

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Dalam penyusunan penelitian ini, penulis terinspirasi dari penelitian-penelitian sebelumnya yang berkaitan dengan latar belakang masalah pada penelitian ini.

Penelitian yang dilakukan oleh Heri Purwan, dkk (2021) yakni tentang Sistem Informasi Penyewaan Lapangan Futsal Berbasis Web. Pembuatan pada aplikasi ini memakai metodologi waterfall. Futsal merupakan olahraga yang digemari oleh hampir semua kalangan, namun masih kebanyakan di tempat Futsal tidak mempunyai Website untuk mempermudah proses penyewaan dari jarak jauh maupun jarak dekat. Pembuatan sistem data web futsal ini diimplementasikan memakai Visual Studio Code, bahasa pemrograman PHP, database MySQL selaku buat penyimpanan informasi. [1]

Penelitian yang dilakukan oleh Geubrina Rizka Utami Sinaga dan Samsudin (2021) yakni tentang Implementasi Framework Laravel dalam Sistem Reservasi pada Restoran Cindelas Kota Medan. Penerapan sistem ini menggunakan pengembangan Waterfall dan menggunakan tools UML dibangun dengan bahasa pemrograman PHP serta MySQL sebagai database server dan Laravel sebagai frameworknya. Restoran merupakan salah satu jenis usaha pangan dilengkapi dengan penyajian dan penjualan makanan dan minuman. Reservasi merupakan aktifitas memesan tempat atau barang, reservasi dan restoran sering dikaitkan karena pemesanan tempat dan menu mempermudah

pelanggan restoran sehingga pilihan reservasi menjadi pilihan yang tepat untuk pelanggan yang ingin memesan terlebih dahulu dan akan datang setelahnya. Restoran Ayam Penyet Cindelaras merupakan salah satu restoran yang populer bagi masyarakat jika ingin melakukan reservasi, masyarakat atau pelanggan yang melakukan reservasi pada restoran cindelaras masih menggunakan cara manual, yaitu dengan mendatangi restoran. Hal ini masih belum efektif dan efisien mengingat tidak semua pelanggan yang melakukan reservasi dapat meluangkan waktu untuk sekedar datang melakukan reservasi. Oleh sebab itu, diperlukan sebuah sistem informasi pemesanan yang mampu mewujudkan reservasi yang efektif dan efisien. Sistem Informasi Layanan Pemesanan ini menyajikan cara reservasi tempat dan menu yang ada pada restoran dan cara pembayaran melalui moneytransfer / m-banking.[2]

Penelitian yang dilakukan oleh Ersi Putri Utami dan Afrizal Zein (2023) yakni tentang Perancangan Sistem Informasi Reservasi Meja Kafe Menggunakan Metode Rad Rapid Application Development Berbasis Web (Studi Kasus : Cafeteria Citra Sawangan Depok. Perancangan dan pengembangan sistem informasi aplikasi ini dibangun menggunakan metode RAD Rapid Application Development untuk mengurangi atau mempersingkat waktu yang dibutuhkan untuk membuat atau mendesain dan menyebarkannya dalam tiga tahap yang terdiri dari tahap perencanaan, tahap perancangan dan pengembangan sistem, dan tahap implementasi. Pada era globalisasi sekarang ini para pedagang bisnis makanan memiliki sistem online yang tepat untuk mempromosikan

produknya. Dengan melalui promosi, konsumen atau pelanggan mengetahui peluang bisnis yang nyata, produk saat ini yang sedang berkembang pesat.

Dengan terus mengembangkan, merancang, mensimulasikan, dan melaksanakan promosi online. Berdasarkan permasalahan di atas maka penulis memberi solusi untuk mendirikan sebuah rancangan dan pengembangan sistem informasi aplikasi berbasis website agar menunjang seperti promosi dan dapat melakukan pemesanan tempat atau reservasi secara on-line. Hasil dari aplikasi ini akan menghasilkan keluaran dan hasil berupa fasilitas, latar belakang restoran, informasi seperti menu dan harga makanan, alamat dan lokasi restoran yang terhubung dengan Google Maps, cara pemesanan atau reservasi, dan laporan penjualan. Dengan dibuatnya sistem informasi promosi dan pemesanan atau reservasi dapat membantu pemilik restoran mempromosikan menu dan harga serta lokasi makanan, serta memudahkan pelanggan untuk menemukan restoran di daerah tersebut. Hasil yang diharapkan dari perancangan sistem informasi berbasis web untuk mempermudah dan lebih efisien mengenai waktu dalam melakukan reservasi secara on-line. Tanpa harus tatap muka dan juga dapat memberi akses kepada pelanggan agar mengetahui profile tempat, juga sebagai sarana promosi yang ada didalam sebuah web tersebut.[3]

2.2 Landasan Teori

2.2.1. Sistem

Sistem adalah suatu kelompok unsur yang memiliki hubungan erat satu dengan yang lain yang mempunyai tujuan yang sama. Suatu sistem dibuat untuk mengatasi sesuatu yang terjadi berulang kali. Pendekatan sistem dengan menekankan pada suatu komponen akan lebih mudah digunakan untuk menganalisis dan melakukan perancangan. Sistem dibuat untuk maksud tertentu.

Ada yang mengatakan suatu sistem dibuat untuk tujuan tertentu (*goal*) dan ada juga yang mengatakan sistem dibuat untuk mencapai suatu sasaran (*bjectives*). Tujuan ditujukan untuk ruang lingkup yang luas sementara sasaran untuk yang lebih sempit.[4]

2.2.2. Booking

Booking atau pemesanan merupakan tindakan untuk mengamankan suatu tempat atau layanan. Ketika melakukan *booking* artinya orang tersebut telah berkomitmen untuk menggunakan tempat atau layanan sesuai dengan tanggal yang telah ditentukan. Setelah melakukan *booking*, seseorang akan memiliki hak resmi untuk tempat atau layanan tersebut. Pembatalan *booking* biasanya akan memberi denda (biaya pembatalan) atau bahkan kehilangan seluruh pembayaran sesuai dengan kebijakan pada tempat tersebut.

2.2.3. Billiard

Billiard berasal dari kata “*billiard*” yang artinya tongkat kayu atau “*bille*” yang artinya bola. *Billiard* adalah olahraga yang dimainkan dengan tongkat untuk menyodok bola-bola kecil yang akan di masukkan ke dalam lubang. Menurut KBBI (Kamus Besar Bahasa Indonesia) tahun 2016, definisi “*biliar*” adalah permainan yang menggunakan bola (kecil) dari gading dan sebagainya dan tongkat panjang di atas meja persegi panjang yang berlapis kain laken.

Billiard memiliki banyak jenis permainan, namun umumnya hanya ada dua yang dipertandingkan. Jumlah bola menjadi pembeda dari kedua jenis permainan itu (Sutanto, 2016: hal 59). Permainan 8 (Delapan) Bola atau 8 Ball dan Permainan 9 (Sembilan) Bola atau 9 Ball.

Ada beberapa teknik yang bisa dilakukan dalam permainan olahraga *billiard*. Teknik ini dilakukan pada saat dalam situasi tertentu yang bisa membuat bola putih melaju sesuai keinginan. Berikut merupakan teknik-teknik dasar dalam olahraga *billiard*.

1. Stop shot adalah teknik yang membuat bola putih berhenti saat mengenai bola target. Teknik ini bisa dilakukan dengan menyodok bagian tengah pada bola putih.
2. Draw shot adalah sebuah teknik yang memungkinkan pemain untuk membuat bola putih mundur ke kiri, ke kanan atau ke tempat semula saat bola putih mengenai bola target. Teknik ini dapat dilakukan dengan menyodok bagian sedikit di bawah bola putih.
3. Follow shot adalah teknik yang memungkinkan pemain untuk membuat bola putih melaju mengikuti bola target setelah mengenai sasaran. Teknik ini dapat dilakukan dengan menyodok bagian atas bola putih.
4. Spin shot adalah yang memungkinkan pemain untuk membuat bola putih berputar melengkung ke kanan atau kiri. Teknik ini dapat dilakukan dengan menyodok bagian atas kanan atau kiri bola putih.
5. Jump shot merupakan teknik yang biasa dilakukan bila bola target terhalang dengan bola lain. Teknik ini biasa dilakukan dengan tongkat khusus. Teknik ini bisa membuat bola putih melambung melewati bola yang menghalangi bola target. Teknik ini juga merupakan teknik yang membutuhkan keahlian yang sangat tinggi dari seorang pemain. Teknik ini bisa dilakukan dengan memukul bola putih di bagian depan paling atas.

Walaupun olahraga ini banyak dianggap negatif bagi yang mendengarnya karena olahraga ini banyak ditemukan di tempat-tempat bar dan sebagainya. Namun semakin mendalami olahraga ini membuat tingkat konsentrasi semakin tinggi sehingga kemampuan otak semakin meningkat. Karena saat bermain *billiard* jumlah oksigen dalam darah meningkat sehingga aliran darah ke otak dipercepat.[5]

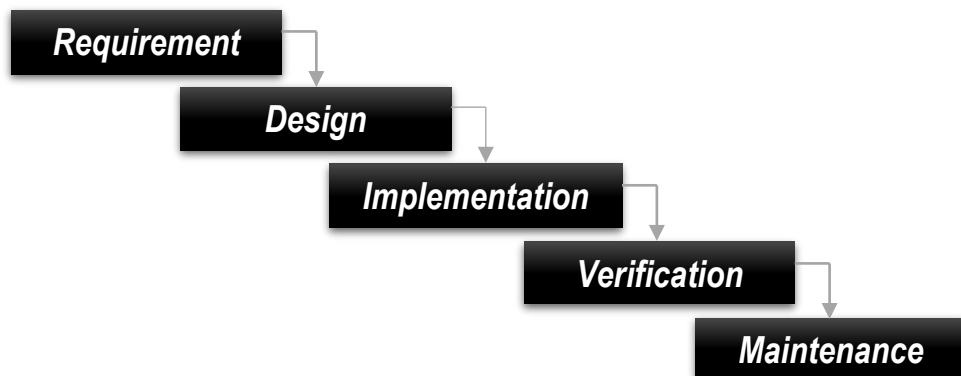
2.2.4. Metode Waterfall

Penelitian ini menggunakan Metode Waterfall SDLC (*System Development Life Cycle*) atau Metode Air Terjun dalam Rekayasa Perangkat Lunak. SDLC atau *System Development Life Cycle* adalah proses pengembangan atau perubahan sistem perangkat lunak menggunakan model dan metode yang digunakan orang untuk mengembangkan sistem perangkat lunak sebelumnya berdasarkan praktik terbaik yang sudah diuji .[6]

Pada metode waterfall menggunakan beberapa langkah yaitu :

- 1) *Requirement*, pada tahap ini diperlukan komunikasi antara peneliti dengan pengguna untuk mengetahui apa yang perlukan dan batasan perangkat yang akan dibuat.
- 2) *Design*, tahapan kedua adalah desain. Pada tahap ini peneliti membuat desain tentang apa yang akan dilakukan sistem dari hasil analisa tahap pertama.
- 3) *Implementation*, pada tahapan ini peneliti melakukan pengkodean yang menghasilkan perangkat lunak.

- 4) *Verification*, selanjutnya adalah tahap verifikasi dimana peneliti melakukan verifikasi dan pengujian terhadap sistem yang dibuat.
- 5) *Maintenance*, ini adalah tahap akhir dari metode waterfall. Pada tahap ini dilakukan pemeliharaan termasuk memperbaiki kesalahan terhadap perangkat yang sudah dijalankan sebelumnya.



Gambar 2. 1 Metode Pengembangan Waterfall

2.2.5. Unified Modelling Language (UML)

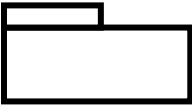
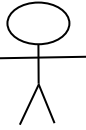
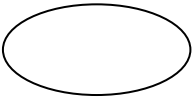
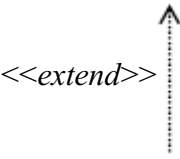
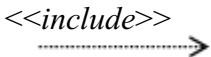
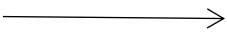
Unified Modeling Language (UML) adalah alat pengembangan perangkat lunak berorientasi objek yang telah menjadi standar dalam dunia pengembangan. UML adalah bahasa yang kuat untuk merancang sistem perangkat lunak. Dengan bantuan UML, hasil rancangan dapat dikomunikasikan kepada pihak lain. UML berarti bahasa permodelan standar. Dengan bantuan UML perancang sistem dapat membuat gambar dalam format standar yang mudah dipahami.[7]

Terdapat 4 (empat) macam diagram yang tersedia dalam UML untuk memodelkan aplikasi perangkat lunak berorientasi objek, yaitu antara lain :

1. *Use case Diagram*

Use case diagram merupakan sebuah diagram yang menunjukkan hubungan antara *actors* dan *use cases*. *Use case* digunakan untuk analisis dan desain sebuah sistem.

Tabel 2.1 Simbol-simbol *Use case Diagram*.

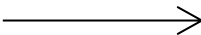
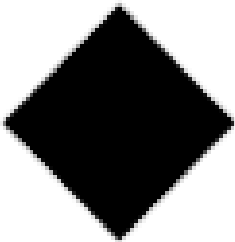
Simbol	Keterangan
	<i>Package</i> Menggambarkan pengelompokan dari <i>usecase</i>
	<i>Actor</i> Menggambarkan orang atau sistem yang menyediakan atau menerima informasi dari sistem
	<i>UseCase</i> Menggambarkan proses sistem kebutuhan sistem yang menyediakan atau menerima informasi dari sistem
	<i>AssociationExtend</i> Menggambarkan perluasan dari <i>usecase</i> diagram arah tidak boleh ke arah extending <i>usecase</i>
	<i>AssociationInclude</i> Menggambarkan pemanggilan <i>usecase</i> oleh <i>usecase</i> lain, arah panah tidak boleh ke arah <i>base</i> atau <i>parentusecase</i>
	<i>Association Aktif</i> Menggambarkan bagaimana <i>actor</i> terlibat dalam <i>usecase</i>
	<i>Association Pasif</i> Menggambarkan bagaimana <i>actor</i> tidak terlibat secara langsung dalam <i>usecase</i>

Sumber: S. Anardani, 2019

2. Class Diagram

Class diagram adalah sebuah diagram yang menunjukkan struktur atau hubungan antar *class* yang didalamnya terdapat atribut dan fungsi dari suatu objek.

Tabel 2.2 Simbol-simbol Class Diagram

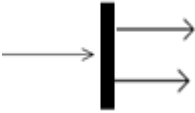
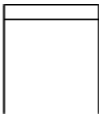
Simbol	Keterangan
Class - Attribute + Method()	<i>Class</i> Penggambaran dari <i>classname</i>, atribut atau <i>property</i> atau data dan <i>method</i> atau <i>function</i> atau <i>behavior</i>
	<i>Association</i> Menggambarkan hubungan antara <i>initialnode</i>, <i>activities</i>, <i>fork</i>, <i>decisionpoint</i> dan <i>activityfinalnode</i> dalam sebuah sistem
	<i>Agregation</i> Menggambarkan perluasan dari <i>class</i>
	Generalisasi / <i>Inheritance</i> Menggambarkan satu atau sekumpulan class mewarisi atribut atau <i>method</i> dari suatu <i>class</i>

Sumber: S. Anardani, 2019

3. Activity Diagram

Diagram aktivitas adalah diagram yang menggambarkan konsep aliran informasi, kegiatan yang direncanakan dengan baik dan terstruktur dalam suatu sistem.

Tabel 2.3 Simbol-simbol Activity Diagram.

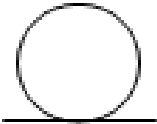
Simbol	Keterangan
	<i>StartPoint</i> Menggambarkan permulaan dari sebuah sistem yang akan dikerjakan, biasanya terletak pada pojok kiri atas
	<i>EndPoint</i> Menggambarkan akhir dari sebuah sistem
	<i>Association</i> Menggambarkan hubungan antara <i>initial no deactivities fork decision point</i> dan <i>activity final node</i> sebuah sistem
	<i>Activity</i> Menggambarkan aktivitas yang dilakukan pada sistem
	<i>Fork</i> Menggambarkan sebuah <i>activity</i> yang berjalan secara bersamaan, biasanya mempunyai 1 transisi masuk dan dua atau lebih transisi keluar atau bisa lebih transisi masuk dan hanya satu transisi keluar
	<i>DecisionPoint</i> Menggambarkan hubungan transisi sebuah garis dari atau ke <i>decisionpoint</i>
	<i>Swimlane</i> Menggambarkan sebuah cara untuk mengelompokkan <i>activity</i>

Sumber: S. Anardani, 2019

4. Sequence Diagram

Sequence diagram merupakan sebuah diagram yang menggambarkan kolaborasi dari objek-objek yang saling berinteraksi antar elemen dari suatu *class*.

Tabel 2.4 Simbol-simbol Sequence Diagram.

Simbol	Keterangan
	<i>Actor Object</i> Menggambarkan pihak yang melakukan interaksi atau yang memicu sistem untuk berfungsi
	<i>Boundary Object</i> Menggambarkan obyek yang menjadi <i>interface</i>
	<i>Control Object</i> Menggambarkan obyek yang menjadi control sistem
	<i>Life Time</i> Menggambarkan eksekusi obyek selama <i>sequence</i> (<i>Message</i> dikirim atau diterima dan aktifitasnya)
	<i>Execution Occurrence</i> Menunjukkan <i>focus control</i> obyek pada suatu waktu
	<i>Message</i> Menggambarkan komunikasi yang terjadi antar obyek

Sumber: S. Anardani, 2019

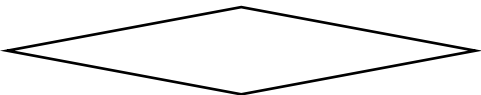
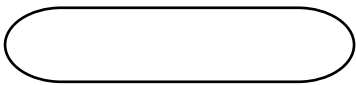
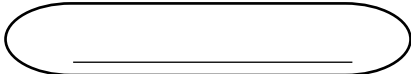
2.2.6. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan model data berisi informasi detail yang memanfaatkan gambar grafis untuk mempermudah menyelesaikan pengembangan suatu sistem dan membantu memenuhi kebutuhan sistem analis. ERD adalah suatu diagram yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang dilengkapi atribut.[8]

Pada dasarnya terdapat 3 komponen yang digunakan dalam *Entity Relationship Diagram* (ERD), yaitu:

1. *Entity*. *Entity* adalah objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbolnya berbentuk persegi panjang.
2. *Attribute*. Atribut atau biasa disebut elemen berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. *Attribute* berisi sesuatu yang dapat mengidentifikasi isi elemen satu dengan elemen yang lain.
3. Hubungan / Relasi. Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

Tabel 2.5 Simbol-simbol Entity Relationship Diagram.

Simbol	Keterangan
	<i>Entity</i>
	Relasi atau aktivitas antar <i>entity</i>
	<i>Fiels</i> atau <i>attribute</i>
	<i>Fiels</i> atau <i>attribute primary key</i>

Sumber: S. Anardani, 2019

2.2.7. *CodeIgniter*

CodeIgniter adalah *framework* aplikasi berbasis web yang mengusung model MVC (*Model, View, Controller*). Salah satu *framework* PHP ini dapat menjadi tools bagi seorang web developer untuk mengembangkan suatu situs dengan lebih mudah. *CodeIgniter* bersifat *open source* (tidak berbayar) sehingga *framework* PHP ini dapat dengan mudah Anda dapatkan.[9]

2.2.8. *Hypertext Preprocessor (PHP)*

PHP merupakan salah satu bahasa pemrograman berbasis web yang ditulis oleh dan untuk pengembang web. PHP pertama kali dikembangkan oleh Rasmus Lerdorf, seorang pengembang *software* dan anggota tim Apache, dan dirilis pada akhir tahun 1994. PHP dikembangkan dengan tujuan awal hanya untuk mencatat pengunjung pada website pribadi Rasmus Lerdorf. Pada rilis keduanya, ditambahkan Form Interpreter, sebuah *tools* untuk melakukan penerjemahan perintah SQL. Rilis kedua disebut dengan PHP/FI. Sejak itu, PHP mulai diterima sebagai sebuah bahasa pemrograman baru yang sangat diminati. Terbukti pada pertengahan tahun 1997, tercatat sekitar 50.000 situs di seluruh dunia telah menggunakan PHP.

Dengan bertambah banyaknya pengguna PHP di seluruh dunia, maka PHP tidak memungkinkan lagi untuk dikelola oleh satu orang saja. Sehingga dibentuk sebuah tim pengembang yang mengorganisir kontributor dari seluruh dunia dengan model pengembangan proyek *open source* "benevolent junta". Tim tersebut dipimpin oleh dua orang programmer, Zeev Suraski dan Andi Gutmans. Keduanya lalu mendirikan sebuah perusahaan PHP dengan nama Zend (akronim

dari nama Zeev Suraski dan Andi Gutmans). Selanjutnya Zend merilis versi PHP3 dan PHP4.

2.2.9. *Mysql*

MySQL adalah salah satu jenis database server yang sangat terkenal. MySQL menggunakan bahasa SQL untuk mengakses databasenya. MySQL tersedia untuk beberapa platform, di antara nya adalah untuk versi windows dan versi linux. Untuk melakukan administrasi secara lebih mudah terhadap Mysql, anda dapat menggunakan software tertentu, di antara nya adalah phpmyadmin dan mysql yog. Pada kesempatan kali ini, kita akan menggunakan phpmyadmin, yang terdapat dalam bundle xampp, dapat di peroleh di www.apachefriends.org. [10]

2.2.10. *Black box testing*

Black box testing merupakan pengujian yang dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian ini dapat digambarkan sebagai suatu kotak hitam dimana penguji hanya bisa melihat luarnya saja tanpa mengetahui isi dari kotak hitam tersebut. Pengujian *black box* didesain dan dibuat untuk mendeteksi kesalahan (*error*) pada suatu program yang dibuat tanpa mempedulikan masalah internal. Teknik pengujian ini dilakukan dengan menjalankan suatu program (test case) dan melakukan input untuk memperoleh output dari suatu program. Pengujian perangkat lunak ini dilakukan dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian ini berfungsi untuk fungsi masukan dan keluaran dari suatu perangkat lunak sudah sesuai dengan yang dibutuhkan.

Metode *Black box testing* adalah metode yang hanya memerlukan batas bawah dan batas atas. Estimasi banyaknya jumlah data yang diuji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dengan metode ini dapat diketahui apabila fungsionalitas masih dapat menerima data yang tidak diinginkan maka data yang disimpan belum dapat valid secara sempurna.[11]

2.2.11. *User Acceptance Testing* (UAT)

User Acceptance Testing (UAT) merupakan proses pengujian solusi yang dibuat dalam suatu sistem, apakah sudah sesuai untuk pengguna. Proses ini berbeda dengan pengujian sistem tetapi solusi dalam sistem tersebut akan bekerja untuk pengguna yaitu tes bahwa pengguna menerima solusi di dalam sistem.

UAT adalah pengujian yang dilakukan oleh end-user , di mana user tersebut

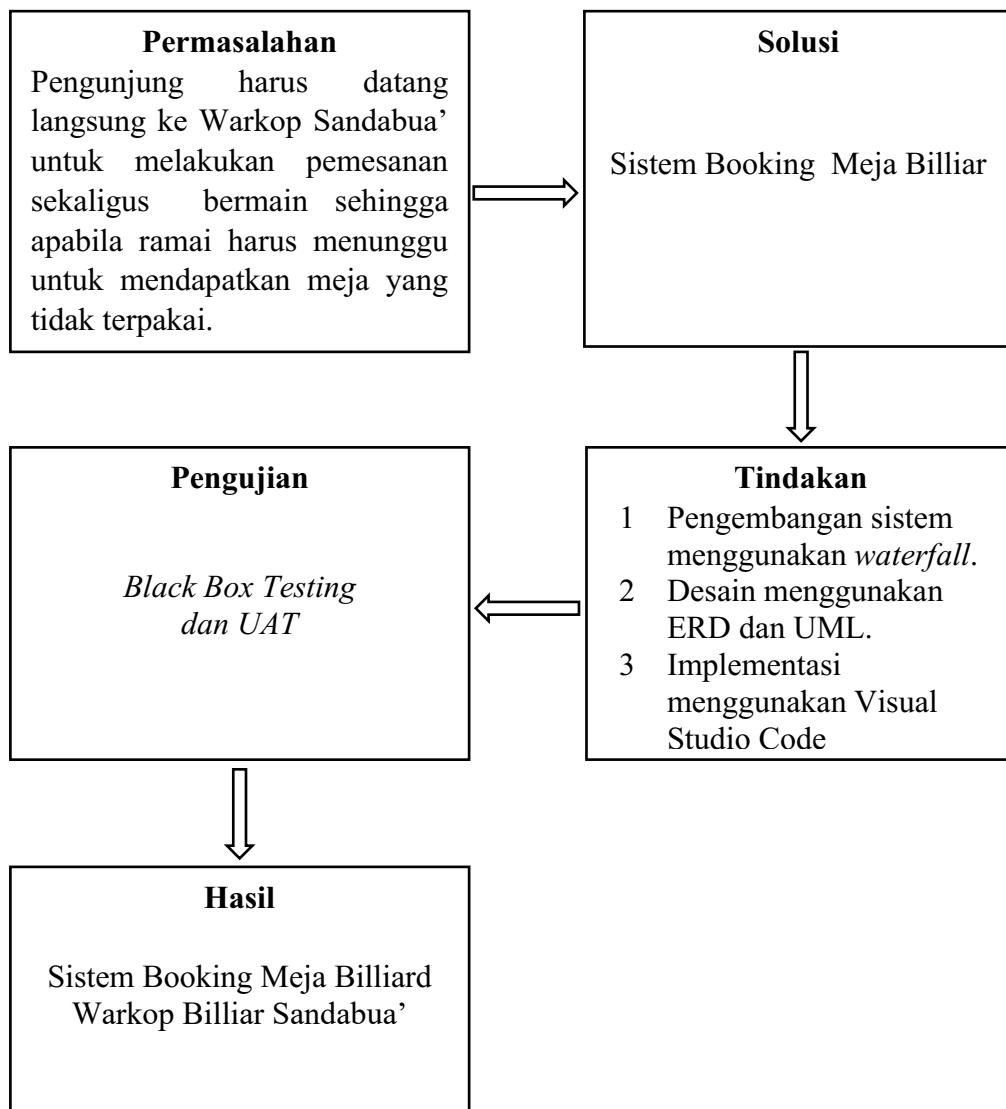
biasanya adalah staff atau karyawan perusahaan yang langsung berinteraksi dengan

sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan

kebutuhan sehingga UAT dapat menghasilkan dokumen yang bisa dijadikan bukti bahwa produk yang dibuat dapat diterima pengguna.

2.3. Kerangka Pikir

Berdasarkan latar belakang dan rumusan masalah yang telah dijelaskan sebelumnya, maka kerangka pikir dari penelitian ini dapat dilihat pada Gambar 2.2.



Gambar 2. 2 Kerangka Pikir