

# APLIKASI *GATE PORTABLE* PADA TERMINAL PETI KEMAS PELABUHAN BATU AMPAR PT PERSERO BATAM BERBASIS *MOBILE*

Anggun Dini Wahyuni<sup>1</sup>, Muchammad Fajri Amirul Nasrullah<sup>2</sup>

<sup>1,2</sup> Teknik Informatika, Politeknik Negeri Batam

[anggundiniwahyuni@gmail.com](mailto:anggundiniwahyuni@gmail.com)<sup>1</sup>, [fajri@polibatam.ac.id](mailto:fajri@polibatam.ac.id)<sup>2</sup>

## Article Info

### Article history:

Received ...

Revised ...

Accepted ...

### Keyword:

Container Terminal, Mobile – based Application, QR Code Scanning

## ABSTRACT

The development of a mobile-based Portable Gate application to facilitate the Staff in process of receiving and delivery at the Batu Ampar Port Container Terminal PT Persero Batam. This application is designed to assist and simplify the process of verifying and recording container data automatically at the work site at gate in and gate out activities during receiving and delivery which is built using the Dart programming language and Flutter framework, and integrates the REST API from the BCTOS system. With the mobile-based application, field officers can verify container data by scanning the QR code through a camera and recording activity data directly at the work site using a mobile device. The system development method used is a prototyping method that involves users and system owners in the application development and evaluation process. The test results show that all application features can run as their functions.



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

## I. PENDAHULUAN

Dalam era digital, perkembangan teknologi memberikan dampak yang signifikan di berbagai sektor, sehingga meningkatnya fleksibilitas akses terhadap data dan informasi. Salah satunya perkembangan aplikasi berbasis *mobile* yang memiliki keunggulan dengan tingkat aksesibilitas yang tinggi dengan memanfaatkan berbagai fitur pada perangkat mobile seperti kamera, GPS, dan lainnya. Hal ini memungkinkan pengguna untuk mengakses informasi, mengerjakan tugas dan mendapatkan pengalaman yang lebih interaktif di mana pun [1].

PT Perusahaan Daerah Industri Pulau Batam atau dikenal dengan PT Persero Batam merupakan perusahaan BUMN yang bergerak di bidang logistik dan saat ini berperan sebagai operator Terminal Peti Kemas di Pelabuhan Batu Ampar. Pelabuhan ini memiliki nilai bongkar muat yang terus meningkat setiap tahunnya [2].

Salah satu alur kerja bongkar muat adalah aktivitas *receiving* dan *delivery* pemindahan barang dari lapangan penumpukan untuk diserahkan ke kendaraan maupun sebaliknya [3]. Dalam proses ini, verifikasi data *Receiving Card (RC)* saat *receiving* dan Surat Penyerahan Peti Kemas

(SP2) saat *delivery* sangat penting karena memuat identitas kontainer dan sebagai bukti penerimaan dan penyerahan kontainer [4]. Durasi dan akurasi dalam aktivitas ini bersifat krusial, karena dapat mempengaruhi kegiatan lainnya.

Dalam pengoperasiannya, PT Persero Batam menggunakan sistem berbasis *website* yang bernama *Batam Container Terminal Operation System (BCTOS)*. Namun, sistem ini tidak dapat digunakan dalam mendukung proses *receiving* dan *delivery* saat *gate in* dan *gate out* secara langsung di lapangan. Sehingga, petugas sulit untuk memverifikasi dan mencatat data secara manual di kertas yang beresiko kehilangan data. Data pada kegiatan ini diperlukan saat penagihan kepada agen atau *customer*. Dengan pengolahan data yang manual ini, juga dapat menghambat proses penagihannya.

Berdasarkan hasil diskusi dengan pihak PT Persero Batam, pengembangan aplikasi *Gate Portable* berbasis *mobile* menjadi solusi yang dapat diberikan. Aplikasi dibangun menggunakan bahasa pemrograman *Dart*, *framework Flutter*, dan pengintegrasian *REST API* dari sistem *BCTOS* dengan beberapa fitur pendukung seperti pemindaian kode QR, pendataan, dan monitoring data dari

kegiatan *gate in* dan *gate out* pada *receiving* dan *delivery*. Dengan aplikasi ini, diharapkan dapat memudahkan dan membantu petugas saat proses verifikasi dan pencatatan data sekaligus memonitoring data secara langsung di lapangan kerja melalui perangkat *mobile*.

## II. LANDASAN TEORI

### A. Penelitian Terdahulu

Pada penelitian yang dilakukan oleh Nyayu Yulia Irianti dengan judul “Aplikasi *Container Location Barcode Scanner Berbasis Android* pada Pelabuhan Indonesia II (Persero) Cabang Palembang dengan Metode *Agile*”, membantu petugas *Gate In* dan *Tally* untuk mengakses informasi lokasi *container* secara akurat, mengurangi antrian dan waktu penyimpanan *container*, serta memudahkan proses pemuatan *container* ke kapal dengan fitur yang mendukung, seperti *scan barcode* melalui kamera atau laser *barcode scanner* [5].

Penelitian yang dilakukan oleh Nur Hanifah dengan judul “Perancangan Sistem Aplikasi *Gate Mobile* Berbasis *Android* di TPS dan TPB (Studi Kasus: KPPBC TMP A Bekasi)”, sistem berbasis *mobile* yang digunakan di wilayah operasional, dirancang untuk merekam waktu dan kondisi pemasukan atau pengeluaran barang dari Tempat Penimbunan Sementara (TPS) atau Tempat Penimbunan Berikat (TPB) di wilayah Pabean dengan fitur yang terdiri dari *Gate In* atau *Gate Out* pada TPS berdasarkan kontainer atau kemasan, dokumen, tanggal awal dan tanggal akhir, dan fitur *scan* nomor polisi. Dengan adanya sistem ini, membantu petugas dalam pencatatan data secara langsung di lapangan, tanpa harus melakukan pencatatan manual [6].

Pada penelitian yang dilakukan oleh Bay Haqi, Jonser Sinaga dan Heri Satria Setiawan dengan judul “Aplikasi *Mobile Scan* Parkir Berbasis *QR Code* pada *Co-Working Space* Ruang Hampa”, sistem parkir yang digital dengan memanfaatkan aplikasi *mobile* sebagai *scanner* dan menggunakan teknologi *QR Code* untuk meningkatkan keamanan dan validasi sistem parkir dengan fitur diantaranya *scan QR Code* pada struk cetak bukti parkir pengendara, pencatatan waktu masuk dan keluar parkir pengendara, dan pemeriksaan validitas saat kendaraan masuk dan keluar dari area parkir. Dengan aplikasi ini, digitalisasi dan peningkatan keamanan sistem pada prosedural parkir dapat terealisasi [7].

### B. Aplikasi Mobile

Aplikasi *mobile* merupakan salah satu perangkat lunak yang dikembangkan menggunakan pemrograman komputer yang berjalan di perangkat *mobile* seperti ponsel, tablet, dan lainnya [1]. Sistem operasi pada perangkat *mobile* adalah *Android* dan *iOS*. Aplikasi ini dapat dibangun menggunakan beberapa bahasa pemrograman,

antara lain adalah *Kotlin*, *Java*, *Objective C*, *Swift*, *Dart*, *JavaScript*, dan *React Native*. Beberapa contoh aplikasi berbasis *mobile*, yaitu Instagram, Grab, Gojek, Spotify, dan lainnya [8].

### C. Alur Kerja Receiving dan Delivery Terminal Peti Kemas

Terminal Peti Kemas merupakan salah satu layanan di Pelabuhan yang berperan sebagai tempat transit antar moda transportasi, yaitu kapal dan truk kontainer untuk melakukan kegiatan operasionalnya. Salah satu kegiatan operasionalnya adalah kegiatan *receiving/delivery*. Menurut Peraturan Pemerintah Republik Indonesia Tahun 2010 tentang Angkutan Perairan, bahwa kegiatan *receiving* dan *delivery* merupakan kegiatan pemindahan barang dari lokasi penumpukan ke kendaraan ataupun sebaliknya [3].

*Receiving* merupakan aktivitas masuknya kontainer yang akan ditumpuk di gudang maupun lapangan penumpukan untuk dimuat di kapal dengan memvalidasi *Receiving Card*. Adapun beberapa tahapan kegiatan *gate in* dan *gate out* dari aktivitas *receiving*, antara lain adalah [9][10]:

- Petugas *gate* menerima kartu ekspor/muat/*receiving card* dari sopir truk yang membawa kontainer.
- Pada kartu ekspor/muat/*receiving card* memiliki detail informasi mengenai kontainer, hal ini akan diverifikasi oleh petugas apakah informasi sudah sesuai. Jika informasi tidak sesuai maka kartu akan dikembalikan kepada sopir truk.
- Ketika data terverifikasi, sopir akan menuju ke *Container Yard (CY)* untuk menyerahkan kontainer untuk diletakkan berdasarkan lokasi yang terdapat di kartu ekspor/muat/*receiving card*.
- Setelah itu sopir truk akan keluar melalui pintu *gate out*

*Delivery* merupakan proses keluarnya kontainer yang di bongkar dari kapal dan akan diserahkan kepada pelanggan/pemilik barang dengan memvalidasi Surat Penyerahan Peti Kemas (SP2). Berikut adalah beberapa tahapan kegiatan *gate in* dan *gate out* dari aktivitas *delivery*, antara lain adalah:

- Petugas *gate* menerima Surat Penyerahan Peti Kemas (SP2) dan melakukan verifikasi apakah SP2 sudah sesuai. Jika belum, SP2 akan dikembalikan kepada sopir truk.
- Saat terverifikasi oleh petugas *gate*, sopir truk akan menuju ke *Container Yard (CY)* untuk mengambil kontainer sesuai dengan lokasi yang terdapat di SP2.
- Setelah mengangkut kontainer, sopir truk menuju pintu *gate out* dan menunjukkan kembali SP2 kepada petugas *gate*, untuk diverifikasi kembali apakah kontainer yang diangkut sesuai dengan SP2.
- Jika sudah sesuai, kontainer dapat dibawa keluar.

#### D. Dart

*Dart* merupakan salah satu bahasa pemrograman bersifat *cross platform* yang dapat dijalankan pada sistem operasi yang berbeda, seperti *Windows, Linux, Unix, MacOS, Android, iOS*, dan lainnya. Bahasa pemrograman ini dikembangkan oleh *Google* dan disetujui sebagai standar oleh *Ecma* untuk membangun aplikasi berbasis *web, server, desktop, dan mobile*. *Dart* secara resmi diperkenalkan dalam Konferensi *GOTO* pada Oktober 2011 dan versi pertamanya dirilis pada November 2013[11].

#### E. Flutter

*Flutter* merupakan *framework open – source* yang dikembangkan oleh *Google* untuk membangun aplikasi *mobile*. *Framework* ini mendukung aplikasi dibangun untuk memiliki performa yang tinggi dan aplikasi dapat dipublikasikan pada *platform Android* maupun *iOS* dengan menggunakan *codebase* tunggal. *Flutter* menggunakan bahasa pemrograman *Dart* [12].

*Flutter* memiliki beberapa keunggulan yang tidak dimiliki oleh *framework* lainnya, antara lain adalah [13]:

- *Cross Platform* merupakan keunggulan *Flutter* dapat dijalankan di berbagai *platform* yang berbeda secara bersamaan seperti berbasis *mobile, website, dan desktop* yang menguntungkan seorang *developer* dalam membangun aplikasi, dikarenakan tidak perlu lagi untuk mempelajari bahasa pemrograman *native* dari setiap *platform*.
- *Hot Reload* merupakan fitur pada *flutter* yang berfungsi untuk memperbarui aplikasi saat adanya perubahan pada kode program. Dengan fitur ini mempercepat dan mempermudah proses pengembangan aplikasi, dimana *developer* dapat melihat perubahan aplikasi secara langsung dalam kurun waktu kurang dari 1 detik tanpa harus melakukan *rebuild*.
- *Beautiful UI* merupakan salah satu keunggulan *Flutter* yang menghasilkan *UI* yang menarik dengan menggunakan *widget* yang dapat dikustomisasi untuk menggambarkan tampilan berdasarkan konfigurasi dan *state* yang ada. Ada 2 *widget* yang digunakan untuk membuat *UI* pada *Flutter*, yaitu *Material Design* dan *Cupertino*.

#### F. REST API

*REST API* merupakan suatu metode yang digunakan untuk transaksi data antara klien dan server. Metode ini memfasilitasi klien untuk mengakses dan memanipulasi data yang tersimpan di server. Dengan metode ini, pengguna dapat mengakses aplikasi dari berbagai *platform*[14].

Penggunaan *REST API* didukung oleh *URL* dan *HTTP*. *HTTP* digunakan untuk menentukan *request* dan *response*. *Response* yang dihasilkan oleh *API server* adalah berupa *JSON* agar informasi bersifat sederhana dan dapat di analisa pada sisi client. Pada *REST API*, terdapat beberapa method dan tipe *request* untuk melakukan komunikasi data, antara lain adalah [15] :

- *GET* adalah suatu metode yang digunakan untuk memperoleh data dari server.
- *POST* adalah suatu metode untuk membuat data (*create*) yang dikirimkan dan disimpan kedalam database.
- *DELETE* adalah suatu metode untuk menghapus data di database.
- *PUT* adalah suatu metode untuk memperbarui data yang sudah ada di database.

### III. METODE PENGEMBANGAN SISTEM

Metode pengembangan sistem yang digunakan adalah metode *prototyping*. *Prototyping* merupakan metode pengembangan perangkat lunak menggunakan *prototype* dalam menggambarkan sistem secara umum. Penggunaan metode ini akan menyesuaikan kebutuhan dengan permintaan. Pengguna dan pemilik sistem terlibat dalam pengembangan sistem untuk memberikan feedback dalam memastikan kebutuhan aplikasi yang diharapkan dan memungkinkan perubahan yang cepat [16]. Tahapan pada metode *prototyping* dalam pengembangan aplikasi ini dapat dilihat pada Gambar 1.



Gambar 1. Metode *Prototyping*

Pada Gambar 1 menggambarkan beberapa tahapan dari metode *prototyping* mulai dari analisis kebutuhan, membangun *prototype*, evaluasi *prototype*, membangun atau mengkodekan sistem, pengujian sistem, evaluasi sistem, dan penggunaan sistem. Berikut adalah penjelasan dari setiap tahapan [17] :

1. *Analisis Kebutuhan*: Analisis kebutuhan merupakan tahapan untuk melakukan identifikasi dan mengumpulkan kebutuhan sistem. Pada tahapan ini dilakukan diskusi bersama pihak PT Persero Batam mengenai fitur dan kebutuhan dari sistem. Adapun hasil analisis kebutuhan sistem bahwa petugas gate membutuhkan sebuah sistem untuk dapat memverifikasi, merekam, dan memonitoring data kontainer dari RC dan SP2 yang memiliki kode QR yang digunakan di lapangan kerja.

2. *Membangun Prototype:* Membangun *prototype* merupakan tahapan membangun rancangan *prototype* dari sistem yang telah didiskusikan bersama pengguna atau pemilik sistem (PT Persero Batam). *Prototype* yang dibangun menggambarkan alur kerja dan visualisasi dari sistem. *Prototype* dibangun menggunakan *Figma*.

3. *Evaluasi Prototype:* Evaluasi *prototype* merupakan tahapan untuk meninjau dan menilai *prototype* apakah sudah sesuai dengan kebutuhan sistem. Jika *prototype* belum memenuhi kebutuhan ataupun kriteria yang diinginkan, maka pengembang dapat melakukan analisis kembali. Pada tahapan ini ditinjau oleh pengguna atau pemilik sistem, yaitu pihak PT Persero Batam.

4. *Membangun Sistem:* Membangun sistem merupakan tahapan metode *prototype* untuk mengimplementasikan *prototype* kedalam kode yang akan menjadi sebuah sistem menggunakan bahasa pemrograman. Pada sistem *Gate Portable* dibangun menggunakan bahasa pemrograman *Dart* dengan *framework Flutter*.

5. *Pengujian Sistem:* Pengujian sistem merupakan tahapan untuk melakukan uji coba pada sistem yang sudah dibangun. Pengujian dilakukan untuk mengetahui apakah sistem sudah dapat berjalan dengan semestinya. Pengujian sistem menggunakan metode *black box*.

6. *Evaluasi Sistem:* Evaluasi sistem merupakan tahapan untuk meninjau dan menilai apakah sistem sudah dapat atau belum digunakan. Jika sistem belum dapat digunakan, maka akan dilakukan tahap membangun sistem (mengkodakan sistem) dan pengujian sistem kembali. Evaluasi dilakukan oleh pengguna atau pemilik sistem, yaitu PT Persero Batam. Pada tahapan ini akan memperoleh *feedback* dari pemilik sistem.

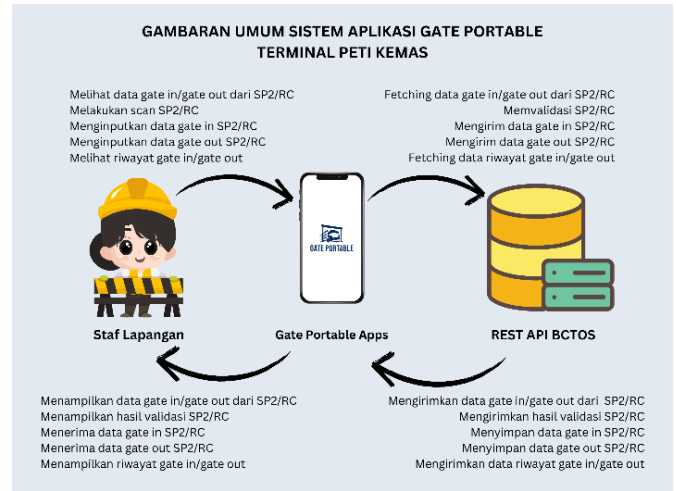
7. *Penggunaan Sistem:* Penggunaan sistem adalah tahapan akhir, yaitu menggunakan sistem yang telah diuji dan disetujui untuk digunakan. Sistem akan digunakan di *Gate Terminal Peti Kemas* untuk melakukan kegiatan *gate in* dan *gate out* pada *receiving* dan *delivery*.

**IV. HASIL DAN PEMBAHASAN**

**A. Gambaran Umum Sistem**

Aplikasi *Gate Portable* merupakan aplikasi untuk memfasilitasi proses verifikasi dan pendataan kegiatan truk *get in* dan *gate out* pada aktivitas *receiving* dan *delivery* di Terminal Peti Kemas Pelabuhan Batu Ampar PT Persero Batam. Aplikasi ini bertujuan untuk memudahkan petugas dalam memverifikasi, mencatat, dan monitoring kegiatan secara langsung di lapangan kerja. Pengguna dari aplikasi ini adalah Staf Lapangan. Aplikasi terintegrasi dengan sistem *Batam Container Terminal Operation (BCTOS)* melalui *REST API*. Fitur aplikasi ini meliputi verifikasi SP2 dan RC menggunakan pemindaian kode *QR*, pencatatan

data *gate in* dan *gate out*, serta monitoring persentase total kegiatan *gate in* dan *gate out* pada *receiving* dan *delivery*. Gambaran umum sistem dapat dilihat pada Gambar 2.



Gambar 2 Gambaran Umum Sistem

**B. Analisis Kebutuhan Sistem**

Berdasarkan hasil analisis dapat diidentifikasi kebutuhan sistem yang terdiri dari kebutuhan fungsional dan kebutuhan non – fungsional. Kebutuhan fungsional dan kebutuhan non – fungsional dapat dilihat pada Tabel 1 dan Tabel 2.

TABEL I  
TABEL KEBUTUHAN FUNGSIONAL SISTEM

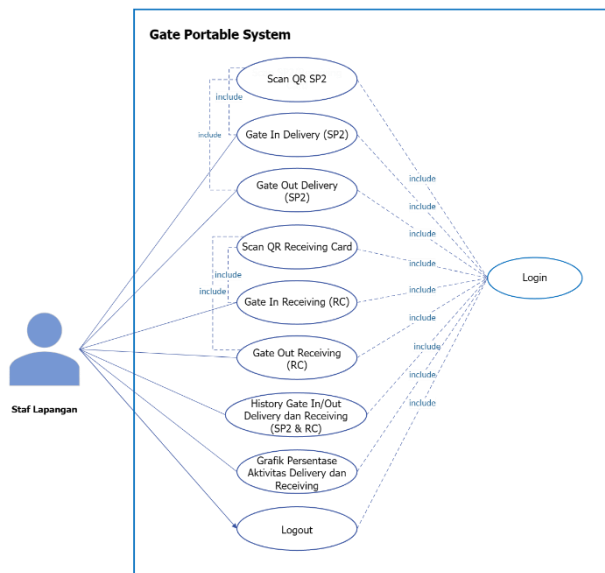
No	Kebutuhan Fungsional
F001	Staf dapat melakukan <i>login</i> menggunakan akun BCTOS
F002	Staf dapat melakukan pemindaian kode <i>QR SP2</i>
F003	Staf dapat melihat detail data SP2
F004	Staf dapat menyimpan data <i>gate in</i> menggunakan SP2
F005	Staf dapat menyimpan data <i>gate out</i> menggunakan SP2
F006	Staf dapat melihat riwayat data <i>gate in</i> dan <i>gate out</i> kegiatan <i>delivery</i>
F007	Staf dapat mencari data kontainer yang sudah <i>gate in</i> atau <i>gate out</i> kegiatan <i>delivery</i>
F008	Staf dapat melakukan pemindaian kode <i>QR RC</i>
F009	Staf dapat melihat detail data RC
F010	Staf dapat menyimpan data <i>gate in</i> menggunakan RC
F011	Staf dapat menyimpan data <i>gate out</i> menggunakan RC
F012	Staf dapat melihat riwayat data <i>gate in</i> dan <i>gate out</i> kegiatan <i>receiving</i>
F013	Staf dapat mencari data kontainer yang sudah <i>gate in</i> atau <i>gate out</i> kegiatan <i>receiving</i>
F014	Staf dapat melihat persentase dan total kegiatan <i>gate in</i> dan <i>gate out</i> pada <i>receiving</i> dan <i>delivery</i> dalam bentuk grafik

TABEL II  
TABEL KEBUTUHAN NON FUNGSIONAL SISTEM

No	Kebutuhan Non Fungsional
NF001	Aplikasi diakses menggunakan perangkat <i>mobile</i> dengan sistem operasi <i>Android</i>
NF002	Aplikasi memiliki tampilan antarmuka yang mudah dimengerti oleh Staf
NF003	Aplikasi dapat mengakses kamera dari perangkat yang digunakan
NF004	Aplikasi dapat diakses menggunakan jaringan internet

C. Use Case Diagram

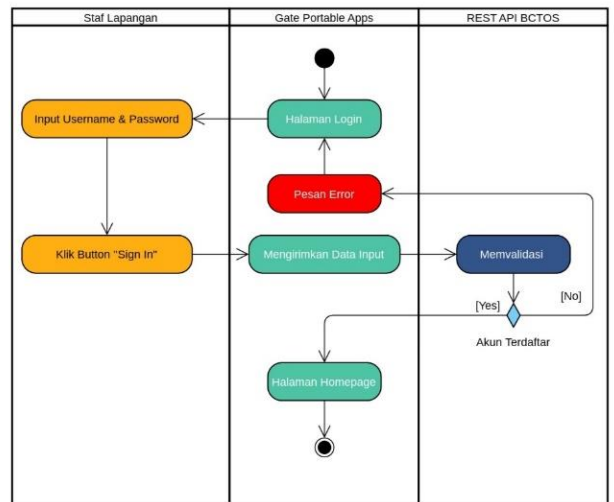
Use case diagram merupakan diagram untuk menggambarkan hubungan antara aktor dengan fungsional pada sistem [18]. Aktor dari aplikasi *Gate Portable* adalah Staf Lapangan. Use case pada sistem ini dapat dilihat pada Gambar 3.



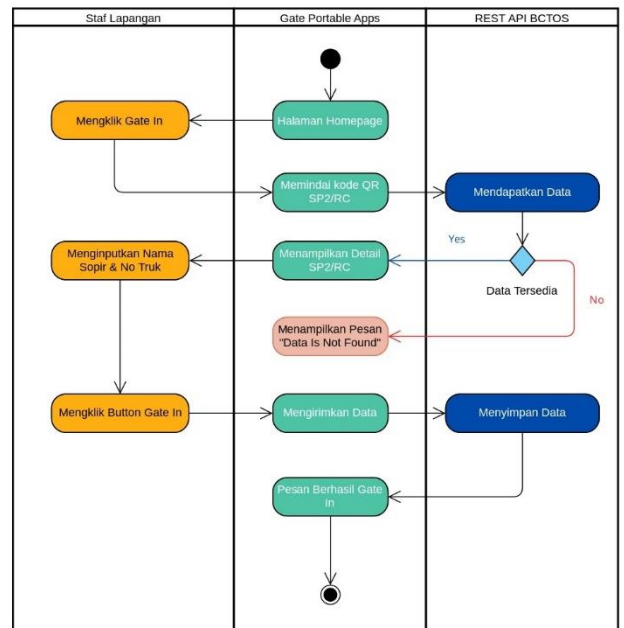
Gambar 3 Use Case Diagram

D. Activity Diagram

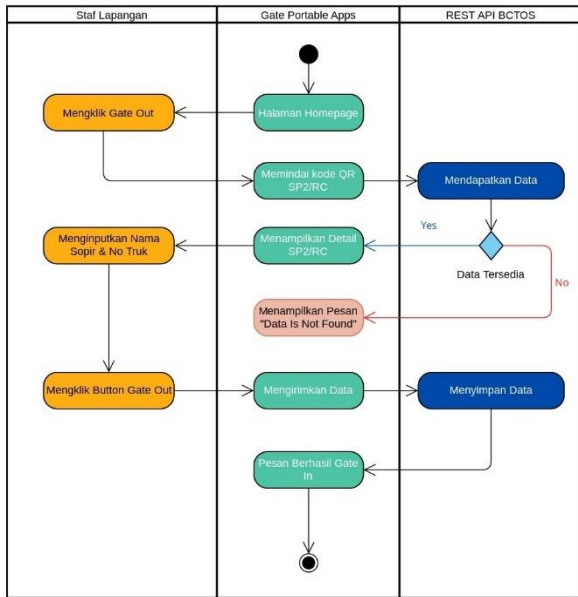
Activity diagram dirancang untuk menggambarkan interaksi pengguna dengan setiap fungsi yang ada [19]. Activity diagram aplikasi *Gate Portable* dapat dilihat pada Gambar 3, 4, 5, 7, 8, dan 9.



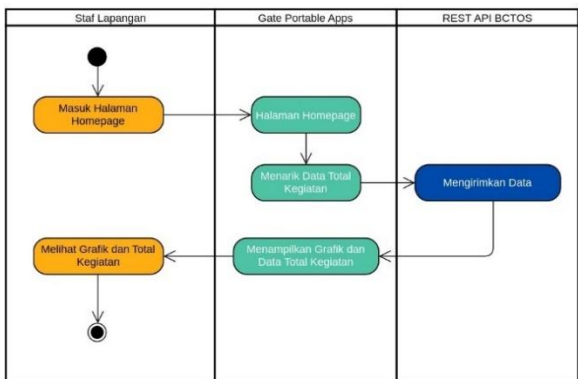
Gambar 4 Activity Diagram Login



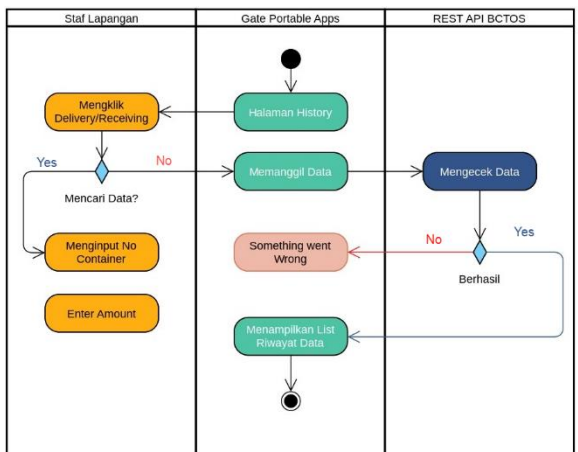
Gambar 5 Activity Diagram Gate In



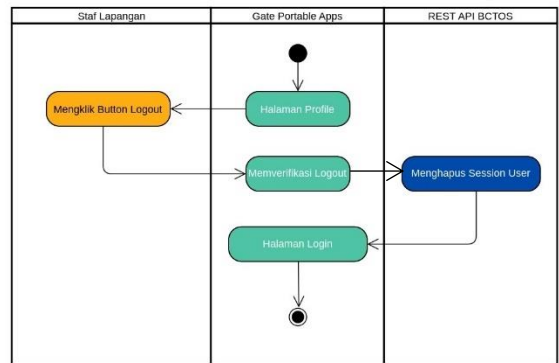
Gambar 6 Activity Diagram Gate Out



Gambar 7 Activity Diagram Grafik dan Total Kegiatan



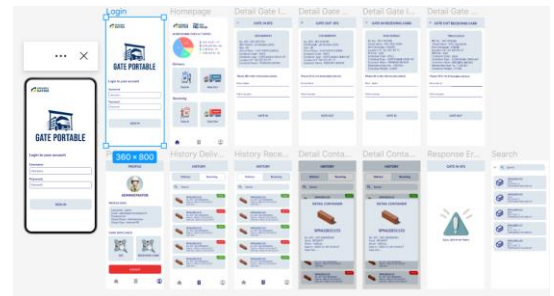
Gambar 8 Activity Diagram History Kegiatan



Gambar 9 Activity Diagram Logout

E. Rancangan Desain Prototyping Sistem

Desain *prototyping* sistem digunakan untuk menggambarkan alur dan visualisasi dari tampilan antarmuka sistem. *Prototyping* didesain menggunakan *software Figma*. Pada perancangan ini terjadi beberapa kali perubahan desain setelah dilakukannya evaluasi. desain tampilan antarmuka sistem terdiri dari halaman *login*, *homepage*, halaman *history*, *profile*, halaman setiap *gate in* dan *gate out* pada SP2 dan RC. Rancangan ini dapat dilihat pada Gambar 10.



Gambar 10 Rancangan Desain Prototyping Sistem

F. Hasil Implementasi

Aplikasi ini memiliki beberapa tampilan antarmuka untuk mendukung *experience* pengguna pada sistem. Desain *prototyping* yang dibangun akan diimplementasikan kedalam kode yang menjadi sebuah sistem menggunakan bahasa pemrograman *Dart* dan *framework Flutter*. Data didapatkan melalui pengintegrasian *REST API* sistem *BCTOS*.

Saat pengguna pertama kali membuka aplikasi akan tampil halaman *login*. Pada halaman ini pengguna akan menginputkan *username* dan *password* yang terdaftar di sistem *BCTOS* untuk dapat masuk ke aplikasi. Tampilan halaman *login* dapat dilihat pada Gambar 11.



Gambar 11 Halaman Login

Setelah berhasil login, pengguna akan masuk ke halaman homepage seperti pada Gambar 12. Pada halaman ini menampilkan grafik persentase kegiatan, fitur scan gate in dan gate out pada delivery dan receiving.



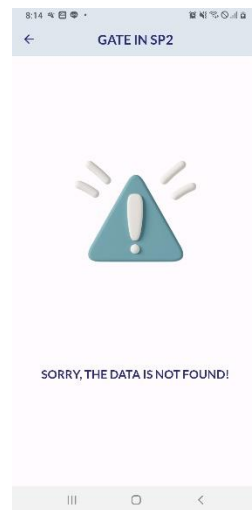
Gambar 12 Homepage

Saat terjadi kegiatan delivery maupun receiving, pengguna dapat mengklik gate in saat truk masuk dan mengklik gate out ketika truk keluar. Gambar 13 menggambarkan saat fitur di klik maka akan menampilkan scanner kode QR untuk memverifikasi SP2 atau RC.



Gambar 13 Tampilan Scanner QR Code

Ketika SP2 atau RC terverifikasi maka akan tampil detail datanya, jika tidak maka akan menampilkan pesan bahwa data tidak ditemukan seperti pada Gambar 14.



Gambar 14 Tampilan Data Tidak Ditemukan

Pada fitur gate in di delivery, saat SP2 terverifikasi maka akan muncul tampilan detail data SP2 untuk dilakukan gate in. Pengguna harus memasukkan nama sopir dan nomor polisi truk. SP2 hanya akan terverifikasi ketika kontainer sudah berstatus 07 di sistem BCTOS seperti pada Gambar 15.

Gambar 15 Tampilan Gate In SP2

Kemudian pada fitur *gate out* di *delivery*, saat SP2 terverifikasi maka akan muncul tampilan detail data SP2 untuk melakukan *gate out*. SP2 hanya akan terverifikasi ketika kontainer sudah memiliki status *unplacement* di sistem BCTOS. Pengguna juga harus tetap menginputkan nama sopir dan nomor polisi truk. Halaman ini dapat dilihat pada Gambar 16.

Gambar 16 Tampilan Gate Out SP2

Pada fitur *gate in* di *receiving*, saat RC terverifikasi maka akan muncul tampilan detail data RC untuk melakukan *gate in*. pengguna harus menginputkan nama sopir dan nomor polisi truk. RC hanya akan terverifikasi ketika status kontainer sudah berstatus 58 sistem BCTOS seperti pada Gambar 17.

Gambar 17 Tampilan Gate In RC

Kemudian pada fitur *gate out* di *receiving*, saat RC terverifikasi maka akan muncul tampilan detail data RC untuk melakukan *gate out*. RC hanya akan terverifikasi ketika kontainer sudah memiliki status *placement* di sistem BCTOS. Pengguna juga harus tetap menginputkan nama sopir dan nomor polisi truk. Halaman ini dapat dilihat pada Gambar 18.

Gambar 18 Tampilan Gate Out RC

Pengguna dapat melihat riwayat kegiatan *delivery* dan *receiving* saat *gate in* dan *gate out*. Pada halaman ini pengguna dapat mengecek kontainer yang sudah memiliki status *gate in* atau *gate out* seperti Gambar 19.



Gambar 19 Tampilan *History*



Gambar 21 Tampilan *Profile*

Selain itu pengguna juga dapat melihat detail data dari kontainer yang sudah *gate in* atau *gate out*. Pada detail data tersebut menampilkan nomor SP2 atau RC, nomor truk, nama sopir, waktu *gate in*, dan waktu *gate out*. Tampilan ini dapat dilihat pada Gambar 20



Gambar 20 Tampilan Detail *History Container*

Pada halaman *profile*, terdapat data dari pengguna dan dapat keluar dari aplikasi dengan mengklik tombol *logout* seperti pada Gambar 21. Setelah berhasil keluar, pengguna akan kembali ke halaman *login*.

G. Hasil Uji Coba

Tahap pengujian aplikasi menggunakan metode *black box* yang berfokus pada pengujian fungsionalitas pada aplikasi agar berjalan dengan baik. Pengujian dilakukan secara langsung oleh Tim IT Persero Batam dan Staf Lapangan. Hasil pengujian aplikasi dapat dilihat pada Tabel III

TABEL III  
TABEL PENGUJIAN SISTEM

Fitur	Skenario	Output	Hasil Pengujian
Login	Login dengan memasukkan <i>username</i> dan <i>password</i> yang terdaftar di BCTOS	Berhasil <i>login</i> dan menampilkan halaman <i>homepage</i>	Valid
	Login dengan memasukkan <i>username</i> dan <i>password</i> yang tidak terdaftar di BCTOS	Gagal <i>login</i> dan menampilkan pesan <i>error</i>	Valid
	Login dengan mengosongkan <i>username</i> dan <i>password</i>	Gagal <i>login</i> dan menampilkan pesan <i>error</i>	Valid
Verifikasi <i>Gate In Delivery</i>	Memindai kode QR yang benar	Menampilkan detail data kontainer SP2 dan <i>form Gate In</i>	Valid
	Memindai kode QR yang salah	Data tidak ditemukan	Valid

Verifikasi <i>Gate Out Delivery</i>	Memindai kode QR yang benar	Menampilkan detail data kontainer SP2 dan <i>form Gate Out</i>	Valid
	Memindai kode QR yang salah	Data tidak ditemukan	Valid
<i>Gate In Delivery</i>	Menginputkan nama sopir dan nomor polisi truk	Berhasil <i>Gate In Delivery</i>	Valid
<i>Gate Out Delivery</i>	Menginputkan nama sopir dan nomor polisi truk	Berhasil <i>Gate Out Delivery</i>	Valid
Verifikasi <i>Gate In Receiving</i>	Memindai kode QR yang benar	Menampilkan detail data kontainer RC dan <i>form Gate In</i>	Valid
	Memindai kode QR yang salah	Data tidak ditemukan	Valid
Verifikasi <i>Gate Out Delivery</i>	Memindai kode QR yang benar	Menampilkan detail data kontainer RC dan <i>form Gate Out</i>	Valid
	Memindai kode QR yang salah	Data tidak ditemukan	Valid
<i>Gate In Receiving</i>	Menginputkan nama sopir dan nomor polisi truk	Berhasil <i>Gate In Receiving</i>	Valid
<i>Gate Out Receiving</i>	Menginputkan nama sopir dan nomor polisi truk	Berhasil <i>Gate Out Receiving</i>	Valid
Monitoring Kegiatan	Melihat grafik persentase dan data total kegiatan <i>delivery</i> dan <i>receiving</i> saat <i>gate in</i> dan <i>gate out</i> di <i>homepage</i>	Menampilkan grafik persentase dan data total kegiatan <i>delivery</i> dan <i>receiving</i> saat <i>gate in</i> dan <i>gate out</i> di <i>homepage</i>	Valid
Riwayat Kegiatan	Mengklik tab <i>delivery</i>	Menampilkan list riwayat data kontainer yang <i>gate in</i> dan <i>gate out delivery</i>	Valid
	Mengklik tab <i>receiving</i>	Menampilkan list riwayat data kontainer yang <i>gate in</i>	Valid

		dan <i>gate out receiving</i>	
	Mengklik <i>card data</i>	Menampilkan detail data kontainer sesuai dengan statusnya	Valid
	Menginputkan nomor kontainer untuk mencari data kontainer	Menampilkan data kontainer	Valid
<i>Logout</i>	Mengklik <i>button logout</i> di halaman profile	Berhasil <i>logout</i> dan kembali ke halaman <i>login</i>	Valid

Berdasarkan hasil pengujian pada Tabel III menunjukkan bahwa aplikasi sudah berjalan dengan baik sesuai dengan fungsionalitas yang diharapkan.

## V. KESIMPULAN

Berdasarkan dari hasil pembahasan di atas, pengembangan aplikasi *Gate Portable* berbasis *mobile* menggunakan bahasa pemrograman *Dart* dan *framework Flutter* dengan integrasi pada sistem *BCTOS*, untuk memudahkan proses verifikasi dan pencatatan data dalam kegiatan *delivery* dan *receiving* di Terminal Peti Kemas Pelabuhan Batu Ampar oleh PT Persero Batam. Pengembangan aplikasi menggunakan metode *prototyping* yang melibatkan pengguna dan pemilik sistem untuk memastikan keberhasilan implementasinya. Dengan memanfaatkan kamera untuk memindai kode QR, Staf Lapangan dapat langsung memverifikasi dan mencatat data secara langsung di lapangan menggunakan *smartphone* maupun perangkat *mobile* lainnya. Dari hasil pengujian, seluruh fitur aplikasi sudah berjalan sesuai dengan fungsinya dan dapat digunakan di lapangan kerja Terminal Peti Kemas Pelabuhan Batu Ampar.

## DAFTAR PUSTAKA

- [1] Herlambang, R. R. A., & Zuhri, Z. (2023). Penggunaan Firebase Analytics pada Pengembangan Aplikasi Mobile I'm UII dengan Framework Flutter. *AUTOMATA*, 4(2).
- [2] Anggraini, D., Ramdlani, S., & Pamungkas, S. T. (2015). *Pola Spasial Pelabuhan Batu Ampar, Batam* (Doctoral dissertation, Brawijaya University).
- [3] Dandi, I., Setiawan, F., Ramdhan, L., & Lalu, H. P. (2019). Proses dan pelayanan pengiriman dan penerimaan di pelabuhan. Proses dan Pelayanan Pengiriman dan Penerimaan Di Pelabuhan (1st ed.). AKMI, Suaka Bahari Cirebon.
- [4] Firgiawan, A. L. (2020). Pelayanan Pendistribusian Container Empty di Depo PT Spil Cabang Jakarta Dengan Menggunakan Trucking Order Management System (Toms) Di Pelabuhan Tanjung Priok Jakarta. *KARYA TULIS*.
- [5] Nyayu, Y. I. (2018). *Aplikasi Container Location Menggunakan Barcode Scanner Berbasis Android Pada Pt. Pelabuhan Indonesia*

- li (Persero) Cabang Palembang Dengan Metode Agile* (Doctoral dissertation, POLITEKNIK NEGERI SRIWIJAYA).
- [6] Hanifah, N. (2022). PERANCANGAN SISTEM APLIKASI GATE MOBILE BERBASIS ANDROID DI TPS DAN TPB (STUDI KASUS: KPPBC TMP A BEKASI). *Eclipse Jurnal Sistem Informasi*, 1(1).
- [7] Haqi, B., Sinaga, J., & Setiawan, H. S. (2019, August). Aplikasi Mobile Scan Parkir Berbasis QR Code Pada CO-Working Space Ruang Hampa. In *Prosiding Seminar Sains Nasional dan Teknologi* (Vol. 1, No. 1).
- [8] Dicoding. (2023, June 23). *3 Jenis Aplikasi yang Harus Kamu Ketahui*. Dicoding. Diakses dari 3 Jenis Aplikasi yang Harus Kamu Ketahui - Dicoding Blog
- [9] Tedy Herdian. (2023, December 3). Alur Logistik Peti Kemas (Bagian 1 dari 2 tulisan). Supply Chain Indonesia. <https://supplychainindonesia.com/alur-logistik-peti-kemas-bagian-1-dari-2-tulisan/Tedy>
- [10] Herdian. (2023, December 6). Alur Logistik Peti Kemas (Bagian 2 dari 2 tulisan). Supply Chain Indonesia. <https://supplychainindonesia.com/alur-logistik-peti-kemas-bagian-2-dari-2-tulisan/>
- [11] Fauziah Nabilah. (2022, February). *Bahasa Pemrograman Dart – Pengertian, Sejarah, Fitur, dan Contoh Aplikasi*. Diakses dari <https://academy.alterra.id/blog/bahasa-pemrograman-dart/>
- [12] Muslim. (2022). Implementasi Framework Flutter pada Sistem Perpustakaan Mesjid. *Jurnal Komputer dan Aplikasi*, Vol 10(01), 46–59
- [13] Setiawan, P. R., Ramadhan, R. A., & Labellapansa, A. (2022). Pelatihan Pemrograman Flutter. *Jurnal Pengabdian Masyarakat dan Penerapan Ilmu Pengetahuan*, 3(1), 22-27.
- [14] Renardi, F., Faizah, N. M., & Koryanto, L. (2023). Reimbursement Application Using REST API Methods and Android-Based Flutter Framework (Case Study: PT. Protonema). *Journal Mobile Technologies (JMS)*, 1(1), 44-52.
- [15] Affrianto, M. K. N. F., & Cahyono, A. B. (2022). Implementasi REST API Pada Fitur Rencana Strategis Dalam Aplikasi website E-Government (Studi Kasus CV. Atsoft Teknologi). *AUTOMATA*, 3(2).
- [16] Mulyani, S. (2017). *Metode Analisis dan perancangan sistem*. Abdi Sistematika.
- [17] Rizky Dimas. (2019, January 15). *Mengenal Prototyping*. Diakses dari <https://medium.com/dot-intern/sdlc-metode-prototype-8f50322b14bf>
- [18] Erzed, N., Anwar, N., Widodo, A. M., Prasetyo, E., & Juman, K. K. (2022). Implementasi Flutter Pada Aplikasi Presensi Karyawan Berbasis Mobile. *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 6(3).
- [19] Sihotang, J. I., & Limbong, R. Y. (2020). Perancangan Sistem Transaksi Non-Tunai Berbasis QR Code di Platform Android. *TeIka*, 10(1), 25-37.