

**PENGEMBANGAN APLIKASI *MOBILE HEALTH, SAFETY,*
AND ENVIRONMENT MENGGUNAKAN
IONIC FRAMEWORK**



Disusun Oleh:

N a m a : Arya Andhika Yudhistiro
NIM : 21523097

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2025**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN APLIKASI *MOBILE HEALTH, SAFETY,*
AND ENVIRONMENT MENGGUNAKAN
IONIC FRAMEWORK**

TUGAS AKHIR JALUR MAGANG



Yogyakarta, 15 Juli 2025

Pembimbing,

Dr. Nur Wijayaning Rahayu, S.Kom., M.Cs.

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN APLIKASI *MOBILE HEALTH, SAFETY,*
AND ENVIRONMENT MENGGUNAKAN
IONIC FRAMEWORK**

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, 6 Agustus 2025

Tim Penguji

Ketua Penguji

Dr. Nur Wijayaning Rahayu, S.Kom., M.Cs.

Anggota 1

Erika Ramadhani, S.T., M.Eng.

Anggota 2

Sri Mulyati, S.Kom., M.Kom.

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 : Arya Andhika Yudhistiro
NIM : 21523097

Tugas akhir dengan judul:

PENGEMBANGAN APLIKASI *MOBILE HEALTH, SAFETY,* *AND ENVIRONMENT* MENGGUNAKAN IONIC FRAMEWORK

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 apapun.

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

Yogyakarta, 15 Juli 2025



A handwritten signature in black ink, appearing to be "Arya Andhika Yudhistiro".

Arya Andhika Yudhistiro

HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, Tuhan Yang Maha Pengasih lagi Maha Penyayang atas berbagai karunia-Nya. Terima kasih ku panjatkan atas ridho orang tua karena ridho Allah tak luput dari ridho mereka. Yang tak kalah penting, terima kasih kepada dosen pembimbing dan pihak PT Nocola IoT Solution yang telah memberikan bimbingan berharga dalam penyelesaian laporan akhir ini.

HALAMAN MOTO

“Dan janganlah kamu berputus asa dari rahmat Allah. Sesungguhnya tiada berputus asa dari rahmat Allah, kecuali kaum kafir.” (Q.S. Yusuf ayat 87)

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan hidayat-Nya sehingga penulis dapat menyelesaikan penulisan tugas akhir skripsi ini dengan judul “Pengembangan Aplikasi Mobile Health, Safety & Environment Menggunakan Ionic Framework. Skripsi ini disusun ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknologi Industri Universitas Islam Indonesia. Penulisan skripsi ini tidak lepas dari bantuan berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika Program Sarjana Fakultas Teknologi Industri Universitas Islam Indonesia.
2. Ibu Dr. Novi Setiani, S.T., M.T., selaku Dosen Pembimbing Akademik karena telah membantu dan mengarahkan dalam kebutuhan administrasi dalam perkuliahan.
3. Ibu Dr. Nur Wijyaning Rahayu, S.Kom., M.Cs. selaku Dosen Pembimbing Tugas Akhir karena telah memberikan arahan, bimbingan dan ilmu kepada penulis agar laporan tugas akhir ini dapat selesai.
4. Kedua Orang Tua Bapak dan Ibu yang telah selalu memberikan dukungan yang tak terhingga kepada penulis sehingga penulis dapat kuat berjuang dari awal studi hingga akhir dalam menyusun skripsi ini hingga selesai.
5. Cygnus, Ikhwan, dan dava yang telah memberikan dukungan, inspirasi dan waktunya selama perjalanan akademik ini
6. Tegar, dan Alwi sebagai rekan satu panjaluran yang selalu memberikan dukungan dan bantuan dalam menyelesaikan tugas akhir.
7. Teman-teman seperjuangan “Wong Angel” yang telah memberikan dukungan, semangat, dan kebersamaan selama perkuliahan ini.

Yogyakarta, 15 Juli 2025



Arya Andhika Yudhistiro

SARI

Skripsi ini membahas pengembangan aplikasi *mobile* Health, Safety, & Environment (HSE) menggunakan Ionic Framework yang bertujuan untuk mendukung pelaporan dan pemantauan keselamatan kerja secara digital. Aplikasi yang dikembangkan adalah S2P-Mobile, yang dapat berjalan pada platform Android dan iOS dengan satu basis kode yang sama, serta dapat beroperasi secara *offline* dengan bantuan SQLite.

Pengembangan aplikasi ini menggunakan metode *Agile Development* untuk memungkinkan iterasi cepat dan penyesuaian berkelanjutan sesuai kebutuhan pengguna. Fitur utama yang dikembangkan meliputi halaman Profil pengguna, fitur ganti kata santi pembaca RFID berbasis NFC dan halaman informasi perusahaan. Integrasi API dilakukan untuk memastikan data dapat diperbarui secara *real-time*, sedangkan penyimpanan lokal mendukung fungsi *offline*.

Hasil pengujian aplikasi menunjukkan tingkat keberhasilan tinggi dengan Sebagian besar fitur berjalan sesuai spesifikasi, meskipun tinggi dengan Sebagian besar fitur berjalan sesuai spesifikasi, meskipun masih ada tantangan dalam sinkronisasi data yang perlu diperbaiki pada pengembangan berikutnya. Pengalaman magang di PT Nocola IoT Solution memberikan pembelajaran berharga terkait pengembangan aplikasi *mobile*, khususnya dalam pengelolaan *frontend* menggunakan Ionic Framework.

Kata Kunci: Pengembangan aplikasi *mobile*, Health Safety, & Environment (HSE), Ionic Framework, *Agile Development*, SQLite, RFID.

GLOSARIUM

API	Application Programming Interface, sebuah antarmuka yang memungkinkan aplikasi berkomunikasi dengan layanan lain.
Capacitor	Library pada Ionic Framework untuk mengakses fitur <i>native</i> perangkat seperti kamera, NFC dan penyimpanan lokal.
HSE	Health, Safety <i>and</i> Environment merupakan pendekatan manajemen untuk menjaga Kesehatan dan keselamatan kerja.
K3	Keselamatanm dan Kesehatan kerja merupakan pendekatan manajemen untuk menjaga Kesehatan dan keselamatan kerja.
Ionic Framework	Framework open source yang digunakan untuk membangun aplikasi <i>mobile cross-platform</i> dengan HTML, CSS, dan JavaScript
NFC	Near Field Communication merupakan teknologi komunikasi jarak dekat yang digunakan untuk membaca tag RFID.
QR Code	Quick Response Code merupakan kode matriks dua dimensi untuk menyimpan informasi yang dapat dipindai menggunakan kamera.
SQLite	Sistem manajemen basis data relasional yang ringan dan digunakan untuk penyimpanan local pada aplikasi <i>mobile</i> .

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.....	viii
GLOSARIUM	ix
DAFTAR ISI.....	x
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	4
BAB II DASAR TEORI.....	5
2.1 Aplikasi Mobile	5
2.2 Cross-platform Framework.....	5
2.3 Health, Safety & Environment	6
2.4 Angular Framework.....	7
2.5 Ionic Framework.....	7
2.6 Agile Development.....	9
2.7 Capacitor.....	10
2.8 Tinjauan Pustaka.....	10
BAB III PELAKSANAAN MAGANG	14

3.1 Manajemen Proyek	14
3.1.1 Perkenalan Lingkungan Perusahaan.....	14
3.1.2 Instalasi Ionic Framework.....	15
3.1.3 Pelatihan Ionic Framework	17
3.1.4 Pengembangan Aplikasi S2P-Mobile.....	19
3.2 Hasil.....	26
3.2.1 Halaman Profil	29
3.2.2 Halaman Halaman Ganti Password.....	31
3.2.3 Halaman Pembaca RFID	32
3.2.4 Halaman Tentang Kami	34
3.3 Pembahasan	36
BAB IV REFLEKSI PELAKSANAAN MAGANG	38
4.1 Relevansi Akademik.....	38
4.2 Pembelajaran Magang	39
4.2.1 Manfaat.....	40
4.2.2 Kendala, Hambatan dan Tantangan	40
BAB V PENUTUP.....	42
5.1 Kesimpulan	42
5.2 Saran	43
DAFTAR PUSTAKA	44
LAMPIRAN	48

DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka.....	11
Tabel 3.1 Aktivitas Magang.....	14
Tabel 3.2 Daftar Penugasan pada Proyek S2P-Mobile.....	21
Tabel 4.1 Perbandingan pembelajaran.....	38

DAFTAR GAMBAR

Gambar 2.1 Agile Development	9
Gambar 3.1 Halaman <i>Template</i> Memulai Proyek.....	16
Gambar 3.2 Halaman Detail Aplikasi Pokedex	18
Gambar 3.3 Desain Figma Halaman Profil.....	22
Gambar 3.4 Desain Figma Halaman Ganti Password.....	23
Gambar 3. 5 Tampilan Figma Halaman Cek RFID	23
Gambar 3. 6 Tampilan Figma Halaman Tentang Kami.....	24
Gambar 3.7 Pengujian Fungsional Pada Simulator Menggunakan Xcode	25
Gambar 3.8 Pengujian Fungsional Pada <i>Real-Device</i> Menggunakan Xcode	25
Gambar 3.9 Kode Program untuk Menampilkan Judul Halaman Profil.....	26
Gambar 3.10 Kode Program untuk konten halaman.....	27
Gambar 3.11 Kode Program unntuk Menampilkan informasi Profil	27
Gambar 3.12 Kode Program untuk Komponen Foto Profil.....	28
Gambar 3.13 Kode Program untuk List Daftar Menu	28
Gambar 3.14 Kode Program untuk Komponen Modal.....	29
Gambar 3.15 Tampilan Halaman Profil.....	29
Gambar 3.16 Kode Program untuk Menampilkan Data Pengguna.....	30
Gambar 3.17 Kode Program untuk Layanan SharedService	30
Gambar 3.18 Kode Program untuk Sinkronisasi SQLite.....	30
Gambar 3.19 Tampilan Halaman Ganti Password.....	31
Gambar 3.20 Kode Program untuk Fungsi Ganti Password.....	31
Gambar 3.21 Kode Program untuk Menu Ganti Sandi Akun.....	32
Gambar 3.22 Tampilan Halaman Cek RFID	32
Gambar 3.23 Kode Program untuk Fungsi Pembaca NFC	33
Gambar 3.24 Kode Program Tombol Navigasi Menu Pembaca RFID	33
Gambar 3.25 Tampilan Halaman Tentang Kami.....	34
Gambar 3.26 Kode Program untuk Menampilkan Peta	35
Gambar 3.27 Kode Program untuk Menu Tentang Kami.....	35

BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam lingkungan industri modern yang dinamis dan penuh kompleksitas, penerapan prinsip Health, Safety, & Environment (HSE) menjadi hal yang tidak hanya menjamin keselamatan dan kesehatan para pekerja, tetapi juga cukup penting untuk menjaga keberlangsungan dan reputasi operasional perusahaan. Seiring dengan meningkatnya kesadaran akan risiko kerja serta diperketatnya standar regulasi keselamatan oleh pihak yang berwenang, perusahaan kini dituntut untuk beralih dari metode tradisional ke pengelolaan aspek HSE yang lebih sistematis, efisien, dan berbasis data untuk pengambilan keputusan yang lebih tepat.

Pentingnya peran HSE sebagai faktor cukup penting karena penerapan HSE yang baik terbukti secara signifikan mampu mengurangi angka kecelakaan kerja dan mendukung kelancaran aktivitas operasional perusahaan (A. D. Putra et al., 2021). Hal ini menegaskan bahwa investasi dalam sistem keselamatan kerja bukan lagi sekadar biaya untuk pemenuhan regulasi, melainkan sebuah investasi strategis yang berdampak langsung pada reputasi perusahaan yang bisa berpengaruh kepada keberlanjutan bisnis karena meningkatnya hasil produksi.

Namun dalam praktiknya, banyak perusahaan masih menghadapi tantangan dalam implementasi HSE. Pelaksanaan dan pelaporan yang masih bergantung pada metode manual sering kali menjadi sumber kendala, seperti keterlambatan informasi insiden, potensi kesalahan pencatatan data oleh manusia (*human error*) dan kurangnya akses data secara *real-time* oleh manajemen untuk pengambilan keputusan cepat. Keterbatasan ini sering kali berujung pada respon yang lambat terhadap insiden yang menyebabkan kesulitan dalam menganalisis tren kecelakaan untuk tindakan pencegahan dan budaya kerja yang cenderung reaktif daripada proaktif.

Berangkat dari permasalahan tersebut serta untuk mendorong transformasi digital di lingkungan kerja, PT Sumber Segara Primadaya bekerja sama dengan PT Nocola IoT Solution untuk mengembangkan aplikasi *mobile* bernama S2P-Mobile. Aplikasi ini dirancang secara khusus untuk mendukung implementasi HSE secara digital dan terintegrasi. Fitur-fitur utamanya mencakup pelaporan kondisi dan insiden keselamatan kerja secara *real-time* dari lapangan, penyampaian notifikasi darurat yang cepat kepada pihak terkait, serta pemantauan status dan kepatuhan pekerja terhadap prosedur keselamatan yang berlaku.

Pengembangan aplikasi S2P-Mobile ini juga sejalan dengan tren transformasi digital yang lebih luas, khususnya dalam konteks revolusi industri 4.0. Pada era ini, integrasi teknologi seperti Internet of Things (IoT), analisis data dan komputasi *mobile* tidak lagi menjadi pilihan, melainkan sebuah keharusan bagi perusahaan untuk mempertahankan daya saing dan keunggulan operasional. Aplikasi S2P-Mobile merupakan salah satu manifestasi dari penerapan prinsip industri 4.0 pada bidang HSE, di mana data yang dikumpulkan secara *real-time* dari perangkat *mobile* dapat menjadi dasar untuk analisis prediktif guna mencegah kecelakaan sebelum terjadi, bukan sekedar melaporkannya.

Untuk mencapai tujuan tersebut, aplikasi S2P-Mobile dikembangkan menggunakan Ionic Framework didasarkan pada kemampuannya untuk pengembangan aplikasi lintas platform (*cross-platform*) yang memungkinkan aplikasi untuk berjalan di sistem operasi Android dan iOS hanya dengan menggunakan satu basis kode, sehingga menghemat waktu dan sumber daya. Selain itu, untuk mengatasi tantangan konektivitas di lingkungan industri yang terkadang tidak stabil, aplikasi ini diintegrasikan dengan SQLite yang mendukung penggunaan data secara *offline*. Fitur ini memastikan data tetap dapat diakses dan dicatat oleh pekerja di area dengan koneksi internet terbatas, yang selanjutnya disinkronkan saat terhubung kembali dengan jaringan.

Melalui pengembangan aplikasi ini, diharapkan perusahaan dapat secara signifikan meningkatkan efektivitas dan efisiensi dalam pengelolaan HSE. Lebih dari itu, aplikasi S2P-Mobile diharapkan dapat menjadi solusi dalam membangun ekosistem keselamatan kerja yang terintegrasi, di mana data dari lapangan dapat dianalisis untuk pengambilan keputusan yang lebih baik, sehingga mampu mendorong dan memperkuat budaya kerja yang lebih aman, tanggap dan berbasis teknologi di perusahaan.

1.2 Ruang Lingkup

Proyek ini mencakup pengembangan aplikasi *mobile* S2P-Mobile yang bertujuan untuk mendukung pengelolaan HSE di PT Sumber Segara Primadaya. Ruang lingkup proyek ini adalah mengembangkan aplikasi *mobile* untuk platform Android dan iOS dengan menggunakan Ionic Framework.

Proyek ini tidak melibatkan integrasi langsung dengan sistem *back-end* perusahaan atau pengelolaan data tingkat lanjut, melainkan fokus pada pengembangan antarmuka pengguna dan fungsionalitas dasar aplikasi yang mendukung HSE.

1.3 Tujuan

Tujuan dari laporan tugas akhir ini adalah untuk:

- a. Mengembangkan aplikasi *mobile* S2P-Mobile menggunakan Ionic Framework untuk meningkatkan keselamatan pekerja.
- b. Membuat aplikasi yang mudah digunakan, responsif dan dapat berjalan di perangkat Android dan iOS dengan satu basis kode yang sama.
- c. Meningkatkan efisiensi pemantauan keselamatan kerja dengan memanfaatkan teknologi *mobile* dan SQLite untuk pengelolaan data secara lokal.

1.4 Manfaat

Manfaat yang diharapkan dari pengembangan aplikasi S2P-Mobile ini antara lain:

- a. Peningkatan efisiensi dalam memantau dan melaporkan kondisi HSE di perusahaan secara *real-time*.
- b. Peningkatan keselamatan bagi karyawan dengan adanya fitur notifikasi darurat dan pemantauan status HSE.
- c. Kemudahan akses bagi pekerja yang berada di lokasi dengan keterbatasan koneksi internet berkat fitur *offline-first* menggunakan SQLite.

Pengembangan aplikasi ini tidak hanya membantu perusahaan dalam mengelola keselamatan kerja secara digital, tetapi juga memperkenalkan teknologi baru yang mempermudah pengelolaan data HSE.

1.5 Sistematika Penulisan

Laporan tugas akhir ini disusun secara sistematis agar memudahkan pembaca dalam memahami proses pengembangan aplikasi *mobile* berbasis Health, Safety & Environment (HSE) menggunakan Ionic Framework. Adapun sistematika penulisan laporan ini adalah sebagai berikut:

- a. BAB I membahas pendahuluan, latar belakang, ruang lingkup, tujuan, manfaat, serta sistematika penulisan dari laporan tugas akhir. Penjelasan ini memberikan gambaran umum mengenai alasan pemilihan topik dan cakupan proyek magang.
- b. BAB II membahas landasan teori dan tinjauan pustaka, Bab ini membahas teori-teori yang relevan dengan pengembangan aplikasi *mobile*, termasuk pemahaman tentang Ionic Framework, SQLite serta konsep dasar Health, Safety, & Environment (HSE). Bab ini juga memuat tinjauan pustaka dari penelitian atau proyek sejenis yang pernah dilakukan sebelumnya.
- c. BAB III membahas pelaksanaan magang, berisi penjelasan mengenai proses pelaksanaan magang di PT. Nocola IoT Solution, aktivitas-aktivitas yang dilakukan selama magang, serta kontribusi nyata dalam proyek pengembangan aplikasi S2P-Mobile.
- d. BAB IV membahas refleksi pelaksanaan magang, relevansi antara teori yang telah dipelajari di bangku perkuliahan dengan praktik nyata di lapangan berdasarkan hasil pelaksanaan magang.
- e. BAB V membahas penutupan, Berisi kesimpulan dari kegiatan magang yang telah dilakukan serta berisi saran yang bersifat membangun untuk penelitian selanjutnya.

BAB II

DASAR TEORI

2.1 Aplikasi Mobile

Aplikasi *mobile* adalah perangkat lunak yang dirancang untuk dijalankan pada perangkat selular seperti *smartphone* dan tablet. Aplikasi ini memberikan kemudahan akses layanan dan informasi, baik secara *online* maupun *offline*, serta dirancang agar efisien dalam penggunaan sumber daya perangkat. Dengan pertumbuhan pengguna perangkat selular, aplikasi *mobile* menjadi sarana penting dalam berbagai bidang seperti edukasi, bisnis, kesehatan, dan hiburan.

Penelitian (C. G. & Devi, 2021) memberikan gambaran bahwa industri aplikasi *mobile* terus mengalami pertumbuhan pesat, didorong oleh dominasi platform Android dan iOS. Studi tersebut menyoroti bagaimana aplikasi *mobile* telah menjadi bagian penting dari kehidupan sehari-hari dan strategi bisnis modern. Oleh karena itu, pengembangan aplikasi *mobile* tidak lagi hanya menjadi sebuah opsi teknologi melainkan sebuah kebutuhan strategis yang sangat berguna di masa kini.

Menurut (Y. W. S. Putra et al., 2023) aplikasi *mobile* memiliki karakteristik utama berupa antarmuka yang intuitif, responsif, dan mampu beradaptasi dengan ukuran layar perangkat. Pengembangan aplikasi *mobile* dapat dilakukan dengan pendekatan *native* yang dibangun khusus untuk platform tertentu, atau *hybrid* yang memungkinkan pengembangan aplikasi dengan satu basis kode yang dapat dijalankan di berbagai platform secara bersamaan.

2.2 Cross-platform Framework

Penelitian (Pamungkas & Imrona, 2020) mengatakan salah satu tantangan utama dalam pengembangan aplikasi *mobile* adalah keharusan untuk membuat aplikasi yang berbeda untuk platform Android dan iOS. Kedua platform tersebut memiliki bahasa dan logika pemrograman yang tidak sama, sehingga pengembangan memerlukan waktu, usaha dan biaya dua kali lipat untuk satu proyek aplikasi yang sama. Framework *cross-platform* muncul sebagai solusi atas permasalahan tersebut, karena memungkinkan pengembangan untuk membangun aplikasi hanya dengan satu basis kode yang kemudian dapat dijalankan kedua platform.

Menurut (Ilhami, 2021) munculnya *cross-platform framework* memungkinkan sebuah basis kode tunggal dapat digunakan dan di *deploy* ke beberapa platform sekaligus, seperti aplikasi *mobile*, aplikasi web atau PWA (*Progressive Web Apps*) dan aplikasi *desktop*. Pendekatan ini tidak merujuk pada satu bahasa pemrograman atau *tool* spesifik. Beberapa *framework cross-platform* yang sering digunakan adalah PhoneGap, Ionic Framework dan

React Native. Adopsi *framework cross-platform* didukung oleh beberapa faktor seperti perkembangan masif teknologi *website*, meningkatnya jumlah pengguna *smartphone* serta tingginya jumlah *website developer* dibandingkan *native app developer*. Hal ini menjadikan teknologi *cross-platform* berbasis JavaScript sebagai alternatif yang menjanjikan untuk dikuasai oleh para *programmer* baru yang ingin berkarir di industri ini.

2.3 Health, Safety & Environment

Health, Safety & Environment (HSE) merupakan pendekatan manajemen yang fokus pada perlindungan kesehatan dan keselamatan kerja serta lingkungan. Penerapan HSE bertujuan untuk menciptakan lingkungan kerja yang aman, sehat, dan produktif. Namun, penelitian (Purnawati et al., 2023) menyebutkan bahwa kecelakaan kerja masih sering terjadi meskipun beberapa perusahaan telah menerapkan sistem manajemen HSE secara manual maupun digital. Berdasarkan penelitian tersebut perlu adanya dukungan teknologi untuk meningkatkan sistem yang ada, salah satunya melalui perancangan aplikasi digital. Studi tersebut secara spesifik merancang sebuah aplikasi bernama “K3 Agent” dengan antarmuka berbasis Android. Aplikasi ini berfungsi sebagai alat bantu bagi petugas K3 untuk mendokumentasikan dan mengelola berbagai keperluan administrasi HSE, seperti pembuatan analisis, laporan, dan pedoman keselamatan secara lebih efisien.

Seperti pada penelitian oleh (Swatika et al., 2022) penerapan HSE secara signifikan meningkatkan produktivitas karyawan, sebab rasa aman yang timbul karena perusahaan menerapkan HSE sehingga membuat mereka dapat bekerja dengan lebih tenang dan semangat serta mencegah kerugian perusahaan akibat kecelakaan kerja.

Menurut (Cantika & Sofyan, 2024) penerapan HSE secara konsisten mampu meningkatkan kesadaran karyawan terhadap risiko kerja, menurunkan angka kecelakaan, serta mendukung efisiensi operasional perusahaan. Selain itu, budaya keselamatan yang kuat dan audit rutin menjadi kunci keberhasilan implementasi HSE di lingkungan industri.

2.4 Angular Framework

Menurut (Wijaya, 2023) *framework* Angular terbukti secara signifikan meningkatkan performa aplikasi *website* melalui mekanisme *client-side routing* yang membuat interaksi dengan server hanya terjadi pada pemuatan awal. Halaman yang menggunakan *routing* Angular dimuat dalam waktu mendekati 0 detik dengan transfer data sebesar 43,1 KB, jauh lebih efisien dibandingkan tanpa *routing* yang membutuhkan 2,20 detik dengan transfer data sebesar 6,3 MB data. Hal ini menunjukkan implementasi Angular sangat efektif mengurangi beban server dan jumlah data yang ditransfer.

Penelitian (Sinlae et al., 2024) membuktikan bahwa penggunaan *framework* Angular secara signifikan meningkatkan efisiensi pengembangan *website*. Hasil studi kasus dalam penelitian tersebut menunjukkan bahwa Angular mampu mengurangi waktu pengembangan hingga 30% dan menurunkan jumlah bug hingga 27%. Peningkatan efisiensi ini didukung oleh fitur-fitur canggih Angular seperti *two-way data binding* dan *dependency injection* yang mengurangi penulisan kode berulang seperti pada **Error! Reference source not found.**, serta sistem modular yang solid agar mempermudah kolaborasi tim. Selain efisiensi teknis, studi tersebut juga melaporkan tingkat kepuasan pengembang yang tinggi, dimana 85% merasa lebih mudah dalam menambahkan fitur baru dan melakukan pemeliharaan aplikasi. Meskipun memiliki kurva belajar yang cukup tinggi bagi pemula, penelitian tersebut menyimpulkan bahwa Angular merupakan pilihan efektif untuk pengembangan.

2.5 Ionic Framework

Ionic *Framework* adalah platform pengembangan aplikasi *mobile* yang menggunakan HTML, CSS, dan Angular untuk membangun aplikasi *cross-platform* sehingga dapat membangun antarmuka pengguna yang konsisten seperti pada penelitian yang dilakukan oleh (Chaudhary, 2018). Sementara itu, (Singh & G, 2021) menambahkan bahwa Ionic kini juga mendukung integrasi fitur *native* melalui Capacitor yang semakin memperluas fungsionalitas aplikasi *hybrid* secara efisien.

Penelitian (Pamungkas & Imrona, 2020) membandingkan beberapa *cross-platform framework*, yang menyebutkan bahwa Ionic Framework, termasuk dalam kategori *framework* yang menggunakan pendekatan *webview*. Pendekatan ini menjadi pembeda dengan *framework* lain seperti React Native dan Flutter yang berbasis pendekatan *native*, sehingga memberikan pemahaman mengenai arsitektur fundamental yang digunakan Ionic.

Penelitian lain (Syarifudin & Siswanto, 2018) juga menyebutkan bahwa Ionic Framework merupakan sebuah *software development kit* (SDK) yang bersifat *open-source* dan ditunjukkan

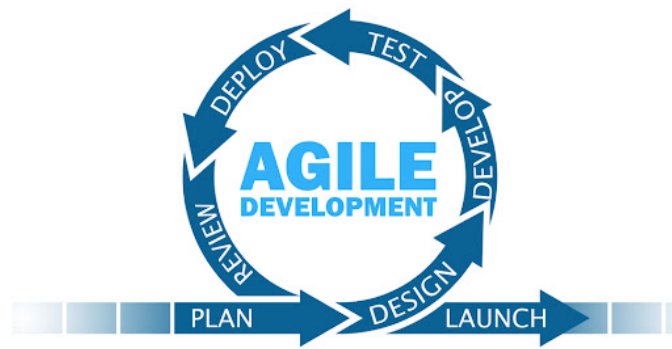
untuk pengembangan aplikasi *mobile hybrid*. Penelitian tersebut juga menjelaskan bahwa Ionic menyediakan semua komponen, peralatan, dan fungsionalitas yang setara dengan yang ada pada SDK pengembangan *native*. Persyaratan untuk melakukan instalasi Ionic harus sudah terpasang NodeJS beserta npm sebagai manajer pakatnya.

Penelitian (Waranashiwar & Ukey, 2018) menyebutkan Ionic Framework menjadi platform yang lebih baik dibandingkan pengembangan aplikasi *native* yang spesifik untuk setiap platform, memakan waktu, dan mahal. Selain karena pengembangan Ionic menggunakan teknologi berbasis *website*, Ionic juga didukung oleh akses langsung ke API perangkat melalui Apache Cordova yang merupakan salah satu dari tiga kerangka kerja kolaboratif Ionic dan Angular.

Penelitian oleh (Zohud & Zein, 2021) menambahkan bahwa Ionic memungkinkan pengembangan untuk membuat aplikasi yang dapat berjalan di iOS dan Android dengan satu basis kode sehingga dapat menghemat waktu dan biaya pengembangan. Selain itu, penelitian yang dilakukan oleh (Asyahari et al., 2025) menunjukkan bahwa implementasi Ionic dan Capacitor pada aplikasi mobile berhasil meningkatkan efisiensi pengelolaan data dan aksesibilitas aplikasi karena dapat dijalankan di berbagai perangkat dengan satu basis kode.

Penelitian (Efendi & Ubaidi, 2020) mengatakan bahwa Ionic Framework merupakan *framework front-end* untuk membuat aplikasi seluler *hybrid*. Penelitian tersebut menjelaskan bahwa Ionic Framework memanfaatkan AngularJS dan HTML5 sebagai basis teknologi untuk menciptakan aplikasi.

2.6 Agile Development



Gambar 2.1 Agile Development

Sumber: (kompasiana, 2023)

Menurut (Fahrudin & Ilyasa, 2021) metode agile adalah sebuah pendekatan pengembangan perangkat lunak di mana tahapan-tahapannya dapat diulang Kembali hingga membentuk sebuah siklus seperti pada Gambar 2.1. Pengulangan ini dilakukan karena metode agile sangat berfokus pada nilai sebuah produk yang dikembangkan. Pendekatan ini juga dinilai sangat responsif dan adaptif terhadap perubahan. Oleh karena itu, metode agile sering kali dikombinasikan dengan metode lain seperti *Design Thinking*, di mana agile berperan untuk membantu mengimplementasikan ide dan solusi yang dihasilkan agar produk yang dibangun tetap sesuai dengan kebutuhan pengguna yang bisa berubah seiring waktu.

Penelitian (Lutfi Wirananda Silvana et al., 2024) meneliti lebih lanjut tahapan-tahapan yang ada dalam siklus metode agile. Prosesnya diawali dengan tahap perencanaan untuk menganalisis kebutuhan dan merancang solusi, dilanjutkan dengan desain arsitektur sistem dan basis data. Setelah itu, masuk ke tahap pengembangan di mana proses koding dilakukan, yang kemudian akan dievaluasi pada tahap pengujian untuk memastikan fungsionalitasnya berjalan sesuai harapan. Tahap selanjutnya adalah *deploy* ke server, diikuti dengan review untuk memeriksa kesesuaian hasil akhir, sebelum akhirnya aplikasi diluncurkan agar dapat diakses oleh publik. Rangkaian tahapan ini menunjukkan bagaimana metode Agile memecahkan proyek besar menjadi fase yang terstruktur namun tetap fleksibel.

Penelitian (Candra Pamungkas et al., 2023) penggunaan metode *Agile Development* bertujuan untuk membuat proses pengembangan perangkat lunak menjadi lebih terstruktur, terarah, dan tetap adaptif terhadap perubahan. Dalam pendekatan Agile, proses pengembangan yang berulang dan bertahap tersebut diorganisir ke dalam siklus kerja yang disebut *sprint*, di mana setiap *sprint* memiliki tujuan yang jelas untuk menghasilkan bagian produk yang fungsional.

2.7 Capacitor

Untuk menjembatani antara aplikasi berbasis *website* dengan fungsionalitas perangkat keras Ionic Framework mengandalkan Capacitor yang merupakan *native runtime open-source* yang dikembangkan langsung oleh tim Ionic untuk menggantikan teknologi lama seperti Cordova. Menurut (Tahvanainen, 2023) Capacitor memungkinkan aplikasi untuk mengakses berbagai fitur asli perangkat (*native feature*) seperti sistem *file*, kamera, dan layanan lokasi melalui *Application Programming Interface* (API) Typescript yang terpadu.

Salah satu keunggulan utama Capacitor dibandingkan pendahulunya adalah cara pengelolaannya terhadap proyek *native* (Android Studio dan Xcode) yang dianggap sebagai aset sumber. Hal ini memberikan kemudahan bagi pengembang untuk melakukan kustomisasi kode *native* jika diperlukan.

Selain itu, Capacitor juga memiliki sistem manajemen *plugin* yang dinilai tidak rentan terhadap kesalahan, yang terdiri dari *plugin* resmi, *plugin* komunitas, dan *plugin* premium untuk kebutuhan perusahaan.

2.8 Tinjauan Pustaka

Beberapa penelitian telah dilakukan untuk menerapkan teknologi dalam meningkatkan sistem Health, Safety, and Environment (HSE) di berbagai industri. Fokus utamanya adalah mendigitalisasi proses yang sebelumnya manual untuk meningkatkan efisiensi, akurasi, dan kecepatan pelaporan serta analisis. Upaya digitalisasi ini dapat dikelompokkan ke dalam beberapa pendekatan utama.

Fokus pertama adalah pengembangan aplikasi untuk pelaporan dan manajemen data HSE secara terpusat pada aplikasi S2P-Mobile. Pendekatan ini banyak diimplementasikan pada platform berbasis *website* yang dapat diakses melalui *desktop*. Penelitian oleh (Irawan et al., 2025) merancang sistem informasi HSE untuk mengelola data secara terpusat di PT Multi Jaya Teknik yang sebelumnya masih mengandalkan proses manual. Serupa dengan itu, (Ningtyas et al., 2024) juga mengembangkan sistem informasi berbasis *website* untuk pelaporan statistik HSE yang bertujuan mempermudah analisis data insiden. Penelitian (Nuristi & Farell, 2022) secara spesifik merancang sistem informasi untuk proses HIRADC (*Hazard Identification Risk Assessment & Determining Control*), yang merupakan inti dari manajemen risiko HSE. Di lingkungan pelayanan Kesehatan, (Askar et al., 2021) membangun sistem pelaporan berbasis *website* untuk kegiatan HSE di puskesmas, yang terbukti meningkatkan efektivitas dan efisiensi pelaporan.

Fokus kedua adalah pengembangan aplikasi untuk manajemen prosedur keselamatan yang lebih spesifik di lapangan. Dengan mengimplementasikan aplikasi berbasis *mobile*, dapat menjadi platform yang mudah diakses di lapangan. (Saputra et al., 2021) secara spesifik merancang aplikasi *checklist safety* berbasis *mobile* dengan metode Job Safety Analysis (JSA) untuk mempermudah proses audit K3 di lingkungan ritel. Sementara itu, (Sihombing et al., 2021) mengembangkan aplikasi HSE di PT Altrak 1978 yang mencakup manajemen data APD dan limbah B3 (Bahan Berbahaya dan Beracun). Dalam konteks yang lebih spesifik, (Sukoco & N, 2023) menganalisis efektivitas aplikasi “Inspekta” di lingkungan PT PLN, yang menunjukkan bahwa aplikasi *mobile* dapat secara signifikan memotivasi karyawan untuk melaporkan isu-isu HSE. (Purnawati et al., 2023) juga menyoroti pentingnya dukungan teknologi dalam manajemen HSE. Meskipun sistem HSE sudah ada, kecelakaan kerja masih sering terjadi, sehingga diperlukan aplikasi digital yang lebih praktis. Penelitian tersebut secara spesifik merancang sebuah aplikasi bernama “K3 Agent” dengan antarmuka berbasis Android yang berfungsi sebagai alat bantu bagi petugas HSE untuk mendokumentasikan dan mengelola berbagai keperluan administrasi HSE, seperti pembuatan analisis, laporan, dan pedoman keselamatan secara lebih efisien.

Secara keseluruhan, tinjauan Pustaka ini menunjukkan peningkatan pemanfaatan teknologi baik web maupun *mobile* untuk menciptakan sistem HSE yang lebih terstruktur, responsif dan berbasis data. Berikut perbandingan dari beberapa penelitian terdahulu pada Tabel 2.1.

Tabel 2.1 Tinjauan Pustaka

No	Sumber	Judul	Tujuan	Platform Aplikasi	Hasil
1.	(Purnawati et al., 2023)	Design of Health, Safety, and Environment Digital Application with Android-Based User Interface	Merancang aplikasi digital untuk membantu pengelolaan HSE.	Aplikasi Mobile (Android)	Berhasil menciptakan prototipe aplikasi "K3 Agent" yang fungsional untuk analisis dan pelaporan HSE.
2.	(Irawan et al., 2025)	Perancangan Sistem Informasi Keselamatan dan Kesehatan Kerja (K3) di PT. Multi Jaya Teknik	Mengatasi pengelolaan data HSE yang masih manual dengan sistem terkomputerisasi.	Website (Desktop)	Menghasilkan rancangan sistem informasi HSE yang dapat mengelola data insiden, APD, dan pelatihan secara efektif.

3.	(Saputra et al., 2021)	Aplikasi Checklist Safety Berbasis Mobile Menggunakan Metode Job Safety Analysis (JSA)	Menganalisis kebutuhan dan membuat aplikasi <i>checklist</i> HSE untuk mempermudah proses audit.	Aplikasi Mobile (Android)	Berhasil merancang aplikasi <i>checklist</i> HSE berbasis <i>mobile</i> sebagai alat audit di lingkungan perusahaan.
4.	(Nuristi & Farell, 2022)	Rancang Bangun Sistem Informasi HIRADC	Membantu perusahaan mengelola data bahaya dan risiko secara terstruktur dan efisien.	Website	Menghasilkan rancangan sistem informasi untuk proses HIRADC (<i>Hazard Identification Risk Assessment & Determining Control</i>).
5.	(Askar et al., 2021)	Sistem Pelaporan Berbasis Web Pada Pelaksanaan Kegiatan Keselamatan dan Kesehatan Kerja (K3) di Puskesmas	Membuat sistem pelaporan kegiatan HSE yang lebih efektif dan efisien.	Website (Responsif)	Sistem pelaporan berbasis web yang dihasilkan terbukti dapat meningkatkan efektivitas dan efisiensi serta memuaskan pengguna.
6.	(Sihombing et al., 2021)	Aplikasi Health Safety Environment Pada PT Altrak 1978 Samarinda	Mengelola data HSE seperti data karyawan, APD, dan limbah B3 (Bahan Beracun dan Berbahaya) secara digital.	Aplikasi Web	Menghasilkan aplikasi HSE yang fungsional dan telah divalidasi dengan metode pengujian <i>blackbox</i> .
7.	(Ningtyas et al., 2024)	Perancangan Sistem Informasi Pelaporan Statistik K3 Berbasis Website	Merancang sistem informasi untuk mempermudah pelaporan dan analisis data statistik HSE.	Website	Menghasilkan sistem informasi yang mampu mengelola dan melaporkan data statistik HSE secara digital.
8.	(Sukoco & N, 2023)	Efektifitas Aplikasi Inspekta dalam Peningkatan Budaya K3 di PLN ULP Kisaran	Menganalisis efektivitas aplikasi "Inspekta" dalam meningkatkan budaya HSE.	Aplikasi Mobile	Aplikasi Inspekta terbukti efektif dalam memotivasi karyawan untuk melaporkan isu HSE dan membantu mengatasi risiko lebih cepat.
9.	(Revaldo & Putra, 2025)	Evaluasi Kepuasan Pengguna Aplikasi Inspekta di PT. PLN UP2D S2JB	Mengevaluasi tingkat kepuasan pengguna terhadap aplikasi "Inspekta" di	Website	Aplikasi Inspekta dinilai "puas" oleh pengguna, terutama pada aspek efisiensi, namun perlu peningkatan

		Menggunakan Metode Pieces	lingkungan PT PLN.		pada keamanan dan dukungan teknis.
--	--	------------------------------	-----------------------	--	---------------------------------------

BAB III

PELAKSANAAN MAGANG

3.1 Manajemen Proyek

Kegiatan magang pada PT. Nocola IoT Solution selama enam bulan dimulai dari 15 September 2024 hingga 16 Maret 2025. Aktivitas magang yang dilaksanakan di PT. Nocola IoT Solution dapat dilihat pada Tabel 3.1. Tabel tersebut memberikan gambaran mengenai berbagai kegiatan yang dilakukan selama magang berlangsung. Kegiatan tersebut mencakup tugas-tugas yang diberikan, proyek yang dikerjakan serta tanggung jawab selama magang di PT Nocola IoT Solution.

Tabel 3.1 Aktivitas Magang

No.	Deskripsi Kegiatan	Waktu Kegiatan	Durasi
1.	Perkenalan Lingkungan Perusahaan	17 September 2024	1 hari
2.	<i>Installation</i> Ionic Framework	18 September 2024	1 hari
3.	<i>Trainning</i> Ionic Framework	19 September – 24 Oktober 2024	26 hari
4.	Pengembangan Aplikasi S2P-Mobile bersama tim	25 Oktober 2024 – 4 Januari 2025	51 hari

3.1.1 Perkenalan Lingkungan Perusahaan

Nocola atau PT. Nocola IoT Solution adalah perusahaan pengembangan Internet of Things asal Indonesia yang didirikan di Cilacap pada tahun 2016. Perusahaan ini memiliki tim profesional yang handal dan terlatih dengan pengetahuan serta keterampilan yang kompeten di berbagai bidang.

Tidak hanya berfokus pada IoT, Nocola juga menjalin kerja sama strategis dengan mitra teknologi, baik di tingkat nasional, regional, maupun global, guna memperkuat keunggulan produk yang dihasilkan. Bidang utama yang dikerjakan mencakup integrasi sistem, pengembangan IoT, pengembangan perangkat lunak, layanan navigasi dan layanan komunikasi jaringan. Selain itu, Nocola menyediakan layanan desain, implementasi, dan manajemen proyek sesuai kebutuhan pelanggan.

Kantor pusat nocola berlokasi di Jl. Ir. Djuanda No. 117, Sidanegara, Cilacap, Jawa Tengah, sementara workshop-nya berada di Jl. Lengkong No. 160, Mertasinga, Cilacap Utara, Cilacap, Jawa Tengah.

Pada tahap awal magang, penulis diperkenalkan dengan lingkungan kerja, ruangan serta staf yang ada di PT. Nocola IoT Solution oleh Bapak Karier Susastra selaku *Human Resource Development* (HRD). Pengenalan ini mencakup informasi tentang divisi-divisi perusahaan, peraturan serta budaya kerja yang bertujuan untuk memudahkan penulis beradaptasi dengan lingkungan perusahaan.

Sebagai bagian dari standar profesionalisme dan etika kerja, penulis di tahap awal diwajibkan untuk menandatangani dokumen *Non-Disclosure Agreement* (NDA). Penandatanganan dokumen ini merupakan bentuk komitmen formal penulis untuk menjaga kerahasiaan seluruh informasi sensitif, baik yang berkaitan dengan strategi internal perusahaan maupun data spesifik dari proyek yang dikerjakan bersama klien. Prosedur ini menekankan pentingnya integritas dan tanggung jawab profesional dalam mengelola informasi rahasia selama dan setelah periode magang.

Setelah menyelesaikan proses administrasi dengan HRD (*Human Resource Development*), penulis kemudian dipertemukan kepada Hamzah Alvana selaku *Supervisor Mobile Developer* sekaligus menjadi pembimbing lapangan selama periode magang. Pada sesi pengenalan teknis ini, penulis mendapatkan pengarahan awal mengenai proyek S2P-Mobile. Penjelasan tersebut meliputi latar belakang proyek, tujuan yang ingin dicapai, status pengembangan terkini, serta teknologi yang digunakan. Melalui pengarahan tersebut, penulis memperoleh pemahaman yang komprehensif mengenai lingkup pekerjaan, tanggung jawab dan target yang telah ditentukan.

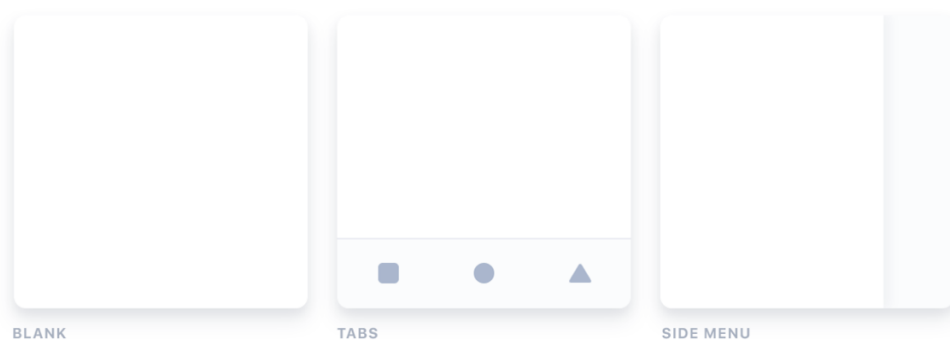
3.1.2 Instalasi Ionic Framework

Pada awal masa magang, penulis diperkenalkan pada *framework* dan teknologi yang digunakan oleh perusahaan oleh Hamzah Alvana selaku supervisor pengembangan aplikasi *mobile* yang sekaligus menjadi mentor selama program magang berlangsung. Salah satu yang dipelajari adalah *framework* Ionic yang bertujuan untuk membantu penulis memahami dan menyesuaikan diri dengan teknologi yang diterapkan pada proyek perusahaan untuk klien mereka. Sebagai bagian dari pembelajaran, mentor memberikan dokumen dalam format PDF kepada penulis untuk dipelajari. Selain itu, penulis juga disarankan untuk mempelajari dokumentasi *framework* Ionic yang tersedia di situs *website* resminya pada alamat <https://ionicframework.com/docs/cli>.

Pada langkah awal, penulis melakukan instalasi Ionic Framework pada perangkat penulis dimulai dengan memastikan persyaratan sistem telah terpenuhi, seperti Node.js, npm, dan Git.

Node.js merupakan *runtime* JavaScript yang menjadi dasar pengoperasian Ionic Framework, sedangkan npm digunakan sebagai modul. Untuk memeriksa apakah Node.js dan npm telah terinstal, penulis menggunakan `node -version` dan `npm -version` pada terminal. Setelah Node.js dan npm terverifikasi, langkah berikutnya adalah menginstal Ionic CLI, yang merupakan antarmuka baris perintah untuk mengelola proyek Ionic. Instalasi dilakukan dengan menjalankan perintah `npm install -g @ionic/cli`. Setelah instalasi selesai, penulis dapat memverifikasi bahwa Ionic CLI telah terpasang dengan benar menggunakan perintah `ionic -version`.

Setelah itu, penulis dapat membuat proyek Ionic baru menggunakan perintah `ionic start nama-proyek blank` atau `ionic start nama-proyek tabs` atau `ionic start nama-proyek sidemenu`. Perbedaan perintah ini akan mempengaruhi jenis halaman yang dibangun secara otomatis sesuai *template* yang dipilih, seperti yang ditampilkan pada Gambar 3.1 berikut:



Gambar 3.1 Halaman *Template* Memulai Proyek

Setelah proyek dibuat, penulis masuk ke dalam direktori proyek menggunakan perintah `CD nama-proyek`. Untuk menjalankan aplikasi Ionic dalam mode pengembangan, penulis menggunakan perintah `ionic cap run ios -liveroad` untuk pengembangan aplikasi pada simulator perangkat iOS. Sebelum menjalankan perintah pada perangkat iOS, penulis perlu memastikan Xcode telah terinstal pada perangkat pengembang dan tidak lupa menambahkan platform iOS pada proyek yang akan dikembangkan menggunakan perintah `ionic capacitor add ios`.

3.1.3 Pelatihan Ionic Framework

Aplikasi pertama yang pemegang buat bernama “Pokedex” yang merupakan sebuah aplikasi berbasis *mobile* yang dirancang untuk menampilkan data rinci tentang berbagai jenis Pokemon. Aplikasi ini terintegrasi dengan API publik bernama PokeApi, sebuah RESTful API yang menyediakan basis data sangat lengkap dan mendetail mengenai informasi Pokemon. Dalam aplikasi ini, terdapat sejumlah fitur utama yang dirancang untuk membuat pengalaman pengguna menjadi lebih interaktif dan informatif.

Salah satu fitur utama aplikasi ini adalah *dashboard*, yang menampilkan berbagai jenis Pokemon dalam antarmuka yang menarik dan mudah digunakan. *Dashboard* ini memudahkan pengguna untuk melihat daftar Pokemon secara keseluruhan. Selain itu, tersedia fitur pencarian yang memungkinkan pengguna mencari Pokemon berdasarkan nama. Fitur ini membantu pengguna menemukan Pokemon yang diinginkan dengan cepat tanpa perlu menelusuri daftar pokemon secara manual.

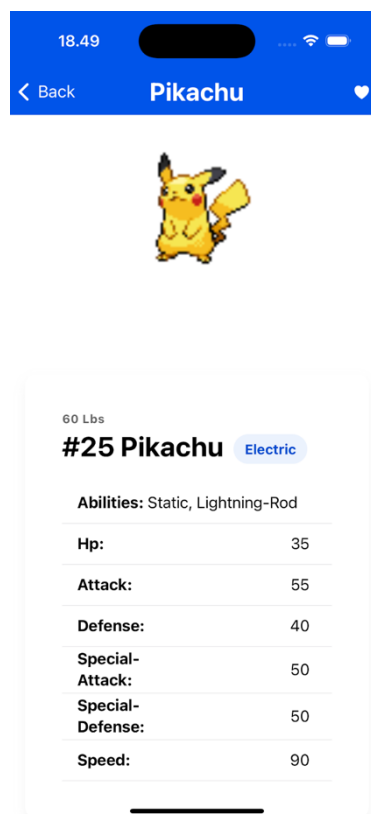
Setiap Pokemon juga memiliki halaman detail yang menampilkan informasi lengkap. Data yang ditampilkan mencakup nomor identitas Pokemon, seperti contoh “#25 Pikachu”, tipe elemen seperti “Listrik”, berat badan Pokemon yang biasanya ditampilkan dalam satuan pon (lbs) atau kilogram (kg), serta kemampuan (*abilities*) yang dimiliki Pokemon, seperti “Static” dan “Lightning-Rod”. Selain itu, halaman ini juga menyajikan statistik dasar Pokemon, meliputi HP (*Health Points*), *Attack*, *Defense*, *Special Attack*, *Special Defense*, dan *speed*. Informasi ini memberikan gambaran mendalam mengenai kekuatan dan kelemahan setiap Pokemon, sehingga pengguna dapat memahami karakteristik masing-masing Pokemon dengan lebih baik.

Aplikasi ini juga menyediakan fitur untuk menambahkan Pokemon ke dalam daftar favorit. Fitur ini memungkinkan pengguna menyimpan pokemon yang disukai sehingga dapat diakses kembali dengan mudah. Selain itu, pengguna juga dapat menghapus Pokemon dari daftar favorit apabila tidak lagi dibutuhkan. Fitur ini dirancang untuk memberikan pengalaman yang lebih personal dan sesuai dengan preferensi setiap pengguna.

Pengembangan aplikasi dimulai dengan integrasi PokeAPI melalui koneksi RESTful API. Penulis memastikan bahwa data yang ditampilkan dalam aplikasi selalu diperbarui dan akurat sesuai dengan informasi terkini yang ada di basis data PokeAPI. Menggunakan teknologi *front-end* modern seperti React Native atau Vue.js yang berguna untuk menciptakan antarmuka yang responsif dan ramah pengguna. Di sisi back-end, framework seperti Node.js digunakan untuk menangani permintaan API serta mengelola data pengguna terkait daftar favorit Pokemon.

Aplikasi ini dirancang sebagai latihan penulis agar dapat menguasai penggunaan *framework* ionic dalam pengembangan aplikasi *mobile*. Melalui pengembangan aplikasi “Pokedex”, penulis mempelajari untuk melakukan integrasi API publik. Proses ini memberikan pemahaman mendalam tentang arsitektur aplikasi, para penulis tidak hanya memahami teknologi yang digunakan tetapi juga memperoleh keterampilan yang relevan untuk diterapkan dalam pengembangan proyek ke depannya.

Rancangan yang telah dibuat mencakup antar muka dan fitur pada aplikasi, berikut Gambar 3.2 tangkapan layar halaman detail aplikasi Pokedek pada



Gambar 3.2 Halaman Detail Aplikasi Pokedex

Contoh fungsi yang tertampil di atas antara lain sebagai berikut:

- a) `loadPokemonDetails(url: string)` : Fungsi ini berguna untuk memuat detail Pokemon dari API berdasarkan *base url* yang ada pada *file* `pokemon.service.ts` untuk mengambil data melalui *HTTP request* dan kemudian memperbarui data Pokemon yang sedang ditampilkan.

- b) `async loadFavorites()` : Fungsi ini berguna untuk memuat daftar Pokemon Favorite pengguna yang disimpan dalam *local storage* menggunakan `this.storage.get('favorites')`.
- c) `checkIfFavorite()` : Fungsi ini berguna untuk memeriksa apakah Pokemon yang sedang ditampilkan adalah salah satu dari Pokemon favorit yang disimpan di dalam *array* favorite. Jika ditemukan, maka variabel `isFavorite` akan menjadi *true*, menandakan bahwa Pokemon tersebut adalah favorit. Namun jika tidak ditemukan, maka `isFavorite` akan menjadi *false*.
- d) `toggleFavorite()` : Fungsi ini digunakan untuk menambah atau menghapus Pokemon yang sedang ditampilkan dari daftar favorit yang tersimpan pada *local storage*. Pokemon yang sedang ditampilkan akan diperiksa apakah sudah ada dalam daftar favorite, jika akan ditambahkan pada daftar favorit maka ID, nama dan URL-nya akan ditambahkan kedalam *array* dan selanjutnya akan disimpan ke dalam *local storage* menggunakan `this.storage.set('favorites', this.favorites)`. Setelah itu, kita menjalankan fungsi `checkIfFavorite()` kembali untuk memperbarui data pokemon yang sedang ditampilkan.

3.1.4 Pengembangan Aplikasi S2P-Mobile

Pencatatan Health, safety, & Environment (HSE) di PT Sumber Segara Primadaya pada awalnya dilakukan secara manual menggunakan kertas dan pencatatan tertulis. Setiap laporan kejadian, seperti insiden kerja, potensi bahaya, atau kondisi tidak aman, dicatat oleh petugas atau karyawan yang bertanggung jawab, kemudian diserahkan ke bagian HSE untuk direkap. Proses ini memerlukan waktu yang cukup lama karena data harus dikumpulkan secara fisik, dipindahkan dari satu pihak ke pihak lain, dan kemudian diinput ulang ke dalam dokumen rekapitulasi.

Metode manual ini memiliki sejumlah kelemahan, salah satunya adalah tingginya risiko human eror. Kesalahan dapat pada tahap pengisian data (misalnya informasi tidak lengkap atau tulisan sulit baca), pada tahap pelaporan (terlambat menyampaikan informasi). Akibatnya, respon terhadap potensi bahaya atau insiden sering kali terlambat, sehingga dapat berdampak pada keselamatan pekerja dan kelancaran operasional.

Selain itu, sistem manual ini juga menyulitkan proses pencarian data historis karena dokumen arsip harus dicari secara fisik dan memakan waktu. Proses pelaporan berkala menjadi kurang efisien karena membutuhkan banyak tenaga untuk mengompilasi data dari berbagai

sumber. Keterbatasan ini membuat pihak manajemen sulit memantau kondisi HSE secara *real-time* dan mengambil keputusan cepat berdasarkan data aktual di lapangan.

Untuk mengatasi permasalahan tersebut dikembangkan aplikasi S2P-Mobile dikembangkan untuk PT Sumber Segara Primadaya guna meningkatkan aspek Health, Safety, & Environment (HSE) secara digital. Penggunaan Ionic Framework memungkinkan pengembangan aplikasi *cross-platform* dengan satu basis kode yang dapat berjalan pada perangkat Android dan iOS serta menyediakan komponen antarmuka bawaan yang mendukung penggunaan Material design dan iOS Human Interface Guidelines untuk antarmuka yang responsif.

Dalam pengembangan aplikasi S2P-Mobile, penulis tergabung dalam tim Mobile Developer yang terdiri atas empat orang, dengan satu orang supervisor bernama Hamzah Alvana, Barkah Ade sebagai senior mobile developer, serta dua orang peserta magang yang termasuk penulis.

Penulis ditempatkan pada proyek ini karena saat penulis masuk sebagai peserta magang, proyek pengembangan aplikasi S2P-Mobile merupakan proyek utama yang sedang berjalan di divisi pengembangan aplikasi mobile. Dengan demikian, penulis dapat terlibat langsung dalam proses pengembangan aplikasi yang relevan dengan bidang keahlian, sekaligus memperoleh pengalaman dalam membangun aplikasi berskala perusahaan.

a. Metode Pengembangan

Proses pengembangan aplikasi ini pihak kantor mengadopsi metode *Agile Development* karena kemampuannya untuk menyediakan proses yang adaptif dan dinamis. Metode ini menekankan pada kolaborasi tim, siklus pengembangan berulang yang cepat, serta pengumpulan umpan balik secara terus-menerus dari pengguna. Keunggulan utama dari model ini adalah fleksibilitas yang tinggi, yang memungkinkan tim untuk melakukan penyesuaian dan perbaikan secara cepat berdasarkan temuan selama pengujian atau masukan dari pengguna pada tahap awal agar memastikan aplikasi yang dihasilkan benar-benar menjawab kebutuhan di lapangan.

Penerapan metode ini dilakukan dengan memfokuskan pengembangan pada fitur-fitur utama dalam setiap *sprint*. Fitur-fitur yang telah dikembangkan tersebut kemudian akan dievaluasi dan disempurnakan pada *sprint* berikutnya. Proses berulang seperti ini memungkinkan aplikasi terus berkembang secara bertahap sehingga hasil akhirnya dapat berkembang sejalan dengan kebutuhan pengguna yang juga terus berubah seiring waktu.

Dengan mengadopsi pendekatan Agile Development, dalam pengembangan aplikasi S2P-Mobile berperan sebagai pengembang *front-end* yang bertanggung jawab mengimplementasikan fitur antarmuka pada aplikasi S2P-Mobile. Pengembangan *front-end* ini menggunakan Ionic Framework di mana setiap fitur dibangun secara modular dan terstruktur dengan mengacu pada spesifikasi teknis dan desain UI/UX dari tim desain agar sesuai dengan rancangan awal aplikasi.

Dukungan Capacitor dalam Ionic memfasilitasi integrasi fitur *native device* seperti RFID berbasis NFC, pemindai kode QR berbasis kamera dan SQLite untuk penyimpanan data lokal. Dengan dukungan data *binding* Angular, aplikasi ini dapat menyinkronkan data secara *real-time* dengan *database* untuk memastikan pembaruan informasi yang akurat. Selain itu, fitur *offline-first* memungkinkan aplikasi tetap berfungsi meskipun tanpa koneksi internet dengan mekanisme penyimpanan sementara menggunakan SQLite yang akan disinkronkan kembali dengan server saat koneksi internet tersedia.

b. Tahapan Pengembangan

Pengembangan aplikasi *mobile* S2P-Mobile dilakukan melalui beberapa tahapan berikut:

No	Deskripsi Kegiatan	Waktu Kegiatan
1	Implementasi desain antarmuka halaman Profil	25 Oktober 2024 – 19 November 2024
2	Pengujian Fungsionalitas halaman Profil pada simulator dan <i>real-device</i>	19 November – 21 November 2024
3	Implementasi desain antarmuka halaman Ganti Password	22 November – 2 Desember 2024
4	Pengujian fungsionalitas halaman Ganti Password pada simulator dan <i>real-device</i>	3 Desember 2024 – 4 Desember 2024
5	Implementasi desain antarmuka halaman Cek RFID	5 Desember 2024 – 17 Desember 2024
6	Pengujian fungsionalitas halaman Cek RFID pada simulator dan <i>real-device</i>	18 Desember 2024 – 19 Desember 2024
7	Implementasi fitur halaman Tentang Kami	20 Desember – 3 Januari 2025
8	Pengujian fungsionalitas halaman Tentang Kami pada simulator dan <i>real-device</i>	3 Januari 2025 – 4 Januari 2025

Tabel 3.2 Daftar Penugasan pada Proyek S2P-Mobile

Tabel 3.2 memuat daftar penugasan beserta waktu pengerjaan yang diberikan oleh supervisor. Rincian penugasan tersebut disajikan pada uraian berikut.

1. Implementasi Fitur

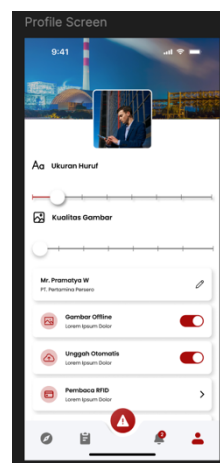
Pada tahap ini, penulis mengembangkan halaman tertentu dalam aplikasi yang dilakukan dengan mengacu pada desain UI/UX yang telah dibuat oleh tim desain agar memastikan konsistensi tampilan dan pengalaman pengguna yang optimal.

Pengembangan dilakukan menggunakan Ionic Framework dengan Angular sebagai *front-end* serta Capacitor untuk integrasi fitur perangkat. Sesuai dengan metode Agile Development, pengembangan fitur-fitur aplikasi S2P-Mobile dibagi

menjadi ke dalam beberapa siklus pendek yang disebut *sprint*. Setiap *sprint* memiliki target yang spesifik untuk menghasilkan bagian dari aplikasi yang fungsional. Pada proyek ini penulis bertanggung jawab pada dua *sprint* untuk pengembangan antarmuka pengguna.

Pada *sprint* pertama, yaitu berfokus pada pengembangan fitur-fitur utama yang berkaitan dengan manajemen akun pengguna untuk menyediakan fungsionalitas dasar yang memungkinkan pengguna mengelola profil dan keamanan akun mereka. Berikut beberapa fitur yang dikembangkan dalam *sprint* pertama:

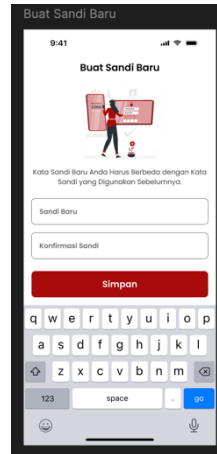
- Halaman Profil: Mengimplementasikan antarmuka dari desain figma pada Gambar x.x untuk menampilkan data pengguna seperti nama lengkap, foto profil dan informasi perusahaan. Pada tahap ini dilakukan integrasi awal dengan API untuk mengambil data pengguna dan menampilkannya menggunakan mekanisme *data binding* Angular.



Gambar 3.3 Desain Figma Halaman Profil

Selama tahap pengembangan terdapat perubahan desain antarmuka atas arahan supervisor divisi pengembangan aplikasi mobile. Desain awal pada halaman Profil Gambar 3.3 diubah menjadi desain baru seperti terlihat pada Gambar 3.15.

- Halaman Ganti Password: Mengimplementasikan antarmuka halaman yang berisi *form* untuk mengubah kata sandi beserta dengan validasi input untuk memastikan kata sandi baru memenuhi kriteria keamanan yang ditetapkan.



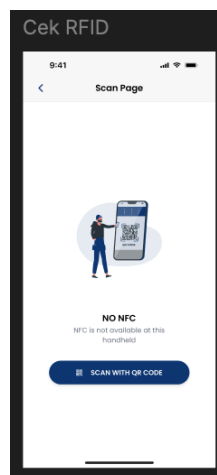
Gambar 3.4 Desain Figma Halaman Ganti Password

Perubahan desain juga terjadi pada halaman Ganti Password dari tampilan awal Gambar 3.4 menjadi tampilan baru pada Gambar 3.19.

Tampilan halaman Profil dan halaman Ganti password yang telah direvisi merupakan penyesuaian yang dilakukan pada tahap pengembangan sesuai arahan supervisor yang diberikan pada sesi review tugas mingguan.

Pada *sprint* kedua pengembangan fitur-fitur pendukung yang dapat memanfaatkan kemampuan perangkat keras dan menyajikan informasi tambahan lainnya berdasarkan desain UI/UX yang telah di desain oleh tim desain pada figma. Berikut beberapa fitur yang dikembangkan dalam *sprint* kedua:

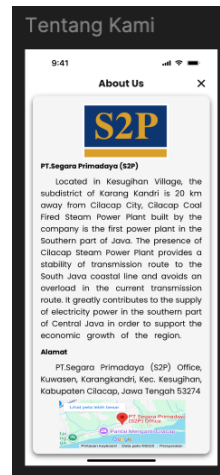
- Halaman Pembaca RFID: Mengimplementasikan antarmuka untuk fitur pemindaian kartu identitas pekerja. Bagian ini melibatkan integrasi dengan *library* Capacitor untuk mengakses fungsionalitas NFC pada perangkat *mobile*.



Gambar 3. 5 Tampilan Figma Halaman Cek RFID

Pada pengembangan fitur ini, Gambar 3. 5 tidak terdapat perubahan desain seperti pada sprint sebelumnya.

- Halaman Tentang Kami: Mengimplementasikan antarmuka halaman Tentang Kami yang berisi deskripsi singkat perusahaan, alamat serta peta lokasi interaktif yang diintegrasikan menggunakan layanan eksternal.



Gambar 3. 6 Tampilan Figma Halaman Tentang Kami

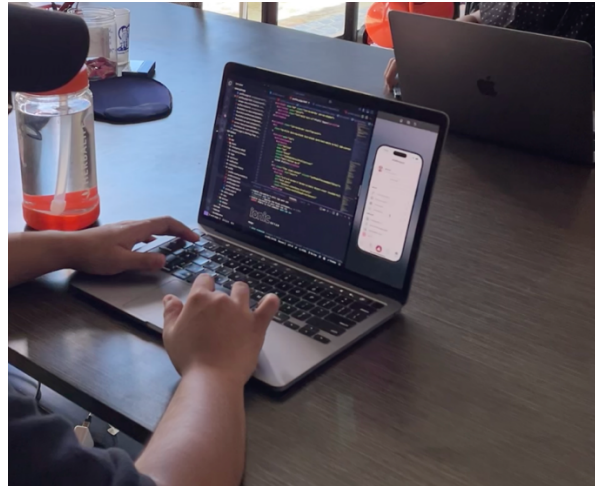
Pada pengembangan halaman Tentang Kami terdapat sedikit perubahan dari desain awal Gambar 3. 6 menjadi desain baru Gambar 3.25.

Perubahan desain antar muka merupakan tanggung jawab sepenuhnya atas perintah supervisor, karena terdapat fitur yang tidak dibutuhkan ataupun mengurangi tingkat efisiensi aplikasi. Oleh karena itu, penyesuaian dilakukan Ketika penulis melakukan review bersama supervisor.

2. Pengujian Fungsional

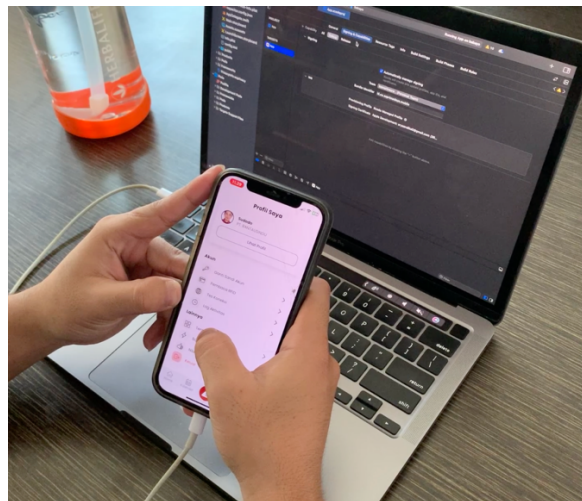
Setelah fitur dikembangkan, dilakukan pengujian fungsional secara manual untuk memastikan bahwa halaman berjalan sesuai dengan desain yang telah ditetapkan.

Pengujian ini dilakukan dengan memeriksa apakah terdapat ikon atau teks yang saling bertumpuk, serta memastikan bahwa fitur telah berjalan dengan baik dalam mengambil data pengguna dan dapat diakses tanpa kendala.



Gambar 3.7 Pengujian Fungsional Pada Simulator Menggunakan Xcode

Proses pengujian dilakukan dengan dua metode. Pertama Gambar 3.7, menggunakan simulator yang tersedia pada Xcode untuk memeriksa fungsionalitas dasar dan tampilan aplikasi pada berbagai ukuran layer perangkat iOS.



Gambar 3.8 Pengujian Fungsional Pada *Real-Device* Menggunakan Xcode

Berikutnya Gambar 3.8, melakukan pengujian langsung pada *real-device* yang terhubung melalui Xcode untuk memastikan kinerja aplikasi optimal pada kondisi penggunaan sebenarnya, termasuk interaksi dengan perangkat keras seperti NFC dan kamera.

3.2 Hasil

Aplikasi S2P-Mobile dikembangkan untuk PT Sumber Segara Primadaya guna meningkatkan aspek Health, Safety, & Environment (HSE) secara digital. Penggunaan Ionic Framework memungkinkan pengembangan aplikasi *cross-platform* dengan satu basis kode yang dapat berjalan pada perangkat Android dan iOS serta menyediakan komponen antarmuka bawaan yang mendukung penggunaan Material design dan iOS Human Interface Guidelines untuk antarmuka yang responsif.

Ionic Framework menyediakan berbagai komponen antarmuka bawaan seperti `<ion-header>`, `<ion-toolbar>`, `<ion-content>` dan `<ion-model>` yang mendukung pengembangan antarmuka interaktif. Pada Ionic juga terdapat dukungan CSS utilities bawaan yang memungkinkan penyesuaian tampilan tanpa memerlukan banyak kode tambahan. Penggunaan grid system dan flexbox bawaan juga mempermudah dalam mengatur tampilan antarmuka yang responsif di berbagai ukuran layar. Penerapan *dark mode* juga dimungkinkan untuk mendukung kenyamanan pengguna dalam menggunakan aplikasi.

Ionic Framework merupakan *framework open source* berbasis HTML, CSS, dan JavaScript yang digunakan untuk mengembangkan aplikasi *mobile* berbasis *hybrid*. Framework ini memungkinkan pengembangan satu basis kode yang dapat berjalan di berbagai *platform*, termasuk Android dan iOS. Dalam pengembangan aplikasi S2P-Mobile, Ionic digunakan sebagai teknologi utama untuk membangun antarmuka pengguna yang responsif dan kompatibel dengan berbagai perangkat.

Berdasarkan kode yang telah diimplementasikan, pengembangan aplikasi ini menggunakan Ionic Components yang mendukung tampilan berbasis Material Design dan iOS Human Interface Guidelines, memastikan pengalaman pengguna yang optimal. Beberapa komponen utama dari Ionic yang digunakan meliputi:

- a. `<ion-header>` dan `<ion-toolbar>`

Komponen ini digunakan untuk membangun *header* pada aplikasi, yang biasanya digunakan untuk menampilkan judul halaman serta tombol navigasi.

```
<ion-header mode="md" class="ion-no-border ion-no-shadow">
<ion-toolbar color="white">
<ion-title class="font-bold text-xl">Profil Saya</ion-title>
</ion-toolbar>
</ion-header>
```

Gambar 3.9 Kode Program untuk Menampilkan Judul Halaman Profil

Pada Gambar 3.9, menunjukkan kode `<ion-header>` yang digunakan untuk menentukan bagian atas halaman dengan tampilan `mode=md` (Material Design). `<ion-toolbar>` digunakan sebagai container yang menampung judul halaman yang ditampilkan menggunakan `<ion-title>`.

b. `<ion-content>` untuk Tampilan Halaman

```
<ion-content class="ion-no-border overflow-auto">
```

Gambar 3.10 Kode Program untuk konten halaman

Pada Gambar 3.10, `<ion-content>` digunakan untuk menampilkan isi utama halaman. Kelas `ion-noborder` digunakan untuk menghilangkan batas (*border*) standar dari Ionic, sedangkan `overflow-auto` memastikan bahwa konten dapat di *scroll* secara otomatis jika melebihi ukuran layar.

c. `<ion-card>` untuk Menampilkan Informasi Profil

```
<ion-card mode="ios">
  <ion-card-header>
    <ion-button
      fill="outline"
      mode="ios"
      expand="block"
      (click)="setModalDetailProfile(true)"
      class="small-button my-3"
    >Lihat Profil</ion-button>
  </ion-card-header>
</ion-card>
```

Gambar 3.11 Kode Program untuk Menampilkan informasi Profil

Pada Gambar 3.11, `<ion-card>` digunakan untuk menampilkan informasi pengguna dalam bentuk kartu yang responsif seperti. Dalam hal ini, komponen `<ion-card-header>` digunakan untuk menampilkan tombol “Lihat Profil” yang dapat membuka modal ketika ditekan.

d. `<ion-avatar>` untuk Menampilkan Foto Profil

```
<ion-avatar slot="start">
<ion-img
class="p-0 m-0 border-2 border-primary object-cover rounded-full mb-4"
alt="male photo"
[src]="dataUser.otherData.fotoProfil"
(ionError)="media.handleImageError($event)"
></ion-img>
</ion-avatar>
```

Gambar 3.12 Kode Program untuk Komponen Foto Profil

Pada Gambar 3.12, `<ion-avatar>` digunakan untuk menampilkan gambar profil pengguna dalam bentuk lingkaran. Atribut `[src]` digunakan untuk mengambil gambar dari *database* pengguna, sedangkan `(ionError)` menangani eror jika gambar gagal dimuat.

e. `<ion-list>` dan `<ion-item>` untuk Navigasi

```
<ion-list class="p-0 w-full bg-white dark:bg-black text-black
dark:text-white">
<ion-item color="transparent" lines="none" class="text-black
dark:text-white">
<ion-label class="font-bold">Akun</ion-label>
</ion-item>
<ion-item (click)="setModalChangePassword(true)" color="transparent"
lines="none">
<ion-icon name="key-outline"></ion-icon>
<p class="pl-4">Ganti Sandi Akun</p>
<ion-icon name="chevron-forward-outline" slot="end" />
</ion-item>
</ion-list>
```

Gambar 3.13 Kode Program untuk List Daftar Menu

Pada Gambar 3.13, `<ion-list>` digunakan untuk menyusun daftar menu navigasi yang berisi beberapa `<ion-item>`, yang masing-masing merepresentasikan menu yang dapat ditekan. Contohnya adalah menu “Ganti Sandi Akun” yang membuka modal pengubah kata sandi ketika diklik.

f. Penggunaan Modal dengan `<ion-modal>`

```
<ion-modal [isOpen]="isModalChangePassword">
```

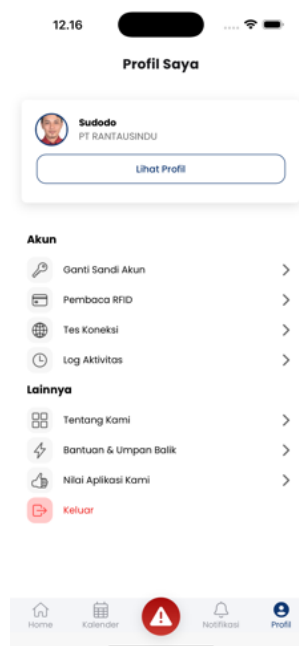
Gambar 3.14 Kode Program untuk Komponen Modal

Pada Gambar 3.14, `<ion-modal>` adalah komponen yang digunakan untuk menampilkan *popup* layar penuh atau sebagian. Dalam aplikasi ini, modal digunakan untuk menampilkan *form* pengubahan kata sandi dan informasi detail profil seperti. Penggunaan modal dipilih karena lebih efisien dibandingkan dengan membuat halaman baru, sehingga tidak menambah jumlah file dalam aplikasi dan mengurangi beban pemrosesan karena modal hanya akan ditampilkan saat diperlukan sehingga membantu mengoptimalkan konsumsi memori serta meningkatkan performa aplikasi agar tetap optimal.

Aplikasi yang dikembangkan dalam penelitian ini dijalankan pada perangkat iOS dengan versi 18.0.1. dan simulator dengan versi 18.0.1.

3.2.1 Halaman Profil

Pada Gambar 3.15 merupakan tampilan halaman profile pada aplikasi S2P-Mobile. Pada halaman profil ini terdiri dari beberapa menu. Pengguna dapat memilih menu sesuai dengan keperluan pengguna untuk menggunakan halaman tersebut.



Gambar 3.15 Tampilan Halaman Profil

Dalam pengembangan aplikasi ini, data pengguna seperti nama, nama perusahaan dan email diambil dari server menggunakan API, kemudian diintegrasikan dengan Angular melalui mekanisme data binding. Hal ini memungkinkan pembaruan data secara otomatis ketika aplikasi melakukan permintaan ke server.

```
<div class="text-sm font-bold">{{dataUser.namaLengkap}}</div>  
<p>{{dataUser.namaPerusahaan}}</p>
```

Gambar 3.16 Kode Program untuk Menampilkan Data Pengguna

Pada Gambar 3.16, nilai `dataUser.namaLengkap` dan `dataUser.namaPerusahaan` berasal dari objek `dataUser`, yang dikelola di dalam komponen TypeScript. Data ini diperoleh dari layanan `SharedService`, sebagaimana terlihat dalam kode pada Gambar 3.17.

```
this.dataUser = this.shared.user;  
console.log('dataUser', this.dataUser);
```

Gambar 3.17 Kode Program untuk Layanan `SharedService`

Dengan pendekatan ini, aplikasi dapat menyimpan dan mengelola data pengguna secara terpusat, sehingga perubahan data dapat segera terdistribusikan di berbagai bagian aplikasi yang membutuhkannya.

Selain itu, aplikasi juga menggunakan SQLite melalui `SqliteService` sebagai opsi penyimpanan lokal seperti pada Gambar 3.18.

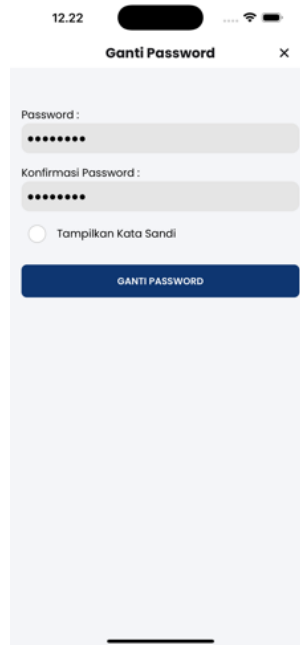
```
await this.sqlite.destroyTables(DatabaseName.K3Reporting);
```

Gambar 3.18 Kode Program untuk Sinkronisasi SQLite

Penyimpanan lokal ini berguna untuk menyimpan sementara data pengguna, sehingga aplikasi tetap dapat diakses meskipun dalam kondisi tanpa koneksi internet. Dengan mekanisme ini, data pengguna dapat dikelola secara efisien, baik dalam kondisi online maupun offline, serta memungkinkan sinkronisasi otomatis dengan server saat koneksi tersedia kembali.

3.2.2 Halaman Halaman Ganti Password

Pada Gambar 3.19 merupakan tampilan halaman Ganti Password. Pada halaman ini pengguna dapat mengganti kata sandi sebelumnya untuk menjaga keamanan akun pengguna.



Gambar 3.19 Tampilan Halaman Ganti Password

```
setModalChangePassword(isOpen: boolean) {  
  this.isModalChangePassword = isOpen;  
}
```

Gambar 3.20 Kode Program untuk Fungsi Ganti Password

Pada Gambar 3.20 fitur Ganti Password diinisiasi saat pengguna menekan menu “Ganti Sandi Akun”. Melalui *event binding (click)* dari Angular, memanggil fungsi `setModalChangePassword(true)` yang ada di dalam komponen TypeScript. Fungsi tersebut kemudian mengubah nilai variable `isModalChangePassword` menjadi `true`.

```

<ion-item (click)="setModalChangePassword(true)" color="transparent"
  lines="none"
  class="text-black dark:text-white">
  <ion-icon name="key-outline" class="icon-bg-gray icon-light"></ion-icon>
  <p class="pl-4 icon-small">Ganti Sandi Akun</p>
  <ion-icon name="chevron-forward-outline" slot="end" class="icon-light" />
</ion-item>

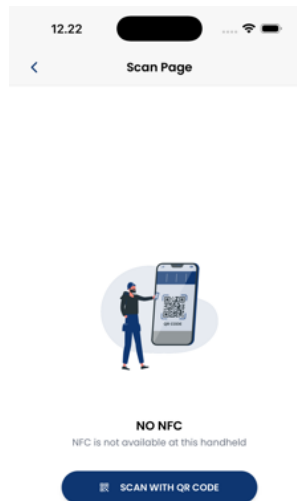
```

Gambar 3.21 Kode Program untuk Menu Ganti Sandi Akun

Pada Gambar 3.21 perubahan status ini secara otomatis dideteksi oleh *property binding* `[isOpen]` pada komponen `<ion-modal>` yang langsung menampilkan *popup* berisi formulir untuk mengganti kata sandi. Proses untuk menutupnya pun bekerja sebaliknya, di mana tombol tutup akan memanggil fungsi `setModalChangePassword(false)` untuk Kembali menyembunyikan *modal* tersebut. Alur kerja ini menunjukkan Angular mengelola interaksi pengguna dan status komponen secara efisien tanpa perlu berpindah halaman.

3.2.3 Halaman Pembaca RFID

Pada Gambar 3.22 adalah tampilan pembaca RFID, pada halaman ini pengguna dapat mengecek kartu identitas pegawai dengan menggunakan fitur NFC pada perangkat pengguna.



Gambar 3.22 Tampilan Halaman Cek RFID

Ionic Framework mendukung Capacitor, yang merupakan *library* untuk mengakses fitur *native* perangkat seperti kamera, GPS dan penyimpanan lokal. Dalam aplikasi ini, fitur Pembaca RFID digunakan untuk membaca kartu RFID, yang ditangani menggunakan Ionic dengan Capacitor.

```
async setupNfc() {
  const status = await this.nfc.enabled();
  if (status) {
    this.nfc.addNdefListener(async (event: any) => {
      const tagData = this.nfc.bytesToString(event.tag?.payload);
      const alert = await this.alertCtrl.create({
        header: 'Result',
        message: tagData,
        mode: 'ios',
        buttons: [{ text: 'Cancel', role: 'cancel' }, { text: 'Copy', handler: ()
=> Clipboard.write({ string: tagData }) }]
      });
      alert.present();
    });
  }
}
```

Gambar 3.23 Kode Program untuk Fungsi Pembaca NFC

Pada Gambar 3.23 fungsi `setupNFC()` yang berfungsi untuk menginisiasi fitur pembaca NFC pada perangkat *mobile* agar aplikasi siap untuk memindai dan membaca data dari kartu RFID atau tag NFC lainnya. Aplikasi memeriksa status NFC menggunakan `this.nfc.enabled()`, yang bertujuan untuk mendeteksi apakah fitur NFC dalam keadaan aktif atau tidak. Jika fitur ini aktif, aplikasi akan mulai membaca tag NFC menggunakan metode `addNdefListener()`, yang merupakan bagian dari Capacitor NFC Plugin. Saat tag NFC berhasil dipindai, data yang diperoleh kemudian ditampilkan dalam *alert* dan dapat disalin ke *clipboard* menggunakan Clipboard Capacitor, sebagaimana terlihat pada perintah `Clipboard.write({ string: tagData })`.

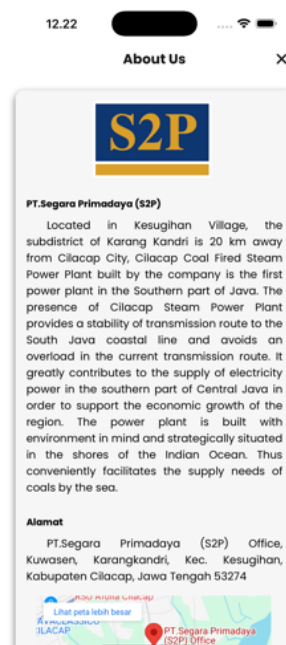
```
<ion-item (click)="setModalCheckRFID(true)" color="transparent"
lines="none">
<ion-icon name="card-outline" class="icon-bg-gray icon-light"></ion-icon>
<p class="pl-4 icon-small">Pembaca RFID</p>
<ion-icon name="chevron-forward-outline" slot="end" class="icon-light" />
</ion-item>
```

Gambar 3.24 Kode Program Tombol Navigasi Menu Pembaca RFID

Pada Gambar 3.24, terdapat tombol navigasi berupa `<ion-item>`, yang berfungsi untuk membuka halaman pembaca RFID saat diklik. Elemen ini terhubung dengan fungsi `SetModalCheckRFID(true)`, yang akan membuka modal berisi tampilan hasil pemindaian NFC. Modal ini dikendalikan oleh Angular melalui binding properti `[isOpen]`, yang bergantung pada nilai dari variabel `isModalCheckRFID`. Jika `isModalCheckRFID` bernilai `true`, modal akan ditampilkan, sedangkan jika bernilai `false`, modal akan tertutup.

3.2.4 Halaman Tentang Kami

Gambar 3.25 merupakan tampilan halaman tentang kami, pada halaman ini pengguna dapat membaca informasi singkat mengenai perusahaan beserta alamatnya.



Gambar 3.25 Tampilan Halaman Tentang Kami

Terdapat juga halaman Tentang Kami yang juga dibuat menggunakan Ionic dan Angular untuk memungkinkan integrasi data secara dinamis dan interaktif. Informasi yang ditampilkan dalam halaman ini meliputi deskripsi perusahaan, alamat, serta peta lokasi yang diproses menggunakan data *binding* Angular sehingga setiap pembaruan pada objek data akan langsung tercermin dalam tampilan aplikasi.

```
<iframe class="h-auto pr-8"
src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d15815.982436309
452!2d109.0873806!3d-
7.6836182!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x2e656d8acd10196d%3A0xe102b3
65e80ca179!2sPT.Segara%20Primadaya%20(S2P)%20Office!5e0!3m2!1sid!2sid!4v1
723003295595!5m2!1sid!2sid"
loading="lazy" style="width: 100%; margin-left: 15px;"
referrerpolicy="no-referrer-when-downgrade"></iframe>
```

Gambar 3.26 Kode Program untuk Menampilkan Peta

Pada Gambar 3.26 kode program halaman Tentang Kami dilengkapi dengan peta lokasi interaktif yang menggunakan layanan Google Maps yang meningkatkan pengalaman pengguna. Penggunaan iframe memungkinkan integrasi peta langsung dalam aplikasi, yang memungkinkan pengguna melihat lokasi perusahaan secara langsung tanpa harus membuka aplikasi lain.

```
<ion-item (click)="setModalAboutUs(true)" color="transparent"
lines="none">
<ion-icon name="grid-outline" class="icon-bg-gray icon-light"></ion-icon>
<p class="pl-4 icon-small">Tentang Kami</p>
<ion-icon name="chevron-forward-outline" slot="end" class="icon-
light"></ion-icon>
</ion-item>
```

Gambar 3.27 Kode Program untuk Menu Tentang Kami

Pada Gambar 3.27 penggunaan modal yang digunakan pada halaman Tentang Kami menggunakan Angular dengan properti `[isOpen]` yang memungkinkan pengguna menampilkan dan menutup modal sesuai kebutuhan. Modal akan terbuka jika pengguna memilih opsi Tentang Kami pada halaman Profil dengan memanggil metode `setModalAoutUs(true)`.

3.3 Pembahasan

Pengembangan aplikasi S2P-Mobile telah berhasil membuktikan bahwa penerapan teknologi *mobile* dengan *framework cross-platform* merupakan solusi yang efektif untuk menjawab tantangan dalam system Health, Safety, and Environment (HSE) yang diuraikan pada Bab I. Proyek ini tidak hanya berhasil menciptakan aplikasi yang fungsional, tetapi juga memvalidasi penerapan teori dan teknologi yang telah dibahas pada Bab II dalam sebuah studi kasus nyata di lingkungan industri.

Pemilihan Ionic Framework sebagai teknologi utama selaras dengan Tinjauan Pustaka, khususnya penelitian oleh (Asyahari et al., 2025) dan (Gerges & Elgalb, 2024) yang menegaskan efisiensi pendekatan *hybrid* dalam pengembangan aplikasi lintas platform. Dengan menggunakan satu basis kode, S2P-Mobile berhasil dikembangkan untuk platform Android dan iOS yang secara langsung menjawab kebutuhan efisiensi waktu dan sumber daya. Arsitektur aplikasi ini juga sesuai dengan klasifikasi oleh (Pamungkas & Imrona, 2020) yang menempatkan Ionic dalam kategori *framework* dengan pendekatan *webview* dimana aplikasi *website* dibungkus dalam wadah *native*.

Keberhasilan implementasi fitur-fitur interaktif pada aplikasi ini tidak terlepas dari peran sentral Angular sebagai kerangka kerja yang menggerakkan logika pada antarmuka pengguna. Berdasarkan implementasi pada sub-bab 3.2, fungsi Angular sangat krusial dalam beberapa aspek. Pertama dalam manajemen data dinamis, di mana data pengguna seperti nama foto profil yang diambil dari API secara otomatis ditampilkan ke antarmuka menggunakan mekanisme *data binding*, seperti pada kode `[src]="sataUser.otherData.fotoProfil"` dan `{{dataUser.namaLengkap}}`. Selanjutnya dalam pengelolaan interaksi pengguna, seluruh aksi seperti menekan tombol ditangani oleh *event binding Angular*, contohnya pada `(click)="setModalDetailProfile(true)"`. Terakhir, Angular juga mengontrol status dan fungsionalitas komponen, seperti pada penggunaan `<ion-modal [isOpen]="isModalChangePassword">` yang status buka-tutupnya bergantung pada nilai *variable* di dalam komponen TypeScript.

Selain itu, tantangan untuk mengakses perangkat keras *native*, yang sering menjadi kelemahan *framework hybrid*, berhasil diatasi dengan penggunaan Capacitor. Seperti yang dijelaskan pada sub bab 3.2.2, implementasi fitur pembaca RFID berbasis NFC pada fungsi `setupNFC()` menunjukkan bagaimana Capacitor menjadi jembatan yang efektif antara kode berbasis web dengan fitur asli perangkat. Ditambah lagi, penggunaan SQLite untuk penyimpanan data local berhasil menjawab kebutuhan aplikasi untuk dapat berfungsi secara

offline, sebuah syarat penting untuk aplikasi yang digunakan di lingkungan industry dengan konektivitas yang tidak menentu, seperti yang telah diidentifikasi pada latar belakang.

Dari sisi manajemen proyek, penerapan Agile Development memungkinkan pengembangan fitur-fitur seperti Halaman Profil, Ganti Password, Cek RFID dan Tentang Kami dilakukan secara bertahap dan terstruktur dalam *sprint*. Pendekatan ini terbukti efektif, seperti yang terlihat pada hasil Pengujian Alpha pada sub-bab 3.3, dimana aplikasi menunjukkan tingkat keberhasilan fungsional yang tinggi, yaitu 83,3% (11 dari 12 pengujian lulus). Satu-satunya pengujian yang gagal terkait masalah sinkronisasi data menjadi temuan paling penting yang dapat dijadikan dasar untuk perbaikan pada iterasi atau pengembangan selanjutnya.

Secara keseluruhan, pembahasan ini menunjukkan bahwa pengembangan aplikasi S2P-Mobile telah berhasil mengintegrasikan berbagai teknologi seperti Ionic Framework, Angular, Capacitor dan SQLite serta metode Agile Development untuk menciptakan solusi digital yang relevan dan fungsional bagi sistem manajemen HSE.

BAB IV

REFLEKSI PELAKSANAAN MAGANG

4.1 Relevansi Akademik

Pengembangan aplikasi S2P-Mobile yang merupakan aplikasi *mobile* penunjang sistem Health, Safety and Environment (HSE) di PT. Sumber Segara Primadaya. Aplikasi ini dirancang untuk memudahkan pemantauan dan pelaporan terkait keselamatan dan Kesehatan kerja karyawan saat bekerja. Pemilihan platform *mobile* dinilai strategis karena menawarkan mobilitas tinggi, kemudahan akses, serta pelaporan secara *real-time*.

Pengembangan aplikasi S2P-Mobile menggunakan pendekatan *cross-platform* dengan memanfaatkan Ionic yang memungkinkan satu basis kode yang dapat dijalankan pada platform iOS dan Android sehingga dapat lebih efisien dalam pengembangan dan pemeliharaan aplikasi di berbagai perangkat. Penggunaan Ionic sebagai *crossplatform* framework digunakan oleh penulis selama pelaksanaan magang dan menjadi pengalaman baru bagi penulis.

Pengalaman pengembangan menggunakan Ionic Framework ini memberikan wawasan baru yang berbeda secara signifikan dengan pembelajaran yang penulis dapatkan di lingkungan akademik. Perbedaan metode dan teknologi antara praktik industri saat magang dengan materi perkuliahan dapat dilihat dalam Tabel 4.1 perbandingan berikut:

Tabel 4.1 Perbandingan Pembelajaran

No	Aspek Pemanding	Pembelajaran saat Magang	Pembelajaran saat kuliah
1	Pendekatan Pengembangan	<i>Cross-Platform</i>	Native
2	Framework & Teknologi	Ionic Framework	Kotlin dengan Android Studio
3	Target Platform	Android dan iOS	Android
4	Fokus Utama	Efisiensi pengembangan dan pemeliharaan lintas platform	Penguasaan fundamental pada platform Android
5	Konteks Proyek	Kebutuhan industri nyata	Tugas dan materi sesuai kurikulum mata kuliah

Tabel 4.1 perbandingan di atas secara jelas menunjukkan adanya *gap* antara pembelajaran akademik dengan praktik di dunia industri. Di lingkungan kampus, fokus pembelajaran cenderung pada penguasaan fundamental teknologi *native* pada satu platform Android. Jika pada industri lebih menekankan pada efisiensi pengembangan lintas platform untuk menjangkau lebih banyak pengguna. Oleh karena itu, pengalaman magang ini menjadi sangat berharga karena memberikan pemahaman praktis mengenai tren teknologi yang relevan di

dunia kerja, sekaligus melatih kemampuan penulis dalam beradaptasi dan belajar mandiri di luar materi yang diajarkan secara formal.

Bagi penulis saat penyesuaian pengembangan aplikasi *mobile native* ke *cross-platform* melatih kemampuan adaptasi teknis dan membuktikan bahwa fundamen pemrograman yang didapat dari perkuliahan dapat menjadi bekal untuk menyesuaikan teknologi baru secara mandiri sesuai dengan kebutuhan industri.

Selain itu, selama pelaksanaan magang perusahaan menerapkan metode Agile pada proyek pengembangan aplikasi. Penerapan metode ini bertujuan untuk menciptakan alur kerja yang lebih terstruktur, terarah, serta memungkinkan adanya evaluasi dan perbaikan berkelanjutan di setiap tahapan pengembangan. Dalam penjalanannya, proyek dibagi menjadi beberapa sprint dengan target yang jelas pada setiap periode, sehingga tim pengembangan dapat lebih focus menyelesaikan bagian aplikasi secara bertahap namun terukur. Setiap sprint diakhiri dengan sesi evaluasi dan *review* bersama supervisor, yang menjadi wadah diskusi bagi anggota tim untuk memberikan masukan, mengidentifikasi kendala, serta menentukan solusi yang tepat sebelum melanjutkan ke sprint berikutnya.

Pengalaman ini sangat berbeda dengan yang diperoleh saat perkuliahan. Meskipun materi mengenai manajemen proyek telah diajarkan dalam mata kuliah Rekayasa Perangkat Lunak, pembagian tugas pada umumnya masih dilakukan secara acak tanpa adanya manajemen yang terstruktur.

Selama proses magang, penulis bertanggung jawab untuk mengimplementasikan antarmuka pengguna (*user interface*) pada beberapa halaman utama aplikasi, yakni halaman Profil, Ganti Password, Cek RFID dan Tentang Kami. Pembuatan antarmuka ini dilakukan berdasarkan desain *UI/UX* yang telah disusun oleh tim desain. Dalam pengerjaannya, penulis berdiskusi secara aktif dengan tim pengembang terkait kebutuhan teknis serta penyesuaian fungsionalitas agar implementasi sesuai dengan harapan perusahaan.

4.2 Pembelajaran Magang

Selama proses magang di PT. Nocola IoT Solution, banyak hal yang memiliki peran penting dalam pengembangan keterampilan dan pengalaman profesional bagi penulis. Penjelasan mengenai manfaat serta kendala, hambatan dan tantangan.

4.2.1 Manfaat

Kegiatan magang di PT Nocola IoT Solution memberikan pengalaman yang berharga bagi penulis dalam memahami secara langsung proses pengembangan aplikasi *mobile* berbasis Ionic Framework, khususnya dalam proyek pengembangan aplikasi S2P-Mobile. Melalui keterlibatan dalam proyek tersebut, penulis dapat mengimplementasikan desain antarmuka pengguna berdasarkan rancangan UI/UX yang telah ditentukan, serta berkontribusi dalam proses integrasi fitur aplikasi.

Selama proses magang, penulis dihadapkan pada lingkungan kerja profesional yang menuntut kepatuhan terhadap standar kualitas dan tenggat waktu yang ketat. Bekerja di bawah tenggat waktu yang terstruktur melatih kedisiplinan dan efektivitas penulis dalam mengelola tugas secara mandiri, Menyusun prioritas pekerjaan berdasarkan urgensinya, serta mengembangkan kemampuan pemecahan masalah untuk mengatasi tantangan teknis yang muncul selama proses implementasi. Penulis juga berkesempatan untuk secara rutin menyampaikan hasil pekerjaan kepada supervisor dalam sesi *review*, berdiskusi mengenai masukan yang diberikan dan melakukan penyesuaian yang dibutuhkan. Proses ini tidak hanya memastikan kesesuaian hasil dengan target awal tetapi juga mengasah kemampuan komunikasi teknis dan adaptasi penulis.

Secara keseluruhan, pelaksanaan magang ini memberikan kontribusi nyata yang signifikan terhadap kesiapan penulis dalam menghadapi dunia kerja. Pengalaman ini berfungsi sebagai wadah untuk beradaptasi antara pengetahuan akademik yang bersifat teoritis dengan tantangan praktis di dunia industri. Penulis tidak hanya mengaplikasikan konsep yang dipelajari di bangku kuliah, tetapi juga memperkaya diri dengan keterampilan baru yang relevan dengan kebutuhan industri saat ini. Dengan demikian, pengalaman ini tidak hanya memperkuat pondasi teknis, tetapi juga membangun mentalitas profesional yang siap untuk berkontribusi secara efektif dalam lingkungan kerja yang dinamis.

4.2.2 Kendala, Hambatan dan Tantangan

Selama menjalani kegiatan magang sebagai Mobile Developer di PT Nocola IoT Solution selama kurang lebih enam bulan, penulis menghadapi sejumlah tantangan dalam proses pengembangan aplikasi S2P-Mobile. Salah satu tantangan utama yang dihadapi adalah keterbatasan pengalaman dalam menggunakan Ionic Framework, yang merupakan teknologi utama dalam pengembangan aplikasi di perusahaan. Sebelumnya, penulis hanya memiliki latar belakang dalam pengembangan aplikasi menggunakan Android Studio dengan pendekatan

native, sehingga memerlukan waktu beradaptasi dengan sintaksis, struktur proyek dan ekosistem kerja Ionic.

Keterbatasan tersebut sempat menjadi hambatan dalam menyelesaikan tugas-tugas yang diberikan oleh supervisor, terutama pada tahap awal implementasi antarmuka pengguna. Namun kendala ini dapat diatasi dengan cara mempelajari dokumentasi resmi Ionic, bertanya kepada supervisor serta melakukan eksplorasi langsung melalui latihan mandiri dan eksperimen terhadap komponen UI yang digunakan dalam aplikasi S2P-Mobile.

Salah satu pengalaman yang cukup berarti selama proses magang adalah saat penulis diberi tanggung jawab untuk mengembangkan tampilan antarmuka pada halaman *dashboard* aplikasi. Penulis menyelesaikan pengembangan halaman tersebut sesuai dengan arahan dan desain yang diberikan. Namun, setelah selesai, pihak perusahaan memutuskan untuk tidak menggunakan hasil pekerjaan yang telah dibuat oleh penulis dan menggantinya dengan versi yang dikembangkan oleh karyawan lainnya. Meskipun keputusan tersebut menjadi pengalaman yang cukup mengecewakan, penulis menyadari bahwa hasil kerja yang dibuat mungkin masih perlu peningkatan, baik dari segi konsistensi struktur kode, efisiensi penggunaan komponen, maupun standar penulisan kode yang digunakan oleh tim pengembang utama. Keputusan ini menjadi bahan refleksi bagi penulis untuk terus meningkatkan kualitas teknis dalam pengembangan antarmuka dan memastikan bahwa hasil pekerjaan sesuai dengan standar perusahaan.

Tantangan lain yang dihadapi adalah penempatan lokasi magang yang tidak berada di kantor pusat, melainkan di kantor workshop. Hal ini menyebabkan komunikasi langsung dengan supervisor menjadi terbatas, terutama saat membutuhkan penjelasan teknis. Untuk mengatasi hal tersebut, penulis dan tim magang memanfaatkan media komunikasi daring seperti WhatsApp dan Google Meet untuk menjaga koordinasi serta menyampaikan progres pekerjaan secara rutin.

Selain itu, penulis juga menghadapi keterbatasan dalam mendapatkan bimbingan intensif dari supervisor dikarenakan jadwal kerja supervisor yang padat dan tanggung jawab terhadap beberapa proyek sekaligus. Kendala ini diatasi dengan menjadwalkan sesi review dan diskusi bersama supervisor, sehingga kendala teknis dapat didiskusikan secara bersama.

Secara keseluruhan, tantangan-tantangan tersebut memberikan pelajaran berharga bagi penulis dalam hal kemandirian, kemampuan adaptasi, inisiatif belajar, serta kemampuan untuk menyusun strategi pemecahan masalah secara mandiri maupun kolaboratif. Pengalaman ini juga memperkuat kesiapan penulisan untuk menghadapi situasi kerja yang dinamis dan tidak selalu ideal di dunia profesional.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil kegiatan penelitian yang dilakukan melalui program magang di PT. Nocola IoT Solution memberikan kontribusi yang signifikan terhadap pengembangan kompetensi penulis, baik dalam aspek teknis maupun non-teknis. Selama masa magang, penulis memperoleh kesempatan untuk menerapkan teknologi Ionic Framework dalam pengembangan aplikasi *mobile* lintas platform, khususnya pada proyek pengembangan aplikasi S2P-Mobile. Pengalaman ini menambah wawasan dan keterampilan penulis dalam mengembangkan antarmuka pengguna sesuai desain UI/UX yang telah ditetapkan.

Penulis juga menghadapi berbagai tantangan, terutama akibat perbedaan fundamental antara latar belakang akademis dalam pengembangan *native* berbasis Kotlin dengan ekosistem *cross-platform* Ionic Framework. Transisi ini memerlukan adaptasi dari pendekatan pemrograman imperatif dan *layouting* berbasis XML di Android Studio ke arsitektur berbasis komponen yang deklaratif, manajemen *state* menggunakan *services*, dan penanganan proses asinkron. Di dalam Angular. Keterbatasan pengalaman ini berdampak langsung pada implementasi awal, seperti pada pengembangan halaman *dashboard* yang akhirnya tidak digunakan karena pertimbangan kualitas dan efisiensi kode yang digunakan. Faktor lain adalah minimnya bimbingan langsung bersama *supervisor* akibat kendala lokasi dan kesibukan *supervisor*. Kendala tersebut dapat diatasi melalui pembelajaran mandiri, eksplorasi dokumentasi resmi, serta diskusi dengan rekan magang dan supervisor secara daring maupun luring.

Berdasar pengalaman magang ini menunjukkan betapa pentingnya kerja sama antara dunia perkuliahan dan industri. Terlihat jelas bahwa kurikulum pendidikan perlu menyertakan materi tentang teknologi *cross-platform*, karena sudah menjadi standar di banyak perusahaan. Di sisi lain, perusahaan akan sangat terbantu jika program magangnya terstruktur dan memiliki bimbingan teknis yang terjadwal, karena mahasiswa bisa beradaptasi lebih cepat dan memberikan kontribusi yang lebih maksimal. Pelajaran penting lainnya dari pengalaman ini adalah bahwa keberhasilan sebuah proyek tidak hanya dinilai dari hasil akhirnya, tetapi juga dari proses belajar, cara menyelesaikan masalah dan perkembangan profesional selama proses tersebut.

Secara keseluruhan, pelaksanaan magang ini memberikan pengalaman yang sangat berarti bagi penulis, tidak hanya dalam aspek teknis pengembangan perangkat lunak, tetapi juga dalam juga dalam penguatan karakter, kedisiplinan, dan kemampuan beradaptasi terhadap dinamika kerja di industri teknologi informasi. Pengalaman ini menjadi pondasi penting dalam mempersiapkan diri menghadapi dunia profesional di masa depan.

5.2 Saran

Setelah menyelesaikan proses program magang di PT Nocola IoT Solution terdapat beberapa saran yang dapat penulis sampaikan baik bagi pihak perusahaan maupun bagi mahasiswa yang akan mengambil program magang ke depannya.

Bagi perusahaan tempat magang, PT Nocola IoT Solution disarankan agar program magang ke depannya dapat memiliki struktur bimbingan yang lebih terjadwal, terutama jika mahasiswa ditempatkan di lokasi yang berbeda dari supervisor. Sesi *review* rutin akan sangat membantu mempercepat proses adaptasi dan penyelesaian masalah. Selain itu, untuk memaksimalkan kontribusi akan sangat bermanfaat jika mahasiswa dilibatkan secara lebih menyeluruh dalam seluruh siklus pengembangan, mulai dari *sprint planning* hingga *sprint review* untuk memberikan pengalaman pengembangan yang lebih menyeluruh.

Selanjutnya bagi mahasiswa yang akan melaksanakan magang di bidang pengembangan aplikasi, disarankan untuk mempersiapkan diri tidak hanya secara teknis, tetapi juga mental dan manajemen waktu agar dapat menjalankan tugas dengan baik meskipun bekerja secara mandiri atau dengan bimbingan dari pembimbing dari perusahaan secara terbatas agar lebih cepat beradaptasi dan berkontribusi secara maksimal dalam proyek.

DAFTAR PUSTAKA

- Askar, N. F., Herawati, D., Susilawati, S., & Pratomo, D. N. (2021). Sistem Pelaporan Berbasis Web Pada Pelaksanaan Kegiatan Keselamatan dan Kesehatan Kerja (K3) Di Puskesmas. *Jurnal JKFT*, 6(2), 17. <https://doi.org/10.31000/jkft.v6i2.5615>
- Asyahari, A., Yuda, F., Meiditra, I., Lubis, S. S., Dehasen Bengkulu, U., Teknologi, I., & Hilir, R. (2025). Implementasi Hybrid Programming Dengan Framework Ionic Dan Capacitor Pada Sistem I-Sekolahku. In *Riau Journal of Computer Science* (Vol. 11, Issue 1). <https://doi.org/https://doi.org/10.30606/rjocs.v11i1.3043>
- C. G., T., & Devi, A. J. (2021). A Study and Overview of the Mobile App Development Industry. *International Journal of Applied Engineering and Management Letters*, 115–130. <https://doi.org/10.47992/IJAEML.2581.7000.0097>
- Candra Pamungkas, Diva Yuannisa Nur Berlian, Nabila Renaldy, Fawwaz Naufaldini, & Muhammad Nur Yasin. (2023). Implementasi Pembuatan Sistem Informasi Artha Puspa dengan Framework Agile (Scrum). *JUMINTAL: Jurnal Manajemen Informatika Dan Bisnis Digital*, 2(2), 179–191. <https://doi.org/10.55123/jumintal.v2i2.2401>
- Cantika, D. A., & Sofyan, M. (2024). Penerapan Keselamatan dan Kesehatan Kerja: Studi kasus Pada PT Mega Cipta Bangsa. *Jurnal Ekonomi, Manajemen, Bisnis, Dan Sosial (EMBISS)*, 4(3), 237–246. <https://doi.org/10.59889/embiss.v4i3.305>
- Chaudhary, P. (2018). Ionic Framwork. *International Research Journal of Engineering and Technology*. www.irjet.net
- Efendi, A., & Ubaidi. (2020). Sistem Informasi Pkpri (Pusat Koperasi Pegawai Republik Indonesia) Kabupaten Sampang. *Jurnal Insand Comtech*, 5(1). <https://doi.org/http://dx.doi.org/10.53712/jic.v5i1.828>
- Fahrudin, R., & Ilyasa, R. (2021). Perancangan Aplikasi “Nugas” Menggunakan Metode Design Thinking dan Agile Development. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 8(1), 35–44. <https://doi.org/10.33197/jitter.vol8.iss1.2021.714>
- Gerges, M., & Elgalb, A. (2024). Comprehensive Comparative Analysis of Mobile Apps Development Approaches. *Journal of Artificial Intelligence General Science (JAIGS) ISSN:3006-4023*, 6(1), 430–437. <https://doi.org/10.60087/jaigs.v6i1.269>

- Ilhami, M. (2021). Tren dan Peluang Cross-Platform Mobile App untuk Developer Pemula. *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi*, 1(2), 402–411. <https://doi.org/10.24002/konstelasi.v1i2.4320>
- Irawan, P., Jend Yani, J. A., Naga, K., & Kisaran Timur, K. (2025). Perancangan Sistem Informasi Keselamatan dan Kesehatan Kerja (K3) di PT. Multi Jaya Teknik. *Jurnal Riset Sistem Informasi Dan Teknik Informatika*, 3(2). <https://doi.org/10.61132/merkurius.v3i2.719>
- kompasiana. (2023, October 2). *Definisi serta Kelebihan dan Kekurangan Metode Agile Crystal*.
- Lutfi Wirananda Silvana, Wahyudi Agustiono, & Fifin Ayu Mufarroha. (2024). Implementasi Payment Gateway Pada Marketplace Digital Product Berbasis Website Dengan Menggunakan Metode Agile. *Journal of Scientech Research and Development*, 6(2), 451–464. <https://doi.org/10.56670/jsrd.v6i2.523>
- Ningtyas, P. N., Kurniasih, D., & Arninputranto, W. (2024). Perancangan Sistem Informasi Pelaporan Statistik K3 Berbasis Website. *IJCIT (Indonesian Journal on Computer and Information Technology)*, 9(1). <https://doi.org/10.31294/ijcit.v9i1.16411>
- Nuristi, N. D., & Farell, G. (2022). Rancang Bangun Sistem Informasi HIRADC (Hazard Identification Risk Assessment & Determining Control). *Voteteknika (Vocational Teknik Elektronika Dan Informatika)*, 10(2), 32. <https://doi.org/10.24036/voteteknika.v10i2.117333>
- Pamungkas, L. A. B., & Imrona, D. M. (2020). Analisa Perbandingan Kinerja Cross Platform Mobile Framework React Native dan Flutter. *e-Proceeding of Engineering*, 7(1), 2195–2203. <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/11807>
- Purnawati, E., Syafa'at, A. Y., Febianto, D. D., Sari, D., & Adhimah, L. F. (2023). Design of Health, Safety, and Environment Digital Application with Android-Based User Interface. *SISTEMASI*, 12(1), 257. <https://doi.org/10.32520/stmsi.v12i1.2446>
- Putra, A. D., Syamsuir, E., & Wahyuni, F. I. (2021). Analisis penerapan kesehatan dan keselamatan kerja (K3) di perusahaan jasa konstruksi Kota Payakumbuh. *Rang Teknik Journal*, 4(1), 76–82. <https://doi.org/10.31869/rtj.v4i1.2034>
- Putra, Y. W. S., Dawis, A. M., Novi, N., Natsir, F., Fitria, F., Widhiyanti, A. A. S., & Maniah, M. (2023). *Pengantar Aplikasi Mobile*. Haura Utama.

- Revaldo, A., & Putra, S. (2025). Evaluasi Kepuasan Pengguna Aplikasi Inspekta di PT. PLN UP2D S2JB Menggunakan Metode Pieces. *Jurnal Riset Sistem Informasi Dan Teknik Informatika*, 3(4). <https://doi.org/10.61132/mercurius.v3i4.927>
- Saputra, M. A. Y., Fadillah, M. N., & Setiadi, D. (2021). Aplikasi Checklist Safety Berbasis Mobile Menggunakan Metode Job Safety Analysis (JSA) Untuk Audit Kesehatan dan Keselamatan Kerja (K3) Di PT.Matahari Putra Prima,Tbk. *Jurnal Sistem Informasi Universitas Suryadarma*, 8(2). <https://doi.org/10.35968/jsi.v8i2.740>
- Sihombing, A. P., Fahrullah, F., & Riyayatsyah, R. (2021). Aplikasi Health Safety Environment Pada PT Altrak 1978 Samarinda. *Jurnal Ilmu Komputer Dan Informatika*, 1(1), 29–42. <https://doi.org/10.54082/jiki.5>
- Singh, M., & G, S. (2021). Comparative Analysis of Hybrid Mobile App Development Frameworks. *International Journal of Soft Computing and Engineering*, 10(6), 21–26. <https://doi.org/10.35940/ijscce.F3518.0710621>
- Sinlae, F., Haryanto, B. D., Saputra, I. E., & Alfaridzy, M. D. (2024). Pengaruh Penggunaan Framework Angular Terhadap Efisiensi Pengembangan Website. *Scientica: Jurnal Ilmiah Sain dan Teknologi*, 2(9), 108–112. <https://jurnal.kolibi.org/index.php/scientica/article/view/2319>
- Sukoco, D., & N, A. (2023). Efektifitas Aplikasi Inspekta Dalam Peningkatan Budaya K3 Di PLN ULP Kisaran. *J-Com (Journal of Computer)*, 3, 221–228. <https://doi.org/10.33330/j-com.v3i3.2900>
- Swatika, B., Wibowo, P. A., & Abidin, Z. (2022). Pengaruh Keselamatan dan Kesehatan Kerja (K3) Terhadap Produktivitas Kerja Karyawan. *Jurnal Ilmu Kesehatan Masyarakat*, 11(02), 197–204. <https://doi.org/10.33221/jikm.v11i02.1220>
- Syarifudin, D. M., & Siswanto, Ir. (2018). Rancang Bangun Sistem Monitoring Berbasis Location Based Service Menggunakan Ionic Framework. *SKANIKA*, 1(1), 297–302. <https://jom.fti.budiluhur.ac.id/SKANIKA/article/view/196>
- Tahvanainen, M. (2023). *Hybrid mobile development using Ionic framework* [University of Helsinki]. <https://helda.helsinki.fi/handle/10138/352932>
- Waranashiwar, J., & Ukey, M. (2018). Ionic Framework with Angular for Hybrid App Development. *International Journal of New Technology and Research*, 4(5).
- Wijaya, T. (2023). Implementasi Framework Angular untuk Meningkatkan Performa Aplikasi Web Modern. *Seminar Nasional Corisindo*, 33–37. <https://ojs.stmikpontianak.ac.id/corisindo/article/view/4>

Zohud, T., & Zein, S. (2021). Cross-Platform Mobile App Development in Industry: A Multiple Case-Study. *International Journal of Computing*, 20(1), 46–54.
<https://doi.org/10.47839/ijc.20.1.2091>

LAMPIRAN



