

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Dalam penelitian ini terdapat beberapa artikel penelitian terdahulu yang telah digunakan sebagai sumber referensi untuk membandingkan metode, *framework*, serta *tools* yang membantu menyelesaikan penyusunan penelitian ini.

Tabel 2.1 Penelitian Terdahulu

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
<p>Judul Jurnal: Desain Sistem Raport Online SMP Berbasis <i>Website</i> ASP.NET C# Dengan Pendekatan Konsep MVC</p> <p>Nama Jurnal: Vol 9 No 01 (2019): SMATIKA Jurnal : STIKI Informatika Jurnal</p> <p>Tahun: 2019</p> <p>Penulis: Mardiana Andarwati, Bima Sakti Putra, Kukuh Yudhistira</p> <p>Referensi: [9]</p>	<p>Dalam penelitian ini masalah utama yang dihadapi adalah perkembangan IT yang masih kurang di beberapa sekolah di Indonesia. Upaya untuk menyelesaikan ini khususnya untuk sekolah SMP adalah pengembangan aplikasi raport online berbasis web yang memudahkan siswa dan orang tuanya untuk melihat nilai tanpa harus pergi kesekolah.</p>	<p>Hasil dari penelitian ini menghasilkan aplikasi raport online SMPN 2 <i>website</i> menggunakan C# dengan konsep MVC sangat membantu bagi pihak sekolah.</p> <p>Kesimpulan dari penelitian ini adalah dengan konsep MVC, pembuatan aplikasi raport online dapat dipermudah sehingga menghasilkan aplikasi yang efektif dan efisien bagi pengguna yang dapat dilihat pada hasil wawancara. Diharapkan dengan penelitian ini, lebih banyak</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan framework ASP.NET dengan metode MVC yang dibangun dengan Visual Studio 2014, C# dan SQL Server 2014.</p>	<p>Penulis memberi saran kepada semua sekolah agar memaksimalkan penggunaan teknologi. Untuk kedepannya aplikasi-aplikasi selanjutnya dapat dibangun dengan design UI lebih menarik dan UX yang lebih <i>intuitive</i>.</p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
		sekolah dapat memaksimalkan penggunaan IT untuk menghadapi era global yang terus berkembang.		
<p>Judul Jurnal: Rancang Bangun Aplikasi Marketplace Jual Beli Sampah Daur Ulang Berbasis Web</p> <p>Nama Jurnal: Jurnal Teknologi Sistem Informasi dan Aplikasi</p> <p>Tahun: 2023</p> <p>Penulis: Luthfi Kamal Ananda, Khafiizh Hastuti</p> <p>Referensi : [10]</p>	<p>Dalam penelitian ini masalah yang ingin diselesaikan adalah pengelolaan sampah di Indonesia yang masih sangat kurang. Oleh karena itu penulis memberikan solusi untuk membangun Reforma Trash, sebuah web e-commerce yang berfungsi sebagai market penjualan sampah kepada pihak yang membutuhkan untuk keperluan daur ulang, kerajinan, dsb.</p>	<p>Hasil dari penelitian ini menghasilkan <i>website</i> menggunakan model <i>waterfall</i> dengan <i>framework</i> Node.Js untuk marketplace pengelolaan sampah</p> <p>Kesimpulan dari penelitian ini adalah perancangan web dengan model <i>waterfall</i> yang sangat cocok diimplementasikan untuk proyek kecil sampai sedang dikarenakan sifatnya yang sekuensial dan teratur sehingga memudahkan pembuatan aplikasi web tersebut.</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>waterfall</i> dengan NoSQL database</p>	<p>Dalam penelitian ini penulis memberikan saran untuk pengembangan aplikasi dapat ditambahkan fitur filtering untuk memudahkan pencarian produk, melakukan perubahan layout aplikasi, menggunakan autentikasi untuk login, dan menambahkan fitur <i>payment</i>.</p>
<p>Judul Jurnal: Developing a Web-based Information Sistem to Enhance Operations in Hajj and Umrah Travel (Case</p>	<p>Dalam penelitian ini masalah yang ingin diselesaikan adalah tantangan yang dihadapi perusahaan ketika masih mengalami ketergantungan dengan</p>	<p>Penelitian ini menghasilkan sistem informasi berbasis web untuk PT Mutiara Cinta Imani dalam menyelesaikan masalah terkait</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>Rapid Application Development</i> (RAD)</p>	<p>Dalam penelitian ini aplikasi dapat dikembangkan lagi pada bagian security agar tidak ada <i>vulnerabilities</i> yang dapat mengakibatkan</p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
<p>Study: PT Mutiara Cinta Imani)</p> <p>Nama Jurnal: Ultima Infosys: Jurnal Ilmu Sistem Informasi</p> <p>Tahun: 2023</p> <p>Penulis: Rosalina, Muhammad Syahir</p> <p>Referensi: [11]</p>	<p>penggunaan kertas dan excel dalam melakukan apa? sehingga terjadi potensi kesalahan dan keterbatasan yang terkait dengan praktik manajemen data. Oleh karena itu penulis memberikan solusi dalam bentuk pengembangan sistem informasi berbasis web.</p>	<p>efisiensi operasi, mitigasi error, dan limitasi dalam data <i>management</i> manual.</p> <p>Kesimpulan dari penelitian ini adalah dengan <i>Management Information Systems</i>, PT Mutiara Cinta Imani dapat meningkatkan fasilitas bisnis dan prosedur operasi. Dengan sistem informasi berbasis web pelanggan dimudahkan dalam pengimputan data.</p>		kehilangan data perusahaan.
<p>Judul Jurnal: Pengembangan Sistem Pengelolaan dan Pemantauan Proyek dengan Metode <i>Agile</i> Pola Scrum</p> <p>Nama Jurnal: Jurnal Teknik Informatika dan Sistem Informasi</p> <p>Tahun: 2020</p> <p>Penulis: Adhe Rama Febrianto, Anita Wulansari,</p>	<p>Dalam penelitian ini masalah yang dihadapi adalah kurang efektifnya <i>management process</i> yang masih manual yang dilakukan dengan menggunakan <i>Whats App</i> yang menyebabkan potensi <i>miscommunication</i>. Oleh karena itu penulis memberi solusi dalam bentuk sistem pengelolaan dan pemantauan proyek. sehingga apa?</p>	<p>Hasil dari penelitian ini adalah aplikasi sistem dengan model <i>Agile</i> yang dapat membantu Visualproyek dalam pengelolaan proyek dan service kepada customers</p> <p>Kesimpulan dari penelitian ini adalah penggunaan <i>Agile Development Method</i> sangat membantu karena sistem ini dapat dikembangkan dengan lebih cepat</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>Agile</i> pola <i>Scrum</i></p>	<p>Dalam penelitian ini, perkembangan selanjutnya dapat difokuskan pada <i>security</i> sehingga hanya karyawan perusahaan yang bersangkutan saja yang dapat melakukan pengelolaan <i>managemet process</i>.</p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
Latipah Referensi: [12]				
Judul Jurnal: Model Sistem Informasi Pemesanan Dan Produksi Berbasis Web Menggunakan Metode Agile Nama Jurnal: Jurnal Ilmiah MATRIK Tahun: 2023 Penulis: Taufik Hidayat, Henderi, Euis Nurninawati, Ruli Supriati Referensi: [13]	Penelitian ini membahas masalah yang ditemukan dalam proses pemesanan dan produksi sebuah perusahaan. Masalah ini meliputi inefisiensi dalam proses pemesanan dan produksi yang masih manual sementara perusahaan memiliki kebutuhan untuk aplikasi yang cepat.	Hasil dari penelitian ini adalah <i>website</i> untuk melakukan pemesanan dan produksi menggunakan metode <i>Agile</i> dan Bahasa pemrograman PHP Kesimpulan dari penelitian ini adalah <i>website</i> yang sudah dibangun berhasil memudahkan <i>user</i> dalam proses pemesanan.	Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>Agile</i>	Pengembangan selanjutnya dalam penelitian ini dapat ditekankan pada pemisahan <i>service</i> ke dalam bagian-bagian apa sehingga aplikasi akhir dapat terus dikembangkan kedepannya sesuai dengan kebutuhan perusahaan
Judul Jurnal: Penerapan Metode <i>Waterfall</i> Untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang Nama Jurnal: PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer	Penelitian ini membahas masalah dalam pengelolaan inventori Toko Keramik Bintang Terang yang saat ini masih dilakukan secara manual dimana memberikan potensi kesulitan pelacakan barang.	Hasil dari penelitian ini adalah sistem informasi berbasis web untuk pengelolaan <i>inventory</i> toko keramik bintang terang dengan menggunakan bahasa pemrograman PHP dengan <i>database</i> MySQL dan metode <i>waterfall</i> Kesimpulan dari penelitian ini adalah sistem	Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>waterfall</i> dengan <i>database</i> MySQL dan Bahasa pemogramam PHP	Pengembangan selanjutnya dalam penelitian ini dapat difokuskan pada peningkatan <i>user interface</i> sehingga <i>user</i> dapat memiliki pengalaman yang lebih lancar dalam menggunakan aplikasi.

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
<p>Tahun: 2021</p> <p>Penulis: Kurniawati, Mohammad Badrul</p> <p>Referensi: [14]</p>		<p>informasi berbasis web ini memudahkan toko keramik dalam pencatatan keluar masuk <i>inventory</i> dari pencatatan manual sebelumnya sehingga meningkatkan efisiensi waktu kegiatan pencatatan dan penyusunan data barang.</p>		
<p>Judul Jurnal: Perancangan Sistem Informasi Inventory Berbasis Web Pada Toko Galuh Kosmetik</p> <p>Nama Jurnal: JORAPI : Journal of Research and Publication Innovation</p> <p>Tahun: 2023</p> <p>Penulis: Galuh Surya Permana, Mochamad Sofyan Sauri, Syahrul Arifin</p> <p>Referensi: [15]</p>	<p>Penelitian ini membahas masalah dalam pengelolaan inventory pada toko kosmetik yang masih manual dengan pencatatan pada buku sehingga potensi terjadinya kesalahan atau kehilangan barang sering terjadi.</p>	<p>Hasil penelitian ini adalah sistem informasi berbasis web untuk pengelolaan <i>inventory</i> pada toko Galuh kosmetik menggunakan PHP, Javascript, Html, Css, dan <i>database</i> MySQL.</p> <p>Kesimpulan dari penelitian ini adalah sistem informasi yang telah dibangun dapat memudahkan toko dalam pengorganisasian stok dan asset, manajemen stok, pengelolaan stok, penghematan waktu dan biaya, dan peningkatan efisiensi dikarenakan inventaris yang teratur</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>waterfall</i> dengan database MySQL dan Bahasa pemogramam PHP,HTML, javascript dan CSS</p>	<p>Pengembangan selanjutnya dalam penelitian ini dapat difokuskan pada penambahan fitur lainya seperti sistem informasi untuk pembelian barang dalam sistem inventori yang sudah dibuat sehingga tidak diperlukan lagi pencatatan manual.</p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
<p>Judul Jurnal: Pembentukan dan Penerapan Enterprise Resource Planning (ERP) pada UMKM (Usaha Kecil Menengah) Toko Budi dan M-Bisy Mart</p> <p>Nama Jurnal: Jurnal Pengabdian kepada Masyarakat</p> <p>Tahun: 2019</p> <p>Penulis: Kelly Rossa Sungkono, Riyanarto Sarno, Azzam Jihad Ulhaq, Muhammad Taufiqulsa'di, Isnaini Nurul KurniaSari, Zahrul Zizki DinantoSyahrul Arifin</p> <p>Referensi: [16]</p>	<p>Penelitian ini membahas masalah dalam kegiatan bisnis Toko Budi dan M-Bisy mart yang masih dilakukan secara manual sehingga tidak sejalan dengan <i>society 5.0</i>. Oleh karena itu, penulis memberikan solusi dalam bentuk sebuah sistem ERP yang dapat memantau bisnis secara <i>real-time</i>.</p>	<p>Hasil dari penelitian ini adalah pembangunan ERP untuk UMKM Toko Budi Mulya dengan tujuan membantu pengelolaan penjualan dan memantau kegiatan perusahaan secara <i>real-time</i>. Pembangunan ERP dilakukan dengan metode <i>Agile</i>.</p> <p>Kesimpulan dari penelitian ini adalah aplikasi ERP mendapatkan respon positif dari penggunaanya yang mengatakan bahwa dengan aplikasi web ERP, pencatatan transaksi serta monitoring dalam toko dapat menghematkan waktu dalam pembuatan laporan keuangan serta mencatat stok.</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>Agile</i></p>	<p>Pengembangan selanjutnya dalam penelitian ini dapat difokuskan pada penampilan UI/UX aplikasi menggunakan tools seperti React JS atau Angular sehingga <i>user</i> dapat memiliki pengalaman yang lebih lancar dalam menggunakan aplikasi.</p>
<p>Judul Jurnal: An Enterprise Resource Planning Sistem Solution for Small-Mid Size Enterprises: An Information Sistem</p>	<p>Penelitian ini membahas masalah yang dihadapi oleh perusahaan kecil-menengah yang belum mempunyai sistem ERPnya sendiri. Hal ini</p>	<p>Hasil dalam penelitian ini adalah aplikasi sistem ERP "Quinova" untuk bisnis kecil-menengah menggunakan ASP.Net, Virtual</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan metodologi <i>waterfall</i> dengan framework ASP.NET</p>	<p>Pengembangan selanjutnya dalam penelitian ini dapat difokuskan pada penambahan fitur-fitur dan modul baru seperti modul untuk <i>Human</i></p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
<p>Development Case Study</p> <p>Nama Jurnal: Journal of Computer Science, Information Technology and Telecommunication Engineering (JCoSITTE)</p> <p>Tahun: 2021</p> <p>Penulis: Ashari Imamuddin</p> <p>Referensi: [17]</p>	<p>dikarenakan <i>cost</i>, <i>complexity</i>, atau keterbatasan perusahaan tersebut. Oleh karena itu, salah satu perusahaan <i>waste management</i> berkolaborasi dengan Sekolah Tinggi Teknologi Muhammadiyah Cileungsi untuk membangun sistem ERP dengan <i>minimal cost</i>.</p>	<p>Basic, dan <i>database</i> MySQL yang dapat diakses melalui browser atau perangkat seluler.</p> <p>Kesimpulan dari penelitian ini adalah dengan Qinova ERP, sistem ERP dapat mengelola data pelanggan, dapat menghasilkan invoice pelanggan, menyediakan fungsi untuk pembayaran bank, mendukung pengadaan barang, mencatat jurnal umum, dan menghasilkan buku besar untuk menerbitkan laporan keuangan.</p>	<p>dengan database MySQL dan Bahasa pemrograman Visual Basic</p>	<p><i>Resources</i> atau untuk <i>Supply Chain Managment</i>.</p>
<p>Judul Jurnal: Rancang Bangun API untuk Odoo ERP pada Modul Sales</p> <p>Nama Jurnal: JURNAL TEKNIK ITS</p> <p>Tahun: 2019</p> <p>Penulis: Salma Nurkhafidoh, Nurul Fajrin Ariyani, Abdul Munif</p>	<p>Penelitian ini membahas masalah terkait dengan keterbatasan ERP Odoo dimana modul sales harus mempunyai koneksi dengan internet sehingga mengakibatkan masalah ketika <i>user</i> ingin mengakses data sales namun tidak mempunyai koneksi internet.</p>	<p>Hasil penelitian adalah API <i>endpoint</i> untuk modul <i>sales</i> dalam Odoo ERP. Pembangunan API menggunakan Firebase Realtime Database. Hasil API berhasil dijalankan saat dilaksanakan uji coba dengan dua skenario A dan B. Skenario A adalah uji coba tanpa koneksi internet. Skenario B adalah uji coba</p>	<p>Pengembangan aplikasi dalam penelitian ini menggunakan framework Laravel dan MVC dengan database PostgreSQL dan Bahasa pemrograman PHP</p>	<p>Pengembangan selanjutnya dalam penelitian ini dapat difokuskan pada modul-modul lainnya dalam Odoo ERP yang dapat diimplementasi dengan <i>offline storage</i>.</p>

Jurnal	Masalah	Hasil dan Kesimpulan	Metode, Framework dan tools	Future Research
Referensi: [18]		<p>sinkronisasi antara <i>device</i> dengan Oodo server.</p> <p>Kesimpulan dari penelitian ini adalah API Odo dalam modul <i>sales</i> dapat diimplementasikan pada aplikasi web dan mobile menggunakan pemograman. API berhasil dijalankan dalam 2 kondisi internet yang terputus dan terhubung. Untuk kedepannya terdapat saran untuk menggunakan Firebase yang berbayar agar <i>data</i> dapat tersimpan dalam cache.</p>		

Berdasarkan Tabel 2.1 diatas, terdapat 10 artikel jurnal yang digunakan sebagai referensi penelitian terdahulu untuk rancang bangun aplikasi *Smart Sparepart Inventory* berbasis web untuk PT XYZ.

Dimulai dari jurnal pertama dan kesembilan, aplikasi berbasis web dan sistem *ERP* digunakan dengan *framework* ASP.NET. Berdasarkan hasil dan kesimpulan dari penelitian pertama oleh Mardiana Andarwati, Bima Sakti Putra, dan Kukuh Yudhistira, ASP.NET dengan MVC dapat mempermudah *developer* dalam pengembangan aplikasi tersebut dikarenakan pattern *Model-View-Controller* memudahkan proses debugging bagi *developer* sehingga hasil aplikasi web yang

dihasilkan lebih berkualitas [9]. Untuk hasil dan kesimpulan dari jurnal kesembilan, *ERP* yang dibangun dengan ASP.Net, Visual Basic, dan MySQL *database* dengan modul *Customer, Vendor, Accounts, Company*, dan *Sistem Administration*. Sistem *ERP* telah diimplementasi dan *go-live* di salah satu perusahaan pengangkut sampah [17].

Penelitian kedua oleh Luthfi Kamal Ananda dan Khafiizh Hastuti merupakan pengembangan aplikasi *e-commerce* berbasis web yang menggunakan *framework* dan metodologi yang berbeda yaitu Node.js dengan metode *Waterfall* yang berbeda dibandingkan dengan penelitian pertama dan kesembilan. Hasil dari penelitian tersebut adalah *website marketplace* jual beli sampah daur ulang yang menunjukkan bahwa bagi aplikasi *platform marketplace*, metode *waterfall* memudahkan proses pengembangan dikarenakan bersifat sekuensial dan teratur sehingga cocok bagi sistem skala kecil. *Websitenya* sendiri memberikan berbagai dampak baik bagi 3 pengguna survei yaitu, dari memberikan sarana untuk pengelolaan sampah daur ulang dan memudahkan akses dikarenakan dapat diakses kapan pun [10].

Penelitian ketiga oleh Rosalina dan Muhammad Syahir merupakan pengembangan sistem informasi berbasis web untuk PT Mutiara Cinta Imani yang menggunakan metode *Rapid Application Development*. Hasil dari penelitian ini dapat menyelesaikan masalah utama yang dihadapi PT Mutiara Cinta Imani yaitu proses bisnis yang tidak efisien seperti penggunaan kertas serta excel untuk pengelolaan *Travel Package*. Dengan sistem informasi berbasis web ini, karyawan dapat melakukan pengelolaan data *Travel Package* yang lebih efisien dari melihat *list package, create package, update, dan delete*. [11].

Penelitian keempat oleh Adhe Rama Febrianto, Anita Wulansari, dan Latipah, penelitian kelima oleh Taufik Hidayat, Henderi, Euis Nurninawati, dan Ruli Supriati, serta penelitian kedelapan oleh Kelly Rossa Sungkono, Riyanarto

Sarno, Azzam Jihad Ulhaq, Muhammad Taufiqulsa'di, Isnaini, Nurul KurniaSari, Zahrul Zizki, dan Dinanto Syahrul Arifin merupakan pengembangan aplikasi dengan metode *Agile*. Penelitian keempat bertujuan untuk menyelesaikan masalah dalam Visi Prima Group dalam pengelolaan *message* serta *notes* pada suatu proyek. Hasil dari penelitian ini menghasilkan aplikasi web yang dapat membantu administrator Visi Prima Group dalam pengelolaan pekerjaan proyek. Selain itu, pengembangan sistem dengan *Agile* pola scrum memberikan kemudahan bagi *developer* dikarenakan dapat secara cepat mendapatkan *feedback* langsung sehingga setiap input yang diterima dapat langsung dikerjakan [12]. Penelitian kelima merupakan pengembangan sistem informasi untuk pemesanan dan produksi berbasis web. Hasil penelitian ini berhasil dalam pengembangan aplikasi berbasis web untuk pemesanan dan produksi dengan metode *Agile*. Dengan aplikasi ini, *user* dapat dimudahkan dalam proses pemesanan barang sesuai kebutuhannya serta mempercepat proses pemesanan dan produksi dan proses pengiriman barang kepada *customer* [13]. Pembangunan ERP pada penelitian 8 menggunakan metode *Agile* sehingga aplikasi akan terus dikembangkan sesuai dengan keperluan toko. Hasil dari penelitian adalah ERP app yang berhasil membantu karyawan toko dalam pengelolaan barang [31].

Penelitian keenam oleh Kurniawati, dan Mohammad Badrul merupakan pengembangan sistem informasi berbasis web untuk pengelolaan *inventory* untuk toko keramik Bintang Terang dengan tujuan menggantikan metode pencatatan manual sebelumnya yang kurang efisien. Pengembangan sistem informasi ini menggunakan bahasa pemrograman PHP dengan *database* MySQL dan metode *waterfall* berbeda dengan penelitian keempat, kelima dan kedelapan sebelumnya. Hasil dari penelitian ini adalah sistem informasi berbasis web yang dapat membantu admin toko keramik dalam pencatatan keluar dan masuk inventori sehingga meningkatkan efisiensi waktu dalam pencatatan [14].

Penelitian ketujuh oleh Galuh Surya Permana, Mochamad Sofyan Sauri, dan Syahrul Arifin merupakan perancangan dan pembangunan sistem informasi berbasis web untuk *inventory* toko Galuh kosmetik dengan tujuan meningkatkan efisiensi dan akurasi manajemen *inventory*. Pembangunan sistem menggunakan pemrograman PHP, Javascript, Html, Css, dan *database* MySQL. Hasil dari penelitian ini adalah aplikasi web dimana *user* dapat dimudahkan dalam pengorganisasian stok dan asset [15].

Penelitian kesepuluh oleh Salma Nurkhafidoh, Nurul Fajrin Ariyani, dan Abdul Munif merupakan pembangunan API *endpoints* untuk modul *Sales* dalam ERP Odoo. Pembangunan API ini dilakukan agar aplikasi Odoo dapat digunakan dalam keadaan *offline*. Pembangunan API menggunakan *offline storage* dengan Firebase *Database*. Hasil dari penelitian adalah endpoint API berhasil digunakan tanpa adanya koneksi internet [18].

Dari kesepuluh penelitian sebelumnya, dapat ditemukan keterkaitan antar semua penelitian terutama dalam pembuatan aplikasi atau sistem berbasis web yang bertujuan untuk menyelesaikan masalah terkait kurangnya efektifitas atau efisiensi suatu proses bisnis dari pengelolaan *inventory*, pemesanan, dan pemantauan. Selain itu beberapa penelitian juga menggunakan metode yang berbeda, seperti penelitian keempat dan kelima menggunakan metode *Agile*, sedangkan penelitian kedua menggunakan metode *waterfall*. Jika diteliti maka dapat ditemukan bahwa untuk proyek dengan skala yang lebih besar sebaiknya menggunakan metode *Agile* sementara untuk proyek skala kecil dapat diterapkan metode *waterfall*.

Penelitian di PT XYZ, memiliki beberapa hal yang diadopsi dan hal yang baru dari penelitian terdahulu. Dari hal yang diadopsi, penelitian di PT XYZ menggunakan metode *Agile* sama seperti penelitian keempat dan kelima. Selain itu, bahasa pemrograman yang digunakan serta *framework*, sama seperti penelitian pertama yaitu Bahasa C# dengan *framework* MVC. Hal yang baru dapat dilihat dari

penggunaan dua *software* database dari dua vendor berbeda yaitu *SQL Server* dan *Oracle Databases*. Selain itu, aplikasi sistem berbasis web dalam penelitian ini akan dihubungkan dengan sistem *ERP Oracle EBS* dan memfokuskan pada pengelolaan *asset management* di PT XYZ. Semua pengembangan yang dilakukan dalam penelitian ini akan disesuaikan untuk menyelesaikan masalah perusahaan.

2.2 Enterprise Resource Planning (ERP)

ERP atau *Enterprise Resource Planning* merupakan gabungan sistem dalam proses bisnis yang dapat mengumpulkan dan mengintegrasikan *data* menggunakan satu *database* saja. *ERP* ini menjadi populer pada tahun 1990an dimana setiap organisasi mengimplementasikan *ERP* sistem untuk menyatukan semua sistem di setiap bagian divisi dalam organisasi yang dulunya terpisah menjadi satu sehingga meningkatkan efektivitas dan membantu memudahkan pembagian informasi di dalam organisasi tersebut [4]. Selain itu *ERP* juga dapat menolong kualitas informasi di dalam organisasi. Sejarah *ERP* dapat dibagikan ke dalam 4 tahap yaitu:

1. *MRP (Materials Requirements Planning)* = Tahap ini merupakan pemula dari sistem *ERP* sekarang. *MRP* merupakan sistem *software* yang keluar pada tahun 1970-an yang digunakan untuk mendukung *manager* dalam *production planning* dan *inventory control*. Tujuan utamanya adalah untuk memastikan ketersediaan material untuk produksi, sinkronisasi *manufacturing* dengan *delivery*, dan sinkronisasi *purchasing* dan *manufacturing*.
2. *MRP II / ERP* = Tahap kedua adalah munculnya *software* *MRP II* dimana fitur seperti *financial accounting sistem*, *sales planning*, dan *customer order processing* ditambahkan untuk mengekspansi kemampuan *MRP* ke seluruh *enterprise*. *MRP II* ini lalu berkembang menjadi *ERP*. Pada tahun 1990an *ERP* sistem keluar dengan tujuan integrasi fungsi bisnis utama dan integrasi proses bisnis ke dalam *ERP software*. Dengan *ERP* sistem ini, suatu organisasi dapat menghasilkan

aliran *data* yang mulus dalam seluruh organisasi yang memenuhi semua kebutuhan customer external dan internal.

3. *ERP II* = Tahap ketiga merupakan pengembangan dari *ERP* di tahun 2000an dimana model bisnis mulai berubah menjadi *web-based* mengikuti perkembangan teknologi. Oleh karena itu *ERP II* dikembangkan untuk memenuhi kebutuhan baru ini dengan implementasi *ERP* dengan modul lainnya seperti CRM (*Customer Relationship Management*), SCM (*Supply Chain Management*), APS (*Advanced Planning and Scheduling*), BI (*Business Intelligence*), dan *e-business*. Dengan modul CRM dan SCM suatu organisasi dapat secara lebih efektif mengatur relationship dengan organisasi, *supplier* dan *customer* dari *procurement materials* hingga *delivery* produk.
4. *Cloud ERP* = Perkembangan terbaru dari *ERP* merupakan munculnya *Cloud Computing* dimana suatu organisasi dapat memiliki *processing*, *storage* dan *software* komputer tanpa memiliki *hardware*nya sendiri.. Hal ini mendorong perkembangan *Cloud ERP* pada tahun 2010 dimana organisasi skala kecil dapat mengimplementasikan *ERP* tanpa mengeluarkan biaya yang besar untuk mempersiapkan implementasinya. Di dalam *Cloud ERP* sendiri terdapat 3 model yaitu:
 - a) *Infrastructure as a Service (IaaS)*
Model ini memungkinkan untuk perusahaan melakukan *control* aplikasi, *security*, dan *database* namun *hardware*nya sendiri dikelola oleh service provider. Dalam hal ini, Perusahaan tidak perlu membangun infrastrukturnya sendiri untuk pengimplementasian dan pemakaian *ERP*.
 - b) *Platform as a Service (PaaS)*
Model ini merupakan gabungan IaaS dengan tools pengembangan *software* yang cocok untuk perusahaan yang ingin mengimplementasikan aplikasi dalam *cloud platform*.

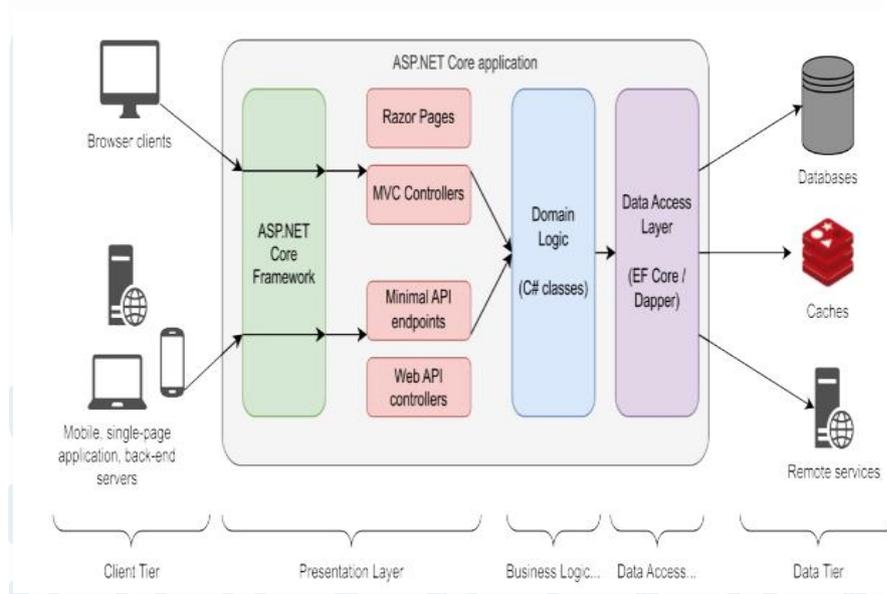
c) *Software as a Service (SaaS)*

Model ini menyediakan aplikasi yang siap untuk digunakan oleh perusahaan untuk kebutuhan fungsi bisnis spesifik mereka. Semua *control* terhadap *ERP* termasuk semua *hardware*, *software* dan *application* dilakukan oleh *service provider*.

2.3 Framework

2.3.1 ASP .NET Core

ASP .NET Core merupakan *framework* perangkat lunak *cross-platform* dari *Microsoft* yang bertujuan untuk memudahkan pengembangan aplikasi-aplikasi untuk berbagai macam aplikasi. Secara *default* ASP .NET menggunakan bahasa pemrograman C#, F# dan *Virtual Basic* [19]. ASP .NET memberikan struktur dasar untuk membangun aplikasi dengan *libraries* yang lengkap dan *helper functions* sehingga memudahkan pembangunan berbagai macam aplikasi dari *web applications*, *backend server*, HTTP APIs, MVC (*Model-View-Controller*) *applications* dan *mobile applications* dengan *minimal code* [20].



Gambar 2.1 ASP .NET Application

Gambar 2.1 diatas dari <https://livebook.manning.com/book/asp-net-core-in-action-second-edition/chapter-1/v-6/11> merupakan contoh aplikasi ASP.NET pada umumnya dengan beberapa lapisan sebagai berikut:

1. *Client Tier*

Klien melakukan *request* terhadap aplikasi untuk suatu *task* tertentu dari UI yang ada dari *browser clients, mobile, atau backend servers*.

2. *Presentation Layer*

Presentation layer merupakan tampilan-tampilan view yang dapat diakses oleh *user* dan bekerja untuk memanggil *logic tier* untuk memenuhi *request* dari *user* dan mengembalikan data dari logic kepada *user*.

3. *Business Logic / Logic Tier*

Logic Tier merupakan bagian dari aplikasi yang menghubungkan *request* dengan data yang ingin dipanggil. Disini terdapat *business logic* dimana *request user* dipastikan sudah benar sebelum diproses oleh *data tier* dan dikembalikan kepada *presentation tier*.

4. *Data Tier*

Data tier berisi semua informasi yang tersimpan oleh organisasi dalam *services* atau *database*.

2.3.2 Agile Methodology

Metodologi *Agile* merupakan salah satu *framework* pengembangan perangkat lunak yang memfokuskan pengembangan *iterative* secara berulang dan bertahap selama siklus hidup proyek. Tujuan utama dari metode *Agile* ini adalah mengurangi *overhead* saat proses pengembangan perangkat lunak dengan memberikan kemampuan *developer* untuk

mengadopsi perubahan tanpa perlunya mempertaruhkan proses atau mengulangi proyek dari awal [21]. Terdapat 4 nilai utama dalam proses *Agile* yaitu:

1. Interaksi atas proses dan alat

Agile menekankan bahwa proses pembuatan perangkat lunak lebih dipengaruhi oleh komunikasi, interaksi dan kualitas *developer* dibanding alat dan proses yang digunakan.

2. Perangkat lunak yang berjalan dibanding dokumentasi lengkap

Pengerjaan perangkat lunak dengan *Agile* tidak memerlukan dokumentasi yang intensif, melainkan membutuhkan dokumentasi yang menambah nilai pada proses pengembangan. Penilaian progress dalam *Agile* dilihat dari aplikasi yang sedang berjalan yang dapat menyelesaikan masalah atau tidak.

3. *Customer collaboration over contract negotiation*

Metodologi *Agile* menekankan kepada kolaborasi antar *developer* dengan klien sehingga dapat memudahkan proses perubahan *requirements* di titik manapun dalam pengembangan.

4. *Responding to change over following a plan*

Selama pengembangan perangkat lunak, kedua *developer* dan klien akan mendapatkan pengertian yang lebih mendalam terkait sistem yang sedang dibangun sehingga *user requirements* dapat diubah.

2.4 Tools dan Software

2.4.1 Bahasa Pemrograman C#

Bahasa pemrograman C# merupakan bahasa pemrograman *object-oriented* yang dikembangkan oleh *Microsoft*. C# sendiri termasuk ke dalam salah satu bahasa pemrograman yang paling populer di dunia. Popularitas ini disebabkan oleh kemudahan mempelajari Bahasa C# jika sudah *familiar*

dengan bahasa pemrograman C, C++, dan Java karena syntax yang menyerupai. Penggunaan utama Bahasa C# adalah pembuatan aplikasi web bersama dengan *framework* ASP.NET dan .Net Core [22]. Berikut merupakan fitur utama dari Bahasa C#:

1. *Unified type* Sistem

Blok-blok penyusun dari C# merupakan unit data yang dienkapsulasi disebut tipe. C# memiliki tipe sistem terbaru yang berarti semua tipe akhirnya berbagi tipe dasar yang sama sehingga semua tipe baik objek bisnis maupun tipe primitif memiliki dasar yang sama penggunaannya. Sebagai contoh, semua tipe data dapat dikonversi menjadi string menggunakan metode ToString.

2. *Classes and Interface*

Dalam C# terdapat beberapa tipe *class* lainya yang tidak ada pada bahasa pemrograman *object oriented* pada umumnya yaitu *interface*. *Interface* merupakan class, namun tidak dapat menampung data sehingga digunakan untuk mendefinisikan *behavior* dan memungkinkan adanya *inheritance* berganda.

3. *Properties, Methods and events*

Dalam Bahasa pemrograman *object oriented* semua fungsi merupakan *method*. Namun, dalam C# metode merupakan satu jenis anggota fungsi saja yang mencakup *property* dan *events* [23].

2.4.2 Next.js

Next.js merupakan *Framework open-source* berbasis javascript yang dibangun atas React *library* yang bertujuan untuk memudahkan pembangunan aplikasi web *full-stack* dengan memberikan berbagai macam

tools dan *features* yang lengkap. Berikut merupakan fitur-fitur utama yang disediakan oleh Next.js:

1. *Routing*

Next.js memberikan fitur file-sistem berbasis *router* yang dibangun atas komponen-komponen server yang mendukung *layout*, *nested routing*, *loading states*, dan *error handling*

2. *Rendering*

Next.js memberikan fitur *rendering* untuk *client* dan *server side* dengan komponen *server* yang dapat dioptimalisasikan lagi dengan *dynamic and static rendering* pada server.

3. *Data Fetching*

Next.js memberikan fitur yang memudahkan *fetching data* dengan komponen server *async* dan *await* serta API *fetch* untuk request memorization, *data caching* dan *revalidation*

4. *Styling*

Next.Js dapat dibangun dengan *syling* dari CSS Modules, Tailwind CSS, dan CSS-in-JS

5. *Optimizations*

Fitur dalam next.js ini memberikan optimalisasi lebih untuk menciptakan *user experience* yang lebih baik dengan *images*, *fonts*, dan optimisasi *scripts*.

6. *TypeScript*

Next.Js memberikan kompilasi yang lebih efisien serta plugin TypeScript dan type checker.

2.4.2 Swagger

Swagger merupakan *framework* untuk dokumentasi REST APIs sama seperti *Postman* yang memberikan *developer* kemudahan dalam pengujian API aplikasi dengan format bentuk JSON, XML, dan RAW TEXT. Secara default, *Swagger* sudah terintegrasi dalam proyek ASP .NET Core dalam *Visual Studio* sehingga dokumentasi dan pengujian aplikasi dapat dilakukan langsung tanpa perlunya aplikasi *3rd party*. Elemen utama dari *Swagger* terdiri dari versi *Swagger*, metode HTTP yang ada (POST, GET, PUT, DELETE) dan *status code* yang dikembalikan saat API tersebut dijalankan [24].

2.4.3 Postman

Postman merupakan *framework* untuk melakukan pengujian REST API URLs dalam bentuk JSON, XML, atau RAW TEXT. Untuk pengujian REST API, *Postman* dapat melakukan pengujian terhadap API dengan metode POST, GET, PUT, dan DELETE serta mengembalikan *status code* setelah url API dijalankan. Namun, *Postman* sendiri tidak dapat dimasukkan ke dalam satu project ASP .NET tersendiri seperti *Swagger* sehingga murni digunakan untuk pengujian API diluar aplikasi ASP .NET [20].

2.4.4 JavaScript Object Notation (JSON)

JavaScript Object Notation atau dikenal sebagai JSON merupakan *formatting* ringan berbasis bahasa pemrograman JavaScript yang mudah dibaca oleh kedua mesin dan manusia sehingga menjadi salah satu format terpopuler untuk menukar data dalam web. Hal ini dapat dilihat dalam web *services* yang memiliki API *request* dan *response* dalam protokol HTTP. Kesederhanaan format JSON dapat dikaitkan dengan dokumentasi JSON yang terdiri dari pasangan nilai kunci dimana *value* bisa menjadi dokumen JSON sehingga memungkinkan tingkat *nesting* yang sewenang-wenang [25].

2.4.5 SQL Server

SQL Server merupakan salah satu *software database management* sistem dari *Microsoft* yang digunakan untuk *create, use, dan maintain database* dalam bentuk sebuah interface antar *database* dengan *user*. Komponen dari *Microsoft SQL Server* yaitu *Database Engine* merupakan relational *database* sistem yang memiliki konsep data model berelasi dimana data tersimpan dalam tabel yang terdiri dari kolom dan baris, setiap tabel memiliki ID mereka tersendiri sebagai *primary key* dari tabel tersebut [26]. Dalam SQL Server, Bahasa T-SQL (*Transact-Structured Query Language*) digunakan untuk mengelola data dalam relational *database* dari pengambilan data, memperbaharui, menghapus, membuat *database*, membuat tabel serta data isinya, membuat *stored procedure*, dan mengatur *permissions* dalam *database* [27].

2.4.6 Oracle E-Business Suite

Oracle E-Business Suite atau disingkat *Oracle EBS* merupakan sistem aplikasi *ERP* pertama yang dikembangkan oleh *Oracle*. Sama seperti *ERP SAP* atau *Odoo*, *Oracle EBS* bertujuan untuk memudahkan perusahaan dalam meningkatkan efektivitas dan efisiensi dalam kegiatan proses bisnis dari sisi manufaktur, *inventory, reporting, business intelligence, CRM (Customer relationship management), HR, finance, sales & order, dan procurement* [28]. Modul-modul dalam *Oracle E-Business* dapat dikategorikan ke dalam 9 yaitu:

1. *Order Management*

Modul *Order Management* memastikan bahwa semua transaksi dan proses transaksi oleh perusahaan dipastikan akurat dan efisien

2. *Logistics*

Modul *Logistics* dalam *Oracle E-Business Suite* meliputi pengelolaan *Inventory, Cost Management, Supply Chain Applications, Warehouse Management, dan Yard Management.*

3. *Procurement*

Modul *Procurement* dalam *Oracle E-Business Suite* meliputi pengelolaan *purchasing, procurement, sourcing, supplier management, dan service contracts.*

4. *Projects*

Modul *Projects* dalam *Oracle EBS* meliputi *Project Billing, Project Contracts, Project Costing, Project Planning and Control, Oracle Project Portfolio Analysis, dan Project Resource Management.*

5. *Manufacturing*

Modul *Manufacturing* dalam *Oracle EBS* meliputi *Flow Manufacturing, Master Production Scheduling, Outsourced Manufacturing, Quality, Project Manufacturing, Process Manufacturing, dsb.*

6. *Asset Lifecycle Management*

Modul *Asset Lifecycle Management* dalam *Oracle EBS* meliputi *Assets, Maintenance Repair and Overhaul, dan Enterprise Asset Management.*

7. *Service*

Modul *Service* dalam *Oracle EBS* meliputi *Telephony, Scheduler, Repair, Email Center, Field Service*, dsb.

8. *Financials*

Financials dalam *Oracle EBS* meliputi *Accounts Receivable Deduction Settlement, Cash and Treasury Management, Treasury, Credit-to-Cash, Loans, Credit Management, Payments, Receivables, General Ledger, Payables*, dsb.

9. *Human Capital*

Modul *Human Capital* dalam *Oracle EBS* meliputi *Human Capital Management, Talent Management*, dan *Workforce Management*.

2.4.7 Oracle Enterprise Asset Management (EAM)

Oracle Enterprise Asset Management atau EAM merupakan aplikasi dalam *Oracle E-Business Suite* yang dirancang untuk memberikan kemampuan kepada perusahaan untuk perencanaan komprehensif, pelacakan, dan eksekusi support untuk pemantauan asset. Dirancang untuk pemeliharaan asset dalam pabrik, *property*, dan infrastruktur. EAM menghapus kelebihan dan hal-hal yang tidak diperlukan lagi dalam *storage inventory*, mempromosikan kebijakan lingkungan, dan meningkatkan koordinasi jadwal produksi dan pemeliharaan [29].

2.4.8 Visual Studio

Visual Studio merupakan perangkat lunak *integrated development environment* dari *Microsoft* yang digunakan untuk mengedit, debug, dan

menulis *code*. *Visual Studio* merupakan IDE terlengkap untuk pengembangan aplikasi .NET dengan fitur yang lengkap yang memudahkan dan mempercepat semua tahap perancangan dan pembangunan aplikasi [30].

2.4.9 GitLab

GitLab merupakan aplikasi *workflow web-based* yang digunakan untuk mengelola *Continuous Integration* dan *Continuous Development*. GitLab menyediakan *team management* dan *tools* seperti *responsibilities*, *milestone*, dan *issues* yang dirancang untuk memudahkan dan memperlancar komunikasi dan kolaborasi dalam tim sepanjang siklus pengembangan aplikasi [31].

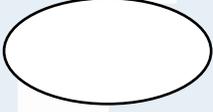
2.4.10 UML (*Unified Modelling Language*)

Unified Modelling Language atau UML merupakan salah satu tool untuk pemodelan yang terdiri dari berbagai diagram yang bertujuan untuk membantu *developer* perangkat lunak dalam memvisualisasikan, membangun, dan mendokumentasikan struktur, *flow*, dan *behavior* sebuah sistem yang ingin dibangun. UML sendiri telah menjadi standar untuk perancangan *blueprint* sistem yang meliputi semua konsep proses bisnis secara *detail* dengan tujuan mengkomunikasikan kepada *developer* maupun *non-developer* terkait fungsi sistem dari fitur, *requirements*, serta proses [32].

2.4.10.1 *Use case Diagrams*

Use case Diagrams merupakan diagram yang menggambarkan fungsi suatu sistem yang mengilustrasikan relasi antar berbagai agent luar (*Actors*) dengan sistem. Berikut merupakan komponen dari *Use case Diagram* dilihat pada tabel 2.2 dibawah:

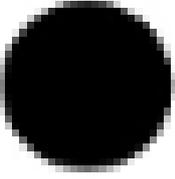
Tabel 2.2 Komponen Use case Diagram

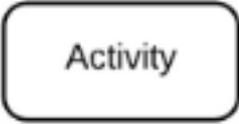
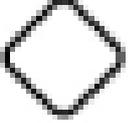
Gambar	Nama	Keterangan
	<i>Actor</i>	<i>Actor</i> merupakan entitas eksternal yang berada diluar sistem dan dapat berinteraksi langsung dengan sistem. <i>Actors</i> terdiri dari <i>user</i> , <i>admin</i> , <i>hardware</i> , dsb
	<i>Use case</i>	<i>Use case</i> merupakan gambaran dari fitur-fitur atau proses yang dapat dilakukan dalam sistem
	<i>Association Relationship</i>	Merupakan relasi atau komunikasi antar actor dengan <i>use case</i>

2.4.10.2 Activity Diagrams

Activity Diagrams merupakan diagram yang digunakan untuk menjelaskan flow dari semua aktivitas atau proses yang terjadi dalam sistem yang ingin dibangun. Berikut merupakan komponen dalam *Activity Diagram* dilihat pada tabel 2.3 dibawah:

Tabel 2.3 Komponen Activity Diagram

Gambar	Nama	Keterangan
	<i>Start</i>	Merupakan simbol yang merepresentasikan mulainya suatu flow atau aktivitas

	<i>Activity</i>	Merupakan aktivitas dalam suatu proses
	<i>Connection</i>	Menunjukkan arah <i>flow</i> dari suatu aktivitas
	<i>Decision</i>	Merupakan kondisi dengan 2 jalur bercabang
	<i>Flow Final</i>	Merupakan simbol berakhirnya satu proses, bukan akhirnya dari semua aktivitas
	<i>End</i>	Merupakan simbol berakhirnya semua aktivitas dan proses

2.4.10.3 Class Diagrams

Class Diagrams merupakan diagram static yang digunakan untuk memvisualisasikan model *object oriented* sistem. Terdiri dari class, atribut dari class dengan tipe *datanya* masing-masing seperti integer dan string, method yang menunjukkan fungsi dari suatu class, dan relasi antar setiap class tersebut. Hasil dari *Class Diagrams* ini akan menggambarkan *blueprint* dari semua tabel-tabel dalam *database* yang akan dibangun nanti.

2.4.11 UAT

User Acceptance Testing atau disingkat sebagai UAT merupakan proses terakhir dalam proses pengembangan aplikasi sebelum deployment yang terdiri dari penilaian dan pengujian terhadap solusi di dalam suatu sistem apakah sudah dapat menyelesaikan masalah dari *user/klien*. Tujuan utama dilakukannya UAT adalah agar *user/klien* dapat memastikan bahwa sistem yang telah dibuat dapat memenuhi semua kebutuhan (*Requirements*) yang sudah didefinisikan sebelumnya dan apakah sistem tersebut sudah berjalan dengan baik tanpa kendala dan sesuai dengan keinginan *user/klien*.

User Acceptance Test pada umumnya dilaksanakan setelah unit *testing* dan *System Integration testing* (SIT), dan diberikan kepada *user/klien* untuk berinteraksi dengan hasil sistem dan mencari apakah terdapat fitur yang kurang atau *bugs* dalam pengetesan [33]. Masalah-masalah yang muncul saat pelaksanaan UAT dapat dikategorikan menjadi tiga, yaitu:

1. *Critical*

Masalah *critical* mengganggu fungsi utama dari sistem yang berdampak langsung dari keseluruhan proses bisnis. Masalah *critical* ini harus diselesaikan agar sistem dapat diimplementasikan dalam ranah produksi.

2. *Major*

Kegagalan *major* merupakan masalah-masalah terhadap fungsi-fungsi penting dalam sistem namun hanya mempengaruhi terhadap penggunaan sistem, dan tidak terhadap proses bisnis.

3. *Minor*

Kegagalan minor tidak termasuk ke dalam masalah-masalah darurat dan hanya menyebabkan gangguan kecil dalam penggunaan sistem.

2.4.12 *Intranet*

Intranet merupakan jaringan privat dalam suatu badan atau organisasi yang digunakan untuk menjamin keamanan data dengan memastikan *information sharing* hanya dilakukan oleh anggota atau karyawan dalam organisasi yang bersangkutan. Berbeda dengan Internet dan Extranet dimana Internet bersifat umum dengan *user* dari seluruh dunia yang dapat mengakses *network*, sedangkan extranet merupakan *network* yang privat namun dikhususkan untuk *sharing information* dari satu organisasi kepada organisasi lainnya, ESS-BPPM dalam penelitian ini merupakan aplikasi *Intranet* yang selain bertujuan untuk menjamin keamanan data PT XYZ bertujuan pula untuk pengelolaan bisnis di PT XYZ dan dalam penelitian ini [34], pengelolaan sparepart mesin untuk karyawan internal dari PT XYZ.

