LAPORAN AKHIR PROJECT WORK

SISTEM HAGAPLUS PADA PT. MORA CIPTA SOLUSI

Oleh :

LUKMAN MULUDIN 221011700337

SITI KHODIJAH 221011700488

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS PAMULANG

2026

SISTEM HAGAPLUS PADA PT. MORA CIPTA SOLUSI

PROJECT WORK

Laporan Ini Disusun Guna Memenuhi Salah Satu Syarat untuk Memperoleh Gelar Sarjana di Universitas Pamulang

Oleh :
LUKMAN MULUDIN 221011700337
SITI KHODIJAH 221011700488

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PAMULANG

2026

LEMBAR PERSETUJUAN PROJECT WORK

Judul Project Work	:	Sistem HagaPlus Pada PT. Mora Cipta Solusi
Mitra	:	PT. Mora Cipta Solusi
Alamat Mitra	:	Ruko Alicante D 56, Kel. Medang, Kec. Pagedangan, Kab. Tangerang, Provinsi Banten 15334
Jumlah Mahasiswa Terlibat	:	2 Mahasiswa
Biaya Project	:	-
Sumber Pendanaan	:	-
Jangka Waktu Pengerjaan	:	6 Bulan

Tangerang Selatan, 20 Juli 2026

Dosen Pembimbing Project	Koordinator Project
Heri Haerudin, S.Kom., M.Kom. NIDN. 0403028306	Siti Khodijah NIM. 221011700488
Pimpinan Mitra	Menyetujui, Ketua Program Studi Sistem Informasi
Geri Debataraja Simamora, S.Kom	Heri Haerudin, S.Kom., M.Kom. NIDN. 0403028306

LEMBAR PENGESAHAN PROJECT WORK

Judul Project Work	:	Sistem HagaPlus Pada PT. Mora Cipta Solusi
Mitra	:	PT. Mora Cipta Solusi
Alamat Mitra	:	Ruko Alicante D 56, Kel. Medang, Kec. Pagedangan, Kab. Tangerang, Provinsi Banten 15334
Jumlah Mahasiswa Terlibat	:	2 Mahasiswa
Biaya Project	:	-
Sumber Pendanaan	:	-
Jangka Waktu Pengerjaan	:	6 Bulan

Tangerang Selatan, 19 Agustus 2026

Penguji I	Penguji II
Fingki Marwati, S.Kom., M.Kom. NIDN. 0408039501	Ika Setiawan, S.Kom., M.Kom. NIDN. 0410129005

Dosen Pembimbing Project

Heri Haerudin, S.Kom., M.Kom.
NIDN. 0403028306

Mengetahui
Ketua Program Studi Sistem Informasi

Heri Haerudin, S.Kom., M.Kom.
NIDN. 0403028306

LEMBAR PERNYATAAN KEASLIAN PROJECT WORK

Kami yang bertanda tangan di bawah ini:

1. Nama : Lukman Muludin
   NIM : 221011700337
2. Nama : Siti Khodijah
   NIM : 221011700488

Menyatakan bahwa *Project Work* berjudul "**SISTEM HAGAPLUS PADA PT. MORA CIPTA SOLUSI"** merupakan karya asli kami. Apabila di kemudian hari, *Project Work* ini disinyalir bukan merupakan karya asli kami, maka kami bersedia untuk dibatalkan gelar sarjana kami beserta hak dan kewajiban yang melekat pada gelar tersebut dan kami siap mempertanggungjawabkan sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Tangerang Selatan, 20 Juli 2026

Yang Menyatakan,

(Materai 10000)	(Materai 10000)
Lukman Muludin	Siti Khodijah
221011700337	221011700488

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyelesaikan Laporan *Project Work* yang berjudul "Sistem HagaPlus pada PT. Mora Cipta Solusi" dengan baik.

Laporan ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pamulang. Selain itu, penyusunan laporan ini merupakan bentuk implementasi ilmu pengetahuan yang telah diperoleh selama masa perkuliahan melalui kegiatan perancangan dan pengembangan sistem informasi yang disesuaikan dengan kebutuhan dunia kerja.

Pelaksanaan *Project Work* ini memberikan banyak pengalaman dan pengetahuan kepada penulis, khususnya dalam menganalisis kebutuhan pengguna, merancang sistem, mengembangkan aplikasi berbasis web, serta bekerja sama dalam tim untuk menghasilkan solusi yang bermanfaat bagi mitra.

Penulis menyadari bahwa penyusunan laporan ini tidak akan terselesaikan tanpa adanya bantuan, bimbingan, dukungan, dan doa dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Dr. Pranoto, S.E., M.M., selaku Ketua Yayasan Sasmita Jaya yang telah memberikan dukungan dalam penyelenggaraan pendidikan di Universitas Pamulang.
2. Dr. Drs. E. Nurzaman AM, M.M., M.Si., selaku Rektor Universitas Pamulang yang telah memberikan kesempatan kepada penulis untuk menempuh pendidikan di Universitas Pamulang.
3. Dr. Yan Mitha Djaksana, S.Kom., M.Kom., selaku Dekan Fakultas Ilmu Komputer Universitas Pamulang yang telah memberikan dukungan dan motivasi selama proses perkuliahan.
4. Heri Haerudin, S.Kom., M.Kom., selaku Ketua Program Studi Sistem Informasi Universitas Pamulang yang telah memberikan arahan selama pelaksanaan *Project Work*.
5. Heri Haerudin, S.Kom., M.Kom., selaku Dosen Pembimbing *Project Work* yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan

bimbingan, arahan, serta masukan sehingga laporan *Project Work* ini dapat diselesaikan dengan baik.

6. Geri Debataraja Simamora, S.Kom., selaku Direktur PT Mora Cipta Solusi yang telah memberikan kesempatan, dukungan, serta informasi yang diperlukan selama pelaksanaan dan pengembangan Sistem HagaPlus.
7. Kedua orang tua dan keluarga yang senantiasa memberikan doa, kasih sayang, dukungan, serta motivasi kepada penulis selama proses penyusunan *Project Work*.
8. Rekan satu tim *Project Work* serta seluruh teman-teman Program Studi Sistem Informasi Universitas Pamulang yang telah memberikan bantuan, kerja sama, dan semangat selama proses penyusunan laporan ini.
9. Semua pihak yang tidak dapat disebutkan satu per satu yang telah memberikan bantuan, dukungan, dan kontribusi, baik secara langsung maupun tidak langsung, sehingga laporan *Project Work* ini dapat diselesaikan.

Penulis menyadari bahwa laporan *Project Work* ini masih memiliki keterbatasan dan jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan sebagai bahan evaluasi dan perbaikan di masa yang akan datang.

Tangerang Selatan, 12 Desember 2025

Hormat Kami,

Penulis

DAFTAR ISI

HALAMAN JUDUL ..... **i**
LEMBAR PERNYATAAN KEASLIAN PROJECT WORK ..... **iv**
LEMBAR PERSETUJUAN PROJECT WORK ..... **ii**
LEMBAR PENGESAHAN PROJECT WORK ..... **iii**
KATA PENGANTAR ..... **iv**
DAFTAR ISI ..... **vii**
DAFTAR GAMBAR ..... **xi**
DAFTAR TABEL ..... **xv**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.2 Maksud dan Tujuan 3
1.3 Sasaran 4
1.4 Mitra 5
1.5 Tim dan Pembagian Tugas 6
1.6 *Timeline* 7
BAB II ANALISIS DAN PERANCANGAN **8**
2.1 Analisis Sistem 8
2.2 Analisis Masalah 9
2.2.1 Analisis Kebutuhan Sistem 10
2.2.2 Analisis *Role* Sistem 12
2.3 Perancangan Sistem 14
2.4 Perancangan *Database* 15
2.4.1 *Entity Relationship Diagram* (ERD) 15
2.4.2 Transformasi *Entity Relationship Diagram* (ERD) ke *Logical Record Structure* (LRS) 17
2.4.3 *Logical Record Structure* (LRS) 19
2.5 Spesifikasi *Database* 20
2.5.1 Spesifikasi Tabel *activity_logs* 21
2.5.2 Spesifikasi Tabel *announcements* 22

2.5.3 Spesifikasi Tabel *attendance_policies* 23
2.5.4 Spesifikasi Tabel *attendance_revisions* 26
2.5.5 Spesifikasi Tabel *attendances* 27
2.5.6 Spesifikasi Tabel *branches* 29
2.5.7 Spesifikasi Tabel *cache* 30
2.5.8 Spesifikasi Tabel *cache_locks* 30
2.5.9 Spesifikasi Tabel *departments* 30
2.5.10 Spesifikasi Tabel *discount_usage* 31
2.5.11 Spesifikasi Tabel *discounts* 32
2.5.12 Spesifikasi Tabel *division_policies* 33
2.5.13 Spesifikasi Tabel *divisions* 35
2.5.14 Spesifikasi Tabel *email_templates* 35
2.5.15 Spesifikasi Tabel *employee_policies* 36
2.5.16 Spesifikasi Tabel *employee_payroll_components* 36
2.5.17 Spesifikasi Tabel *payroll_components* 37
2.5.18 Spesifikasi Tabel *employee_schedules* 38
2.5.19 Spesifikasi Tabel *employee_tax_forms* 39
2.5.20 Spesifikasi Tabel *employees* 40
2.5.21 Spesifikasi Tabel *failed_jobs* 43
2.5.22 Spesifikasi Tabel *holidays* 44
2.5.23 Spesifikasi Tabel *instansi_role_permissions* 44
2.5.24 Spesifikasi Tabel *instansi_roles* 45
2.5.25 Spesifikasi Tabel *instansis* 45
2.5.26 Spesifikasi Tabel *job_batches* 46
2.5.27 Spesifikasi Tabel *jobs* 47
2.5.28 Spesifikasi Tabel *leaves* 47
2.5.29 Spesifikasi Tabel *login_logs* 49
2.5.30 Spesifikasi Tabel *packages* 50
2.5.31 Spesifikasi Tabel *password_reset_tokens* 50
2.5.32 Spesifikasi Tabel *payment_methods* 51
2.5.33 Spesifikasi Tabel *payrolls* 52
2.5.34 Spesifikasi Tabel *permissions* 54

2.5.35 Spesifikasi Tabel *positions* 54
2.5.36 Spesifikasi Tabel *reimbursements* 55
2.5.37 Spesifikasi Tabel *sessions* 56
2.5.38 Spesifikasi Tabel *shift_swap_requests* 57
2.5.39 Spesifikasi Tabel *subscription_requests* 58
2.5.40 Spesifikasi Tabel *subscriptions* 59
2.5.41 Spesifikasi Tabel *system_roles* 60
2.5.42 Spesifikasi Tabel *system_settings* 60
2.5.43 Spesifikasi Tabel *tax_settings* 61
2.5.44 Spesifikasi Tabel *users* 62
2.6 *Unified Modeling Language* (UML) 63
2.6.1 *Use Case* 64
2.6.2 *Activity Diagram* 65
2.6.3 *Sequence Diagram* 88
2.6.4 *Class Diagram* 105
2.7 Perangkat yang Dibutuhkan 107
2.7.1 Perangkat Keras (*Hardware*) 107
2.7.2 Perangkat Lunak (*Software*) 107
BAB III IMPLEMENTASI DAN PENGGUNAAN 108
3.1 Implementasi dan Rancangan Waktu 108
3.2 Instalasi Sistem 111
3.2.1 Persyaratan Sistem 111
3.2.2 Instalasi Lokal 116
3.2.3 Panduan *Deployment* ke *Virtual Private Server* (VPS) 117
3.3 Rincian Perhitungan Penggajian 125
3.3.1 Rumus Perhitungan Penggajian 126
3.3.2 Sampel Perhitungan PPh Pasal 21 Nihil 129
3.3.3 Perhitungan PPh Pasal 21 Desember 130
3.3.4 Sampel Perhitungan PPh Pasal 21 Tidak Nihil 131
3.3.5 Perhitungan PPh Pasal 21 Desember 132
3.3.6 Formulir 1721-A1 134
3.4 Cara Penggunaan 135

3.4.1 Akses Awal Sistem 135
3.4.2 Penggunaan Sistem oleh *Superadmin* 137
3.4.3 Penggunaan Sistem oleh *Admin*/HR 140
3.4.4 Penggunaan Sistem oleh Karyawan 146
3.5 *Troubleshoot* 150
3.5.1 Masalah Tampilan & Aset (CSS / JS) 151
3.5.2 Masalah Koneksi & Konfigurasi *Database* 152
3.5.3 Masalah *Runtime* & *Server* 152
3.5.4 Masalah Fitur Aplikasi (Absensi & Notifikasi) 153
3.5.5 *Int*egrasi Pihak Ketiga (*Third-Party*) 154
BAB IV KESIMPULAN 156
4.1 Kesimpulan 156
LAMPIRAN 157

DAFTAR GAMBAR

Gambar 1.1 Logo PT. Mora Cipta Solusi 5
Gambar 2.1 *Entity Relationship Diagram* 16
Gambar 2.2 Transformasi ERD ke LRS 18
Gambar 2.3 *Logical Record Structure* 20
Gambar 2.4 *Use Case* 65
Gambar 2.5 Activity Diagram Register Subscription 67
Gambar 2.6 Activity Digram Login 68
Gambar 2.7 Activity Diagram Submit Attendance 69
Gambar 2.8 *Activity Diagram View Attendance* 70
Gambar 2.9 *Activity Diagram Submit Leave Request* 71
Gambar 2.10 Activity Diagram Submit Reimbursement 72
Gambar 2.11 Activity Diagram Swap *Shift* 73
Gambar 2.12 Activity Diagram View Salary Slip 74
Gambar 2.13 Activity Diagram Up*date* Profile Karyawan 75
Gambar 2.14 Activity Diagram Manage Employee 76
Gambar 2.15 Activity Diagram Manage Attendance 77
Gambar 2.16 Activity Diagram Manage Employee Request 78
Gambar 2.17 Activity Diagram Company Profile 79
Gambar 2.18 Activity Diagram Manage *Branches* 80
Gambar 2.19 Activity Diagram Manage Payroll 81
Gambar 2.20 *Activity Diagram Manage Subscription* 82
Gambar 2.21 *Activity Diagram Manage Package* 83
Gambar 2.22 *Activity Diagram Manage Users* 84
Gambar 2.23 *Activity Diagram Manage Instansi* 85
Gambar 2.24 *Activity Diagram Manage Payment Methods* 86
Gambar 2.25 *Activity Diagram Manage Announcement* 87
Gambar 2.26 *Sequence Diagram Registrasi Subscription* 89
Gambar 2.27 *Sequence Diagram Login* 90
Gambar 2.28 *Sequence Diagram Submit Attendance* 91

Gambar 2.29 ***Sequence Diagram View Attendance*** 91
Gambar 2.30 ***Sequence Diagram Submit Leave Request*** 92
Gambar 2.31 ***Sequence Diagram Submit Reimbursement*** 92
Gambar 2.32 ***Sequence Diagram Swap Shift*** 93
Gambar 2.33 ***Sequence Diagram View Salary Slip*** 94
Gambar 2.34 ***Sequence Diagram Update Profile*** 94
Gambar 2.35 ***Sequence Diagram Manage Employee*** 95
Gambar 2.36 ***Sequence Diagram Manage Attendance*** 96
Gambar 2.37 ***Sequence Diagram Manage Employee Request*** 97
Gambar 2.38 ***Sequence Diagram Manage Company Profile*** 97
Gambar 2.39 ***Sequence Diagram Manage Branches*** 98
Gambar 2.40 ***Sequence Diagram Manage Payroll*** 99
Gambar 2.41 ***Sequence Diagram Manage Subscription*** 100
Gambar 2.42 ***Sequence Diagram Manage Package*** 101
Gambar 2.43 ***Sequence Diagram Manage Users*** 102
Gambar 2.44 ***Sequence Diagram Manage Instansi*** 103
Gambar 2.45 ***Sequence Diagram Manage Payment Methods*** 104
Gambar 2.46 ***Sequence Diagram Manage Announcements*** 105
Gambar 2.47 ***Class Diagram HagaPlus*** 106
Gambar 3.1 Kompres seluruh *project* tanpa node_modules 118
Gambar 3.9 Verifikasi 2 Langkah akun google 120
Gambar 3.10 *Input* nama aplikasi di *app password* 120
Gambar 3.11 Sandi Aplikasi 120
Gambar 3.12 Menentukan *project* yang akan digunakan 121
Gambar 3.13 Membuat *project* baru 121
Gambar 3.14 *Project* berhasil dibuat 121
Gambar 3.15 Tambahkan *URL callback* yang digunakan 122
Gambar 3.16 *OAuth client details* 122
Gambar 3.17 Konfigurasi google *secret* pada *file* .env 122
Gambar 3.18 *Import database* ke phpmyadmin hosting panel 124
Gambar 3.19 Rincian perhitungan waktu dan uang lembur pada penggajian Eko Pratama periode November 2025. 127

Gambar 3.20 Rincian komponen potongan gaji Eko Pratama 127
Gambar 3.21 Tarif Iuran BPJS yang Digunakan dalam Perhitungan Penggajian 128
Gambar 3.22 Perhitungan PPh Pasal 21 Masa Pajak Terakhir Gaji Eko Pratama periode Desember 2025 129
Gambar 3.23 Perhitungan PPh21 Gaji Eko Pratama periode November 2025 130
Gambar 3.24 Perhitungan PPh21 Gaji Eko Pratama periode Desember 2025 131
Gambar 3.25 Rincian penghasilan dan potongan Lukman Muludin periode November 2025. 132
Gambar 3.26 Rincian Perhitungan PPh21 Gaji Lukman Muludin periode Desember 2026 133
Gambar 3.27 Rincian perhitungan penggajian dan koreksi PPh Pasal 21 masa pajak terakhir pada Lukman Muludin. 133
Gambar 3.28 Formulir 1721-A1 yang dihasilkan oleh sistem HagaPlus. 134
Gambar 3.29 *Landing Page* 135
Gambar 3.30 Halaman *Register* 136
Gambar 3.31 Halaman *Login* 136
Gambar 3.32 Halaman *Dashboard Superadmin* 137
Gambar 3.33 Halaman Instansi *Management* 137
Gambar 3.34 Halaman *Subscription Management* 138
Gambar 3.35 Halaman *Package Management* 138
Gambar 3.36 Halaman *User Management* 139
Gambar 3.37 Halaman *Payment Methods* 139
Gambar 3.38 Halaman *Landing Page Announcement* 140
Gambar 3.39 Halaman *Dashboard Admin* 140
Gambar 3.40 Halaman *Manage Attendance* (Absensi) 141
Gambar 3.41 Halaman *Employee Management* 141
Gambar 3.42 Halaman *Leave Requests Management* 142
Gambar 3.43 Halaman *Reimbursement* 142
Gambar 3.44 Halaman *Shift Swaps* 143

Gambar 3.45 Halaman *Payroll Management* 143
Gambar 3.46 Halaman *Manage Holidays* 144
Gambar 3.47 Halaman *Organization Management* 144
Gambar 3.48 Halaman *Branch Management* 145
Gambar 3.49 Halaman *Company Profile* 145
Gambar 3.50 Halaman *Dashboard* Karyawan 146
Gambar 3.51 Halaman *Attendance* (Absensi) Karyawan 147
Gambar 3.52 Halaman *Leave Requests* Karyawan 147
Gambar 3.53 Halaman *Reimbursement* Karyawan 148
Gambar 3.54 Halaman *Shift Swap* Karyawan 149
Gambar 3.55 Halaman Slip Gaji Karyawan 149
Gambar 3.56 Halaman Profil Karyawan 150

DAFTAR TABEL

Tabel 1.1 Identifikasi Kebutuhan Mitra 2
Tabel 1.2 Tim dan Pembagian Tugas 6
Tabel 1.3 *Timeline* 7
Tabel 2.1 Identifikasi Kebutuhan dan Rancangan Solusi 9
Tabel 2.2 Kebutuhan Fungsional Sistem 10
Tabel 2.3 Kebutuhan Non-Fungsional Sistem 12
Tabel 2.4 *Role* dan Hak Akses 13
Tabel 2.5 Spesifikasi Tabel *activity_logs* 21
Tabel 2.6 Spesifikasi Tabel *announcements* 22
Tabel 2.7 Spesifikasi Tabel *attendance_policies* 24
Tabel 2.8 Spesifikasi Tabel *attendance_revisions* 26
Tabel 2.9 Spesifikasi Tabel *attendances* 28
Tabel 2.10 Spesifikasi Tabel *branches* 29
Tabel 2.11 Spesifikasi Tabel *cache* 30
Tabel 2.12 Spesifikasi Tabel *cache_locks* 30
Tabel 2.13 Spesifikasi Tabel *departments* 31
Tabel 2.14 Spesifikasi Tabel *discount_usage* 31
Tabel 2.15 Spesifikasi Tabel *discounts* 32
Tabel 2.16 Spesifikasi Tabel *division_policies* 33
Tabel 2.17 Spesifikasi Tabel *divisions* 35
Tabel 2.18 Spesifikasi Tabel *email_templates* 35
Tabel 2.19 Spesifikasi Tabel *employee_policies* 36
Tabel 2.20 Spesifikasi Tabel *employee_payroll_components* 36
Tabel 2.21 Spesifikasi Tabel *payroll_components* 37
Tabel 2.22 Spesifikasi Tabel *employee_schedules* 38
Tabel 2.23 Spesifikasi Tabel *employee_tax_forms* 39
Tabel 2.24 Spesifikasi Tabel *employees* 40
Tabel 2.25 Spesifikasi Tabel *failed_jobs* 43
Tabel 2.26 Spesifikasi Tabel *holidays* 44

Tabel 2.27 Spesifikasi Tabel *instansi_role_permissions* 44
Tabel 2.28 Spesifikasi Tabel *instansi_roles* 45
Tabel 2.29 Spesifikasi Tabel *instansis* 45
Tabel 2.30 Spesifikasi Tabel *job_batches* 46
Tabel 2.31 Spesifikasi Tabel *jobs* 47
Tabel 2.32 Spesifikasi Tabel *leaves* 48
Tabel 2.33 Spesifikasi Tabel *login_logs* 50
Tabel 2.34 Spesifikasi Tabel *packages* 50
Tabel 2.35 Spesifikasi Tabel *password_reset_tokens* 51
Tabel 2.36 Spesifikasi Tabel payment_methods 51
Tabel 2.37 Spesifikasi Tabel *payrolls* 52
Tabel 2.38 Spesifikasi Tabel *permissions* 54
Tabel 2.39 Spesifikasi Tabel *positions* 54
Tabel 2.40 Spesifikasi Tabel *reimbursements* 55
Tabel 2.41 Spesifikasi Tabel sessions 57
Tabel 2.42 Spesifikasi Tabel *shift_swap_requests* 57
Tabel 2.43 Spesifikasi Tabel *subscription_requests* 58
Tabel 2.44 Spesifikasi Tabel *subscriptions* 59
Tabel 2.45 Spesifikasi Tabel *system_roles* 60
Tabel 2.46 Spesifikasi Tabel *system_settings* 61
Tabel 2.47 Spesifikasi Tabel *tax_settings* 61
Tabel 2.48 Spesifikasi Tabel *users* 62
Tabel 2.49 Perangkat Keras yang dibutuhkan 107
Tabel 2.50 Perangkat Lunak yang dibutuhkan 107
Tabel 3.1 Rancangan Waktu Pengembangan Sistem 108
Tabel 3.2 Persyaratan *Hardware* 111
Tabel 3.3 Persyaratan *Software* 113
Tabel 3.4 Komponen Pendukung Sistem 114
Tabel 3.5 Spesifikasi VPS yang digunakan untuk *deployment* 117

BAB I
PENDAHULUAN

1.1 Latar Belakang

Transformasi digital telah mendorong perubahan dalam berbagai bidang, termasuk dalam pengelolaan sumber daya manusia pada perusahaan. Perkembangan teknologi informasi memungkinkan proses pencatatan kehadiran dan penggajian karyawan dilakukan melalui sistem informasi berbasis digital. Penggunaan sistem informasi dalam proses tersebut dapat membantu perusahaan mengelola data kehadiran dan penggajian secara lebih terstruktur serta mendukung proses administrasi yang lebih efektif. *Int*egrasi antara data kehadiran dan penggajian juga diperlukan karena informasi kehadiran dapat menjadi salah satu dasar dalam pengelolaan gaji karyawan.

Dalam proses penggajian, perhitungan gaji tidak hanya berkaitan dengan gaji pokok, tetapi dapat mencakup berbagai komponen penghasilan seperti tunjangan, insentif, lembur, dan komponen lainnya sesuai dengan kebijakan masing-masing perusahaan. Selain komponen penghasilan, terdapat pula potongan yang perlu diperhitungkan, seperti iuran jaminan sosial dan Pajak Penghasilan (PPh) Pasal 21. Perbedaan struktur penghasilan dan kebijakan penggajian pada setiap perusahaan menyebabkan sistem *payroll* perlu memiliki komponen gaji yang fleksibel dan dapat disesuaikan dengan kebutuhan perusahaan.

Pengelolaan jaminan sosial merupakan salah satu aspek yang perlu diperhatikan dalam proses penggajian. Berdasarkan Undang-Undang Nomor 24 Tahun 2011 tentang Badan Penyelenggara Jaminan Sosial, Pasal 15 ayat (1) mengatur bahwa pemberi kerja secara bertahap wajib mendaftarkan dirinya dan pekerjanya sebagai peserta kepada BPJS sesuai dengan program jaminan sosial yang diikuti. Pemberi kerja juga wajib memberikan data dirinya dan pekerjanya secara lengkap dan benar kepada BPJS sebagaimana diatur dalam Pasal 15 ayat (2). Ketentuan tersebut menunjukkan bahwa data pekerja dan informasi kepesertaan jaminan sosial perlu diperhatikan dalam pengelolaan penggajian.

Selain jaminan sosial, pemberi kerja juga perlu memperhatikan kewajiban pemotongan Pajak Penghasilan (PPh) Pasal 21 atas penghasilan yang diterima

karyawan. Ketentuan pemotongan PPh Pasal 21 antara lain diatur dalam Peraturan Pemer*int*ah Nomor 58 Tahun 2023 dan Peraturan Menteri Keuangan Nomor 168 Tahun 2023 yang berlaku sejak 1 Januari 2024. Ketentuan tersebut mengatur penggunaan tarif efektif dalam penghitungan pemotongan PPh Pasal 21. Untuk pegawai tetap, tarif efektif bulanan digunakan pada masa pajak selain masa pajak terakhir, sedangkan pada masa pajak terakhir digunakan tarif berdasarkan Pasal 17 ayat (1) huruf a Undang-Undang Pajak Penghasilan. Ketentuan tersebut menyebabkan proses penggajian perlu memperhatikan data penghasilan, status pegawai, komponen gaji, serta ketentuan perpajakan yang berlaku.

Kebutuhan pengelolaan absensi dan penggajian tersebut dapat diterapkan pada berbagai bentuk dan skala badan usaha, termasuk Usaha Mikro, Kecil, dan Menengah (UMKM). Salah satu alternatif penyediaan sistem adalah melalui konsep *Software as a Service* (SaaS) dengan mekanisme *subscription* atau berlangganan. Melalui konsep tersebut, perusahaan dapat menggunakan sistem tanpa harus membangun dan menyediakan infrastruktur sistem secara mandiri. Model berlangganan juga memungkinkan layanan sistem disediakan dalam bentuk paket sesuai dengan kebutuhan pengguna. Oleh karena itu, HagaPlus dirancang agar dapat menjadi alternatif bagi UMKM dan badan usaha lainnya yang membutuhkan sistem absensi dan penggajian berbasis web.

Perancangan HagaPlus didasarkan pada kebutuhan yang disampaikan oleh pihak mitra terhadap pengelolaan proses kehadiran dan penggajian secara ter*int*egrasi. Kebutuhan tersebut mencakup pencatatan kehadiran karyawan, pengelolaan cuti dan izin, *reimbursement*, pengelolaan komponen penghasilan secara dinamis, proses penggajian, perhitungan BPJS dan PPh Pasal 21, penyediaan formulir SPT 1721-A1, serta pengelolaan data perusahaan dan karyawan. Identifikasi kebutuhan tersebut menjadi dasar dalam menentukan fitur dan fungsi yang dirancang pada HagaPlus.

Tabel 1.1 Identifikasi Kebutuhan Mitra

No	Kebutuhan Mitra	Fitur Sistem
1.	Pengelolaan kehadiran karyawan	Absensi berbasis GPS dan foto
2.	Pengelolaan cuti dan izin	Modul cuti dan izin

3.	Pengelolaan pengeluaran karyawan	Reimbursement
4.	Pengelolaan komponen penghasilan yang dapat disesuaikan	Komponen gaji dinamis
5.	Pengelolaan proses penggajian	Modul payroll
6.	Perhitungan potongan jaminan sosial	Perhitungan BPJS
7.	Perhitungan pajak penghasilan karyawan	Perhitungan PPh Pasal 21
8.	Penyediaan dokumen perpajakan karyawan	Formulir SPT 1721-A1
9.	Pengelolaan data perusahaan dan karyawan	Manajemen organisasi

Berdasarkan kebutuhan tersebut, dirancang HagaPlus sebagai sistem informasi berbasis web dengan konsep *Software as a Service* (SaaS) atau berlangganan yang berfokus pada *int*egrasi proses pencatatan kehadiran hingga penggajian karyawan. HagaPlus mencakup fitur absensi berbasis GPS dan foto sebagai bagian dari pencatatan kehadiran, serta fitur pendukung berupa cuti dan izin, *reimbursement*, pengelolaan komponen gaji secara dinamis, penggajian, perhitungan BPJS dan PPh Pasal 21, formulir SPT 1721-A1, serta manajemen organisasi. Perancangan HagaPlus diharapkan dapat memberikan alternatif sistem berbasis web yang membantu proses absensi dan penggajian dilakukan secara lebih terstruktur serta dapat digunakan oleh UMKM dan badan usaha lainnya melalui mekanisme berlangganan.

1.2 Maksud dan Tujuan

Maksud dari perancangan Sistem HagaPlus adalah merancang sistem informasi berbasis web yang dapat membantu proses pengelolaan absensi dan penggajian karyawan secara ter*int*egrasi pada Usaha Mikro, Kecil, dan Menengah (UMKM). Sistem ini dirancang sebagai solusi yang praktis, mudah digunakan, serta dapat diakses secara *online* guna meningkatkan efektivitas dan efisiensi pengelolaan sumber daya manusia. Selain itu, HagaPlus dikembangkan dengan konsep *Software as a Service* (*SaaS*) berbasis *subscription* sehingga pengguna tidak

perlu membangun sistem secara mandiri maupun menyediakan infrastruktur teknologi yang kompleks.

Adapun tujuan dari perancangan Sistem HagaPlus adalah sebagai berikut:

a. Merancang dan membangun sistem informasi absensi, penggajian, dan pengelolaan pajak karyawan berbasis web yang ter*int*egrasi dalam satu *platform*.
b. Menghasilkan sistem penggajian yang mampu melakukan perhitungan *payroll* secara otomatis berdasarkan data absensi, data karyawan, serta perhitungan BPJS dan PPh21.
c. Menyediakan sistem yang fleksibel, mudah digunakan, dan sesuai dengan kebutuhan operasional UMKM.
d. Mendukung transformasi digital UMKM melalui penerapan sistem informasi yang praktis, efektif, dan terjangkau.
e. Menghasilkan sistem informasi berbasis web dengan model layanan *SaaS* yang dapat diimplementasikan sebagai solusi digital dalam pengelolaan sumber daya manusia pada UMKM.
f. Menghasilkan sistem yang mampu menghitung potongan BPJS Kesehatan, BPJS Ketenagakerjaan, serta PPh21 secara otomatis berdasarkan data karyawan dan pengaturan pajak yang berlaku.

1.3 Sasaran

Sasaran dari sistem HagaPlus adalah menghasilkan produk perangkat lunak berupa aplikasi web dengan spesifikasi sebagai berikut:

a. Modul Absensi Digital: Sistem kehadiran berbasis web yang mencatat waktu kerja secara akurat serta dilengkapi fitur pengajuan revisi kehadiran.
b. Modul Manajemen Alur Kerja (*Workflow*): Sistem pengajuan, pembatalan, dan persetujuan cuti, klaim reimbursement, serta pertukaran *shift* kerja secara mandiri (*self-service*).
c. Mesin Penggajian Otomatis (*Payroll Engine*): Sistem kalkulasi gaji bulanan yang secara otomatis meng*int*egrasikan komponen gaji pokok, tunjangan, potongan kehadiran, iuran BPJS, dan potongan PPh21.

d. Portal Mandiri Karyawan (*Employee Self-Service*): Halaman khusus bagi karyawan untuk melihat profil pribadi, riwayat absensi, serta mengunduh slip gaji secara mandiri.
e. Modul *Admini*strasi Pajak: Fitur perhitungan PPh21 sesuai aturan perpajakan yang berlaku dan *generator* formulir SPT 1721-A1 digital untuk pelaporan pajak tahunan.
f. Model Layanan *SaaS Multi-Tenant*: Infrastruktur aplikasi berbasis *subscription* yang memungkinkan pendaftaran bisnis secara instan, pemilihan paket langganan, dan manajemen transaksi pembayaran.

1.4 Mitra

PT. Mora Cipta Solusi merupakan perusahaan yang bergerak di bidang pengembangan solusi teknologi ter*int*egrasi yang mencakup *audio visual*, keamanan, *Information and Communication Technology* (ICT), serta berbagai perangkat elektronik pendukung. Perusahaan ini berkomitmen untuk menghadirkan solusi teknologi yang inovatif dan sesuai dengan kebutuhan pengguna dengan memanfaatkan teknologi terkini.

Dalam *Project Work* ini, PT. Mora Cipta Solusi berperan sebagai Mitra dalam proses pengembangan sistem HagaPlus. Mitra memberikan dukungan berupa informasi kebutuhan sistem, proses bisnis, serta masukan terkait fitur-fitur yang dikembangkan agar sistem dapat berjalan sesuai dengan kebutuhan pengguna. Melalui kerja sama tersebut, diharapkan sistem HagaPlus dapat menjadi solusi yang membantu proses pengelolaan absensi dan penggajian secara lebih efektif dan ter*int*egrasi.

Gambar 1.1 Logo PT. Mora Cipta Solusi

1.5 Tim dan Pembagian Tugas

Dalam pengerjaan proyek ini, tim terdiri dari dua orang. Pembagian tugas dilakukan agar proses pengerjaan lebih terarah dan setiap anggota dapat fokus pada tanggung jawabnya masing-masing.

Adapun pembagian tugas dalam proyek ini adalah sebagai berikut:

Tabel 1.2 Tim dan Pembagian Tugas

Nama	Peran	Tanggung Jawab
Siti Khodijah	*UI/UX Designer* dan *System Analyst*	1. Menyusun laporan proyek secara keseluruhan. 2. Melakukan perancangan sistem dan pembuatan UML (*Use Case* Diagram, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, dan diagram lainnya). 3. Mendesain antarmuka pengguna (UI/UX) sesuai kebutuhan pengguna. 4. Melakukan pengujian (*testing*) sistem. 5. Menyusun dokumentasi dan *guidebook* penggunaan sistem.
Lukman Muludin	*Full Stack Developer*	1. Mengembangkan sistem secara teknis. 2. Mengimplementasikan *frontend* berdasarkan desain UI/UX. 3. Mengembangkan *backend* dan logika sistem. 4. Meng*int*egrasikan seluruh fitur sistem. 5. Menangani *bug* dan *error* selama proses pengembangan.

1.6 *Timeline*

Penelitian ini direncanakan akan berlangsung selama periode sekitar 6 (enam) bulan terhitung mulai dari penandatanganan kontrak kesepakatan penelitian. Tempat pelaksanaan penelitian ditetapkan di Universitas Pamulang yang berlokasi di Jalan Raya Puspiptek, Kelurahan Buaran, Kecamatan Pamulang, Kota Tangerang Selatan, Provinsi Banten 15310.

Tabel 1.3 *Timeline*

No	Kegiatan	Waktu (Bulan)						PIC
		1	2	3	4	5	6	
1	Pengumpulan data dan analisis kebutuhan sistem							Tim
2	Penyusunan instrumen dan dokumentasi *project*							Siti Khodijah
3	Perancangan desain sistem meliputi *UI/UX* dan *database*							Siti Khodijah
4	Analisis dan perancangan sistem							Tim
5	Pengembangan sistem HagaPlus							Lukman Muludin
6	*Testing* dan implementasi sistem							Tim
7	Penyusunan laporan akhir *Project Work*							Siti Khodijah
8	Publikasi hasil project dan pengajuan HKI							Tim

BAB II
ANALISIS DAN PERANCANGAN

2.1 Analisis Sistem

Analisis sistem merupakan tahapan awal dalam proses perancangan sistem yang bertujuan untuk mengidentifikasi kebutuhan pengguna dan proses yang berkaitan dengan pencatatan kehadiran hingga penggajian karyawan. Analisis dilakukan untuk menentukan kebutuhan sistem yang menjadi dasar dalam perancangan HagaPlus sebagai sistem informasi berbasis web. HagaPlus dirancang untuk meng*int*egrasikan proses pencatatan kehadiran dengan proses penggajian serta menyediakan fitur pendukung seperti pengelolaan cuti dan izin, *reimbursement*, dan data karyawan dalam satu *platform*.

Berdasarkan kebutuhan yang disampaikan oleh pihak mitra, diperlukan sistem yang dapat mendukung proses pengelolaan kehadiran dan penggajian secara ter*int*egrasi. Kebutuhan tersebut meliputi pengelolaan data karyawan, pencatatan absensi, pengelolaan cuti dan izin, *reimbursement*, pengelolaan komponen penghasilan, serta proses penggajian. Proses penggajian juga perlu memperhatikan komponen potongan seperti BPJS dan PPh Pasal 21 serta penyediaan formulir SPT 1721-A1. Selain itu, komponen penghasilan perlu dapat disesuaikan dengan kebijakan masing-masing perusahaan karena struktur penghasilan karyawan dapat berbeda.

Berdasarkan kebutuhan tersebut, HagaPlus dirancang sebagai sistem informasi berbasis web yang meng*int*egrasikan proses pencatatan kehadiran hingga penggajian karyawan. Sistem dirancang dengan pembagian hak akses berdasarkan peran pengguna, yaitu *User*, Superadmin, Admin, dan *Employee*. Setiap aktor memiliki fungsi dan hak akses yang berbeda sesuai dengan kebutuhan masing-masing. Pembagian hak akses tersebut digunakan sebagai dasar dalam menentukan fungsi yang dapat dilakukan oleh setiap pengguna pada sistem.

Hasil analisis kebutuhan tersebut selanjutnya menjadi dasar dalam perancangan fitur, alur proses, struktur basis data, dan antarmuka HagaPlus. Analisis juga menjadi dasar dalam menentukan keterkaitan antara data kehadiran dengan proses penggajian, termasuk pengelolaan komponen gaji dinamis,

perhitungan BPJS dan PPh Pasal 21, serta penyediaan dokumen SPT 1721-A1. Dengan demikian, hasil analisis digunakan sebagai acuan dalam menyusun rancangan sistem HagaPlus sesuai dengan kebutuhan yang telah diidentifikasi.

2.2 Analisis Masalah

Berdasarkan kebutuhan yang disampaikan oleh pihak mitra, terdapat beberapa kebutuhan yang menjadi dasar dalam perancangan sistem HagaPlus. Kebutuhan tersebut berkaitan dengan proses pencatatan kehadiran hingga penggajian karyawan serta beberapa fitur pendukung pengelolaan kepegawaian. Analisis dilakukan dengan mengidentifikasi kebutuhan yang perlu didukung oleh sistem dan menentukan rancangan solusi yang sesuai. Adapun kebutuhan dan rancangan solusi yang menjadi dasar perancangan sistem adalah sebagai berikut:

Tabel 2.1 Identifikasi Kebutuhan dan Rancangan Solusi

No	Kebutuhan Mitra	Rancangan Solusi
1.	Pengelolaan kehadiran perlu dilakukan secara terstruktur	Absensi GPS dan foto
2.	Pengelolaan cuti dan izin membutuhkan proses pengajuan dan persetujuan	Modul cuti dan izin
3.	Pengelolaan *reimbursement* membutuhkan proses pengajuan dan *approval*	Modul *reimbursement*
4.	Komponen penghasilan dapat berbeda sesuai kebijakan perusahaan	Komponen gaji dinamis
5.	Proses penggajian perlu mengakomodasi berbagai komponen penghasilan dan potongan	Modul *payroll*
6.	Perhitungan iuran jaminan sosial perlu ter*int*egrasi dengan penggajian	Perhitungan BPJS
7.	Perhitungan PPh Pasal 21 perlu mengikuti ketentuan perpajakan	Perhitungan PPh 21
8.	Karyawan membutuhkan dokumen perpajakan tahunan	SPT 1721-A1
9.	Data perusahaan dan karyawan perlu dikelola secara ter*int*egrasi	Manajemen organisasi

2.2.1 Analisis Kebutuhan Sistem

Berdasarkan kebutuhan mitra, HagaPlus dirancang sebagai sistem informasi berbasis web dengan konsep *Software as a Service* (SaaS) yang mendukung proses pencatatan absensi hingga penggajian karyawan secara ter*int*egrasi. Sistem mencakup pencatatan absensi, pengelolaan cuti dan izin, *reimbursement*, pengelolaan komponen penghasilan secara dinamis, hingga proses penggajian. Selain itu, sistem dirancang untuk mendukung perhitungan BPJS dan PPh Pasal 21 serta penyediaan formulir SPT 1721-A1 sebagai bagian dari proses penggajian. Konsep SaaS memungkinkan pengguna memanfaatkan layanan sistem tanpa perlu melakukan instalasi secara mandiri.

Kebutuhan sistem dibagi menjadi kebutuhan fungsional dan kebutuhan nonfungsional.

a. Kebutuhan Fungsional

Tabel 2.2 Kebutuhan Fungsional Sistem

No	Kebutuhan Fungsional	Keterangan
1.	*Login* Sistem	Sistem dapat melakukan *login* sesuai hak akses pengguna
2.	Manajemen Absensi	Sistem dapat mencatat absensi karyawan secara digital berbasis GPS dan foto
3.	Pengelolaan Cuti	Sistem dapat mengelola pengajuan cuti karyawan
4.	Pengelolaan Izin	Sistem dapat mengelola pengajuan izin karyawan
5.	*Reimbursement*	Sistem dapat mengelola pengajuan *reimbursement* karyawan
6.	Komponen Gaji Dinamis	Sistem dapat mengelola komponen penghasilan yang dapat disesuaikan dengan kebijakan perusahaan
7.	Penggajian	Sistem dapat menghitung penggajian berdasarkan data karyawan, data

		absensi, komponen penghasilan, dan potongan
8.	Slip Gaji	Sistem dapat menampilkan dan mencetak slip gaji karyawan
9.	*Dashboard*	Sistem dapat menampilkan informasi dan ringkasan data absensi serta penggajian sesuai dengan hak akses pengguna
10.	Manajemen Data Karyawan	Sistem dapat mengelola data karyawan yang meliputi data pribadi, data pekerjaan, dan informasi terkait kepegawaian
11.	*Approval* Data	Sistem dapat melakukan *approval* terhadap pengajuan cuti, izin, dan *reimbursement*
12.	Pertukaran Jadwal Kerja	Sistem dapat mengelola perm*int*aan pertukaran jadwal kerja antar karyawan
13.	Manajemen Hari Libur	Sistem dapat mengelola data hari libur nasional dan cuti bersama
14.	Manajemen Organisasi	Sistem dapat mengelola data divisi, departemen, posisi, dan cabang
15.	Formulir Pajak (SPT 1721-A1)	Sistem dapat menyediakan formulir SPT 1721-A1 sebagai dokumen perpajakan karyawan
16.	Perhitungan BPJS dan PPh Pasal 21	Sistem dapat menghitung komponen BPJS dan PPh Pasal 21 berdasarkan data penghasilan dan ketentuan yang diterapkan
17.	Manajemen Paket dan *Subscription*	Superadmin dapat mengelola paket berlangganan dan transaksi pembayaran pengguna

18.	*Manajemen Kode Promo*	Superadmin dapat membuat dan mengelola kode promo untuk pelanggan
19.	*Onboarding* Admin	Sistem menyediakan panduan langkah demi langkah bagi Admin dalam menggunakan sistem
20.	Ekspor Data	Sistem dapat mengekspor data absensi dan penggajian ke dalam format Excel

b. Kebutuhan Non-Fungsional

Tabel 2.3 Kebutuhan Non-Fungsional Sistem

No	Aspek	Keterangan
1.	Keamanan	Sistem memiliki autentikasi *login* sesuai hak akses pengguna
2.	*Usability*	Sistem memiliki tampilan yang mudah dipahami dan digunakan (*user friendly*)
3.	*Portability*	Sistem dapat diakses melalui *browser* pada berbagai perangkat
4.	*Performance*	Sistem dapat memproses data absensi dan penggajian dengan waktu respons yang sesuai
5.	*Availability*	Sistem dapat diakses secara *online* selama layanan tersedia
6.	*Database*	Sistem menggunakan *database* sebagai media penyimpanan dan pengelolaan data

2.2.2 Analisis *Role* Sistem

Dalam sistem HagaPlus terdapat beberapa aktor yang memiliki hak akses dan fungsi yang berbeda sesuai dengan tahapan penggunaan sistem. Pembagian hak akses dilakukan untuk membatasi fungsi yang dapat diakses oleh setiap aktor sesuai

dengan perannya, mulai dari calon pengguna layanan hingga pengguna yang telah menggunakan sistem.

Tabel 2.4 *Role* dan Hak Akses

No	Pengguna	Hak Akses
1.	Superadmin	Mengelola keseluruhan sistem, data instansi, Mengelola data instansi, pengguna, paket *subscription*, metode pembayaran, kode promo, dan pengumuman
2.	Admin	Mengelola data karyawan, absensi, cuti, izin, *reimbursement*, jadwal kerja, profil perusahaan, cabang, serta proses penggajian
3.	*Employee*	Melakukan absensi, melihat riwayat absensi, mengajukan cuti dan *reimbursement*, melakukan pertukaran jadwal kerja, melihat slip gaji, dan memperbarui profil

Berikut penjelasan dari masing-masing pengguna pada sistem HagaPlus:

a. Superadmin

Superadmin merupakan pihak yang mengelola layanan HagaPlus sebagai penyedia layanan berbasis *Software as a Service* (SaaS). Superadmin memiliki hak akses untuk mengelola data instansi, data pengguna, paket *subscription*, metode pembayaran, kode promo, serta informasi dan aktivitas yang berkaitan dengan layanan HagaPlus.

b. Admin

Admin merupakan pengguna dari perusahaan atau instansi yang menggunakan layanan HagaPlus untuk mengelola proses absensi dan penggajian serta fitur pendukung kepegawaian. Admin memiliki hak akses untuk mengelola data karyawan, data absensi, pengajuan cuti dan izin, *reimbursement*, pertukaran jadwal kerja, manajemen organisasi yang meliputi divisi, departemen, posisi, dan cabang, manajemen hari libur, komponen gaji, serta proses penggajian termasuk perhitungan BPJS dan PPh Pasal 21.

c. *Employee*

Employee merupakan karyawan yang menggunakan HagaPlus untuk melakukan pencatatan kehadiran dan mengakses informasi yang berkaitan dengan kepegawaiannya. *Employee* dapat melakukan absensi, melihat riwayat absensi, mengajukan cuti dan izin, mengajukan *reimbursement*, melakukan pertukaran jadwal kerja, melihat slip gaji dan formulir SPT 1721-A1, serta memperbarui data profil.

2.3 Perancangan Sistem

Perancangan sistem merupakan tahapan yang dilakukan setelah proses analisis kebutuhan selesai dilaksanakan. Tahap ini bertujuan untuk menerjemahkan hasil analisis kebutuhan ke dalam bentuk rancangan sistem yang menjadi dasar dalam perancangan HagaPlus. Perancangan dilakukan dengan memperhatikan kebutuhan pengguna, proses pengelolaan kepegawaian dan penggajian, serta hubungan antar komponen sistem agar setiap fungsi dapat saling ter*int*egrasi.

Pada penelitian ini, proses perancangan menggunakan pendekatan *Unified Modeling Language* (UML) sebagai alat bantu untuk memvisualisasikan struktur, perilaku, dan *int*eraksi sistem. Pemodelan UML meliputi *Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram* yang digunakan untuk menggambarkan *int*eraksi pengguna dengan sistem, alur proses, komunikasi antar objek, serta struktur sistem.

Selain perancangan UML, perancangan basis data dilakukan menggunakan *Entity Relationship Diagram* (ERD), *Logical Record Structure* (LRS), dan spesifikasi basis data**.** ERD digunakan untuk menggambarkan hubungan antarentitas, LRS digunakan untuk menggambarkan struktur tabel dan relasi data secara logis, sedangkan spesifikasi basis data digunakan untuk menjelaskan struktur setiap tabel, atribut, tipe data, dan panjang data yang digunakan dalam sistem.

Perancangan antarmuka juga dilakukan untuk menggambarkan tampilan dan *int*eraksi pengguna dengan sistem HagaPlus. Perancangan tersebut mencakup antarmuka Superadmin, Admin dan Karyawan sesuai dengan hak akses masing-masing. Seluruh hasil perancangan digunakan sebagai acuan untuk memastikan fungsi, alur proses, struktur data, dan tampilan HagaPlus sesuai dengan kebutuhan yang telah diidentifikasi.

2.4 Perancangan *Database*

Perancangan *database* pada sistem HagaPlus dilakukan untuk menentukan struktur data yang digunakan dalam pengelolaan kepegawaian dan penggajian. *Database* dirancang agar data perusahaan, karyawan, absensi, jadwal kerja, cuti, izin, *reimbursement*, penggajian, komponen gaji, BPJS, PPh Pasal 21, serta data pendukung lainnya dapat disimpan dan dikelola secara terstruktur dan saling ter*int*egrasi.

Perancangan *database* HagaPlus terdiri atas *Entity Relationship Diagram* (ERD), *Logical Record Structure* (LRS), dan spesifikasi *database*. ERD digunakan untuk menggambarkan entitas dan hubungan antarentitas yang terdapat dalam sistem. LRS digunakan untuk menggambarkan struktur tabel beserta hubungan antar tabel berdasarkan hasil perancangan ERD. Sementara itu, spesifikasi *database* digunakan untuk menjelaskan struktur setiap tabel secara lebih rinci, meliputi nama *field*, tipe data, panjang data (*length*), *primary key*, *foreign key*, dan keterangan.

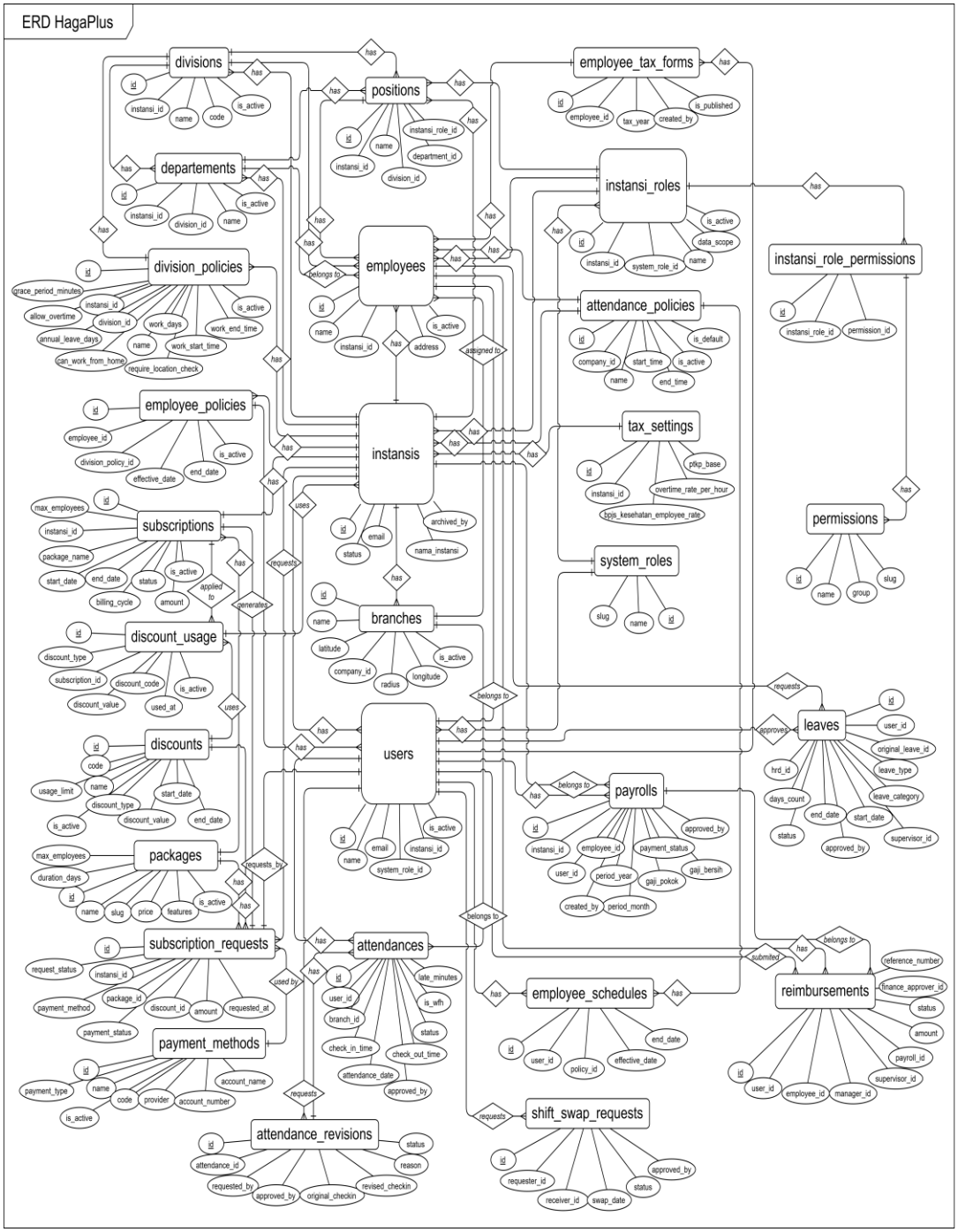
Struktur *database* HagaPlus disusun untuk mendukung proses utama sistem, mulai dari pengelolaan perusahaan dan karyawan, pencatatan absensi, pengelolaan jadwal kerja, cuti dan izin, *reimbursement*, hingga proses penggajian. *Database* juga mendukung pengelolaan komponen gaji yang bersifat dinamis sehingga komponen penghasilan dapat disesuaikan dengan kebijakan masing-masing perusahaan. Selain itu, struktur *database* mencakup data yang diperlukan dalam perhitungan BPJS dan PPh Pasal 21 serta penyediaan formulir SPT 1721-A1.

Pada sisi layanan *subscription*, *database* juga digunakan untuk menyimpan informasi paket, perusahaan yang berlangganan, riwayat *subscription*, pembayaran, dan kode promo. Dengan struktur tersebut, setiap data dapat dikelola berdasarkan hubungan antar tabel dan hak akses pengguna sehingga mendukung proses pengelolaan HagaPlus secara ter*int*egrasi.

2.4.1 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) merupakan diagram yang digunakan untuk menggambarkan entitas, atribut, serta hubungan antarentitas yang terdapat pada *database* HagaPlus. ERD dirancang berdasarkan kebutuhan sistem yang meliputi pengelolaan perusahaan, karyawan, absensi, jadwal kerja, cuti dan izin, *reimbursement*, penggajian, komponen gaji, BPJS, PPh Pasal 21, serta pengelolaan

layanan *subscription*. Setiap entitas memiliki hubungan dengan entitas lainnya melalui *primary key* dan *foreign key* sehingga data dapat saling terhubung dan dikelola secara terstruktur.



Gambar 2.1 ***Entity Relationship Diagram***

ERD HagaPlus digunakan sebagai dasar dalam menentukan struktur *database* sebelum ditransformasikan ke dalam bentuk LRS dan spesifikasi

database. Hubungan antarentitas dirancang untuk mendukung proses pengelolaan mulai dari data perusahaan dan karyawan, pencatatan kehadiran, pengajuan cuti dan izin, *reimbursement*, hingga proses penggajian dan perhitungan potongan.

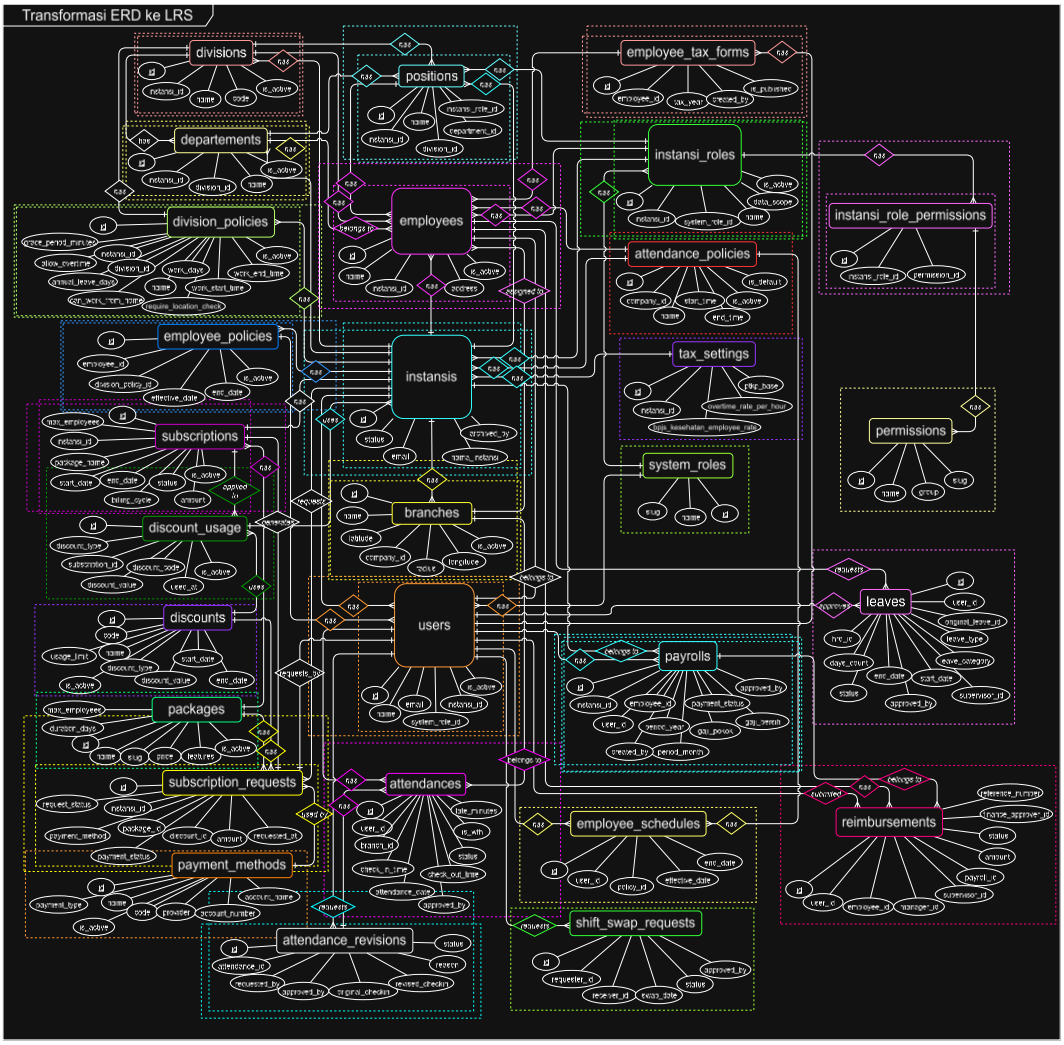
Perancangan ERD juga memperhatikan keterkaitan data yang digunakan dalam proses absensi hingga penggajian. Data karyawan memiliki keterkaitan dengan data kehadiran, jadwal kerja, komponen penghasilan, dan data penggajian. Data tersebut selanjutnya digunakan untuk mendukung proses perhitungan komponen gaji, termasuk potongan BPJS dan PPh Pasal 21. Dengan demikian, ERD HagaPlus menggambarkan hubungan antarentitas yang diperlukan agar proses pengelolaan data dapat berjalan secara ter*int*egrasi dan menjadi dasar dalam pembentukan struktur tabel pada *database*.

2.4.2 Transformasi *Entity Relationship Diagram* (ERD) ke *Logical Record Structure* (LRS)

Transformasi *Entity Relationship Diagram* (ERD) ke *Logical Record Structure* (LRS) dilakukan untuk mengubah rancangan hubungan antar entitas pada ERD menjadi struktur tabel yang dapat digunakan dalam perancangan *database* HagaPlus. Proses transformasi dilakukan dengan menentukan tabel berdasarkan entitas yang terdapat pada ERD, kemudian menentukan atribut, *primary key*, dan *foreign key* pada setiap tabel.

Pada sistem HagaPlus, transformasi ERD ke LRS mencakup tabel yang mendukung pengelolaan perusahaan, karyawan, absensi, jadwal kerja, cuti dan izin, *reimbursement*, penggajian, komponen gaji, BPJS, PPh Pasal 21, serta layanan *subscription*. Hubungan antar tabel ditentukan berdasarkan relasi antar entitas pada ERD sehingga data dapat saling terhubung dan mendukung proses pengelolaan kepegawaian dan penggajian.

Hasil transformasi ERD ke LRS selanjutnya digunakan sebagai dasar dalam menentukan struktur tabel pada *database* HagaPlus. Setiap tabel memiliki *primary key* sebagai identitas data, sedangkan *foreign key* digunakan untuk menghubungkan tabel yang memiliki hubungan. Struktur LRS yang dihasilkan kemudian menjadi acuan dalam penyusunan spesifikasi *database*, termasuk penentuan nama atribut, tipe data, dan panjang data (*length*).



Gambar 2.2 Transformasi ERD ke LRS

Dalam proses transformasi tersebut, setiap entitas pada ERD diubah menjadi tabel dengan atribut yang disesuaikan dengan kebutuhan penyimpanan data pada sistem HagaPlus. Relasi *one-to-many* dire*present*asikan dengan menempatkan *foreign key* pada tabel yang berada di sisi *many*, sedangkan relasi yang membutuhkan tabel penghubung disesuaikan dengan struktur hubungan yang telah ditentukan pada ERD. Atribut pada setiap entitas selanjutnya diterapkan sebagai *field* pada tabel, dengan *primary key* digunakan sebagai identitas data dan *foreign key* sebagai penghubung antar tabel. Proses transformasi ini mempertahankan hubungan antarentitas yang telah dirancang pada ERD sekaligus menyesuaikan struktur data dengan kebutuhan sistem, mulai dari pengelolaan data perusahaan dan karyawan, pencatatan absensi, cuti dan izin, *reimbursement*, komponen gaji, hingga

penggajian serta perhitungan BPJS dan PPh Pasal 21. Hasil transformasi tersebut selanjutnya menjadi dasar dalam penyusunan LRS dan spesifikasi *database*, sehingga setiap tabel memiliki keterkaitan yang jelas dan dapat digunakan secara ter*int*egrasi dalam mendukung proses pada sistem HagaPlus.

2.4.3 *Logical Record Structure* (LRS)

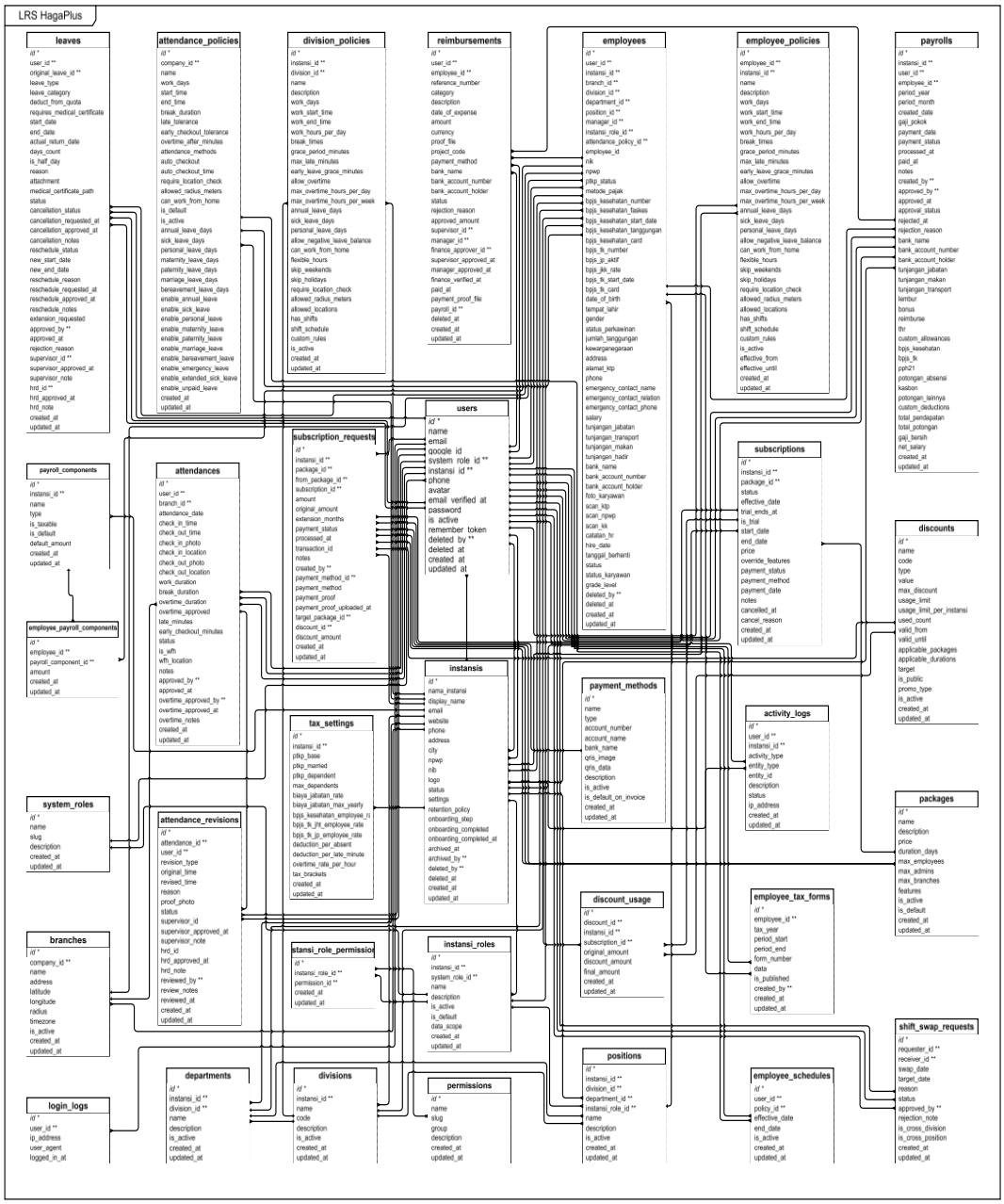
Logical Record Structure (LRS) merupakan re*present*asi dari struktur tabel dalam *database* yang menunjukkan tabel, atribut, serta hubungan antar tabel yang terdapat dalam suatu sistem. LRS pada sistem HagaPlus disusun berdasarkan hasil transformasi *Entity Relationship Diagram* (ERD) yang telah dibuat sebelumnya sehingga dapat menggambarkan struktur data secara lebih terperinci dan menjadi dasar dalam perancangan *database*.

LRS HagaPlus mencakup tabel-tabel yang digunakan untuk mendukung proses pengelolaan perusahaan, karyawan, absensi, jadwal kerja, cuti dan izin, *reimbursement*, penggajian, komponen gaji, BPJS, PPh Pasal 21, serta layanan *subscription*. Setiap tabel memiliki atribut yang disesuaikan dengan kebutuhan sistem dan memiliki *primary key* sebagai identitas data. Hubungan antar tabel ditunjukkan melalui *foreign key* yang menghubungkan data antar tabel sesuai dengan relasi yang telah ditentukan pada ERD.

Penyusunan LRS dilakukan dengan memastikan seluruh tabel yang diperlukan oleh sistem tercantum dan memiliki hubungan yang sesuai. Hal ini dilakukan agar struktur *database* dapat mendukung seluruh fungsi HagaPlus secara ter*int*egrasi, mulai dari pencatatan absensi hingga proses penggajian. LRS yang telah disusun selanjutnya menjadi dasar dalam penyusunan spesifikasi *database*, termasuk penentuan nama atribut, tipe data, panjang data (*length*), *primary key*, dan *foreign key*.

Struktur LRS juga disusun dengan memperhatikan kebutuhan *int*egrasi antarporses dalam HagaPlus, sehingga data yang digunakan pada satu proses dapat mendukung proses lainnya. Sebagai contoh, data karyawan digunakan dalam proses absensi dan penggajian, sedangkan data absensi dan komponen penghasilan dapat menjadi bagian dari informasi yang digunakan dalam proses perhitungan gaji. Dengan struktur tersebut, LRS dapat membantu memastikan keterkaitan data

antarproses tetap terorganisasi dan mendukung pengelolaan data HagaPlus secara ter*int*egrasi.



Gambar 2.3 ***Logical Record Structure***

2.5 Spesifikasi *Database*

Spesifikasi *database* merupakan bagian dari perancangan yang menjelaskan struktur setiap tabel yang digunakan dalam sistem HagaPlus. Pada tahap ini, setiap atribut dalam tabel dijabarkan secara rinci, meliputi nama *field*, tipe data, panjang data (*length*), *primary key*, *foreign key*, aturan *null*, serta keterangan dari masing-

masing atribut. Penyusunan spesifikasi *database* bertujuan untuk memberikan gambaran yang lebih rinci mengenai struktur tabel berdasarkan hasil perancangan *Logical Record Structure* (LRS), sehingga hubungan antarentitas yang telah dirancang sebelumnya dapat diterjemahkan ke dalam struktur tabel.

Spesifikasi *database* pada sistem HagaPlus disusun berdasarkan hasil perancangan dan transformasi LRS. Struktur tabel disesuaikan dengan kebutuhan sistem yang meliputi pengelolaan perusahaan dan *subscription*, data karyawan, struktur organisasi, pencatatan absensi berbasis GPS dan foto, pengelolaan cuti dan izin, *reimbursement*, komponen gaji dinamis, serta proses penggajian. Struktur *database* juga mendukung penyimpanan data yang diperlukan untuk perhitungan BPJS dan PPh Pasal 21 serta penyediaan formulir SPT 1721-A1. Dengan adanya spesifikasi *database* yang terstruktur, setiap tabel dan atribut dapat ditentukan sesuai dengan kebutuhan sistem sehingga hubungan dan pengelolaan data HagaPlus dapat dilakukan secara terorganisasi.

2.5.1 Spesifikasi Tabel *activity_logs*

Tabel *activity_logs* berfungsi untuk mencatat riwayat aktivitas pengguna sebagai *audit trail* dalam sistem HagaPlus. Tabel ini memiliki *primary key id* serta *foreign key user_id*, instansi_id, dan *entity_id* yang digunakan untuk menghubungkan data aktivitas dengan pengguna, instansi, dan entitas terkait. Informasi aktivitas disimpan melalui *field activity_type, entity_type, description, status,* dan *ip_address,* serta dilengkapi dengan *created_at* dan *updated_at* untuk mencatat waktu pembuatan dan pembaruan data.

Tabel 2.5 Spesifikasi Tabel *activity_logs*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik log aktivitas dalam sistem
2.	*user_id*	*BIGINT UNSIGNED*	-	Pengguna yang melakukan aktivitas
3.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi terkait aktivitas yang dicatat
4.	*activity_type*	*VARCHAR*	50	Jenis aktivitas yang dilakukan

5.	*entity_type*	*VARCHAR*	100	Nama entitas atau tabel yang terdampak
6.	*entity_id*	*BIGINT UNSIGNED*	-	ID spesifik dari entitas yang terdampak
7.	*description*	*TEXT*	-	Deskripsi detail tentang aktivitas
8.	*status*	*VARCHAR*	50	Hasil aktivitas, yaitu *success* atau *failed*
9.	*ip_address*	*VARCHAR*	45	Alamat IP perangkat pengguna
10.	*created_at*	*TIMESTAMP*	-	Tanggal dan waktu aktivitas dibuat
11.	*updated_at*	*TIMESTAMP*	-	Tanggal dan waktu log terakhir diperbarui

2.5.2 Spesifikasi Tabel *announcements*

Tabel *announcements* berfungsi untuk mengelola data pengumuman atau papan informasi yang ditampilkan kepada pengguna dalam sistem HagaPlus. Tabel ini memiliki *primary key* id sebagai identitas setiap data pengumuman. Informasi pengumuman disimpan melalui *field* title, slug, subtitle, description, dan content yang digunakan untuk menyimpan judul, alamat halaman, subjudul, deskripsi, serta isi pengumuman.

Tabel 2.6 Spesifikasi Tabel *announcements*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengumuman
2.	*title*	*VARCHAR*	150	Judul utama pengumuman
3.	*slug*	*VARCHAR*	150	Versi *URL-friendly* dari judul
4.	*subtitle*	*VARCHAR*	150	Subjudul atau ringkasan singkat
5.	*description*	*TEXT*	-	Penjelasan atau uraian tambahan

6.	*content*	*LONGTEXT*	-	Informasi *content*
7.	*how_to_use*	*LONGTEXT*	-	Informasi *how to use*
8.	*image*	*VARCHAR*	255	*Path* atau *URL* gambar sampul
9.	*promo_code*	*VARCHAR*	50	Informasi *promo code*
10.	*link_url*	*VARCHAR*	255	Tautan eksternal
11.	*link_text*	*VARCHAR*	100	Informasi *link text*
12.	*cta_type*	*VARCHAR*	50	Tipe tombol aksi
13.	*type*	*ENUM*	-	Informasi *type*
14.	*background_color*	*VARCHAR*	20	Informasi *background color*
15.	*text_color*	*VARCHAR*	20	Informasi *text color*
16.	*is_active*	*TINYINT*	1	Status aktif
17.	*display_order*	*INT*	-	Urutan tampilan
18.	*start_date*	*DATETIME*	-	Tanggal mulai
19.	*end_date*	*DATETIME*	-	Tanggal berakhir
20.	*deleted_at*	*TIMESTAMP*	-	Waktu data dihapus
21.	*created_at*	*TIMESTAMP*	-	Waktu pengumuman dibuat
22.	*updated_at*	*TIMESTAMP*	-	Waktu pengumuman diperbarui

2.5.3 Spesifikasi Tabel *attendance_policies*

Tabel *attendance_policies* berfungsi untuk menyimpan dan mengelola data terkait kebijakan absensi yang diterapkan pada perusahaan. Tabel ini memiliki *primary key* id dan *foreign key* company_id yang digunakan untuk menghubungkan data kebijakan absensi dengan perusahaan terkait. Informasi kebijakan absensi disimpan melalui *field* name, work_days, start_*time*, end_*time*, dan break_duration yang digunakan untuk menyimpan nama kebijakan, hari kerja, waktu mulai kerja, waktu selesai kerja, serta durasi istirahat.

Tabel 2.7 Spesifikasi Tabel *attendance_policies*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik kebijakan kehadiran
2.	*company_id*	*BIGINT UNSIGNED*	-	Referensi ke data *company*
3.	*name*	*VARCHAR*	100	Nama kebijakan kehadiran
4.	*work_days*	*JSON*	-	Informasi *work days*
5.	*start_time*	*TIME*	-	Waktu mulai kerja
6.	*end_time*	*TIME*	-	Waktu selesai kerja
7.	*break_duration*	*INT*	-	Durasi istirahat
8.	*late_tolerance*	*INT*	-	Toleransi keterlambatan
9.	*early_checkout_tolerance*	*INT*	-	Toleransi pulang lebih awal
10.	*overtime_after_minutes*	*INT*	-	Batas mulai lembur
11.	*attendance_methods*	*JSON*	-	Metode absensi
12.	*auto_checkout*	*TINYINT*	1	Status *auto checkout*
13.	*auto_checkout_time*	*TIME*	-	Waktu *auto checkout*
14.	*require_location_check*	*TINYINT*	1	Memerlukan pengecekan lokasi

15.	*allowed_radius_meters*	*DECIMAL*	10,2	Radius lokasi yang diperbolehkan
16.	*can_work_from_home*	*TINYINT*	1	Dapat bekerja dari rumah
17.	*is_default*	*TINYINT*	1	Status sebagai kebijakan default
18.	*is_active*	*TINYINT*	1	Status aktif
19.	*annual_leave_days*	*INT UNSIGNED*	-	Jumlah cuti tahunan
20.	*sick_leave_days*	*INT UNSIGNED*	-	Jumlah cuti sakit
21.	*personal_leave_days*	*INT UNSIGNED*	-	Jumlah cuti pribadi
22.	*maternity_leave_days*	*INT UNSIGNED*	-	Jumlah cuti melahirkan
23.	*paternity_leave_days*	*INT UNSIGNED*	-	Jumlah cuti ayah
24.	*marriage_leave_days*	*INT UNSIGNED*	-	Jumlah cuti menikah
25.	*bereavement_leave_days*	*INT UNSIGNED*	-	Jumlah cuti duka
26.	*enable_annual_leave*	*TINYINT*	1	Mengaktifkan cuti tahunan
27.	*enable_sick_leave*	*TINYINT*	1	Mengaktifkan cuti sakit
28.	*enable_personal_leave*	*TINYINT*	1	Mengaktifkan cuti pribadi
29.	*enable_maternity_leave*	*TINYINT*	1	Mengaktifkan cuti melahirkan
30.	*enable_paternity_leave*	*TINYINT*	1	Mengaktifkan cuti ayah

31.	*enable_marriage_leave*	*TINYINT*	1	Mengaktifkan cuti menikah
32.	*enable_bereavement_leave*	*TINYINT*	1	Mengaktifkan cuti duka
33.	*enable_emergency_leave*	*TINYINT*	1	Mengaktifkan cuti darurat
34.	*enable_extended_sick_leave*	*TINYINT*	1	Mengaktifkan cuti sakit panjang
35.	*enable_unpaid_leave*	*TINYINT*	1	Mengaktifkan cuti tanpa bayaran
36.	*created_at*	*TIMESTAMP*	-	Waktu kebijakan dibuat
37.	*updated_at*	*TIMESTAMP*	-	Waktu kebijakan diperbarui

2.5.4 Spesifikasi Tabel *attendance_revisions*

Tabel *attendance_revisions* berfungsi menyimpan dan mengelola data terkait *attendance revisions*. Tabel ini memiliki *Primary Key* id dan *Foreign Key* attendance_id, user_id, *supervisor*_id, hrd_id. Rincian utamanya disimpan pada *field* revision_*type*, original_*time*, revised_*time*, reason, serta *field* proof_photo yang berisi path bukti foto pendukung pengajuan revisi.

Tabel 2.8 Spesifikasi Tabel *attendance_revisions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengajuan revisi
2.	*attendance_id*	*BIGINT UNSIGNED*	-	Data kehadiran yang direvisi
3.	*user_id*	*BIGINT UNSIGNED*	-	Karyawan yang mengajukan revisi
4.	*revision_type*	*VARCHAR*	50	Jenis revisi *check_in/check_out*

5.	*original_time*	*DATETIME*	-	Waktu asli
6.	*revised_time*	*DATETIME*	-	Waktu revisi
7.	*reason*	*TEXT*	-	Alasan pengajuan revisi
8.	*proof_photo*	*VARCHAR*	255	Path bukti foto
9.	*status*	*ENUM*	-	*pending, approved, rejected*
10.	*supervisor_id*	*BIGINT UNSIGNED*	-	Referensi *supervisor*
11.	*supervisor_approved_at*	*TIMESTAMP*	-	Waktu persetujuan *supervisor*
12.	*supervisor_note*	*TEXT*	-	Catatan *supervisor*
13.	*hrd_id*	*BIGINT UNSIGNED*	-	Referensi HRD
14.	*hrd_approved_at*	*TIMESTAMP*	-	Waktu persetujuan HRD
15.	*hrd_note*	*TEXT*	-	Catatan HRD
16.	*reviewed_by*	*BIGINT UNSIGNED*	-	Pengguna yang melakukan *review*
17.	*review_notes*	*TEXT*	-	Catatan *review*
18.	*reviewed_at*	*TIMESTAMP*	-	Waktu *review*
19.	*created_at*	*TIMESTAMP*	-	Waktu pengajuan dibuat
20.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.5 Spesifikasi Tabel *attendances*

Tabel *attendances* berfungsi untuk merekam data absensi harian karyawan, termasuk waktu masuk, waktu pulang, dan lokasi saat melakukan absensi. Tabel ini memiliki *primary key* id serta *foreign key user_id* dan *branch_id* yang digunakan untuk menghubungkan data absensi dengan pengguna dan cabang terkait. Informasi absensi disimpan melalui *field attendance_date, check_in_time, check_out_time*, *check_in_photo*, dan *check_in_location* yang digunakan untuk menyimpan tanggal absensi, waktu masuk, waktu pulang, foto saat melakukan absensi masuk, serta koordinat GPS lokasi karyawan saat melakukan absensi masuk.

Tabel 2.9 Spesifikasi Tabel *attendances*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik data kehadiran
2.	*user_id*	*BIGINT UNSIGNED*	-	Karyawan pemilik data
3.	*branch_id*	*BIGINT UNSIGNED*	-	Cabang tempat absensi
4.	*attendance_date*	*DATE*	-	Tanggal kehadiran
5.	*check_in_time*	*TIMESTAMP*	-	Waktu absen masuk
6.	*check_out_time*	*TIMESTAMP*	-	Waktu absen pulang
7.	*check_in_photo*	*VARCHAR*	255	Foto selfie saat masuk
8.	*check_in_location*	*VARCHAR*	255	Koordinat GPS saat masuk
9.	*check_out_photo*	*VARCHAR*	255	Foto selfie saat pulang
10.	*check_out_location*	*VARCHAR*	255	Koordinat GPS saat pulang
11.	*work_duration*	*INT*	-	Durasi kerja dalam menit
12.	*break_duration*	*INT*	-	Durasi istirahat
13.	*overtime_duration*	*INT*	-	Durasi lembur
14.	*overtime_approved*	*TINYINT*	1	Status persetujuan lembur
15.	*late_minutes*	*INT*	-	Jumlah menit keterlambatan
16.	*early_checkout_minutes*	*INT*	-	Jumlah menit pulang lebih awal
17.	*status*	*ENUM*	-	*present*, *late*, *absent*, *partial*, *leave*
18.	*is_wfh*	*TINYINT*	1	Penanda *WFH*
19.	*wfh_location*	*VARCHAR*	255	Lokasi saat *WFH*

20.	*notes*	*TEXT*	-	Catatan kehadiran
21.	*approved_by*	*BIGINT UNSIGNED*	-	ID atasan
22.	*approved_at*	*TIMESTAMP*	-	Waktu persetujuan
23.	*overtime_approved_by*	*BIGINT UNSIGNED*	-	ID atasan lembur
24.	*overtime_approved_at*	*TIMESTAMP*	-	Waktu persetujuan lembur
25.	*overtime_notes*	*TEXT*	-	Catatan lembur
26.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
27.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.6 Spesifikasi Tabel *branches*

Tabel *branches* berfungsi untuk menyimpan dan mengelola data cabang perusahaan yang digunakan dalam sistem HagaPlus. Tabel ini memiliki *primary key* id serta *foreign key* company_id yang digunakan untuk menghubungkan data cabang dengan perusahaan terkait. Informasi cabang disimpan melalui *field* name, address, latitude, longitude, dan radius yang digunakan untuk menyimpan nama cabang, alamat, koordinat lokasi, serta radius dalam meter sebagai batas area yang valid untuk melakukan absensi pada cabang tersebut.

Tabel 2.10 Spesifikasi Tabel *branches*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik cabang
2.	*company_id*	*BIGINT UNSIGNED*	-	Referensi *company*
3.	*name*	*VARCHAR*	100	Nama kantor cabang
4.	*address*	*TEXT*	-	Alamat lengkap cabang
5.	*latitude*	*DECIMAL*	10,8	Koordinat latitude
6.	*longitude*	*DECIMAL*	11,8	Koordinat longitude
7.	*radius*	*INT*	-	Radius area absensi

8.	*timezone*	*VARCHAR*	50	Zona waktu
9.	*is_active*	*TINYINT*	1	Status aktif
10.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
11.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.7 Spesifikasi Tabel *cache*

Tabel *cache* berfungsi untuk menyimpan dan mengelola data *cache* yang digunakan oleh sistem HagaPlus. Tabel ini tidak memiliki *primary key* khusus. Informasi *cache* disimpan melalui *field* key, value, dan expiration yang digunakan untuk menyimpan kunci data *cache*, nilai data yang disimpan, serta waktu kedaluwarsa data *cache*.

Tabel 2.11 Spesifikasi Tabel *cache*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*key*	*VARCHAR*	255	ID unik data *cache*
2.	*value*	*MEDIUMTEXT*	-	Nilai *cache*
3.	*expiration*	*INT*	-	Waktu kedaluwarsa

2.5.8 Spesifikasi Tabel *cache_locks*

Tabel *cache_locks* berfungsi untuk menyimpan dan mengelola data *cache lock* yang digunakan dalam sistem HagaPlus. Tabel ini tidak memiliki *primary key* khusus. Informasi *cache lock* disimpan melalui *field* key, owner, dan expiration yang digunakan untuk menyimpan kunci *cache*, identitas pemilik kunci, serta waktu kedaluwarsa *cache lock*.

Tabel 2.12 Spesifikasi Tabel *cache_locks*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*key*	*VARCHAR*	255	ID unik *cache* lock
2.	*owner*	*VARCHAR*	255	Pemilik lock
3.	*expiration*	*INT*	-	Waktu kedaluwarsa

2.5.9 Spesifikasi Tabel *departments*

Tabel *departments* berfungsi untuk menyimpan dan mengelola data departemen yang terdapat dalam struktur organisasi perusahaan pada sistem

HagaPlus. Tabel ini memiliki *primary key* id serta *foreign key* instansi_id dan division_id yang digunakan untuk menghubungkan data departemen dengan instansi dan divisi terkait. Informasi departemen disimpan melalui *field* name, description, dan is_active yang digunakan untuk menyimpan nama departemen, deskripsi departemen, serta status aktif atau tidak aktifnya departemen..

Tabel 2.13 Spesifikasi Tabel *departments*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik departemen
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik departemen
3.	*division_id*	*BIGINT UNSIGNED*	-	Referensi divisi
4.	*name*	*VARCHAR*	100	Nama departemen
5.	*description*	*TEXT*	-	Penjelasan fungsi departemen
6.	*is_active*	*TINYINT*	1	Status aktif
7.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
8.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.10 Spesifikasi Tabel *discount_usage*

Tabel *discount_usage* berfungsi untuk menyimpan dan mengelola data penggunaan diskon dalam sistem HagaPlus. Tabel ini memiliki *primary key* id serta *foreign* key discount_id, instansi_id, dan *subscription_id* yang digunakan untuk menghubungkan data penggunaan diskon dengan data diskon, instansi, dan *subscription* terkait. Informasi penggunaan diskon disimpan melalui *field original_amount, discount_amount*, dan *final_amount* yang digunakan untuk menyimpan nominal sebelum diskon, nominal diskon, serta nominal akhir setelah diskon diterapkan.

Tabel 2.14 Spesifikasi Tabel *discount_usage*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik
2.	*discount_id*	*BIGINT UNSIGNED*	-	Referensi discount
3.	*instansi_id*	*BIGINT UNSIGNED*	-	Referensi instansi
4.	*subscription_id*	*BIGINT UNSIGNED*	-	Referensi *subscription*

5.	*original_amount*	*DECIMAL*	15,2	Nilai sebelum diskon
6.	*discount_amount*	*DECIMAL*	15,2	Nilai diskon
7.	*final_amount*	*DECIMAL*	15,2	Nilai setelah diskon
8.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
9.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.11 Spesifikasi Tabel *discounts*

Tabel *discounts* berfungsi untuk menyimpan dan mengelola data diskon yang tersedia dalam sistem HagaPlus. Tabel ini memiliki *primary key* id sebagai identitas setiap data diskon. Informasi diskon disimpan melalui *field* name, code, *type*, value, dan max_discount yang digunakan untuk menyimpan nama diskon, kode diskon, jenis diskon, nilai diskon, serta batas maksimum nominal potongan harga untuk diskon dengan jenis persentase.

Tabel 2.15 Spesifikasi Tabel *discounts*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik kode diskon
2.	*name*	*VARCHAR*	100	Nama promo atau diskon
3.	*code*	*VARCHAR*	50	Kode unik diskon
4.	*type*	*ENUM*	-	percentage atau fixed
5.	*value*	*DECIMAL*	15,2	Nilai diskon
6.	*max_discount*	*DECIMAL*	15,2	Batas maksimum diskon
7.	*usage_limit*	*INT*	-	Batas total penggunaan
8.	*usage_limit_per_instansi*	*INT*	-	Batas penggunaan per instansi
9.	*used_count*	*INT*	-	Jumlah penggunaan
10.	*valid_from*	*DATE*	-	Tanggal mulai berlaku
11.	*valid_until*	*DATE*	-	Tanggal berakhir

12.	*applicable_packages*	JSON	-	Paket yang berlaku
13.	*applicable_durations*	JSON	-	Durasi yang berlaku
14.	*target*	*VARCHAR*	50	Target penerima
15.	*is_public*	*TINYINT*	1	Dapat dilihat publik
16.	*promo_type*	*ENUM*	-	*package, extension, both*
17.	*is_active*	*TINYINT*	1	Status aktif
18.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
19.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.12 Spesifikasi Tabel *division_policies*

Tabel *division_policies* berfungsi menyimpan dan mengelola data terkait division policies. Tabel ini memiliki *Primary Key* id dan *Foreign Key* instansi_id, division_id. Rincian utamanya disimpan pada *field* name, description, work_days, work_start_*time*, serta *field* work_end_*time* yang menyimpan data work end *time*.

Tabel 2.16 Spesifikasi Tabel *division_policies*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik kebijakan
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
3.	*division_id*	*BIGINT UNSIGNED*	-	Referensi divisi
4.	*name*	*VARCHAR*	100	Nama kebijakan
5.	*description*	*TEXT*	-	Penjelasan kebijakan
6.	*work_days*	JSON	-	Hari kerja
7.	*work_start_time*	*TIME*	-	Jam mulai kerja
8.	*work_end_time*	*TIME*	-	Jam selesai kerja
9.	*work_hours_per_day*	*INT*	-	Jam kerja per hari
10.	*break_times*	JSON	-	Waktu istirahat

11.	*grace_period_minutes*	*INT*	-	Masa toleransi
12.	*max_late_minutes*	*INT*	-	Maksimal keterlambatan
13.	*early_leave_grace_minutes*	*INT*	-	Toleransi pulang awal
14.	*allow_overtime*	*TINYINT*	1	Mengizinkan lembur
15.	*max_overtime_hours_per_day*	*INT*	-	Maksimal lembur per hari
16.	*max_overtime_hours_per_week*	*INT*	-	Maksimal lembur per minggu
17.	*annual_leave_days*	*INT*	-	Cuti tahunan
18.	*sick_leave_days*	*INT*	-	Cuti sakit
19.	*personal_leave_days*	*INT*	-	Cuti pribadi
20.	*allow_negative_leave_balance*	*TINYINT*	1	Saldo cuti negatif
21.	*can_work_from_home*	*TINYINT*	1	Dapat *WFH*
22.	*flexible_hours*	*TINYINT*	1	Jam fleksibel
23.	*skip_weekends*	*TINYINT*	1	Lewati akhir pekan
24.	*skip_holidays*	*TINYINT*	1	Lewati hari libur
25.	*require_location_check*	*TINYINT*	1	Perlu pengecekan lokasi
26.	*allowed_radius_meters*	*DECIMAL*	10,2	Radius lokasi
27.	*allowed_locations*	JSON	-	Lokasi yang diperbolehkan
28.	*has_shifts*	*TINYINT*	1	Memiliki *shift*
29.	*shift_schedule*	JSON	-	Jadwal *shift*
30.	*custom_rules*	JSON	-	Aturan khusus
31.	*is_active*	*TINYINT*	1	Status aktif
32.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
33.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.13 Spesifikasi Tabel *divisions*

Tabel *divisions* berfungsi menyimpan dan mengelola data terkait *divisions*. Tabel ini memiliki *Primary Key* id dan *Foreign Key* instansi_id. Rincian utamanya disimpan pada *field* name, code, description, serta *field* is_active yang menyimpan data is active.

Tabel 2.17 Spesifikasi Tabel *divisions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik divisi
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
3.	*name*	*VARCHAR*	100	Nama divisi
4.	*code*	*VARCHAR*	10	Kode divisi
5.	*description*	*TEXT*	-	Penjelasan fungsi divisi
6.	*is_active*	*TINYINT*	1	Status aktif
7.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
8.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.14 Spesifikasi Tabel *email_templates*

Tabel *email_templates* berfungsi untuk menyimpan dan mengelola data templat email yang digunakan dalam sistem HagaPlus. Tabel ini memiliki *primary key* id sebagai identitas setiap templat email. Informasi templat email disimpan melalui *field name, slug, subject, body,* dan *variables* yang digunakan untuk menyimpan nama templat, identitas templat, subjek email, isi email, serta variabel yang dapat digunakan dalam templat email.

Tabel 2.18 Spesifikasi Tabel *email_templates*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik temp*late*
2.	*name*	*VARCHAR*	100	Nama temp*late* email
3.	*slug*	*VARCHAR*	100	Kunci unik temp*late*
4.	*subject*	*VARCHAR*	150	Subjek email
5.	*body*	*TEXT*	-	Isi email HTML
6.	*variables*	JSON	-	Variabel temp*late*

7.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
8.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.15 Spesifikasi Tabel *employee_policies*

Tabel *employee_policies* berfungsi untuk menyimpan dan mengelola data kebijakan kerja yang diterapkan kepada karyawan dalam sistem HagaPlus. Tabel ini memiliki *primary key* id serta *foreign key* employee_id dan instansi_id yang digunakan untuk menghubungkan data kebijakan dengan karyawan dan instansi terkait. Informasi kebijakan kerja disimpan melalui *field* name, description, work_days, work_start_*time*, dan work_end_*time* yang digunakan untuk menyimpan nama kebijakan, deskripsi kebijakan, hari kerja, waktu mulai kerja, serta waktu selesai kerja.

Tabel 2.19 Spesifikasi Tabel *employee_policies*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik
2.	*user_id*	*BIGINT UNSIGNED*	-	Referensi user
3.	*policy_id*	*BIGINT UNSIGNED*	-	Referensi policy
4.	*effective_date*	*DATE*	-	Tanggal mulai berlaku
5.	*end_date*	*DATE*	-	Tanggal berakhir
6.	*is_active*	*TINYINT*	1	Status aktif
7.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
8.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.16 Spesifikasi Tabel *employee_payroll_components*

Tabel *employee_payroll_components* berfungsi menyimpan dan mengelola data terkait employee payroll components. Tabel ini memiliki *Primary Key* id dan *Foreign Key* employee_id, payroll_component_id. Rincian utamanya disimpan pada *field* amount.

Tabel 2.20 Spesifikasi Tabel *employee_payroll_components*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik data komponen *payroll*

2.	*employee_id*	*BIGINT UNSIGNED*	-	ID karyawan
3.	*payroll_component_id*	*BIGINT UNSIGNED*	-	ID komponen *payroll*
4.	*amount*	*DECIMAL*	15,2	Nominal komponen payroll
5.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
6.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.17 Spesifikasi Tabel *payroll_components*

Tabel *payroll_components* berfungsi menyimpan dan mengelola data terkait payroll components. Tabel ini memiliki *Primary Key* id dan *Foreign Key* instansi_id. Rincian utamanya disimpan pada *field* name, *type*, is_taxable, is_default, serta *field* default_amount yang menyimpan data default amount.

Tabel 2.21 Spesifikasi Tabel *payroll_components*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik komponen payroll
2.	*instansi_id*	*BIGINT UNSIGNED*	-	ID instansi
3.	*name*	*VARCHAR*	255	Nama komponen payroll
4.	*type*	*ENUM*	-	Jenis komponen: allowance atau deduction
5.	*is_taxable*	*TINYINT*	1	Menentukan apakah komponen dikenakan pajak
6.	*is_default*	*TINYINT*	1	Menentukan apakah komponen merupakan komponen default
7.	*default_amount*	*DECIMAL*	15,2	Nilai nominal default komponen
8.	*created_at*	*TIMESTAMP*	-	Waktu data dibuat
9.	*updated_at*	*TIMESTAMP*	-	Waktu data diperbarui

2.5.18 Spesifikasi Tabel *employee_schedules*

Tabel *employee_schedules* berfungsi menyimpan dan mengelola data terkait employee schedules. Tabel ini memiliki *Primary Key* id dan *Foreign Key* user_id, policy_id. Rincian utamanya disimpan pada *field* effective_*date*, end_*date*, serta *field* is_active yang menyimpan data is active.

Tabel 2.22 Spesifikasi Tabel *employee_schedules*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik kebijakan
2.	*employee_id*	*BIGINT UNSIGNED*	-	Referensi employee
3.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
4.	*name*	*VARCHAR*	100	Nama kebijakan
5.	*description*	*TEXT*	-	Penjelasan
6.	*work_days*	JSON	-	Hari kerja
7.	*work_start_time*	*TIME*	-	Jam mulai
8.	*work_end_time*	*TIME*	-	Jam selesai
9.	*work_hours_per_day*	*INT*	-	Jam kerja per hari
10.	*break_times*	JSON	-	Waktu istirahat
11.	*grace_period_minutes*	*INT*	-	Masa toleransi
12.	*max_late_minutes*	*INT*	-	Maksimal terlambat
13.	*early_leave_grace_minutes*	*INT*	-	Toleransi pulang awal
14.	*allow_overtime*	*TINYINT*	1	Mengizinkan lembur
15.	*max_overtime_hours_per_day*	*INT*	-	Maksimal lembur harian
16.	*max_overtime_hours_per_week*	*INT*	-	Maksimal lembur mingguan

17.	*annual_leave_days*	*INT*	-	Cuti tahunan
18.	*sick_leave_days*	*INT*	-	Cuti sakit
19.	*personal_leave_days*	*INT*	-	Cuti pribadi
20.	*allow_negative_leave_balance*	*TINYINT*	1	Saldo cuti negatif
21.	*can_work_from_home*	*TINYINT*	1	Dapat *WFH*
22.	*flexible_hours*	*TINYINT*	1	Jam fleksibel
23.	*skip_weekends*	*TINYINT*	1	Lewati akhir pekan
24.	*skip_holidays*	*TINYINT*	1	Lewati hari libur
25.	*require_location_check*	*TINYINT*	1	Pengecekan lokasi
26.	*allowed_radius_meters*	*DECIMAL*	10,2	Radius lokasi
27.	*allowed_locations*	JSON	-	Lokasi diperbolehkan
28.	*has_shifts*	*TINYINT*	1	Memiliki *shift*
29.	*shift_schedule*	JSON	-	Jadwal *shift*
30.	*custom_rules*	JSON	-	Aturan khusus
31.	*is_active*	*TINYINT*	1	Status aktif
32.	*effective_from*	*TIMESTAMP*	-	Mulai berlaku
33.	*effective_until*	*TIMESTAMP*	-	Berakhir berlaku
34.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
35.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.19 Spesifikasi Tabel *employee_tax_forms*

Tabel *employee_tax_forms* berfungsi menyimpan dan mengelola data terkait employee tax forms. Tabel ini memiliki *Primary Key* id dan *Foreign Key* employee_id. Rincian utamanya disimpan pada *field tax_year, period_start, period_end, form_number,* serta *field* data yang berisi data lengkap formulir pajak dalam format json terenkripsi.

Tabel 2.23 Spesifikasi Tabel *employee_tax_forms*

No	Nama *Field*	*Type*	*Length*	Keterangan

1.	*id*	*BIGINT UNSIGNED*	-	ID unik bukti potong
2.	*employee_id*	*BIGINT UNSIGNED*	-	Karyawan pemilik
3.	*tax_year*	YEAR	-	Tahun pajak
4.	*period_start*	*TINYINT*	-	Bulan awal periode
5.	*period_end*	*TINYINT*	-	Bulan akhir periode
6.	*form_number*	*VARCHAR*	50	Nomor formulir
7.	*data*	LONG*TEXT*	-	Data formulir pajak
8.	*is_published*	*TINYINT*	1	Status publikasi
9.	*created_by*	*BIGINT UNSIGNED*	-	Admin pembuat
10.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
11.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.20 Spesifikasi Tabel *employees*

Tabel *employees* berfungsi menyimpan profil lengkap dan data kepegawaian dari setiap karyawan. Tabel ini memiliki *Primary Key* id dan *Foreign Key user_id, instansi_id, branch_id, division_id, department_id, position_id, manager_id, instansi_role_id, attendance_policy_id, employee_id*. Rincian utamanya disimpan pada *field* nik, npwp, ptkp_status, metode_pajak, serta *field* bpjs_kesehatan_number yang berisi nomor kartu bpjs kesehatan karyawan (data sensitif terenkripsi).

Tabel 2.24 Spesifikasi Tabel *employees*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik karyawan
2.	*user_id*	*BIGINT UNSIGNED*	-	Akun pengguna yang terhubung
3..	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi/perusahaan
4.	*branch_id*	*BIGINT UNSIGNED*	-	Kantor cabang

5.	*division_id*	*BIGINT UNSIGNED*	-	Divisi
6.	*department_id*	*BIGINT UNSIGNED*	-	Departemen
7.	*position_id*	*BIGINT UNSIGNED*	-	Jabatan
8.	*manager_id*	*BIGINT UNSIGNED*	-	Atasan langsung
9.	*instansi_role_id*	*BIGINT UNSIGNED*	-	Peran karyawan
10.	*attendance_policy_id*	*BIGINT UNSIGNED*	-	Kebijakan kehadiran
11.	*employee_id*	*VARCHAR*	50	Nomor ID karyawan
12.	*nik*	*VARCHAR*	255	Nomor Induk Kependudukan
13.	*npwp*	*VARCHAR*	255	Nomor Pokok Wajib Pajak
14.	*ptkp_status*	*ENUM*	-	Status PTKP
15.	*metode_pajak*	*ENUM*	-	Metode PPh21
16.	*bpjs_kesehatan_number*	*VARCHAR*	255	Nomor BPJS Kesehatan
17.	*bpjs_kesehatan_faskes*	*VARCHAR*	100	Faskes BPJS
18.	*bpjs_kesehatan_start_date*	*DATE*	-	Tanggal mulai BPJS Kesehatan
19.	bpjs_kesehatan_tanggungan	*INT*	-	Jumlah tanggungan
20.	*bpjs_kesehatan_card*	*VARCHAR*	255	*Scan* kartu BPJS Kesehatan
21.	*bpjs_tk_number*	*VARCHAR*	255	Nomor BPJS Ketenagakerjaan
22.	*bpjs_jp_aktif*	*TINYINT*	1	Status Jaminan Pensiun

23.	*bpjs_jkk_rate*	*DECIMAL*	5,2	Tarif JKK
24.	*bpjs_tk_start_date*	*DATE*	-	Tanggal mulai BPJS TK
25.	*bpjs_tk_card*	*VARCHAR*	255	*Scan* kartu BPJS TK
26.	*date_of_birth*	*DATE*	-	Tanggal lahir
27.	tempat_lahir	*VARCHAR*	100	Tempat lahir
28.	*gender*	*ENUM*	-	Jenis kelamin
29.	status_perkawinan	*ENUM*	-	Status perkawinan
30.	jumlah_tanggungan	*INT*	-	Jumlah tanggungan pajak
31.	kewarganegaraan	*VARCHAR*	50	Kewarganegaraan
32.	*address*	*TEXT*	-	Alamat domisili
33.	alamat_ktp	*TEXT*	-	Alamat KTP
34.	*phone*	*VARCHAR*	255	Nomor telepon
35.	*emergency_contact_name*	*VARCHAR*	100	Nama kontak darurat
36.	*emergency_contact_relation*	*VARCHAR*	50	Hubungan kontak darurat
37.	*emergency_contact_phone*	*VARCHAR*	255	Nomor kontak darurat
38.	*salary*	*DECIMAL*	15,2	Gaji pokok
39.	tunjangan_jabatan	*DECIMAL*	15,2	Tunjangan jabatan
40.	tunjangan_transport	*DECIMAL*	15,2	Tunjangan transportasi
41.	tunjangan_makan	*DECIMAL*	15,2	Tunjangan makan
42.	tunjangan_hadir	*DECIMAL*	15,2	Tunjangan kehadiran
43.	*bank_name*	*VARCHAR*	50	Nama bank
44.	*bank_account_number*	*VARCHAR*	255	Nomor rekening
45.	*bank_account_holder*	*VARCHAR*	100	Pemilik rekening
46.	foto_karyawan	*VARCHAR*	255	Foto karyawan
47.	*scan_ktp*	*VARCHAR*	255	*Scan* KTP
48.	*scan_npwp*	*VARCHAR*	255	*Scan* NPWP
49.	*scan_kk*	*VARCHAR*	255	*Scan* KK

50.	*catatan_hr*	*TEXT*	-	Catatan HR
51.	*hire_date*	*DATE*	-	Tanggal mulai bekerja
52.	tanggal_berhenti	*DATE*	-	Tanggal berhenti
53.	status	*ENUM*	-	Status karyawan
54.	status_karyawan	*ENUM*	-	Jenis hubungan kerja
55.	*grade_level*	*VARCHAR*	50	Tingkat/golongan jabatan
56.	*deleted_by*	*BIGINT UNSIGNED*	-	Pengguna yang menghapus
57.	*deleted_at*	*TIMESTAMP*	-	Waktu penghapusan
58.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
59.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.21 Spesifikasi Tabel *failed_jobs*

Tabel *failed_jobs* berfungsi untuk menyimpan dan mengelola data pekerjaan (*job*) yang gagal diproses oleh sistem HagaPlus. Tabel ini memiliki *primary key* id sebagai identitas setiap data pekerjaan yang gagal. Informasi pekerjaan disimpan melalui *field uuid, connection, queue, payload, dan exception* yang digunakan untuk menyimpan identitas unik pekerjaan, koneksi yang digunakan, antrean pekerjaan, data pekerjaan yang diproses, serta informasi kesalahan yang menyebabkan pekerjaan gagal.

Tabel 2.25 Spesifikasi Tabel *failed_jobs*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik
2.	*uuid*	*VARCHAR*	255	*UUID*
3.	*connection*	*TEXT*	-	*Connection*
4.	*queue*	*TEXT*	-	*Queue*
5.	*payload*	LONG*TEXT*	-	*Payload*
6.	*exception*	LONG*TEXT*	-	*Exception*
7.	*failed_at*	*TIMESTAMP*	-	Waktu gagal

2.5.22 Spesifikasi Tabel *holidays*

Tabel *holidays* berfungsi menyimpan dan mengelola data terkait holidays. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* start_*date*, end_*date*, name, description, serta *field is_national_holiday* yang menyimpan data is national holiday.

Tabel 2.26 Spesifikasi Tabel *holidays*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik hari libur
2.	*start_date*	*DATE*	-	Tanggal mulai
3.	*end_date*	*DATE*	-	Tanggal selesai
4.	*name*	*VARCHAR*	100	Nama hari libur
5.	*description*	*TEXT*	-	Penjelasan
6.	*is_national_holiday*	*TINYINT*	1	hari libur nasional
7.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
8.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.23 Spesifikasi Tabel *instansi_role_permissions*

Tabel *instansi_role_permissions* berfungsi menyimpan dan mengelola data terkait instansi role permissions. Tabel ini memiliki *Primary Key* id dan *Foreign Key instansi_role_id, permission_id*. Tabel ini hanya berisi struktur relasi atau data pendukung tanpa rincian atribut spesifik.

Tabel 2.27 Spesifikasi Tabel *instansi_role_permissions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	id	*BIGINT UNSIGNED*	-	ID unik
2.	instansi_role_id	*BIGINT UNSIGNED*	-	Referensi role instansi
3.	permission_id	*BIGINT UNSIGNED*	-	Referensi permission
4.	created_at	*TIMESTAMP*	-	Waktu dibuat
5.	up*dat*ed_at	*TIMESTAMP*	-	Waktu diperbarui

2.5.24 Spesifikasi Tabel *instansi_roles*

Tabel *instansi_roles* berfungsi menyimpan dan mengelola data terkait instansi roles. Tabel ini memiliki *Primary Key* id dan *Foreign Key instansi_id, system_role_id*. Rincian utamanya disimpan pada *field* name, description, is_active, is_default, serta *field data_scope* yang menyimpan *data scope*.

Tabel 2.28 Spesifikasi Tabel *instansi_roles*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik *role*
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik *role*
3.	*system_role_id*	*BIGINT UNSIGNED*	-	Referensi system *role*
4.	*name*	*VARCHAR*	100	Nama *role*
5.	*description*	*TEXT*	-	Penjelasan *role*
6.	*is_active*	*TINYINT*	1	Status aktif
7.	*is_default*	*TINYINT*	1	Status default
8.	*data_scope*	*ENUM*	-	Lingkup data
9.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
10.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.25 Spesifikasi Tabel *instansis*

Tabel *instansis* berfungsi menyimpan data profil perusahaan atau instansi yang menggunakan sistem. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* nama_instansi, display_name, email, website, serta *field* phone yang menyimpan data phone.

Tabel 2.29 Spesifikasi Tabel *instansis*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik instansi
2.	*nama_instansi*	*VARCHAR*	100	Nama resmi instansi
3.	*display_name*	*VARCHAR*	50	Nama tampilan
4.	*email*	*VARCHAR*	100	Email resmi
5.	*website*	*VARCHAR*	100	Website resmi

6.	*phone*	*VARCHAR*	20	Nomor telepon
7.	*address*	*TEXT*	-	Alamat
8.	*city*	*VARCHAR*	50	Kota
9.	*npwp*	*VARCHAR*	30	NPWP
10.	*nib*	*VARCHAR*	30	NIB
11.	*logo*	*VARCHAR*	255	Logo
12.	*status*	*ENUM*	-	Status instansi
13.	*settings*	JSON	-	Pengaturan instansi
14.	*retention_policy*	JSON	-	Kebijakan retensi
15.	*onboarding_step*	*TINYINT* UNSIGNED	-	Tahapan onboarding
16.	*onboarding_completed*	*TINYINT*	1	Status onboarding
17.	*onboarding_completed_at*	*TIMESTAMP*	-	Waktu onboarding selesai
18.	*archived_at*	*TIMESTAMP*	-	Waktu diarsipkan
19.	*archived_by*	*BIGINT UNSIGNED*	-	Pengguna yang mengarsipkan
20.	*deleted_by*	*BIGINT UNSIGNED*	-	Pengguna yang menghapus
21.	*deleted_at*	*TIMESTAMP*	-	Waktu penghapusan
22.	*created_at*	*TIMESTAMP*	-	Waktu didaftarkan
23.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.26 Spesifikasi Tabel *job_batches*

Tabel job_batches berfungsi menyimpan dan mengelola data terkait job batches. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* name, *total_jobs*, *pending_jobs*, *failed_jobs*, serta *field failed_job_ids* yang menyimpan data *failed job ids*.

Tabel 2.30 Spesifikasi Tabel *job_batches*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*VARCHAR*	255	ID unik

2.	*name*	*VARCHAR*	255	Nama entitas
3.	*total_jobs*	*INT*	-	Total *jobs*
4.	*pending_jobs*	*INT*	-	*Jobs* tertunda
5.	*failed_jobs*	*INT*	-	*Jobs* gagal
6.	*failed_job_ids*	LONG*TEXT*	-	ID *jobs* gagal
7.	*options*	MEDIUM*TEXT*	-	Options
8.	*cancelled_at*	*INT*	-	Waktu pembatalan
9.	*created_at*	*INT*	-	Waktu dibuat
10.	*finished_at*	*INT*	-	Waktu selesai

2.5.27 Spesifikasi Tabel *jobs*

Tabel *jobs* berfungsi menyimpan dan mengelola data terkait *jobs*. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field queue, payload*, *attempts, reserved_at,* serta *field available_at* yang menyimpan data *available at*.

Tabel 2.31 Spesifikasi Tabel *jobs*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik
2.	*queue*	*VARCHAR*	255	*Queue*
3.	*payload*	LONG*TEXT*	-	*Payload*
4.	*attempts*	*TINYINT* UNSIGNED	-	Jumlah percobaan
5.	*reserved_at*	*INT* UNSIGNED	-	Waktu reservasi
6.	*available_at*	*INT* UNSIGNED	-	Waktu tersedia
7.	*created_at*	*INT* UNSIGNED	-	Waktu dibuat

2.5.28 Spesifikasi Tabel *leaves*

Tabel *leaves* berfungsi mencatat riwayat dan status pengajuan cuti atau izin karyawan. Tabel ini memiliki *Primary Key* id dan *Foreign Key user_id, original_leave_id, supervisor_id, hrd_id*. Rincian utamanya disimpan pada *field leave_type*, *leave_category, deduct_from_quota, requires_medical_certificate*, serta *field* start_*date* yang berisi tanggal mulai cuti.

Tabel 2.32 Spesifikasi Tabel *leaves*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengajuan cuti
2.	*user_id*	*BIGINT UNSIGNED*	-	Karyawan pengaju
3.	*original_leave_id*	*BIGINT UNSIGNED*	-	Referensi original *leave*
4.	*leave_type*	*ENUM*	-	Jenis cuti
5.	*leave_category*	*ENUM*	-	Kategori cuti
6.	*deduct_from_quota*	*TINYINT*	1	Potong kuota cuti
7.	*requires_medical_certificate*	*TINYINT*	1	Memerlukan surat dokter
8.	*start_date*	*DATE*	-	Tanggal mulai
9.	*end_date*	*DATE*	-	Tanggal selesai
10.	*actual_return_date*	*DATE*	-	Tanggal kembali aktual
11.	*days_count*	*INT*	-	Jumlah hari
12.	*is_half_day*	*TINYINT*	1	Setengah hari
13.	*reason*	*TEXT*	-	Alasan cuti
14.	*attachment*	*VARCHAR*	255	Dokumen pendukung
15.	*medical_certificate_path*	*VARCHAR*	255	Surat dokter
16.	*status*	*VARCHAR*	255	Status pengajuan
17.	*cancellation_status*	*ENUM*	-	Status pembatalan
18.	*cancellation_requested_at*	*TIMESTAMP*	-	Waktu perm*int*aan pembatalan
19.	*cancellation_approved_at*	*TIMESTAMP*	-	Waktu persetujuan pembatalan
20.	*cancellation_notes*	*TEXT*	-	Catatan pembatalan
21.	*reschedule_status*	*ENUM*	-	Status *reschedule*

22.	*new_start_date*	*DATE*	-	Tanggal mulai baru
23.	*new_end_date*	*DATE*	-	Tanggal selesai baru
24.	*reschedule_reason*	*TEXT*	-	Alasan *reschedule*
25.	*reschedule_requested_at*	*TIMESTAMP*	-	Waktu perm*int*aan *reschedule*
26.	*reschedule_approved_at*	*TIMESTAMP*	-	Waktu persetujuan *reschedule*
27.	*reschedule_notes*	*TEXT*	-	Catatan *reschedule*
28.	*extension_requested*	*TINYINT*	1	Perm*int*aan perpanjangan
29.	*approved_by*	*BIGINT UNSIGNED*	-	Atasan pemroses
30.	*approved_at*	*TIMESTAMP*	-	Waktu persetujuan
31.	*rejection_reason*	*TEXT*	-	Alasan penolakan
32.	*supervisor_id*	*BIGINT UNSIGNED*	-	*Supervisor*
33.	*supervisor_approved_at*	*TIMESTAMP*	-	Waktu persetujuan *supervisor*
34.	*supervisor_note*	*TEXT*	-	Catatan *supervisor*
35.	*hrd_id*	*BIGINT UNSIGNED*	-	HRD
36.	*hrd_approved_at*	*TIMESTAMP*	-	Waktu persetujuan HRD
37.	*hrd_note*	*TEXT*	-	Catatan HRD
38.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
39.	up*date*d_at	*TIMESTAMP*	-	Waktu diperbarui

2.5.29 Spesifikasi Tabel *login_logs*

Tabel *login_logs* berfungsi menyimpan dan mengelola data terkait login logs. Tabel ini memiliki *Primary Key* id dan *Foreign Key* user_id. Rincian utamanya disimpan pada *field* ip_address, user_agent, serta *field* logged_in_at yang menyimpan data logged in at.

Tabel 2.33 Spesifikasi Tabel *login_logs*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik log masuk
2.	*user_id*	*BIGINT UNSIGNED*	-	Pengguna yang login
3.	*ip_address*	*VARCHAR*	45	Alamat IP
4.	*user_agent*	*TEXT*	-	*Browser*/perangkat
5.	*logged_in_at*	*TIMESTAMP*	-	Waktu *login*

2.5.30 Spesifikasi Tabel *packages*

Tabel *packages* berfungsi menyimpan dan mengelola data terkait packages. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* name, description, price, duration_days, serta *field* max_employees yang menyimpan data max employees.

Tabel 2.34 Spesifikasi Tabel *packages*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik paket
2.	*name*	*VARCHAR*	100	Nama paket
3.	*description*	*TEXT*	-	Deskripsi paket
4.	*price*	*DECIMAL*	15,2	Harga paket
5.	*duration_days*	*INT*	-	Durasi paket
6.	*max_employees*	*INT*	-	Maksimal karyawan
7.	*max_admins*	*INT*	-	Maksimal admin
8.	*max_branches*	*INT*	-	Maksimal cabang
9.	*features*	JSON	-	Fitur paket
10.	*is_active*	*TINYINT*	1	Status aktif
11.	*is_default*	*TINYINT*	1	Paket default
12.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
13.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.31 Spesifikasi Tabel *password_reset_tokens*

Tabel *password_reset_tokens* berfungsi untuk menyimpan dan mengelola data yang digunakan dalam proses pengaturan ulang kata sandi pengguna pada

sistem HagaPlus. Tabel ini tidak memiliki *primary key* khusus. Informasi pengaturan ulang kata sandi disimpan melalui *field* email dan token yang digunakan untuk menyimpan alamat email pengguna serta token yang digunakan dalam proses pengaturan ulang kata sandi.

Tabel 2.35 Spesifikasi Tabel *password_reset_tokens*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*email*	*VARCHAR*	255	ID unik
2.	*token*	*VARCHAR*	255	*Token reset password*
3.	*created_at*	*TIMESTAMP*	-	Waktu dibuat

2.5.32 Spesifikasi Tabel *payment_methods*

Tabel *payment_methods* berfungsi menyimpan dan mengelola data terkait payment methods. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* name, *type*, account_number, account_name, serta *field* bank_name yang berisi nama bank atau penyedia layanan pembayaran.

Tabel 2.36 Spesifikasi Tabel payment_methods

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik metode pembayaran
2.	*name*	*VARCHAR*	100	Nama metode pembayaran
3.	*type*	*ENUM*	-	Jenis metode pembayaran
4.	*account_number*	*VARCHAR*	50	Nomor rekening/akun
5.	*account_name*	*VARCHAR*	100	Nama pemilik akun
6.	*bank_name*	*VARCHAR*	50	Nama bank
7.	*qris_image*	*VARCHAR*	255	Gambar QRIS
8.	*qris_data*	*TEXT*	-	Data QRIS
9.	*description*	*TEXT*	-	Penjelasan
10.	*is_active*	*TINYINT*	1	Status aktif
11.	*is_default_on_invoice*	*TINYINT*	1	Default pada *invoice*

12.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
13.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.33 Spesifikasi Tabel *payrolls*

Tabel *payrolls* berfungsi menyimpan rekapan perhitungan gaji bulanan karyawan. Tabel ini memiliki *Primary Key* id dan Foreign Key instansi_id, user_id, employee_id. Rincian utamanya disimpan pada *field* period_year, period_month, created_*date*, gaji_pokok, serta *field* payment_*date* yang berisi tanggal pembayaran gaji dilakukan.

Tabel 2.37 Spesifikasi Tabel *payrolls*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	id	*BIGINT UNSIGNED*	-	ID unik slip gaji
2.	instansi_id	*BIGINT UNSIGNED*	-	Instansi penerbit
3.	user_id	*BIGINT UNSIGNED*	-	Karyawan penerima
4.	employee_id	*BIGINT UNSIGNED*	-	Data karyawan
5.	period_year	*INT*	-	Tahun periode
6.	period_month	*INT*	-	Bulan periode
7.	created_*date*	*DATE*	-	Tanggal dibuat
8.	gaji_pokok	*DECIMAL*	15,2	Gaji pokok
9.	payment_*date*	*DATE*	-	Tanggal pembayaran
10.	payment_status	*ENUM*	-	draft, processed, paid
11.	processed_at	*TIMESTAMP*	-	Waktu mulai diproses
12.	paid_at	*TIMESTAMP*	-	Waktu dibayar
13.	notes	*TEXT*	-	Catatan pembayaran
14.	created_by	*BIGINT UNSIGNED*	-	Admin pembuat

15.	*approved*_by	*BIGINT UNSIGNED*	-	Admin penyetuju
16.	*approved*_at	*TIMESTAMP*	-	Waktu persetujuan
17.	approval_status	*ENUM*	-	Status persetujuan
18.	rejected_at	*TIMESTAMP*	-	Waktu ditolak
19.	rejection_reason	*TEXT*	-	Alasan penolakan
20.	bank_name	*VARCHAR*	50	Bank tujuan
21.	bank_account_number	*VARCHAR*	50	Nomor rekening
22.	bank_account_holder	*VARCHAR*	100	Pemilik rekening
23.	tunjangan_jabatan	*DECIMAL*	15,2	Tunjangan jabatan
24.	tunjangan_makan	*DECIMAL*	15,2	Tunjangan makan
25.	tunjangan_transport	*DECIMAL*	15,2	Tunjangan transport
26.	lembur	*DECIMAL*	15,2	Upah lembur
27.	bonus	*DECIMAL*	15,2	Bonus
28.	reimburse	*DECIMAL*	15,2	Reimbursement
29.	thr	*DECIMAL*	15,2	THR
30.	bpjs_kesehatan	*DECIMAL*	15,2	Potongan BPJS Kesehatan
31.	bpjs_tk	*DECIMAL*	15,2	Potongan BPJS TK
32.	pph21	*DECIMAL*	15,2	Potongan PPh21
33.	potongan_absensi	*DECIMAL*	15,2	Potongan absensi
34.	kasbon	*DECIMAL*	15,2	Potongan kasbon
35.	potongan_lainnya	*DECIMAL*	15,2	Potongan lainnya
36.	total_pendapatan	*DECIMAL*	15,2	Total pendapatan
37.	total_potongan	*DECIMAL*	15,2	Total potongan
38.	gaji_bersih	*DECIMAL*	15,2	Gaji bersih
39.	net_salary	*DECIMAL*	15,2	Nilai gaji bersih cadangan
40.	created_at	*TIMESTAMP*	-	Waktu dibuat
41.	up*date*d_at	*TIMESTAMP*	-	Waktu diperbarui

2.5.34 Spesifikasi Tabel *permissions*

Tabel *permissions* berfungsi menyimpan dan mengelola data terkait permissions. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* name, slug, group, serta *field* description yang berisi penjelasan atau uraian tambahan.

Tabel 2.38 Spesifikasi Tabel *permissions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	id	*BIGINT UNSIGNED*	-	ID unik permission
2.	name	*VARCHAR*	100	Nama izin akses
3.	slug	*VARCHAR*	100	Nama permission untuk pemrograman
4.	group	*VARCHAR*	50	Kelompok modul
5.	description	*TEXT*	-	Penjelasan permission
6.	created_at	*TIMESTAMP*	-	Waktu dibuat
7.	up*date*d_at	*TIMESTAMP*	-	Waktu diperbarui

2.5.35 Spesifikasi Tabel *positions*

Tabel *positions* berfungsi menyimpan dan mengelola data terkait positions. Tabel ini memiliki *Primary Key* id dan *Foreign Key instansi_id, division_id, department_id, instansi_role_id*. Rincian utamanya disimpan pada *field name*, description, serta *field is_active* yang menyimpan data *is active*.

Tabel 2.39 Spesifikasi Tabel *positions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik jabatan
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
3.	*division_id*	*BIGINT UNSIGNED*	-	Referensi divisi
4.	*department_id*	*BIGINT UNSIGNED*	-	Referensi departemen

5.	*instansi_role_id*	*BIGINT UNSIGNED*	-	Referensi role
6.	*name*	*VARCHAR*	100	Nama jabatan
7.	*description*	*TEXT*	-	Penjelasan jabatan
8.	*is_active*	*TINYINT*	1	Status aktif
9.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
10.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.36 Spesifikasi Tabel *reimbursements*

Tabel *reimbursements* berfungsi menyimpan dan mengelola data terkait *reimbursements*. Tabel ini memiliki *Primary Key id* dan *Foreign Key user_id, employee_id, supervisor_id, manager_id, finance_approver_id, payroll_id*. Rincian utamanya disimpan pada *field reference_number, category, description, date_of_expense*, serta *field amount* yang menyimpan data *amount*.

Tabel 2.40 Spesifikasi Tabel *reimbursements*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik
2.	*user_id*	*BIGINT UNSIGNED*	-	Referensi user
3.	*employee_id*	*BIGINT UNSIGNED*	-	Referensi employee
4.	*reference_number*	*VARCHAR*	50	Nomor referensi
5.	*category*	*VARCHAR*	50	Kategori
6.	*description*	*TEXT*	-	Deskripsi
7.	*date_of_expense*	*DATE*	-	Tanggal pengeluaran
8.	*amount*	*DECIMAL*	15,2	Jumlah biaya
9.	*currency*	*VARCHAR*	10	Mata uang
10.	*proof_file*	*VARCHAR*	255	File bukti
11.	*project_code*	*VARCHAR*	50	Kode proyek
12.	*payment_method*	*VARCHAR*	50	Metode pembayaran
13.	*bank_name*	*VARCHAR*	50	Nama bank

14.	*bank_account_number*	*VARCHAR*	50	Nomor rekening
15.	*bank_account_holder*	*VARCHAR*	100	Pemilik rekening
16.	*status*	*ENUM*	-	Status *reimbursement*
17.	*rejection_reason*	*TEXT*	-	Alasan penolakan
18.	*approved_amount*	*DECIMAL*	15,2	Jumlah disetujui
19.	*supervisor_id*	*BIGINT UNSIGNED*	-	*Supervisor*
20.	*manager_id*	*BIGINT UNSIGNED*	-	*Manager*
21.	*finance_approver_id*	*BIGINT UNSIGNED*	-	*Approver finance*
22.	*supervisor_approved_at*	*TIMESTAMP*	-	Waktu persetujuan *supervisor*
23.	*manager_approved_at*	*TIMESTAMP*	-	Waktu persetujuan *manager*
24.	*finance_verified_at*	*TIMESTAMP*	-	Waktu verifikasi *finance*
25.	*paid_at*	*TIMESTAMP*	-	Waktu pembayaran
26.	*payment_proof_file*	*VARCHAR*	255	Bukti pembayaran
27.	*payroll_id*	*BIGINT UNSIGNED*	-	Referensi payroll
28.	*deleted_at*	*TIMESTAMP*	-	Waktu *soft delete*
29.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
30.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.37 Spesifikasi Tabel *sessions*

Tabel *sessions* berfungsi menyimpan dan mengelola data terkait *sessions*. Tabel ini memiliki *Primary Key id* dan *Foreign Key user_id*. Rincian utamanya disimpan pada *field ip_address, user_agent, payload*, serta *field* last_activity yang menyimpan data last *activity*.

Tabel 2.41 Spesifikasi Tabel *sessions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*VARCHAR*	255	ID unik session
2.	*user_id*	*BIGINT UNSIGNED*	-	Referensi *user*
3.	*ip_address*	*VARCHAR*	45	Alamat IP
4.	*user_agent*	*TEXT*	-	*User agent*
5.	*payload*	*LONGTEXT*	-	*Payload session*
6.	*last_activity*	*INT*	-	Aktivitas terakhir

2.5.38 Spesifikasi Tabel *shift_swap_requests*

Tabel *shift_swap_requests* berfungsi menyimpan dan mengelola data terkait *shift* swap requests. Tabel ini memiliki *Primary Key* id dan *Foreign Key* requester_id, receiver_id. Rincian utamanya disimpan pada *field swap_date, target_date, reason, status*, serta *field approved_by* yang berisi id atasan yang menyetujui penukaran *shift*.

Tabel 2.42 Spesifikasi Tabel *shift_swap_requests*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengajuan tukar *shift*
2.	*requester_id*	*BIGINT UNSIGNED*	-	Karyawan pengaju
3.	*receiver_id*	*BIGINT UNSIGNED*	-	Karyawan penerima
4.	*swap_date*	*DATE*	-	Tanggal *shift* pengaju
5.	*target_date*	*DATE*	-	Tanggal *shift* penerima
6.	*reason*	*TEXT*	-	Alasan tukar *shift*
7.	*status*	*ENUM*	-	Status pengajuan
8.	*approved_by*	*BIGINT UNSIGNED*	-	ID atasan
9.	*rejection_note*	*TEXT*	-	Catatan penolakan
10.	*is_cross_division*	*TINYINT*	1	Antar divisi
11.	*is_cross_position*	*TINYINT*	1	Antar jabatan
12.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
13.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.39 Spesifikasi Tabel *subscription_requests*

Tabel *subscription_requests* berfungsi menyimpan dan mengelola data terkait *subscription requests*. Tabel ini memiliki *Primary Key* id dan *Foreign Key instansi_id, package_id, from_package_id, subscription_id, transaction_id, payment_method_id, target_package_id, discount_id*. Rincian utamanya disimpan pada *field amount*, *original_amount, extension_months, payment_status*, serta *field processed_at* yang berisi waktu permintaan diproses oleh *admin*.

Tabel 2.43 Spesifikasi Tabel *subscription_requests*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik perm*int*aan
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pengaju
3.	*package_id*	*BIGINT UNSIGNED*	-	Paket baru
4.	*from_package_id*	*BIGINT UNSIGNED*	-	Paket lama
5.	*subscription_id*	*BIGINT UNSIGNED*	-	Langganan terkait
6.	*amount*	*DECIMAL*	15,2	Total tagihan setelah diskon
7.	*original_amount*	*DECIMAL*	15,2	Harga sebelum diskon
8.	*extension_months*	*INT*	-	Jumlah bulan perpanjangan
9.	*payment_status*	*ENUM*	-	Status pembayaran
10.	*processed_at*	*TIMESTAMP*	-	Waktu diproses
11.	*transaction_id*	*VARCHAR*	100	ID transaksi
12.	*notes*	*TEXT*	-	Catatan admin
13.	*created_by*	*BIGINT UNSIGNED*	-	Pengguna pengaju

14.	*payment_method_id*	*BIGINT UNSIGNED*	-	Metode pembayaran
15.	*payment_method*	*VARCHAR*	50	Nama metode pembayaran
16.	*payment_proof*	*VARCHAR*	255	Bukti pembayaran
17.	*payment_proof_uploaded_at*	*TIMESTAMP*	-	Waktu upload bukti
18.	*target_package_id*	*BIGINT UNSIGNED*	-	Paket tujuan
19.	*discount_id*	*BIGINT UNSIGNED*	-	Kode diskon
20.	*discount_amount*	*DECIMAL*	15,2	Nilai diskon
21.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
22.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.40 Spesifikasi Tabel *subscriptions*

Tabel *subscriptions* berfungsi menyimpan dan mengelola data terkait subscriptions. Tabel ini memiliki *Primary Key* id *dan Foreign Key* instansi_id, package_id. Rincian utamanya disimpan pada *field* status, effective_*date*, trial_ends_at, is_trial, serta *field* start_*date* yang berisi tanggal mulai periode langganan berbayar.

Tabel 2.44 Spesifikasi Tabel *subscriptions*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik langganan
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
3.	*package_id*	*BIGINT UNSIGNED*	-	Paket layanan
4.	*status*	*ENUM*	-	Status langganan
5.	*effective_date*	*DATE*	-	Tanggal efektif
6.	*trial_ends_at*	*DATE*	-	Akhir masa *trial*
7.	*is_trial*	*TINYINT*	1	Status *trial*
8.	*start_date*	*DATE*	-	Tanggal mulai berbayar
9.	*end_date*	*DATE*	-	Tanggal berakhir

10.	*price*	*DECIMAL*	15,2	Harga langganan
11.	*override_features*	*TEXT*	-	Fitur tambahan
12.	*payment_status*	*ENUM*	-	Status pembayaran
13.	*payment_method*	*VARCHAR*	50	Metode pembayaran
14.	*payment_date*	*DATE*	-	Tanggal pembayaran
15.	*notes*	*TEXT*	-	Catatan langganan
16.	*cancelled_at*	*DATETIME*	-	Waktu pembatalan
17.	*cancel_reason*	*TEXT*	-	Alasan pembatalan
18.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
19.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.41 Spesifikasi Tabel *system_roles*

Tabel *system_roles* berfungsi menyimpan dan mengelola data terkait *system roles*. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field* name, slug, serta *field* description yang berisi penjelasan singkat tentang cakupan akses peran ini.

Tabel 2.45 Spesifikasi Tabel *system_roles*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik peran sistem
2.	*name*	*VARCHAR*	100	Nama peran sistem
3.	*slug*	*VARCHAR*	100	Versi singkat nama peran
4.	*description*	*TEXT*	-	Penjelasan cakupan akses
5.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
6.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.42 Spesifikasi Tabel *system_settings*

Tabel *system_settings* berfungsi menyimpan dan mengelola data terkait *system settings*. Tabel ini memiliki *Primary Key* id. Rincian utamanya disimpan pada *field key, value, type, group*, serta *field description* yang berisi penjelasan singkat fungsi dari pengaturan ini.

Tabel 2.46 Spesifikasi Tabel *system_settings*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	id	*BIGINT UNSIGNED*	-	ID unik pengaturan
2.	key	*VARCHAR*	100	Kunci pengaturan
3.	value	*TEXT*	-	Nilai pengaturan
4.	*type*	*VARCHAR*	50	Tipe data
5.	group	*VARCHAR*	50	Kelompok pengaturan
6.	description	*TEXT*	-	Penjelasan pengaturan
7.	is_public	*TINYINT*	1	Dapat diakses pengguna
8.	created_at	*TIMESTAMP*	-	Waktu dibuat
9.	up*dated*_at	*TIMESTAMP*	-	Waktu diperbarui

2.5.43 Spesifikasi Tabel *tax_settings*

Tabel *tax_settings* berfungsi menyimpan dan mengelola data terkait *tax settings*. Tabel ini memiliki *Primary Key* id dan *Foreign Key* instansi_id. Rincian utamanya disimpan pada *field ptkp_base, ptkp_married, ptkp_dependent, max_dependents,* serta *field* biaya_jabatan_*rate* yang berisi persentase biaya jabatan (%) sebagai pengurang pajak.

Tabel 2.47 Spesifikasi Tabel *tax_settings*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengaturan pajak
2.	*instansi_id*	*BIGINT UNSIGNED*	-	Instansi pemilik
3.	*ptkp_base*	*DECIMAL*	15,2	PTKP dasar
4.	*ptkp_married*	*DECIMAL*	15,2	Tambahan PTKP menikah
5.	*ptkp_dependent*	*DECIMAL*	15,2	Tambahan PTKP tanggungan

6.	*max_dependents*	*INT*	-	Maksimal tanggungan
7.	*biaya_jabatan_rate*	*DECIMAL*	5,2	Tarif biaya jabatan
8.	*biaya_jabatan_max_yearly*	*DECIMAL*	15,2	Maksimal biaya jabatan tahunan
9.	*bpjs_kesehatan_employee_rate*	*DECIMAL*	5,2	Tarif BPJS Kesehatan karyawan
10.	*bpjs_tk_jht_employee_rate*	*DECIMAL*	5,2	Tarif JHT karyawan
11.	*bpjs_tk_jp_employee_rate*	*DECIMAL*	5,2	Tarif JP karyawan
12.	*deduction_per_absent*	*DECIMAL*	15,2	Potongan per hari tidak hadir
13.	*deduction_per_late_minute*	*DECIMAL*	15,2	Potongan per menit terlambat
14.	*overtime_rate_per_hour*	*DECIMAL*	15,2	Upah lembur per jam
15.	*tax_brackets*	JSON	-	Lapisan tarif pajak
16.	*created_at*	*TIMESTAMP*	-	Waktu dibuat
17.	*updated_at*	*TIMESTAMP*	-	Waktu diperbarui

2.5.44 Spesifikasi Tabel *users*

Tabel users berfungsi sebagai pusat data autentikasi untuk seluruh pengguna sistem. Tabel ini memiliki *Primary Key* id dan *Foreign Key* google_id, system_role_id, instansi_id. Rincian utamanya disimpan pada *field* name, email, phone, avatar, serta *field* password yang berisi kata sandi pengguna yang telah dienkripsi (*bcrypt hash*).

Tabel 2.48 Spesifikasi Tabel *users*

No	Nama *Field*	*Type*	*Length*	Keterangan
1.	*id*	*BIGINT UNSIGNED*	-	ID unik pengguna
2.	*name*	*VARCHAR*	100	Nama lengkap pengguna
3.	*email*	*VARCHAR*	100	Email pengguna

4.	*google_id*	*VARCHAR*	50	ID akun Google
5.	*system_role_id*	*BIGINT UNSIGNED*	-	Peran sistem
6.	*instansi_id*	*BIGINT*	-	Instansi pengguna
7.	*phone*	*VARCHAR*	20	Nomor telepon
8.	*avatar*	*VARCHAR*	255	Foto profil
9.	*email_verified_at*	*TIMESTAMP*	-	Waktu verifikasi email
10.	*password*	*VARCHAR*	255	Password terenkripsi
11.	*is_active*	*TINYINT*	1	Status akun
12.	*remember_token*	*VARCHAR*	100	Token sesi
13.	*deleted_by*	*BIGINT*	-	Pengguna yang menghapus
14.	*deleted_at*	*TIMESTAMP*	-	Waktu penghapusan
15.	*created_at*	*TIMESTAMP*	-	Waktu akun dibuat
16.	*updated_at*	*TIMESTAMP*	-	Waktu akun diperbarui

2.6 *Unified Modeling Language* (UML)

Unified Modeling Language (UML) merupakan bahasa pemodelan yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan suatu sistem. UML membantu dalam memvisualisasikan kebutuhan, proses, *int*eraksi pengguna, serta struktur sistem sebelum rancangan diterapkan ke dalam sistem.

Pada perancangan sistem HagaPlus, UML digunakan untuk menggambarkan proses pengelolaan kepegawaian dan penggajian, mulai dari pencatatan absensi, pengelolaan cuti dan izin, *reimbursement*, pengelolaan komponen penghasilan, hingga proses penggajian. UML juga digunakan untuk menggambarkan *int*eraksi antara pengguna dengan sistem berdasarkan hak akses Superadmin, Admin dan Karyawan, sehingga setiap fungsi sistem dapat dirancang sesuai dengan kebutuhan masing-masing pengguna.

Penggunaan UML dalam perancangan HagaPlus bertujuan untuk memberikan gambaran yang terstruktur mengenai fungsi, alur proses, *int*eraksi antarobjek, dan struktur sistem sebelum dilakukan implementasi. Dengan adanya pemodelan tersebut, hubungan antara kebutuhan sistem dan rancangan yang dibuat dapat lebih

mudah dipahami serta menjadi acuan dalam perancangan database dan antarmuka sistem.

Diagram UML yang digunakan dalam perancangan sistem HagaPlus meliputi *Use Case Diagram, Activity Diagram, Sequence Diagram,* dan *Class Diagram*. *Use Case Diagram* digunakan untuk menggambarkan *int*eraksi aktor dengan fungsi sistem, *Activity Diagram* untuk menggambarkan alur aktivitas, *Sequence Diagram* untuk menggambarkan urutan *int*eraksi antarobjek, sedangkan *Class Diagram* digunakan untuk menggambarkan struktur kelas, atribut, operasi, dan hubungan antar kelas dalam sistem HagaPlus.

2.6.1 *Use Case*

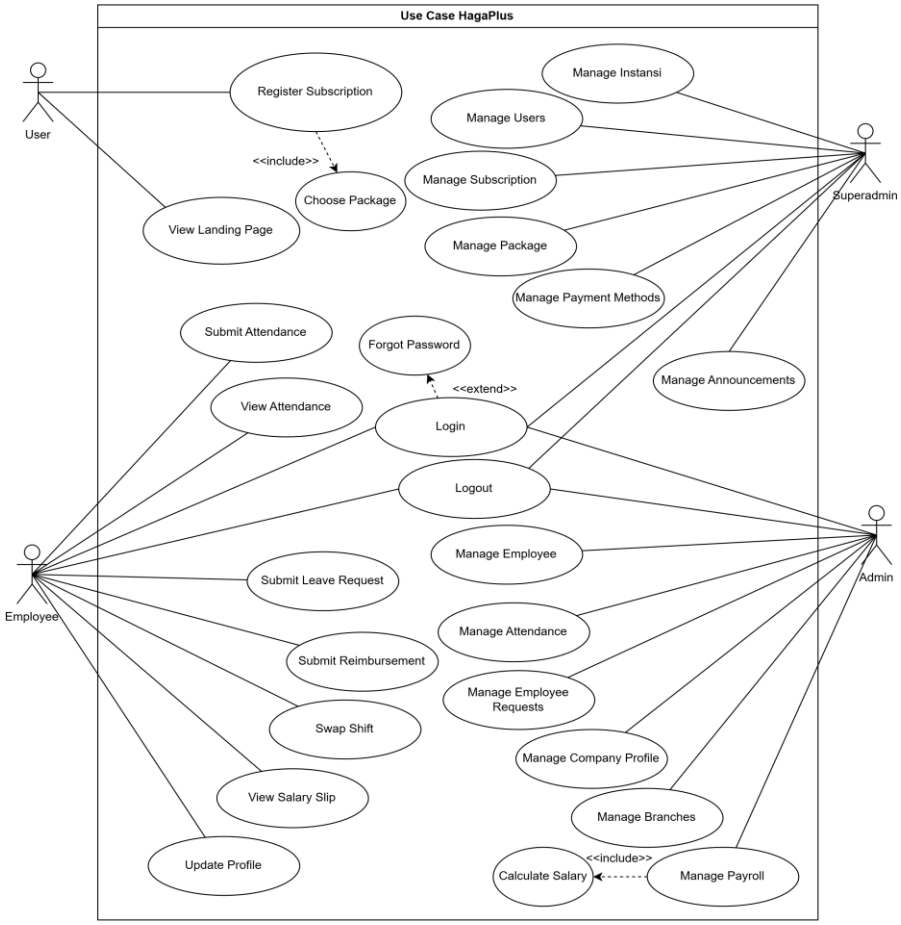
Use Case Diagram digunakan untuk menggambarkan hubungan antara aktor dengan sistem serta fungsi-fungsi yang dapat dilakukan pada sistem HagaPlus. Diagram ini menunjukkan *int*eraksi antara pengguna dengan sistem berdasarkan hak akses dan tanggung jawab masing-masing aktor.

Pada sistem HagaPlus terdapat empat aktor, yaitu *User*, Superadmin, Admin, dan *Employee*. *User* merupakan pengguna yang belum menjadi pengguna aktif sistem dan dapat melakukan proses registrasi *subscription* serta memilih paket layanan yang tersedia. Superadmin merupakan pihak yang mengelola layanan HagaPlus, meliputi pengelolaan instansi, pengguna, *subscription*, paket, metode pembayaran, dan pengumuman.

Admin merupakan pengguna dari perusahaan atau instansi yang menggunakan HagaPlus untuk mengelola data dan proses kepegawaian. Admin memiliki hak akses untuk mengelola data karyawan, absensi, perm*int*aan karyawan, profil perusahaan, cabang, serta proses penggajian dan perhitungan gaji. Sementara itu, *Employee* merupakan karyawan yang menggunakan sistem untuk melakukan absensi, melihat data kehadiran, mengajukan cuti dan *reimbursement*, melakukan pertukaran jadwal kerja, melihat slip gaji, serta memperbarui profil.

Selain hubungan antara aktor dan *use case*, terdapat hubungan <<*include*>> dan <<*extend*>> pada beberapa fungsi. *Register Subscription* memiliki hubungan *include* dengan *Choose Package* karena pemilihan paket merupakan bagian dari proses *registrasi subscription*. Sementara itu, *Forgot Password* memiliki hubungan *extend* dengan *Login* karena proses tersebut merupakan fungsi tambahan yang dapat

dilakukan ketika pengguna mengalami kendala saat melakukan *login*. Pada proses penggajian, *Manage Payroll* memiliki hubungan *include* dengan *Calculate Salary* karena perhitungan gaji merupakan bagian dari proses pengelolaan penggajian.



Gambar 2.4 *Use Case*

2.6.2 *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur aktivitas dan proses yang terjadi dalam sistem HagaPlus berdasarkan fungsi yang telah ditentukan pada *Use Case Diagram*. Diagram ini menunjukkan urutan aktivitas yang dilakukan oleh pengguna maupun sistem, mulai dari proses awal hingga proses selesai. *Activity Diagram* juga digunakan untuk menggambarkan kondisi percabangan (*decision*) serta aktivitas yang dapat dilakukan secara bersamaan (*parallel*).

Pada sistem HagaPlus, *Activity Diagram* digunakan untuk menggambarkan beberapa proses utama, seperti *login*, registrasi *subscription*, absensi, pengajuan cuti dan izin, pengajuan *reimbursement*, pertukaran jadwal kerja, pengelolaan data karyawan, pengelolaan absensi, hingga proses penggajian. Setiap aktivitas disusun

berdasarkan hak akses pengguna, yaitu *Superadmin, Admin*, dan *Employee*. Penggambaran aktivitas tersebut disesuaikan dengan fungsi yang dapat dilakukan oleh masing-masing pengguna sehingga alur proses dapat menunjukkan hubungan antara tindakan pengguna dan respons sistem.

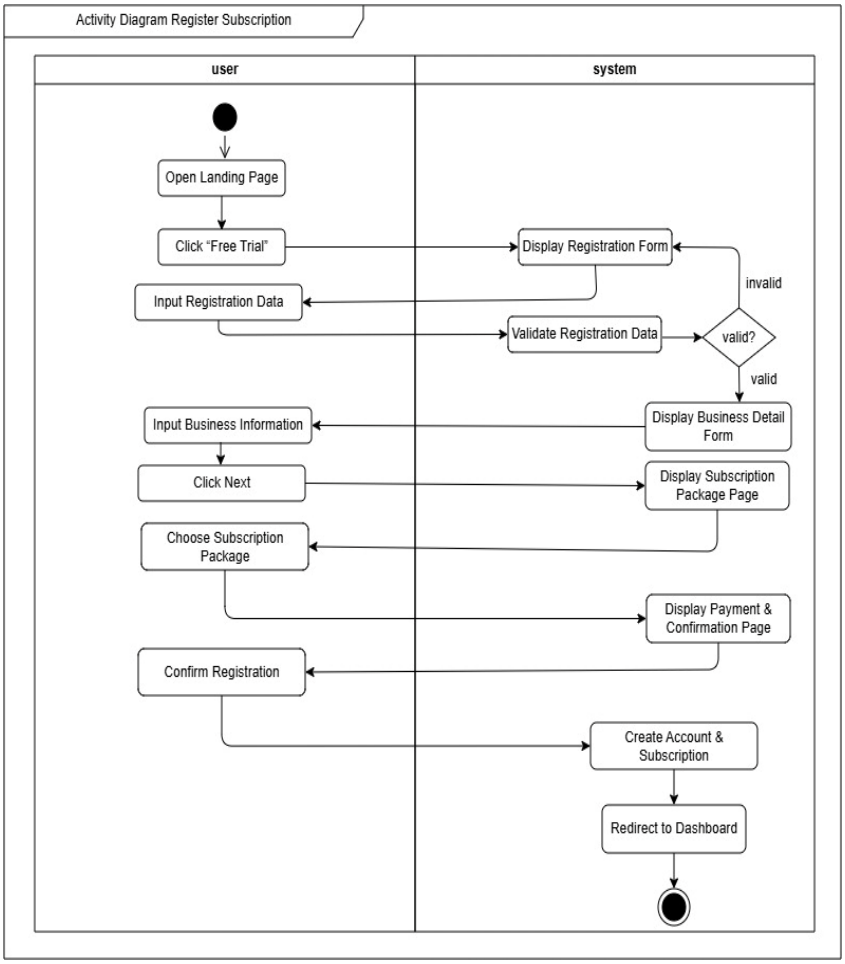
Dalam proses yang memiliki lebih dari satu aktivitas yang dapat berjalan secara bersamaan, digunakan *Fork* untuk membagi alur menjadi beberapa aktivitas paralel dan *Join* untuk menggabungkan kembali aktivitas tersebut sebelum proses dilanjutkan ke tahap berikutnya. Penggunaan *Fork* dan *Join* disesuaikan dengan proses bisnis yang memang memiliki aktivitas yang dapat berlangsung secara paralel sehingga alur diagram tetap menggambarkan proses sistem secara tepat.

Selain itu, *Decision Node* digunakan untuk menggambarkan kondisi yang menghasilkan pilihan alur berdasarkan suatu keputusan. Contohnya, pada proses pengajuan cuti, izin, atau *reimbursement*, sistem dapat menentukan apakah pengajuan diterima atau ditolak berdasarkan hasil *approval* dari Admin. Pada proses absensi, kondisi tertentu juga dapat menentukan apakah data absensi dapat diproses berdasarkan ketentuan yang telah ditetapkan, seperti lokasi dan waktu absensi. Setiap alur aktivitas memiliki *initial node* sebagai titik awal dan *activity final node* sebagai titik akhir.

Perancangan *Activity Diagram* dilakukan dengan mengacu pada kebutuhan fungsional yang telah dianalisis sebelumnya. Setiap diagram menggambarkan proses atau fungsi tertentu sehingga alur yang dihasilkan dapat menjelaskan tahapan proses secara terperinci. Hasil perancangan *Activity Diagram* selanjutnya digunakan sebagai acuan dalam penyusunan *Sequence Diagram*, terutama dalam menentukan urutan *int*eraksi antara pengguna, sistem, dan data yang terlibat dalam setiap proses.

Dengan demikian, *Activity Diagram* HagaPlus memberikan gambaran mengenai urutan aktivitas, percabangan, aktivitas paralel, serta hubungan antara aktivitas pengguna dan sistem. Penggunaan *initial node*, *activity*, *decision node*, *fork*, *join*, dan *activity final node* disesuaikan dengan kebutuhan masing-masing proses agar diagram tetap jelas dan mudah dipahami.

a. ***Activity Diagram Register Subscription***

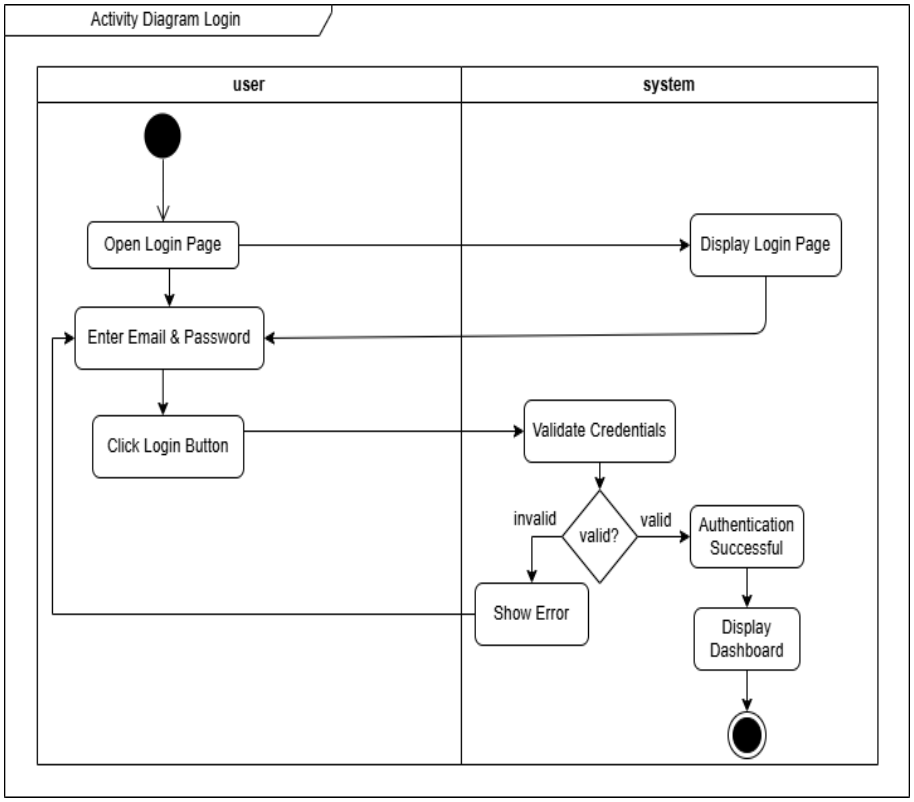


Gambar 2.5 ***Activity Diagram Register Subscription***

Activity Diagram Register Subscription menggambarkan alur aktivitas pengguna dalam melakukan proses *registrasi subscription* pada sistem HagaPlus. Proses diawali dengan pengguna mengakses halaman registrasi dan mengisi data registrasi. Selanjutnya sistem melakukan validasi terhadap data yang telah diinput. Apabila data valid, pengguna melanjutkan dengan mengisi data instansi, memilih paket *subscription*, dan melakukan konfirmasi pendaftaran. Setelah itu, sistem memproses pembuatan akun beserta *subscription*, kemudian mengarahkan pengguna ke *dashboard* sebagai tanda bahwa proses registrasi telah berhasil diselesaikan. Alur ini menunjukkan tahapan registrasi secara berurutan hingga pengguna berhasil terdaftar pada

layanan HagaPlus. Setiap tahapan memiliki proses validasi untuk memastikan data yang digunakan sesuai dengan kebutuhan sistem.

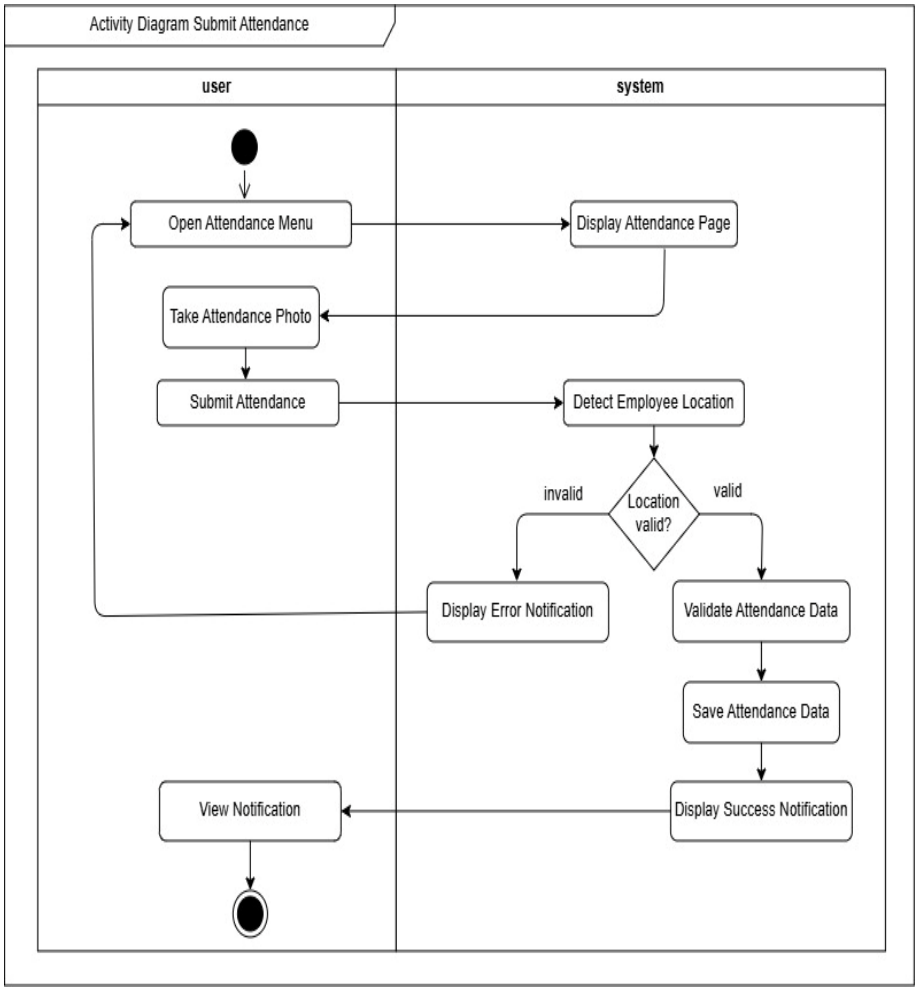
b. *Activity Diagram Login*



Gambar 2.6 *Activity Digram Login*

Activity Diagram Login menggambarkan alur aktivitas pengguna dalam melakukan proses *login* ke dalam sistem HagaPlus menggunakan akun yang telah terdaftar. Proses diawali dengan pengguna membuka halaman *login*, kemudian memasukkan alamat email dan kata sandi, lalu menekan tombol *login*. Selanjutnya sistem melakukan validasi terhadap data yang diinput. Apabila data yang dimasukkan valid, sistem melakukan autentikasi pengguna dan menampilkan *dashboard*. Sebaliknya, apabila data tidak valid, sistem akan menampilkan pesan kesalahan sehingga pengguna dapat melakukan *login* kembali. Alur ini menunjukkan proses autentikasi untuk memastikan hanya pengguna dengan data akun yang valid yang dapat mengakses sistem.

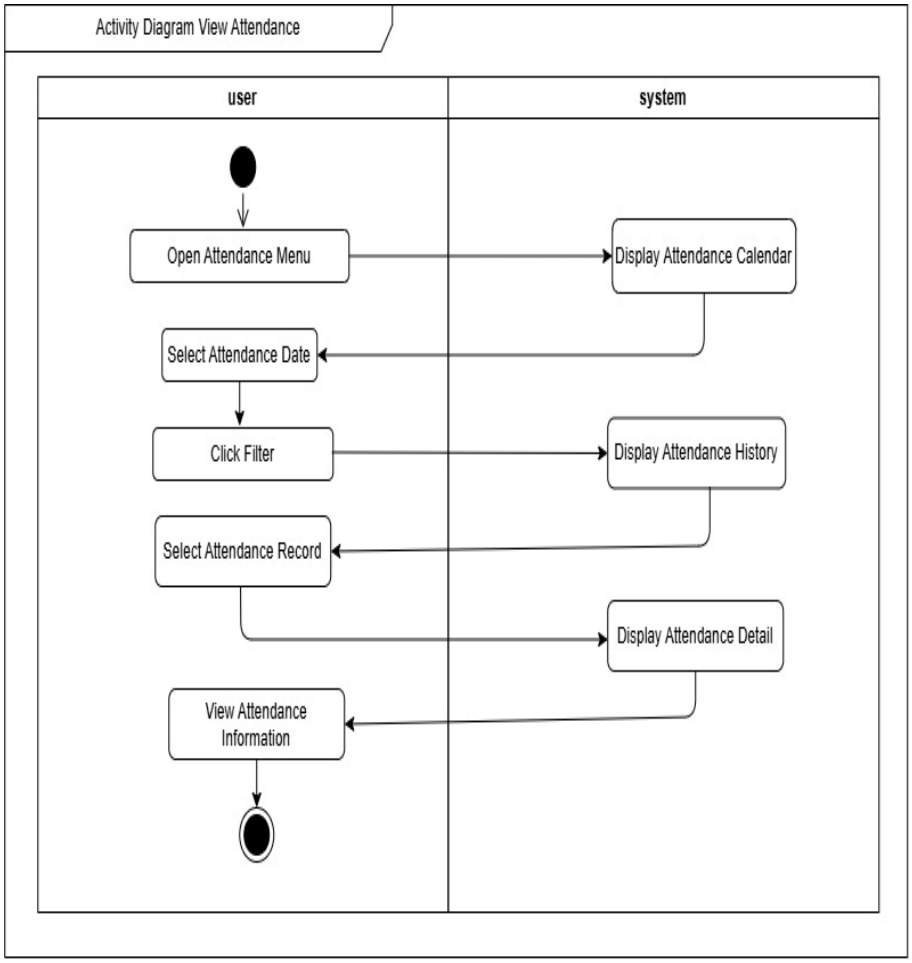
c. ***Activity Diagram Submit Attendance***



Gambar 2.7 ***Activity Diagram Submit Attendance***

Activity Diagram Submit Attendance menggambarkan alur aktivitas karyawan dalam melakukan proses absensi melalui sistem HagaPlus. Proses diawali dengan karyawan membuka menu absensi dan mengambil foto sebagai bukti kehadiran, kemudian mengirimkan data absensi ke sistem. Selanjutnya sistem mendeteksi lokasi karyawan untuk memastikan bahwa absensi dilakukan pada lokasi yang telah ditentukan. Apabila lokasi valid, sistem melakukan validasi data absensi, menyimpan data kehadiran, dan menampilkan notifikasi keberhasilan kepada pengguna. Sebaliknya, apabila lokasi tidak valid, sistem akan menampilkan notifikasi kesalahan sehingga pengguna dapat melakukan proses absensi kembali.

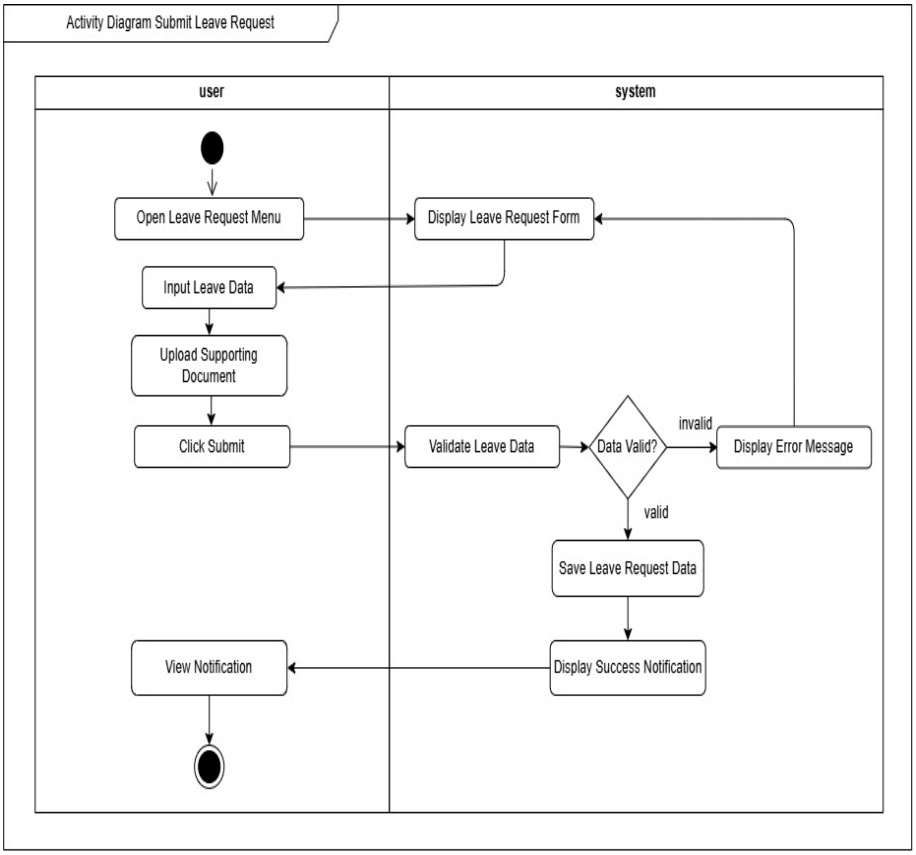
d. ***Activity Diagram View Attendance***



Gambar 2.8 ***Activity Diagram View Attendance***

Activity Diagram View Attendance menggambarkan alur aktivitas karyawan dalam melihat riwayat dan informasi kehadiran pada sistem HagaPlus. Proses diawali dengan karyawan membuka menu absensi, kemudian memilih tanggal absensi yang ingin ditampilkan dan menekan tombol filter. Selanjutnya sistem menampilkan riwayat absensi sesuai dengan tanggal yang dipilih. Karyawan kemudian memilih data absensi yang diinginkan, sehingga sistem menampilkan detail informasi kehadiran yang dapat dilihat oleh karyawan.

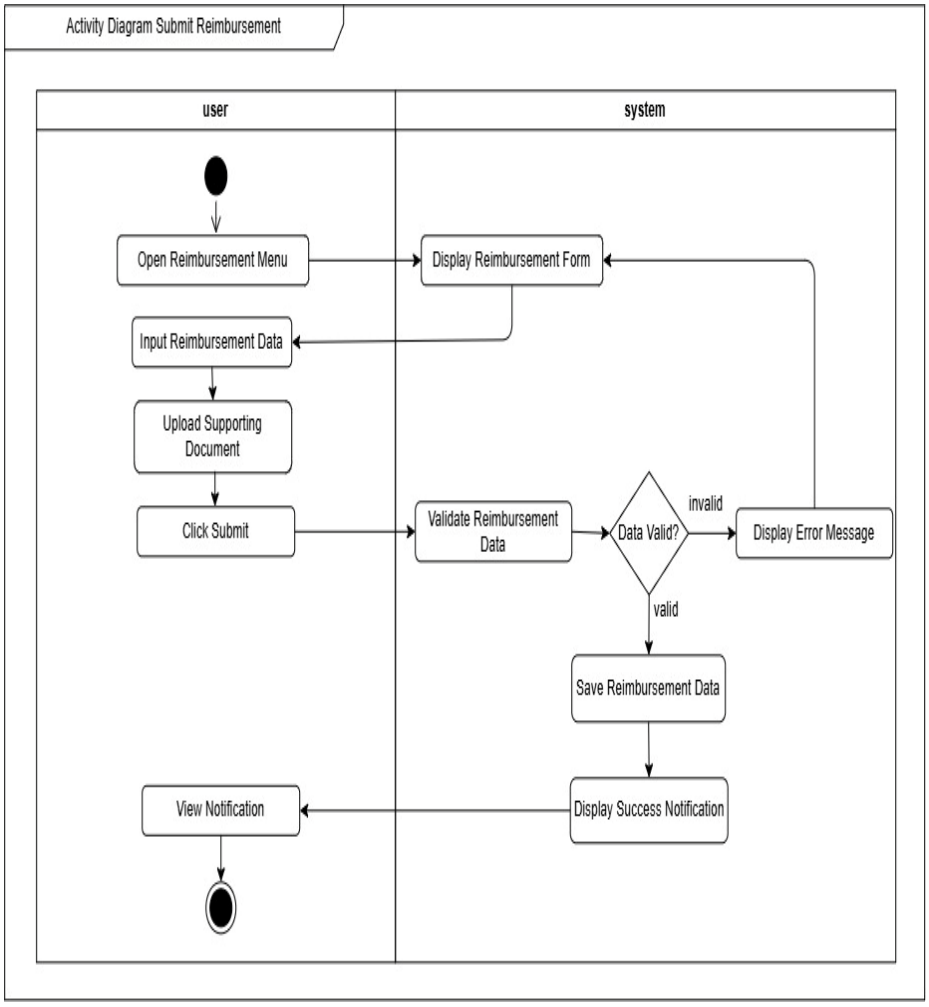
e. ***Activity Diagram Submit Leave Request***



Gambar 2.9 ***Activity Diagram Submit Leave Request***

Activity Diagram Submit Leave Request menggambarkan alur aktivitas karyawan dalam melakukan pengajuan cuti melalui sistem HagaPlus. Proses diawali dengan karyawan membuka menu pengajuan cuti, kemudian mengisi data cuti dan mengunggah dokumen pendukung sebagai persyaratan pengajuan. Setelah pengguna menekan tombol submit, sistem melakukan validasi terhadap data yang diinput. Apabila data valid, sistem menyimpan data pengajuan cuti dan menampilkan notifikasi keberhasilan kepada pengguna. Sebaliknya, apabila data tidak valid, sistem akan menampilkan pesan kesalahan sehingga pengguna dapat memperbaiki data dan mengajukan cuti kembali.

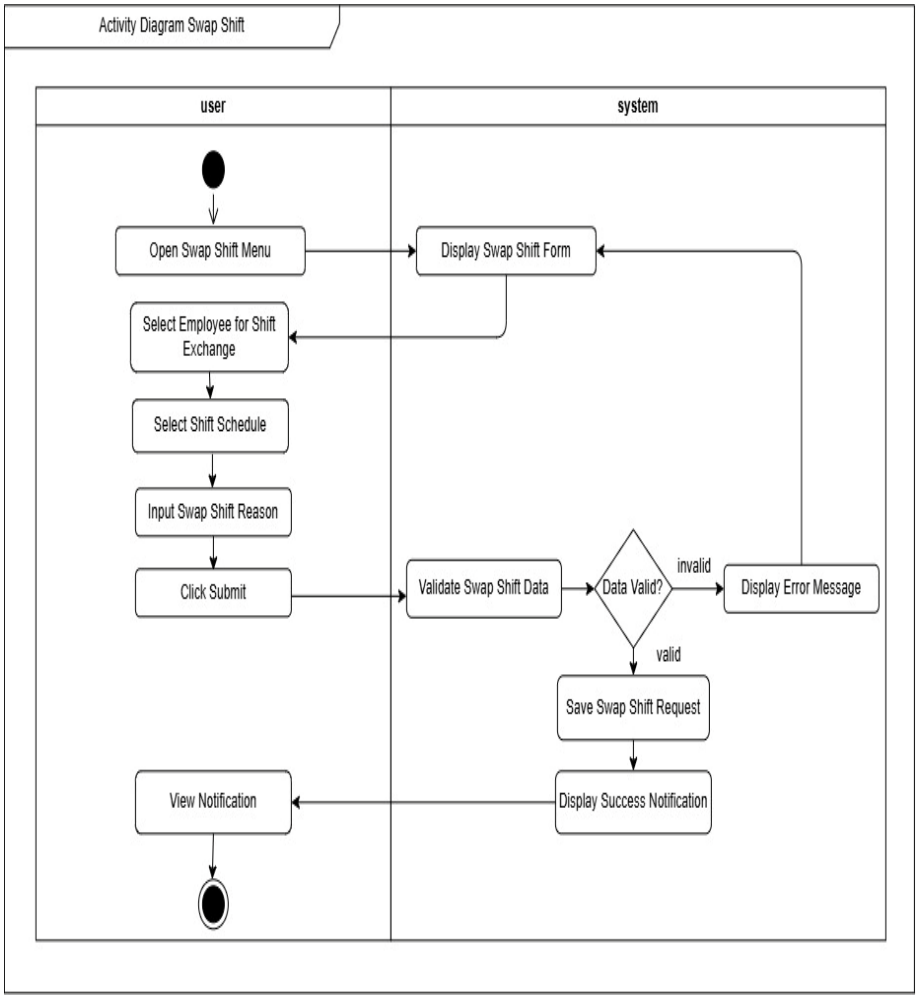
f. *Activity Diagram Submit Reimbursement*



Gambar 2.10 ***Activity Diagram Submit Reimbursement***

Activity Diagram Submit Reimbursement menggambarkan alur aktivitas karyawan dalam melakukan pengajuan *reimbursement* melalui sistem HagaPlus. Proses diawali dengan karyawan membuka menu *reimbursement*, kemudian mengisi data *reimbursement* dan mengunggah dokumen pendukung sebagai bukti pengajuan. Setelah pengguna menekan tombol submit, sistem melakukan validasi terhadap data yang telah diinput. Apabila data valid, sistem menyimpan data *reimbursement* dan menampilkan notifikasi keberhasilan kepada pengguna. Sebaliknya, apabila data tidak valid, sistem akan menampilkan pesan kesalahan sehingga pengguna dapat memperbaiki data dan mengajukan *reimbursement* kembali.

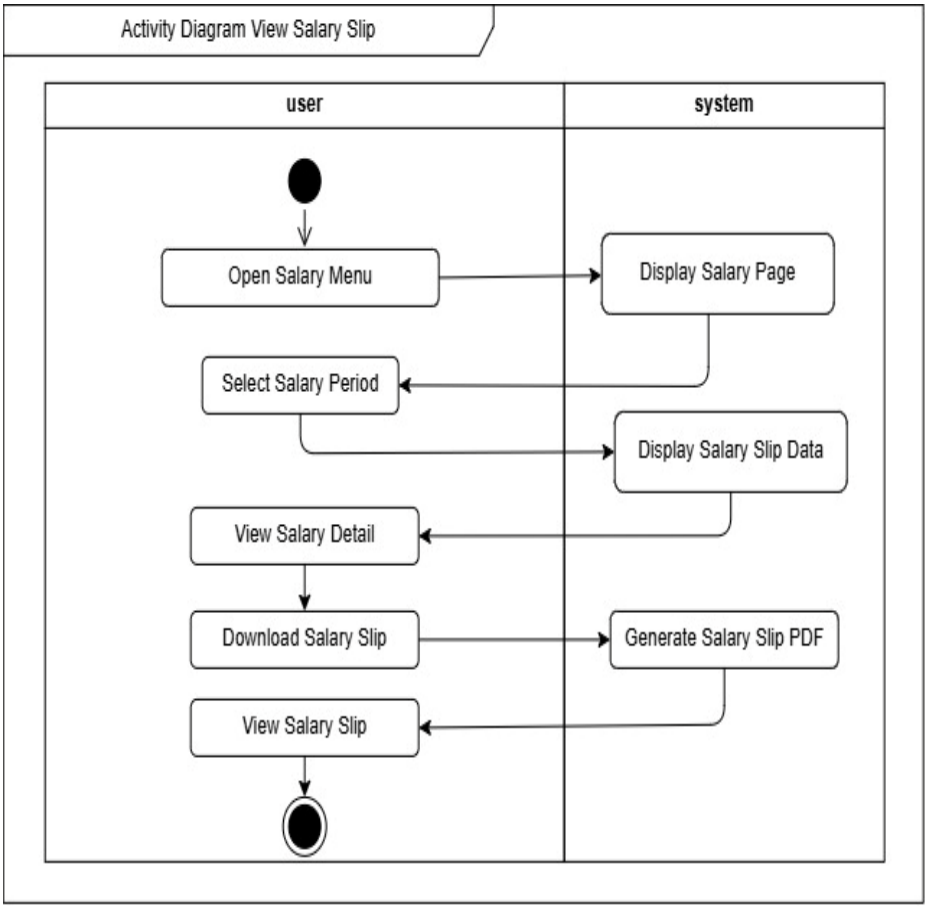
g. *Activity Diagram Swap Shift*



Gambar 2.11 *Activity Diagram Swap Shift*

Activity Diagram Swap Shift menggambarkan alur aktivitas karyawan dalam melakukan pengajuan pertukaran jadwal kerja *(swap shift)* melalui sistem HagaPlus. Proses diawali dengan karyawan membuka menu *swap shift*, kemudian memilih rekan kerja untuk melakukan pertukaran *shift*, memilih jadwal kerja yang akan ditukar, serta mengisi alasan pertukaran *shift*. Setelah pengguna menekan tombol *submit*, sistem melakukan validasi terhadap data yang telah diinput. Apabila data valid, sistem menyimpan data pengajuan pertukaran *shift* dan menampilkan notifikasi keberhasilan kepada pengguna. Sebaliknya, apabila data tidak valid, sistem akan menampilkan pesan kesalahan sehingga pengguna dapat memperbaiki data dan mengajukan pertukaran *shift* kembali.

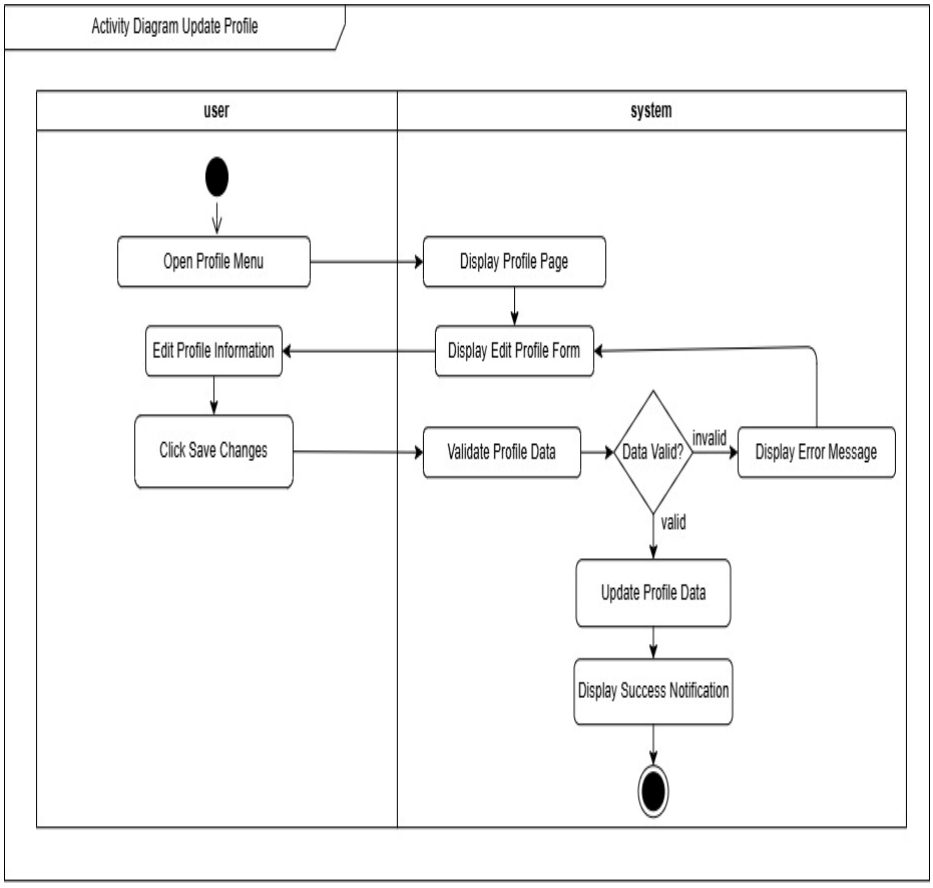
h. ***Activity Diagram View Salary Slip***



Gambar 2.12 ***Activity Diagram View Salary Slip***

Activity Diagram View Salary Slip menggambarkan alur aktivitas karyawan dalam melihat informasi slip gaji pada sistem HagaPlus. Proses diawali dengan karyawan membuka menu gaji dan memilih periode gaji yang ingin ditampilkan. Selanjutnya, sistem menampilkan data slip gaji sesuai dengan periode yang dipilih. Karyawan dapat melihat detail slip gaji dan mengunduhnya dalam format PDF. Setelah proses selesai, sistem menyediakan dokumen slip gaji yang dapat digunakan oleh karyawan. Slip gaji yang ditampilkan mencakup informasi penghasilan dan potongan yang telah dihitung berdasarkan data penggajian. Dengan adanya fitur ini, karyawan dapat mengakses informasi gaji secara mandiri tanpa perlu mem*int*a dokumen secara manual kepada Admin.

i. *Activity Diagram Update Profile*

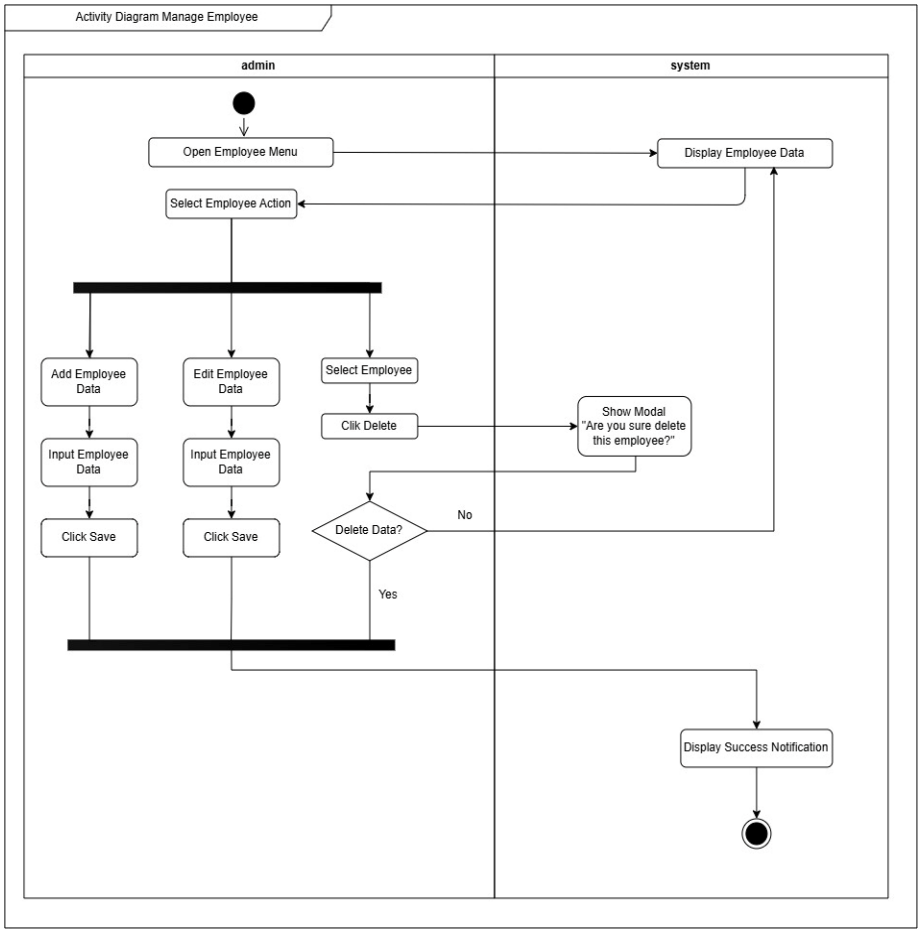


Gambar 2.13 ***Activity Diagram Update Profile Karyawan***

Activity Diagram Update Profile menggambarkan alur aktivitas pengguna dalam memperbarui informasi profil pada sistem HagaPlus. Proses diawali dengan pengguna membuka menu profil, kemudian sistem menampilkan halaman profil beserta formulir pengubahan data. Selanjutnya, pengguna melakukan perubahan informasi profil dan menekan tombol *Save Changes*. Sistem kemudian melakukan validasi terhadap data yang telah diinput. Apabila data valid, sistem memperbarui informasi profil dan menampilkan notifikasi keberhasilan. Sebaliknya, apabila data tidak valid, sistem menampilkan pesan kesalahan sehingga pengguna dapat memperbaiki data dan melakukan penyimpanan kembali. Proses ini memastikan informasi profil yang tersimpan sesuai dengan data yang diberikan oleh pengguna.

Setelah pembaruan berhasil, data profil terbaru dapat langsung digunakan pada sistem.

j. *Activity Diagram Manage Employee*

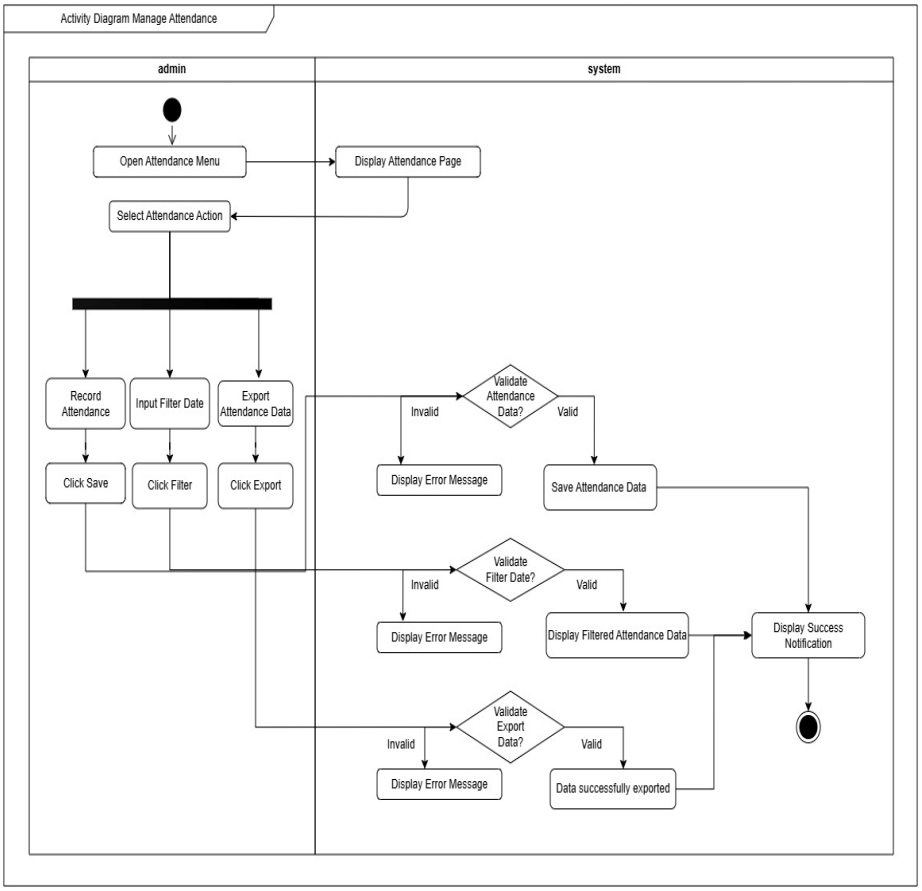


Gambar 2.14 *Activity Diagram Manage Employee*

Activity Diagram Manage Employee menggambarkan alur aktivitas admin dalam mengelola data karyawan pada sistem HagaPlus. Proses diawali dengan admin membuka menu karyawan, kemudian sistem menampilkan daftar data karyawan. Selanjutnya admin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data karyawan. Untuk proses penambahan dan perubahan data, admin mengisi atau memperbarui informasi karyawan, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, admin memilih data karyawan yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data tersebut. Setelah setiap proses

berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada admin.

k. ***Activity Diagram Manage Attendance***

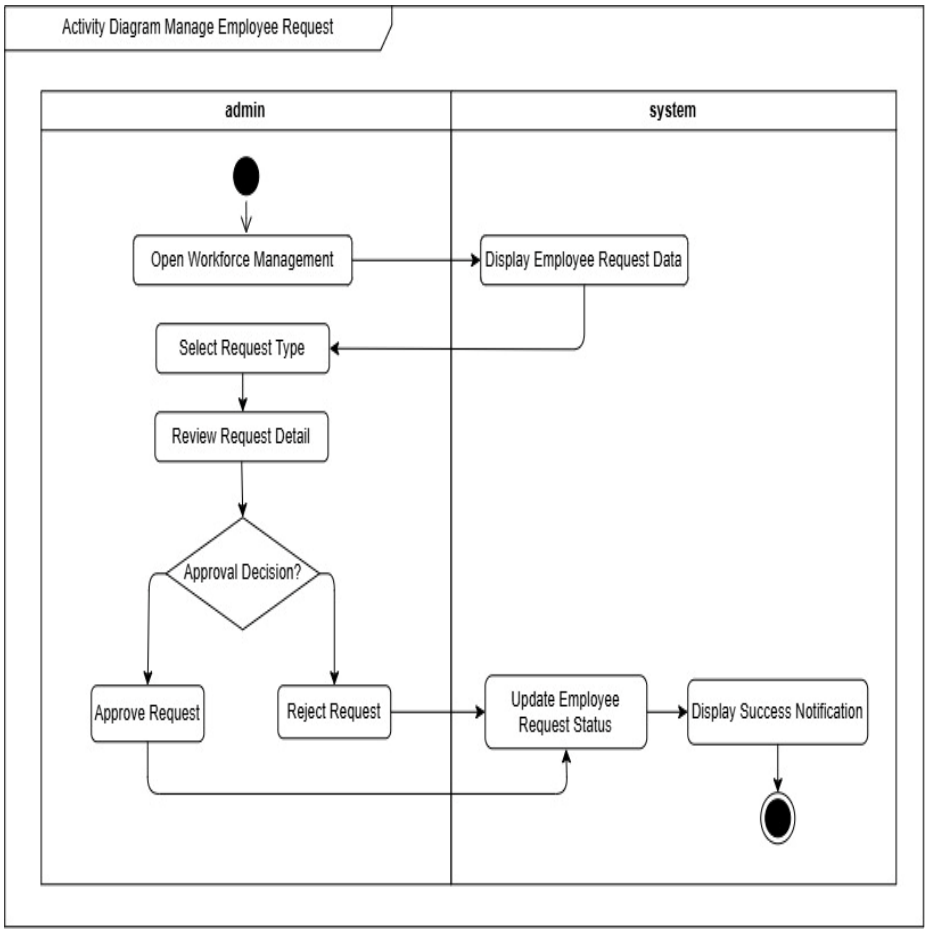


Gambar 2.15 ***Activity Diagram Manage Attendance***

Activity Diagram Manage Attendance menggambarkan alur aktivitas admin dalam mengelola data absensi karyawan pada sistem HagaPlus. Proses diawali dengan admin membuka menu absensi, kemudian sistem menampilkan halaman data absensi. Selanjutnya admin memilih aksi yang akan dilakukan, yaitu mencatat data absensi, memfilter data berdasarkan tanggal, atau mengekspor data absensi. Pada proses pencatatan absensi, admin memasukkan data absensi, kemudian sistem melakukan validasi sebelum menyimpan data. Pada proses penyaringan data, admin menentukan filter tanggal dan sistem menampilkan data absensi yang sesuai dengan kriteria yang dipilih. Sementara itu, pada proses ekspor, sistem menghasilkan file data

absensi yang dapat diunduh oleh admin. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi telah berhasil kepada admin.

l. ***Activity Diagram Manage Employee Requests***

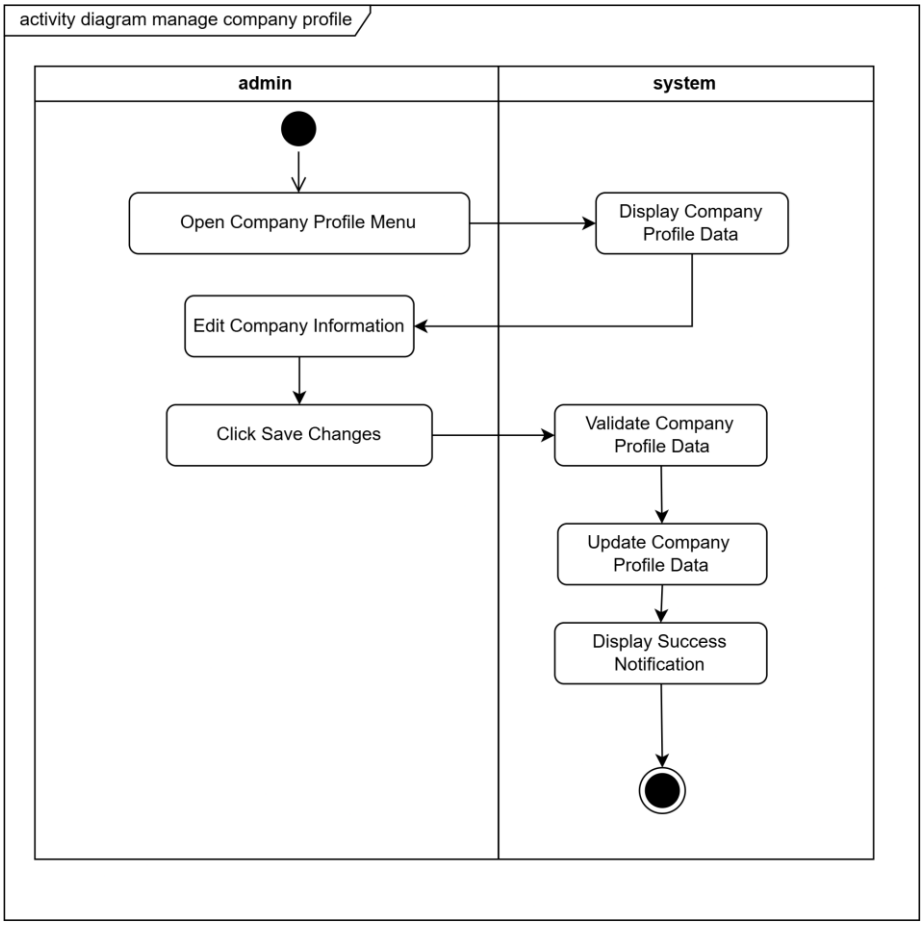


Gambar 2.16 ***Activity Diagram Manage Employee Request***

Diagram Manage Employee Request menggambarkan alur aktivitas Admin dalam mengelola pengajuan yang diajukan oleh karyawan pada sistem HagaPlus, seperti pengajuan cuti, *reimbursement*, dan pertukaran jadwal kerja (*swap shift*). Proses diawali dengan Admin membuka menu *Workforce Management*, kemudian sistem menampilkan daftar pengajuan karyawan. Selanjutnya, Admin memilih jenis pengajuan dan meninjau detail pengajuan yang diajukan. Berdasarkan hasil peninjauan, Admin menentukan keputusan untuk menyetujui atau menolak pengajuan. Setelah keputusan diberikan, sistem memperbarui status pengajuan karyawan sesuai dengan

keputusan Admin dan menampilkan notifikasi keberhasilan. Perubahan status pengajuan menjadi informasi bagi karyawan mengenai hasil pengajuan yang telah diproses. Proses ini membantu Admin mengelola berbagai pengajuan karyawan secara lebih terstruktur dalam satu sistem.

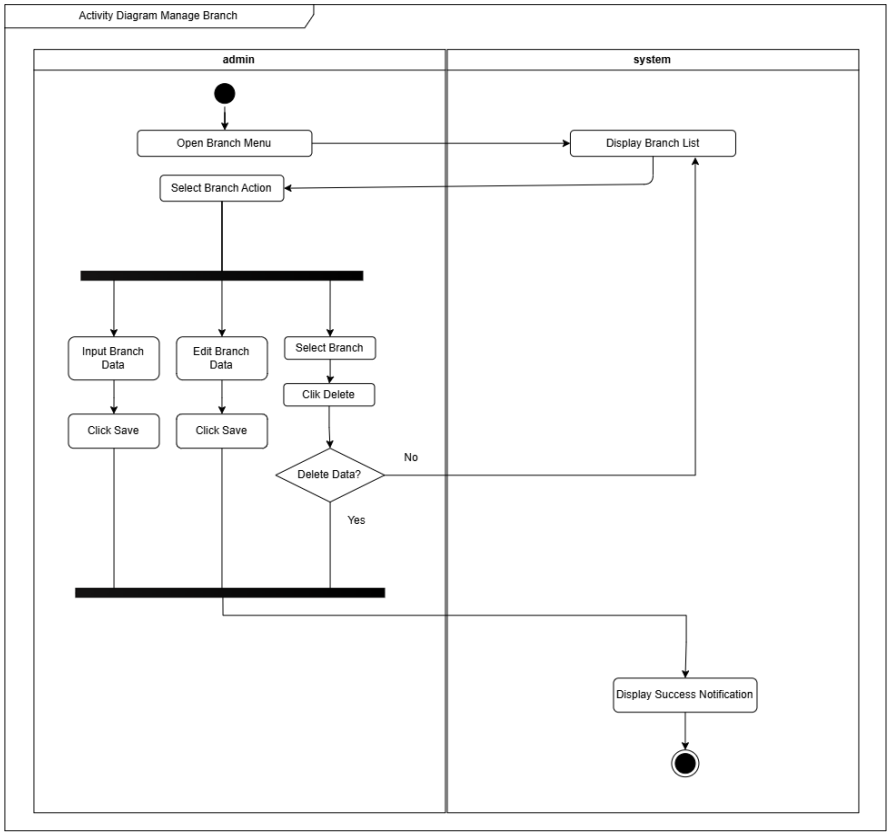
m. ***Activity Diagram Manage Company Profile***



Gambar 2.17 ***Activity Diagram Company Profile***

Activity Diagram Manage Company Profile menggambarkan alur aktivitas admin dalam mengelola informasi profil perusahaan pada sistem HagaPlus. Proses diawali dengan admin membuka menu *Company Profile*, kemudian sistem menampilkan informasi profil perusahaan. Selanjutnya admin melakukan perubahan pada informasi perusahaan dan menekan tombol *Save Changes*. Sistem kemudian melakukan validasi terhadap data yang telah diinput. Apabila data valid, sistem memperbarui informasi profil perusahaan dan menampilkan notifikasi telah berhasil kepada admin.

n. ***Activity Diagram Manage Branches***

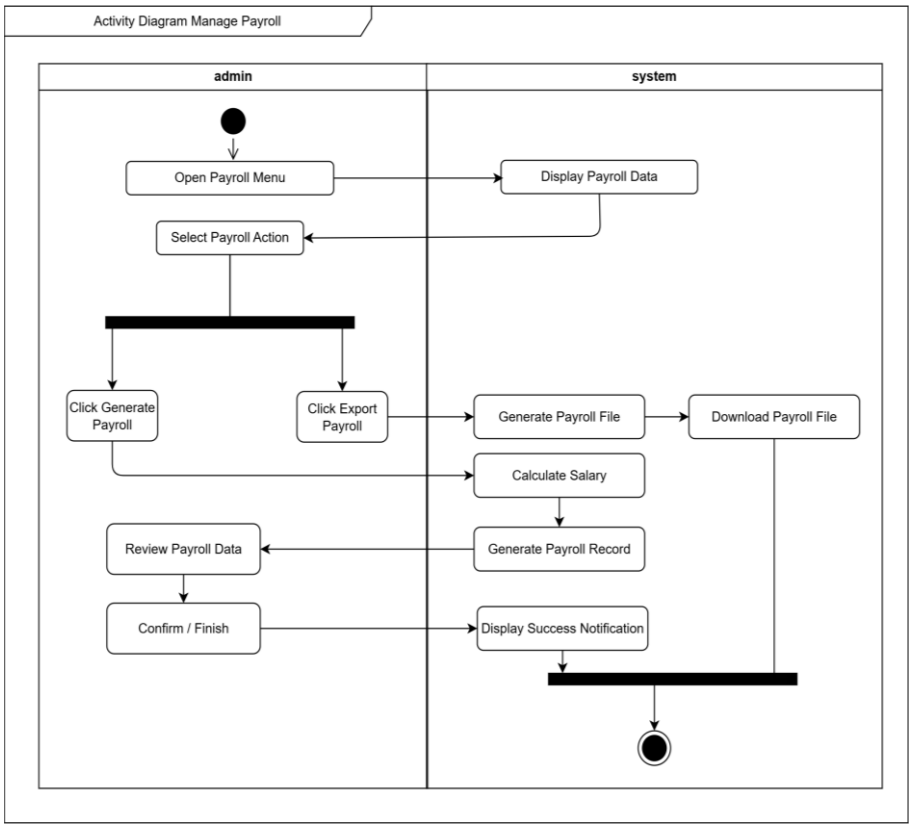


Gambar 2.18 ***Activity Diagram Manage Branches***

Activity Diagram Manage Branch menggambarkan alur aktivitas Admin dalam mengelola data cabang perusahaan pada sistem HagaPlus. Proses diawali dengan Admin membuka menu cabang, kemudian sistem menampilkan daftar cabang yang telah terdaftar. Selanjutnya, Admin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data cabang. Untuk proses penambahan dan perubahan data, Admin mengisi atau memperbarui informasi cabang, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, Admin memilih data cabang yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data cabang tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada Admin. Data cabang yang tersimpan dapat digunakan dalam pengelolaan organisasi dan

mendukung proses absensi berdasarkan lokasi cabang. Dengan adanya fitur ini, Admin dapat mengelola informasi cabang secara terpusat dan terstruktur.

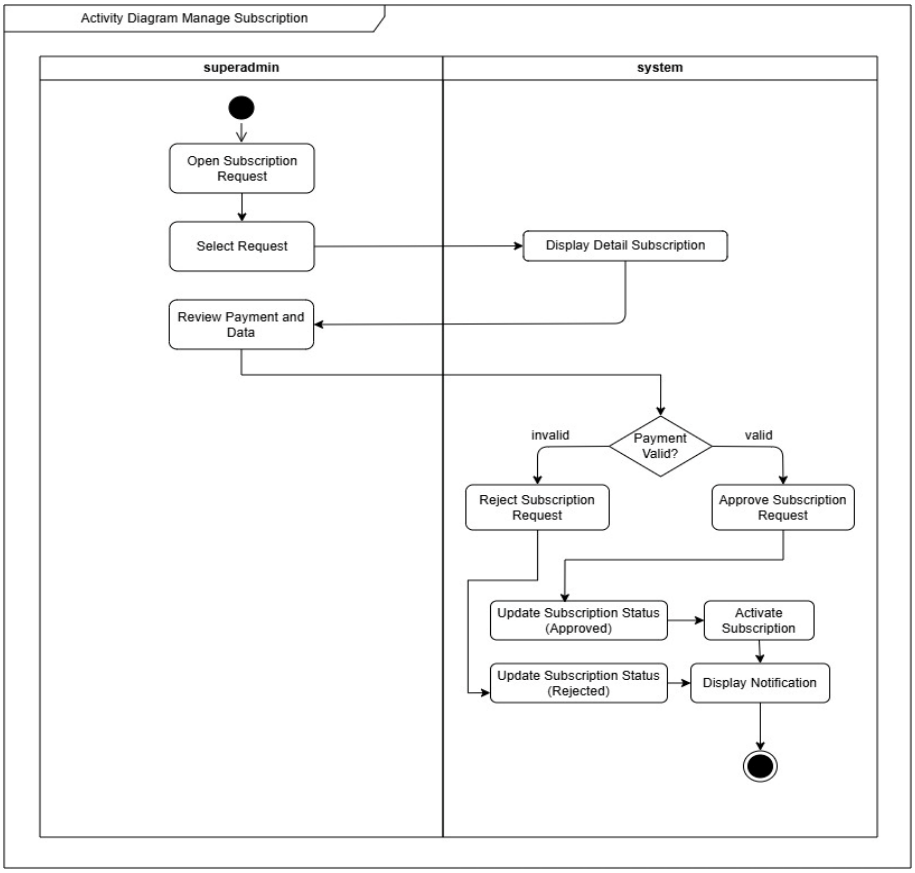
o. ***Activity Diagram Manage Payroll***



Gambar 2.19 ***Activity Diagram Manage Payroll***

Activity Diagram Manage Payroll menggambarkan alur aktivitas admin dalam mengelola data penggajian karyawan pada sistem HagaPlus. Proses diawali dengan admin membuka menu *payroll*, kemudian sistem menampilkan data penggajian. Selanjutnya admin memilih aksi yang akan dilakukan, yaitu menghasilkan *(generate)* data *payroll* atau mengekspor data payroll. Pada proses *generate*, sistem melakukan perhitungan gaji berdasarkan data yang tersedia, kemudian menghasilkan data *payroll* untuk ditinjau oleh admin. Setelah admin melakukan peninjauan dan konfirmasi, sistem menampilkan notifikasi telah berhasil. Sementara itu, pada proses ekspor, sistem menghasilkan file *payroll* yang dapat diunduh oleh admin.

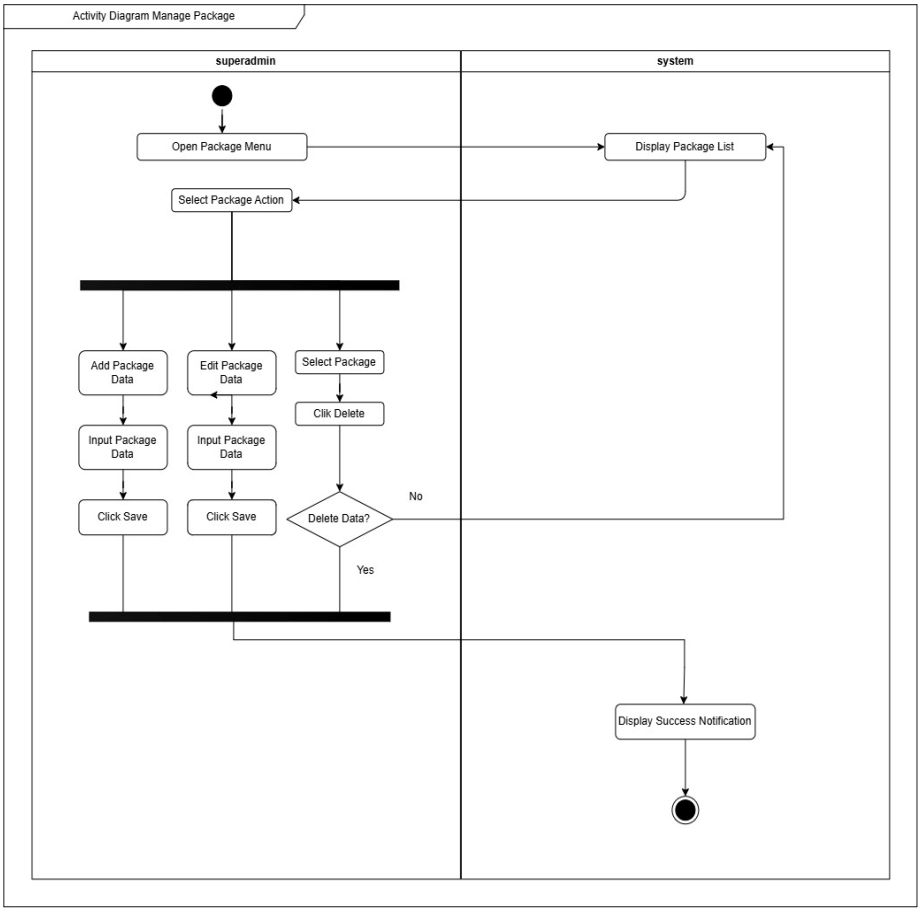
p. ***Activity Diagram Manage Subscription***



Gambar 2.20 ***Activity Diagram Manage Subscription***

Activity Diagram Manage Subscription menggambarkan alur aktivitas Superadmin dalam mengelola pengajuan *subscription* pada sistem HagaPlus. Proses diawali dengan Superadmin membuka menu *subscription*, kemudian memilih pengajuan *subscription* yang akan diproses. Selanjutnya, sistem menampilkan detail *subscription* beserta data pembayaran untuk ditinjau. Berdasarkan hasil peninjauan, Superadmin menentukan keputusan untuk menyetujui atau menolak pengajuan *subscription*. Apabila pengajuan disetujui, sistem memperbarui status *subscription*, mengaktifkan layanan, dan menampilkan notifikasi kepada Superadmin. Sebaliknya, apabila pengajuan ditolak, sistem memperbarui status *subscription* menjadi ditolak dan menampilkan notifikasi. Proses ini membantu Superadmin memastikan setiap pengajuan dan pembayaran *subscription* telah diperiksa sebelum layanan diaktifkan.

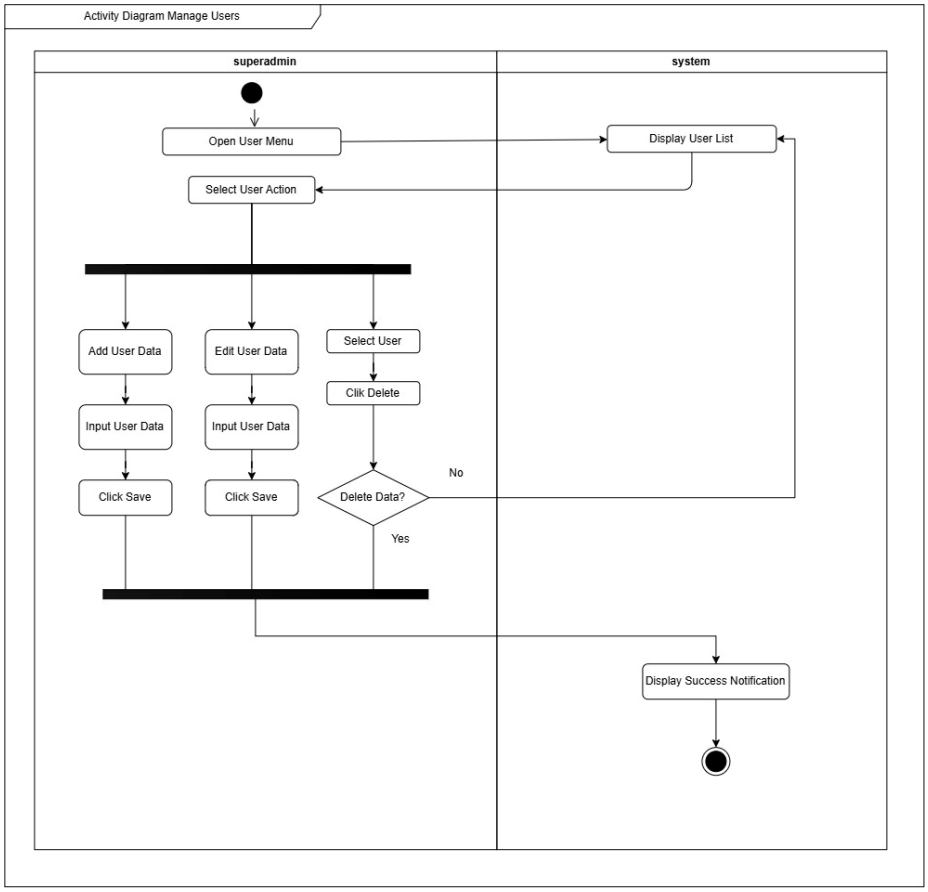
q. ***Activity Diagram Manage Package***



Gambar 2.21 ***Activity Diagram Manage Package***

Activity Diagram Manage Package menggambarkan alur aktivitas superadmin dalam mengelola data paket langganan pada sistem HagaPlus. Proses diawali dengan superadmin membuka menu paket, kemudian sistem menampilkan daftar paket langganan yang tersedia. Selanjutnya superadmin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data paket. Untuk proses penambahan dan perubahan data, superadmin mengisi atau memperbarui informasi paket, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, superadmin memilih paket yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data paket tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi telah berhasil kepada superadmin.

r. *Activity Diagram Manage Users*

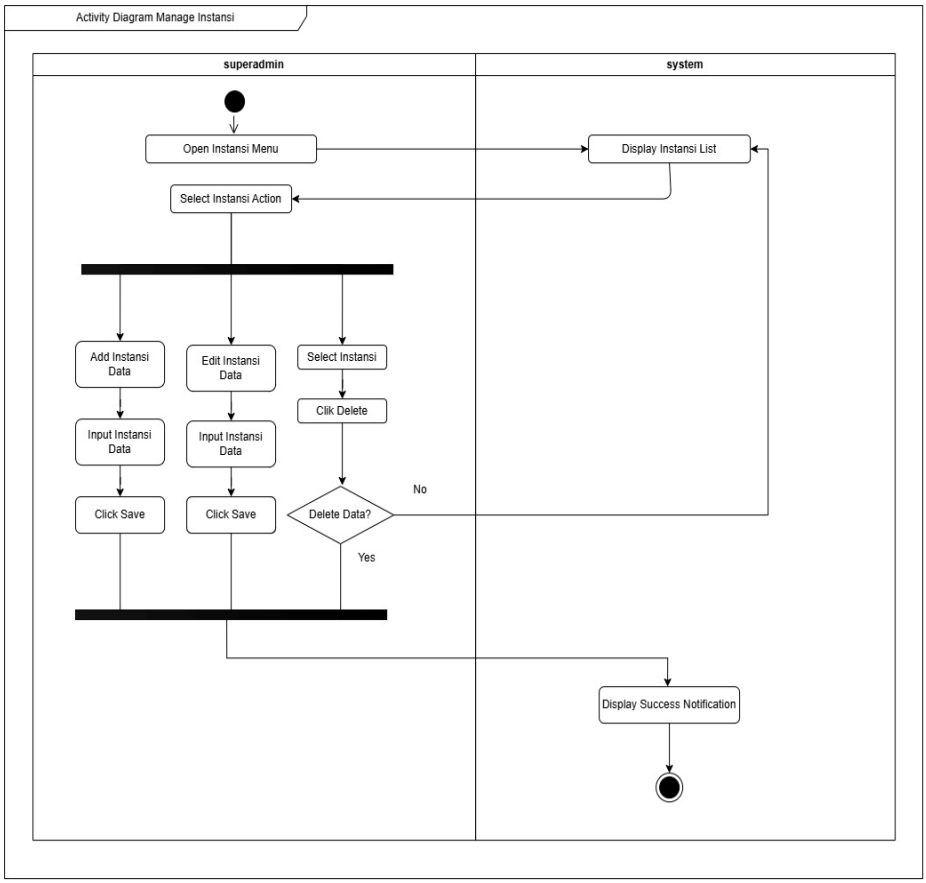


Gambar 2.22 ***Activity Diagram Manage Users***

Activity Diagram Manage Users menggambarkan alur aktivitas Superadmin dalam mengelola data pengguna pada sistem HagaPlus. Proses diawali dengan Superadmin membuka menu *User*, kemudian sistem menampilkan daftar pengguna yang telah terdaftar. Selanjutnya, Superadmin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data pengguna. Untuk proses penambahan dan perubahan data, Superadmin mengisi atau memperbarui informasi pengguna, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, Superadmin memilih pengguna yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data pengguna tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada Superadmin. Pengelolaan data pengguna

dilakukan untuk memastikan informasi akun yang digunakan dalam sistem tetap terdata dan sesuai dengan kebutuhan pengelolaan HagaPlus.

s. ***Activity Diagram Manage Instansi***

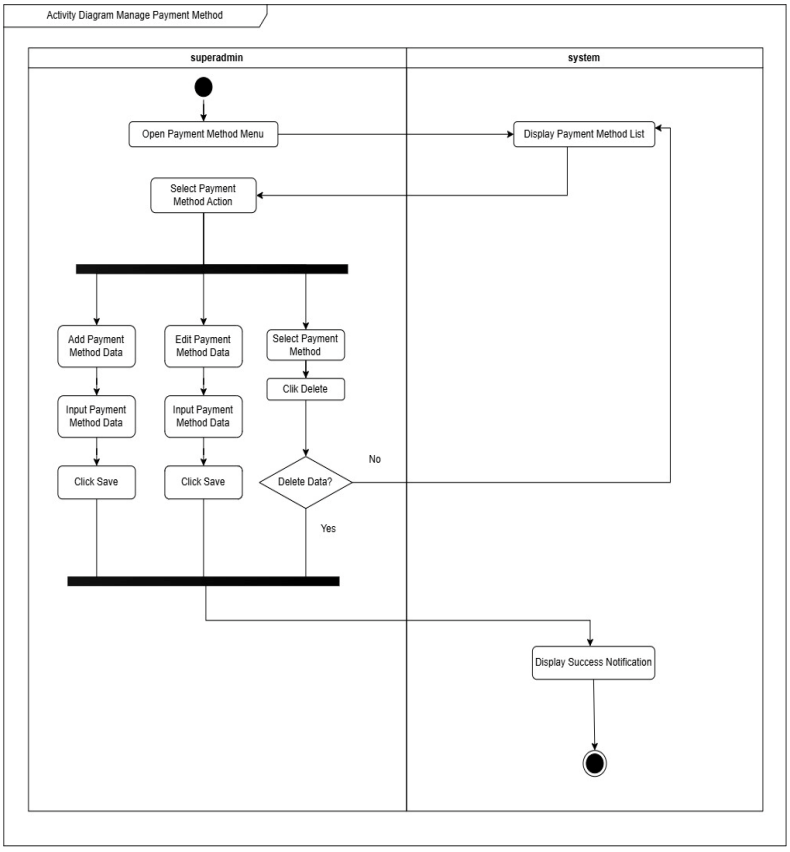


Gambar 2.23 ***Activity Diagram Manage Instansi***

Activity Diagram Manage Instansi menggambarkan alur aktivitas superadmin dalam mengelola data instansi pada sistem HagaPlus. Proses diawali dengan superadmin membuka menu Instansi, kemudian sistem menampilkan daftar instansi yang telah terdaftar. Selanjutnya superadmin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data instansi. Untuk proses penambahan dan perubahan data, superadmin mengisi atau memperbarui informasi instansi, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, superadmin memilih instansi yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data

instansi tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada superadmin.

t. *Activity Diagram Manage Payment Methods*

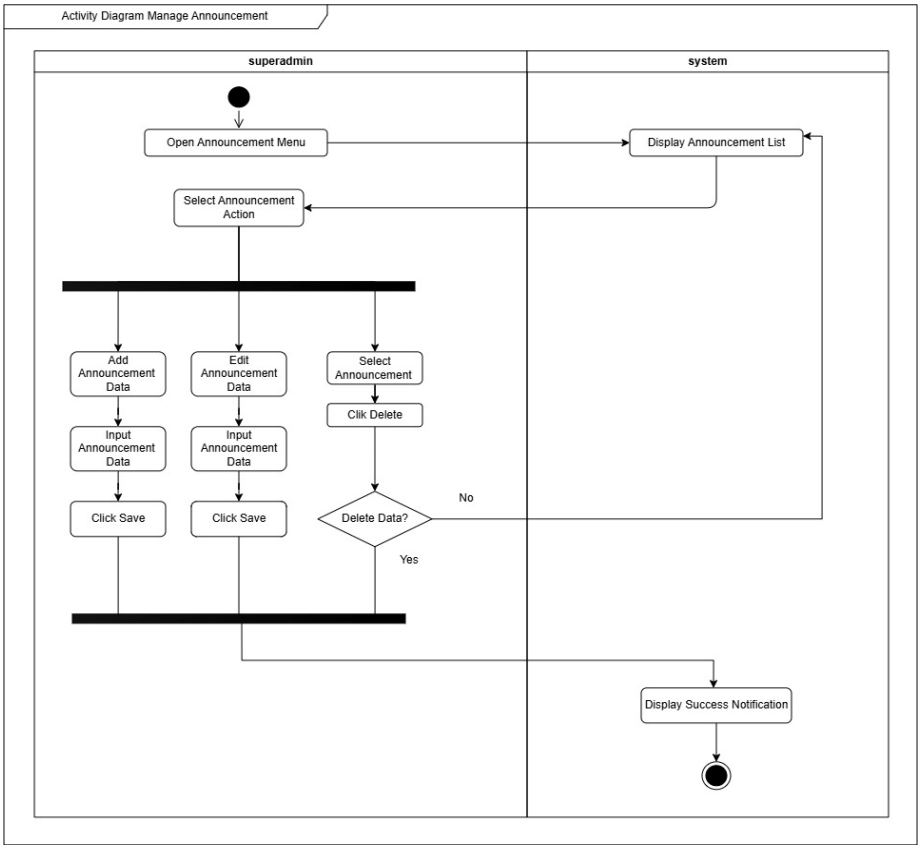


Gambar 2.24 *Activity Diagram Manage Payment Methods*

Activity Diagram Manage Payment Method menggambarkan alur aktivitas Superadmin dalam mengelola data metode pembayaran pada sistem HagaPlus. Proses diawali dengan Superadmin membuka menu *Payment Method*, kemudian sistem menampilkan daftar metode pembayaran yang tersedia. Selanjutnya, Superadmin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data metode pembayaran. Untuk proses penambahan dan perubahan data, Superadmin mengisi atau memperbarui informasi metode pembayaran, kemudian sistem melakukan validasi sebelum menyimpan atau memperbarui data. Sementara itu, untuk proses penghapusan, Superadmin memilih metode pembayaran yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data metode

pembayaran tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada Superadmin. Pengelolaan metode pembayaran dilakukan untuk memastikan pilihan pembayaran *subscription* yang tersedia dapat dikelola sesuai kebutuhan layanan HagaPlus.

u. *Activity Diagram Manage Announcements*



Gambar 2.25 *Activity Diagram Manage Announcement*

Activity Diagram Manage Announcement menggambarkan alur aktivitas superadmin dalam mengelola data pengumuman pada sistem HagaPlus. Proses diawali dengan superadmin membuka menu *Announcement*, kemudian sistem menampilkan daftar pengumuman yang tersedia. Selanjutnya superadmin memilih aksi yang akan dilakukan, yaitu menambah, mengubah, atau menghapus data pengumuman. Untuk proses penambahan dan perubahan data, superadmin mengisi atau memperbarui informasi pengumuman, kemudian sistem melakukan validasi sebelum menyimpan atau

memperbarui data. Sementara itu, untuk proses penghapusan, superadmin memilih pengumuman yang akan dihapus dan mengonfirmasi penghapusan, kemudian sistem menghapus data pengumuman tersebut. Setelah setiap proses berhasil diselesaikan, sistem menampilkan notifikasi keberhasilan kepada superadmin.

2.6.3 *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan urutan *int*eraksi antara aktor, sistem, dan komponen yang terlibat dalam suatu proses berdasarkan urutan waktu. Diagram ini menunjukkan bagaimana suatu perm*int*aan dikirimkan dari pengguna kepada sistem, bagaimana sistem memproses perm*int*aan tersebut, serta bagaimana hasil atau respons dikembalikan kepada pengguna. Dengan demikian, *Sequence Diagram* dapat memberikan gambaran yang lebih rinci mengenai komunikasi yang terjadi pada setiap proses dalam sistem HagaPlus.

Pada perancangan HagaPlus, *Sequence Diagram* digunakan untuk menggambarkan *int*eraksi pada proses-proses utama, seperti *login*, registrasi *subscription*, pengelolaan data karyawan, absensi, pengajuan cuti dan izin, *reimbursement*, pengelolaan *subscription*, hingga proses penggajian. Setiap diagram disusun berdasarkan *Use Case Diagram* dan *Activity Diagram* yang telah dirancang sebelumnya sehingga urutan *int*eraksi yang ditampilkan sesuai dengan kebutuhan dan alur proses sistem.

Komponen yang digunakan dalam *Sequence Diagram* meliputi aktor, *lifeline*, *message*, *activation*, serta *return message*. Aktor menunjukkan pihak yang menjalankan suatu proses, sedangkan *lifeline* menggambarkan objek atau komponen yang terlibat selama proses berlangsung. *Message* digunakan untuk menunjukkan komunikasi atau perm*int*aan yang dikirimkan antarobjek, sedangkan *return message* menunjukkan respons yang diberikan setelah suatu proses selesai dilakukan.

Urutan proses pada *Sequence Diagram* dibaca dari atas ke bawah berdasarkan waktu. Sebagai contoh, pada proses *login*, pengguna memasukkan email dan kata sandi melalui halaman *login*, kemudian sistem mengirimkan data tersebut untuk dilakukan validasi. Sistem selanjutnya memberikan respons berupa

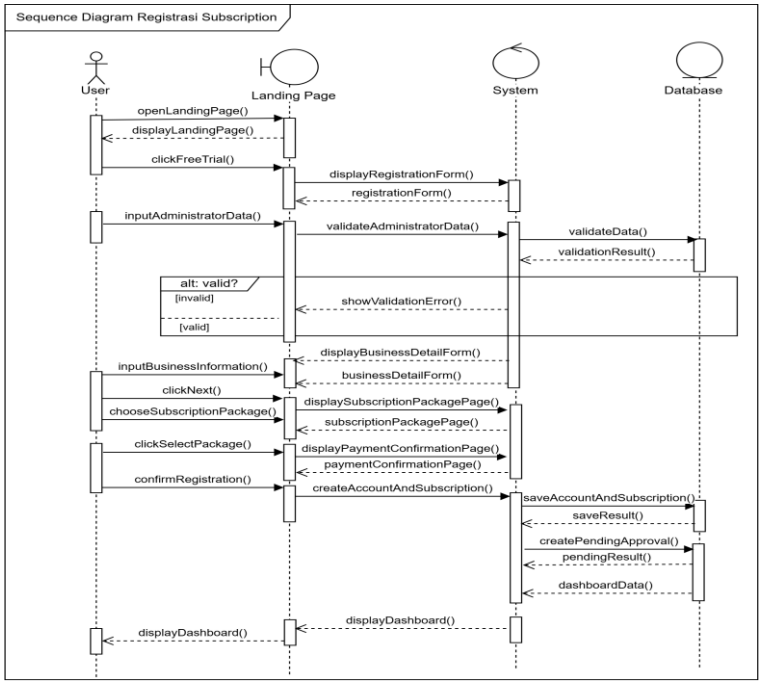
keberhasilan autentikasi dan mengarahkan pengguna ke *dashboard* apabila data valid, atau menampilkan pesan kesalahan apabila data tidak sesuai.

Perancangan *Sequence Diagram* juga memperhatikan proses validasi dan kondisi yang dapat terjadi selama sistem berjalan. Hal tersebut digunakan untuk menunjukkan bagaimana sistem memberikan respons terhadap input pengguna, termasuk ketika data yang diberikan tidak valid atau proses tidak dapat dilanjutkan. Dengan adanya penggambaran tersebut, hubungan antara aktivitas pengguna, proses sistem, dan data yang digunakan dapat dipahami dengan lebih jelas.

Hasil perancangan *Sequence Diagram* selanjutnya menjadi acuan dalam memahami mekanisme *int*eraksi antarbagian sistem sebelum proses implementasi dilakukan. Diagram ini membantu memastikan bahwa setiap fungsi HagaPlus memiliki urutan proses yang jelas serta mendukung keterkaitan antara antarmuka, proses aplikasi, dan *database*.

a. *Sequence Diagram Register Subscription*

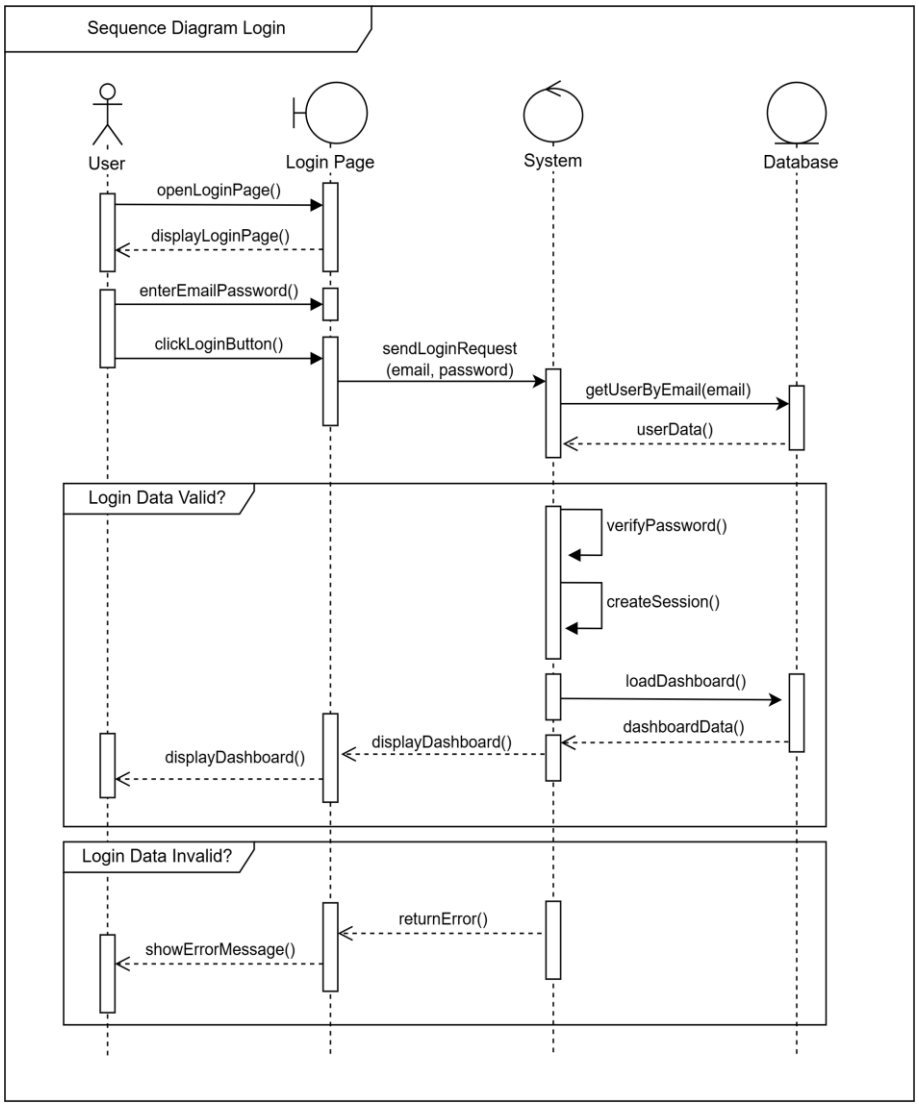
Sequence diagram register subscription menggambarkan *int*eraksi antara pengguna dan sistem dalam proses pendaftaran *subscription*, mulai dari memilih paket hingga data berhasil tersimpan.



Gambar 2.26 *Sequence Diagram Registrasi Subscription*

b. *Sequence Diagram Login*

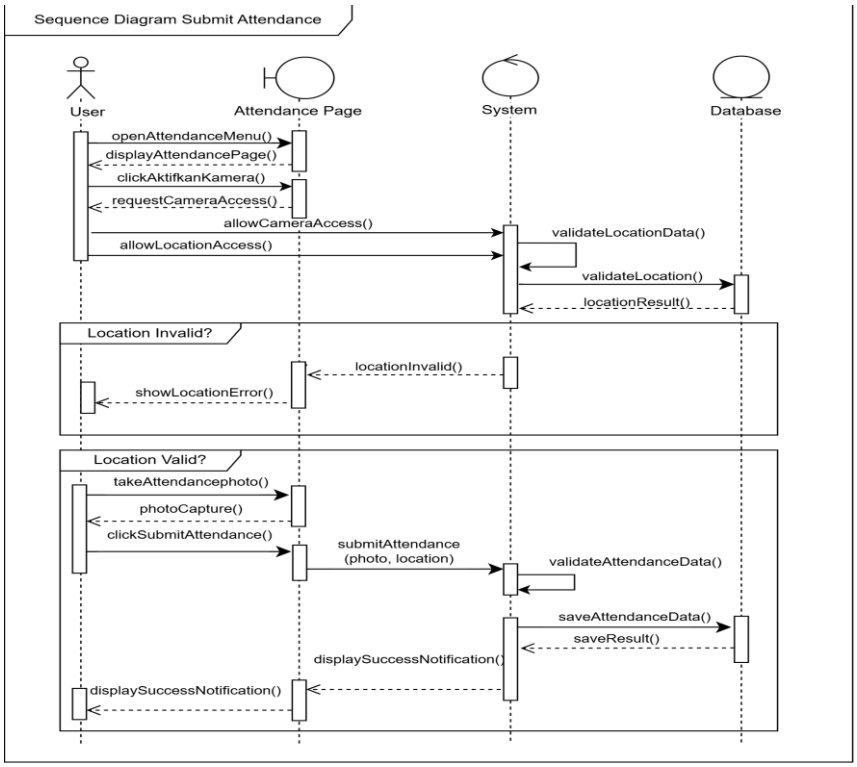
Sequence diagram login menggambarkan alur *int*eraksi pengguna dengan sistem saat proses masuk ke dalam aplikasi menggunakan akun yang telah terdaftar.



Gambar 2.27 *Sequence Diagram Login*

c. *Sequence Diagram Submit Attendance*

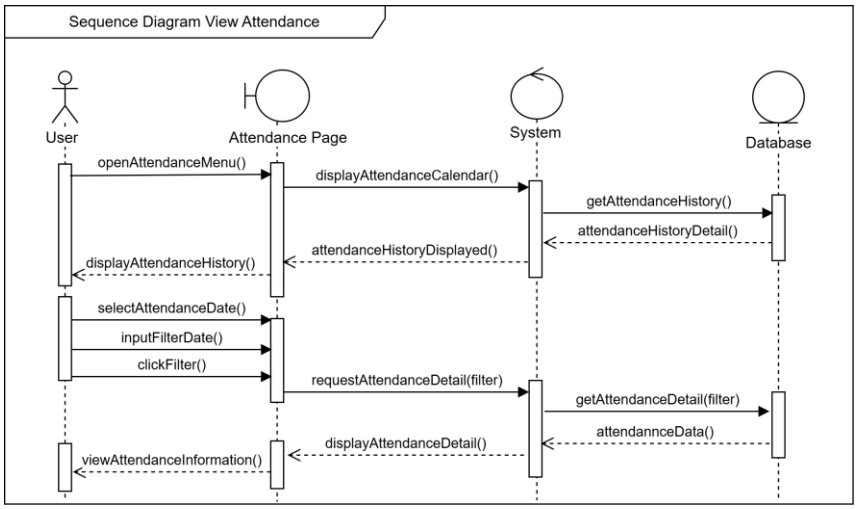
Sequence diagram submit attendance digunakan untuk menggambarkan proses *int*eraksi antara pengguna dan sistem saat melakukan absensi pada HagaPlus. Proses ini meliputi akses halaman absensi, aktivasi kamera dan lokasi, validasi lokasi pengguna, hingga penyimpanan data kehadiran dan menampilkan notifikasi berhasil.



Gambar 2.28 ***Sequence Diagram Submit Attendance***

d. ***Sequence Diagram View Attendance***

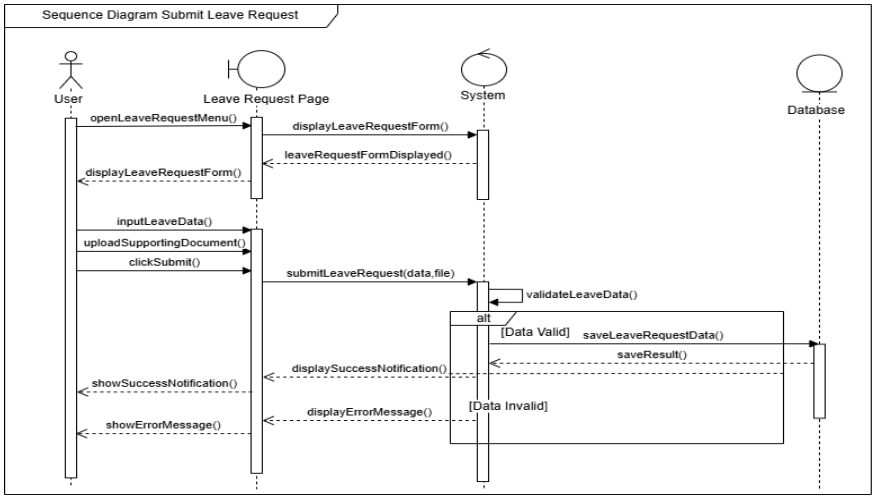
Sequence diagram view attendance digunakan untuk menggambarkan proses pengguna dalam melihat data dan riwayat kehadiran pada sistem HagaPlus. Diagram ini menunjukkan alur *int*eraksi saat sistem menampilkan informasi absensi berdasarkan data yang tersimpan.



Gambar 2.29 ***Sequence Diagram View Attendance***

e. ***Sequence Diagram Submit Leave Request***

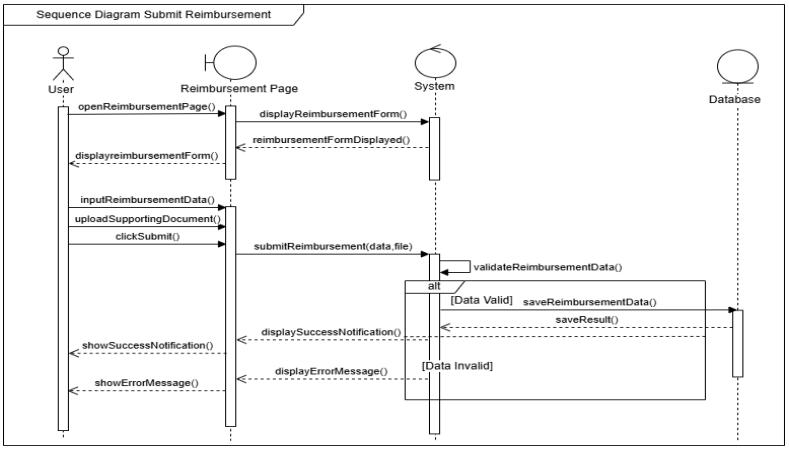
Sequence diagram submit leave request digunakan untuk menggambarkan proses *int*eraksi antara pengguna dan sistem saat melakukan pengajuan cuti pada HagaPlus. Diagram ini menunjukkan alur pengisian data cuti, unggah dokumen pendukung, validasi data, hingga penyimpanan data pengajuan cuti oleh sistem.



Gambar 2.30 ***Sequence Diagram Submit Leave Request***

f. ***Sequence Diagram Submit Reimbursement***

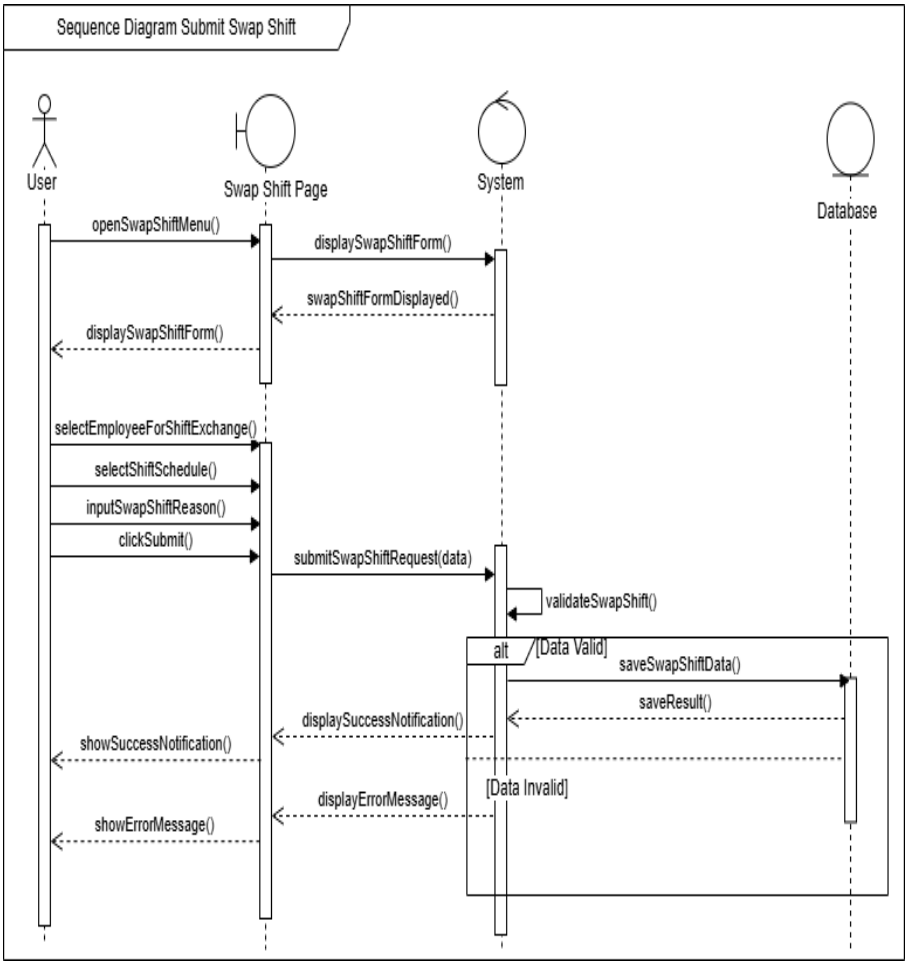
Sequence diagram submit reimbursement digunakan untuk menggambarkan proses pengguna dalam mengajukan *reimbursement* pada sistem HagaPlus. Proses ini meliputi pengisian data *reimbursement*, unggah dokumen pendukung, validasi data, serta penyimpanan data pengajuan oleh sistem.



Gambar 2.31 ***Sequence Diagram Submit Reimbursement***

g. *Sequence Diagram Swap Shift*

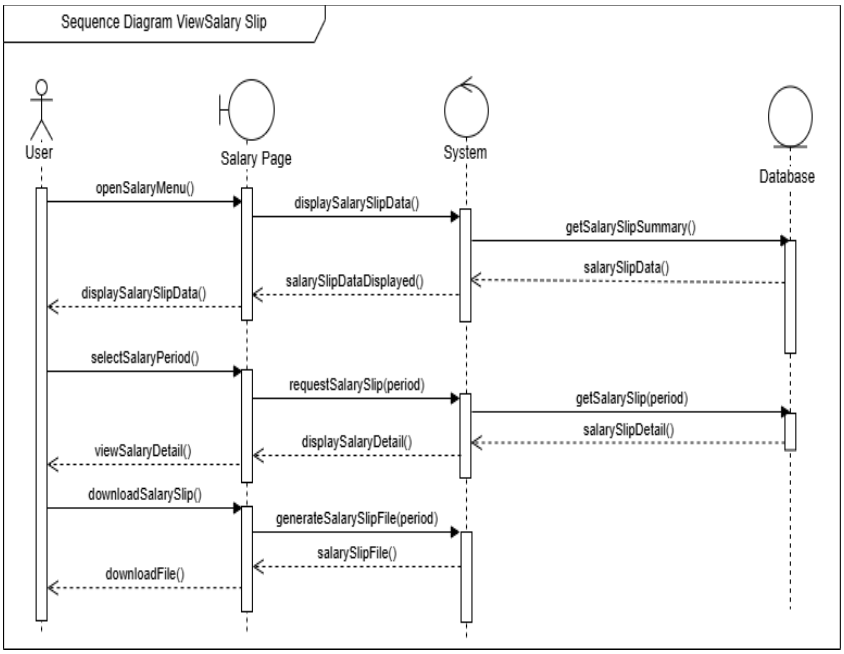
Sequence diagram swap shift digunakan untuk menggambarkan proses pengguna dalam melakukan pengajuan pertukaran jadwal kerja pada sistem HagaPlus. Diagram ini menunjukkan alur pemilihan karyawan, pemilihan jadwal *shift*, pengisian alasan pertukaran *shift*, validasi data oleh sistem.



Gambar 2.32 *Sequence Diagram Swap Shift*

h. *Sequence Diagram View Salary Slip*

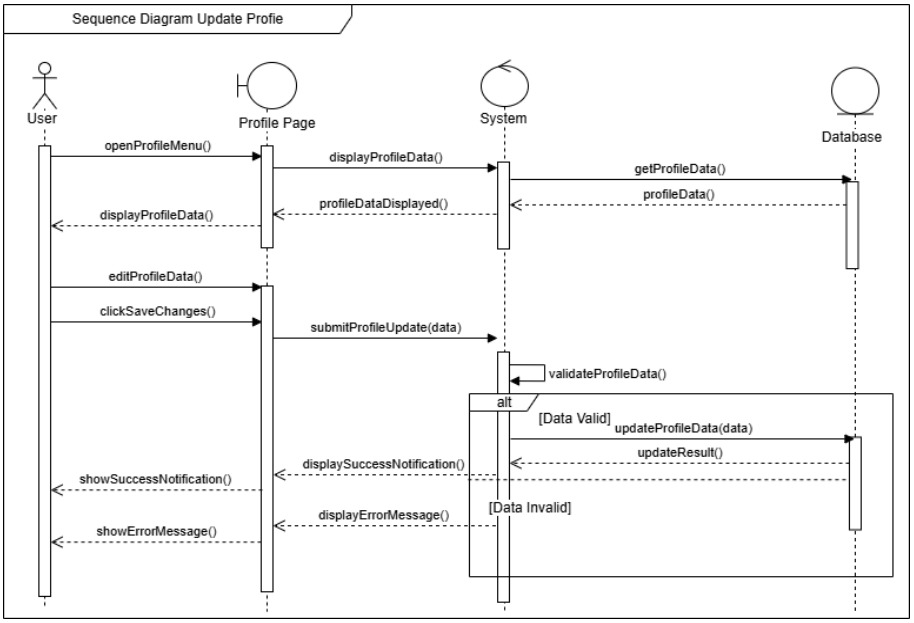
Sequence diagram view salary slip digunakan untuk menggambarkan proses pengguna dalam melihat slip gaji pada sistem HagaPlus. Diagram ini menunjukkan alur saat pengguna membuka menu gaji, memilih periode gaji, melihat detail slip gaji, hingga mengunduh slip gaji dari sistem.



Gambar 2.33 ***Sequence Diagram View Salary Slip***

i. *Sequence Diagram Update Profile*

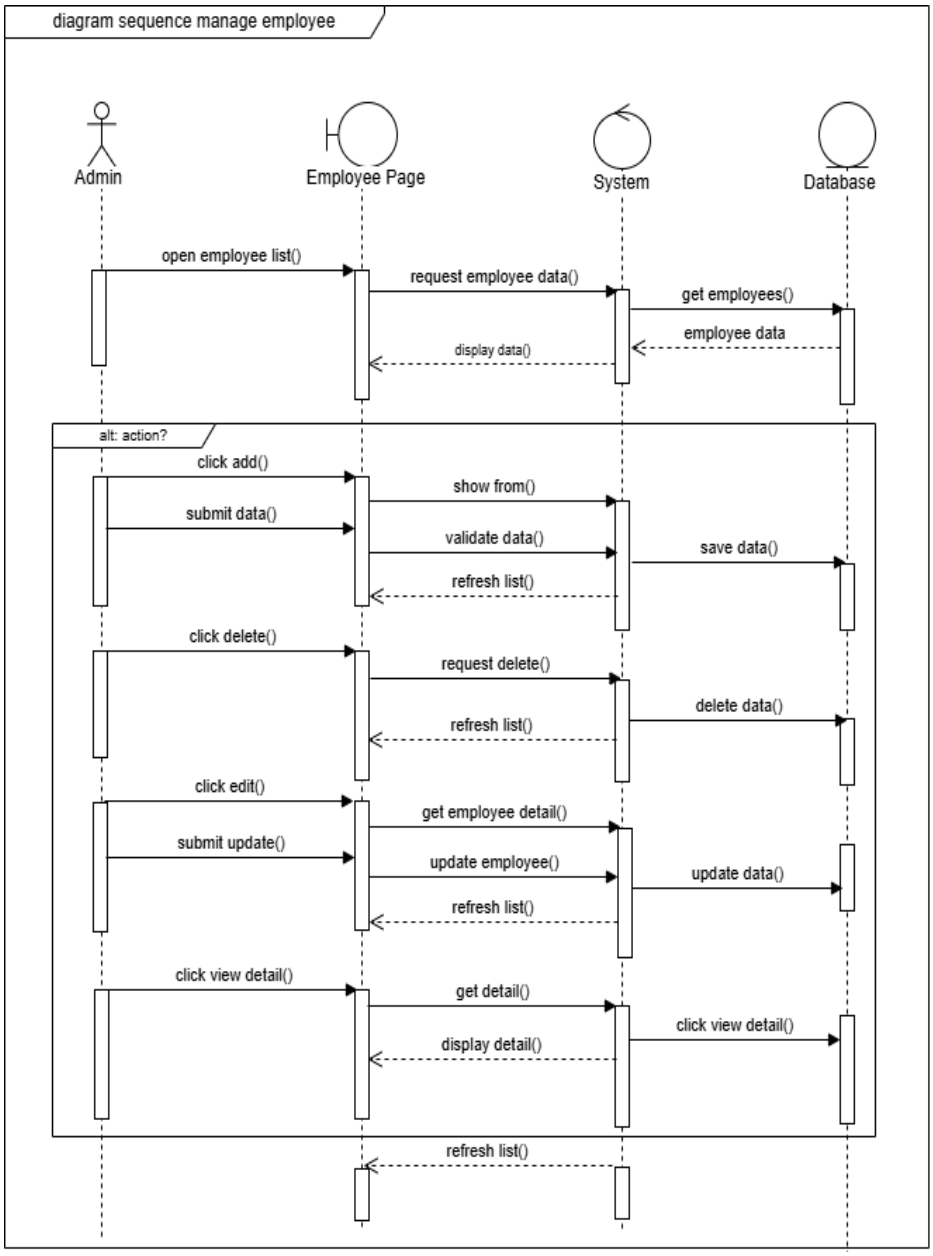
Sequence diagram update profile digunakan untuk menggambarkan proses pengguna dalam memperbarui informasi profil pada sistem HagaPlus. Diagram ini menunjukkan alur mulai dari membuka menu profil, mengubah data profil, validasi data oleh sistem, hingga penyimpanan perubahan data profil.



Gambar 2.34 ***Sequence Diagram Update Profile***

j. *Sequence Diagram Manage Employee*

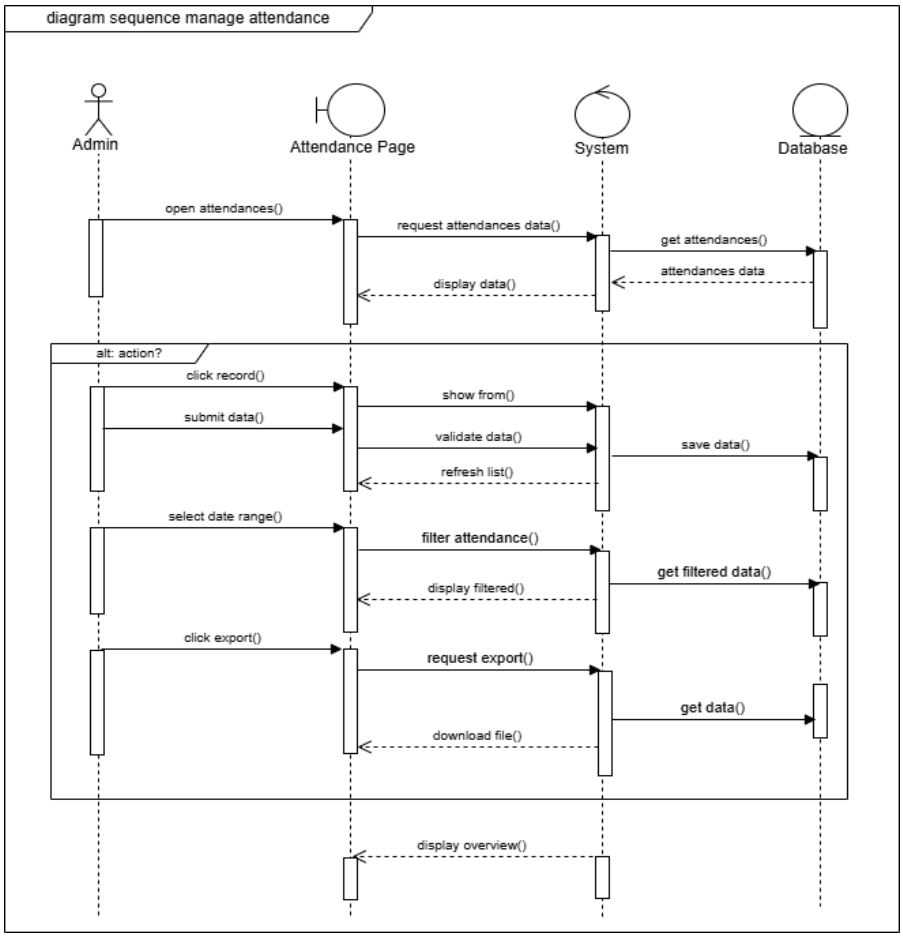
Sequence diagram manage employee digunakan untuk menggambarkan proses *int*eraksi antara admin dan sistem dalam mengelola data karyawan pada HagaPlus. Diagram ini menunjukkan alur pengelolaan data karyawan seperti menambah, mengubah, menghapus, dan menampilkan data karyawan yang dilakukan oleh admin melalui sistem.



Gambar 2.35 *Sequence Diagram Manage Employee*

k. *Sequence Diagram Manage Attendance*

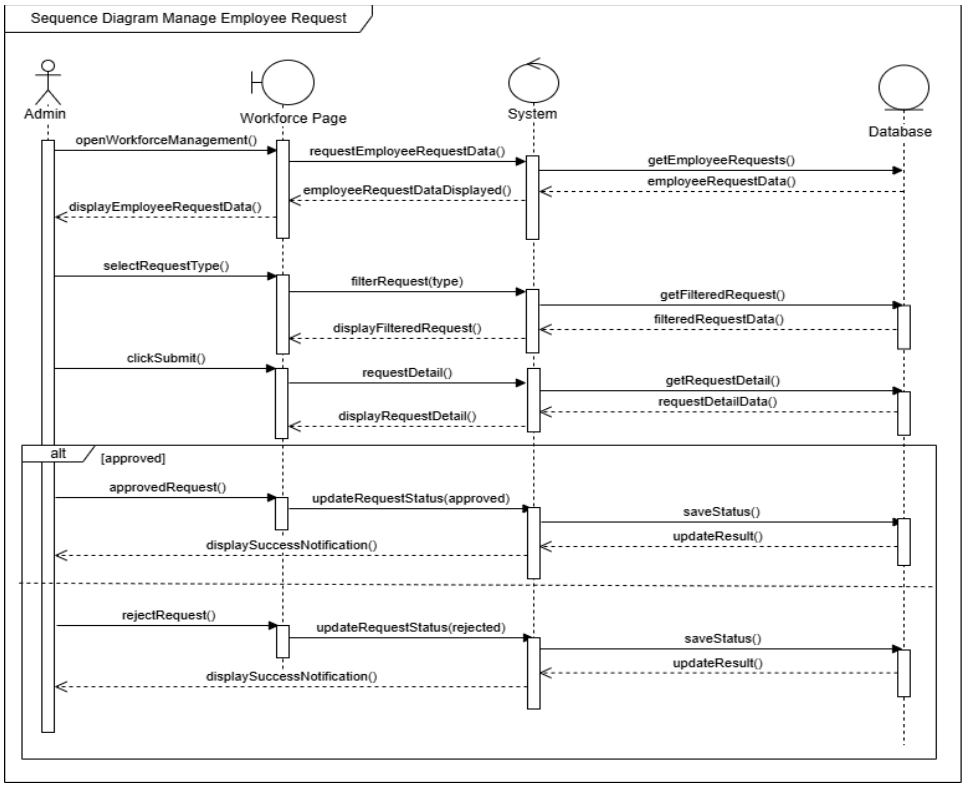
Sequence diagram manage attendance digunakan untuk menggambarkan proses *int*eraksi antara admin dan sistem dalam mengelola data absensi karyawan pada HagaPlus. Diagram ini menunjukkan alur pengelolaan data kehadiran seperti pencatatan, pencarian, filter data, hingga menampilkan data absensi karyawan pada sistem.



Gambar 2.36 *Sequence Diagram Manage Attendance*

l. *Sequence Diagram Manage Employee Requests*

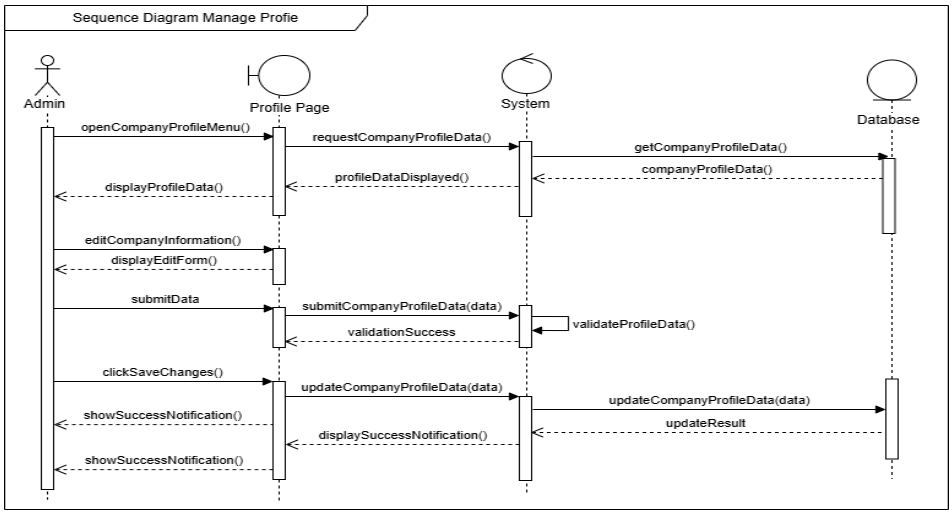
Sequence diagram manage employee requests digunakan untuk menggambarkan proses admin dalam mengelola pengajuan karyawan pada sistem HagaPlus. Diagram ini menunjukkan alur peninjauan dan persetujuan pengajuan seperti cuti, *reimbursement*, dan pertukaran *shift* yang dilakukan melalui sistem.



Gambar 2.37 ***Sequence Diagram Manage Employee Request***

m. ***Sequence Diagram Manage Company Profile***

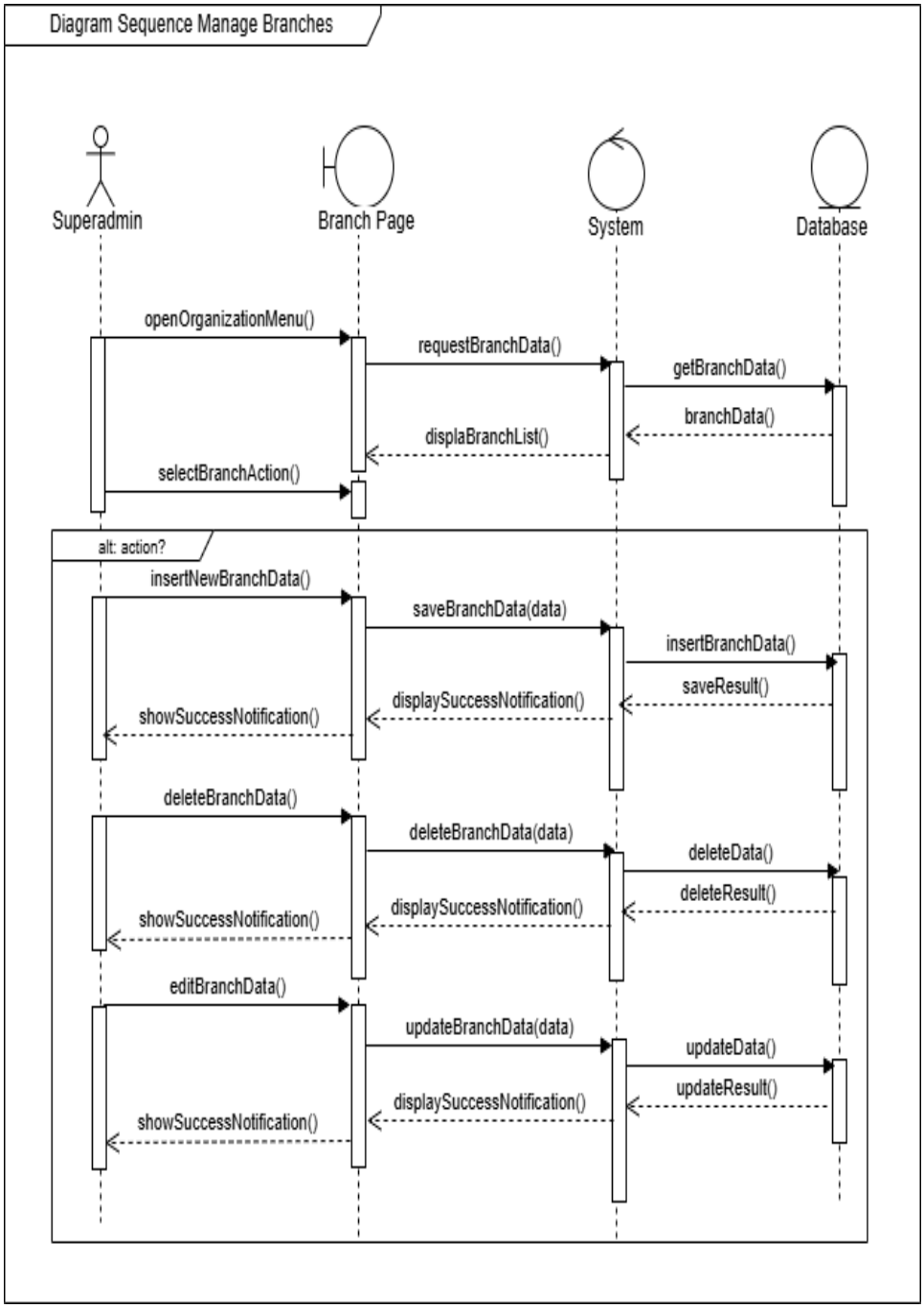
Sequence diagram manage company profile digunakan untuk menggambarkan proses admin dalam mengelola informasi profil perusahaan pada sistem HagaPlus. Diagram ini menunjukkan alur pengubahan dan penyimpanan data profil perusahaan yang dilakukan melalui sistem.



Gambar 2.38 ***Sequence Diagram Manage Company Profile***

n. *Sequence Diagram Manage Branches*

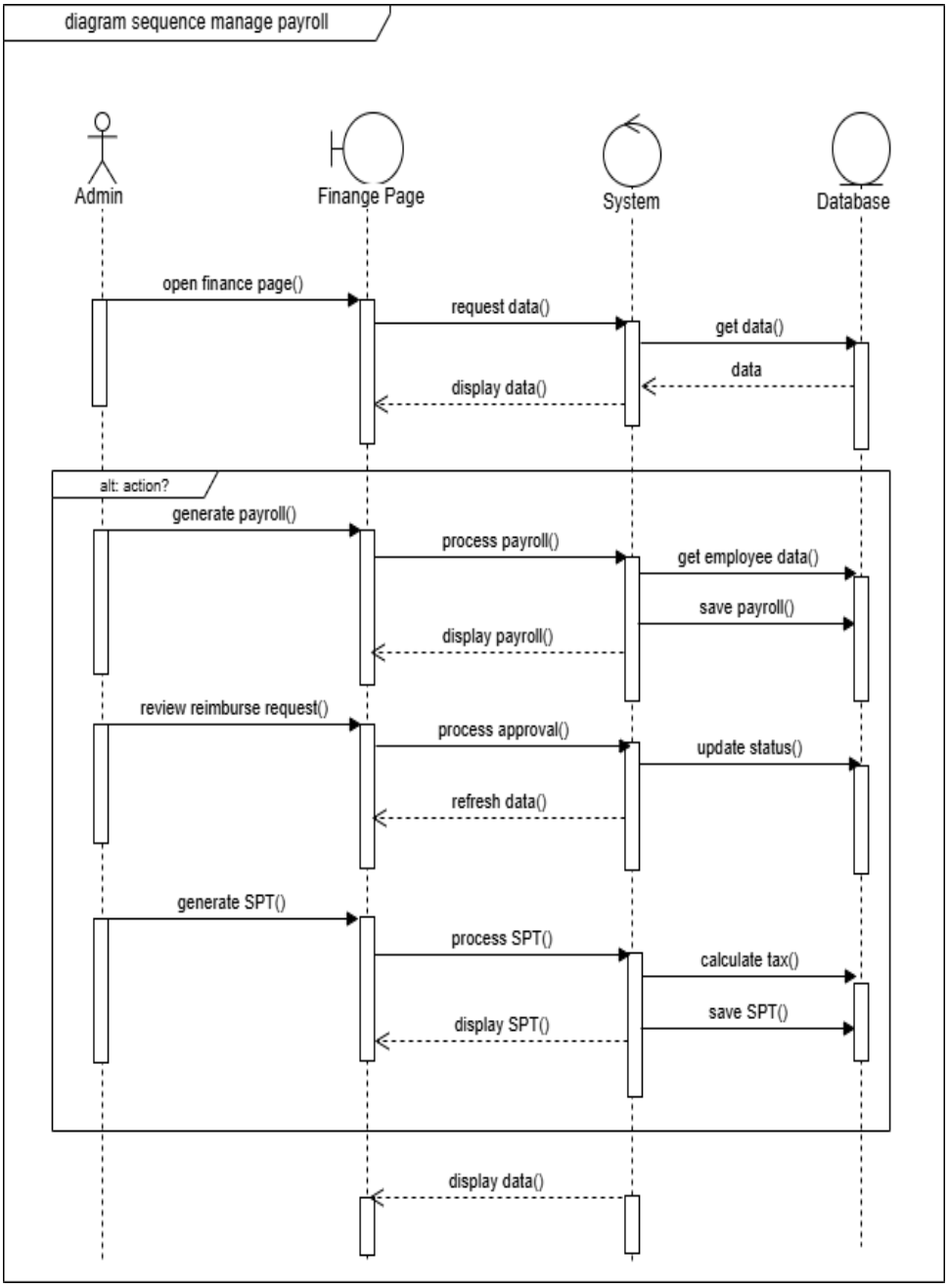
Sequence diagram manage branches digunakan untuk menggambarkan proses admin dalam mengelola data cabang perusahaan pada sistem HagaPlus. Diagram ini menunjukkan alur penambahan, pengubahan, dan pengelolaan informasi cabang yang dilakukan melalui sistem.



Gambar 2.39 *Sequence Diagram Manage Branches*

o. ***Sequence Diagram Manage Payroll***

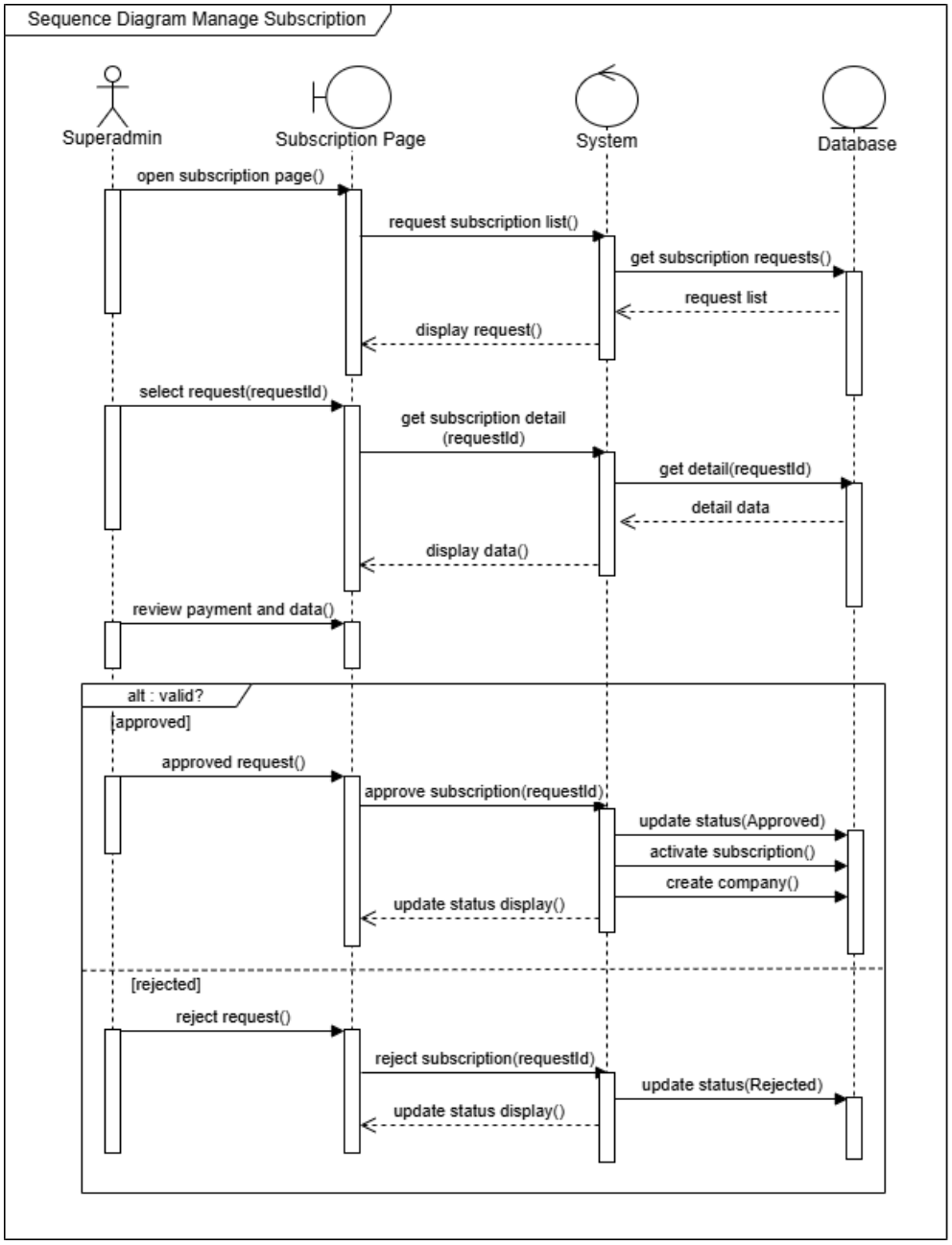
Sequence diagram manage payroll digunakan untuk menggambarkan proses admin dalam mengelola data penggajian karyawan pada sistem HagaPlus. Diagram ini menunjukkan alur perhitungan gaji, pembuatan data *payroll*, peninjauan data *payroll*, hingga proses ekspor data penggajian melalui sistem.



Gambar 2.40 ***Sequence Diagram Manage Payroll***

p. *Sequence Diagram Manage Subscription*

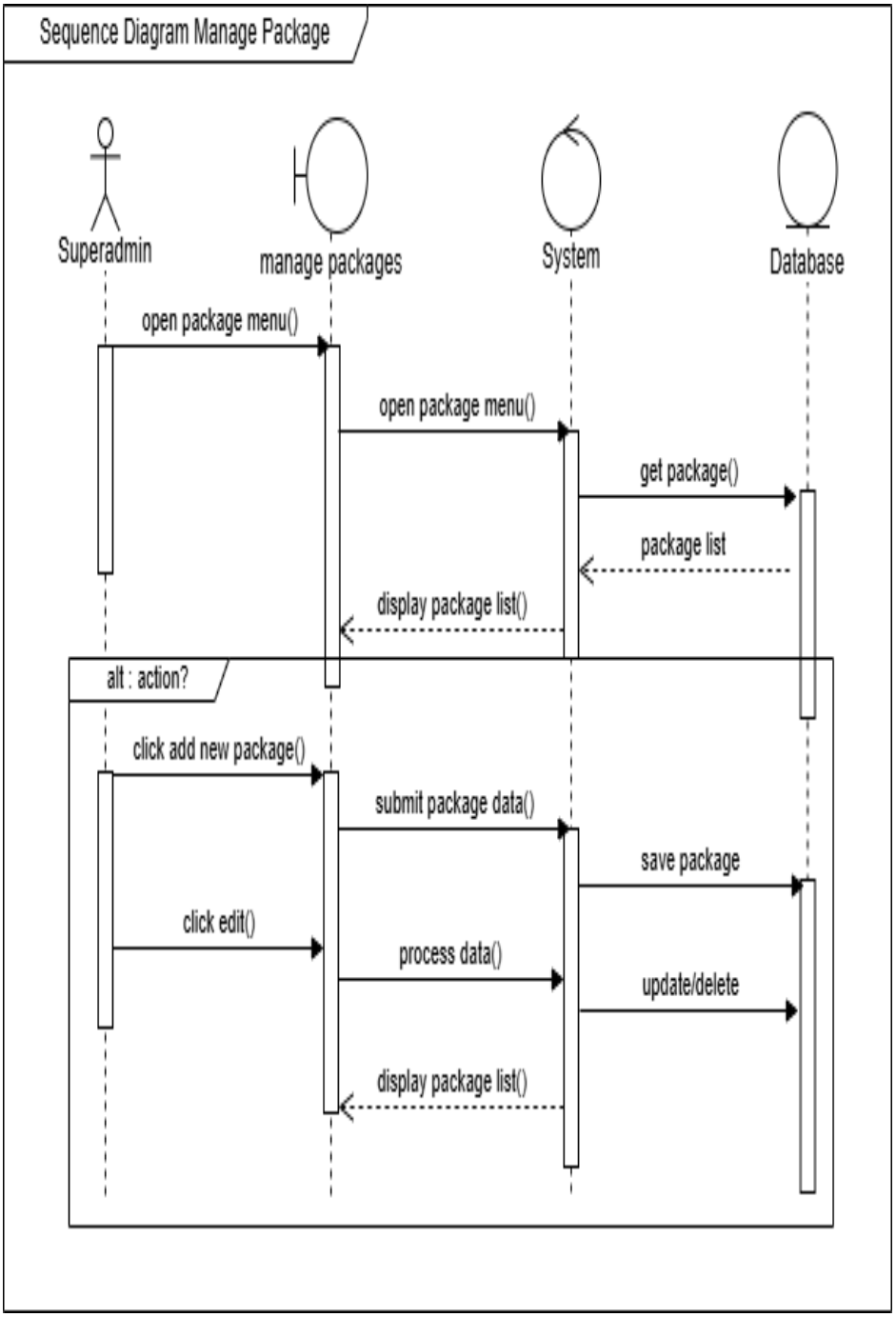
Sequence diagram manage subscription digunakan untuk menggambarkan proses *int*eraksi antara superadmin dan sistem dalam mengelola perm*int*aan *subscription* pada HagaPlus. Diagram ini menunjukkan alur peninjauan data *subscription*, validasi pembayaran, hingga proses persetujuan atau penolakan *subscription* oleh superadmin.



Gambar 2.41 *Sequence Diagram Manage Subscription*

q. *Sequence Diagram Manage Package*

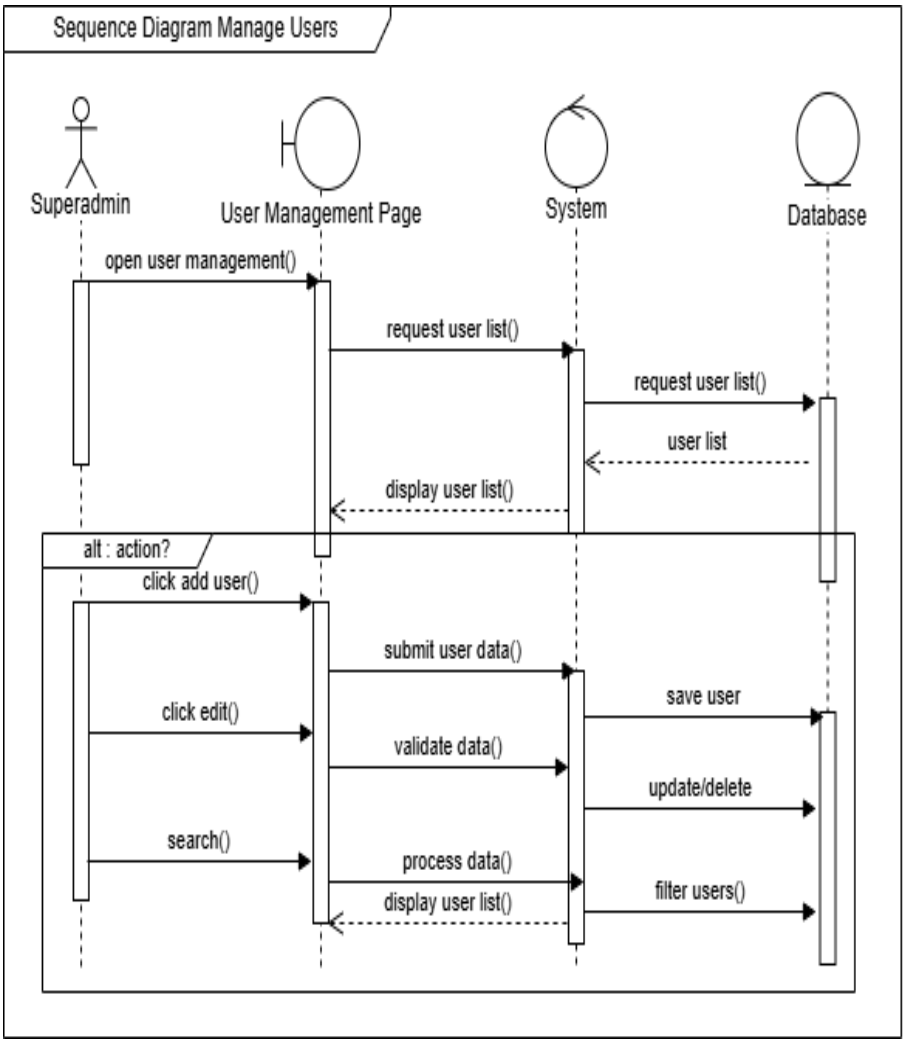
Sequence diagram manage package digunakan untuk menggambarkan proses superadmin dalam mengelola data paket *subscription* pada sistem HagaPlus. Diagram ini menunjukkan alur penambahan, pengubahan, validasi, dan penyimpanan data *package* yang dilakukan melalui sistem.



Gambar 2.42 *Sequence Diagram Manage Package*

r. ***Sequence Diagram Manage Users***

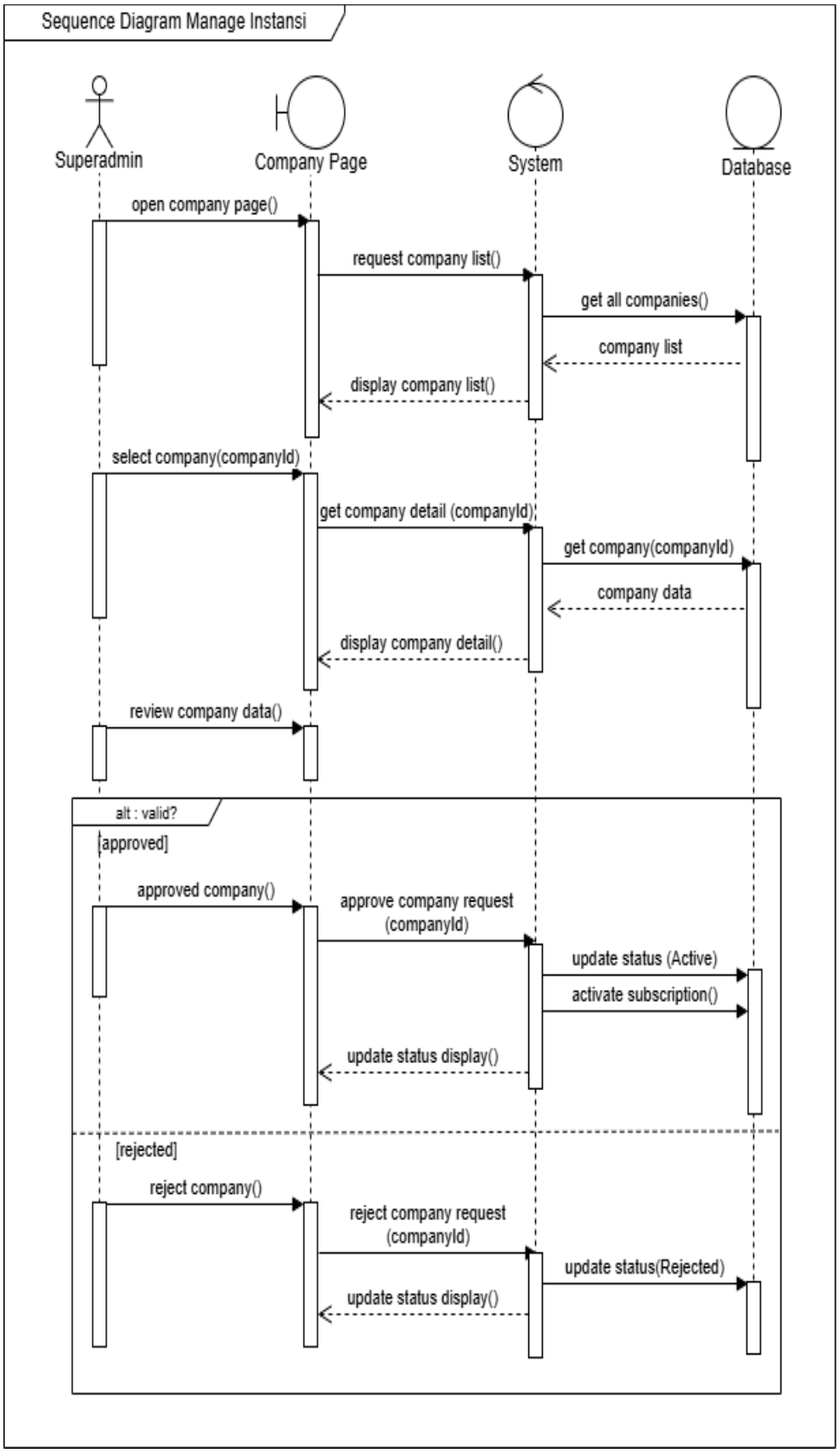
Sequence diagram manage users digunakan untuk menggambarkan proses superadmin dalam mengelola data pengguna pada sistem HagaPlus. Diagram ini menunjukkan alur penambahan, pengubahan, validasi, dan penyimpanan data pengguna yang dilakukan melalui sistem.



Gambar 2.43 ***Sequence Diagram Manage Users***

s. ***Sequence Diagram Manage Instansi***

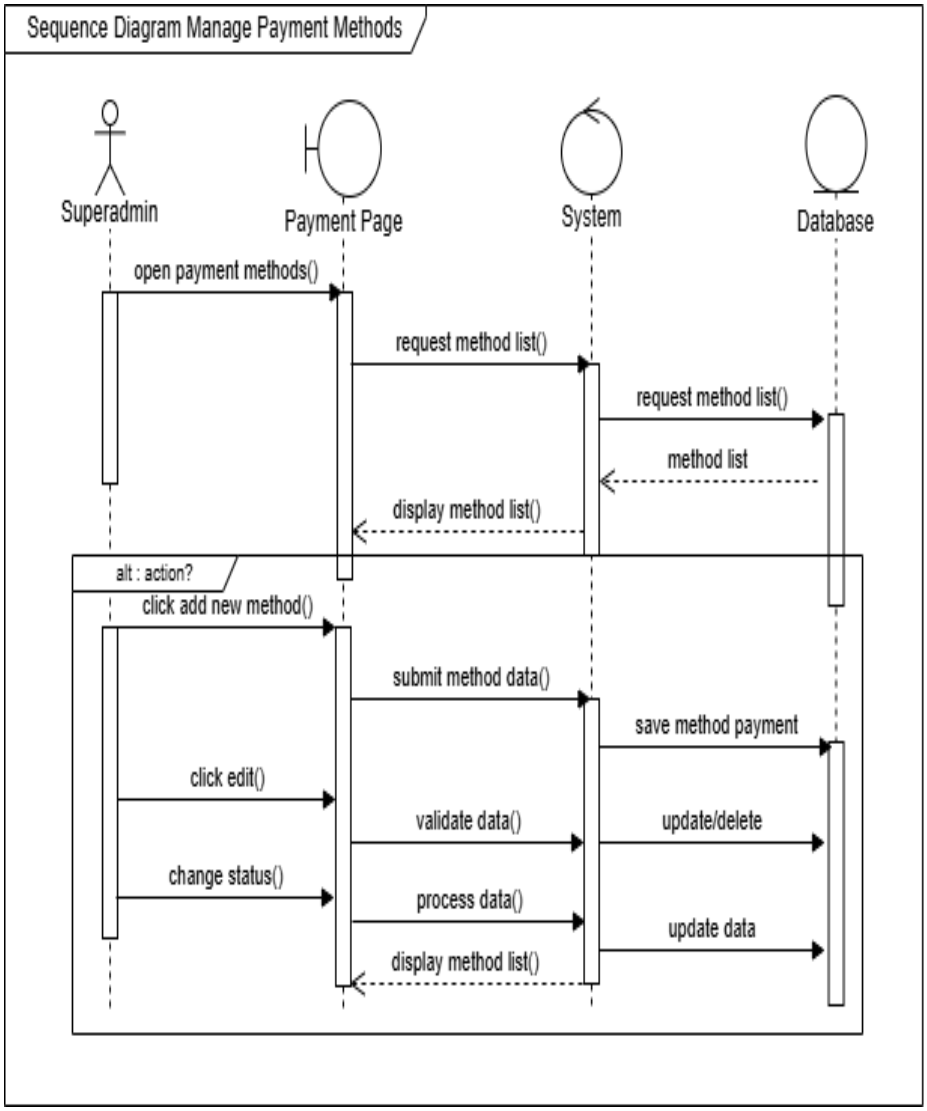
Sequence diagram manage instansi digunakan untuk menggambarkan proses superadmin dalam mengelola data instansi pada sistem HagaPlus. Diagram ini menunjukkan alur pengubahan dan penyimpanan informasi instansi yang dilakukan melalui sistem.



Gambar 2.44 ***Sequence Diagram Manage Instansi***

t. *Sequence Diagram Manage Payment Methods*

Sequence diagram manage payment methods digunakan untuk menggambarkan proses superadmin dalam mengelola metode pembayaran pada sistem HagaPlus. Diagram ini menunjukkan alur penambahan, validasi, dan penyimpanan data metode pembayaran yang dilakukan melalui sistem.

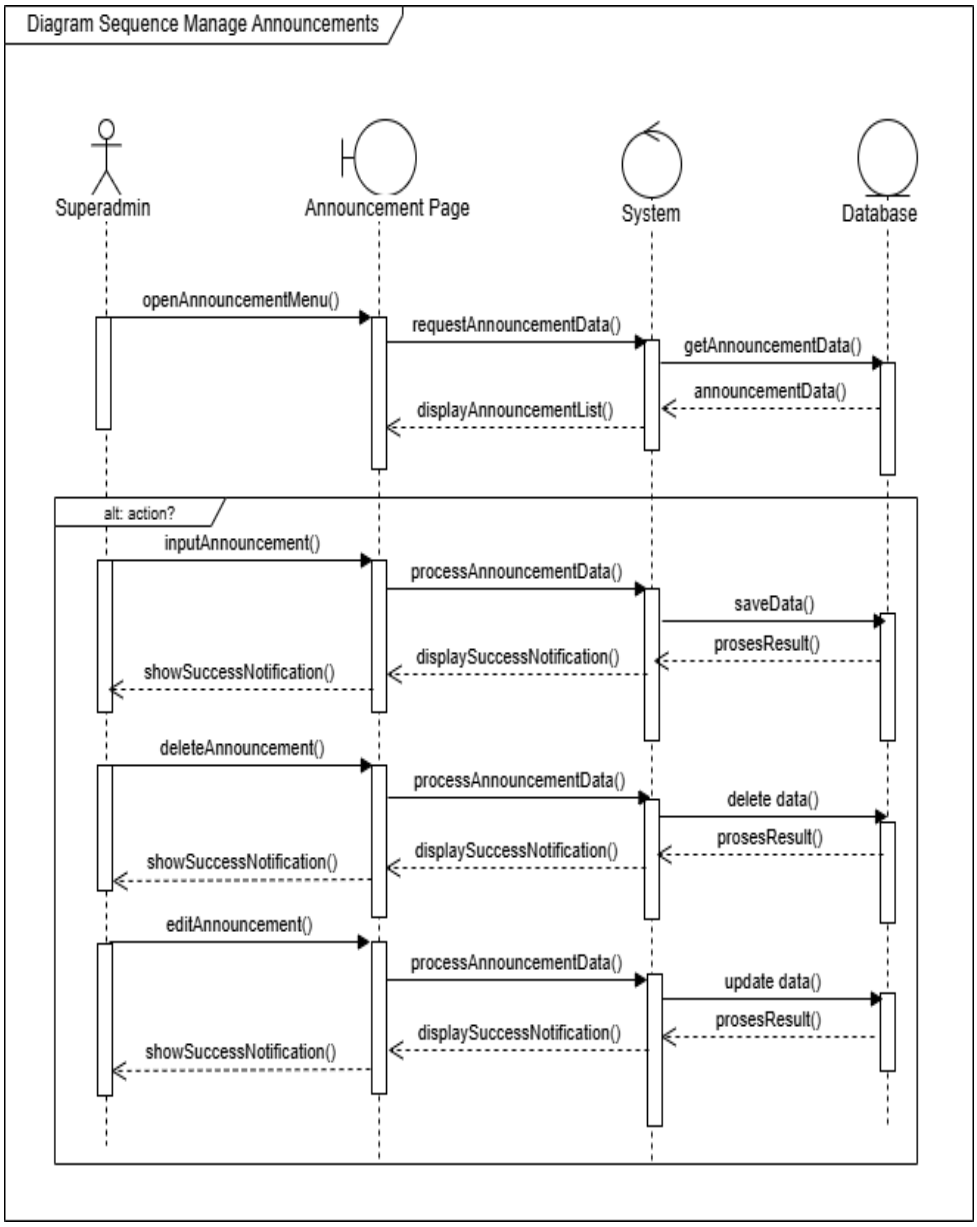


Gambar 2.45 *Sequence Diagram Manage Payment Methods*

u. *Sequence Diagram Manage Announcements*

Sequence diagram manage announcements digunakan untuk menggambarkan proses superadmin dalam mengelola pengumuman pada sistem HagaPlus. Diagram ini menunjukkan alur pembuatan,

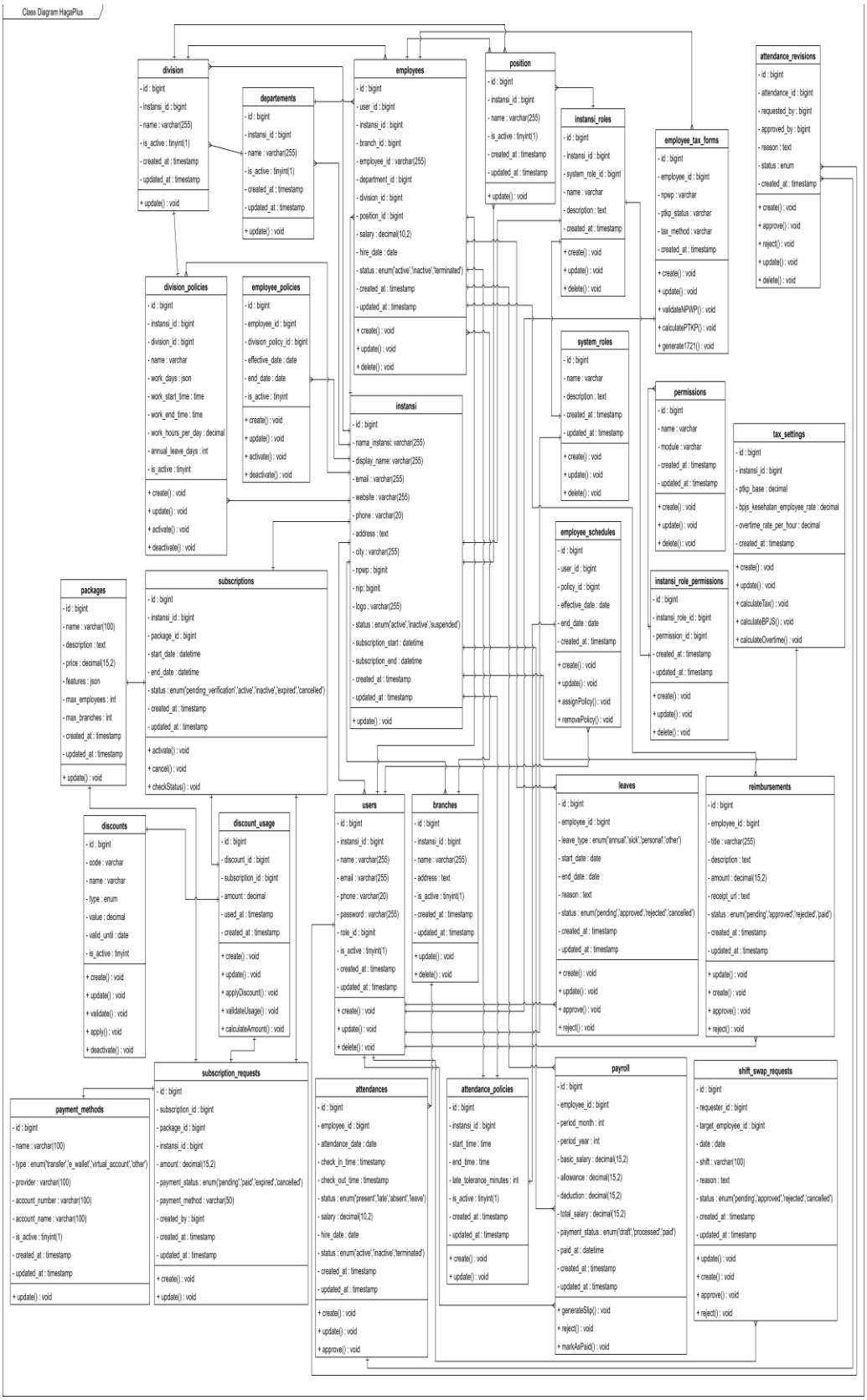
penyimpanan, dan publikasi pengumuman yang dilakukan melalui sistem.



Gambar 2.46 ***Sequence Diagram Manage Announcements***

2.6.4 *Class Diagram*

Class diagram digunakan untuk menggambarkan struktur sistem dalam bentuk kelas yang saling berhubungan, termasuk atribut dan fungsi yang dimiliki oleh masing-masing kelas. Pada sistem HagaPlus, *class diagram* mencakup entitas utama seperti *user*, absensi, cuti, dan penggajian, yang menjadi dasar dalam perancangan sistem serta pengembangan *database*.



Gambar 2.47 ***Class Diagram HagaPlus***

2.7 Perangkat yang Dibutuhkan

Dalam perancangan dan pengembangan sistem HagaPlus, diperlukan beberapa perangkat pendukung baik dari sisi perangkat keras (*hardware*) maupun perangkat lunak (*software*). Perangkat tersebut digunakan untuk mendukung jalannya sistem serta pengolahan data yang diperlukan.

2.7.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam pengembangan sistem ini berupa perangkat komputer atau laptop yang digunakan untuk menjalankan tools pengembangan serta pengujian sistem.

Tabel 2.49 Perangkat Keras yang dibutuhkan

No	Perangkat	Spesifikasi
1	Laptop	*Processor Int*el Core i5-8365 U
2	RAM	16 GB
3	*Storage*	256 GB

2.7.2 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam pengembangan sistem HagaPlus merupakan tools yang mendukung pembuatan aplikasi berbasis web, pengelolaan *database*, serta proses pengujian sistem.

Tabel 2.50 Perangkat Lunak yang dibutuhkan

No	Perangkat Lunak	Keterangan
1	Laravel	*Framework* untuk membangun sistem berbasis web
2	MySQL	*Database* untuk menyimpan data sistem
3	Laragon	*Web server* lokal untuk menjalankan aplikasi
4	PhpMyAdmin	*Tools* untuk mengelola *database* MySQL
5	Visual Studio Code	*Code editor* untuk menulis program
6	Browser	Untuk mengakses dan menjalankan sistem

BAB III
IMPLEMENTASI DAN PENGGUNAAN

3.1 Implementasi dan Rancangan Waktu

Implementasi Sistem HagaPlus dilakukan secara bertahap berdasarkan hasil analisis kebutuhan yang telah dilakukan sebelumnya. Proses pengembangan dimulai dari tahap perencanaan project, analisis kebutuhan, perancangan sistem, implementasi aplikasi, pengujian sistem, hingga penyusunan dokumentasi. Setiap tahapan saling berkaitan untuk memastikan sistem yang dibangun sesuai dengan kebutuhan mitra dan dapat berfungsi secara optimal.

Rancangan waktu pengembangan disusun sebagai acuan pelaksanaan project mulai dari tahap perencanaan pada bulan Desember hingga penyelesaian laporan, pengajuan HKI, dan persiapan sidang pada bulan Mei. Adapun rincian tahapan dan waktu pelaksanaan pengembangan Sistem HagaPlus disajikan pada Tabel 3.1.

Tabel 3.1 Rancangan Waktu Pengembangan Sistem

Nama Kegiatan	Rentang Waktu	Penjelasan	PIC
Perencanaan *Project*	Desember (Minggu 1-2)	Menentukan topik project, identifikasi kebutuhan mitra, menentukan ruang lingkup sistem, pembagian tugas anggota tim, serta penyusunan jadwal pengerjaan project.	Tim
Analisis Kebutuhan Sistem	Desember (Minggu 3-4)	Melakukan observasi, identifikasi permasalahan, analisis kebutuhan fungsional dan non-fungsional, serta analisis proses bisnis yang akan diterapkan pada sistem.	Tim

Perancangan Sistem	Januari (Minggu 1-3)	Menyusun UML (*Use Case*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*), ERD, LRS, serta merancang struktur *database* dan alur sistem.	Siti Khodijah
Perancangan *UI/UX*	Januari (Minggu 3-4)	Mendesain antarmuka sistem menggunakan Figma/alat desain meliputi halaman *login*, dashboard, absensi, *payroll*, cuti, *reimbursement*, dan halaman *admin*.	Siti Khodijah
Pengembangan Sistem (*Frontend* & *Backend*)	Februari (Minggu 1) – Maret (Minggu 2)	Implementasi sistem menggunakan Laravel, Bootstrap, MySQL, meliputi pembuatan fitur autentikasi, absensi, *payroll*, cuti, *reimbursement*, *subscription*, serta dashboard.	Lukman Muludin
*Int*egrasi dan Penyempurnaan Sistem	Maret (Minggu 3-4)	Meng*int*egrasikan seluruh modul, memperbaiki bug, melakukan optimasi *database*, validasi data, dan penyempurnaan tampilan antarmuka.	Lukman Muludin
Pengujian Sistem (*Testing*)	April (Minggu 1-2)	Melakukan *Usability Testing* pada seluruh fitur, memperbaiki *error* yang ditemukan, dan memastikan	Tim

		seluruh fungsi berjalan sesuai kebutuhan pengguna.	
Deployment dan Implementasi	April (Minggu 3)	Melakukan *deployment* ke hosting, konfigurasi *database production*, pengujian akses sistem secara *online*, serta memastikan sistem dapat digunakan oleh pengguna.	Lukman Muludin
Penyusunan Laporan *Project Work*	April (Minggu 4) - Mei (Minggu 2)	Menyusun laporan akhir meliputi pendahuluan, analisis, perancangan, implementasi, pengujian, dokumentasi sistem, serta penyusunan lampiran.	Siti Khodijah
Penyusunan *Guidebook*	Mei (Minggu 1-2)	Menyusun panduan instalasi, *deployment*, penggunaan sistem, serta *troubleshooting* bagi *administrator* dan pengguna.	Tim
Pengajuan Hak Kekayaan *Int*elektual (HKI)	Mei (Minggu 2-3)	Menyiapkan dokumen persyaratan, deskripsi program komputer, identitas pencipta, serta melakukan pendaftaran HKI untuk sistem HagaPlus.	Tim
Finalisasi Laporan dan Persiapan Sidang	Mei (Minggu 4)	Melakukan revisi laporan, pengecekan format, persiapan *present*asi, demo aplikasi, serta latihan menjawab pertanyaan sidang.	Tim

3.2 Instalasi Sistem

Sebelum melakukan instalasi dan pengembangan sistem HagaPlus, terdapat beberapa persyaratan yang perlu dipenuhi agar sistem dapat berjalan dengan baik. Persyaratan tersebut meliputi perangkat keras (*hardware*), perangkat lunak (*software*), serta komponen pendukung yang digunakan dalam proses pengembangan dan implementasi sistem.

3.2.1 Persyaratan Sistem

a. Persayaratan *Hardware*

Persyaratan *hardware* merupakan spesifikasi minimum perangkat yang digunakan untuk menjalankan sistem HagaPlus. Perangkat harus mampu menjalankan *web server*, *database*, PHP, dan Node.js secara bersamaan. Selain itu, fitur absensi berbasis lokasi dan *selfie* memerlukan perangkat yang dilengkapi kamera dan GPS. Rincian persyaratan *hardware* ditunjukkan pada Tabel 3.2.

Tabel 3.2 Persyaratan *Hardware*

No	Komponen	Minimum	Rekomendasi	Keterangan
1	Prosesor (CPU)	Dual-Core 2 GHz	Quad-Core 2.5 GHz atau lebih	Menjalankan *web server*, PHP, dan MySQL secara bersamaan.
2	Memori (RAM)	4 GB	8 GB atau lebih	Mendukung proses pengembangan dengan PHP, MySQL, Node.js, dan browser.
3	Penyimpanan (*Storage*)	10 GB ruang kosong	20 GB atau lebih (SSD)	Menyimpan *source code*, *database*,

				dependensi proyek, dan *file upload*.
4	Kartu Jaringan (NIC)	100 Mbps	300 Mbps	Mendukung akses *int*ernet saat instalasi dependensi dan penggunaan aplikasi.
5	Kamera	Tersedia	Resolusi minimal 2 MP	Digunakan untuk fitur *selfie* saat *check-in* dan *check-out*.
6	GPS / Penangkap Lokasi	Tersedia	GPS bawaan perangkat	Digunakan untuk validasi lokasi absensi berbasis geolokasi.
7	Layar	1280 x 720 px	1920 x 1080 px atau lebih	Memberikan tampilan dashboard yang optimal.

b. Persyaratan *Software*

Persyaratan *software* mencakup sistem operasi, *web server*, bahasa pemrograman, *database*, serta perangkat lunak pendukung yang digunakan dalam pengembangan sistem. HagaPlus dikembangkan menggunakan Laravel 12 dengan PHP 8.2, MySQL sebagai *database*, serta Node.js dan NPM untuk pengelolaan aset *frontend*. Rincian persyaratan *software* ditunjukkan pada Tabel 3.3.

Tabel 3.3 Persyaratan *Software*

No	Komponen	Versi Minimum	Versi yang Digunakan	Keterangan
1	Sistem Operasi	Windows 10 / macOS 12 / Ubuntu 20.04	Windows 11 / macOS 14 / Ubuntu 22.04	Mendukung pengembangan dan *deployment* pada berbagai *platform*.
2	*Web Server*	Nginx 1.18 / Apache 2.4 / Laragon	Laragon	Direkomendasikan karena mudah dikonfigurasi di Windows.
3	PHP	8.2	8.2 atau lebih	Laravel 12 membutuhkan PHP 8.2 beserta ekstensi pendukung.
4	Composer	2.x	2.8 atau lebih	Mengelola *dependency* proyek Laravel.
5	MySQL	5.7	8.0 atau lebih	*Database* utama untuk menyimpan data sistem.
6	Node.js	18.x	20.x atau lebih	Digunakan untuk Vite, Tailwind CSS, Alpine.js, dan build *frontend*.
7	NPM	9.x	10.x atau lebih	Mengelola *dependency* JavaScript.

8	Git	2.x	2.40 atau lebih	*Version control* selama pengembangan.
9	*Browser Web*	Chrome 90 / Firefox 90 / Edge 90	Chrome atau Firefox terbaru	Mendukung JavaScript modern, kamera, *geolocation*, dan *WebSocket*.
10	*PHP Extension*	-	bcmath, c*type*, fileinfo, json, mbstring, openssl, pdo_mysql, tokenizer, xml	Ekstensi PHP yang wajib diaktifkan pada php.ini.

c. Komponen Pendukung

Selain *hardware* dan *software* utama, HagaPlus juga menggunakan beberapa *library* dan *package* tambahan untuk mendukung berbagai fitur sistem, seperti pembuatan dokumen PDF, ekspor data ke Excel, autentikasi Google, komunikasi *real-time*, serta pemindaian *QR Code*. Daftar komponen pendukung dapat dilihat pada Tabel 3.4.

Tabel 3.4 Komponen Pendukung Sistem

No	Kategori	Paket	Versi	Fungsi
1	*Framework*	Laravel	12.x	*Framework* utama dengan arsitektur MVC.
2	PDF	DomPDF	3.x	Mencetak slip gaji dan dokumen PDF.
3	PDF	TCPDF	6.x	Mencetak formulir SPT 1721-A1.

4	Excel	Maatwebsite/ Excel	1.1	Mengekspor data absensi dan penggajian ke Excel.
5	WebSocket	Laravel Reverb	1.6	Mendukung notifikasi real-*time*.
6	WebSocket	Pusher PHP Server	7.2	Driver komunikasi WebSocket.
7	Autentikasi	Laravel Socialite	5.23	*Login* menggunakan akun Google *(OAuth).*
8	Captcha	Cloudflare Turnstile	2.3	Memverifikasi bahwa pengguna adalah manusia serta melindungi proses *login* dan registrasi dari bot.
9	*Frontend*	Tailwind CSS	3.4	*Framework* CSS untuk membangun antarmuka pengguna.
10	*Frontend*	Alpine.js	3.15	*Framework* JavaScript ringan untuk *int*eraktivitas halaman.
11	HTTP *Client*	Axios	1.11	Mengirim *request* HTTP antara *frontend* dan *backend*.

3.2.2 Instalasi Lokal

Instalasi lokal merupakan tahapan untuk mempersiapkan lingkungan pengembangan sehingga aplikasi dapat dijalankan pada komputer lokal. Pada tahap ini dilakukan proses instalasi dan konfigurasi yang diperlukan sebelum aplikasi digunakan.

1. *Clone* Repositori

   Clone proyek ini ke folder *web server* lokal Anda (misal C:\laragon\www\hagaplus):

   ```
   git clone https://github.com/skhodijah/hagaplus.git
   cd hagaplus
   ```

2. *Install Dependency* PHP & Node.js

   Jalankan per*int*ah berikut untuk meng *install package composer* dan NPM:

   ```
   composer install
   npm install
   ```

3. Salin *File Environment*

   Buat salinan *file* .env dari .env.example:

   ```
   cp .env.example .env
   ```

4. Konfigurasi *Environment* (.env)

   Buka *file* .env yang baru dibuat dan sesuaikan konfigurasi *database* Anda:

   ```
   APP_NAME=Hagaplus
   APP_ENV=local
   APP_KEY=
   APP_DEBUG=true
   APP_URL=http://hagaplus.test #atau http://localhost:8000

   DB_CONNECTION=mysql
   DB_HOST=127.0.0.1
   DB_PORT=3306
   DB_DATABASE=hagaplus #sesuaikan dengan nama database yang digunakan
   DB_USERNAME=root #sesuaikan dengan username database yang digunakan
   DB_PASSWORD= #sesuaikan dengan password databsae yang digunakan
   ```

5. *Generate Application Key*

   Jalankan per*int*ah artisan untuk meng-*generate application key*:

   ```
   php artisan key:generate
   ```

6. Migrasi *Database* dan *Seeding*

   Buat *database* bernama 'hagaplus' (bisa disesuaikan) di MySQL Anda, lalu jalankan migrasi beserta data awal (*seeder*):

   ```
   php artisan migrate -seed
   ```

7. Jalankan Server Pengembangan

   Gunakan perintah *internal composer* untuk menyalakan Laravel server, *queue worker*, dan *Vite bundler* secara bersamaan:

```
composer run dev
```

   Atau Anda bisa menjalankannya secara manual di terminal terpisah:

```
# Terminal 1: Laravel Server
php artisan serve

# Terminal 2: Queue Listener (Untuk proses asynchronous/notifikasi)
php artisan queue:listen

# Terminal 3: Vite Dev Server (Aset Frontend)
npm run dev
```

   Akses sistem melalui *browser* pada url yang tertera (contoh: http://localhost:8000).

3.2.3 Panduan *Deployment* ke *Virtual Private Server* (VPS)

Virtual Private Server (VPS) merupakan layanan *server virtual* yang menyediakan lingkungan server dengan sumber daya komputasi yang dialokasikan untuk pengguna. VPS memberikan akses dan kontrol konfigurasi yang lebih luas dibandingkan *shared hosting*, sehingga pengguna dapat melakukan konfigurasi terhadap sistem operasi, aplikasi, *web server*, dan basis data.

Pada implementasi HagaPlus, VPS digunakan sebagai lingkungan *deployment* karena sistem dirancang untuk dapat digunakan dalam skala pengguna yang lebih besar. Penggunaan VPS menyediakan sumber daya komputasi dan penyimpanan yang lebih memadai serta memberikan fleksibilitas dalam melakukan konfigurasi server.

Spesifikasi VPS yang digunakan dalam proses *deployment* ditampilkan pada Tabel dibawah ini:

Tabel 3.5 Spesifikasi VPS yang digunakan untuk *deployment*

Komponen	Spesifikasi
vCPU Core	8
RAM	32 GB
Disk Space	400 GB
IPv4 dan IPv6	1
Inode	*Unlimited*
I/O	300 MB/s
Bandwidth	32 TB

Spesifikasi tersebut mencakup kapasitas pemrosesan, memori, penyimpanan, *bandwidth*, serta batas *inode* yang menjadi pertimbangan dalam menyediakan lingkungan server untuk menjalankan sistem HagaPlus.

Berikut adalah panduan *deployment* aplikasi Hagaplus ke VPS dari awal hingga selesai:

a. Persiapan Aset & *File* Project (Lokal)

Sebelum mengunggah proyek ke server, lakukan beberapa langkah persiapan di komputer lokal Anda terlebih dahulu:

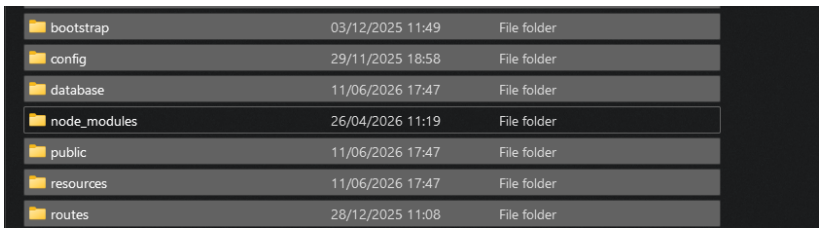
1. Kompilasi Aset *Frontend* (Build): Jalankan per*int*ah berikut pada terminal lokal Anda untuk mem-*build* aset CSS (Tailwind) dan JavaScript (Alpine.js/Vite) ke folder public/build:

```
npm run build
```

2. Instalasi *Dependency* Produksi (Opsional): Bila hosting Anda tidak menyediakan akses Composer atau SSH, pastikan *dependency* Laravel sudah terinstal lengkap secara lokal dengan per*int*ah:

```
composer install --no-dev --optimize-autoloader
```

3. Kompresi Folder Proyek ke Format ZIP: Buat arsip .zip (misalnya: hagaplus.zip) dari seluruh isi folder proyek Anda.
   a) PENTING: Jangan sertakan folder node_modules untuk menghemat alokasi ruang penyimpanan dan batas jumlah *file* (*inodes*) pada akun hosting Anda.



Gambar 3.1 Kompres seluruh *project* tanpa node_modules

   b) Pastikan folder vendor/ ikut dikompres jika hosting Anda tidak mendukung terminal SSH/Composer.

b. Konfigurasi *Environment* (.env)

1. Di dalam *File Manager* hosting Anda, salin berkas .env.example lalu ubah namanya menjadi .env.

2. Edit *file* .env tersebut dan sesuaikan variabel konfigurasi dengan kondisi hosting Anda, kode di seluruh *file* .env bersifat rahasia, berikut contoh konfigurasi yang telah disesuaikan:

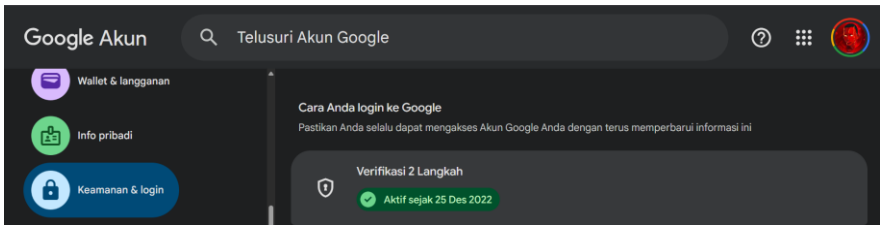
```
APP_NAME=Hagaplus
APP_ENV=production
APP_DEBUG=false
APP_URL=https://nama-domain-anda.com #Sesuaikan dengan domain HTTPS
Anda

# Sesuaikan dengan yang kita dapatkan dari menu MySQL Database di panel hosting
DB_CONNECTION=mysql
DB_HOST= sql204.infinityfree.com
DB_PORT=3306
DB_DATABASE= if0_42179653_hagaplus
DB_USERNAME= if0_42179653
DB_PASSWORD= 72onevif

QUEUE_CONNECTION=database
SESSION_DRIVER=database
CACHE_STORE=database

# Konfigurasi SMTP Mail (Gmail)
MAIL_MAILER=smtp
MAIL_HOST=smtp.gmail.com
MAIL_PORT=587
MAIL_USERNAME=hagaplus@gmail.com
MAIL_PASSWORD=kgkvzulzhtbyhquv #Masukkan Password Aplikasi Google
(App Password)
MAIL_ENCRYPTION=tls
MAIL_FROM_ADDRESS=hagaplus@gmail.com #Sesuaikan dengan email ygang
digunakan
MAIL_FROM_NAME="Haga+"

# Konfigurasi Google Client (OAuth / Socialite Login)
GOOGLE_CLIENT_ID=679795148946-
43k4palrabr1q3d6ad55c20ok10aghpj.apps.googleusercontent.com
GOOGLE_CLIENT_SECRET=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
GOOGLE_REDIRECT_URI="${APP_URL}/auth/google/callback"
```

c. Cara Memperoleh *Password* Aplikasi Gmail (*App Password*)

Aplikasi menggunakan layanan SMTP Gmail untuk mengirim *email* seperti notifikasi dan *reset password*. Untuk alasan keamanan, Google tidak memperbolehkan penggunaan *password* akun Gmail secara langsung sehingga diperlukan *App Password*.

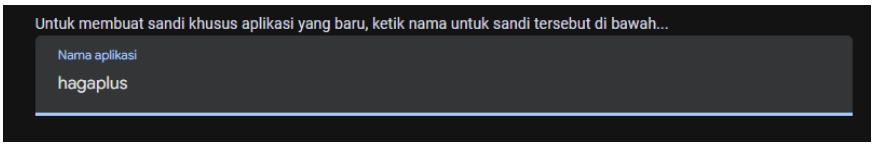
Langkah-langkah untuk memperoleh *App Password* adalah sebagai berikut:

1. Buka akun Google yang akan digunakan sebagai pengirim email.
2. Buka halaman pengaturan keamanan akun Google melalui : https://myaccount.google.com/security
3. Aktifkan fitur Verifikasi 2 Langkah (*2-Step Verification*) apabila belum aktif.



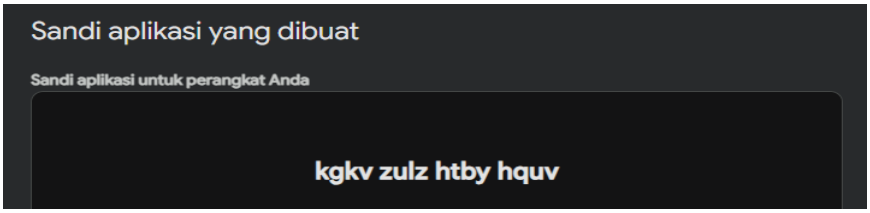
Gambar 3.2 Verifikasi 2 Langkah akun google

4. Setelah verifikasi dua langkah aktif, buka menu *App Passwords*. https://myaccount.google.com/apppasswords
5. Pada bagian *form input* nama aplikasi masukkan nama aplikasi, misalnya Hagaplus.



Gambar 3.3 *Input* nama aplikasi di *app password*

6. Klik tombol *Generate*/Buat.
7. Google akan menampilkan kode sepanjang 16 karakter.



Gambar 3.4 Sandi Aplikasi

8. Salin kode tersebut dan masukkan ke dalam *variable* tanpa spasi

   Contoh:

   ```
   MAIL_PASSWORD=kgkvzulzhtbyhquv
   ```

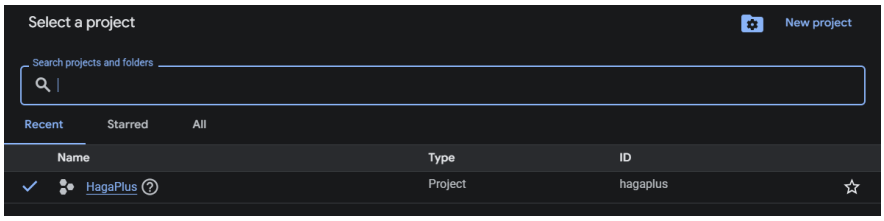
d. Cara Membuat *Google OAuth Client ID* dan *Client Secret*

Fitur login menggunakan akun Google memerlukan *OAuth Client ID* dan *Client Secret* yang diperoleh melalui *Google Cloud Console*.

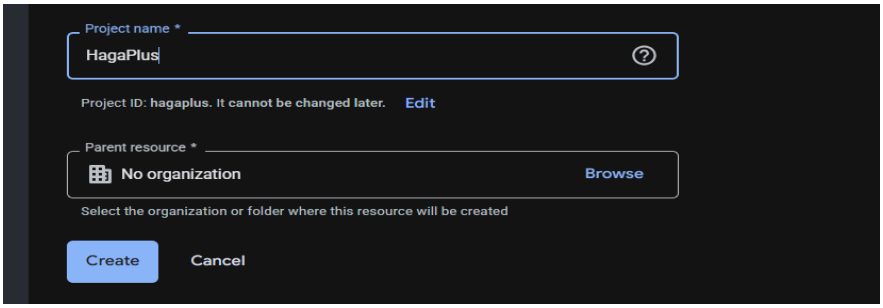
*Lang*kah-langkah konfigurasi *OAuth adalah* sebagai berikut:

1. Buka Google *Cloud Console menggu*nakan akun Google yang *a*kan digunakan melalui https://console.cloud.google.com/

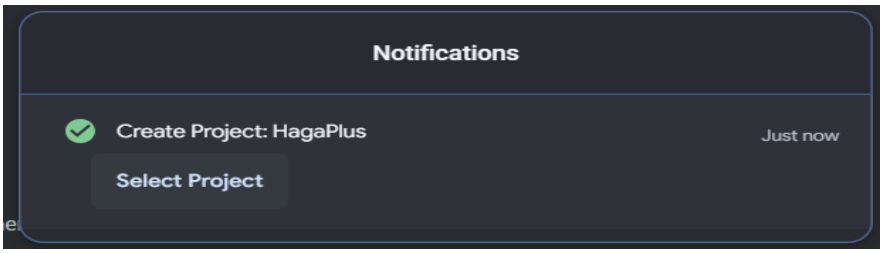
2. Gunakan *project* yang sudah ada, atau buat *project* baru dengan klik *new project.*



Gambar 3.5 Menentukan *project* yang akan digunakan

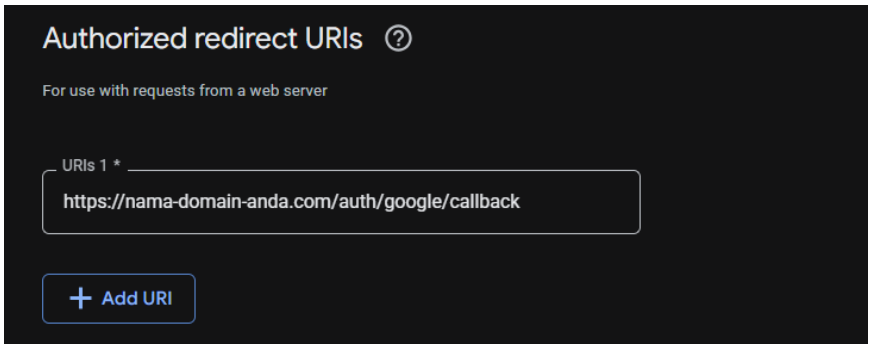


Gambar 3.6 Membuat *project* baru



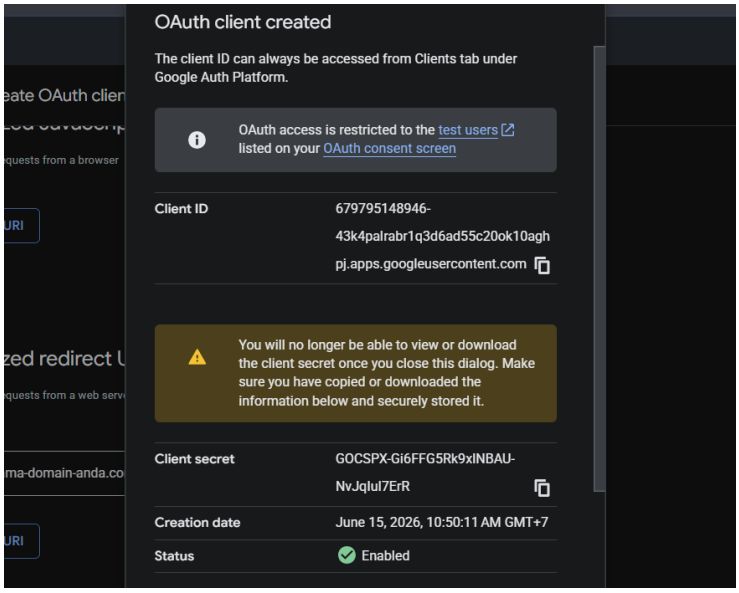
Gambar 3.7 *Project* berhasil dibuat

3. Masuk ke menu *APIs & Services* lalu menu *OAuth Consent Screen*.
4. Klik *Get Started* lalu isi informasi aplikasi seperti nama aplikasi, email pengembang, dan kontak dukungan. Untuk *Audience* pilih *External*
5. Simpan konfigurasi *OAuth Consent Screen* hingga selesai.
6. Buka menu *Client.*
7. Klik *Create Client* → *OAuth Client ID*.
8. Pilih tipe aplikasi *Web Application*.
9. Masukkan nama aplikasi, misalnya Hagaplus.
10. Tambahkan *Authorized Redire*ct URI berikut:



Gambar 3.8 Tambahkan *URL callback* yang digunakan

11. Klik *Create*
12. Google akan menampilkan *Client ID* dan *Client Secret.*



Gambar 3.9 *OAuth client details*

13. Masukkan nilai tersebut ke dalam *file* .env:

```
GOOGLE_CLIENT_ID=679795148946-43k4palrabr1q3d6ad55c20ok10aghpj.apps.googleusercontent.com
GOOGLE_CLIENT_SECRET=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
GOOGLE_REDIRECT_URI="${APP_URL}/auth/google/callback"
```

Gambar 3.10 Konfigurasi google *secret* pada *file* .env

e. Mengakses phpMyAdmin dan *Import Database* Secara Manual

Apabila *database* telah digunakan pada lingkungan pengembangan lokal dan ingin dipindahkan ke server hosting tanpa menjalankan migrasi Laravel, proses pemindahan dapat dilakukan menggunakan fitur ekspor dan impor *database* melalui phpMyAdmin.

Langkah pertama adalah mengekspor *database* yang digunakan pada lingkungan lokal menjadi *file* SQL. Jalankan layanan Apache dan MySQL pada Laragon, XAMPP, atau aplikasi server lokal yang digunakan.

1. Buka *browser*, kemudian akses alamat berikut: http://localhost/phpmyadmin
2. Pada panel sebelah kiri, pilih *database* yang digunakan oleh aplikasi, misalnya: hagaplus
3. Setelah *database* terbuka, klik tab *Export* yang berada pada bagian atas halaman phpMyAdmin.
4. Pada bagian *Export Method*, pilih opsi *Quick* agar seluruh struktur tabel dan data diekspor secara otomatis.
5. Pastikan format ekspor yang dipilih adalah SQL.
6. Klik tombol *Go*.
7. *Browser* akan mengunduh *file database* dengan ekstensi .sql, misalnya: hagaplus.sql
8. Simpan *file* tersebut karena akan digunakan pada proses impor ke server hosting.

Setelah *file* SQL berhasil diperoleh, langkah berikutnya adalah mengimpor *database* tersebut ke hosting.

1. *Login* ke panel hosting (cPanel, hPanel Hostinger, atau panel lainnya).
2. Buka menu phpMy*Admin*.
3. Pada panel sebelah kiri, pilih *database* yang telah dibuat sebelumnya.
4. Klik tab *Import* pada bagian atas halaman.
5. Pada bagian *File to Import*, klik tombol *Choose File*.
6. Pilih *file database* hasil ekspor sebelumnya, misalnya hagaplus.sql contoh:

Gambar 3.11 *Import database* ke phpmyadmin hosting panel

7. Biarkan pengaturan lainnya menggunakan nilai bawaan (*default*).
8. Klik tombol *Import* atau *Go* untuk memulai proses impor.
9. Tunggu hingga proses selesai. Jika berhasil, phpMyAdmin akan menampilkan pesan:

   "Import has been successfully finished."

   Pastikan seluruh tabel berhasil masuk ke *database* dengan memeriksa daftar tabel pada panel sebelah kiri phpMyAdmin.

f. Inisialisasi *Server* (*Generate Key*, Migrasi *Database*, & *Symlink)*

1. Metode A: Jika Hosting Mendukung SSH

   Hubungkan terminal SSH Anda ke server hosting, masuk ke direktori folder proyek Anda, lalu jalankan rangkaian per*int*ah berikut:

```
# Membuat Application Key baru jika masih kosong
php artisan key:generate

# Menjalankan migrasi database beserta data awal (seeder)
php artisan migrate --force --seed

# Membuat link folder storage ke folder publik untuk berkas upload
php artisan storage:link
```

2. Metode B: Jika Hosting Tidak Mendukung SSH

   Jika hosting Anda tidak menyediakan akses SSH, Anda dapat memicu per*int*ah artisan di atas dengan membuat *Route* sementara:

   a) Buka berkas routes/web.php menggunakan *File* Editor hosting.

   b) Tambahkan baris kode berikut pada bagian paling bawah file:

```
Route::get('/init-server', function() {
    // Generate key jika belum ada
    Artisan::call('key:generate');

    // Jalankan migrasi database + seeder
     Artisan::call('migrate', ['--force' => true, '--seed' =>
true]);

    // Buat storage link
```

```
    Artisan::call('storage:link');

    return "Server initialization success!";
});
```

c) Akses alamat URL https://nama-domain-anda.com/init-server satu kali melalui *web browser* Anda

d) PENTING: Demi menjaga keamanan sistem, segera hapus kembali kode *Route* tersebut dari berkas routes/web.php setelah proses inisialisasi selesai.

g. Konfigurasi *Cron Job untuk Scheduler & Queue*

Karena *shared* hosting membatasi jalannya background *process* yang terus-menerus, Anda perlu menggunakan fitur *Cron Job* pada panel hosting Anda:

1. Masuk ke halaman dashboard panel hosting Anda dan cari menu *Cron Jobs.*
2. Tambahkan *Cron Job* baru untuk Laravel *Scheduler agar* berjalan setiap menit:

```
* * * * * cd /path/ke/proyek/anda && php artisan schedule:run >> /dev/null 2>&1
```

3. Tambahkan *Cron Job* baru untuk *Queue Worker* agar memproses antrean data absensi/penggajian setiap menit:

```
* * * * * cd /path/ke/proyek/anda && php artisan queue:work --once >> /dev/null 2>&1
```

(Silakan sesuaikan /path/ke/proyek/anda dengan alamat folder absolut proyek Anda di hosting, seperti /home/username/hagaplus atau /home/username/public_html).

3.3 Rincian Perhitungan Penggajian

Perhitungan penggajian pada sistem HagaPlus dilakukan berdasarkan komponen pendapatan dan potongan yang dimiliki oleh masing-masing karyawan. Komponen pendapatan dapat meliputi gaji pokok, tunjangan, uang lembur, bonus, dan komponen penghasilan lainnya, sedangkan komponen potongan dapat meliputi iuran jaminan sosial, Pajak Penghasilan (PPh) Pasal 21, serta potongan lainnya sesuai dengan data penggajian karyawan. Perhitungan tersebut digunakan untuk

memperoleh total penghasilan, total potongan, dan gaji bersih (*take home pay*) karyawan.

Khusus untuk perhitungan PPh Pasal 21, sistem HagaPlus mengacu pada Peraturan Pemer*int*ah Nomor 58 Tahun 2023, Peraturan Menteri Keuangan Nomor 168 Tahun 2023, serta panduan *Cermat Pemotongan PPh Pasal 21/26* yang diterbitkan oleh Direktorat Jenderal Pajak pada tahun 2024. Panduan tersebut menjelaskan mekanisme pemotongan PPh Pasal 21, termasuk penggunaan Tarif Efektif Rata-Rata (TER) untuk pegawai tetap pada masa pajak selain masa pajak terakhir dan penghitungan pada masa pajak terakhir. Panduan DJP juga menjelaskan bahwa ketentuan tersebut ditujukan untuk memberikan kemudahan dan kesederhanaan dalam proses penghitungan PPh Pasal 21.

3.3.1 Rumus Perhitungan Penggajian

Perhitungan penggajian pada HagaPlus menggunakan beberapa komponen perhitungan sebagai berikut.

a. Total Penghasilan

Total penghasilan merupakan jumlah seluruh komponen pendapatan yang diterima oleh karyawan pada periode penggajian.

Rumus:

Total Penghasilan = Gaji Pokok + Tunjangan + Lembur + Bonus + Reimbursement + THR + Komponen Penghasilan Lainnya

Komponen yang digunakan dapat berbeda pada setiap karyawan sesuai dengan data dan pengaturan penghasilan yang terdapat pada sistem.

b. Perhitungan Waktu dan Uang Lembur

Waktu lembur yang tercatat dalam sistem berupa satuan menit kemudian dikonversi menjadi jam.

Rumus:

Jam Lembur = Total Menit Lembur ÷ 60

Selanjutnya, nilai lembur dihitung berdasarkan tarif lembur yang digunakan pada sistem.

Uang Lembur = Jam Lembur × Tarif Lembur per Jam

Sebagai contoh, pada sampel Eko Pratama terdapat waktu lembur sebesar 255 menit. Sistem mengonversinya menjadi 4,25 jam dan menghasilkan uang lembur sebesar Rp85.000.

PT Hagaplus Solusi Digital
Landmark Tower, Jl. Jend. Gatot Subroto Kav. 52
Telp: 02180863534

SLIP GAJI
Periode: Nov 2025

ID Karyawan: MKT22005
Nama: Eko Pratama
Organization: Marketing

Jabatan: Officer
PTKP: TK/0
NPWP: -

Grade / Level: Staff
Tanggal Masuk: 17 May 2022

Pendapatan

Gaji Pokok	1.500.000
Tunjangan Jabatan	0
Tunjangan Makan	50.000
Tunjangan Transport	100.000
Lembur	85.000
Bonus	0
Reimburse	0
THR	0
Tunjangan Komunikasi	150.000
Total Pendapatan	**1.885.000**

Potongan

BPJS Kesehatan	15.000
BPJS Ketenagakerjaan	45.000
PPh21	0
Potongan Absensi	139.000
Kasbon	0
Potongan Lainnya	0
Iuran Paguyuban	20.000
Total Potongan	**219.000**

Take Home Pay	**1.666.000**

Attendance Summary

Actual Working Day	19
Schedule Working Day	0
Absent	1
Late (minutes)	39
Overtime (minutes)	255

Mengetahui,
Manager

Gambar 3.12 Rincian perhitungan waktu dan uang lembur pada penggajian Eko Pratama periode November 2025.

c. Perhitungan Iuran BPJS

Iuran BPJS yang menjadi tanggungan karyawan dihitung berdasarkan dasar perhitungan dan tarif masing-masing program.

Rumus umum:

Iuran BPJS = Dasar Perhitungan × Tarif Iuran

Potongan

BPJS Kesehatan	15.000
BPJS Ketenagakerjaan	45.000
PPh21	0
Potongan Absensi	139.000
Kasbon	0
Potongan Lainnya	0
Iuran Paguyuban	20.000
Total Potongan	**219.000**

Gambar 3.13 Rincian komponen potongan gaji Eko Pratama

Gambar 3.14 Tarif Iuran BPJS yang Digunakan dalam Perhitungan Penggajian

Pada sampel Eko Pratama, sistem menampilkan iuran BPJS Kesehatan sebesar 1% dari gaji pokok, JHT sebesar 2%, dan JP sebesar 1%. Total iuran BPJS yang menjadi potongan karyawan adalah Rp60.000.

d. **Perhitungan PPh Pasal 21 Bulanan**

Untuk periode Januari sampai November, sistem menggunakan metode Tarif Efektif Rata-Rata (TER). Tarif yang digunakan ditentukan berdasarkan status PTKP dan besarnya penghasilan bruto bulanan.

Rumus:

PPh Pasal 21 = Penghasilan Bruto Bulanan × Tarif TER

Pada sampel Eko Pratama, status PTKP adalah TK/0 sehingga termasuk dalam kategori TER A. Dengan penghasilan bruto sebesar Rp1.885.000, tarif TER yang digunakan adalah 0%, sehingga PPh Pasal 21 yang dipotong sebesar Rp0.

e. **Perhitungan PPh Pasal 21 Masa Pajak Terakhir**

Pada masa pajak terakhir, sistem melakukan penghitungan kembali PPh Pasal 21 berdasarkan penghasilan selama satu tahun. Nilai PPh Pasal 21 yang telah dipotong pada periode sebelumnya kemudian diperhitungkan untuk memperoleh nilai PPh Pasal 21 pada masa pajak terakhir.

Rumus:

PPh Pasal 21 Masa Pajak Terakhir = PPh Pasal 21 Terutang Setahun − PPh Pasal 21 yang Telah Dipotong pada Masa Sebelumnya.

PPh21 — Metode Koreksi Tahunan (Desember) (Pasal 17 UU PPh)

Status PTKP: TK/0 → PTKP = Rp 54.000.000
Biaya Jabatan (5%, maks Rp 6.000.000) = Rp 1.117.400
Netto Setahun = Rp 20.690.600
PPh21 Setahun (Pasal 17) = Rp 0

Bruto Bulanan × 12 = Rp 22.348.000
Iuran Pensiun Setahun = Rp 540.000
PKP (Netto − PTKP) = Rp 0
Sudah dipotong Jan–Nov = Rp 0

Lihat Riwayat Bruto & Potongan Jan-Nov

Bulan	Bruto	PPh21 (TER)
January	Rp 1.800.000	Rp 0
February	Rp 1.800.000	Rp 0
March	Rp 1.990.000	Rp 0
April	Rp 1.945.333	Rp 0
June	Rp 1.969.000	Rp 0

PPh21 Desember = Rp 0 − Rp 0 = Rp 0

Gambar 3.15 Perhitungan PPh Pasal 21 Masa Pajak Terakhir Gaji Eko Pratama periode Desember 2025

f. Penghasilan Neto dan Penghasilan Kena Pajak

Dalam penghitungan tahunan, sistem menghitung penghasilan neto dengan memperhitungkan penghasilan bruto dan komponen pengurang yang digunakan pada perhitungan.

Penghasilan Neto = Penghasilan Bruto − Total Pengurang

Selanjutnya:

PKP = Penghasilan Neto − PTKP

Nilai PKP kemudian digunakan dalam penghitungan PPh Pasal 21 tahunan berdasarkan lapisan tarif yang berlaku.

g. Gaji Bersih

Gaji bersih atau *take home pay* merupakan penghasilan yang diterima karyawan setelah dikurangi seluruh komponen potongan.

Gaji Bersih = Total Penghasilan − Total Potongan

Komponen potongan dapat terdiri atas iuran BPJS karyawan, PPh Pasal 21, potongan ketidakhadiran, iuran paguyuban, dan potongan lain yang diterapkan pada penggajian.

3.3.2 Sampel Perhitungan PPh Pasal 21 Nihil

Sampel pertama menggunakan data karyawan Eko Pratama dengan status PTKP TK/0. Pada periode November 2025, karyawan memiliki gaji pokok sebesar Rp1.500.000, tunjangan makan Rp50.000, tunjangan transport Rp100.000, uang lembur Rp85.000, dan tunjangan komunikasi Rp150.000. Total penghasilan yang diperoleh adalah Rp1.885.000.

Status PTKP: **TK/0** → Kategori TER: **A**

Kategori A: TK/0, TK/1, K/0 | B: TK/2, TK/3, K/1, K/2 | C: K/3

Komponen Kena Pajak (Penambah Bruto):

Gaji Pokok	Rp 1.500.000
Tunjangan Makan	Rp 50.000
Tunjangan Transport	Rp 100.000
Lembur	Rp 85.000
Tunjangan Komunikasi	Rp 150.000

Total Penghasilan Bruto Sebulan:

Rp 1.885.000

Tarif TER Kategori A pada bruto ini:

0.00%

Lihat Tabel TER Kategori A

Penghasilan Bruto Bulanan	Tarif
s.d. Rp 5.400.000 ✓ BRUTO ANDA	0%
Rp 5.400.000 - Rp 5.650.000	0,25%
Rp 5.650.000 - Rp 5.950.000	0,5%
Rp 5.950.000 - Rp 6.300.000	0,75%
Rp 6.300.000 - Rp 6.750.000	1%
Rp 6.750.000 - Rp 7.500.000	1,25%
Rp 7.500.000 - Rp 8.550.000	1,5%
Rp 8.550.000 - Rp 9.650.000	1,75%
Rp 9.650.000 - Rp 10.050.000	2%
Rp 10.050.000 - Rp 10.350.000	2,25%

PPh21 = 0.00% × Rp 1.885.000 = Rp 0

Gambar 3.16 Perhitungan PPh21 Gaji Eko Pratama periode November 2025

Berdasarkan penghasilan bruto sebesar Rp1.885.000 dan status PTKP TK/0, karyawan termasuk dalam kategori TER A. Penghasilan tersebut berada pada lapisan penghasilan sampai dengan Rp5.400.000 dengan tarif TER sebesar 0%. Oleh karena itu, PPh Pasal 21 pada periode November adalah:

PPh 21 = Rp1.885.000 × 0%

PPh 21 = Rp0

Sistem juga menghitung potongan BPJS karyawan sebesar Rp60.000 yang terdiri atas BPJS Kesehatan Rp15.000, JHT Rp30.000, dan JP Rp15.000. Setelah memperhitungkan potongan ketidakhadiran sebesar Rp139.000 dan iuran paguyuban sebesar Rp20.000, total potongan menjadi Rp219.000 sehingga gaji bersih yang diterima sebesar Rp1.666.000.

3.3.3 Perhitungan PPh Pasal 21 Desember

Pada periode Desember, sistem melakukan penghitungan PPh Pasal 21 tahunan. Sampel Eko memiliki penghasilan bruto setahun sebesar Rp22.348.000.

Setelah memperhitungkan biaya jabatan sebesar Rp1.117.400 dan iuran pensiun/BPJS TK sebesar Rp540.000, diperoleh penghasilan neto sebesar Rp20.690.600. Nilai tersebut masih berada di bawah PTKP sebesar Rp54.000.000 sehingga PKP yang diperoleh adalah Rp0.

PPh21 — Metode Koreksi Tahunan (Desember) (Pasal 17 UU PPh)

Status PTKP: **TK/0** → PTKP = **Rp 54.000.000**
Biaya Jabatan (5%, maks Rp 6.000.000) = **Rp 1.117.400**
Netto Setahun = **Rp 20.690.600**
PPh21 Setahun (Pasal 17) = **Rp 0**

Bruto Bulanan × 12 = **Rp 22.348.000**
Iuran Pensiun Setahun = **Rp 540.000**
PKP (Netto − PTKP) = **Rp 0**
Sudah dipotong Jan–Nov = **Rp 0**

Lihat Riwayat Bruto & Potongan Jan-Nov

Bulan	Bruto	PPh21 (TER)
January	Rp 1.800.000	Rp 0
February	Rp 1.800.000	Rp 0
March	Rp 1.990.000	Rp 0
April	Rp 1.945.333	Rp 0
June	Rp 1.969.000	Rp 0

PPh21 Desember = Rp 0 − Rp 0 = Rp 0

Gambar 3.17 Perhitungan PPh21 Gaji Eko Pratama periode Desember 2025

Dengan demikian:

PKP = Rp20.690.600 − Rp54.000.000

PKP = Rp0

Karena PKP bernilai Rp0, maka PPh Pasal 21 terutang selama satu tahun juga sebesar Rp0. Sistem kemudian menghitung PPh Pasal 21 Desember sebagai selisih antara PPh Pasal 21 setahun dan pajak yang telah dipotong pada Januari sampai November. Karena keduanya bernilai Rp0, PPh Pasal 21 Desember juga sebesar Rp0.

3.3.4 Sampel Perhitungan PPh Pasal 21 Tidak Nihil

Sampel kedua menggunakan data karyawan Lukman Muludin dengan status PTKP TK/0. Pada periode November 2025, karyawan memiliki gaji pokok Rp8.500.000, tunjangan jabatan Rp2.000.000, tunjangan makan Rp750.000, tunjangan transport Rp750.000, uang lembur Rp21.667, dan tunjangan komunikasi Rp150.000. Total penghasilan bruto bulan tersebut adalah Rp12.171.667.

PT Hagaplus Solusi Digital
Landmark Tower, Jl. Jend. Gatot Subroto Kav. 52
Telp: 02180863534

SLIP GAJI
Periode: Nov 2025

ID Karyawan: OPS26001
Nama: Lukman Muludin
Organization: Operations

Jabatan: Team Leader
PTKP: TK/0
NPWP: 12.345.678.9-7772.123

Grade / Level: Supervisor
Tanggal Masuk: 01 Jan 2026

Pendapatan

Gaji Pokok	8.500.000
Tunjangan Jabatan	2.000.000
Tunjangan Makan	750.000
Tunjangan Transport	750.000
Lembur	21.667
Bonus	0
Reimburse	0
THR	0
Tunjangan Komunikasi	150.000
Total Pendapatan	**12.171.667**

Potongan

BPJS Kesehatan	85.000
BPJS Ketenagakerjaan	170.000
PPh21	486.867
Potongan Absensi	36.000
Kasbon	0
Potongan Lainnya	0
Iuran Paguyuban	20.000
Total Potongan	**797.867**

Take Home Pay	**11.373.800**

Gambar 3.18 Rincian penghasilan dan potongan Lukman Muludin periode November 2025.

Penghasilan bruto sebesar Rp12.171.667 berada pada lapisan TER kategori A sebesar 4%. Dengan demikian, PPh Pasal 21 bulan November dihitung sebagai berikut:

PPh 21 = Rp12.171.667 × 4%

PPh 21 = Rp486.867

Nilai tersebut sesuai dengan hasil perhitungan yang ditampilkan sistem.

3.3.5 Perhitungan PPh Pasal 21 Desember

Pada masa pajak Desember, sistem melakukan koreksi tahunan. Berdasarkan data penghasilan, sistem menghasilkan bruto setahun sebesar Rp149.308.000, biaya jabatan sebesar Rp6.000.000, dan iuran pensiun/BPJS TK sebesar Rp2.040.000. Penghasilan neto yang diperoleh adalah Rp141.268.000. Setelah dikurangi PTKP sebesar Rp54.000.000, diperoleh PKP sebesar Rp87.268.000.

PKP sebesar Rp87.268.000 masuk ke dua lapisan tarif, yaitu:

Rp60.000.000 × 5% = Rp3.000.000

Rp27.268.000 × 15% = Rp4.090.200

Sehingga:

PPh 21 Setahun = Rp3.000.000 + Rp4.090.200

PPh 21 Setahun = Rp7.090.200

Selanjutnya, sistem menjumlahkan PPh Pasal 21 yang telah dipotong pada Januari sampai November sebesar Rp4.893.227. Dengan demikian, PPh Pasal 21 yang dipotong pada Desember adalah:

PPh 21 Desember = Rp7.090.200 − Rp4.893.227

PPh 21 Desember = Rp2.196.973

PPh21 — Metode Koreksi Tahunan (Desember) (Pasal 17 UU PPh)

Status PTKP: **TK/0** → PTKP = **Rp 54.000.000**
Biaya Jabatan (5%, maks Rp 6.000.000) = **Rp 6.000.000**
Netto Setahun = **Rp 141.268.000**
PPh21 Setahun (Pasal 17) = **Rp 7.090.200**

Bruto Bulanan × 12 = **Rp 149.308.000**
Iuran Pensiun Setahun = **Rp 2.040.000**
PKP (Netto − PTKP) = **Rp 87.268.000**
Sudah dipotong Jan–Nov = **Rp 4.893.227**

Lihat Riwayat Bruto & Potongan Jan-Nov

Bulan	Bruto	PPh21 (TER)
July	Rp 12.283.333	Rp 491.333
August	Rp 12.228.333	Rp 489.133
September	Rp 12.270.667	Rp 490.827
October	Rp 12.310.667	Rp 492.427
November	Rp 12.171.667	Rp 486.867

PPh21 Desember = Rp 7.090.200 − Rp 4.893.227 = Rp 2.196.973

Gambar 3.19 Rincian Perhitungan PPh21 Gaji Lukman Muludin periode Desember 2026

Hasil tersebut kemudian ditampilkan sebagai komponen potongan pada penggajian Desember.

PT Hagaplus Solusi Digital
Landmark Tower, Jl. Jend. Gatot Subroto Kav. 52
Telp: 02180863534

SLIP GAJI
Periode: Dec 2025

ID Karyawan: OPS26001
Nama: Lukman Muludin
Organization: Operations

Jabatan: Team Leader
PTKP: TK/0
NPWP: 12.345.678.9-7772.123

Grade / Level: Supervisor
Tanggal Masuk: 01 Jan 2026

Pendapatan

Gaji Pokok	8.500.000
Tunjangan Jabatan	2.000.000
Tunjangan Makan	750.000
Tunjangan Transport	750.000
Lembur	292.333
Bonus	0
Reimburse	0
THR	0
Tunjangan Komunikasi	150.000
Total Pendapatan	**12.442.333**

Potongan

BPJS Kesehatan	85.000
BPJS Ketenagakerjaan	170.000
PPh21	2.196.973
Potongan Absensi	76.000
Kasbon	0
Potongan Lainnya	0
Iuran Paguyuban	20.000
Total Potongan	**2.547.973**

Take Home Pay	**9.894.360**

Gambar 3.20 Rincian perhitungan penggajian dan koreksi PPh Pasal 21 masa pajak terakhir pada Lukman Muludin.

3.3.6 Formulir 1721-A1

Selain menghasilkan rincian perhitungan penggajian, sistem juga menyediakan Formulir 1721-A1 sebagai dokumen bukti pemotongan PPh Pasal 21 bagi pegawai tetap. Formulir tersebut memuat identitas penerima penghasilan, identitas pemotong, rincian penghasilan bruto, pengurangan, penghasilan neto, PTKP, PKP, serta PPh Pasal 21 terutang.

a r e a s t a p l e s

KEMENTERIAN KEUANGAN RI
DIREKTORAT JENDERAL PAJAK

BUKTI PEMOTONGAN PAJAK PENGHASILAN PASAL 21 BAGI PEGAWAI TETAP ATAU PENERIMA PENSIUN ATAU TUNJANGAN HARI TUA/JAMINAN HARI TUA BERKALA

NOMOR : H.01 1.1-12.25-0000001

FORMULIR 1721 - A1

Lembar ke-1 : untuk Penerima Penghasilan
Lembar ke-2 : untuk Pemotong

MASA PEROLEHAN PENGHASILAN
01 - 12

NPWP PEMOTONG : H.03 01.000.013.1-093.000
NAMA PEMOTONG : H.04 PT Hagaplus Solusi Digital

A. IDENTITAS PENERIMA PENGHASILAN YANG DIPOTONG

1.	NPWP	12.345.678.9-7772.123	**6. STATUS /JUMLAH TANGGUNGAN KELUARGA UNTUK PTKP**
2.	NIK/NO. PASPOR	3271053007772123	K / **TK / 0** HB /
3.	NAMA	Lukman Muludin	7. NAMA JABATAN : Team Leader
4.	ALAMAT	Serdang Wetan, Kec. Legok, Kabupaten Tangerang, Banten 15820	8. KARYAWAN ASING : **TIDAK**
5.	JENIS KELAMIN	**PEREMPUAN**	9. KODE NEGARA DOMISILI : **ID**

B. RINCIAN PENGHASILAN DAN PENGHITUNGAN PPh PASAL 21

	URAIAN	JUMLAH (Rp)
	KODE OBJEK PAJAK : [X] 21-100-01 [ ] 21-100-02	
	PENGHASILAN BRUTO:	
1	GAJI/PENSIUN ATAU THT/JHT	132.000.000
2	TUNJANGAN PPh	0
3	TUNJANGAN LAINNYA, UANG LEMBUR DAN SEBAGAINYA	1.123.000
4	HONORARIUM DAN IMBALAN LAIN SEJENISNYA	0
5	PREMI ASURANSI YANG DIBAYAR PEMBERI KERJA	0
6	PENERIMAAN DALAM BENTUK NATURA DAN KENIKMATAN LAINNYA YANG DIKENAKAN PEMOTONGAN PPh PASAL 21	0
7	TANTIEM, BONUS, GRATIFIKASI, JASA PRODUKSI DAN THR	0
8	**JUMLAH PENGHASILAN BRUTO (1 S.D 7)**	**133.123.000**
	PENGURANGAN:	
9	BIAYA JABATAN/ BIAYA PENSIUN	6.000.000
10	IURAN PENSIUN ATAU IURAN THT/JHT	2.805.000
11	**JUMLAH PENGURANGAN (9 S.D 10)**	**8.805.000**
	PENGHITUNGAN PPh PASAL 21:	
12	JUMLAH PENGHASILAN NETO (8-11)	124.318.000
13	PENGHASILAN NETO MASA SEBELUMNYA	0
14	JUMLAH PENGHASILAN NETO UNTUK PENGHITUNGAN PPh PASAL 21 (SETAHUN/DISETAHUNKAN)	124.318.000
15	PENGHASILAN TIDAK KENA PAJAK (PTKP)	54.000.000
16	PENGHASILAN KENA PAJAK SETAHUN/DISETAHUNKAN (14 - 15)	70.318.000
17	PPh PASAL 21 ATAS PENGHASILAN KENA PAJAK SETAHUN/DISETAHUNKAN	4.547.700
18	PPh PASAL 21 YANG TELAH DIPOTONG MASA SEBELUMNYA	0
19	**PPh PASAL 21 TERUTANG**	**4.547.700**
20	PPh PASAL 21 DAN PPh PASAL 26 YANG TELAH DIPOTONG DAN DILUNASI	4.547.700

C. IDENTITAS PEMOTONG

1. NPWP : 01.000.013.1-093.000
2. NAMA : PT Hagaplus Solusi Digital

3. TANGGAL & TANDA TANGAN

12-08-2026

Gambar 3.21 Formulir 1721-A1 yang dihasilkan oleh sistem HagaPlus.

3.4 Cara Penggunaan

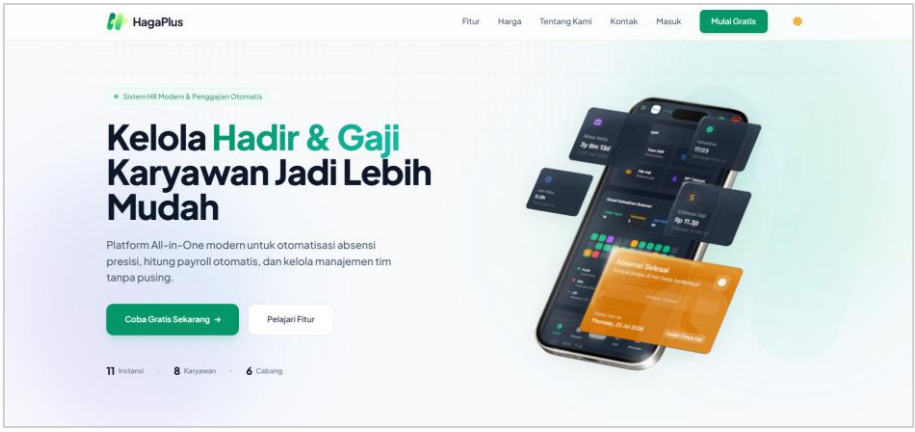
Setelah proses implementasi dan instalasi sistem selesai dilakukan, tahap selanjutnya adalah penggunaan sistem HagaPlus oleh pengguna sesuai dengan hak akses yang dimiliki. Sistem HagaPlus dirancang dengan antarmuka yang sederhana dan mudah digunakan sehingga setiap pengguna dapat mengakses fitur-fitur yang tersedia sesuai dengan perannya masing-masing.

Pada sistem HagaPlus terdapat tiga jenis pengguna, yaitu *Superadmin*, *Admin*/HR, dan Karyawan. Masing-masing pengguna memiliki menu dan fungsi yang berbeda sesuai dengan kebutuhan operasional. Pada bagian ini dijelaskan fungsi setiap menu utama beserta tampilan antarmuka sistem sebagai gambaran penggunaan aplikasi dalam mendukung proses pengelolaan sumber daya manusia.

3.4.1 Akses Awal Sistem

a. *Landing Page*

Landing Page merupakan p*int*u gerbang utama informasi sistem HagaPlus yang dapat diakses oleh publik secara bebas. Halaman ini menyajikan informasi fitur unggulan, paket harga berlangganan yang transparan, informasi kontak, serta statistik riil mengenai jumlah instansi, karyawan, dan cabang aktif. Pengguna juga dapat melihat pengumuman atau promo terbaru melalui *Announcement Carousel*.

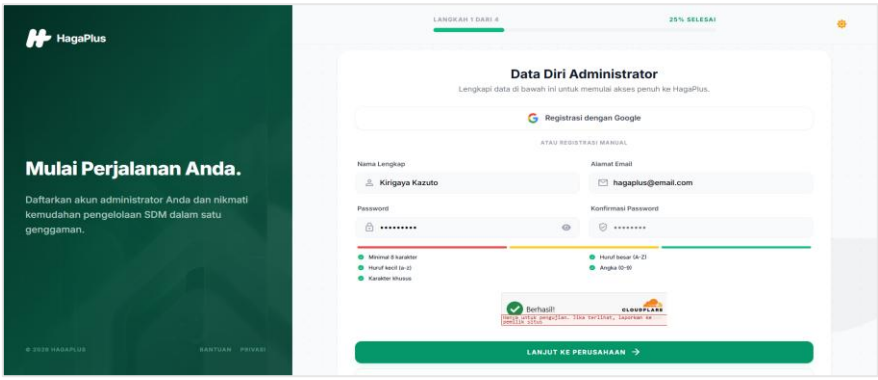


Gambar 3.22 *Landing Page*

b. Fitur *Register*

Fitur *Register* membantu calon pengguna untuk mendaftarkan akun baru melalui proses *multi-step wizard* sebanyak empat langkah terstruktur. Langkah pertama meliputi pengisian data diri *administrator*

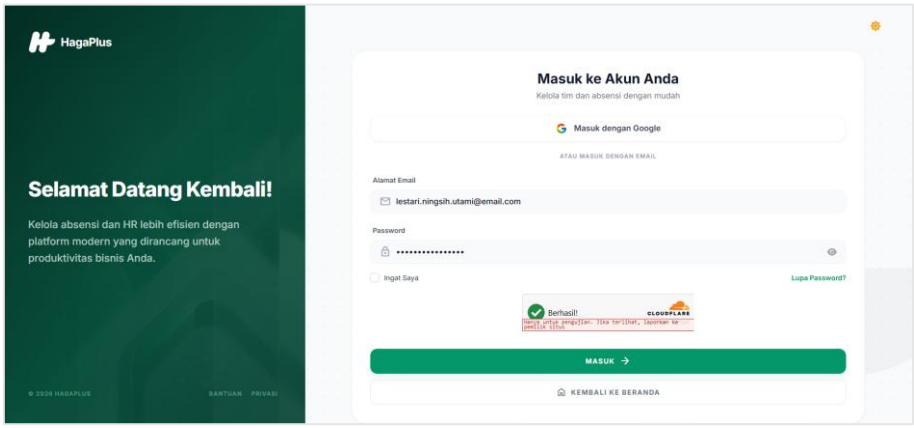
atau menghubungkan langsung melalui *Google Auth*. Langkah kedua dilanjutkan dengan pengisian profil bisnis seperti nama instansi dan alamat kantor. Pada langkah ketiga dan keempat, pendaftar dapat memilih paket langganan yang sesuai, menerapkan kode promo jika ada, memilih metode pembayaran melalui transfer bank atau QRIS, serta mengunggah bukti pembayaran untuk diverifikasi oleh *Superadmin*.



Gambar 3.23 Halaman *Register*

c. Halaman *Login*

Halaman *Login* digunakan oleh pengguna yang telah memiliki akun terdaftar untuk masuk ke dalam sistem. Akun terdaftar tersebut meliputi peran sebagai *Superadmin*, *Admin* Instansi atau HR, maupun Karyawan. Opsi masuk cepat menggunakan *Google Auth*, serta pengaman tambahan berupa *Cloudflare Turnstile Captcha* untuk mencegah serangan *spam*. Setelah proses autentikasi berhasil, sistem akan mendeteksi peran pengguna secara otomatis dan mengarahkannya ke *dashboard* masing-masing.

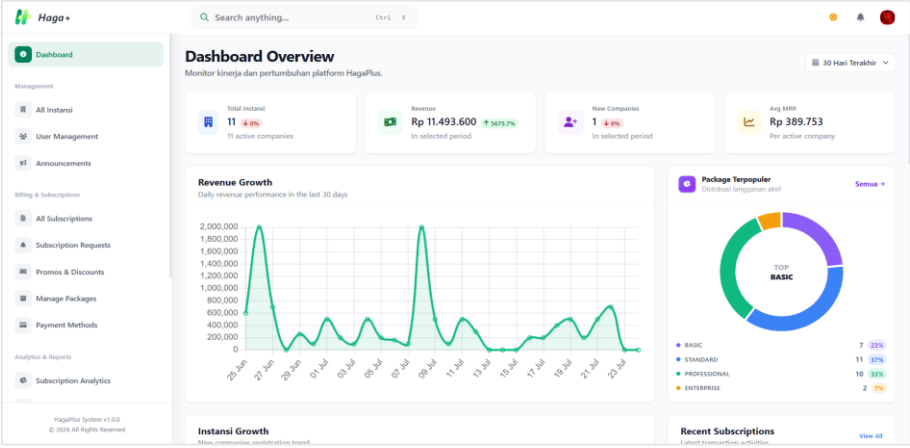


Gambar 3.24 Halaman *Login*

3.4.2 Penggunaan Sistem oleh *Superadmin*

a. Halaman *Dashboard*

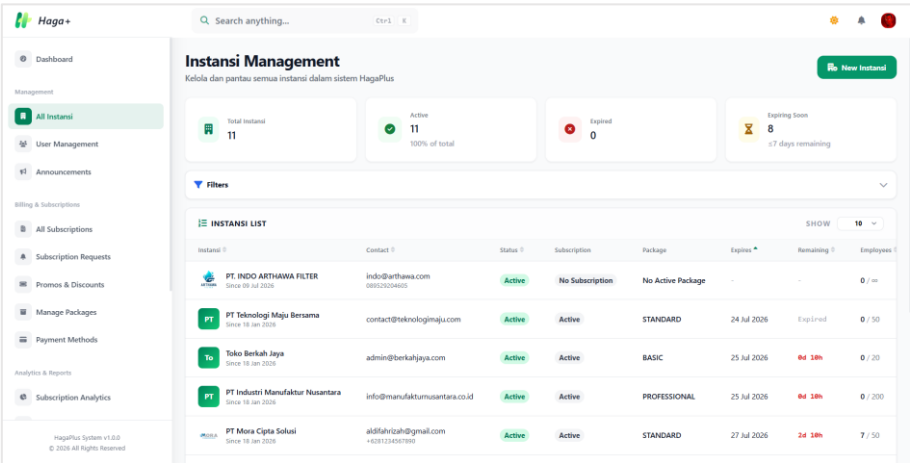
Dashboard Superadmin menyediakan ringkasan eksekutif berskala makro mengenai kondisi ekosistem HagaPlus secara keseluruhan. Di halaman ini, *Superadmin* dapat memantau indikator kinerja utama seperti total pendapatan bulanan, jumlah instansi yang aktif atau ditangguhkan, total paket langganan yang terjual, serta daftar aktivitas transaksi terbaru..



Gambar 3.25 Halaman *Dashboard Superadmin*

b. Halaman Instansi

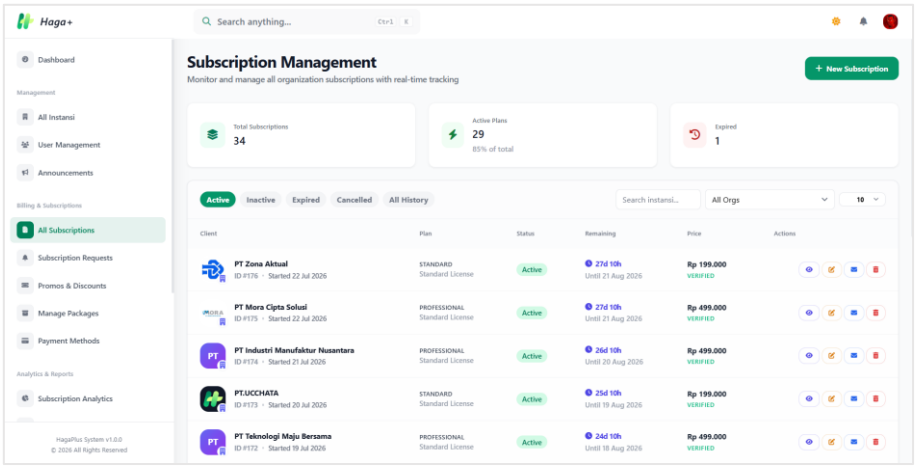
Menu Instansi memungkinkan *Superadmin* untuk mengelola seluruh profil perusahaan yang menggunakan platform HagaPlus. *Superadmin* memiliki wewenang penuh untuk melihat rincian detail setiap instansi dan memperbarui informasi kontak instansi.



Gambar 3.26 Halaman Instansi *Management*

c. Halaman *Subscription*

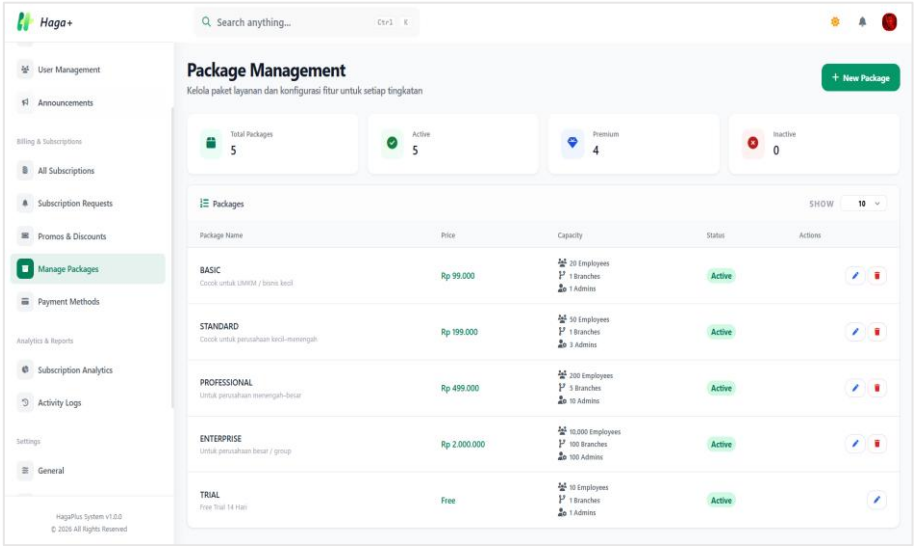
Menu *Subscription* digunakan untuk melacak dan mengelola seluruh riwayat langganan instansi yang ada di *platform*. *Superadmin* dapat memantau masa aktif paket tiap instansi.



Gambar 3.27 Halaman *Subscription Management*

d. Halaman *Package*

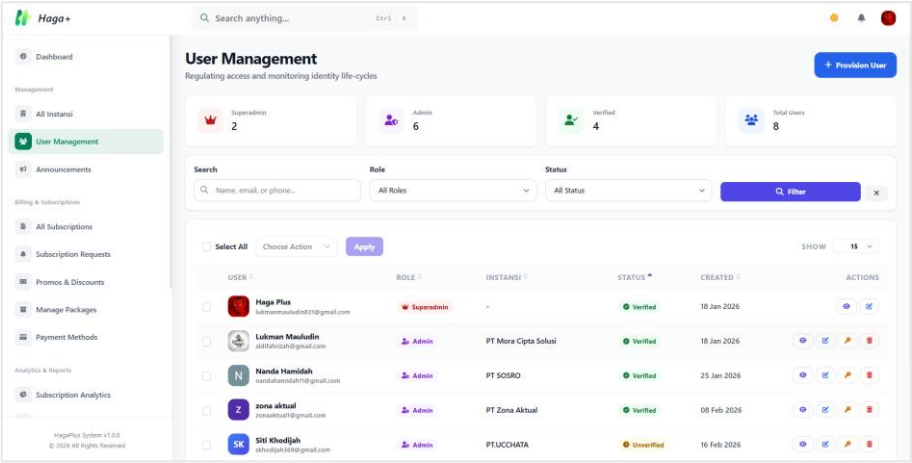
Menu *Package* memberikan kendali kepada *Superadmin* untuk merancang dan memodifikasi paket-paket langganan yang ditawarkan kepada calon instansi. Di menu ini, *Superadmin* bisa menentukan batas kapasitas maksimal karyawan, masa aktif paket, tarif harga dasar, serta menerapkan diskon promo aktif yang bersifat publik maupun privat.



Gambar 3.28 Halaman *Package Management*

e. Halaman *Users*

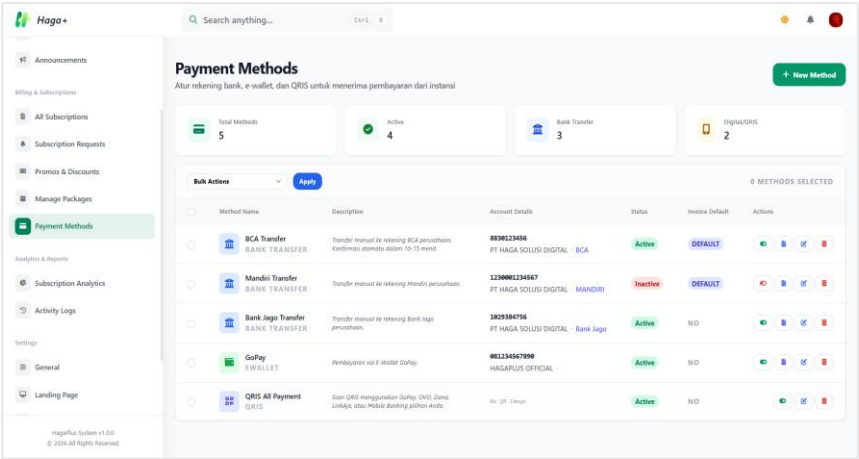
Menu *Users* merupakan tempat pengelolaan data akun pengguna yang memiliki hak akses di dalam sistem, baik *administrator* instansi maupun staf *int*ernal *Superadmin*. *Superadmin* dapat menambahkan pengguna baru, memperbarui informasi profil, mereset kata sandi, mengubah hak akses.



Gambar 3.29 Halaman *User Management*

f. Halaman *Payment Method*

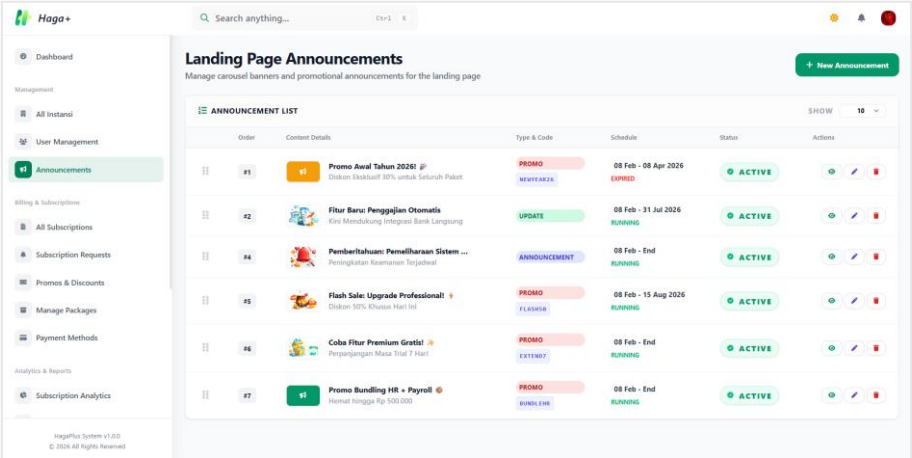
Menu *Payment Method* berfungsi untuk mengatur rekening penerima pembayaran langganan yang sah di HagaPlus. *Superadmin* dapat mendaftarkan berbagai metode pembayaran seperti transfer bank manual dengan menyertakan nama pemilik dan nomor rekening, atau mengunggah data kode QRIS statis.



Gambar 3.30 Halaman *Payment Methods*

g. Halaman *Announcement*

Menu *Announcement* digunakan oleh *Superadmin* untuk membuat dan menyebarkan pengumuman atau promo yang akan tampil pada *Landing Page* utama HagaPlus..

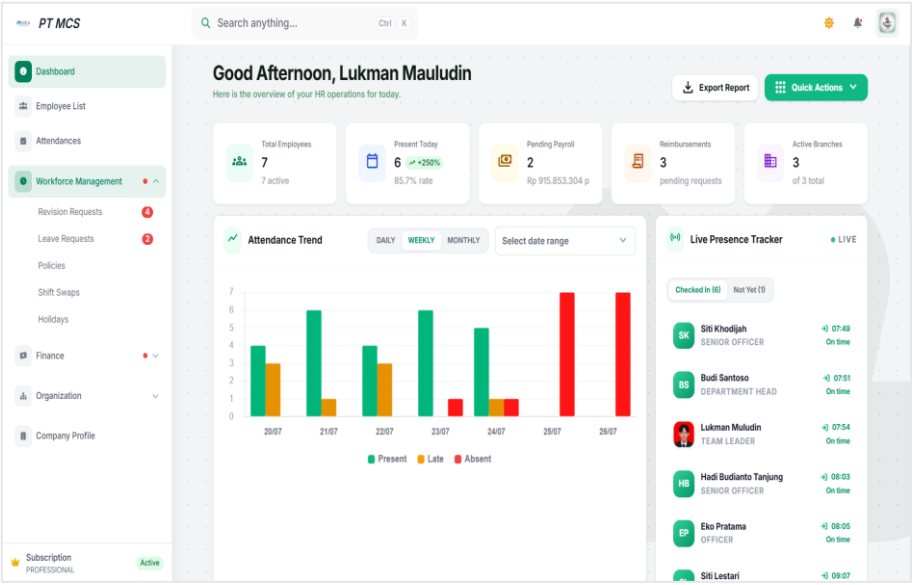


Gambar 3.31 Halaman *Landing Page Announcement*

3.4.3 Penggunaan Sistem oleh *Admin*/HR

a. Halaman *Dashboard Admin*

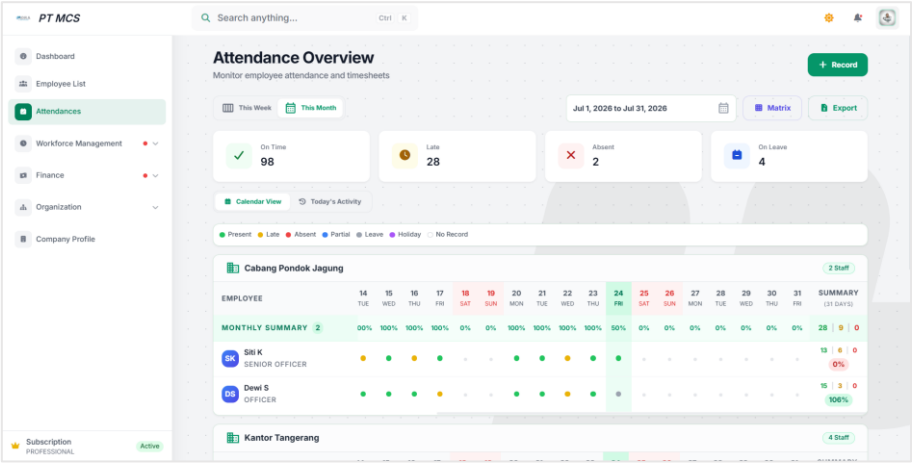
Dashboard *Admin*/HR menyajikan panel informasi operasional harian instansi secara terpusat untuk mempermudah pemantauan staf. Panel ini menampilkan statistik kehadiran hari ini seperti karyawan yang tepat waktu, terlambat, dan tidak hadir.



Gambar 3.32 Halaman *Dashboard Admin*

b. Halaman *Manage Attendance* (Absensi)

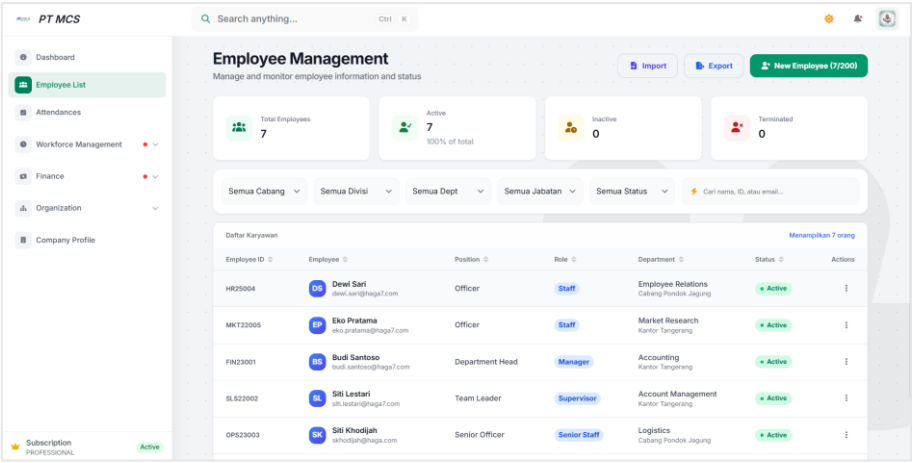
Menu *Manage Attendance* memuat seluruh rekaman *log* absensi harian karyawan instansi secara terperinci. *Admin*/HR dapat memvalidasi absensi yang dikirimkan karyawan saat masuk atau pulang kerja, mengunduh laporan kehadiran bulanan, serta mengekspor data absensi ke format Excel untuk kebutuhan audit.



Gambar 3.33 Halaman *Manage Attendance* (Absensi)

c. Halaman *Employee*

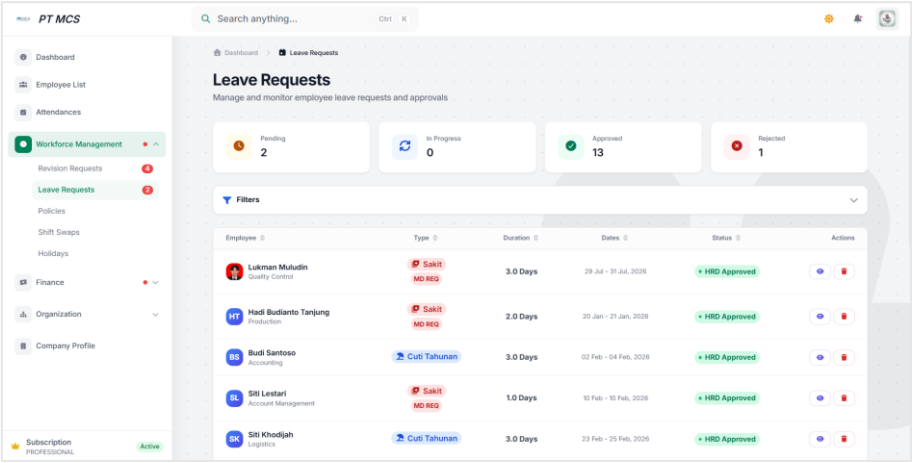
Menu *Employee* digunakan untuk mengelola data seluruh staf dan karyawan di perusahaan. HR dapat menambahkan karyawan baru secara manual maupun impor massal, menetapkan jenis jam kerja, menentukan besaran gaji pokok beserta tunjangan tetap, serta menonaktifkan akun karyawan yang sudah tidak bekerja di perusahaan.



Gambar 3.34 Halaman *Employee Management*

d. Halaman *Leave Request*

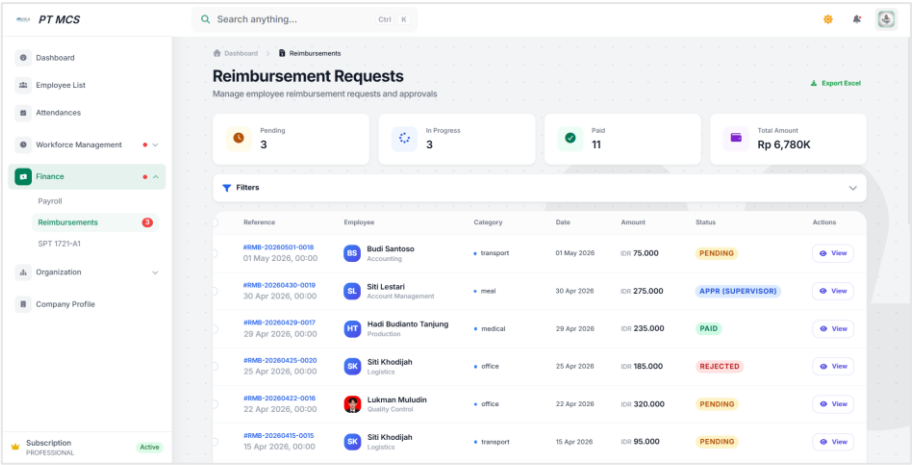
Menu *Leave Request* berfungsi sebagai pusat persetujuan permohonan cuti dan izin yang diajukan oleh karyawan. HR dapat meninjau alasan pengajuan, melihat lampiran dokumen pendukung seperti surat dokter, kemudian memutuskan untuk menyetujui atau menolak permohonan tersebut secara digital.



Gambar 3.35 Halaman *Leave Requests Management*

e. Halaman *Reimbursement*

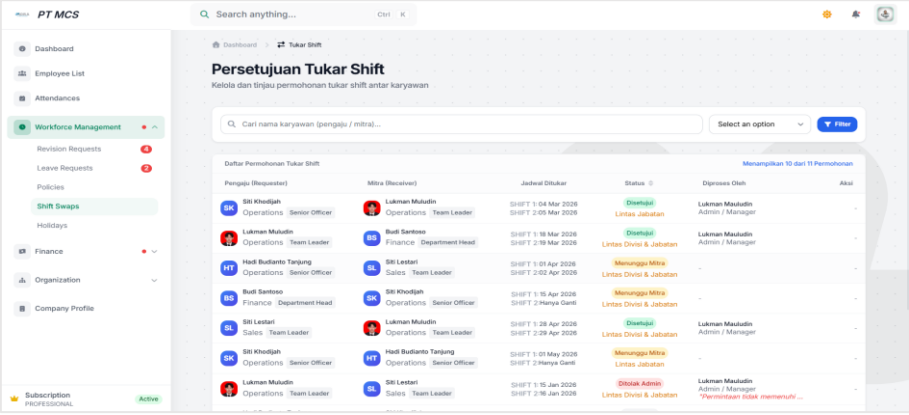
Menu *Reimbursement* digunakan untuk menampung dan memproses klaim pengembalian dana biaya operasional yang diajukan karyawan. *Admin*/HR dapat memeriksa nominal klaim, memverifikasi keaslian foto nota atau kwitansi yang dilampirkan, serta memperbarui status klaim menjadi disetujui atau ditolak.



Gambar 3.36 Halaman *Reimbursement*

f. Halaman *Swap Shift*

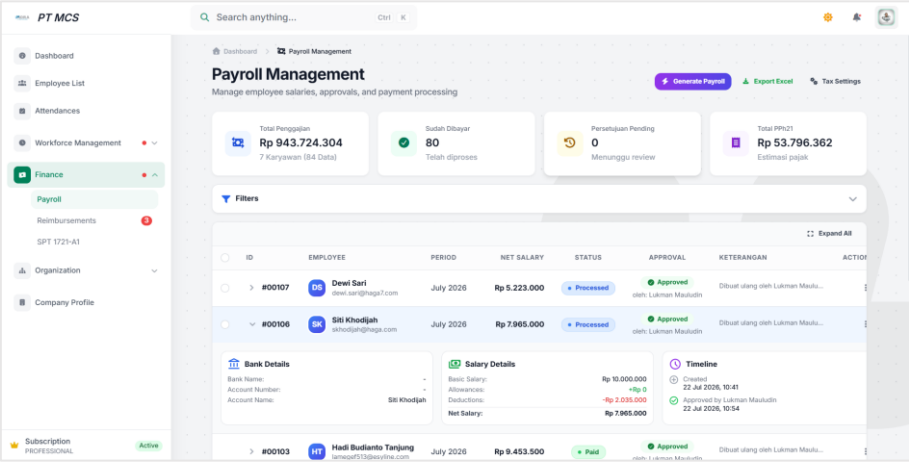
Menu *Swap Shift* memfasilitasi persetujuan pengajuan tukar jadwal kerja antar karyawan. HR bertindak sebagai *validator* akhir untuk memastikan bahwa pertukaran jadwal yang diajukan oleh kedua belah pihak tidak mengganggu operasional divisi terkait dan tetap menjaga keseimbangan alokasi sumber daya manusia pada hari tersebut.



Gambar 3.37 Halaman *Shift Swaps*

g. Halaman *Payroll*

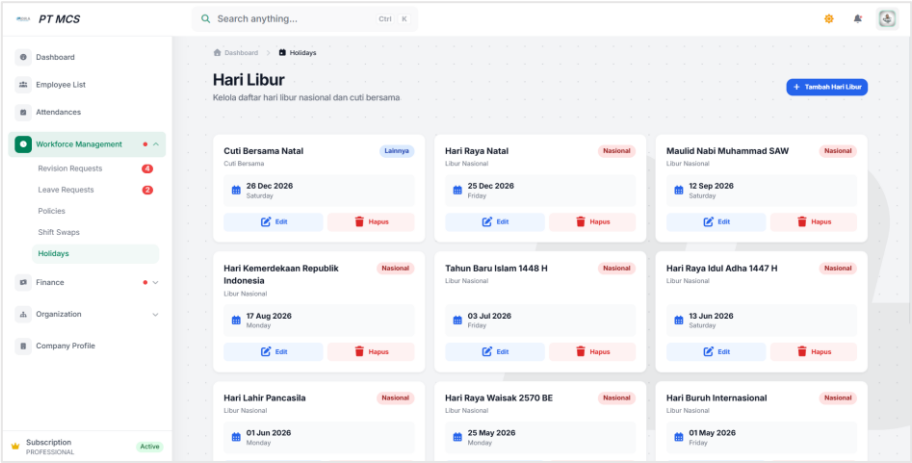
Menu *Payroll* memproses perhitungan gaji bulanan seluruh karyawan secara otomatis berdasarkan kehadiran, keterlambatan, pengajuan cuti, potongan kasbon, serta kalkulasi pajak jika dikonfigurasi. HR dapat memicu pembuatan slip gaji bulanan secara massal, melacak riwayat pembayaran gaji bulanan instansi, serta mengirimkan slip gaji langsung ke akun masing-masing karyawan.



Gambar 3.38 Halaman *Payroll Management*

h. Halaman *Holiday*

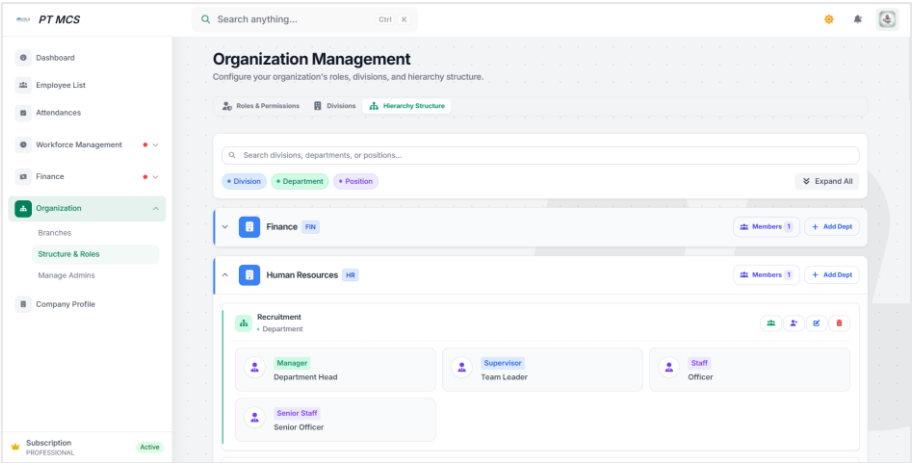
Menu *Holiday* digunakan HR untuk mendefinisikan hari libur nasional atau hari libur khusus *int*ernal instansi dalam kalender kerja tahunan. Hari libur yang didaftarkan di sini secara otomatis akan meliburkan jadwal absensi harian karyawan dan mengecualikan pemotongan gaji akibat tidak hadir pada tanggal tersebut.



Gambar 3.39 Halaman *Manage Holidays*

i. Halaman *Organization*

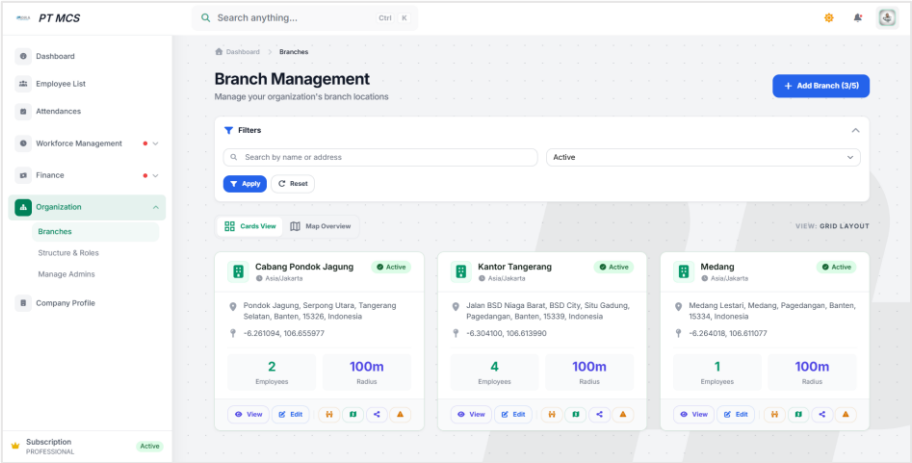
Menu *Organization* membantu HR dalam memetakan struktur organisasi perusahaan. Melalui menu ini, HR dapat menambahkan kantor cabang, mendirikan departemen seperti divisi teknologi, keuangan, dan pemasaran, serta menyusun tingkatan jabatan untuk menjaga kerapian alokasi dan penugasan karyawan.



Gambar 3.40 Halaman *Organization Management*

j. Halaman *Branches*

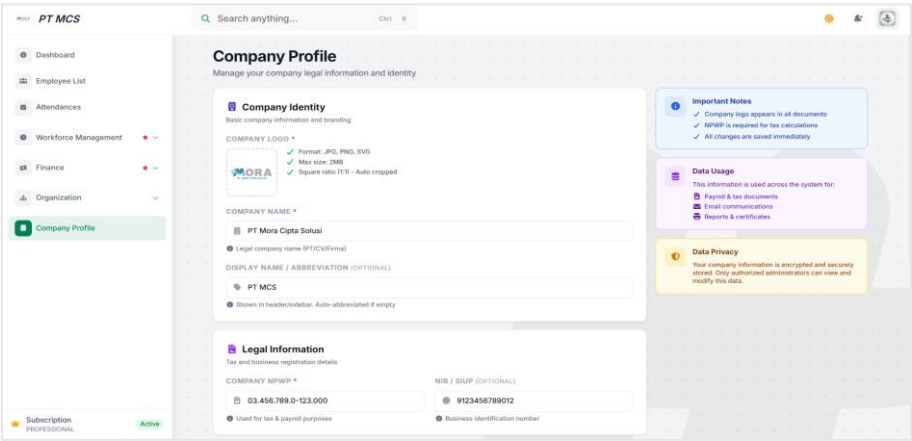
Fitur *Branches* memungkinkan *Admin* atau HR untuk mengelola berbagai lokasi kantor cabang yang dimiliki oleh perusahaan secara terpusat dan ter*int*egrasi. Melalui menu ini, pengguna dapat mendaftarkan cabang baru, mengatur alamat fisik, serta menentukan titik koordinat GPS lengkap dengan radius toleransi presisi atau *geofencing*. Pengaturan lokasi ini berfungsi sebagai acuan utama sistem untuk memvalidasi posisi *check-in* dan *check-out* karyawan saat melakukan absensi seluler guna mencegah kecurangan lokasi.



Gambar 3.41 Halaman *Branch Management*

k. Halaman *Company Profile*

Menu *Company Profile* digunakan untuk mengonfigurasi informasi dasar instansi, seperti nama perusahaan, logo instansi, alamat email, serta nomor kontak resmi.



Gambar 3.42 Halaman *Company Profile*

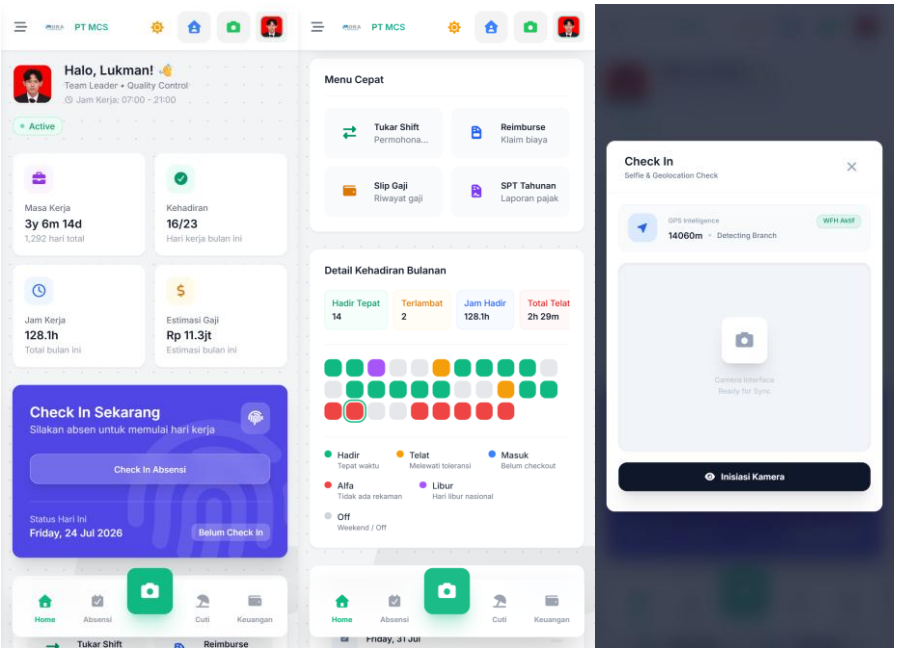
3.4.4 Penggunaan Sistem oleh Karyawan

Halaman karyawan dirancang menggunakan konsep responsive sehingga dapat diakses melalui perangkat *desktop* maupun *mobile*. Namun, mengingat aktivitas karyawan lebih sering dilakukan melalui *smartphone*, maka tampilan yang ditampilkan pada bagian ini menggunakan versi *mobile*.

Berikut merupakan menu-menu utama yang dapat diakses oleh karyawan beserta fungsinya:

a. Halaman *Dashboard* Karyawan

Dashboard Karyawan dirancang *int*uitif untuk mempermudah akses operasional sehari-hari staf dari perangkat mereka. Halaman utama ini menyediakan tombol cepat untuk melakukan *Clock In* dan *Clock Out*, menampilkan informasi *shift* kerja hari ini dan status kehadiran harian pribadi.

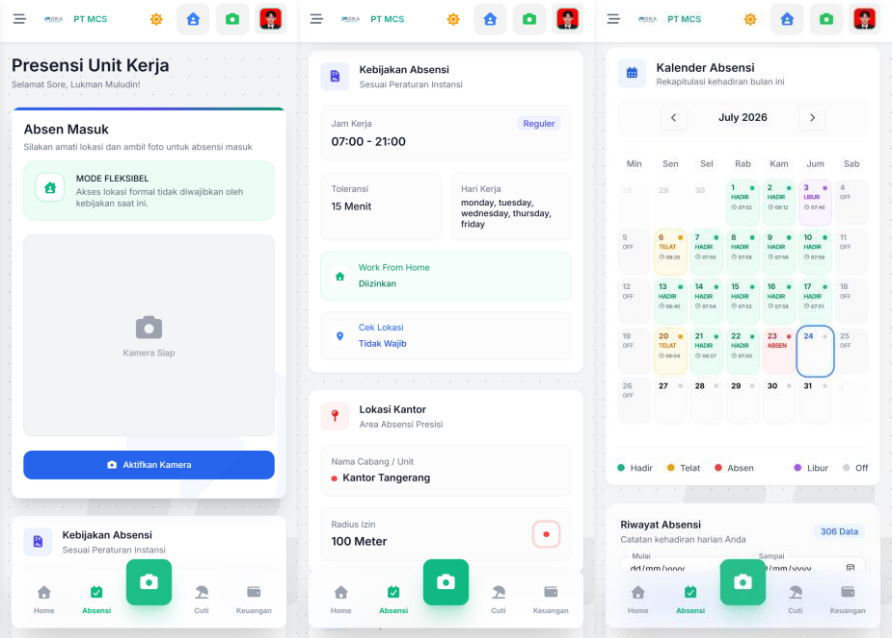


Gambar 3.43 Halaman *Dashboard* Karyawan

b. Halaman *Attendance* (Absensi)

Menu *Attendance* memungkinkan karyawan untuk mencatatkan kehadiran secara mandiri menggunakan teknologi absensi GPS dan foto *selfie* langsung dari *smartphone* mereka. Selain melakukan absensi harian, karyawan juga dapat melihat histori riwayat kehadiran pribadi

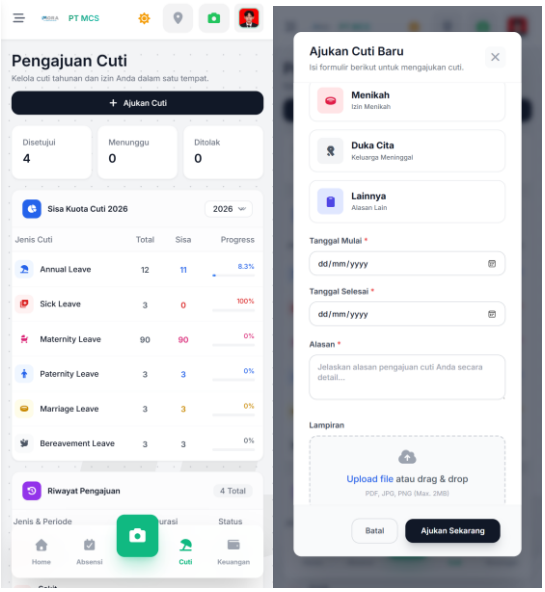
mereka sepanjang bulan berjalan untuk meminimalkan kekeliruan pencatatan.



Gambar 3.44 Halaman *Attendance* (Absensi) Karyawan

c. Halaman *Leave Request* (Pengajuan Cuti)

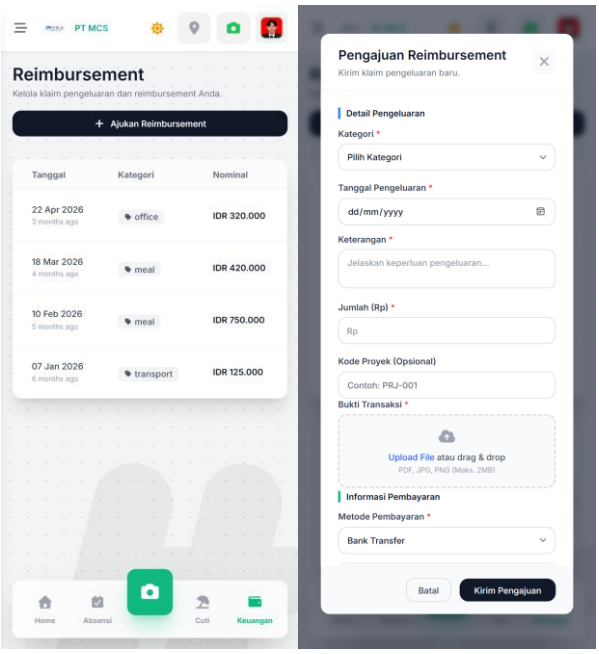
Menu *Leave Request* digunakan oleh karyawan untuk mengajukan permohonan cuti secara mandiri. Karyawan hanya perlu memilih tanggal cuti, mengisi alasan, mengunggah dokumen pendukung seperti surat dokter, dan memantau status persetujuan dari HR secara langsung.



Gambar 3.45 Halaman *Leave Requests* Karyawan

d. Halaman *Reimbursement*

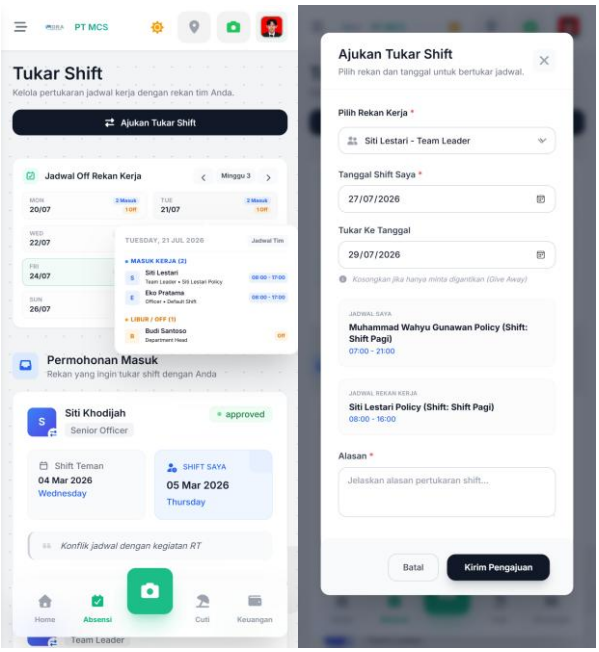
Menu *Reimbursement* mempermudah karyawan untuk mengajukan klaim pengembalian biaya yang telah dikeluarkan untuk kepentingan perusahaan. Karyawan dapat mengisi form nominal klaim, menulis deskripsi pengeluaran, mengunggah foto struk atau nota pembayaran secara digital, serta melacak status pembayaran klaim tersebut baik yang masih pending, disetujui, maupun ditolak.



Gambar 3.46 Halaman *Reimbursement* Karyawan

e. Halaman *Swap Shift* (Tukar *Shift*)

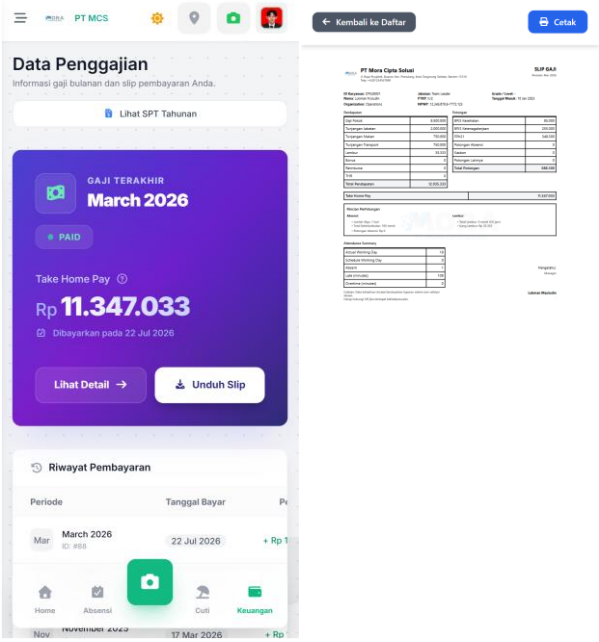
Menu *Swap Shift* memberikan fleksibilitas kepada karyawan untuk mengajukan pertukaran jadwal kerja dengan rekan sejawatnya ketika berhalangan hadir pada jadwal yang telah ditentukan. Pengajuan yang dibuat akan diteruskan kepada rekan kerja yang ditunjuk untuk memperoleh konfirmasi atas pertukaran tersebut. Setelah rekan kerja memberikan persetujuan, pengajuan selanjutnya dapat melalui proses validasi oleh pihak HR untuk memastikan pertukaran jadwal sesuai dengan ketentuan dan kebutuhan operasional perusahaan. Selain itu, fitur *Swap Shift* membantu mengurangi kebutuhan koordinasi secara manual dan memudahkan pihak HR dalam memantau perubahan jadwal kerja karyawan.



Gambar 3.47 Halaman *Shift Swap* Karyawan

f. Halaman Slip Gaji

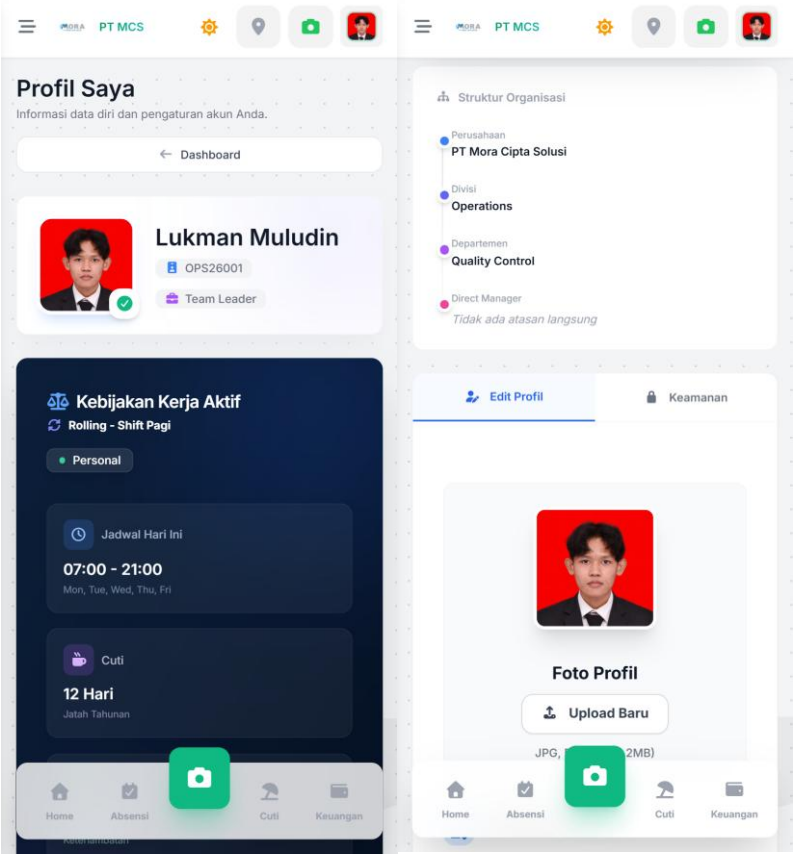
Menu Slip Gaji memberikan hak akses kepada karyawan untuk mengunduh dokumen slip rincian pendapatan bulanan mereka secara transparan. Dokumen digital ini memuat informasi lengkap mengenai rincian gaji yang dapat dicetak atau disimpan ke format PDF.



Gambar 3.48 Halaman Slip Gaji Karyawan

g. Halaman *Profile*

Menu *Profile* memuat informasi data diri pribadi karyawan seperti nama lengkap, kontak darurat, foto profil, dan jabatan pekerjaan. Karyawan dapat memperbarui data pribadi tertentu secara mandiri seperti alamat tempat tinggal terbaru atau nomor kontak yang bisa dihubungi, serta mengunggah foto profil baru yang dapat disesuaikan menggunakan fitur image cropping.



Gambar 3.49 Halaman Profil Karyawan

3.5 *Troubleshoot*

Dalam proses penggunaan sistem HagaPlus, terdapat beberapa kendala yang mungkin terjadi, baik pada saat instalasi maupun saat sistem dijalankan. Kendala tersebut umumnya berkaitan dengan konfigurasi sistem, koneksi *database*, maupun kesalahan dalam penggunaan. Oleh karena itu, diperlukan langkah penanganan agar sistem dapat kembali berjalan dengan normal.

Adapun beberapa permasalahan yang umum terjadi beserta cara penanganannya adalah sebagai berikut:

3.5.1 Masalah Tampilan & Aset (CSS / JS)

a. Tampilan Aplikasi Berantakan (CSS / JS Tidak Muat)

1. Gejala: Halaman web hanya memuat teks HTML polos tanpa gaya desain (CSS) dan *int*eraksi JavaScript tidak berfungsi.
2. Penyebab:
   a) Aset *frontend* belum dikompilasi (di-build) ke folder produksi public/build.
   b) Alamat APP_URL pada *file* .env tidak sesuai dengan domain/URL yang diakses.
3. Solusi:
   a) Di komputer lokal Anda, jalankan per*int*ah kompilasi sebelum mengunggah *file* ke hosting:

```
npm run build
```

   Pastikan folder public/build ikut terunggah ke folder publik hosting Anda (public_html atau htdocs).

   b) Buka *file* .env di server hosting dan pastikan APP_URL diatur dengan benar menggunakan protokol yang tepat (misal: https://nama-domain-anda.com).

b. *Error Mixed Content* (CSS/JS Diblokir oleh *Browser*)

1. Gejala: Aset tidak termuat dan konsol browser menampilkan *error*: "*Mixed Content: The page at 'https://...' was loaded over HTTPS, but requested an insecure stylesheet 'http://...'.*"
2. Penyebab: Laravel menghasilkan URL aset menggunakan protokol http:// sedangkan situs web diakses menggunakan https://.
3. Solusi: Buka berkas app/Providers/AppServiceProvider.php dan tambahkan pemaksaan skema HTTPS pada *method boot()* jika aplikasi berjalan di lingkungan produksi:

```
use Illuminate\Support\Facades\URL;

public function boot(): void
{
    if (app()->environment('production')) {
        URL::forceScheme('https');
    }

}
```

3.5.2 Masalah Koneksi & Konfigurasi *Database*

a. *Error*: SQLSTATE[HY000] [2002] *Connection refused* atau *Connection timed out*

1. Gejala: Aplikasi menampilkan halaman *error database* saat mencoba mengakses data atau menjalankan migrasi.
2. Penyebab:
   a) Konfigurasi DB_HOST pada *file* .env tidak sesuai. Pada *shared* hosting, menggunakan IP 127.0.0.1 sering kali ditolak.
   b) Kredensial *database* (*user*name, nama *database*, atau *password*) salah.
3. Solusi:
   a) Ubah nilai DB_HOST pada *file* .env dari 127.0.0.1 menjadi localhost:

```
DB_HOST=localhost
```

   b) Pastikan nama *database* dan *user*name menyertakan prefix unik bawaan hosting Anda (contoh: u123456_hagaplus, bukan hanya hagaplus).
   c) Pastikan *user database* telah diberikan *All Privileges* (Hak Akses Penuh) ke *database* terkait di menu kontrol panel hosting.

3.5.3 Masalah *Runtime* & *Server*

a. Halaman Menampilkan "500 | *Int*ernal Server *Error*" atau Layar Putih Polos

1. Gejala: Aplikasi sama sekali tidak dapat diakses dan hanya menampilkan status kode 500 atau halaman kosong.
2. Penyebab:
   a) File .env belum dibuat atau kosong.
   b) Kunci aplikasi (APP_KEY) belum di-generate.
   c) Hak akses (*permission*) folder penyimpanan Laravel tidak memadai.
3. Solusi:
   a) Pastikan berkas .env sudah ada di server hosting.
   b) *Generate application key* dengan menjalankan:

```
php artisan key:generate
```

Jika tidak memiliki SSH, gunakan Metode B (*Route* Sementara) pada README.md untuk memicu Artisan::call('key:generate').

c) Ubah hak akses direktori *storage*/ dan bootstrap/*cache*/ menjadi dapat ditulis oleh server (biasanya CHMOD 775 atau 755, pada beberapa hosting memerlukan 777).

3.5.4 Masalah Fitur Aplikasi (Absensi & Notifikasi)

a. Foto Karyawan atau Berkas *Upload* Tidak Muncul (*Error* 404)

1. Gejala: *Avatar* pengguna atau dokumen yang diunggah memicu *error 404* (*Not Found*) meskipun filenya ada di server.
2. Penyebab: P*int*asan direktori (*symlink) dari* folder *public/storage* ke *storage/app/public* belum dibuat atau rusak (*broken link*) akibat pemindahan *file* ke hosting.
3. Solusi:
   a) Solusi Hosting Standar: Hapus folder/link public/*storage* yang lama terlebih dahulu jika ada di dalam folder publik hosting Anda, lalu buat ulang tautan simbolis dengan menjalankan per*int*ah:

   ```
   php artisan storage:link
   ```

   (Jika tidak ada SSH, akses URL https://domain-anda.com/init-server untuk memicu pembuatannya).

   b) Solusi Hosting Ketat (*Bypass* Jalur *Storage*): Kami telah mengimplementasikan sistem *bypass* otomatis menggunakan rute /public-media/{path} dan *middleware RewriteStorageUrl* yang aktif secara otomatis ketika berjalan di server produksi.
      1) Sistem ini akan mem-*bypass web server* untuk menyajikan *file* secara aman dari server tanpa menggunakan *symlink.*
      2) *Pasti*kan Anda mengosongkan nilai PUBLIC_STORAGE_URL di lokal Anda, namun Anda dapat mendefinisikannya secara opsional di .env hosting dengan nilai:

      ```
      PUBLIC_STORAGE_URL=https://nama-domain-anda.com/public-media
      ```

(Sistem kami juga mendeteksi otomatis alamat domain produksi ini secara dinamis jika tidak diset di .env).

b. Proses Absensi Lambat atau Email Slip Gaji Tidak Terkirim

1. Gejala: Saat karyawan melakukan absensi, aplikasi terasa menggantung beberapa detik. Email notifikasi atau slip gaji juga tidak pernah masuk ke email karyawan.
2. Penyebab: Aplikasi menggunakan sistem antrean (*Queue*) berbasis *database* (*QUEUE_CONNECTION=database*), namun tidak ada *worker* yang berjalan secara terus-menerus di hosting untuk memproses antrean tersebut.
3. Solusi:
   a) Pastikan Anda telah menambahkan *Cron Job* di panel hosting yang memicu per*int*ah *queue worker* setiap menit:

```
* * * * * cd /path/ke/proyek/anda && php artisan queue:work --once >> /dev/null 2>&1
```

   b) Jika hosting Anda benar-benar membatasi *Cron Job, Anda* bisa mengubah konfigurasi .env sementara untuk memproses tugas secara sinkron (langsung tanpa antrean), meskipun ini akan memperlambat respon aplikasi saat mengirim email:

```
QUEUE_CONNECTION=sync
```

3.5.5 *Int*egrasi Pihak Ketiga (*Third-Party*)

a. *Error: redirect_uri_mismatch* Saat *Login* Google ***(Socialite*)**

1. Gejala: Ketika pengguna mengklik tombol "*Login via* Google*",* halaman Google menampilkan pesan *error redirect_uri_mismatch*.
2. Penyebab: URL *callback yang d*idaftarkan di Google *Cloud Console* tidak sama persis dengan GOOGLE_REDIRECT_URI di *file* .env. Perbedaan kecil seperti http vs https atau adanya / di akhir URL akan memicu *error* ini.
3. Solusi:
   a) Periksa nilai GOOGLE_REDIRECT_URI di .env Anda:

```
GOOGLE_REDIRECT_URI=https://domain-anda.com/auth/google/callback
```

b) Masuk ke Google *Cloud Console* -> *Credentials* -> edit *OAuth 2.0 Client ID* Anda.

c) Di bawah bagian *Authorized redirect* URIs, tambahkan URL yang sama persis dengan yang tertulis di .env Anda tersebut.

BAB IV
KESIMPULAN

4.1 Kesimpulan

Berdasarkan hasil perancangan dan implementasi sistem HagaPlus yang telah dilakukan, dapat disimpulkan bahwa sistem ini mampu memberikan solusi dalam pengelolaan absensi, penggajian, perhitungan iuran jaminan sosial (BPJS Kesehatan & Ketenagakerjaan), serta pemotongan pajak (PPh21) karyawan secara ter*int*egrasi, khususnya bagi UMKM yang masih menggunakan metode manual. Sistem yang dibangun berbasis web ini memungkinkan proses pencatatan kehadiran, pengelolaan cuti dan izin, perhitungan gaji, hingga penyusunan dokumen pelaporan pajak tahunan (SPT 1721-A1) dilakukan secara lebih efisien dan terstruktur.

Penerapan HagaPlus juga menunjukkan bahwa penggunaan sistem digital dapat membantu mengurangi risiko kesalahan dalam pengolahan data, meningkatkan akurasi perhitungan potongan wajib karyawan sesuai regulasi pemer*int*ah, serta mempercepat proses administrasi. Selain itu, fitur yang disediakan mampu meningkatkan transparansi karena informasi kehadiran, slip gaji, dan dokumen perpajakan dapat diakses secara lebih mudah oleh pengguna (karyawan) secara mandiri.

Dari sisi pengembangan, sistem ini dirancang dengan mempertimbangkan kemudahan penggunaan serta fleksibilitas agar dapat menyesuaikan kebutuhan UMKM. Konsep berbasis *subscription* (berlangganan) yang diterapkan juga menjadi solusi yang lebih terjangkau dibandingkan harus membangun sistem secara mandiri, sehingga lebih realistis untuk diimplementasikan.

Dengan demikian, HagaPlus dapat menjadi alternatif solusi digital yang membantu UMKM dalam meningkatkan efisiensi operasional, akurasi data, serta kesiapan dalam menghadapi transformasi digital di era saat ini.

LAMPIRAN

1. Implementing Agreement (IA)

IMPLEMENTATION ARRANGEMENT FORM
PROGRAM STUDI SISTEM INFORMASI S1
UNIVERSITAS PAMULANG

MORA
PT. MORA CIPTA SOLUSI

Implementasi Kerja Sama di Bidang Pendidikan
Nomor : 041033/IA/2025/26200

Yang bertandatangan di bawah ini:

Nama	:	Geri Debataraja Simamora, S.Kom.
Nama Instansi	:	PT MORA CIPTA SOLUSI
Jabatan	:	Direktur
Alamat	:	Ruko Alicante D 56, Kel. Medang, Kec.Pagedangan, Kab. Tangerang, Provinsi Banten, Kode Pos: 15334

Dengan ini menerangkan dosen-dosen/mahasiswa Program Studi SISTEM INFORMASI S1 Fakultas ILMU KOMPUTER Universitas Pamulang, Berikut ini:

No.	Nama Dosen/Mahasiswa	NIDN/NIM	Keterangan
1.	HERI HAERUDIN S.Kom., M.Kom.	0403028306	Dosen Pembimbing
2.	LUKMAN MULUDIN	221011700337	Mahasiswa
3.	SITI KHODIJAH	221011700488	Mahasiswa

Dosen-dosen/Mahasiswa tertera di atas telah melaksanakan kegiatan, sebagai berikut:

Nama Kegiatan	:	Karya Tulis / Project Work
Tema	:	SISTEM HAGA PLUS PADA PT. MORA CIPTA SOLUSI
Hari/Tanggal Pelaksanaan Kegiatan	:	Sabtu, 06 September 2025
Tempat	:	Lokasi kegiatan

Kegiatan tersebut di atas merupakan implementasi dari Nota Kesepahaman (MOU)/ Perjanjian Kerja Sama (MOA/PKS) antara Universitas pamulang dan mitra Nomor: 041033/IA/2025/26200 Tentang: Karya Tulis/Project Work

Tangerang Selatan, 9/6/2025

Direktur	Ketua Program Studi
PT MORA CIPTA SOLUSI	SISTEM INFORMASI S1
	Universitas Pamulang.
MORA	PROGRAM STUDI SISTEM INFORMASI FAKULTAS
Geri Debataraja Simamora, S.Kom.	Heri Haerudin, S.Kom., M.Kom.

2. Laporan Pelaksanaan Kerja Sama

LAPORAN PELAKSANAAN KERJA SAMA

1.	JUDUL KERJA SAMA	:	Penelitian, Tugas Akhir Karya Tulis/Project Work Mahasiswa yang Ber-HKI
2.	REFERENSI KERJA SAMA (MoA/IA)	:	041033/IA/2025/26200
3.	MITRA KERJA SAMA	:	PT MORA CIPTA SOLUSI
4.	RUANG LINGKUP	:	Implementasi Kerjasama
5.	HASIL PELAKSANAAN (OUTPUT & OUTCOME)	:	Penelitian, Tugas Akhir Karya Tulis/Project Work Mahasiswa yang Ber-HKI
6.	TAUTAN/LINK DOKUMENTASI KEGIATAN	:	

PENANGGUNG JAWAB KEGIATAN
06 September 2025
Ketua Pelaksana,

HERI HAERUDIN S.Kom., M.Kom.
NIDN.0403028306

Pimpinan Mitra,
PT MORA CIPTA SOLUSI

Geri Debataraja Simamora, S.Kom.

Mengetahui,
Pimpinan,
Program Studi Sistem Informasi
Universitas Pamulang

Heri Haerudin, S.Kom., M.Kom.
NIDN.0403028306

3. Berita Acara Serah Terima

BERITA ACARA SERAH TERIMA

Saya yang bertanda tangan di bawah ini, pada hari Senin, tanggal 8 bulan Juni, tahun 2026

1. NIM : 221011700488
   Nama : Siti Khodijah
   Institusi : Universitas Pamulang
   Program Studi : Sistem Informasi
   Alamat : Jl. Puspitek, Buaran, Kec. Pamulang, Kota Tangerang Selatan, Banten 15310
   Telepon : 0217412566

Selanjutnya disebut **PIHAK PERTAMA.**

2. Nama : Geri Debataraja Simamora, S.Kom
   Jabatan : Direktur
   Alamat : PT MORA CIPTA SOLUSI
   Ruko Alicante D 56, Kel. Medang, Kec. Pagedangan, Kab. Tangerang, Provinsi Banten 15334
   Telepon : 0822-9993-0700

Selanjutnya disebut **PIHAK KEDUA.**

PIHAK PERTAMA menyerahkan aplikasi kepada **PIHAK KEDUA** dan **PIHAK KEDUA** menyatakan telah menerima aplikasi dari **PIHAK PERTAMA**, berupa sistem HagaPlus.

Demikian berita acara serah terima aplikasi ini dibuat dan aplikasi tersebut dalam keadaan baik, sejak dilakukan penandatanganan berita acara ini sehingga aplikasi tersebut menjadi tanggung jawab **PIHAK KEDUA** untuk digunakan sebagaimana mestinya.

PIHAK PERTAMA Mahasiswa	**PIHAK KEDUA** Pimpinan Mitra
Siti Khodijah 221011700488	Geri Debataraja Simamora, S.Kom

Berita Acara Serah Terima program HagaPlus yang telah di tandatangani oleh mitra project

4. Wawancara Bersama Pimpinan PT. Mora Cipta Solusi

Tim Project Work melakukan wawancara bersama pihak mitra untuk membahas kebutuhan sistem, fitur yang dikembangkan, serta memperoleh masukan dan klarifikasi terkait sistem HagaPlus. Hasil wawancara digunakan sebagai bahan dalam proses pengembangan dan penyempurnaan sistem agar sesuai dengan kebutuhan yang telah disampaikan oleh mitra.

5. Instrumen Wawancara seputar sistem Hagaplus

INSTRUMEN DAN HASIL WAWANCARA ANALISIS KEBUTUHAN SISTEM HAGAPLUS

A. Identitas Wawancara

Nama Mitra/Perusahaan	: PT Mora Cipta Solusi
Nama Narasumber	: Geri Debataraja Simamora, S.Kom
Jabatan	: Direktur
Tanggal Wawancara	: 19 Desember 2025
Tempat Wawancara	: Bengkel Lapar, Telaga Kahuripan
Pewawancara	: Siti Khodijah dan Lukman Muludin

B. Hasil Wawancara Analisis Kebutuhan

No.	Pertanyaan	Jawaban Mitra
1.	Apa kebutuhan utama UMKM atau perusahaan dalam mengelola absensi dan penggajian karyawan?	menurut mitra sistem yang dibutuhkan UMKM adalah sistem yang praktis dan mudah digunakan untuk membantu mengatur absensi dan penggajian agar lebih mudah dan mengurangi pekerjaan manual
2.	Mengapa UMKM dipilih sebagai target utama pengguna sistem ini dibandingkan perusahaan yang lebih besar?	Karena UMKM memiliki skala usaha yang lebih sederhana dan membutuhkan solusi yang praktis dengan biaya ~~[illegible]~~ terjangkau, tanpa perlu membangun sistem sendiri dari awal
3.	Apa saja fitur yang dibutuhkan dalam sistem pengelolaan karyawan?	Fitur utamanya ada data karyawan, absensi, cuti, reimbursement, tukar shift, penggajian, dan laporan

4.	Mengapa diperlukan sistem yang mengintegrasikan absensi dan penggajian dalam satu platform?	Supaya data absensi bisa langsung digunakan dalam proses penggajian, Jadi ga perlu rekap secara manual lagi
5.	Mengapa konsep SaaS dan sistem berlangganan sesuai untuk UMKM?	karena lebih Fleksibel dan sistem kita harganya harus lebih terjangkau dengan sistem Saas yang sudah ada, UMKM bisa pilih paket sesuai jumlah karyawan mereka
6.	Siapa saja yang akan menggunakan sistem dan bagaimana pembagian hak aksesnya?	Bisa pemilik, pimpinan, atau HRD dan karyawan, Hak aksesnya bisa disesuaikan dengan tugas dan bekasan masing-masing
7.	Siapa yang memiliki kewenangan untuk melakukan approval terhadap cuti, izin, reimbursement, dan tukar shift?	bisa pemilik, pimpinan, atau yang memiliki hak akses yang sudah ditentukan, atau yang jabatannya lebih tinggi
8.	Di mana sistem akan digunakan oleh pengguna dalam menjalankan aktivitas absensi dan administrasi karyawan?	Sistem bisa diakses secara online, melalui browser atau ~~[illegible]~~ saja, Jadi bisa diakses menggunakan perangkat apa saja
10.	Bagaimana metode absensi yang diharapkan?	metode absensinya cukup selfie dan berdasarkan lokasi aja, karena prosesnya sedar hanya dan mudah digunakan untuk kedamanan bisa tambahkan validasi waktu dan lokasi
11.	Bagaimana data absensi dapat digunakan dalam proses perhitungan penggajian?	data kehadiran, keterlambatan ketidakhadiran, dan lembur bisa digunakan sebagai dasar perhitungan gaji

12.	Bagaimana proses pengajuan dan approval cuti, izin, reimbursement, serta tukar shift yang diharapkan?	karyawan mengajukan melalui sistem, kemudian atasan mereka memeriksa dan mengambil keputusan, jika tukar shift maka rekan mereka yang harus konfirmasi terlebih dahulu
13.	Bagaimana sistem dapat membantu proses perhitungan PPh 21, BPJS, dan komponen penggajian lainnya?	sistem dapat membantu menghitung PPh21, BPJS, dan komponen lainnya berdasarkan aturan dari pemerintah yang berlaku, untuk komponen lainnya sesuai kebutuhan masing-masing
15.	Bagaimana sistem SaaS dapat menyediakan layanan yang terjangkau bagi UMKM melalui paket berlangganan?	Paket ditentukan berdasarkan jumlah karyawan, cabang dan admin, sehingga umkm dapat memilih paket sesuai dengan kebutuhan mereka

C. Pengesahan

Bogor, 19 Desember 2025

PT. Mora Cipta Solusi	Mahasiswa
Geri Debataraja Simamora, S.Kom	**Lukman Muludin** **221011700337**

6. Foto Bersama Pimpinan PT Mora Cipta Solusi

Pada kegiatan ini, pihak mitra melakukan penandatanganan lembar persetujuan Project Work sebagai bentuk persetujuan dan konfirmasi terhadap pelaksanaan proyek yang telah dilakukan.

7. Surat Pencatatan Ciptaan (HAKI)

REPUBLIK INDONESIA
KEMENTERIAN HUKUM

SURAT PENCATATAN CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan	:	EC002026094921, 23 Juni 2026
Pencipta		
Nama	:	**Lukman Muludin, Siti Khodijah dkk**
Alamat	:	Kp. Bojong RT001/002, Desa Kemuning, Kecamatan Legok, Kabupaten Tangerang, Legok, Kab. Tangerang, Banten, 15820
Kewarganegaraan	:	Indonesia
Pemegang Hak Cipta		
Nama	:	**Lukman Muludin, Siti Khodijah dkk**
Alamat	:	Kp. Bojong RT001/002, Desa Kemuning, Kecamatan Legok, Kabupaten Tangerang, Legok, Kab. Tangerang, Banten, 15820
Kewarganegaraan	:	Indonesia
Jenis Ciptaan	:	**Program Komputer**
Judul Ciptaan	:	**HagaPlus**
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	:	23 Juni 2026, di Kota Tangerang Selatan
Jangka waktu pelindungan	:	Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor Pencatatan	:	001297991

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL
u.b
Direktur Hak Cipta dan Desain Industri

Agung Damarsasongko,SH.,MH.
NIP. 196912261994031001

Balai Besar Sertifikasi Elektronik

Disclaimer:
1. Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.
2. Surat Pencatatan ini telah disegel secara elektronik menggunakan segel elektronik yang ditertibkan oleh Balai Besar Sertifikasi Elektronik, Badan Siber dan Sandi Negara.
3. Surat Pencatatan ini dapat dibuktikan keasliannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Lukman Muludin	Kp. Bojong RT001/002, Desa Kemuning, Kecamatan Legok, Kabupaten Tangerang Legok, Kab. Tangerang
2	Siti Khodijah	Meruya Utara No.17 RT.005 RW.011, Meruya Utara, kembangan Kembangan, Kota Adm. Jakarta Barat
3	Heri Haerudin	Kp. Ciater RT. 001/003 Kel. Rawa Mekar Jaya Kec. Serpong Serpong, Kota Tangerang Selatan
4	Khanif Faozi	Jl. H. Jamat GG. Rais RT. 001/004 Kel. Buaran Kec. Serpong Serpong, Kota Tangerang Selatan
5	Gusmayeni	Komp Sekneg Blok A-22/21 RT. 008/008 Kel. Sunter Agung Kec. Tanjung Priok Tanjung Priok, Kota Adm. Jakarta Utara

LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Lukman Muludin	Kp. Bojong RT001/002, Desa Kemuning, Kecamatan Legok, Kabupaten Tangerang Legok, Kab. Tangerang
2	Siti Khodijah	Meruya Utara No.17 RT.005 RW.011, Meruya Utara, kembangan Kembangan, Kota Adm. Jakarta Barat
3	Heri Haerudin	Kp. Ciater RT. 001/003 Kel. Rawa Mekar Jaya Kec. Serpong Serpong, Kota Tangerang Selatan
4	Khanif Faozi	Jl. H. Jamat GG. Rais RT. 001/004 Kel. Buaran Kec. Serpong Serpong, Kota Tangerang Selatan
5	Gusmayeni	Komp Sekneg Blok A-22/21 RT. 008/008 Kel. Sunter Agung Kec. Tanjung Priok Tanjung Priok, Kota Adm. Jakarta Utara

Balai Besar Sertifikasi Elektronik

Disclaimer:
1. Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.
2. Surat Pencatatan ini telah disegel secara elektronik menggunakan segel elektronik yang ditertibkan oleh Balai Besar Sertifikasi Elektronik, Badan Siber dan Sandi Negara.
3. Surat Pencatatan ini dapat dibuktikan keasliannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.

8. Kartu Bimbingan

KARTU KONSULTASI MAHASISWA

Nama Lengkap : LUKMAN MULUDIN
Nomor Induk Mahasiswa : 221011700337
Fakultas / Jurusan : ILMU KOMPUTER / SISTEM INFORMASI S1
Kampus : UNIVERSITAS PAMULANG
Judul Karya tulis : SISTEM HAGA PLUS PADA PT. MORA CIPTA SOLUSI

Pembimbing : HERI HAERUDIN, S.Kom., M.Kom.

Dokumen ini harus dilampirkan pada KARYA TULIS

No.	Tanggal	Catatan Bimbingan	Tanda Tangan
1	9 November 2025	Bimbingan bab I dan bab II, fokus pada revisi UML bai usecase, ERD, LRS, dan diagram lainnya.	
2	2 Desember 2025	Revisi BAB 2 dan revisi UML beserta penyesuaian programnya	
3	3 Desember 2025	Pertemuan dengan mitra untuk menyesuaikan kebutuhan program	
4	11 Desember 2025	revisi beberapa fitur	
5	29 Mei 2026	Pendaftaran HKI dan revisi BAB 2	
6	4 Juli 2026	update bimbingan bab 1, lalu penomoran sub bab 2 headingnya, serta bab 3 bagian instalasi sistem isinya	
7	13 Juli 2026	bimbingan laporan BAB 1 sampai kesimpulan	
8	17 Juli 2026	Bimbingan update revisi UML dan Diagram di BAB 2	

KARTU KONSULTASI MAHASISWA

Nama Lengkap : SITI KHODIJAH
Nomor induk Mahasiswa : 221011700488
Fakultas / Jurusan : ILMU KOMPUTER / SISTEM INFORMASI S1
Kampus : UNIVERSITAS PAMULANG
Judul Karya tulis : SISTEM HAGA PLUS PADA PT. MORA CIPTA SOLUSI

Pembimbing : HERI HAERUDIN, S.Kom., M.Kom.

Dokumen ini harus dilampirkan pada KARYA TULIS

No.	Tanggal	Catatan Bimbingan	Tanda Tangan
1	11 Oktober 2025	Bimbingan bab I dan bab II, fokus pada revisi UML baik usscase, ERD, LRS, dan diagram lainnya.	
2	9 November 2025	Bimbingan update revisi Bab I dan bab II.	
3	24 November 2025	Presentasi project work bersama direktur PT.Mora Cipta Solusi untuk pengembangan aplikasi Hagaplus.	
4	9 Aprli 2026	update informasi mengenal uml dan buku panduan hari rabu tanggal 8 april 2026	
5	29 Mei 2026	proses daftar HKI san cek manual book	
6	4 Juni 2026	update bimbingan bab 1. lalu ponomoran sub bab 2 headingnya, serta bab 3 bagian instalasi sistem isinya	
7	13 Juli 2026	Bimbingan Akhir Laporan dari Bab I sampe kesimpulan.	
8	16 Juli 2026	Bimbingan update revisi uml dan diagram di bab 2.	