

**RANCANG BANGUN APLIKASI PENDATAAN TRANSAKSI JASA  
PENYEBERANGAN DENGAN FRAMEWORK REACT NATIVE  
BERBASIS ANDROID (STUDI KASUS: B.C. BAKAUHENI)**

*The Design of Crossing Service Transaction Data Collection Application for  
Framework Service with React Native Framework Based on Android (Case Study:  
Bc. Bakauheni)*

Usulan penelitian untuk skripsi S-1

Diusulkan oleh:  
Rahmad Suryadi  
17312031



**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNIK ILMU KOMPUTER  
UNIVERSITAS TEKNOKRAT INDONESIA  
BANDAR LAMPUNG  
2022**

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Allah SWT, karena atas berkat rahmat-Nya, penulis dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar sarjana pada Program Studi S1 Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan laporan ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

- 1 Bapak Dr. H. M. Nasrullah Yusuf, S.E., M.B.A., selaku Rektor Universitas Teknokrat Indonesia
- 2 Bapak H. Mahathir Muhammad, S.E., M.M., selaku Dekan Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia.
- 3 Bapak Andi Nurkholis, S.Kom., M.Kom. selaku pembimbing yang telah meluangkan waktu untuk membimbing penulis menyelesaikan skripsi ini.
- 4 Bapak/Ibu selaku dosen penguji.

Akhir kata, penulis berharap semoga Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu dan semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Bandar Lampung, Mei 2022

Penulis

## DAFTAR ISI

	Hal.
KATA PENGANTAR .....	i
DAFTAR ISI .....	ii
DAFTAR TABEL .....	iv
DAFTAR GAMBAR .....	v
1 PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
2 LANDASAN TEORI .....	4
2.1 Tinjauan Pustaka .....	4
2.2 Pengertian Android.....	9
2.3 Framework .....	9
2.4 React Native.....	10
2.5 QR-code .....	10
2.6 Metode Pengembangan Sistem .....	11
2.7 <i>Unified Modeling Language (UML)</i> .....	13
2.7.1 <i>Use Case</i> .....	13
2.7.2 <i>Activity Diagram</i> .....	14
2.7.3 <i>Class diagram</i> .....	15
2.8 <i>Class Responsibility Collaborator Card (CRC Card)</i> .....	16
2.9 Firebase.....	17
2.10 Pengujian ISO 25010.....	17
2.11 Pengertian Figma.....	19
3 METODOLOGI PENELITIAN.....	20
3.1 Kerangka Penelitian.....	20
3.2 <i>Planning</i> .....	21
3.2.1 <i>User Stories</i> .....	21
3.3.2 <i>Values</i> .....	23
3.3.3 <i>Acceptance Test Criteria</i> .....	24
3.3.4 <i>Iteration Plan</i> .....	27
3.3.4.2. Analisis Kebutuhan Sistem .....	27

3.4. <i>Design</i> .....	29
3.4.1 <i>Use Case Diagram</i> .....	29
3.4.2 <i>Activity Diagram</i> .....	32
3.4.3 <i>Class Diagram</i> .....	46
3.4.1 <i>Class Responsibility Collaborator Card (CRC Card)</i> .....	47
3.4.2 <i>Prototype Design</i> .....	50
3.5 <i>Jadwal Penelitian</i> .....	57
DAFTAR PUSTAKA .....	58
LAMPIRAN .....	60

## DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka .....	4
Tabel 2.2 Simbol-Simbol <i>Use Case</i> .....	14
Tabel 2.3 Simbol – Simbol <i>Activity Diagram</i> .....	15
Tabel 2.4 Simbol – Simbol <i>Class Diagram</i> .....	16
Tabel 2.5 <i>Class Responsibility Collaborator Card (CRC Card)</i> .....	17
Tabel 3.1 <i>User Stories</i> .....	21
Tabel 3.2 <i>User Stories</i> (lanjutan 1).....	22
Tabel 3.3 <i>User Stories</i> (lanjutan 2).....	23
Tabel 3.4 Kerangka Pengujian <i>Co-existence</i> .....	25
Tabel 3.5 Kerangka Pengujian <i>Operating System Dan Device</i> .....	26
Tabel 3.6 Penjelasan <i>Use Case</i> .....	31
Tabel 3.7 <i>CRC Card AUTH</i> .....	47
Tabel 3.8 <i>CRC Card Pegawai</i> .....	48
Tabel 3.9 <i>CRC Card Pimpinan</i> .....	48
Tabel 3.10 <i>CRC Card Transaksi Penyeberangan</i> .....	49
Tabel 3.11 <i>CRC Card Kendaraan</i> .....	49
Tabel 3.12 <i>CRC Card rekap</i> .....	49
Tabel 3.13 <i>CRC Card Transaksi Kreditur</i> .....	49

## DAFTAR GAMBAR

Gambar 2.1 Logo android .....	9
Gambar 2.2 Logo React Native .....	10
Gambar 2.3 Metode Extreme Programming .....	11
Gambar 2.4 Model ISO 25010 .....	18
Gambar 3.1 Kerangka Penelitian.....	20
Gambar 3.2 <i>Use Case Diagram</i> Aplikasi Pendataan .....	30
Gambar 3.3 <i>Activity Diagram</i> Mengelola Data Transaksi .....	32
Gambar 3.4 <i>Activity Diagram</i> Menampilkan Data Transaksi .....	33
Gambar 3.5 <i>Activity Diagram</i> Memfilter Data Transaksi .....	34
Gambar 3.6 <i>Activity Diagram</i> Pencarian Data Transaksi .....	35
Gambar 3.7 <i>Activity Diagram</i> Menampilkan Data Kendaraan dan Pegawai .....	36
Gambar 3.8 <i>Activity Diagram</i> Mengelola Data Kendaraan dan Pegawai.....	37
Gambar 3.9 <i>Activity Diagram</i> Melakukan Transaksi Penyeberangan .....	38
Gambar 3.10 <i>Activity Diagram</i> Melakukan Transaksi Kredit.....	39
Gambar 3.11 <i>Activity Diagram</i> Menampilkan Rekap Harian .....	40
Gambar 3.12 <i>Activity diagram</i> Menampilkan dan mengelola saldo.....	41
Gambar 3.13 <i>Activity Diagram</i> Menampilkan dan Mengelola Harga Pelabuhan .	42
Gambar 3.14 <i>Activity Diagram</i> Menambahkan Pengeluaran.....	43
Gambar 3.15 <i>Activity Diagram</i> Filter Data Kendaraan.....	44
Gambar 3.16 <i>Activity Diagram</i> Pencarian Data Kendaraan.....	45
Gambar 3.17 <i>Activity Diagram</i> Simpan QR-Code Kendaraan .....	46
Gambar 3.18 <i>Class diagram</i> aplikasi pendataan transaksi.....	47
Gambar 3.19 Tampilan <i>Login</i> .....	50
Gambar 3.20 Halaman Home .....	51
Gambar 3.21 Halaman Data Transaksi .....	51
Gambar 3.22 Halaman <i>Scanner</i> .....	52
Gambar 3.23 Halaman Transaksi Penyeberangan .....	53
Gambar 3.24 Halaman Transaksi <i>Kredit</i> .....	53
Gambar 3.25 Halaman Data Kendaraan Dan Pegawai .....	54
Gambar 3.26 Halaman Harga Pelabuhan .....	55
Gambar 3.27 Halaman Tambah Pengeluaran Harian .....	55
Gambar 3.28 Halaman Rekap Harian .....	56
Gambar 3.29 Halaman Edit Saldo .....	57

# **1 PENDAHULUAN**

## **1.1 Latar Belakang**

Penggunaan komputer merupakan suatu hal yang penting di berbagai kehidupan dengan berbagai komunitas saat ini. Sehingga dapat disimpulkan bahwa komputer menjadi media informasi yang dapat membantu suatu pekerjaan yang rumit. Bukan hanya pada lingkungan kerja, komputer juga dapat memudahkan pengguna dalam kehidupan sehari-harinya. Seiring dalam perkembangan teknologi, kemajuan perangkat lunak dan perangkat keras berjalan dengan baik, perkembangan ini berpengaruh dalam hal pembuatan aplikasi yang semakin lebih mudah dan cepat. Hal ini pun membuat meningkatnya pemanfaatan komputer pada lingkungan masyarakat dan tentunya pada lingkungan kerja. (Loveri 2018).

Jasa merupakan suatu aktivitas bukan berupa benda yang ditawarkan oleh penyedia jasa ke pihak lain (Gultom dkk., 2014). Berdasarkan observasi peneliti pada jasa penyeberangan, layanan yang ditawarkan jasa berupa pembayaran tiket secara *online* dan juga membantu menangani kendala-kendala yang terjadi pada truk yang menghambat perjalanan. Pencatatan transaksi pada jasa penyeberangan B.C. Bakauheni masih dilakukan dengan cara konvensional yang membuat pencatatan tidak efisien yang mengharuskan pegawai mencatat data kendaraan di setiap transaksinya, rentan terjadinya kesalahan dalam pencatatan data, pencarian data transaksi yang sulit, dan kekhawatiran terjadi hilangnya data. Sebelumnya pencatatan pernah dilakukan dengan menggunakan komputer, hal ini dirasa kurang efisien karena pegawai harus mencatat pada buku catatan saat di lapangan dan kemudian melakukan pencatatan lagi setelah kembali ke kantor dan keluhan tagihan listrik yang meningkat.

Pada Penelitian sebelumnya telah banyak mengembangkan aplikasi untuk menangani permasalahan pendataan dengan memanfaatkan aplikasi *mobile*. Pertama, pengembangan aplikasi pengelolaan uang berbasis android dengan melakukan pencatatan terperinci dan menghitung pemasukan dan pengeluaran untuk mengontrol keuangan pengguna (Juhardi and Khairullah 2019) Selanjutnya pengembangan aplikasi pendataan pada CV. Nana Beautyskin dapat berguna dalam pendataan produk, pencatatan transaksi pencatatan bonus pegawai dan keefektifan

dalam pembuatan struk (Saputra, dkk. 2021). Selanjutnya perancangan aplikasi kasir *point of sales* berbasis android untuk gerai makanan yang dapat memudahkan dalam perekapan harian (Iskandar & Abdurrahman, 2020). Selanjutnya pengembangan aplikasi pengelolaan pinjaman berbasis *mobile* pada koperasi PKK sejahtera sukabumi yang membuat pembukuan simpan pinjam dan pengajuan pinjaman menjadi lebih efektif (Maranti dkk., 2018). Selanjutnya pengembangan sistem pencatatan keuangan toko berbasis android meningkatkan efisiensi pengelolaan keuangan (Romadony dkk., 2019).

Pada penelitian terdahulu yang telah dipaparkan, mendapatkan hasil bahwasanya penggunaan aplikasi berbasis android dapat membantu dalam pencatatan dan pengelolaan data transaksi sehingga proses bisnis dapat berjalan lebih efisien. Selain itu dengan adanya sistem dengan *database online* pengguna dapat melihat hasil transaksi kapanpun dan dimanapun. Pengembangan yang dapat dilakukan dari penelitian sebelumnya adalah menambahkan fitur pemindai *QR-code* dalam melakukan transaksinya, dan juga menampilkan secara *realtime* hasil transaksi yang telah terjadi.

Penelitian ini bertujuan mengembangkan aplikasi pendataan transaksi dengan studi kasus jasa penyeberangan B.C. Bakauheni. aplikasi dikembangkan menggunakan metode *extreme programming* dengan empat tahapan yaitu *planning*, *design*, *coding* dan *testing*. Metode ini dipilih karena fleksibilitas yang tinggi terhadap *user* dan klien dalam pengembangannya. sehingga mampu membuat aplikasi yang sesuai dengan fungsi yang dibutuhkan klien (Widodo 2008). Cakupan fungsi yang dimaksud yaitu mencakup pencatatan data dengan memindai *QR-code*, pencarian data dengan mudah, rekap harian dan menampilkan data transaksi secara *realtime*. Pada tahap *coding* dalam pengembangan aplikasi menggunakan bahasa pemrograman JavaScript dengan *framework* React Native dan pengujian aplikasi menggunakan ISO 25010. Aplikasi ini diharapkan dapat menyelesaikan masalah-masalah pendataan yang terjadi dan dapat meningkatkan kinerja pegawai dalam setiap transaksinya sehingga proses bisnis dapat berjalan lebih efisien.



## 1.2 Rumusan Masalah

Berdasarkan dari pemaparan yang telah penulis jelaskan diatas, maka rumusan masalah yang diambil penulis adalah bagaimana mengembangkan sebuah aplikasi berbasis android yang dapat membantu jasa penyeberangan dalam hal pencatatan, pencarian, penyimpanan, dan perekapan data transaksi?

## 1.3 Batasan Masalah

Dalam penelitian rancang bangun aplikasi pendataan transaksi pada jasa penyeberangan ini ditentukan batasan masalah sebagai berikut:

1. Aplikasi hanya melakukan pendataan transaksi, pencarian, dan perekapan data transaksi pada jasa penyeberangan B.C. Bakauheni.
2. Aplikasi di bangun menggunakan *framework* React Native dan *database* Firebase.

## 1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk merancang dan mengembangkan sebuah aplikasi pendataan transaksi untuk jasa penyeberangan B.C Bakauheni yang dapat melakukan pencatatan, pencarian, penyimpanan, dan perekapan data transaksi untuk meningkatkan efisiensi dalam proses bisnisnya.

## 1.5 Manfaat Penelitian

Adapun manfaat yang diperoleh secara langsung maupun tidak langsung dari penelitian ilmiah ini adalah sebagai berikut:

1. Bagi penulis dapat meningkatkan kemampuan menyelesaikan masalah di suatu kasus dengan memanfaatkan teknologi dan juga meningkatkan kemampuan pengembangan perangkat lunak berbasis android khususnya menggunakan *framework* react native.
2. Bagi jasa penyeberangan B.C. Bakauheni manfaat penelitian ini diharap dapat memudahkan jasa penyeberangan dalam menyelesaikan masalah - masalah pendataan yang ada dan dapat membantu meningkatkan efisiensi dalam proses bisnisnya.

## 2 LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Pada penelitian ini menggunakan tinjauan Pustaka dari penelitian sebelumnya yang nantinya akan digunakan sebagai pendukung penelitian, tinjauan Pustaka yang digunakan penulis dalam penelitian ini dapat dilihat pada Gambar 2.1.

**Tabel 2.1** Tinjauan Pustaka

Nomor	Detail Jurnal	
1	Judul	Sistem Pencatatan dan Pengelolaan Keuangan Pada Aplikasi Manajemen Keuangan E-Dompet Berbasis Android
	Tahun Terbit	2019
	Penulis	Ujang Juhardi, Khairullah
	Metode Penelitian	Waterfall
	Analisis Masalah	Dalam penelitian ini disebutkan bahwa pencatatan pemasukan dan pengeluaran uang pribadi pada umumnya masih dilakukan dengan cara konvensional. hal tersebut sangat tidak efisien, jumlah transaksi yang tidak sedikit di setiap harinya membuat pencatatan, perhitungan, dan pembuatan laporan membutuhkan waktu yang tidak sedikit, maka dibutuhkan aplikasi pengelolaan keuangan yang dapat digunakan seseorang untuk mempermudah pencatatan laporan keuangan.
	Hasil	Hasil dari penelitian ini adalah sebuah Aplikasi Manajemen Keuangan E-Dompet Berbasis Android yang dapat melakukan pencatatan laporan keuangan secara rinci sehingga memudahkan seseorang melihat rincian laporan keuangannya.

	Kelebihan	Penyimpanan data pada aplikasi ini bersifat <i>offline</i> sehingga pengguna tidak harus menggunakan internet setiap menggunakan aplikasi.
	Kekurangan	Karena penyimpanan yang digunakan <i>offline</i> atau local, jika pengguna menggunakan <i>device</i> yang berbeda tidak dapat membuka catatan yang sudah ada sebelumnya.
2	Judul	Perancangan Aplikasi Kasir <i>Point of Sales</i> Berbasis Android Menggunakan Metode Rapid Application Development Untuk Usaha Retail
	Tahun Terbit	2020
	Penulis	Iskandar, Umar Tsani Abdurrahman
	Metode Penelitian	Rapid Application Development
	Analisis Masalah	Dalam penelitian ini disebutkan bahwa kegiatan transaksi pada gerai pada umumnya dilakukan belum menggunakan sistem komputer, sehingga banyak menemukan kendala, yaitu dalam proses penghitungan sering terjadi kesalahan hitung dan proses rekap transaksi yang relatif lama karena harus menghitung ulang setelah gerai selesai berjualan sehingga membutuhkan waktu extra.
	Hasil	Sebuah aplikasi kasir <i>point of sales</i> berbasis android yang dapat digunakan gerai dalam melakukan pencatatan transaksi dan dapat mempermudah kegiatan transaksinya.
	Kelebihan	Aplikasi dapat terhubung dengan printer melalui bluetooth sehingga dapat Melakukan pencetakan struk penjualan
	Kekurangan	<i>Interface</i> pada aplikasi dirasa kurang menarik dan pemilihan <i>color palette</i> yang kurang selaras atau terlalu kontras.

3	Judul	Rancang Bangun Aplikasi Pengelolaan Pinjaman Koperasi Berbasis Mobile Pada Koperasi Pkk Sejahtera Sukabumi
	Tahun Terbit	2018
	Penulis	Ovi Sovia Maranti, Lis Saumi Ramdhani, Rusli Nugraha, Khairul Rizal
	Metode Penelitian	Waterfall
	Analisis Masalah	Dalam penelitian ini disebutkan bahwa jumlah anggota yang cukup banyak pada koperasi PKK Sejahtera Sukabumi mengakibatkan sistem pelaporan, pembukuan simpan pinjam dan pengajuan pinjaman menjadi kurang efektif, karena masih menggunakan cara konvensional, terutama dalam pengajuan pinjam, hal tersebut menyebabkan perputaran modal berjalan lambat, oleh karena itu dibutuhkan sistem terkomputerisasi agar anggota dapat mengajukan pinjaman kapanpun dan dimana saja.
	Hasil	Hasil dari penelitian ini yaitu aplikasi pengelolaan pinjaman koperasi berbasis <i>mobile</i> yang dibuat terintegrasi dengan web server supaya memudahkan dalam pengelolaan datanya.
	Kelebihan	Penyimpanan data pinjaman yang ditegrasikan dengan web server dapat memudahkan dalam pengolahan data.
	Kekurangan	Halaman admin dan anggota dibuat pada platform yang berbeda sehingga tidak efisien dan user interface yang dirasa masih kurang baik.
	Judul	Pengembangan Aplikasi Klinik Kecantikan sebagai Pengelola Transaksi berbasis Android

4		menggunakan Metode Prototype (Studi Kasus: Klinik Kecantikan CV Nana Beautyskin).
	Tahun Terbit	2021
	Penulis	Agung Dwi Saputra, Agi Putra Kharisma, Lutfi Fanani.
	Metode Penelitian	Prototype
	Analisis Masalah	Dalam penelitian ini disebutkan bahwa kegiatan pencatatan transaksi yang dilakukan pada CV. Nana Beauty Skin masih dilakukan dengan cara tradisional, metode pencatatan tersebut masih dirasa efektif saat kegiatan transaksi masih sedikit, namun saat kegiatan transaksi yang mulai banyak hal tersebut memiliki beberapa resiko dalam pelaksanaannya. Sehingga dibutuhkan sistem pengelolaan data transaksi yang lebih baik dari sebelumnya.
	Hasil	Aplikasi klinik kecantikan sebagai pengelola transaksi berbasis Android, yang dapat digunakan untuk pencatatan setiap transaksi, pencatatan bonus pegawai pada setiap pelayanan jasa, dan pembuatan struk yang lebih efektif.
	Kelebihan	Penyimpanan yang sudah secara online, UI yang menarik dan pengujian aplikasi yang mendapat nilai yang bagus.
	Kekurangan	Pengujian kompatibilitas aplikasi yang tidak dilakukan pada <i>real device</i> .
	Judul	Sistem Pencatatan Keuangan Toko Berbasis Android
	Tahun Terbit	2019

5	Penulis	Willi Alham Romadony, Muhammad Ardianto, Wisnu Kartiko Arie Pangestu, Didih Rizki Chandranegara, Wildan Suharso
	Metode Penelitian	Waterfall
	Analisis Masalah	pengelolaan keuangan dengan cara konvensional dirasa kurang efektif dan juga memerlukan waktu yang lebih untuk mencatat pengeluaran yang telah dilakukan saat itu sehingga terkadang lupa untuk melakukan pencatatan pengeluaran kecil yang telah dilakukan.
	Hasil	aplikasi pencatatan keuangan berbasis android yang dapat digunakan dalam mengelola keuangan.
	Kelebihan	Terdapat rekap harian yang sudah langsung terekap dan dapat di cetak
	Kekurangan	Tidak disebutkan penyimpanan yang digunakan dan juga UI yang masih kurang menarik.

Kesimpulan yang dapat diambil dari tinjauan literatur diatas yaitu masalah yang muncul karena pengelolaan data transaksi yang masih menggunakan cara konvensional dapat teratasi dengan sistem yang terkomputerisasi dengan memanfaatkan aplikasi *mobile* berbasis android sehingga proses transaksi yang dilakukan akan lebih efisien. Perbedaan yang terdapat dari tinjauan literatur dengan penelitian penulis yaitu pada metode pengembangan sistem yang dipakai, pada penelitian yang akan diteliti penulis menggunakan metode Extreme Programming (XP), karena fleksibilitas yang tinggi terhadap *user* dan klien dalam pengembangannya. sehingga mampu membuat aplikasi yang sesuai dengan fungsi yang dibutuhkan klien (Widodo 2008). Kemudian terdapat juga perbedaan dalam penggunaan teknologi yang digunakan penulis dalam mengembangkan aplikasi berbasis android, dalam penelitian ini penulis menggunakan *Framework* React Native. Pada aplikasi yang akan peneliti kembangkan juga memanfaatkan fitur *QR-code scanner* yang berguna untuk menginputkan data pelanggan yang membuat

pegawai tidak harus menginputkan data pelanggan setiap transaksinya dan fitur lainnya yang akan ada pada aplikasi yaitu dapat menampilkan *history* transaksi secara *realtime* supaya pimpinan dapat mudah mengetahui dan memantau transaksi yang terjadi kapan saja dan dimana saja. Kemudian dalam pengujian sistem penulis menggunakan pengujian ISO 25010.

## 2.2 Pengertian Android

Android adalah sebuah *software* yang digunakan pada perangkat *mobile* yang mencakup sistem operasi, *middleware*, dan aplikasi kunci yang dirilis oleh Google. Hal tersebut membuat android dapat mencakup keseluruhan dari suatu aplikasi, mulai dari pengembangan aplikasi sampai sistem operasi itu sendiri. Pengembangan aplikasi yang diterapkan pada android, menggunakan dasar bahasa pemrograman Java. Tapi secara sempit, Android biasanya mengacu pada sistem operasinya saja. Sistem operasi ini bersifat open source atau terbuka, sehingga pengembang bebas mengembangkan atau mengembangkan aplikasi dengan biaya yang sedikit, dan pengembang dapat menjual aplikasi yang diciptakan tanpa ada lisensi ke produsen atau vendor tertentu. Kemudian pengembang diperbolehkan memodifikasi atau mengubah sistem operasi android. Dari beberapa faktor tersebut lah yang membuat android menjadi salah satu sistem operasi yang populer (Tim, 2015).



**Gambar 2.1** Logo android

Sumber: (Dieter, 2019)

## 2.3 Framework

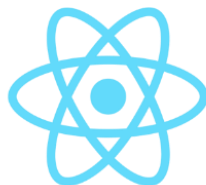
*Framework* merupakan sebuah kerangka kerja yang bertujuan untuk memudahkan programmer dalam pengembangan aplikasi dengan memanfaatkan library yang sudah terorganisir untuk dapat membuat suatu program lebih cepat

(Solikin 2014). Dalam Bahasa Indonesia yaitu kerangka kerja dan dapat diartikan sebagai library (*class*) yang bisa diturunkan, atau dapat digunakan fungsinya oleh modul modul atau fungsi yang akan dikembangkan. (Cahyati and Murti 2018).

Berdasarkan definisi-definisi di atas dapat diambil kesimpulan bahwa framework adalah sistem yang terstruktur yang digunakan sebagai kerangka dalam mengembangkan sesuatu yang bertujuan untuk menyelesaikan suatu permasalahan atau isu – isu yang cukup kompleks lebih cepat.

## 2.4 React Native

React Native adalah salah satu *framework* yang digunakan untuk mengembangkan aplikasi *mobile* dan menggunakan Bahasa pemrograman JavaScript. Dengan menggunakan *framework* React Native, pengembang dapat merender *user interface* yang di peruntukan untuk digunakan pada platform iOS maupun Android. React Native juga merupakan *framework* yang bersifat *open source*, dan dalam penggunaannya dapat berjalan pada sistem operasi seperti Windows maupun macOS (Eisenman B. , 2016). dalam pengembangannya juga, 75% kode program dapat digunakan jika ingin mengembangkan aplikasi ke *platform* lain tanpa harus menulis ulang kode (Hansson and Vidhall 2016).



## React Native

**Gambar 2.2** Logo React Native

Sumber: (John3, 2019)

## 2.5 QR-code

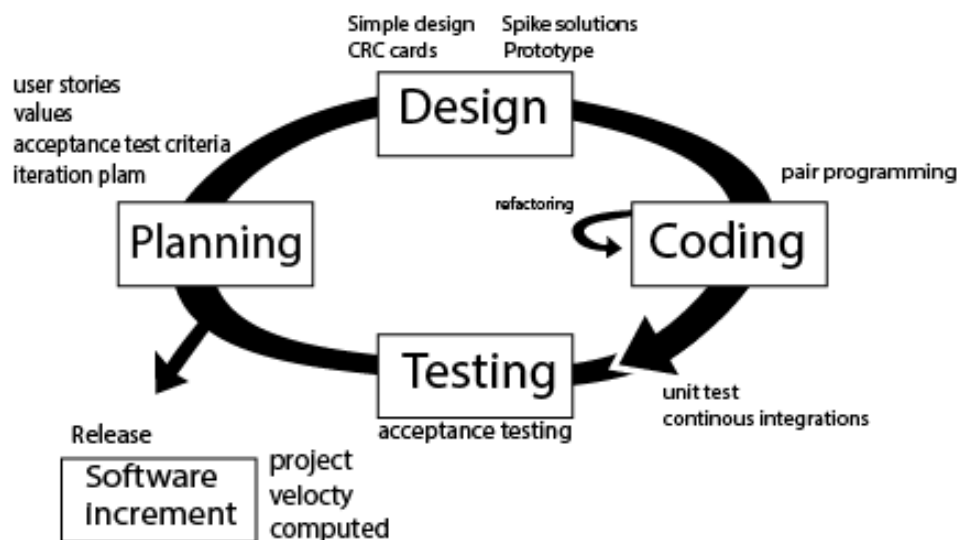
QR *code* adalah jenis kode matriks atau kode batang dua dimensi dan dikembangkan oleh Denso Wave, salah satu divisi Denso Corporation yang merupakan perusahaan dari Jepang dan dipublikasikan pada tahun 1994. Sesuai dengan namanya, *quick response* atau respon cepat fungsi utama *QR-code* dapat dengan mudah dibaca oleh pemindai *QR-code*. tujuan adalah untuk diciptakannya



*QR-code* yaitu untuk menyampaikan informasi dengan cepat dan mendapatkan respons yang cepat. Berbeda dengan kode batang, yang hanya menyimpan informasi secara *horizontal QR-code* dapat menampung informasi yang lebih banyak dari pada kode batang. (Ismail dkk., 2021).

## 2.6 Metode Pengembangan Sistem

*Extreme Programming* (XP) merupakan metodologi yang digunakan untuk pengembangan perangkat lunak yang ditujukan dalam meningkatkan kualitas perangkat lunak terhadap perubahan serta kebutuhan pelanggan. Jenis pengembangan perangkat lunak semacam ini dimaksudkan untuk meningkatkan produktivitas dan memperkenalkan pro pemeriksaan dimana persyaratan pelanggan baru dapat diadopsi. Ada beberapa tahapan yang ada pada *Extreme Programming* yaitu terdiri dari Perencanaan (*Planning*) seperti memahami kriteria pengguna dan perencanaan pengembangan, designing seperti perancangan *prototype* dan tampilan, pengkodean juga termasuk dalam pengintegrasian, terakhir adalah testing (Ariyanti dkk., 2020). *Extreme Programming* adalah metode pengembangan perangkat lunak untuk menyederhanakan proses pengembangan sehingga lebih fleksibel, adaptif, dan dikerjakan oleh satu atau dua orang pengembang (Fatoni and Dwi 2016). Ada empat tahapan dalam pengembangan perangkat lunak menggunakan metode extreme programming (Pressman, 2005).



**Gambar 2.3** Metode Extreme Programming

Sumber: (Pressman, 2005)

### 1. *Planning* (Perencanaan)

Pada tahapan ini merupakan tahapan yang diperlukan sebelum pengembang membuat sistem, tahapan ini penting karena ketika membuat sebuah sistem harus direncanakan atau dianalisis kebutuhan-kebutuhan yang diperlukan *user*. Dalam tahapan *planning* ini terdapat beberapa bagian yaitu:

- a. *User Stories* yaitu Pengguna menggambarkan atau menceritakan permasalahan dan kebutuhan *user* yang diperlukan untuk sistem yang akan dibangun.
- b. *Values* merupakan poin atau nilai-nilai yang dapat diambil dari *user stories*.
- c. *Acceptable test criteria* yaitu menentukan kriteria tes sebagai acuan terhadap kebutuhan sistem yang akan dibangun.
- d. *Iteration plan* merupakan rencana untuk menentukan berapa kali peneliti melakukan pertemuan terhadap pengguna.

### 2. *Design* (Perancangan)

Setelah pada tahapan perencanaan selesai, maka tahapan selanjutnya adalah perancangan. Pada tahapan ini pengembang melakukan perancangan dengan membuat sebuah pemodelan, yang dimulai dari pemodelan sistem, kemudian pemodelan arsitektur, dan yang terakhir adalah pemodelan basis data. Dalam tahapan ini terdapat beberapa bagian yaitu:

- a. *Simple design*: Pengembang mengembangkan perangkat lunak dengan desain yang sederhana.
- b. *Spike solution* Jika dalam praktiknya desain yang dibuat sangatlah sulit. Extreme programming akan menggunakan spike solution dimana pembuatan design akan dibuat langsung ke tujuannya.
- c. *CRC card* Digunakan untuk mengidentifikasi dan mengorganisasikan object-oriented *classes*.
- d. *Prototype*: Merupakan perancangan *user interface* biasanya dalam bentuk *wireframing* untuk mempermudah pengembang dan klien dalam melihat gambaran sistem

### 3. *Coding* (Pengkodean)

Tahapan ini merupakan tahapan untuk menerapkan pemodelan yang sudah dirancang di tahapan perancangan yang sudah dibuat kedalam bentuk *user interface* dan menggunakan bahasa pemrograman.

- a. *Pair programming*: Dalam proses pengembangan terdapat dua orang *programmer*, dimana seorang *programmer* membuat *coding* dan *programmer* lainnya mengoreksi code yang dibuat.
  - b. *Refactory*: Merupakan tahapan yang dilakukan ketika terjadi ketidaksesuaian kode program kemudian dilakukan perbaikan untuk mendapatkan hasil yang diinginkan.
4. *Testing* (Pengujian)

Pada tahapan ini pengembang sistem melakukan pengujian terhadap sistem yang sudah dibuat untuk mengetahui kesalahan yang terdapat pada sistem dan untuk mengecek sistem telah dibuat sudah sesuai dengan kebutuhan pengguna atau belum.

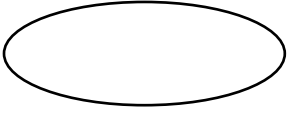



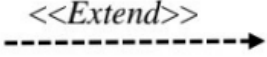
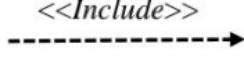
## **2.7 Unified Modeling Language (UML)**

UML (*Unified Modelling Language*) adalah sebuah Bahasa berbentuk grafik atau gambar yang digunakan untuk memvisualisasikan sebuah sistem yang akan di kembangkan berbasis *Object-Oriented*. UML juga memiliki standar penulisan sebuah sistem blue print, meliputi kelas dalam Bahasa pemrograman yang spesifik, proses bisnis, skema *database* dan komponen lainnya yang diperlukan sistem (Mubarak dkk., 2019). Terdapat tiga diagram UML yang memiliki fungsi masing – masing yaitu:

### **2.7.1 Use Case**

*Use Case* adalah teknik untuk menggambarkan kebutuhan-kebutuhan fungsional dari sebuah sistem yang ingin di kembangkan atau sistem baru yang akan dibuat. Setiap *use case* memiliki satu atau lebih skenario yang menjelaskan bagaimana sebuah sistem akan berinteraksi dengan pengguna atau sistem lainnya untuk tercapainya suatu sasaran bisnis tertentu. (Artina, 2006). Simbol – simbol yang terdapat pada *use case* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol-Simbol *Use Case*


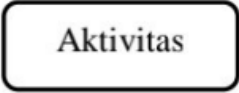
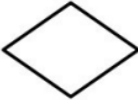


Simbol	Keterangan
	<i>Use Case</i> : Fungsionalitas yang disediakan sistem, Biasanya diawali dengan kata kerja sebelum frase nama <i>use case</i> .
	Aktor: pengguna, proses atau sistem lain yang berinteraksi dengan sistem yang akan di kembangkan, walau simbol berbentuk orang aktor belum tentu orang, nama aktor biasanya menggunakan kata benda
	Asosiasi: sebuah komunikasi antar aktor dan <i>use case</i> digunakan saat <i>use case</i> memiliki interaksi dengan aktor
	Generalisasi: hubungan umum sampai khusus antara dua buah <i>use case</i> di mana salah satu <i>use</i> memiliki fungsi yang lebih umum.
	<i>Extend</i> : menunjukan <i>use case</i> tambahan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
	<i>Include</i> : Menunjukan bahwa <i>use case</i> tambahan akan dipanggil saat <i>use case</i> tambahan dijalankan

### 2.7.2 Activity Diagram

*Activity diagram* merupakan diagram yang berfungsi untuk menjelaskan secara prosedural alur proses sebuah sistem. Pada diagram ini memungkinkan mengevaluasi adanya lebih dari satu jalur yang terbentuk dan berjalan beriringan. pembuatan *activity diagram* dimulai dari *initial node* dan diakhiri dengan *end node*. *initial node* dalam sebuah *activity diagram* diperbolehkan lebih dari satu, hal ini

berguna untuk mengakomodasi jika sistem yang dikembangkan akan memiliki input lebih dari satu (Ayu, 2017). adapun simbol-simbol yang terdapat pada *activity diagram* dapat dilihat pada Tabel 2.3.

**Tabel 2.3** Simbol – Simbol *Activity Diagram*

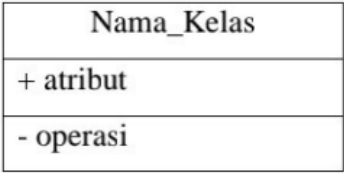




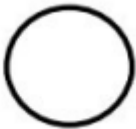

Simbol	Keterangan
	Status awal: simbol yang memiliki fungsi menunjukkan status awal dari sebuah diagram
	Aktivitas: simbol yang memiliki fungsi menunjukkan aktivitas yang dilakukan sistem, biasanya diawali dengan kata kerja.
	<i>Decision</i> : asosiasi percabangan, dimana jika ada pilihan aktivitas lebih dari satu
	<i>Joint</i> : asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Status akhir: simbol yang menunjukkan aktivitas akhir diagram

### 2.7.3 *Class diagram*

*Class diagram* adalah sebuah diagram UML yang menggambarkan hubungan antar kelas dan memiliki penjelasan detail dari sebuah sistem di dalam model desain, dan juga memperhatikan entitas perilaku sistem. Pada *class diagram* memiliki atribut-atribut dan operasi yang dimiliki dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan. Komponen-komponen yang terdapat pada *class diagram* yaitu *class*, relasi, asosiasi, generalisasi dan agregasi, atribut, operasi, dan visibilitas, tingkat akses objek eksternal pada suatu operasi. Juga hubungan antar kelas disebut multiplicity dan *cardinality* (Hendini 2016).

Adapun simbol-simbol yang terdapat pada *class diagram* dapat dilihat pada Tabel 2.4.

**Tabel 2.4** Simbol – Simbol *Class Diagram*

Simbol	Keterangan
	Menggambarkan kelas yang terdapat pada struktur
	Asosiasi berarah: relasi antar kelas yang memiliki arti kelas satu digunakan oleh kelas yang lain dan biasanya disertai dengan <i>multiplicity</i> .
	Generalisasi: relasi antar kelas yang memiliki makna umum khusus
	Ketergantungan: <i>dependency</i> merupakan relasi antar kelas dengan arti ketergantungan antar kelas.
	Agregasi: merupakan relasi antar kelas dengan arti semua bagian
	Antar muka: yaitu menunjukkan interface hal ini sama dengan konsep interface pada pemrograman berorientasi objek.
	Asosiasi: merupakan relasi antar kelas dengan makna umum dan biasanya disertai <i>multiplicity</i> .

## 2.8 Class Responsibility Collaborator Card (CRC Card).

Menurut Zulhalim (2018), *Class Responsibility Collaborator Card (CRC Card)* adalah suatu teknik yang bertujuan untuk mengecek adanya interaksi suatu objek, mengenali, dan menentukan *class* yang diperlukan. Pada *CRC Card* terdapat *Class Name* yaitu sebagai suatu kelas yang dijabarkan, *Responsibilities* yang

berfungsi sebagai tugas dan fungsi dari kelas tersebut, dan *Collaborators* berfungsi sebagai objek atau kelas yang berkaitan dengan kelas tersebut. Berikut ini merupakan gambar bagian-bagian *CRC Card*.

**Tabel 2.5** *Class Responsibility Collaborator Card (CRC Card)*

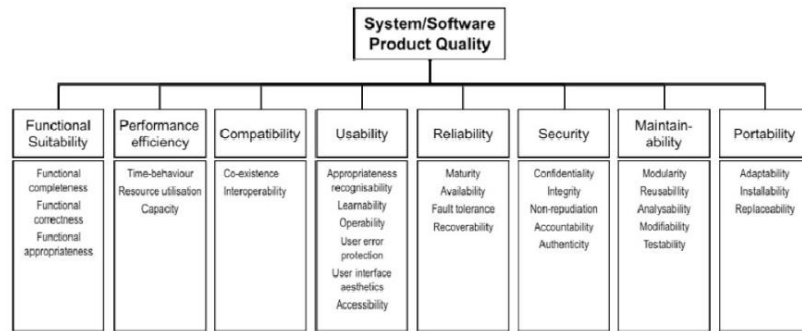
<i>Class Name</i>	
<i>Responsibilities</i>	<i>Collaborators</i>

## 2.9 Firebase

Firebase adalah penyedia layanan *cloud* dengan *back-end* sebagai servis yang berbasis di San Fransisco, California. Firebase membuat sejumlah produk untuk pengembangan aplikasi *mobile* ataupun web. Firebase didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011 dan diluncurkan dengan *cloud database* secara *realtime* di tahun 2012 (Kumala & Winard, 2020).

## 2.10 Pengujian ISO 25010

Menurut (Fadli H. Wattiheluw, 2019) Pengujian ISO 25010 merupakan bagian dari *Systems and Software Quality Requirements and Evaluation (SQuaRE)* yang merupakan versi lanjutan dari ISO 91261, yang telah direvisi secara teknis dengan menambahkan beberapa struktur dan bagian dari standar model kualitas. Tujuan dari penggunaan kualitas ini adalah untuk mengukur sejauh mana produk atau sistem tersebut bisa digunakan oleh pengguna untuk memenuhi kebutuhan dalam mencapai tujuan yang diinginkan dengan efisiensi, efektivitas, kepuasan dalam konteks penggunaan yang spesifik, dan bebas dari resiko. Menurut (Harun, 2018) ISO 25010 terdiri dari delapan karakteristik yang dibagi menjadi beberapa bagian yang berhubungan dengan sifat-sifat statis perangkat lunak dan sifat dinamis dari sistem komputer, yang dapat dilihat pada Gambar 2.4



**Gambar 2.4** Model ISO 25010

Sumber: (Harun, 2018)

Berdasarkan gambar diatas, dapat dijelaskan mengenai delapan karakteristik tersebut, sebagai berikut:

1. *Functional Suitability*, produk yang memberikan fungsional untuk memenuhi kebutuhan saat sistem atau produk tersebut digunakan pada keadaan tertentu.
2. *Reliability*, sistem dapat mempertahankan kinerjanya pada level tertentu ketika digunakan pada keadaan tertentu.
3. *Performance Efficiency*, sistem menyediakan performa yang baik dengan sejumlah *resource* yang akan digunakan pada sistem atau produk.
4. *Usability*, sistem atau produk mudah dimengerti, mudah dipakai, dan menarik untuk digunakan.
5. *Security*, sistem menyediakan layanan untuk melindungi akses, ataupun pengungkapan yang berbahaya.
6. *Compatibility*, merupakan kemampuan pada suatu komponen atau sistem untuk bertukar informasi.
7. *Maintainability*, merupakan tingkat suatu sistem dapat dimodifikasi, perbaikan, pengembangan untuk menyesuaikan dengan lingkungan, modifikasi pada kriteria, dan spesifikasi fungsi.
8. *Portability*, sistem dapat dipindahkan dari satu ruang ke ruang lainnya.

Berdasarkan dari kebutuhan dalam penelitian ini, dalam pengujian aplikasi *mobile* menggunakan empat karakteristik yang digunakan yaitu *functional suitability*, *compatibility*, *usability*, dan *performance efficiency*. (David, 2011)



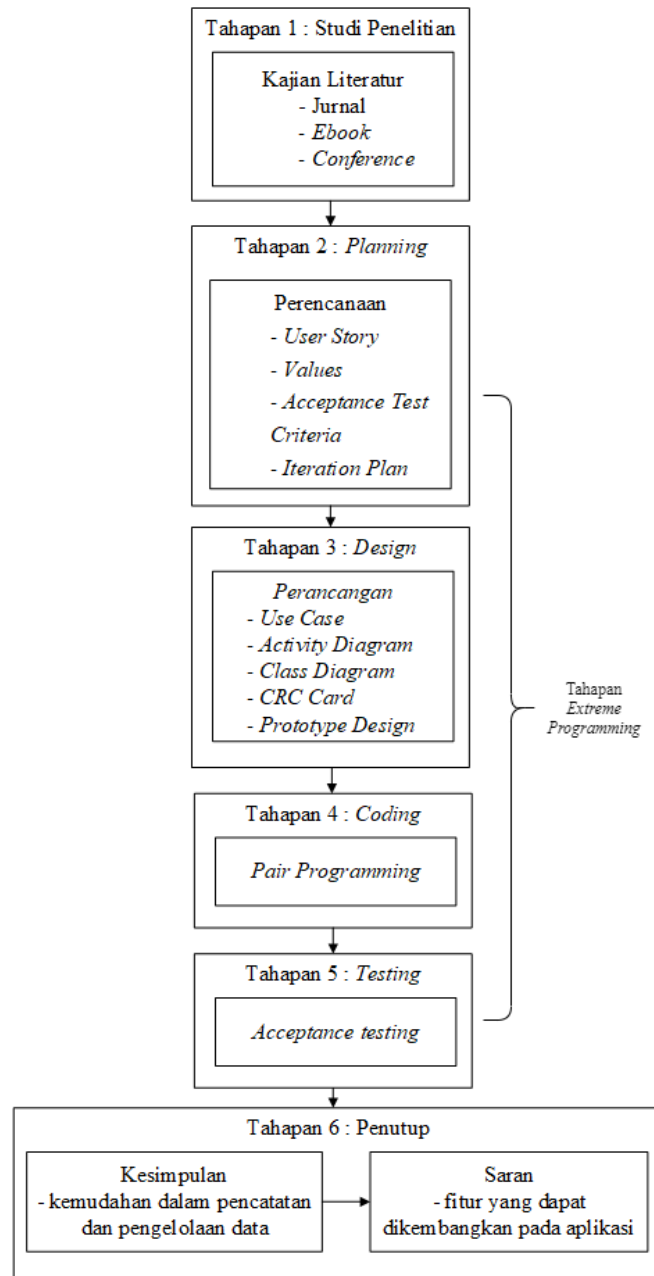
### 2.11 Pengertian Figma

Figma adalah *software* yang digunakan untuk pembuatan desain berbasis cloud dengan konsep fungsionalitas sketch, dan dapat melakukan kolaborasi antar desainer lainnya. figma dapat bekerja pada berbagai sistem operasi karena dapat berjalan pada browser, fitur slack yang terdapat pada figma sebagai media komunikasi dari satu designer ke designer lainnya dapat memudahkan suatu tim dalam berkomunikasi secara *realtime* dalam melakukan editing, keunggulan lain yaitu pengguna dapat dengan mudah mendapatkan kode dalam bentuk css sehingga memudahkan front end dalam pengimplementasian desain yang ada (Nugraha dkk., 2020)

### 3 METODOLOGI PENELITIAN

#### 3.1 Kerangka Penelitian

Kerangka Penelitian adalah sebuah konsep atau gambaran yang telah dibuat oleh penulis untuk dilakukannya penelitian. Dari uraian yang sudah dijelaskan sebelumnya, maka akan menghasilkan sebuah kerangka penelitian yang dapat dilihat pada gambar 3.1.



**Gambar 3.1** Kerangka Penelitian

Proses pengembangan sistem melalui empat tahapan yaitu tahap perencanaan (*planning*) yaitu bertujuan untuk mengetahui permasalahan dan mengetahui keinginan pengguna, tahapan perancangan (*design*) berupa perancangan *use case*, *activity diagram*, *class diagram*, *CRC Card* dan *interface*, pengkodean (*coding*) yaitu penerapan rancangan dalam bentuk *coding*, dan tahapan pengujian (*testing*) yaitu melakukan pengujian terhadap sistem.

### 3.2 Planning

Perencanaan merupakan tahapan utama yang penulis lakukan untuk mendapatkan kebutuhan pengguna, mengkaji literatur, pembuatan *user stories*, dan melakukan komunikasi secara berkala kepada pihak jasa penyeberangan B.C Bakauheni. Berikut ini merupakan tahapan pada perencanaan:

#### 3.2.1 User Stories

Berikut ini merupakan tabel *user stories* yang telah dirangkum berdasarkan wawancara yang telah dilakukan dengan pimpinan. Tahap ini bertujuan untuk menggambarkan atau menceritakan permasalahan dan kebutuhan *user* yang diperlukan untuk sistem yang akan dibangun. *user stories* dapat dilihat pada Tabel 3.1 Tabel 3.2 dan Tabel 3.3.

**Tabel 3.1** *User Stories*

Kode <i>Stories</i>	<i>User Stories</i>
<i>Story-01</i>	Saya ingin aplikasi dapat mencatat data transaksi dan setiap transaksi data kendaraan otomatis terinput agar memudahkan dalam melakukan transaksi dan mengurangi kesalahan pencatatan.
<i>Story-02</i>	Sebagai pimpinan, saya ingin aplikasi dapat mencatat pembayaran kredit pengendara agar dapat mengetahui <i>history</i> pembayaran kredit pengendara pada sebelumnya.

**Tabel 3.2** *User Stories* (lanjutan 1)

<b>Kode Stories</b>	<b><i>User Stories</i></b>
<i>Story-03</i>	Sebagai pimpinan, saya ingin setiap transaksi dapat terlihat langsung secara <i>realtime</i> agar dapat memantau transaksi yang terjadi walau saya sedang tidak berada di kantor dan terdapat nama dari pegawai yang melakukan transaksi.
<i>Story-04</i>	Sebagai pimpinan, saya ingin aplikasi dapat memisahkan data kendaraan pembayaran langsung dan bulanan agar mudah dalam membuat invoice untuk perusahaan yang bersangkutan.
<i>Story-05</i>	Sebagai pimpinan, saya ingin aplikasi dapat menampilkan nama pegawai yang melakukan transaksi supaya dapat bertanya langsung jika ada kesalahan atau hal yang ingin ditanyakan dengan transaksi tersebut.
<i>Story-06</i>	Sebagai pimpinan, saya ingin aplikasi dapat melakukan pencarian dan mengedit data transaksi dan data pembayaran kredit agar dapat mengubah data jika ada kesalahan input.
<i>Story-07</i>	Sebagai pimpinan, saya ingin aplikasi dapat mengurutkan data sesuai perusahaan untuk bagian kendaraan yang melakukan pembayaran bulanan agar mempermudah dalam pembuatan invoice
<i>Story-08</i>	Sebagai pimpinan, saya ingin aplikasi dapat mengurutkan data sesuai perusahaan untuk bagian kendaraan yang melakukan pembayaran bulanan agar mempermudah dalam pembuatan invoice.
<i>Story-09</i>	Sebagai pimpinan, Saya ingin aplikasi dapat mengubah dan menambahkan data kendaraan dan pegawai, dan juga dapat mengubah password maupun <i>username</i> pegawai. Agar dengan mudah mengubah data jika ada data yang salah ataupun jika ada kendaraan baru atau pegawai baru.

**Tabel 3.3** *User Stories* (lanjutan 2)

<b>Kode Stories</b>	<b><i>User Stories</i></b>
<i>Story-10</i>	Sebagai pimpinan, Saya ingin aplikasi dapat melakukan edit saldo untuk menambahkan atau pun mengurangi saldo, karena terkadang jika saldo kurang akan ditambah dan jika saldo banyak akan dikurangi.
<i>Story-11</i>	Sebagai Pimpinan, saya ingin aplikasi dapat melakukan rekap harian untuk mengetahui uang yang masuk dan keluar dan hasil dari rekap tersebut diambil menjadi uang saldo.
<i>Story-12</i>	Sebagai pimpinan, Saya ingin aplikasi dapat merubah Harga Penyeberangan karena terkadang harga naik dan turun.
<i>Story-13</i>	Sebagai pimpinan, Saya ingin aplikasi dapat mencatat pengeluaran seperti buku/pena dan lainnya biaya tersebut diambil dari saldo dan data masuk kedalam rekap harian
<i>Story-14</i>	Sebagai pimpinan, Saya ingin pegawai tidak dapat mengubah maupun menambahkan data pegawai, data kendaraan, data kreditur, data transaksi, saldo, dan harga penyeberangan, karena takut terjadi kecurangan oleh pegawai.

### 3.3.2 *Values*

Setelah dilakukan tahapan pembuatan *user stories* tahapan selanjutnya yang dilakukan yaitu mengambil nilai atau *value*. berdasarkan cerita pengguna dapat disimpulkan bahwa yang dibutuhkan pimpinan adalah sebuah aplikasi yang dapat memudahkan jasa dalam melakukan pencatatan transaksi debit dan kredit, pencarian data transaksi maupun data kendaraan, melakukan filter pada data, pembuatan rekap harian, dan menampilkan data transaksi secara realtime agar pimpinan dapat memonitoring transaksi kapanpun.

### 3.3.3 *Acceptance Test Criteria*

Pada tahapan ini bertujuan untuk membuat suatu daftar kriteria yang diambil dari *user stories*, yang bertujuan untuk menentukan batasan-batasan *user stories*, pada tahap ini juga bertujuan untuk menentukan aplikasi yang dibuat sudah berjalan sesuai kebutuhan dan keinginan pengguna. Berikut kriteria yang digunakan berdasarkan empat aspek pengujian ISO 25010 adalah sebagai berikut:

#### 1. *Functional Suitability*

Aspek pertama yang akan dilakukan yaitu pengujian pada bagian fungsional, memiliki tujuan untuk melakukan pengecekan terhadap fungsi-fungsi yang ada. Berikut ini merupakan karakteristik pengujian yang dibagi menjadi beberapa karakteristik yaitu:

- a. *Functional completeness*, sejauh mana fungsi yang telah disediakan mencakup seluruh tugas dan tujuan pengguna secara spesifik.
- b. *Functional correctness*, sejauh mana produk atau sistem menyediakan hasil yang benar sesuai dengan kebutuhan pengguna.
- c. *Functional appropriateness*, sejauh mana fungsi yang telah disediakan mampu memfasilitasi penyelesaian tugas atau suatu tujuan tertentu.

#### 2. *Usability*

Tes kriteria yang akan dilakukan selanjutnya yaitu *usability* yang bertujuan untuk mengetahui aplikasi yang dibuat mudah dimengerti, mudah dipakai dan menarik. Karakteristik dalam pengujian *usability* ini dibagi menjadi beberapa karakteristik yaitu:

- a. *Appropriateness recognisability*, sejauh mana pengguna mengetahui apakah sistem atau produk sesuai kebutuhan mereka.
- b. *Operability*, sejauh mana produk atau sistem mudah dioperasikan dan dikendalikan.
- c. *Learnability*, sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai suatu tujuan tertentu, belajar menggunakan sistem atau produk dengan efisien, efektif dan kebebasan dari resiko serta dalam hal kepuasan pada konteks tertentu.

- d. *User interface aesthetics*, sejauh mana tampilan antarmuka pengguna dari sistem memungkinkan interaksi yang menyenangkan dan memuaskan pengguna.
- e. *User error protection*, sejauh mana produk atau sistem melindungi pengguna terhadap suatu kesalahan.
- f. *Accessibility*, sejauh mana produk atau sistem dapat digunakan oleh semua kalangan untuk mencapai suatu tujuan tertentu sesuai dengan konteks

### 3. *Compatibility*

Pada tahap ini menjalankan fungsi lain yang diperlukan secara bersamaan ketika berbagi perangkat keras dan environment perangkat lunak yang sama. Aspek yang diuji pada tahap *compatibility* meliputi *Co-existence*, Operating System, Device (Jayanto & Jati, 2017). Berikut tahapan yang dilakukan pada uji *compatibility*:

#### a. *Co-existence*

Pada tahap ini aplikasi akan diuji apakah dapat berjalan berdampingan dengan aplikasi lainnya dengan menggunakan *smartphone* Pocophone f1. Kerangka pengujian *Co-existence* dapat dilihat pada Tabel 3.7.

**Tabel 3.4** Kerangka Pengujian *Co-existence*

NO	Aplikasi yang dijalankan		Berhasil	Gagal
1	Aplikasi pendataan BC	Play Store		
2	Aplikasi pendataan BC	Google Chrome		
3	Aplikasi pendataan BC	Whatsapp		
4	Aplikasi pendataan BC	Facebook		
5	Aplikasi pendataan BC	Maps		
6	Aplikasi pendataan BC	Apex Legends		
<b>Total</b>				

#### b. *Operating System dan device*

Pada tahap ini akan dilakukan pengujian di berbagai sistem operasi android dan beberapa perangkat, pengujian dibagi menjadi tiga bagian yaitu *setup suite* mewakili instalasi aplikasi di perangkat, *built-in explorer* yaitu menjalankan

aplikasi pada perangkat pengujian, dan terakhir *teardown suite* yaitu melakukan penghapusan pada perangkat. Perangkat - perangkat yang digunakan untuk pengujian dapat dilihat pada Tabel 3.8.

**Tabel 3.5** Kerangka Pengujian *Operating System Dan Device*

<i>No</i>	<i>Perangkat</i>	<i>Setup Suite</i>	<i>Built-in Explorer Suite</i>	<i>Teardown Suite</i>	<i>Jumlah</i>
1	Xiaomi Pochophone F1 (android 10)				
2	Xiaomi Pocophone X3 nft (android 11)				
3	Oppo f1s (android 5.1)				
4	Redmi 4x (android 7)				
Total					

#### 4. *Performance Efficiency*

Pengujian pada tahap ini berguna untuk mengetahui sejauh mana kinerja *relative* aplikasi dengan perangkat yang digunakan pada kondisi tertentu. Pengujian dilakukan menggunakan tools *Firebase test lab* secara *automation* menggunakan lima perangkat dan hasil pengujian dibagi menjadi tiga bagian. Aspek yang akan di uji pada tahap *performance Efficiency* sebagai berikut:

- Time Behaviour* pada pengujian time behaviour nilai didapatkan berdasarkan persentase thread yang di dapat dari pengujian pada firebase test lab.
- CPU Utilization* pada tahap ini melakukan pengujian terhadap CPU ketika menjalankan aplikasi dengan rentang waktu yang ditentukan dan akan didapatkan persentase dari penggunaan CPU
- Memory Utilization* pada pengujian ini melakukan pengujian pada *memory* Ketika menjalankan aplikasi dan akan didapatkan nilai rata-rata *memory* yang



digunakan. Setelah pembuatan kriteria penerimaan tahapan selanjutnya dalam penelitian ini yaitu melakukan pembuatan rencana iterasi (*iteration plan*).

### 3.3.4 *Iteration Plan*

*Iteration Plan* merupakan proses komunikasi dan perencanaan fase iterasi pengembangan sistem yang akan di bangun, yang bertujuan untuk menentukan berapa lama pembuatan aplikasi yang akan dibangun, berdasarkan prioritas dan *value* pada *user stories*. rencana iterasi yang telah dibuat dapat dilihat pada Tabel 3.9.

#### 3.3.4.2. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk memudahkan penelitian ini dalam pembuatan rancangan sistem. analisis dibagi menjadi dua kategori yaitu kebutuhan fungsional dan kebutuhan non fungsional, hasil analisa yang didapatkan dapat dilihat sebagai berikut:

##### 1. Kebutuhan Fungsional

Berikut ini merupakan kebutuhan fungsional berupa fungsi atau proses yang nantinya akan dibutuhkan sistem sesuai kebutuhan pengguna:

##### a. *User* Pimpinan

*User* pimpinan adalah aktor yang dapat melihat semua informasi yang ada pada aplikasi. Kegiatan yang dapat dilakukan oleh *user* pimpinan adalah sebagai berikut:

1. Mengelola data transaksi
2. Menampilkan data transaksi
3. Memfilter data transaksi
4. Mencari data transaksi
5. Menampilkan data kendaraan dan pegawai
6. Mengelola data kendaraan dan pegawai
7. Melakukan transaksi penyeberangan dan kreditur
8. Menampilkan rekap harian
9. Menampilkan dan mengelola saldo
10. Menampilkan dan mengelola harga pelabuhan
11. Menambahkan pengeluaran

##### b. *User* Pegawai

*User* pegawai adalah aktor yang dapat semua informasi yang ada pada aplikasi tanpa bisa melakukan perubahan data. Kegiatan yang dapat dilakukan oleh *user* pimpinan adalah sebagai berikut:

1. Menampilkan data transaksi
2. Memfilter data transaksi
3. Mencari data transaksi
4. Menampilkan data kendaraan dan pegawai
5. Melakukan transaksi penyeberangan dan kreditur
6. Menampilkan rekap harian
7. Menampilkan saldo
8. Menampilkan harga pelabuhan
9. Menambah pengeluaran

## 2. Kebutuhan Non Fungsional

Berikut ini merupakan kebutuhan non fungsional atau kebutuhan berupa perangkat-perangkat untuk mendukung sistem yang akan dikembangkan:

- a. Aplikasi dapat dijalankan di beberapa versi android diantaranya android 10, 11, 5 dan 7.
- b. Aplikasi memiliki tampilan *user* interface yang friendly serta responsif.
- c. Aplikasi dapat menyimpan data-data kedalam *database*.

## 3. Spesifikasi Perangkat

Berikut merupakan kebutuhan berupa perangkat keras dan perangkat lunak yang digunakan dalam penelitian:

- a. Perangkat keras (*Hardware*)

Perangkat keras yang digunakan pengembang dalam penelitian ini yaitu sebuah komputer dengan spesifikasi sebagai berikut:

1. *Processor* Ryzen 5 3500X 6-Core ~3.6GHz
2. *Memory* RAM 16 GB
3. *Memory internal* 1.25 TB
4. *VGA card* Radeon RX5500XT

- b. Perangkat lunak (*Software*)

Perangkat lunak yang digunakan pengembang dalam penelitian ini yaitu sebagai berikut:

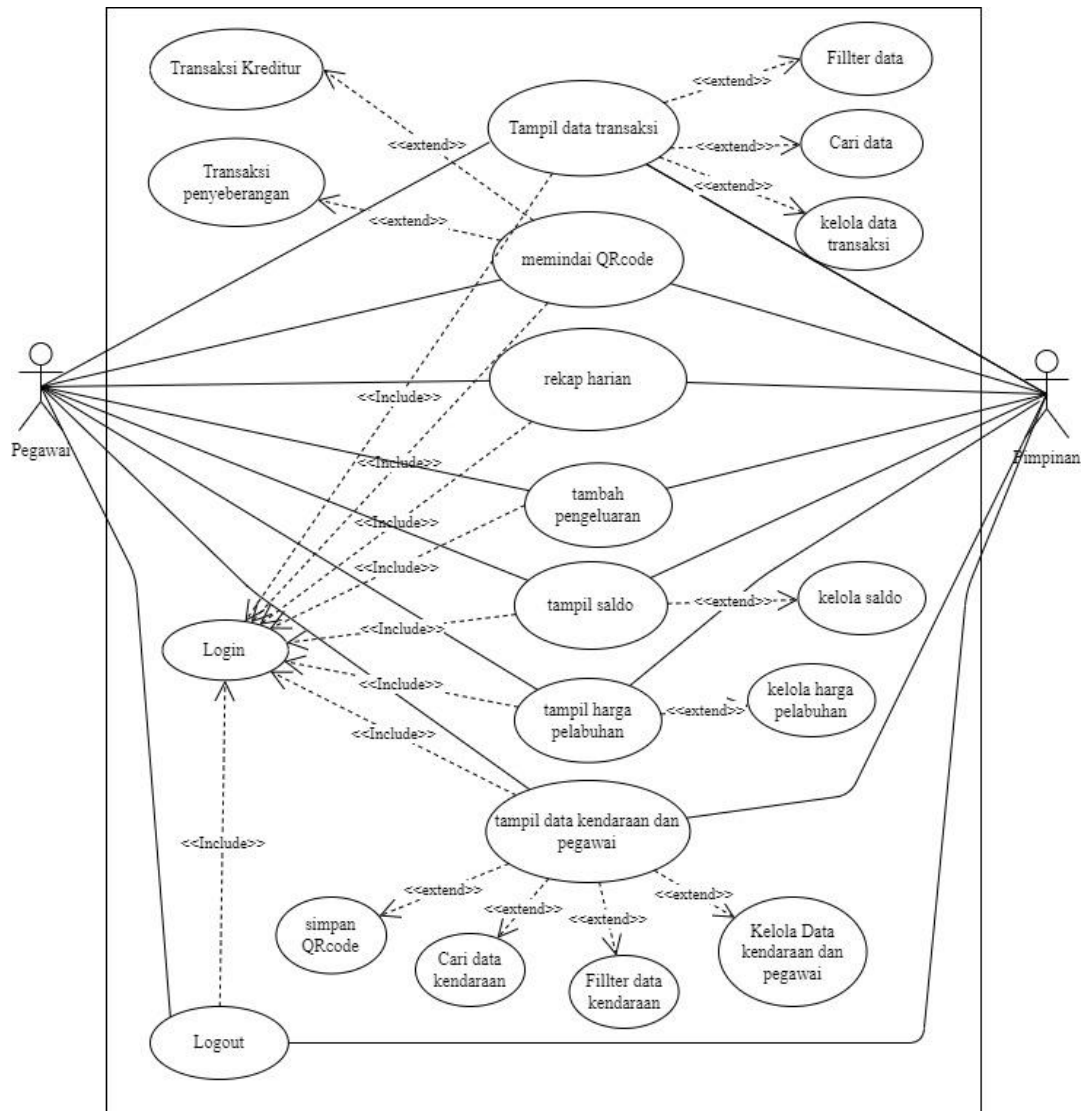
1. Sistem operasi windows 10 (64-bit)
2. Sistem operasi android
3. Visual Code
4. Figma

### **3.4. Design**

Perancangan ini bertujuan untuk merancang atau mendesain sistem dalam rangka memenuhi kebutuhan pengguna sistem dan untuk memberikan gambaran yang jelas. Pada tahap perancangan sistem ini terdiri dari *use case*, *activity diagram*, CRC dan *prototype design*.

#### **3.4.1 Use Case Diagram**

Berikut ini merupakan sebuah proses kegiatan dan proses bisnis yang dilakukan oleh aktor yang berinteraksi dengan *use case*. Pada penelitian ini penulis menggunakan dua aktor yaitu pegawai dan pengurus. *Use case* dapat dilihat pada Gambar 3.2.



**Gambar 3.2** Use Case Diagram Aplikasi Pendataan

Penjelasan dari proses kegiatan pada *use case* yang telah dibuat dapat dilihat pada Tabel 3.11.

**Tabel 3.6** Penjelasan *Use Case*

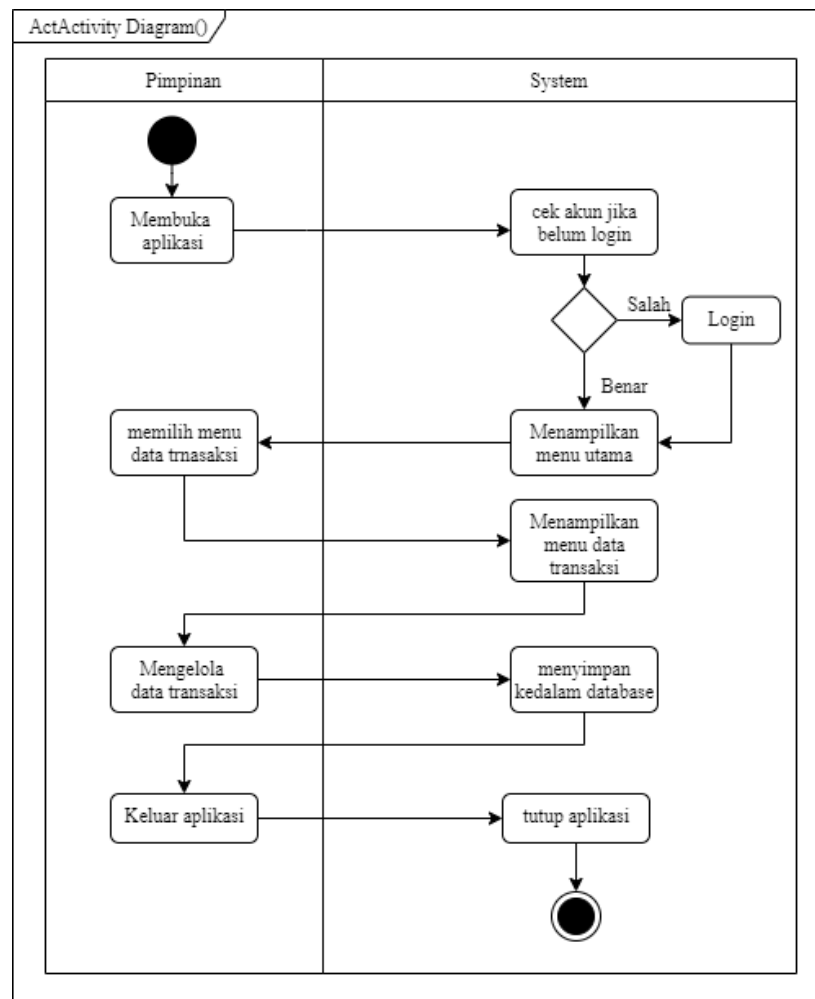
<b>Aktor</b>	<b>Kegiatan</b>
Pimpinan	Kegiatan yang dilakukan pimpinan yaitu login terlebih dahulu untuk masuk ke halaman utama aplikasi, kemudian pimpinan dapat Mengelola data transaksi, menampilkan data transaksi, memfilter data transaksi, mencari data transaksi, menampilkan data kendaraan dan pegawai, mengelola data kendaraan dan pegawai, melakukan transaksi penyeberangan dan kreditur, menampilkan rekap harian, menampilkan dan mengelola saldo, menampilkan dan mengelola harga pelabuhan, menambahkan pengeluaran
Pegawai	Kegiatan yang di lakukan pegawai yaitu login terlebih dahulu untuk masuk ke halaman utama aplikasi, kemudian pegawai dapat menampilkan data transaksi, memfilter data transaksi, mencari data transaksi, menampilkan data kendaraan dan pegawai, melakukan transaksi penyeberangan dan kreditur, menampilkan rekap harian, menampilkan saldo, menampilkan harga pelabuhan, menambah pengeluaran

### 3.4.2 Activity Diagram

*Activity diagram* pada rancangan aplikasi pendataan transaksi dapat dilihat pada gambar dibawah ini:

1. *Activity diagram* mengelola data transaksi

Hasil pembuatan *activity diagram* mengelola data transaksi yang dilakukan oleh pimpinan dapat dilihat pada Gambar 3.3.



**Gambar 3.3** Activity Diagram Mengelola Data Transaksi

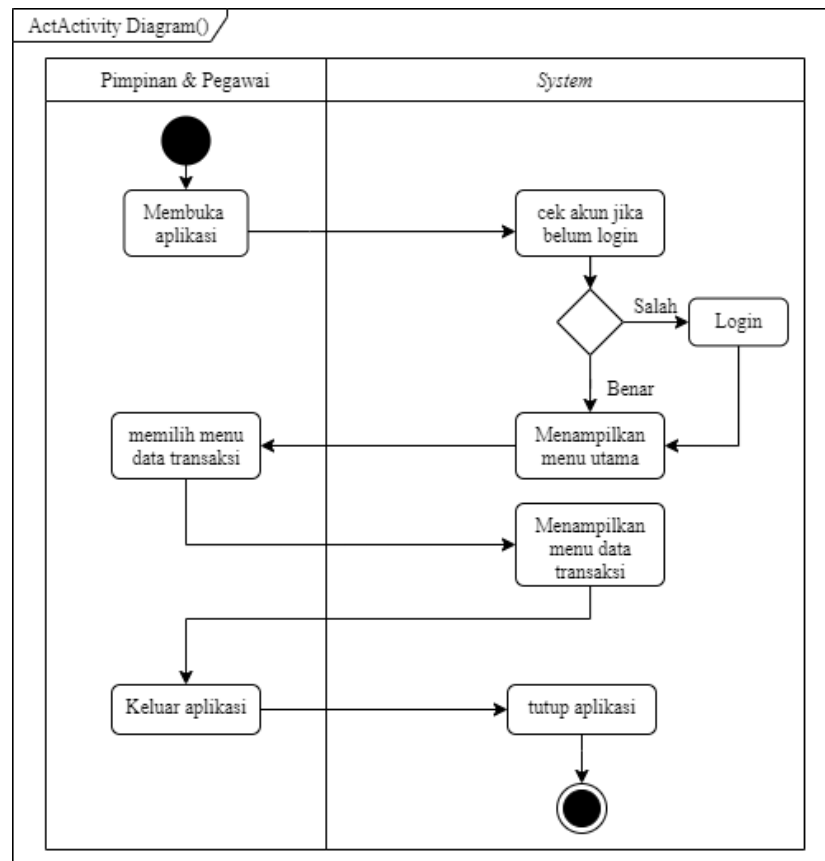
Penjelasan:

Berdasarkan activity diagram mengelola data transaksi yang ditunjukkan pada gambar 3.3 tahapan pertama untuk menampilkan halaman mengelola data transaksi yaitu sistem melakukan autentikasi akun dan kemudian menampilkan halaman utama, selanjutnya pimpinan dapat langsung memilih menu data transaksi dan dapat

langsung mengelola data yang ada, setelah proses selesai sistem akan langsung menyimpan data ke dalam *database*.

## 2. *Activity diagram* menampilkan data transaksi

Hasil pembuatan *activity diagram* menampilkan data transaksi yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.4.



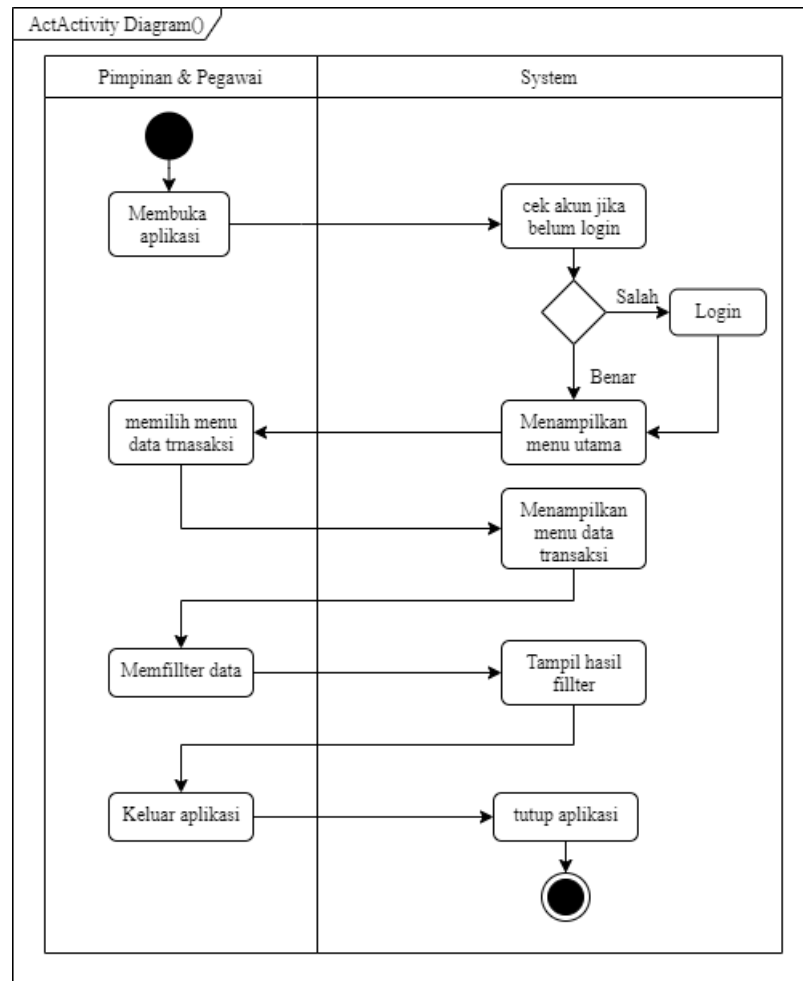
**Gambar 3.4** *Activity Diagram* Menampilkan Data Transaksi

Penjelasan:

Dapat dilihat pada gambar 3.4 tahapan untuk menampilkan menu data transaksi yaitu diawali dengan sistem melakukan autentikasi akun dan dilanjutkan dengan menampilkan halaman utama, kemudian pimpinan dapat memilih menu data transaksi dan sistem akan langsung menampilkan menu yang dipilih.

## 3. *Activity Diagram* memfilter data transaksi

Hasil pembuatan *activity diagram* memfilter data transaksi yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.5.



**Gambar 3.5** Activity Diagram Memfilter Data Transaksi

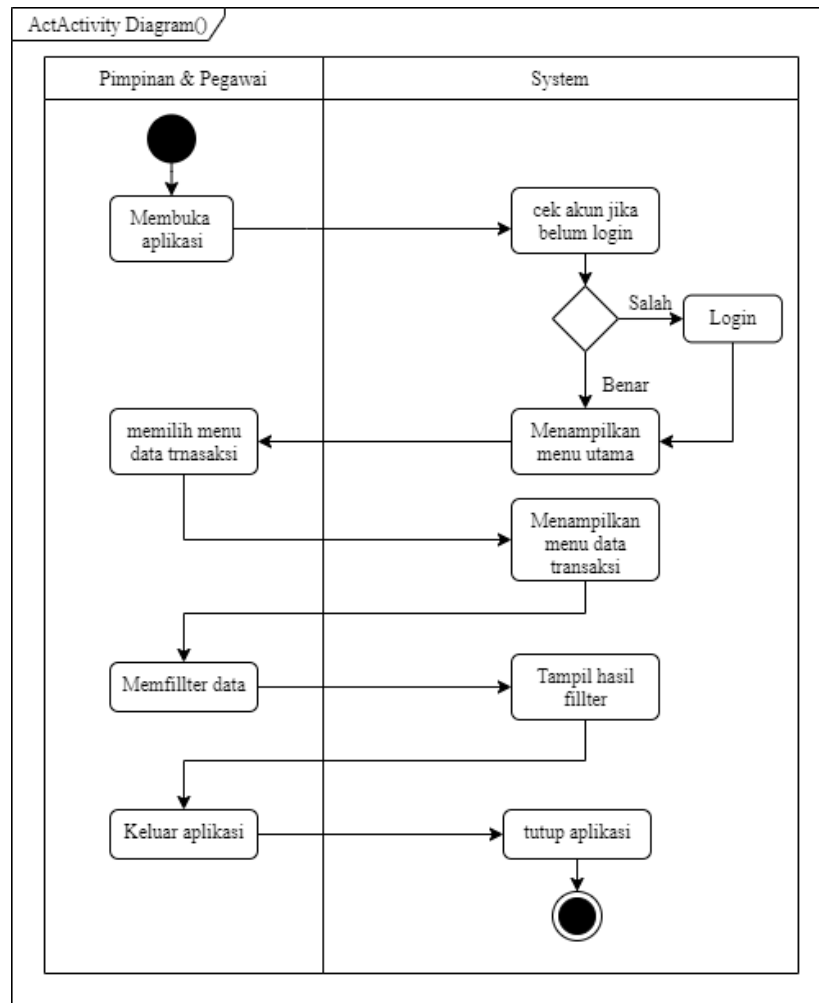
Penjelasan:

Dapat dilihat pada gambar 3.5 tahapan pertama untuk menggunakan filter data transaksi yaitu sistem melakukan autentikasi akun dan kemudian pengguna memilih menu data transaksi selanjutnya pengguna dapat melakukan filter untuk seleksi data yang diinginkan.

#### 4. Activity diagram pencarian data transaksi

Hasil pembuatan *activity diagram* pencarian data transaksi yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.6.





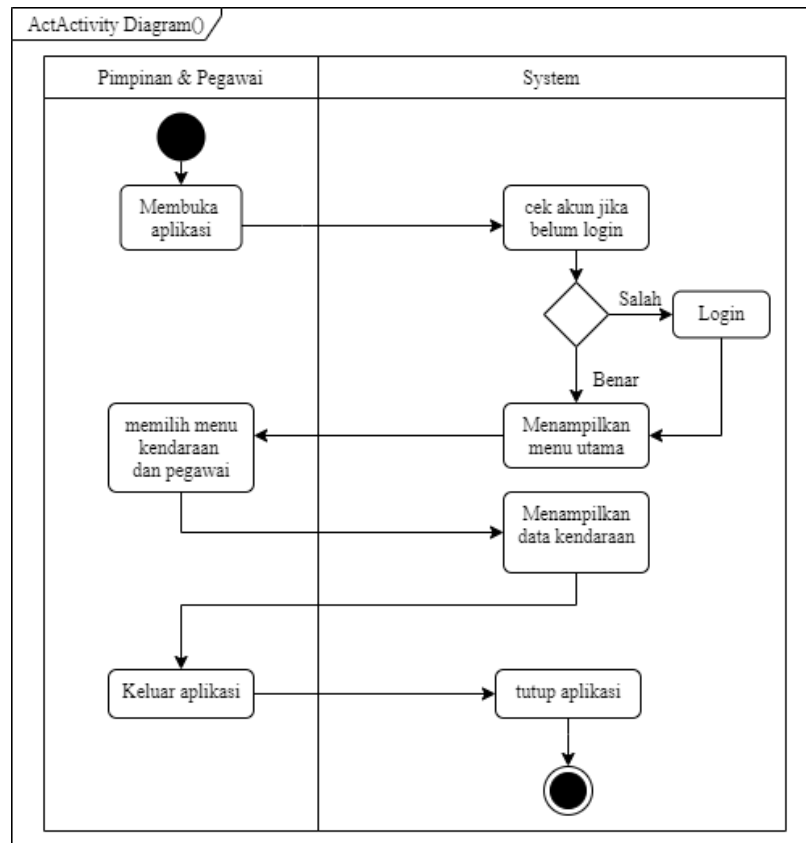
**Gambar 3.6** Activity Diagram Pencarian Data Transaksi

Penjelasan:

Dapat dilihat pada gambar 3.6 hal pertama yang harus dilakukan untuk menggunakan fitur pencarian data yaitu sistem melakukan autentikasi akun dan masuk pada tampilan utama, kemudian pengguna memilih menu data transaksi dan dapat langsung menggunakan filter untuk menampilkan data sesuai kebutuhan.

5. Activity diagram menampilkan data kendaraan dan pegawai

Hasil pembuatan *activity diagram* menampilkan data kendaraan dan pegawai yang dilakukan pimpinan atau pegawai dapat dilihat pada Gambar 3.7.



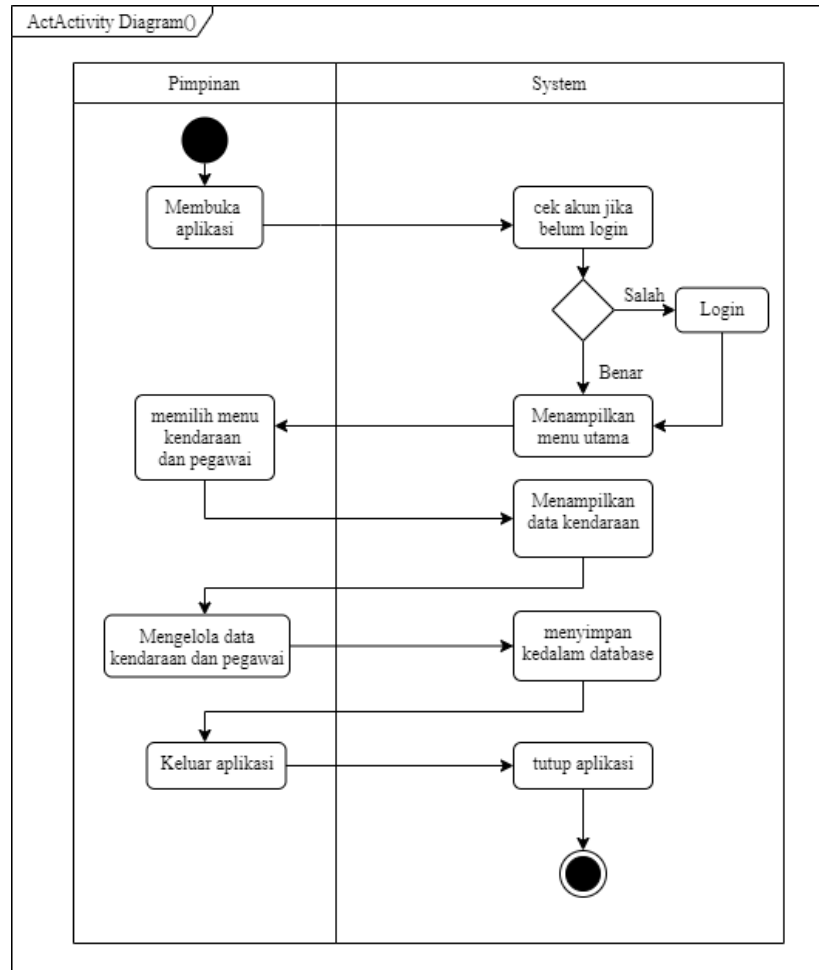
**Gambar 3.7** *Activity Diagram* Menampilkan Data Kendaraan dan Pegawai

Penjelasan:

Berdasarkan activity diagram pada gambar 3.7 tahapan awal untuk menampilkan menu data kendaraan dan pegawai yaitu sistem melakukan autentikasi akun kemudian pengguna dapat langsung memilih menu data kendaraan dan pegawai setelah halaman utama tampil.

#### 6. *Activity diagram* mengelola data kendaraan dan pegawai

Hasil pembuatan *activity diagram* mengelola data kendaraan dan pegawai dapat dilihat pada Gambar 3.8.



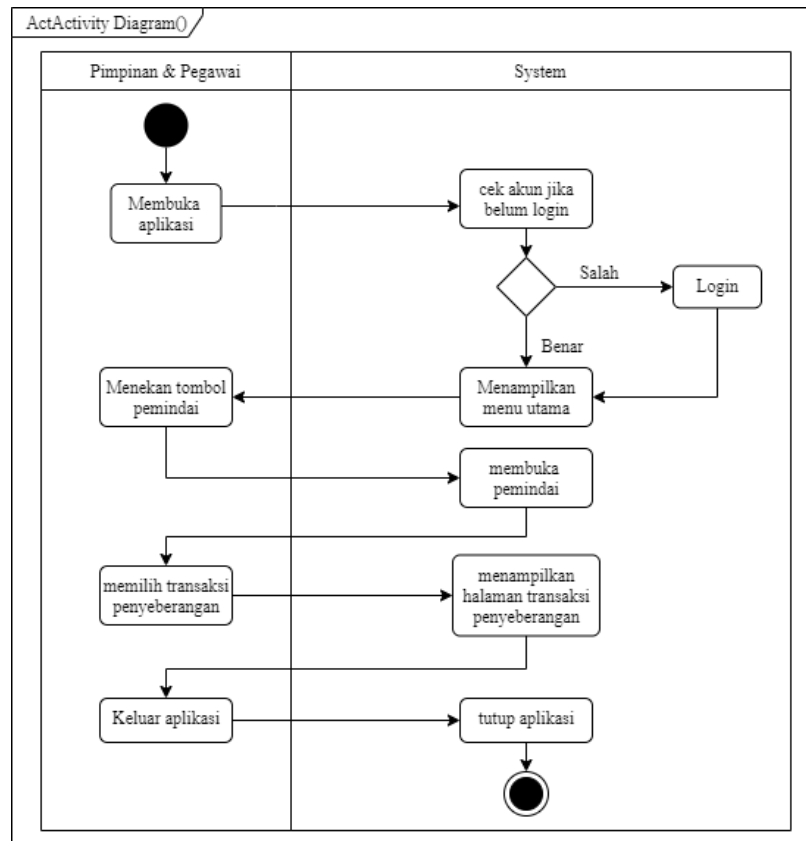
**Gambar 3.8** Activity Diagram Mengelola Data Kendaraan dan Pegawai

Penjelasan:

Pada gambar 3.8 untuk mengelola data kendaraan dan pegawai hal pertama yang dilakukan sama seperti pada activity diagram sebelumnya yaitu sistem melakukan autentikasi akun dan menampilkan halaman utama aplikasi, kemudian pimpinan memilih menu data kendaraan dan transaksi untuk mengelola data yang di inginkan, tahap selanjutnya sistem akan menyimpan perubahan yang dilakukan ke dalam *database*.

#### 7. Activity diagram melakukan transaksi penyeberangan

Hasil pembuatan *Activity diagram* melakukan transaksi penyeberangan yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.9.



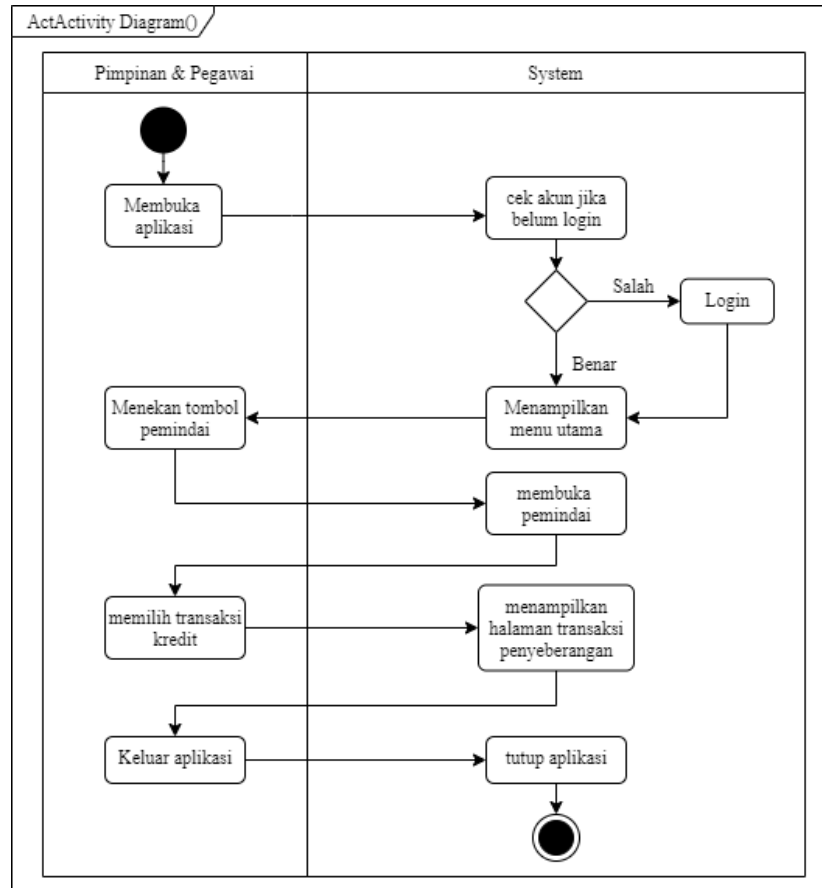
**Gambar 3.9** *Activity Diagram* Melakukan Transaksi Penyeberangan

Penjelasan:

Dapat dilihat pada gambar 3.9 untuk melakukan transaksi penyeberangan hal yang dilakukan setelah melakukan autentikasi akun yaitu menekan tombol pemindai dan dilanjutkan memindai QR-code yang disediakan, kemudian pengguna dapat memilih jenis transaksi penyeberangan dan sistem akan menampilkan halaman untuk mengisi data transaksi.

#### 8. *Activity diagram* melakukan transaksi kredit

Hasil pembuatan *activity diagram* melakukan transaksi kredit yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.10.



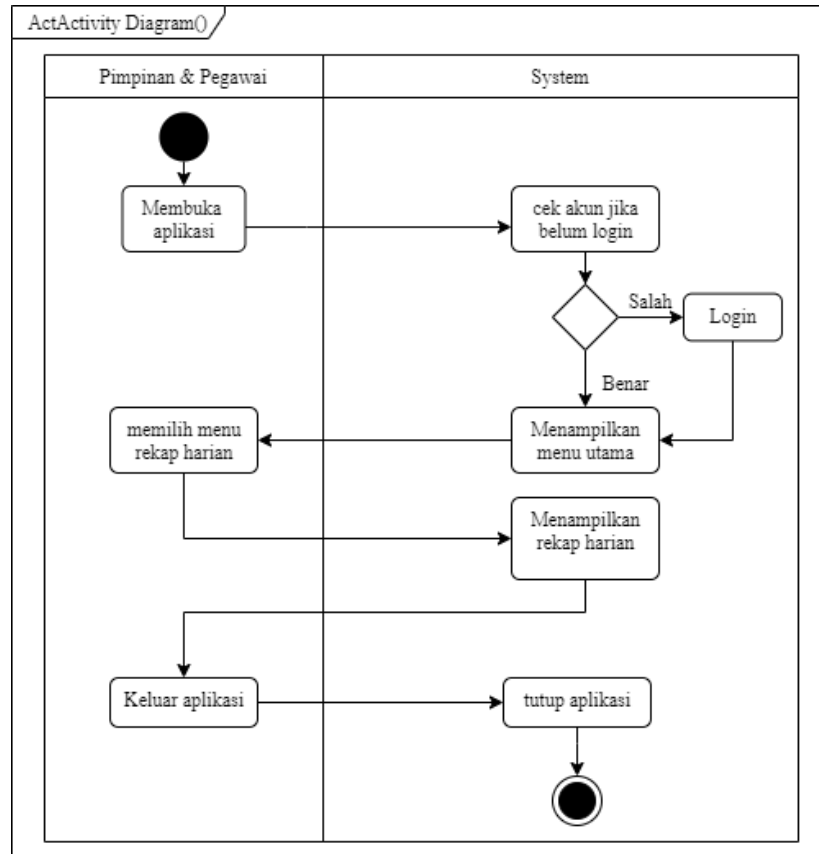
**Gambar 3.10** Activity Diagram Melakukan Transaksi Kredit

Penjelasan:

Berdasarkan gambar 3.10 kegiatan untuk melakukan transaksi kredit tahapan yang dilakukan hampir sama dengan activity diagram pada gambar 3.9, setelah melakukan pemindaian pengguna memilih jenis transaksi kredit dan sistem akan menampilkan halaman untuk mengisi data transaksi kredit.

#### 9. Activity diagram Menampilkan rekap harian

Hasil pembuatan *activity diagram* Menampilkan rekap harian yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.11.



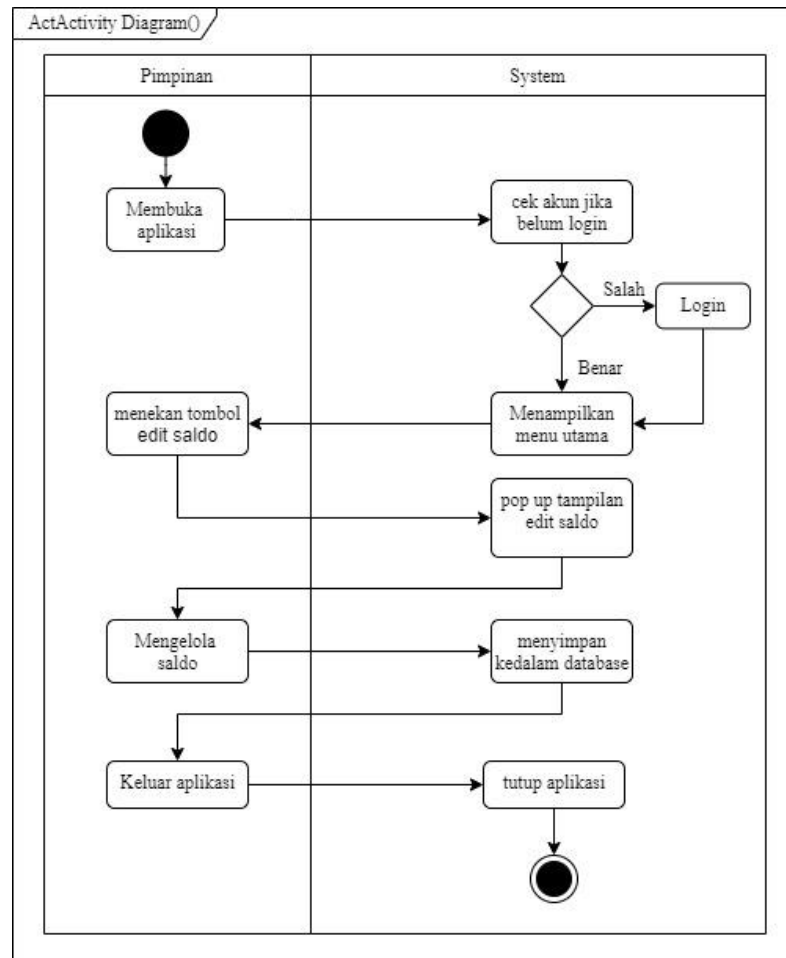
**Gambar 3.11** Activity Diagram Menampilkan Rekap Harian

Penjelasan:

Dapat dilihat pada gambar 3.11 tahapan pertama yang digambarkan pada activity diagram untuk melakukan rekap harian yaitu dengan melakukan login dan dilanjutkan dengan memilih menu rekap harian yang ada pada halaman utama, sistem akan otomatis menghitung rekap dan langsung menampilkan hasil rekap.

#### 10. Activity diagram Menampilkan dan mengelola saldo

Hasil pembuatan Activity diagram Menampilkan dan mengelola saldo yang dilakukan oleh pimpinan yang dapat dilihat pada Gambar 3.12.



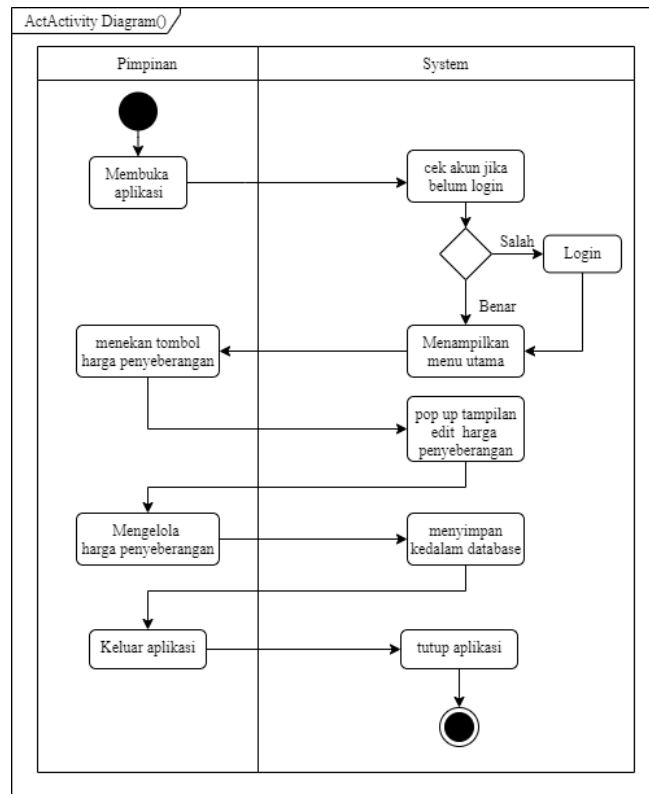
**Gambar 3.12** Activity diagram Menampilkan dan mengelola saldo

Penjelasan:

Berdasarkan activity diagram yang dapat dilihat pada gambar 3.12 tahapan untuk melakukan edit saldo diawali dengan melakukan login dan menekan tombol edit saldo yang berada pada halaman utama. Setelah selesai melakukan perubahan saldo, sistem akan kembali menyimpan perubahan ke *database*.

#### 11. Activity diagram Menampilkan dan mengelola harga pelabuhan

Hasil pembuatan Activity diagram Menampilkan dan mengelola harga pelabuhan yang dilakukan oleh pimpinan yang dapat dilihat pada Gambar 3.13.



**Gambar 3.13** Activity Diagram Menampilkan dan Mengelola Harga Pelabuhan

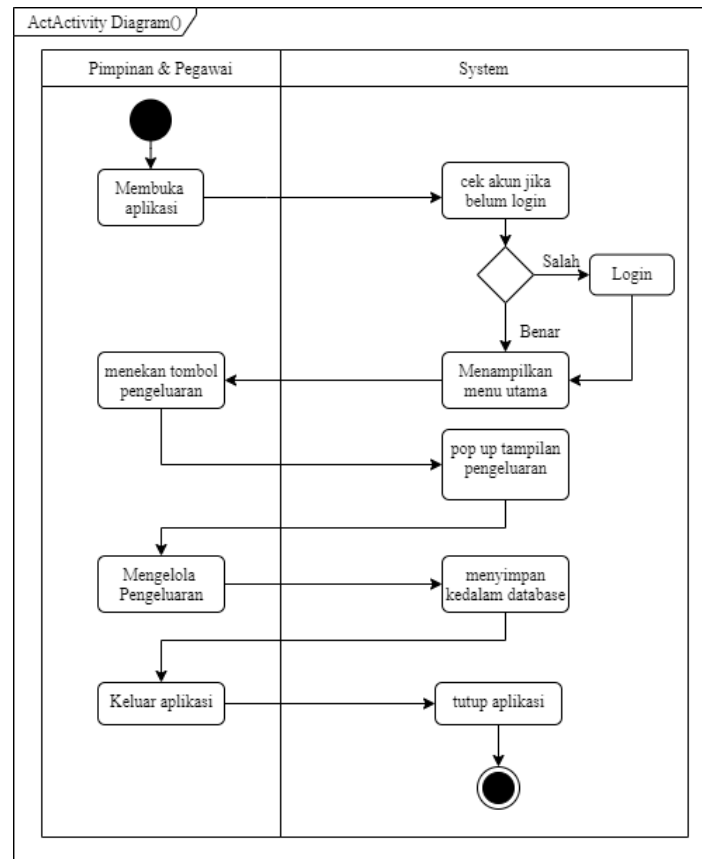
Penjelasan:

Berdasarkan activity diagram pada gambar 3.13 tahapan yang dilakukan untuk menampilkan dan mengelola harga penyeberangan diawali dengan sistem melakukan autentikasi akun kemudian pengguna dapat menekan tombol harga penyeberangan untuk menampilkan dan melakukan perubahan, setelah perubahan dirasa cukup pimpinan dapat menutup tampilan dan sistem akan menyimpan data kedalam *database*.

## 12. Activity diagram Menambahkan pengeluaran

Hasil pembuatan *Activity diagram* menambahkan pengeluaran yang dilakukan oleh pimpinan dan pegawai yang dapat dilihat pada Gambar 3.14.





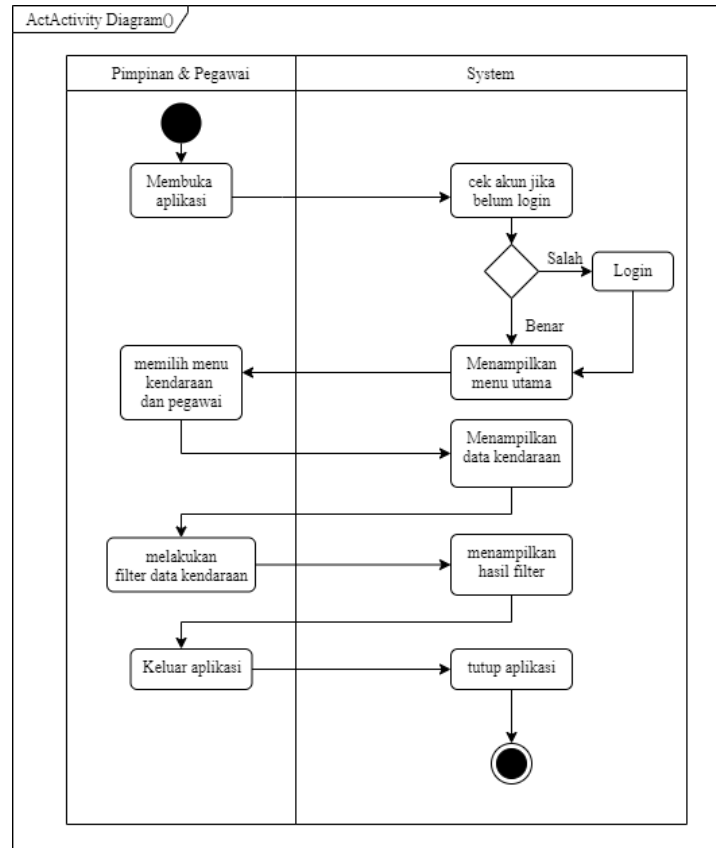
**Gambar 3.14** Activity Diagram Menambahkan Pengeluaran

Penjelasan:

Menurut activity diagram pada gambar 3.14 tahapan yang ditunjukkan untuk menambahkan pengeluaran yaitu diawali dengan sistem melakukan autentikasi akun dan menampilkan halaman utama aplikasi, kemudian pengguna dapat langsung menekan tombol pengeluaran yang berada pada menu utama dan pengguna dapat langsung menambahkan pengeluaran, setelah proses selesai data pengeluaran akan langsung tersimpan ke dalam *database*.

### 13. Activity diagram filter data kendaraan

Hasil pembuatan *Activity diagram* filter data kendaraan yang dilakukan oleh pimpinan dan pegawai yang dapat dilihat pada Gambar 3.15.



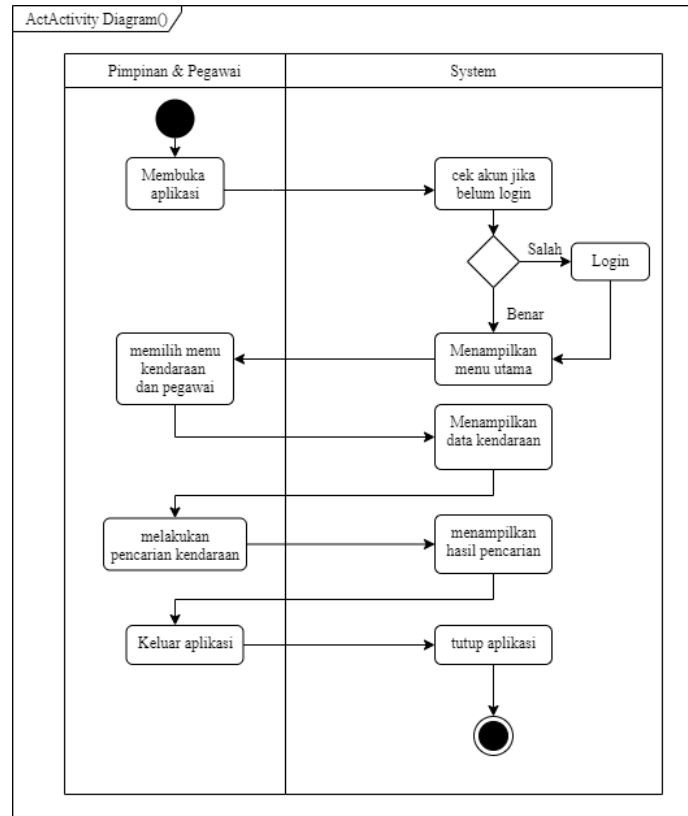
**Gambar 3.15** *Activity Diagram* Filter Data Kendaraan

Penjelasan:

Dapat dilihat pada gambar 3.15 untuk menggunakan filter data kendaraan tahapan yang harus dilakukan yaitu sistem melakukan autentikasi akun dan menampilkan halaman utama, kemudian pengguna memilih halaman data kendaraan dan kemudian pengguna dapat langsung menggunakan filter untuk menampilkan data kendaraan sesuai kebutuhan.

#### 14. *Activity diagram* pencarian data kendaraan

Hasil pembuatan *activity diagram* pencarian data kendaraan yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.16.



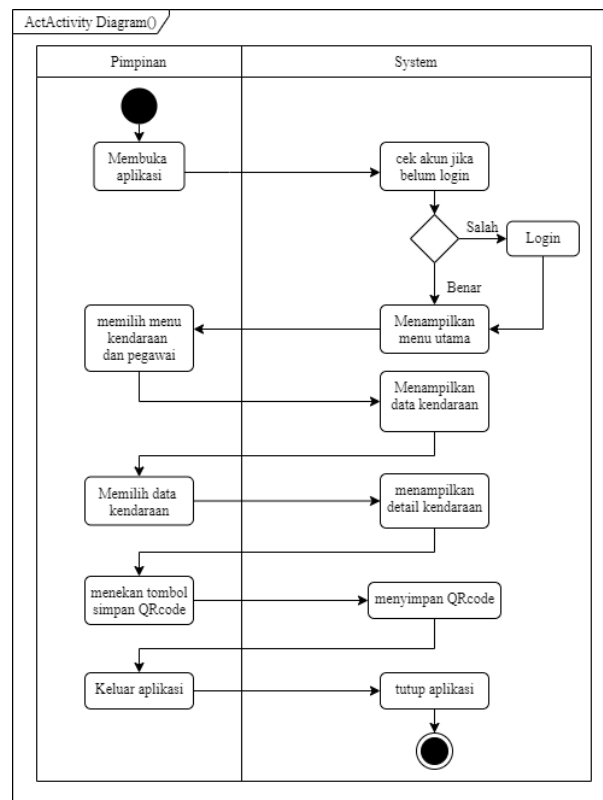
**Gambar 3.16** Activity Diagram Pencarian Data Kendaraan

Penjelasan:

Berdasarkan activity diagram Pada gambar 3.16 langkah awal untuk menggunakan pencarian data kendaraan yaitu sistem melakukan autentikasi akun dan menampilkan halaman utama aplikasi, selanjutnya pengguna memilih menu data kendaraan dan fitur pencarian dapat langsung digunakan.

#### 15. Activity diagram simpan QR-code

Hasil pembuatan activity diagram simpan QR-code yang dilakukan oleh pimpinan dan pegawai dapat dilihat pada Gambar 3.17.



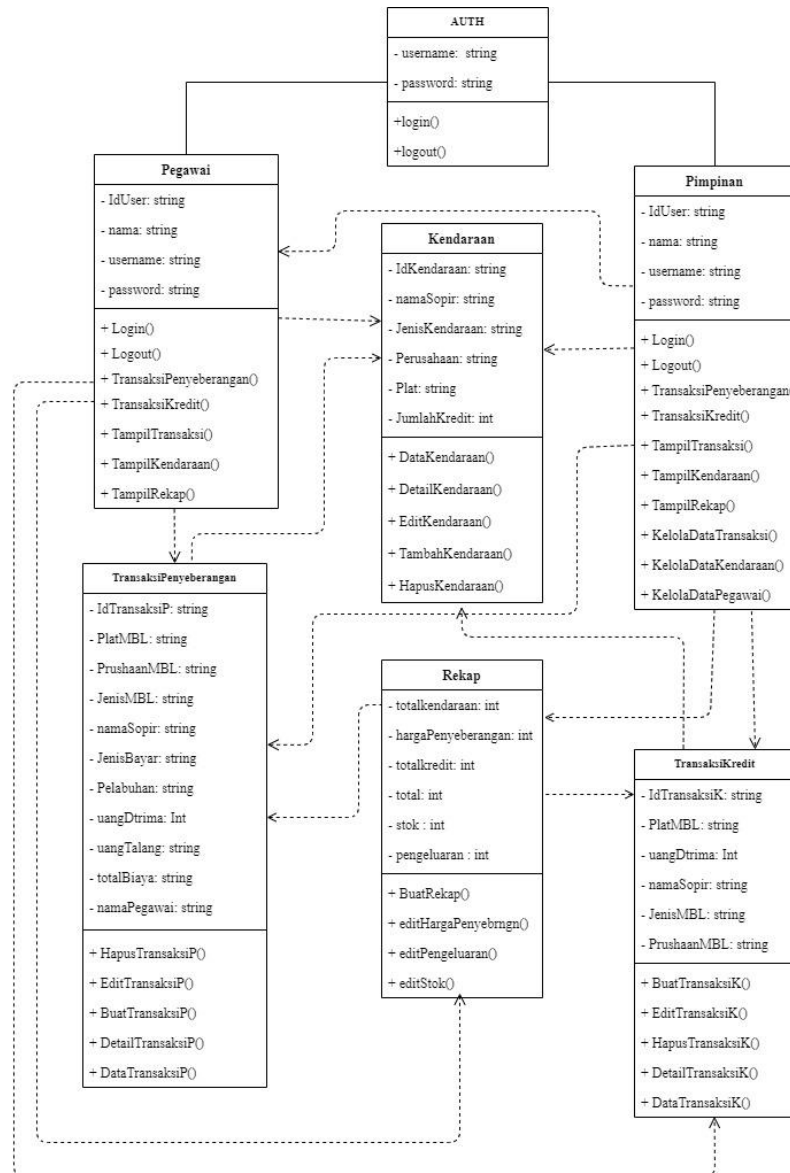
**Gambar 3.17** Activity Diagram Simpan QR-Code Kendaraan

Penjelasan:

Berdasarkan activity diagram pada gambar 3.17 tahapan awal untuk menyimpan QR-code kendaraan yaitu sistem melakukan autentikasi akun dan masuk kehalaman utama aplikasi, kemudian pengguna membuka menu data kendaraan dan memilih data kendaraan yang diinginkan dan selanjutnya sistem akan langsung menampilkan detail dari kendaraan tersebut, pada tampilan detail kendaraan pengguna dapat langsung menekan tombol simpan QRcode dan sistem akan menyimpan file Qrcode ke dalam perangkat.

### 3.4.3 Class Diagram

*Class diagram* yang dibuat untuk menggambarkan suatu struktur dari kelas-kelas yang akan digunakan dalam mengembangkan sistem pada penelitian ini dapat dilihat pada Gambar 3.18.



**Gambar 3.18** Class diagram aplikasi pendataan transaksi

### 3.4.1 Class Responsibility Collaborator Card (CRC Card).

Berikut ini merupakan *CRC Card* yang terdiri dari *class*, *responsibility*, dan *collaborator* dari sistem yang akan dibuat yang di mana pegawai dan pimpinan sudah melakukan login terlebih dahulu:

**Tabel 3.7** CRC Card AUTH

Nama Class: Auth	
Responsibility	Collaborator
Dapat login ke sistem	Pegawai
	Pimpinan

**Tabel 3.8 CRC Card Pegawai**

<b>Nama Class: Pegawai</b>	
<b><i>Responsibility</i></b>	<b><i>Collaborator</i></b>
Melakukan transaksi penyeberangan	TransaksiPenyeberangan
Menampilkan data transaksi	
Menampilkan data kreditur	Kreditur
Melakukan transaksi kreditur	
Menampilkan data kendaraan	Kendaraan
Menampilkan data rekap harian	Rekap

**Tabel 3.9 CRC Card Pimpinan**

<b>Nama Class: Pimpinan</b>	
<b><i>Responsibility</i></b>	<b><i>Collaborator</i></b>
Melakukan transaksi penyeberangan	TransaksiPenyeberangan
Menampilkan dan mengelola data transaksi	
Melakukan transaksi kreditur	TransaksiKreditur
Menampilkan dan mengelola data kreditur	
Menampilkan dan mengelola data kendaraan	Kendaraan
Menampilkan dan engelola data pegawai	Pegawai
Menampilkan data rekap harian	Rekap

**Tabel 3.10 CRC Card Transaksi Penyeberangan**

<b>Nama Class: TransaksiPenyeberangan</b>	
<b>Responsibility</b>	<b>Collaborator</b>
Menghapus data transaksi penyeberangan	
Mengubah data transaksi penyeberangan	
Membuat transaksi penyeberangan	Kendaraan
Membuat detail transaksi	

**Tabel 3.11 CRC Card Kendaraan**

<b>Nama class: Kendaraan</b>	
<b>Responsibility</b>	<b>Collaborator</b>
Mengubah data kendaraan	
Menambah data kendaraan	
Menghapus data kendaraan	

**Tabel 3.12 CRC Card rekap**

<b>Nama class: Rekap</b>	
<b>Responsibility</b>	<b>Collaborator</b>
Membuat rekap	TransaksiPenyeberangan
	TransaksiKreditur

**Tabel 3.13 CRC Card Transaksi Kreditur**

<b>Nama class: TransaksiKreditur</b>	
<b>Responsibility</b>	<b>Collaborator</b>
Menampilkan data transaksi kreditur	Kendaraan
Membuat detail transaksi kreditur	

### 3.4.2 *Prototype Design*

Pada tahap ini pembuatan *prototype user interface* sesuai dengan kebutuhan menggunakan Figma. Bertujuan untuk memberi gambaran aplikasi yang akan dibuat kepada klien. Berikut ini merupakan tampilan yang telah penulis buat:

#### 1. Halaman *login*

Tampilan login merupakan tampilan awal ketika mengakses aplikasi. Pada tampilan ini sebagai pemisah antara *user* pegawai dan pimpinan, kegiatan harus dilakukan pengguna yaitu memasukkan *username* dan *password*, tampilan login dapat dilihat pada Gambar 3.19.

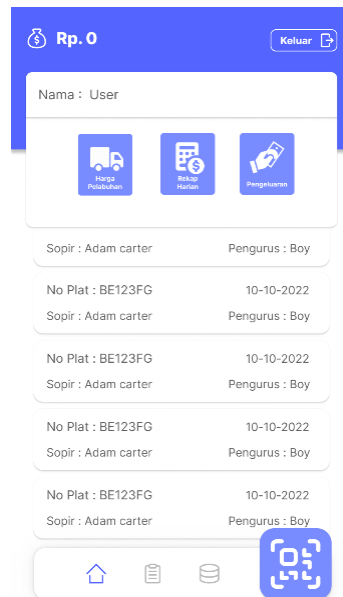


**Gambar 3.19** Tampilan *Login*

#### 2. Halaman home pimpinan dan karyawan

Ketika pegawai maupun pimpinan berhasil *login* maka akan muncul halaman *home*, pada tampilan awal terdapat jumlah saldo, list transaksi secara *real time*, dan tombol yang berguna untuk melakukan pemindan. halaman *home* pengguna dapat dilihat pada Gambar 3.20.

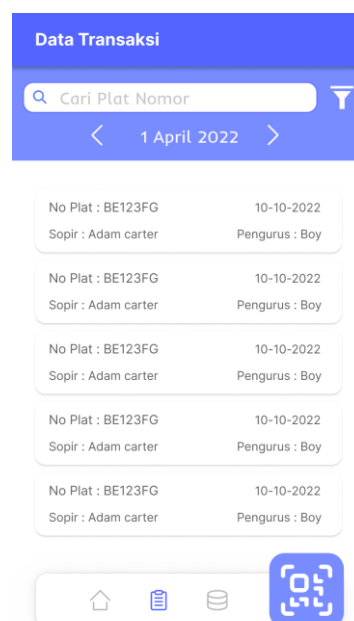




**Gambar 3.20** Halaman Home

### 3. Halaman data transaksi

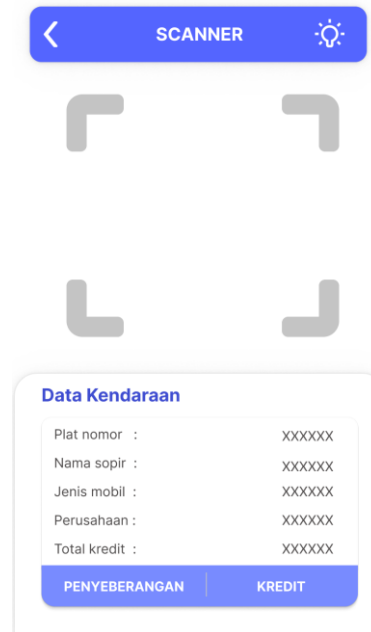
Pada halaman transaksi pengguna dapat melakukan pencarian data transaksi dan melihat detail transaksi, pada halaman ini pimpinan dapat merubah detail transaksi jika ada kesalahan, tampilan data transaksi dapat dilihat pada gambar 3.21.



**Gambar 3.21** Halaman Data Transaksi

#### 4. Halaman *Scanner*

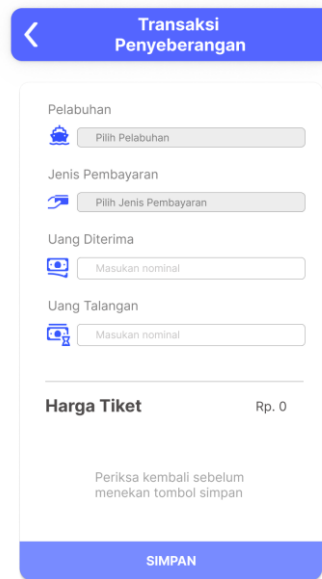
Halaman *scanner* berguna untuk pemindaian *QR-code* kendaraan sebelum melakukan transaksi, pada tampilan ini dapat menampilkan data kendaraan setelah melakukan pemindaian untuk memastikan kendaraan yang melakukan transaksi sesuai, tampilan *scanner* dapat dilihat pada gambar 3.22.



**Gambar 3.22** Halaman *Scanner*

#### 5. Halaman proses transaksi penyeberangan

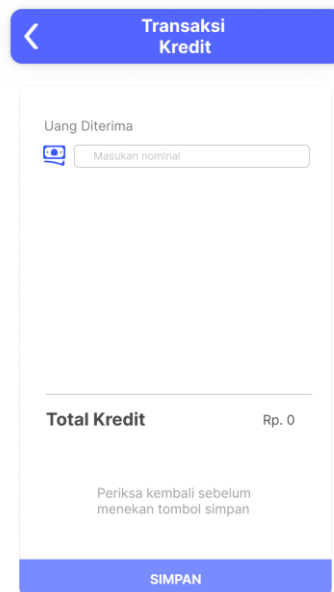
Halaman proses transaksi berguna untuk menginputkan data transaksi penyeberangan seperti jenis Pelabuhan, jenis pembayaran dan uang diterima dan uang talangan, halaman proses transaksi penyeberangan dapat dilihat pada gambar 3.23.



**Gambar 3.23** Halaman Transaksi Penyeberangan

6. Halaman proses transaksi kredit

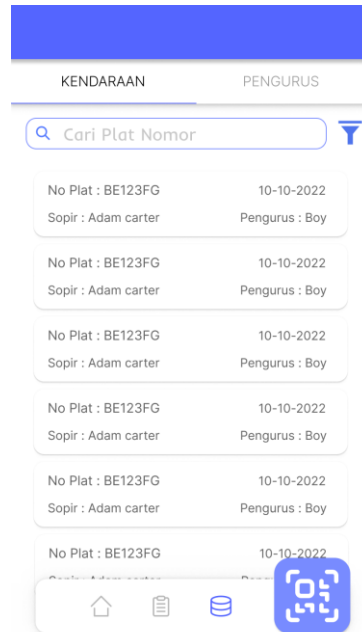
Halaman proses transaksi kreditur berguna untuk melakukan transaksi pembayaran kredit dan sebelum melakukan transaksi harus melakukan pemindaian *QR-code* terlebih dahulu, halaman proses transaksi kreditur dapat di lihat pada gambar 3.24.



**Gambar 3.24** Halaman Transaksi *Kredit*

## 7. Halaman kendaraan dan pegawai

Halaman data kendaraan dan pegawai berguna untuk menampilkan data kendaraan dan data pegawai, pada halaman ini pegawai hanya bisa melihat detail data kendaraan dan pegawai sedangkan pimpinan dapat melakukan edit, hapus dan menambah data, halaman data dapat dilihat pada gambar 3.25.



**Gambar 3.25** Halaman Data Kendaraan Dan Pegawai

## 8. Halaman harga pelabuhan.

Pada halaman ini terdapat fitur untuk menampilkan harga-harga pelabuhan sesuai dengan jenis kendaraan dan jenis pelabuhan, fitur ini dapat digunakan oleh pimpinan dan pegawai, selanjutnya di dalam halaman dapat mengubah harga pelabuhan yang hanya bisa dilakukan oleh, halaman dapat dilihat pada gambar 3.26.

**Gambar 3.26** Halaman Harga Pelabuan

9. Halaman tambah pengeluaran harian

Pada halaman ini memiliki fungsi menambah pengeluaran harian yang berisi text input untuk memasukan nilai pengeluaran dan keterangan pengeluaran. Halaman dapat dilihat pada gambar 3.267.

**Gambar 3.27** Halaman Tambah Pengeluaran Harian

#### 10. Halaman rekap harian

Pada halaman ini berfungsi menghitung dan menampilkan hasil rekap setiap harinya, perhitungan rekap dilakukan oleh sistem agar mempermudah jasa dalam perekapan harian, halaman ini dapat diakses oleh pimpinan dan pegawai. Halaman dapat dilihat pada gambar 3.28.

Rekap Harian  
1 April 2022

Kendaraan	Pelabuhan	Jumlah	Harga
truk	eksekutif	2	1400000
tronton	reguler	2	1000000
truk	swasta	2	1200000
Total			3600000

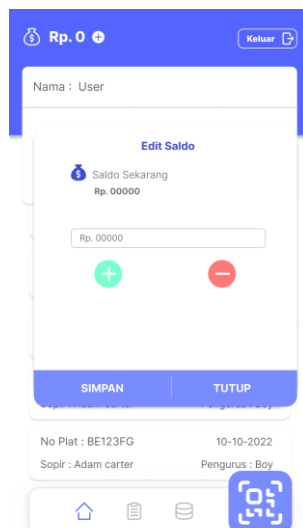
Uang diterima	3000000
Kredit	600000
Modal	4000000
Uang masuk	8200000
Bayar tiket Penyeberangan	3600000
Pengeluaran	600000
<b>Hasil Akhir</b>	<b>Rp. 6.660.000</b>

TUTUP

**Gambar 3.28** Halaman Rekap Harian

#### 11. Halaman edit saldo

Fungsi utama dari halam ini yaitu mengubah saldo yang hanya dapat diakses oleh pimpinan, pimpinan dapat mengurangi dan menambahkan saldo yang ada jika dibutuhkan. Halaman dapat dilihat pada gambar 2.29.



**Gambar 3.29** Halaman Edit Saldo

### 3.5 Jadwal Penelitian

Dibawah ini adalah jadwal penelitian proposal skripsi yang telah dibuat oleh penulis:

**Tabel 3.18** Jadwal Penelitian

No	Kegiatan Penelitian	Tahun 2022																			
		Maret				April				Mei				Juni				juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1.	Wawancara																				
2.	Studi Literatur																				
3.	Perencanaan																				
4.	Perancangan																				
5.	Penyusunan Proposal Skripsi																				
6.	Seminar Proposal Skripsi																				
7.	Pengkodean																				
8.	Pengujian																				
9.	Sidang Skripsi																				

Keterangan:  = Telah Dilaksanakan  = Belum Dilaksanakan

## DAFTAR PUSTAKA




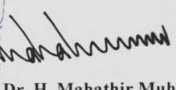
- Alham Romadony, Willi, Muhammad Ardianto, Wisnu Kartiko Arie Pangestu, Didih Rizki Chandranegara, Wildan Suharso, Universitas Muhammadiyah Malang, Malang Kontak Person, and Wildan Suharso Jalan Raya Tlogomas No. 2019. "Sistem Pencatatan Keuangan Toko Berbasis Android." *Seminar Nasional Teknologi Dan Rekayasa (SENTRA)2019* 2527–6050.
- Ariyanti, Lisa, Muhammad Najib, Dwi Satria, and Debby Alita. 2020. *Sistem Informasi Akademik Dan Administrasi Dengan Metode Extreme Programming Pada Lembaga Kursus Dan Pelatihan*. Vol. 1.
- Artina, Nyimas. 2006. "Penerapan Analisis Kebutuhan Metode *Use Case* Pada Metode Pengembangan Terstruktur." *Jurnal Ilmiah STMIK G1 MDP* 2(3):1–6.
- Ayu, Vittalis. 2017. *Pemodelan Proses Pemilihan Rute Pada Protokol Babel Dengan Activity Diagram Dan Transition System*. Vol. 12.
- Cahyati, Yuliana, and Hari Murti. 2018. *Sistem E-Surat Pada Government Resource Management System Provinsi Jawa Tengah Berbasis Framework Codeigniter*.
- Fatoni, Ahmad, and Dhany Dwi. 2016. "Rancang Bangun Sistem Extreme Programming Sebagai Metodologi Pengembangan Sistem." 3(1).
- Gultom, Dedek K., Paham Ginting, and Beby Kf Sembiring. 2014. *Pengaruh Bauran Pemasaran Jasa Dan Kualitas Pelayanan Terhadap Kepuasan Mahasiswa Program Studi Manajemen Fakultas Ekonomi Universitas Muhammadiyah Sumatera Utara*. Vol. 14.
- Hansson, Niclas, and Tomas Vidhall. 2016. *Institutionen För Datavetenskap Effects on Performance and Usability for Cross-Platform Application Development Using React Native*.
- Hendini, Ade. 2016. *Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak)*. Vol. IV.
- Iskandar, and Umar Tsani Abdurrahman. 2020. "Perancangan Aplikasi Kasir *Point Of Sales* Berbasis Android Menggunakan Metode Rapid Application Development Untuk Usaha Retail." *INFOTECH: Jurnal Informatika & Teknologi* 1(2):67–77. doi: 10.37373/infotech.v1i2.62.
- Ismail, Muh, Al Ghazali Syam, Riwayat Artikel, and Kata kunci. 2021. *Aplikasi Qr Code Sebagai Sarana Penyampaian Informasi Pohon Dikebun Raya Jompie Informasi Artikel*. Vol. 1.




- Juhardi, Ujang, and Khairullah. 2019. "Sistem Pencatatan Dan Pengolahan Keuangan Pada Aplikasi Manajemen Keuangan E-Dompet Berbasis Android." *JTIS* 2(1):2614–3089.
- Loveri, Tomi Tm. 2018. "Sistem Informasi Aplikasi Pengelolaan Transaksi Keuangan Dan Pendataan Konsumen Pada Cv. Puplas." *Jurnal Sains Dan Informatika* 4(2):139. doi: 10.22216/jsi.v4i2.3584.
- Mubarak, Abdul, Jl Jati Metro, and Kota Ternate Selatan. 2019. *Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek*. Vol. 02.
- Nugraha, Made Muliana, Nur Muhammad Luthf, and Hetti Hidayati. 2020. "Aplikasi Pengelolaan Sport Club Universitas Telkom Berbasis Website." *E-Proceeding of Applied Science* 6(2):2562.
- Saputra, Agung Dwi, Agi Putra Kharisma, and Lutfi Fanani. 2021. *Pengembangan Aplikasi Klinik Kecantikan Sebagai Pengelola Transaksi Berbasis Android Menggunakan Metode Prototype (Studi Kasus :Klinik Kecantikan CV Nana Beautyskin)*. Vol. 5.
- Solikin, Imam. 2014. "Perancangan Sistem Informasi Penjualan Berbasis Framework MVC Pada PT Thamrin Brother Perancangan Sistem Infomasi Penjualan Berbasis Framework Model View Controller (Mvc) Pada Pt Thamrin Brother Cabang Oki." *Jurnal Media Informatika Dan Komputer* 4(1).
- Sovia Maranti, Ovi, Lis Saumi Ramdhani, Rusli Nugraha, Khairul Rizal, Stmik Nusa, and Mandiri Sukabumi. 2018. "Rancang Bangun Aplikasipengelolaanpinjaman Koperasi Berbasis *Mobile* pada Koperasi Pkk Sejahtera Sukabumi." *Jurnal Swabumi* 6(1).
- Widodo. 2008. *Extreme Programming : Pengembangan Perangkat Lunak Semi Formal*.

## LAMPIRAN


## Lampiran 1. Dokumen Penelitian

 <b>DOKUMEN SURAT PENELITIAN</b>	
<div style="display: flex; justify-content: space-between; align-items: center;">  <div> <b>UNIVERSITAS TEKNOKRAT INDONESIA</b>  <b>FAKULTAS TEKNIK DAN ILMU KOMPUTER</b> </div> </div>	
Nomor : 008/FTIK-S1.I/B.7.12/X/2021	14 Oktober 2021
Perihal : Permohonan Izin Penelitian	
Yth. <b>Pimpinan Jasa Penyebrangan BC Bakauheni</b> Jl. Lintas Sumatera, Bakauheni Lampung Selatan	
Dengan hormat,	
Sehubungan dengan penulisan skripsi mahasiswa Program Studi S1 Informatika Fakultas Teknik dan Ilmu Komputer Universitas Teknokrat Indonesia, dengan ini kami mohon kesediaan Bapak/Ibu untuk memperkenalkan mahasiswa/i kami melakukan penelitian pada perusahaan yang Bapak/Ibu pimpin.	
Adapun mahasiswa/i kami yang melakukan penelitian adalah:	
nama	: Rahmad Suryadi
NPM	: 17312031
judul penelitian	: Rancang Bangun Aplikasi Pedataan Transaksi Jasa Penyebrangan Dengan <i>Framework React Native</i> Berbasis Android (Studi Kasus: BC. Bakauheni)
Demikian permohonan ini, atas izin yang diberikan kami ucapkan terima kasih.	
 <div> Hormat kami,  Dekan,    <b>Dr. H. Mahathir Muhammad, S.E, M.M.</b> </div>	
<b>FAKULTAS TEKNIK DAN ILMU KOMPUTER</b> <small>TEKNIK ELEKTRO (S1), TEKNIK SIPIL (S1), TEKNIK KOMPUTER (S1), TEKNOLOGI INFORMASI (S1), INFORMATIKA (S1),</small>	

## Lampiran 2. Dokumen hasil kegiatan wawancara

	<b>DOKUMEN HASIL WAWANCARA</b>

## Lampiran 3. Foto kegiatan wawancara

	<b>FOTO WAWANCARA</b>