BAB II

LANDASAN TEORI

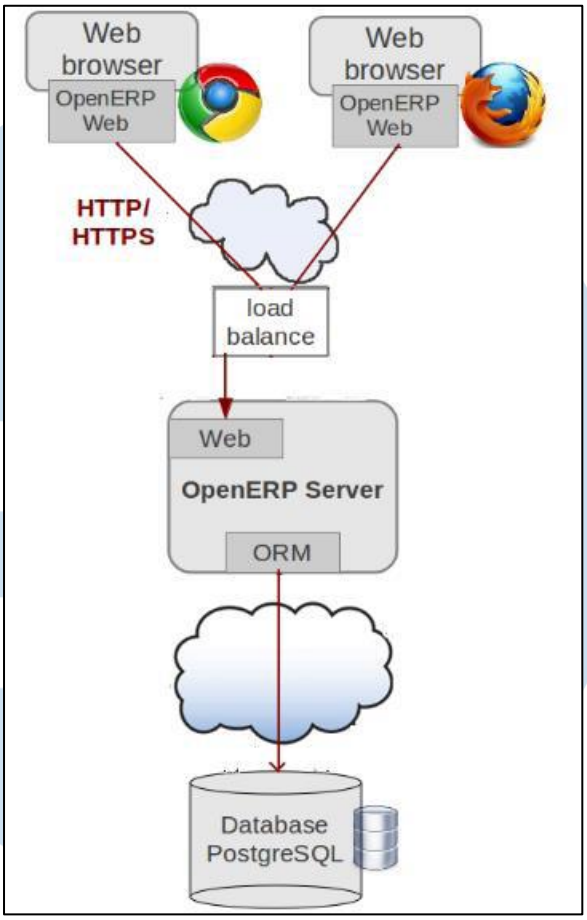
2.1 Tinjauan Teori

2.1.1 Odoo

Di era digital ini banyak perusahaan sudah mulai beralih menggunakan *software ERP* untuk mendukung proses bisnisnya. *Software* ini dinilai dapat meningkatkan efisiensi pengelolaan sumber daya perusahaan karena memiliki kemampuan untuk mengintegrasikan departemen-departemen yang terdapat dalam perusahaan. Salah satu *software ERP* yang tersedia saat ini adalah Odoo.

Odoo adalah sebuah *business management tools* yang berbasiskan pada *website* dan bersifat *open source.* Sistem ini dibuat dengan konsep sistem modul yang memungkinkannya untuk memenuhi kebutuhan kecil hingga kebutuhan besar berbagai organisasi maupun perusahaan. Tujuan utama Odoo adalah menjadi sistem *Enterprise Resource Planning* yang dimaksudkan untuk mengelola proses bisnis dan melacak informasi tentang mereka [2]. Proses bisnis yang dicakup antara lain penjualan, pembelian, stok, akuntansi, dan manajemen proyek. Odoo memiliki beberapa modul yang dapat disesuaikan dengan kebutuhan bisnis seperti *Point of Sales, Manufacturing, Transportation, Inventory, Human Resources, Purchase,* dan lain-lain. Modul-modul tersebut merupakan aplikasi tunggal yang dapat "dicopot pasang" sehingga dapat menyesuaikan kebutuhan perusahaan.

Secara sederhana odoo diciptakan berdasarkan artitektur MVC (*Model-view-controller*) yang terdiri dari 3 *layers* yaitu: *database, server,* dan *client* [3]. Gambar 2.1 berikut menunjukkan arsitektur yang terdapat pada Odoo:



Gambar 2.1. Arsitektur Odoo

Selain itu, berikut merupakan beberapa kelebihan dari Odoo dibandingkan dengan *software ERP* lainnya [4]:

a. Lebih terjangkau

Dapat dikatakan bahwa Odoo cenderung lebih terjangkau dibandingkan dengan *software ERP* lainnya. Harga modul yang ditawarkan oleh Odoo bervariatif sehingga memberikan kebebasan pengguna dalam memilih modul yang diinginkan. Selain itu Odoo juga menyediakan beberapa modul yang dapat digunakan secara gratis oleh penggunanya. Hal ini tentu dapat meringankan beban keuangan perusahaan karena perusahaan bebas untuk memilih modul yang diinginkan.

b. Bersifat fleksibel

Dikarenakan Odoo bersifat *open source* maka hal tersebut memungkinkan pengguna untuk dapat memodifikasi modul-modul yang terdapat di dalamnya. Modifikasi dapat dilakukan tanpa adanya batasan dan dapat disesuaikan dengan kebutuhan perusahaan. Selain itu pengguna juga dapat meng-*upgrade* Odoo ke versi yang lebih baru kapanpun ketika sudah tersedia.

c. Mudah untuk di-*upgrade*

Manfaat lain yang didapatkan dari sistem yang bersifat *open source* adalah kemudahan untuk melakukan *upgrade* Odoo. Proses *upgrade* dapat dilakukan tanpa mengubah *framework*. Selain itu, proses *upgrade* juga tidak mengganggu *production server* yang sedang berjalan sehingga tidak menyebabkan *server down* dalam jangka panjang.

2.1.2 Python

Salah satu bahasa pemrograman yang banyak digunakan saat ini adalah Python. Python merupakan bahasa pemrograman level tinggi dan bersifat *object-oriented* yang diciptakan oleh Guido van Rossum. Python memungkinkan *programmer* untuk mengekspresikan konsep dalam baris kode yang lebih sedikit dari pada bahasa pemrograman lainnya seperti C# [5]. Python juga disebut sebagai bahsa pemrograman yang *user-friendly* dan sangat ramah terhadap pemula. Popularitasnya bahkan menghalahkan bahasa pemrograman Java dikarenakan penggunaannya yang mudah. Berikut merupakan beberapa kelebihan dari bahasa pemrograman Python [5]:

a. Bahasa pemrograman yang sederhana

Python merupakan salah satu bahasa pemrograman yang banyak digunakan serta sederhana. Python memiliki tampilan yang sederhana sehingga lebih mudah untuk membaca dan menulis

program Python dibandingkan dengan Java, C++, C#, dan lain-lain.

b. Bersifat *portable*

Python bersifat *portable* karena dapat digunakan pada sistem operasi lain seperti Mac OS, Windows, Linux, UNIX, dan lain-lain tanpa adanya perubahan sintaks.

c. Memiliki *libraries* yang dapat digunakan untuk melakukan tugas

Python memiliki sejumlah *libraries* yang dapat digunakan untuk mempermudah *programmer* dalam melakukan tugasnya. *Programmer* tidak perlu menuliskan seluruh *code*-nya sendiri, cukup dengan melakukan *import library* yang dibutuhkan.

d. Dapat digunakan untuk banyak pemrograman

Cakupan pemrograman Python sangat luas sehingga dapat digunakan untuk beberapa kebutuhan seperti *system programming, graphical user interface (GUI), game programming, database programming, image processing, robotic programming, artificial intelligence, website,* dan sebagainya.

Selain kelebihan, Python juga memiliki beberapa kekurangan yaitu [5]:

a. Perawatan yang tidak mudah

Python merupakan bahasa pemrograman yang mudah digunakan namun semakin banyak sintaks yang digunakan maka semakin sulit juga untuk melakukan perawatan apabila suatu saat terjadi kesalahan.

b. Lambat

Python tergolong lambat karena sifatnya yang terlalu fleksibel dan terlalu banyak definisi yang akhirnya memperlambat kinerja Python

2.1.3 eXtensible Markup Language (XML)

eXtensible Markup Language atau disingkat XML adalah bahasa *meta* yang dapat digunakan untuk menyimpan informasi atau data.

Struktur XML terdiri dari *logical* dan *physical structure. Logical structure* merupakan *template* yang merinci elemen yang akan dimasukkan ke dalam dokumen dan urutannya. *Physical structure* berisi data aktual yang dignakan dalam sebuah dokumen. Pada dasarnya XML memiliki persamaan dengan *HyperText Markup Language* (HTML) namun terdapat beberapa perbedaan mendasar yaitu [6]:

a. Pemisahan bentuk dan konten

   HTML sebagian besar terdiri dari *tag* yang mendefinisikan tampilan teks sedangkan XML mendefinisikan struktur dan konten data dengan tampilan aktual yang ditentkan oleh aplikasi tertentu atau *stylesheet* terkait.

b. XML dapat diperluas

   Tag dapat ditentukan oleh individu atau organisasi untuk beberapa aplikasi tertentu sedangkan kumpulan *tag* standar HTML ditentukan oleh *World Wide Web Consortium* (W3C).

Pada umumnya XML dapat digunakan pada beberapa skenario berikut [7]:

a. Pemrograman berbasis dokumen

   Dapat digunakan sebagai wadah untuk membangun antarmuka aplikasi.

b. Archiving

   Dapat digunakan sebagai tempat dimana dokumen disimpan / diarsipkan.

c. Binding

   Merupakan sebuah skema yang dapat mendefinisikan struktur data XML dan secara otomatis dapat digunakan untuk memproses data dari suatu aplikasi.

Selain itu, XML juga memiliki kelebihan dan kekurangan. Berikut adalah beberapa kelebihan dari XML yaitu [6]:

a. XML memisahkan data dari HTML

Jika perlu menampilkan data dinamis dalam dokumen HTML, dibutuhkan banyak pekerjaan untuk mengedit HTML setiap kali data berubah. Dengan XML, data dapat disimpan dalam file XML terpisah sehingga *developer* dapat berkonsentrasi menggunakan HTML untuk tata letak dan tampilan serta memastikan bahwa perubahan pada data mendasar tidak memerlukan perubahan apa pun pada HTML.

b. XML menyerderhanakan *data sharing*

Pada umumnya, sistem komputer dan *database* berisi data dalam format yang tidak kompatibel. Data XML disimpan dalam format teks biasa dengan menyediakan cara yang tidak bergantung pada *software* dan *hardware* untuk menyimpan data sehingga mempermudah dalam membuat data.

c. XML menyederhanakan *data transport*

Salah satu tantangan yang memakan waktu bagi *developer* adalah pertukaran data antara sistem yang tidak kompatibel melalui internet. Pertukaran data XML dapat mengurangi kerumitan ini karena data dapat dibaca oleh berbagai aplikasi yang tidak kompatibel.

d. XML menyederhanakan perubahan *platform*

Upgrade ke sistem baru selalu memakan waktu, bahkan sejumlah besar data harus dikonversi dan data yang tidak kompatibel seringkali hilang. Data XML disimpan dalam format teks yang memudahkan untuk memperluas atau meningkatkan ke sistem baru tanpa harus kehilangan data.

e. XML dapat digunakan untuk membuat bahasa internet baru

Bahasa internet baru yang dapat dibuat dengan XML seperti XHTML, WSDL (untuk mendeskripsikan layanan *web* yang tersedia), WAP dan WML (bahasa *markup* untuk telepon genggam), RSS (bahasa untuk umpan berita), RDF dan OWL

(untuk mendeskripsikan sumber daya dan ontology), dan SMIL (untuk mendeskripsikan multimedia dalam *web*).

Berikut adalah beberapa kelemahan yang dimiliki XML antara lain [6]:

a. Kurangnya pengolahan aplikasi XML
   XML membutuhkan sistem pemrosesan aplikasi sedangkan saat ini belum ada *browser* yang dapat membaca XML. Untuk dapat dibaca dalam *browser*, XML masih bergantung pada HTML dan tidak terlepas darinya.
b. XML merupakan bahasa *verbose*
   XML dikenal sebagai bahasa *verbose* yang berarti sangat beragantung pada siapa penulisnya. Bahasa *verbose* dapat menimbulkan masalah bagi pengguna lain.
c. XML dan Unicode
   Ketika XML terikat erat dengan Unicode, Unicode mengubah atribut XML yang mungkin dapat menghasilkan file yang berbeda dengan aslinya.

2.1.4 PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional kelas perusahaan yang setara dengan basis data berpemilik terbaik seperti Oracle, Microsoft, SQL Server, dan IBM DB2 [9]. Fungsi utama dari PostgreSQL adalah sebagai tempat penyimpanan serta pengelolaan data melalui perintah atau *SQL Query*. PostgreSQL dapat juga mendukung beberapa bahasa pemrograman seperti Python, PHP, Java, SQL, .NET, C/C++, Ruby, dan lain-lain [9]. PostgreSQL juga memiliki beberapa kelebihan antara lain [8]:

a. *Maintenance* yang mudah
   Sistem ini dapat di-*maintenance* dengan mudah baik *developer* profesional maupun pemula sekalipun.
b. Aman dan tidak memerlukan biaya

PostgreSQL adalah sistem berbasis *open source* dan tidak memerlukan lisensi khusus sehingga *user* tidak perlu mengeluarkan uang sepeser pun alias gratis.

c. Dapat direplikasi

PostgreSQL juga memungkinkan untuk direplikasi secara gratis dan tidak memerlukan lisensi yang dapat membantu dalam melancarkan kerja sistem.

Selain kelebihan, PostgreSQL juga memiliki beberapa kekurangan yaitu [8]:

a. File yang berantakan

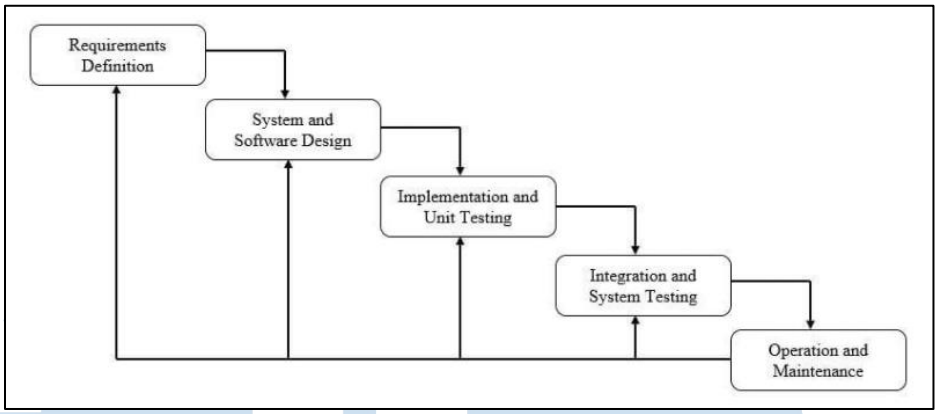
Physical database file yang terdapat pada sistem ini cukup berantakan sehingga akan merepotkan *developer* dalam mengolah *database.*

b. Tidak mendukung *selectable stored procedure*

PostgreSQL tidak mendukung *selectable stored procedure* sehingga akan meningkatkan lalu lintas jaringan dan memperlambat waktu respon *database.*

2.1.5 Software Development Life Cycle (SDLC)

Untuk dapat menciptakan *software* yang dapat memenuhi kebutuhan *user*, diperlukan adanya tahapan-tahapan yang dikenal dengan istilah *Software Development Life Cycle (SDLC).* SDLC atau *Software Development Life Cycle* merupakan suatu proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem perangkat lunak sebelumnya [10]. Gambar 2.2 berikut ini merupakan alur dari metodologi SDLC *Waterfall*:



Gambar 2.2. Metodologi SDLC *Waterfall*

SDLC *Waterfall* merupakan metodologi yang umum digunakan dalam pengembangan suatu sistem. Metodologi ini memiliki beberapa kelebihan serta kekurangan. Berikut merupakan beberapa kelebihan yang dimilikinya antara lain [11]:

a. Tahapan proses pengembangannya mudah diaplikasikan, tetap (pasti, dan prosesnya teraktur.
b. Cocok digunakan untuk produk *software* yang sudah jelas kebutuhannya di awal, sehingga minim kesalahannya.
c. Metode ini biasanya menghasilkan kualitas *software* yang baik.
d. Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.

Selain kelebihan, metode ini juga memiliki beberapa kekurangan antara lain [10]:

a. Proyek yang sebenarnya jarang mengikuti alur sekuensial seperti diusulkan, sehingga perubahan yang terjadi dapat menyebabkan hasil yang sudah didapatkan tim pengembang harus diubah kembali sering menyebabkan masalah baru.
b. Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses.

c. Sulit untuk mengalami perubahan kebutuhan yang diinginkan oleh pelanggan.
d. Adanya waktu kosong (menganggur) bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

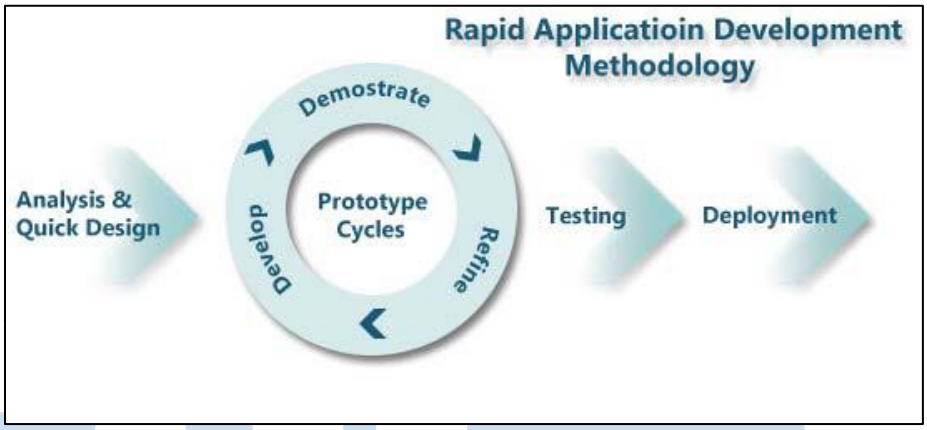
2.1.6 Rapid Application Development (RAD)

Selain SDLC, terdapat juga metodologi lain yang dinamai dengan RAD *(Rapid Application Development)*. RAD adalah kombinasi dari berbagai teknik terstruktur dengan teknik *prototyping* dan pengembangan aplikasi bersama teknik untuk mempercepat pengembangan sistem [12]. Metodologi ini bertujuan agar pengembangan sistem dapat diselesaikan dengan waktu yang relatif cepat. Metodologi dapat dicapai dengan menerapkan beberapa poin penting antara lain [12]:

a. Konstruksi berbasis komponen.
b. Penekanan pada penggunaan kembali komponen perangkat lunak yang ada.
c. Pembuatan kode program yang dilakukan secara otomatis maupun semi-otomatis.
d. Pembagian tugas yang terbagi menjadi beberapa tim.

Dengan menggunakan metodologi RAD, diperkirakan perangkat lunak lengkap akan selesai dalam 60 sampai 90 hari yang prosesnya dapat dilihat pada gambar 2.3 berikut ini:

UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 2.3. Metodologi RAD

Walaupun metodologi RAD dinilai lebih cepat serta efisien namun metodologi ini juga memiliki kelebihan dan kekurangan. Berikut merupakan kelebihan yang dimiliki RAD [12]:

a. Penghematan waktu dalam fase proyek secara keseluruhan dapat tercapai.
b. RAD mengurangi semua kebutuhan yang terkait dengan biaya proyek dan sumber daya manusia.
c. RAD membantu mengembangkan aplikasi yang foks pada proyek waktu penyelesaian.
d. Perubahan desain sistem bisa lebih berpengaruh cepat dari pada pendekatan SDLC tradisional.
e. Sudut pandang pengguna disajikan dalam sistem akhir baik melalui fungsi sistem atau antarmuka pengguna.
f. RAD dapat menciptakan rasa kepemilikan yang kuat di antara semua pemangku kepentingan proyek.

Berikut adalah beberapa kelemahan yang terdapat pada RAD yaitu [12]:

a. Dengan metode RAD, *analyst* mencoba untuk mempercepat proyek dengan tergesa-gesa.
b. Kelemahan terkait waktu dan perhatian terhadap detail.

c. RAD mempersulit *programmer* yang tidak berpengalaman untuk menggunakan *toolkit*. *Programmer* dan *analyst* harus menguasai keterampilan baru dengan cepat sementara pada saat yang sama mereka harus bekerja dalam mengembangkan sistem.

2.1.7 PyCharm

Dalam perancangan Odoo, diperlukan adanya suatu *environment* atau tempat dimana *developer* melakukan proses *coding* Python. Salah satu *environment* yang digunakan adalah PyCharm. PyCharm adalah *Python Integrated Development Environment* (IDE) khusus yang menyediakan berbagai alat penting untuk pengembang Python, terintegrasi erat untuk menciptakan lingkungan yang nyaman bagi pengembang Python, web, dan ilmu data yang produktif [13]. Berikut merupakan beberapa fitur yang ditawarkan oleh PyCharm [14]:

a. Memeriksa *code* untuk mencari *error*, menyediakan fasilitas penyelesaian *code*, dan melakukan audit kualitas PEP8.
b. Memfaktorkan ulang *code* dengan cara yang cerdas.
c. Menyediakan terminal SSH terintegrasi serta integrasi Docker.
d. Terintegrasi dengan ShellCheck untuk memeriksa file *bash*.
e. Menawarkan dukungan untuk kerangka kerja seperti Django dan Google App Engine.
f. Mendukung banyak *packages* seperti Numpy dan Scikit-learn.

PyCharm memiliki beberapa kelebihan antara lain [14]:

a. Dapat diunduh secara gratis.
b. Memiliki tampilan antarmuka pengguna grafis dan mudah digunakan.

Selain kelebihan, PyCharm memiliki beberapa kekurangan yaitu [14]:

a. *Loading time* yang tergolong lambat.
b. Bersifat memori intensif.

2.1.8 DBeaver

Untuk mengelola *database* yang terdapat pada Odoo maka diperlukan adanya suatu *software* yang mampu mengelola *database* tersebut. DBeaver merupakan alat administrasi basis data grafis bersifat *open source* yang tidak terbatas untuk *database designer* dan *administrator* [15]. *Software* ini mempermudah *developer* dalam mengelola pergerakan data yang terjadi pada suatu aplikasi. DBeaver dapat mendukung beberapa *database* pada umumnya seperti MySQL, PostgreSQL, MariaDB, SQLite, Oracle, DB2, SQL Server, Sybase, Microsoft Access, Teradata, Firebird, Derby dan lain-lain [15].

2.2 Penelitian Terdahulu

Tabel 2.1. Penelitian Terdahulu 1

Penulis	A. Terminanto, H.A. Swantoro, A.N. Hidayanto
Nama Jurnal	IOP Conference Series: Materials Science and Engineering
Judul Artikel	Configurations and implementation of payroll system using open source erp: A case study of Koperasi PT Sri
Permasalahan	Tidak terintegrasinya sistem gaji karyawan pada Koperasi Karyawan PT. Suryaraya Rubberindo Industries (KKPS) yang menyebabkan alur informasi dan data antar departemen menjadi tidak akurat. Penghitungan dan pengelolaan sistem gaji masih dilakukan secara manual tanpa menggunakan *database* sehingga *output* tidak dapat diperoleh secara *real-time*
Metode	• *ASAP (Accelerate SAP)*
Kesimpulan	Implementasi modul *Human Resources* dan *Payroll* menggunakan sistem Odoo membantu perusahaan dalam

	mengelola data karyawan serta membantu dalam mengurangi kesalahan dalam penggajian karyawan. Sistem yang telah menggunakan *database* memudahkan karyawan dalam mendapatkan informasi secara *real-time* sehingga dalam memperlancar proses bisnis perusahaan

Tabel 2.2. Penelitian Terdahulu 2

Penulis	Hennry Syahreza Arifin, Ari Yanuar Ridwan, Muhari Saputra
Nama Jurnal	2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA)
Judul Artikel	Design of Green ERP System Reverse Logistic Module Based on Odoo in Leather Tanning Industry
Permasalahan	Proses *input data* produk pada PT. ELCO Indonesia yang masih dilakukan secara *manual* sehingga menyebabkan data produk pada *inventory* tidak sinkron antara satu dengan yang lain
Metode	• *SAP Activate*
Kesimpulan	Terciptanya modul *Reverse Logistic* dengan memanfaatkan sistem Odoo mampu mencatat setiap pergerakan produk pada *Inventory* dan tersedianya *report* yang membantu perusahaan dalam melihat jumlah *Inventory* yang dimilikinya

Tabel 2.3. Penelitian Terdahulu 3

Penulis	Kurnia Dwi, Avon Budiyono, R. Wahjoe Witjaksono
Nama Jurnal	e-Proceeding of Engineering: Vol.8, No.5 Oktober 2021
Judul Artikel	Implementasi Sistem ERP Proses Pengadaan Pada Restaurant De Braga Hotel Dengan Metode Rapid

	Application Development Berbasis Odoo e-Proceeding of Engineering: Vol.8, No.5 Oktober 2021
Permasalahan	Proses pengadaan bahan baku pada Restoran De Braga Hotel yang masih *semi manual* dan belum terintegrasi dengan proses produksi, penjualan, dan pembayaran
Metode	• *RAD (Rapid Application Development)*
Kesimpulan	Implementasi sistem Odoo dengan menggunakan beberapa modul Odoo seperti *purchasing, manufacturing, inventory, sales,* dan *point of sales* pada Restoran De Braga Hotel yang mampu mengintegrasikan proses pengadaan bahan baku dengan proses produksi, penjualan, dan pembayaran

Tabel 2.4. Penelitian Terdahulu 4

Penulis	Risa Mutiara, Yuli Adam Prasetyo, Muhammad Azani
Nama Jurnal	e-Proceeding of Engineering: Vol.4, No.2 Agustus 2017
Judul Artikel	Implementasi Aplikasi Enterprise Resource Planning Odoo Modul Sales Menggunakan Metode Rapid Application Development di UD. Permatasari
Permasalahan	Belum terintegrasinya proses bisnis *sales order* dari *salesman* dengan sistem penjualan di UD. Permatasari karena proses penjualan masih dilakukan secara *manual* atau tanpa menggunakan sistem
Metode	• *RAD (Rapid Application Development)*
Kesimpulan	Tersedianya modul *sales* Odoo yang telah dikonfigurasi dengan kebutuhan *sales* sehingga pengelolaan persediaan menjadi semakin efektif, efisien, dan selaras dengan proses bisnis perusahaan

Tabel 2.5. Penelitian Terdahulu 5

Penulis	Yusup Firmansyah, Rahmat Hidayat, Ulinnuha Latifa

Nama Jurnal	Research of Science and Informatic v7.11
Judul Artikel	Perancangan Aplikasi Sistem Parkir Otomatis Menggunakan ERP Odoo Berbasis Internet of Things
Permasalahan	Sistem parkir UNSIKA (Universitas Singaperbangsa Karawang) yang kurang baik khususnya pada bidang keamanan karena sering terjadi tindak pencurian kendaraan bermotor
Metode	• *Waterfall Development Model*
Kesimpulan	Penggunaan aplikasi Odoo berbasis *web* yang dinilai efisien dan fleksibel karena dapat diakses dimanapun melalui *web browser*. Sistem Odoo yang sudah menerapkan konsep IoT (*Internet of Things*) sehingga memudahkan memonitor kondisi parkiran serta mampu mengelola data parkiran dengan memanfaatkan teknologi *barcode* dan *fingerprint sensor* yang dapat mengurangi resiko tindak pencurian

Berdasarkan beberapa penelitian terdahulu, penggunaan *software* Odoo terbukti telah menjawab beberapa permasalahan yang dialami oleh pelaku usaha dan instansi pendidikan sekalipun. Permasalahan seperti sistem yang belum terintegrasi, proses *input* data secara *manual*, hingga pencurian kendaraan bermotor dapat teratasi dengan menggunakan modul-modul seperti *sales, purchasing, inventory, human resources*, dan lain-lain.

Perbedaan antara penelitian terdahulu dengan penelitian ini terletak pada modul Odoo yang digunakan. Penelitian ini akan berfokus pada perancangan modul *employee shift* yang akan digunakan oleh *manager* untuk mengatur jadwal *shift* karyawan Kompas Gramedia. Metode penelitian yang digunakan adalah SDLC *Waterfall* dengan menggunakan bahasa pemrograman Python, XML, dan PostgreSQL.