

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama menjalani kerja magang di PT Nusantara Compnet Integrator, penempatan sebagai *Developer Intern* dilakukan dalam tim *IT Support Software* yang berada di bawah naungan Divisi *Engineering Operation*. Tim ini memiliki peran dalam memastikan kelancaran operasional sistem teknologi informasi internal, baik melalui pengembangan aplikasi baru maupun pemeliharaan sistem yang sudah ada. Koordinasi juga dilakukan secara langsung dengan beberapa pihak terkait, antara lain:

- **Pieter Yanuar Tanosady** (*IT Support Manager*): Bertindak sebagai supervisor proyek, menetapkan arah strategis pengembangan, serta mengevaluasi progres tim secara keseluruhan.
- **Ria Romasari Girsang** (*Application Support Supervisor*): Berperan sebagai mentor teknis, memberikan panduan dalam implementasi fitur, dan menjadi penghubung antara tim IT dengan divisi pengguna akhir (*end user*).
- **Anggota Tim IT Support Software Lainnya**: Termasuk *developer* lainnya yang berkolaborasi dalam siklus kerja *Agile* (Scrum).
- **Divisi Lain di Perusahaan**: Termasuk pihak pengguna atau *end user* dari berbagai divisi yang terlibat dalam proses pengumpulan kebutuhan, pengujian, serta evaluasi fitur selama pengembangan aplikasi berlangsung.

Koordinasi dalam tim dilakukan melalui pendekatan metodologi *Agile* (scrum) dengan siklus dua minggu (*sprint*). Berikut adalah alur koordinasi yang diterapkan:

1. Rapat Harian (*Daily Sprint Meeting*)

- **Durasi**: 15–30 menit setiap pagi.
- **Tujuan**: Menyelaraskan prioritas harian, melaporkan progres, dan mengidentifikasi kendala teknis.
- **Platform**: Microsoft Teams.

2. Rapat Mingguan (*Weekly Sprint Planning*)

- **Pelaksanaan:** Setiap dua minggu untuk merencanakan tugas dalam *sprint* berikutnya.
- **Diskusi:** Alokasi *task*, estimasi waktu pengerjaan, dan penyesuaian target berdasarkan *feedback end user*.

3. Internal Demo

- **Proses:** Setelah fase pengembangan selesai, hasil kerja dipresentasikan kepada Application Support Supervisor dan Tim untuk evaluasi awal.
- **Tindak Lanjut:** Jika ditemukan *bug* atau kebutuhan perbaikan, tim diberikan waktu untuk menindaklanjuti sebelum demo lanjutan.

4. Komunikasi Informal

- **Platform:** *WhatsApp* atau *Microsoft Teams*.
- **Tujuan:** Diskusi cepat terkait perubahan jadwal, masalah mendadak, atau koordinasi dengan tim lain.

5. Manajemen Tugas

- **Trello:** Digunakan untuk memantau *task* dalam kolom "*Sprint Backlog*", "*In Progress*", "*Pending*", "*Ready for Test*", dan "*Done*".
- **GitLab:** Platform utama untuk *version control*, kolaborasi kode, dan *merge request* sebelum integrasi ke *branch* utama.

3.2 Tugas yang dilakukan

Selama periode magang, aktivitas pengembangan perangkat lunak internal perusahaan dilakukan. Pengembangan aplikasi internal perusahaan serta pemeliharaan sistem yang sudah ada menjadi tanggung jawab magang. Tugas ini dilakukan dalam kerangka kerja *sprint* mingguan selama dua proyek utama. Setiap proyek mencakup dua hingga tiga *sprint*, dengan satu *sprint* berdurasi masing-masing selama sekitar dua minggu. Tugas yang dilakukan setiap minggu didokumentasikan dalam Tabel 3.1 berikut.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

| Minggu Ke- | Pekerjaan yang dilakukan |
|------------|--|
| 1 | Pengenalan alur kerja dan tools, menyiapkan environment serta repository, dan membuat ERD dan UI/UX aplikasi Permainan Kuis Tim. |
| 2 | Memulai pengembangan aplikasi Permainan Kuis Tim, serta membuat form <i>create/update</i> Game dan eksplorasi <i>real-time update</i> . |
| 3 | Uji coba awal <i>real-time update</i> dengan Pusher dan bantu tim <i>hardware</i> proses pergantian perangkat kantor <i>batch</i> pertama. |
| 4 | Fokus membantu pergantian perangkat kantor <i>batch</i> pertama. |
| 5 | Selesai bantu tim <i>hardware</i> , melaksanakan demo pertama aplikasi Permainan Kuis Tim, dan menindaklanjuti langsung ke demo kedua. |
| 6 | Fokus membantu pergantian perangkat kantor <i>batch</i> kedua. |
| 7 | Selesai bantu tim <i>hardware</i> dan mulai pengembangan proyek <i>PPE</i> . |
| 8 | Membuat fitur permintaan dan persetujuan APD (proyek <i>PPE</i>). |
| 9 | Perbaiki <i>bugs</i> , penyesuaian email, dan menyesuaikan fitur dengan kebutuhan user. |
| 10 | Demo internal pertama proyek <i>PPE</i> dan revisi sesuai masukan. Lanjut demo internal kedua. |
| 11 | Demo proyek <i>PPE</i> dengan user dan tindak lanjut berupa revisi berdasarkan masukan serta kebutuhan pengguna. |
| 12 – 13 | Menyusun dokumentasi proyek <i>PPE</i> , menunggu <i>user trial</i> , dan melakukan perbaikan <i>bugs</i> serta tampilan. |
| 14 | Melaksanakan demo akhir proyek <i>PPE</i> kepada user, menyusun panduan admin, dan menindaklanjuti revisi akhir sesuai kebutuhan. |
| 15 | Mengembangkan fitur penomoran pada portal <i>Legal</i> . |
| 16 | Menambahkan fitur <i>import</i> dan menyempurnakan penomoran pada portal <i>Legal</i> . |
| 17 | Memulai pengembangan awal proyek portal <i>GA</i> untuk manajemen ruang rapat. |

3.3 Uraian Pelaksanaan Magang

Selama periode magang, terlibat secara langsung dalam dua proyek utama. Proyek umumnya dimulai dengan tahap perencanaan pada *weekly sprint meeting* yang mencakup penjelasan mengenai ruang lingkup proyek serta identifikasi kebutuhan dari *user*. Setelah kebutuhan tersebut berhasil dirumuskan secara jelas, proses dilanjutkan ke fase pengembangan inti aplikasi sesuai dengan rencana dan prioritas yang telah ditetapkan.

Setelah fase pengembangan utama selesai, dilakukan sesi demo *internal* sebagai tahap evaluasi awal. Tujuan dari demo *internal* pertama adalah untuk menilai kelayakan, kualitas, serta kesesuaian hasil pengembangan dengan kebutuhan yang telah ditentukan. Apabila dalam sesi ini ditemukan *bugs* atau terdapat kebutuhan pengembangan tambahan akibat ketidaksesuaian dengan ekspektasi pengguna, maka perbaikan dan penyempurnaan akan dilakukan setelah sesi demo *internal* tersebut. Selanjutnya, sesi demo *internal* kedua dilaksanakan sebagai bentuk validasi akhir. Sesi ini bertujuan untuk memastikan bahwa seluruh permasalahan telah diselesaikan dan aplikasi telah memenuhi seluruh kriteria yang dibutuhkan sebelum dipresentasikan kepada pengguna akhir (*end user*).

Apabila hasil evaluasi internal telah memenuhi standar kualitas yang ditetapkan, maka proses dilanjutkan dengan penjadwalan sesi demo dengan pengguna. Jika dalam demo tersebut pengguna memberikan masukan, menemukan *bugs*, atau mengusulkan perbaikan fitur, maka tim akan diberikan waktu untuk menindaklanjuti dan melakukan penyesuaian. Setelah seluruh penyempurnaan selesai, sesi demo lanjutan kepada pengguna akan dilakukan untuk memastikan bahwa solusi yang dikembangkan telah sepenuhnya sesuai dengan kebutuhan dan harapan pengguna. Detail pekerjaan yang dilakukan setiap minggunya terdapat pada tabel 3.1.

Pengerjaan *version control* proyek dilakukan melalui platform *Git* dan *GitLab* Compnet. Meskipun tidak ada ketentuan khusus mengenai editor yang digunakan, seluruh anggota tim umumnya menggunakan *Visual Studio Code*. Untuk kebutuhan pengujian lokal dan basis data, digunakan *XAMPP* sebagai *local server* serta *HeidiSQL* sebagai alat bantu manajemen basis data. Seluruh aktivitas pengembangan dan pengujian dilakukan secara sistematis pada development server perusahaan. Setelah seluruh proses pengembangan selesai dan *user* memberikan persetujuan terhadap hasil yang telah dicapai, aplikasi kemudian di-*deploy* ke server produksi.

Proyek aplikasi Permainan Kuis Tim adalah proyek pertama yang dikembangkan selama masa magang. Permainan Kuis Tim merupakan permainan kuis interaktif berbasis survei, di mana dua tim saling menebak jawaban terbanyak dari pertanyaan yang telah disurvei sebelumnya. Proyek ini merupakan *request* dari pihak internal perusahaan untuk dikembangkan sebagai media permainan yang dapat digunakan dalam berbagai acara internal, seperti perayaan ulang tahun *PT Nusantara Compnet Integrator* maupun peringatan Hari Kemerdekaan pada tanggal 17 Agustus. Tujuan utama dari pengembangan proyek ini adalah untuk mendukung kelancaran acara serta menciptakan suasana yang lebih meriah dan interaktif. Permainan ini dirancang agar sesuai digunakan dalam sesi permainan kelompok, *ice breaking*, maupun hiburan ringan di tengah rangkaian kegiatan perusahaan.

Alasan utama aplikasi Permainan Kuis Tim perlu dikembangkan menjadi sebuah aplikasi adalah agar permainan ini dapat digunakan secara berkelanjutan dalam berbagai acara perusahaan di masa mendatang. Dengan adanya aplikasi, proses permainan dapat berjalan lebih efisien, terstruktur, dan bebas dari kendala teknis yang kerap muncul jika dilakukan secara manual. Selain itu, aplikasi ini juga memungkinkan adanya pencatatan data permainan, seperti pertanyaan, jawaban, dan skor tim, yang dapat berguna untuk evaluasi atau digunakan kembali di acara selanjutnya. Di sisi lain, penggunaan aplikasi dapat meningkatkan kualitas interaksi saat acara berlangsung karena seluruh proses—mulai dari tampilan pertanyaan, penghitungan poin, hingga transisi antar sesi—dapat dilakukan secara otomatis, responsif, dan lebih menarik secara visual. Hal ini tentunya akan menciptakan pengalaman yang lebih menyenangkan bagi peserta dan membantu panitia dalam menjalankan acara dengan lebih mulus dan profesional.

Proyek kedua yang dikerjakan selama masa magang adalah pengembangan aplikasi *Personal Protective Equipment* atau disingkat *PPE*. Aplikasi *PPE* merupakan sistem berbasis web yang dirancang untuk mempermudah proses manajemen peminjaman Alat Pelindung Diri (APD) di lingkungan perusahaan. Fungsi utama dari aplikasi ini adalah sebagai platform *resource management*, yang memungkinkan pengguna untuk mengajukan permintaan peminjaman APD secara digital melalui formulir bernama FPA (Form Peminjaman APD).

Dengan adanya aplikasi ini, seluruh proses peminjaman menjadi lebih terdokumentasi, terstruktur, dan transparan, mulai dari pengisian formulir, proses persetujuan, hingga pengambilan APD oleh peminjam. Sebelumnya, proses peminjaman dilakukan secara manual melalui *spreadsheet Excel*, yang sering kali menimbulkan berbagai kendala, seperti pencatatan data yang tidak konsisten,

duplikasi peminjaman, serta permintaan yang mendadak. Kondisi ini turut menyulitkan divisi *Health, Safety, and Environment* (HSE) dalam melakukan pengelolaan APD secara efektif. Pengembangan aplikasi ini diharapkan dapat meningkatkan efisiensi operasional, meminimalkan kesalahan administratif, serta mempercepat proses pelacakan dan pengembalian APD. Selain itu, aplikasi ini juga memberikan visibilitas yang lebih baik terhadap data penggunaan APD, sehingga memudahkan perusahaan dalam melakukan evaluasi dan perencanaan kebutuhan di masa mendatang.

3.3.1 Perangkat Penunjang

Dalam pelaksanaan praktik kerja magang, terdapat beberapa perangkat keras dan lunak yang digunakan. Berikut adalah rincian perangkat yang dipakai:

- **Perangkat Keras**

- Dell Latitude E6450 (Laptop Kantor sebelum transisi Laptop)
- Lenovo V14 G2 IAP (Laptop Kantor)
- Intel Core i3-1215U
- RAM 8GB
- SSD 256GB
- Mouse Lenovo (Asset Kantor)

- **Perangkat Lunak**

- **Windows 11** sebagai sistem operasi utama
- **PHP 8.x** sebagai bahasa pemrograman backend
- **Laravel 10.x** sebagai framework web development
- **MySQL** sebagai Database Relasional
- **HeidiSQL** sebagai Database Viewer
- **Git** untuk Version Control dengan **GitLab** sebagai repository hosting
- **Visual Studio Code** sebagai Text Editor utama
- **Microsoft Teams** sebagai platform kolaborasi tim
- **Trello** sebagai alat manajemen tugas untuk melacak progres pengembangan

3.3.2 Permainan Kuis Tim

Aplikasi Permainan Kuis Tim dirancang untuk digunakan oleh admin sebagai pengendali utama jalannya permainan. Para peserta atau *participants* tidak berinteraksi langsung dengan aplikasi, melainkan mengikuti kuis secara langsung di lokasi acara. Aplikasi ini berfungsi sebagai tampilan visual untuk mendukung permainan, di mana admin mengakses halaman khusus untuk mengelola kuis, sedangkan pemain, pembawa acara (*MC*), dan penonton menyaksikan jalannya permainan melalui halaman *ongoing game* yang bersifat publik.

A Requirements

Pada tahap perencanaan proyek, ditetapkan beberapa fitur utama yang perlu dikembangkan. Fitur-fitur ini merupakan elemen dasar yang menjadi inti dari aplikasi, serta dirancang untuk mendukung jalannya kuis secara efektif dan efisien. Meskipun demikian, secara keseluruhan fitur yang dibutuhkan tergolong sederhana, selama aplikasi dapat menjalankan mekanisme permainan kuis tim seperti menampilkan pertanyaan, mencatat jawaban, dan menghitung skor secara otomatis. Pendekatan ini diambil agar pengembangan dapat difokuskan pada fungsionalitas inti permainan dan dapat diselesaikan dalam waktu yang terbatas.

B Perancangan Sistem

Proses perancangan sistem didukung oleh dokumentasi teknis menggunakan standar *Unified Modeling Language* (UML), seperti *Use Case Diagram*, *Activity Diagram* dan *Entity Relationship Diagram* (ERD). Selain itu, *flowchart* disusun sebagai representasi visual dari alur kerja utama dalam aplikasi, guna memberikan pemahaman yang lebih jelas terhadap logika proses dan keterkaitan antar komponen sistem.

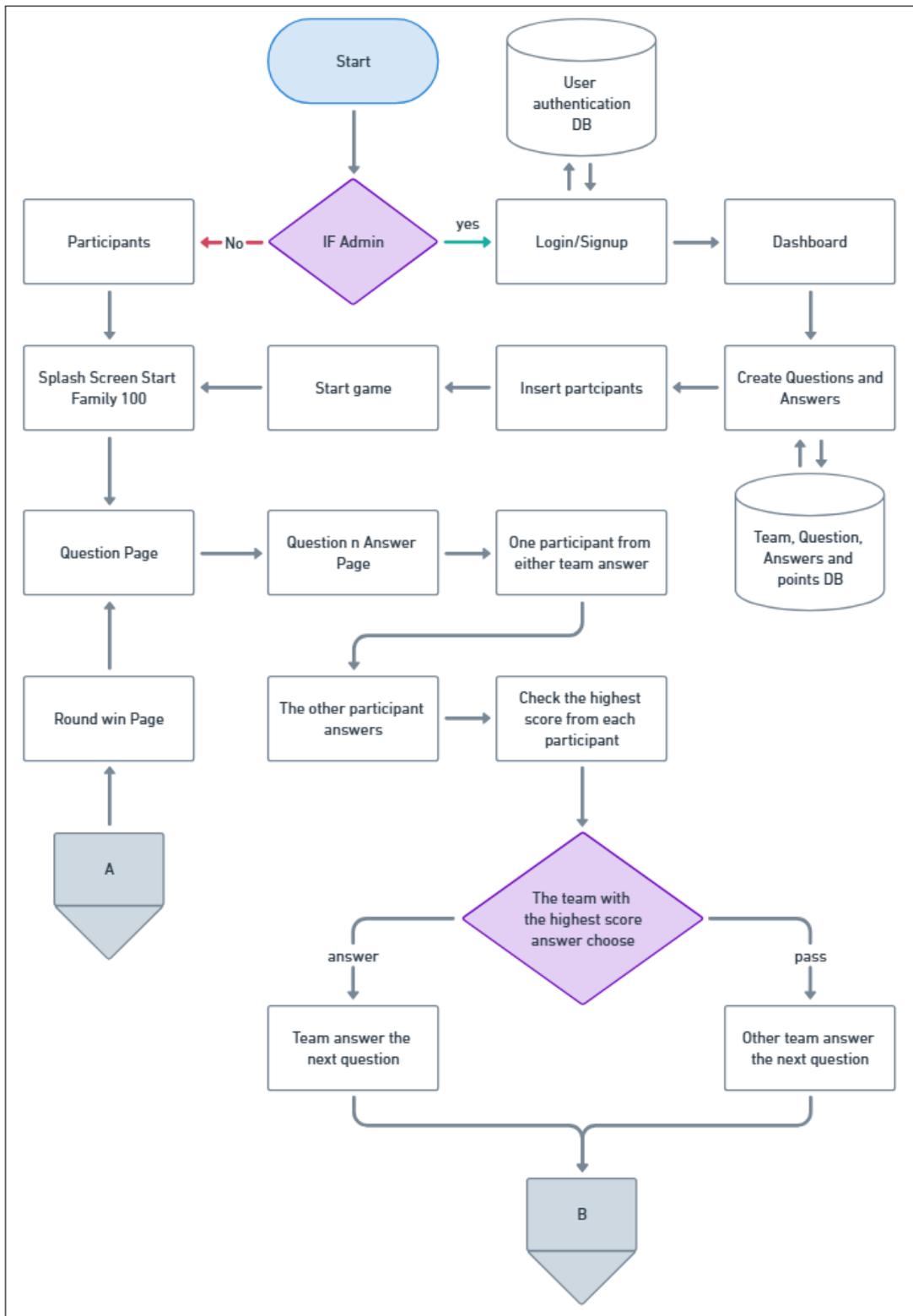
B.1 Flow Chart

Permainan Kuis Tim merupakan permainan tanya jawab di mana dua tim bersaing untuk menebak jawaban-jawaban terbanyak dari suatu pertanyaan. Permainan diawali dengan admin yang memilih dan memulai satu pertanyaan. Pertanyaan tersebut akan ditampilkan pada halaman peserta dan penonton. Tim yang mendapat giliran akan menjawab secara bergantian. Jika jawaban benar

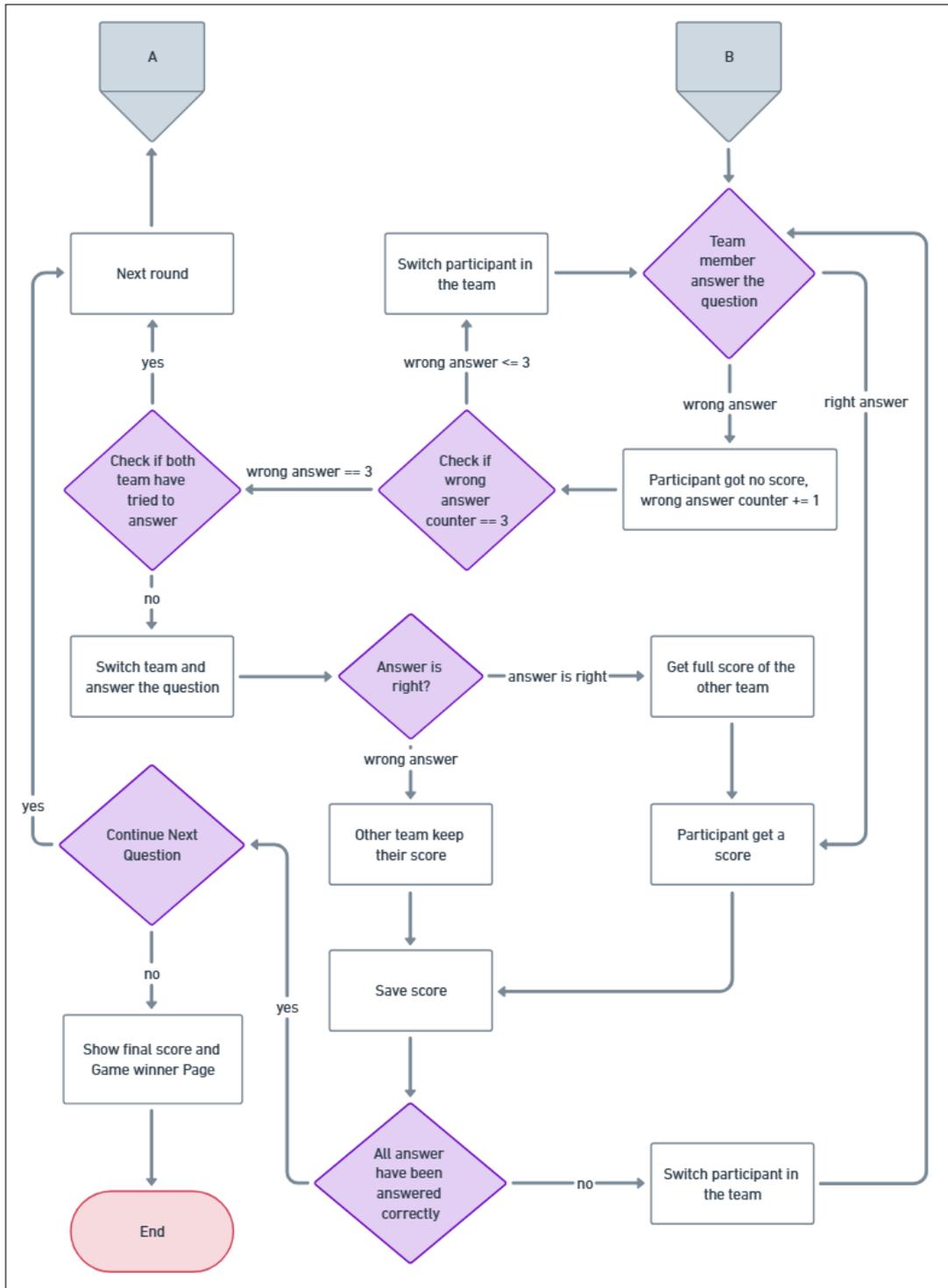
(berada dalam daftar jawaban), maka jawaban tersebut akan ditampilkan pada layar beserta nilai poinnya. Namun, jika jawabannya salah, maka akan ditampilkan simbol silang besar sebagai tanda kesalahan. Ketika semua jawaban benar telah ditemukan atau kesempatan tim habis, maka poin dari jawaban-jawaban yang benar akan diberikan kepada tim yang terakhir aktif. Setelah satu ronde selesai, permainan dilanjutkan ke ronde berikutnya dengan pertanyaan baru.

Alur permainan ini divisualisasikan melalui sebuah *flowchart* yang menggambarkan proses dari awal hingga akhir permainan, termasuk logika pergantian giliran, validasi jawaban, dan pemberian poin. Flowchart dirancang untuk menggambarkan keseluruhan alur permainan secara menyeluruh, sebagaimana ditunjukkan pada Gambar 3.1 dan Gambar 3.2.





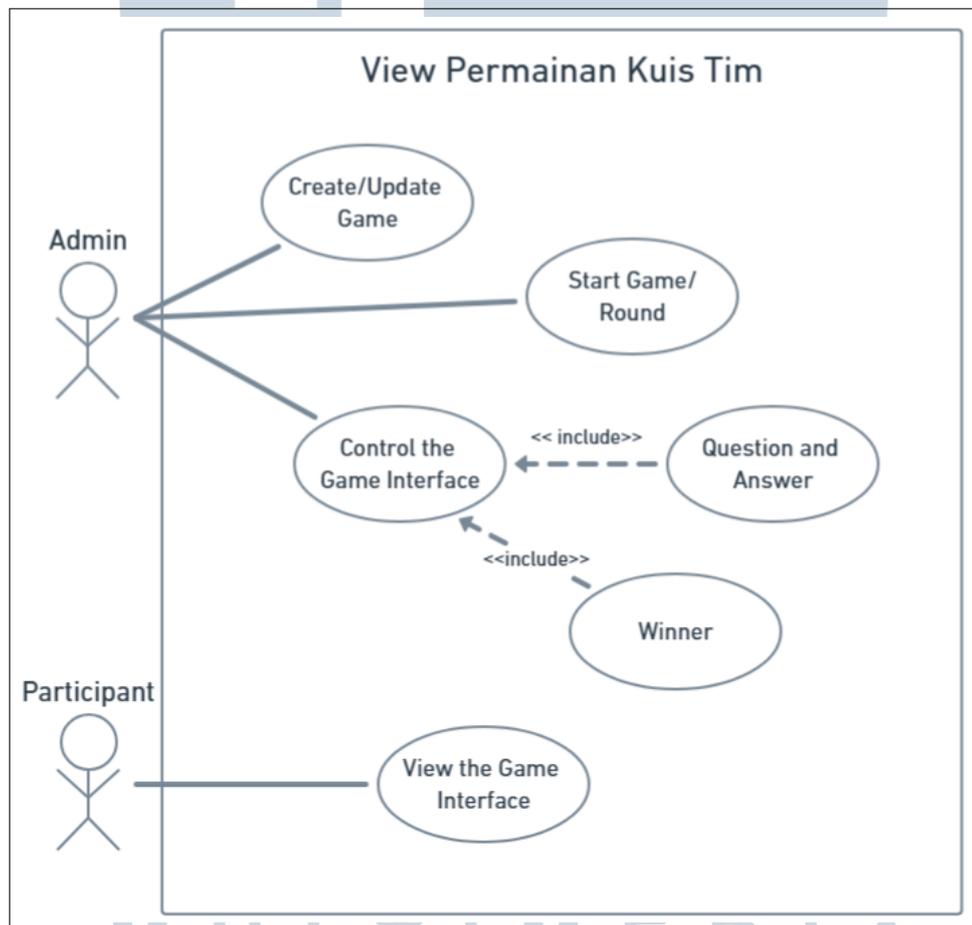
Gambar 3.1. Flowchart Aplikasi Permainan Kuis Tim – Bagian 1



Gambar 3.2. Flowchart Aplikasi Permainan Kuis Tim – Bagian 2

B.2 Use Case Diagram

Untuk memahami interaksi antara pengguna dan sistem, disusun *Use Case Diagram* sebagai representasi visual terhadap fungsi utama dan aktor yang terlibat dalam sistem. Diagram ini membantu dalam mengidentifikasi proses-proses inti yang dilakukan oleh masing-masing peran pengguna selama pengoperasian aplikasi. Pada aplikasi Permainan Kuis Tim, aktor utama yang terlibat adalah admin yang bertanggung jawab terhadap pengaturan dan kendali permainan secara penuh, serta peserta dan penonton sebagai penerima output dari sistem. *Use Case Diagram* dapat dilihat pada Gambar 3.3.

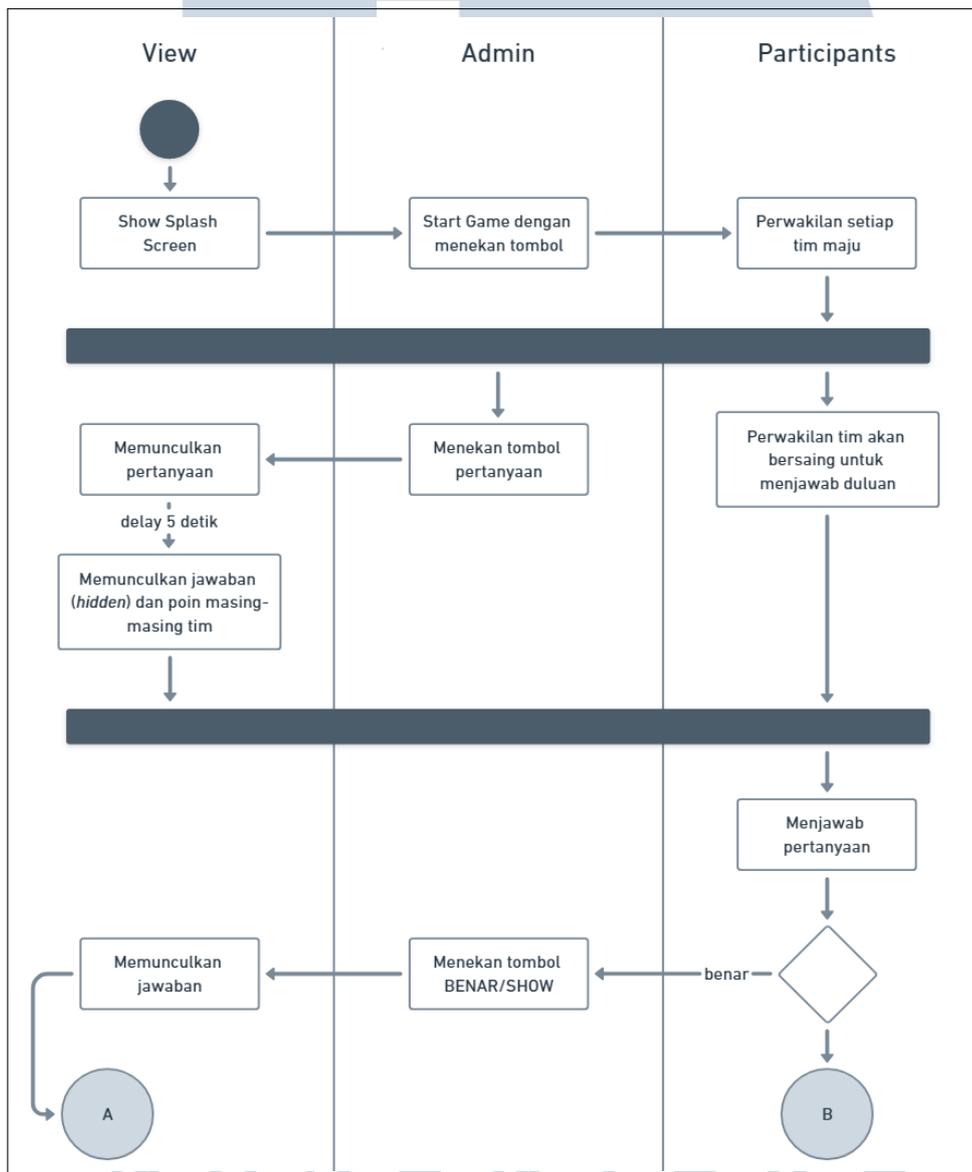


Gambar 3.3. *Use Case Diagram* Aplikasi Permainan Kuis Tim

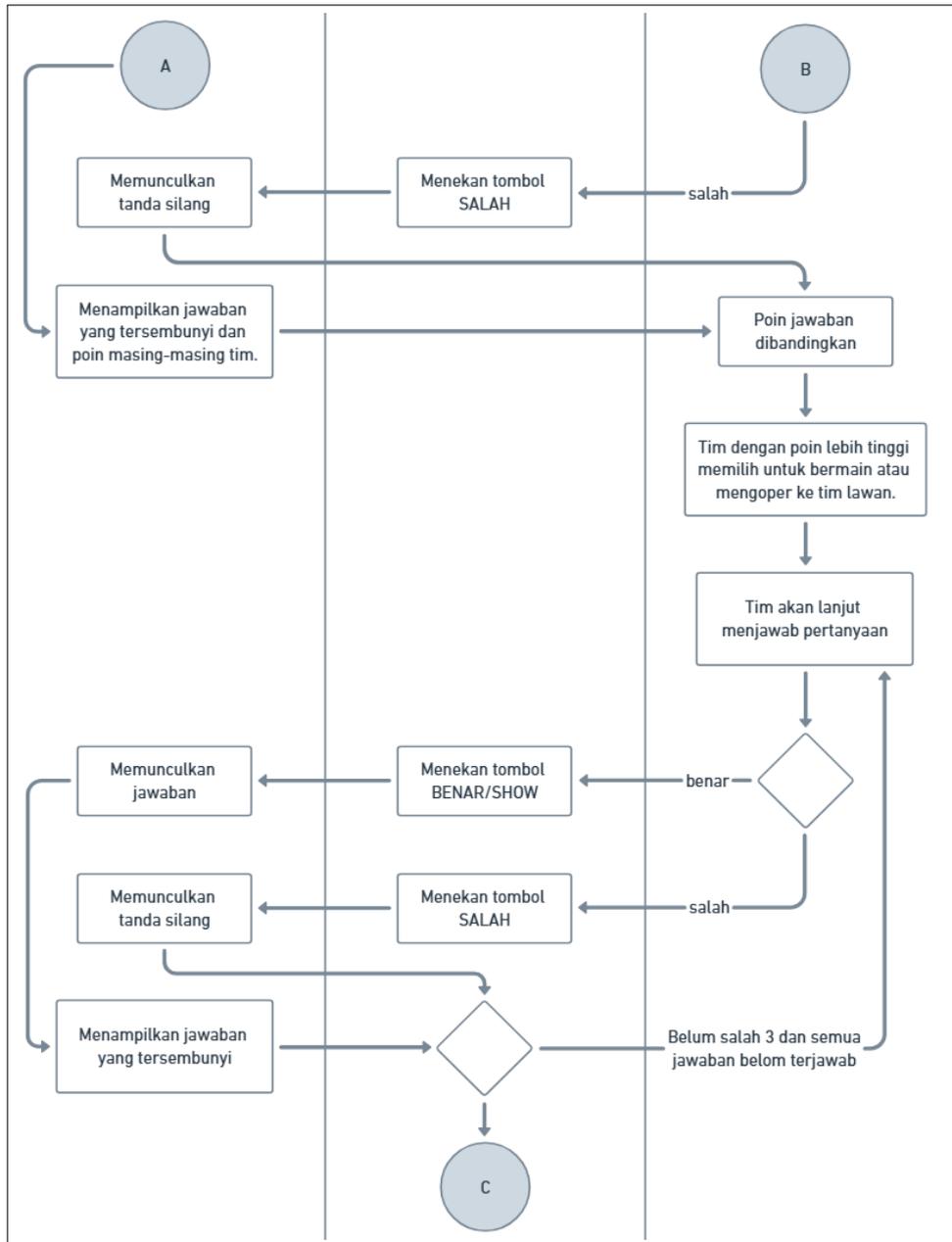
B.3 Activity Diagram

Setelah berhasil mengidentifikasi semua skenario utama beserta aktor yang terlibat dengan bantuan *Use Case Diagram*, langkah berikutnya adalah memperinci

alur proses dalam sistem menggunakan *Activity Diagram*. Aplikasi ini secara operasional dikendalikan oleh admin selama permainan berlangsung, sementara *participants* merupakan dua tim yang mengikuti kuis secara langsung. *Activity Diagram* berikut menggambarkan alur aktivitas selama satu ronde permainan, termasuk interaksi admin dengan sistem dan proses tampilan pertanyaan hingga pemberian skor. Karena diagram cukup panjang, visualisasinya dibagi menjadi tiga bagian tampilan, seperti ditunjukkan pada Gambar 3.4, 3.5, dan 3.6.

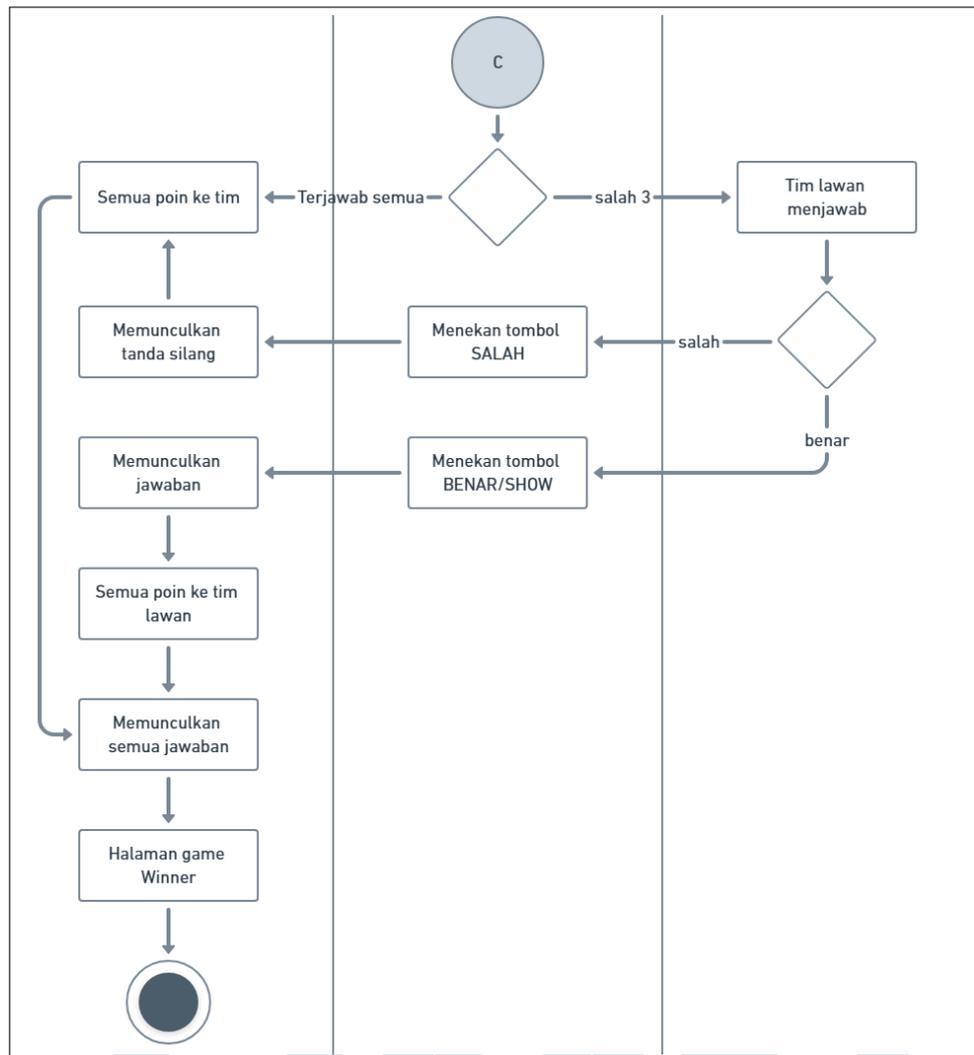


Gambar 3.4. *Activity Diagram* Aplikasi Permainan Kuis Tim – Bagian 1



Gambar 3.5. Activity Diagram Aplikasi Permainan Kuis Tim – Bagian 2

UNIVERSITAS
MULTIMEDIA
NUSANTARA

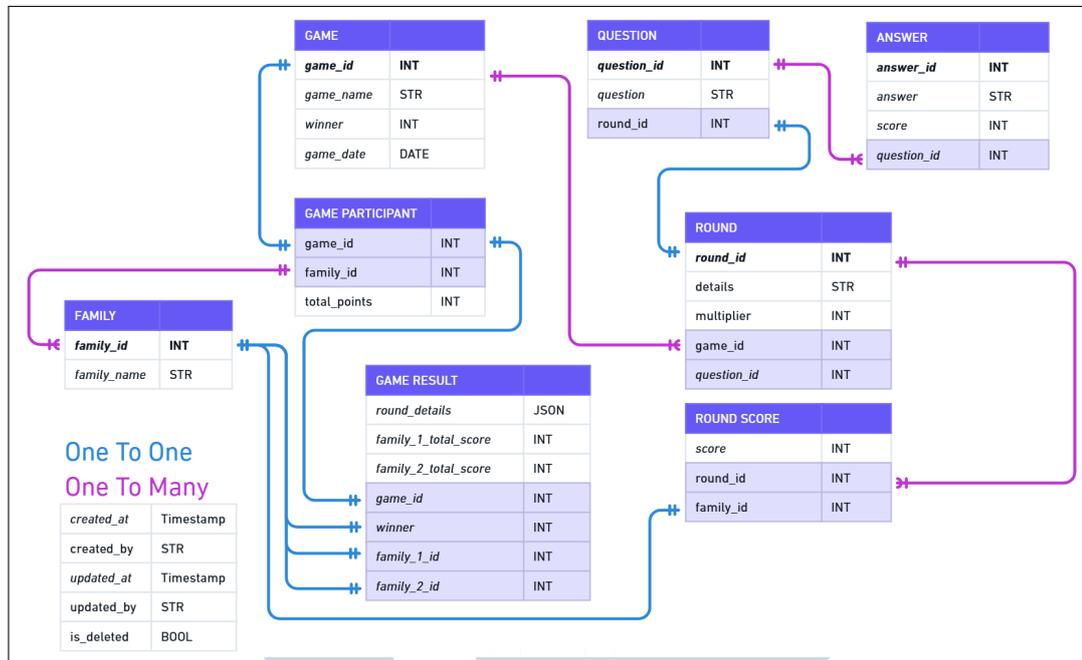


Gambar 3.6. Activity Diagram Aplikasi Permainan Kuis Tim – Bagian 3

B.4 Entity Relationship Diagram

ERD pada Gambar 3.7 memvisualisasikan struktur data dari proyek aplikasi Permainan Kuis Tim. Diagram ini berfungsi untuk menunjukkan relasi antara entitas dalam sistem, mencakup tabel-tabel utama seperti *family*, *game*, *question*, *answer*, *round*, dan *game_result*. Tabel *family* menyimpan informasi mengenai tim peserta, sementara tabel *game* berisi data permainan, seperti pemenang dan tanggal. Interaksi antara permainan dan tim diatur melalui tabel *game-participant*. Tabel *question* dan *answer* menyimpan pertanyaan serta jawaban untuk setiap ronde. Tabel *round* dan *round_score* menyimpan informasi tentang putaran dan nilai skornya. Hasil akhir permainan, yang mencakup total skor dan pemenang,

direkam dalam tabel *game_result*.

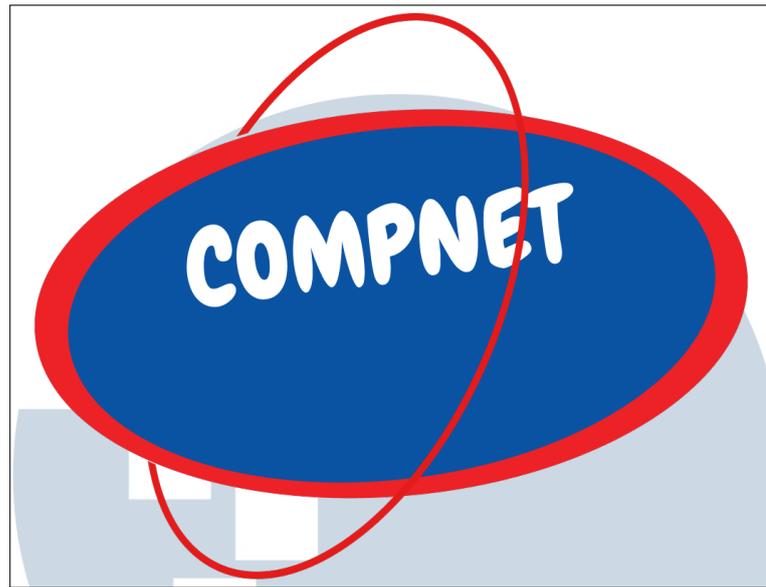


Gambar 3.7. ERD Aplikasi Permainan Kuis Tim

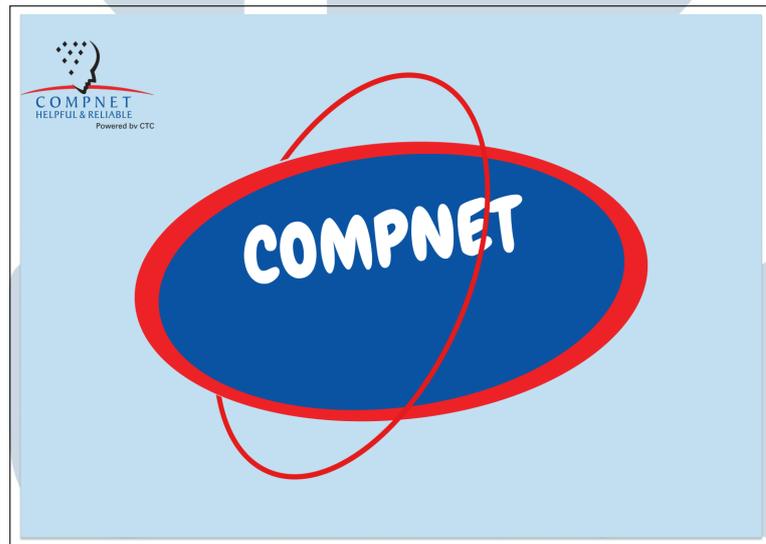
B.5 Prototype

Prototype dari aplikasi Permainan Kuis Tim dirancang menggunakan *Figma* untuk memberikan representasi visual yang menyeluruh terhadap rancangan antarmuka pengguna (*user interface*). Penyusunan prototype bertujuan untuk memastikan bahwa alur interaksi dan tampilan aplikasi telah sesuai dengan kebutuhan fungsional maupun pengalaman pengguna yang diharapkan.

Platform *Figma* dipilih karena mendukung kolaborasi desain secara daring dan memungkinkan pembuatan antarmuka yang dinamis serta mudah disesuaikan. Penggunaan visualisasi ini mempercepat proses validasi desain, meminimalkan revisi pada tahap implementasi, dan memastikan konsistensi tampilan aplikasi. Logo utama dari aplikasi Permainan Kuis Tim ditampilkan pada Gambar 3.8. Sementara itu, Gambar 3.27 memperlihatkan halaman awal atau *splash screen* yang ditampilkan saat aplikasi pertama kali dijalankan oleh pengguna.



Gambar 3.8. Figma Logo Permainan Kuis Tim



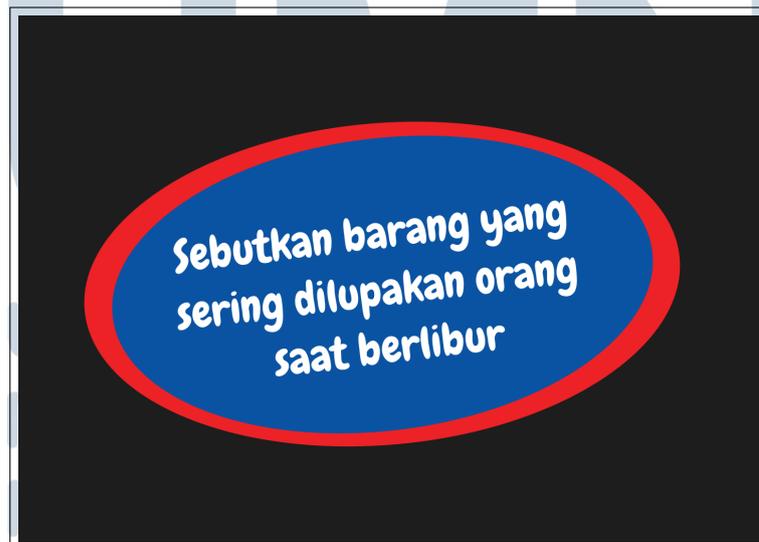
Gambar 3.9. Figma *Splash Screen* Permainan Kuis Tim

Setelah admin memulai permainan, peserta dan penonton akan di tampilkan halaman *Versus* yang memperlihatkan dua tim yang akan bertanding lengkap dengan nama tim masing-masing. Halaman *Versus* dapat dilihat di Gambar 3.10.



Gambar 3.10. Figma *Versus Page* Permainan Kuis Tim

Setelah ronde dimulai, sistem akan menampilkan halaman soal kuis yang diberikan kepada peserta, sebagaimana ditunjukkan pada Gambar 3.11. Soal ditampilkan secara jelas dan terpusat pada bagian tengah halaman untuk memudahkan peserta dalam membaca. Selanjutnya, Gambar 3.12 memperlihatkan tampilan soal beserta pilihan jawaban yang tersedia. Apabila peserta memilih jawaban yang tidak tepat, sistem secara otomatis menampilkan tanda silang sebagai indikator kesalahan.

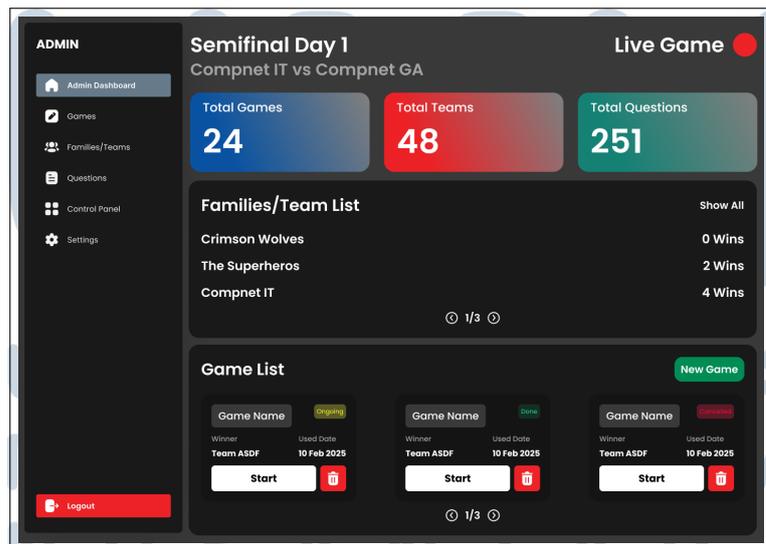


Gambar 3.11. Figma *Question Page* Permainan Kuis Tim

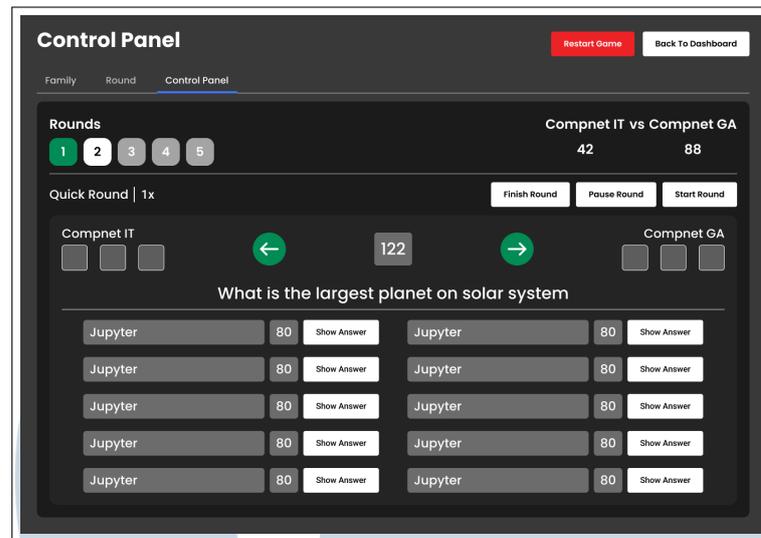


Gambar 3.12. Figma *Question & Answer Page* Permainan Kuis Tim

Gambar 3.13 memperlihatkan halaman *dashboard* yang dirancang khusus untuk admin. Halaman ini berfungsi untuk menampilkan status pertandingan secara keseluruhan. Sementara itu, Gambar 3.14 menampilkan halaman *control panel* yang digunakan oleh admin untuk mengatur jalannya kuis secara langsung, seperti memulai ronde, mengirimkan soal, serta mengungkap jawaban sesuai dengan alur permainan yang berlangsung.



Gambar 3.13. Figma *Dashboard* Permainan Kuis Tim



Gambar 3.14. Figma *Control Panel* Permainan Kuis Tim

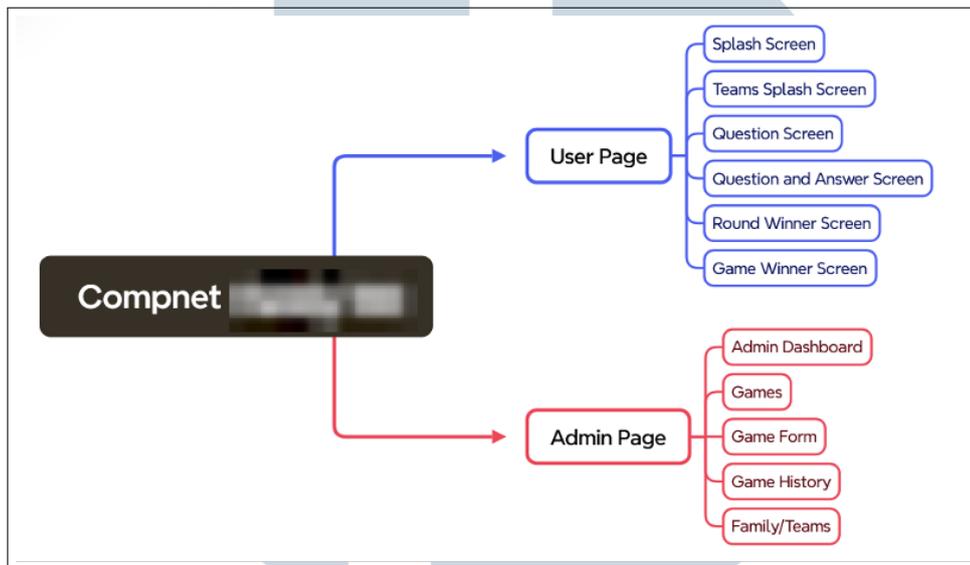
C Pengembangan

Proyek ini dikembangkan dengan menggunakan Laravel sebagai kerangka kerja utama di sisi *backend*, serta React untuk membangun antarmuka pengguna di sisi *frontend*. Keduanya terintegrasi melalui *Inertia.js*, yang memungkinkan interaksi langsung antara logika backend dan tampilan frontend. Laravel menggunakan pola arsitektur *Model-View-Controller* (MVC), sementara React berperan dalam menyajikan antarmuka pengguna yang dinamis dan responsif.

Untuk memenuhi kebutuhan pembaruan data secara langsung (*real-time update*), proyek ini mengimplementasikan *Socket.IO*. Sebelumnya, fitur ini sempat dikembangkan menggunakan *Pusher*, namun tidak dilanjutkan karena keterbatasan lisensi berbayar. Sebagai sistem manajemen basis data, proyek ini menggunakan *MySQL*, dengan pembuatan dan pengelolaan struktur databasenya dilakukan melalui fitur *migration* Laravel. Untuk keperluan visualisasi dan pengaturan basis data, digunakan aplikasi *HeidiSQL* yang memudahkan pengembang dalam melihat dan memanipulasi data selama proses pengembangan aplikasi.

Tampilan menu pada Gambar 3.15 menunjukkan struktur antarmuka pengguna yang terbagi menjadi dua bagian utama, yaitu halaman untuk admin dan halaman untuk penonton/peserta. Menu admin terdiri dari fitur-fitur seperti *dashboard*, *game form*, *family/team*, melihat *game history*, serta pengelolaan sesi *game*. Meskipun seluruhnya dapat diakses oleh admin, fitur pengontrol permainan (seperti pemilihan pertanyaan, penilaian, dan navigasi antar ronde) disediakan

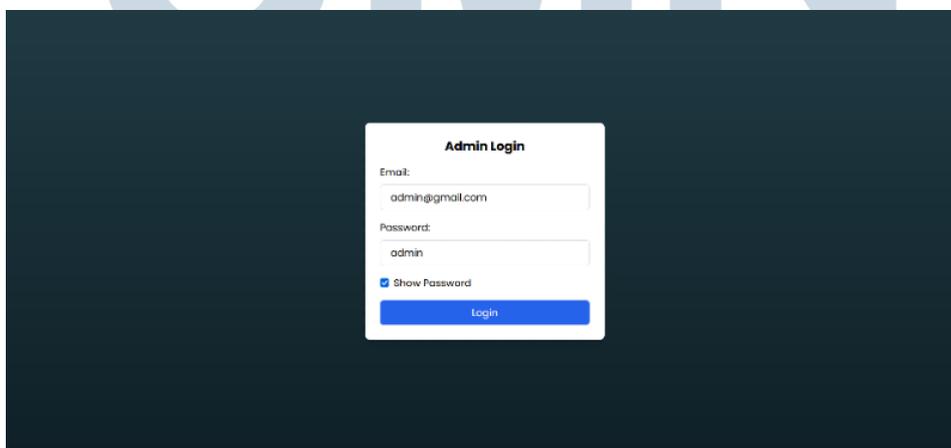
dalam halaman *control panel* terpisah untuk memudahkan pengelolaan selama kuis berlangsung. Sementara itu, tampilan untuk peserta hanya menampilkan informasi permainan secara real-time, seperti pertanyaan yang sedang aktif, jawaban yang telah ditemukan, skor masing-masing tim, serta pemenang ronde dan permainan.



Gambar 3.15. Menu Aplikasi Permainan Kuis Tim

C.1 Tampilan Admin

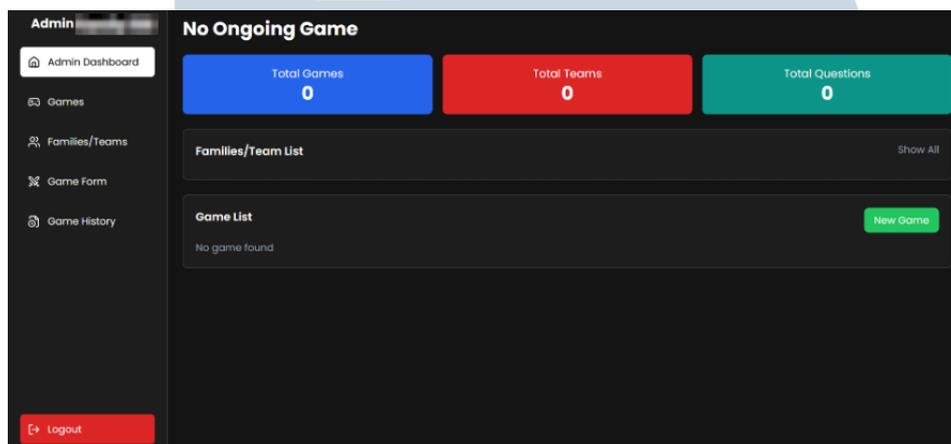
Halaman administrator dapat diakses melalui URL khusus yang menampilkan formulir autentikasi. Formulir ini meminta pengguna untuk memasukkan alamat surel dan kata sandi sebagai kredensial akses. Tampilan halaman *login* administrator ditunjukkan pada Gambar 3.16.



Gambar 3.16. Halaman Admin *Login* Aplikasi Permainan Kuis Tim

C.1.1 Halaman Dashboard

Setelah proses autentikasi berhasil, pengguna akan diarahkan ke halaman *dashboard*. Halaman ini berfungsi sebagai pusat kendali dan penyajian informasi statistik penting, seperti jumlah total permainan (*games*), tim peserta (*teams*), serta daftar pertanyaan (*questions*) yang tersedia dalam sistem. Tampilan antarmuka dari *dashboard* admin ditampilkan pada Gambar 3.17.

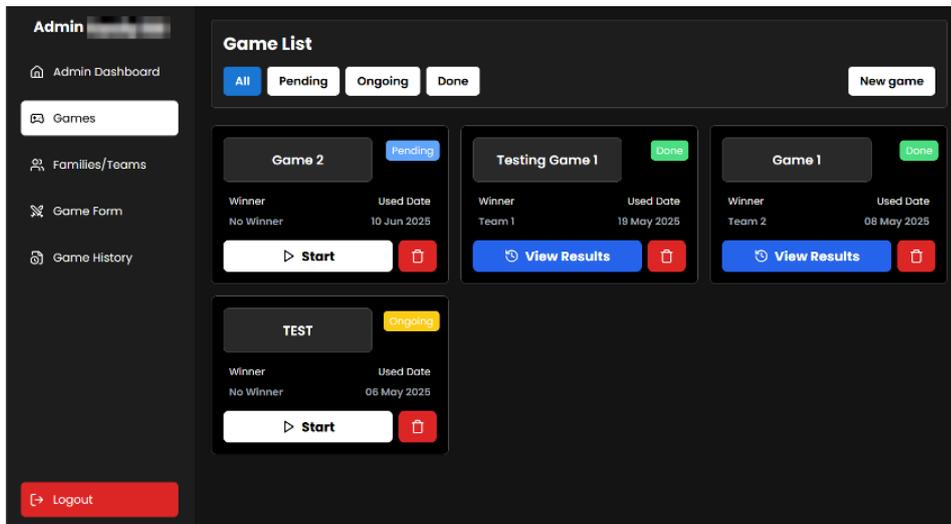


Gambar 3.17. Halaman Admin *Dashboard* Aplikasi Permainan Kuis Tim

C.1.2 Halaman Games

Halaman *Games* berfungsi sebagai pusat pengelolaan seluruh sesi permainan yang telah dibuat. Setiap permainan ditampilkan dalam bentuk *card* dengan informasi nama permainan, status (misalnya *Pending*, *Ongoing*, atau *Done*), waktu pelaksanaan, pemenang serta opsi untuk melihat detail. Tampilan ini memudahkan pemantauan status dan pengelolaan permainan secara terstruktur, sebagaimana ditunjukkan pada Gambar 3.18.

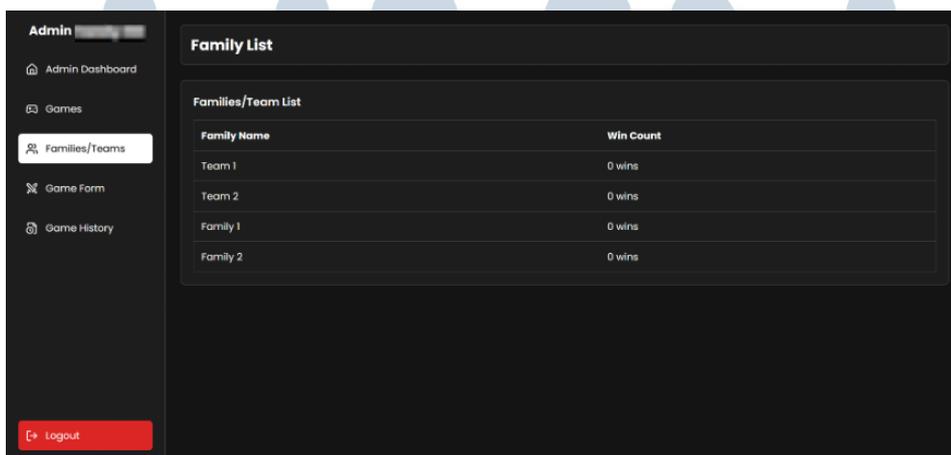
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.18. Halaman *Games* Aplikasi Permainan Kuis Tim

C.1.3 Halaman Families/Teams

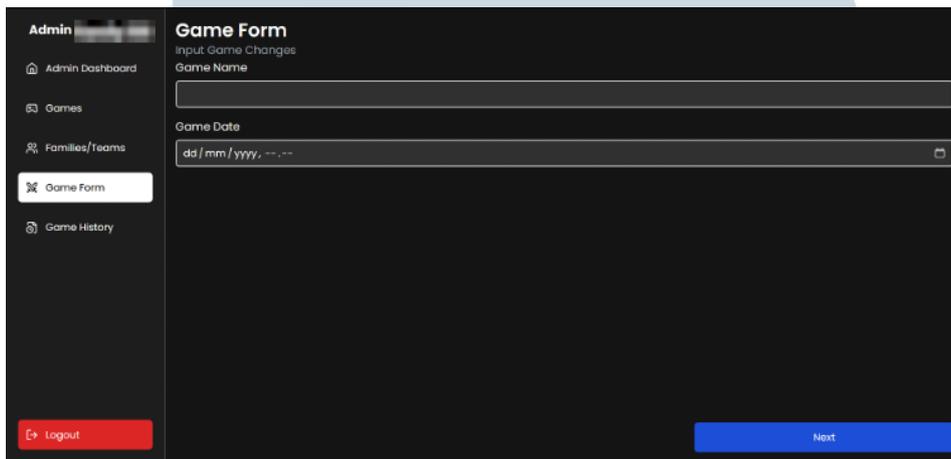
Halaman *Families* menyajikan daftar tim atau keluarga yang dapat dilihat di Gambar 3.19. Informasi yang ditampilkan hanya nama tim serta jumlah kemenangan yang telah diraih. Halaman ini bersifat informatif tanpa fitur pengelolaan langsung, dan ditujukan untuk memberikan gambaran umum mengenai performa masing-masing tim.



Gambar 3.19. Halaman *Families* Aplikasi Permainan Kuis Tim

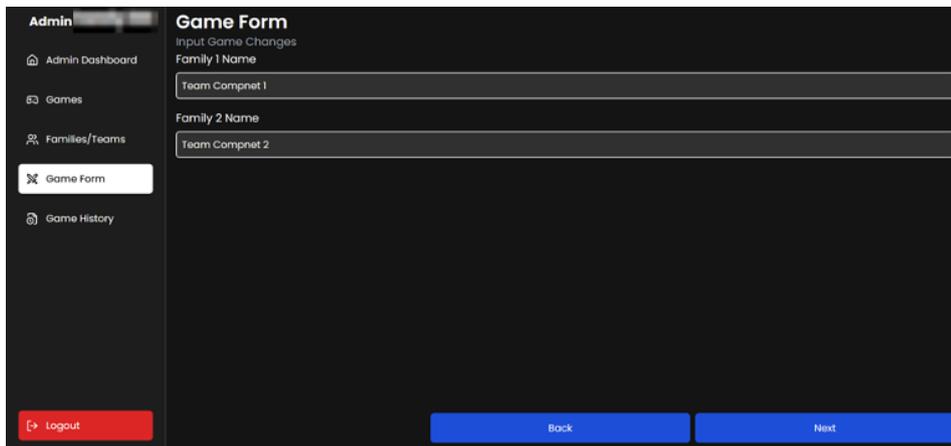
C.1.4 Halaman Game Form

Halaman *Game Form* dirancang secara bertahap dan modular guna mengakomodasi berbagai konfigurasi permainan. Bagian awal formulir (Gambar 3.20) memuat kolom untuk nama permainan dan waktu pelaksanaan. Selanjutnya, pengguna mengisi nama kedua tim yang berpartisipasi pada Gambar 3.21, lalu mengatur jumlah dan isi ronde permainan melalui Gambar 3.22.



The screenshot shows a dark-themed web interface. On the left is a sidebar with navigation items: Admin Dashboard, Games, Families/Teams, Game Form (highlighted), and Game History. At the bottom of the sidebar is a red 'Logout' button. The main content area is titled 'Game Form' with the subtitle 'Input Game Changes'. It contains two input fields: 'Game Name' (a simple text box) and 'Game Date' (a date picker showing 'dd/mm/yyyy, --:--'). At the bottom right of the main area is a blue 'Next' button.

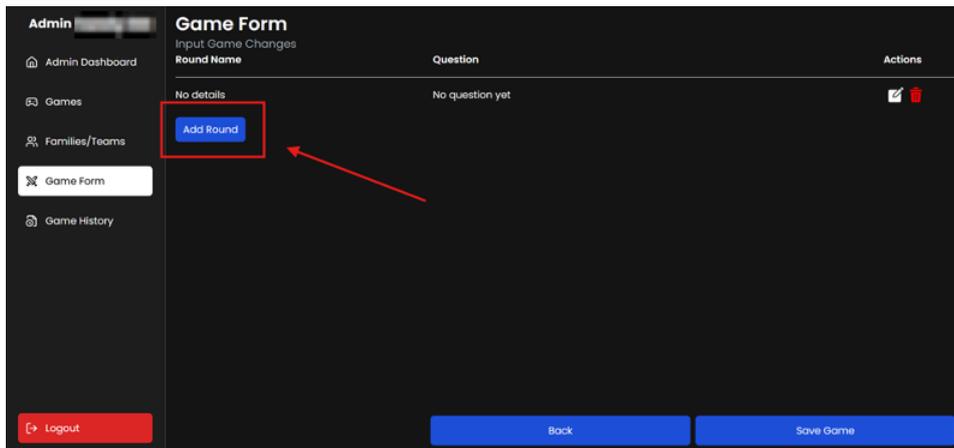
Gambar 3.20. *Game Form* Nama dan Waktu Permainan



The screenshot shows the same dark-themed web interface. The sidebar is identical. The main content area is titled 'Game Form' with the subtitle 'Input Game Changes'. It contains two input fields: 'Family 1 Name' (with 'Team Compnet 1' entered) and 'Family 2 Name' (with 'Team Compnet 2' entered). At the bottom of the main area are two blue buttons: 'Back' and 'Next'.

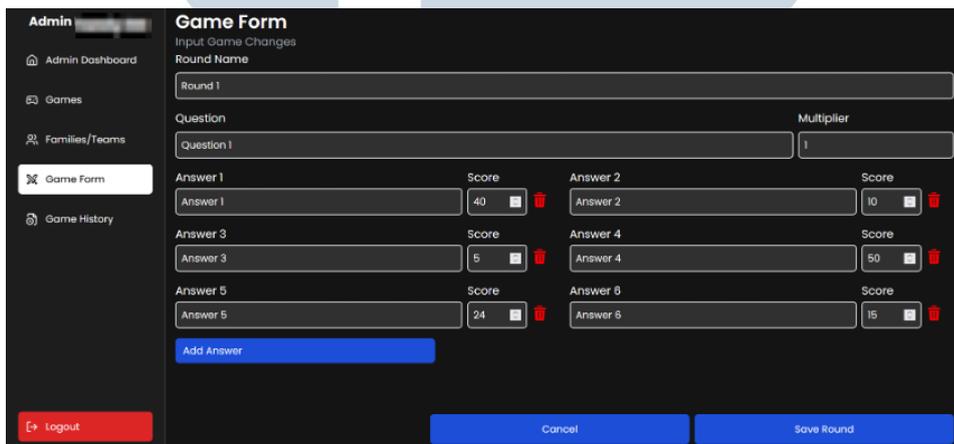
Gambar 3.21. *Game Form* Nama Tim

M U L T I M E D I A
N U S A N T A R A



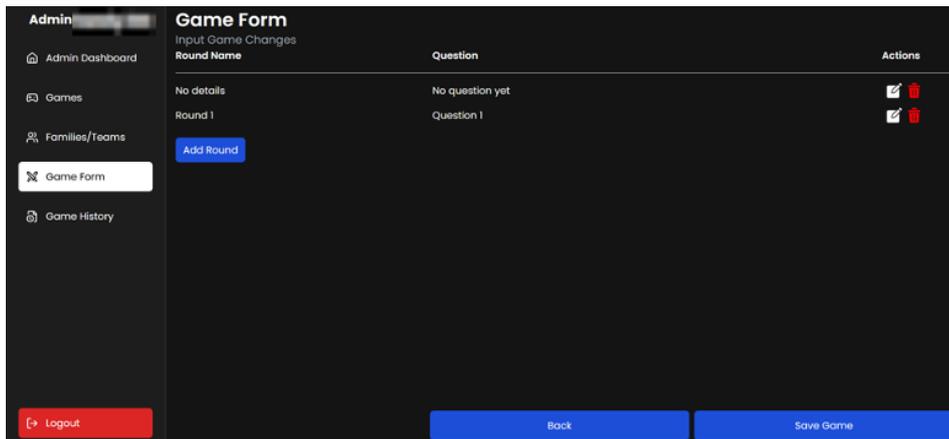
Gambar 3.22. *Game Form* Pengaturan Ronde

Penyusunan pertanyaan dan jawaban dilakukan pada Gambar 3.23 dan Gambar 3.24, memastikan materi kuis lengkap sebelum dijalankan.



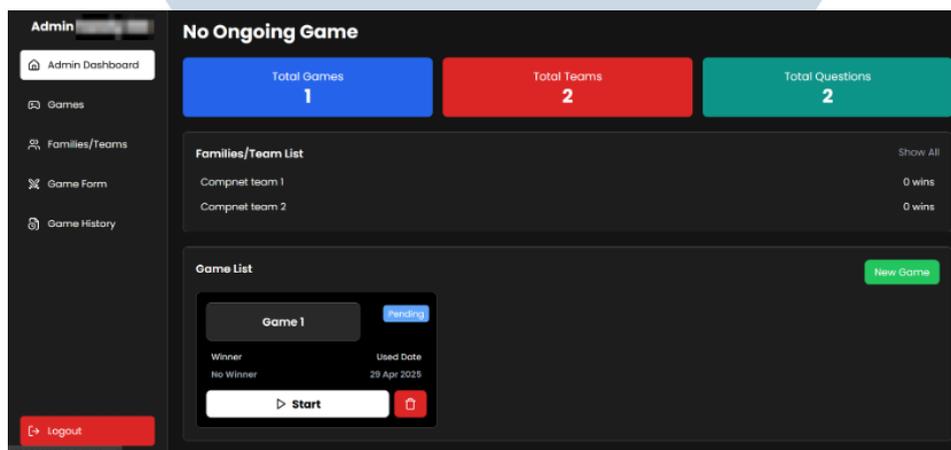
Gambar 3.23. *Game Form* Pertanyaan dan Jawaban – Bagian 1

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.24. *Game Form* Pertanyaan dan Jawaban – Bagian 2

Permainan yang telah dibuat akan muncul di halaman *dashboard* atau daftar permainan, seperti terlihat pada Gambar 3.25, siap untuk dikelola lebih lanjut.

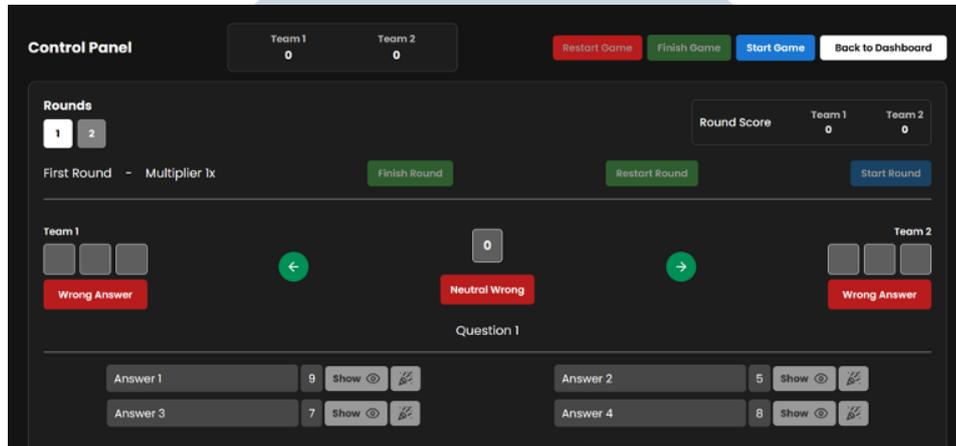


Gambar 3.25. Halaman admin *Dashboard* dengan permainan

C.1.5 Halaman Control Panel

Fitur utama dari sisi admin adalah kemampuan untuk mengelola jalannya permainan secara langsung melalui halaman *Control Panel*. Halaman ini menyediakan kontrol penuh terhadap setiap sesi permainan yang sedang berlangsung, seperti memulai ronde, menampilkan pertanyaan ke layar penonton, mengatur waktu, menampilkan skor, hingga menentukan pemenang pada akhir permainan. Seluruh interaksi ini didukung oleh integrasi *Socket.IO* yang memungkinkan pembaruan data secara *real-time* antara admin dan tampilan pengguna/penonton tanpa perlu memuat ulang halaman.

Antarmuka pada halaman ini dirancang agar interaktif dan intuitif, sehingga admin dapat menjalankan permainan dengan lancar dan tanpa hambatan teknis. Tampilan halaman *Control Panel* dapat dilihat pada Gambar 3.26.



Gambar 3.26. Halaman *Control Panel*

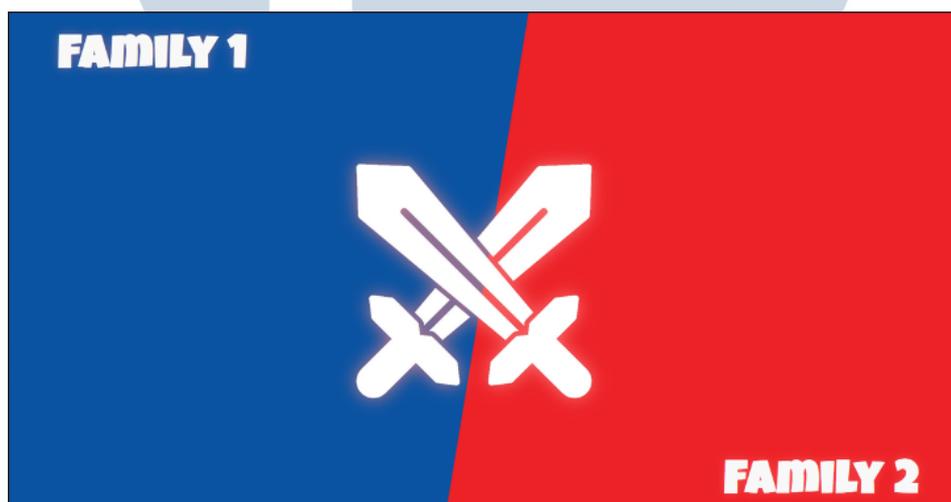
C.2 Tampilan User/Penonton

Antarmuka pengguna untuk penonton dirancang agar responsif terhadap status permainan secara real-time. Sebelum permainan dimulai, penonton akan disajikan halaman *splash screen* yang berfungsi sebagai layar penunggu, sebagaimana ditunjukkan pada Gambar 3.27. Setelah permainan diaktifkan oleh admin, tampilan secara otomatis diperbarui menjadi halaman yang menampilkan dua tim yang saling berkompetisi, lengkap dengan nama masing-masing tim seperti terlihat pada Gambar 3.28.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

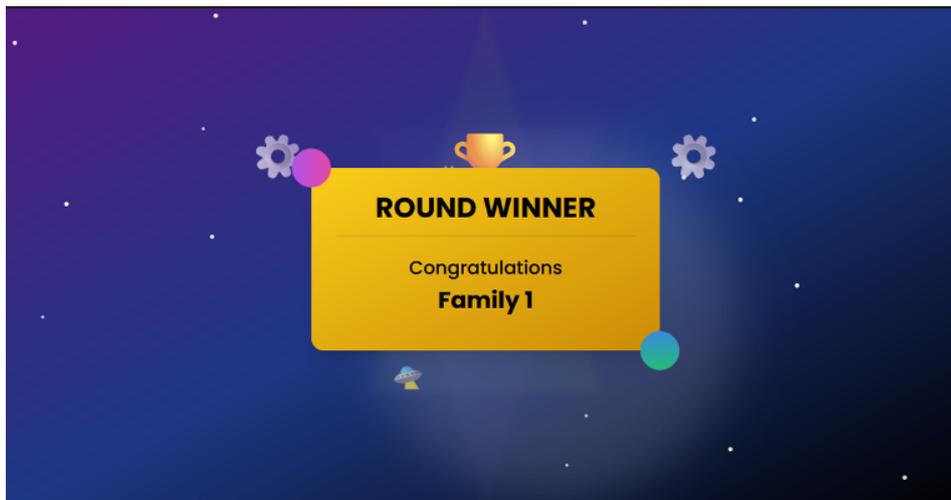


Gambar 3.27. Halaman *Splash Screen*

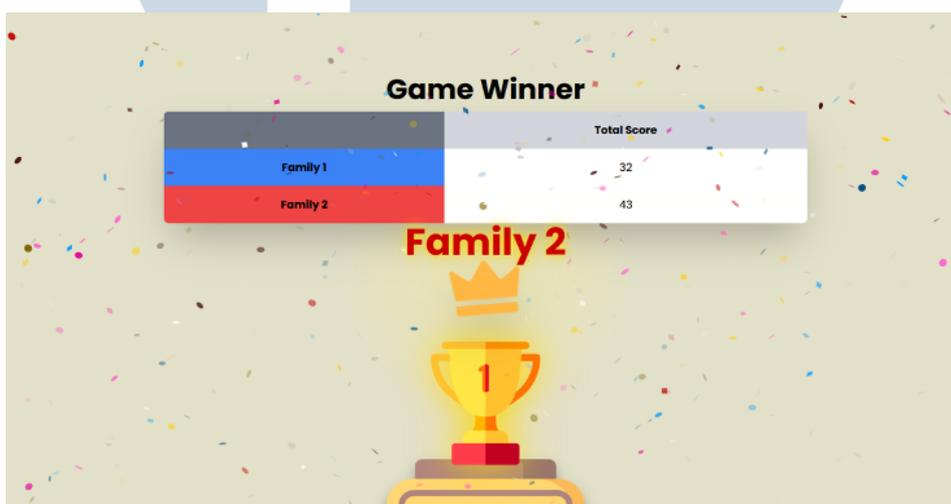


Gambar 3.28. Halaman Tim lawan Tim

Sepanjang permainan, ketika sebuah tim memenangkan satu ronde, sistem akan memunculkan halaman khusus yang menyoroti kemenangan tersebut. Ini menciptakan pengalaman visual yang interaktif bagi para penonton, seperti diperlihatkan pada Gambar 3.29. Usai pertandingan berakhir, halaman tersebut menampilkan pemenang utama pertandingan dalam bentuk animasi atau grafik perayaan, sebagaimana ditunjukkan pada Gambar 3.30.



Gambar 3.29. Halaman Menang Ronde Permainan



Gambar 3.30. Halaman Pemenang Permainan

3.3.3 Personal Protective Equipment (PPE)

A Requirements

Pada tahap perencanaan awal proyek aplikasi *PPE*, sejumlah fitur utama telah dirancang untuk mendukung sistem manajemen peminjaman Alat Pelindung Diri (APD) secara efektif. Fitur Form Peminjaman APD (FPA) memungkinkan pengguna mengajukan permintaan peminjaman melalui formulir digital yang tersedia. Selain itu, terdapat fitur *Extend* dan *Return* untuk memfasilitasi perpanjangan masa peminjaman secara mandiri serta pengembalian APD yang

dikelola oleh admin agar proses pengembalian berjalan akurat dan tepat waktu.

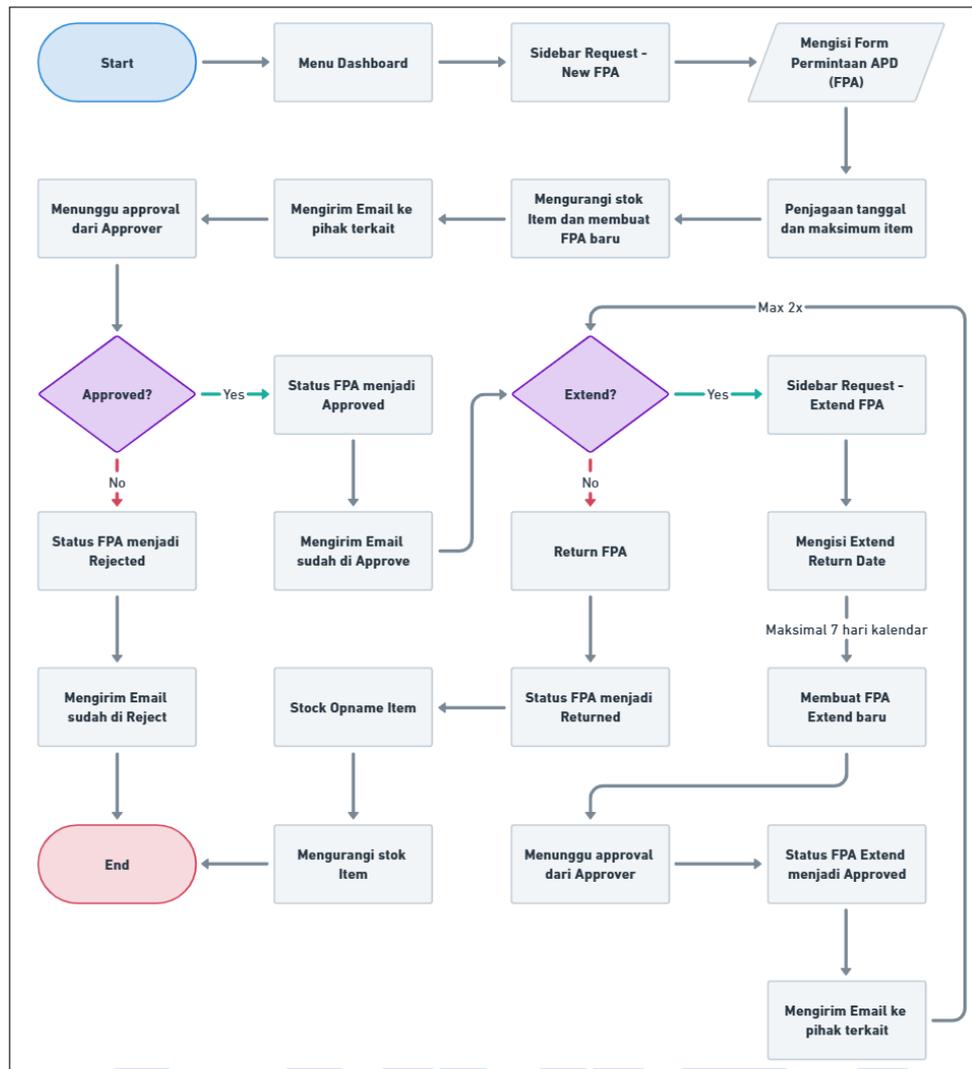
Untuk memastikan kontrol dan akuntabilitas, sistem juga menyediakan fitur *Approval* oleh tim HSE selaku pihak yang berwenang menyetujui permintaan peminjaman maupun perpanjangan. Sementara itu, fitur *Stock Opname* berperan dalam pemantauan dan pembaruan stok APD secara berkala, sehingga data ketersediaan tetap akurat dan dapat diakses langsung melalui sistem.

B Perancangan Sistem

Dalam proses perancangan, digunakan standar dokumentasi teknis seperti *Unified Modeling Language (UML)*, meliputi *Use Case Diagram*, *Activity Diagram* dan *Entity Relationship Diagram (ERD)*, untuk menggambarkan alur proses dan struktur data dalam sistem. Selain itu, *flowchart* disusun sebagai representasi visual dari alur kerja aplikasi secara umum, sehingga mempermudah pemahaman terhadap logika dan mekanisme kerja sistem secara keseluruhan.

B.1 Flow Chart

Gambar 3.31 menunjukkan alur utama dari proses permintaan dan pengelolaan Alat Pelindung Diri (APD) melalui sistem PPE. Proses dimulai dari pengisian form permintaan APD (FPA) oleh user hingga tahap akhir berupa pengembalian atau perpanjangan FPA. Setelah pengisian form, sistem akan menunggu persetujuan dari pihak approver. FPA yang telah disetujui dapat diperpanjang maksimal dua kali. Jika tidak diperpanjang, maka user akan melakukan pengembalian APD. FPA yang ditolak atau dikembalikan juga akan mengubah statusnya dalam sistem dan dilanjutkan dengan pengurangan stok saat dilakukan opname. Pada setiap tahap proses, mulai dari pembuatan FPA, persetujuan atau penolakan, hingga perpanjangan, sistem akan secara otomatis mengirimkan notifikasi melalui email kepada pihak terkait.

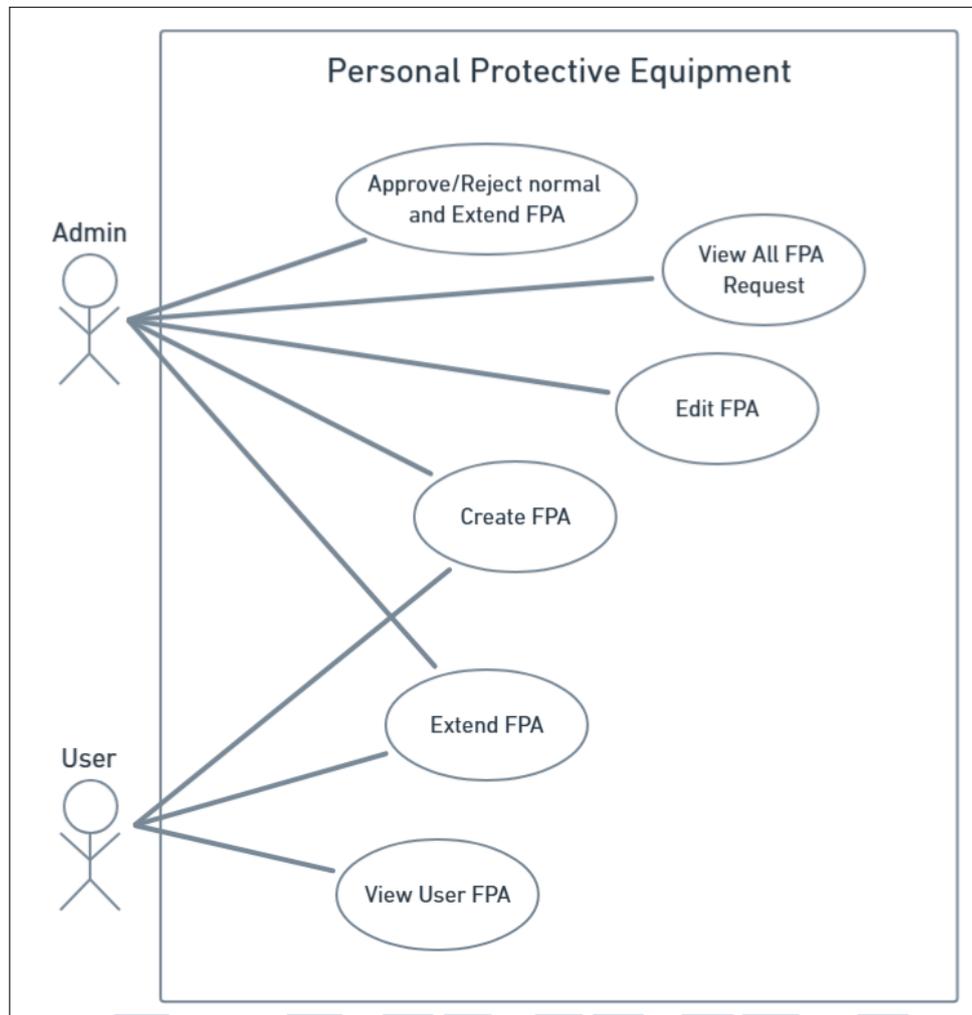


Gambar 3.31. Flowchart PPE

B.2 Use Case Diagram

Untuk mendapatkan pemahaman yang lebih baik mengenai bagaimana pengguna berinteraksi dengan sistem, dibuatlah sebuah *Use Case Diagram* sebagai representasi visual dari fungsi-fungsi utama serta aktor-aktor yang terlibat di dalam sistem. Diagram ini berperan dalam mengidentifikasi berbagai proses inti yang dilakukan oleh masing-masing peran pengguna selama penggunaan aplikasi berlangsung. Dalam konteks aplikasi PPE, aktor meliputi user atau peminjam, admin yang mengelola sistem, serta approver dari divisi HSE yang memiliki otoritas untuk menyetujui ataupun menolak pengajuan peminjaman alat pelindung diri (APD). Untuk referensi lebih lanjut, *Use Case Diagram* tersebut dapat dilihat pada

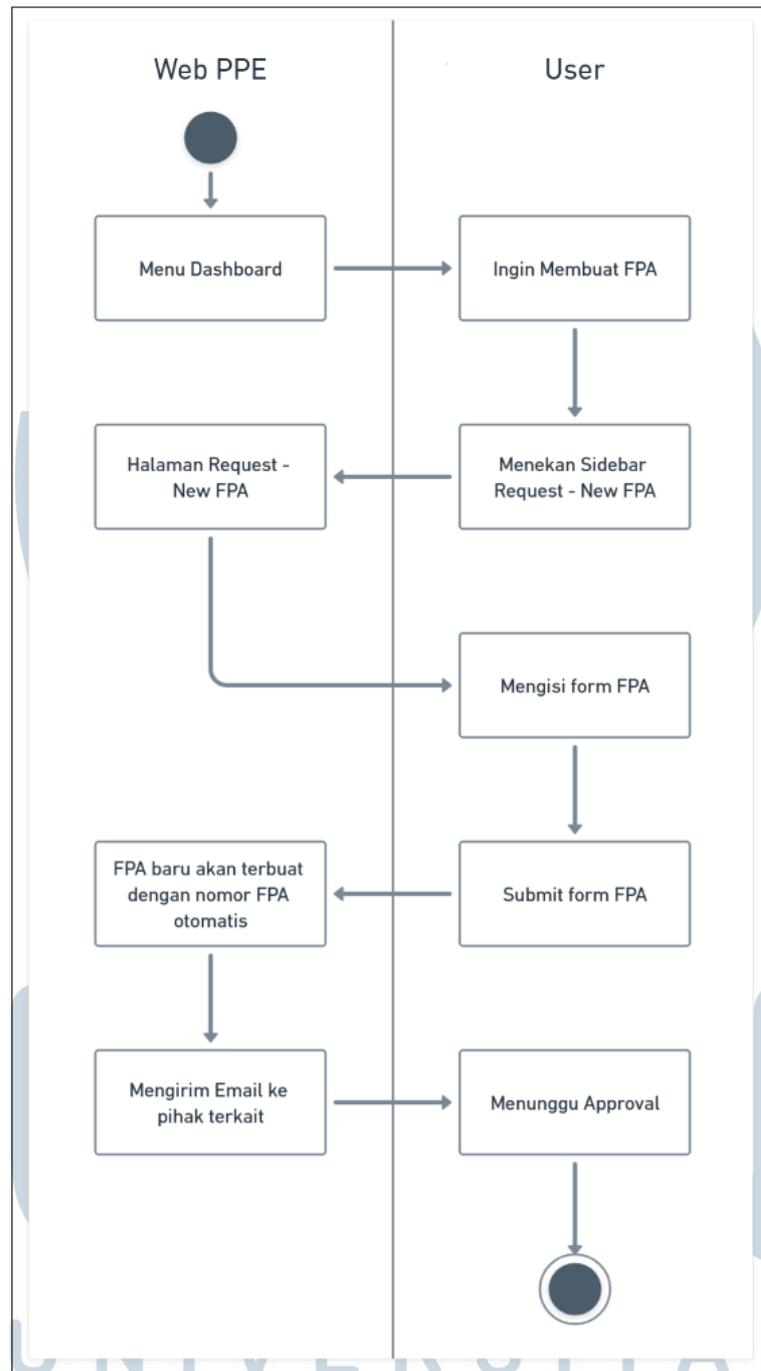
Gambar 3.32.



Gambar 3.32. Use Case Diagram PPE

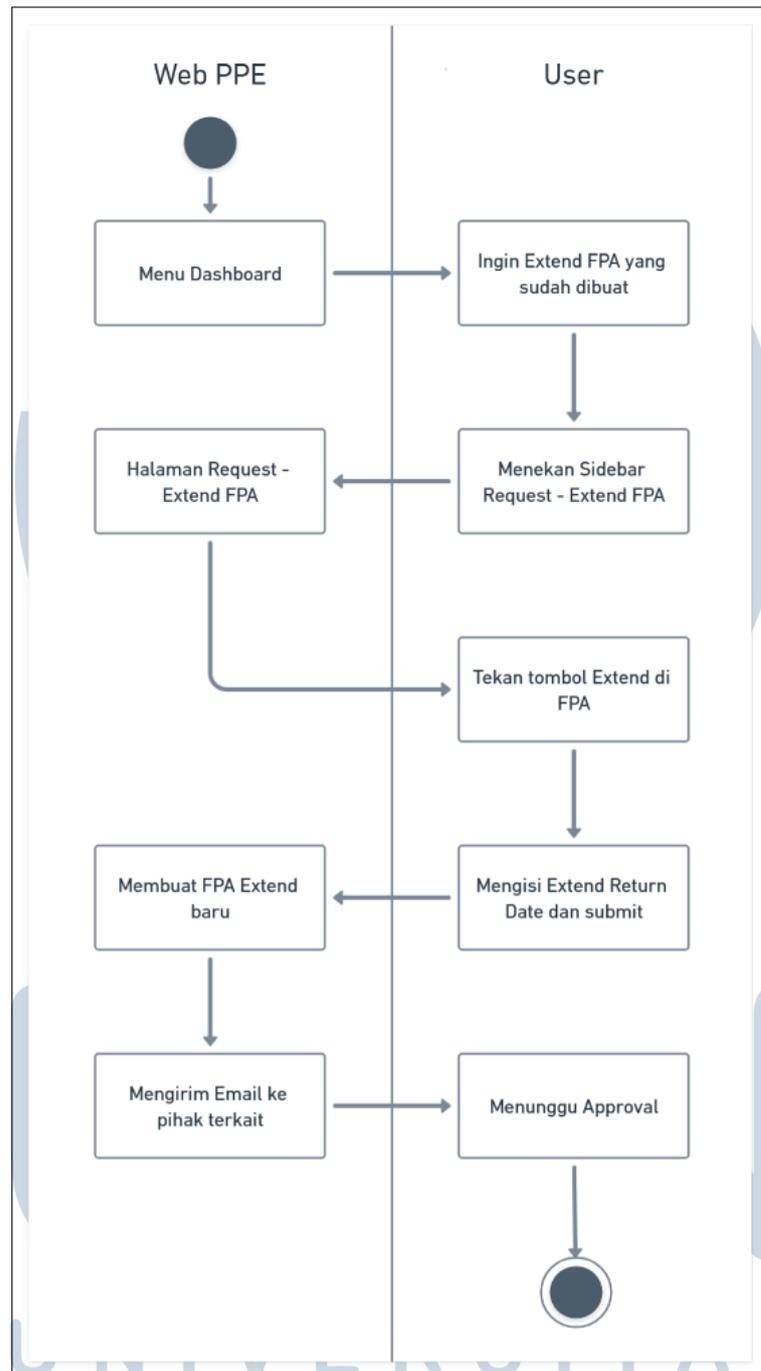
B.3 Activity Diagram

Setelah seluruh skenario utama dan aktor yang terlibat berhasil diidentifikasi melalui *Use Case Diagram*, tahap selanjutnya adalah mendetailkan alur proses yang terjadi di dalam sistem melalui *Activity Diagram*. Diagram ini menggambarkan urutan aktivitas dalam skenario tertentu, termasuk keputusan, pengulangan, serta interaksi antara pengguna dan sistem dalam menjalankan suatu proses. Gambar 3.33 menggambarkan alur aktivitas saat user membuat permintaan FPA baru. Proses ini dimulai dari user yang memilih menu request FPA, kemudian mengisi form dan mengirimkannya. Sistem secara otomatis menghasilkan nomor FPA dan mengirimkan notifikasi email ke pihak terkait untuk diproses lebih lanjut.



Gambar 3.33. Activity Diagram PPE Membuat FPA

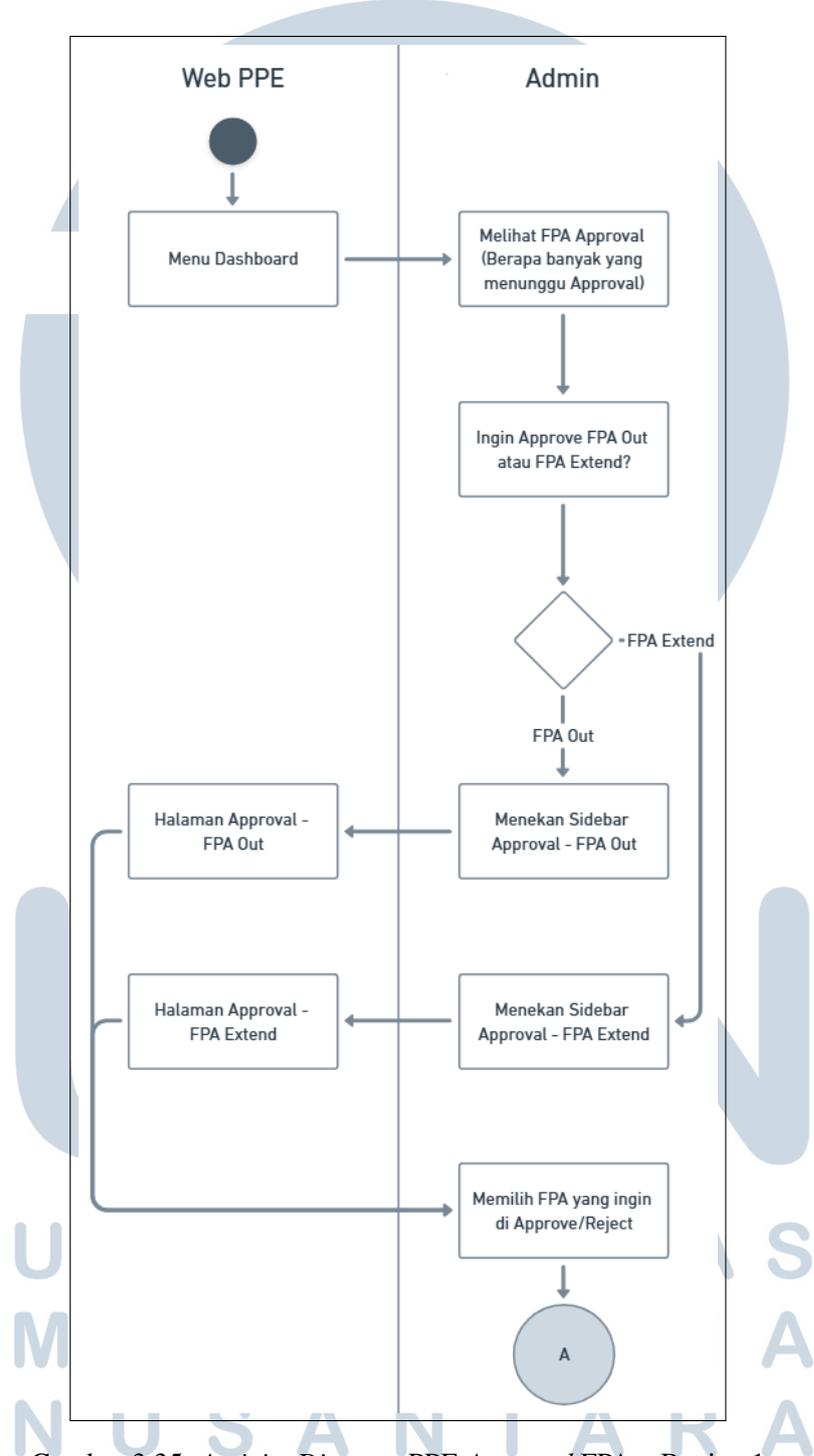
Gambar 3.34 menjelaskan proses perpanjangan (extend) FPA. User memilih FPA yang ingin diperpanjang, kemudian mengisi tanggal pengembalian baru. Sistem akan membuat FPA extend baru dan mengirimkan notifikasi kepada pihak terkait. Setelah itu, proses menunggu approval akan dimulai kembali.



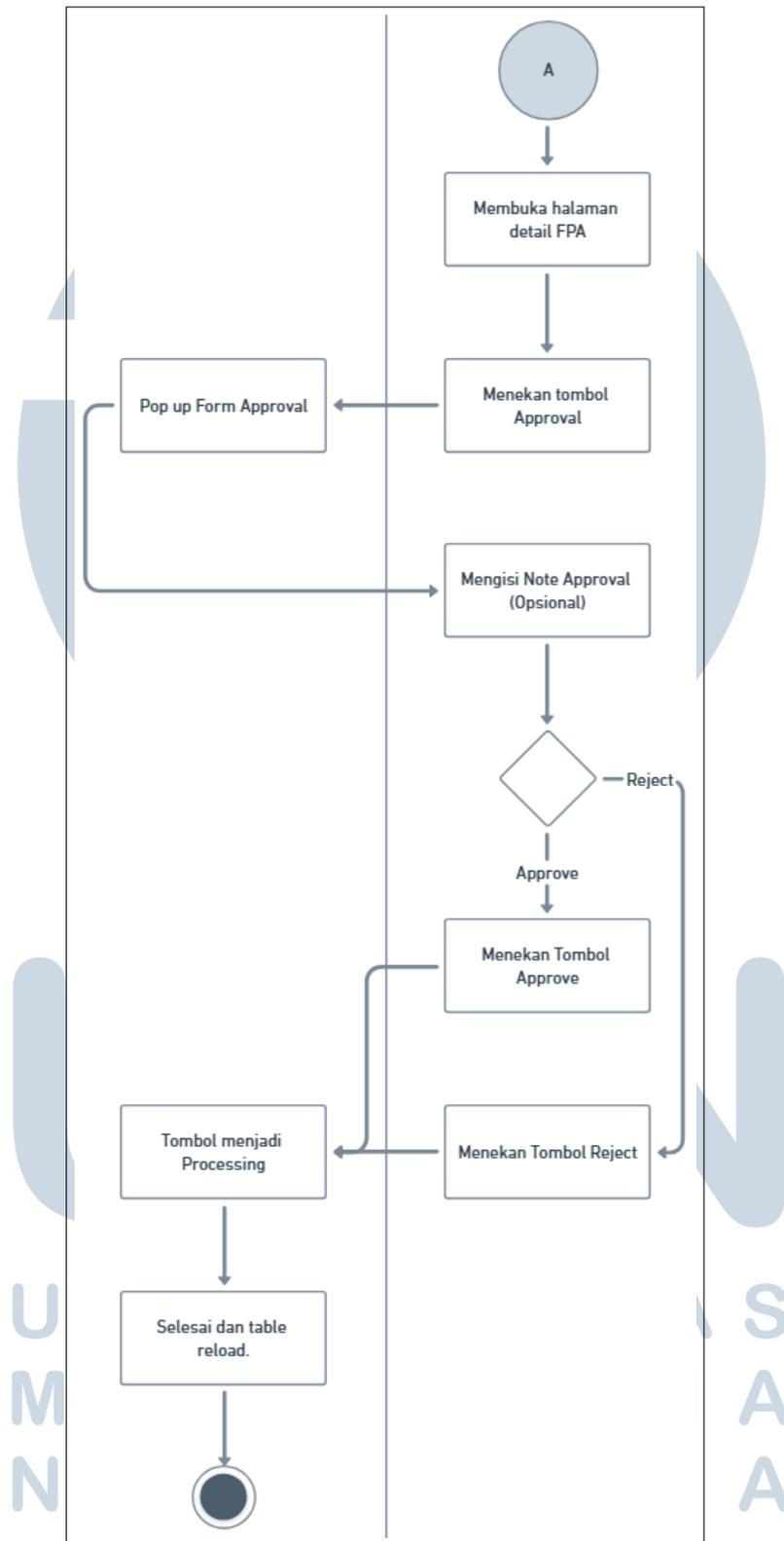
Gambar 3.34. Activity Diagram PPE Extend FPA

Gambar 3.35 dan 3.36 merupakan dua bagian dari proses approval FPA. Bagian pertama menjelaskan interaksi user saat membuka detail FPA, mengisi catatan (jika ada), dan memilih untuk menyetujui atau menolak. Bagian kedua menunjukkan proses dari sisi admin dalam memilih apakah akan menyetujui FPA out atau extend. Admin akan memilih FPA dari daftar yang tersedia dan

memberikan persetujuan atau penolakan sesuai kebutuhan. Seluruh proses ini akan memperbarui status FPA dan mengubah tampilan tabel secara otomatis.



Gambar 3.35. Activity Diagram PPE Approval FPA – Bagian 1

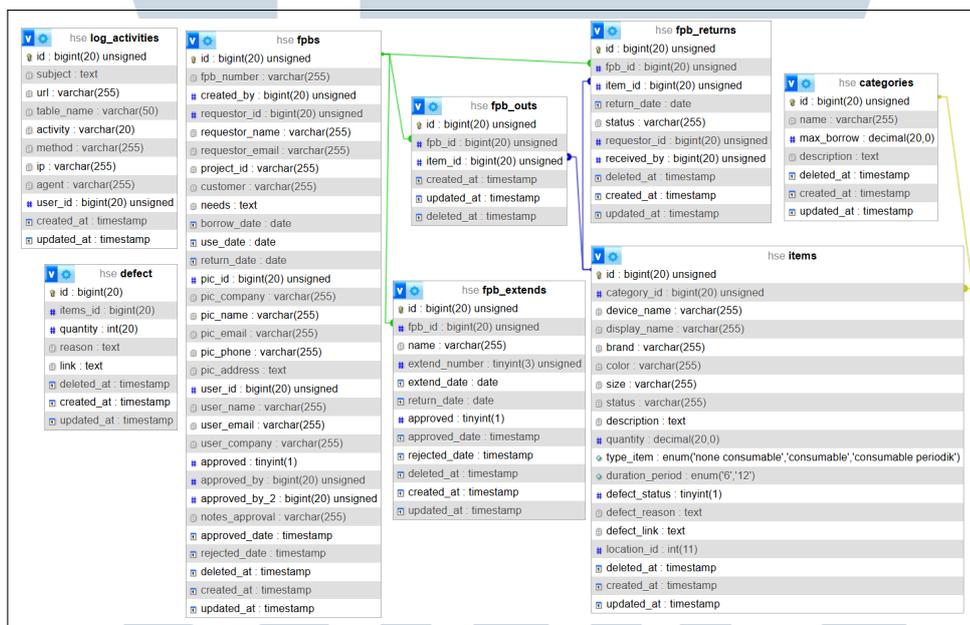


Gambar 3.36. Activity Diagram PPE Approval FPA – Bagian 2

B.4 Entity Relationship Diagram

ERD pada Gambar 3.37 memvisualisasikan struktur data dari sistem PPE. Diagram ini menunjukkan relasi antar entitas dalam sistem, mencakup tabel-tabel utama seperti *fpbs*, *items*, *fpb_outs*, *fpb_returns*, dan *fpb_extends*.

Tabel *fpbs* menyimpan data permintaan APD dari pengguna, termasuk informasi peminjam, tanggal pinjam dan kembali, serta persetujuan. Tabel *items* berisi detail setiap alat, seperti nama, kategori, kuantitas, dan status. Tabel *fpb_outs* dan *fpb_returns* mencatat proses pengeluaran dan pengembalian APD. Proses perpanjangan FPA dicatat dalam tabel *fpb_extends*. Selain itu, terdapat tabel *log_activities* untuk mencatat aktivitas sistem, serta tabel *defect* dan *categories* untuk pengelompokan serta pencatatan kerusakan barang.



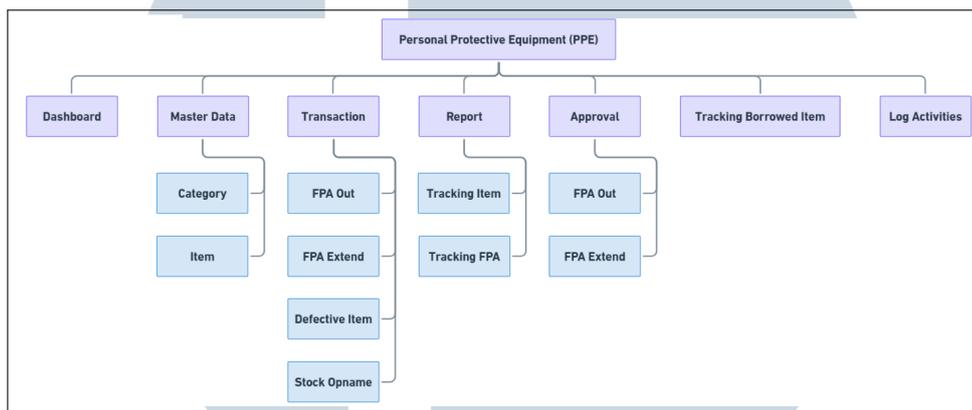
Gambar 3.37. ERD Aplikasi Permainan Kuis Tim

C Pengembangan

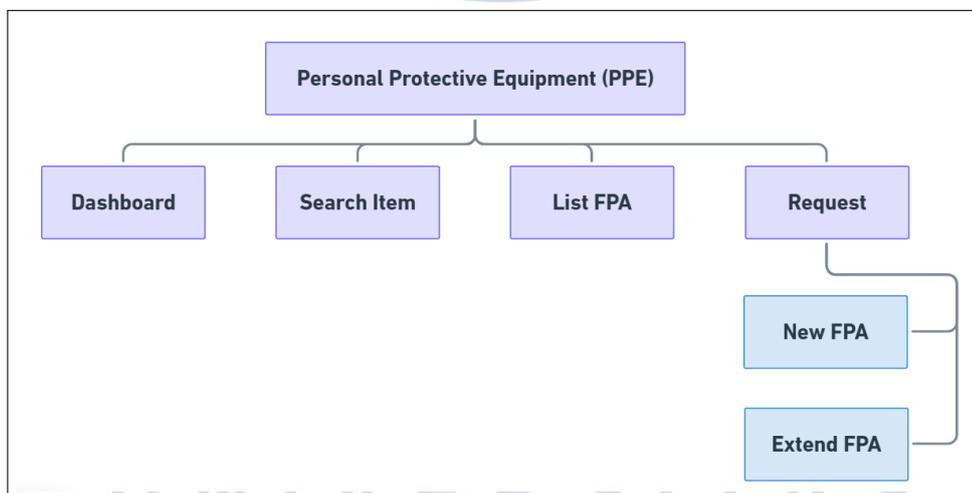
Aplikasi PPE dibuat menggunakan Laravel sebagai *full-stack framework* yang mengintegrasikan pengelolaan *backend* dan *frontend*. Sistem data menggunakan *MySQL*, sementara aplikasi *HeidiSQL* digunakan untuk mempercepat dan menyederhanakan proses interaksi dengan data, termasuk dalam hal pengujian *query*, tabel, serta manajemen data.

Antarmuka aplikasi diatur untuk mendukung dua kategori peran pengguna:

admin dan karyawan. Setiap peran diberikan akses dan tampilan menu yang berbeda, sesuai dengan kebutuhan dan hak aksesnya. Gambar 3.38 menunjukkan tampilan menu untuk peran admin, sedangkan Gambar 3.39 memperlihatkan tampilan menu yang dapat diakses oleh semua karyawan. Pembagian ini dirancang untuk mempermudah navigasi, memperjelas fungsi, serta menjaga keamanan dan integritas data dalam sistem.



Gambar 3.38. Menu Aplikasi PPE Admin

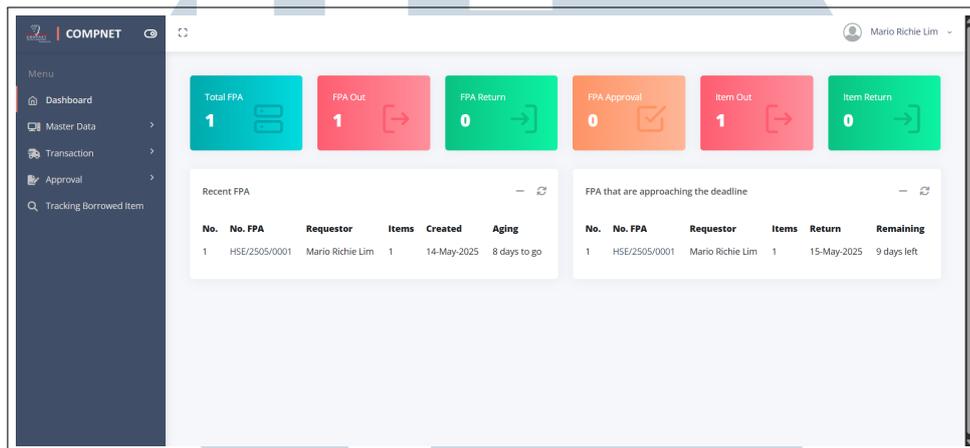


Gambar 3.39. Menu Aplikasi PPE Pengguna

C.1 Tampilan Antarmuka

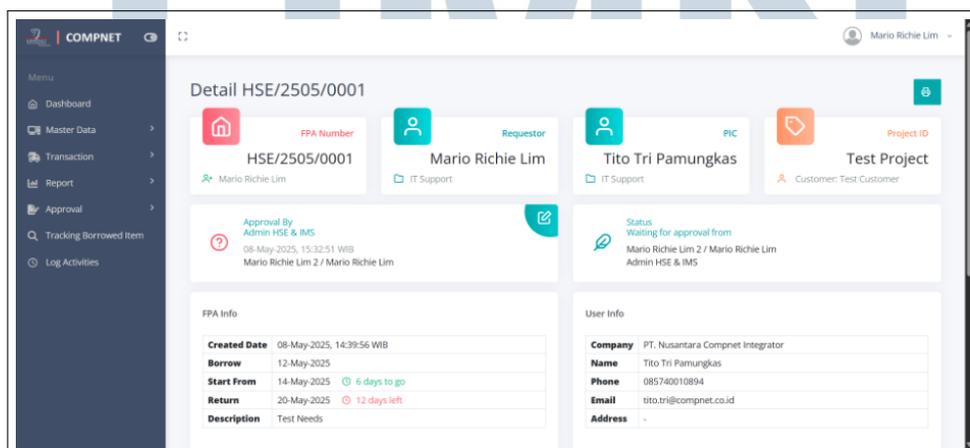
Tampilan antarmuka aplikasi disesuaikan berdasarkan hak akses pengguna yang telah ditentukan secara otomatis melalui data karyawan yang tersedia. Perbedaan akses ini terlihat dari elemen tampilan yang muncul, seperti tombol

aksi (contohnya *edit* dan *delete*) dan beberapa modul yang hanya tersedia untuk admin. Pengguna dengan peran admin memiliki hak akses penuh terhadap seluruh data FPA, sedangkan pengguna umum hanya diberikan akses terhadap data yang relevan dengan perannya dalam proses pengajuan, persetujuan, atau penggunaan APD. Tampilan *dashboard* aplikasi dapat dilihat pada Gambar 3.40.

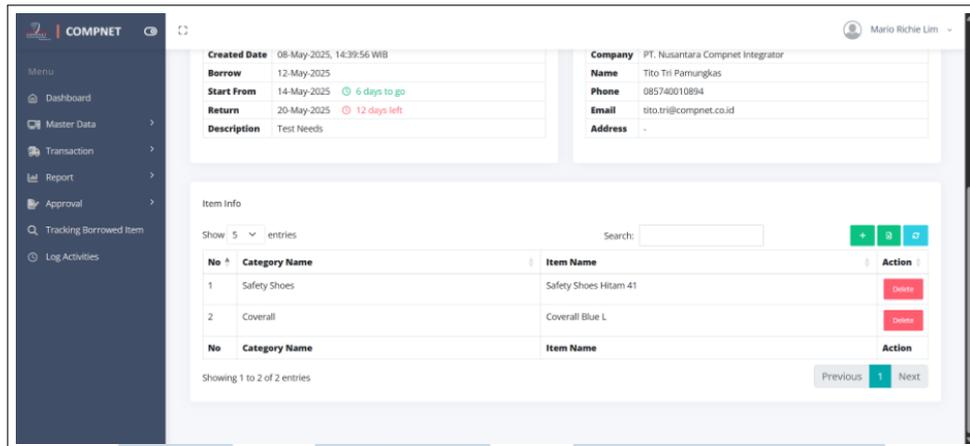


Gambar 3.40. Halaman *Dashboard* Admin

Pengguna juga dapat menavigasi ke halaman *detail* FPA dengan mengklik nomor FPA yang tersedia. Halaman ini menampilkan informasi lengkap dari suatu permintaan APD, mulai dari pihak yang terlibat, tanggal pinjam dan kembali, hingga daftar barang yang dipinjam dan statusnya. Tampilan halaman detail FPA dapat dilihat pada Gambar 3.41 dan Gambar 3.42.



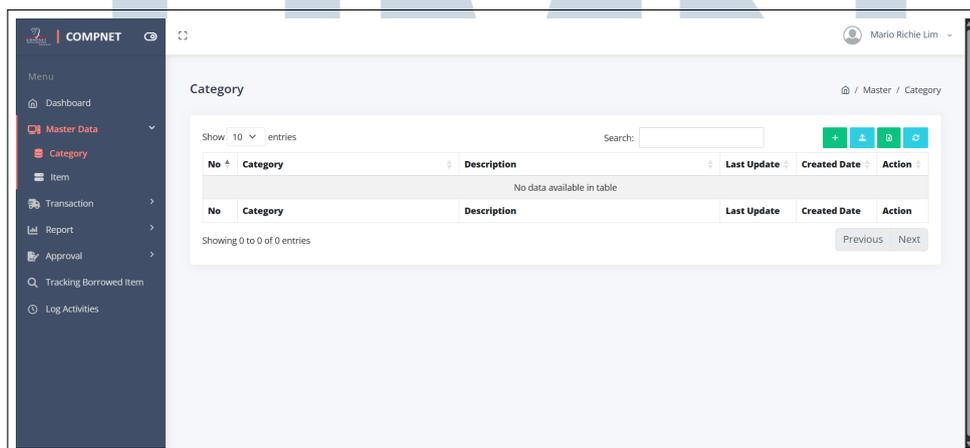
Gambar 3.41. Halaman *Detail* FPA – Bagian 1



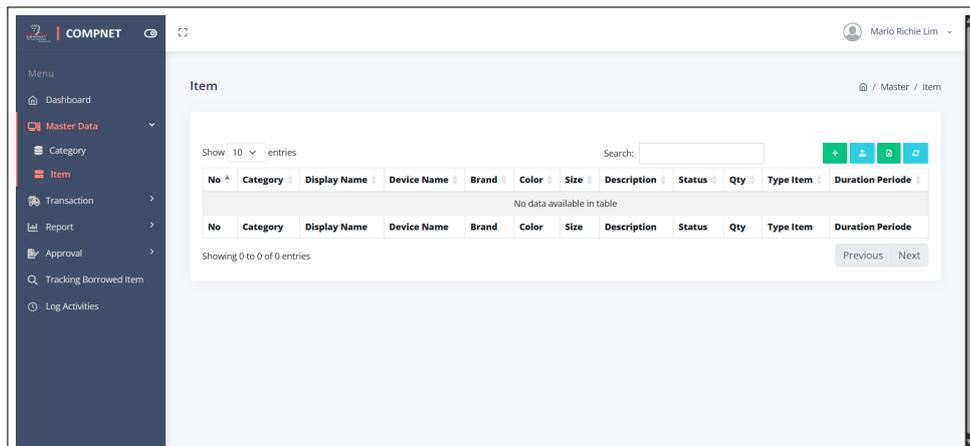
Gambar 3.42. Halaman *Detail FPA* – Bagian 2

C.1.1 Master Data Admin

Modul *Master Data* berfungsi sebagai pusat pengelolaan data APD beserta kategorinya. Fitur utama mencakup pembuatan kategori atau *item* baru, pemantauan yang sudah ada, *edit* dan penghapusan data. Untuk mempermudah proses pengelolaan, disediakan fitur *import* dan *export* data dalam format *spreadsheet*, yang mendukung pengelolaan data dalam jumlah besar dan memudahkan pelacakan riwayat perubahan. Tampilan dari modul ini dapat dilihat pada Gambar 3.43 untuk kategori dan Gambar 3.44 untuk *item*.

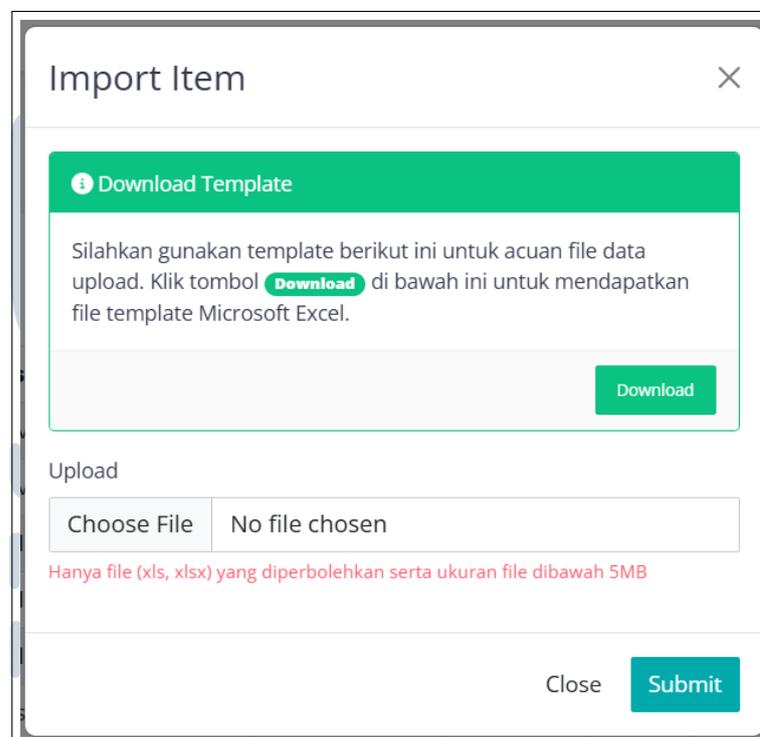


Gambar 3.43. Halaman *Master Data Category* PPE



Gambar 3.44. Halaman *Master Data Item PPE*

Tombol *import* dan *export* terletak di bagian kanan atas tabel. Saat tombol *import* ditekan, akan muncul modal yang menyediakan opsi untuk mengunduh *template spreadsheet* serta mengunggah file data yang telah disesuaikan dengan format tersebut. Sementara itu, tombol *export* berfungsi untuk langsung mengunduh data yang ditampilkan pada tabel ke dalam bentuk file spreadsheet. Tampilan modal untuk *import item* ditunjukkan pada Gambar 3.45.



Gambar 3.45. Modal *Import Item PPE*

C.1.2 Transaction FPA Admin

Modul *Transaction* pada sistem FPA berfungsi untuk mengelola berbagai aktivitas terkait FPA, meliputi *FPA Out* (pengajuan peminjaman yang sedang berjalan), *FPA Extend* (perpanjangan peminjaman), *Defect Item* (pengelolaan barang yang mengalami kerusakan), serta *Stock Opname* (pencocokan jumlah fisik Alat Pelindung Diri). Tampilan antarmuka untuk masing-masing fitur tersebut dapat dilihat pada Gambar 3.46 hingga Gambar 3.49.

FPA Out

Show: 10 entries

| No | No FPA | Requestor | PIC | User | Borrow | Start From | Return | Status |
|-----|---------------|------------------|--------------------|--------------------|-------------|-------------|-------------|---------------|
| ▶ 1 | HSE/2506/0003 | Mario Richie Lim | Mario Richie Lim | Mario Richie Lim 2 | 19-Jun-2025 | 23-Jun-2025 | 27-Jun-2025 | Returned |
| ▶ 2 | HSE/2506/0002 | Mario Richie Lim | Mario Richie Lim | Mario Richie Lim | 18-Jun-2025 | 19-Jun-2025 | 20-Jun-2025 | Approved |
| ▶ 3 | HSE/2506/0001 | Mario Richie Lim | Tito Tri Pamungkas | Tito Tri Pamungkas | 13-Jun-2025 | 16-Jun-2025 | 20-Jun-2025 | Need Approval |

Showing 1 to 3 of 3 entries

Gambar 3.46. Halaman *Transaction FPA Out* PPE

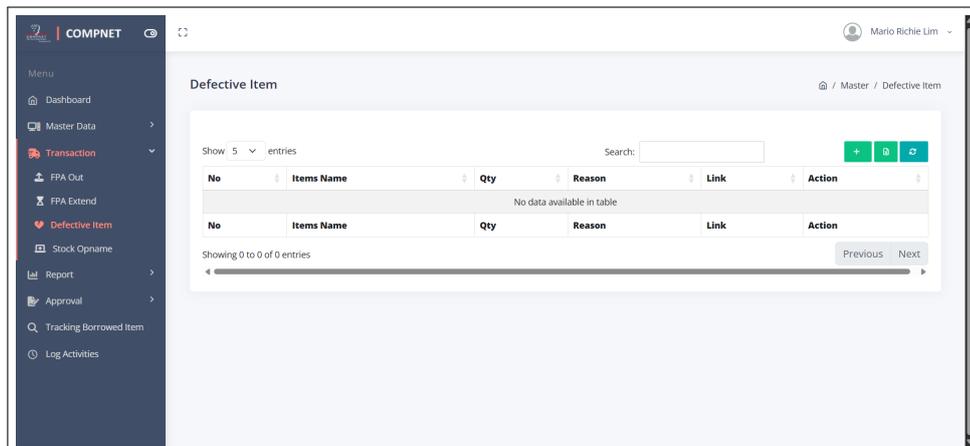
FPA Extend

Show: 5 entries

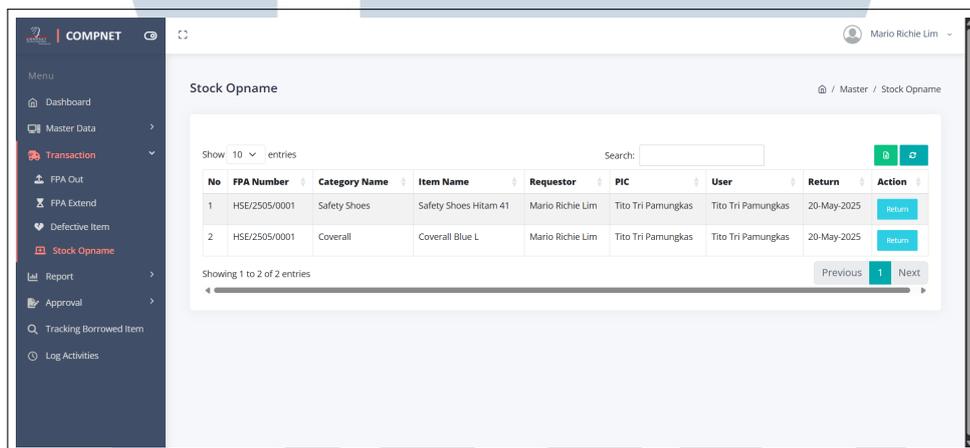
| No | No FPA | Extend | Requestor | PIC | User | Needs | Borrow | Return / Extend | Status |
|---------------------|---------------|---------|------------------|--------------------|--------------------|------------|-------------|-----------------|-----------------|
| ▼ 1 | HSE/2506/0003 | Not Yet | Mario Richie Lim | Mario Richie Lim | Mario Richie Lim 2 | 3333 | 19-Jun-2025 | 27-Jun-2025 | Ready To Extend |
| Total Item 2 | | | | | | | | | |
| ▶ 2 | HSE | Extend | Mario Richie Lim | Mario Richie Lim | Mario Richie Lim | aaa | 18-Jun-2025 | 20-Jun-2025 | Ready To Extend |
| ▶ 3 | HSE/2506/0001 | Not Yet | Mario Richie Lim | Tito Tri Pamungkas | Tito Tri Pamungkas | Test Needs | 13-Jun-2025 | 20-Jun-2025 | Ready To Extend |

Showing 1 to 3 of 3 entries

Gambar 3.47. Halaman *Transaction FPA Extend* PPE



Gambar 3.48. Halaman *Transaction Defect Item* PPE



Gambar 3.49. Halaman *Transaction Stock Opname* PPE

Untuk melakukan pengisian data FPA, tersedia tombol dengan simbol “+” yang terletak di bagian kanan atas tabel. Tombol ini akan memunculkan modal form FPA yang dapat digunakan untuk mengisi data peminjaman baru. Tampilan antarmuka untuk pengisian FPA *Out* ditunjukkan pada Gambar 3.50, sedangkan tampilan untuk pengajuan perpanjangan FPA *Extend* ditampilkan pada Gambar 3.51. Jika proses pengisian FPA berhasil, maka akan muncul jendela *pop-up* notifikasi berhasil, sebagaimana ditunjukkan pada Gambar 3.52.

Gambar 3.50. Modal FPA *Out PPE*

| No | Category Name | Item Name |
|----|---------------|-----------------------|
| 1 | Safety Shoes | Safety Shoes Hitam 42 |
| 2 | Masker N95 | Masker N95 Putih |
| No | Category Name | Item Name |

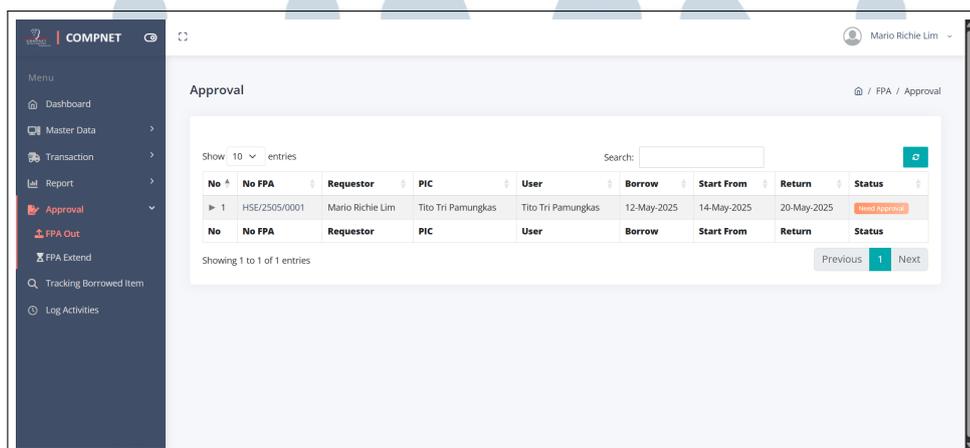
Gambar 3.51. Modal FPA *Extend PPE*



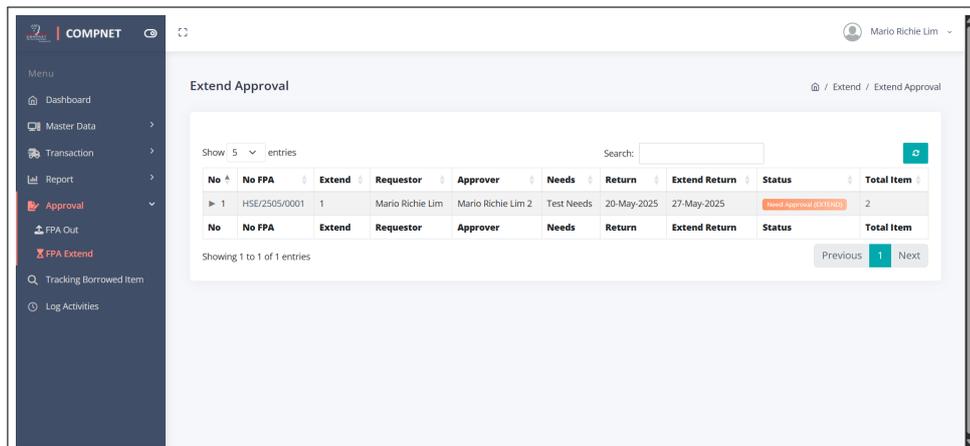
Gambar 3.52. *Pop-up* berhasil FPA PPE

C.1.3 Approval

Proses persetujuan bertujuan untuk memastikan validitas permintaan dan ketersediaan barang sesuai kebutuhan. Modul Approval terdiri dari dua jenis: persetujuan untuk permintaan awal (*FPA Out*) dan untuk perpanjangan permintaan (*FPA Extend*). Setiap permintaan hanya dapat diproses lebih lanjut apabila telah disetujui oleh pihak yang memiliki otorisasi sebagai approver. Gambar 3.53 menunjukkan halaman antarmuka untuk melakukan persetujuan *FPA Out*, sedangkan Gambar 3.54 menunjukkan halaman persetujuan untuk perpanjangan *FPA*.



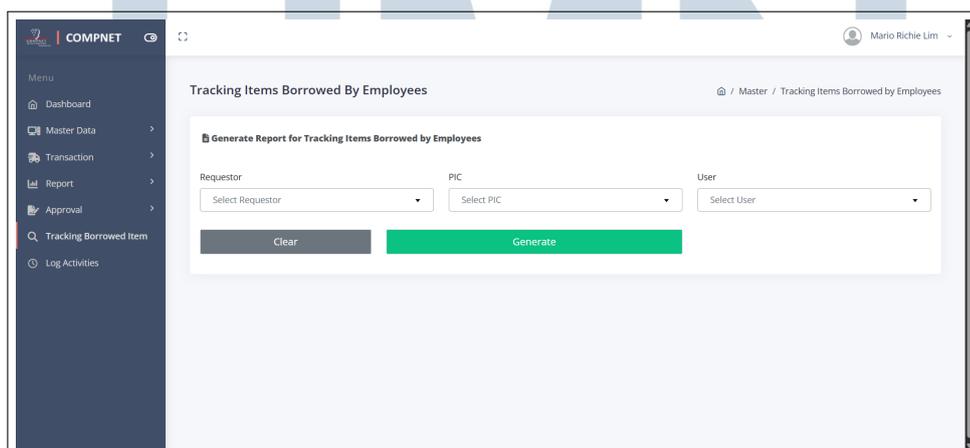
Gambar 3.53. Halaman *Approval FPA Out* PPE



Gambar 3.54. Halaman *Approval FPA Extend PPE*

C.1.4 Tracking Borrowed Item

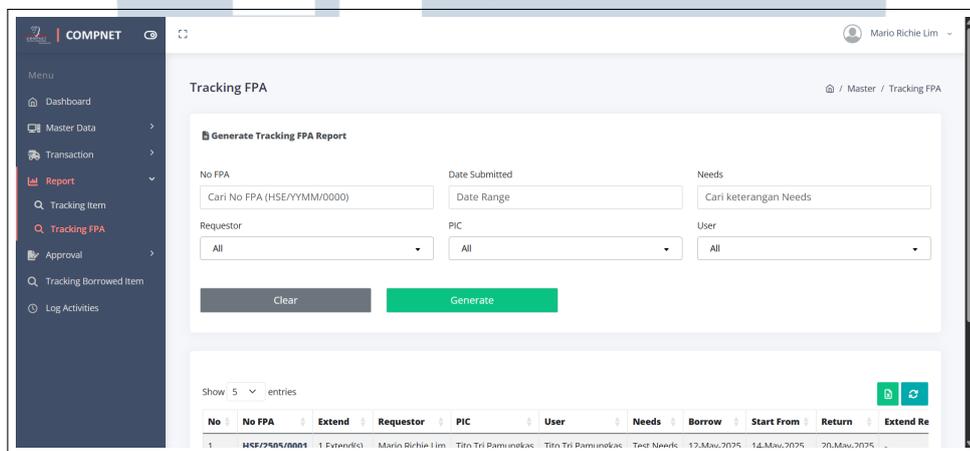
Modul ini digunakan untuk memantau status peminjaman APD untuk pengguna. Melalui halaman ini, *user* dapat melihat daftar APD yang sedang dipinjam, termasuk informasi detail seperti nama peminjam, nama barang, tanggal pinjam, tanggal kembali, serta status pengembalian. Fitur ini membantu dalam memastikan akuntabilitas penggunaan APD, serta memudahkan proses monitoring untuk keperluan perpanjangan atau pengembalian barang. Tampilan antarmuka dari modul ini ditunjukkan pada Gambar 3.55.



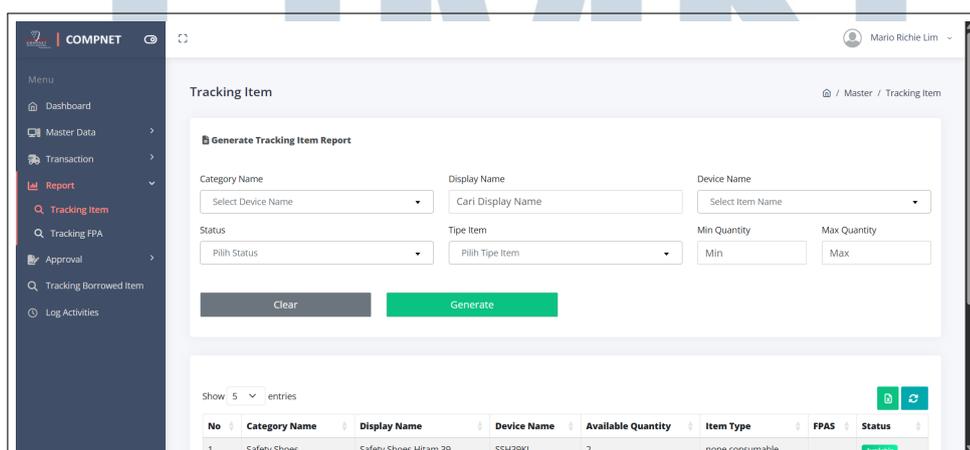
Gambar 3.55. Halaman Tracking Peminjaman APD

C.1.5 Report

Modul *Report* menawarkan dua fitur utama, yaitu *Tracking FPA* dan *Tracking Item*. Fitur ini dirancang untuk mendukung admin dalam melakukan pemantauan dan pelacakan data secara menyeluruh. *Tracking FPA* memaparkan daftar seluruh permintaan APD (FPA) yang telah diajukan. Sebaliknya, *Tracking Item* menyajikan informasi mengenai status setiap *item* dan nomor FPA yang meminjamnya. Dengan fitur ini, admin dapat dengan mudah menilai penggunaan dan mengelola stok APD secara lebih efektif. Antarmuka modul ini terlihat pada Gambar 3.56 dan Gambar 3.57.



Gambar 3.56. Halaman *Tracking FPA*



Gambar 3.57. Halaman *Tracking Item*

3.4 Kendala dan Solusi yang Ditemukan

3.4.1 Kendala

Selama pelaksanaan kegiatan magang, dialami beberapa kendala, baik teknis maupun non-teknis. Berikut adalah beberapa kendala utama yang dialami selama proses kerja magang:

- Penyesuaian terhadap alur kerja (*workflow*) serta pola komunikasi tim yang memerlukan adaptasi, terutama dalam hal kolaborasi antar tim.
- Kesulitan dalam berkomunikasi dengan *end user*, di mana terdapat perbedaan pemahaman bahasa teknis antara tim pengembang dan pengguna non-teknis.
- Banyaknya *bug* yang muncul selama proses pengembangan serta penyesuaian kebutuhan pengguna yang dinamis menyebabkan waktu pengembangan menjadi lebih lama.

3.4.2 Solusi

Dari berbagai kendala yang dihadapi selama proses kegiatan kerja magang, diperoleh beberapa solusi yang diterapkan untuk mengatasinya, antara lain:

- Aktif beradaptasi dengan ritme kerja tim melalui *Microsoft Teams*, *Whatsapp*, dan mengikuti alur kerja yang telah ditetapkan.
- Untuk meminimalkan miskomunikasi, digunakan bahasa yang sederhana dan diminta klarifikasi secara langsung saat diperlukan..
- Dalam menghadapi *bugs* dan perubahan kebutuhan, dilakukan pengujian berkala dan dicatat revisi untuk mempercepat identifikasi masalah.