

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Penelitian ini menggunakan studi literatur dari beberapa penelitian sebelumnya. Jurnal yang dipilih didasarkan dengan judul yang diambil yaitu Pengembangan Modul Kas Kecil dalam Aplikasi ERP *Odoo* dengan Metode *Rapid Application Development* (Studi Kasus Breezelabs.id) yang dapat dijelaskan sebagai berikut:

Penelitian pertama dengan judul Sistem Informasi Akuntansi Pengolahan Dana Kas Kecil Menggunakan Metode *Imprest* Pada PT Sinar Sosro Bandarlampung [6]. Tujuan dari penelitian ini adalah memberikan solusi efektif untuk menangani isu terkait manajemen dana kas kecil, dengan harapan dapat memfasilitasi proses pengeluaran, pengelolaan, dan pelaporan. Aplikasi yang dibangun menggunakan *Borland Delphi 7*. Permasalahan yang muncul dalam penelitian ini adalah bahwa proses pengeluaran bukti klaim masih dilakukan secara manual dan belum mengadopsi sistem komputerisasi, termasuk dalam perhitungan bukti klaim. Berdasarkan evaluasi fungsional menggunakan *black box testing*, sistem ini dapat berfungsi dengan baik tanpa terdapat kesalahan yang berasal dari aplikasinya.

Penelitian pertama dengan judul Implementasi Sistem ERP Proses Pengadaan pada Restaurant De Braga Hotel dengan Metode *Rapid Application Development* Berbasis *Odoo* [7]. Penelitian ini dimaksudkan untuk mengembangkan sistem ERP *Odoo* yang akan mengintegrasikan proses pengadaan bahan baku di sektor produksi, penjualan, dan pembayaran di restoran De Braga. Penelitian ini memiliki masalah terkait dengan pengoperasian sistem yang masih menggunakan pendekatan semi manual dan belum terintegrasi dengan baik antara satu dengan yang lainnya pada restoran De Braga. Temuan dari penelitian menunjukkan bahwa integrasi penggunaan modul *purchasing, manufacturing, inventory, sales, dan point of sale* secara simultan memberikan hasil data yang terintegrasi, memberikan dukungan

pada berbagai proses di departemen *Food and Beverage*, terutama dalam konteks pengadaan bahan baku.

Penelitian kedua dengan judul Analisis dan Implementasi *Timesheet Odoo ERP 10* dalam Manajemen Proyek Perangkat Lunak Pembelajaran PAI Berbasis ICT [8]. Tujuan dari penelitian ini adalah untuk menilai dan menerapkan sistem manajemen *Timesheet Odoo ERP 10* untuk memenuhi kebutuhan manajemen proyek pembelajaran PAI berbasis ICT. Penelitian ini menghadapi masalah terkait dengan kompleksitas manajemen waktu yang lebih tinggi dibandingkan dengan proyek yang dilakukan secara individu. Peneliti mengharapkan hasil yang optimal dan sesuai dengan jadwal perencanaan proyek pembelajaran PAI berbasis ICT.

Penelitian ketiga dengan judul Implementasi Aplikasi ERP Modul *Purchase Management* Menggunakan *Odoo 10* (Studi Kasus PT. Albasia Nusa Karya) [9]. Tujuan penelitian ini adalah untuk mengembangkan sebuah sistem ERP yang akan menggabungkan akuntansi, pengadaan, dan gudang organisasi. Kendala utama dalam penelitian ini berhubungan dengan kekurangan regulasi dalam pengadaan bahan baku dari tahap bisnis di PT. ANK. Hasil dari penelitian ini adalah proses pembelian bahan baku dapat dilakukan dengan lebih baik dari sebelumnya.

Penelitian keempat dengan judul Pengembangan *Green ERP Modul Sales and Distribution* untuk Industri Penyamakan Kulit dengan Metode ASAP [10]. Penelitian ini bertujuan untuk membangun sistem informasi menggunakan ERP (*Enterprise Resource Planning*) untuk modul penjualan dan distribusi dengan memanfaatkan aplikasi *Odoo 10.0*, menggunakan pendekatan metodologi ASAP. Permasalahan yang ada pada penelitian ini adalah proses pemasaran barang seperti hasil penjualan belum terintegrasi antar sistem. Konteks penelitian ini menunjukkan bahwa *output* yang dihasilkan adalah modul penjualan dan distribusi yang terintegrasi dengan *platform Odoo 10.0*, sesuai dengan kebutuhan perusahaan. Misi utamanya adalah memberikan kemudahan dalam mengatasi hambatan terkait dengan proses pemasaran barang.

Penelitian kelima dengan judul Pengembangan Manajemen Proyek Pembelajaran Berbasis ICT Menggunakan Metode *Accelerated SAP* Pada *Odoo ERP* [11]. Penelitian ini bertujuan untuk pembuatan sistem yang dapat mengatur

timeline menggunakan sistem ERP. Penelitian ini menghadapi masalah berupa kurangnya koordinasi yang efektif antara tim pengembang perangkat lunak, yang menyebabkan peningkatan waktu yang dibutuhkan untuk mengembangkan perangkat lunak melebihi batas waktu yang telah direncanakan. Hasil dari penelitian ini menunjukkan bahwa realisasi pengerjaan proyek dapat dikategorikan sebagai hasil yang memuaskan.

Penelitian keenam dengan judul Pengembangan Teknologi ERP untuk Modul *Human Resource Management* Studi Kasus PT Raja Digital Media [12]. Penelitian ini bertujuan mengimplementasikan *business process* manajemen sumber daya manusia (HRM) menggunakan perangkat lunak ERP, yaitu *Odoo*. Masalah yang dihadapi dalam penelitian ini terkait dengan pengelolaan bisnis proses yang dilakukan dengan cara tradisional, dan dampak mengenai performa perusahaan. Penelitian ini menghasilkan sistem pencarian informasi yang disederhanakan sehingga pencarian informasi penggajian pegawai jadi lebih mudah dengan waktu pencarian yang lebih efisien.

Penelitian ketujuh dengan judul Perancangan *Enterprise Resource Planning* Modul *Human Resource* dengan menggunakan *Odoo* pada PT Kinarya Alihdaya Mandiri [13]. Penelitian ini bertujuan untuk memberikan dukungan kepada bagian HRD PT. Kinarya Alihdaya Mandiri mengembangkan Sistem Informasi terintegrasi berbasis ERP (*Enterprise Resource Planning*) terutama dalam aspek modul Rekrutmen. Permasalahan yang dihadapi dalam penelitian ini terkait dengan penggunaan *Microsoft Excel* untuk pendataan karyawan baru yang masih kurang optimal dan dilakukan secara manual. *Output* dari penelitian ini mencakup desain modul yang memfasilitasi divisi HRD dalam memperlancar tahapan penerimaan pegawai baru, administrasi data, dan pencatatan arsip karyawan. Seluruh aktivitas rekrutmen direkam secara langsung dan *real-time* dalam aplikasi *Odoo*.

Penelitian kedelapan dengan judul Perancangan ERP Menu *Hr-Training* Berbasis *Odoo* Menggunakan Metode SDLC Studi Kasus PT.XYZ [14]. Merancang sistem manajemen ERP yang dapat mengatur pelatihan karyawan merupakan tujuan utama penelitian ini. Kendala yang dihadapi dalam penelitian ini berkaitan dengan isu administratif dalam manajemen sumber daya manusia di

lingkungan perusahaan. Hasil dari penelitian ini adalah pengembangan menu *HR Training* yang disesuaikan dengan kebutuhan pengguna sistem dan terdiri dari enam tahapan yang terperinci.

Penelitian kesembilan dengan judul Rancang Bangun API untuk *Odoo* ERP pada Modul CRM (*Customer Relationship Management*) [15]. Tujuan dari penelitian ini adalah membangun API *Odoo* yang terletak pada modul CRM dan menerapkan teknologi *Couchbase* di dalam *Odoo*. Ini dilakukan dengan maksud memungkinkan pertukaran data lokal ketika akses internet tidak tersedia, dan menyelaraskan data ketika koneksi internet kembali aktif. Permasalahan yang dihadapi dalam penelitian ini adalah ketika perangkat tidak dapat terhubung ke internet, tidak dapat melakukan transaksi data apa pun pada modul CRM. *Output* dari penelitian ini adalah berhasilnya pengembangan API *Odoo* di modul CRM, yang bertujuan untuk menggabungkan aplikasi mobile dan web dengan memanfaatkan bahasa pemrograman PHP, dengan *Laradoo* sebagai alat pendukung.

Penelitian kesepuluh dengan judul *Reengineering* Proses Bisnis Toko Oleh-Oleh Menggunakan *Enterprise Resource Planning Odoo 13* dengan *User Acceptance Test* sebagai Metode Pengujian Sistem [16]. Fokus penelitian ini adalah untuk mengenali proses bisnis pembelian, persediaan, dan penjualan barang yang ada pada Toko Wardani, serta menyusun usulan sistem terintegrasi dengan mengimplementasikan *Odoo 13*. Permasalahan yang dihadapi dalam penelitian ini terkait dengan penggunaan metode manual dalam fungsi bisnis pembelian, persediaan, dan penjualan di Toko Wardani. Hasil dari penelitian ini didasarkan pada pengujian menggunakan metode UAT (*User Acceptance Testing*), dan penilaian sistem *Odoo* mencapai angka 707, menunjukkan keberhasilannya.

Tabel 2.1 Tinjauan Pustaka

No	Judul	Comparing	Contrasting	Synthesize	Summarize
1	Sistem Informasi Akuntansi Pengolahan Dana Kas Kecil Menggunakan Metode <i>Imprest</i> Pada PT Sinar Sosro Bandarlampung	Tujuan dari penelitian ini adalah memberikan solusi efektif untuk menangani isu terkait manajemen dana kas kecil, dengan harapan dapat memfasilitasi proses pengeluaran, pengelolaan, dan pelaporan. Aplikasi yang dibangun menggunakan <i>Borland Delphi 7</i>	Penelitian sebelumnya merancang sistem menggunakan <i>Use Case Diagram</i> lalu dibangun menggunakan <i>Borland Delphi 7</i>	Permasalahan yang muncul dalam penelitian ini adalah bahwa proses pengeluaran bukti klaim masih dilakukan secara manual dan belum mengadopsi sistem komputerisasi, termasuk dalam perhitungan bukti klaim.	Berdasarkan evaluasi fungsional menggunakan <i>black box testing</i> , sistem ini dapat berfungsi dengan baik tanpa terdapat kesalahan yang berasal dari aplikasinya.
2	Implementasi Sistem ERP Proses Pengadaan pada Restaurant De Braga Hotel dengan Metode <i>Rapid Application Development</i> Berbasis <i>Odoo</i>	Penelitian ini dimaksudkan untuk mengembangkan sistem ERP <i>Odoo</i> yang akan mengintegrasikan proses pengadaan bahan baku di sektor	Penelitian sebelumnya menggunakan <i>Rapid Application Development</i>	Masalah yang dihadapi dalam penelitian ini terkait dengan pengoperasian sistem yang masih menggunakan	Temuan dari penelitian menunjukkan bahwa integrasi penggunaan modul <i>purchasing, manufacturing, inventory, sales, dan point of sale</i> secara

No	Judul	Comparing	Contrasting	Synthesize	Summarize
		produksi, penjualan, dan pembayaran di restoran De Braga.		pendekatan semi manual dan belum terintegrasi dengan baik antara satu dengan yang lainnya pada restoran De Braga	simultan memberikan hasil data yang terintegrasi, memberikan dukungan pada berbagai proses di departemen <i>Food and Beverage</i> , terutama dalam konteks pengadaan bahan baku.
3	Analisis dan Implementasi <i>Timesheet Odoo ERP 10</i> dalam Manajemen Proyek Perangkat Lunak Pembelajaran PAI Berbasis ICT	Penelitian ini memiliki tujuan untuk menganalisis dan mengimplementasikan sistem manajemen <i>Timesheet Odoo ERP 10</i> dalam konteks manajemen proyek pembelajaran PAI berbasis ICT	Penelitian sebelumnya tidak menjelaskan metode apa yang digunakan.	Penelitian ini menghadapi masalah terkait dengan kompleksitas manajemen waktu yang lebih tinggi dibandingkan dengan proyek yang dilakukan secara individu.	Peneliti mengharapkan hasil yang optimal dan sesuai dengan jadwal perencanaan proyek pembelajaran PAI berbasis ICT.
4	Implementasi Aplikasi ERP Modul <i>Purchase Management</i> Menggunakan <i>Odoo 10</i>	Tujuan penelitian ini adalah untuk mengembangkan sebuah sistem ERP	Penelitian sebelumnya menggunakan <i>iterative</i>	Kendala utama dalam penelitian ini berhubungan dengan	Hasil dari penelitian ini adalah proses pembelian bahan baku dapat dilakukan

No	Judul	Comparing	Contrasting	Synthesize	Summarize
	(Studi Kasus PT. Albasia Nusa Karya)	yang akan menggabungkan akuntansi, pengadaan, dan gudang organisasi.	<i>waterfall.</i>	kekurangan regulasi dalam pengadaan bahan baku dari tahap bisnis di PT. ANK.	dengan lebih baik dari sebelumnya.
5	Pengembangan <i>Green ERP Modul Sales and Distribution</i> untuk Industri Penyamakan Kulit dengan Metode ASAP	Penelitian ini bertujuan untuk membangun sistem informasi menggunakan ERP (<i>Enterprise Resource Planning</i>) untuk modul penjualan dan distribusi dengan memanfaatkan aplikasi <i>Odoo 10.0</i> , menggunakan pendekatan metodologi ASAP.	Penelitian sebelumnya menggunakan <i>Accelerated SAP</i> .	Permasalahan yang ada pada penelitian ini adalah proses pemasaran barang seperti hasil penjualan belum terintegrasi antar sistem.	Konteks penelitian ini menunjukkan bahwa <i>output</i> yang dihasilkan adalah modul penjualan dan distribusi yang terintegrasi dengan <i>platform Odoo 10.0</i> , sesuai dengan kebutuhan perusahaan. Misi utamanya adalah memberikan kemudahan dalam mengatasi hambatan terkait dengan proses pemasaran barang.
6	Pengembangan Manajemen Proyek Pembelajaran Berbasis ICT	Penelitian ini bertujuan untuk pembuatan sistem yang dapat mengatur	Penelitian sebelumnya menggunakan <i>Accelerated</i>	Penelitian ini menghadapi masalah berupa kurangnya	Hasil dari penelitian ini menunjukkan bahwa realisasi pengerjaan proyek

No	Judul	Comparing	Contrasting	Synthesize	Summarize
	Menggunakan Metode <i>Accelerated</i> SAP Pada <i>Odoo</i> ERP	<i>timeline</i> menggunakan sistem ERP	SAP.	koordinasi yang efektif antara tim pengembang perangkat lunak, yang menyebabkan peningkatan waktu yang dibutuhkan untuk mengembangkan perangkat lunak melebihi batas waktu yang telah direncanakan.	dapat dikategorikan sebagai hasil yang memuaskan.
7	Pengembangan Teknologi ERP Untuk Modul <i>Human Resource Management</i> Studi Kasus PT Raja Digital Media	Tujuan dari penelitian ini adalah mengimplementasikan proses bisnis manajemen sumber daya manusia (HRM) menggunakan perangkat lunak ERP, yaitu <i>Odoo</i> .	Penelitian sebelumnya menggunakan metode survei dan wawancara, studi literatur, analisa sistem, serta desain sistem.	Masalah yang dihadapi dalam penelitian ini terkait dengan pengelolaan bisnis proses yang dilakukan dengan cara tradisional, dan dampak mengenai performa	Hasil dari penelitian ini adalah sistem yang dirancang untuk memudahkan pencarian informasi penggajian pegawai dengan waktu pencarian yang lebih efisien.

No	Judul	Comparing	Contrasting	Synthesize	Summarize
				perusahaan.	
8	Perancangan <i>Enterprise Resource Planning</i> Modul <i>Human Resource</i> dengan menggunakan <i>Odoo</i> pada PT Kinarya Alihdaya Mandiri	Penelitian ini bertujuan untuk memberikan dukungan kepada bagian HRD PT. Kinarya Alihdaya Mandiri mengembangkan Sistem Informasi terintegrasi berbasis ERP (<i>Enterprise Resource Planning</i>) terutama dalam aspek modul Rekrutmen	Penelitian sebelumnya menggunakan metode dengan beberapa fase yaitu identifikasi masalah, pengumpulan data, analisa, desain dan penerapan.	Permasalahan yang dihadapi dalam penelitian ini terkait dengan penggunaan <i>Microsoft Excel</i> untuk pendataan karyawan baru yang masih kurang optimal dan dilakukan secara manual.	<i>Output</i> dari penelitian ini mencakup desain modul yang memfasilitasi divisi HRD dalam memperlancar tahapan penerimaan pegawai baru, administrasi data, dan pencatatan arsip karyawan. Seluruh aktivitas rekrutmen direkam secara langsung dan <i>real-time</i> dalam aplikasi <i>Odoo</i> .
9	Perancangan ERP Menu <i>Hr-Training</i> Berbasis <i>Odoo</i> Menggunakan Metode SDLC Studi Kasus PT.XYZ	Merancang sistem manajemen ERP yang dapat mengatur pelatihan karyawan merupakan tujuan utama penelitian ini.	Penelitian sebelumnya menggunakan metode SDLC (<i>Software Development Life Cycle</i>).	Kendala yang dihadapi dalam penelitian ini berkaitan dengan isu administratif dalam manajemen sumber daya manusia di	Hasil dari penelitian ini adalah pengembangan menu <i>HR Training</i> yang disesuaikan dengan kebutuhan pengguna sistem dan terdiri dari enam tahapan yang terperinci.

No	Judul	Comparing	Contrasting	Synthesize	Summarize
				lingkungan perusahaan.	
10	Rancang Bangun API untuk <i>Odoo</i> ERP pada Modul CRM (<i>Customer Relationship Management</i>)	Tujuan dari penelitian ini adalah membangun API <i>Odoo</i> yang terletak pada modul CRM dan menerapkan teknologi <i>Couchbase</i> di dalam <i>Odoo</i> . Ini dilakukan dengan maksud memungkinkan pertukaran data lokal ketika akses internet tidak tersedia, dan menyelaraskan data ketika koneksi internet kembali aktif.	Penelitian sebelumnya tidak menjelaskan metode apa yang digunakan.	Permasalahan yang dihadapi dalam penelitian ini adalah ketika perangkat tidak dapat terhubung ke internet, tidak dapat melakukan transaksi data apa pun pada modul CRM.	<i>Output</i> dari penelitian ini adalah berhasilnya pengembangan API <i>Odoo</i> di modul CRM, yang bertujuan untuk menggabungkan aplikasi mobile dan web dengan memanfaatkan bahasa pemrograman PHP, dengan Laradoo sebagai alat pendukung.
11	<i>Reengineering</i> Proses Bisnis Toko Oleh-Oleh Menggunakan <i>Enterprise Resource Planning Odoo</i> 13 dengan <i>User Acceptance Test</i> sebagai Metode Pengujian	Fokus penelitian ini adalah untuk mengenali proses bisnis pembelian, persediaan, dan penjualan barang yang ada pada Toko	Penelitian sebelumnya menggunakan metode <i>User Acceptance Test</i>	Permasalahan yang dihadapi dalam penelitian ini terkait dengan penggunaan metode manual	Hasil dari penelitian ini didasarkan pada pengujian menggunakan metode UAT (<i>User Acceptance Testing</i>), dan penilaian sistem

No	Judul	Comparing	Contrasting	Synthesize	Summarize
	Sistem	Wardani, serta menyusun usulan sistem terintegrasi dengan mengimplementasikan <i>Odoo</i> 13.		dalam fungsi bisnis pembelian, persediaan, dan penjualan di Toko Wardani.	<i>Odoo</i> mencapai angka 707, menunjukkan keberhasilannya.

Berdasarkan pada sepuluh penelitian terdahulu yang telah diuraikan. Inti dari penelitian terdahulu ialah membangun atau mengembangkan sistem untuk memudahkan pekerjaan manusia. Perbedaan antara penelitian Pengembangan Modul Kas Kecil dalam Aplikasi ERP *Odoo* dengan Metode *Rapid Application Development* (Studi Kasus Breezelabs.id) dengan penelitian terdahulu yaitu pada metode yang digunakan dan fitur yang dibangun atau dikembangkan. Fitur – fitur yang dikembangkan di penelitian terdahulu adalah modul *Purchase, Sales, Customer Relationship Management*, dan lainnya. Sedangkan metode pengembangan yang digunakan di penelitian terdahulu terdapat *Rapid Application Development, User Acceptance Test, Accelerated SAP*, dan *Software Development Life Cycle*. Akan tetapi, semua penelitian terdahulu menggunakan *Odoo* ERP sebagai *framework* dan *platform* untuk mengembangkan projeknya.

2.2. Landasan Teori

2.2.1. Odoo

Perangkat lunak ERP *Odoo Open-Source* telah menjadi pilihan populer bagi perusahaan untuk mengimplementasikan sistem ERP dalam kegiatan bisnis mereka. Pada versi komunitasnya, pengguna dapat mengunduh *Odoo* secara gratis dan menyesuaikan aplikasinya sesuai kebutuhan, serta menggunakannya tanpa biaya. *Odoo* menyediakan berbagai modul yang lengkap, seperti CRM, Manajemen Pembelian, Akuntansi, Manajemen Penjualan, dan modul lainnya. Melalui pemanfaatan sistem ERP *Odoo*, perusahaan dapat mengintegrasikan semua aspek kegiatan bisnisnya, membentuk struktur yang tertata rapi, dan mengurangi risiko dengan cara yang signifikan. [17].

Dibangun dengan menggunakan bahasa pemrograman *Python, XML, dan JavaScript*, *Odoo (OpenERP)* adalah aplikasi web yang manajemen basis datanya mengandalkan *PostgreSQL*. *Odoo* dirancang untuk mengelola beragam jenis bisnis, mencakup perusahaan jasa, perdagangan, dan manufaktur. Setiap individu memiliki kemampuan untuk menggabungkan dan mengkonfigurasi *Odoo* sesuai dengan kebutuhannya sendiri [18].

2.2.2. Kas Kecil

Menurut [19], Kas kecil merujuk pada sejumlah uang tunai yang dialokasikan perusahaan untuk menutupi pengeluaran dengan nilai yang tergolong kecil, karena penggunaan cek tidak memadai secara ekonomis. Contoh pengeluaran dengan jumlah kecil tersebut meliputi penyetakan makanan kecil, pembayaran *wifi*, pembelian ATK, atau sejenisnya. Sebagai tanggung jawab, perusahaan menyerahkan dana kas kecil kepada petugas yang bertugas mengelola kas kecil, yang mempunyai tanggung jawab penuh terhadap pengisian dan penggunaan dana kas kecil dalam rentang waktu tertentu. Pada pengelolaan kas kecil, terdapat dua metode yang dapat

diterapkan, yakni metode *imprest* dan metode fluktuasi [20]. Rincian perbandingan perbedaan antara kedua metode tersebut dapat diakses melalui Tabel 2.2.

Tabel 2.2 Perbandingan Metode *Imprest* dan Fluktuasi

Metode <i>Imprest</i>	Metode <i>Fluktuasi</i>
Metode pengisian dan pengendalian kas kecil yang digunakan adalah metode <i>imprest</i> , di mana jumlah kas kecil tetap dari waktu ke waktu. Ini disebabkan oleh kenyataan bahwa pengisian kembali kas kecil selalu setara dengan jumlah yang telah dikeluarkan sebelumnya.	Metode pencatatan dan pengendalian kas kecil diimplementasikan dengan prinsip bahwa kas kecil akan selalu mengalami perubahan karena pengisian kembali selalu dilakukan dengan jumlah yang tetap dari waktu ke waktu.

2.2.3. Enterprise Resource Planning (ERP)

Menurut penelitian [21], Konsep atau perangkat lunak yang dikenal sebagai *Enterprise Resource Planning* (ERP) berperan dalam mengatur dan mengelola sumber daya perusahaan. Dampaknya melibatkan segala lapisan manajemen, mulai dari tingkat eksekutif hingga ke aktivitas operasional perusahaan. ERP biasanya menggabungkan modul berdasarkan fungsi operasional perusahaan.

Pada penelitian [22] mengatakan ERP adalah perangkat lunak yang menawarkan solusi bagi perusahaan besar. Sistem ERP mencakup sejumlah modul perangkat lunak yang mendukung berbagai fungsi, termasuk pemasaran dan penjualan, layanan lapangan, desain dan pengembangan produk, manajemen produksi dan inventaris, pengadaan, penjualan, manajemen fasilitas industri, desain dan pengembangan proses, manufaktur, kualitas, sumber daya manusia, keuangan dan akuntansi, serta layanan informasi. Menurut [23] ERP adalah sistem terintegrasi berbasis komputer

untuk meningkatkan kinerja inovasi dan kualitas kinerja, untuk meningkatkan akurat informasi antar departemen dalam suatu perusahaan, untuk daya tanggap dan dukungan pelanggan kepada pelanggan secara langsung. Dan mempengaruhi kinerja bisnis dengan menyediakan Perusahaan dalam pengambilan keputusan dan penggunaan sumber daya yang bijak.

Dapat disimpulkan pengertian ERP mampu mengurus semua kegiatan bisnis yang berlangsung di perusahaan, dengan baik dan sesuai aturan yang berlaku. Selain itu ERP juga update data secara real time. Jika proses bisnis dikelola dengan baik maka dapat sebagai acuan pengambil keputusan.

2.2.4. Keuangan dan Akuntansi

Menurut [24] bidang akuntansi keuangan memusatkan perhatiannya pada penyampaian laporan informasi kepada pihak-pihak yang berkepentingan di luar perusahaan. Sedangkan menurut [25] laporan keuangan yang melibatkan pendapatan dan aliran kas perusahaan memberikan nilai tambah dalam mendukung pengambilan keputusan oleh berbagai pihak. Adapun menurut [26] sistem akuntansi keuangan adalah suatu mekanisme yang mencatat dan mengolah data ekonomi perusahaan, melibatkan langkah - langkah mulai dari pencatatan, pendaftaran, rekapitulasi, hingga penyusunan laporan keuangan dan informasi keuangan. Semua ini bertujuan sebagai dasar pada tahap perencanaan, pelaksanaan, dan pertanggungjawaban dalam hal pengambilan keputusan

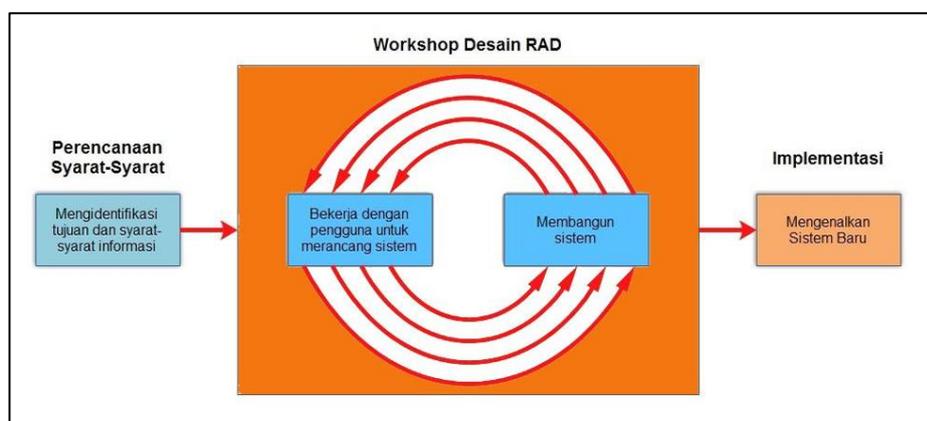
Kesimpulan akuntansi keuangan adalah laporan yang disusun untuk pengambilan keputusan oleh orang yang berkepentingan di perusahaan.

2.2.5. *Rapid Application Development (RAD)*

Rapid Application Development (RAD) adalah pendekatan tertentu dalam pengembangan sistem perangkat lunak atau aplikasi yang mengutamakan kecepatan dan efisiensi dalam proses pengembangan yang singkat. Metode ini menggunakan teknik yang berfokus pada pendekatan berorientasi objek, termasuk pengembangan dan pembuatan perangkat lunak.

Pendekatan RAD melibatkan penggunaan metode iteratif dan pembangunan model sistem pada tahap awal pengembangan, dengan tujuan utama mengidentifikasi kebutuhan pengguna. Metode RAD dikenal sebagai proses perancangan aplikasi yang menekankan pengembangan dengan siklus singkat [27].

Metode Rapid Application Development terdiri dari tiga tahap yang harus dilalui untuk mencapai sistem yang diinginkan, yaitu:



Gambar 2.1 Metode RAD

a. Rencana Kebutuhan

Dalam fase awal, terjadi keterlibatan antara pengguna dan analis dalam suatu pertemuan untuk menetapkan sasaran sistem dan menggali kebutuhan informasi demi mencapai kesepakatan tujuan yang telah disepakati. Proses perencanaan kebutuhan sangat penting karena melibatkan kedua belah pihak dan memastikan keterlibatan aktif dari mereka [27].

b. Desain Sistem

Pada fase selanjutnya, partisipasi aktif pengguna memegang peran krusial dalam mencapai tujuan sistem, karena desain dan perbaikan sistem terjadi melalui proses yang melibatkan umpan balik antara pengguna dan peneliti. Jika terdapat ketidaksesuaian dalam desain, pengguna dapat menghubungi peneliti dan memberikan komentar langsung. Fase ini memberikan hasil spesifikasi *software* yang melibatkan struktur sistem,

database, dan hal lainnya. [27].

c. Implementasi

Fase terakhir melibatkan peran *programmer* dalam mengembangkan program sesuai dengan pengguna dan peneliti sudah mencapai kesepakatan bersama. Perlu dilakukan proses pengujian untuk mendeteksi adanya kesalahan atau bug sebelum program tersebut diimplementasikan dalam organisasi. Partisipasi pengguna mencakup memberikan tanggapan terhadap sistem yang telah dibuat dan menyetujui keberlanjutan penggunaan sistem tersebut [27].

2.2.6. Black Box Testing

Metode pengujian *Black Box* digunakan untuk mengevaluasi fitur dan fungsi sistem yang telah dikembangkan. Langkah ini penting sebelum sistem dirilis karena dapat mengidentifikasi masalah atau kesalahan yang mungkin terjadi. Pengujian fungsionalitas sistem dengan menggunakan metode ini relatif mudah dan sederhana [28]. *Black Box Testing* memiliki peran yang penting dalam pengujian sebuah sistem dan membantu validasi fungsionalitas secara keseluruhan sistem. *Black Box Testing* dieksekusi berdasarkan kebutuhan pelanggan, sehingga fitur yang tidak lengkap atau belum sesuai ekspektasi *customer* akan mudah untuk diidentifikasi. Proses melakukan pengujiannya berdasarkan perspektif pengguna akhir. Semua tim *developer* dilibatkan dalam melakukan pengujian. Menyiapkan *test scenario* termasuk dalam proses pengujian [29].

2.2.7. Unified Modelling Language (UML)

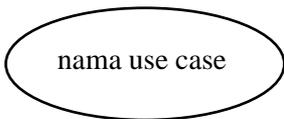
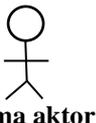
Unified Modeling Language (UML) digunakan sebagai pendekatan dalam pengembangan sistem untuk menyajikan spesifikasi secara visual. Kegunaan UML melibatkan desain perangkat lunak, fungsi sebagai alat komunikasi antara perangkat lunak dan proses bisnis, dan penggunaan elemen grafis dalam pembuatan diagram [30]. Penelitian ini menggunakan beberapa jenis diagram UML diantaranya adalah *Use Case Diagram*, *Activity Diagram*,

Sequence Diagram, dan *Class Diagram*.

a. *Use Case Diagram*

Dalam lingkup UML, *Use Case Diagram* digunakan sebagai alat untuk menggambarkan serta memodelkan cara aktor dan *use case* berinteraksi dalam suatu sistem. Fungsinya melibatkan penyusunan dan visualisasi perilaku sistem secara statis. Diagram ini memperlihatkan hubungan serta interaksi antara aktor (entitas eksternal) dan *use case* (fungsi sistem) yang terlibat dalam sistem [30]. Pada Tabel 2.3 dijelaskan terkait simbol – simbol dari *use case diagram*.

Tabel 2.3 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use case</i></p> 	Bertindak sebagai perantara untuk entitas luar yang terlibat dalam interaksi dengan sistem, seperti pengguna, sistem lain, atau perangkat eksternal.
<p>Aktor / <i>actor</i></p> 	Menyatakan peran sebagai perwakilan entitas dari luar yang terlibat dalam interaksi dengan sistem, seperti pengguna, sistem lain atau perangkat eksternal.
<p>Asosiasi / <i>association</i></p> <p>.</p>	Memberikan gambaran tentang hubungan antara aktor dan <i>use case</i> , menunjukkan partisipasi aktor dalam suatu skenario penggunaan.
<p>Extensi / <i>extend</i></p> <p>▼ <<extend>></p>	Mengindikasikan bahwa dalam pelaksanaannya, suatu <i>use case</i> memiliki kemampuan untuk memperluas atau mengambil alih sebagian fungsi dari <i>use</i>

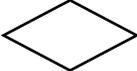
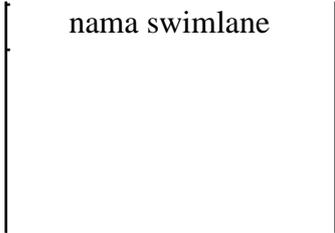
Simbol	Deskripsi
	<i>case</i> lain.
Generalisasi <i>generalization</i> 	/ Menggambarkan struktur hirarkis antara <i>use case</i> , di mana <i>use case</i> utama berfungsi sebagai <i>use case</i> umum, dan <i>use case</i> anak mengadopsi fungsionalitas dari <i>use case</i> utama.
Include / <i>uses</i> <<include>> ▼ <<uses>> ▼	Mengindikasikan bahwa satu <i>use case</i> membutuhkan atau melibatkan eksekusi <i>use case</i> lain.

b. *Activity Diagram*

Diagram aktivitas adalah salah satu jenis diagram UML yang digunakan untuk mengilustrasikan aliran kerja atau urutan aktivitas dalam suatu sistem. Diagram ini memberikan gambaran tentang bagaimana aktivitas - aktivitas terkait dan bagaimana aliran kerja berlangsung dari satu aktivitas ke aktivitas lainnya [30]. Pada tabel 2.4 dijelaskan terkait simbol – simbol dari *activity diagram*.

Tabel 2.4 Simbol *Activity Diagram*

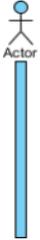
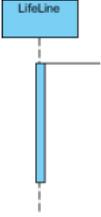
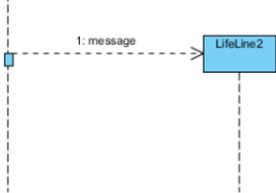
Simbol	Deskripsi
Status awal 	Komponen yang menunjukkan status permulaan dari aktivitas yang dijalankan oleh sistem.
Aktivitas 	Bagian yang mencirikan kegiatan di dalam sistem, umumnya dimulai dengan kata kerja.

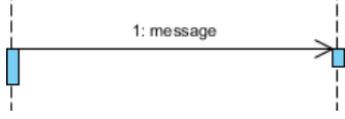
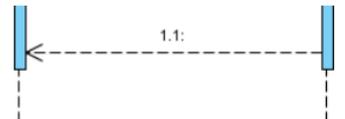
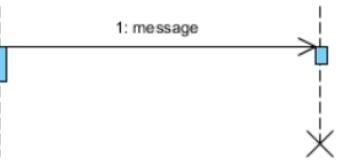
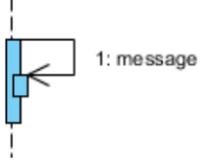
Simbol	Deskripsi
Percabangan / <i>decision</i> 	Menggambarkan percabangan keputusan dalam alur aktivitas, di mana syarat tertentu harus terpenuhi sebelum melanjutkan eksekusi aktivitas berikutnya.
Penggabungan / <i>join</i> 	Dipakai untuk menyatukan kembali jalur - jalur menjadi satu alur.
Status akhir 	Komponen yang mencirikan lokasi akhir atau titik selesainya <i>Activity Diagram</i> .
Control Flow 	Komponen yang menggambarkan perjalanan alur dari satu aktivitas ke aktivitas lainnya, mencirikan urutan dan ketergantungan antara aktivitas.
Swimlane 	Komponen yang menunjukkan wadah untuk setiap aktivitas yang berlangsung.

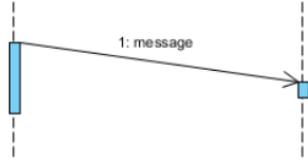
c. Sequence Diagram

Sequence Diagram digunakan untuk memvisualisasikan pemrosesan yang terjadi dalam skenario *use case*. Secara praktis, diagram ini sering diterapkan dalam proses desain sistem untuk menggambarkan interaksi, hubungan, dan metode -metode yang dimiliki oleh objek - objek dalam sistem [30]. Pada tabel 2.5 dijelaskan terkait simbol – simbol *sequence diagram*.

Tabel 2.5 Simbol *Sequence Diagram*

Simbol	Deskripsi
<p data-bbox="427 450 507 479">Aktor</p> 	<p data-bbox="826 450 1356 539">Peran yang dijalankan oleh entitas yang terlibat dalam interaksi dengan subjek.</p>
<p data-bbox="427 786 699 815">Garis hidup / <i>lifeline</i></p> 	<p data-bbox="826 786 1356 869">Menggambarkan peran individu dalam interaksi</p>
<p data-bbox="427 1167 587 1196">Waktu aktif</p> 	<p data-bbox="826 1167 1356 1317">Garis panjang tipis pada <i>lifeline</i> mencerminkan durasi saat elemen melakukan operasi.</p>
<p data-bbox="427 1503 655 1532">Pesan tipe <i>create</i></p> 	<p data-bbox="826 1503 1356 1585">Kategori pesan yang menunjukkan pembentukan dari <i>lifeline</i> yang dituju.</p>
<p data-bbox="427 1845 624 1874">Pesan tipe <i>call</i></p>	<p data-bbox="826 1845 1356 1989">Tipe pesan yang melambangkan permintaan untuk pelaksanaan operasi oleh <i>lifeline</i> yang menjadi tujuan.</p>

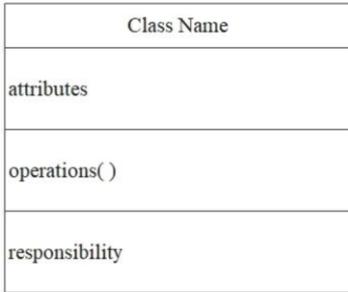
Simbol	Deskripsi
	
<p>Pesan tipe <i>send</i></p> <p>▼ <u>1: masukan</u></p>	<p>Indikator yang menunjukkan suatu objek yang melakukan pengiriman data, masukan, atau informasi ke objek lain.</p>
<p>Pesan tipe <i>return</i></p> 	<p>Ini mencakup pesan-pesan yang ditujukan untuk mengirimkan data respons kembali ke pesan panggilan yang sudah dikirim sebelumnya, menjaga hubungan korespondensi dengan komunikasi awal.</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Kategori pesan yang mengindikasikan permohonan untuk menghapus target lifeline.</p>
<p>Pesan tipe <i>self</i></p> 	<p>Kategori pesan yang mengindikasikan permintaan untuk melakukan operasi pada <i>target lifeline</i>.</p>
<p>Pesan tipe <i>Recursive</i></p> 	<p>Jenis pesan yang menandakan pengambilan sebuah pesan dari lifeline yang sama. Tujuan pesan ini merujuk pada aktivasi yang terletak di atas aktivasi di mana pesan itu dimulai.</p>
<p>Pesan tipe durasi</p>	<p>Mengukur interval waktu antara dua</p>

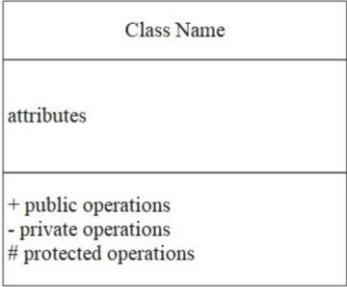
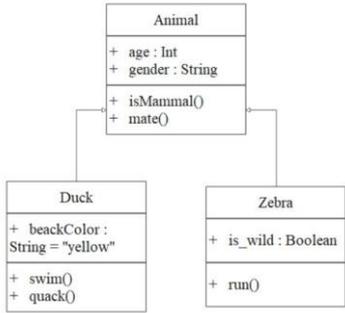
Simbol	Deskripsi
	instan untuk memanggil pesan.
<p data-bbox="427 577 491 611"><i>Note</i></p> 	Memberikan fasilitas untuk melampirkan catatan tambahan pada elemen spesifik. Catatan ini tidak mengubah signifikansi semantik dari elemen tersebut, namun menyediakan informasi tambahan yang bermanfaat bagi individu yang melakukan pemodelan.

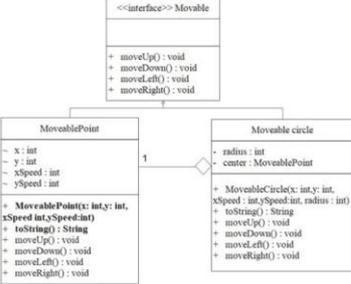
d. *Class Diagram*

Class Diagram dirancang untuk menggambarkan hubungan dan struktur antara kelas - kelas dalam suatu sistem. Fokusnya adalah mengilustrasikan entitas - entitas dan koneksi di antara kelas - kelas tersebut, melibatkan aspek fungsi - fungsi sistem dan kebutuhan menu utama serta basis data [30]. Pada tabel 2.6 dijelaskan terkait simbol – simbol *class diagram*.

Tabel 2.6 Simbol *Class Diagram*

Simbol	Deskripsi
<p data-bbox="427 1603 507 1637"><i>Class</i></p> 	Nama kelas selalu ditampilkan di bagian pertama, atribut di bagian kedua, dan operasi di bagian ketiga. Atribut adalah nilai-nilai yang mendefinisikan sebuah kelas. Kelas dapat menjalankan proses yang dikenal sebagai operasi.

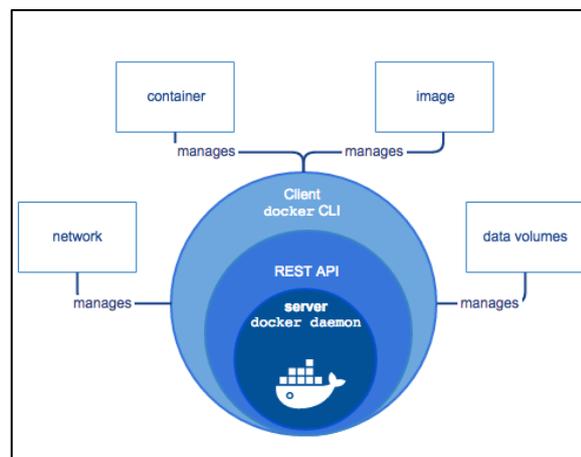
Simbol	Deskripsi
<p><i>Visibility</i></p>  <pre> classDiagram class Animal { + age : Int + gender : String + isMammal() + mate() } class Duck { + beakColor : String = "yellow" + swim() + quack() } class Zebra { + is_wild : Boolean + run() } Animal < -- Duck Animal < -- Zebra </pre>	<p>Simbol visibilitas digunakan untuk menentukan aksesibilitas informasi yang terdapat dalam kelas.</p>
<p><i>Bidirectional Association</i></p> <p>.</p>	<p>Asosiasi bilateral direpresentasikan oleh garis lurus yang menghubungkan dua kelas. Ini hanya menunjukkan bahwa kelas-kelas tersebut mengetahui hubungan mereka satu sama lain.</p>
<p><i>Unilateral Association</i></p> <p>→</p>	<p>Asosiasi unilateral direpresentasikan oleh ujung panah terbuka yang menghubungkan satu kelas ke kelas lain. Ini menunjukkan bahwa satu kelas mengetahui hubungannya dengan kelas lain.</p>
<p><i>Inheritance</i></p>  <pre> classDiagram class Animal { + age : Int + gender : String + isMammal() + mate() } class Duck { + beakColor : String = "yellow" + swim() + quack() } class Zebra { + is_wild : Boolean + run() } Animal < -- Duck Animal < -- Zebra </pre>	<p>Menunjukkan hubungan <i>child-parent</i> antara kelas-kelas. Kelas anak adalah suatu kelas yang dispesialisasi, merupakan sub-kelas dari kelas induk.</p>
<p><i>Realization</i></p>	<p>Satu kelas mengimplementasikan perilaku yang ditentukan oleh kelas lain.</p>

Simbol	Deskripsi
 <pre> classDiagram class Movable { <<interface>> +moveUp() void +moveDown() void +moveLeft() void +moveRight() void } class MoveablePoint { -x int -y int -xSpeed int -ySpeed int +MoveablePoint(x: int, y: int, xSpeed: int, ySpeed: int) +toString() string +moveUp() void +moveDown() void +moveLeft() void +moveRight() void } class MoveableCircle { -radius int -center MoveablePoint +MoveableCircle(x: int, y: int, radius: int, xSpeed: int, ySpeed: int) +toString() string +moveUp() void +moveDown() void +moveLeft() void +moveRight() void } Movable < -- MoveablePoint Movable < -- MoveableCircle MoveableCircle ..> MoveablePoint </pre>	
<p><i>Dependency</i></p> <p>▼</p>	<p>Satu kelas bergantung pada kelas lain.</p> <p>Panah berputus menunjukkan ini.</p>
<p><i>Aggregation</i></p> 	<p>Ini mewakili hubungan unilateral antara kelas. Satu kelas adalah bagian dari kelas lain. Dalam konteks ini, kelas <i>child</i> dan <i>parent</i> dapat berdiri secara independen.</p>
<p><i>Composition</i></p> 	<p>Ini merupakan bentuk agregasi di mana satu kelas bergantung pada kelas lain. Satu kelas adalah bagian dari yang lain. Dalam konteks ini, kelas <i>child</i> dan <i>parent</i> tidak dapat berdiri secara independen.</p>

2.2.8. Docker

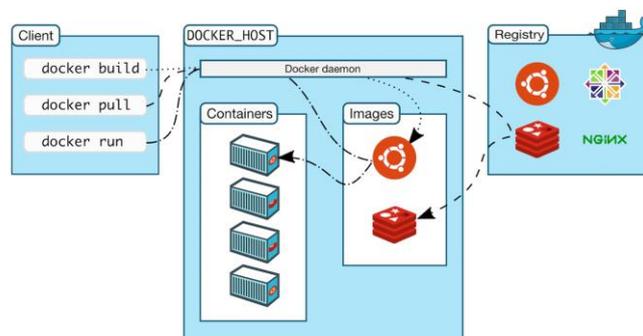
Docker adalah sebuah projek *open-source* yang menyediakan *platform* terbuka untuk *developer* maupun *sysadmin* untuk dapat membangun, mengemas, dan menjalankan aplikasi dimanapun didalam sebuah *container* [31].

Docker menggunakan arsitektur *client-server*. Di mana *client* dan *docker* berkomunikasi dengan *daemon docker*, yang melakukan suatu tindakan untuk membangun, menjalankan, dan mendistribusikan *container docker*. *Client docker* dan *daemon* dapat berjalan pada sistem yang sama. *Client docker* dan *daemon* berkomunikasi menggunakan *REST API*, melalui soket UNIX atau antarmuka jaringan [31]. Teknis kerja *docker* dapat dilihat pada gambar 2.2 sebagai berikut:



Gambar 2.2 Teknis Kerja Docker

Arsitektur *Docker* pada gambar 2.3 mempunyai beberapa komponen yaitu terdiri dari *Docker daemon*, *Docker client*, *Docker images*, *Docker container*, dan *Docker registry*. *Docker* menggunakan teknologi *client-server* untuk menghubungkan antara *Docker client* dan *Docker daemon* [31].



Gambar 2.3 Arsitektur Docker

Docker sendiri memiliki banyak komponen dalam menjalankan proses dan tugasnya. Masing - masing komponen pada *Docker* saling ketergantungan, sehingga jika ada komponen yang tidak tersedia, hal itu tentu saja akan mempengaruhi kinerja *Docker*.

Berikut merupakan komponen yang terdapat pada *Docker*:

1. *Docker Daemon* adalah sebuah *service* yang dijalankan di dalam *host* dalam *Operating System* (OS). Fungsinya adalah membangun, mendistribusikan, dan menjalankan *container Docker*. Pengguna tidak dapat langsung menggunakan *Docker Daemon*, akan tetapi untuk menggunakan *Docker Daemon* maka pengguna menggunakan *Docker Client* sebagai perantara atau CLI.
2. *Docker Client* adalah seperangkat perintah *command line* untuk mengoperasikan *Docker container*, misalnya membuat *container*, *start/stop container*, menghapus (*destroy*), dan sebagainya yang ada di *Docker Daemon*.
3. *Docker Images* adalah sebuah komponen *Docker* yang berupa *template* dasar yang bersifat *read-only*. *Template* ini sebenarnya adalah sebuah OS atau OS yang telah di-*install* berbagai aplikasi. *Docker Images* berfungsi untuk membuat *Docker container*, dengan hanya satu *Docker Image* dapat membuat banyak *Docker container*.
4. *Docker Container* merupakan sebuah *image* yang dapat dikemas dan bersifat *read-write*, *container* berjalan di atas *image*. *Docker container* juga bisa dikatakan sebagai sebuah *folder*, di mana *Docker container* ini dibuat dengan menggunakan *Docker container*. Setiap *Docker container* disimpan maka akan terbentuk layer baru tepat di atas *Docker image* atau *base image* di atasnya.
5. *Docker Registry* merupakan *repository* distribusi kumpulan *Docker images* yang bersifat *private* maupun *public* yang dapat diakses melalui *Docker Hub*.