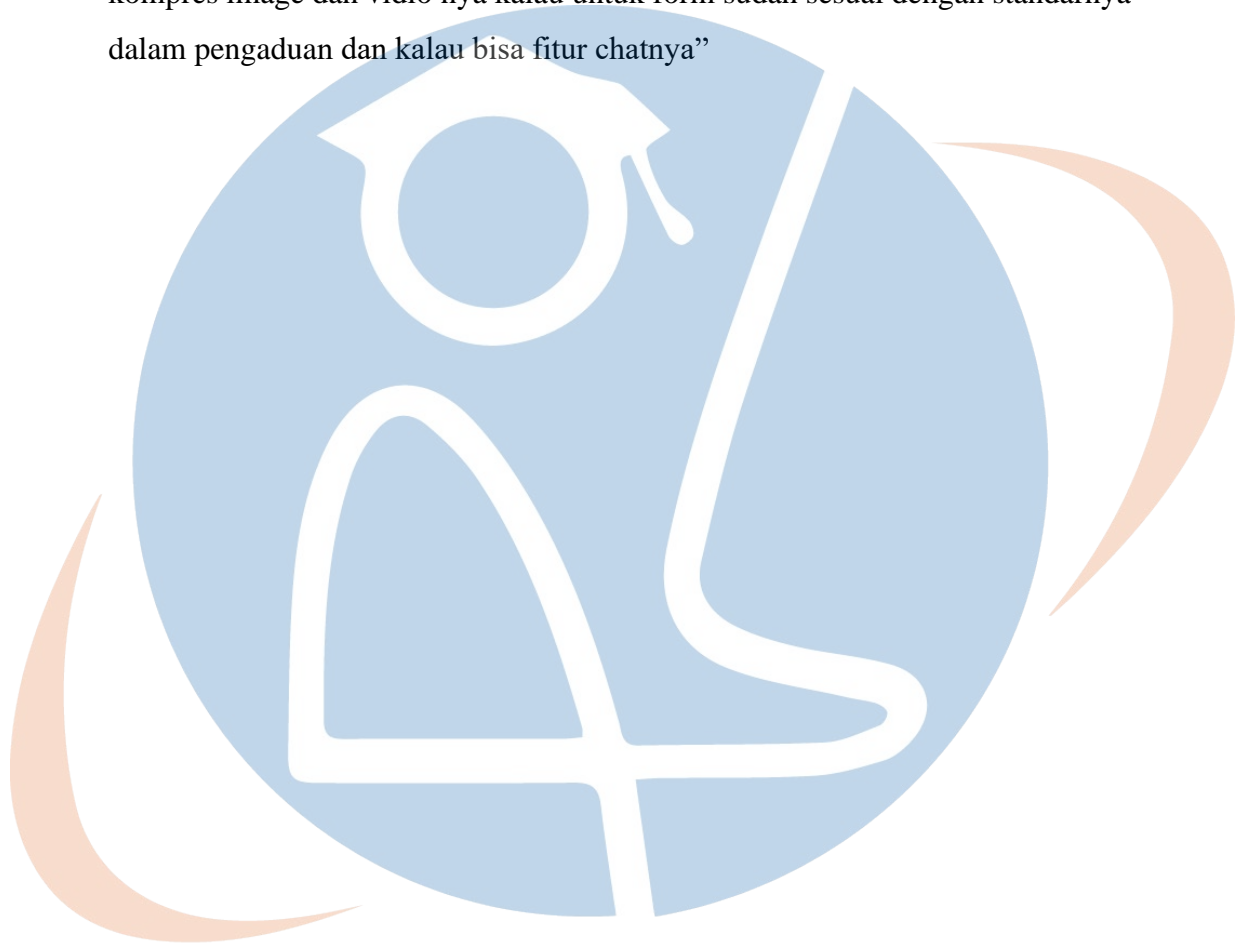
“ee.. mungkin kalau dalam skala kecil customer care ini aplikasinya ee.. bisa kehandle si seharusnya nah harusnya si bisa pakai firebase di backendnya”

Pertanyaan:

3. Fitur apa yang diharapkan dalam sistem layanan pengaduan pada customer?

Jawaban:

“ee.. rating jika sudah selesai kasih rating begitu untuk feedbacknya ketika sudah direspon oleh admin ohh jadi ini untuk general kalau untuk saran dari gua untuk di kompres image dan vidio nya kalau untuk form sudah sesuai dengan standarnya dalam pengaduan dan kalau bisa fitur chatnya”



STT - NF