

BAB II

LANDASAN TEORI

2.1. *Enterprise Resource Planning dan Material Requirement Planning*

2.1.1 Enterprise Resource Planning (ERP)

Enterprise Resource Planning (ERP) adalah sebuah aplikasi yang dapat mengintegrasikan informasi ke seluruh divisi perusahaan dengan menggabungkan seluruh informasi yang ada dalam satu unit *database*[2]. *Database* dapat disimpan secara lokal jika sistem ERP diimplementasikan secara *on premise*, *database* juga dapat disimpan dalam *cloud* jika sistem ERP diimplementasikan secara *cloud-based*. Sistem ERP merupakan salah satu aspek penting dalam perubahan organisasi dalam perusahaan dengan implikasi besar terhadap organisasi dan manajemen kerja[3]. ERP menjadi sebuah pondasi yang ada dalam teknologi *e-business* yang dapat menghubungkan struktur kerja dalam bentuk transaksi yang terjadi di seluruh dunia. Hubungan yang terjadi bisa berupa proses manajemen penjualan, gudang, kontrol keuangan, dll[4]. ERP juga dapat memungkinkan terjadinya aktivitas integrasi dan pemrosesan transaksi serta aktivitas perencanaan. ERP mendukung berbagai macam situasi yang ada dan memfasilitasi berbagai jenis bahasa. Selain itu, ERP mendukung praktik bisnis pada industri tertentu dan dapat dimodifikasi tanpa perlu banyak pengkodean[5].

Terdapat tiga jenis metode implementasi sistem *enterprise resource planning* (ERP) yaitu[6]:

1. On Premise

On Premise merupakan sistem ERP yang dimana perusahaan menyediakan infrastruktur ERP seperti *server database* sendiri. Jenis ini memerlukan biaya dan waktu yang lebih lama dibandingkan jenis lainnya. Hal ini dikarenakan perusahaan harus mengeluarkan biaya lebih untuk menyediakan infrastruktur dan juga untuk pemeliharaan infrastruktur tersebut kedepannya. Waktu yang dibutuhkan untuk

implementasi juga lebih lama karena perusahaan harus melakukan konfigurasi dari awal pada infrastruktur yang baru dibuat. Meskipun membutuhkan biaya dan waktu yang lebih besar, jenis ini memberikan keuntungan seperti keamanan data yang lebih aman dan kemudahan kontrol dan modifikasi sistem sesuai keperluan perusahaan.

2. Cloud

Cloud ERP adalah sistem ERP yang menggunakan *cloud service* dalam sistem ERP. Dalam *cloud ERP*, infrastruktur disediakan *provider* dalam *cloud*. Hal ini membuat pengguna tidak perlu menyediakan infrastruktur apapun. *Software* aplikasi sistem ERP juga disediakan lewat *cloud* sehingga dapat diakses dimana saja selama ada koneksi ke *cloud*. Biaya yang diperlukan untuk implementasi juga lebih murah karena pengguna tidak perlu membayar infrastruktur dan hanya membayar biaya langganan sistem ERP secara berkala.

3. Hybrid

Hybrid ERP merupakan sistem ERP yang terdiri dari campuran *on premise* dan *cloud*. Secara umum, dalam *hybrid erp* menggunakan 2 jenis tingkatan fondasi. Pondasi tingkat 1 menggunakan *on premise* untuk menampung bagian bisnis yang penting sedangkan pondasi tingkat 2 menggunakan *cloud* untuk mendukung bagian bisnis lainnya. Pembagian tingkatan ini disesuaikan dengan keinginan dan kebutuhan dari perusahaan.

2.1.2 Material Requirement Planning (MRP)

Material Requirement Planning (MRP) adalah salah satu metode yang dapat digunakan dalam mengelola *inventory*. MRP merupakan sistem informasi yang digunakan dalam mengatur jadwal proses produksi beserta bahan baku yang dibutuhkan[7]. MRP juga merupakan salah satu pendahulu sistem ERP sebagai

generasi kedua kedua sistem komputasi tersentralisasi. Sistem MRP juga bertujuan untuk membantu mengontrol *demand-dependent inventory* secara otomatis. Tujuan utama dari sistem MRP yaitu memastikan bahan baku dialokasikan sesuai kebutuhan di waktu yang tepat. Sistem ini menggunakan informasi dari *master production schedule*, *bill of materials*, *production cycle*, dll dalam menentukan bahan baku yang dibutuhkan setiap proses produksi beserta waktu pengalokasiannya. Pada penelitian ini, sistem MRP yang ada dibangun memiliki fitur yaitu manajemen *Manufacturing Order (MO)* dan *Bill Of Material (BOM)*.

Sebagai suatu bentuk aktivitas proses, material requirement planning memiliki beberapa fungsi dengan input nya masing-masing yaitu[8]:

1. Data dari total permintaan sebagai salah satu data utama atau sumber data bagi proses penjadwalan produksi. Data ini berasal dari data hasil prediksi penjualan atau *forecasting* penjualan serta dari data historis penjualan di tahun-tahun sebelumnya. Selain itu, data ini juga berasal dari data pemesanan atau order dari pelanggan.
2. Status dari gudang atau status *inventory* yang berkaitan dengan departemen gudang. Status gudang ini meliputi informasi terkait bagaimana kondisi stok persediaan barang terutama bahan baku yang ada di gudang. Informasi ini digunakan untuk merencanakan kapan suatu proses produksi dapat dilakukan berdasarkan kondisi stok bahan baku di gudang. Status gudang ini merupakan bentuk integrasi dari departemen gudang dengan departemen produksi.
3. Rencana produksi merupakan salah satu aktivitas utama yang dapat dilakukan dalam *material requirement planning*. Rencana produksi dilakukan untuk menjadwalkan produksi suatu barang agar kegiatan produksi dapat berjalan dengan teratur dan dapat memenuhi semua kebutuhan akan produksi dengan baik. Rencana produksi menentukan kapan barang diproduksi, jumlah bahan baku yang dibutuhkan, dan sumber daya lain yang diperlukan oleh rencana produksi tersebut.

4. *Bill of material* merupakan salah satu bagian penting dari *material requirement planning*. *Bill of material* berfungsi untuk mengalokasikan bahan baku terhadap suatu produk. Tujuan untuk dari *bill of material* adalah membuat struktur produk untuk kebutuhan perencanaan produksi. Hal ini dibutuhkan pada saat kegiatan perencanaan produksi dilakukan sehingga struktur dari suatu produk yang ingin diproduksi sudah ada. Struktur produk ini meliputi jumlah setiap bahan baku atau komponen yang diperlukan untuk membuat suatu produk. Saat melakukan perencanaan produksi maka *bill of material* ini hanya tinggal dipanggil saja menyesuaikan produk yang diproduksi sehingga kegiatan perencanaan produksi lebih mudah tanpa perlu menentukan komponen produk yang diproduksi.

Salah satu hasil dari material requirement planning pada dasarnya adalah jadwal kegiatan produksi yang sebelumnya sudah dibuat dan direncanakan lewat pembuatan manufacturing order. Jadwal produksi ini mengatur kegiatan produksi suatu barang agar alur produksi barang tersebut terstruktur. Hasil dari jadwal perencanaan produksi inilah yang dapat mengatasi masalah seperti stok kurang, waktu bertabrakan dengan produksi lain, *overproduction*, dll.

2.2 MySQL

MySQL adalah sebuah sistem manajemen basis data relasional yang bersifat *open source*. MySQL menggunakan *Structured Query Language* sebagai bahasa pemrograman yang digunakan untuk memanipulasi basis data. Selain itu, MySQL juga dapat mendefinisikan struktur tabel data beserta relasi antar tabel data tersebut menggunakan skema basis data. MySQL memiliki arsitektur yaitu *server client side* yang cocok digunakan dalam perancangan dan pengembangan sistem berbasis web. *Database* MySQL juga memiliki fitur yaitu dapat diakses di berbagai *platform* berbeda[9].

2.3 PhpMyAdmin

PhpMyAdmin adalah sebuah aplikasi web berbasis PHP yang digunakan untuk mengelola basis data dari MySQL. PhpMyAdmin menggunakan bahasa pemrograman yaitu PHP. PHP adalah bahasa pemrograman server side untuk mengembangkan aplikasi berbasis web. Sama seperti MySQL, PhpMyAdmin juga menggunakan skema basis data untuk mengelola data tabel yang ada beserta dengan relasinya. PhpMyAdmin juga memiliki antarmuka pengguna yang intuitif dan mudah digunakan[10].

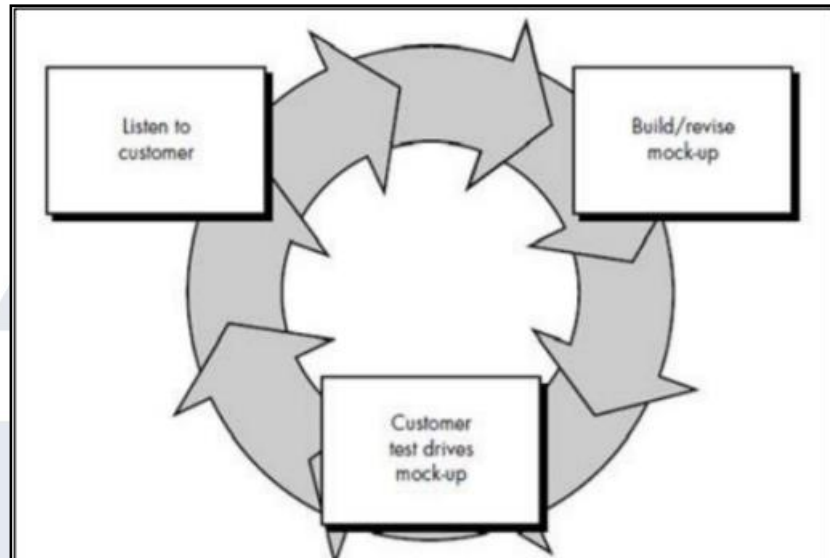
2.4 System Development Lifecycle

2.4.1 Prototype

Mode *prototype* adalah suatu teknik dalam mengumpulkan informasi tertentu terkait suatu kebutuhan pengguna secara cepat. Metode ini berfokus pada penyajian aspek aplikasi yang akan nampak bagi pelanggan atau pemakai . Prototype ini akan dievaluasi oleh pengguna atau pemakai untuk mendapatkan masukan dari sisi pengguna. Masukan ini akan digunakan untuk melakukan tahapan pengembangan *prototype* tersebut. *Prototype* didefinisikan sebagai sebuah alat pemberi ide bagi perancang dan pengguna aplikasi. *Prototype* membantu memberikan penjelasan lengkap terkait cara kerja sistem serta bagaimana bentuk dari sistem tersebut.

Metode *prototype* digunakan untuk mendapatkan gambaran awal dari sistem yang akan dibuat. Rancangan sistem dari awal hingga *prototype* terbentuk dan dievaluasi oleh pengguna. Setelah dievaluasi oleh pengguna, masukan yang ada menjadi rujukan untuk pengembangan sistem kedepannya menghasilkan sistem akhir.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 2.1 Model *Prototype* [11]

Metode prototype memiliki tiga tahapan utama yaitu:

1. *Listen to customer*: Tahapan pertama ini bertujuan untuk mendapatkan informasi terkait keinginan atau kebutuhan pengguna akan sistem yang akan dibuat. Pada tahapan ini dilakukan analisa kebutuhan, masalah, dan semua *requirement* yang dibutuhkan.
1. *Build/Revise mock-up*: Tahapan kedua ini dilakukan untuk merancang dan membuat *prototype* sistem berdasarkan *requirement* yang sudah diberikan oleh pengguna. Perancangan sistem dilakukan hingga *prototype* sistem terbentuk untuk selanjutnya dievaluasi.
2. *Customer test drives mock-up*: Tahapan terakhir ini bertujuan untuk mengevaluasi hasil *prototype* sistem yang sudah dibuat di tahapan sebelumnya. Jika pengguna memiliki masukan atau saran maka kembali ke tahapan kedua untuk melakukan revisi *prototype* sistem sesuai dengan masukan pengguna.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

2.4.2 Iterative

Iterative system development lifecycle adalah salah satu metodologi yang dapat digunakan dalam mengembangkan sistem. Iterative development merupakan strategi penjadwalan ulang dalam pengembangan perangkat lunak di mana waktu dihabiskan untuk merevisi dan meningkatkan bagian-bagian sistem. Dalam hal ini, kode fitur dirancang, dikembangkan, dan diuji dalam siklus berulang. Tidak diperlukan persyaratan terperinci untuk memulai, karena perbaikan dimulai dengan menunjukkan dan menjalankan bagian-bagian perangkat lunak. Prosedur ini kemudian diulang untuk menciptakan versi baru dari perangkat lunak pada akhir setiap siklus model. Model pengembangan iteratif memecah pengembangan produk aplikasi yang besar menjadi bagian-bagian kecil. Oleh karena itu, disebut sebagai proses memecah pengembangan perangkat lunak aplikasi besar menjadi potongan-potongan kecil.

Salah satu fitur penting dari model iteratif adalah adanya model sistem yang berfungsi secara keseluruhan pada awal latihan pengembangan, sehingga memudahkan untuk segera menemukan kesalahan konfigurasi dan mengambil tindakan perbaikan secara tepat waktu dalam perencanaan biaya. Keunggulan dari model iteratif memastikan bahwa versi sistem yang lebih baru secara bertahap ditingkatkan dari siklus sebelumnya. Model iteratif juga tidak memakan waktu yang lama karena setiap tahapan pengembangan disesuaikan dengan kebutuhan awal dan langsung dikerjakan oleh tim sehingga sistem dapat dipakai secepat mungkin. Kekurangan dari model ini yang paling terlihat adalah kebutuhan waktu dan biaya yang tidak tetap. Hal ini dikarenakan biaya yang dibutuhkan bisa bertambah dari perkiraan awal jika memang ada banyak perubahan yang dilakukan diluar sistem awal yang dirancang. Hal ini juga akan memakan waktu yang lebih lama dari perkiraan awal[12]

2.4.3 Spiral

Model ini menyediakan dukungan untuk Penanganan Risiko. Ketika kita mengikuti model spiral untuk merancang aplikasi web, kita akan menemukan bahwa prosesnya fleksibel, tangkas, mudah dipelihara, dan lebih cocok. Model spiral membagi proses secara keseluruhan menjadi berbagai modul, dan setiap modul disediakan kepada pelanggan sehingga mereka dapat memanfaatkannya sejak awal. Model yang diusulkan juga menguntungkan karena jika kebutuhan klien berubah di tengah jalan, kita dapat dengan mudah mengimplementasikannya karena kita mengembangkan proyek dalam tahap-tahap. Seperti namanya, semua kegiatan proyek diorganisir dalam bentuk spiral. Anggaran proyek yang dikembangkan pada setiap titik waktu dapat ditentukan oleh jari-jari spiral, dan atribut sudut membantu kita menentukan kemajuan yang dicapai dalam setiap siklus.

Ada empat fase dalam model spiral, dan setiap proses melewati semua fase, dengan peningkatan pada setiap spiral terjadi penyempurnaan. Jumlah loop spiral yang tepat tidak diketahui dan dapat berubah untuk setiap proyek. Setiap loop spiral disebut sebagai Fase Proses Pengembangan Perangkat Lunak. Jumlah langkah yang diperlukan untuk pengembangan produk dapat berubah oleh administrator proyek berdasarkan risiko yang ditentukan untuk proyek tersebut. Karena orang yang mengelola proyek menentukan jumlah langkah dengan akurat, maka manajer proyek memainkan peran penting dalam pengembangan perangkat lunak menggunakan model Spiral. Penentuan harga produk perangkat lunak dapat ditentukan oleh seberapa luas cakupan proyek tersebut[13].

2.5 Framework

1. Codeigniter

Penelitian ini menggunakan bahasa pemrograman PHP dalam pembuatan rancang bangun sistem ERP berbasis web. Framework bahasa pemrograman PHP yang digunakan adalah Codeigniter. Codeigniter adalah sebuah *framework* PHP yang terdiri dari sekumpulan *file* dan *folder*. *File* dan *folder* yang terdapat dalam Codeigniter terdiri dari *file* PHP, JavaScript, CSS, TXT, dan *file web base* lainnya. Codeigniter adalah *framework* PHP yang lebih mudah digunakan dibandingkan *framework* PHP lainnya. Codeigniter membutuhkan sebuah *web server* untuk dapat berjalan. Oleh karena itu, pada penelitian ini digunakan XAMPP sebagai *web server* dan Codeigniter sebagai *framework*. Codeigniter dipilih sebagai *framework* karena terdapat konsep MVC (*Model.View,Controller*) dalam *framework* tersebut.

Model ini yang membuat Codeigniter lebih mudah dipelajari dan dikuasai dibandingkan *framework* lainnya. Model ini memungkinkan pembagian *file* berdasarkan jenis kode pemrograman yang ada. *File Model* berisikan kode pemrograman yang mengatur *database*. *File View* mengatur kode pemrograman yang memberikan tampilan antarmuka web ke pengguna. *File Controller* berisikan kode pemrograman yang menghubungkan *view* dengan *model*. Model ini memberikan kemudahan bagi *user* dalam pengkodean sistem karena kode-kode pemrograman tidak tercampur dalam satu *file* yang sama. Hal ini memudahkan proses *debugging* dan lebih mudah untuk dibaca.

2. Bootstrap

Bootstrap merupakan salah satu *framework* paling populer untuk bahasa pemrograman HTML, CSS, dan Javascript. *Framework* ini banyak digunakan dalam pembuatan *website* karena fitur lengkap yang diberikan. Bootstrap membuat *website* yang dikembangkan menjadi lebih *responsive* dan *mobile friendly*. Bootstrap memberikan kemudahan bagi pengembangan front-end karena sudah terdapat template untuk tipografi, *forms*, *button*, *tables*, *navigation*, dll yang sering digunakan dalam perancangan *front-end web*. Hal ini membuat *developer* atau *programmer*

lebih mudah dalam merancang *website* karena *template* dasar yang sudah disediakan oleh Bootstrap. *Programmer* bisa menggunakan *template* ini atau memodifikasinya sesuai keinginan pengguna. Hal ini memungkinkan karena *template* yang diberikan Bootstrap berupa *code* yang dapat dimodifikasi untuk mendapatkan hasil yang sesuai[14].

Selain itu, Bootstrap juga dapat menggunakan fitur *plug-ins* dari Javascript. Javascript inilah yang membuat Bootstrap lebih *responsive* dan *mobile friendly* dari *framework* lain. Selain *template* HTML dan CSS yang jauh lebih baik ditambah dengan fitur Javascript yang *responsive* membuat *framework* ini lebih unggul dari *framework* lainnya. Bootstrap juga merupakan salah satu *framework* yang mudah digunakan bagi pemula. Hal ini dikarenakan semua fitur yang diberikan Bootstrap didasari oleh HTML, CSS, dan Javascript. Jika pengguna mengetahui dasar dari ketiga bahasa pemrograman tersebut maka penggunaan Bootstrap akan jauh lebih mudah. Dalam penelitian ini, *framework* Bootstrap digunakan karena kemudahan dalam perancangan *front-end web* dengan *template* yang sudah diberikan dan fitur *customisation* yang mudah dilakukan serta fleksibel. Bootstrap akan menjadi *framework front-end* utama dalam perancangan sistem web ini.





2.6 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah sebuah notasi yang dihasilkan dari penggabungan *Object Modelling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Tujuan utama dari UML adalah untuk menyediakan notasi standar yang dapat digunakan oleh semua metode objek yang terorientasi. UML juga bertujuan untuk memilih serta mengintegrasikan elemen terbaik dari notasi pendahulunya. UML divisualisasikan dalam penggambaran diagram yang terdiri dari beberapa jenis seperti *class diagram*, *activity diagram*, dan *use case diagram*. Berikut ini adalah penjelasan dari masing-masing diagram tersebut[15].

1. Class Diagram

Class diagram adalah sebuah diagram yang digunakan untuk mendeskripsikan struktur dari sebuah sistem. Terdapat beberapa komponen dalam *class diagram* yaitu *class* dan *object*. *Class* adalah sebuah komponen yang menentukan struktur umum dan perilaku dari sekumpulan objek. *Object* adalah sebuah instansi dari *class* yang sudah dibuat, dimodifikasi, dan dihapus selama sistem dieksekusi. Sebuah objek memiliki keadaan yang mencakup nilai dari atributnya dan hubungan dengan objek lain[34]. *Class diagram* memiliki relasi atau hubungan yaitu:


Tabel 2.1 Tabel Keterangan *Class Diagram*





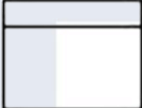
Simbol	Keterangan
	Agregasi: Relasi antar kelas dengan makna whole-part atau bagian dari kelas lain.
	Asosiasi: Relasi antar kelas yang ada dengan makna umum.
	Komposisi: Relasi antar kelas dengan makna kebergantungan. Satu kelas tidak bisa berdiri tanpa kelas lainnya.
	Generalisasi: Relasi antar kelas dengan makna yaitu umum-khusus (generalisasi-spesialisasi)

2. Activity Diagram

Activity diagram menggambarkan seluruh aktifitas yang ada dalam sistem tersebut. *Activity diagram* menggambarkan alur darai serangkaian aktivitas yang terjadi. *Activity diagram* memiliki beberapa komponen sebagai berikut:

Tabel 2.2 Tabel Keterangan *Activity Diagram*

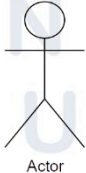
Simbol	Keterangan
	Status Awal: Merupakan status awal dari sebuah activity diagram. Menggambarkan permulaan dari activity diagram tersebut.

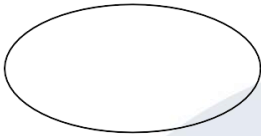

	Aktivitas: Aktivitas yang dilakukan oleh sistem dan biasanya diawali dengan kata kerja.
	Percabangan: Percabangan adalah kondisi dimana terdapat pilihan keputusan yang lebih dari satu pilihan aktivitas.
	Penggabungan: Penggabungan adalah keadaan dimana ada lebih dari satu aktivitas yang digabungkan menjadi satu.
	Status Akhir: Status akhir adalah status terakhir yang dilakukan.
	Swimlane: Swimlane memisahkan bagian bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. Use Case Diagram

Use Case Diagram adalah sebuah diagram yang bertujuan untuk mengumpulkan persyaratan dan analisa untuk mewakili fungsionalitas sebuah sistem. *Diagram* ini berfokus pada perilaku sistem dari sudut pandang eksternal. *Use Case Diagram* menggambarkan fungsi yang diberikan sistem beserta aktor yang terlibat dalam fungsi tersebut. *Use Case Diagram* membedakan fungsionalitas yang diberikan sistem beserta aktor-aktor yang berperan. Berikut ini adalah komponen yang ada dalam *Use Case Diagram*.

Tabel 2.3 Tabel Keterangan *Use Case Diagram*

Simbol	Keterangan
	Aktor: Aktor adalah segala hal diluar sistem yang mencakup pengguna sistem untuk melakukan fungsionalitas yang ada dalam sistem.

	<p><i>Use Case:</i> <i>Use Case</i> adalah gambaran fungsionalitas sistem yang diberikan. Komponen ini menjelaskan fungsionalitas apa saja yang diberikan oleh sistem.</p>
	<p>Relasi Asosiasi: Relasi ini adalah relasi yang menghubungkan antara aktor dan <i>use case</i> yang ada pada <i>use case diagram</i></p>

2.7 *Programming Language*

1. PHP

PHP adalah salah satu bahasa pemrograman yang paling sering digunakan dalam pengembangan aplikasi karena fleksibilitas yang ada dan kemudahan dalam penggunaan dan pembelajaran bahasa pemrograman PHP[16]. PHP dapat digunakan untuk 2 halam utama yaitu *server side scripting* dan *command line scripting*. *Server side scripting* digunakan untuk pembuatan konten web dinamis sedangkan *command line scripting* digunakan untuk menjalankan *scripts* dari command line seperti Perl, awk, atau Unix *shell*[17]. PHP dapat dijalankan di mayoritas sistem operasi yang ada, mulai dari varian Unix seperti Linux hingga Windows dan MacOS.

2. *Hypertext Markup Language* (HTML)

HTML adalah bahasa pemrograman standar yang umum dan paling sering digunakan oleh pengembang web dalam mengembangkan aplikasi web. HTML juga merupakan bahasa pemrograman yang digunakan untuk melakukan pertukaran dokumen antar web dengan menghubungkan kedua web tersebut[18].

3. Cascading Style Sheets (CSS)

CSS adalah bahasa pemrograman yang umum digunakan untuk mendefinisikan isi dokumen terstruktur dan antarmuka pengguna[19]. CSS menunjukkan *style* yang digunakan naskah HTML. CSS mendefinisikan bagaimana seharusnya elemen dari HTML ditampilkan[20].

4. Javascript

Javascript adalah bahasa pemrograman yang memberikan situs web dinamis yang dapat secara aktif berinteraksi dengan pengguna. Javascript memungkinkan pengguna atau *user* berinteraksi dengan web serta untuk mengembangkan aplikasi web yang interaktif. Pada aplikasi web saat ini, Javascript digunakan sebagai bahasa yang mengatur *client side script* dan *server side script*[21]. Javascript mendukung arsitektur *Model View Controller* (MVC) yang memelihara kode sehingga dapat dibaca dengan jelas dan memisahkan bagian-bagian dari kode program.

Javascript mendukung beberapa jenis dan tipe data seperti *string*, *integer*, *array*, *object*, dll. Selain itu, Javascript juga dapat digunakan untuk melakukan manipulasi dan interaksi terhadap komponen elemen HTML pada halaman web. Aktivitas seperti merubah data, mengubah bentuk atau fitur dari sebuah atribut, menambah atau menghapus sebuah elemen semuanya dimungkinkan dengan Javascript. Situasi seperti penanganan *error* atau peristiwa lain yang merupakan hasil dari suatu tindakan juga dapat dilakukan oleh Javascript. Fitur Javascript yang lengkap didukung dengan teknologi API yang semakin berkembang. API yang sering digunakan untuk Javascript seperti AJAX dan Fetch API juga ikut mendukung kinerja Javascript semakin optimal.

2.8 Dashboard

Dashboard dalam organisasi adalah suatu istilah dari sebuah sistem yang dapat memvisualisasikan atau menampilkan data yang berguna untuk pengambilan keputusan perusahaan. *Dashboard* adalah sebuah *tools* yang membantu meningkatkan kontrol atas berbagai macam data bisnis yang ada. *Dashboard* juga dapat membantu dalam menentukan

trend, pola, dan kejanggalan beserta alasan atas apa yang dilihat untuk membantu mengarahkan ke pengambilan keputusan yang efektif. *Dashboard* juga dapat diterapkan dalam sistem aplikasi MRP berbasis web untuk dapat menampilkan informasi data seperti jadwal produksi yang ada di perusahaan[22].

2.9 *Black Box Testing*

Metode pengujian ini merupakan salah satu metode pengujian yang mudah digunakan dan dilakukan. Metode ini hanya memerlukan batas bawah dan batas atas dari data yang diharapkan oleh penguji. Metode ini menguji perangkat lunak dari segi fungsionalitasnya tanpa menguji desain dan kode pemrograman. *Blackbox testing* menguji *expected output* berdasarkan data yang diinput tanpa melihat desain atau kode pemrograman suatu sistem. Tujuan dari dilakukannya *black box testing* adalah untuk mengetahui apakah fungsional dari perangkat lunak sudah sesuai dengan spesifikasi yang dibutuhkan.

Metode ini menguji suatu fungsi dari perangkat lunak dan mencocokkan apakah data yang dimasukkan dapat memberikan hasil yang sesuai dengan kebutuhan fungsi tersebut atau tidak. Metode ini juga dapat membantu dalam mengetahui kelemahan suatu sistem dari segi fungsional. Metode ini memungkinkan penguji memastikan bahwa data yang dihasilkan sesuai dengan data yang dimasukkan di suatu sistem[23].

2.10 *User Acceptance Testing (UAT)*

User Acceptance Testing (UAT) adalah sebuah metode pengujian yang ditujukan untuk pengguna dari sebuah sistem. UAT memiliki fokus utama yaitu terhadap uji coba atau *tutorial* pengguna terhadap sistem. Tujuan utama dari metode pengujian UAT adalah untuk menentukan apakah suatu sistem atau perangkat lunak layak digunakan. Metode UAT penting dilakukan pada suatu sistem atau perangkat lunak karena objek pengujian adalah pengguna dari sistem atau perangkat lunak. Hal ini memungkinkan untuk mendeteksi kekurangan dari sistem yang sebelumnya tidak dapat dilakukan oleh metode pengujian sistem seperti *black box testing*. UAT dilakukan menggunakan metode survei atau kuesioner yang ditujukan terhadap pengguna dari sistem atau perangkat lunak tersebut. Survei atau kuesioner ini akan memiliki skala nilai untuk mengukur tingkat kelayakan sistem mulai dari

skala 1 yang mengindikasikan pengguna sangat tidak setuju hingga skala nilai 5 yang mengindikasikan bahwa pengguna sangat setuju terhadap pernyataan penilaian tersebut[24].

Langkah awal dalam melakukan UAT adalah menentukan target pengguna sistem serta kegiatan fungsional yang akan dilakukan oleh masing-masing pengguna. Penguji akan memberikan *tutorial* terhadap pengguna terkait suatu fungsi pada sistem yang nantinya juga akan dilakukan oleh pengguna. Setelah tutorial dan pengguna melakukan apa yang dijelaskan pada *tutorial*, survei akan diberikan kepada pengguna. Survei ini berisikan pertanyaan terkait fungsi sistem dan menentukan apakah pengguna setuju dengan pernyataan tersebut atau tidak. Nilai inilah yang akan menjadi patokan dalam mengevaluasi sistem tersebut.

Jika nilai yang didapatkan mayoritas mengatakan tidak setuju maka sistem tersebut masih banyak kekurangannya dan dapat dikatakan tidak layak digunakan. Tujuan akhir dari UAT untuk memastikan pengguna setuju bahwa sistem tersebut sudah memenuhi kriteria penilaian yang tertera pada survei. UAT ini dilakukan untuk semua skenario yang dapat terjadi dalam sistem untuk memastikan pengguna dapat menggunakan sistem dengan baik dan layak sesuai dengan kebutuhan. Perhitungan persentase keberhasilan UAT secara individual dan keseluruhan menggunakan rumus *average* sebagai berikut[25].

1. Individual

$$\text{Persentase Nilai} = \frac{\text{Jumlah Pengujian yang Berhasil}}{\text{Total Jumlah Pengujian}} \times 100\%$$

2. Keseluruhan

$$\text{Persentase Keseluruhan} = \frac{\text{Total Hasil Persentase UAT (\%)}}{\text{Jumlah UAT yang Dilakukan}}$$

2.11 XAMPP dan Microsoft Visual Studio Code

2.11.1 XAMPP

XAMPP adalah sebuah aplikasi *web server* yang menyediakan basis data MySQL dan mendukung bahasa pemrograman PHP[26]. XAMPP juga merupakan solusi *open source software* yang bersifat *cross platform web server*. Selain MySQL dan PHP, XAMPP juga menyediakan fitur Apache HTTP *server*, dan bahasa pemrograman Perl. XAMPP dapat digunakan pada Windows, Linux, Solaris, dan Mac OS[27]. Pada penelitian ini, XAMPP akan digunakan sebagai *web server* untuk menghubungkan basis data dengan *website* sistem yang dibuat. XAMPP akan memungkinkan data yang diinput dari *website* dapat disimpan didalam basis data dan sebaliknya.

2.11.2 Microsoft Visual Studio Code

Visual Studio Code adalah salah satu aplikasi *code editor* yang memiliki fitur lengkap lingkungan pengembangan terpadu (IDE). Aplikasi ini juga mengakomodasi bahasa pemrograman seperti PHP, Javascript,C,C++.dll[28]. *Tools* ini juga bersifat *lightweight* dan memiliki fitur yang lengkap seperti fitur komprehensif revisi dan fitur utilitas kompilasi. Antarmuka dari *tools* ini juga memiliki tampak modern. Visual Studio juga menyediakan fitur seperti hubungan terhadap Git. Semua pengkodean baik *frontend* dan *backend* dilakukan menggunakan *tools* Visual Studio ini. Bahasa pemrograman yang digunakan dalam pengkodean menggunakan bahasa pemrograman PHP.

2.12 Penelitian Terdahulu

Tabel 2.4 Tabel Penelitian Terdahulu

Judul Artikel	Nama Artikel	Penulis/Tahun	Hasil	Kesimpulan
The Development of Web-based Sales	Ultima Infosys : Jurnal Ilmu	Suryasari, Jansen Wiratama, Ririn Ikana	Hasil dari penelitian ini adalah sebuah sistem informasi	Hal yang dijadikan referensi adalah metode evaluasi yang digunakan

Judul Artikel	Nama Artikel	Penulis/Tahun	Hasil	Kesimpulan
Reporting Information Systems using Rapid Application Development Method	Sistem Informasi	Desanti / 2023	berbasis <i>web</i> untuk kebutuhan <i>sales reporting</i> PT Artindo Pratama Sejahtera. Sistem ini akan dikembangkan menggunakan metode <i>Rapid Application Development</i> (RAD). Evaluasi sistem menggunakan evaluasi <i>blackbox testing</i> dan <i>User Acceptance Testing</i> (UAT) untuk uji fungsionalitas dan kelayakan sistem.	dalam mengevaluasi sistem akan menggunakan <i>blackbox testing</i> serta UAT untuk menguji fungsionalitas sistem.
Aplikasi Manajemen Proyek Berbasis Framework CodeIgniter dan Bootstrap di PT Pura Barutama.	Jurnal Informatika: Jurnal Pengembangan IT (JPIT)	Ramos Somya / 2018	Hasil dari penelitian ini untuk menghasilkan sebuah sistem aplikasi manajemen proyek untuk memudahkan perusahaan dalam melakukan <i>request</i> dan <i>monitoring</i> proyek. Aplikasi ini dibuat menggunakan <i>framework</i> CodeIgniter serta Bootstrap sebagai <i>framework front-end</i> dan <i>back-end code</i> .	Hal yang diadopsi dari artikel jurnal tersebut adalah penggunaan <i>framework</i> CodeIgniter dan Bootstrap sebagai <i>framework</i> utama <i>front-end</i> dan <i>back-end programming</i> .
Knowledge Management System pada PT. TASPEN KCU Bandung	Ultima Infosys : Jurnal Ilmu Sistem Informasi	Anggun Pratiwi dan Leonardi Paris Hasugian / 2018	Hasil penelitian ini berupa sebuah sistem <i>knowledge management</i> pada PT TASPEN KCU Bandung untuk meningkatkan pelayanan dan	Hal yang dapat dijadikan referensi yaitu penggunaan metode penelitian <i>System Development Life Cycle</i> yaitu <i>prototype</i> dalam

Judul Artikel	Nama Artikel	Penulis/Tahun	Hasil	Kesimpulan
			pengelolaan aset perusahaan. Peneliti menggunakan metode penelitian menggunakan <i>System Development Life Cycle (SDLC)</i> yaitu <i>Prototype</i> dalam mengembangkan sistem tersebut..	merencanakan dan mengembangkan sistem MRP.
RANCANG BANGUN APLIKASI PERENCANAAN BAHAN BAKU MENGGUNAKAN METODE MATERIAL REQUIREMENT PLANNING PADA PT MULTI KADERA SEJATI	Undergraduate thesis, Institut Bisnis dan Informatika Stikom Surabaya	MUHAMMAD RISQI KURNIAWAN / 2018	Hasil dari penelitian ini adalah sebuah aplikasi perencanaan bahan baku menggunakan metode <i>material requirement planning</i> PT Multi Kadera Sejati. Penelitian ini menjelaskan fitur atau fungsi utama yang terdapat dalam <i>material requirement planning</i> beserta solusi yang diberikan terhadap permasalahan yang ada di perusahaan yang diteliti.	Hal yang diadopsi dari penelitian tersebut adalah pemilihan fungsi utama sistem MRP serta menjadikan sistem yang dibuat dalam penelitian tersebut sebagai referensi sistem yang akan dibuat.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Judul Artikel	Nama Artikel	Penulis/Tahun	Hasil	Kesimpulan
Evaluasi Website Pemko Pekanbaru Menggunakan Metode Heuristic Evaluation	Skripsi thesis, Universitas Islam Negeri Sultan Syarif Kasim Riau	Faris Nazari / 2019	Hasil dari penelitian tersebut adalah nilai evaluasi kelayakan <i>website</i> Pemko Pekanbaru berdasarkan kriteria <i>website</i> , kualitas informasi, serta desain <i>web</i> yang efektif. Pengujian ini dilakukan untuk mengetahui seberapa baik <i>website</i> tersebut.	Hal yang diadopsi dari penelitian tersebut adalah kriteria dari <i>website</i> yang baik dan efektif. Kriteria ini akan menjadi patokan dalam perancangan sistem MRP berbasis <i>web</i> yang akan dilakukan.



Terdapat 4 artikel jurnal yang digunakan sebagai referensi penelitian ini, Penelitian pertama dilakukan untuk merancang sistem informasi *sales reporting* berbasis web untuk PT Artindo Pratama Sejahtera berdasarkan permasalahan dalam penumpukan *sales reporting document* yang terjadi di perusahaan. Metode penelitian yang digunakan adalah *system development lifecycle* (SDLC) berjenis *Rapid Application Development Method* (RAD) dan menggunakan metode *testing User Acceptance Testing* (UAT). Berdasarkan analisa dari keadaan sistem perusahaan yang sedang digunakan saat ini, didapatkan rekomendasi fitur yang dapat diterapkan dalam sistem baru. Hasil dari penelitian ini berupa sistem *sales reporting* berbasis web yang memuat rekomendasi fitur berdasarkan hasil analisa sistem sebelumnya. Fitur yang diberikan sistem seperti pengelolaan produk, pengelolaan penjualan, pelaporan penjualan, pemantauan target penjualan, dan pengelolaan dokumen penjualan. Hasil sistem tersebut mendapatkan nilai rata-rata sebesar 82.6% dari *User Acceptance Testing* yang dilakukan oleh 38 orang *salesperson* perusahaan[29].

Penelitian kedua dilakukan untuk merancang sistem manajemen proyek berbasis web pada PT Pura Barutama. Sistem ini dirancang berdasarkan permasalahan dalam membuat dan monitoring rancangan anggaran biaya serata surat perintah kerja yang dialami oleh perusahaan. Hasil penelitian ini berupa sistem aplikasi *material requirement planning* dengan fitur rekomendasi berfokus pada kegiatan pembuatan laporan perpindahan bahan baku dan pelaporan kegiatan produksi. Berdasarkan hasil pengujian yang dilakukan menggunakan metode *black box*, didapatkan bahwa seluruh fungsi yang diuji berhasil sesuai dengan harapan dari penguji[30].

Penelitian ketiga dilakukan untuk merancang sistem *knowledge management* untuk PT Taspen KCU Bandung. Sistem dirancang untuk meningkatkan pelayanan dan pengelolaan asset pengetahuan yang dimiliki perusahaan karena belum adanya sistem *knowledge management*. Metode penelitian yang digunakan dalam merancang sistem adalah metode *prototype* dan metode pengujian sistem menggunakan metode *black box*. Berdasarkan hasil evaluasi sistem yang sedang digunakan dan permasalahan yang ada maka direkomendasikan beberapa solusi untuk ditambahkan ke dalam sistem baru. Hasil dari penelitian ini berupa sistem aplikasi *knowledge management* yang memiliki fitur sesuai dengan solusi yang direkomendasikan yaitu fitur pengelolaan data aset terintegrasi antar

karyawan dan cabang perusahaan, fitur klaim kartu bantuan kesehatan dan SPPD, dan fitur *frequently ask question* (FAQ). Hasil dari pengujian yang didapatkan menggunakan pengujian *black box* menunjukkan semua skenario fungsi yang terdapat pada sistem bekerja sesuai dengan hasil yang diharapkan oleh penguji[31].

Penelitian keempat dilakukan untuk merancang sistem *material requirement planning* pada PT Multi Kadera Sejati. Sistem dirancang berdasarkan permasalahan perusahaan yaitu proses inventarisasi masih manual sehingga data yang ada kurang *up to date* dan penyampaian informasi ke PIC memakan waktu lama dan bahkan terlambat. Metode penelitian yang digunakan untuk merancang sistem adalah *waterfall model* dan metode pengujian sistem yang digunakan adalah metode *black box*. Hasil dari penelitian ini berupa sistem aplikasi *material requirement planning* dengan fokus pada kegiatan inventarisasi di perusahaan PT Multi Kadera Sejati. Sistem memiliki fitur yang sudah ditentukan berdasarkan hasil wawancara dan observasi. Fitur yang diberikan sistem seperti memasukkan barang, pengelolaan barang, dan kegiatan produksi. Berdasarkan hasil pengujian sistem menggunakan metode *black box* didapatkan hasil uji skenario fungsi sistem sesuai dengan yang diharapkan oleh penguji[32].

Penelitian kelima dilakukan untuk menganalisa dan mengevaluasi sistem berbasis web yang dimiliki oleh Pemerintah Kota Pekanbaru. Metode penelitian yang digunakan adalah metode *heuristic evaluation*. Tujuan penelitian dilakukan untuk mengetahui seberapa baik sistem berbasis web tersebut dari sisi usabilitynya. Menggunakan metode *heuristic evaluation*, pengukuran tingkat *usability* sistem dilakukan berdasarkan kriteria web yang efektif. Kriteria web efektif ditentukan berdasarkan aspek tampilan, fungsionalitas, dan manfaat yang diberikan. Hasil dari penelitian ini berupa nilai *usability* dari sistem tersebut sebesar 63% yang masuk kedalam kategori cukup baik[33].

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Dalam penelitian sebelumnya, dapat ditemukan keterkaitan antar penelitian satu dengan penelitian lainnya. Penelitian-penelitian tersebut sama-sama membahas tentang sistem informasi terutama sistem informasi berbasis web. Beberapa penelitian tersebut memiliki keterkaitan dalam melakukan perancangan sistem informasi baik berbasis web atau berbasis aplikasi. Penelitian kelima juga memiliki keterkaitan dengan empat penelitian lainnya meskipun tidak membahas perancangan sistem. Hal ini dapat dilihat dari kriteria website yang baik pada penelitian kelima dapat ditemukan pada pembahasan di keempat penelitian sebelumnya.

Penelitian ini mengadopsi beberapa hal dari penelitian sebelumnya seperti metode perancangan sistem yaitu *system development lifecycle prototype*. Selain itu, penelitian ini juga mengadopsi metode pengujian sistem yaitu *user acceptance testing* dan *black box* dalam mengukur fungsionalitas dari sistem yang sudah dibuat dan mendapatkan masukan dari pengguna. Framework CodeIgniter dan Bootstrap juga diadopsi dari penelitian sebelumnya untuk digunakan dalam penelitian ini.

Penelitian ini memiliki beberapa keterbaruan dari penelitian terdahulu. Keterbaruan ini dapat dilihat dari industri yang dijadikan obyek penelitian yaitu industri manufaktur yang berbeda dengan objek penelitian terdahulu. Selain itu, penelitian ini berfokus pada perencanaan bahan baku yaitu pengelolaan *bill of material* dan *manufacturing order*. Desain antarmuka serta fungsi yang dimiliki sistem juga dibuat khusus sesuai hasil wawancara dan observasi langsung dengan pihak perusahaan untuk memenuhi kebutuhan perusahaan.