

APLIKASI ISSUE TRACKING SYSTEM BERBASIS WEB (Studi Kasus: PT Mitra Kuadran Teknologi)

Muchamad Fajri Amirul Nasrullah¹, Rina Yulius², Linargian Pratama³

¹Politeknik Negeri Batam

Jln. Ahmad Yani, Batam Centre, Batam 29461, Indonesia, +62-778-469858 Ext.1017 / +62-778-463620,

fajri@polibatam.ac.id

²Politeknik Negeri Batam

Jln. Ahmad Yani, Batam Centre, Batam 29461, Indonesia, +62-778-469858 Ext.1017 / +62-778-463620,

rinayulius@polibatam.ac.id

³Politeknik Negeri Batam

Jln. Ahmad Yani, Batam Centre, Batam 29461, Indonesia, +62-778-469858 Ext.1017 / +62-778-463620,

linargian00@gmail.com

ARTICLE INFO

Article history:

Received 30 Agustus 2021

Received in revised form 2 September 2021

Accepted 10 September 2021

Available online 22 September 2021

ABSTRACT

PT Mitra Kuadran Teknologi is a software development company. Applications that have been used by customers; problems are often found that need to be resolved as soon as possible.

Customers used to submit complaints by sending messages directly to the programmer in person. Customers sometimes need to send the same complaint multiple times or to multiple people for resolution. Complaint handling like this is still not effective, because the complaint process is still done manually so that it affects the performance of programmers in solving problems. In this research, the author will make an application based on these problems. Helpdesk Kuadran application was developed as an issue tracking system application with the purpose of making it easier for customers to register problems and programmers to solve problems. The application is built using the waterfall method. The user acceptance testing (UAT) method is used to measure the value of user acceptance. The results of usability testing using the System Usability Scale (SUS) method which were carried out resulted in an average final score of 82.5. This value is above the 68 approval limit and is considered to make it easier for programmers to facilitate customer complaints.

Keywords: Issue tracking system, ASP.NET Core, complaint.

1. PENDAHULUAN

PT Mitra Kuadran Teknologi merupakan perusahaan software house yaitu perusahaan yang memberikan layanan untuk pengembangan perangkat lunak. PT Mitra kuadran teknologi memiliki misi untuk memungkinkan pelanggan kami mengembangkan bisnis mereka dengan memberikan pengalaman perangkat lunak terintegrasi menggunakan pengembangan aplikasi layanan kami. PT Mitra Kuaran Teknologi juga memiliki visi untuk Menjadi perusahaan yang memberikan pelayanan profesional dan produk terkait pengembangan teknologi perangkat lunak, untuk membuat layanan yang lebih baik untuk aplikasi sistem perusahaan.

Proses penyelesaian masalah di PT Mitra Kuadran Teknologi dilakukan secara manual. Pelanggan aplikasi biasa menanyakan langsung ke programmer atau pegawai lainnya menggunakan aplikasi pesan singkat WhatsApp. Pelanggan biasa mengirimkan pesan ke grup atau langsung ke akun personal lalu programmer menjawab pesan. Hal yang disampaikan pelanggan bisa berupa pertanyaan mengenai perangkat lunak yang digunakan, perubahan pada aplikasi untuk meningkatkan efektivitas hingga pengaduan masalah serius yang melumpuhkan fitur utama aplikasi, masalah seperti ini perlu segera diselesaikan.

Meningkatnya jumlah pelanggan dan aplikasi yang dipasarkan juga meningkatkan pengaduan masalah yang perlu diselesaikan. Pelanggan terkadang perlu mengirim pengaduan masalah yang sama beberapa kali atau ke beberapa orang terlebih dahulu untuk diselesaikan. Hal ini dapat menyebabkan tersebarnya pengaduan isu di banyak tempat dan tertimbunnya pengaduan dengan pesan atau bahkan pengaduan lain yang menyebabkan lambatnya masalah diselesaikan. Programmer dan pelanggan juga kesulitan untuk melacak pengaduan dan mencari informasi yang diberikan pelanggan. Hal ini dapat berdampak negatif pada kepuasan pelanggan jika tidak dapat diselesaikan.

Terkait dengan masalah di atas, penulis memberikan solusi yang dapat menyelesaikan permasalahan di atas dengan membuat aplikasi issue tracking system. Aplikasi issue tracking system adalah yang menyediakan sistem tiket untuk mencatat dan mengikuti perkembangan setiap masalah yang diajukan oleh pelanggan hingga masalah teratasi [2,4]. Aplikasi dapat melakukan Manipulasi data pengguna manipulasi data aplikasi, pembuatan tiket masalah, dan notifikasi pelanggan [8]. Dari penjabaran yang ada, penulis berencana untuk merancang sebuah aplikasi yang diberi judul “ APLIKASI ISSUE TRACKING SYSTEM BERBASIS WEB ”

2. TINJAUAN PUSTAKA

2.1. Issue Tracking System

Issue Tracking System (ITS) adalah aplikasi yang menyediakan sistem tiket untuk mencatat dan mengikuti perkembangan setiap masalah yang diajukan oleh pengguna hingga masalah teratasi [2]. Masalah dapat berupa apa saja, mulai dari laporan kesalahan dalam aplikasi hingga pertanyaan pengguna dan pertanyaan dalam pengembangan perangkat lunak. Ketika pengguna yang berbeda melaporkan masalah yang disebabkan oleh kesalahan yang sama, maka setiap pengguna membuka tiket berbeda di ITS dan dilacak hingga diselesaikan, biasanya dengan menerapkan perbaikan pada aplikasi dan memverifikasi dengan pengguna yang terpengaruh bahwa mereka tidak lagi mengalami masalah yang dilaporkan.

Setiap masalah di ITS biasanya memiliki beberapa detail yang terkait dengannya. Beberapa masalah mungkin memiliki urgensi yang lebih besar daripada yang lain, mungkin karena mereka mempengaruhi sebagian besar pengguna atau karena mereka mewakili hambatan serius yang perlu dipecahkan agar proyek dapat berlanjut dengan lancar.

Demikian juga, masalah juga dapat memiliki urgensi rendah atau nol untuk menunjukkan bahwa mereka harus diselesaikan jika waktu memungkinkan.

Detail lainnya mencakup semuanya, mulai dari pelanggan yang mengalami masalah hingga deskripsi terperinci tentang masalah yang dialami hingga solusi yang dicoba dan informasi relevan lainnya.

2.2. ASP.NET Core

ASP.NET adalah framework pengembangan web yang populer untuk mengembangkan aplikasi web pada platform .NET. ASP.NET Core adalah versi open source dari ASP.NET, yang berjalan di macOS, Linux, dan Windows. ASP.NET Core pertama kali dirilis pada tahun 2016 dan merupakan desain ulang dari versi ASP.NET khusus Windows sebelumnya. [12, 13, 14]

ASP.Net Core memiliki berbagai fitur sebagai kelebihan, antara lain :

1. Cross-platform, framework ASP.Net Core dapat berjalan di berbagai platform seperti Windows, Linux, macOS.
2. Hosting, ASP.Net Core dapat dijalankan di berbagai hosting. Contohnya IIS, Apache, dan Docker.
3. Satu model pengembangan. ASP.Net Core hanya perlu satu model pengembangan untuk aplikasi MVC web dan juga API web.
4. Open source, yaitu ASP.Net Core dikembangkan oleh Microsoft tetapi berkolaborasi melalui komunitas pengembang open source yang luas.
5. Modular, ASP.Net Core menyediakan komponen middleware sehingga pengembangan dilakukan secara modular dapat dicapai secara luas.

2.3. PostgreSQL

PostgreSQL adalah sistem basis data relasional open source yang kuat. PostgreSQL menggunakan dan memperluas bahasa SQL yang dikombinasikan dengan banyak fitur yang secara aman menyimpan data yang paling rumit. PostgreSQL dibuat pada tahun 1982 yang awalnya diberi nama Ingres Project di University of California di Berkeley dan hingga saat ini masih aktif dikembangkan [15].

PostgreSQL adalah sistem basis data relasional tingkat tinggi yang mendukung SQL (relasional) dan JSON (non-relasional). Ini adalah sistem manajemen basis data yang sangat stabil, didukung oleh lebih dari 20 tahun pengembangan komunitas yang telah berkontribusi pada tingkat ketahanan, integritas, dan kebenarannya yang tinggi [16].

PostgreSQL sebagai sistem manajemen basis data memiliki banyak fitur yang disediakan, antar lain:

1. Open source, kode sumber tersedia secara bebas di bawah lisensi sumber terbuka.
2. Bebas biaya, sebagai produk open source sejati, PostgreSQL tidak dikenakan biaya apa pun
3. Keandalan, beberapa perusahaan dan individu berkontribusi pada proyek dan telah mendorong inovasi selama lebih dari 25 tahun sekarang. Komunitas yang kuat memastikan bahwa bug diperbaiki tanpa penundaan.
4. Keamanan, ada banyak fitur untuk meningkatkan keamanan.
5. Skalabilitas, basis data PostgreSQL dapat berkembang sesuai kebutuhan. Ada beberapa opsi teknis untuk mengoperasikan PostgreSQL dalam skala besar.

3. METODOLOGI PENELITIAN

3.1. Metode Pengumpulan Data

3.1.1. Wawancara

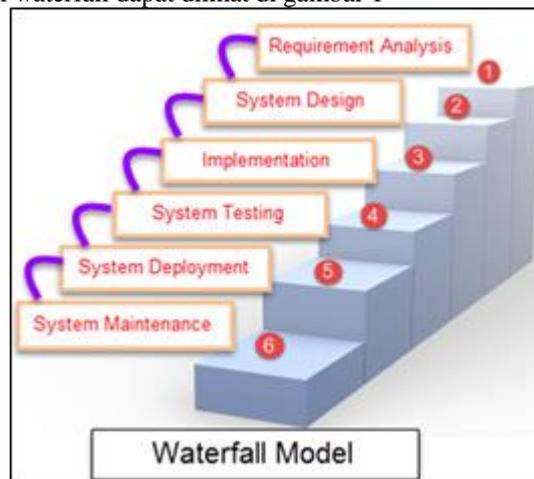
Metode pengumpulan data dengan melakukan sesi tanya jawab dengan Bapak M. Syukri Ultari selaku programmer di PT Mitra Kuadran Teknologi. Wawancara ini bertujuan untuk mendapatkan informasi tentang data dan sistem yang berjalan saat ini tentang cara penanganan pengaduan masalah oleh pengguna.

3.1.2. Studi Pustaka

Metode pengumpulan data yang dihasilkan dari artikel internet, jurnal yang berkaitan tentang issue tracking system dan hal lainnya yang menunjang dalam proses pembuatan aplikasi ini.

3.2. Metode Pengembangan Sistem

Pengembangan dengan model waterfall adalah Model Systems development life cycle (SDLC) yang sering digunakan pengembangan perangkat lunak. Model waterfall merupakan model pengembangan di mana seluruh proses pengembangan perangkat lunak dibagi ke dalam fase-fase [1,5]. Pembagian fase pengembangan dalam model waterfall dapat dilihat di gambar 1



Gambar 1. Model Waterfall (Sumber : www.guru99.com)

i. Requirement Analysis

Pada tahap ini, penulis menganalisis dan memahami semua kemungkinan persyaratan sistem yang akan dikembangkan dan didokumentasikan dalam dokumen spesifikasi kebutuhan.

ii. System Design

Pada tahap ini, penulis mengubah kebutuhan pengguna ke sebuah perancangan desain aplikasi. Desain sistem berguna untuk membuat persyaratan sistem, perangkat keras yang digunakan dan arsitektur dari sistem yang akan dikembangkan secara keseluruhan.

iii. Implementation

Pada tahap ini, penulis mengubah perancangan desain aplikasi ke sebuah program yang dimengerti oleh mesin. Kode program yang dihasilkan berupa modul atau program kecil.

iv. System Testing

Pada tahap ini, penulis menggabungkan semua modul kecil yang dihasilkan pada tahap implementasi ke dalam sebuah sistem. Seluruh sistem lalu dilakukan pengujian secara keseluruhan.

v. System Deployment

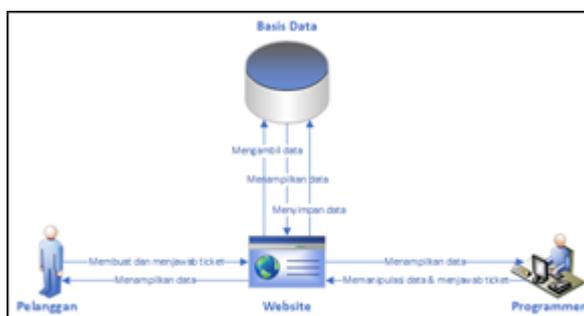
Pada tahap ini, penulis memasang aplikasi yang telah di uji di tahap sebelumnya ke pengguna agar dapat digunakan. Penulis memastikan seluruh sistem terpasang dengan benar dan berjalan sesuai yang diharapkan.

vi. Maintenance

Pada tahap terakhir ini, penulis melakukan pemeliharaan aplikasi. Pemeliharaan dilakukan dengan memperbaiki kesalahan yang ditemukan penulis atau pengguna dan menambah fitur meningkatkan produk menjadi lebih efektif.

3.3. Gambaran Umum Sistem

Aplikasi web yang akan dirancang ini untuk membantu pelanggan dan programmer dalam membuat, mengubah dan menyelesaikan pengaduan masalah. Pada gambar 2, programmer pertama mengisi data aplikasi atau layanan, data perusahaan dan data pengguna yang dapat menggunakan aplikasi web ini. Setiap pelanggan di hubungkan dengan data perusahaan dan data aplikasi yang berkaitan. pelanggan kemudian masuk ke dalam sistem menggunakan kredensial yang telah diberikan. Setelah masuk pelanggan dapat membuat tiket isu baru dengan memilih aplikasi yang bermasalah dan menjelaskan isu yang dimiliki. Programmer kemudian dapat menyelesaikan isu yang dimiliki pelanggan. Aplikasi akan mengirimkan notifikasi setiap pembuatan, tambahan dan manipulasi isu lainnya ke alamat email orang yang bersangkutan. Jika dianggap sudah selesai programmer dan pelanggan dapat menutup tiket. Pelanggan dapat membuka kembali tiket jika masih memiliki isu tambahan.

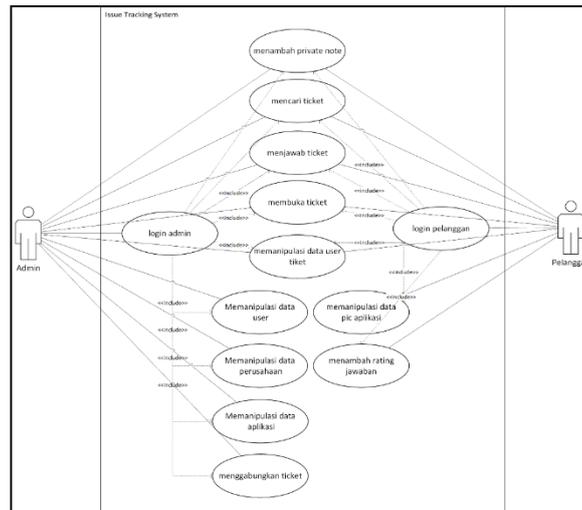


Gambar 2. Gambaran Umum Sistem

3.4. Use Case Diagram

Use Case Diagram, yaitu diagram yang mendefinisikan hubungan antara fungsi yang terdapat pada sistem dengan aktor-aktornya [3].

Gambar 3 di bawah terdiri dari 3 komponen yaitu sistem, aktor, dan *use case*. Sistem digambarkan dengan bentuk persegi berisi semua *use case* dalam sistem tersebut. Komponen aktor digambarkan ke dalam bentuk orang, fungsi dari aktor adalah untuk menggambarkan jenis pengguna dan interaksinya dengan sistem. Komponen *use case* digambarkan ke dalam bentuk lingkaran lonjong yang berfungsi untuk menggambarkan fungsional dalam sebuah sistem.



Gambar 3. Use Case Diagram

Diagram ini memiliki banyak *use case* dengan fungsi masing-masing. Penjelasan fungsionalitas diagram *use case* pada gambar 3 dijelaskan dalam tabel 1 di bawah ini.

Tabel 1. Definisi *Use Case*

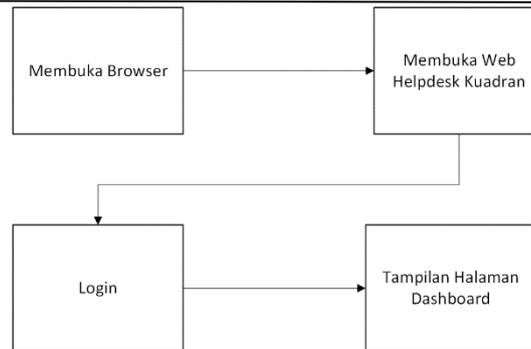
No	Use Case	Deskripsi
1.	Login Admin	Pemeriksaan <i>username</i> dan <i>password</i> yang digunakan untuk masuk ke sistem sebagai admin.
2.	Login Pelanggan	Pemeriksaan <i>username</i> dan <i>password</i> yang digunakan untuk masuk ke sistem sebagai pelanggan.
3.	Memanipulasi data <i>user</i>	Admin dapat menambah, mengubah, dan menghapus data <i>user</i> sesuai skema yang ditentukan.
4.	Memanipulasi data perusahaan	Admin dapat menambah, mengubah, dan menghapus data perusahaan sesuai skema yang ditentukan.
5.	Memanipulasi data aplikasi	Admin dapat menambah, mengubah, dan menghapus data aplikasi sesuai skema yang ditentukan.
6.	Menggabungkan tiket	Admin menggabungkan tiket yang sama menjadi satu tiket.
7.	Memanipulasi data <i>PIC</i> aplikasi	Pelanggan dapat menambah, mengubah atau menghapus data <i>PIC</i> aplikasi untuk mendapatkan notifikasi tiket terkait.
8.	Memanipulasi data <i>user</i> tiket	<i>User</i> dapat memanipulasi data <i>user</i> yang tergabung dalam satu perusahaan.
9.	Membuka tiket	<i>User</i> dapat membuka tiket baru.
10.	Menjawab tiket	<i>User</i> dapat menjawab tiket aktif jika tergabung dalam tiket.
11.	Mencari tiket	<i>User</i> dapat mencari tiket yang sudah menggunakan kata kunci dan data lainnya.

4. HASIL DAN PEMBAHASAN

4.1. Hasil

Tahapan pada saat menggunakan *issue tracking system* Helpdesk Kuadran adalah pertama dengan membuka browser kemudian mengakses alamat web Helpdesk Kuadran. Halaman pertama yang akan muncul adalah halaman *login*. Pengguna bisa *login* sebagai pelanggan ataupun *programmer* atau admin dengan memasukkan *username* dan *password* yang telah dibuat sebelumnya. Pelanggan akan diberikan akun oleh admin untuk dapat masuk ke dalam web.

Jika *username* dan *password* yang dimasukkan di halaman *login* benar maka pengguna berhasil masuk. Halaman pertama yang ditampilkan setelah *login* adalah halaman *dashboard*.

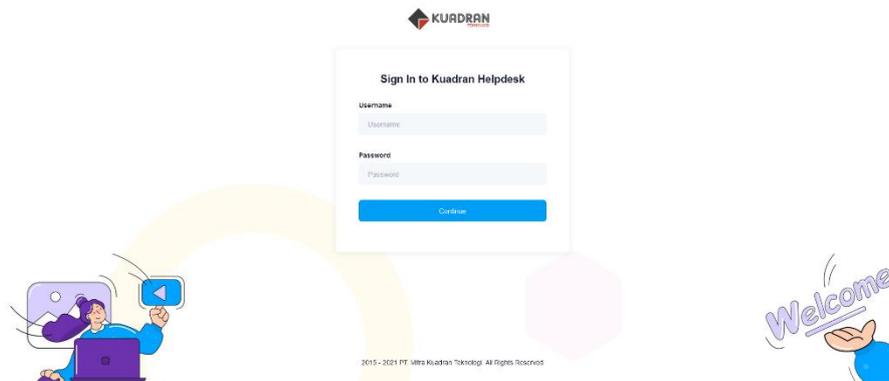


Gambar 4. Implementasi Sistem

Berikut adalah tampilan dari sistem yang telah diimplementasi

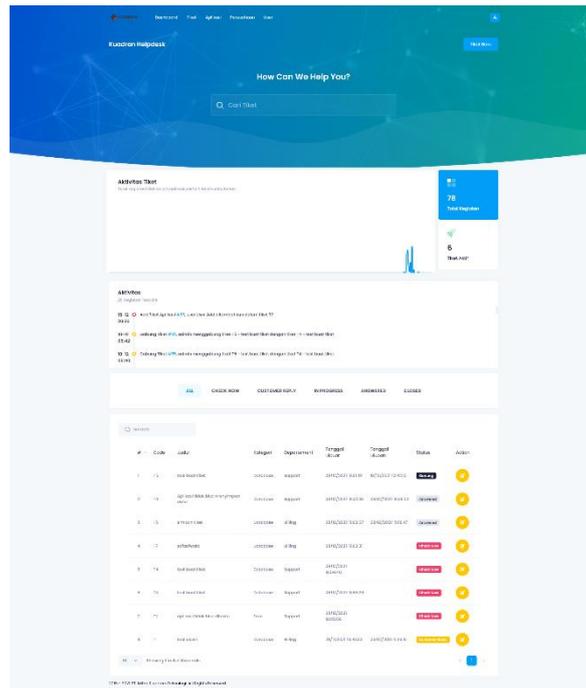
A. Tampilan Halaman *Login*

Tampilan halaman *login* sistem sesuai gambar 5. Halaman *login* tersebut digunakan oleh pelanggan dan admin untuk masuk dalam aplikasi.

Gambar 5. Tampilan Halaman *Login*

B. Halaman *Dashboard* Admin

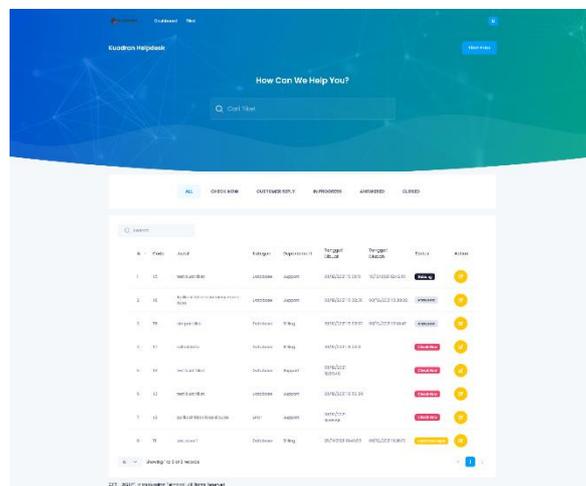
Tampilan antarmuka halaman *dashboard* admin disajikan gambar 6. Halaman *dashboard* admin berisi data seperti jumlah aktivitas tiket per hari, total dan aktivitas terakhir yang terjadi. Halaman ini juga berisi daftar tiket yang ada dalam sistem.



Gambar 6. Tampilan Halaman *Dashboard* Admin

C. Halaman *Dashboard* Pelanggan

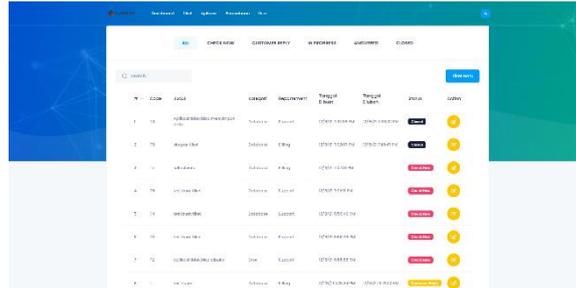
Tampilan antarmuka halaman *dashboard* pelanggan disajikan gambar 7. Halaman *dashboard* pelanggan berisi daftar tiket yang bisa di akses, menu pencarian tiket dan membuat baru tiket.



Gambar 7. Tampilan Halaman *Dashboard* Pelanggan

D. Halaman Daftar Tiket

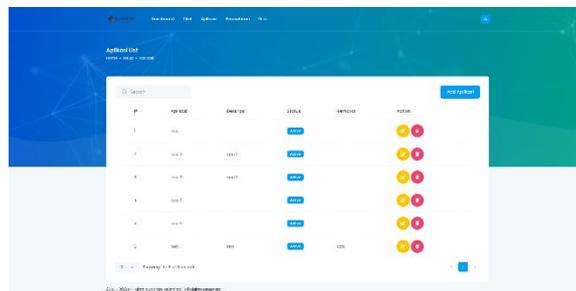
Tampilan antarmuka daftar tiket disajikan gambar 8. Halaman daftar tiket berisi seluruh tiket yang dapat diakses pelanggan dipisah berdasarkan status. Pelanggan dapat membuat tiket baru, mengubah tiket, menggabungkan tiket dan menghapus tiket.



Gambar 8. Tampilan Halaman Daftar Tiket

E. Halaman Daftar Aplikasi

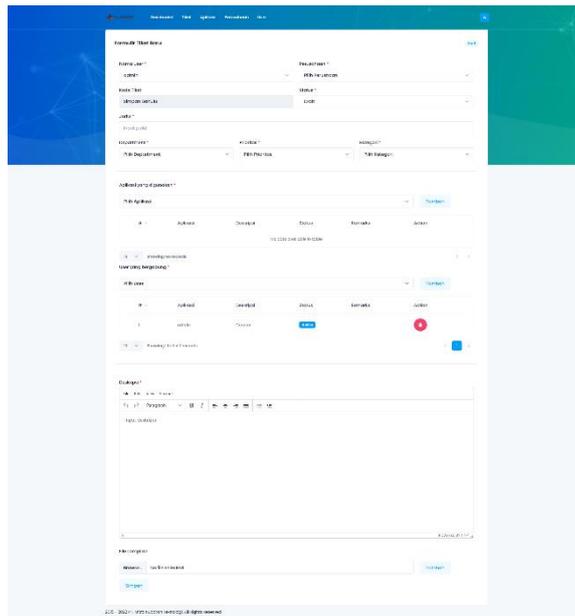
Tampilan antarmuka halaman daftar aplikasi disajikan gambar 9. Halaman daftar aplikasi berisi daftar semua aplikasi yang dapat diajukan oleh pelanggan dalam tiket. Admin dapat membuat aplikasi baru, mengubah aplikasi dan menghapus aplikasi.



Gambar 9. Tampilan Halaman Daftar Aplikasi

F. Halaman Formulir Tiket Baru

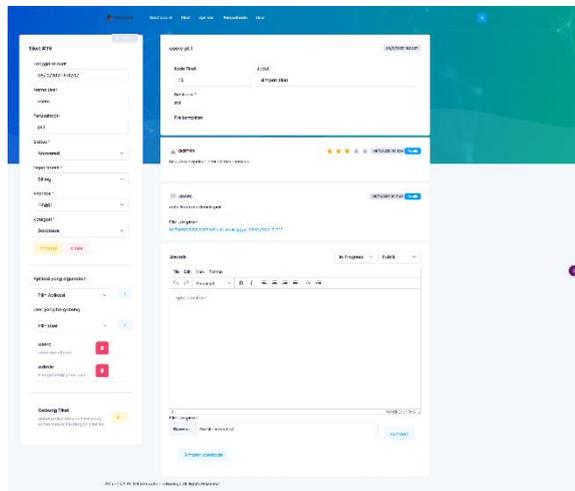
Tampilan antarmuka halaman formulir tiket baru disajikan gambar 10. Halaman formulir tiket baru berisi formulir yang harus diisi pelanggan untuk membuat tiket baru. Pelanggan menggunakan formulir untuk membuat tiket baru. Pelanggan dapat mengatur *user* yang tergabung, aplikasi yang tergabung, detail, kategori, dan data lainnya.



Gambar 10. Tampilan Halaman Formulir Tiket Baru

G. Halaman Detail Tiket

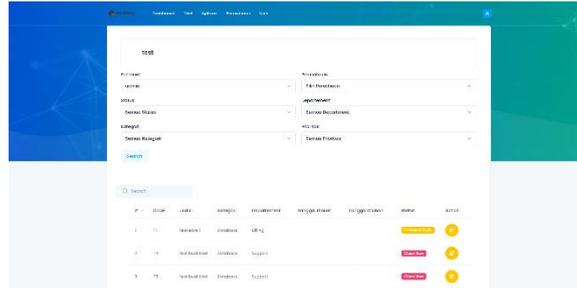
Tampilan antarmuka halaman detail tiket disajikan gambar 11. Halaman detail tiket digunakan untuk melihat data lengkap dari tiket yang sudah dibuat. Halaman detail tiket berfungsi untuk melihat semua jawaban yang ada dalam tiket. Pelanggan dapat menilai jawaban yang diberikan admin, pengguna dapat menambah aplikasi dan user yang bergabung dalam tiket. Pengguna yang bergabung juga dapat menjawab tiket, jawaban dapat diatur agar hanya diri sendiri atau kelompok pengguna tertentu yang dapat melihat.



Gambar 1. Tampilan Halaman Detail Tiket

H. Halaman Pencarian Tiket

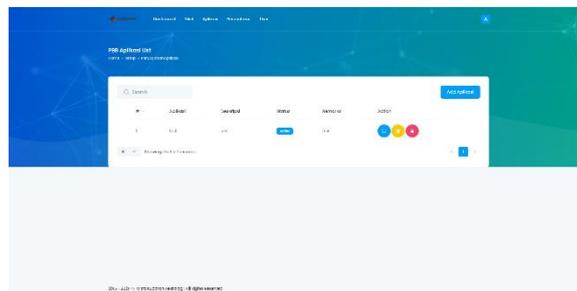
Tampilan antarmuka halaman pencarian tiket disajikan gambar 12. Halaman pencarian tiket digunakan pengguna untuk mencari tiket yang dapat diakses menggunakan kata kunci, departemen, prioritas, kategori dan pembuat. Pelanggan dapat melihat detail tiket yang muncul, menghapus tiket dan menggabungkan tiket.



Gambar 12. Tampilan Halaman Pencarian Tiket

I. Halaman Daftar Aplikasi Perusahaan

Tampilan antarmuka halaman daftar aplikasi perusahaan disajikan gambar 13. Halaman daftar aplikasi perusahaan berisi daftar aplikasi yang digunakan dalam satu perusahaan. Pengguna dapat menambahkan aplikasi yang terdaftar ke dalam perusahaan, menghapus dan melihat detail pengguna dalam aplikasi.



Gambar 13. Tampilan Halaman Daftar Aplikasi Perusahaan

4.2. Pengujian

Pengujian Aplikasi Helpdesk Kuadran sebagai Issue Tracking System dilakukan dengan menguji berdasarkan kebutuhan fungsionalnya. Pengujian ini dilakukan untuk mengurangi kesalahan atau berbedanya fungsi yang telah dibuat.

Metode pengujian yang dilakukan adalah metode Black Box. Metode pengujian black box adalah pengujian yang hanya memedulikan masukan dan keluaran aplikasi tanpa mengetahui kode aplikasi [10]. Metode ini dipilih agar pengujian yang dilakukan difokuskan kepada fungsionalitas yang ada dalam sistem yang telah di implementasi.

Pengujian aplikasi dilakukan oleh tim dari PT Mitra Kuadran Teknologi terutama tim programmer. Tim pengujian berjumlah 8 orang, 6 orang terdiri dari divisi software dan 2 orang merupakan bagian dari manajemen. Mereka dipilih sebagai pengujian karena mereka adalah pemilik dari aplikasi ini dan terlebih karena mereka adalah target pengguna aplikasi. Berikut adalah hasil dari pengujian aplikasi Helpdesk Kuadran yang disajikan pada tabel 2 di bawah :

Tabel 2. Hasil Pengujian Aplikasi Helpdesk Kuadran

No	Use Case	Skenario Pengujian	Data Pengujian	Hasil Pengujian	Valid
1.	Login Admin	Memasukkan <i>username</i> dan <i>password</i> yang sudah terdaftar.	- <i>Username</i> : linargian00@gmail.com - <i>Password</i> : 12345	Berhasil masuk dan diarahkan ke halaman <i>dashboard</i> .	√
2.	Login Pelanggan	Memasukkan <i>username</i> dan <i>password</i> yang sudah terdaftar.	- <i>Username</i> : userc <i>Password</i> : 123	Berhasil masuk dan diarahkan ke halaman <i>dashboard</i> .	√
3.	Memanipulasi data <i>user</i>	Admin membuka menu <i>User</i> dan melakukan aksi	- Kode : 1 - Nama : Hendra	Menampilkan pesan data tersimpan.	√

		seperti menambah, mengubah dan menghapus.	- Email : hendr@mail.com - No. HP : 0811111111 - Perusahaan : pt1 - Jabatan : IT - Tipe : pengguna - Status : <i>Active</i> - <i>Username</i> : hendra - <i>Password</i> : 123		
4.	Memanipulasi data perusahaan	Admin membuka menu perusahaan dan melakukan aksi seperti menambah, mengubah dan menghapus.	- Nama : Polibatam - Deskripsi : Perguruan tinggi negeri - Status : <i>Active</i> - <i>Remarks</i> : -	Menampilkan pesan data tersimpan.	√
5.	Memanipulasi data aplikasi	Admin membuka menu aplikasi dan melakukan aksi seperti menambah, mengubah dan menghapus.	- Nama : SIA - Deskripsi : sistem informasi akademik - Status : <i>Active</i> - <i>Remarks</i> : -	Menampilkan pesan data tersimpan.	√
6.	Memanipulasi data <i>PIC</i> aplikasi	Admin membuka menu perusahaan lalu memilih salah satu perusahaan dan aplikasi di dalamnya, kemudian menambah dan menghapus <i>PIC</i> .	- Perusahaan : pt2 - Aplikasi : app2 - <i>PIC</i> : admin	Data <i>PIC</i> tersimpan sesuai aksi yang dilakukan.	√
7.	Membuka tiket	Pelanggan membuat tiket baru dari menu <i>dashboard</i> .	- Judul : masalah aplikasi - <i>Departement</i> : <i>support</i> - Prioritas : tinggi - Kategori : <i>Error</i> - Deskripsi : aplikasi fungsi <i>error</i>	Menampilkan pesan data tersimpan, halaman tiket yang sudah tersimpan dan email pesan terbuat.	√
8.	Memanipulasi data <i>user</i> tiket	Admin membuka tiket yang dibuat sebelumnya, lalu menambah admin sebagai <i>user</i> .	Data tiket yang sudah dibuat sebelumnya.	Menampilkan pesan data tersimpan.	√
9.	Menjawab tiket	Admin membuka tiket yang dibuat sebelumnya, lalu menjawab tiket.	- Jawaban : akan kami cek. - Status : <i>in progress</i> - Hak akses : publik	Menampilkan halaman detail tiket dengan jawaban tersimpan.	√
10.	Mencari tiket	Pelanggan mencari tiket dengan menulis kata kunci di halaman <i>dashboard</i> .	- Kata kunci : masalah	Menampilkan daftar tiket sesuai dengan kata kunci.	√
11.	Menggabungkan tiket	Admin membuka detail tiket. Menggabungkan tiket ke tiket lain.	Gabungkan tiket dengan salah satu tiket yang sudah ada.	Status tiket berubah menjadi gabung dan	√

				muncul info tiket telah digabung.	
--	--	--	--	-----------------------------------	--

Hasil pengujian dari tabel 2 menunjukkan bahwa seluruh use case telah tervalidasi dan dapat berjalan sesuai kebutuhan. Hasil pengujian ini menunjukkan bahwa sistem telah terimplementasi sesuai dengan kebutuhan dan telah berhasil dijalankan dengan baik.

Pengujian Usability atau kegunaan menggunakan metode System Usability Scale (SUS) untuk menguji kegunaan aplikasi terhadap pengguna [11]. Aplikasi dianggap memiliki kegunaan yang baik jika nilai akhirnya sama dengan atau di atas 68. Pengujian dilakukan dengan cara pengujian mencoba aplikasi yang telah dikembangkan. Pengujian lalu mengisi kuesioner mengenai kegunaan aplikasi yang telah mereka lakukan. Daftar pertanyaan dari pengujian tersaji pada tabel 3.

Tabel 3. Kuesioner SUS

No	Pertanyaan
1.	Saya berpikir akan menggunakan sistem ini lagi
2.	Saya merasa sistem ini rumit untuk digunakan
3.	Saya merasa sistem ini mudah digunakan
4.	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini
5.	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6.	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)
7.	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat
8.	Saya merasa sistem ini membingungkan
9.	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10.	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini

Bobot jawaban telah ditentukan dalam tabel 4 di bawah. Bobot jawaban digunakan untuk menghitung skor dari nilai yang di dapatkan dari kuesioner. Hasil analisis skor kuesioner setelah perhitungan tersaji pada tabel 5. Perhitungan dilakukan dengan mengurangi satu nilai skor pertanyaan bernomor ganjil. Dan nilai 5 dikurangi skor pertanyaan bernomor genap. Nilai akhir didapat dengan mengkali total skor dengan 2.5

Tabel 4. Bobot Jawaban Kuesioner SUS

No	Pilihan Jawaban	Interval Persentase
1.	Sangat Setuju (SS)	5
2.	Setuju (S)	4
3.	Netral (N)	3
4.	Tidak Setuju (TS)	2
5.	Sangat Tidak Setuju (STS)	1

Tabel 5. Rekapitulasi Hasil Kuesioner SUS

Responden No.	Pertanyaan										Total Skor	Nilai (total x 2,5)
	1	2	3	4	5	6	7	8	9	10		
1.	4	4	4	4	4	4	4	4	4	4	40	100
2.	3	3	4	0	3	3	2	2	3	2	25	62,5
3.	3	3	3	2	3	3	3	4	3	2	29	72,5
4.	4	3	4	3	4	3	4	3	4	2	34	85
5.	3	3	3	4	3	4	3	4	3	2	32	80
6.	3	3	3	2	3	3	3	3	3	2	28	70
7.	2	2	2	2	2	2	2	2	2	2	20	50

Berdasarkan rekapitulasi hasil kuesioner pada tabel 5 nilai akhir lebih besar dengan nilai 82.5 dari nilai maksimal 100. Nilai ini di atas nilai batas 68 dan dianggap memiliki kegunaan yang baik.

5. KESIMPULAN DAN SARAN

Berdasarkan seluruh tahap pengembangan dan pengujian yang telah dilakukan untuk membuat aplikasi Helpdesk Kuadran sebagai Issue Tracking System. Dapat disimpulkan sebagai berikut :

1. Sistem ini telah dirancang dari hasil analisis kebutuhan berupa gambaran umum sistem, kebutuhan fungsional, kebutuhan non fungsional, kebutuhan perangkat keras, dan kebutuhan perangkat lunak berdasarkan pengumpulan data yang telah dilakukan. Kemudian berdasarkan hasil analisis kemudian perancangan sistem berupa UML (use case diagram, activity diagram, dan class diagram), perancangan ERD, dan perancangan antarmuka.
2. Pengujian yang dilakukan oleh tim PT Mitra Kuadran Teknologi menggunakan metode black box menunjukkan seluruh fungsi dapat berjalan sesuai kebutuhan. Hasil kuesioner Usability atau kegunaan menggunakan metode System Usability Scale mendapatkan nilai akhir rata-rata 82.5. Hasil pengujian dan kuesioner ini menandakan bahwa aplikasi Helpdesk Kuadran sebagai Issue Tracking System dapat memfasilitasi pelaksanaan pelaporan pengaduan, pendataan aduan, manajemen aduan dan penyelesaian aduan sehingga mempermudah pelanggan dan programmer dalam menyelesaikan pengaduan.

Ucapan Terima Kasih

Jika perlu berterima kasih kepada pihak tertentu, misalnya sponsor penelitian, nyatakan dengan jelas dan singkat, hindari pernyataan terima kasih yang berbunga-bunga.

DAFTAR PUSTAKA

Daftar Pustaka ditulis mengikuti format *IEEE style* berikut:

Referensi Cetak:

Buku

Penulis. *Judul buku*. Lokasi Penerbit: Penerbit, tahun, halaman.

- [1] W.K. Chen. *Linear Networks and Systems*. Belmont, CA: Wadsworth, 1993, pp. 123-35.

Artikel dalam Buku

Penulis. "Judul Artikel" in *Judul Buku*, edisi, volume. Nama Editors, Ed. Lokasi Penerbit: Penerbit, tahun, halaman.

- [2] J.E. Bourne. "Synthetic structure of industrial plastics," in *Plastics*, 2nd ed., vol. 3. J. Peters, Ed. New York: McGraw-Hill, 1964, pp.15-67.

Jurnal

Penulis. "Judul Artikel". *Nama Jurnal*, vol., halaman, tanggal/tahun, DOI.

- [3] Christopher S. Goldenstein, et. al. "Infrared laser-absorption sensing for combustion gases." *Progress in Energy and Combustion Science*, Volume 60, May 2017, Pages 132-176, <https://doi.org/10.1016/j.pecs.2016.12.002>.

Prosiding

Penulis. "Judul Artikel." in *Conference proceedings*, tahun, halaman.

- [4] D.B. Payne and H.G. Gunhold. "Digital sundials and broadband technology," in *Proc. IOOC-ECOC*, 1986, pp. 557-998.

Peraturan Pemerintah

- [5] Republik Indonesia. Undang-Undang Nomor 3 Tahun 2014 tentang Perindustrian.

Standar

- [6] British Standards Institution. *B.S. 764*. London: British Standards Institution, 1990.

Dokumen Paten

Inventor."Nama/Judul." Negara dimana paten terdaftar. Nomor, tanggal.

- [7] E.E. Rebecca. "Alternating current fed power supply." U.S. Patent 7 897 777, Nov. 3, 1987.

Gambar Teknik

[8] F. Afrinaldi. *Rangka Belt Conveyor*. [Gambar Teknik]. Universitas Andalas: Padang, 2005.

Referensi Elektronik:Buku

Penulis. (Tahun, bulan tanggal). *Judul buku*. (Edisi). [On-line]. Volume(Nomor). Available: site/path/file [tanggal diakses].

[9] S. Calmer. (1999, June 1). Engineering and Art. (2nd edition). [On-line]. 27(3). Available: www.enggart.com/examples/students.html [May 21, 2003].

Web

Penulis. "Judul." Internet: complete URL, tanggal di-update [tanggal diakses].

[10] M. Duncan. "Engineering Concepts on Ice. Internet: www.iceengg.edu/staff.html, Oct. 25, 2000 [Nov. 29, 2003].

- [1] Aliya, Humaira. (2021). Model Waterfall: Pendekatan Sequential Life Cycle Model Paling Awal. <https://glints.com/id/lowongan/model-waterfall-adalah/> (diakses 28 Agustus 2021)
- [2] Cornell, Jared. (2021). What is Issue Tracking Software? The What, Why, and How Explained. <https://www.proprofsdesk.com/blog/what-is-issue-tracking-software/> (diakses 28 Agustus 2021)
- [3] Daly, Nicky. (2021). What Is a Use Case?. <https://www.wrike.com/blog/what-is-a-use-case> (diakses 10 Desember 2021)
- [4] Darmojo, Hardjito S, Mohammad Ridwan, dan Ryan Muhammad Bahrudin. Penerapan Helpdesk Ticketing System Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web. *JUTIS*, 7(1), 71-82. 2019.
- [5] Martin, Matthew. (2021). What is Waterfall Model in SDLC? Advantages and Disadvantages. <https://www.guru99.com/what-is-sdlc-or-waterfall-model.html> (diakses 28 Agustus 2021)
- [6] Moedjahedy, Jimmy H, Oktoverano Lengkong, dan Stenly Ibrahim Adam. Pengembangan IT Helpdesk Ticketing Sistem Berbasis Web di Universitas Klabat. *Cogito Smart Journal*, 6(2), 217-228. 2020.
- [7] Nishadha. (2021). UML Class Diagram Relationships Explained with Examples. <https://creately.com/blog/diagrams/class-diagram-relationships/> (diakses 10 Desember 2021)
- [8] Rochim, Adian Fatchur, Reza Taftazani, dan Sudjadi. PERANCANGAN TROUBLE TICKET SYSTEM PADA UPT PUSKOM UNIVERSITAS DIPONEGORO. *TRANSIENT*, 4(1), 215-218. 2015.
- [9] Setiawan, Rony. (2021). Memahami ERD, Model Data, dan Komponennya. <https://www.dicoding.com/blog/memahami-erd/> (diakses 13 November 2021)
- [10] Sonmez, John. 2020. 7 Common Types of Software Testing. <https://usersnap.com/blog/software-testing-basics/> (diakses 14 Januari 2022)
- [11] Z. Sharfina and H. B. Santoso, "An Indonesian adaptation of the System Usability Scale (SUS)," in *International Conference on Advanced Computer Science and Information Systems, ICACISIS 2016*, pp. 145–148. 2017.
- [12] Microsoft. (2021). What is ASP.NET Core?. <https://dotnet.microsoft.com/learn/aspnet/what-is-aspnet-core> (diakses 28 Agustus 2021)
- [13] Roth, Daniel, Rick Anderson, Shaun Luttin. (2021). Introduction to ASP.NET Core. <https://aspnetcore.readthedocs.io/en/stable/intro.html> (diakses 28 Agustus 2021)
- [14] Watson, Matt. (2017). Top 13 ASP.NET Core Features You Need to Know. <https://stackify.com/asp-net-core-features/> (diakses 28 Agustus 2021)
- [15] PostgreSQL. (2021). What is PostgreSQL?. <https://www.postgresql.org/about/> (diakses 28 Agustus 2021)
- [16] Peterson, Richard. (2021). What is PostgreSQL? Introduction, Advantages & Disadvantages. <https://www.guru99.com/introduction-postgresql.html> (diakses 28 Agustus 2021)