

# LAPORAN PRAKTIK KERJA LAPANGAN PADA PT DWANSOFT GLOBAL INDONESIA

Periode 4 November 2024 – 17 April 2025

*Disusun Sebagai Pertanggungjawaban Pelaksanaan Praktik Kerja Lapangan*



Oleh :

Isabell Gaol

NIS. 13422043

Kelas XII RPL 2

**SMK NEGERI 7 BATAM**  
**TAHUN 2025**

# **LEMBAR PENGESAHAN**

**LAPORAN PRAKTIK KERJA LAPANGAN PADA  
PT DWANSOFT GLOBAL INDONESIA  
Periode 4 November 2024 – 17 April 2025**

Oleh:

Isabell Gaol

NIS. 13422043

Kelas XII RPL 2

Telah Disetujui dan disahkan pada tanggal 17 April 2025

Oleh :

Instruktur IDUKA,

Guru Mata Pelajaran PKL,

**RIFAN RIZKY**

**Gr. ADAM KRIZAL, S.Kom., S.Pd., M.M.**  
NIPPPK. 198608282022211005

Menyetujui,

Pimpinan PT Dwansoft Global Indonesia,

Wakil Kepala Sekolah Bidang HUMAS

**RIDWAN WINANTO**

**ENJANG SUHAEDIN, S.Pd., Gr., M.M.**  
NIP. 198703192019031001

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena dengan rahmat-Nya penulis dapat menyelesaikan Laporan Praktik Kerja Lapangan (PKL) di PT Dwansoft Global Indonesia. Laporan ini disusun sebagai bentuk pertanggungjawaban atas kegiatan PKL yang telah penulis jalani selama di perusahaan tersebut, yang bertujuan untuk memberikan gambaran umum tentang pengalaman dan pembelajaran yang penulis peroleh.

Dalam laporan ini, akan dijelaskan mengenai kegiatan selama PKL, tugas yang diberikan, serta hasil yang dicapai. Laporan ini juga berisi evaluasi dan penilaian terkait dengan perkembangan diri penulis selama program PKL.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ridwan Winanto sebagai Pimpinan PT Dwansoft Global Indonesia.
2. Bapak Rifan Rizky Gumilar sebagai pembimbing PT Dwansoft Global Indonesia.
3. Seluruh Karyawan PT Dwansoft Global Indonesia.
4. Orang tua dan wali yang selalu memberikan dukungan moral dan materiil.
5. Ibu Nursya'bani, M.Pd., selaku Kepala SMK Negeri 7 Batam.
6. Bapak Enjang Suhaedin, S.Pd., Gr., M.M., selaku Wakasek Humas SMK Negeri 7 Batam.
7. Bapak Gr. Adam Krizal, S.Kom., S.Pd., M.M., selaku Guru Pengampu Mata Pelajaran PKL SMK Negeri 7 Batam.
8. Bapak/Ibu Guru dan Tenaga Kependidikan SMK Negeri 7 Batam.

Semoga laporan ini dapat memberikan manfaat dan menjadi referensi yang berguna bagi pihak-pihak yang berkepentingan.

Batam, 17 April 2025

Penulis

Isabell Gaol

# DAFTAR ISI

LEMBAR JUDUL .....	ii
LEMBAR PENGESAHAN .....	iii
KATA PENGANTAR .....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vi
DAFTAR TABEL.....	ix
BAB I PENDAHULUAN.....	1
A. Latar Belakang.....	1
B. Tujuan PKL .....	2
C. Manfaat PKL .....	2
1. Manfaat bagi Peserta Didik.....	2
2. Manfaat bagi Sekolah.....	3
3. Manfaat bagi IDUKA (Industri dan Dunia Kerja) .....	3
D. Waktu dan Tempat Pelaksanaan PKL .....	1
1. Waktu Pelaksanaan PKL .....	1
2. Tempat Pelaksanaan PKL .....	1
BAB II PROFIL INDUSTRI DAN DUNIA KERJA .....	2
A. Sejarah Berdirinya IDUKA .....	2
B. Bidang Usaha.....	3
1. Aplikasi Akuntansi .....	3
2. Aplikasi Pinjaman Dana.....	3
3. Aplikasi Toko Online .....	3
4. Sistem Informasi Kepegawaian .....	3
5. Sistem Booking Lapangan Futsal .....	4
6. Sistem Pembelian Kepiting .....	4
C. Visi dan Misi IDUKA.....	4
1. Visi IDUKA .....	4
2. Misi IDUKA .....	4
D. Struktur Organisasi IDUKA .....	5
1. <i>Chief Executive Officer (CEO)</i> .....	5
2. <i>Human Resource Department (HRD)</i> .....	6
3. <i>Programmer &amp; Networking</i> .....	6

4.	<i>Internet Marketing</i> .....	6
E.	Budaya Kerja dan Tata Tertib Karyawan di IDUKA.....	7
1.	Nilai-Nilai Budaya Kerja .....	7
2.	Tata Tertib dan Peraturan Perusahaan .....	7
BAB III PROSES DAN HASIL BELAJAR DI IDUKA .....		9
A.	Proses Belajar di IDUKA .....	9
1.	Konsep Dasar yang Dipelajari .....	9
2.	Penerapan Teknologi dalam Sistem Website Booking dan Transaksi .....	22
3.	Penerapan Teknologi dalam Sistem <i>Website</i> Penjualan-Pembelian Kepiting.....	46
B.	Hasil Belajar di IDUKA .....	56
1.	Capaian Sikap .....	56
2.	Capaian Pengetahuan .....	56
3.	Capaian Keterampilan.....	57
BAB IV PENUTUP.....		58
A.	Kesimpulan .....	58
B.	Saran .....	58
1.	Saran Bagi Siswa PKL Berikutnya .....	58
2.	Saran Bagi Sekolah .....	58
3.	Saran Bagi IDUKA .....	59
DAFTAR LAMPIRAN		

## DAFTAR GAMBAR

Gambar 1. 1 Lokasi IDUKA.....	1
Gambar 2. 1 Logo PT Dwansoft.....	2
Gambar 2. 2 Struktur Organisasi .....	5
Gambar 3. 1 Java .....	12
Gambar 3. 2 Php .....	13
Gambar 3. 3 Javascript .....	13
Gambar 3. 4 Css.....	14
Gambar 3. 5 Laravel .....	16
Gambar 3. 6 Yii2.....	16
Gambar 3. 7 Postgresql.....	19
Gambar 3. 8 Mysql .....	20
Gambar 3. 9 iReport .....	21
Gambar 3. 10 Visual Studio Code .....	21
Gambar 3. 11 WinSCP.....	22
Gambar 3. 12 Use Case Diagram .....	23
Gambar 3. 13 Basis Data Gama.....	24
Gambar 3. 14 Login Gama .....	25
Gambar 3. 15 Code Login Gama .....	26
Gambar 3. 16 Dashboard Gama.....	27
Gambar 3. 17 Code Dashboard Gama .....	28
Gambar 3. 18 Index Produk Gama .....	28
Gambar 3. 19 Code Index Produk Gama .....	29
Gambar 3. 20 Form Produk Gama.....	29
Gambar 3. 21 Code Form Produk Gama .....	30
Gambar 3. 22 Index Contact Gama .....	30
Gambar 3. 23 Form Produk Gama.....	31
Gambar 3. 24 Function Setno .....	31
Gambar 3. 25 Function Getno .....	32
Gambar 3. 26 Function Nextno .....	32
Gambar 3. 27 Index Booking.....	33
Gambar 3. 28 Form Booking .....	33
Gambar 3. 29 Form Booking 2 .....	34

Gambar 3. 30 Code Booking .....	34
Gambar 3. 31 Invoice Booking Gama .....	35
Gambar 3. 32 iReport Booking Gama .....	35
Gambar 3. 33 Jadwal Gama .....	37
Gambar 3. 34 Code Jadwal .....	37
Gambar 3. 35 Tombol Sinkron .....	38
Gambar 3. 36 Code Tombol Sinkron .....	38
Gambar 3. 37 Modul API.....	39
Gambar 3. 38 Api Code .....	39
Gambar 3. 39 Index Cash Gama .....	40
Gambar 3. 40 Form Cash Gama .....	41
Gambar 3. 41 Code Cash Gama .....	41
Gambar 3. 42 Index Report Gama .....	42
Gambar 3. 43 Code Report Gama.....	42
Gambar 3. 44 iReport Report Gama .....	42
Gambar 3. 45 Query Report Gama .....	43
Gambar 3. 46 Index User Gama .....	43
Gambar 3. 47 Form User Gama.....	44
Gambar 3. 48 Form User Gama 2.....	44
Gambar 3. 49 Form User Gama 3.....	45
Gambar 3. 50 Form User Gama 4.....	45
Gambar 3. 51 Table User Gama.....	46
Gambar 3. 52 Flowmap .....	47
Gambar 3. 53 Basis Data Crab .....	48
Gambar 3. 54 Login Crab .....	49
Gambar 3. 55 Dashboard Crab .....	50
Gambar 3. 56 Index Produk Crab .....	51
Gambar 3. 57 Form Produk Crab .....	51
Gambar 3. 58 Index Stok Crab .....	51
Gambar 3. 59 Query Stok Crab .....	52
Gambar 3. 60 Index Contact Crab .....	52
Gambar 3. 61 Form Contact Crab.....	53
Gambar 3. 62 Code Form Contact.....	53
Gambar 3. 63 Index Pembelian Crab.....	54
Gambar 3. 64 Dynamic Form Contact.....	54

Gambar 3. 65 Index Penjualan Crab .....	54
Gambar 3. 66 Form Penjualan Crab .....	55
Gambar 3. 67 Cash Penjualan Crab .....	55
Gambar 3. 68 Index Report Crab .....	56



## DAFTAR TABEL

Tabel 1. 1 Jadwal PKL .....	7
-----------------------------	---

# BAB I

## PENDAHULUAN

### A. Latar Belakang

Praktik Kerja Lapangan (PKL) di Sekolah Menengah Kejuruan (SMK) merupakan program yang bertujuan untuk melatih siswa agar dapat mengaplikasikan pengetahuan yang diperoleh di sekolah ke dalam dunia industri. Program ini tidak hanya sebagai tolak ukur pemahaman siswa terhadap materi pelajaran, tetapi juga sebagai sarana untuk mengenal langsung proses kerja di industri dan mempersiapkan diri dalam persaingan di dunia kerja.

Dalam era digital saat ini, kemajuan teknologi telah membawa perubahan besar dalam pengelolaan data dan sistem informasi. Perusahaan membutuhkan sistem yang efisien dan terintegrasi untuk mengelola transaksi, pencatatan data, serta pengolahan informasi guna meningkatkan produktivitas dan akurasi dalam operasional bisnis. Oleh karena itu, pemahaman tentang sistem informasi berbasis web, pengelolaan *database*, serta pembuatan laporan keuangan menjadi keterampilan yang sangat dibutuhkan di dunia kerja.

Keputusan strategis dalam memilih Yii2 sebagai *framework* pengembangan tidak hanya didasarkan pada kehandalan dan kemudahan skalabilitasnya, melainkan juga pada reputasinya yang terbukti memberikan kinerja tinggi, keamanan yang kuat, dan fleksibilitas yang penting dalam mengatasi tantangan pengembangan aplikasi web kompleks di era digital saat ini. Dengan menggunakan Yii2 sebagai fondasi pengembangan, sistem yang dikembangkan dapat lebih optimal dalam mendukung operasional perusahaan, meningkatkan efisiensi kerja, serta memberikan pengalaman pengguna yang lebih baik dalam mengelola transaksi dan data.

Selain itu, implementasi teknologi berbasis web dalam pengelolaan bisnis memberikan banyak keuntungan, seperti aksesibilitas yang lebih luas, kemudahan dalam monitoring data secara real-time, serta peningkatan efektivitas dalam pengelolaan informasi. Dengan adanya sistem yang terstruktur dan terintegrasi, perusahaan dapat mengurangi kesalahan dalam pencatatan data, meningkatkan kecepatan layanan, serta mempercepat proses pengambilan keputusan berdasarkan data yang akurat dan terkini. Oleh karena itu, pemanfaatan teknologi yang tepat sangat diperlukan dalam menghadapi dinamika industri yang terus berkembang.

Dengan berbagai pengalaman tersebut, diperoleh wawasan yang lebih luas mengenai implementasi teknologi informasi dalam dunia industri, khususnya dalam pengelolaan data dan sistem transaksi berbasis web. Pengalaman ini sangat bermanfaat sebagai bekal untuk menghadapi tantangan di dunia kerja yang semakin kompetitif dan berbasis teknologi.

## B. Tujuan PKL

PKL bertujuan untuk mengembangkan kemampuan berpikir kritis dan kreatif bagi siswa yang menjalani program ini. Selain itu, PKL juga bertujuan untuk memberikan pengalaman langsung dalam dunia industri, sehingga siswa dapat memahami bagaimana penerapan teori yang telah dipelajari di sekolah ke dalam lingkungan kerja nyata. Dengan adanya PKL, siswa diharapkan dapat memperoleh keterampilan teknis dan non-teknis yang berguna dalam dunia kerja.

Berdasarkan permasalahan yang telah diuraikan, tujuan PKL dapat dirumuskan sebagai berikut:

1. Meningkatkan pemahaman tentang perancangan dan pengembangan sistem informasi pencatatan transaksi menggunakan Yii2 sebagai *framework*.
2. Memperdalam keterampilan dalam penggunaan *dynamic form* dan *Select2* dalam sistem berbasis Yii2 untuk pengelolaan data.
3. Mengembangkan pemahaman tentang sistem keamanan dalam sistem informasi berbasis Yii2 guna memastikan perlindungan data dan transaksi.
4. Meningkatkan kemampuan dalam mengoptimalkan *query* dalam *PostgreSQL* untuk meningkatkan kinerja sistem.
5. Memahami pembuatan laporan keuangan menggunakan *iReport* yang terintegrasi dengan database guna menunjang efisiensi pencatatan keuangan.

Dengan adanya PKL, siswa diharapkan dapat mengembangkan keterampilan yang dibutuhkan di dunia industri serta memahami bagaimana teknologi informasi dapat diterapkan dalam berbagai aspek bisnis dan operasional perusahaan.

## C. Manfaat PKL

Adapun manfaat dari Program Praktik Kerja Lapangan (PKL) adalah sebagai berikut:

1. Manfaat bagi Peserta Didik

Adapun manfaat bagi Peserta Didik sebagai berikut:

- a. Mengasah keterampilan yang telah diperoleh dengan menerapkan konsep-konsep teoritis dalam lingkungan kerja.
- b. Mengembangkan sikap disiplin, tanggung jawab, dan profesionalisme dalam dunia kerja.
- c. Memperoleh pengalaman langsung dalam industri sehingga lebih siap menghadapi dunia kerja setelah lulus.

- d. Meningkatkan keterampilan teknis dan pemecahan masalah dalam bidang teknologi informasi dan pengelolaan sistem.
- e. Menambah wawasan serta memperluas jaringan profesional yang dapat bermanfaat dalam karier di masa depan.

## 2. Manfaat bagi Sekolah

Adapun manfaat bagi Sekolah sebagai berikut:

- a. Meningkatkan efisiensi dan efektivitas dalam mendidik serta melatih siswa agar memiliki kompetensi yang sesuai dengan kebutuhan industri.
- b. Mempererat kerja sama antara sekolah dan dunia industri (IDUKA), sehingga dapat membuka peluang lebih besar bagi siswa untuk mendapatkan pengalaman kerja.
- c. Memberikan gambaran nyata kepada sekolah mengenai perkembangan industri dan tren teknologi terbaru, yang dapat digunakan untuk menyempurnakan kurikulum pembelajaran.
- d. Meningkatkan reputasi sekolah dalam mencetak lulusan yang siap bersaing di dunia kerja.

## 3. Manfaat bagi IDUKA (Industri dan Dunia Kerja)

Adapun manfaat bagi IDUKA sebagai berikut:

- a. Mendapatkan tenaga kerja magang yang dapat membantu dalam berbagai tugas operasional perusahaan.
- b. Memiliki kesempatan untuk membina dan membentuk calon tenaga kerja yang potensial sesuai dengan kebutuhan industri.
- c. Meningkatkan produktivitas perusahaan dengan adanya tambahan tenaga kerja yang dapat menangani berbagai aspek pekerjaan.
- d. Memperkuat kerja sama dengan institusi pendidikan dalam rangka pengembangan sumber daya manusia yang kompeten.
- e. Meningkatkan citra perusahaan sebagai pihak yang berkontribusi dalam dunia pendidikan dan pengembangan keterampilan generasi muda.

#### **D. Waktu dan Tempat Pelaksanaan PKL**

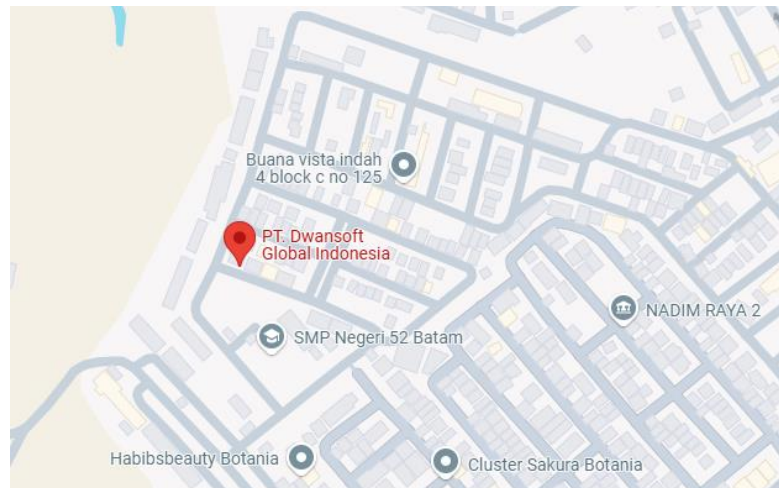
Berikut adalah rincian waktu dan tempat pelaksanaan Praktik Kerja Lapangan (PKL):

##### **1. Waktu Pelaksanaan PKL**

- a. Periode PKL: 4 November 2024 hingga 17 April 2025.
- b. Hari Kerja: Senin hingga Sabtu.
- c. Jam Kerja:
  - 1). Senin hingga Jumat: 08.30 pagi hingga 16.30 sore.
  - 2). Sabtu: 08.30 pagi hingga 12.00 siang.

##### **2. Tempat Pelaksanaan PKL**

- a. Nama Industri Dunia Kerja (IDUKA): PT. Dwansoft Global Indonesia.
- b. Alamat: Cluster Daisy Blok E17 No.18-19, Kelurahan Belian, Kecamatan Batam Kota, Kota Batam, Kepulauan Riau, 29454.



*Gambar 1. 1 Lokasi IDUKA*

## BAB II

### PROFIL INDUSTRI DAN DUNIA KERJA

#### A. Sejarah Berdirinya IDUKA



*Gambar 2. 1 Logo PT Dwansoft*

Seiring dengan pesatnya perkembangan teknologi informasi dan meningkatnya kebutuhan bisnis akan sistem berbasis digital, muncul tantangan dalam pengelolaan data, transaksi, serta integrasi sistem yang efisien. Banyak perusahaan mengalami kendala dalam mengelola operasional secara optimal karena keterbatasan teknologi yang digunakan. Hal ini mendorong perlunya solusi perangkat lunak yang dapat membantu meningkatkan efisiensi, akurasi, serta kecepatan dalam proses bisnis.

Melihat kebutuhan tersebut, PT. Dwansoft Global Indonesia didirikan sebagai perusahaan yang berfokus pada pengembangan perangkat lunak, sistem informasi, dan aplikasi berbasis web serta *mobile*. Dengan visi untuk menyediakan solusi teknologi yang inovatif dan terintegrasi, perusahaan ini mulai mengembangkan berbagai produk dan layanan yang dapat membantu perusahaan dalam mengoptimalkan operasional mereka.

Sejak awal berdirinya, PT. Dwansoft Global Indonesia berkomitmen untuk menghadirkan layanan yang profesional, inovatif, dan sesuai dengan kebutuhan industri. Dengan dukungan tenaga ahli di bidang teknologi informasi, perusahaan ini terus berkembang dan menjalin kerja sama dengan berbagai pihak guna menciptakan solusi digital yang dapat bersaing di era industri 4.0.

Melalui pengalaman dan keahlian dalam pengembangan perangkat lunak, perusahaan ini telah banyak berkontribusi dalam membangun sistem yang membantu berbagai sektor, mulai dari manajemen bisnis, sistem transaksi, hingga integrasi data berbasis cloud. Dengan komitmen yang kuat terhadap kualitas dan kepuasan pelanggan, PT. Dwansoft

Global Indonesia terus berkembang menjadi salah satu penyedia solusi teknologi informasi yang andal di Indonesia.

## **B. Bidang Usaha**

PT. Dwansoft Global Indonesia merupakan perusahaan yang bergerak di bidang pengembangan perangkat lunak dan teknologi informasi. Perusahaan ini fokus pada pembuatan sistem berbasis web dan aplikasi yang dirancang untuk membantu berbagai sektor industri dalam meningkatkan efisiensi operasional, pengelolaan data, serta pelayanan kepada pelanggan.

Sebagai penyedia solusi teknologi, PT. Dwansoft Global Indonesia mengembangkan berbagai produk dan layanan berbasis digital yang dapat disesuaikan dengan kebutuhan bisnis. Produk dan jasa yang dihasilkan meliputi:

### **1. Aplikasi Akuntansi**

Aplikasi ini dirancang untuk membantu perusahaan dalam mengelola keuangan secara lebih efektif dan akurat. Fitur-fitur utama dalam aplikasi akuntansi meliputi pencatatan transaksi keuangan, pembuatan laporan keuangan, pengelolaan pajak, serta analisis keuangan yang dapat membantu dalam pengambilan keputusan bisnis.

### **2. Aplikasi Pinjaman Dana**

Aplikasi ini dikembangkan untuk mendukung layanan keuangan, khususnya dalam bidang pinjaman dana. Dengan sistem yang terintegrasi, aplikasi ini memudahkan proses pengajuan pinjaman, verifikasi data peminjam, perhitungan bunga, serta manajemen pembayaran cicilan secara otomatis dan transparan.

### **3. Aplikasi Toko Online**

*Platform e-commerce* yang memungkinkan bisnis untuk menjual produk secara digital. Aplikasi ini menyediakan fitur seperti katalog produk, sistem pembayaran *online*, manajemen pesanan, serta integrasi dengan sistem logistik untuk pengiriman barang.

### **4. Sistem Informasi Kepegawaian**

Aplikasi ini membantu perusahaan dalam mengelola data karyawan, termasuk informasi pribadi, absensi, penggajian, hingga evaluasi kinerja. Dengan sistem berbasis digital, pengelolaan sumber daya manusia menjadi lebih terstruktur dan efisien.

## 5. Sistem Booking Lapangan Futsal

Sistem ini dirancang untuk memudahkan pengelolaan penyewaan lapangan futsal secara *online*. Fitur utama meliputi pemesanan jadwal lapangan, pembayaran digital, serta manajemen pengguna yang dapat membantu pemilik lapangan dalam mengoptimalkan layanan mereka.

## 6. Sistem Pembelian Kepiting

Sistem ini dikembangkan untuk mendukung industri perikanan, khususnya dalam proses pembelian dan distribusi kepiting. Aplikasi ini mencakup fitur pencatatan stok, manajemen transaksi pembelian, pemantauan ukuran dan kondisi kepiting, serta laporan penjualan yang dapat digunakan untuk analisis bisnis.

### C. Visi dan Misi IDUKA

Setiap perusahaan memiliki visi dan misi yang menjadi pedoman dalam menjalankan usahanya. Visi dan misi ini menjadi arah serta tujuan utama dalam setiap langkah strategis yang diambil oleh perusahaan. PT. Dwansoft Global Indonesia pun memiliki visi dan misi yang mencerminkan komitmen dalam menghadirkan solusi teknologi informasi yang inovatif dan berkualitas.

#### 1. Visi IDUKA

- a. Menjadi perusahaan teknologi informasi yang profesional dan inovatif dengan menghadirkan solusi serta layanan optimal yang memiliki daya saing tinggi.
- b. Berperan aktif dalam perkembangan dunia teknologi informasi dengan menyediakan layanan yang terintegrasi, efisien, dan berorientasi pada kebutuhan pelanggan.

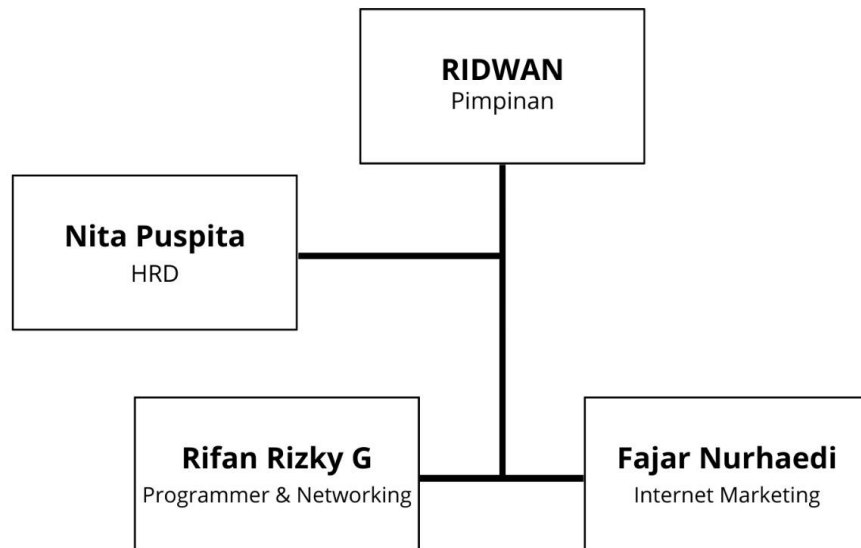
#### 2. Misi IDUKA

- a. Menyediakan layanan teknologi informasi yang terpadu dan dapat diandalkan dalam mendukung kebutuhan bisnis berbagai sektor industri.
- b. Mengembangkan produk dan layanan berkualitas tinggi dengan memberikan layanan purna jual yang responsif untuk memastikan kepuasan pelanggan.
- c. Menjunjung tinggi profesionalisme serta mengembangkan kompetensi individu dan tim guna menciptakan solusi teknologi yang unggul dan inovatif.
- d. Memberikan solusi yang optimal dan efisien bagi mitra kerja dalam menghadapi tantangan dan permasalahan di dunia teknologi informasi.
- e. Terus berinovasi dalam pengembangan produk berbasis teknologi modern untuk mendukung transformasi digital di berbagai bidang usaha.



Dengan visi dan misi ini, PT. Dwansoft Global Indonesia terus berupaya untuk menjadi perusahaan yang tidak hanya berorientasi pada kemajuan teknologi, tetapi juga mampu memberikan dampak positif bagi dunia industri serta masyarakat luas.

#### D. Struktur Organisasi IDUKA



Gambar 2. 2 Struktur Organisasi

PT Dwansoft Global Indonesia memiliki struktur organisasi yang dirancang untuk mendukung efektivitas operasional perusahaan dalam mengembangkan serta menyediakan layanan teknologi informasi. Perusahaan ini dipimpin oleh seorang *Chief Executive Officer* (CEO) yang bertanggung jawab atas seluruh strategi dan arah perusahaan. Dalam menjalankan operasionalnya, CEO didukung oleh beberapa departemen utama, yaitu *Human Resource Department* (HRD), *Programmer & Networking*, dan *Internet Marketing*.

##### 1. *Chief Executive Officer* (CEO)

Sebagai pemimpin tertinggi dalam perusahaan, CEO bertanggung jawab dalam merancang visi dan misi perusahaan serta mengarahkan strategi bisnis agar tetap kompetitif di industri teknologi. Tugas utama CEO meliputi:

- Menentukan kebijakan serta arah perkembangan perusahaan.
- Memimpin dan memotivasi tim untuk mencapai tujuan strategis perusahaan.
- Mengembangkan strategi bisnis dan membangun relasi dengan investor serta mitra kerja.
- Mengawasi pengelolaan anggaran dan memastikan efisiensi keuangan perusahaan.

## 2. *Human Resource Department (HRD)*

HRD memiliki peran penting dalam mengelola sumber daya manusia agar perusahaan dapat berjalan dengan efektif. Tanggung jawab utama HRD antara lain:

- a. Melakukan proses rekrutmen dan seleksi guna mendapatkan karyawan terbaik.
- b. Mengelola pelatihan serta pengembangan karyawan agar selalu memiliki keterampilan yang relevan.
- c. Menjaga hubungan kerja yang harmonis di dalam perusahaan serta memastikan kepatuhan terhadap peraturan ketenagakerjaan.
- d. Mengatur kebijakan kesejahteraan karyawan, termasuk kompensasi dan tunjangan.

## 3. *Programmer & Networking*

Departemen ini merupakan inti dari pengembangan produk perusahaan, yang berfokus pada pembuatan serta pemeliharaan aplikasi dan infrastruktur jaringan.

Tanggung jawab utama departemen ini meliputi:

- a. Mengembangkan, menguji, dan memelihara aplikasi berbasis web maupun *mobile*.
- b. Memastikan keamanan dan stabilitas jaringan perusahaan serta layanan yang diberikan kepada klien.
- c. Mengatasi permasalahan teknis yang berkaitan dengan pengembangan *software* dan infrastruktur TI.
- d. Berkolaborasi dengan tim lain untuk menghasilkan solusi berbasis teknologi yang efisien.

## 4. *Internet Marketing*

Departemen ini berperan dalam meningkatkan visibilitas dan daya saing perusahaan melalui strategi pemasaran digital. Beberapa tanggung jawab utama dalam bidang ini antara lain:

- a. Menyusun dan menjalankan strategi pemasaran *online* untuk meningkatkan penjualan produk dan layanan.
- b. Mengelola promosi digital melalui berbagai platform seperti media sosial, email *marketing*, dan iklan berbasis internet.
- c. Membangun serta menjaga hubungan baik dengan klien dan mitra bisnis melalui komunikasi digital.
- d. Menganalisis data pemasaran guna mengoptimalkan strategi yang lebih efektif.

## E. Budaya Kerja dan Tata Tertib Karyawan di IDUKA

Budaya kerja di PT Dwansoft Global Indonesia mengedepankan profesionalisme, disiplin, dan kerja sama tim untuk mencapai kinerja yang optimal. Setiap karyawan diharapkan memiliki sikap yang bertanggung jawab, proaktif, serta berorientasi pada hasil dalam setiap tugas yang diberikan. Selain itu, perusahaan juga menerapkan tata tertib yang bertujuan untuk menciptakan lingkungan kerja yang kondusif dan produktif.

### 1. Nilai-Nilai Budaya Kerja

Budaya kerja yang diterapkan di perusahaan ini mencakup beberapa aspek utama:

- Profesionalisme: Setiap karyawan diharapkan bekerja secara efisien, bertanggung jawab, dan sesuai dengan etika profesi.
- Disiplin dan Integritas: Menjunjung tinggi kedisiplinan dalam kehadiran, pelaksanaan tugas, serta kepatuhan terhadap peraturan yang berlaku.
- Kerja Sama Tim: Mendorong komunikasi yang baik serta kolaborasi antar karyawan dalam menyelesaikan tugas dan mencapai target perusahaan.
- Inovasi dan Adaptasi: Mengembangkan ide-ide baru dalam teknologi dan terus belajar untuk meningkatkan keterampilan sesuai perkembangan zaman.
- Pelayanan Prima: Mengutamakan kepuasan klien dan memberikan solusi terbaik bagi mitra bisnis perusahaan.

### 2. Tata Tertib dan Peraturan Perusahaan

Untuk menjaga kedisiplinan serta menciptakan lingkungan kerja yang harmonis, PT Dwansoft Global Indonesia menerapkan peraturan berikut:

#### a. Jam Kerja

Jam Kerja	Keterangan
08.30 s/d 16.30	Senin s/d Jumat
08.30 s/d 12.00	Sabtu
Libur	Minggu

*Tabel 1. 1 Jadwal PKL*

#### b. Kewajiban dan Etika Kerja

- Karyawan diwajibkan hadir sesuai dengan jadwal kerja yang telah ditetapkan dan absen sebelum serta sesudah bekerja.
- Jika mengalami keterlambatan, harus meminta izin terlebih dahulu kepada pimpinan perusahaan.
- Jika berhalangan hadir, wajib melaporkan ketidakhadirannya sejak hari pertama dengan cara apa pun yang memungkinkan.

- 4). Setiap karyawan wajib menyelesaikan tugas dan tanggung jawab yang diberikan dengan penuh dedikasi.
- 5). Berpakaian sopan dan rapi sesuai dengan ketentuan perusahaan.
- 6). Menjaga kebersihan dan kerapihan lingkungan kerja.
- 7). Menjaga kerahasiaan data dan informasi perusahaan.

Penerapan budaya kerja dan tata tertib ini bertujuan untuk menciptakan lingkungan kerja yang kondusif, profesional, serta mendukung pertumbuhan perusahaan dalam industri teknologi informasi.

## **BAB III**

### **PROSES DAN HASIL BELAJAR DI IDUKA**

#### **A. Proses Belajar di IDUKA**

Selama melaksanakan Praktik Kerja Lapangan (PKL), berbagai teknologi dan konsep sistem informasi diterapkan dalam pekerjaan. Pemahaman mengenai sistem, sistem informasi, database, serta bahasa pemrograman menjadi dasar dalam mengembangkan aplikasi yang digunakan di industri.

##### **1. Konsep Dasar yang Dipelajari**

###### **a. Perancangan Sistem**

Menurut Rosa A.S dan M. Shalahuddin (dalam Rezki, 2018), perancangan sistem dalam konteks pembangunan perangkat lunak melibatkan upaya untuk merancang sistem yang memenuhi berbagai spesifikasi kebutuhan fungsional. Hal ini mencakup pemenuhan target, kebutuhan yang disampaikan secara terbuka atau tersirat terkait dengan performa sistem, serta penggunaan sumber daya yang efisien. Perancangan sistem juga harus mempertimbangkan batasan yang ada, termasuk batasan biaya, waktu, dan perangkat yang tersedia. Proses ini bertujuan untuk menjamin bahwa sistem yang dikembangkan dapat memberikan kepuasan kepada pengguna dan pemangku kepentingan lainnya.

Menurut Kendall (dalam Asyhadi, 2021), perancangan sistem melibatkan penyiapan file atau basis data yang mampu menyimpan informasi yang esensial bagi para pengambil keputusan. Perlu diperhatikan bahwa perancangan sistem tidak hanya terbatas pada penyusunan struktur file atau basis data, tetapi juga melibatkan pemilihan metode penyimpanan yang efisien dan strategi pengelolaan data yang dapat mendukung kebutuhan pengambil keputusan.

Berdasarkan pengertian para ahli di atas, dapat disimpulkan perancangan sistem merupakan suatu proses yang melibatkan identifikasi kebutuhan data, penggambaran, perencanaan, dan pembuatan gambaran dari berbagai komponen yang terpisah dalam suatu sistem. Tujuan utamanya adalah untuk memastikan bahwa sistem tersebut dapat memenuhi kebutuhan penggunanya secara efektif dan efisien serta menciptakan sebuah entitas yang lengkap dan beroperasi dengan efisien. Dalam konteks pembangunan

perangkat lunak, perancangan sistem juga melibatkan usaha untuk mengembangkan sistem yang memenuhi berbagai spesifikasi kebutuhan fungsional. Salah satu aspek penting dari perancangan sistem adalah penyiapan file atau basis data yang mampu menyimpan informasi yang esensial bagi para pengambil keputusan, sehingga mendukung proses pengambilan keputusan yang akurat dan tepat.

Perancangan sistem bertujuan untuk mencapai kebutuhan dan tujuan yang telah ditentukan sebelumnya oleh organisasi atau entitas yang bersangkutan, serta untuk menyusun suatu struktur atau kerangka kerja yang optimal dalam mengelola informasi, memproses data, dan menjalankan fungsi-fungsi tertentu sesuai dengan kebutuhan yang spesifik. Merancang sistem secara cermat, diharapkan dapat meningkatkan kinerja, produktivitas, dan efisiensi organisasi, serta memungkinkan untuk menyediakan layanan atau produk dengan kualitas yang lebih baik kepada pengguna akhir. Selain itu, perancangan sistem juga bertujuan untuk memastikan keselarasan antara berbagai komponen teknologi informasi, proses bisnis, dan kebijakan yang ada dalam suatu organisasi, sehingga dapat mendukung pencapaian tujuan strategis secara keseluruhan.

#### b. Sistem

Menurut Bambang Hartono (dalam Putri dan Khana, 2021), sistem adalah kumpulan berbagai bagian dan elemen yang terorganisasi dan saling terhubung berdasarkan fungsinya, membentuk satu kesatuan yang terorganisir. Pengorganisasian sistem secara efektif memerlukan pembagian tugas dan tanggung jawab yang jelas antara berbagai bagian dan elemen dalam sistem tersebut, serta koordinasi yang baik antara mereka untuk mencapai tujuan bersama.

Berdasarkan pengertian ahli di atas, dapat disimpulkan bahwa sistem adalah kumpulan elemen yang saling berinteraksi dan terorganisasi dalam suatu kesatuan untuk mencapai tujuan tertentu. Sistem terdiri dari dua atau lebih komponen yang bekerja bersama dan saling terhubung, baik secara internal maupun dengan lingkungan eksternal. Sistem juga mungkin terbagi menjadi bagian yang lebih kecil dari sistem yang mendukung struktur yang lebih besar. Integritas, keandalan, dan koordinasi antara elemen-elemen

sistem menjadi kunci dalam mencapai keberhasilan dalam pencapaian tujuan sistem.

c. Web

Menurut Boone dan Kurtz (dalam Mohamad, 2022), web adalah sebuah kumpulan beragam informasi yang disajikan dalam format visual yang terhubung satu sama lain melalui jaringan internet yang luas. Menurut Andi (Mohamad, 2022), web merupakan bentuk komunikasi interaktif yang digunakan di dalam jaringan komputer, memungkinkan pengguna untuk berinteraksi dengan berbagai konten multimedia dan berbagi informasi secara global.

Menurut Bakti (dalam Ochi, Arie, dan Randy, 2020), web adalah sekumpulan halaman yang dirancang untuk menampilkan berbagai jenis informasi seperti teks, gambar diam atau bergerak, animasi, serta suara. Halaman-halaman ini dapat bersifat statis atau dinamis, dan saling terhubung membentuk struktur yang terorganisir, dengan masing-masing terhubung ke jaringan halaman lainnya. Menurut Asropudin (dalam Muchamad, Sutarman, dan Deri, 2019), web adalah serangkaian halaman dimulai dari halaman utama yang memuat informasi, iklan, dan program aplikasi.

Berdasarkan definisi para ahli di atas, dapat disimpulkan bahwa web adalah sebuah sistem informasi yang terdiri dari kumpulan halaman terhubung melalui internet, dirancang untuk menampilkan berbagai jenis konten multimedia seperti teks, gambar, animasi, dan suara. Pengguna dapat berinteraksi dengan konten tersebut dan berbagi informasi secara global yang memungkinkan pengguna untuk mengakses informasi dan layanan dengan lebih cepat dan mudah dari berbagai perangkat.

d. Bahasa Pemrograman

Menurut pandangan Kusrini dan Koniyo (dalam Ramadhan, 2024), Bahasa pemrograman mengacu pada serangkaian instruksi yang dapat dipahami oleh komputer untuk menjalankan tugas-tugas tertentu. Ini menggambarkan bahwa bahasa pemrograman adalah kumpulan perintah yang memungkinkan komputer untuk memproses dan menjalankan fungsi-fungsi yang diinginkan. Bahasa pemrograman memiliki peran utama dalam proses

pengembangan berbagai aplikasi dan sistem komputer, dari perangkat lunak hingga aplikasi web dan mobile.

Menurut Fitria (2022), bahasa pemrograman adalah rangkaian simbol dan peraturan yang digunakan untuk mengirimkan instruksi kepada komputer dalam bentuk program. Definisi ini menekankan bahwa bahasa pemrograman merupakan kerangka kerja simbolik yang memungkinkan para pengembang untuk mengekspresikan logika, algoritma, dan langkah-langkah yang diinginkan agar dapat dieksekusi oleh komputer sesuai kebutuhan aplikasi yang dikembangkan. Pentingnya bahasa pemrograman terletak pada kemampuannya untuk memungkinkan komunikasi antara manusia dan mesin, sehingga memfasilitasi pengembangan berbagai jenis perangkat lunak dan aplikasi komputer.

Berdasarkan definisi para ahli di atas, dapat disimpulkan bahwa bahasa pemrograman adalah sistem sintaksis yang memungkinkan programmer untuk menuliskan serangkaian instruksi dalam komputer. Ini memungkinkan para programmer untuk membuat program aplikasi dengan menggunakan kumpulan perintah yang dapat dipahami oleh komputer. Bahasa pemrograman juga merupakan kumpulan simbol dan aturan yang digunakan untuk menyampaikan instruksi kepada komputer dalam bentuk program.

e. Bahasa Pemrograman yang Digunakan

Terdapat banyak jenis bahasa pemrograman yang digunakan dalam pengembangan perangkat lunak. Beberapa jenis bahasa pemrograman yang digunakan oleh IDUKA adalah sebagai berikut.

1) Java



*Gambar 3. 1 Java*

Java adalah bahasa pemrograman yang dikembangkan oleh James Gosling di Sun Microsystems pada tahun 1995, setelah diakuisisi oleh Oracle Corporation pada tahun 2010, Java terus berkembang dengan



rilis reguler dan digunakan luas dalam berbagai industri, termasuk pengembangan perangkat lunak perusahaan, aplikasi *mobile*, permainan, dan sistem terdistribusi.

## 2) PHP



*Gambar 3. 2 Php*

PHP (Hypertext Preprocessor) dikembangkan oleh Rasmus Lerdorf pada tahun 1994 sebagai solusi untuk membuat halaman web pribadi dengan lebih mudah. Rasmus Lerdorf awalnya menciptakan kumpulan skrip CGI (*Common Gateway Interface*) dalam bahasa C yang memungkinkan dia untuk melacak pengunjung halaman webnya. Seiring waktu, skrip ini berkembang menjadi PHP yang kita kenal sekarang. Sejak itu, PHP telah menjadi bahasa pemrograman yang sangat populer untuk pengembangan aplikasi web dinamis dan berbasis server.

## 3) Javascript



*Gambar 3. 3 Javascript*

JavaScript, awalnya dikembangkan oleh Brendan Eich di Netscape pada tahun 1995, diciptakan sebagai bahasa skrip untuk meningkatkan interaktivitas halaman web. Digunakan untuk pengembangan aplikasi web, pengembangan front-end, pengembangan back-end, serta pengembangan aplikasi mobile dan permainan.

#### 4) Cascading Style Sheets



Gambar 3. 4 Css

CSS (*Cascading Style Sheets*) pertama kali dikembangkan oleh Håkon Wium Lie pada tahun 1996 sebagai standar untuk mendesain tampilan halaman web. CSS digunakan untuk mengontrol tata letak, warna, font, dan elemen visual lainnya pada sebuah situs web. Saat ini, CSS banyak digunakan dalam pengembangan antarmuka web, desain responsif, animasi web, serta penyesuaian tampilan aplikasi mobile dan desktop berbasis web.

#### f. Framework

Menurut Raharjo (dalam Ade, 2020), *framework* adalah sebuah rangkaian kode yang terdiri dari pustaka (*library*) dan alat (*tool*) yang digabungkan secara terstruktur untuk membentuk suatu kerangka kerja (*framework*). Tujuan dari *framework* tersebut adalah untuk mempermudah dan mempercepat proses pengembangan aplikasi web. Adanya *framework*, pengembang dapat mengakses berbagai fungsi dan komponen yang telah disediakan.

Menurut Yudhanto dan Prasetyo (dalam A, 2020), *framework* merupakan komponen pemrograman yang tersedia untuk digunakan kembali dengan mudah, sehingga para pengembang tidak perlu menulis skrip yang sama berulang kali untuk tugas yang serupa sehingga pengembang dapat menghemat waktu dan upaya dalam proses pengembangan aplikasi, karena mereka dapat mengakses fungsi-fungsi dan modul-modul yang sudah ada dalam *framework* tersebut untuk memenuhi kebutuhan aplikasi yang sedang dibangun.

Menurut Saputra (dalam A, 2020), *framework* merupakan kerangka kerja yang terdiri dari struktur dasar yang dapat digunakan kembali (*reusable*) dalam pemrograman berorientasi objek. *Framework* ini terdiri dari kelas-kelas abstrak dan kelas konkret yang membantu dalam membangun aplikasi

dengan cara yang terstruktur. Menggunakan *framework*, pengembang dapat memanfaatkan struktur dasar yang sudah ada untuk mempercepat proses pengembangan dan memastikan konsistensi dalam pengembangan aplikasi berbasis objek.

Berdasarkan definisi para ahli di atas, dapat disimpulkan bahwa *framework* adalah sebuah struktur dasar yang dapat digunakan kembali dalam pemrograman berorientasi objek. *Framework* ini merupakan komponen pemrograman yang siap digunakan ulang untuk memudahkan pembuatan aplikasi dengan menyediakan kumpulan potongan program yang terorganisir secara terpadu, mengurangi kebutuhan untuk membuat kode yang sama berulang kali, dan mempercepat proses pengembangan aplikasi web. Artinya *framework* dapat membantu dalam pembuatan aplikasi dengan menyediakan struktur dan fungsi dasar yang telah tersedia sehingga pengembang tidak perlu memulai dari awal setiap kali membuat aplikasi baru.

*Framework* juga memiliki fungsi untuk mempromosikan praktik-praktik terbaik dalam pengembangan perangkat lunak. Hal ini dapat mencakup penggunaan desain arsitektur yang baik, penerapan pola-pola desain yang telah terbukti, serta pemisahan antara logika bisnis dan presentasi data. *Framework* membantu meningkatkan kualitas perangkat lunak yang dihasilkan, serta mempermudah proses pengembangan tim yang terdiri dari beberapa pengembang. Selain itu, *framework* juga menyediakan dokumentasi yang lengkap dan komprehensif, serta komunitas pengguna yang aktif. Ini membantu pengembang untuk memahami dan memanfaatkan *framework* secara efektif, serta mempercepat proses pembelajaran dan pemecahan masalah saat mengembangkan aplikasi menggunakan *framework* tersebut.

g. *Framework* yang Digunakan

Berikut adalah beberapa contoh *framework* yang digunakan oleh pengembang *website* dengan bahasa pemrograman PHP.

1) Laravel



Gambar 3. 5 Laravel

Laravel adalah *framework* PHP yang pertama kali dirilis pada tahun 2011 oleh Taylor Otwell, dengan tujuan menyederhanakan dan mempercepat proses pengembangan web menggunakan PHP. Dikenal dengan fitur-fitur kuat seperti sistem *routing*, ORM (*Object-Relational Mapping*), *templating Blade*, migrasi *database*, serta berbagai fitur bawaan seperti otentikasi pengguna, *caching*, dan pengujian yang terintegrasi.

2) Yii



Gambar 3. 6 Yii2

Yii adalah *framework* PHP yang pertama kali dirilis pada tahun 2008 oleh Qiang Xue. Merupakan kelanjutan dari Yii *framework* versi sebelumnya, Yii didesain dengan fokus pada kinerja yang tinggi, fleksibilitas, dan keamanan. *Framework* ini menyediakan berbagai fitur termasuk sistem ORM (*Object Relational Mapping*) yang kuat, pembangkit kode (*code generator*) otomatis, dan integrasi yang mudah dengan teknologi-teknologi *modern* seperti jQuery dan Bootstrap.

#### h. *Yii Framework*

Menurut Ari Budi Warsito (dalam Zar, 2018), Yii adalah sebuah *framework* PHP berbasis-komponen yang memiliki kinerja tinggi dan dirancang untuk memudahkan pengembangan aplikasi web berskala besar. Menggunakan Yii, pengembang dapat membangun aplikasi web yang kuat dan efisien dengan cepat dan mudah, berkat fitur-fitur yang disediakan oleh kerangka kerja ini. Menurut penjelasan Samuel Demoen (dalam Zar, 2018), Yii merupakan sebuah *framework* PHP yang tersusun dengan baik dan dilengkapi dengan berbagai fitur penting untuk membangun aplikasi web masa kini. Menurut Chris Backhouse (dalam Femi, 2018), Yii adalah sebuah kerangka kerja PHP yang memiliki kecepatan dan efisiensi tinggi, serta didukung oleh komunitas pengembang yang aktif. Menurut Praptiningsih dan Bambang Eka Purnama (dalam Femi, 2018), Yii adalah platform pengembangan aplikasi web berbasis PHP yang dapat disesuaikan dan simpel, serta ideal untuk berbagai jenis web.

Berdasarkan definisi para ahli di atas, dapat disimpulkan bahwa Yii adalah sebuah *framework* PHP yang sangat efisien, terstruktur dengan baik, dan mudah digunakan. Yii menawarkan berbagai fitur yang berguna untuk pengembangan aplikasi web *modern*, sehingga cocok digunakan untuk proyek-proyek berskala besar maupun kecil. Kecepatan dan efisiensi dalam pengembangan aplikasi web juga menjadi salah satu keunggulan Yii, yang didukung oleh komunitas pengembang yang aktif.

Yii adalah kerangka kerja PHP yang kuat dan terstruktur secara modular, dirancang untuk memudahkan pengembangan aplikasi web *modern*. Konsep komponen berbasis, Yii memungkinkan pengembang untuk menggunakan dan menggabungkan komponen-komponen yang diperlukan dengan mudah, menjadikan pengembangan aplikasi lebih terstruktur dan mudah dipelihara. Pola desain *Model-View-Controller* (MVC) memisahkan logika aplikasi menjadi tiga bagian utama, Model untuk mengelola data, *View* untuk menampilkan informasi kepada pengguna, dan *Controller* untuk mengatur interaksi antara Model dan *View*. Selain itu, Yii dilengkapi dengan Gii, sebuah *code generator* yang menghasilkan kode otomatis untuk model, *controller*, *form*, dan operasi CRUD, membantu pengembang menghemat waktu dalam membuat kode dasar aplikasi.

Yii juga menyediakan *Active Record*, yang memfasilitasi interaksi dengan basis data menggunakan objek yang merepresentasikan tabel-tabel dalam basis data, serta *Dependency Injection Container* yang membantu dalam efisien mengatur ketergantungan antara komponen-komponen aplikasi. Dukungan untuk RESTful API memungkinkan pembangunan dan penerapan API (*Application Programming Interface*) dengan cepat dan efisien menggunakan konvensi standar REST. Selain itu, Yii menyediakan berbagai *widget* untuk membangun antarmuka pengguna yang dinamis dan interaktif, serta mendukung konsep tema untuk memudahkan pengembangan tata letak dan tampilan aplikasi.

i. Basis Data

Menurut Rosa dan Shalahuddin (dalam Della, 2019), basis data merupakan sebuah sistem yang digunakan untuk menyimpan data dengan tujuan memudahkan dan mempercepat akses terhadap data tersebut. Menurut Anhar (dalam Syafnidawaty, 2020), basis data merupakan koleksi tabel yang berisi data dan terdiri dari *field* atau kolom. Struktur file yang membentuk basis data mencakup data *record* dan *field*. Menurut Connolly (dalam Andy, 2008), basis data adalah suatu kumpulan data yang terhubung secara logis, dapat dimanfaatkan bersama, dan deskripsi data ini dibuat untuk memenuhi keperluan yang ada. Basis data ini sering digunakan dalam aplikasi-aplikasi seperti sistem manajemen basis data (DBMS) untuk mengorganisir dan mengelola informasi secara efisien. Menurut Date (dalam Andy, 2008), basis data adalah kumpulan data yang digunakan secara berkelanjutan oleh sistem aplikasi dari suatu perusahaan.

Berdasarkan pengertian para ahli yang di atas, dapat disimpulkan basis data adalah sistem yang diciptakan untuk menyimpan, mengelola, dan mengatur data secara terstruktur dengan tujuan memudahkan penggunaan dan akses data. Hal ini sering digunakan dalam berbagai konteks seperti bisnis, organisasi, dan aplikasi teknologi informasi untuk menyediakan informasi yang terorganisir dengan baik dan memenuhi kebutuhan pengguna. struktur yang mencakup data *record* dan *field*, serta koneksi logis antara data, *database* memungkinkan penggunaan dan akses data yang efisien oleh berbagai sistem aplikasi dalam suatu entitas.

j. Basis Data PostgreSQL



Gambar 3. 7 Postgresql

PostgreSQL adalah sistem manajemen basis data (DBMS) objek relasional yang *open-source*. PostgreSQL menawarkan dukungan yang kuat untuk bahasa SQL dan menyediakan fitur-fitur canggih seperti transaksi, kunci asing, tipe data kustom, dan banyak lagi. PostgreSQL juga terkenal karena keandalannya, fleksibilitasnya, dan kemampuannya untuk menangani beban kerja yang kompleks. Banyak organisasi dan perusahaan besar menggunakan PostgreSQL untuk aplikasi-aplikasi kritis mereka karena performa dan keamanannya yang tinggi.

Menurut Mochamad Alfian Rosid (dalam Anggita, 2021), PostgreSQL merupakan Sistem Manajemen Basis Data Relasional Objek yang bersifat *open source*. Menurut Fauzan Ardiansyah dan Munawaroh (2023), PostgreSQL adalah sebuah sistem basis data yang open source, memungkinkan penggunaannya tanpa melanggar hukum terkait lisensi karena dapat digunakan secara bebas dan untuk publik. Menurut Adithya Gilang Maulana, Dyah Ayu Irawati, dan Budi Harijanto (dalam Anggita, 2021), PostgreSQL merupakan salah satu perangkat lunak aplikasi basis data yang menggabungkan aspek objek relasional. Meskipun demikian, PostgreSQL tetap mempertahankan fitur-fitur khas yang ditemukan dalam sistem basis data tradisional. Hal ini menunjukkan bahwa PostgreSQL memiliki kemampuan relasional dan objek yang luas. Menurut Ariffud Muhammad (2021) PostgreSQL adalah sistem manajemen basis data objek relasional yang didasarkan pada *open source*, dan telah menjadi salah satu RDBMS (*Relational Database Management System*) yang sangat terkenal di dunia.

Berdasarkan pengertian para ahli yang di atas, dapat disimpulkan PostgreSQL adalah sistem manajemen basis data relasional objek yang *open source*. Penggunaannya dapat dilakukan tanpa melanggar hukum terkait

lisensi, sehingga bisa digunakan secara bebas oleh publik. PostgreSQL menggabungkan aspek objek relasional dan mempertahankan fitur-fitur khas sistem basis data tradisional seperti dukungan untuk bahasa SQL yang kuat, transaksi, kunci asing, dan kemampuan untuk menjaga integritas referensial data.

k. Basis Data MySQL



Gambar 3. 8 Mysql

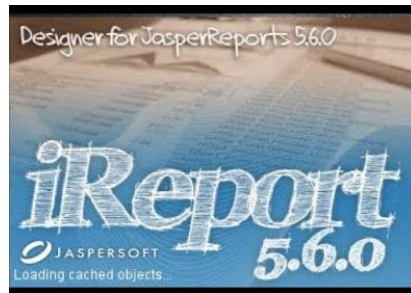
MySQL adalah salah satu sistem manajemen basis data (DBMS) relasional yang paling populer di dunia. Dikembangkan oleh perusahaan Oracle, MySQL mendukung banyak platform sistem operasi dan bahasa pemrograman. MySQL menggunakan bahasa SQL (*Structured Query Language*) untuk mengakses dan memanipulasi data dalam *database*. MySQL sering digunakan dalam berbagai aplikasi web dan aplikasi bisnis karena keandalannya, performa tinggi, dan kemampuan untuk mengelola volume data yang besar.

l. Teknologi dan Alat yang Digunakan

Dalam pengembangan perangkat lunak dan pengelolaan sistem, berbagai alat digunakan untuk mendukung efisiensi kerja. Beberapa alat yang sering digunakan meliputi editor kode, aplikasi transfer file, dan perangkat lunak pembuatan laporan.



## 1) iReport



Gambar 3. 9 iReport

iReport, dikembangkan oleh Jaspersoft, adalah sebuah alat desain laporan berbasis GUI yang digunakan untuk membuat laporan dalam format JasperReports. Pertama kali dirilis pada awal 2000-an, iReport digunakan untuk mendesain dan menghasilkan laporan dalam berbagai format seperti PDF, Excel, HTML, dan lainnya. Alat ini banyak digunakan dalam pengelolaan data perusahaan, sistem informasi bisnis, serta aplikasi berbasis Java dan *database* untuk pembuatan laporan yang kompleks dan interaktif.

## 2) Visual Studio Code



Gambar 3. 10 Visual Studio Code

Visual Studio Code (VS Code), dikembangkan oleh Microsoft dan pertama kali dirilis pada tahun 2015, adalah sebuah kode editor sumber terbuka yang mendukung berbagai bahasa pemrograman. VS Code banyak digunakan dalam pengembangan web, aplikasi berbasis *cloud*, pengelolaan kode sumber, serta debugging aplikasi. Dengan fitur seperti IntelliSense, integrasi Git, dan dukungan ekstensi yang luas, VS Code menjadi salah satu editor yang populer di kalangan *developer*.

### 3) WinSCP



Gambar 3. 11 WinSCP

WinSCP (*Windows Secure Copy*), pertama kali dirilis oleh Martin Přikryl pada tahun 2000, adalah sebuah klien SFTP dan FTP untuk Windows yang digunakan dalam transfer file yang aman antara komputer lokal dan server jarak jauh. WinSCP sering digunakan oleh pengembang dan *administrator* sistem untuk mengunggah, mengelola, serta menyinkronkan file di server, terutama dalam pengembangan web dan administrasi sistem berbasis Linux atau Unix.

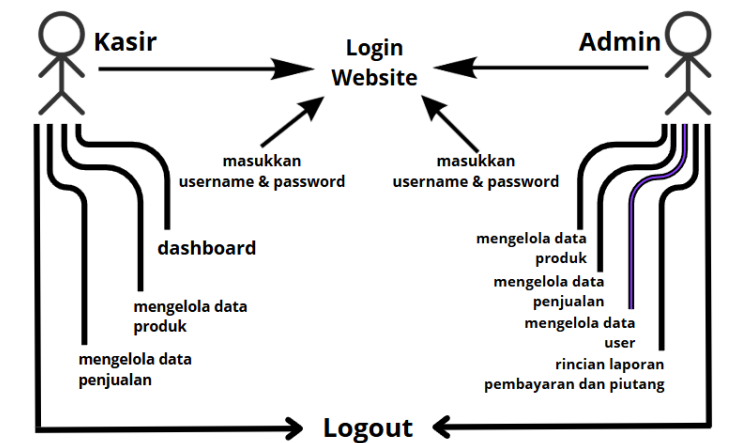
## 2. Penerapan Teknologi dalam Sistem Website Booking dan Transaksi

Menggambarkan bagaimana teknologi diaplikasikan dalam pengembangan sistem berbasis web yang digunakan untuk kebutuhan *booking* dan transaksi. Dalam dunia industri, sistem berbasis web menjadi solusi yang efisien untuk mengotomatisasi proses bisnis, meningkatkan akurasi data, serta memberikan pengalaman yang lebih baik bagi pengguna. Penerapan teknologi dalam sistem ini melibatkan berbagai aspek, mulai dari penggunaan framework Yii2 sebagai fondasi pengembangan, PostgreSQL sebagai basis data untuk menyimpan informasi dengan aman dan terstruktur, hingga pemanfaatan berbagai alat bantu seperti Visual Studio Code untuk pengembangan kode, WinSCP untuk transfer file ke server, dan iReport untuk pembuatan laporan transaksi.

Dalam prosesnya, sistem dikembangkan dengan menerapkan konsep pemrograman berbasis objek, manajemen database yang optimal, serta penerapan teknologi web seperti CSS, dan JavaScript guna meningkatkan tampilan dan interaktivitas. Dengan kombinasi teknologi ini, sistem dapat berjalan secara efisien, mendukung operasional bisnis, serta memberikan kemudahan bagi pengguna dalam melakukan *booking* lapangan.

a. Proses Perancangan

Proses perancangan sistem informasi pencatatan penjualan mencakup penyusunan alur kerja sistem, pembuatan diagram UML, pengembangan ERD, *flowmap*, normalisasi data, dan pembuatan *database* serta antarmuka pengguna. Melibatkan gambaran umum tentang operasi sistem, struktur dan interaksi antar komponen, model hubungan entitas dalam *database*, optimasi struktur basis data, implementasi struktur *database*, dan pembuatan antarmuka pengguna yang mudah digunakan. Berikut ini adalah perancangan yang menggunakan *Use Case Diagram* untuk *Website Transaksi Booking* :



Gambar 3. 12 Use Case Diagram

Kasir mengakses sistem dengan memasukkan *username* dan *password* untuk masuk ke dalam *dashboard*. Setelah *login*, kasir dapat mengelola data produk dan mencatat transaksi penjualan. Hak akses kasir terbatas pada operasional penjualan tanpa kewenangan untuk mengelola pengguna atau melihat laporan keuangan.

Admin juga masuk ke dalam sistem dengan *login* menggunakan *username* dan *password*. Namun, admin memiliki hak akses yang lebih luas dibandingkan kasir. Selain mengelola data produk dan penjualan, admin juga bertanggung jawab atas manajemen pengguna, melihat rincian laporan, serta mengelola pembayaran dan piutang.

Baik kasir maupun admin dapat keluar dari sistem dengan melakukan *logout* setelah selesai menggunakan aplikasi. Diagram ini memberikan gambaran bagaimana sistem membagi peran dan tugas antara kasir dan admin dalam pengelolaan transaksi dan data perusahaan.

## b. Pembuatan Database

Pembuatan database untuk sistem informasi *booking* dan transaksi lapangan merupakan langkah kunci dalam pengembangan sistem yang efisien dan terstruktur. Dengan menggunakan PostgreSQL sebagai sistem manajemen basis data, sistem ini dapat menyediakan lingkungan penyimpanan data yang handal dan aman. PostgreSQL dipilih karena kemampuannya dalam menangani beban kerja yang tinggi serta fitur-fiturnya yang mendukung kueri kompleks dan pengelolaan akses yang fleksibel.

Melalui pembuatan *database* ini, sistem bertujuan untuk memastikan bahwa proses *booking* dan transaksi lapangan berjalan dengan lancar, mulai dari pencatatan jadwal pemesanan hingga proses pembayaran. Struktur *database* dirancang agar dapat menangani volume data yang besar, memastikan kinerja optimal dalam operasional sehari-hari. Dengan demikian, sistem dapat mencatat, mengelola, dan mengakses data pemesanan serta transaksi dengan akurasi tinggi, memungkinkan pengelolaan jadwal dan pembayaran yang lebih efisien.



Gambar 3. 13 Basis Data Gama

Dalam sistem database yang digunakan untuk *booking* dan transaksi lapangan, terdapat beberapa tabel utama yang memiliki peran penting dalam pengelolaan data, seperti produk, varian, produkqty, tran, dan cash.

Tabel produk berisi informasi utama tentang layanan atau barang yang tersedia dalam sistem, termasuk nama produk, ukuran, dan kategori. Tabel varian menyimpan detail spesifik dari produk, seperti jam main dan harga yang

berbeda. Ini memungkinkan sistem mendukung variasi layanan atau barang yang ditawarkan. Tabel *produkty* mencatat ketersediaan dari setiap varian produk, memastikan bahwa data selalu terupdate saat terjadi transaksi. Tabel *tran* digunakan untuk mencatat seluruh transaksi yang terjadi, baik pemesanan maupun pembayaran, dengan informasi seperti tanggal transaksi, pelanggan, dan total pembayaran. Tabel *cash* berfungsi untuk menyimpan data terkait pembayaran yang dilakukan oleh pelanggan, memastikan bahwa setiap transaksi keuangan terdokumentasi dengan baik.

Struktur database ini memastikan bahwa sistem dapat berjalan dengan lancar, mengelola pemesanan, ketersediaan produk, dan transaksi keuangan dengan akurat.

c. Halaman *Login*

Halaman *login* adalah sebuah antarmuka yang disediakan oleh suatu sistem saat pertama kali pengguna akan mengakses sistem, yang memungkinkan pengguna untuk memasukkan data identifikasi mereka, seperti *username* dan *password*, agar bisa mengakses akun atau fitur tertentu. Fungsinya adalah sebagai langkah awal dalam proses verifikasi identitas, di mana pengguna harus membuktikan identitasnya sebelum diberikan akses ke informasi atau fitur tertentu, bertujuan untuk meningkatkan keamanan dengan memastikan bahwa hanya pengguna yang terdaftar yang dapat mengakses akun atau informasi yang diinginkan.



Gambar 3. 14 Login Gama

```

<?php $form = ActiveForm::begin(['id' => 'login-form', 'enableClientValidation' => false]); ?>

<?=
    $form
    ->field($model, 'username', $fieldOptions1)
    ->label(false)
    ->textInput(['placeholder' => $model->getAttributeLabel('username')])
>>
<?=
    $form
    ->field($model, 'password', $fieldOptions2)
    ->label(false)
    ->passwordInput(['placeholder' => $model->getAttributeLabel('password')])
>>

<div class="row" style="margin-top: 10px;">
    <div class="col-xs-8">
        <?= $form->field($model, 'rememberMe')->checkbox() >>
    </div>

    <!-- /.col -->
    <div class="col-xs-4">
        <?= Html::submitButton('Sign in', ['class' => 'btn ', 'name' => 'login-button']) >>
    </div>
    <!-- /.col -->
</div>

<?php ActiveForm::end(); >>

```

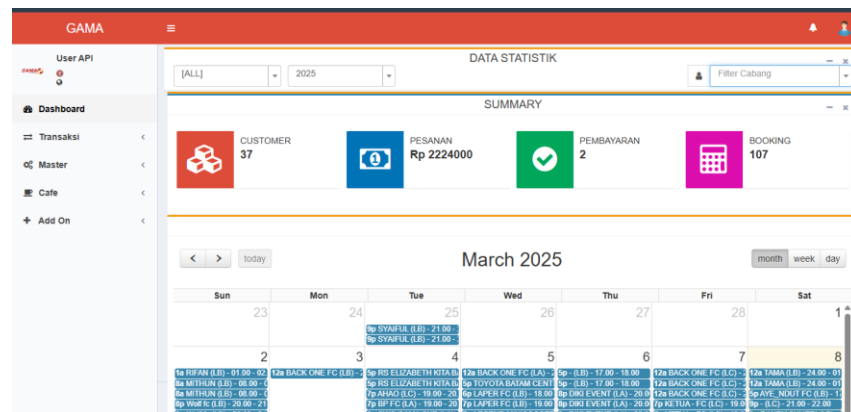
Gambar 3. 15 Code Login Gama

Kode ini merupakan tampilan *form login* yang dikembangkan menggunakan *Yii2 Framework* dengan fitur *ActiveForm* untuk menangani input dari pengguna. *Form* ini dibuat dengan *ActiveForm::begin()*, yang mendefinisikan atribut seperti *id* untuk form dan menonaktifkan validasi klien (*enableClientValidation* disetel ke *false*). Dalam *form* ini, terdapat dua input utama, yaitu *username* dan *password*. Masing-masing *input* dibuat menggunakan *\$form->field()* dengan *textInput()* untuk *username* dan *passwordInput()* untuk *password*. Kedua input ini menggunakan *placeholder* yang diambil dari label atribut model. Label *form* dinonaktifkan dengan *label(false)*, sehingga tampilan lebih bersih dan hanya menampilkan *placeholder* dalam *field* input.

Setelah input *username* dan *password*, terdapat *checkbox* "*rememberMe*", yang memungkinkan pengguna untuk tetap masuk tanpa harus *login* ulang setiap kali membuka aplikasi. *Checkbox* ini dibuat dengan *\$form->field(\$model, 'rememberMe')->checkbox()*. Bagian terakhir dari *form* ini adalah tombol *Sign in*, yang dibuat menggunakan *Html::submitButton()*. Tombol ini memiliki kelas CSS *btn*, yang dapat digunakan untuk *styling* tampilan, serta memiliki atribut *name* sebagai *login-button* agar mudah diidentifikasi dalam proses autentikasi. Terakhir, *ActiveForm::end()* digunakan untuk menutup *form*. Dengan struktur ini, *form login* dapat digunakan untuk menangani autentikasi pengguna dalam sistem berbasis *Yii2* secara dinamis dan fleksibel.

d. Menu *Dashboard*

Menu *Dashboard* merupakan tampilan utama yang muncul setelah pengguna berhasil *login* ke dalam sistem. Pada menu ini, pengguna dapat melihat ringkasan statistik bisnis secara *real-time*. Informasi utama yang ditampilkan meliputi jumlah pelanggan (*customer*), total pesanan dalam bentuk nominal rupiah, jumlah pembayaran yang telah dilakukan, serta total *booking* yang telah tercatat dalam sistem. Selain itu, terdapat kalender di bagian bawah yang menampilkan jadwal aktivitas atau pemesanan berdasarkan tanggal. Pengguna dapat melihat detail jadwal dalam tampilan bulanan, mingguan, atau harian sesuai kebutuhan. Kalender ini membantu dalam memantau dan mengelola jadwal operasional dengan lebih efisien.



Gambar 3. 16 Dashboard Gama

Di sisi kiri, terdapat *sidebar* menu yang berisi beberapa kategori utama, seperti *Dashboard*, *Transaksi*, *Master*, *Cafe*, dan *Add On*. Menu ini memungkinkan pengguna untuk dengan mudah mengakses berbagai fitur dalam sistem. Pada bagian atas halaman, terdapat panel filter untuk menampilkan data statistik berdasarkan tahun yang dipilih. Dengan tampilan yang terstruktur ini, pengguna dapat dengan mudah memahami kondisi bisnis dan mengelola operasional dengan lebih efektif.

```

<div class="col-md-12 no-padding">
  <div class="box box-warning col-md-12">
    <div class="box-header with-border">
      <div class="col-lg-2 col-md-1 col-sm-6 col-xs-6 no-padding">
        $data = [
          ['<!-- --> 'ALL'], 1 => 'JAN', 2 => 'FEB', 3 => 'MAR', 4 => 'APR', 5 => 'MAY', 6 => 'JUN', 7 => 'JUL', 8 => 'AUG', 9 => 'SEP', 10 => 'OCT', 11 => 'NOV', 12 => 'DEC'],
          ['name' => 'month',
            'id' => 'month',
            'data' => $data,
            'value' => $_GET['month'] == "" ? "" : $_GET['month'],
            'hideSearch' => true,
            'options' => ['placeholder' => 'Month'],
            'pluginOptions' => [
              'allowClear' => false
            ],
            'pluginEvents' => [
              'change' => "function() { loaddata(); }",
            ]
          ];
        </div>
        <div class="col-lg-2 col-md-1 col-sm-6 col-xs-6 no-padding">
          <div>
            <div>
              $data = [
                ['2024' => '2024', '2025' => '2025'],
                ['name' => 'year',
                  'id' => 'year',
                  'data' => $data,
                  'value' => $_GET['year'] == "" ? date("Y") : $_GET['year'],
                  'hideSearch' => true,
                  'options' => ['placeholder' => 'Year'],
                  'pluginOptions' => [
                    'allowClear' => false
                  ],
                  'pluginEvents' => [
                    'change' => "function() { loaddata(); }",
                  ]
                ];
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

Gambar 3. 17 Code Dashboard Gama

Pada halaman dashboard, terdapat fitur filter untuk memilih data berdasarkan bulan dan tahun. Filter ini menggunakan *Select2* untuk mempermudah pengguna dalam memilih opsi yang tersedia. *Dropdown* bulan menampilkan daftar Januari hingga Desember, sedangkan *dropdown* tahun menyediakan opsi 2024 dan 2025, dengan nilai *default* mengikuti tahun saat ini. Setiap kali pengguna mengubah pilihan, sistem akan memperbarui data secara otomatis dengan memanggil fungsi *loaddata()*. Filter ini bekerja dengan mengambil nilai dari URL menggunakan metode *\$\_GET*. Dengan adanya fitur ini, pengguna dapat menyaring data sesuai periode waktu yang diinginkan untuk analisis lebih lanjut.

#### e. Master Produk Lapangan

Actions	Kode	Nama	Kategori	Unit	Deskripsi	Stok
	LC	Lapangan C Kecil (50 x 30 meter)	Lapangan Kecil	Jam	Ukuran 50 x 30 meter	
	LA	Lapangan A Besar (50 x 40 meter)	Lapangan Besar	Jam	Ukuran 60 x 40 meter	
	LB	Lapangan B Kecil (50 x 30 meter)	Lapangan Kecil	Jam	Ukuran 50 x 30 meter	0

Gambar 3. 18 Index Produk Gama

Master Produk Lapangan digunakan untuk mengelola data lapangan yang tersedia dalam sistem. Halaman ini menampilkan daftar lapangan dengan informasi seperti kode, nama, kategori, unit, deskripsi, dan stok. Pengguna dapat menambahkan data lapangan baru menggunakan tombol "New", serta melakukan pencarian berdasarkan kategori atau kata kunci tertentu. Setiap lapangan memiliki kategori yang membedakan ukuran dan jenisnya. Fitur ini



mempermudah pengelolaan lapangan, memastikan data yang tersimpan selalu akurat dan *up to date*.

Halaman *index* produk ini menggunakan *Kartik GridView*, yaitu komponen tabel yang lebih interaktif dan kaya fitur di Yii2. Salah satu fitur yang digunakan adalah *ExpandRowColumn*, yang memungkinkan pengguna untuk melihat detail tambahan dari suatu produk dengan cara mengklik tombol ekspansi pada baris tabel.

```

},
[
  'class' => 'kartikGridExpandRowColumn',
  'value' => function ($model, $key, $index, $column) {
    return GridView::ROW_COLLAPSED;
  },
  'detailUrl' => Url::to(['produkt/indexdetail', 'produkttype' => $produkttype, 'parentid'=>$model['produktid']]),
  'headerOptions' => ['class' => 'kartik-sheet-style'],
  'expandOneOnly' => true
],
[
  'attribute' => 'produktno',
  'format' => 'raw',
  'label' => 'Kode',
  'contentOptions' => ['class' => 'text-left'],
  'headerOptions' => ['class' => 'text-center'],
],
[

```

Gambar 3. 19 Code Index Produk Gama

f. *Form* Produk

*Form* tambah lapangan ini dirancang untuk memasukkan data produk lapangan secara detail dan fleksibel. Pengguna dapat memilih kategori dan satuan unit terlebih dahulu, lalu mengisi informasi seperti kode, nama, dan deskripsi lapangan. Selain itu, terdapat fitur untuk menambahkan varian lapangan, seperti kode varian, kapasitas, jam operasional, harga, dan tag. Setiap varian bisa ditambah atau dihapus dengan tombol yang tersedia. Form ini dibuat agar proses input data lebih terstruktur dan efisien, dengan tampilan yang jelas dan mudah dipahami.

Tambah Lapangan

Kategori

Lapangan Kecil

Unit

Jam

Kode

FD-00001

Nama

Deskripsi

Data Varian

No.	Kode	Varian	Jam Mulai	Jam Selesai	Harga	Tag
1	C.51	90	12:00:00	12:00:00	99.990	Masukkan Tag
2	Masukkan Kod	Masukkan Data	12:00:00	12:00:00	0	Masukkan Tag

+ Tambah Baris

Submit

Close

*Gambar 3. 20 Form Produk Gama*

*Form* produk ini menggunakan *ActiveForm* di Yii2 untuk mempermudah pengelolaan input data. Setiap *field* dirancang agar terstruktur dan interaktif.

```

<td class="vcenter">
</php
echo $form->field($rowdetail, "{{$indexdetail}}jora")->textInput([
    // 'value' => $rowdetail->1,
    // 'placeholder' => 'No.',
    'class' => 'ord form-control',
    'readonly' => true,
    'style' => 'width:50px;text-align:center;',
])->label(false);
})
</td>

<td>
<?php
echo $form->field($rowdetail, "{{$indexdetail}}varianno")->textInput([
    'placeholder' => 'Masukkan Kode',
    'class' => 'form-control',
    // 'readonly' => true,
])->label(false); }
</td>

<td class="vcenter">
</php
$variantext_id = $rowdetail->variantext_id != null ? '(' . Varii::findOne($rowdetail->variantext_id)->varianno . ' ' :
echo $form->field($rowdetail, "{{$indexdetail}}variantext_id")->textInput([
    'placeholder' => 'Masukkan data',
    'class' => 'form-control',
])->label(false);
})
</td>

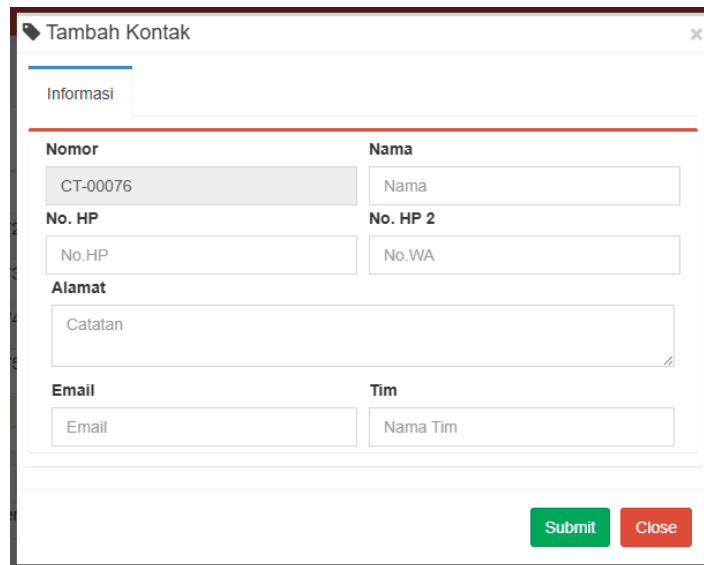
```

g. Menu *Contact*

<div>Tambah</div>		<div>Data Kontak</div> <div>Show 61-64 of 64 items</div>		<div>10</div> <div>Data</div>	
				<div> <div>Carli Nomor &gt; (ENTER)</div> <div>Carli</div> </div>	
Actions	Nomor	Nama	Tim	No. HP	No. HP 2
<div>☰</div>	CT-00072	JAGAD	JAGAD	0	0
<div>☰</div>	CT-00073	NAIN	NAIN	00	0
<div>☰</div>	CT-00074	EDGAR	EDGAR	00	0
<div>☰</div>	CT-00075	ZENIN FC	ZENIN FC	00	0
<div> <div>«</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>6</div> <div>7</div> <div>»</div> </div> <div>Show 61-64 of 64 items</div>					

*Gambar 3. 22 Index Contact Gama*

Halaman ini menampilkan daftar kontak menggunakan *GridView* di Yii2. Setiap kontak memiliki nomor unik, nama, tim, dan nomor HP. Data ditampilkan dalam bentuk tabel agar mudah dicari dan dikelola. Tombol tambah digunakan untuk menambahkan kontak baru. Setiap baris memiliki tombol aksi yang memungkinkan pengguna melakukan edit atau hapus data dengan cepat. Sistem pencarian di bagian kanan atas memungkinkan pengguna mencari kontak berdasarkan nomor. *Pagination* diterapkan untuk membagi data menjadi beberapa halaman agar tampilan tetap ringan dan mudah dinavigasi.



Gambar 3. 23 Form Produk Gama

*Form* ini digunakan untuk menambahkan data kontak baru. Pengguna bisa mengisi Nama, No. HP, No. HP 2 (WA), Alamat, Email, dan Tim. Bagian Nomor tidak bisa diedit (*readonly*) karena sistem akan mengisinya secara otomatis saat pengguna mengklik tombol Tambah

```
function setNo() {
$.ajax({
url: '<?php echo Url::to(['contact/getno'], true); ?>',
type: 'post',
data: {
csrf: '<?php echo Yii::$app->request->getCsrfToken(); ?>',
cabangid: $('#<?php echo Html::getInputId($model, 'cabangid'); ?>').val(),
cabangid: params.term,parentid:$('#penjualan-cabangid').val()
},
success: function(data) {
alert('123');
var json = $.parseJSON(data);
var receiptno = $('#receipt-receiptno').val();
}
});
}
```

Gambar 3. 24 Function Setno

Fungsi *setNo()* ini digunakan untuk mengambil nomor kontak baru secara otomatis saat pengguna menekan tombol tambah. Saat dipanggil, fungsi ini mengirimkan permintaan *AJAX* ke *contact/getno* menggunakan metode

*POST*. Sistem juga mengirimkan token *CSRF* untuk keamanan. Setelah server merespons, data yang diterima diubah menjadi *JSON*, meskipun dalam kode ini hasilnya belum langsung dimasukkan ke dalam inputan. Intinya, ini memastikan bahwa setiap kontak yang dibuat memiliki nomor unik secara otomatis tanpa perlu diinput manual.

```
public function actionGetno()
{
    $array = array(
        "no" => Contact::nextNo()
    );
    return \yii\helpers\Json::encode($array);
}
```

Gambar 3. 25 Function Getno

*ActionGetno()* ini ada di *controller* dan berfungsi untuk memberikan nomor kontak baru saat ada permintaan dari *AJAX* di *setNo()*. Ketika fungsi ini dipanggil, dia memanggil *Contact::nextNo()* yang adalah metode di model *Contact* untuk menghasilkan nomor berikutnya. Hasilnya kemudian dikembalikan dalam format *JSON*, yang nantinya akan diambil oleh *AJAX* untuk menampilkan nomor baru di form.

```
public function nextNo()
{
    try {
        $filter = "";
        $sql = "select cast(max(NULLIF(regexp_replace(contactno, '[0-9]', ''), ''))::numeric as bigint)+1 as kode from contact where status <> '10'";
        $nextKode = "1";
        $results = \Yii::$app->db->createCommand($sql)->queryAll();
        foreach ($results as $result) {
            $nextKode = $result['kode'];
        }

        if ($nextKode == "") {
            $nextKode = 1;
        }

        while (strlen($nextKode) < 5) {
            $nextKode = "0" . $nextKode;
        }

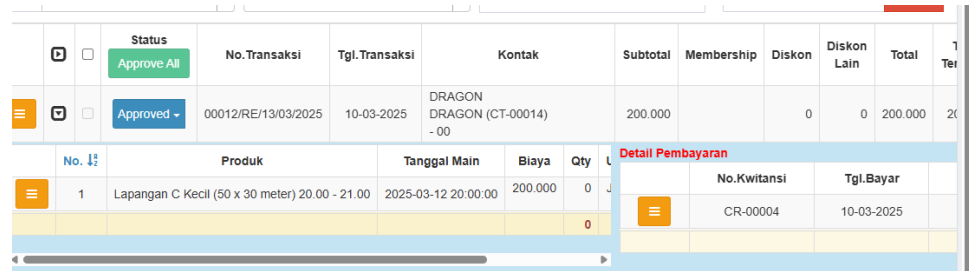
    } catch (Exception $e) {
        $nextKode = "00001";
    }
    $nextKode = "CT-" . $nextKode;
    return $nextKode;
}
```

Gambar 3. 26 Function Nextno

Fungsi *nextNo()* ini menentukan nomor kontak berikutnya secara otomatis. Pertama, *query* mengambil nomor kontak terbesar yang ada di database, dengan menghapus semua karakter selain angka dan mengubahnya menjadi tipe numerik. Setelah itu, nilai terbesar ditambah satu agar mendapatkan nomor kontak baru. Kalau tidak ada hasil, maka nomor *default* yang digunakan adalah 1. Setelah nomor diperoleh, sistem memastikan bahwa panjangnya minimal lima digit dengan menambahkan nol di depannya. Terakhir, sistem menambahkan *prefix* "CT-" sehingga format akhirnya menjadi "CT-00001", "CT-00002", dan seterusnya.

#### h. Menu *Booking*

Menu *Booking* digunakan untuk mengelola semua data pemesanan yang masuk ke dalam sistem. Di dalamnya, pengguna bisa melihat daftar *booking* yang sudah dibuat, menambahkan *booking* baru, mengedit detail pemesanan, serta membatalkan *booking* jika diperlukan. Menu ini juga menampilkan informasi penting seperti tanggal, waktu, status pembayaran, serta data pelanggan yang melakukan pemesanan. Dengan adanya menu ini, proses pencatatan dan pengelolaan *booking* menjadi lebih terstruktur.



No.	Status	No. Transaksi	Tgl. Transaksi	Kontak	Subtotal	Membership	Diskon	Diskon Lain	Total	1 Ter
	<span>Approve All</span>									
	<span>Approved</span>	00012/RE/13/03/2025	10-03-2025	DRAGON DRAGON (CT-00014) - 00	200.000		0	0	200.000	20

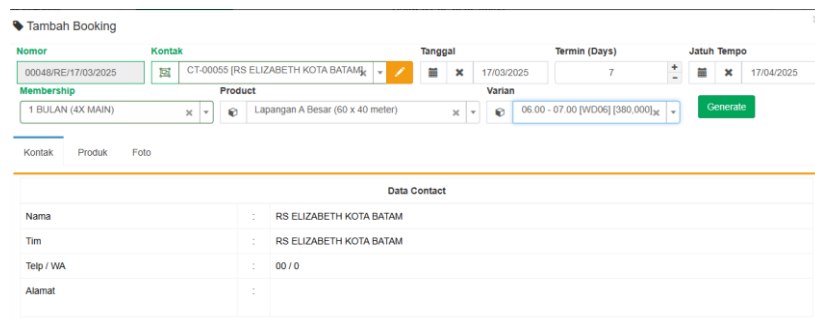
No.	Produk	Tanggal Main	Biaya	Qty
1	Lapangan C Kecil (50 x 30 meter) 20.00 - 21.00	2025-03-12 20:00:00	200.000	0

No. Kwitansi	Tgl. Bayar
CR-00004	10-03-2025

Gambar 3. 27 Index Booking

Tampilan ini menampilkan data transaksi *booking* yang mencakup nomor transaksi, tanggal transaksi, dan informasi kontak pelanggan, dan lainnya. Di dalamnya terdapat rincian produk yang dipesan, termasuk nama lapangan, ukuran, jadwal main, dan biaya. Selain itu, terdapat bagian detail pembayaran yang menampilkan nomor kwitansi dan tanggal pembayaran. Subtotal dan total transaksi juga ditampilkan untuk memastikan kesesuaian dengan pembayaran yang telah dilakukan.



**Tambah Booking**

Nomor: 00048/RE/17/03/2025 Kontak: CT-00055 [RS ELIZABETH KOTA BATAM] Tanggal: 17/03/2025 Termin (Days): 7 Jatuh Tempo: 17/04/2025

Membership: 1 BULAN (4X MAIN) Product: Lapangan A Besar (60 x 40 meter) Varian: 06.00 - 07.00 [WD06] [380.000] Generate

Kontak Produk Foto

Data Contact	
Nama	: RS ELIZABETH KOTA BATAM
Tim	: RS ELIZABETH KOTA BATAM
Telp / WA	: 00 / 0
Alamat	:

Gambar 3. 28 Form Booking

Tampilan ini adalah form tambah *booking* yang mencakup informasi nomor transaksi, kontak pelanggan, tanggal *booking*, termin pembayaran, dan jatuh tempo. Pada bagian kontak, jika pengguna memilih kontak dari *Select2*, maka detail informasi kontak akan otomatis ditampilkan di tab kontak, termasuk nama, tim, nomor telepon, dan alamat. Terdapat juga informasi *membership* yang bisa dipilih, dan saat memilih *membership*, sistem akan otomatis menyediakan

pilihan produk berupa lapangan yang bisa dipilih. Setelah produk dipilih, varian lapangan berdasarkan jam dan tarifnya juga dapat dipilih.

No.	Tanggal	Produk	Unit	Biaya	Disc	Total
1	17/03/2025	Lapangan A Besar (60 x 40)	Jam	380.000	19.000	361.000
2	24/03/2025	Lapangan A Besar (60 x 40)	Jam	380.000	19.000	361.000
3	31/03/2025	Lapangan A Besar (60 x 40)	Jam	380.000	19.000	361.000
4	07/04/2025	Lapangan A Besar (60 x 40)	Jam	380.000	19.000	361.000
SubTotal					1.520.000	
Disc					76.000	
Other Disc					0	
Total					1.444.000	

Gambar 3. 29 Form Booking 2

Tab produk ini menampilkan daftar lapangan yang telah dipilih berdasarkan varian yang tersedia. Setiap baris menunjukkan tanggal *booking*, produk (lapangan), unit waktu, biaya per sesi, serta diskon yang diberikan. Sistem secara otomatis mengisi daftar ini berdasarkan varian lapangan yang dipilih pada form sebelumnya. Jika pengguna mengklik "*Generate*" pada tab utama, sistem akan membuat entri *booking* sesuai dengan jadwal yang tersedia, misalnya setiap minggu pada hari yang dipilih. Pengguna juga dapat menambahkan baris baru secara manual dengan menekan tombol + *Add Line*, serta menghapus baris yang tidak diperlukan dengan tombol merah di setiap baris. Di bagian bawah terdapat perhitungan total biaya yang mencakup subtotal, total diskon, diskon tambahan (jika ada), dan total akhir yang harus dibayarkan.

```
// console.log(sumitemid);
// alert(s_umsbtotal);
$("#tran-subtotal").val(sumsbtotal);
$("#tran-disc").val(sumdisc);
$("#tempdetail").val(tempdetail);
$("#tempdate").val(tempdate);
var subtotal = $('#tran-subtotal').val();
subtotal = (subtotal == "") ? 0 : Number(subtotal.split(".").join(""));
var disc = 0;
sumtotal = sumsubtotal - sumdisc;
var otherdisc = $('#tran-otherdisc').val();
otherdisc = (otherdisc == "") ? 0 : Number(otherdisc.split(".").join(""));

sumtotal = sumtotal - otherdisc;
//alert(otherdisc);
// $('#pembelian-ketberat').val(sumberat);
$('#tran-total').val(sumtotal).trigger('change');
}
$('#tran-otherdisc').on('input', function() {
    setTotal();
});
```

Gambar 3. 30 Code Booking

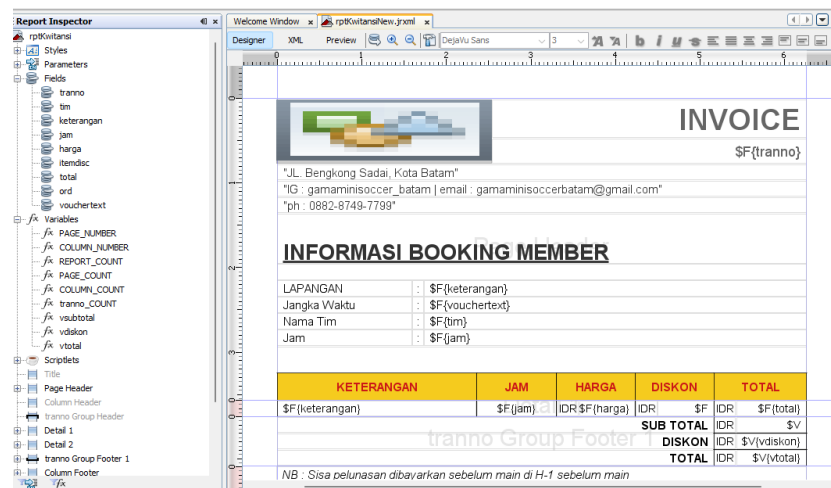
Kode *JavaScript* ini digunakan untuk menghitung total harga setelah memperhitungkan diskon. Nilai subtotal diambil dari elemen dengan ID *#tran-subtotal* dan dikonversi menjadi angka, jika kosong maka dianggap nol. Diskon awalnya diatur ke nol, lalu nilai diskon lain diambil dari elemen *#tran-otherdisc* dan dikonversi menjadi angka dengan menghapus titik pemisah ribuan. Setelah itu, subtotal dikurangi dengan diskon transaksi dan dikurangi lagi dengan diskon

tambahan untuk mendapatkan total akhir. Hasil perhitungan ini dimasukkan ke dalam elemen *#tran-total* dan dipicu *event change* agar data diperbarui secara otomatis. Selain itu, terdapat *event listener* pada elemen *#tran-otherdisc* yang akan memanggil fungsi *setTotal()* setiap kali ada perubahan input, sehingga total langsung menyesuaikan saat diskon diubah.



Gambar 3. 31 Invoice Booking Gama

*Invoice* ini berisi elemen utama seperti informasi *booking*, rincian pembayaran, serta total biaya. Bagian utama terdiri dari nomor *invoice*, detail pemesanan seperti tanggal dan jam *booking*, serta tabel yang mencakup harga, diskon, dan total pembayaran. Harga dalam *invoice* bisa berubah tergantung pada diskon atau kebijakan lainnya, sehingga total akhir dapat berbeda sesuai potongan yang diberikan. *Invoice* ini digunakan sebagai bukti transaksi bagi pelanggan yang melakukan *booking* lapangan, memastikan pembayaran sesuai dengan kesepakatan, serta memudahkan pengelola dalam pencatatan keuangan dan administrasi.



Gambar 3. 32 iReport Booking Gama

Template *invoice* ini dibuat menggunakan JasperReports, yang memanfaatkan *field* dan *variable* untuk menampilkan data transaksi secara dinamis dari *database*.

Di bagian kiri tampilan terdapat *Report Inspector*, yang berisi *Fields* dan *Variables*. *Fields* seperti  $\$F\{keterangan\}$ ,  $\$F\{jam\}$ ,  $\$F\{harga\}$ , dan  $\$F\{tim\}$  adalah data langsung dari *database*.  $\$F\{keterangan\}$  menampilkan nama lapangan yang dipesan,  $\$F\{jam\}$  menunjukkan jam bermain yang sudah dipesan,  $\$F\{harga\}$  menampilkan harga per sesi, dan  $\$F\{tim\}$  adalah nama tim yang melakukan *booking*.  $\$F\{transno\}$  digunakan untuk menampilkan nomor transaksi unik yang membedakan setiap *invoice*.

Sementara itu, *Variables* seperti  $\$V\{subtotal\}$ ,  $\$V\{vtotal\}$ , dan  $\$V\{vdiskon\}$  digunakan untuk perhitungan sebelum data ditampilkan.  $\$V\{subtotal\}$  menghitung jumlah harga sebelum diskon,  $\$V\{vdiskon\}$  menyimpan nilai diskon yang diberikan, dan  $\$V\{vtotal\}$  adalah hasil akhir setelah diskon diterapkan.

*Field* diperlukan untuk mengambil data dari database tanpa perubahan, sedangkan *variable* digunakan untuk memproses dan menghitung nilai yang membutuhkan operasi tambahan, seperti menghitung total setelah diskon. Dengan kombinasi *field* dan *variable* ini, *invoice* bisa dibuat otomatis sesuai dengan data *booking* tanpa harus memasukkan informasi secara manual untuk setiap transaksi.

#### i. Menu Jadwal

Menu Jadwal berfungsi untuk menampilkan dan mengelola jadwal yang sudah terdaftar dalam sistem. Di dalamnya, pengguna dapat melihat jadwal *booking* yang sudah dibuat, memastikan tidak ada bentrokan waktu, serta mengatur perubahan jika diperlukan. Menu ini membantu dalam perencanaan operasional dengan memberikan gambaran yang jelas mengenai waktu-waktu yang sudah terisi dan yang masih tersedia. Dengan fitur ini, pengguna dapat lebih mudah mengatur ketersediaan layanan dan memastikan semua jadwal berjalan sesuai rencana.



+ New Transaction

Data Transaksi

Show 1-20 of 20 items

20

▼

Data

16/03/202

Carli > (ENTER)

Carli

Waktu	Lapangan A	Lapangan B	Lapangan C
01.00 - 02.00	390.000	290.000	290.000
06.00 - 07.00	580.000	480.000	480.000
07.00 - 08.00	580.000	480.000	480.000
08.00 - 09.00	580.000	BOOKED TERATAI FC	480.000
09.00 - 10.00	580.000	BOOKED TERATAI FC	480.000
10.00 - 11.00	480.000	380.000	380.000
11.00 - 12.00	480.000	380.000	380.000
12.00 - 13.00	480.000	380.000	380.000

Gambar 3. 33 Jadwal Gama

Tampilan ini adalah tabel untuk sistem pemesanan lapangan yang menampilkan daftar waktu dan harga penyewaan untuk setiap lapangan. Beberapa waktu sudah terisi dengan status "BOOKED TERATAI FC", yang berarti lapangan tersebut sudah dipesan oleh klub Teratai FC. Saat pengguna mengklik harga di tabel, mereka akan diarahkan ke *form booking*, di mana data seperti nama lapangan, jam pemesanan, tanggal yang dipilih, dan harga akan otomatis terisi sesuai dengan yang dipilih. Fitur ini memudahkan pengguna agar tidak perlu mengisi detail secara manual.

```

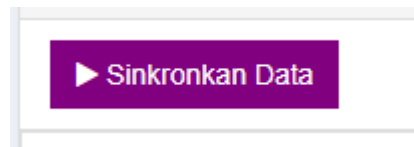
{
  'attribute' => 'lapangan_a',
  'format' => 'raw',
  'label' => 'Lapangan A',
  'hAlign' => 'center',
  'vAlign' => 'middle',
  'headerOptions' => [{ 'class' => 'text-center' },
  'contentOptions' => function ($model) {
    return [
      'class' => strpos($model['lapangan_a'], 'BOOKED') !== false ? 'bg-gray' : '',
      'style' => 'text-align: center; vertical-align: middle;',
    ];
  },
  'value' => function ($model) {
    // Split value into number and status
    $data = explode(' ', $model['lapangan_a']);
    // if (isset($data[2]) && ($data[2] === 'BOOKED' || $data[3] === 'BOOKED')) {
    if (isset($data[0]) && $data[0] === 'BOOKED') {
      // Jika BOOKED, hanya tampilkan teks
      $number = isset($data[0]) ? $data[0] . ' ' . $data[1] : '';
      $status = isset($data[2]) ? strtoupper($data[2]) : '';
      return $number . '<br>' . $status;
    } else {
      // Jika tidak BOOKED, buat hyperlink
      $number = isset($data[0]) ? number_format((int)$data[0], 0, ',', '.') : '';
      $url = Url::to(['tran/create', 'trantype' => 'trantype.re', 'voucher' => '5a5a8bbe-b3fb-49fd-be5d-38548391d080', 'produkid' => $data[1], 'varianid' => $data[2], 'trandate' => isset($_GET['filterdate']) ? urlencode($_GET['filterdate']) : date("d/m/Y")]);
      return Html::a($number, $url, ['target' => '_blank', 'class' => 'tambah']) . '<br>';
    }
  },
}

```

Gambar 3. 34 Code Jadwal

Saat sistem membaca data, pertama-tama ia akan memeriksa apakah dalam kolom harga ada kata "BOOKED". Kalau ada, maka tampilan akan diberi *background* abu-abu sebagai tanda kalau slot sudah dipesan. Kemudian, sistem akan memisahkan data yang ada dalam satu kolom, misalnya harga dan statusnya, supaya bisa diolah lebih lanjut. Kalau statusnya "BOOKED", maka yang muncul hanya teks biasa tanpa *link*. Tapi kalau slotnya masih tersedia, harga akan berubah menjadi tombol yang bisa diklik. Saat diklik, sistem akan membawa pengguna ke halaman booking dengan informasi seperti jenis

transaksi, kode produk, varian, dan tanggal yang otomatis terisi sesuai dengan data yang dipilih.




Gambar 3. 35 Tombol Sinkron

Tombol "Sinkronkan Data" berfungsi untuk mengambil data dari API dan menyimpannya ke dalam database sistem. Saat tombol ini ditekan, sistem akan melakukan *request* ke API untuk mendapatkan data terbaru, kemudian data tersebut diproses dan dimasukkan ke dalam tabel yang sesuai di *database*. Ini memastikan bahwa informasi dalam sistem selalu diperbarui sesuai dengan data dari sumber eksternal.

```
document.getElementById("apilist-button").addEventListener("click", function() {  
  let filterDate = document.querySelector("input[name='filterdate']").value; //inikan ambil dr filterdetnyaaa iyaaa  
  let formattedDate = filterDate.split("/").reverse().join("-"); // ini ubah dmy ke ymd aja  
  
  fetch(`https://svr5.gobra.id/index.php?r=tran/apilist&date=${formattedDate}`)  
    .then(response => response.json())  
    .then(data => console.log("Response dari API:", data))  
    .catch(error => console.error("Error:", error));  
});
```


Gambar 3. 36 Code Tombol Sinkron

Kode ini menangani *event* ketika tombol dengan ID "*apilist-button*" atau sinkronkan data diklik. Pertama, kode mengambil nilai tanggal dari input yang bernama "*filterdate*", lalu mengonversinya dari format DD/MM/YYYY menjadi YYYY-MM-DD agar sesuai dengan format yang dibutuhkan API. Setelah itu, kode melakukan *fetch request* ke *endpoint* API dengan parameter tanggal yang sudah diformat. Jika request berhasil, data yang diterima dari API akan ditampilkan di *console log*. Jika terjadi *error*, pesan *error* akan dicetak di *console*.



AYO  
INDONESIA

API Documentation



AYO  
INDONESIA

booking\_source

No

There are 2 types of booking sources: "orders" and "reservation", please use one of them if you want to filter by the booking source.

Request Method:

GET

API Endpoint:

API-BASE-URL/list-bookings/venue\_code

Example Request:

```
curl --location --request GET
'https://api.ayo.co.id/api/v2/third-party/list-bookings/BptdYPUeMr' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json' \
--data '{
  "signature":
    "d78f3ed192026799060b68b4531d327110a44b3829382a50e21ee8acc749fba1201cd42c4f
a8fc867b8008cf2deb9092c319dba0dcab00bc1dc0b3ea2c05666",
  "token": "3qUFJksFEG8U5X15aXF21Y2WT8jYRDYCFerLmHjWILrawlVpeyjm00V6tQr1"
}'
```

Gambar 3. 37 Modul API

Dokumentasi ini menunjukkan cara mengambil data daftar pemesanan dari API Ayo Indonesia menggunakan metode *GET*. Endpoint API yang digunakan memerlukan kode *venue* untuk mendapatkan daftar pemesanan terkait. Contoh permintaan menggunakan *curl* menunjukkan bahwa API ini memerlukan *header JSON* dan *signature* + token sebagai autentikasi.

```
ksort($params);
$data = http_build_query($params);
$signature = hash_hmac('sha512', $data, $privateKey);
$postData = json_encode([
  "signature" => $signature,
  "token" => $apiToken,
  // "date" => "2025-03-22"
  "date" => $date,
  "limit" => $limit
]);

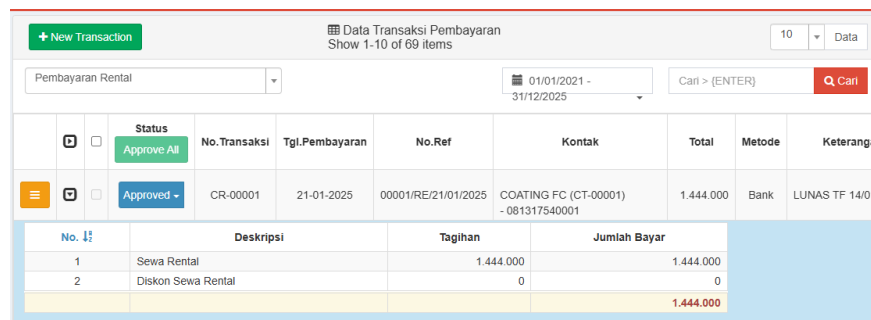
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $apiUrl);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, [
  'Accept: application/json',
  'Content-Type: application/json',
]);
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "GET");
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);
```

Gambar 3. 38 API Code

Kode ini digunakan untuk mengirim permintaan API menggunakan *cURL* di PHP. Data parameter diurutkan dan dikonversi menjadi *query string*, lalu dibuat *signature* menggunakan *hash HMAC SHA-512* dengan *private key* untuk keamanan. Data yang dikirim diencode dalam format *JSON*, termasuk *signature*, token API, tanggal, dan limit data yang diminta. Kemudian, *cURL* diinisialisasi dengan metode *GET*, URL API, *header JSON*, dan *payload* data. Hasilnya akan dikembalikan sebagai *respons JSON* setelah eksekusi *cURL*.

Setelah data diperoleh melalui *cURL* dalam bentuk *respons JSON*, data tersebut diproses dan dimasukkan ke dalam *database*. Prosesnya mencakup pengecekan apakah transaksi sudah ada, pembaruan jika diperlukan, atau pembuatan entri baru jika transaksi belum ada. Selain itu, informasi pelanggan juga dicek dan disimpan jika belum ada dalam *database*. Data yang disimpan mencakup informasi pemesan, tanggal, harga, status pembayaran, dan produk yang dipesan. Ini memastikan bahwa semua transaksi tercatat dan dapat digunakan dalam sistem untuk keperluan pelacakan dan analisis lebih lanjut.

#### j. Menu Pembayaran



No.	Status	No. Transaksi	Tgl. Pembayaran	No. Ref	Kontak	Total	Metode	Keterangan
1	Approved	CR-00001	21-01-2025	00001/RE/21/01/2025	COATING FC (CT-00001) - 081317540001	1.444.000	Bank	LUNAS TF 14/01/2025

No.	Deskripsi	Tagihan	Jumlah Bayar
1	Sewa Rental	1.444.000	1.444.000
2	Diskon Sewa Rental	0	0
			1.444.000

Gambar 3. 39 Index Cash Gama

Halaman ini menampilkan daftar transaksi pembayaran untuk penyewaan, dengan berbagai informasi penting seperti nomor transaksi, tanggal pembayaran, no ref, kontak pelanggan, total pembayaran, dan metode pembayaran.

Dalam contoh yang terlihat, transaksi dengan nomor CR-00001 telah disetujui dan memiliki total pembayaran sebesar Rp1.444.000 yang dilakukan melalui transfer bank. Statusnya juga sudah lunas, yang menandakan bahwa tidak ada kewajiban pembayaran yang tersisa. Di bagian bawah, terdapat rincian transaksi yang menunjukkan biaya sewa rental sebesar Rp1.444.000 tanpa adanya diskon, sehingga jumlah yang harus dibayar tetap sama. Tampilan ini mempermudah pemantauan pembayaran dan statusnya, memastikan bahwa semua transaksi tercatat dengan jelas.

No.	Detail	Tagihan	Jumlah Bayar
1	Sewa Rental	380.000	300.000 (+)
2	Diskon Sewa Rental	0	0 (-)
SubTotal			300.000 (+)
Discount			0 (-)
Total			300.000

Gambar 3. 40 Form Cash Gama

Di halaman ini, jumlah pembayaran bisa diinput secara manual sesuai dengan kebutuhan, memungkinkan pembayaran dilakukan secara bertahap atau dicicil. Dalam contoh ini, dari total tagihan Rp380.000, pengguna hanya membayar Rp300.000 terlebih dahulu, sehingga masih ada sisa pembayaran sebesar Rp80.000. Sistem akan tetap menampilkan referensi transaksi selama masih ada sisa pembayaran. Jika pembayaran sudah lunas, maka nomor referensi transaksi tidak akan muncul lagi di formulir pembayaran tunai, sehingga mencegah terjadinya pembayaran ganda atau kelebihan pembayaran.

```
});
$("#cash-subtotal").val(sumsubtotal);
$("#cash-disc").val(subdiscotal);

var subtotal = $('#cash-subtotal').val();
subtotal = (subtotal == "") ? 0 : Number(subtotal.split(".").join(""));

var disc = $('#cash-disc').val();
disc = (disc == "") ? 0 : Number(disc.split(".").join(""));

sumtotal = sumsubtotal - disc;
// $("#pembelian-ketberat").val(sumberat);
$("#cash-total").val(sumtotal).trigger('change');
```

Gambar 3. 41 Code Cash Gama

Kode ini mengambil nilai dari input dengan ID *cash-subtotal* dan *cash-disc*, lalu mengonversinya ke angka. Jika input kosong, nilainya akan menjadi 0. Setelah itu, dilakukan perhitungan dengan mengurangi *subtotal* dengan diskon. Hasil akhirnya dimasukkan ke dalam *input cash-total* dan memicu *event change*, yang bisa digunakan untuk memperbarui elemen lain jika ada.

## k. Menu Laporan

The screenshot shows a web application interface. On the left is a sidebar menu with two main categories: 'Pembayaran' (Payment) and 'Penjualan' (Sales). Under 'Pembayaran', there are sub-items: 'Laporan Pembayaran Rental', 'Laporan Pembayaran Penjualan', 'Laporan Pembayaran Perlengkapan', and 'Laporan Piutang'. Under 'Penjualan', there are sub-items: 'Laporan Piutang Rental', 'Laporan Piutang Penjualan', 'Laporan Piutang Perlengkapan', and 'Laporan Piutang'. The main area displays a report titled 'LAPORAN PEMBAYARAN RENTAL'. It includes a search bar, a date range selector (01/03/2025 - 31/03/2025), and a table with columns: No, No Transaksi, Tgl Pembayaran, Kontak, Ref No, Note, and Total Terbayar. The table contains 12 rows of data.

No	No Transaksi	Tgl Pembayaran	Kontak	Ref No	Note	Total Terbayar
1	CR-00016	03/03/2025	CT-00017, KUDA LUMPING (KUDA LUMPING)	00017REF1415139	Cash diterima dari API	280.000
2	CR-00024	03/03/2025	CT-00021, INDOHARET (INDOHARET)	00021REF1415139	Cash diterima dari API	0
3	CR-00023	03/03/2025	CT-00021, INDOHARET (INDOHARET)	00021REF1415139	Cash diterima dari API	0
4	CR-00022	03/03/2025	CT-00021, INDOHARET (INDOHARET)	00021REF1415139	Cash diterima dari API	0
5	CR-00038	15/03/2025	CT-00029, BACK ONE FC (BACK ONE FC)	00029REF1415139	Cash diterima dari API	351.000
6	CR-00015	13/03/2025	CT-00029, BACK ONE FC (BACK ONE FC)	00029REF1415139	Cash diterima dari API	261.000
7	CR-00024	17/03/2025	CT-00035, HANO (HANO)	00035REF1415139	Cash diterima dari API	580.000
8	CR-00004	10/03/2025	CT-00014, DRAGON (DRAGON)	00014REF1415139	Cash diterima dari API	200.000
9	CR-00049	17/03/2025	CT-00014, DRAGON (DRAGON)	00014REF1415139	Cash diterima dari API	200.000
10	CR-00056	15/03/2025	CT-00015, ASBP (ASBP)	00015REF1415139	Cash diterima dari API	580.000
11	CR-00005	05/03/2025	CT-00036, METRO FC (METRO FC)	00036REF1415139	Cash diterima dari API	580.000
12	CR-00047	07/03/2025	CT-00040, (-)	00040REF1415139	Cash diterima dari API	380.000

Gambar 3. 42 Index Report Gama

Tampilan ini adalah sistem laporan pembayaran rental yang menampilkan data transaksi dalam bentuk tabel yang dapat difilter dan diekspor. Di sebelah kiri terdapat menu navigasi dengan dua kategori utama, yaitu "Pembayaran" dan "Penjualan". Di dalam laporan pembayaran rental, sistem menampilkan data transaksi yang berisi nomor transaksi, tanggal pembayaran, kontak yang melakukan pembayaran, nomor referensi, catatan terkait transaksi, dan total yang telah dibayarkan. Data ini berasal dari sistem pembayaran yang mencatat setiap transaksi yang masuk, baik secara manual maupun otomatis melalui API.

Tampilan ini juga memiliki fitur filter yang memungkinkan pengguna memilih kontak tertentu serta rentang tanggal transaksi yang ingin ditampilkan. Selain itu, ada tombol untuk menyegarkan data dan mengunduh laporan dalam format tertentu.

```

else if ($jenis == 2) {
    //var_dump($scashtype);exit;
    $judul = "Laporan Pembayaran Rental";
    $subjudul = "";
    //var_dump($statuspaid);exit;
    $periode = \Yii::$app->function->convertMonthIntoSingkat($month) . " - " . $year;

    $this->getQueryPembayaran($jenis, $sareaid, $stipe, $subareaid, $smoid, $scashtype, $month, $year, $filterdate, $statuspaid, $contacts, $invoicepaid, $stipe, $sql, $);
    //var_dump($statuspaid);exit;
    //var_dump($contacts); exit;
    $report = $completemanager->compileReport(realpath(\Yii::getAlias('@runtime') . "/Report/rptlistPembayaran-fxml"));
    // var_dump($contacts); exit;
} else if ($jenis == 3) {

```

Gambar 3. 43 Code Report Gama

Kode ini adalah bagian dari sistem yang menangani laporan pembayaran rental. Ketika variabel \$jenis bernilai 2, sistem akan menampilkan Laporan Pembayaran Rental. Variabel \$judul berisi teks yang akan ditampilkan sebagai judul laporan.

The screenshot shows a web application interface. On the left is a sidebar menu with two main categories: 'Fields' and 'Variables'. Under 'Fields', there are sub-items: 'cashno', 'cashdate', 'contactname', 'refno', 'note', and 'total'. Under 'Variables', there are sub-items: 'PAGE\_NUMBER', 'COLUMN\_NUMBER', 'REPORT\_COUNT', 'PAGE\_COUNT', 'COLUMN\_COUNT', 'count\_row', and 'vtotal'. The main area displays a report titled 'SP(pjudul).toUpperCase()'. It includes a search bar, a date range selector, and a table with columns: No, No Transaksi, Tgl, Kontak, Ref No, Note, and Total Terbayar. The table contains 12 rows of data.

No	No Transaksi	Tgl	Kontak	Ref No	Note	Total Terbayar
SV	SP(cashno)	SP(cashdate)	SP(contactname)	SP(refno)	SP(note)	SP(total)
TOTAL						SV{total}

Gambar 3. 44 iReport Report Gama

Setelah data pembayaran diperoleh, sistem akan menggunakan `$compileManager->compileReport()` untuk mengolah template laporan dari file `rptlistPembayaran.jrxml`, yang merupakan file *template* JasperReports untuk menghasilkan laporan dalam format yang dapat dicetak atau diekspor.

```

class ReportController extends Controller
public function getQueryPembayaran($jenis, $areaid, $stipe, $subareaid, $cmoid, $cashtype, $month, $year, $filterdate)
{
    $temp = "
    select A.cashno, A.cashdate, C.contactno || ' ' || C.contactname || ' (' || C.ig || ')' as contactname,
    B.tranno as refno, B.statuspaid, A.note, coalesce(A.total, 0) as total
    from cash A
    left join tran B ON B.tranid=A.refid
    left join contact C ON C.contactid=B.contactid
    $filter
    ";

    $sql = "select * from(
        $temp
    ) as result
    $orderby
    ";
}

```

Gambar 3. 45 Query Report Gama

Kode ini digunakan untuk mengambil data laporan pembayaran dari *database*. *Query* mengambil informasi pembayaran dari tabel *cash*, menghubungkannya dengan transaksi di tabel *tran* dan data kontak di tabel *contact*. Hasilnya mencakup nomor kas, tanggal pembayaran, nama kontak, nomor referensi transaksi, status pembayaran, catatan, dan total pembayaran. Data ini kemudian difilter sesuai parameter yang diberikan sebelum ditampilkan dalam laporan.

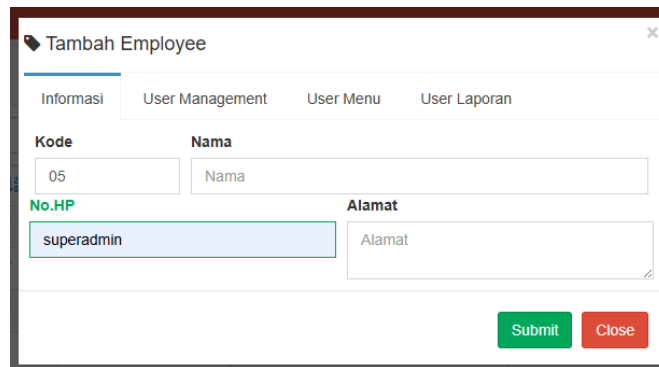
## 1. Menu User

<div> <div>+ Baru</div> <div>Data Employee</div> <div>Show 1-4 of 4 items</div> </div>						
<div> <div>[ ALL ]</div> <div>Pilih Cabang</div> <div>Caril Nomor &gt; {ENTER}</div> </div>						
Actions	ID	Nama	Posisi	Cabang	Username	No.HP
	04	User API			User API	
	03	kasir2			kasir2	
	02	Widia			Widia	
	01	admin_01			admin01	08165955xxxx

Gambar 3. 46 Index User Gama

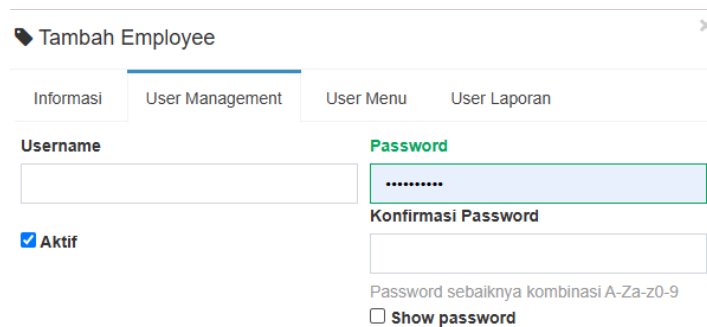
Menu User digunakan untuk mengelola informasi pengguna dalam sistem. Di dalam menu ini, admin atau pengguna dengan akses tertentu bisa melihat daftar pengguna yang terdaftar, menambahkan pengguna baru, mengedit informasi pengguna yang sudah ada, serta menghapus pengguna jika diperlukan. Menu ini juga bisa mencakup pengaturan hak akses, memungkinkan admin menentukan peran dan izin yang dimiliki oleh masing-masing pengguna dalam sistem. Selain itu, menu ini berfungsi untuk memastikan data pengguna tetap

akurat dan terorganisir, sehingga sistem dapat berjalan dengan lebih efektif dan aman.

The image shows a web application window titled "Tambah Employee". It has four tabs: "Informasi", "User Management", "User Menu", and "User Laporan". The "Informasi" tab is selected. The form contains four input fields: "Kode" with the value "05", "Nama" with the value "Nama", "No.HP" with the value "superadmin", and "Alamat" with the value "Alamat". At the bottom right, there are two buttons: "Submit" (green) and "Close" (red).

Gambar 3. 47 Form User Gama

Tab Informasi dalam *form Tambah User* memiliki beberapa *field* yang bisa diisi, yaitu Kode, Nama, No. HP, dan Alamat. Kode berisi identifikasi unik untuk setiap *user*. Nama digunakan untuk memasukkan nama *user*. No. HP diisi dengan nomor handphone *user*. Alamat digunakan untuk menuliskan alamat lengkap *user*.

The image shows the same "Tambah Employee" window, but with the "User Management" tab selected. The "Informasi" tab is now disabled. The form contains three input fields: "Username", "Password" (masked with dots), and "Konfirmasi Password". There is a checkbox labeled "Aktif" which is checked. Below the password fields, there is a text label "Password sebaiknya kombinasi A-Za-z0-9" and a checkbox labeled "Show password" which is unchecked.

Gambar 3. 48 Form User Gama 2

Tab *User Management* dalam *form Tambah User* memiliki beberapa *field* yang bisa diisi, yaitu *Username*, *Password*, dan *Konfirmasi Password*. *Username* digunakan sebagai identitas *login user*. *Password* berisi kata sandi yang harus dikombinasikan dengan huruf dan angka untuk keamanan. *Konfirmasi Password* digunakan untuk memastikan bahwa *password* yang dimasukkan benar. Ada juga opsi *Aktif*, yang menandakan apakah *user* tersebut diaktifkan atau tidak, serta *checkbox Show password* untuk menampilkan atau menyembunyikan kata sandi yang diketik.



Menu	Lihat	Tambah	Ubah	Hapus	Cetak	Persetujuan
Makanan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Cabang	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Kontak	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Peralatan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Lapangan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Kategori Peralatan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Kategori Makanan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Jadwal	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Gambar 3. 49 Form User Gama 3

Tab *User Menu* dalam *form Tambah User* berisi daftar menu yang bisa diakses oleh *user*. Setiap menu memiliki beberapa hak akses yang bisa diatur, yaitu Lihat, Tambah, Ubah, Hapus, Cetak, dan Persetujuan. Admin dapat mencentang kotak sesuai dengan izin yang diberikan kepada *user* untuk setiap menu. Jika kotak tertentu dicentang, *user* akan mendapatkan izin untuk melakukan tindakan sesuai dengan kategori tersebut.

Nama	Lihat	Print
Laporan Laporan Piutang Rental	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Pembayaran Rental	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Piutang Penjualan	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Piutang Perlengkapan	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Pembayaran Penjualan	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Pembayaran Perlengkapan	<input type="checkbox"/>	<input type="checkbox"/>
Laporan Laporan Piutang	<input type="checkbox"/>	<input type="checkbox"/>

Gambar 3. 50 Form User Gama 4

Tab *User Laporan* dalam *form Tambah User* berisi daftar laporan yang bisa diakses oleh *user*. Setiap laporan memiliki dua hak akses yang bisa diatur, yaitu Lihat dan Print. Jika kotak Lihat dicentang, *user* dapat melihat laporan tersebut, sedangkan jika kotak *Print* dicentang, *user* dapat mencetak laporan.

1	select * from users		
Data Output		Explain	Messages
	userid [PK] uuid	username character varyin	password character vary
1	6bad87...	admin	\$1\$75x7qP...
2	ce297bf...	admin01	\$1\$AmdQB...
3	6bad87...	superadmin	\$1\$z2gTlid...
4	25e5dd...	kasir2	\$1\$wImmi...
5	6ddc37...	Widia	\$1\$bEj2M...
6	ff317d9...	User API	\$1\$1FvFOT...

Gambar 3. 51 Table User Gama

Setelah *user* ditambahkan melalui *form*, data *user* akan masuk ke dalam tabel *users* di *database*. Tabel ini menyimpan informasi seperti *userid* (sebagai *primary key*), *username*, dan *password* yang telah di-*hash* untuk keamanan. Data yang ada dalam tabel inilah yang akan digunakan untuk proses *login*. Saat *user* mencoba masuk, sistem akan mencocokkan *username* dan *password* yang diinput dengan data di tabel ini. Jika sesuai, maka *user* dapat mengakses sistem sesuai dengan hak akses yang telah diberikan.

### 3. Penerapan Teknologi dalam Sistem *Website* Penjualan-Pembelian Kepiting

Sistem *website* ini dirancang sebagai alat manajemen internal untuk mengelola proses penjualan dan pembelian kepiting. Tidak seperti toko online yang memungkinkan pelanggan langsung melakukan transaksi, sistem ini lebih berfokus pada pengelolaan data dan operasional bisnis.

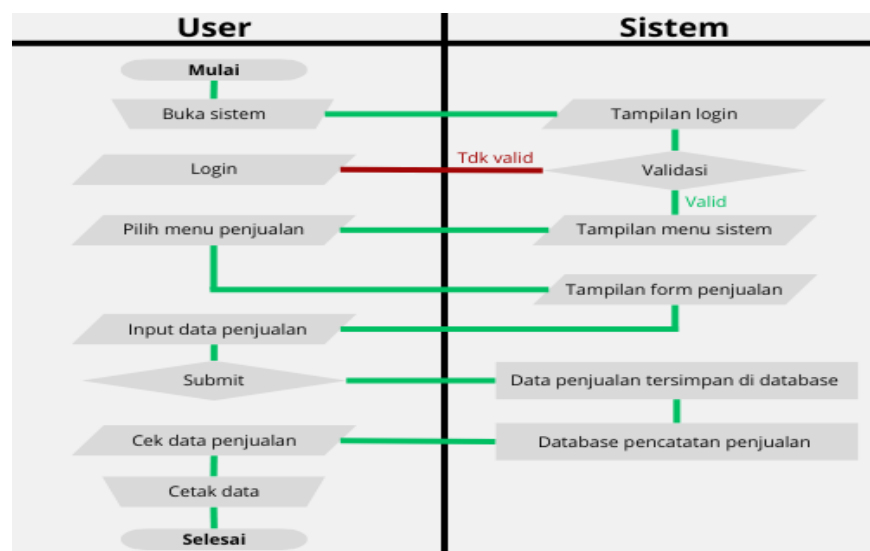
Dalam sistem ini, terdapat berbagai fitur yang memungkinkan pengguna untuk melihat statistik penjualan, jumlah pesanan, pembayaran yang telah diterima, serta jumlah pelanggan. Semua informasi tersebut ditampilkan dalam bentuk *dashboard* untuk memudahkan pemantauan aktivitas bisnis secara keseluruhan.

Selain itu, terdapat menu yang digunakan untuk mengelola berbagai aspek bisnis, seperti penjualan, pembelian, produk, serta pengelolaan data pengguna dan lokasi. Fitur ini membantu dalam pencatatan dan pemantauan transaksi, sehingga mempermudah pengambilan keputusan dalam operasional bisnis. Dengan adanya sistem ini, proses administrasi menjadi lebih terstruktur dan efisien, tanpa perlu menggunakan pencatatan manual.

a. Proses Perancangan

Proses perancangan sistem informasi pencatatan penjualan mencakup penyusunan alur kerja sistem, pembuatan diagram UML, pengembangan ERD, *flowmap*, normalisasi data, dan pembuatan database serta antarmuka pengguna. Melibatkan gambaran umum tentang operasi sistem, struktur dan interaksi antar komponen, model hubungan entitas dalam *database*, optimasi struktur basis data, implementasi struktur *database*, dan pembuatan antarmuka pengguna yang mudah digunakan.

*Flowmap* adalah sebuah visualisasi yang mengilustrasikan aktivitas yang sedang berlangsung, menyajikan gambaran mengenai aliran dokumen yang terjadi dalam proses tersebut, serta menguraikan bagaimana data dan informasi berhubungan dengan berbagai bagian dalam aktivitas yang sedang dilakukan.

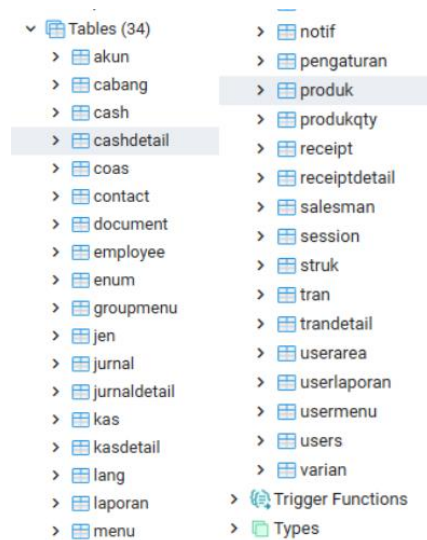


Gambar 3. 52 Flowmap

*Flowmap* ini menggambarkan alur proses penggunaan sistem dari sisi pengguna dan sistem. Pengguna memulai dengan membuka sistem, kemudian *login*. Jika *login* tidak valid, proses akan terhenti, tetapi jika valid, sistem akan menampilkan menu utama. Pengguna kemudian memilih menu penjualan dan sistem menampilkan *form input* data penjualan. Setelah pengguna memasukkan data dan mengirimkannya, sistem menyimpan data ke dalam database. Pengguna dapat mengecek data yang telah dimasukkan, dan sistem menampilkan informasi dari database. Jika diperlukan, pengguna dapat mencetak data sebelum menyelesaikan proses.

## b. Pembuatan *Database*

Pembuatan *database* untuk sistem penjualan dan pembelian kepiting merupakan langkah penting dalam membangun sistem yang efisien dan terstruktur. Dengan menggunakan PostgreSQL sebagai sistem manajemen basis data, sistem ini dapat menyimpan dan mengelola data dengan aman dan andal. PostgreSQL dipilih karena kemampuannya dalam menangani transaksi dalam jumlah besar serta fitur-fiturnya yang mendukung pengelolaan data secara fleksibel dan akurat.



Gambar 3. 53 Basis Data Crab

Melalui pembuatan *database* ini, sistem bertujuan untuk memastikan bahwa seluruh proses penjualan dan pembelian kepiting dapat berjalan dengan lancar, mulai dari pencatatan stok hingga transaksi pembayaran. Struktur *database* dirancang agar dapat menangani volume data yang besar dan memastikan performa tetap optimal dalam operasional sehari-hari. Dengan demikian, sistem dapat mencatat, mengelola, dan mengakses data penjualan serta pembelian dengan akurasi tinggi untuk memastikan transaksi berjalan dengan efisien.

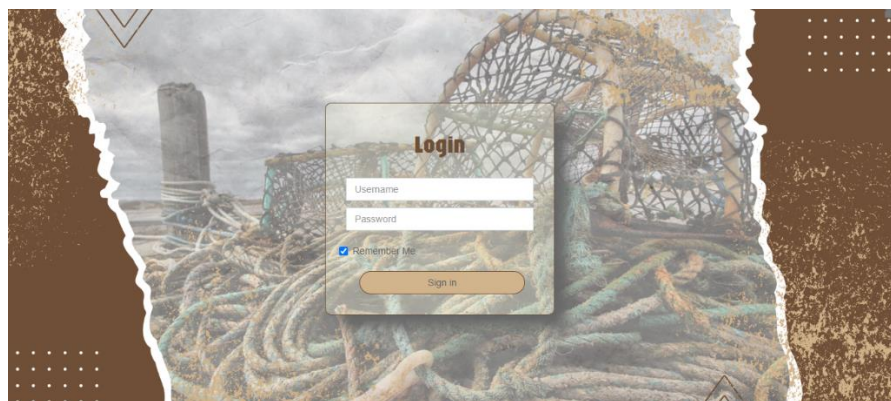
Dalam sistem database ini, terdapat beberapa tabel utama yang memiliki peran penting dalam pengelolaan data, seperti produk, varian, tran, dan cash. Tabel produk berisi informasi utama mengenai kepiting yang dijual, termasuk jenis, ukuran, dan harga dasar. Tabel varian menyimpan kode peti dari kepiting, seperti perbedaan ukuran yang ditawarkan kepada pelanggan. Tabel tran digunakan untuk mencatat seluruh transaksi yang terjadi dalam sistem, baik dari sisi penjualan kepada pelanggan maupun pembelian dari pemasok, dengan informasi seperti tanggal transaksi, jumlah, dan total pembayaran. Tabel cash

berfungsi untuk menyimpan data terkait pembayaran yang dilakukan, baik yang diterima dari pelanggan maupun yang dibayarkan kepada pemasok, sehingga seluruh transaksi keuangan terdokumentasi dengan baik.

Dengan struktur *database* yang terorganisir, sistem ini dapat membantu dalam pengelolaan penjualan dan pembelian kepiting secara lebih efisien, memastikan ketersediaan stok, mencatat transaksi dengan akurat, dan memudahkan dalam pelacakan arus keuangan.

c. Halaman *Login*

Halaman *login* merupakan halaman awal yang digunakan oleh pengguna untuk mengakses sistem dengan cara memasukkan data autentikasi berupa *username* dan *password*. Fungsi utama dari halaman ini adalah sebagai pintu masuk menuju sistem yang hanya dapat diakses oleh pengguna yang telah terdaftar. Dengan adanya sistem login ini, keamanan data serta hak akses setiap pengguna dapat dikendalikan dan dibedakan sesuai dengan peran dan tanggung jawab masing-masing.

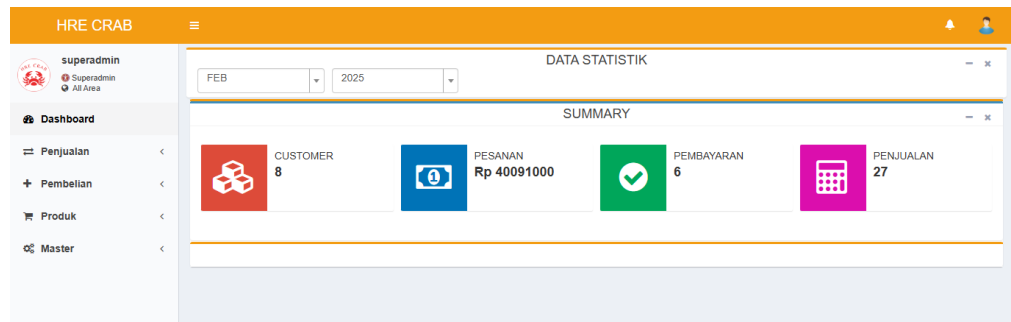


Gambar 3. 54 Login Crab

Dalam sistem penjualan dan pembelian kepiting, halaman *login* ini dirancang dengan tampilan yang sesuai dengan tema laut, seperti yang terlihat pada latar belakang gambar jaring dan tali kapal yang memberikan nuansa aktivitas nelayan. Elemen estetika ini tidak hanya memperkuat identitas sistem, tetapi juga memberikan pengalaman visual yang khas dan profesional kepada pengguna. Selain itu, tersedia juga fitur *Remember Me* yang memudahkan pengguna untuk tetap masuk tanpa harus *login* ulang setiap saat, selama menggunakan perangkat yang sama. Setelah pengguna berhasil *login*, mereka akan diarahkan ke halaman utama sistem untuk mengakses fitur-fitur seperti pencatatan transaksi, data stok, hingga laporan keuangan.

d. Menu *Dashboard*

*Dashboard* adalah halaman utama yang menyajikan rangkuman informasi penting dari sebuah sistem secara ringkas dan interaktif. Di dalam sistem informasi penjualan dan pembelian kepiting ini, *dashboard* berfungsi sebagai pusat kontrol yang memberikan gambaran umum mengenai aktivitas sistem berdasarkan data yang telah dimasukkan. Pengguna yang berhasil *login* akan langsung diarahkan ke halaman *dashboard* untuk memantau data terkini seperti jumlah pelanggan, total pesanan, pembayaran, serta jumlah penjualan.



Gambar 3. 55 Dashboard Crab

Pada bagian atas *dashboard* terdapat fitur filter berupa pilihan bulan dan tahun. Filter ini memungkinkan pengguna untuk menampilkan data statistik berdasarkan periode waktu tertentu. Setelah pengguna memilih bulan dan tahun yang diinginkan, sistem akan secara otomatis memperbarui tampilan data di bagian *summary*.

*Summary* merupakan bagian yang menampilkan ringkasan kinerja sistem berdasarkan filter waktu yang dipilih. Informasi yang ditampilkan yang terjadi pada periode tersebut. Seluruh data ini berguna untuk analisis cepat dan pengambilan keputusan oleh pihak manajemen, khususnya dalam mengevaluasi performa penjualan dan pembelian dalam periode tertentu.

e. Menu Produk

Menu Produk adalah fitur dalam sistem yang digunakan untuk mengelola semua data kepiting, dalam hal ini khususnya produk kepiting. Di dalamnya, pengguna seperti admin atau *superadmin* bisa melihat daftar produk yang sudah terdaftar, termasuk informasi seperti nama *container*, kategori kepiting, satuan unit, lokasi penyimpanan, dan stok yang tersedia. Tampilan ini adalah halaman *index*, tempat semua data produk ditampilkan secara ringkas. Kalau salah satu baris diklik, akan muncul detail stok berdasarkan sub-kode dari container tersebut, lengkap dengan jumlah stok masing-masing.

**Data Produk Makanan**  
Show 1-4 of 4 items

Pilih Kategori:  Pilih Produk:  Cari > (ENTER)

Actions	Kode	Nama	Kategori	Unit	Lokasi	Deskripsi
	B	Container B	Kepiting Besar	Gram	Container B	
	D	Container D	Kepiting Sedang	Gram	Container D	
	C	Container C	Kepiting Besar	Gram	Container C	

No.	Kode	Stok
1	C.51	0
2	C.52	10
3	C.53	0
4	C.54	0
		<b>10</b>

Actions	Kode	Nama	Kategori	Unit	Lokasi	Deskripsi
	A	Container A	Kepiting Raja	Gram	Container A	

Gambar 3. 56 Index Produk Crab

*Form* tambah kepiting ini digunakan untuk memasukkan data produk baru ke dalam sistem. Pengguna bisa mengisi kategori, unit, lokasi, kode, nama produk, dan deskripsi. Di bagian bawah juga tersedia fitur untuk menambahkan varian kode produk jika satu produk punya lebih dari satu jenis. Setelah semua data diisi, klik *submit* untuk menyimpan atau *close* untuk membatalkan.

**Tambah Makanan**

Kategori:  Unit:  Lokasi:

Kode:  Nama:

Deskripsi:

**Data Varian**

No.	Kode
1	Masukkan Kode <input type="text"/>
2	Masukkan Kode <input type="text"/>

Gambar 3. 57 Form Produk Crab

Halaman *index* stok ini digunakan untuk menampilkan sisa stok terakhir (*lastsize*) dari masing-masing produk setelah melalui proses penjualan atau pembelian kepiting. Data ini memudahkan pemantauan ketersediaan stok secara *real-time*.

10 Data

Pilih Kategori:  Container A  Cari > (ENTER)

Kode	Nama	Kategori	Stok	Unit	Lokasi	Deskripsi
A.51	Container A (A.51)	Kepiting Raja	200	Gram	Container A	
			<b>200</b>			

Show 1-1 of 1 items

Gambar 3. 58 Index Stok Crab

*Query* ini berfungsi untuk menampilkan stok akhir dari masing-masing produk dan variannya di halaman *index* stok. Prosesnya dimulai dari tabel varian

yang berelasi dengan tabel produk untuk mengambil informasi detail produk seperti nama, kategori, unit, lokasi, dan keterangan. Kemudian dilakukan *join* tambahan ke beberapa tabel *enum* untuk mengubah ID menjadi teks yang terbaca.

```
class Produk extends \yii\db\ActiveRecord
{
    public function searchstokbaru($params, $produktipe = '') {
        $sql = "
        SELECT A.varianid, A.lastsize, P.produktid, P.produktipe, B.enumtext_id as categorytext, C.enumtext_id as unittext, D.enumtext_id as lokasitext,
        B.enumno, A.varianno, P.produktid || '-' || A.varianno || '-' || B.enumtext_id || '-' || C.enumtext_id || '-' || D.enumtext_id as stokakhir
        FROM varian A LEFT JOIN produk P ON P.produktid = A.produktid
        LEFT JOIN enum B ON B.categoryid = B.enumid
        LEFT JOIN enum C ON C.unitid = C.enumid
        LEFT JOIN enum D ON D.locationid = D.enumid
        LEFT JOIN (
            -- Data dari pembelian/restock
            SELECT A.itemid, SUM(A.itemqty) as qty FROM trandetail A
            LEFT JOIN tran B ON A.tranid = B.tranid WHERE B.trantype = 'trantype.cr'
            GROUP BY A.itemid
        ) PB ON A.varianid = PB.itemid
        LEFT JOIN (
            -- Data dari penjualan/pengeluaran
            SELECT A.itemid, SUM(A.itemqty) as qty FROM trandetail A
            LEFT JOIN tran B ON A.tranid = B.tranid WHERE B.trantype = 'trantype.sa'
            GROUP BY A.itemid
        ) PJ ON A.varianid = PJ.itemid
        WHERE ($filter)
        ORDER BY
        P.produktid, A.varianid, B.enumtext_id, C.enumtext_id, D.enumtext_id
        "
    }
}
```

Gambar 3. 59 Query Stok Crab

Setelah data dasar diambil, *query* ini juga melakukan perhitungan stok akhir dengan menjumlahkan total pembelian atau *restock* lalu mengurangnya dengan total penjualan atau pengeluaran. Nilai akhir dari stok ini disebut *lastsize*, yang ditampilkan dalam kolom stok di halaman *index*. Nilai ini akan terus berubah mengikuti transaksi pembelian dan penjualan yang terjadi, sehingga selalu menampilkan jumlah stok terbaru.

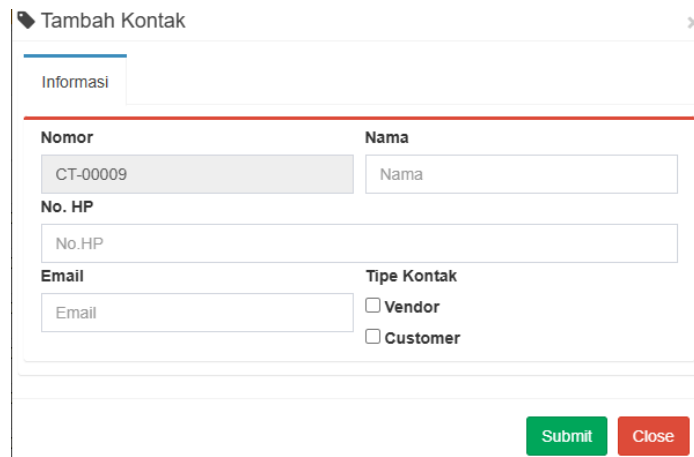
#### f. Menu Contact

<div> <div>+ Tambah</div> <div>Data Kontak</div> <div>Show 1-8 of 8 items</div> <div>10</div> <div>Data</div> </div>			
<div> <div>Carli Nomor &gt; {ENTER}</div> <div>Q Cari</div> <div>Filter Contact</div> </div>			
Actions	Nomor	Nama	No. HP
<div>⋮</div>	CT-00001	COATING FC	081317540001
<div>⋮</div>	CT-00002	LAPRAN	08116923262
<div>⋮</div>	CT-00003	哥哥	0800000000
<div>⋮</div>	CT-00004	Gege	0800000000

Gambar 3. 60 Index Contact Crab

Dalam menu *Contact* pada sistem transaksi penjualan dan pembelian keping ini hanya terdapat dua jenis data utama, yaitu *customer* dan *vendor*. *Customer* adalah pihak yang membeli keping, sedangkan *vendor* adalah pihak yang menjual atau memasok keping ke perusahaan. Informasi yang disimpan mencakup nomor kontak, nama, dan nomor HP. Data ini digunakan sebagai referensi saat membuat transaksi pembelian atau penjualan, agar pihak yang terlibat dalam transaksi dapat dipilih langsung tanpa perlu diinput ulang. Hal ini membantu mempercepat proses pencatatan dan memastikan data kontak selalu konsisten di setiap transaksi.





Gambar 3. 61 Form Contact Crab

Form ini digunakan untuk menentukan apakah sebuah kontak termasuk sebagai *vendor* atau *customer*. Di bagian ini terdapat dua *checkbox*, satu untuk menandai kontak sebagai *vendor* dan satu lagi sebagai *customer*. Jika *checkbox* *vendor* dicentang, maka data *isvendor* akan bernilai 1. Sebaliknya, jika *checkbox* *customer* dicentang, maka data *iscustomer* akan bernilai 2. Nilai ini kemudian akan disimpan ke dalam *database* sebagai bagian dari data kontak, sehingga saat transaksi dilakukan, sistem dapat dengan mudah membedakan apakah kontak tersebut adalah pihak yang menjual atau membeli kepiting.

```
<div class="col-lg-6 col-md-6 col-sm-12 col-xs-12">
  <div class="form-group">
    <label class="control-label">Tipe Kontak</label>
    <div class="d-flex">
      <div class="mr-3">
        <?=$form->field($model, 'isvendor', [
          'options' => ['tag' => false]
        ]->checkbox([
          'label' => 'Vendor',
          'value' => 1,
          'checked' => $model->isvendor == 1 ? true : false
        ]->label(false) ?>
      </div>
      <div>
        <?=$form->field($model, 'iscustomer', [
          'options' => ['tag' => false]
        ]->checkbox([
          'label' => 'Customer',
          'value' => 2,
          'checked' => $model->iscustomer == 2 ? true : false
        ]->label(false) ?>
      </div>
    </div>
  </div>
</div>
```

Gambar 3. 62 Code Form Contact

#### g. Menu Pembelian

Menu *restock* digunakan untuk mencatat transaksi pembelian kepiting dari *vendor*. Di dalamnya terdapat informasi seperti nomor transaksi, tanggal, nama *vendor*, serta detail produk yang dibeli. Data yang ditampilkan meliputi jumlah, harga, dan total biaya.

+ Restock

Data Transaksi Restock

Show 1-1 of 1 items

10

Data

\*

Pilih Contact

Semua

01/01/2021 - 30/12/2025

Cari > [ENTER]

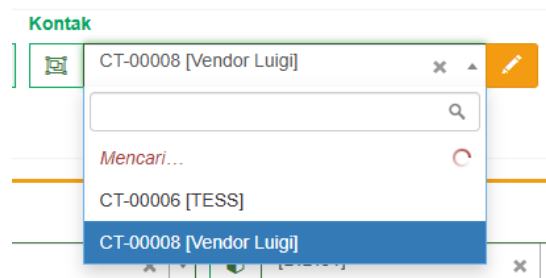
Q Cari

<div><div></div><div></div></div>	<div>Status</div> <div>Approve All</div>	No.Transaksi	Tgl.Transaksi	Kontak	Subtotal	Diskon	Total	Total Terbayar	Sisa	Status Bayar
<div><div></div><div></div></div>	<div>Approved</div>	CR-00016	16-04-2025	Vendor Luigi (CT-00008) marie@gmail.com - 0855555555	1.000.000	0	1.000.000	0	1.000.000	Belum Lu

<div><div></div><div></div></div>	No. 1	Produk	Biaya	Qty	Unit	Diskon	Total	Detail Pembayaran		
<div><div></div><div></div></div>	1	Container D	1.000.000	100	Gram	0	1.000.000	No.Kwitansi	Tgl.Bayar	Tot
				100		0	1.000.000	Tidak ada data yang ditemukan.		

Gambar 3. 63 Index Pembelian Crab

Pada *form* pembelian atau *restock*, ketika pengguna memilih kontak, sistem hanya akan menampilkan daftar kontak yang berjenis *isvendor*. Ini memastikan bahwa hanya vendor yang terdaftar yang bisa dipilih sebagai pemasok barang dalam proses pembelian. Kontak lain seperti *iscustomer* tidak akan muncul dalam daftar pilihan ini.



Gambar 3. 64 Dynamic Form Contact

## h. Menu Penjualan

+ New Transaction

Data Transaksi Penjualan

Show 1-1 of 1 items

Pilih Contact

Semua

01/01/2021 - 30/12/2025

Cari > [ENTER]

Q Cari

<div><div></div><div></div></div>	<div>Status</div> <div>Approves All</div>	No.Transaksi	Tgl.Transaksi	Kontak	Subtotal	Diskon	Total	Total Terbayar	Sisa	Status Bayar
<div><div></div><div></div></div>	<div>Approved -</div>	SA-00013	16-04-2025	火凤凰队01 Gege (CT-00004) 邮箱@gmail.com - 0800000000	2.000.000	0	2.000.000	0	2.000.000	Belum Lun

<div><div></div><div></div></div>	No. ↓	Produk	Biaya	Qty	Unit	Diskon	Total	Detail Pembayaran			
	1	Container D	2.000.000	100	Gram	0	2.000.000	No.Kwitansi			
				100		0	2.000.000	Tgl.Bayar			
								Tidak ada data yang ditemukan.			
							2.000.000	0	2.000.000	0	2.000.000

Gambar 3. 65 Index Penjualan Crab

Menu penjualan merupakan bagian dari sistem yang digunakan untuk mencatat dan mengelola transaksi penjualan produk ke pelanggan. Di halaman ini, pengguna dapat melihat daftar seluruh transaksi penjualan yang telah dilakukan, lengkap dengan informasi penting seperti nomor transaksi, tanggal, kontak pelanggan, rincian produk yang dijual, jumlah, harga, hingga status pembayaran. Menu ini juga memungkinkan pengguna untuk menambahkan

transaksi baru, mencari data penjualan berdasarkan tanggal atau kontak, serta memantau apakah pembayaran sudah dilakukan atau masih belum lunas.

Gambar 3. 66 Form Penjualan Crab

Pada *form* penjualan, saat pengguna memilih varian kepiting yang ingin dijual, kolom Gram akan otomatis terisi berdasarkan nilai dari kolom *lastsize*. Nilai *lastsize* ini mencerminkan sisa stok gram terakhir dari varian kepiting tersebut. Jika varian tersebut sudah pernah dibeli, maka *lastsize* akan diperbarui menjadi 0. Sebaliknya, saat dilakukan restock, nilai *lastsize* akan bertambah sesuai jumlah gram yang ditambahkan. Oleh karena itu, varian kepiting yang muncul dan bisa dipilih di *form* penjualan hanyalah varian yang masih memiliki isi, alias *lastsize*-nya tidak 0.

+ New Transaction

Data Transaksi Pembayaran

Show 1-1 of 1 items

10Data

Pembayaran Penjualan

01/01/2021 - 30/12/2025

gege

Car

			Status	No. Transaksi	Tgl. Pembayaran	No. Ref	Kontak	Total	Metode	Keterangan
			Approve All							
			Approved	CS-00006	16-04-2025	SA-00013	Gege (CT-00004) Paradise Island Walli Maria - 0800000000	2.000.000	Tunai	

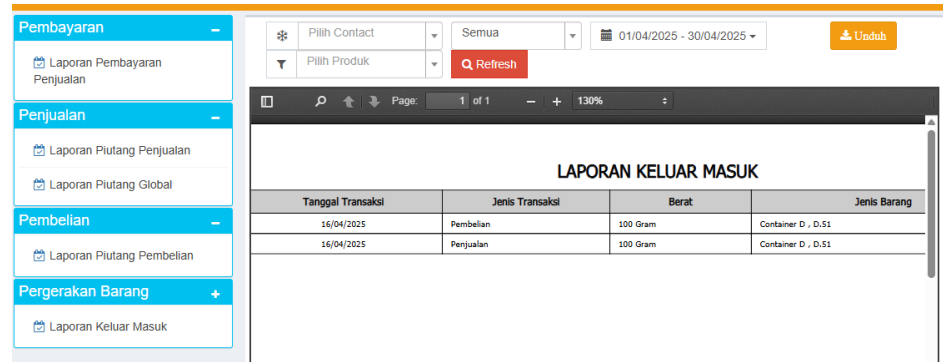
No.	Deskripsi	Tagihan	Jumlah Bayar
1	Penjualan	2.000.000	2.000.000
2	Diskon Penjualan	0	0
			2.000.000
			2.000.000

Gambar 3. 67 Cash Penjualan Crab

Menu pembayaran penjualan digunakan untuk mencatat dan memantau pembayaran yang dilakukan oleh pelanggan atas transaksi penjualan yang sudah terjadi. Di halaman ini, pengguna bisa melihat daftar pembayaran beserta informasi seperti nomor transaksi, tanggal pembayaran, referensi penjualan, nama kontak, total yang dibayarkan, metode pembayaran, dan keterangan tambahan. Menu ini berfungsi untuk memastikan bahwa setiap transaksi penjualan tercatat dengan jelas status pembayarannya dan membantu dalam pelacakan piutang atau pelunasan dari pelanggan.

#### i. Menu Laporan

Menu laporan berfungsi untuk menyajikan data-data penting yang berkaitan dengan transaksi dan aktivitas dalam sistem, sehingga memudahkan pengguna dalam melakukan pengecekan, evaluasi, maupun pengambilan keputusan. Salah satu menu laporan yang tersedia adalah laporan pergerakan barang.



LAPORAN KELUAR MASUK			
Tanggal Transaksi	Jenis Transaksi	Berat	Jenis Barang
16/04/2025	Pembelian	100 Gram	Container D , D.51
16/04/2025	Penjualan	100 Gram	Container D , D.51

Gambar 3. 68 Index Report Crab

### B. Hasil Belajar di IDUKA

Setelah melaksanakan Praktik Kerja Lapangan di PT Dwansoft Global Indonesia, penulis memperoleh sejumlah hasil belajar yang mencakup sikap, pengetahuan, dan keterampilan. Kegiatan yang dilakukan selama PKL memberi dampak positif terhadap perkembangan diri dan kesiapan penulis dalam menghadapi dunia kerja secara nyata.

#### 1. Capaian Sikap

Penulis menunjukkan perubahan sikap yang lebih baik setelah melaksanakan PKL. Penulis menjadi lebih disiplin terhadap waktu, bertanggung jawab dalam menyelesaikan tugas, serta lebih menghargai proses kerja di lingkungan profesional. Sikap kerja seperti teliti, sabar, dan mampu bekerja sama dalam tim juga mulai tumbuh dan terbentuk selama kegiatan berlangsung.

#### 2. Capaian Pengetahuan

Dari sisi pengetahuan, penulis memahami bagaimana sistem informasi dibangun dan diterapkan, khususnya sistem transaksi penjualan dan pembelian produk seperti kepiting. penulis mengetahui alur pengelolaan data kontak, transaksi *restock*, hingga laporan, serta mempelajari struktur database dan penerapan *query* SQL dalam sistem nyata.

Penulis memperoleh pemahaman nyata tentang alur kerja di industri, termasuk struktur organisasi, sistem operasional, serta penggunaan teknologi dan

*tools* yang sebelumnya hanya dikenal secara teori di sekolah. Penulis juga mengetahui pentingnya dokumentasi, manajemen data, dan alur komunikasi antar departemen.

### 3. Capaian Keterampilan

Penulis menjadi lebih terampil dalam menggunakan perangkat lunak dan sistem kerja yang digunakan di perusahaan, seperti input data, penggunaan form transaksi, validasi informasi, hingga pengolahan data berbasis *query* SQL. Selain itu, penulis juga mengasah kemampuan teknis seperti *debugging* sederhana, pemahaman terhadap sistem stok dan transaksi, serta keterampilan dokumentasi pekerjaan. Penulis juga menguasai penggunaan *framework* Yii2, pengintegrasian data antartabel, dan mampu membangun modul-modul fungsional untuk keperluan bisnis.

## **BAB IV**

### **PENUTUP**

#### **A. Kesimpulan**

Pelaksanaan Praktik Kerja Lapangan di PT Dwansoft Global Indonesia memberikan pengalaman langsung bagi penulis dalam memahami dan mengaplikasikan pengetahuan yang telah dipelajari di sekolah ke dalam lingkungan kerja nyata. Selama kegiatan PKL, penulis tidak hanya meningkatkan sikap profesional seperti kedisiplinan, tanggung jawab, dan etika kerja, tetapi juga memperoleh pemahaman yang lebih luas mengenai proses bisnis, sistem kerja, dan alur operasional perusahaan, khususnya dalam konteks sistem informasi berbasis Yii2.

Melalui keterlibatan dalam pengembangan dan pengelolaan sistem informasi penjualan, penulis memahami penerapan struktur *database*, relasi antar tabel, serta penggunaan *query* SQL untuk pengelolaan stok dan transaksi. Penulis juga terlibat dalam proses pengujian fitur seperti menu penjualan, pembelian, hingga manajemen pengguna, serta memahami pentingnya aspek keamanan seperti sistem *login*. Dengan demikian, PKL memberikan kontribusi nyata dalam membentuk kompetensi siswa baik dari sisi sikap, pengetahuan, maupun keterampilan teknis, yang semuanya menjadi bekal penting untuk memasuki dunia kerja.

#### **B. Saran**

##### **1. Saran Bagi Siswa PKL Berikutnya**

Bagi siswa yang akan melaksanakan PKL selanjutnya, disarankan untuk mempersiapkan diri sejak awal, terutama dalam penguasaan dasar-dasar *framework* dan bahasa pemrograman yang digunakan di industri seperti Yii2 dan SQL. Pemahaman terhadap alur kerja di dunia nyata juga penting agar lebih mudah beradaptasi. Siswa juga sebaiknya aktif bertanya, mencatat setiap proses yang dikerjakan, serta menjaga sikap profesional dalam bekerja agar pengalaman PKL menjadi lebih bermakna dan berdampak.

##### **2. Saran Bagi Sekolah**

Pihak sekolah diharapkan terus menjalin kerja sama yang aktif dengan perusahaan mitra yang relevan dengan kompetensi siswa, terutama di bidang pengembangan sistem informasi. Sekolah juga dapat mengembangkan kurikulum yang lebih sinkron dengan kebutuhan industri saat ini, serta menyediakan pelatihan

tambahan agar siswa lebih siap secara teknis dan mental sebelum turun ke lapangan. Selain itu, pembekalan mengenai etika kerja dan komunikasi profesional akan sangat membantu siswa saat berada di dunia kerja.

### 3. Saran Bagi IDUKA

Untuk pihak industri yang menjadi tempat PKL, sangat disarankan untuk terus membuka ruang belajar bagi siswa dengan memberikan proyek-proyek yang sesuai kemampuan namun tetap menantang. Pendampingan teknis yang konsisten juga akan sangat membantu siswa dalam memahami pekerjaan secara lebih mendalam. Selain itu, memberi kesempatan pada siswa untuk terlibat langsung dalam pengembangan sistem nyata akan memberikan pengalaman yang berharga dan memotivasi siswa dalam meningkatkan keterampilannya.

## DAFTAR LAMPIRAN

### A. Biodata Penulis

1. Nama Penulis : Isabell Gaol
2. Jenis Kelamin : Perempuan
3. NIS : 13422053
4. Kelas : XII Rpl 2
5. Kompetensi Keahlian : Rekayasa Perangkat Lunak
6. Tempat Tanggal Lahir : Batam, 23 Juni 2006
7. Agama : Kristen Protestan
8. Alamat Rumah : Taman Raya 2
9. No. Telepon / HP : 081261459656
10. E-mail : washiwaisabell@gmail.com
11. Nama Orangtua : Maria Nining Tri Lukita Ningsih
12. No HP Orang Tua : 08117777115
13. Jarak Rumah ke IDUKA : 2,6 Km
14. Transportasi ke IDUKA : Motor

### B. Data yang Relevan



Gambar Modul Dokumentasi API







*Foto Kegiatan Break*