

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Beberapa penelitian yang terkait dengan penelitian ini, dalam bentuk tabel 2.1 berikut ini :

Tabel 2. 1 Data Perbandingan Penelitian

Penulis	Judul	Metode / Teknologi	Interface
<i>Andre Nofriandi</i> (2015)	Penerapan Enterprise Resource Planning (ERP) pada Sistem Pembelian, Penjualan dan Inventory Barang di Mini Market 7 Putra	FrontAccounting, Bahasa Pemrograman php	Web Based
Dhani Adiatma Rimen, Ricky Akbar (2016)	Penerapan Enterprise Resource Planning (ERP) Dan Pengelolaan Hubungan Dengan Pelanggan Untuk Sistem Informasi Penjualan Pada Toko Soviah	Odoo versi 8, Bahasa Pemrograman Python	Web Based
Yusuf Eko Nurcahyo (2012)	Penerapan Enterprise Resource Planning (ERP) Adempiere Pada Perusahaan PT Global Agrotek Nusantara	Adempiere, Bahasa Pemrograman JAVA	Dekstop
Usulan	Aplikasi ERP pada	Odoo Versi 9,	Web Based

Penulis (2017)	Distributor Bahan Bangunan PT Alisan Catur Adhirajasa	Bahasa Pemrograman Python	
-------------------	---	---------------------------------	--

Pada Penelitian sebelumnya (Dhani Adiatma Rimen , Ricky Akbar (2016)) di analisa bahwa ERP Odoo merupakan perangkat lunak ERP open source yang sangat baik dalam mengelola proses bisnis Purchase Requisition. maka memilih aplikasi Odoo untuk diterapkan di Toko Soviah dengan beberapa alasan antara lain :

1. Aplikasi Odoo merupakan yang paling efisien dibandingkan Openbravo dan ERP Next
2. Aplikasi Odoo dapat mengelola proses bisnis pada perusahaan dengan baik karena fitur yang dimiliki sesuai dengan kebutuhan.
3. Untuk pengelolaan Customer Relationship Management aplikasi Odoo sangat praktis dan mudah digunakan dibandingkan aplikasi lain. Dibuktikan dengan pemakaian aplikasi Odoo yang userfriendly.

Pada Tahun 2015 (Andre Nofriandi, Husnil Kamil. 2015) hasil implementasi ERP Adempiere modul master data management, Quote to Invoice (sales management), Requisition to Invoice (Requisition dan Purchasing), Material Management, Performance Analysis dan manufacturing management, menunjukkan modul yang saling terkait sebagai relasi satu sama lain dalam memperoleh data yaitu manufacturing management, procurement management, sales management dan accounting.

Pada penelitian selanjutnya, Berdasarkan hasil pengujian yang telah dilakukan, menunjukkan bahwa aplikasi ERP FrontAccounting yang diterapkan dapat berjalan dengan baik sesuai dengan laporan yang ada di mini market 7 Putra. Pengujian ini dapat berjalan dengan baik pada sistem pembelian, penjualan dan inventory barang di 7 Putra. Secara fungsional aplikasi yang diterapkan sudah dapat menghasilkan output atau keluaran yang diharapkan.

Kemudian ditahun 2017 mengusulkan pembuatan ERP dengan menggunakan Odoo versi 9 yang lebih *modern* dari penelitian sebelumnya di PT Alisan Catur Adhirajasa yang lebih fokus pada fitur pembelian, penjualan, persediaan dan accounting untuk memudahkan karyawan dalam melakukan pencatatan transaksi yang ada.

2.2. Dasar Teori

2.2.1. Akuntansi

Akuntansi adalah suatu sistem informasi yang mengidentifikasi, mencatat, dan mengkomunikasikan peristiwa-peristiwa ekonomi dari suatu organisasi kepada para pengguna yang berkepentingan

- a. Mengidentifikasi peristiwa-peristiwa ekonomi akan melibatkan pemilihan aktivitas-aktivitas ekonomi yang relevan bagi suatu organisasi tertentu.

- b. Setelah teridentifikasi, peristiwa-peristiwa ekonomi tersebut kemudian dicatat untuk menjadi alur aktivitas keuangan perusahaan. Pencatatan terdiri atas pembuatan jurnal peristiwa-peristiwa secara sistematis dan kronologis.

Aktivitas pengidentifikasian dan pencatatan tidak akan banyak memberikan manfaat kecuali jika informasi tersebut dikomunikasikan kepada pengguna-pengguna yang berkepentingan. Informasi keuangan akan disampaikan melalui laporan-laporan akuntansi yang disebut laporan keuangan (Kieso, 2007).

2.2.2. Kode Akun (*Chart of Account*)

Kode rekening akuntansi dikelola oleh Direktorat Jenderal Perbendaharaan, Departemen Keuangan. Kode akun pemerintah pusat RI disebut Bagan Perkiraan Standar adalah daftar buku besar yang ditetapkan dan disusun secara sistematis untuk memudahkan perencanaan, pelaksanaan, anggaran, serta pertanggung jawaban dan pelaporan keuangan yang wajib melaksanakan Standar Akuntansi. (Sariono, 2007)

Pengodean rekening secara kelompok dilakukan dengan memberi kode pada setiap akun sehingga berbeda antara satu kelompok akun dengan kelompok akun yang lain. Angka pertama menunjukkan kode akun dan angka kedua menunjukkan nomor urut akun termasuk kelompok akun tersebut. (Indriani, 2012)

Tabel 2. 2 Contoh Kode Rekening Akuntansi

Kelompok 1 Harta (Aktiva)	Kelompok 2 Utang	Kelompok 3 Modal	Kelompok 4 Pendapatan	Kelompok 5 Beban-beban
1.1 Aktiva Lancar 111 Kas 112 Piutang dagang 113 Perlengkapan 114 Peralatan 115 Persediaan barang	2.1 Utang lancar 211 Utang dagang 212 Utang pajak	311 Modal 312 Prive	411 Penjualan 412 Pendapatan sewa	5.1 Beban Oprasi 511 Beban gaji 512 Beban sewa 513 Beban Iklan 514 Beban Perlengkapan

Sumber: (Sariono, 2007)

2.2.3. Jurnal

Jurnal disebut dengan buku pencatatan awal. Jurnal yang paling mendasar yaitu jurnal umum (*general jurnal*). Biasanya jurnal umum memiliki tempat untuk mencantumkan tanggal, nama akun, dan uraiannya, referensi, dan dua kolom jumlah (Kieso, 2005).

Tabel 2. 3 Jurnal Metode Persediaan Perpetual

No.	Transaksi	Metode Perpetual		
		Akun/Uraian	Debet	Kredit
1.	Pembelian baranga dagangan secara tunai	Persediaan barang dagangan Kas	Rp xxx	Rp xxx
2.	Pembelian barang dagangan secara kredit	Persediaan barang dagangan Utang dagang	Rp xxx	Rp xxx
3.	Penjualan barang dagangan tunai	Kas Persediaan barang dagangan	Rp xxx	Rp xxx
4.	Penjualan barang dagangan secara kredit	Piutang dagang Persediaan barang dagangan	Rp xxx	Rp xxx
5.	Pembayaran beban angkut pembelian	Harga perolehan Kas	Rp xxx	Rp xxx
6.	Pembayaran beban angkut penjualan	Beban angkut penjualan Kas	Rp xxx	Rp xxx
7.	Pengiriman kembali sebagian barang yang dibeli tunai	Kas Persediaan barang dagangan	Rp xxx	Rp xxx
8.	Pengiriman kembali sebagian barang dagangan yang dibeli kredit	Utang dagang Persediaan barang dagangan	Rp xxx	Rp xxx
9.	Penerimaan kembali barang yang dijual secara tunai	Persediaan barang dagangan Kas	Rp xxx	Rp xxx
10.	Penerimaan kembali barang yang dijual secara redit	Persediaan barang dagangan Piutang dagang	Rp xxx	Rp xxx
11.	Pembayaran utang tanpa potongan	Utang dagang Kas	Rp xxx	Rp xxx
12.	Pembayaran utang dagang dengan potongan	Utang dagang Potongan pembelian Kas	Rp xxx	Rp xxx Rp xxx
13.	Penerimaan tagihan tanpa potongan	Kas Piutang dagang	Rp xxx	Rp xxx
14.	Penerimaan tagihan dengan potongan	Kas Potongan penjualan Piutang dagang	Rp xxx Rp xxx	Rp xxx

Sumber : (Kieso, 2005)

2.2.4. Persediaan

Menurut Pernyataan Standar Akuntansi Keuangan (PSAK) No. 14 persediaan merupakan aktiva tersedia untuk dijual dalam kegiatan usaha normal dalam proses produksi, dalam perjalanan, dalam bentuk bahan atau perlengkapan (*supplies*) untuk digunakan dalam proses produksi atau pemberian jasa.

2.2.5. Metode *First In First Out* (FIFO)

Metode ini mengansumsikan bahwa barang pertama yang dibeli adalah barang pertama yang digunakan atau di jual, karena itu persediaan yang tersisa merupakan barang yang di beli paling terakhir.

Metode *First In First Out* (FIFO) merupakan harga pokok barang yang pertama kali dibeli akan merupakan barang yang dijual pertama kali. Di dalam metode ini biaya persediaan yang paling awal yang ada terlebih dahulu dibebankan sebagai harga pokok penjualan. Dengan demikian barang yang ada dalam persediaan dianggap berasal dari pembelian sebelumnya dianggap telah dijual atau dikeluarkan. (Yusuf, 2005)

2.2.6. Sistem

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Model dasar dari sebuah sistem adalah masukan, pengolahan dan pengeluaran. Fungsi pengolan informasi sering membutuhkan data yang telah dikumpulkan dan diolah dalam waktu periode sebelumnya. Oleh

karena itu, dalam model sistem informasi ditambahkan pula media penyimpanan data. Maka fungsi pengolahan informasi bukan lagi mengubah data menjadi informasi semata, tetapi juga menyimpan data untuk penggunaan lanjutan. (Jogianto, 1999)

2.2.7. Sistem ERP

ERP adalah *software* infrastruktur yang menghubungkan aplikasi internal perusahaan dan mendukung proses bisnis eksternal perusahaan. Aplikasi ERP adalah modular, dan modul-modul tersebut diintegrasikan untuk memperluas kapabilitas sistem. (Turban & Volonino, 2010). Lebih lanjut, bahwa ERP mengambil bisnis proses dari keseluruhan organisasi untuk mengintegrasikan perencanaan, manajemen, dan kegunaan dari seluruh sumber daya organisasi, memiliki *software* platform dan *database* yang sama. Menurut Turban & Volonino ERP adalah *software* infrastruktur yang menghubungkan aplikasi internal perusahaan dan mendukung proses bisnis eksternal perusahaan. Aplikasi ERP adalah modular, dan modul-modul tersebut diintegrasikan untuk memperluas kapabilitas sistem.

Lebih lanjut, Rainer & Cigielski menjelaskan bahwa ERP mengambil bisnis proses dari keseluruhan organisasi untuk mengintegrasikan perencanaan, manajemen, dan kegunaan dari seluruh sumber daya organisasi, memiliki *software platform* dan *database* yang sama.

2.2.8. Odoo / Open Erp

Berdasarkan Odoo, OpenERP atau sekarang lebih dikenal dengan Odoo adalah sebuah sistem *software* manajemen berbasis python yang memiliki

berbagai aplikasi bisnis untuk memenuhi segala kebutuhan bisnis. OpenERP ditemukan oleh Fabien Pinckaers dari 2002 dan baru di tahun 2005 resmi diluncurkan dengan format sebagai *open source software*. OpenERP telah tersedia dalam 18 bahasa dan memiliki partner serta kontributor dari seluruh dunia. Sekitar 1500 developer telah berpartisipasi dalam proyek pengembangan sistem OpenERP dan sekitar 500 lebih perusahaan telah menjadi *official partner* dari OpenERP. Hingga saat ini terdapat lebih dari 4000 aplikasi yang bisa diterapkan pada OpenERP. Aplikasi tersebut dikelompokkan menjadi 6 grup besar:

1. *Front-end apps: website builder, blog, e-commerce*
2. *Sales management apps: CRM, point of sales, quotation builder*
3. *Business operations apps: project management, inventory, manufacturing, accounting and purchase*
4. *Marketing apps: mass mailing, lead automation, events, survey, forum, live chat*
5. *Human Resources apps: employee directory, enterprise social network, leaves management, timesheet, fleet management*
6. *Productivity apps: business intelligence, instant messaging, notes*

OpenERP dapat diaplikasikan pada berbagai macam sektor, seperti sektor perdagangan, tekstil, agrikultural, dan lainnya. Perbedaan tersebut menggambarkan tingkat fleksibilitas OpenERP yang sangat tinggi sehingga dapat menjangkau seluruh jenis perusahaan yang ada. OpenERP dibangun dengan menggunakan arsitektur yang modular dan teknologi-teknologi yang bersifat *open source*, tetapi tetap memperhatikan kualitas produk yang dihasilkan.

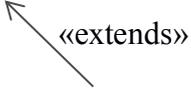
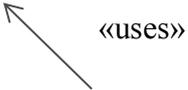
2.2.9. Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks pendukung. Macam dari diagram UML antara lain :

a. Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Tabel 2. 4 Simbol Use Case

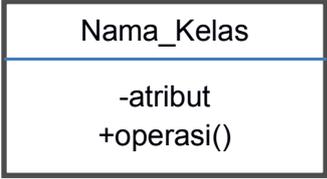
Simbol	Deskripsi
<p><i>Usecase</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor</p>
<p>Aktor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.</p>
<p>Asosiasi</p> 	<p>Komunikasi antara aktor dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan aktor</p>
<p><i>Extend</i></p> 	<p><i>Case</i> tambahan memiliki nama depan yang sama dengan <i>usecase</i> yang ditambahkan. Arah panah mengarah pada <i>usecase</i> yang ditambahkan.</p>
<p><i>Include / uses</i></p> 	<p>Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> dimana <i>usecase</i> yang ditambahkan memerlukan <i>usecase</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>usecase</i> ini.</p>
<p>Generalisasi</p> 	<p>Hubungan umum khusus antara dua buah <i>usecase</i> dimana fungsi yang satu lebih umum dari yang lainnya. Arah panah mengarah pada <i>usecase</i> yang menjadi generalisasinya (umum).</p>

Sumber : (Rosa & Shalahuddin, 2011)

b. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dan segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Tabel 2. 5 Simbol Class Diagram

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>interface</i>  nama_interface	Sama seperti konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna umum khusus
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas

Sumber : (Rosa & Shalahuddin, 2011)

c. Activity Diagram

Activity diagram menggambarkan *workflow* atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan di sini bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor.

2.2.10. MVC

MVC adalah sebuah *pattern* atau teknik pemrograman yang memisahkan antar pengembang aplikasi berdasarkan komponen utama pada sebuah aplikasi seperti memanipulasi data, user interface dan bagian yang menjadi *control* aplikasi. (Murya, 2018)

2.2.11. Python

Python merupakan bahasa pemrograman yang berorientasi obyek dinamis, dapat digunakan untuk bermacam-macam pengembangan perangkat lunak. Python menyediakan dukungan yang kuat untuk integrasi dengan bahasa pemrograman lain dan alat-alat bantu lainnya. Python hadir dengan pustaka-pustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari. Sudah banyak *programmer* Python yang menyatakan bahwa mereka mendapatkan produktivitas yang lebih tinggi. Mereka juga merasakan bahwa Python meningkatkan kualitas pengembangan karena kode sumber yang mereka tulis dapat terus dipelihara. Python dapat berjalan di banyak *platform* / sistem operasi seperti Windows, Linux/Unix, Mac OS X, OS/2, Amiga, Palm Handhelds dan telepon genggam Nokia. (Santoso, 2009)

2.2.12. Docker

Docker adalah sebuah *platform software container*. Para pengembang menggunakan docker untuk menghilangkan permasalahan-permasalahan “*works on my machine*” ketika melakukan kolaborasi kode dengan *workers*. Docker

biasa digunakan untuk menjalankan dan mengelola aplikasi *side-by-side* dalam sebuah container untuk mendapatkan proses komputasi yang lebih baik.

Image adalah paket eksekusi yang ringan, *stand-alone* dan di dalamnya mencakup semua yang dibutuhkan untuk menjalankan perangkat lunak, termasuk kode, *runtime*, *libraries*, *environment variables*, dan file konfigurasi.

Container adalah sebuah *runtime instance* dari Image. Container menjalankan aplikasi secara *native* di *host machine's kernel* dan memiliki karakteristik kinerja yang lebih baik dari pada *Virtual machine*. Docker memiliki kemampuan untuk mengurangi ukuran dari pengembangan aplikasi dan menyediakan beberapa container dengan ukuran sistem operasi yang lebih kecil. Dengan Container, dapat memudahkan pengembang dalam mengembangkan aplikasi. Docker container dapat di *deploy* di komputer fisik, VirtualMachine(VMs), Aws, Google Cloud Platform, Openshift, dll.

2.2.13. Ubuntu

Ubuntu berasal dari bahasa kuno Afrika, yang berarti "rasa perikemanusiaan terhadap sesama manusia". Ubuntu juga bisa berarti "aku adalah aku karena keberadaan kita semua". Tujuan dari distribusi Linux Ubuntu adalah membawa semangat yang terkandung di dalam Ubuntu ke dalam dunia perangkat lunak. Ubuntu adalah sistem operasi lengkap berbasis Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional. Ubuntu sendiri dikembangkan oleh komunitas sukarelawan Ubuntu.

Komunitas Ubuntu dibentuk berdasarkan gagasan yang terdapat di dalam filosofi Ubuntu: bahwa perangkat lunak harus tersedia dengan bebas biaya, bahwa aplikasi perangkat lunak tersebut harus dapat digunakan dalam bahasa lokal masing-masing dan untuk orang-orang yang mempunyai keterbatasan fisik, dan bahwa pengguna mempunyai kebebasan untuk mengubah perangkat lunak sesuai dengan yang dibutuhkan.

Perihal kebebasan inilah yang membuat Ubuntu berbeda dari perangkat lunak berpemilik (proprietary), bukan hanya peralatan yang masyarakat butuhkan tersedia secara gratis, tetapi masyarakat juga mempunyai hak untuk memodifikasi perangkat lunaknya sampai perangkat lunak tersebut bekerja sesuai dengan yang diinginkan.

2.2.14. Nginx

Nginx adalah sebuah server HTTP dan reverse proxy bebas berbasis open-source yang berkemampuan tinggi, juga dapat digunakan sebagai server proxy IMAP/POP3. Perangkat lunak ini diciptakan oleh Igor Sysoev pada tahun 2002, dan dirilis untuk pertama kalinya secara umum pada tahun 2004. Saat ini Nginx digunakan oleh 7,65% (22,8 juta) nama domain di seluruh dunia. Nginx terkenal karena performanya yang tinggi, stabil, memiliki banyak fitur, mudah dikonfigurasi, dan menggunakan sedikit sumber daya pada server. Nginx adalah salah satu dari sebagian kecil perangkat

lunak untuk server yang diciptakan untuk mengatasi masalah C10K. Tidak seperti perangkat lunak server yang umum lainnya, Nginx tidak bergantung kepada thread untuk melayani klien. Sebaliknya, Nginx menggunakan arsitektur asinkronus yang lebih stabil. Arsitektur ini membutuhkan lebih sedikit *memory*, dan yang lebih penting, dapat diperkirakan. Bahkan jika pengguna tidak mengharapkan *server* untuk mengatasi ribuan koneksi pada saat yang bersamaan, pengguna masih dapat diuntungkan dengan pemakaian *memory* yang sedikit namun berkemampuan tinggi. Nginx dapat digunakan dalam semua skala mulai dari VPS kecil hingga dengan *cluster server* dalam jumlah besar.