

BAB II

PENELITIAN TERKAIT

2.1. Tinjauan Pustaka

Penelitian yang sudah ada sebelumnya menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperbanyak sekaligus memperluas cakupan teori-teori yang digunakan dalam mengkaji penelitian yang dilakukan.

Berdasarkan hasil penelitian [2] ditemukan bahwa sistem informasi *top up game* berbasis *website* menjadi solusi untuk mempercepat proses pemesanan dan penjualan produk serta mempermudah transaksi pembelian dan pembayaran bagi pelanggan. Dalam pengembangan sistem ini, peneliti menggunakan *framework Django* dan menggunakan aplikasi *Paypal* sebagai metode pembayaran yang aman dan dapat digunakan di pasar global. Kelebihan dari sistem ini adalah *customer* dapat membeli lebih dari 1 item, transaksi pembayaran bisa menggunakan aplikasi *Paypal* yang mencakup pembayaran di pasar global (*USD, EUR, JYP*, dll), dan aman terhadap akun *game* milik *customer*. Kekurangan dari sistem ini adalah membeli *item* yang berbeda tidak dapat dilakukan secara bersamaan. Perbedaan penelitian [2] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [2] menggunakan metode *waterfall*, sedangkan penelitian yang akan dilakukan menggunakan metode *Rapid Application Development*. Lalu, metode pembayaran penelitian [2] hanya menyediakan pembayaran melalui *Paypal*, sedangkan penelitian yang akan dilakukan menggunakan *Payment Gateway* dari *Tripay*.

Lalu, penelitian [3] menunjukkan bahwa dengan diterapkannya sistem *website* penjualan *voucher game*, dapat membantu pelanggan untuk mengetahui informasi *voucher game* yang dijual oleh *Good Gaming Store*. Metode yang digunakan dalam pengembangan *website* ini juga menggunakan model *waterfall*. Hasil survey yang dilakukan menunjukkan bahwa sebagian

besar responden menyatakan bahwa *website* ini dapat mempermudah proses pembelian, *posting voucher game*, keamanan dan transparansi data serta tampilan yang mudah dipahami. Kelebihan dari sistem ini adalah terdapat teknologi *Single Page Applications (SPA)* yang merupakan suatu paradigma aplikasi *web* yang menghindari *refresh* pada halaman web untuk menampilkan konten baru. Kekurangan dari sistem ini adalah pengguna atau pembeli harus melakukan daftar terlebih dahulu untuk melakukan transaksi. Perbedaan penelitian [3] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [3] menggunakan metode *waterfall*, sedangkan penelitian yang akan dilakukan menggunakan metode *Rapid Application Development*. Perbedaan lainnya penelitian [3] dengan penelitian yang akan dilakukan yaitu terletak pada subjek penelitian. Selain itu sistem ini mengharuskan pengguna melakukan daftar terlebih dahulu untuk melakukan transaksi.

Kemudian, penelitian [8] menunjukkan bahwa proses transaksi penjualan *voucher* menjadi efektif, karena di sistem ini sudah otomatis tersedia data *voucher* dan stok *voucher*. Proses transaksi sudah terkomputerisasi, sehingga dapat mempermudah dan mempercepat waktu pekerjaan dalam pengolahan data penjualan *voucher*. Antarmuka yang sederhana dapat mempermudah *user* dalam mempelajari sistem ini. Kelebihan dari sistem ini adalah terdapat fitur pembuatan laporan serta mencetaknya untuk diberikan kepada pemilik setiap sebulan sekali. Baik itu laporan stok *voucher game online*, data pegawai, penjualan *voucher game online*, maupun data pelanggan. Kekurangan dari sistem ini adalah masih dibutuhkannya *operator* atau pegawai untuk melakukan suatu transaksi. Perbedaan penelitian [8] dengan penelitian yang akan dilakukan yaitu pada penelitian [8] sistem transaksi berbasis *desktop* dan masih membutuhkan *operator* atau pegawai, sedangkan penelitian yang akan dilakukan berbasis *website* dan tidak membutuhkan *operator* atau pegawai dalam melakukan transaksi. Perbedaan lainnya penelitian [8] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [8] menggunakan metode

waterfall, sedangkan penelitian yang akan dilakukan menggunakan metode *Rapid Application Development*.

Selanjutnya penelitian [9] menunjukkan bahwa pengembangan aplikasi penjualan *Rolling Door* dengan menggunakan metode *RAD* membantu dalam memenuhi kebutuhan pengguna dengan cara menghemat waktu dalam proses merancang dan membangun aplikasi. Hasil pengujian menunjukkan bahwa sistem memiliki fitur navigasi yang mempermudah konsumen saat mengunjungi *e-commerce* dan melakukan proses pemesanan barang secara *online* tanpa dibatasi ruang dan waktu. Kelebihan dari sistem ini adalah terdapat fitur pembuatan laporan data order produk yang dapat disesuaikan pada tanggal tertentu. Kekurangan dari sistem ini adalah proses sistem pembayaran pada aplikasi *e-commerce* sistem penjualan masih dilakukan dengan sistem transfer secara manual melalui rekening Bank.

Berikutnya penelitian [10] menunjukkan bahwa berdasarkan hasil dari studi ini, dapat disimpulkan bahwa pengembangan aplikasi *Point of Sale* (POS) berbasis Android dapat dilakukan menggunakan metode *Rapid Application Development* (*RAD*). Periode pengembangan sistem selama 72 hari, dari 5 Maret 2020 hingga 18 Mei 2020, dengan pengembangan aplikasi *web* selama 22 hari, pengembangan *API* selama 15 hari, dan pengembangan aplikasi Android dan integrasi selama 35 hari. Hasil pengujian menunjukkan bahwa aplikasi berjalan dengan baik pada provider Telkomsel dengan *bandwith download* sekitar 39 Mbps dan *upload* sekitar 17 Mbps, saat pengujian dilakukan di Jalan Pasundan, Kota Samarinda. Kelebihan dari sistem ini adalah terdapatnya sistem berbasis *web* dan sistem berbasis Android sehingga memudahkan pengguna dalam menggunakannya. Kekurangan dari penelitian ini adalah dibutuhkannya arsitektur *microservices* agar aplikasi POS dapat melayani perubahan sistem yang cepat dan melayani transaksi yang jauh lebih banyak. Perbedaan penelitian [10] dengan penelitian yang akan dilakukan yaitu, objek penelitian [10] merupakan sistem *Point of Sale*, sedangkan penelitian yang akan dilakukan objeknya adalah sistem informasi transaksi *topup game* berbasis *website*.

Pada tabel 2.1 ini penulis memaparkan beberapa penelitian terkait rancang bangun *website* menggunakan metode *Rapid Application Development*.

Tabel 2.1 Penelitian Terdahulu

Peneliti	Metode	Hasil	Kelebihan dan Kekurangan	Perbedaan
R. N. Setiawan (2021)	<i>SDLC Waterfall</i>	Hasil penelitian [2] ditemukan bahwa sistem informasi <i>top up game</i> berbasis <i>website</i> menjadi solusi untuk mempercepat proses pemesanan dan penjualan produk serta mempermudah transaksi pembelian dan pembayaran bagi pelanggan. Dalam pengembangan sistem ini, peneliti menggunakan <i>framework Django</i> dan menggunakan aplikasi <i>Paypal</i> sebagai metode pembayaran yang aman dan dapat digunakan di pasar global.	Kelebihan dari sistem ini adalah <i>customer</i> dapat membeli lebih dari 1 item, transaksi pembayaran bisa menggunakan aplikasi <i>Paypal</i> yang mencakup pembayaran di pasar global (<i>USD, EUR, JYP</i> , dll), dan aman terhadap akun <i>game</i> milik <i>customer</i> . Kekurangan dari sistem ini adalah membeli <i>item</i> yang berbeda tidak dapat dilakukan secara bersamaan.	Perbedaan penelitian [2] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [2] menggunakan metode <i>waterfall</i> , sedangkan penelitian yang akan dilakukan menggunakan metode <i>Rapid Application Development</i> . Lalu, metode pembayaran penelitian [2] hanya menyediakan pembayaran melalui <i>Paypal</i> , sedangkan penelitian yang akan dilakukan menggunakan <i>Payment Gateway</i> dari <i>Tripay</i> .

Peneliti	Metode	Hasil	Kelebihan dan Kekurangan	Perbedaan
F. Setiawan dan H. Zakaria (2022)	<i>SDLC Waterfall</i>	Hasil penelitian [3] menunjukkan bahwa dengan diterapkannya sistem <i>website</i> penjualan <i>voucher game</i> , dapat membantu pelanggan untuk mengetahui informasi <i>voucher game</i> yang dijual oleh Good Gaming Store. Metode yang digunakan dalam pengembangan <i>website</i> ini juga menggunakan model <i>waterfall</i> . Hasil survey yang dilakukan menunjukkan bahwa sebagian besar responden menyatakan bahwa <i>website</i> ini dapat mempermudah proses pembelian, <i>posting voucher game</i> , keamanan dan transparansi data serta tampilan yang mudah dipahami.	Kelebihan dari sistem ini adalah terdapat teknologi <i>Single Page Applications (SPA)</i> yang merupakan suatu paradigma aplikasi <i>web</i> yang menghindari <i>refresh</i> pada halaman <i>web</i> untuk menampilkan konten baru. Kekurangan dari sistem ini adalah pengguna atau pembeli harus melakukan daftar terlebih dahulu untuk melakukan transaksi.	Perbedaan penelitian [3] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [3] menggunakan metode <i>waterfall</i> , sedangkan penelitian yang akan dilakukan menggunakan metode <i>Rapid Application Development</i> . Perbedaan lainnya penelitian [3] dengan penelitian yang akan dilakukan yaitu terletak pada subjek penelitian. Selain itu sistem ini mengharuskan pengguna melakukan daftar terlebih dahulu untuk melakukan transaksi.

Peneliti	Metode	Hasil	Kelebihan dan Kekurangan	Perbedaan
A. Ananditya, Sriyono, dan S. Yani (2020)	<i>SDLC Waterfall</i>	Hasil penelitian [8] menunjukkan bahwa proses transaksi penjualan <i>voucher</i> menjadi efektif, karena di sistem ini sudah otomatis tersedia data <i>voucher</i> dan <i>stok voucher</i> . Proses transaksi sudah terkomputerisasi, sehingga dapat mempermudah dan mempercepat waktu pekerjaan dalam pengolahan data penjualan <i>voucher</i> . Antarmuka yang sederhana dapat mempermudah <i>user</i> dalam mempelajari sistem ini.	Kelebihan dari sistem ini adalah terdapat fitur pembuatan laporan serta mencetaknya untuk diberikan kepada pemilik setiap sebulan sekali. Baik itu laporan <i>stok voucher game online</i> , data pegawai, penjualan <i>voucher game online</i> , maupun data pelanggan. Kekurangan dari sistem ini adalah masih dibutuhkannya <i>operator</i> atau pegawai untuk melakukan suatu transaksi.	Perbedaan penelitian [8] dengan penelitian yang akan dilakukan yaitu pada penelitian [8] sistem transaksi berbasis <i>desktop</i> dan masih membutuhkan operator atau pegawai, sedangkan penelitian yang akan dilakukan berbasis <i>website</i> dan tidak membutuhkan operator atau pegawai dalam melakukan transaksi. Perbedaan lainnya penelitian [8] dengan penelitian yang akan dilakukan yaitu terletak pada metode penelitian [8] menggunakan metode <i>waterfall</i> , sedangkan penelitian yang akan dilakukan

Peneliti	Metode	Hasil	Kelebihan dan Kekurangan	Perbedaan
				menggunakan metode <i>Rapid Application Development</i> .
M. Hasanudin, H. M. Khoirudin, W. A. Amroni, dan S. Silen (2019)	<i>Rapid Application Development</i>	Pengembangan aplikasi penjualan <i>Rolling Door</i> dengan menggunakan metode <i>RAD</i> membantu dalam memenuhi kebutuhan pengguna dengan cara menghemat waktu dalam proses merancang dan membangun aplikasi. Hasil pengujian menunjukkan bahwa sistem memiliki fitur navigasi yang mempermudah konsumen saat mengunjungi <i>e-commerce</i> dan melakukan proses pemesanan barang secara <i>online</i> tanpa dibatasi ruang dan waktu.	Kelebihan dari sistem ini adalah terdapat fitur pembuatan laporan data order produk yang dapat disesuaikan pada tanggal tertentu. Kekurangan dari sistem ini adalah proses sistem pembayaran pada aplikasi <i>e-commerce</i> sistem penjualan masih dilakukan dengan sistem transfer secara manual melalui rekening Bank.	Perbedaan penelitian [9] dengan penelitian yang akan dilakukan yaitu pada penelitian [9] masih menggunakan metode pembayaran secara manual, sedangkan penelitian yang akan dilakukan telah menggunakan <i>payment gateway</i> sehingga pembayaran bisa dilakukan secara otomatis.
E. Junirianto dan N.	<i>Rapid Application Development</i>	Berdasarkan hasil dari studi ini, dapat disimpulkan bahwa pengembangan aplikasi <i>Point of Sale (POS)</i> berbasis	Kelebihan dari sistem ini adalah terdapatnya sistem berbasis <i>web</i> dan sistem berbasis Android	Perbedaan penelitian [10] dengan penelitian yang akan dilakukan yaitu, objek

Peneliti	Metode	Hasil	Kelebihan dan Kekurangan	Perbedaan
Kurniadin (2020)		Android dapat dilakukan menggunakan metode <i>Rapid Application Development (RAD)</i> . Periode pengembangan sistem selama 72 hari, dari 5 Maret 2020 hingga 18 Mei 2020, dengan pengembangan aplikasi <i>web</i> selama 22 hari, pengembangan <i>API</i> selama 15 hari, dan pengembangan aplikasi Android dan integrasi selama 35 hari. Hasil pengujian menunjukkan bahwa aplikasi berjalan dengan baik pada provider Telkomsel dengan <i>bandwith download</i> sekitar 39 Mbps dan <i>upload</i> sekitar 17 Mbps, saat pengujian dilakukan di Jalan Pasundan, Kota Samarinda.	sehingga memudahkan pengguna dalam menggunakannya. Kekurangan dari penelitian ini adalah dibutuhkan arsitektur <i>microservices</i> agar aplikasi POS dapat melayani perubahan sistem yang cepat dan melayani transaksi yang jauh lebih banyak.	penelitian [10] merupakan sistem <i>Point of Sale</i> , sedangkan penelitian yang akan dilakukan objeknya adalah sistem informasi transaksi <i>topup game</i> berbasis <i>website</i> .

2.2. Landasan Teori

Terdapat beberapa dasar teori yang berkaitan dengan Perancangan Sistem Informasi Transaksi pada *Evo Game Store* berbasis *website* menggunakan metode *Rapid Application Development*.

2.2.1. Sistem Informasi

Sistem informasi merupakan sebuah kombinasi dari manusia, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, sumber data, kebijakan, dan prosedur yang memiliki kemampuan untuk mengumpulkan, menyimpan, mengubah, dan menyebarkan informasi dari sebuah organisasi[11]. Sistem Informasi adalah suatu sistem di dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi, dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan[12].

Dengan kata lain, sistem informasi dapat diartikan seperti susunan dari beberapa komponen yaitu manusia, data, alur, dan teknologi informasi yang berkaitan erat untuk dilakukannya pengumpulan, pemrosesan, penyimpanan, dan penyediaan output informasi yang dibutuhkan untuk menunjang kegiatan suatu lembaga. Sedangkan, Sistem Informasi Transaksi adalah sub sistem informasi yang mencakup kumpulan prosedur yang melaksanakan, mencatat, mengkalkulasi, membuat dokumen dan informasi transaksi untuk keperluan manajemen mulai dari diterimanya order penjualan sampai mencatat timbulnya tagihan dagang [13].

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa komponen-komponen dari sistem informasi transaksi secara umum terdiri dari:

1. Pencatatan transaksi
2. Pengecekan stok barang
3. Kalkulasi jumlah harga
4. Pencetakan nota transaksi
5. Pembuatan dokumen atau informasi penjualan untuk keperluan manajemen [13].

2.2.2. *Top-Up Game*

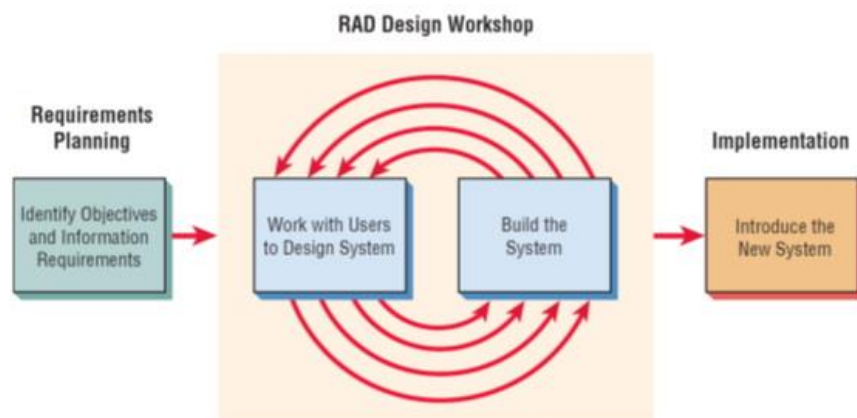
Topup Game termasuk ke dalam Micro-transaction yang merupakan transaksi keuangan yang dilakukan secara online. *Top-up Game* merupakan sebuah proses membeli atau menambahkan uang virtual untuk dapat memainkan sebuah game [14]. Uang virtual tersebut digunakan untuk membeli item atau membuka fitur dalam game. *Top-up game* bisa dilakukan melalui beberapa metode seperti transfer bank, pulsa, atau kartu kredit.

2.2.3. *Website*

Website yakni layanan yang didapati oleh pemakai komputer yang terhubung dengan jaringan internet. *Website* merupakan aplikasi tertentu yang berjalan di atas platform atau operation system browser. Dengan demikian *website* yang dimaksud dalam penelitian ini berarti sebuah halaman informasi yang tersedia secara online dan dapat diakses di seluruh dunia selama tersambung dengan jaringan internet. *Website* pada umumnya terdiri dari format teks, gambar, tabel, grafik, kutipan, video, musik, dan format visual lainnya yang menarik bagi pengunjung *website* tersebut[15]. *Website* saat ini dinilai sangat efektif bila digunakan untuk media informasi data, hal ini dikarenakan *website* memiliki bagian-bagian penting seperti visualisasi gambar dan teks yang membuat para pencari data dan informasi merasa lebih baik.

2.2.4. Metode *Rapid Application Development (RAD)*

Pada penelitian rancang bangun sistem informasi ini, peneliti menggunakan metode *Rapid Application Development*. Tahapan dari model pengembangan *Rapid Application Development* digambarkan pada gambar 2.1



Gambar 2.1 Metode *Rapid Application Development* [16]

RAD atau *Rapid Application Development* adalah model pembangunan perangkat lunak yang tergolong dalam teknik bertingkat yang inkremental. *RAD* menekankan pada siklus pembangunan cepat, pendek dan singkat, dan waktu yang singkat adalah batasan penting untuk model tersebut [5]. Dalam mengembangkan sistem, *RAD* menggunakan metode iteratif dengan membangun working model sistem pada awal tahap pengembangan untuk menetapkan kebutuhan pengguna [10]. Terdapat beberapa langkah yang dilakukan dalam menggunakan model *Rapid Application Development*, yaitu sebagai berikut:

1. *Requirements Planning* (Perencanaan Kebutuhan)

Tahap ini peneliti memerlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi

dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan dalam penelitian.

2. *RAD Design Workshop*

Pada tahap ini, partisipasi aktif dari pengguna yang terlibat sangat kritikal untuk mencapai tujuan, karena di sini terjadi proses desain dan perbaikan desain yang berulang sampai sesuai dengan kebutuhan pengguna yang telah dikenali sebelumnya. Hasil dari tahap ini adalah spesifikasi perangkat lunak yang mencakup dalam sistem, struktur data, dan lain sebagainya.

3. *Implementation*

Tahapan implementasi adalah saat peneliti mempraktikkan desain sistem yang telah disetujui sebelumnya. Sebelum sistem diimplementasi, pengujian dilakukan pada sistem untuk menemukan kesalahan. Tahap ini biasanya mengakibatkan tanggapan mengenai sistem yang sudah dibuat dan memperoleh persetujuan terhadap sistem tersebut.

2.2.5. Bahasa pemrograman

Bahasa pemrograman adalah bahasa yang digunakan oleh *programer* untuk memberikan perintah atau instruksi-instruksi kepada komputer dalam melakukan suatu kegiatan dan menentukan cara kerja yang tepat [17]. Terdapat berbagai jenis bahasa pemrograman yang digunakan untuk tujuan penggunaan yang berbeda-beda, seperti pemrograman untuk merancang aplikasi, pemrograman untuk merancang sistem, pemrograman untuk merancang perangkat keras, dan lain-lain. Setiap bahasa pemrograman memiliki ciri khas sintaks dan struktur yang unik, yang membuatnya lebih cocok untuk digunakan dalam konteks penggunaan tertentu.

Terdapat macam-macam contoh bahasa pemrograman yang populer saat ini dan banyak digunakan khususnya dalam perancangan sebuah perangkat lunak yaitu:

1. *PHP*

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman open-source yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi *HTML*. Bahasa *PHP* dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. *PHP* merupakan bahasa *scripting server-side*, dimana pemrosesan datanya dilakukan pada sisi server [18].

2. *Javascript*

JavaScript adalah bahasa pemrograman yang digunakan untuk menambahkan interaksi dan dinamisme pada halaman web. Ini adalah bahasa skrip yang berjalan di sisi klien (*client-side*) dan dapat digunakan untuk mengontrol elemen *HTML* dan *CSS* serta melakukan operasi logika dan matematika. *Client-side* melakukan prosesnya pada *web browser* sehingga dapat menghasilkan suatu tampilan *web* yang menarik [19].

2.2.6. *Database*

Database (basis data) adalah sekumpulan data-data yang tersimpan di dalam komputer secara terorganisir sehingga dapat diakses dengan menggunakan suatu program komputer untuk mendapatkan informasi dari basis data tersebut [20]. Pada penelitian ini peneliti menggunakan basis data *MySQL*.

MySQL adalah sebuah sistem manajemen basis data (*database management system, DBMS*) yang terkenal dan banyak digunakan. *MySQL* dikembangkan oleh perusahaan Swedia, *MySQL AB*, yang sekarang telah menjadi bagian dari *Oracle Corporation*. *MySQL*

merupakan *DBMS* yang menggunakan struktur database relasional dan menggunakan bahasa *SQL (Structured Query Language)* sebagai bahasa untuk mengakses dan memanipulasi data yang tersimpan di dalam database. *MySQL* dapat digunakan secara gratis dan *open source* di banyak sistem operasi, seperti *Windows, Linux, dan MacOS* [21]. *MySQL* banyak digunakan dalam aplikasi web, karena kemampuannya untuk menangani jumlah koneksi yang besar dengan cepat dan mudah di-install.

Pengoperasian database *MySQL* dapat dengan menggunakan aplikasi atau perangkat lunak yang disediakan, salah satu aplikasi atau perangkat lunak yang banyak digunakan pada saat ini ialah *PhpMyAdmin*. *PhpMyAdmin* merupakan perangkat lunak yang dapat digunakan secara gratis untuk melakukan sebuah pengoperasian terhadap database *MySQL* baik dalam membuat, mengelola, atau melakukan hal lain pada database [21].

2.2.7. PHP Storm

Php Storm adalah sebuah *Integrated Development Environment (IDE)* yang efisien dan pintar yang fokus pada efisiensi para pengembang. PHP Storm menyediakan solusi kode yang bijak, navigasi yang cepat dan mendeteksi kesalahan saat proses pengembangan. Selalu siap untuk membantu dalam menyusun kode, menjalankan uji unit, atau memberikan fitur debugging yang visual. Php Storm juga mencakup semua fitur yang terdapat di *WebStorm* seperti editor *HTML/CSS* dan editor *javascript* serta menambahkan dukungan lengkap untuk bahasa pemrograman *PHP* [22].

2.2.8. Laragon

Laragon merupakan perangkat lunak *open-source* yang tersedia dalam berbagai sistem operasi. Laragon digunakan sebagai server virtual atau localhost, dan dapat menggunakan domain sesuai

keinginan atau *pretty URL's*. Aplikasi ini sangat baik untuk pengelolaan aplikasi berbasis *website* [23].

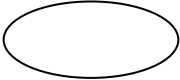
2.2.9. Unified Modelling Language (UML)

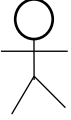


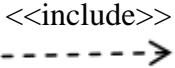
Pemodelan suatu sistem diperlukan pengembang untuk dapat memudahkan dalam pengembangan atau perancangan suatu sistem. Salah satu standar bahasa yang paling banyak digunakan saat ini yaitu *Unified Modelling Language*. *Unified Modelling Language (UML)* banyak digunakan oleh pengembang sistem khususnya dalam metodologi berorientasi objek. *UML* dapat digunakan untuk pemodelan secara visual, penggambaran, pembuatan, dan membuat desain pada sistem [24]. Dalam melakukan penelitian, peneliti menggunakan 3 *UML*, diantaranya:

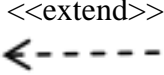
1. Use case Diagram

Use case diagram ialah suatu diagram yang menunjukkan perilaku atau interaksi antara sistem dengan pengguna (aktor). *Use case* diagram dapat menjelaskan bagaimana interaksi yang dilakukan aktor kepada sistem maupun sebaliknya, serta dengan adanya *use case* ini pengembang dapat mengetahui apa saja fungsi yang nantinya dilakukan oleh sistem dan siapa saja yang dapat menggunakan sistem tersebut [25]. Terdapat beberapa simbol yang digunakan dalam membuat sebuah *use case* diagram, yaitu:

Tabel 2.2 Komponen *Use case Diagram* [26]

Simbol	Nama	Keterangan
	<i>Use case</i>	Rangkaian langkah-langkah yang diperlihatkan oleh suatu sistem yang menghasilkan hasil yang dapat diukur bagi seorang aktor.


	<i>Actor</i>	Aktor menentukan sekelompok peran yang mereka ambil ketika terlibat dalam interaksi dengan use case
	<i>Association</i>	Asosiasi merupakan sebuah penggambaran dari penghubung interaksi antara aktor dan sistem.
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	<i>Include</i> menggambarkan bahwa <i>use case</i> sepenuhnya termasuk fungsionalitas dari <i>use case</i> yang lain.

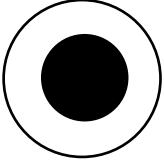

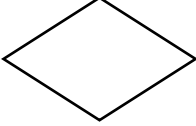

	<i>Extend</i>	<i>Extend</i> merupakan penggambaran jika sebuah <i>use case</i> adalah tambahan fungsionalitas dari <i>use case</i> yang lain.
---	---------------	---

2. Activity Diagram

Activity diagram yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari *use case* yang memiliki alur aktivitas. Alur atau aktivitas berupa bisa berupa runtutan menu-menu atau proses bisnis yang terdapat di dalam sistem tersebut. Diagram aktivitas tidak menjelaskan kelakuan aktor. Diartikan bahwa dalam pembuatan *activity diagram* hanya dapat dipakai untuk menggambarkan alur kerja atau aktivitas sistem saja [27]. Terdapat simbol-simbol yang digunakan dalam pemodelan *activity diagram*, diantaranya:

Tabel 2.3 Komponen *Activity Diagram* [26]

Gambar	Nama	Keterangan
	<i>Start Point</i>	Merupakan penggambaran untuk awal dari aktifitas.


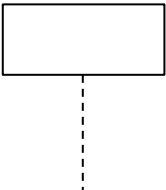
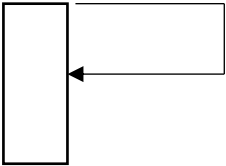
	<i>End Point</i>	Biasanya digunakan untuk akhir dari proses aktifitas.
	<i>Activities</i>	Digunakan untuk menggambarkan suatu aktivitas yang terjadi pada sistem
	<i>Decision Points</i>	Digunakan untuk penggambaran proses pengambilan keputusan antara <i>true</i> atau <i>false</i> .
	<i>Swimlane</i>	Merupakan proses pembagian yang digunakan untuk menentukan siapa dan melakukan apa.

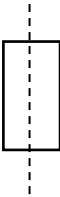
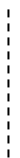

3. Sequence Diagram

Sequence Diagram termasuk diagram *UML* yang berfungsi untuk merepresentasikan visual dari tindakan yang terjadi secara berurutan dalam sistem yang diilustrasikan dalam bentuk diagram yang menunjukkan urutan secara rasional perubahan yang terjadi dalam proses yang menghasilkan sesuatu sesuai dengan *use case diagram*. *Sequence diagram* adalah diagram yang menggambarkan interaksi antara objek atau komponen dalam sistem yang

ditampilkan dalam urutan waktu. Diagram ini menggambarkan tindakan yang terjadi pada objek atau komponen, bagaimana tindakan tersebut terjadi, dan siapa yang melakukannya. Diagram ini juga menggambarkan bagaimana objek atau komponen tersebut berkomunikasi satu sama lain untuk mencapai tujuan tertentu. Sequence diagram sangat berguna untuk menganalisis dan menentukan urutan tindakan yang terjadi dalam suatu sistem, sehingga dapat membantu dalam proses desain sistem[28]. Beberapa simbol yang digunakan dalam penggambaran sequence diagram diantaranya:

Tabel 2.4 Komponen *Sequence Diagram* [26]


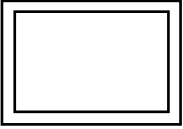
Simbol	Nama	Keterangan
	<i>Actor Lifeline</i>	Menggambarkan sebuah entitas yang berdiri sendiri dan berada diluar sistem dengan tujuan untuk berinteraksi dengan sistem
	<i>Object Lifeline</i>	Menggambarkan entitas yang berdiri sendiri dan berada dalam diagram
	<i>Recursive</i>	Memiliki fungsi untuk melakukan pengiriman proses terhadap dirinya sendiri

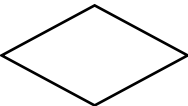
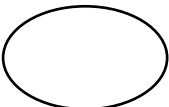
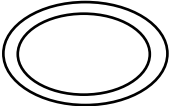

	<i>Activation</i>	Memiliki fungsi untuk mewakili sebuah eksekusi proses yang dilakukan oleh objek
	<i>Lifeline</i>	Digunakan sebagai penghubung antar objek
	<i>Message</i>	Merupakan sebuah simbol yang berfungsi untuk mengirimkan pesan antar kelas

2.2.10. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah sebuah diagram struktural yang digunakan untuk merancang sebuah database. Sebuah *ERD* mendeskripsikan data yang akan disimpan dalam sebuah sistem maupun batasannya. Komponen utama yang terdapat di dalam sebuah *ERD* adalah *entity set*, *relationship set*, dan *juga constraints* [29]. *ERD* ini dapat digambarkan secara sistematis dengan menggunakan simbol-simbol seperti yang terlihat pada tabel 2.6.

Tabel 2.5 Komponen *Entity Relationship Diagram* [30]

Simbol	Nama	Keterangan
	<i>Entity</i>	Entitas adalah objek atau konsep yang ingin Anda simpan informasinya.
	<i>Weak entity</i>	Entitas yang lemah adalah entitas yang harus ditentukan oleh hubungan kunci asing dengan entitas lain karena tidak dapat diidentifikasi

		secara unik dengan atributnya sendiri.
	<i>Relationship</i>	Relasi adalah Tindakan, yang diwakili oleh bentuk intan, menunjukkan bagaimana dua entitas berbagi informasi dalam <i>Database</i> .
	<i>Attribute</i>	Atribut kunci adalah karakteristik entitas yang unik dan berbeda.
	<i>Attribute Multinilai</i>	Atribut multivalued dapat memiliki lebih dari satu nilai.
	<i>Atribut Turunan</i>	Atribut turunan didasarkan pada atribut lain.

2.2.11. *Black Box Testing*

Pengujian *black box* merupakan salah satu jenis metode pengujian yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para penguji memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenai proses testing dibagian luar [31]. Pada *black box testing*, perangkat lunak tersebut akan dieksekusi kemudian berusaha dites apakah telah memenuhi kebutuhan pengguna yang didefinisikan pada saat awal tanpa harus membongkar *listing* program.

Pengujian dengan menggunakan *black box* mempunyai beberapa teknik, di antaranya *Equivalence Partitioning*, *Boundary Value Analysis*, *Robustness Testing*, *Behavior Testing*, dan *Cause-*

Effect Relationship Testing [32]. Pada penelitian ini penulis menggunakan Teknik *Equivalence Partitioning*.

Equivalence partitioning merupakan sebuah metode pengujian yang dirancang dengan memeriksa masukan dan keluaran data yang dikelompokkan berdasarkan fungsinya baik itu valid maupun tidak valid. *Equivalence partitioning* berusaha untuk mendefinisikan *test case* berdasarkan pada evaluasi dari ekuivalensi jenis atau *class* untuk kondisi input berupa nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi *Boolean* [33].