

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Di era globalisasi saat ini, sebagian Masyarakat telah menggunakan teknologi informasi seperti *internet*. Dengan adanya teknologi internet mempermudah manusia dalam menegakses segala sesuatu dengan lebih mudah dan praktis. Canggih nya teknologi saat ini bisa menjadi peluang bagi pebisnis dengan mendirikan usaha mereka agar lebih mudah untuk di akses. Contohnya membuka bisnis warkop yang dimana anak muda zaman sekarang sangat suka nongkrong di tempat-tempat seperti warkop dan *kafe*. Perkembangan usaha warkop secara tidak langsung mendatangkan pesaing jika usaha tersebut semangkin maju maka cara pelayanan pemesanan pada warkop semangkin rumit jika tidak menggunakan bantuan teknologi yang diiringi dengan perkembangan teknologi informasi yang tidak terlepas dari campur tangan manusia memunculkan suatu ide untuk mengembangkan usaha warkop dan memanfaatkan teknologi *internet* untuk membantu mengelola sistem pemesanan makanan dan minuman sehingga teknologi tersebut dapat membantu mempermudah pekerjaan manusia.

Warkop semabung adalah salah satu perusahaan yang bergerak dalam bidang penjualan makanan dan minuman, yang beralamatkan di jln. Depati Hamzah Semabung Baru Girimaya PANGKALPINANG. Warkop Semabung berdiri sejak 15 Oktober 2022 dan sampai sekarang masih beroperasi. Warkop Semabung hanya memiliki tiga karyawan, yang mana ketiga karyawan tersebut memiliki tugas masing-masing yaitu sebagai *chef*, *waiters* dan kasir. Pada umumnya pelanggan datang ke Warkop Semabung untuk duduk lalu memesan makanan dan minuman, yang mana pelanggan memesan menu makanan dan minuman melalui menu kertas yang ada di depan meja kasir. Kemudian mengambil kertas menu dan menuliskan menu yang ingin dipesan. Setelah itu pelanggan mengantar kembali menu kertas

tersebut ke meja kasir dan kasir memberitahu *chef* apa saja menu yang dipesan oleh pelanggan.

Cara seperti ini kurang efisien karna memakan waktu yang cukup lama. Dan tidak menutup kemungkinan terjadinya keluhan dari pelanggan karna lumayan sulitnya untuk memesan menu dimana harus mengambil dan mengantarkan kembali menu ke meja kasir. Dan lamanya proses memesan karna harus mengantri dengan dengan pelanggan lain. Dengan menggunakan salah satu metode Pembangunan *software* yaitu metode *waterfall* untuk menunjang kebutuhan perancangan sistem yang akan dibangun. Dengan menggunakan metode *waterfall* akan dilakukan penelitian secara bertahap dimana jika tahap sebelumnya belum selesai tidak dapat melakukan penelitian berikutnya. Penelitian ini akan membantu kita untuk lebih teliti sehingga tidak ada penelitian yang tertinggal sebelum melangkah ke penelitian berikutnya.

Berdasarkan permasalahan di atas dan hasil wawancara dengan pemilik Warkop Semabung maka perlu dibangun sebuah sistem dengan judul **“SISTEM RANCANG BANGUN PEMESANAN MAKANAN DAN MINUMAN PADA WARKOP SEMABUNG BERBASIS WEB”**.

1.2 Rumusan Masalah

1. Bagaimana sebuah sistem informasi dapat membantu sistem pemesanan makanan dan minuman pada Warkop Semabung?
2. Bagaimana sebuah sistem membantu pelanggan dalam mempermudah pemesanan makanan dan minuman pada Warkop semabung?

1.3 Batasan Masalah

Sistem ini hanya berfokus pada:

- 1) Sistem informasi pemesanan makanan dan minuman pada warkop Semabung ini dibuat hanya untuk data pemesanan saja.

- 2) Sistem pembayaran hanya dilakukan secara langsung atau *offline* setelah pemesanan.

Sistem ini tidak berfokus pada:

1. Sistem ini tidak menyediakan pemesanan melalui rumah dan tidak menyediakan pengantaran makanan dan minuman ke rumah.
2. Sistem pemesanan ini tidak memiliki menu *cancel*.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan Penelitian

1. membangun sebuah sistem yang membantu pelanggan untuk menangani proses pemesanan menu makanan dan minuman.
2. untuk mempermudah Warkop Semabung dalam pengembangan bisnisnya.

1.4.2 Manfaat Penelitian

Manfaat dari pembuatan penelitian sistem rancang bangun berbasis *web* ini

- 1) Manfaat bagi penulis sendiri diharapkan dapat menambah ilmu pengetahuan dan pengalaman dalam merancang sistem dan program yang berbasis *web*.
- 2) Bagi Perusahaan hasil penelitian ini dapat digunakan untuk membantu mempermudah dalam mengelola sistem informasi pelayanan jasa pemesanan makanan dan minuman.

1.5 Metodologi penelitian

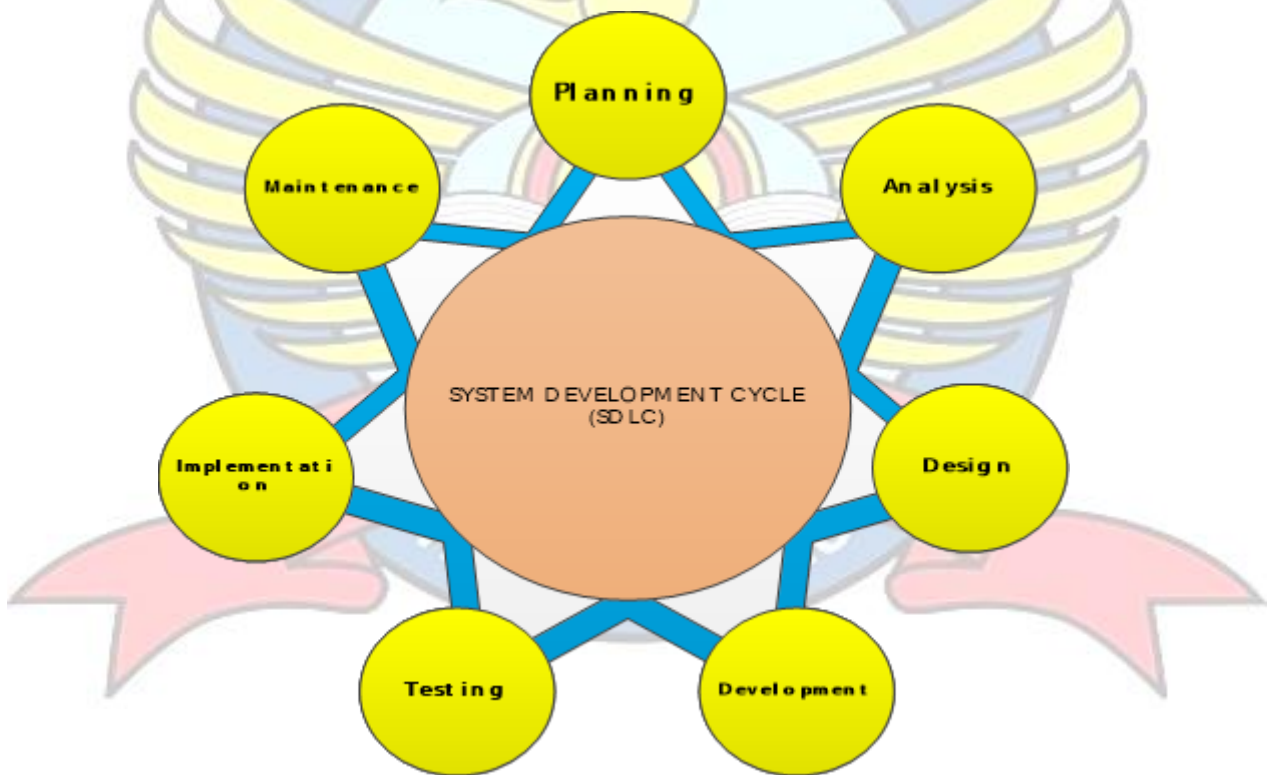
1.5.1 Metodologi Sistem Development Life (SDLC)

Penelitian ini akan menggunakan metodologi *Software Development Life Cycle* (SDLC), Dalam perusahaan perangkat lunak SDLC adalah metode yang digunakan untuk proyek perangkat lunak Ini terdiri dari rencana menyeluruh yang menguraikan cara membuat, Memelihara, Mengganti, dan Memodifikasi atau meningkatkan perangkat lunak tertentu. Siklus hidup

menguraikan proses untuk meningkatkan standar perangkat lunak serta seluruh prosedur pengembangan.

SDLC atau siklus hidup pengembangan perangkat lunak tugas yang dilakukan pada setiap tahap proses pengembangan perangkat lunak ditentukan oleh kerangka kerja. Berfungsi membagi peranan dan tanggung jawab yang jelas antara pengembang, *Desainer*, Analis bisnis, dan Manajer proyek. Fungsi lain dari SDLC ialah dapat memberikan gambaran *input* dan *output* yang jelas dari satu tahap menuju tahap selanjutnya, Sehingga dapat membantu penelitian[1].

Berikut ini adalah tahapan-tahapan yang terdapat dalam SDLC



Gambar 1.1 Gambar Tahapan-Tahapan SDLC[1].

Tahap – tahap Pengembangan sistem yang terdapat dalam metodologi SDLC:

1) *planning and Requirement Analysis*

Perencanaan dan Analisis Kebutuhan adalah *elemen* yang paling penting dan *esensial* dari SDLC adalah analisis kebutuhan. Dengan masukan dari pelanggan, departemen penjualan, *survei* pasar, dan *spesialis* domain dalam bisnis, ini dilakukan oleh anggota tim *senior*. pendekatan proyek fundamental kemudian direncanakan menggunakan pengetahuan ini, dan pengujian produk selanjutnya dilakukan analisis kelayakan dalam domain operasional, teknologi, dan keuangan.

Pada tahap perencanaan juga dilakukan untuk mengidentifikasi risiko proyek dan merencanakan persyaratan penjaminan mutu. Kesimpulan studi kelayakan teknis mengidentifikasi berbagai strategi teknis yang dapat digunakan untuk mengimplementasikan proyek dengan sukses dan dengan risiko seminimal mungkin.

2) *Defining Requirements*

Setelah studi kebutuhan selesai tahap selanjutnya adalah mendeskripsikan dan mencatat kebutuhan produk secara tepat dan mendapatkan persetujuan pelanggan atau analisis pasar. Ini dicapai melalui dokumen SRS, atau Spesifikasi Persyaratan Perangkat Lunak, yang berisi semua persyaratan produk yang harus ditentukan dan dikembangkan sepanjang siklus hidup proyek.

3) *Designing the product architecture*

SRS berfungsi sebagai panduan bagi arsitek produk karena mereka menciptakan arsitektur optimal untuk produk baru. Biasanya, lebih dari satu pendekatan desain untuk arsitektur produk ditawarkan dan didokumentasikan dalam DDS (Spesifikasi Dokumen Desain). Berdasarkan persyaratan yang diberikan dalam SRS, DDS ini ditinjau

oleh semua pemangku kepentingan yang signifikan, dan pendekatan desain yang optimal dipilih untuk produk berdasarkan sejumlah faktor, termasuk penilaian risiko, ketahanan produk, modularitas desain, batasan anggaran dan jadwal. Sebuah strategi desain menguraikan semua komponen arsitektur produk serta bagaimana berkomunikasi dan mewakili aliran data dengan modul eksternal atau pihak ketiga.

4) *Building or Developing the Product*

Selama fase SDLC ini, produk benar-benar dikembangkan dan dibangun. Pada titik ini, kode pemrograman dihasilkan sesuai dengan DDS. Pembuatan kode dapat diselesaikan dengan cepat jika desain dilakukan secara tepat dan sistematis. Pengembang harus mematuhi standar pengkodean yang ditetapkan oleh bisnis mereka, dan kode dihasilkan menggunakan alat pemrograman termasuk kompiler, juru bahasa, dan *debugger*. Bahasa pemrograman tingkat tinggi yang berbeda digunakan untuk pengkodean, termasuk C, C++, *Pascal*, *Java*, dan *PHP*. Jenis perangkat lunak yang diproduksi mempengaruhi pilihan bahasa pemrograman.

5) *Testing the Product*

Karena aktivitas pengujian biasanya disertakan dalam semua tahapan SDLC dalam model SDLC kontemporer, tahapan ini biasanya merupakan subset dari semua tahapan. Namun, tahap ini hanya berkaitan dengan tahap pengujian produk, di mana setiap masalah ditemukan, dilacak, diperbaiki, dan diuji ulang sampai produk memenuhi persyaratan kualitas SRS.

6) *Deployment in the Market and Maintenance*

Setelah produk melalui pengujian dan siap untuk digunakan, produk tersebut secara resmi dirilis di pasar yang relevan. Menurut rencana bisnis organisasi, penyebaran produk kadang-kadang dapat dilakukan

secara bertahap. Produk dapat diuji dalam pengaturan bisnis dunia nyata sebelum tersedia untuk pasar yang lebih besar (UAT). Produk kemudian dapat dirilis dipasar yang ditargetkan baik sebagaimana adanya atau dengan perbaikan yang diusulkan tergantung pada umpan balik. Setelah peluncuran produk di pasar, pemeliharaan dilakukan untuk pelanggan yang sudah dimilikinya.

Sistem Informasi menu pemesanan makanan dan minuman berbasis *Web* pada Warkop Semabung dikembangkan dengan metodologi SDLC karena mampu menampilkan informasi dengan tepat, akurat dan berguna dalam membantu pengambilan keputusan Sistem Informasi menu pemesanan Berbasis *Web* pada Warkop Semabung.

1.5.2 Model Penelitian

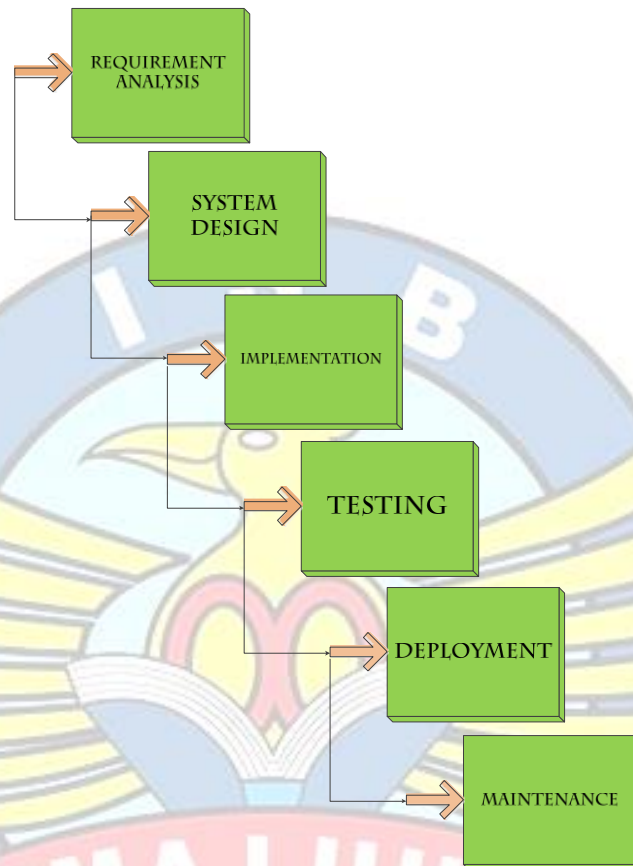
1.5.2.1 Model Waterfall

Model *Waterfall* adalah metodologi SDLC pertama yang diadopsi untuk pengembangan perangkat lunak. Istilah "model siklus hidup sekuensial linier" juga digunakan untuk menggambarkannya. Model Ini sangat mudah digunakan dan dipahami, Urutan dalam Metode *Waterfall* bersifat serial yang dimulai dari proses perencanaan, analisa, desain, dan implementasi pada sistem. Tidak ada tumpang tindih antar *fase* dalam model *waterfall* setiap langkah harus diselesaikan sebelum *fase* berikutnya dapat dimulai kerna itulah metode ini dinamakan dengan metode air terjun[2].

SDLC (*System Development Life Cycle*) merupakan titik keputusan yang sangat penting, kritis dan penting dalam pengembangan perangkat lunak untuk suatu proyek. Berhasil tidaknya proyek dapat diprediksi ketika manajer proyek menentukan model SLDC mana yang akan diterapkan. Model *Waterfall* adalah yang pertama kali digunakan dan sering digunakan oleh proyek-proyek pemerintah dan perusahaan besar. Template ini juga

menyoroti pentingnya dokumentasi, sehingga cocok untuk proyek yang mengutamakan kualitas[3].

Langkah –langkah dalam model *waterfall* dapat dilihat pada gambar berikut:



Gambar 1.2 Gambar Tahapan-Tahapan *Waterfall*[3].

Berikut ini tahapan-tahapan dalam model *waterfall*:

1) *Requirement analysis*

Pengumpulan dan analisis kebutuhan Selama tahap ini, semua kebutuhan sistem potensial dikumpulkan dan dicatat dalam dokumen spesifikasi kebutuhan.

2) *System Design*

fase ini, desain sistem dibuat sementara spesifikasi yang diperlukan dari *fase* pertama diperiksa. Desain sistem membantu dalam menentukan

arsitektur sistem secara keseluruhan serta perangkat keras dan persyaratan sistem.

3) *Implementation*

Sistem awalnya dibangun sebagai program kecil yang dikenal sebagai unit dengan masukan dari desain sistem, dan kemudian digabungkan pada *fase* berikutnya. Pengujian *unit* adalah proses mengembangkan dan mengevaluasi setiap *unit* untuk fungsionalitas.

4) *Integration and Testing*

Setelah pengujian setiap *unit* yang dibuat selama *fase* implementasi, seluruh sistem digabungkan. Seluruh sistem diuji untuk kesalahan dan kegagalan setelah integrasi.

5) *Deployment of system*

Setelah produk menjalani pengujian fungsional dan non-fungsional, produk tersebut akan dipublikasikan ke pasar atau digunakan di lingkungan pelanggan.

6) *Maintenance*

Berbagai masalah dapat muncul di lingkungan *klien*. *Patch* diterbitkan untuk mengatasi masalah tertentu. Selain itu, versi produk yang ditingkatkan dikeluarkan. Untuk membawa perubahan ini pada konsumen, pemeliharaan dilakukan lingkungan.

Sistem Rancang Bangun menu pemesanan makanan dan minuman Berbasis *web* pada Warkop Semabung yang dikembangkan dengan model *waterfall*. Alasan penggunaan metode *waterfall*, karena tahapan-tahapan dalam pengembangan sistem pada model ini terstruktur secara jelas dan dimana sebuah tahapan harus diselesaikan terlebih dahulu sebelum ke tahap berikutnya. Ini meminimalisir terjadinya kesalahan.

1.5.3 Metode Penelitian

1.5.3.1 Unified Modeling Language (UML) sebagai Tools

Diagram *Unified Modelling Language* (UML) akan digunakan untuk membahas diagram *use case* sebagai alat untuk hubungan pengguna dengan sistem dan diagram kelas sebagai alat untuk menggambarkan struktur sistem. Diagram UML digunakan untuk memvisualisasikan dan menggambarkan sistem informasi[4]

Alat untuk memvisualisasikan dan mendokumentasikan hasil analisis dan desain yang berisi sintaks pemodelan sistem secara visual dan juga seperangkat konvensi pemodelan yang digunakan untuk Mengidentifikasi atau mendeskripsikan objek terkait sistem perangkat lunak[5].

1) *Use Case* Diagram

Menurut Tohari dalam Tabrani dan Aghniya (2019: 46) menyimpulkan bahwa “*use case* adalah suatu rangkaian atau deskripsi dari suatu kelompok yang saling bergantung dan membentuk suatu sistem yang teratur yang dijalankan atau dipantau oleh agen”. Menurut Pratama (2019b), “Diagram *use case* adalah representasi grafis dari beberapa atau seluruh *aktor*, *use case*, dan interaksi yang memperkenalkan suatu sistem. *Use Case* Diagram tidak menjelaskan penggunaan *use case* secara *detail*, namun hanya yang memberikan gambaran singkat tentang hubungan antara *use case*, *aktor*, dan sistem. kita akan mengetahui fungsi apa saja yang dimiliki pada sistem yang dibuat”[6].

2) *Activity* Diagram

Menurut Tohari dalam Tabrani dan Aghniya (2019:45), mendefinisikan sebagai berikut: “diagram memodelkan proses bisnis dan urutan aktivitas dalam suatu proses. Diagram ini sangat mirip dengan diagram alur karena memodelkan alur kerja dari aktivitas ke aktivitas atau dari aktivitas ke keadaan. Menurut Novitasari (2018), pengertian “diagram *activity* adalah model yang diimplementasikan pada sistem yang

menggambarkan aktivitas sistem yang sedang berjalan. Diagram *activity* digunakan untuk menjelaskan pengoperasian suatu program tanpa melihat pengkodean atau tampilannya[6].

3) *Sequence* Diagram

Menurut Tohari dalam Tabrani dan Aghniya (2019:46), disimpulkan bahwa “*Sequence* Diagram menggambarkan interaksi antara beberapa objek dalam suatu urutan temporal”. *Sequence* diagram merupakan salah satu diagram pada UML, *sequence* diagram ini merupakan diagram yang menggambarkan kolaborasi dinamis antara beberapa objek. Kegunaannya adalah untuk menampilkan urutan pesan yang dikirim antar objek serta interaksi antar objek. Sesuatu yang terjadi pada titik tertentu selama eksekusi sistem (Pratama 2019)[6].

Diagram *sequence* adalah diagram yang dirancang untuk mengeksplorasi aliran awal hubungan antar objek[7].

4) *Class* Diagram

Diagram kelas harus menggambarkan struktur sistem sedemikian rupa sehingga mengidentifikasi kelas-kelas yang akan dirancang untuk membuat sistem[7].