

**PENGEMBANGAN SISTEM FRONT-END PADA  
PMTCINVENTORY BERBASIS MOBILE  
DAN MENGGUNAKAN  
FRAMEWORK  
FLUTTER**



Disusun Oleh:

Nama : Muhammad Khalifah Milano  
NIM : 19523217

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA  
2023**

**HALAMAN PENGESAHAN DOSEN PEMBIMBING**  
**PENGEMBANGAN SISTEM FRONT-END PADA**  
**PMTCINVENTORY BERBASIS MOBILE**  
**DAN MENGGUNAKAN**  
**FRAMEWORK**  
**FLUTTER**

**TUGAS AKHIR JALUR MAGANG**



الجمعة الاستاذة الاندو

Yogyakarta, 31 Januari 2024

Pembimbing,

( Izzati Muhimmah, S.T., M.Sc., Ph.D)

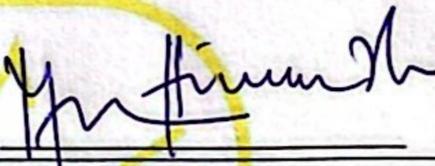
**HALAMAN PENGESAHAN DOSEN PENGUJI**

**PENGEMBANGAN SISTEM FRONT-END PADA  
PMTCINVENTORY BERBASIS MOBILE  
DAN MENGGUNAKAN  
FRAMEWORK  
FLUTTER**

**TUGAS AKHIR JALUR MAGANG**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 8 Januari 2024

Tim Penguji	
Izzati Muhimmah, S.T., M.Sc., Ph.D.	
Anggota 1	
Dr. Novi Setiani, S.T., M.T.	
Anggota 2	
Dr. Sri Kusumadewi, S.Si., M.T.	

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

  
(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

## HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Khalifah Milano  
NIM : 19523217

Tugas akhir dengan judul:

### **PENGEMBANGAN SISTEM FRONT-END PADA PMTCINVENTORY BERBASIS MOBILE DAN MENGGUNAKAN FRAMEWORK FLUTTER**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apa pun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 31 Januari 2024

  
( Muhammad Khalifah Milano )

## **HALAMAN PERSEMBAHAN**

Tugas akhir ini dengan tulus saya persembahkan untuk kedua orang tua saya yang telah memberikan dukungan kepada saya baik secara materi maupun immateri. Tugas akhir ini juga saya persembahkan untuk sahabat terdekat yang saya cintai, Hasna Afifah yang telah menemani saya melalui halangan dan rintangan tidak hanya ketika menjalani masa tugas akhir, namun juga selama masa perkuliahan dan di kehidupan saya. Selain itu, saya juga mau berterimakasih kepada teman-teman saya yaitu Ilham Sabilillah, Amirul Umam Musalam, Yuka Dwi Ananta, Ryan Nugraha, Ananda Tri Putra yang telah menguatkan serta menyemangati saya selama masa pengerjaan tugas akhir. Tugas akhir ini juga saya persembahkan kepada diri saya sendiri yang mampu menyelesaikan tugas akhir walau banyak halangan serta rintangan.

## HALAMAN MOTO

*“I am 56. Just yesterday I was 20. Time flies. I think one day you will regret if you don't reach what you can reach. Consistency is what make difference between top player and top potential. You should demand more from yourself. Not me. But you”*

– Jose Mourinho

*“The saddest thing in life is wasted talent “*

– Lorenzo (Bronx Tale)

*“Waktu terbaik untuk menanam pohon adalah dua puluh tahun lalu. Waktu terbaik kedua adalah sekarang”*

– Pepatah China

## KATA PENGANTAR

Alhamdulillah rabbil 'aalamiin, Puji syukur atas kehadiran Allah SWT yang telah melimpahkan rahmat, hidayah, dan petunjuk-Nya sehingga penulis dapat menyelesaikan laporan akhir dengan judul “Pengembangan Sistem Front-End Pada PMTCInventory Berbasis Mobile dan Menggunakan Framework Flutter”. Tidak lupa sholawat serta salam kepada junjungan kita Nabi Muhammad SAW yang telah menjadi teladan dalam setiap perjuangan dan usaha.

Laporan ini disusun sebagai syarat kelulusan penjaluran magang pada Program Studi Informatika – Program Sarjana. Dalam pengerjaan laporan akhir, terdapat banyak dukungan, bimbingan, serta bantuan dari berbagai pihak. Oleh karena itu, penulis ingin mengutarakan rasa terimakasih penulis kepada :

1. Allah S.W.T karena atas karunianya penulis diberikan kesehatan, kekuatan sehingga bisa menyelesaikan laporan akhir
2. Kedua orang tua yang selalu memberikan dukungan serta do'a
3. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Ketua Jurusan Informatika dan Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. selaku Ketua Program Studi Informatika - Program Sarjana.
4. Ibu Izzati Muhimmah, S.T., M.Sc., Ph.D. selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk membimbing dan mengarahkan penulis sejak pelaksanaan kegiatan magang hingga selesai disusunnya laporan ini.
5. Seluruh pimpinan PT. Pertamina khususnya Pertamina Maritime Training Centre yang telah memberikan kesempatan dan pengalaman magang
6. Divisi IT Pertamina Maritime Training Centre terutama Mas Tama, Mas Dika yang telah membimbing, memberikan dukungan dan ilmu baru, serta membantu penulis akan banyak hal selama magang.
7. Para Dosen Program Studi Informatika - Program Sarjana yang telah memberikan bekal ilmu kepada penulis selama duduk di bangku perkuliahan.
8. Teman seperjuangan ketika magang yaitu Naufal, Kevin, Rafi, Hinggit, Dinda, Aji yang telah memberikan semangat dan mau membantu ketika magang
9. Sahabat yang saya cintai Hasna Afifah dan teman-teman saya Ilham Sabilillah, Amirul Umam Musalam, Yuka Dwi Ananta, Ryan Nugraha Cahyana, Ananda Tri Putra, Roy Andreas Cipta, Candra Rifa'I Putra yang telah memberikan dukungan kepada saya selama pengerjaan laporan akhir.

10. Pihak-pihak yang mungkin tidak saya sebut, terimakasih secara sadar atau tanpa disadari anda telah membantu saya.

Atas dukungan serta bantuan dari pihak-pihak yang telah disebutkan, penulis dapat menyelesaikan laporan akhir. Penulis menyadari bahwa laporan ini sangat jauh dari kesempurnaan. Oleh karena itu, penulis terbuka untuk berbagai bentuk saran dan kritik agar sempurnanya laporan ini. Penulis berharap laporan akhir ini dapat bermanfaat bagi siapapun terutama bagi pembaca. Terima kasih.

Yogyakarta, 31 Januari 2024

A handwritten signature in black ink, consisting of a large, stylized 'M' followed by a series of loops and a long horizontal stroke extending to the right.

( Muhammad Khalifah Milano )

## SARI

Pertamina Maritime Training Center merupakan salah satu bagian dari PT Pertamina (Persero) yang menyelenggarakan pendidikan dan pelatihan maritim yang berstandar nasional dan internasional bagi Masyarakat. Pada saat ini, Pertamina Maritime Training Centre sedang mengembangkan sebuah aplikasi berbasis *mobile* bernama PMTCInventory yang ditujukan untuk karyawan bagian lapangan untuk mempermudah proses pekerjaan mereka. Pada aplikasi ini terdapat beberapa fitur utama, yaitu fitur Home untuk mengolah inventaris perusahaan seperti memberi jadwal perawatan suatu alat, mengkategorikan alat tersebut, memberi keterangan perawatan apa yang diperlukan, penanggung jawab alat tersebut, dan gambar alat tersebut. Kemudian ada fitur Dashboard yang menampilkan *card* yang berisikan gambar alat, penanggung jawab, jadwal perawatan, dan nama perusahaan. Ada pula Detail Alat yang berfungsi agar pengguna mengisi perawatan yang telah diberikan pada fitur Home. Setelah itu ada fitur History yang berfungsi untuk melihat alat apa saja yang telah dilakukan perawatan. Fitur Login yang berfungsi hanya agar pengguna yang terotorisasi saja yang bisa mengakses aplikasi. Fitur Splash Screen yang berfungsi agar pengguna dapat menyadari sudah berada di aplikasi yang dituju sembari menunggu aplikasi dimulai. Halaman ini berada pada awalan ketika aplikasi baru dijalankan dan terakhir ada fitur Saya yang menunjukkan perawatan apa saja yang telah dilakukan oleh pengguna.

Aktivitas magang dilakukan di dalam divisi IT yang ada di Pertamina Maritime Training Centre yang salah satunya bertanggung jawab dalam pengembangan aplikasi PMTCInventory. Proyek ini dikembangkan menggunakan kerangka kerja Flutter dan menggunakan layanan Firebase sebagai *back-end* dan menerapkan metode Waterfall sebagai manajemen proyek, Penggunaan kerangka kerja Flutter dikarenakan kelebihanannya yaitu hanya memerlukan satu *codebase* untuk mengembangkan aplikasi ke berbagai platform seperti iOS, Android, Desktop, dan Website. Keunggulan ini menguntungkan pengembang karena pengembang tidak perlu mempelajari bahasa pemrograman lainnya seperti Swift atau objective-c untuk iOS atau kotlin atau Java untuk Java. Pada pengembangan PMTCInventory, penulis mengembangkan fitur Dashboard, Saya, Login, Detail Alat, Home, Login Page, dan History pada bagian front-end.

Aplikasi PMTCInventory diuji menggunakan *black box testing* dan dari hasil pengujian beberapa hasilnya adalah hanya pengguna yang terotorisasi saja yang dapat mengakses aplikasi dan fitur Dashboard menampilkan *card* dari hasil pengisian data pada menu Home

Kata kunci: aplikasi PMTCInventory, Firebase, Flutter, Waterfall.

## GLOSARIUM

Android	Sistem operasi yang dikembangkan oleh perusahaan Google
Black Box Testing	metode pengujian untuk memastikan apakah keseluruhan fungsionalitas telah melalui input dan output yang dihasilkan pada sistem dengan harapan dapat berjalan sesuai dengan harapan layanan BaaS (Backend as a service) yang dibentuk oleh perusahaan Google
Flutter Framework	Kerangka kerja bahasa pemrograman Dart Kerangka kerja yang digunakan oleh pengembang untuk mempermudah pengembangan perangkat lunak.
iOS	Sistem operasi yang dikembangkan oleh perusahaan Apple untuk perangkat iPhone dan iPad
User Experience	keseluruhan hal yang saling berkaitan dengan alur kerja produk yang mudah dirasakan, dipahami yang kemudian dapat mencapai tujuannya dalam menggunakan aplikasi interaktif
User Interface	Sebuah tampilan visual pada produk yang dapat menjembatani sistem dengan pengguna
Waterfall	Metode pengembangan perangkat lunak.
Wireframe	kerangka penataan item untuk pembuatan laman website atau aplikasi

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN.....	v
HALAMAN MOTO.....	vi
KATA PENGANTAR.....	vii
SARI.....	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Ruang Lingkup.....	2
1.3 Tujuan.....	3
1.4 Manfaat.....	3
1.5 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI DAN TINJAUAN PUSTAKA.....	5
2.1 PMTCInventory.....	5
2.2 User Interface (UI).....	6
2.3 User Experience (UX).....	6
2.4 Flutter.....	6
2.5 Dart.....	7
2.6 Android Studio.....	7
2.7 Vysor.....	8
2.8 Firebase.....	8
2.9 Black Box Testing.....	9
2.10 Tinjauan Pustaka.....	9
BAB III PELAKSANAAN MAGANG.....	12
3.1 Aktivitas Magang.....	12
BAB IV REFLEKSI PELAKSANAAN MAGANG.....	53
4.1 Relevansi Akademik.....	53
4.2 Pembelajaran Magang.....	54
4.3 Kendala, Hambatan, dan Tantangan Selama Magang.....	55
BAB V PENUTUP.....	56
5.1 Kesimpulan.....	56
5.2 Saran.....	56
DAFTAR PUSTAKA.....	57

**DAFTAR TABEL**

Tabel 2.1 Perbandingan penelitian terkait .....	10
Tabel 3.1 Tabel Aktivitas Magang.....	12
Tabel 3.2 Pengujian Black Box Testing .....	46

## DAFTAR GAMBAR

Gambar 2.1 Tampilan Dashboard .....	6
Gambar 2.2 Logo Flutter.....	7
Gambar 2.3 Logo Android Studio .....	8
Gambar 2.4 Logo Vysor .....	8
Gambar 2.5 Logo Firebase.....	9
Gambar 3.1 Usecase Sistem e-inventory pada aplikasi PMTCInventory.....	13
Gambar 3.2 Activity Diagram Halaman Login pada sistem e-inventory PMTCInventory .....	14
Gambar 3.3 Activity Diagram Proses Mendeskripsikan Perusahaan pemilik inventaris .....	15
Gambar 3.4 Activity Diagram Mengkategorikan Alat .....	16
Gambar 3.5 Activity Diagram Memilih Penanggung Jawab Inventaris .....	16
Gambar 3.6 Activity Diagram Penjadwalan .....	17
Gambar 3.7 Activity Diagram Mengunggah Gambar Inventaris.....	18
Gambar 3.8 Activity Diagram Memberi Keterangan Perawatan.....	18
Gambar 3.9 Activity Diagram Melihat Historis Inventaris .....	19
Gambar 3.10 Activity Diagram Melaporkan Inventaris yang Harus dilakukan perawatan.....	19
Gambar 3.11 Activity Diagram Mengisi Hasil dari Pemeriksaan .....	20
Gambar 3.12 Tampilan Codelabs Pembelajaran Kerangka Kerja Flutter.....	21
Gambar 3.13 Tampilan Pembelajaran Layanan Firebase .....	21
Gambar 3.14 Penggunaan Figma.....	22
Gambar 3.15 Low-Fidelity Splash Screen .....	23
Gambar 3.16 Low-Fidelity Login .....	23
Gambar 3.17 6 Low-Fidelity Detail Alat.....	24
Gambar 3.18 Low-Fidelity Tambah Alat.....	24
Gambar 3.19 Low-Fidelity Saya.....	25
Gambar 3.20 Low-Fidelity History.....	25
Gambar 3.21 Alur Fitur atau Menu Dashboard .....	26
Gambar 3.22 High-Fidelity Splash Screen .....	27
Gambar 3.23 High-Fidelity Login .....	27
Gambar 3.24 High-Fidelity Dashboard.....	28
Gambar 3.25 High-Fidelity Detail Alat .....	28
Gambar 3.26 High-Fidelity Saya .....	29
Gambar 3.27 High-Fidelity History .....	29

Gambar 3.28 High-Fidelity Tambah.....	30
Gambar 3.29 High-Fidelity Alur Dashboard .....	30
Gambar 3.30 Tampilan Halaman Splash Screen .....	31
Gambar 3.31 implementasi dari pengembangan splash screen .....	32
Gambar 3.32 Tampilan Halaman Login .....	34
Gambar 3.33 Tampilan Halaman Login Apabila Username dan Password .....	34
Gambar 3.34 Penggunaan library firebase authentication .....	35
Gambar 3.35 Tampilan Isi Navbar Dashboard .....	36
Gambar 3.36 penggalan kode untuk membangun Bottom Navbar untuk Dashboard, Home, History, Saya.....	37
Gambar 3.37 Tampilan Isi Dari Navbar Home.....	38
Gambar 3.38 Tampilan Isi Dari Navbar Home.....	39
Gambar 3.39 Tampilan Penggunaan Date Picker .....	40
Gambar 3.40 Penggalan dari kode untuk dibuatnya date picker .....	41
Gambar 3.41 Tampilan dari isi navbar History .....	42
Gambar 3.42 Tampilan dari isi navbar Saya.....	43
Gambar 3.43 Implementasi firebase authentication.....	44
Gambar 3.44 Tampilan dari Detail Alat .....	45
Gambar 3.45 Tampilan dari Detail Alat .....	45

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Perkembangan teknologi informasi dapat memberikan kemudahan untuk mendapatkan suatu layanan tertentu. Seiring berkembangnya pembangunan perusahaan-perusahaan, banyak juga perusahaan yang memanfaatkan teknologi informasi untuk menunjang produktivitas kerja. Dengan berkembangnya teknologi informasi, memunculkan berbagai sistem informasi seperti sistem informasi management, pelaporan kerja, pemasukan data kerja dan sebagainya. Namun, tidak semua perusahaan memiliki sistem informasi. Maka dari itu Pertamina Training Centre menginginkan sebuah sistem yang dapat membantu para karyawan dalam menjalankan setiap tugasnya.

Pertamina Maritime Training Centre merupakan bagian dari PT. Pertamina (Persero). Perusahaan ini merupakan sebuah perusahaan BUMN. Pertamina Maritime Training Centre merupakan perusahaan yang bergerak di bidang sertifikasi untuk pelaut. Pertamina Maritime Training Centre berlokasi di Jl. Pemuda No.44, RW.4, Jati, Kec. Pulo Gadung, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta. Pertamina Maritime Training Centre saat ini memiliki kurang lebih lima puluh karyawan di seluruh wilayah Indonesia. Misi Pertamina Maritime Training Centre adalah menyelenggarakan pendidikan dan pelatihan maritim yang berstandar nasional dan internasional, menghasilkan sumber daya manusia yang handal dan profesional di bidang maritim berdasarkan tata nilai Pertamina, membangun kerjasama dengan institusi pendidikan dan pelatihan yang terkemuka, meningkatkan penelitian dan pengembangan keilmuan untuk kemajuan industri maritime

Pada Pertamina Maritime Training Centre memiliki divisi bernama OPITO yang bertugas untuk melakukan perawatan inventaris perusahaan. Divisi ini masih menggunakan sistem konvensional, seperti menggunakan Microsoft Word dan Microsoft Excel untuk melakukan pengelolaan data hasil dari pengamatan inventaris perusahaan. Sistem konvensional tersebut dirasa tidak efektif karena terdapat kendala yang terjadi. Di antaranya adalah penyusunan data tidak terorganisir sehingga menyulitkan pengguna untuk melakukan perawatan inventaris. Akibat dari tidak terorganisirnya data, mengakibatkan perawatan berkala alat sering terlewat dari jadwal yang sudah ditetapkan. Oleh karena itu, Pertamina Maritime Training Centre mencoba untuk mencari solusi untuk memecahkan permasalahan tersebut

dengan dibuatnya aplikasi E-Inventory berbasis mobile. Aplikasi tersebut bernama PMTCInventory.

Aplikasi PMTCInventory merupakan aplikasi yang dirancang untuk mempermudah karyawan lapangan dalam mengelola inventaris perusahaan. Aplikasi ini digunakan oleh karyawan lapangan untuk menjalankan pekerjaannya seperti merawat inventaris perusahaan dan mengganti inventaris yang harus diganti. Harapan terciptanya aplikasi ini adalah dapat membantu karyawan dalam menjalankan pekerjaannya sehingga terciptanya perawatan inventaris yang efisien.

Pada pembuatan aplikasi PMTCInventory, penulis melakukan pengembangan dari sisi front end. Pengembangan front end dilakukan dengan menggunakan kerangka kerja Flutter dan berbagai macam library sebagai tambahan untuk membangun antarmuka yang sesuai dengan kebutuhan user.

Pengembangan front end dilakukan mengikuti hasil dari UI/UX (user interface/user experience) berupa bentuk prototipe yang sudah melalui tahapan pengujian dengan pengguna. Rancangan yang sudah dibuat tersebut kemudian diberikan kepada pengembang front end melalui website Figma yang nantinya akan diimplementasikan untuk pembuatan aplikasi PMTCInventory. Terdapat beberapa keunggulan dari kerangka kerangka kerja flutter, diantaranya adalah hanya diperlukan satu codebase untuk lintas platform seperti android, iOS, desktop, dan website. Hal ini menjadikan pengembangan aplikasi menjadi lebih cepat karena tidak perlu melakukan pengembangan masing-masing untuk tiap platform. Kerangka kerja Flutter pun terdokumentasi secara lengkap dan baik sehingga memudahkan pengembang untuk mempelajari kerangka kerja tersebut.

## 1.2 Ruang Lingkup

Pelaksanaan magang di Pertamina Maritime Training Centre berlangsung selama enam bulan terhitung dari 26 September 2022 hingga 21 Maret 2023. Pertamina Maritime Training Centre merupakan perusahaan yang bergerak di bidang sertifikasi untuk pelaut. Pertamina *Maritime* Training Centre berlokasi di Jl. Pemuda No.44, RW.4, Jati, Kec. Pulo Gadung, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta. Selama magang berlangsung peran penulis adalah mengembangkan aplikasi dari sisi *front end* menggunakan kerangka kerja Flutter. Selama magang berlangsung, penulis berkontribusi pada beberapa fitur, diantaranya sebagai berikut:

1. Melakukan konversi dari desain menjadi code pada menu *login*

2. Melakukan konversi dari desain menjadi code pada menu *splash screen*
3. Melakukan konversi dari desain menjadi code pada menu *dashboard*
4. Melakukan konversi dari desain menjadi code pada menu *history*
5. Melakukan konversi dari desain menjadi code pada menu *profile*
6. Melakukan konversi dari desain menjadi code pada menu tambah alat
7. Melakukan konversi dari desain menjadi code pada menu *detail alat*

Pada proyek pembuatan aplikasi PMTCInventory terdiri dari satu orang front end mobile developer, satu orang UI/UX designer, satu orang backend developer, dan disupervisi oleh manager.

### **1.3 Tujuan**

Tujuan dari pelaksanaan magang adalah untuk mengembangkan front-end aplikasi PMTCInventory menggunakan kerangka kerja Flutter serta menggunakan beberapa library sebagai pendukung kemudahan dalam proses pengembangannya

### **1.4 Manfaat**

Manfaat yang diperoleh dari pengembangan *front end* aplikasi PMTCInventory menggunakan kerangka kerja Flutter adalah sebagai berikut:

1. Memudahkan pekerja dalam mengelola data dan merawat inventaris
2. Data dari hasil pengelolaan inventaris menjadi tersusun secara rapih

### **1.5 Sistematika Penulisan**

Sistematika penulisan disusun untuk memudahkan dalam memahami laporan tugas akhir ini. Sistematika penulisan laporan tugas akhir sebagai berikut:

#### **Bab 1: Pendahuluan**

Dalam bab ini berisi latar belakang, ruang lingkup magang, tujuan, manfaat dan sistematika laporan.

#### **Bab 2: Landasan Teori**

Dalam bab ini berisi pembahasan dasar teori dan hasil ringkasan yang mendukung proses pembuatan laporan Tugas Akhir.

#### **Bab 3: Pelaksanaan Magang**

Pada bab ini memberikan aktivitas selama melaksanakan magang serta penjelasan mengenai proses pengembangan front end menggunakan kerangka kerja Flutter pada mobile

#### Bab 4: Refleksi Pelaksanaan Magang

Dalam bab ini berisi beberapa hal yang didapatkan selama kegiatan magang di Pertamina Maritime Training Center

#### Bab 5: Penutup

Pada bab ini berisi kesimpulan dari hasil pengembangan front end menggunakan kerangka kerja Flutter.

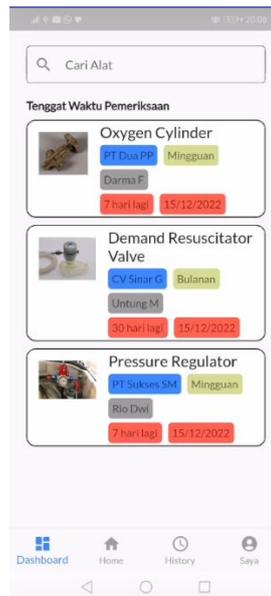
## **BAB II**

### **LANDASAN TEORI DAN TINJAUAN PUSTAKA**

#### **2.1 PMTCInventory**

PMTCInventory merupakan sistem inventaris berbasis mobile sebagai sebuah solusi untuk meningkatkan efisiensi aktivitas pencatatan inventaris di Perusahaan Pertamina Maritime Training Centre. Pembuatan aplikasi ini didasari oleh keluhan karyawan bagian lapangan yang mana harus merawat inventaris kantor secara berkala. Namun, tiap alat memiliki jadwal dan periode berbeda-beda dalam perawatan sehingga dalam penyusunan laporan serta perawatan menjadi tidak tersusun secara rapi. Dengan adanya aplikasi ini, diharapkan dapat memudahkan karyawan dalam mengelola inventaris dan membuat laporan. Pengguna dari sistem ini adalah divisi lapangan, manager, dan jajarannya.

Dalam Pengembangan front end PMTCInventory, digunakan kerangka kerja Flutter. PMTCInventory sendiri memiliki berbagai fitur, seperti: penambahan barang untuk menambahkan barang apa saja yang akan diperiksa nantinya, fitur detail alat pemeriksaan yang bertujuan untuk menampilkan detail dari barang yang akan diperiksa serta memberi update hasil dari pemeriksaan, fitur dashboard yang berfungsi untuk menampilkan kumpulan barang yang akan mencapai tenggat waktu, fitur riwayat berfungsi untuk melihat barang apa saja yang telah diperiksa, dan fitur profile yang bertujuan untuk mengetahui pengguna akun dan aktivitas terbaru pengguna. Gambar 2.1 adalah tampilan halaman dashboard pada aplikasi PMTCInventory



Gambar 2.1 Tampilan Dashboard

## 2.2 User Interface (UI)

User Interface merupakan sebuah tampilan visual pada produk yang dapat menjembatani sistem dengan pengguna (Haryuda et al., 2021)

User Interface merupakan tata letak yang dirancang untuk memberikan representasi visual oleh pengguna akhir atau pengguna ketika berinteraksi langsung dengan suatu aplikasi atau website. UI terdiri atas beberapa komponen, seperti pemilihan warna, bentuk elemen tombol, jenis font yang digunakan, penggunaan gambar, ikon, input field dan karakteristik aplikasi.

## 2.3 User Experience (UX)

User Experience merupakan aspek-aspek yang saling berkaitan dengan pengalaman pengguna ketika menggunakan sebuah layanan atau produk. UX adalah keseluruhan hal yang saling berkaitan dengan alur kerja produk yang mudah dirasakan, dipahami yang kemudian dapat mencapai tujuannya dalam menggunakan aplikasi interaktif (Razi et al., 2018).

## 2.4 Flutter

Flutter merupakan kerangka kerja yang bersifat *open source* yang dibuat oleh Google untuk membuat aplikasi yang dicompile secara native, multi-platform aplikasi dari hanya satu codebase. Kerangka kerja Flutter menggunakan bahasa pemrograman Dart. Kode Flutter dikompilasi ke kode mesin ARM atau Intel serta JavaScript. Kerangka kerja Flutter bersifat

multi-platform dari satu codebase yang memiliki arti pengembang hanya perlu satu codebase untuk men-deploy aplikasi untuk berbagai perangkat, seperti mobile, web, desktop, dan perangkat embedded.

Flutter merupakan Mobile App SDK (Software Development Kit) bertujuan untuk membuat aplikasi bergerak Android dan iOS dari satu codebase yang berperforma tinggi. Artinya, hanya perlu mempelajari Flutter untuk membangun aplikasi mobile untuk beberapa platform (Hendriawan et al., 2021).



Gambar 2.2 Logo Flutter

## 2.5 Dart

Dart merupakan bahasa pemrograman yang dioptimasi untuk client yang cepat untuk berbagai platform. Bahasa pemrograman Dart dibuat oleh perusahaan Google atau Alphabet. Dart dikembangkan dengan bahasa pemrograman yang dispesialisasi berdasarkan kebutuhan pengguna antarmuka.

Bahasa pemrograman Dart membuat proses pengembangan menjadi lebih produktif dengan adanya fitur hot reload. Hot Reload merupakan fitur yang mana pengembang dapat melihat perubahan antarmuka secara instan pada saat aplikasi sedang berjalan.

Bahasa pemrograman Dart dikompilasi ke mesin ARM & x64 untuk perangkat bergerak, desktop, dan backend. Bahasa pemrograman Dart dapat dikompilasi ke JavaScript untuk web. (Hendriawan et al., 2021)

## 2.6 Android Studio

Android Studio merupakan Integrated Development Environment (IDE) atau lingkungan pengembangan yang terintegrasi untuk mengembangkan aplikasi yang menggunakan operating system Android berdasarkan intellij IDEA.



Gambar 2.3 Logo Android Studio

## 2.7 Vysor

Vysor merupakan aplikasi yang memungkinkan pengguna untuk mengontrol perangkat berjalan yang menggunakan sistem operasi Android dan iOS langsung dari komputer, baik itu dari perangkat laptop maupun PC. Pengembang dari aplikasi ini adalah Koushik Dutta. (Ahmadi & Safwan, 2020)

Penggunaan aplikasi ini memudahkan pengguna dikarenakan pengembang tidak perlu menggunakan emulator pada IDE yang mana untuk menjalankannya memerlukan spesifikasi perangkat pengembang harus bagus. Dengan penggunaan aplikasi ini build aplikasi sangat cepat.



Gambar 2.4 Logo Vysor

## 2.8 Firebase

Firebase merupakan sebuah layanan BaaS (Backend as a service) yang dibentuk oleh perusahaan Google. Platform ini memiliki tujuan agar memudahkan pengembang dalam mengembangkan sistem yang menggunakan sumber REST API (Kurniawan et al., 2021). Layanan firebase bisa digunakan oleh berbagai platform seperti website, iOS, Desktop, Flutter. Pada umumnya, untuk berkomunikasi dengan server menggunakan REST API, pada proses

pembuatannya butuh waktu yang cukup lama untuk mengembangkannya dikarenakan beberapa faktor seperti kemudahan akses, kecepatan, dan keamanan. Faktor-faktor penghambat tersebut menjadikan salah satu alasan mengapa firebase dibentuk, yaitu mengurangi kegiatan pengembangan REST API sehingga pengembang menjadi lebih mudah dalam melakukan pengembangannya.



Gambar 2.5 Logo Firebase

## 2.9 Black Box Testing

Black box testing atau pengujian black box adalah metode pengujian untuk memastikan apakah keseluruhan fungsionalitas telah melalui input dan output yang dihasilkan pada sistem dengan harapan dapat berjalan sesuai dengan harapan. Pada proses pengujian black box, tidak memperhatikan kode program didalamnya untuk mengidentifikasi apakah adanya cacat pada sistem ketika pengembangan. (Aryasta, 2022)

## 2.10 Tinjauan Pustaka

Pada beberapa penelitian yang telah dilakukan sebelumnya terkait pengembangan front end berbasis perangkat bergerak menggunakan kerangka kerja Flutter. Terdapat penelitian yang berfokus untuk membahas pengembangan aplikasi e-commerce di PT Putra Sumber Abadi (Hendriawan et al., 2021). Hasil dari penelitian tersebut menjelaskan bahwa Flutter membantu dalam penerapan e-commerce untuk sistem penjualan. Selain itu, membuat aplikasi memiliki tampilan antarmuka yang user friendly sesuai dengan kebutuhan yang diinginkan oleh pengguna. Di dalam penelitian ini juga tercantum bahwa kerangka kerja Flutter memudahkan pembuatan fitur di e-commerce tersebut, seperti: memudahkan pembeli dalam membeli barang, memudahkan dalam pengelolaan barang, mendapatkan informasi secara realtime.

Pada penelitian (Herdiansah et al., 2022), membahas penggunaan kerangka kerja Flutter pada pengembangan aplikasi mobile e-bisnis penyediaan bahan baku bisnis catering. Permasalahan yang coba diselesaikan oleh peneliti adalah bagaimana cara mempermudah

proses manajemen penyediaan bahan baru untuk pembuatan makanan pada usaha katering. Para peneliti berpendapat bahwa permasalahan tersebut dapat diselesaikan dengan dibuatkannya manajemen siste informasi berbasis teknologi komputer. Hasil dari penelitan tersebut adalah aplikasi yang dihasilkan membantu UMKM dalam sektor kuliner dalam rangka mengelola rantai pasokan bahan baku. Selain itu penggunaan kerangka kerja Flutter yang berbasis Android sangat disukai oleh pengguna dikarenakan kerangka kerja Flutter memilih tampilan yang mudah dan digunakan dan menarik.

Pada penelitan (Firmansyah, 2018), peneliti melakukan pengembangan front-end aplikasi mobile e-recruitment menggunakan kerangka kerja Flutter. Pada penelitian ini, peneliti mencoba untuk mengurangi jumlah pengangguran dan mencoba untuk membangun karir untuk mahasiswa. Peneliti berpendapat bahwa untuk menentaskan masalah tersebut, salah satu solusinya adalah dengan membuat aplikasi jobseeker. Hasil dari penelitian tersebut adalah penggunaan Flutter memudahkan peneliti untuk menghasilkan frontend aplikasi mobile e-recruitment yang berdasarkan analisis kebutuhan sistem.

(Fauzi et al., 2021) Melakukan penelitan dengan mengembangkan aplikasi berbasis perangkat aplikasi bergerak tanggap bencana untuk Kabupaten Bengkulu dengan menggunakan kerangka kerja Flutter. Permasalahan yang diangkat oleh para peneliti adalah bagaimana cara menyelamatkan atau mengurangi korban dari bencana kebakaran hutan dan lahan di Kabupaten Bengkulu. Para peneliti berpendapat bahwa penyelesaian masalah tersebut adalah dengan dibuatnya sebuah perangkat aplikasi bergerak tanggap bencana untuk masyarakat sekitar. Peneliti berkesimpulan bahwa aplikasi yang dikembangkan dapat dibuat dengan cepat dikarenakan menggunakan kerangka kerja Flutter.

Tabel 2.1 Perbandingan penelitian terkait

No	Peneliti (Tahun)	Judul	Teknologi	Pengujian
1	Muhammad Hendriawan, Thomas Budiman, Verdi Yasin (2021)	Pengembangan Aplikasi E-Commerce di PT. Putra Sumber Abadi Menggunakan Flutter	Kerangka kerja Flutter dan sistem database PostgreSQL	Tidak dicantumkan
2	Arief Herdiansah, Desi Nurnaningsih, Hengki Rusdianto (2022)	Pemanfaatan Flutter Pada Pengembangan Aplikasi Mobile E-Bisnis Penyediaan Bahan Baku Bisnis Katering	Kerangka kerja Flutter	Black Box Testing
3	Didik Firmansyah (2019)	Pengembangan <i>Front-end</i> Aplikasi Mobile E-Recruitment Menggunakan Flutter	Kerangka kerja Flutter	Black Box Testing

4	Muhammad Fauzi, Agus Tedyana. Depandi Enda	Pengembangan Aplikasi Mobile Tanggap Bencana di Kabutapet Bengkulu Menggunakan Framework Flutter	Kerangka kerja Flutter	Compatibility Test
---	--	--	------------------------	--------------------

## BAB III PELAKSANAAN MAGANG

### 3.1 Aktivitas Magang

Kegiatan magang ini berlangsung selama 6 bulan yaitu dimulai sejak tanggal 26 September 2022 hingga 21 Maret 2023. Selama magang hingga bulan Februari ini penulis telah melakukan aktivitas-aktivitas di Pertamina Maritime Training Centre.

Tabel 3.1 Tabel Aktivitas Magang

No	Aktivitas	Durasi (waktu)
1	Onboarding secara online	Satu hari
2	Onboarding di kantor pusat	Satu hari
3	Onboarding di kantor tempat magang	Dua hari
4	Mengikuti penjelasan mengenai Jobdesc dan lingkungan kerja di perusahaan Pertamina Maritime Training Centre	Tiga hari
5	Wawancara calon pengguna aplikasi	Satu minggu
6	Mempelajari flutter dan firebase	Dua minggu
7	Pembuatan wireframe	Dua minggu
8	Pembuatan design aplikasi	Dua minggu
9	Pembuatan tampilan antarmuka aplikasi PMTCInventory menggunakan kerangka kerja Flutter	Tiga Bulan

#### 3.1.1 Wawancara Calon Pengguna Aplikasi

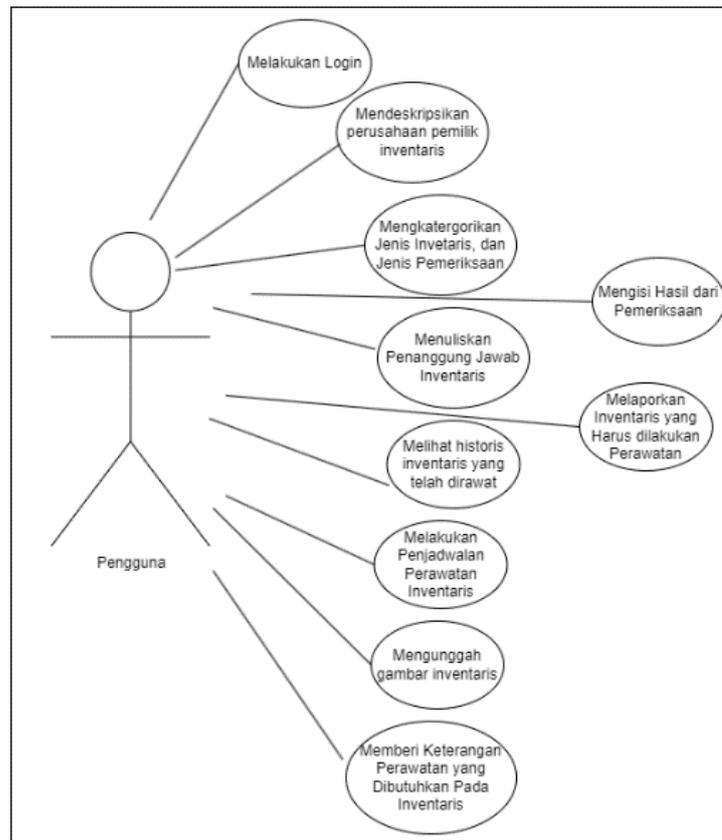
Ketika menjalani masa magang, penulis beserta rekan melakukan wawancara terhadap calon pengguna aplikasi. Hasil dari wawancara ini menghasilkan rincian proses bisnis pada divisi OPITO. Proses perawatan inventaris adalah sebagai berikut :

1. Hanya pekerja yang terotorisasi saja yang dapat melakukan perawatan dan pendataan.
2. Pekerja melihat data terlebih dahulu inventaris apa saja yang harus dilakukan perawatan pada hari tersebut dan melihat historis data tersebut.
3. Pekerja melakukan perawatan berkala pada inventaris tersebut.
4. Pekerja mengisi data seperti mendeskripsikan perusahaan tempat pembelian inventaris, mengkategorikan jenis inventaris dan jenis pemeriksaan, menuliskan

penanggung jawab inventaris, melakukan penjadwalan perawatan inventaris selanjutnya, mengunggah gambar inventaris, memberi keterangan perawatan apa saja yang dibutuhkan pada inventaris, mengisi hasil dari pemeriksaan

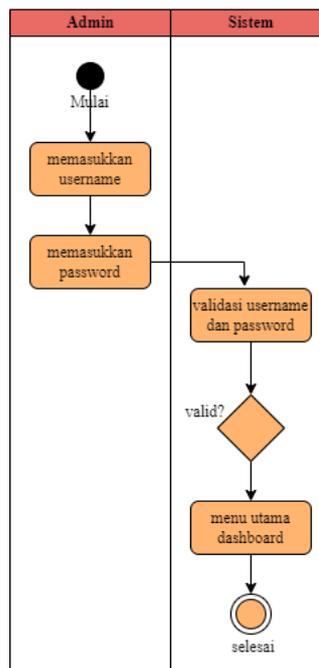
5. Setelah melakukan pengisian data, pekerja melaporkan data tersebut kepada supervisor.

Terdapat *usecase* dan *activity diagram* yang diraih dari hasil wawancara.



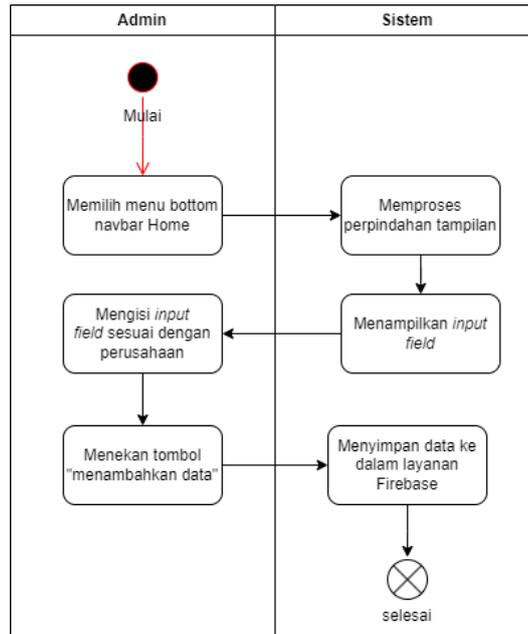
Gambar 3.1 Usecase Sistem e-inventory pada aplikasi PMTCInventory

Gambar 3.1 terdapat satu aktor yang berperan sebagai admin. Aktor admin tersebut merepresentasikan user yang dapat melakukan login melakukan penjadwalan perawatan inventaris, melaporkan inventaris apa saja yang harus dilakukan perawatan, mengkategorikan jenis inventaris perusahaan, melihat historis inventaris yang telah dirawat, memberi keterangan perawatan apa yang dibutuhkan pada inventaris.



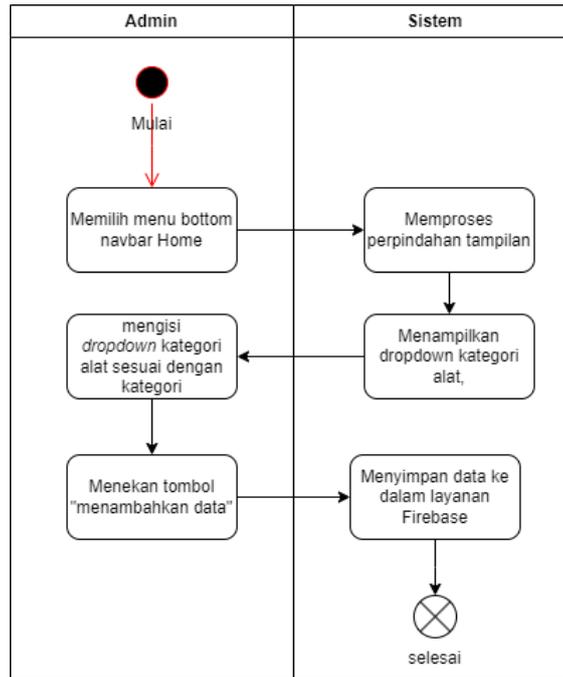
Gambar 3.2 Activity Diagram Halaman Login pada sistem e-inventory PMTCInventory

Gambar 3.2 menunjukkan *Activity Diagram* halaman login yang berfungsi sebagai salah satu keamanan pada sistem yang dibangun. Apabila pengguna ingin mengakses aplikasi, pengguna harus memasukkan *username* dan *password* yang telah diberikan sebelumnya. Jika *username* dan *password* tidak sesuai dengan yang diberikan, maka pengguna akan terus berada di halaman login. Jika sesuai, pengguna memasuki ke halaman utama dashboard.



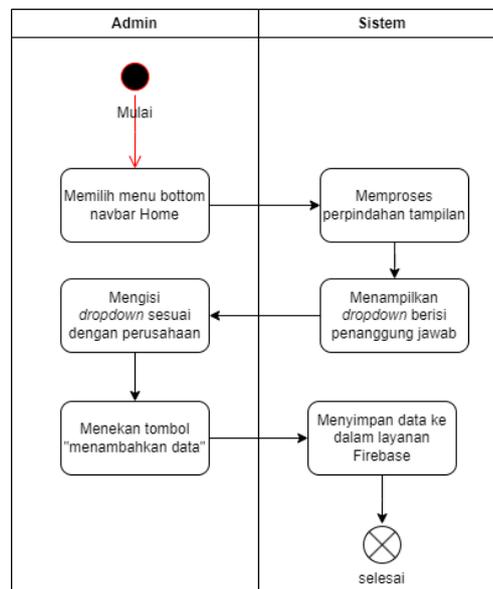
Gambar 3.3 Activity Diagram Proses Mendeskripsikan Perusahaan pemilik inventaris

Gambar 3.3 menunjukkan bagaimana proses mendeskripsikan Perusahaan pemilik inventaris aplikasi PMTCInventory. Prosesnya diawali dengan admin memilih menu bottom navbar Home, kemudian sistem menampilkan tampilan Home dan menampilkan input field, Kemudian admin diharuskan untuk mengisi input field. Setelah semua terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



Gambar 3.4 Activity Diagram Mengkategorikan Alat

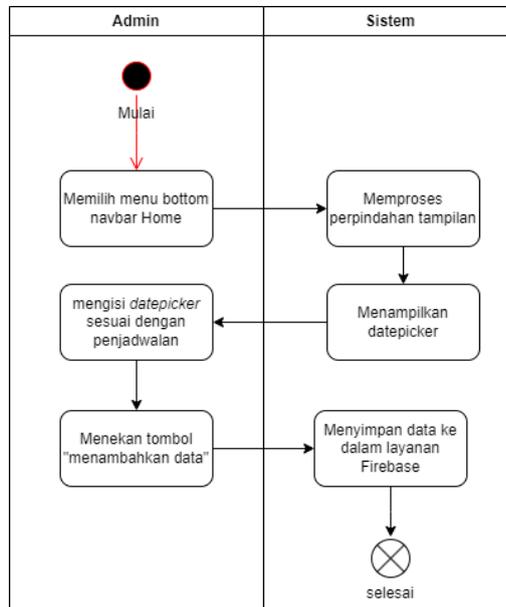
Gambar 3.4 menunjukkan bagaimana proses mengkategorikan alat pada aplikasi PMTCInventory. Prosesnya diawali dengan admin memilih menu bottom navbar Home, kemudian sistem menampilkan tampilan Home dan menampilkan dropdown, Kemudian admin diharuskan untuk mengisi dropdown. Setelah semua terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



Gambar 3.5 Activity Diagram Memilih Penanggung Jawab Inventaris

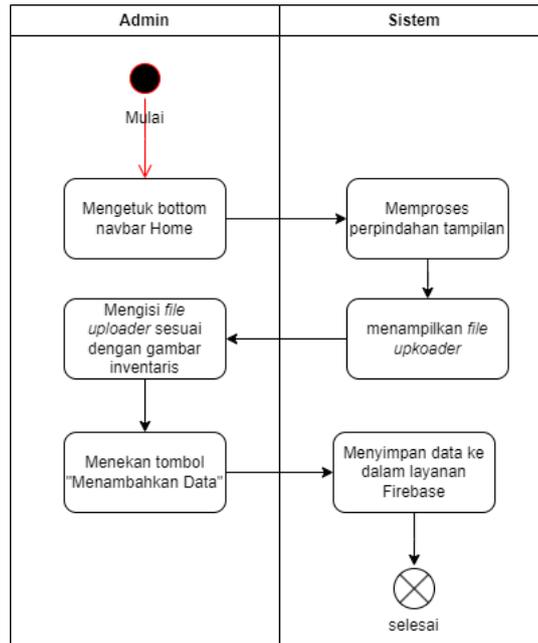
Gambar 3.5 menunjukkan bagaimana proses memilih penanggung jawab inventaris pada aplikasi PMTCInventory. Prosesnya diawali dengan admin memilih menu bottom navbar

Home, kemudian sistem menampilkan tampilan Home dan menampilkan dropdown, Kemudian admin diharuskan untuk mengisi dropdown. Setelah terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



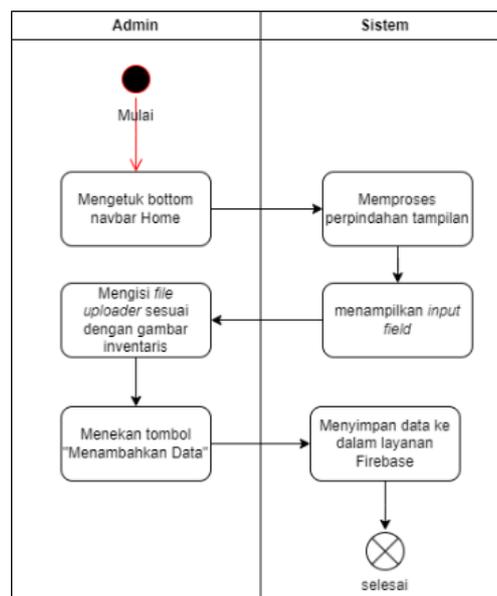
Gambar 3.6 Activity Diagram Penjadwalan

Gambar 3.6 menunjukkan bagaimana proses dilakukan penjadwalan inventaris pada aplikasi PMTCInventory. Prosesnya diawali dengan admin memilih menu bottom navbar Home, kemudian sistem menampilkan tampilan Home dan menampilkan datePicker, Kemudian admin diharuskan untuk mengisi datePicker. Setelah terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



Gambar 3.7 Activity Diagram Mengunggah Gambar Inventaris

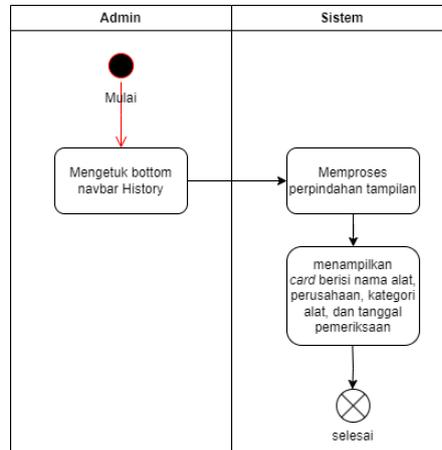
Gambar 3.7 menunjukkan bagaimana proses dilakukan mengunggah gambar inventaris pada aplikasi PMTCInventory. Prosesnya diawali dengan admin memilih menu bottom navbar Home, kemudian sistem menampilkan tampilan Home dan menampilkan file uploader, Kemudian admin diharuskan untuk mengisi file uploader. Setelah terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



Gambar 3.8 Activity Diagram Memberi Keterangan Perawatan

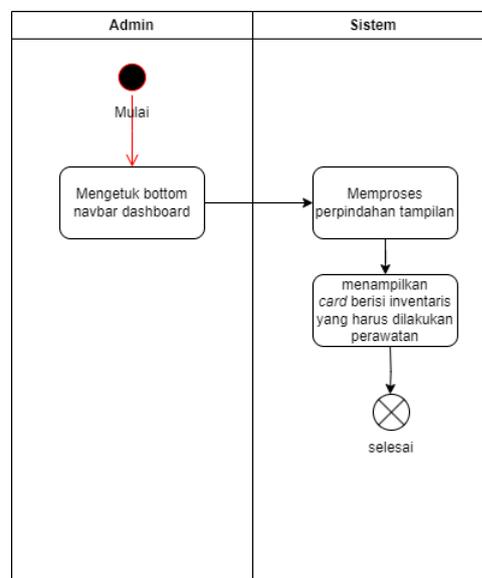
Gambar 3.8 menunjukkan bagaimana proses pemberian keterangan perawatan yang dibutuhkan pada inventaris pada aplikasi PMTCInventory. Prosesnya diawali dengan admin

memilih menu bottom navbar Home, kemudian sistem menampilkan tampilan Home dan menampilkan file uploader, Kemudian admin diharuskan untuk mengisi file uploader. Setelah terisi, admin menekan tombol “tambah data”. Berikutnya, sistem menyimpan data yang telah diisi ke dalam layanan firebase.



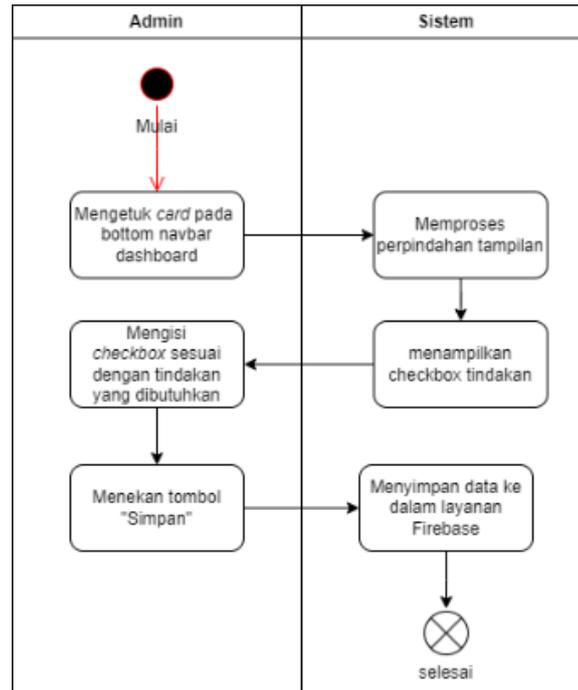
Gambar 3.9 Activity Diagram Melihat Historis Inventaris

Gambar 3.9 menunjukkan proses melihat historis inventaris yang telah dilakukan perawatan. Pertama, admin mengetuk *bottom navbar History*, kemudian sistem akan memproses perpindahan tampilan. Setelah selesai, sistem menampilkan berbagai *Card* yang berisi perusahaan, kategori alat, dan tanggal pemeriksaan.



Gambar 3.10 Activity Diagram Melaporkan Inventaris yang Harus dilakukan perawatan

Gambar 3.10 gambar 10 menunjukkan proses pelaporan inventaris yang harus dilakukan perawatan. Awal proses dimulai dari admin mengetuk bottom navbar dashboard, kemudian sistem melakukan proses perpindahan tampilan. Setelah selesai, sistem menampilkan berbagai Card yang berisi perusahaan, kategori alat, dan deadline tanggal.

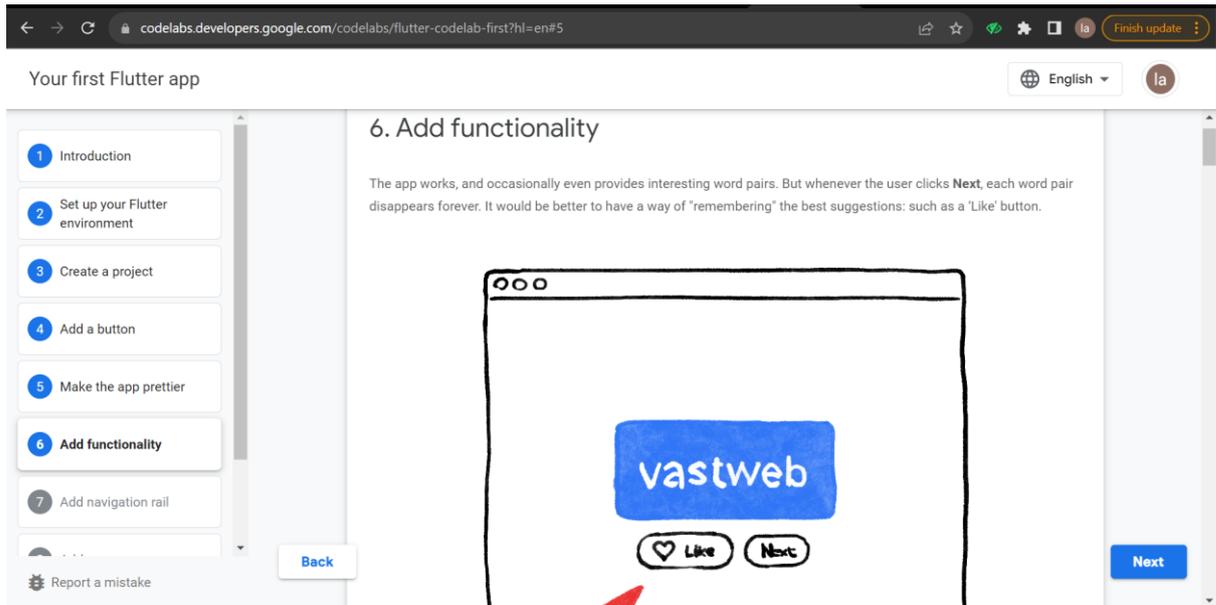


Gambar 3.11 Activity Diagram Mengisi Hasil dari Pemeriksaan

Gambar 3.11 menunjukkan diagram aktivitas mengisi hasil dari pemeriksaan. Proses ini merupakan lanjutan dari proses memberi keterangan perawatan yang dibutuhkan pada inventaris. Awal proses ini adalah admin mengetuk card yang berada pada bottom navbar dashboard. Setelah itu sistem memproses perpindahan tampilan dan sistem menampilkan checkbox tindakan yang telah diberi pada Activity Diagram Memberi Keterangan Perawatan yang Dibutuhkan Pada Inventaris. Kemudian admin mengisi checkbox sesuai dengan tindakan yang dibutuhkan. Lalu admin menekan tombol “simpan” dan sistem akan menyimpan data ke dalam layanan Firebase.

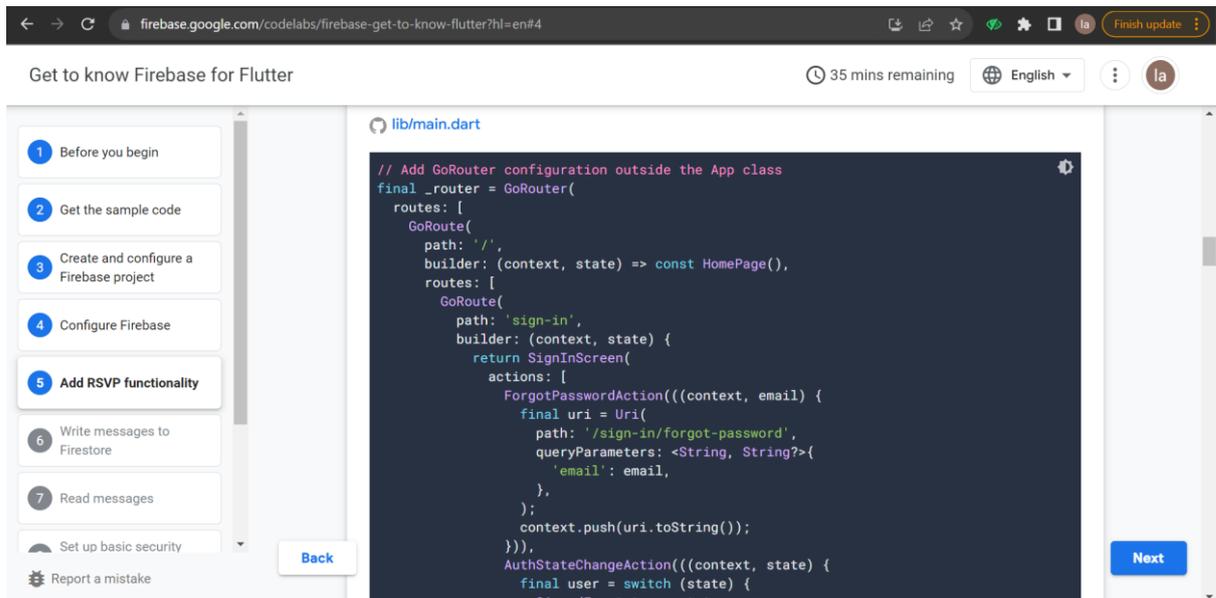
### 3.1.2 Mempelajari Flutter dan Firebase

Salah satu aktivitas yang dilakukan pada minggu kedua pada pelaksanaan magang adalah mempelajari kerangka kerja Flutter serta layanan Firebase. Pada pertemuan ini, supervisor meminta kepada pemegang untuk mempelajari kerangka kerja Flutter, bahasa pemrograman dart, dan layanan Firebase secara mandiri menggunakan layanan dari perusahaan google yang bernama *Codelabs*. Pada layanan *Codelabs*, terdapat pembelajaran untuk kerangka kerja Flutter, bahasa pemrograman Dart, dan layanan Firebase dari pemula hingga mahir. Penulis melakukan pembelajaran kerangka kerja Flutter terlebih dahulu dikarenakan penulis ditugaskan untuk membuat antarmuka aplikasi nantinya



Gambar 3.12 Tampilan Codelabs Pembelajaran Kerangka Kerja Flutter

Setelah melakukan pembelajaran kerangka kerja Flutter, penulis melakukan pembelajaran untuk layanan *Firebase*. Pembelajaran layanan tersebut dilakukan agar nantinya memudahkan integrasi layanan *Firebase* dengan kerangka kerja Flutter.

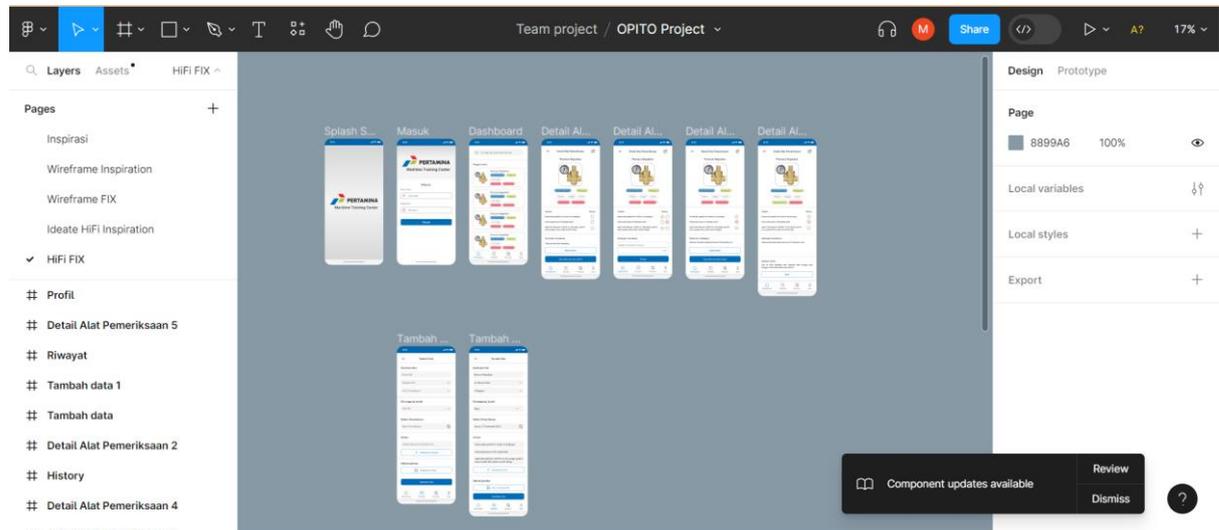


Gambar 3.13 Tampilan Pembelajaran Layanan Firebase

### 3.1.3 Pembuatan Wireframe

Meskipun pemangang bertugas sebagai Front-End perangkat bergerak, supervisor meminta kepada pemangang untuk membantu rekan pemangang dalam mendesain aplikasi. Pada pembuatan desain aplikasi, perlu terlebih dahulu membuat Wireframe. Tujuan utama dari

wireframe adalah untuk mengkomunikasikan ide dan menguji konsep desain dan menyediakan pemahaman dari struktur produk dan fungsionalitas sebelum menjadi desain yang lebih detail. Pada tahapan ini, pemegang dan rekan menggunakan layanan Figma untuk melakukan pembuatan aplikasi serta kolaborasi seperti yang ditampilkan pada gambar 3.3.



Gambar 3.14 Penggunaan Figma

### 3.1.3.1 Pembuatan Low-Fidelity Wireframe

Tahapan awal pembuatan *Wireframe* adalah pemegang dan rekan membuat terlebih dahulu *low-fidelity wireframe*. Terdapat berbagai fitur yang direpresentasikan dengan *low-fidelity* seperti tampilan fitur atau menu *Splash Screen* yang terdapat pada gambar 3.4, *login* terdapat pada gambar 3.5, *Dashboard* terdapat pada gambar 3.6, *Detail Alat* terdapat pada gambar 3.7, *Saya* terdapat pada gambar 3.8, dan *History* pada gambar 3.9.



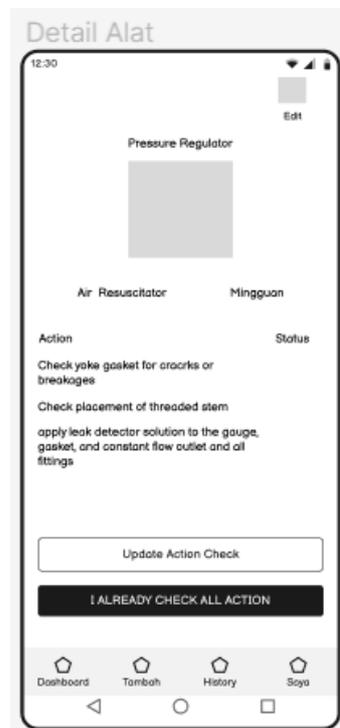
Gambar 3.15 Low-Fidelity Splash Screen



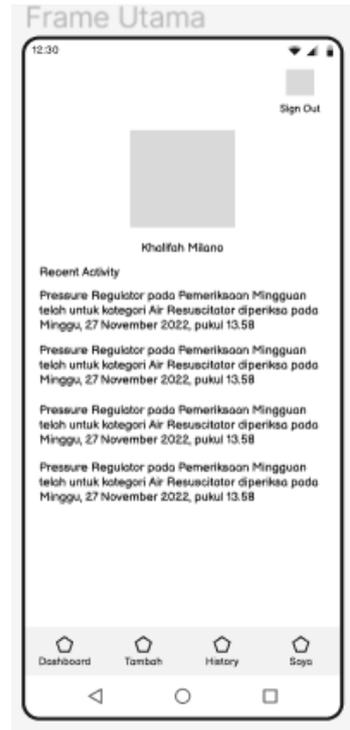
Gambar 3.16 Low-Fidelity Login



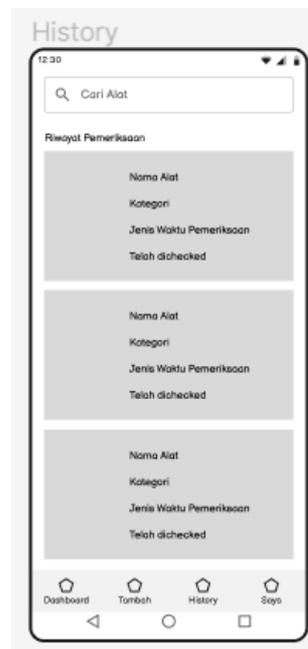
Gambar 3.17 6 Low-Fidelity Detail Alat



Gambar 3.18 Low-Fidelity Tambah Alat

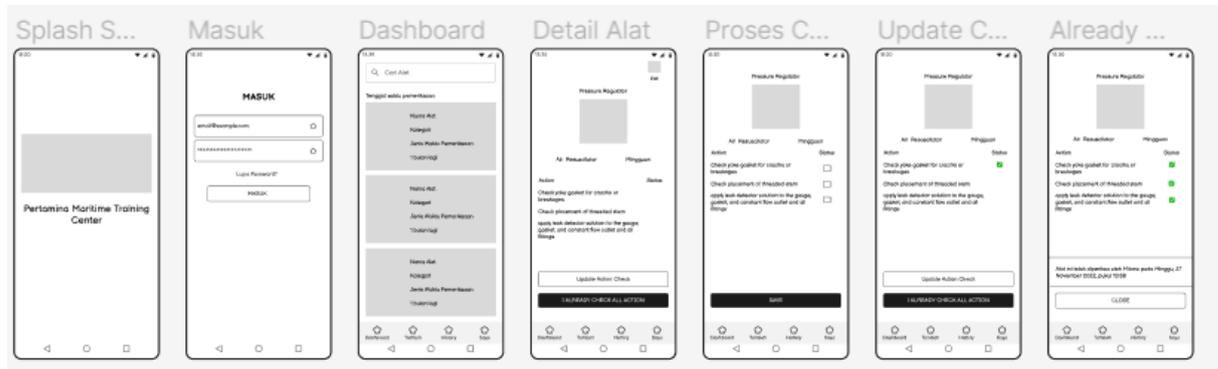


Gambar 3.19 Low-Fidelity Saya



Gambar 3.20 Low-Fidelity History

Selain pembuatan *low-fidelity wireframe*, pemegang dan rekan melakukan alur bagaimana nanti aplikasi akan berjalan menggunakan desain *low-fidelity wireframe* seperti yang akan ditunjukkan pada gambar 3.10



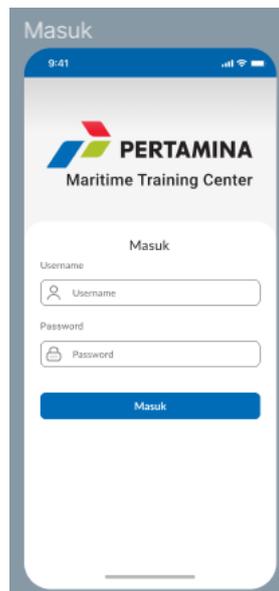
Gambar 3.21 Alur Fitur atau Menu Dashboard

### 3.1.3.2 Pembuatan High-Fidelity Wireframe

Setelah pembuatan *low-fidelity wireframe*, kegiatan selanjutnya adalah pembuatan *high-fidelity wireframe*. Kegiatan ini biasanya mendetailkan apa yang telah dikerjakan pada pembuatan *low-fidelity wireframe*. Tujuan dari *high-fidelity* adalah untuk memastikan bahwa produk akhir bertemu dengan kebutuhan user. Pada tahapan ini, pemegang mendetailkan apa yang telah dikerjakan pada pembuatan *low-fidelity wireframe* seperti tampilan fitur atau menu Splash Screen yang terdapat pada gambar 3.11, login terdapat pada gambar 3.12, Dashboard terdapat pada gambar 3.13, Detail Alat terdapat pada gambar 3.14, Saya terdapat pada gambar 3.15, History terdapat pada gambar 3.16, dan Tambah alat terdapat pada gambar 3.17,



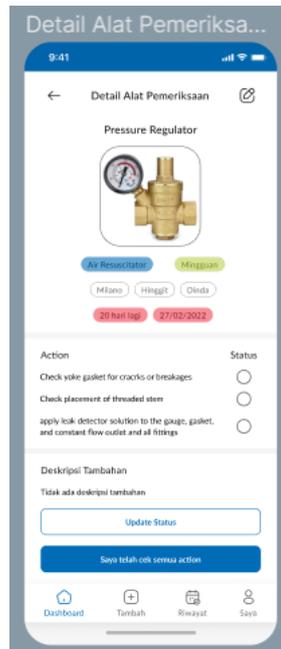
Gambar 3.22 High-Fidelity Splash Screen



Gambar 3.23 High-Fidelity Login



Gambar 3.24 High-Fidelity Dashboard



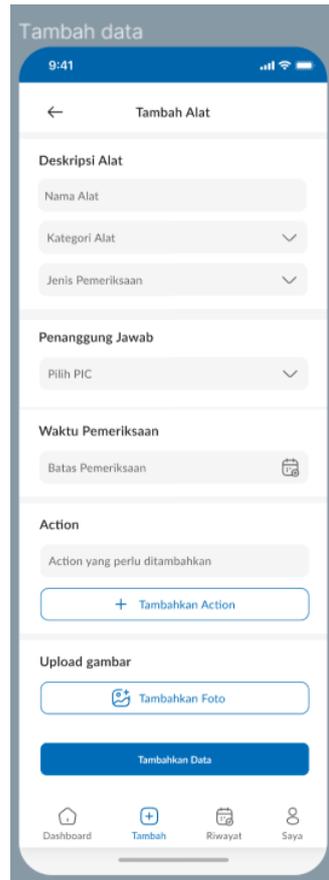
Gambar 3.25 High-Fidelity Detail Alat



Gambar 3.26 High-Fidelity Saya

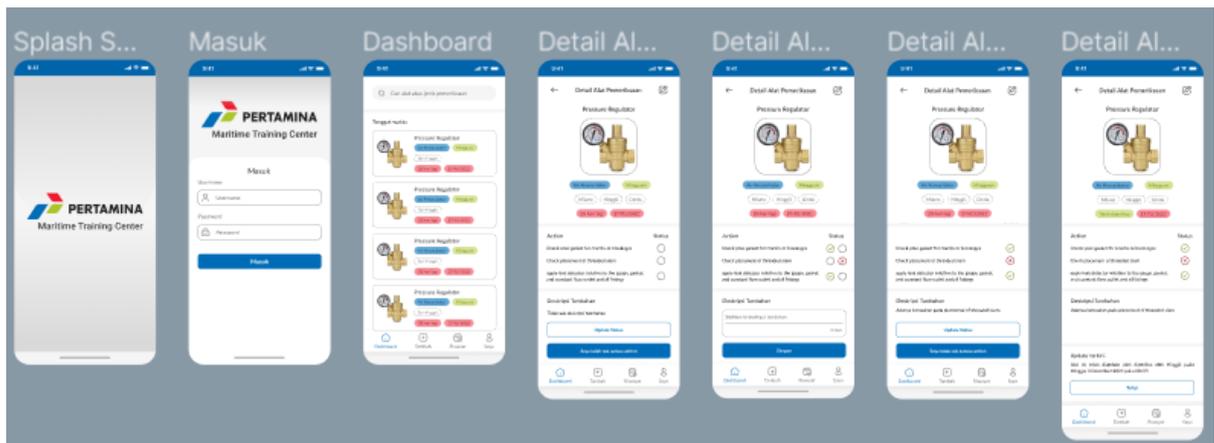


Gambar 3.27 High-Fidelity History



Gambar 3.28 High-Fidelity Tambah

Sama seperti pembuatan *low-fidelity wireframe*, pemegang dan rekan melakukan pembuatan alur bagaimana nanti aplikasi akan berjalan menggunakan desain *high-fidelity wireframe* seperti yang akan ditunjukkan pada gambar 3.18



Gambar 3.29 High-Fidelity Alur Dashboard

### 3.1.4 Pembuatan Tampilan Antarmuka Aplikasi PMTCInventory menggunakan kerangka kerja Flutter

Dalam pembuatan tampilan antarmuka aplikasi PMTCInventory, pemegang menggunakan kerangka kerja flutter dan bahasa pemrograman Dart. Di dalam aplikasi PMTCInventory, pemegang membuat berbagai tampilan untuk menu atau fitur di aplikasi. Fitur atau menu yang dikerjakan oleh pemegang adalah Splash Screen, login, Dashboard, Detail Alat, Saya, dan History.

#### 3.1.3.1 Membuat Halaman *Splash Screen*

Halaman splash screen merupakan halaman yang menampilkan logo aplikasi dengan jangka durasi yang singkat (Fauzi et al., 2021). Halaman ini bertujuan agar pengguna dapat menyadari sudah berada di aplikasi yang dituju sembari menunggu aplikasi dimulai. Halaman ini berada pada awalan ketika aplikasi baru dijalankan. Gambar 3.19 merupakan tampilan halaman untuk *splash screen*.



Gambar 3.30 Tampilan Halaman Splash Screen

Dalam pengembangannya untuk membuat splash screen, pengembang menggunakan *library material.dart*. Dalam pengembangannya pula, digunakan *asset* berupa logo perusahaan. Pada gambar 3.20 merupakan implementasi dari pengembangan *splash screen*.

```

import 'package:flutter/material.dart';
import 'dart:async';

class SplashScreen extends StatefulWidget{
  _SplashScreen createState() => _SplashScreen();
}
class _SplashScreen extends State<SplashScreen>{
  void initState(){
    super.initState();
    splashscreenStart();
  }
  splashscreenStart() async{
    var duration = const Duration(seconds: 3);
    return Timer(duration, (){
      Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => LoginPage()),
      );
    });
  }
  @override
  Widget build(BuildContext context){
    return Scaffold(
      backgroundColor: Colors.blue,
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          crossAxisAlignment: CrossAxisAlignment.center,
          children: <Widget>[
            Image.asset(
              'Images/Logo_MTC.png',
            ),
            SizedBox(height: 24.0,),
            Text("PMTIC Inventory",
              style: TextStyle(
                color: Colors.white,
                fontWeight: FontWeight.bold,
                fontSize: 30.0,
              ),
            ),
          ],
        ),
      ),
    );
  }
}

```

Gambar 3.31 implementasi dari pengembangan splash screen

### 3.1.4.2 Membuat Tampilan Login

Tampilan login merupakan fitur agar hanya pengguna yang terdaftar saja yang bisa mengakses aplikasi. Pada tampilan login, terdapat logo perusahaan, satu input field untuk memasukkan username, satu input field untuk memasukkan password, tombol masuk, serta tombol cancel. Tugas dari tiap komponen-komponen tersebut adalah input field username

harus sesuai dengan apa yang diketikkan oleh pengguna, input field password tidak boleh menunjukkan ketikan yang dimasukkan oleh pengguna, tombol cancel harus menghapus isi dari input field username dan password, tombol masuk harus membiarkan pengguna terverifikasi mengakses aplikasi. Apabila username dan password yang ditunjukkan salah, maka akan tampil pemberitahuan. Pada gambar 3.21 terdapat tampilan login yang belum diisikan. Pada gambar 3.22 terdapat tampilan apabila pengguna mengisi input field username dan password serta menekan tombol masuk namun dengan username dan atau password yang salah.



Gambar 3.32 Tampilan Halaman Login



Gambar 3.33 Tampilan Halaman Login Apabila Username dan Password

Agar pengguna bisa melakukan kegiatan *login*, digunakan lah *library* dari layanan *firebase*. Penggunaan *library* ini memudahkan pengembang dikarenakan proses verifikasi *username* dan *password* akan diserahkan oleh *library firebase authentication*. Pada gambar 3.23 merupakan penggalan kode dari penggunaan *library firebase authentication*.

```
import 'package:firebase_auth/firebase_auth.dart';
```

```

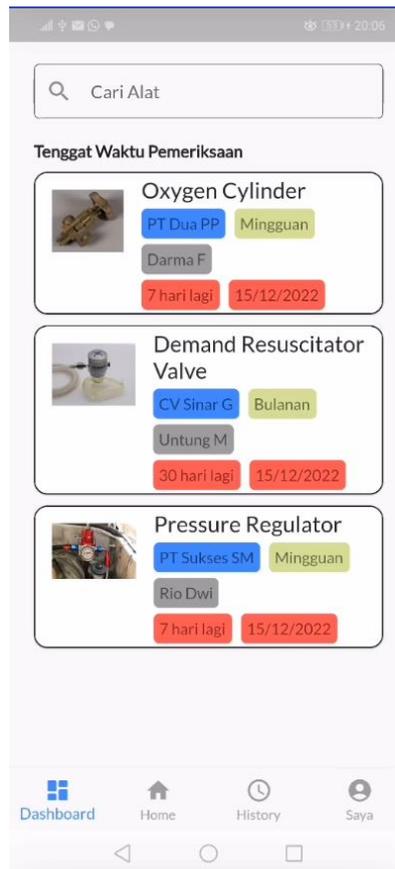
        Future<User?> signInUsingEmailAndPassword({required String email, required
String password,}) async{
    FirebaseAuth auth = FirebaseAuth.instance;
    User? user;
    try{
        UserCredential userCredential = await
auth.signInWithEmailAndPassword(email: email, password: password);
        user = userCredential.user;
        Navigator.push(context, MaterialPageRoute(builder: (context) =>
Home(),));
    } on FirebaseAuthException catch(e){
        if(e.code=="user-not-found"){
            keteranganPasswordUsernameSalah = "Password atau username anda tidak
ditemukan";
        }
    }
    return user;
}

```

Gambar 3.34 Penggunaan library firebase authentication

### 3.1.4.3 Membuat Tampilan Isi Dari Navbar Dashboard

Pada tampilan isi dari navbar dashboard, menampilkan card yang menunjukkan barang apa saja yang akan diperiksa. Isi dari card tersebut adalah perusahaan vendor tempat sebelumnya membeli inventaris tersebut, jenis pemeriksaan seperti, penanggung jawab, deadline, serta nama inventarisnya. Selain itu terdapat pula fitur pencarian alat yang berfungsi untuk mencari alat yang akan diperiksa. Jadi, fungsi daripada navbar dashboard adalah untuk mengingatkan pengguna kapan untuk memeriksa inventaris barang, inventaris apa yang harus diperiksa. Pada gambar 3.22, tampak jelas tampilan daripada isi dari navbar Dashboard.



Gambar 3.35 Tampilan Isi Navbar Dashboard

Untuk mempermudah pembuatan *Bottom Navbar*, digunakanlah *library* dari flutter yakni *material.dart*. *Library* ini pada dasarnya adalah *widget* dari *flutter* yang mengimplementasi dari *Material Design*. *Material Design* sendiri merupakan sistem desain yang dibuat dan didukung oleh pengembang dan desainer Google. *Material Design* juga memberi arahan pada UX dan komponen antarmuka yang diimplementasikan untuk Android, Flutter dan Web. Pengembang dimudahkan dengan adanya *library* ini karena secara desain sudah sesuai dengan standar industri. Pada gambar 3.25 ditunjukkan penggalan kode untuk membangun *Bottom Navbar* untuk Dashboard, Home, History, Saya.

```
class Home extends StatefulWidget {
  const Home({Key? key}) : super(key: key);

  @override
  State<Home> createState() => _HomeState();
}

import 'package:flutter/material.dart';

class _HomeState extends State<Home> {
  int _selectedItems = 0;

  static const List<Widget> _widgetOptions = <Widget>[
    Dashboard(),
    TambahAlat(),
    RiwayatPemeriksaan(),
```

```

    Saya ()
  ];

  void onItemTapped(int index) {
    setState(() {
      _selectedItems = index;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: _widgetOptions.elementAt(_selectedItems),
      bottomNavigationBar: BottomNavigationBar(
        type: BottomNavigationBarType.fixed,
        // <-- This works for fixed
        selectedItemColor: Colors.blueAccent,
        unselectedItemColor: Colors.grey,
        items: const <BottomNavigationBarItem>[
          BottomNavigationBarItem(
            icon: Icon(Icons.dashboard_rounded),
            label: "Dashboard",
          ),
          BottomNavigationBarItem(
            icon: Icon(Icons.home),
            label: "Home",
          ),
          BottomNavigationBarItem(
            icon: Icon(Icons.access_time), label: "History"),
          BottomNavigationBarItem(
            icon: Icon(Icons.account_circle_rounded), label: "Saya"),
        ],
        currentIndex: _selectedItems,

        onTap: onItemTapped,
      ),
    );
  }
}

```

Gambar 3.36 penggalan kode untuk membangun Bottom Navbar untuk Dashboard, Home, History, Saya.

#### 3.1.4.4 Membuat Tampilan Isi Dari Navbar Home

Fungsi daripada navbar Home adalah untuk membuat pengingat pengecekan inventaris kantor. Terdapat *textfield*, *dropdown*, *upload* gambar, *date*. Pada tampilan isi dari navbar home, pengguna diharapkan mengisi nama perusahaan, kategori alat, penanggung jawab, batas waktu, dan mengunggah gambar agar tujuan dari aplikasi berjalan dengan semestinya, yakni mempermudah pengelolaan inventaris kantor. Pada gambar 3.26 dan gambar 3.27, merupakan tampilan dari isi navbar Home.

**Tambah Alat**

**Deskripsi Perusahaan**

Nama Perusahaan

Kategori Alat  
Air Resuscitator

Jenis Pemeriksaan  
Harian

**Penanggung Jawab**

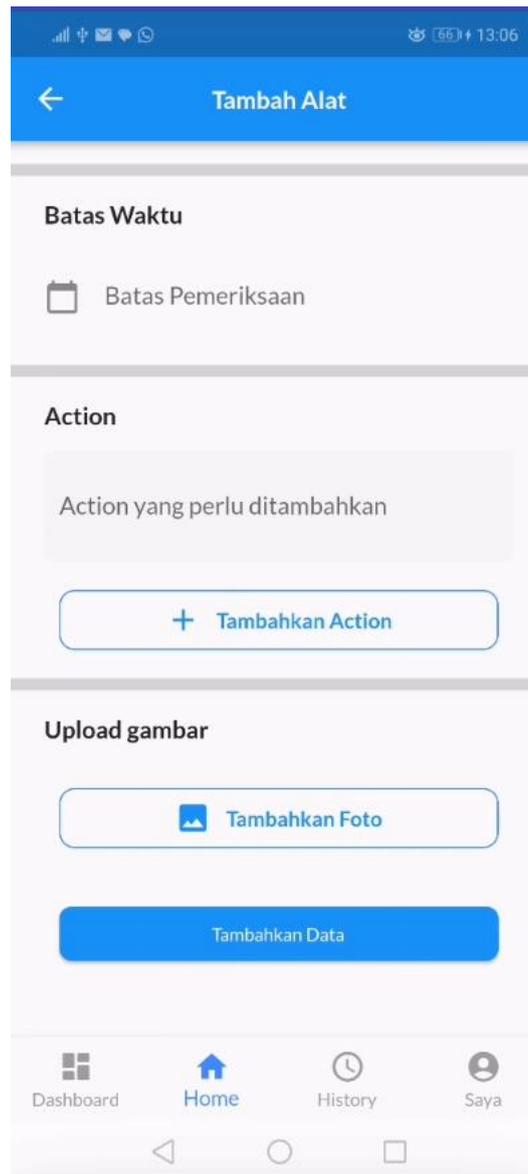
Pilih PIC  
Untung M

**Batas Waktu**

Batas Pemeriksaan

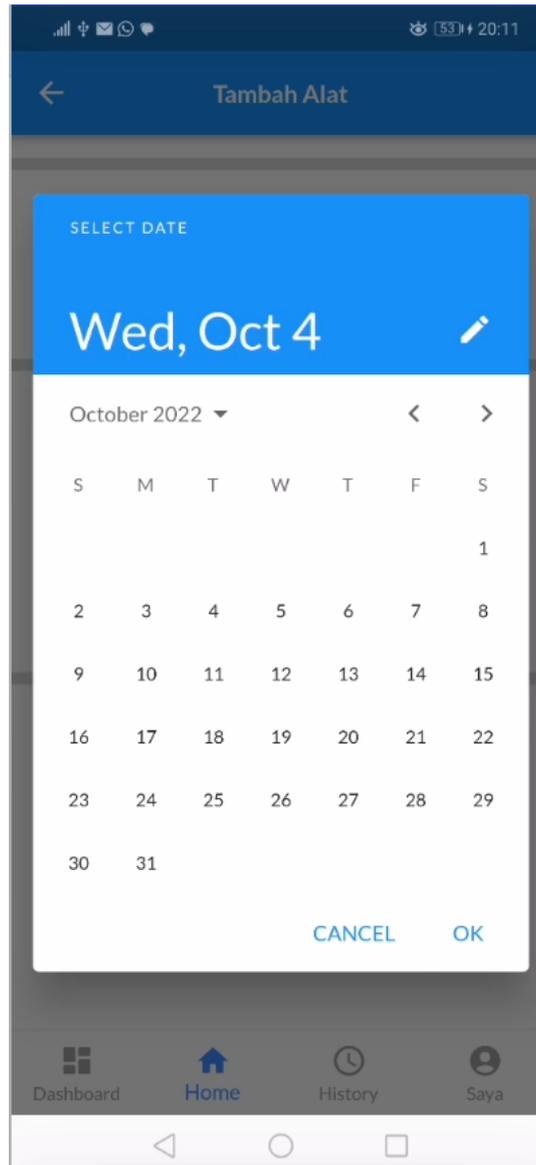
Dashboard Home History Saya

Gambar 3.37 Tampilan Isi Dari Navbar Home



Gambar 3.38 Tampilan Isi Dari Navbar Home

Pada isi dari navbar Home juga terdapat pengisian yang menggunakan *date picker* untuk menentukan kapan waktu untuk dilakukannya perawatan atau pengecekan. Tampilan dari gambar tersebut ditunjukkan pada gambar 3.28.



Gambar 3.39 Tampilan Penggunaan Date Picker

Untuk memudahkan pengembang, digunakan lah *library* dari *flutter* yakni *material.dart* untuk membuat penanggalan. Alasan digunakannya *library* ini adalah karena untuk mengimplementasikannya sangat mudah. Pada gambar 3.29 merupakan penggalan dari kode untuk dibuatnya *date picker*.

```
import 'package:flutter/material.dart';

class datePicker extends StatefulWidget {
  const datePicker({Key? key}) : super(key: key);

  @override
  State<datePicker> createState() => _datePickerState();
}

class _datePickerState extends State<datePicker> {
  TextEditingController dateController = TextEditingController();

  @override
```

```

void initState() {
  // TODO: implement initState
  dateController.text = "";
}

@override
Widget build(BuildContext context) {
  return TextField(
    controller: dateController,
    decoration: const InputDecoration(
      icon: Icon(Icons.calendar_today),
      labelText: 'Batas Pemeriksaan',
      border: InputBorder.none),
    readOnly: true,
    onTap: () async {
      DateTime? pickedDate = await showDatePicker(
        context: context,
        initialDate: DateTime.now(), //get today's date
        firstDate: DateTime(
          2000), //DateTime.now() - not to allow to choose before today.
        lastDate: DateTime(2101));

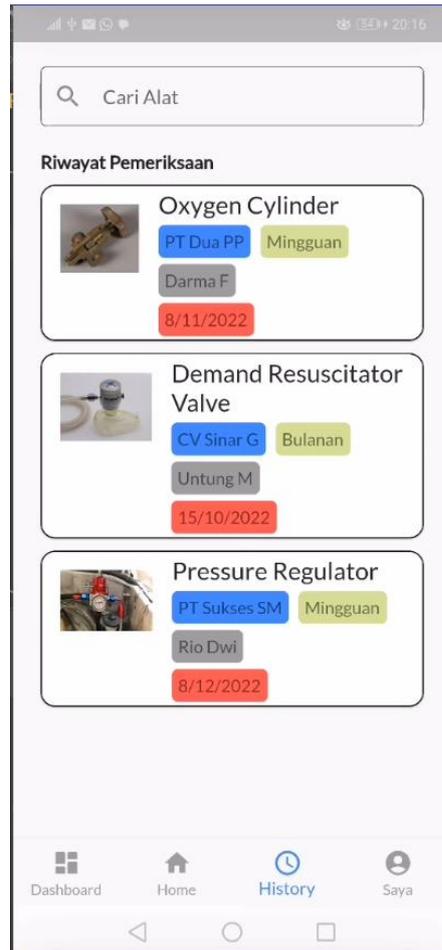
      if (pickedDate != null) {
        setState(() {
          dateController.text =
            formattedDate; //set formatted date to TextField value.
        });
      }
    },
  );
}

```

Gambar 3.40 Penggalan dari kode untuk dibuatnya date picker

### 3.1.4.5 Membuat Tampilan Isi Dari Navbar History

Isi daripada navbar history adalah card yang berisi nama gambar inventaris kantor, nama perusahaan penyedia barang, jenis pemeriksaan, penanggung jawab, dan waktu pemeriksaan yang telah dilakukan. Pada navbar history, terdapat pula fitur pencarian barang yang telah dilakukan pemeriksaan atau perawatan sebelumnya. Tujuan dari fitur pencarian barang adalah untuk memudahkan pengguna dalam mencari alat yang telah diperiksa sebelumnya. Fungsi dari navbar ini adalah untuk menampilkan barang apa saja yang telah diperiksa. Pada gambar 3.30 ditampilkan wujud dari card dan fitur pencarian alat



Gambar 3.41 Tampilan dari isi navbar History

#### 3.1.4.6 Membuat Tampilan Isi Dari Navbar Saya

Isi dari navbar Saya adalah foto diri, email pengguna, dan aktivitas yang telah dilakukan oleh pengguna. Fungsi dari navbar saya adalah agar mudah mengetahui apa saja yang telah dilakukan oleh pengguna. Pada gambar 3.31 menunjukkan isi dari tampilan Navbar Saya.



Gambar 3.42 Tampilan dari isi navbar Saya

Untuk menampilkan email pengguna, digunakanlah *library* dari *firebase*. *Library* yang digunakan adalah *firebase authentication*. Penggunaan *library* ini dikarenakan dari awal proyek ini diinisiasi sudah menggunakan layanan *firebase*. Maka dari itu digunakan lah *Library* ini. Pada gambar 3.32 ditunjukkan implentasi kode.

```
import 'package:firebase_auth/firebase_auth.dart';

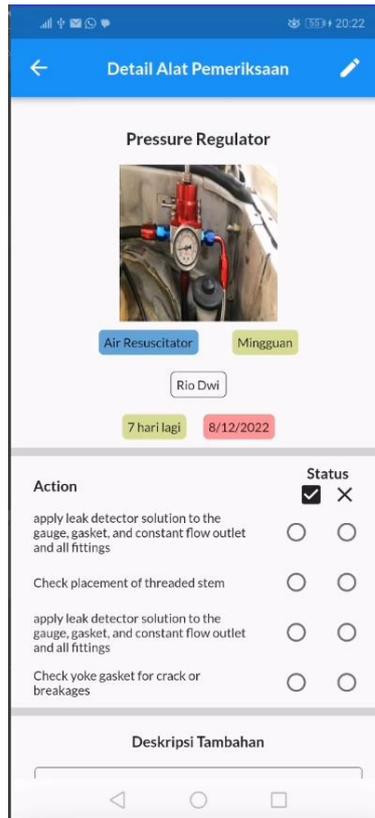
Container(
  margin: EdgeInsets.only(bottom: 15),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Text(
        FirebaseAuth.instance.currentUser!.email!?"No user",
        style: fontLatoJudul,
      )
    ]
  )
)
```



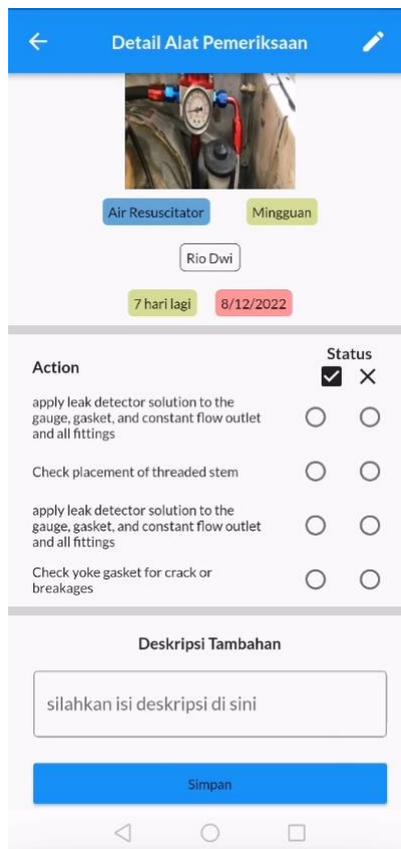
Gambar 3.43 Implementasi firebase authentication

### 3.1.4.7 Membuat Tampilan Detail Alat Pemeriksaan

Tampilan ini merupakan hasil dari dropdown, textfield, upload gambar, dan deadline waktu yang telah pengguna isi pada menu navbar home. Fungsi dari fitur ini adalah agar pada saat hari pemeriksaan, pengguna tinggal mengisi perawatan atau pengecekan apa saja yang diperlukan. Setelah disimpan, data akan dikirimkan ke firebase yang nantinya akan diolah kembali di luar scope aplikasi ini. Pada gambar 3.33 dan 3.34, tampilan dari Detail Alat Pemeriksaan



Gambar 3.44 Tampilan dari Detail Alat



Gambar 3.45 Tampilan dari Detail Alat

### 3.1.5 Pengujian

Pengujian dilakukan hanya sebatas untuk mengetahui kesesuaian antara desain user interface yang telah dibuat dengan implementasi desain ke kode yang telah dibuat. Pengujian ini menggunakan metode Black Box Testing. Pengujian ini dilakukan oleh supervisor.

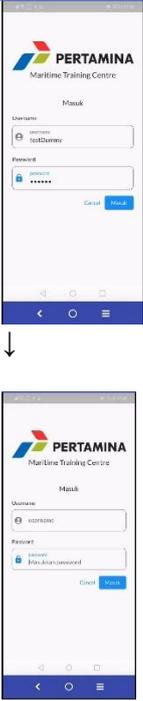
Black box testing atau dapat disebut juga Behavioral Testing adalah pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak. Pengujian ini dilakukan di akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik.

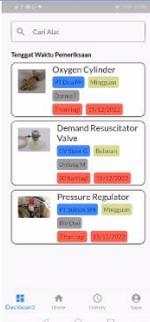
Tujuan dari pengujian adalah untuk memastikan apakah masing-masing fungsionalitas yang telah dikembangkan pada sistem dapat berjalan sesuai dengan proses bisnis dan kebutuhan pelanggan. Tahapan – tahapan pengujian dilakukan dengan menambahkan data dan menguji setiap fitur yang terdapat di masing – masing halaman tersebut menggunakan beberapa kasus pengujian dilakukan secara manual. Pada tabel 3.2 merupakan hasil dari pengujian yang telah dilakukan pada aplikasi PMTCInventory.

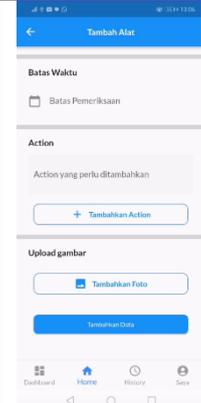
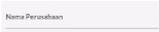
Tabel 3.2 Pengujian Black Box Testing

No	Aktivitas Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Klik Tombol Masuk 	Apabila <i>user</i> yang memiliki akun, user dapat berhasil masuk ke dalam aplikasi jika <i>username</i> dan <i>password</i> yang dimasukkan tepat. Apabila <i>username</i> dan <i>password</i> tidak tepat, maka <i>user</i> tidak dapat mengakses aplikasi	Masuk menggunakan <i>username</i> serta <i>password</i> yang sudah terdaftar → <i>user</i> dapat masuk kedalam aplikasi  Masuk dengan menggunakan <i>password</i> dan <i>username</i> yang tidak terdaftar → terdapat pemberitahuan “Password atau	[x] Sesuai [] Perbaiki

			<p>username salah”</p> 	
2	<p>Klik Tombol Saya</p> 	<p>Apabila <i>user</i> yang memiliki akun, maka <i>user</i> dapat mengakses menu profile</p>	<p>Tombol “Saya” ditekan → Tampilan menu “Saya” muncul</p> 	<p>[x] Sesuai [ ] Perbaiki</p>
3	<p>Klik Tombol Cancel</p> 	<p>Apabila <i>user</i> menekan tombol <i>cancel</i>, maka kolom <i>username</i> dan <i>password</i> akan dikosongkan</p>	<p>Memasukkan <i>username</i> ke dalam kolom <i>username</i> dan memasukkan <i>password</i> kedalam kolom <i>password</i> Kemudian menekan tombol “Cancel” → isi daripada kolom <i>password</i> dan <i>username</i> kosong</p>	<p>[x] Sesuai [ ] Perbaiki</p>

				
4	<p>Memasukkan <i>Password</i></p> 	<p>Apabila <i>user</i> memasukkan <i>password</i>, <i>password</i> tidak boleh terlihat atau <i>obscure</i></p>	<p>Memasukkan <i>password</i> kedalam kolom <i>password</i> → <i>password</i> tidak terlihat</p> 	<p>[x] Sesuai [ ] Perbaiki</p>
5.	<p>Mengklik bottom navbar dashboard</p> 	<p>Apabila user mengklik bottom navbar dashboard, harus tertampil kumpulan card dan searchbox dan masuk kedalam halaman dashboard</p>	<p>Mengklik bottom navbar dashboard → tertampil kumpulan card dan searchbox dan masuk kedalam halaman dashboard</p>	<p>[x] Sesuai [ ] Perbaiki</p>

				
6.	<p>Mengklik bottom navbar Home</p> 	<p>Apabila user mengklik bottom navbar Home, harus tertampil textfield untuk deskripsi perusahaan, dropdown untuk kategori alat, Dropdown jenis pemeriksaan, Dropdown penanggung jawab, Datepicker batas waktu, Textfield action, dan upload gambar, dan menuju halaman home</p>	<p>Mengklik bottom navbar Home → tertampil textfield untuk deskripsi perusahaan, dropdown untuk kategori alat, Dropdown jenis pemeriksaan, Dropdown penanggung jawab, Datepicker batas waktu, Textfield action, dan upload gambar, dan menuju halaman home</p> 	<p>[x] Sesuai [ ] Perbaiki</p>

				
7.	Memasukan text pada textfield nama perusahaan di laman Home	Apabila user memasukkan text, akan tertampil text yang diketik 	User memasukkan text kedalam textfield nama perusahaan → Text yang dimasukkan tertampil	[x] Sesuai [ ] Perbaiki
8.	Mengklik dropdown kategori alat pada laman Home	Apabila user Mengklik dropdown kategori alat pada laman Home, ditampikan jenis-jenis alat	User mengklik dropdown kategori alat → Menampikan keseluruhan kategori alat	[x] Sesuai [ ] Perbaiki
9.	Mengklik dropdown jenis pemeriksaan pada laman Home	Apabila user Mengklik dropdown jenis pemeriksaan pada laman Home, ditampikan jenis-jenis pemeriksaan	User mengklik dropdown jenis pemeriksaan → Menampikan keseluruhan jenis pemeriksaan	[x] Sesuai [ ] Perbaiki
10.	Mengklik dropdown penanggung jawab pada laman Home	Apabila user Mengklik dropdown penanggung jawab pada laman Home, ditampikan nama-nama penanggung jawab	User mengklik dropdown Penanggung Jawab → Menampikan nama-nama penanggung jawab	[x] Sesuai [ ] Perbaiki
11	Mengklik datepicker batas waktu pada laman Home	Apabila user Mengklik datepicker batas waktu laman Home,	User mengklik datepicker batas waktu → Menampikan kalender	[x] Sesuai [ ] Perbaiki

		ditampilkan kalender		
12.	Memasukan text pada textfield action di laman Home	Apabila user memasukkan text pada textfield action, tertampil text yang telah dimasukkan	User memasukkan text pada textfield action → tertampil text yang telah dimasukkan	[x] Sesuai [ ] Perbaiki
13.	Mengklik tombol tambah foto pada file uploader upload gambar di laman Home	Apabila user mengklik tombol tambah foto, user akan diminta untuk memilih gambar dari perangkat	User mengklik tombol tambah foto → User dapat menambahkan gambar dari perangkat	[x] Sesuai [ ] Perbaiki
14	Mengklik tombol tambah data pada laman Home	Apabila user mengklik tombol tambah data, data yang diisi sebelumnya masuk kedalam database dan hasilnya menjadi card pada laman Dashboard	User mengklik tombol tambah data → Data masuk kedalam database dan hasilnya menjadi card pada laman Dashboard	[x] Sesuai [ ] Perbaiki
15	Mengklik card pada laman Dashboard	Apabila user mengklik card pada laman Dashboard, ditampilkan gambar, kategori alat, jenis pemeriksaan, nama perusahaan, nama penanggung jawab, batas waktu, dan action yang telah ditambahkan pada laman Home serta textfield deskripsi tambahan dan tombol simpan	User mengklik card pada laman Dashboard → Menampilkan gambar, kategori alat, jenis pemeriksaan, nama perusahaan, nama penanggung jawab, batas waktu, dan action yang telah ditambahkan pada laman Home serta textfield deskripsi tambahan dan tombol simpan	[x] Sesuai [ ] Perbaiki

Setelah melakukan pengujian *black-box testing*, supervisor berpendapat bahwa aplikasi tersebut sudah cukup untuk memenuhi kebutuhan pengguna.

## **BAB IV**

### **REFLEKSI PELAKSANAAN MAGANG**

#### **4.1 Relevansi Akademik**

PMTCInventory merupakan aplikasi berbasis mobile yang menerapkan kerangka kerja Flutter, bahasa pemrograman Dart dan layanan Firebase dalam pengembangannya. Pada pengembangan aplikasi tersebut, terdapat keterkaitan dengan teori mata kuliah Pengembangan Aplikasi Bergerak. Pada mata kuliah Pengembangan Aplikasi Bergerak, mahasiswa diberi teori dan dasar-dasar dalam pengembangan aplikasi berbasis mobile. Meskipun di dalam mata kuliah tersebut menggunakan bahasa pemrograman Kotlin dan database SQLite, namun secara konsep dasarnya sama. Untuk mengembangkan aplikasi PMTCInventory, menggunakan pola desain Singleton pada praktiknya. Dalam pengembangannya pula, menggunakan library Materials.dart yang mempermudah pengembang dalam membangun tampilan atau front-end. Untuk mengolah dan mengonsumsi data, digunakanlah layanan Firebase sebagai Back-end.

Proses manajemen proyek di Pertamina Maritime Training Centre tidak ditemukan perbedaan antara teori mata kuliah Manajemen Pengembangan Teknologi Informasi (MPTI) dengan pelaksanaan di lapangan. Berdasarkan teori tahapan manajemen proyek dimulai dari inisiasi proyek, perencanaan proyek, eksekusi proyek, pemantauan proyek, dan penutupan proyek. Akan tetapi, pada tahapan pemantauan proyek, tidak ditemukannya penerapan teknologi seperti Jira ataupun Trello. Pemantauan proyek dilakukan secara manual seperti meeting progress mingguan yang menyebabkan pemantauan proyek kurang terkontrol dengan baik. Hal itu mengakibatkan task yang diberikan oleh Supervisor menjadi tidak terstruktur dan supervisor pun harus melakukan update progress secara berkala yang mana permasalahan tersebut dapat diatasi dengan menggunakan teknologi manajemen proyek.

Secara teori yang pada mata kuliah Pengembangan Sistem Informasi (PSI) terdapat aktivitas utama dalam pengembangan proyek dimulai dari tahap System Analysis, System Design, Programming, Testing, Conversation, Production and Maintenance. pengembangan aplikasi PMTCInventory, digunakanlah metode pengembangan proyek atau metode System Development Life Cycle (SDLC) waterfall. Tahapan pada metode waterfall adalah Requirement, Design, Implementation, verification or testing, deployment and maintenance. Pada tahapan Requirement, dilakukanlah wawancara kepada pengguna untuk mengetahui

kebutuhan pengguna. Pada tahapan Design, dilakukanlah pembuatan low-fidelity Wireframe dan high-fidelity wireframe. Pada tahapannya implementation, dilakukannya pemrograman. Yakni mengkonversi design menjadi barisan code. Pada tahapan Verification or Testing, dilakukanlah Black-Box Testing. Pada tahapan Deployment and maintenance, dilakukannya deploy aplikasi

Pada pengujian menggunakan metode black box testing tidak ditemukan perbedaan antara teori dengan praktik dilapangan. Secara teori black box testing merupakan metode pengujian tanpa mengetahui kode program didalamnya untuk memastikan seluruh input dan output berjalan sesuai proses bisnis yang diharapkan. Pengujian dilakukan oleh Supervisor melalui aplikasi PMTCInventory untuk memastikan apakah masukan dan keluaran yang dihasilkan sesuai dengan Requirement.

## **4.2 Pembelajaran Magang**

### **4.2.1 Manfaat Magang**

#### **Mendapatkan pengetahuan dan tools baru.**

Selama melaksanakan kegiatan magang, berbagai pengetahuan baru mengenai tools dan pengetahuan didapatkan. Penulis mendapatkan kesempatan untuk mengembangkan *front-end* menggunakan kerangka kerja Flutter dan bahasa pemrograman yang sebelumnya pernah digunakan oleh penulis dan tidak didapatkan di bangku kuliah. Selain itu, penulis juga merasakan secara langsung dalam proses wawancara pengguna untuk mengetahui keinginan dan kebutuhan pengguna. Penggunaan tools seperti github dan figma bukan pertama kali digunakan oleh penulis, namun penulis tetap mendapatkan pengetahuan-pengetahuan baru mengenai penggunaan tools tersebut. Pada penggunaan figma, untuk pertama kalinya penulis mengetahui bagaimana alur pemakaian aplikasi dan alur mendesain dari awal seperti pembuatan wireframe low-fidelity terlebih dahulu kemudian pembuatan wireframe high-fidelity.

#### **Manajemen diri**

Pada saat pelaksanaan magang, penulis juga sedang menjalankan kuliah dengan mana tiga mata kuliah. Selain itu pemegang juga harus mempelajari tools baru seperti kerangka kerja Flutter, bahasa pemrograman Dart dan layanan Firebase. Dengan banyaknya tanggung jawab

yang harus dijalani oleh penulis, oleh karena itu penulis harus dapat membagi waktu seefektif mungkin. Pemagang harus membagi waktu magang serta harus dapat menyelesaikan kuliah dengan tepat waktu.

### **Bekerja sama dengan tim**

Pada pengembangan aplikasi, terdapat berbagai peran yang dilibatkan pada proyek tersebut. Pembagai peran ini membuat pengembangan menjadi terstruktur karena tugas yang diberikan sesuai dengan scope masing-masing peran.

### **4.3 Kendala, Hambatan, dan Tantangan Selama Magang**

Selama mengikuti kegiatan magang di Pertamina Maritime Training Center penulis merasakan beberapa kendala, hambatan, dan tantangan yang dirasakan baik dari diri sendiri maupun lingkungan sekitar. Kendala, hambatan, dan tantangan tersebut menjadi pembelajaran bagi penulis untuk terus memperbaiki diri dalam bekerja ke depannya.

Ada beberapa kendala yang dihadapi oleh penulis. Adapun kendala-kendala yang dihadapi tersebut antara lain :

1. Pada saat awal pelaksanaan magang, mengalami permasalahan dalam hal berkomunikasi dengan karyawan, karena masih dalam tahap awal dan belum mengetahui karakter dari masing-masing rekan karyawan sehingga cukup sulit untuk meminta penjelasan tentang pekerjaan yang akan penulis lakukan
2. Karena supervisor sibuk dan sering dinas ke luar kota, menjadikan kesulitan untuk berkomunikasi
3. Waktu yang dilaksanakan dalam kegiatan magang sangatlah singkat sehingga banyak pekerjaan yang tidak maksimal
4. Jobdesc yang diberikan kepada penulis tidak jelas, seperti penulis harus membantu desain UI/UX. Bagi penulis bukanlah masalah, namun hal tersebut merupakan di luar jobdesc pemagang sebagai front-end
5. Terdapat periode dimana pendingin ruangan sebagian besar rusak padahal ruangan kantor tidak bisa dibuka jendelanya. Hal ini membuat penulis merasakan kepanasan dan tidak nyaman dalam melaksanakan magang.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Setelah melewati masa magang, pemegang menyimpulkan bahwa penggunaan kerangka kerja Flutter memudahkan pengembang dalam mengembangkan aplikasi karena dokumentasi dan komunitas pengguna Flutter yang banyak. Kemudahan ini juga dikarenakan banyaknya *library* pendukung salah satunya adalah Material.dart yang sangat memudahkan pengembang. Proses untuk mempelajari kerangka kerja Flutter dan *library* pendukungnya dimudahkan dengan adanya platform codelabs. Sehingga untuk mempelajari kerangka kerja Flutter sudah terarah. Kerangka kerja Flutter juga mempercepat proses pengembangan dikarenakan sifatnya yang *multi-platform* sehingga pengembang hanya perlu satu *codebase* untuk berbagai platform.

#### **5.2 Saran**

Terdapat beberapa saran dari penulis untuk perusahaan yang mungkin akan mempermudah pengembangan aplikasi diantaranya :

- a. Sebaiknya menggunakan teknologi untuk manajemen proyek seperti Trello atau Jira, sehingga dapat mempermudah dan mempercepat pengembangan
- b. Pengerjaan di luar jobdesc mungkin tidak masalah karena dapat ilmu baru, namun mungkin akan lebih baik jika pengerjaan sesuai dengan tugas yang telah ditentukan dari awal
- c. Sebaiknya desain UI/UX dirancang hingga final sebelum diimplementasikan ke dalam kode program sehingga meminimalisir terjadinya perubahan pada saat pengembangan.

**DAFTAR PUSTAKA**

- Ahmadi, T., & Safwan. (2020). Rancang Bangun Video Conference Hologram Berbasis Android. *Jurnal J-Innovation*, 9(1), 16–19.
- Aryasta, A. R. (2022). *Pengembangan Front End Sistem Informasi Akuntansi Menggunakan Kerangka Kerja Vue.JS*. Universitas Islam Indonesia.
- Fauzi, M., Teddyana, A., & Enda, D. (2021). Pengembangan Aplikasi Mobile Tanggap Bencana Di Kab. Bengkalis Menggunakan Framework Flutter. *ZONasi: Jurnal Sistem Informasi*, 3(1), 27–36. <https://doi.org/10.31849/zn.v3i1.5856>
- Flutter, E. M., & Firmansyah, D. (2018). *Pengembangan Front-end Aplikasi Mobile*.
- Haryuda, D., Asfi, M., & Fahrudin, R. (2021). Perancangan UI/UX Menggunakan Metode Design Thinking Berbasis Web Pada Laportea Company. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 8(1), 111–117. <https://doi.org/10.33197/jitter.vol8.iss1.2021.730>
- Hendriawan, M., Budiman, T., Yasin, V., & Rini, A. S. (2021). Pengembangan Aplikasi E-Commerce Di Pt. Putra Sumber Abadi Menggunakan Flutter. *Journal of Information System, Informatics and Computing*, 5(1), 69. <https://doi.org/10.52362/jisicom.v5i1.371>
- Herdiansah, A., Nurnaningsih, D., & Rusdianto, H. (2022). Pemanfaatan Flutter Pada Pengembangan Aplikasi Mobile Ebisnis Penyediaan Bahan Baku Bisnis Katering. *Jurnal Teknoinfo*, 16(2), 291. <https://doi.org/10.33365/jti.v16i2.1937>
- Kurniawan, T., Samsudin, S., & Triase, T. (2021). Implementasi Layanan Firebase pada Pengembangan Aplikasi Sewa Sarana Olahraga Berbasis Android. *Jurnal Informatika Universitas Pamulang*, 6(1), 13. <https://doi.org/10.32493/informatika.v6i1.10270>
- Razi, A. A., Mutiaz, I. R., & Setiawan, P. (2018). Penerapan Metode Design Thinking Pada Model Perancangan Ui/Ux Aplikasi Penanganan Laporan Kehilangan Dan Temuan Barang Tercecer. *Desain Komunikasi Visual, Manajemen Desain Dan Periklanan (Demandia)*, 3(02), 219. <https://doi.org/10.25124/demandia.v3i02.1549>

## **LAMPIRAN**