



Tersedia online di : <http://bit.ly/InfoTekJar>

InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan

ISSN (Cetak) 2540-7597 | ISSN (Online) 2540-7600



Software & Development

Startup E-Complaint Dengan Integrasi API Untuk Akses Tanpa Unduh Pada Online Shop

Indah Nur Setiani, Hana Fitri Anisa N W, Jeni Sundari

Universitas Nusa Mandiri Jatiwaringin Jl. Jatiwaringin No. 2, Cipinang Melayu 13620, Indonesia

ARTICLE INFORMATION

Received:
Revised:
Available online:

KEYWORDS

Keluhan pelanggan, Teknologi API, Metode *waterfall*, *Mobile application*

CORRESPONDENCE

Phone: +62 89510399216
E-mail: Indahnrstni04@gmail.com

ABSTRACT

The development of the times that demands everything to be more efficient and effective has prompted various companies to develop their online systems. Along with global trends, online shopping in Indonesia is also becoming more popular. Customer satisfaction in online shopping is a determining factor in their decision to make a repeat purchase. Dissatisfaction can increase customer turnover and the cost of acquiring new customers. A common problem faced by businesses is the quality of customer service, especially in handling complaints manually which is no longer relevant in the midst of technological developments. This research aims to create an online customer complaint system that facilitates the handling of complaints efficiently, and simplifies the administration of reports and documentation. The system uses API technology to ensure an effective interface for admins and customers. The Waterfall method is used in the development of this application to minimize bugs and detect errors. The results of the study show that a web-based complaint application that is connected to a mobile application in real-time can provide ease of access and increase customer trust in online shop services.

INTRODUCTION

Perkembangan zaman yang menuntut efisiensi telah mendorong perusahaan untuk mengembangkan sistem *online*. Seiring dengan tren global, belanja *online* di Indonesia semakin populer. Konsumen berbelanja *online* untuk mendapatkan penawaran harga terbaik dan menghemat waktu. Kepuasan pelanggan dalam berbelanja *online* menjadi faktor penentu dalam keputusan pembelian ulang mereka [1]. Saat pelanggan mengalami pengalaman buruk atau tidak menyenangkan saat berbelanja *online*, mereka akan merasa tidak puas dan mengajukan keluhan [1]. Ketidakpuasan pelanggan dapat meningkatkan kemungkinan mereka beralih ke produk atau merek lain, yang pada akhirnya meningkatkan tingkat pergantian pelanggan [1].

L. Guwiry, pemilik *loveable_id*, menyatakan bahwa masalah umum yang dihadapi adalah kualitas layanan pelanggan. Pelanggan sering mengalami kesulitan dalam mengajukan keluhan dan lambatnya tanggapan dari *marketplace*. Komunikasi yang tidak efektif menyebabkan kesalahpahaman dan ketidakpuasan pelanggan. Oleh karena itu, diperlukan sistem keluhan pelanggan yang lebih mudah diakses dan dioptimalkan.

Kepercayaan (*trust*) sangat berpengaruh terhadap minat menggunakan suatu produk atau jasa seperti yang disebutkan oleh (Flavian, C., Guinaliu, M., & Gurrea, R. 2005) bahwa Studi menyatakan ketika kepercayaan tercipta maka kemauan untuk membeli dari pemasok *online* akan semakin besar dan membuat konsumen menjadi puas sehingga menimbulkan kesetiaan [2]. Penanganan komplain pelanggan secara manual dianggap sudah tidak relevan dengan di tengah kondisi teknologi informasi yang berkembang pesat ini [3]. *Application Programming Interface* (API) merupakan teknologi antarmuka yang dibangun oleh pengembang sistem supaya sebagian atau keseluruhan fungsi sistem dapat diakses secara bersamaan dengan baik [4]. Beberapa penelitian tentang penerapan *web service* metode REST API telah digunakan [5]. Penerapan *web service* pada integrasi aplikasi bank sampah berbasis mobile dan *website* [5]. Metode *Waterfall*. Metode ini dipilih karena dalam pengembangannya hanya membutuhkan tim yang sedikit, Selain itu metode ini juga dapat meminimalisir kecacatan/*bug*, karena pada proses pengembangannya dibagi menjadi beberapa tahapan dan dilakukan pengujian [6]. Sehingga jika terjadi *error* dapat terdeteksi dan dapat segera diperbaiki [6]. Maka dari itu adanya aplikasi komplain ini bisa akses tanpa unduh aplikasi memungkinkan pelanggan untuk dengan cepat dan efisien

mengajukan keluhan di *web application* yang terhubung oleh admin di *application mobile* secara *real time* tanpa kebingungan dan dapat menyediakan kemudahan akses.

Berdasarkan penjelasan latar belakang diatas, maka penulis melakukan penelitian yang berjudul “*Startup E-Complaint Dengan Integrasi API Untuk Akses Tanpa Unduh Pada Online Shop*”.

METHOD

Penggunaan Metode *Waterfall* dalam pengembangan aplikasi komplain dengan integrasi API akses tanpa unduh aplikasi, dengan dasar menggunakan RUP (*Rational Unified Process*), dapat dilakukan dengan mengadaptasi langkah-langkah RUP ke dalam pendekatan Metode *Waterfall*. Berikut adalah langkah-langkah yang dapat diambil :

1. Pengembangan awal
Mengidentifikasi kebutuhan pengguna untuk aplikasi komplain dan fungsionalitas API dari *web* ke aplikasi yang dapat digunakan secara realtime, agar pelanggan dapat mengakses tanpa unduh aplikasi.
- Metode *Waterfall* yang dapat dikembangkan secara bertahap berdasarkan kebutuhan pengguna dan ketergantungan antar-fitur dengan tujuan untuk mencapai fungsi minimal yang dapat digunakan (*Minimum Usable Subset*).
3. Implementasi dan Integrasi
Mulai dengan mengembangkan Metode *Waterfall* pertama, yang mungkin mencakup fitur dasar seperti pengajuan keluhan, *e-ticket* atau status komplain. Lalu integrasikan aplikasi dan web dengan API yang diperlukan.
4. Pengujian dan validasi
Setelah semua sudah diimplementasikan, lalu akan ada pengujian integrasi untuk memverifikasi kinerja aplikasi dan *web* dengan integrasi API. Uji aplikasi secara menyeluruh untuk memastikan tidak ada bug atau masalah yang signifikan.

Mengintegrasikan aplikasi komplain pada online shop dengan menggunakan *link direct* ke *website* yang bisa di gunakan *customer* dimana *web application* ini terhubung dengan *mobile application* yang bisa digunakan admin memungkinkan akses yang lebih mudah dan cepat bagi pelanggan. Dengan menggunakan API, informasi seperti riwayat komplain, *live chat* keluhan dan *e-ticket* komplain dapat diakses langsung dari aplikasi komplain, mempermudah identifikasi dan penyelesaian masalah.

Konsep ini memungkinkan pelanggan untuk mengakses fitur aplikasi melalui *browser* tanpa perlu mengunduh aplikasi secara khusus. Ini memperluas jangkauan pengguna dan mengurangi hambatan untuk memanfaatkan layanan pelanggan.

Customer bisa mengajukan keluhan melalui *web* atau *mobile application* yang sudah terhubung melalui integrasi API. selain itu *customer* juga dapat menginfokan admin apabila ingin menambahkan pesanan atau mengganti produk yang akan dipilih melalui fitur *live chat*. *Customer* dapat mendaftar jika belum mempunyai akun, lalu setelahnya *customer* bisa login dengan menggunakan *username* dan *password* yang sudah di daftarkan pada platform komplain kami.

Pengujian Aplikasi

2 Indah Nur Setiani

Dalam pengujian perangkat lunak ini menggunakan *black box testing* dengan alasan pembuat hanya menguji apakah fungsionalitas dalam aplikasi sudah sesuai dengan yang diharapkan. *Black box Testing* merupakan metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja [7]. Pengetahuan khusus dari kode aplikasi/struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan [7]. Menggunakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus [7]. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional [7]. Perancang uji memilih input yang valid dan tidak valid, lalu menentukan output yang benar [7]. Dengan tahapan yang dilakukan penulis yaitu analisis sistem, desain, pengodean perangkat lunak, pengujian dan implementasi program [7].

Peralatan Pendukung

Peralatan pendukung yang digunakan penulis dalam merancang aplikasi yaitu:

Mobile Application:

1. *Visual Studio Code*: Editor kode sumber yang ringan namun kuat, yang dikembangkan oleh Microsoft. *Visual Studio Code* ini sangat fleksibel dan mendukung untuk berbagai Bahasa pemrograman dan *framework*.
2. Bahasa Dart: Dart adalah bahasa pemrograman yang berorientasi objek (object-oriented), yang berarti semuanya dalam Dart adalah objek. Dart mendukung pemrograman asinkron melalui fitur seperti *async* dan *await*, yang memudahkan penanganan operasi asinkron.
3. React Js: Pustaka (library) *JavaScript* yang digunakan untuk membangun antarmuka pengguna (*user interface*), terutama untuk aplikasi web yang dinamis dengan data yang sering berubah. Pustaka *JavaScript* ini menggunakan pendekatan berbasis komponen, dimana UI dibagi menjadi komponen-komponen yang independent dan dapat digunakan kembali. Konsep utama Pustaka *JavaScript* adalah *Document Object Model* (DOM) *virtual*.
4. Flutter: *Framework UI Open-Source* yang dikembangkan oleh Google untuk membangun aplikasi native untuk *mobile*, *web*, dan desktop dari basis kode tunggal.

Web Application:

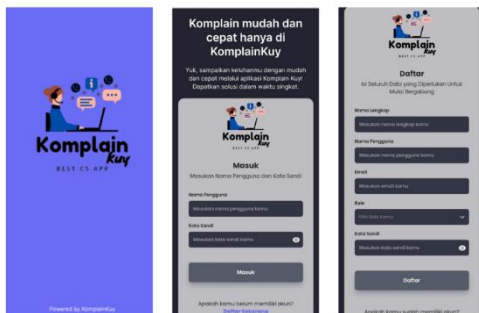
Adapun kesamaan menggunakan *Visual Studio Code* , HTML , CSS , Java.

1. *Database MySQL*: Sistem manajemen basis data relasional yang banyak digunakan untuk mengelola data dalam aplikasi web.
2. *Java Script*: *JavaScript* digunakan untuk membuat halaman *web* lebih interaktif dan dinamis. *JavaScript* adalah bahasa yang serbaguna dan dapat dijalankan di berbagai *platform* dan perangkat. Dengan menggunakan *JavaScript*, pengembang dapat menulis kode yang berjalan di sisi *klien* dan *server*.

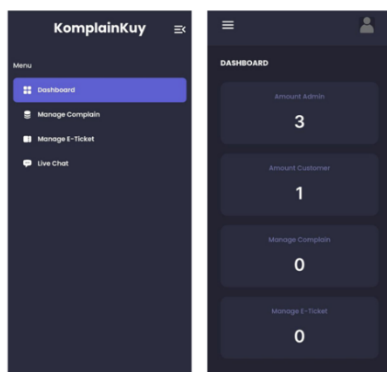
3. REST API (*Representational State Transfer Application Programming Interface*): Standar untuk pengembangan API yang memungkinkan interaksi antara *klien* dan *server* melalui protokol HTTP[8]. Dengan adanya API ini, maka memudahkan *programmer* untuk “membongkar” suatu *software*, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain [8]. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan *programmer* menggunakan sistem function. Proses ini dikelola melalui sistem operasi. Menggunakan metode HTTP standar seperti *GET*, *POST*, *PUT*, *DELETE* untuk operasi *Create*, *Read*, *Update*, dan *Delete* (CRUD Operations).
4. Github: Tools Github ini memudahkan tim untuk bekerja sama dalam proyek yang sama, meskipun berada di lokasi yang berbeda sehingga pengembang dapat mengotomatisasi *build*, pengujian, dan *deployment*.

Tinjauan Startup (Bisnis Digital)

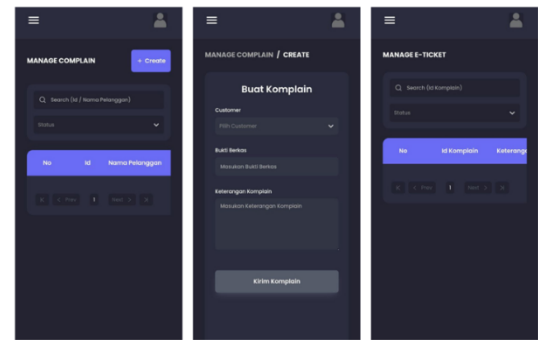
KomplainKuy adalah sebuah *startup e-complaint* yang menyediakan aplikasi komplain untuk *platform online shop* memungkinkan pelanggan untuk mengajukan keluhan tanpa harus mengunduh aplikasi terpisah melalui integrasi API yang mudah digunakan yang bertujuan untuk mengoptimalkan penanganan keluhan pelanggan, sehingga meningkatkan kepuasan pelanggan dan reputasi toko. Pada tahap pembangunan startup ini menggunakan metode *Design Thinking* yang terdiri dari *Empathize*, *Define*, *Ideate*, *Prototype*, dan *Test*. Berikut adalah tampilan *prototype* dari aplikasi yang di buat.



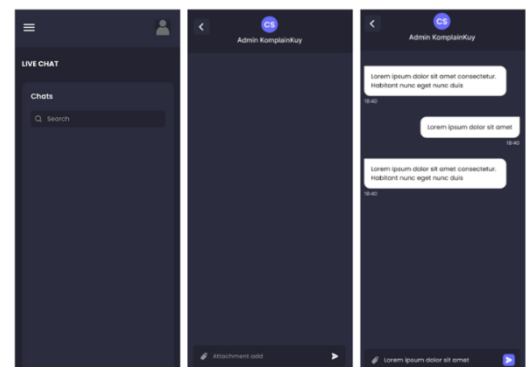
Gambar 1. Login Aplikasi KomplainKuy



Gambar 2. Home Aplikasi KomplainKuy



Gambar 3. Manage Complain & View E-Ticket Aplikasi KomplainKuy

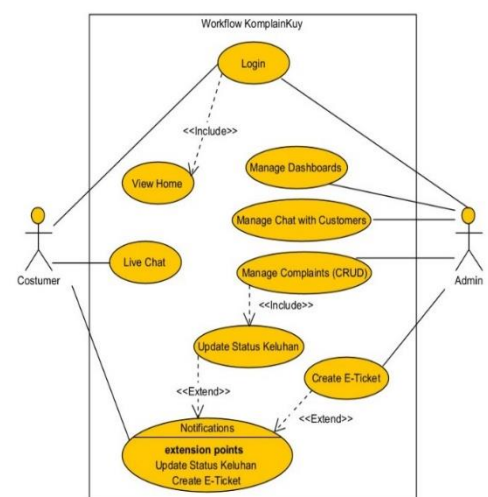


Gambar 4. Live Chat Aplikasi KomplainKuy

Analisis Kebutuhan Aplikasi
Analisa kebutuhan pada aplikasi “KomplainKuy” sebagai berikut:

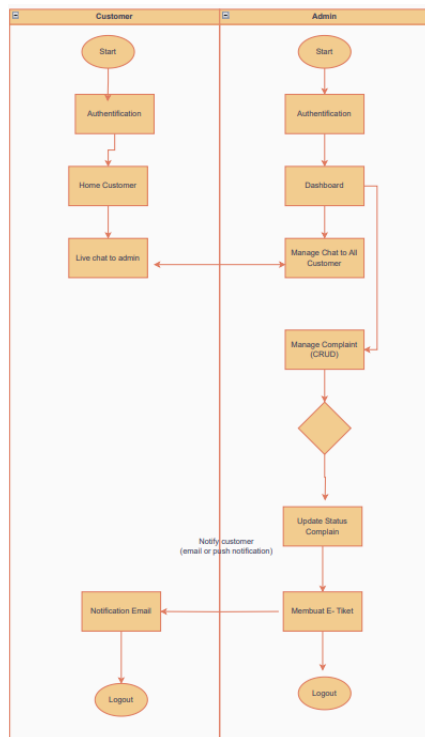
1. Pengguna dapat mengajukan keluhan.
2. Pengguna dapat melihat status keluhan (*E-Ticket*).
3. Pengguna dapat melihat riwayat keluhan.
4. Pengguna dapat menggunakan fitur *live chat* yang terhubung dengan admin.

Rancangan Use Case Diagram



Gambar 5. Use Case KomplainKuy

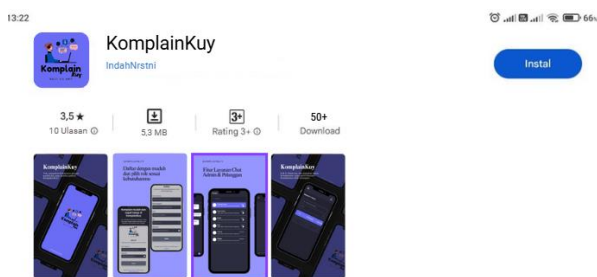
Rancangan Diagram Aktivitas



Gambar 5. Diagram Aktivitas KomplainKuy

RESULTS AND DISCUSSION

Hasil implementasi dari penelitian ini sudah di terbitkan aplikasi kedalam *google play store*. Berikut hasil implementasinya :



Gambar 6. Tampilan Aplikasi KomplainKuy di *Google Play Store*

CONCLUSIONS

Penelitian ini berfokus pada optimalisasi penanganan keluhan pelanggan melalui pengembangan aplikasi KomplainKuy, sebuah solusi *e-complaint* yang terintegrasi dengan API antara aplikasi mobile dengan *website*, memungkinkan akses tanpa perlu mengunduh aplikasi tambahan pada platform online shop. Hasil penelitian menunjukkan bahwa keluhan pelanggan harus di optimalkan karena kepuasan pelanggan dalam berbelanja online menjadi penentu keputusan untuk melakukan pembelian ulang. Jika pelanggan merasa tidak puas, mereka lebih mungkin berpindah ke produk atau merek lain. Selain itu, mengoptimalkan pengelolaan keluhan dengan sistem *online* dapat membuat proses penanganan lebih cepat dan efektif, karena informasi dapat diakses oleh pihak yang bertanggung jawab secara bersamaan.

Antarmuka aplikasi komplain yang diakses melalui API dapat memberikan pengalaman pengguna yang efektif baik untuk admin maupun pelanggan. Hal Ini memastikan bahwa sistem mudah digunakan dan efisien dalam penanganan keluhan, serta memudahkan *administrator* dalam membuat laporan. Kepercayaan pelanggan terhadap bisnis *online* sangat memengaruhi minat dan kesetiaan pelanggan terhadap produk atau jasa yang ditawarkan karena ketika kepercayaan tercipta, kemauan untuk membeli dari pemasok online akan semakin besar. Ini karena kepercayaan membuat konsumen merasa aman dan nyaman dalam bertransaksi.

ACKNOWLEDGMENT

Untuk meningkatkan kualitas aplikasi beberapa saran dapat dipertimbangkan:

- Penyempurnaan pada tampilan dibuat lebih menarik.
- Pemeliharaan berkala akan memastikan aplikasi berjalan lancar dan mengurangi risiko gangguan teknis.
- Tambahan fitur-fitur yang sesuai kebutuhan pengguna.
- Serta peningkatan responsivitas aplikasi akan memberikan pengalaman pengguna yang lebih baik dengan waktu loading yang lebih cepat dan responsif.

REFERENCES

- [1] B. M. Wibawa, R. Octovianisa, R. A. Miyagi, and R. R. Mardhotilah, "Pengaruh Perilaku Komplain dan Kepuasan Penanganan Komplain Terhadap Minat Pembelian Konsumen di E-Commerce," *J. Teknol. Inf. dan Terap.*, vol. 6, no. 2, pp. 41–49, 2019, doi: 10.25047/jtit.v6i2.107.
- [2] H. Rahman and H. Nurlatifah, "Analisis Pengaruh Perceived Ease of Use, Trust, Online Convenience terhadap Purchase Intention melalui Online Shopping Habits (Studi Kasus Pembelian Tiket Bioskop pada Aplikasi Gotix)," *J. Al Azhar Indones. Seri Ilmu Sos.*, vol. 1, no. 1, p. 29, 2020, doi: 10.36722/jaiss.v1i1.458.
- [3] S. N. Zaman, N. Merlina, and N. Nurajijah, "Sistem Informasi Keluhan Pelanggan Berbasis Website," *EVOLUSI J. Sains dan Manaj.*, vol. 9, no. 1, 2021, doi: 10.31294/evolusi.v9i1.9636.
- [4] A. Triawan and A. R. Y. Siboro, "Penerapan Application Programming Interface (API) Pada Push Notification Untuk Informasi Monitoring Stok Barang Minim," *Teknois J. Ilm. Teknol. Inf. dan Sains*, vol. 11, no. 2, pp. 107–114, 2021, doi: 10.36350/jbs.v11i2.120.
- [5] Ridwan and R. Musfika, "Rancangan Dan Implementasi Web service Untuk Integrasi Aplikasi Haba Gamping Menggunakan Metode RESTAPI," *Konf. Nas. Ilmu Komput.* 2021, pp. 420–423, 2021, [Online]. Available: <https://suwarga.bandaacehkota.go.id/>.
- [6] I. Hamal Haris and E. Dwi Wahyuni, "Pengembangan Sistem Marketplace Kadin Kota Batu menggunakan Metode Incremental," *Repositor*, vol. 5, no. 4, pp. 849–864, 2023.
- [7] A. P. Putra, F. Andriyanto, K. Karisman, T. D. M. Harti, and W. P. Sari, "Pengujian Aplikasi Point of Sale

Menggunakan Blackbox Testing,” J. Bina Komput.,
vol. 2, no. 1, pp. 74–78, 2020, doi:
10.33557/binakomputer.v2i1.757.

- [8] Irian and Yusuf Yudhistira, “Implementasi Application Programming Interface (API) Kawal Corona Sebagai Media Informasi Pandemi Covid-19 Berbasis Android,” J. Sist. Inf. Dan Teknol. Perad., vol. 2, no. 1, pp. 22–29, 2021.