



Perancangan dan Pembangunan Sistem Portal *IT Service Management* Berbasis Flutter (Studi Kasus CV. Sabar Maju)

Aulia Ilham Zukri¹, Hendry Gunawan², Binastya Anggara Sekti³, Tri Ismardiko Widyawan⁴
^{1,2,3,4} Fakultas Ilmu Komputer, Universitas Esa Unggul, Jakarta, Indonesia

Article Info

Article history:

Received November 4, 2024
Revised November 5, 2024
Accepted November 7, 2024

Kata Kunci:

ITSM,
Flutter,
Laravel,
API,
Extreme programming,
SUS

Keywords:

ITSM,
Flutter,
Laravel,
API,
Extreme programming,
SUS

ABSTRAK

CV. Sabar Maju sebagai perusahaan yang bergerak dibidang retail sangat bergantung pada sistem IT untuk membantu proses bisnisnya yang beroperasi pada multi cabang. Sistem IT yang berjalan belum didukung oleh sistem layanan IT (*IT services*) sehingga menyebabkan tidak aktualnya data terkait dengan layanan IT yang berjalan seperti data aset IT dan data perbaikan aset IT sehingga riwayat aset tidak jelas. Proses pelaporan kendala juga masih dilakukan secara manual sehingga penanganan kendala menjadi tidak efektif. *IT Service Management (ITSM)* merupakan seperangkat kebijakan dan praktek untuk mengimplementasikan, memberikan, dan mengelola layanan IT untuk *end-user* yang ditentukan dan tujuan bisnis yang ditentukan juga. Pembuatan sistem portal dengan acuan *ITSM* dapat membantu perusahaan dalam menjalankan layanan IT pada perusahaan. *Front-end* sistem portal ini dibangun menggunakan kerangka kerja Flutter yang bermanfaat dengan dukungan multi-platform yang baik sehingga sistem portal dapat berjalan pada platform mobile dan web dengan 1 basis kode. *Back-end* pada sistem ini menggunakan Laravel yang bermanfaat pada kecepatan proses dalam pembangunan sistem. Pengujian sistem dilakukan menggunakan *System Usability Scale (SUS)* dan *black-box testing*. Hasil pengujian menunjukkan bahwa sistem portal yang dibangun sudah membantu memberikan sistem layanan IT yang efektif dan efisien.

ABSTRACT

CV. Sabar Maju as a company operating in the retail sector is highly dependent on IT systems to help its business processes that operate in multiple branches. The IT system that runs has not been supported by an IT services system, causing the data related to IT services to not be up to date, such as IT asset data and IT asset repair data so that the asset history is unclear. The problem reporting process is also still done manually so that the handling of obstacles becomes ineffective. IT Service Management (ITSM) is a set of policies and practices to implement, deliver, and manage IT services for specified end-users and specified business objectives as well. Creating a portal system with an ITSM reference can help companies in running IT services at the company. The front-end of this portal system is built using the Flutter framework which is useful with good multi-platform support so that the portal system can run on mobile and web platforms with 1 code base. The back-end of this system uses Laravel which is beneficial to the speed of the process in system development. System testing is done using System Usability Scale (SUS) and black-box testing. The test results show that the portal system built has helped provide an effective and efficient IT service system.

This is an open access article under the [CC BY](https://creativecommons.org/licenses/by/4.0/) license.



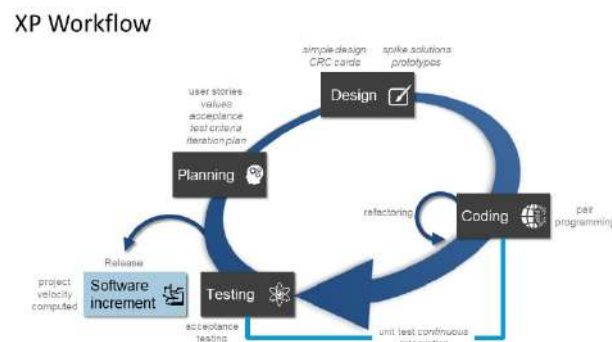
Corresponding Author:

Aulia Ilham Zukri
Fakultas Ilmu Komputer, Universitas Esa Unggul,
Jakarta, Indonesia
Email: zukri.ilham@student.esaunggul.ac.id

1. PENDAHULUAN

Pertumbuhan implementasi TI dalam bisnis di Indonesia meningkat tajam dari tahun ke tahun. Menurut perkiraan dari *International Data Corporation (IDC)*, pengeluaran di bidang Teknologi Informasi dan Komunikasi (TIK) di Indonesia akan mengalami pertumbuhan lebih dari 8 persen dan mencapai US\$30,1 miliar pada tahun 2021 dan diperkirakan akan mencapai US\$33,9 miliar pada tahun 2024. Hal ini menunjukkan pertumbuhan signifikan dalam pemakaian TI dalam dunia bisnis di Indonesia. Teknologi informasi membuat proses bisnis pada suatu perusahaan atau organisasi menjadi lebih efisien dan efektif.

Perkembangan pemanfaatan teknologi informasi ini mendukung adopsi transformasi digital pada perusahaan. Transformasi digital merupakan suatu proses yang dilakukan oleh perusahaan atau organisasi untuk mengubah cara kerjanya dengan menggunakan teknologi digital dalam seluruh aspek bisnis, termasuk strategi, operasi, dan interaksi dengan pelanggan untuk meningkatkan efisiensi, efektivitas, dan inovasi. Manfaat utama dari transformasi digital adalah peningkatan efisiensi, efektivitas, dan inovasi. Teknologi digital memungkinkan perusahaan untuk mengumpulkan dan menganalisis data dalam skala besar, membuat keputusan yang lebih baik, dan meningkatkan interaksi dengan pelanggan. Ini juga memungkinkan perusahaan untuk menciptakan produk dan layanan baru, serta memperluas pasar dan meningkatkan pangsa pasar [1].



Gambar 1. Tahapan *Extreme Programming*

ITSM (*IT Service Management*) merupakan salah satu kerangka kerja untuk mengelola infrastruktur IT dan memberikan layanan terbaik bagi pengguna suatu layanan IT. Dalam pengaplikasian ITSM, terdapat banyak standar kerangka kerja, salah satunya adalah IT

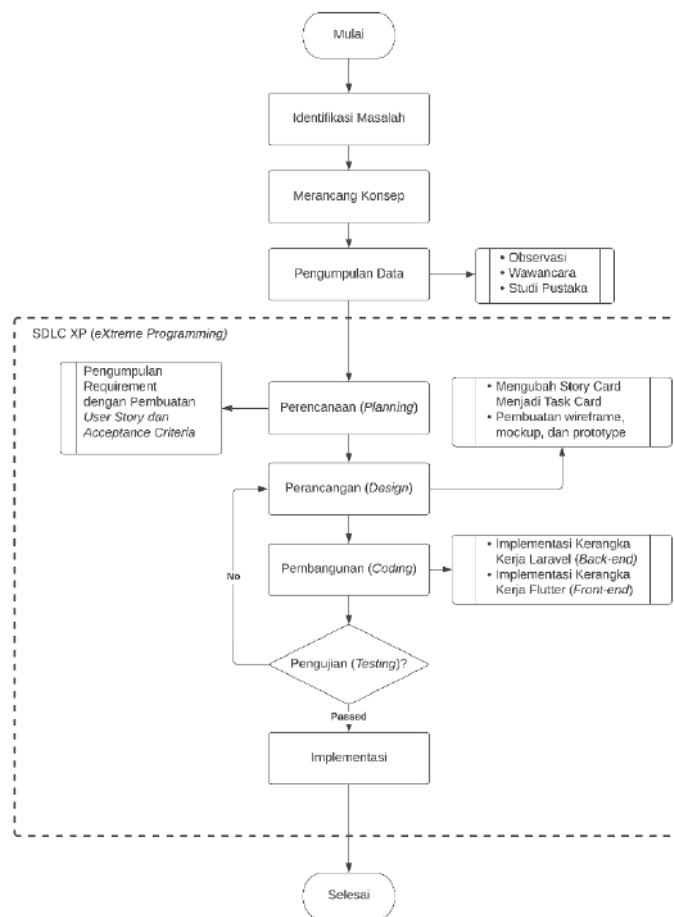
Infrastructure Library (ITIL). IBM mendeskripsikan ITIL sebagai perpustakaan yang berisikan best practice untuk mengelola layanan IT dan meningkatkan dukungan IT serta *Service Level* (SL). Salah satu tujuan utama dari ITIL adalah sejalan dengan layanan IT dengan tujuan bisnis bahkan sejalan dengan perubahan tujuan bisnis [2].

Penerapan portal ITSM pada CV. Sabar Maju bertujuan untuk membantu perusahaan dalam pengelolaan layanan IT yang berjalan pada perusahaan seperti manajemen aset dan *incident handling*. Penerapan sistem ini diharapkan dapat menyelesaikan permasalahan seperti tidak tercatatnya aset dan riwayat aset dan juga penanganan masalah yang tidak terstruktur yang disebabkan oleh tidak adanya sistem terpusat untuk penerimaan masalah IT.

2. METODE

2.1 Kerangka Berpikir

Kerangka berfikir merupakan uraian tentang konsep pemecahan masalah yang sebelumnya sudah dirumuskan[3].



Gambar 2. Kerangka Berpikir

2.2 Extreme Programming

Extreme Programming merupakan metodologi rekayasa perangkat lunak yang dikembangkan oleh Kent Beck pada tahun 1996 yang cocok diimplementasikan untuk proyek yang memiliki *dynamic requirements* atau kebutuhan yang sering berubah dari *client* [4].

Menurut [5], dalam fase pengembangan dengan menggunakan *extreme programming* terdapat sebuah fase yang dinamakan dengan *Iteration to Release Phase* yang dimana fase tersebut terdiri dari 4 tahap, yaitu:

1. *Planning* (Perencanaan)

Pada tahap ini, semua informasi dari semua data dikumpulkan guna memutuskan apa dan bagaimana seluruh arsitektur yang nantinya kelak dibangun.

2. *Design* (Perancangan)

Pada tahap ini, programmer memilih masalah yang timbul untuk kemudian membuat desain (pemodelan) dari masalah tersebut. Desain tersebut bisa berupa tampilan antarmuka dan desain logika bisnis.

3. *Coding* (Pengkodean)

Hasil dari desain atau pemodelan yang telah dibuat sebelumnya, diterjemahkan dalam bentuk kode atau biasa disebut dengan *Coding*. Kode tersebut nantinya akan menghasilkan suatu sistem sebagai upaya dalam solusi pemecahan masalah.

4. *Testing* (Pengujian)

Setelah selesai dalam tahap coding, rangkaian pengujian dilakukan kepada sistem sampai kemudian hasil coding berhasil diintegrasikan. Selain itu, dilakukan juga tahap *refactoring* dan aplikasi siap untuk masuk fase produksi.

2.3 Objek Penelitian

Penelitian ini akan dilakukan pada CV. Sabar Maju. CV. Sabar Maju merupakan perusahaan yang berjalan pada bidang retail yang fokus utamanya adalah terhadap penjualan *home appliances*. CV. Sabar Maju yang berlokasi di Jl. Terusan No.8 RT.005/RW.003, Sukarasa, Kec. Tangerang, Kota Tangerang, Banten.

Dipilihnya CV. Sabar maju sebagai objek penelitian ini dikarenakan peneliti melihat bahwa CV. Sabar Maju masih belum memiliki sistem portal ITSM terpusat yang dapat membantu perusahaan dalam menjalankan layanan IT, seperti *incident handling* dan *asset management*.

2.4 Teknik Pengumpulan Data

Dalam menganalisis kebutuhan pada penelitian ini, diperlukan data dan informasi mengenai pembuatan sistem ITSM pada CV. Sabar Maju. Metode pengumpulan data yang digunakan dalam penelitian ini, sebagai berikut:

- 1) Observasi: Pengamatan dan pencatatan data secara langsung untuk mengetahui bagaimana prosedur dukungan layanan IT dan prosedur manajemen aset IT yang berjalan saat ini. Data yang dimaksud merupakan kebutuhan *user* terhadap sistem yang akan dibuat.
- 2) Wawancara: Wawancara yang dilakukan dalam penelitian ini adalah wawancara tidak terstruktur. Wawancara dilakukan terhadap supervisor IT terkait permasalahan prosedur dukungan layanan IT dan prosedur manajemen aset IT yang berjalan saat ini sebagai pedoman pembuatan usulan sistem ITSM yang dapat membantu layanan IT yang berjalan.
- 3) Studi Pustaka: Studi pustaka dilakukan dengan tujuan pengumpulan informasi melalui buku dan jurnal terkait penelitian yang sedang dilakukan untuk selanjutnya dijadikan bahan referensi untuk pembangunan sistem yang dibutuhkan.

2.5 UML (*Unified Modeling Language*)

UML diartikan sebagai sintesis dari tiga metode desain dan analisis berbasis objek, dan juga menggabungkan keunggulan metode berorientasi objek lainnya (*fusion, shlaer-mellor, coad-yourdon*) yang juga terintegrasi dalam UML, memberikan metode yang sangat baik yang digunakan ada di industri perangkat lunak [6].

Dalam Penelitian ini digunakan 3 jenis diagram UML sebagai medium ilustrasi sistem. Berikut pengertian dari jenis-jenis diagram UML menurut [7]:

1) *Use Case Diagram*

Use Case Diagram menunjukkan interaksi antara komponen yang disebut *use case* dan aktor yang saling berhubungan. Ini digunakan untuk mengatur dan merancang aspek dinamis dari penggunaan sistem [7].

2) *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur dan juga interaksi antar aktor dari suatu sistem. *Activity diagram* juga menggambarkan alur kontrol dari setiap aktivitas yang terjadi dalam sistem tersebut, yang terdiri dari tindakan dan pembagian dari aliran kontrol itu sendiri [7].

3) *Class Diagram*

Class Diagram digunakan untuk menggambarkan entitas-entitas dan hubungan antar entitas yang berjalan pada suatu sistem dengan pendekatan berorientasi objek. Class diagram terdiri dari kelas, antarmuka, dan hubungan antar entitas.[7].

2.6 Flutter

Flutter merupakan SDK (*Software Development Kit*) dari Google yang menawarkan kemudahan kepada pengembang dalam pembuatan aplikasi *multi-platform*. Flutter menawarkan konsep sederhana “*wirte once, deploy everywhere*”. Pada saat ini, pengembangan aplikasi Flutter mendukung hampir seluruh *platform* yang tersedia, mulai dari Android, iOS, aplikasi Web, Linux, Windows, hingga MacOS. Flutter merupakan SDK lengkap yang menyediakan hampir semua yang dibutuhkan oleh pengembang, termasuk mesin rendering, komponen UI, kerangka pengujian, *router*, dan berbagai fitur lainnya [8].

2.7 Laravel

Laravel merupakan sebuah kerangka kerja untuk bahasa pemrograman PHP yang menitikberatkan kesederhanaan dan fleksibilitas pada struktur desainnya yang berbasis MVC (*Model, View, Controller*). Laravel juga dilengkapi dengan “*Artisan*” sebagai *bundle command-line tools* [9].

Menurut [10] Laravel adalah kerangka kerja untuk bahasa pemrograman PHP yang memiliki lisensi MIT dan berbasis MVC (*Model View Controller*) serta menyediakan sintak yang ekspresif dan jelas guna meningkatkan pengalaman bekerja dalam membuat web.

2.8 MySQL

MySQL merupakan DBMS (*Database Management Sistem*) populer yang memberikan dukungan penuh untuk *multiuser* dan *multithread*. MySQL adalah salah satu DBMS gratis berbasis SQL (*Structured Query Language*) merupakan bahasa pemrograman untuk memproses dan menyimpan informasi di dalam sebuah basis data relasional. [11]

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Kebutuhan (*Requirement Gathering*)

Pengumpulan kebutuhan dilakukan dengan wawancara terhadap beberapa aktor atau *role* yang terlibat dalam sistem yang akan dibuat. Dari wawancara tersebut nantinya akan terbuat *user stories*. *User stories* yang terbentuk biasanya masih ambigu dalam penjelasan fiturnya, oleh karena itu setelah *user stories* terbentuk maka selanjutnya *user story* akan diubah menjadi *Software Requirement Document* untuk memperjelas kebutuhan sistem dengan menambahkan *user acceptance criteria*, yaitu kriteria diterimanya suatu fitur oleh *user* untuk menghilangkan ambiguitas yang terbentuk pada *user stories*. Di bawah ini merupakan *user stories* yang terbuat dari hasil wawancara yang sudah dilakukan:

Tabel 1. *User Stories*

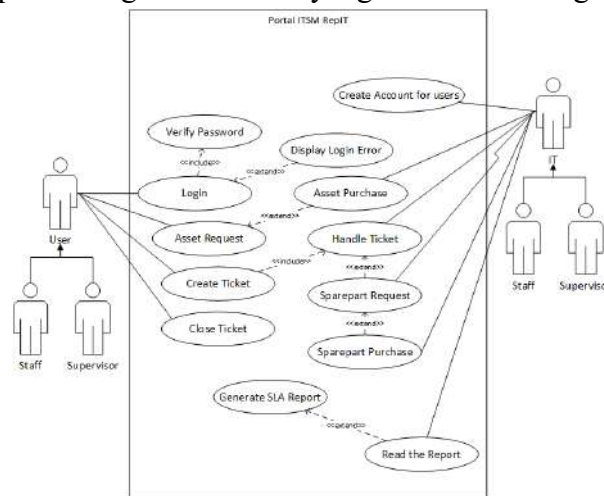
NO	Role	Story
1	Staf IT	Sebagai Staf IT saya ingin memiliki kontrol penuh terhadap akun <i>user</i> seperti pembuatan akun dan <i>reset password</i> akun sehingga keamanan akun terjaga
2	Staf IT	Sebagai <i>Staf IT</i> saya ingin dapat memberikan dan mengelola hak akses user berdasarkan jabatannya, sehingga kerahasiaan data tetap terjaga..
3	Staf IT	Sebagai Staf IT saya ingin mendapatkan notifikasi apabila ada pengajuan aset yang sudah di- <i>approve</i> oleh <i>Supervisor</i> IT.
4	Staf IT	Sebagai Staf IT saya ingin sistem dapat mengeluarkan sebuah dokumen <i>Purchasing Request</i> dalam bentuk PDF sehingga selanjutnya dokumen tersebut bisa diserahkan kepada bagian finance untuk selanjutnya diproses.
5	Staf IT	Sebagai Staf IT saya ingin dapat menginput data aset baru pada sistem yang selanjutnya akan memberikan ID aset baru berbentuk QR yang nantinya QR dapat di cetak. Sehingga aset memiliki data yang faktual. QR pada aset
6	Staf IT	Sebagai Staf IT saya ingin aset memiliki beberapa status seperti <i>preparing</i> , <i>deployed</i> , <i>temp substitution</i> , <i>on repair</i> , dan <i>scrap</i> . Sehingga data tentang aset tetap aktual.
7	Staf IT	Sebagai Staf IT saya ingin agar tiket di assign secara otomatis berdasarkan tipe masalah, sehingga membuat workflow lebih efisien.
8	Staf IT	Sebagai Staf IT saya ingin mendapatkan notifikasi ketika ada tiket baru yang dibuat. Sehingga saya dapat mengetahuinya dengan segera.
9	Supervisor IT	Sebagai Supervisor IT saya ingin pengajuan aset harus mendapatkan persetujuan dari saya, sehingga pengadaan aset tetap dalam pengawasan saya.
10	Supervisor IT	Sebagai Supervisor IT saya ingin mendapatkan notifikasi apabila ada pengajuan aset baru, sehingga saya dapat menindaklanjutinya dengan segera.
11	Supervisor IT	Sebagai <i>IT Supervisor</i> saya ingin segala aktifitas penanganan tiket dihitung waktunya, sehingga saya dapat mengetahui dan mengukur peforma tim.
12	User	Sebagai <i>User</i> saya ingin agar saya dapat mengganti password saya sendiri, agar keamanan akun tetap terjaga.
13	User	Sebagai <i>User</i> saya ingin dapat melakukan pengajuan aset menggunakan aplikasi, sehingga proses pengajuan aset menjadi lebih efisien.
14	User	Sebagai <i>User</i> saya ingin dapat melaporkan masalah pada aset IT saya dengan membuat tiket pelaporan yang selanjutnya akan ditindak lanjuti oleh divisi IT, sehingga mempersingkat proses pelaporan.
15	User	Sebagai <i>User</i> saya ingin dapat melihat progress/status perbaikan aset IT saya, sehingga saya dapat terus memantau status perbaikan aset IT saya.

3.2 Rancangan Sistem Portal ITSM

Portal ITSM ini merupakan aplikasi *multi-platform* yang fungsi utamanya mengontrol dan mengawasi layanan IT yang berjalan pada suatu organisasi atau perusahaan. Portal akan dibangun pada platform android mengingat staf IT memiliki mobilitas tinggi dan juga platform web untuk kebutuhan pengolahan data master dan juga *reporting*. Fitur utama dari aplikasi ini adalah gabungan dari IT *Asset Management* dan *Ticketing* yang terintegrasi satu sama lain. Dengan mengkombinasikan Flutter sebagai *front-end* dan Laravel sebagai *back-end* sistem, diharapkan proses development dari aplikasi ini dapat dijalankan dengan cepat dan efisien. Berikut ini merupakan rancangan sistem portal ITSM RepIT menggunakan *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*.

1) Use Case Diagram

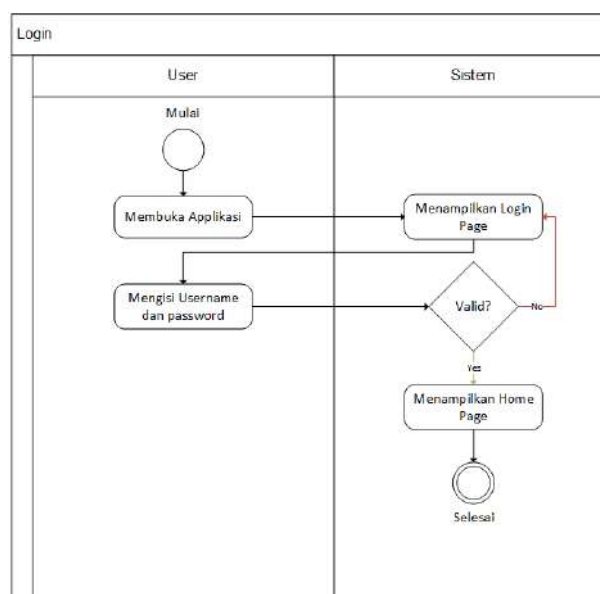
Di bawah ini merupakan diagram *use case* yang sudah dirancang:



Gambar 3. Use Case Diagram RepIT

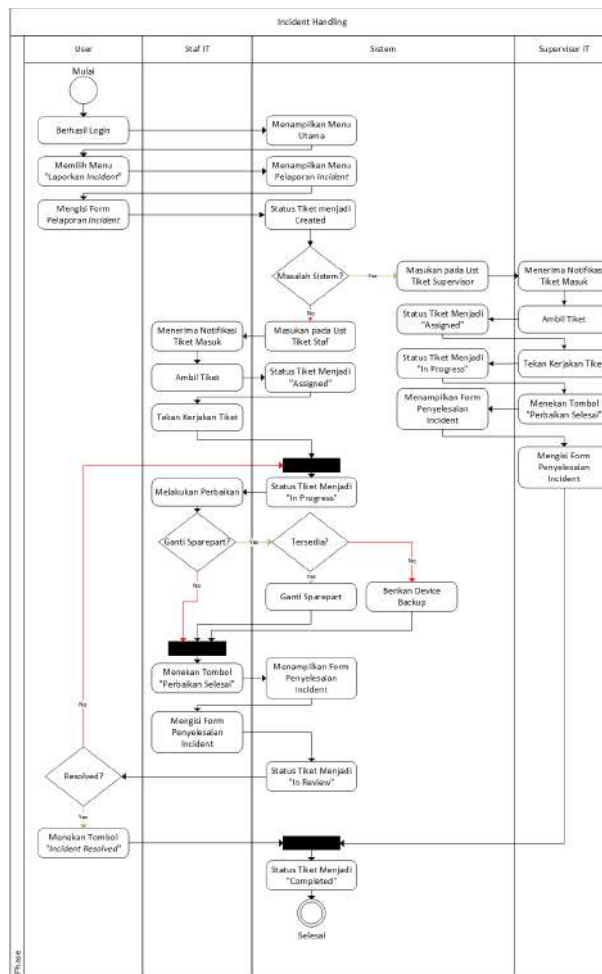
2) Activity Diagram

a. Login

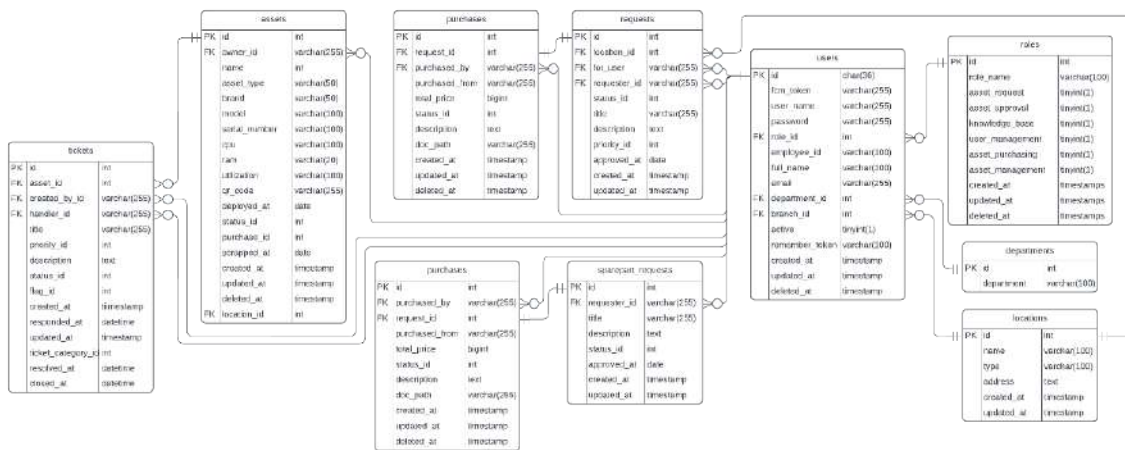


Gambar 4. Activity Diagram Login

d. Incident Handling

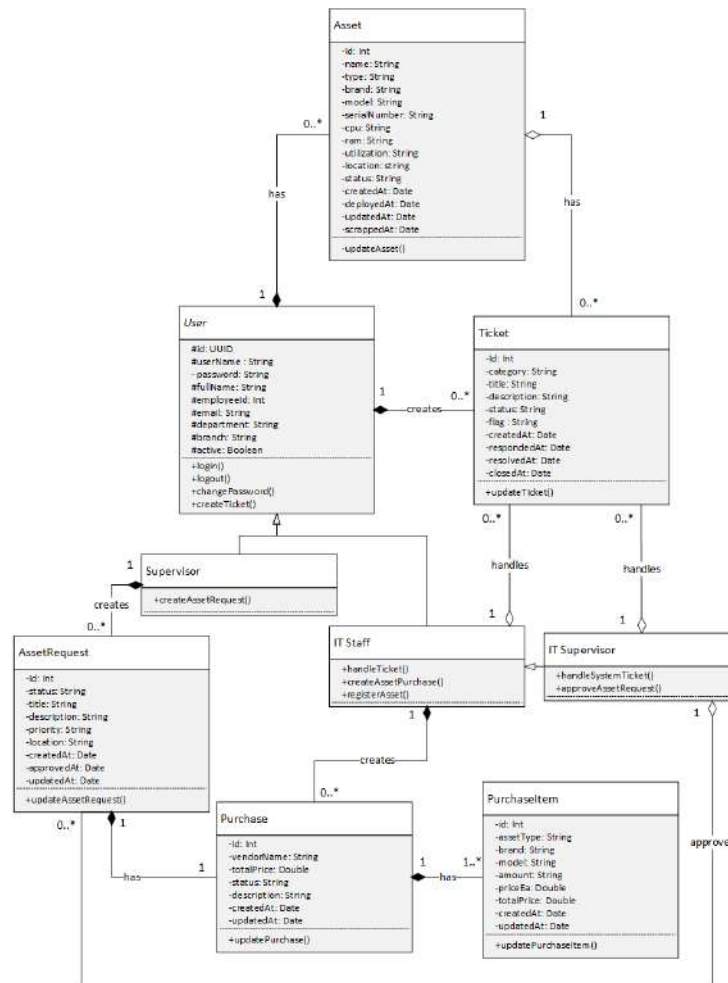


Gambar 7. Activity Diagram Incident Handling



Gambar 8. ERD RepIT

3) Class Diagram



Gambar 9. Class Diagram RepIT

3.3 Entity-Relationship Diagram

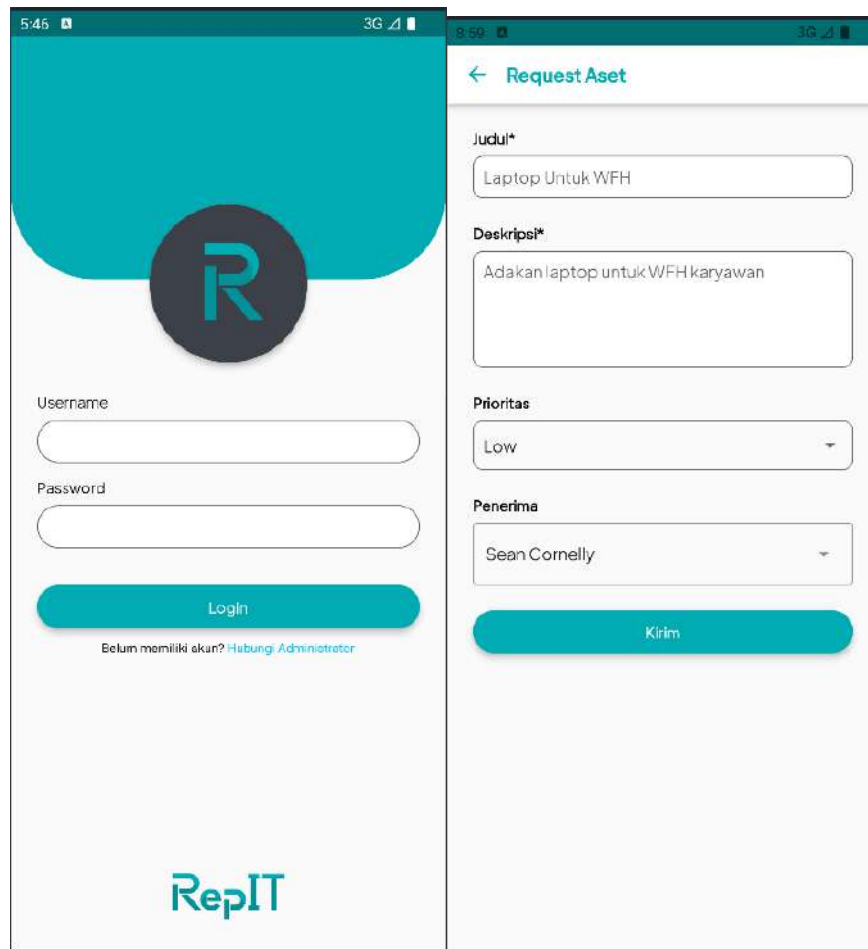
Entity-Relationship Diagram (ERD) adalah representasi visual dari model data yang menggambarkan entitas, atribut, hubungan, dan kardinalitas dalam suatu sistem basis data. Berikut adalah ERD dari basis data yang dimiliki oleh RepIT:

3.3.1 Rancangan User Interface

Di bawah ini merupakan rancangan *user interface* dari aplikasi RepIT:

1) Login Page

Login page merupakan halaman pertama yang ditampilkan setelah membuka aplikasi apabila belum *login*. Apabila *User* sudah *login* dan belum *logout* maka token akan tersimpan pada *memory* perangkat dan aplikasi akan menampilkan *home page* setelah *User* membuka aplikasi.



Gambar 10. Login Page Page dan Create Asset Request Form

2) Home Page

Setelah *login*, *User* diarahkan ke home page. Pada *home page* terdapat 3 (tiga) *tab*, yaitu: *My Assets*, *Tickets*, dan *Request*. Pada *home page* juga terdapat *action* atau menu tambahan seperti *floating action button* yang *action*-nya berisi *speed dial*, *drawer* yang berisi menu-menu tambahan, serta *profile* yang berisi detail *User* yang *ter-login*. Penjelasan tentang *action-action* tambahan tersebut akan dibahas pada poin-poin selanjutnya.



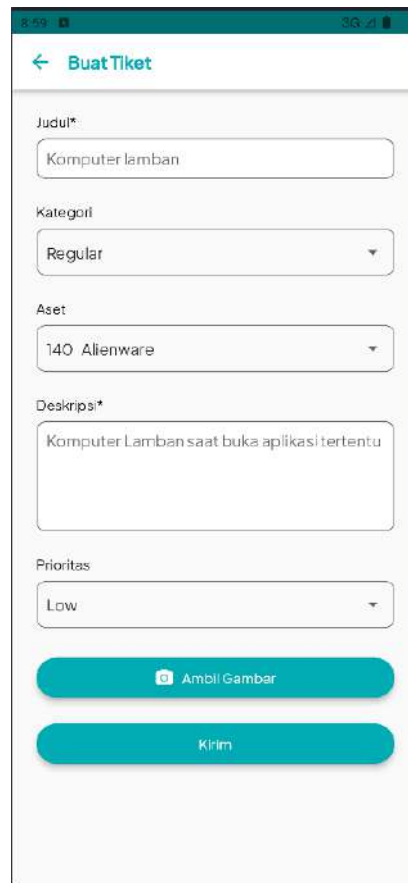
Gambar 11. Home Page

3) Create Asset Request Form

Form yang terdapat pada halaman ini berfungsi untuk membuat permintaan aset baru. Pada *form* ini dapat diatur prioritas permintaan dan penerima aset yang ditujukan dari permintaan aset tersebut.

4) Create Ticket Form

Form pada halaman ini berfungsi untuk membuat tiket baru. Pada halaman ini juga terdapat fitur ambil gambar yang mana berfungsi untuk mengambil foto yang berhubungan dengan tiket tersebut. Foto yang diambil dibatasi hingga 5 (lima) foto.



Gambar 12. *Create Ticket Form*

5) *Asset Detail Page*

Halaman ini memuat informasi *detail* aset. Apabila halaman ini diakses dari *manage asset* oleh divisi IT, maka akan ada 3 (tiga) *button* yang berada di bawah table informasi aset, yaitu: pindah aset, ambil aset dari user, dan *scrap* aset. Halaman ini memiliki 2 (dua) *action* tambahan pada *appbar*-nya, yaitu *repair history* dan *qr code*.



Gambar 13. *Asset Detail Page*

6) *Ticket Detail Page*

Halaman ini memuat detail dari tiket yang sudah dibuat oleh User. Apabila role User merupakan staf IT atau supervisor IT, akan ada button action tambahan yang terletak dibawah gambar atau informasi tiket.



Gambar 14. Ticket Detail Page

3.3.2 Pengkodean Sistem

Setelah perencanaan dan perancangan sudah dilakukan sesuai dengan paradigma *extreme programming*, tahap selanjutnya merupakan pengkodean sistem (*coding*). Pengkodean sistem dilakukan secara paralel antara *front-end* dan *back-end*nya. Untuk pengkodean *front-end* akan dilakukan menggunakan *software development kit* (SDK) Flutter dan kerangka kerja Laravel untuk pengkodean sistem *back-end* yang menghasilkan API yang nantinya akan dikonsumsi oleh *front-end*. Perangkat lunak yang digunakan untuk pengkodean Flutter adalah Android Studio dengan pertimbangan *tools* yang dimilikinya untuk mendukung pengembangan aplikasi mobile. Perangkat lunak yang digunakan untuk pengkodean Laravel adalah Visual Studio serta DBeaver sebagai *DB viewer* untuk database yang akan digunakan, yaitu MySQL.

3.3.3 Pengujian (Testing)

Pengujian sistem RepIT menggunakan 2 (dua) pendekatan yaitu *black-box Testing* dan SUS (*System Usability Score*). Berikut adalah hasil dari test yang sudah dilakukan:

1) Black-Box Testing

Di bawah ini merupakan beberapa *test case* yang diujikan kepada system:

Tabel 2. *Sample Black-Box Testing*

No	Pengujian	Test Case	Hasil yang diharapkan	Hasil
1	<i>Login</i>	Memasukan <i>username</i> dan <i>password</i>	Menampilkan <i>home page</i> apabila berhasil login, dan menampilkan <i>error message</i> apabila gagal	<i>Valid</i>
2	<i>Create Ticket</i>	Menekan <i>speed dial ticket</i> dan menampilkan <i>create ticket page</i> , selanjutnya mengisi semua field yang diperlukan.	Semua <i>dropdown</i> berfungsi sebagaimana mestinya, serta fungsi ambil gambar yang berfungsi dengan baik lalu menampilkan pesan berhasil setelah menekan tombol kirim.	<i>Valid</i>
3	<i>Create Asset Request</i>	Menekan <i>speed dial request</i> dan menampilkan <i>create asset requets page</i> , selanjutnya mengisi semua field yang diperlukan.	Semua <i>dropdown</i> berfungsi sebagaimana mestinya serta menampilkan pesan berhasil setelah menekan tombol kirim.	<i>Valid</i>
4	<i>Self-assign Ticket</i>	Membuka <i>ticket detail page</i> dengan cara menekan <i>card</i> yang berada pada <i>manage ticket page</i> lalu menekan tombol ambil tiket	Status tiket berubah menjadi <i>assigned</i> serta <i>ticket handler</i> berubah menjadi nama lengkap <i>User</i> yang mengambil tiket tersebut	<i>Valid</i>
5	<i>Resolve Ticket</i>	Menekan tombol pengerjaan selesai pada <i>ticket detail page</i> dan mengisi note penyelesaian	Status tiket berubah menjadi <i>In Review</i> serta note penyelesaian terisi pada tiket.	<i>Valid</i>
6	<i>Create Purchase Page</i>	Menekan tombol ajukan pembelian pada <i>request detail page</i>	Menampilkan <i>purchasing form</i> dengan nomor permintaan yang sesuai berdasarkan nomor permintaan yang tercantum pada <i>request detail page</i> .	<i>Valid</i>
7	<i>Logout</i>	Menekan menu <i>logout</i> pada <i>app drawer</i>	<i>Token</i> pada <i>device</i> terhapus dan aplikasi menampilkan <i>login page</i>	<i>Valid</i>

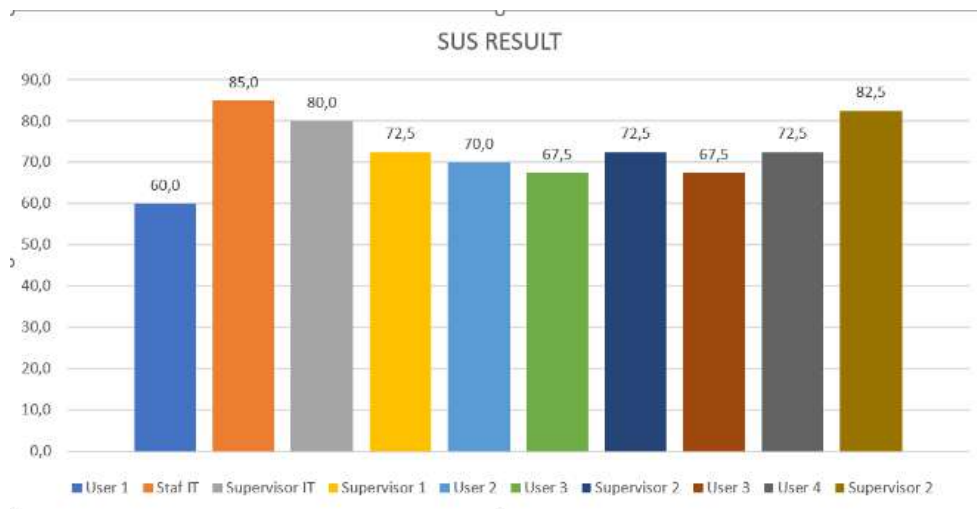
Di bawah ini merupakan kesimpulan dari *black-box testing* yang sudah dilakukan:

Tabel 3. Kesimpulan *Black-Box Testing*

Total Test Case	Total Test Case Berhasil	Kesimpulan
75	75	OK

2) *System Usability Score (SUS)*

Di Bawah ini adalah hasil dari asesmen *SUS (System Usability Score)* yang sudah dilakukan:



Gambar 15. Hasil SUS

Berdasarkan dengan hasil score rata-rata 73 (tujuh puluh tiga), dapat disimpulkan bahwa usability yang dimiliki oleh sistem RepIT adalah baik menurut [12].

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan terkait dengan Perancangan Sistem Portal IT Service Management Berbasis Flutter (Studi Kasus CV. Sabar Maju) maka dapat disimpulkan sebagai berikut:

- 1) Adanya sistem monitoring layanan IT yang berjalan pada CV. Sabar Maju menghasilkan data yang aktual terkait dengan layanan IT yang berjalan pada CV. Sabar Maju.
- 2) Telah berhasil dibangun sistem portal ITSM (RepIT) yang dapat membantu CV. Sabar Maju dalam pengelolaan pelayanan IT yang berjalan pada perusahaan.
- 3) Terdapat fitur penghitungan SLA yang dijalankan terjadwal satu bulan sekali yang mana dapat mengukur performa layanan IT setiap bulannya. Fitur ini dapat menjadi bahan evaluasi divisi IT untuk terus meningkatkan layanannya.
- 4) Pembangunan aplikasi berbasis android memenuhi kebutuhan divisi IT yang memiliki mobilitas tinggi. Divisi IT juga memerlukan portal web untuk keperluan *reporting* dan juga olah data master. Kedua hal tersebut dapat terpenuhi dengan 1 (satu) basis kode Flutter yang memiliki dukungan penuh terhadap pembangunan aplikasi *multiplatform*.

Hasil Pengujian sistem dengan metode *black-box* sesuai dengan yang diharapkan dan *system usability score* yang didapatkan adalah 73 yang berarti *usability* dari sistem RepIT adalah baik.

REFERENSI

- [1] A. McAfee and E. Brynjolfsson, *Machine, Platform, Crowd: Harnessing Our Digital Future*. New York: W. W. Norton & Company, 2017.
- [2] IBM, "IT Service Management (ITSM)," IBM.
- [3] A. Arif, S. Sukuryadi, and F. Fatimaturrahmi, "Pengaruh Ketersediaan Sumber Belajar Di Perpustakaan Sekolah Terhadap Motivasi Belajar Siswa Pada Mata Pelajaran Ips Terpadu Smp Negeri 1 Praya Barat," *JISIP (Jurnal Ilmu Sosial dan Pendidikan)*, vol. 1, no. 2, Mar. 2019, doi: 10.58258/jisip.v1i2.184.

- [4] I. Suryantara, *Merancang Aplikasi dengan Metodologi Extreme Programmings*. Elex Media Komputindo, 2017.
- [5] F. Anwer, S. Aftab, S. Muhammad, and U. Waheed, “Comparative Analysis of Two Popular Agile Process Models: Extreme Programming and Scrum,” *International Journal of Computer Science and Telecommunications*, 2017.
- [6] G. Maulani, M. Karina, and I. Setiawan, “Sistem Informasi UKKO untuk Peningkatan Kinerja Pegawai Studi Kasus PT.PLN (Persero) Tangerang,” *Journal CCIT*, vol. 12, no. 1, 2019.
- [7] J. Osis and U. Donniss, *Topological UML Modeling: An Improved Approach for Domain Modeling and Software Development*. Amsterdam: Elsevier, 2017.
- [8] E. Windmill, *Flutter in action*. Simon and Schuster, 2020.
- [9] S. Fernando, R. Tanaamah, and A. Wijaya, “Rancang Bangun Sistem Informasi Manajemen Mutu Perusahaan Menggunakan Framework Laravel dan Materialize (Studi Kasus: Bagian Pengendalian Dokumen PT. Pura Barutama Divisi Boxindo, Kudus),” *Journal CCIT*, vol. 10, no. 1, 2017.
- [10] Y. Supardi and Sulaeman, *Semua Bisa Menjadi Programmer Laravel Basic*. Jakarta: Elex Media Komputindo, 2019.
- [11] R. Noviana, “Pembuatan Aplikasi Penjualan Berbasis Web Monja Store Menggunakan PHP dan MySQL,” *Jurnal Teknik dan Science*, pp. 114–115, 2022.
- [12] J. R. Lewis and J. Sauro, “Item benchmarks for the system usability scale.,” *J Usability Stud*, vol. 13, no. 3, 2018.