

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Dalam pembuatan tugas akhir ini ada beberapa tinjauan pustaka dijadikan sebagai mendapatkan perbandingan dan referensi untuk menganalisis sistem yang ada, antara lain :

Menurut penelitian yang dilakukan oleh (Indriyani, Efrilina Suawati, Renny Adrian, 2019) dalam jurnal yang berjudul “Aplikasi Pencatatan Pembelian dan Perhitungan Biaya Produksi Tahu (Studi Kasus Pada Perusahaan Lestari Jaya Putri Laksana)” dalam penelitiannya penulis menjelaskan bahwa Perusahaan Lestari Jaya Putri Laksana tidak melakukan pencatatan, baik berupa proses pembelian bahan baku, pembelian bahan penolong serta biaya produksi, karena kurangnya pengetahuan pemilik perusahaan akan ilmu akuntansi biaya dalam hal pencatatan pembelian dan perhitungan biaya produksi yang dikeluarkan selama proses produksi, sehingga mengakibatkan nilai harga pokok penjualan tidak sesuai dengan jumlah biaya produksi yang dikeluarkan. Perusahaan tidak tahu pasti biaya yang dikeluarkan selama satu kali produksi, jumlah biaya produksi bisa lebih tinggi dari pada nilai harga pokok penjualannya. Tujuan penelitian ini adalah membangun Aplikasi Pencatatan Pembelian dan Perhitungan Biaya Produksi Tahu yang dapat mengelola proses pembelian bahan baku dan bahan penolong, memperhitungkan unsur biaya produksi seperti biaya bahan baku, biaya tenaga kerja langsung, dan biaya *overhead* pabrik menggunakan metode *System Development Life Cycle* (SDLC) yang terdiri dari 5 tahapan yaitu, *planning* (perencanaan), *analysis* (analisis), *design* (desain), *implementation* (implementasi), *testing* (uji coba), dan *maintenance* (perbaikan). Sedangkan pengumpulan datanya melalui observasi, wawancara dan mencari literatur yang berhubungan dengan penelitian ini sehingga dapat memecahkan permasalahan yang berkaitan dengan Pencatatan Pembelian dan Perhitungan Biaya Produksi Tahu terutama dalam mengelola proses pembelian bahan baku dan bahan penolong, memperhitungkan unsur biaya produksi seperti biaya bahan baku, biaya tenaga kerja langsung, dan biaya *overhead* pabrik menjadi lebih efektif dan efisien serta dapat meningkatkan

pelayanan kepada pelanggan. Hasil penelitian menunjukkan Aplikasi Pencatatan Pembelian dan Perhitungan Biaya Produksi Tahu ini dapat membantu serta mempermudah mengelola proses pembelian bahan baku dan bahan penolong, memperhitungkan unsur biaya produksi seperti biaya bahan baku, biaya tenaga kerja langsung, biaya *overhead* pabrik, menginput data transaksi pembelian bahan baku secara tunai, dan dapat menyajikan jurnal umum dan buku besar sehingga menjadi lebih cepat dan akurat.

Menurut penelitian yang dilakukan oleh penerapan (Basirun, Muhammad Saputra, Herman Putri, 2022) “harus ada singkatan / nama dibalik dalam jurnal yang berjudul “Penerapan Scm Dalam Upaya Pengelolaan Stock Bahan Baku Produksi Tahu Pada Pabrik Tahu Helen” dalam penelitiannya penulis menjelaskan Alasan mendasar penerapan supply chain management ini karena program ini melibatkan beberapa pihak yang terkait seperti supplier, distributor dan konsumen. Adapun kegiatan utama yang dilakukan yaitu produksi tahu dalam lingkup industri rumah tangga. Kebutuhan bahan baku di Pabrik Tahu Helen setiap hari dan hal ini bisa meningkat tergantung pesanan pelanggan. Beberapa kendala yang dialami oleh Pabrik Tahu Helen adalah sering mengalami kehabisan stok bahan dikarenakan terkadang banyak pesanan mendadak dari pelanggan. Proses pendataan bahan masih dilakukan dengan cara pencatatan secara manual sehingga dapat terjadi kesalahan dalam menentukan jumlah persediaan bahan. Bila diamati secara mendalam, masih banyak kendala yang terlihat sejak usaha ini dilaksanakan. Dari sini penting untuk dikaji ruang lingkup dari penerapan *supply chain management* sebagai wujud dari usaha memajukan pelaku bisnis khususnya industri rumah tangga. Tujuan penelitian ini adalah Penerapan SCM (*Supply Chain Management*) Dalam Upaya Pengelolaan Stock Bahan Baku Produksi Tahu sebagai wujud dari usaha memajukan pelaku bisnis khususnya industri rumah tangga menggunakan metode pendekatan penelitian kualitatif yang merupakan penelitian yang hanya mengumpulkan data dan menjelaskan secara deskriptif/narasi tanpa harus diolah dengan perhitungan. Penelitian ini hanya akan memanfaatkan data yang diperoleh dari tempat riset dan menginputkannya tanpa mengubah apapun. Hasil penelitian menunjukkan Perancangan sistem *Supply Chain Management* berbasis

web ini dibangun untuk mengkoordinasi hubungan antara supplier dengan Kepala Produksi pada Pabrik Tahu Helen., Sistem *Supply Chain Management* berbasis web ini dibangun agar Pabrik Tahu Helen dapat melakukan order kepada supplier yang berkomitmen menyediakan bahan baku dengan tepat waktu., Perangkat lunak *Supply Chain Management* berbasis web telah berhasil dibangun sebagai sistem informasi yang dapat membantu dalam pengelolaan stok bahan baku dan hasil percetakan pada Pabrik Tahu Helen.

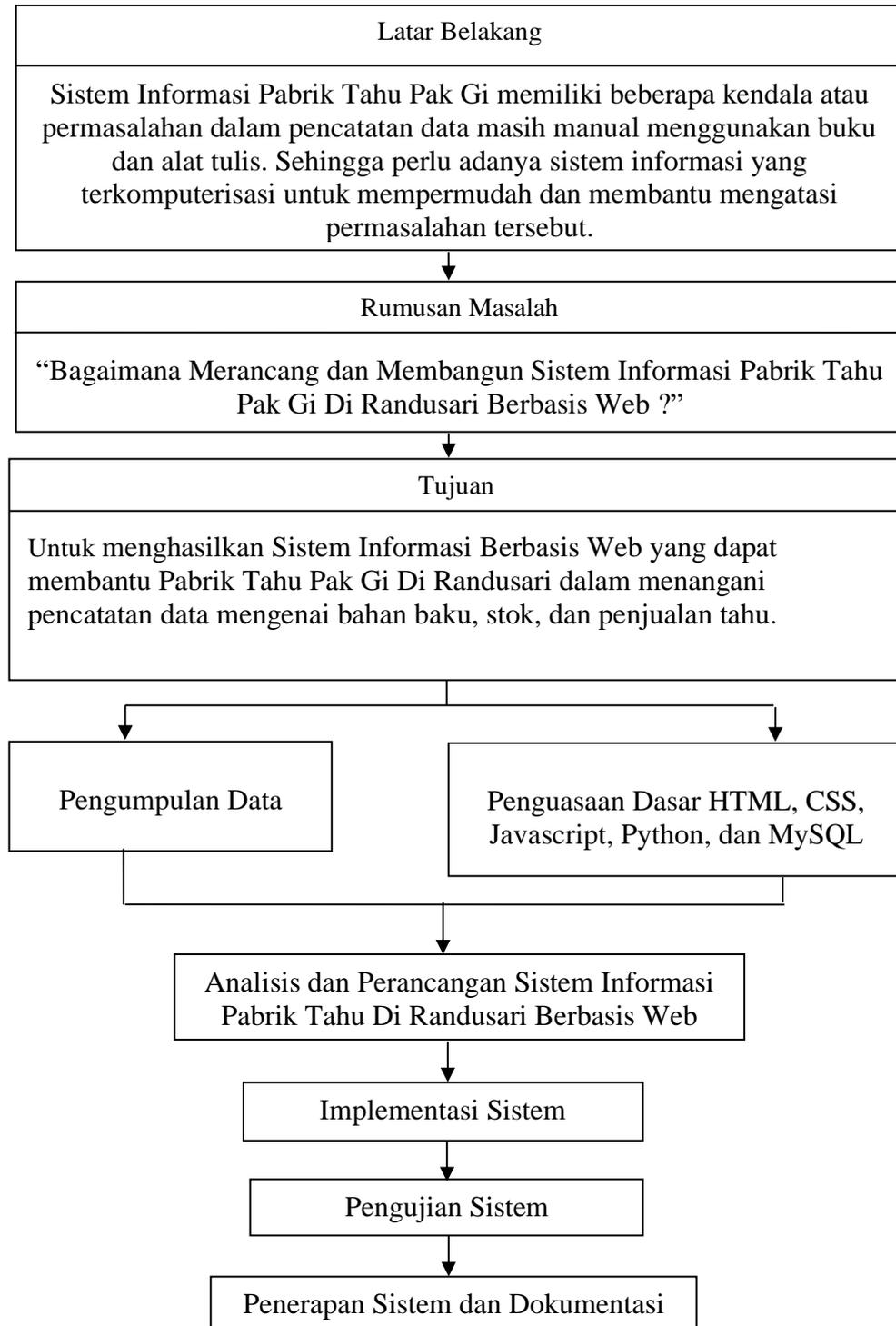
Menurut penelitian yang dilakukan oleh (Yulianeu, Aneu Muhamad Noer, 2019) dalam jurnal yang berjudul “Sistem Informasi Pengolahan Data Produksi Dan Distribusi Di Perusahaan Pabrik Tahu Jajang Suparman JS Kecamatan Cihaurbeuti Kabupaten Tasikmalaya” dalam penelitiannya penulis menjelaskan bahwa merancang Sistem Informasi Pengolahan Data Produksi dan Distribusi Di Perusahaan Tahu Jajang Suparman J.S yang dianggap masih kurang efektif dan efisien karena terbatasnya faktor informasi yang di antaranya : Dalam pembelian bahan baku tahu yang harganya sering naik turun sehingga jarang di ketahui oleh para pengusaha tahu yang mengakibatkan pendapatan tidak sesuai dengan yang diharapkan, Sistem produksi dan distribusi penjualan yang masih kurang efektif dan efisien. Sehingga dalam perancangan sistemnya penulis mencoba mengimplementasikan program pengolahan tahu, yang di kemudian hari dapat di pergunakan oleh para pengusaha tahu untuk memperluas usahanya. Tujuan penelitian ini adalah membangun Sistem Informasi Pengolahan Data Produksi Dan Distribusi Di Perusahaan Pabrik Tahu, yang di kemudian hari dapat di pergunakan oleh para pengusaha tahu untuk memperluas usahanya menggunakan 5 tahapan yaitu, *planning* (perencanaan), *analysis* (analisis), *design* (desain), *implementation* (implementasi), *testing* (uji coba), dan *maintenance* (perbaikan). Sedangkan pengumpulan datanya melalui observasi, wawancara dan mencari literatur yang berhubungan dengan penelitian ini sehingga dapat memecahkan permasalahan yang berkaitan dengan Pengolahan Data Produksi Dan Distribusi Di Perusahaan Pabrik Tahu menjadi lebih efektif dan efisien serta dapat meningkatkan pelayanan kepada pelanggan. Hasil penelitian menunjukkan Sistem Informasi Pengolahan Data Produksi Dan Distribusi Di Perusahaan Pabrik Tahu ini dengan komputerisasinya dapat

membantu serta mempermudah mengelola proses input-output data (bahan baku, penjualan, produksi, biaya operasional dan biaya non operasioanal) agar dapat meningkatkan efisien dalam pekerjaan, proses pembuatan laporan-laporan yang terkomputerisasi dengan sistem database akan memudahkan mendapatkan informasi yang terkait dengan data (pengolahan produksi, biaya dan penjualan) secara cepat tanpa mengurangi tingkat ketelitian maupun ke akuratan data yang terkait, dengan penerapan sistem database dalam hal laporan perhari maupun perbulan bisa juga pertahun dan lebih meningkatkan efektivitas pekerjaan dan lebih meningkatkan keamanan dalam hal penyimpanan data.

Dari keempat penelitian tersebut memiliki persamaan dengan tujuan penelitian penulis yaitu mengembangkan atau menerapkan sistem informasi atau aplikasi untuk mengelola proses produksi, stok bahan baku, dan penjualan di perusahaan Pabrik Tahu. Tujuan umumnya adalah untuk meningkatkan efisiensi, efektivitas, dan akurasi dalam pengelolaan bisnis Pabrik Tahu. Dengan kata lain, semua penelitian tersebut bertujuan untuk mengatasi kendala yang terjadi dalam pencatatan, pengolahan data, dan pengelolaan bisnis di pabrik tahu dengan mengadopsi solusi berbasis teknologi informasi. Kesimpulannya, semua penelitian memiliki tujuan umum untuk meningkatkan pengelolaan bisnis Pabrik Tahu melalui penggunaan teknologi informasi. Namun, perbedaan dalam metode pendekatan penelitian, penerapan metode pengembangan, ruang lingkup aplikasi, dan teknologi yang digunakan dapat menyebabkan perbedaan dalam pendekatan dan solusi yang dihasilkan dalam masing-masing penelitian. Tinjauan pustaka ini bertujuan sebagai bahan referensi dan rujukan terhadap hasil penelitian sebelumnya yang berhubungan dengan penelitian yang akan dilakukan.

2.2. Kerangka Pemikiran

Kerangka pemikiran Rancang Bangun Sistem Informasi Pabrik Tahu Pak Gi Di Randusari Berbasis Web dapat dilihat pada Gambar 2.1.



Gambar 2.1. Kerangka Pemikiran

Keterangan :

1. Latar Belakang

Pokok permasalahan yang mendasari perlunya dibangun sistem informasi berbasis web pada Pabrik Tahu Pak Gi Di Randusari adalah belum adanya sistem yang terkomputerisasi, dalam pencatatan data masih dilakukan secara manual dimana setiap aktivitas pencatatan di simpan dalam buku. Dengan menggunakan buku dan alat tulis dalam mencatat hasil produksi masih rentan terjadi kesalahan tidak sesuainya antara bukti fisik dan datanya. Hal ini membuat Bapak Giyanto selaku pemilik Pabrik Tahu Pak Gi Di Randusari sering mengalami kendala saat memeriksa data yang dicatat sebelumnya, seperti salah mencatat bahan baku, stok tahu, penjualan, perhitungannya yang masih menggunakan alat bantu kalkulator, lambatnya pencarian data dan rekap laporan, hingga kehilangan buku catatannya. Bagi usaha kecil menengah seperti Pabrik Tahu Pak Gi yang bergerak di sektor usaha produksi dan penjualan sebuah produk, sistem informasi memiliki peran yang sangat penting dalam menunjang kegiatan proses bisnis, seperti mengelola data produksi dan penjualan.

2. Rumusan Masalah

Perumusan masalah merupakan inti permasalahan dan jalan keluar menyelesaikan permasalahan yang ada, sehingga dapat dirumuskan masalah “Bagaimana Merancang dan Membangun Sistem Informasi Pabrik Tahu Pak Gi Di Randusari Berbasis Web?”

3. Tujuan

Tujuan dari penelitian ini adalah menghasilkan Sistem Informasi Berbasis Web yang dapat membantu Pabrik Tahu Pak Gi Di Randusari mengelola stok bahan baku yang diperlukan dalam proses produksi. Selain itu, sistem juga akan memungkinkan pemantauan stok tahu jadi lebih efisien, memastikan tersedianya produk yang memadai sesuai permintaan pelanggan. Selanjutnya, sistem juga akan memberikan dukungan dalam mencatat transaksi penjualan tahu, menciptakan catatan yang akurat dan mudah diakses.

4. Pengumpulan Data

- a. Data tertulis meliputi data yang diperlukan untuk membuat Perancangan Sistem Informasi Pabrik Tahu Berbasis Web seperti hasil wawancara, catatan, jurnal, serta semua informasi data yang dibutuhkan untuk membuat sistem informasi.
- b. Data tidak tertulis meliputi contoh tampilan desain Perancangan Sistem Informasi Pabrik Tahu Berbasis Web sebelumnya, dokumentasi, dan bahasa pemrograman yang digunakan.

5. Penguasaan dasar

Penguasaan dasar untuk Merancang dan Membangun Sistem Informasi Pabrik Tahu Berbasis Web ini menggunakan HTML (*Front-End*), CSS (*Front-End*), JavaScript (*client side*), Python (*Back-End server side*), dan MySQL (*database*) sebagai bahasa pemrograman untuk mendesain tampilan, *coding*, dan emulasi.

6. Analisis dan Perancangan Sistem Informasi Pabrik Tahu Berbasis Web

Analisis dan Perancangan Sistem Informasi Pabrik Tahu Berbasis Web menggunakan metode UML (*Unified Modelling, Language*) menggunakan aplikasi Star UML.

7. Implementasi Sistem

Merupakan langkah-langkah atau prosedur-prosedur yang dilakukan dalam menyelesaikan desain aplikasi yang telah disetujui, untuk menguji dan memulai sistem baru atau aplikasi yang diperbaiki. Lingkungan implementasi sistem ini meliputi kebutuhan perangkat lunak, perangkat keras, *form* program yang sesuai, *script* yang digunakan, pemrograman, pengujian program, dan pengujian aplikasi yang telah dirancang sesuai dengan kebutuhan dalam merancang dan membangun Sistem Informasi Pabrik Tahu Pak Gi Berbasis Web Di Randusari.

8. Pengujian Sistem

Metode pengujian sistem menggunakan metode *Black Box*. Metode *Black Box* adalah pengujian antarmuka pengguna setelah memberi tahu pengguna apakah sistem dapat digunakan atau tidak.

Prosedur pengujian ini menggunakan tabel referensi *input* dan *output* untuk menguji perilaku sistem saat diberikan input tertentu. Jika hasilnya sesuai dengan yang diharapkan, maka dapat dikatakan sistem telah lolos uji *Black Box*.

9. Penerapan Sistem dan Dokumentasi

Sistem yang sudah diimplementasikan dan diuji coba kemudian diterapkan pada Pabrik Tahu Pak Gi dan setelah itu dibuat dokumentasi dari keseluruhan kegiatan penyusunan Tugas Akhir.

2.3. Teori Pendukung

2.3.1. Rancang Bangun

Rancang bangun merupakan suatu kegiatan menerjemahkan hasil yang dianalisa dari sistem menggunakan bahasa pemrograman untuk menciptakan sistem baru atau memperbaiki dan mengganti sistem yang sudah ada (Hasan, 2019).

2.3.2. Sistem Informasi

Sistem adalah seperangkat komponen yang saling berhubungan dan saling bekerja sama untuk mencapai beberapa tujuan. Sistem dapat diartikan sebagai kumpulan atau himpunan dari unsur atau variabel-variabel yang terorganisasi, saling berinteraksi dan saling bergantung satu sama lain (Agustin, 2018).

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang (Agustin, 2018).

Sistem informasi adalah sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, membantu dan mendukung kegiatan operasi, bersifat manajerial dari suatu organisasi dan membantu mempermudah penyediaan laporan yang diperlukan (Effendi et al., 2022). Berikut beberapa jenis sistem informasi:

a. Sistem Informasi Keuangan

Sistem informasi keuangan merupakan sistem informasi yang memberikan informasi kepada orang atau kelompok baik di dalam perusahaan maupun di luar perusahaan mengenai masalah keuangan perusahaan.

b. Sistem Informasi Manufaktur

Sistem informasi manufaktur adalah sistem yang digunakan untuk mendukung fungsi produksi yang mencakup seluruh kegiatan yang terkait dalam pembayaran dan pembelian inventaris sekolah dan alat-alat tulis.

c. Sistem Informasi Sumber Daya Manusia

Sistem informasi sumber daya manusia adalah sistem informasi yang menyediakan informasi yang dipakai oleh fungsi personalia.

d. Sistem Informasi Manajemen

Sistem informasi manajemen merupakan bagian dari pengendalian internal suatu bisnis yang meliputi pemanfaatan manusia, dokumen, teknologi, dan prosedur oleh akuntansi manajemen untuk memecahkan masalah bisnis seperti biaya produk, layanan, atau suatu strategi bisnis.

2.3.3. Website

Website adalah keseluruhan halaman web yang terdapat dari sebuah domain yang mengandung informasi. Sebuah *Website* biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut dengan *hyperlink* sedangkan teks yang dijadikan media penghubung disebut *hypertext* (Putra & Wardani, 2019).

Secara umum, terdapat dua jenis *Website* berdasarkan sifatnya, yaitu *Website* statis dan *Website* dinamis. *Website* Statis adalah web yang mempunyai halaman tidak berubah. Sehingga untuk melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengedit *code* yang menjadi struktur dari *Website* tersebut. Sedangkan, *Website* Dinamis merupakan *Website* yang secara struktur diperuntukan untuk *update* sesering mungkin. Biasanya selain utama yang bisa diakses oleh *user* pada umumnya, juga disediakan halaman *backend* untuk mengedit konten dari *Website* (Hermanto & Suyudi, 2018).

2.3.4. Industri Tahu

Usaha pembuatan tahu merupakan salah satu industri pengolahan yang mempunyai prospek yang cerah. Kedelai yang dijadikan sebagai bahan baku dalam pembuatan tahu adalah kedelai yang berkualitas yang memiliki nilai gizi yang tinggi.

Produk yang diolah dilakukan dengan penanganan yang baik disertai dengan syarat-syarat teknis sanitasi dan higienis sesuai dengan standar mutu yang diinginkan. Dulunya tahu ini dianggap sebagai makanan masyarakat golongan bawah, namun saat ini tahu telah diterima oleh masyarakat golongan ekonomi menengah ke atas. Hal ini dikarenakan tahu merupakan bahan makanan yang memiliki rasa yang enak dengan harga yang murah. Tahu dapat dikonsumsi dengan cara digoreng atau digunakan sebagai bahan utama masakan (Yanto et al., 2022).

2.3.5. HTML

HTML (*Hyper Text Markup Language*) adalah sebuah bahasa pemrograman atau file teks yang berisi tag-tag markup yang berguna untuk memberitahukan *browser* bagaimana harus menampilkan sebuah halaman web.

Sebuah file HTML harus memiliki ekstensi htm atau html. HTML merupakan bahasa standar yang digunakan oleh *browser* internet untuk membuat halaman dan dokumen pada sebuah web yang kemudian dapat diakses dan dibaca layaknya sebuah artikel. HTML juga dapat digunakan sebagai *link* antara file-file dalam situs atau dalam komputer dengan menggunakan *localhost*, atau *link* yang menghubungkan antar situs dalam dunia internet (Pangestu & Afuan, 2021).

2.3.6. CSS

CSS (*Cascading Style Sheet*) adalah suatu teknologi bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih indah, rapi, terstruktur, dan seragam (Pangestu & Afuan, 2021).

CSS dipakai untuk memformat tampilan web seperti *style heading, border, navbar, body text, footer, images, sidebar*, dan *style* lainnya dengan tujuan supaya tampilan halaman web lebih bagus untuk dilihat, yang dibuat menggunakan bahasa HTML dan XHTML untuk dapat digunakan bersama-sama di dalam beberapa file (berkas) (Solahudin, 2021).

2.3.7. Python

Didirikan pada Desember 1989 oleh Guido Van, python adalah salah satu bahasa pemrograman umum tingkat tinggi yang paling populer. Menurut Grinberg python adalah bahasa pemrograman yang diketik secara dinamis, sering disebut pengetikan bebek dalam rekayasa perangkat lunak. Ini adalah bahasa yang ditafsirkan. Karena, bahasa ini tidak memerlukan waktu kompilasi terpisah, melainkan dikompilasi di *byte* kode dan dieksekusi secara instan.

Salah satu alasan popularitas Python adalah karena keterbacaan nya. Karena kode yang dapat dibaca adalah impian bagi seorang *software engineer*, itu penting karena pengembang harus dapat memahami pentingnya kode terlepas dari pembuatnya. Keterbacaan kode memang tergantung pada penulisnya. Tapi Python selangkah lebih maju dalam hal ini dengan sangat linguistik. Jenis bahasa pemrograman. Ini juga disebut-sebut sebagai cara mudah untuk mempelajari bahasa pemrograman. Ini sangat membantu *programmer* untuk mengurangi waktu yang dihabiskan untuk mempelajari bahasa pemrograman lain (Darmawan & Iswari, 2022).

2.3.8. Python Flask

Python flask adalah *framework* yang terbilang kecil menurut sebagian besar standar bagi *framework*. Terbilang cukup untuk disebut "*micro-framework*" dan cukup kecil apabila kita sudah terbiasa dengan python flask. Kita dapat membaca dan memahami semua sumber kode nya. Meski dapat terlihat kecil bukan berarti python flask bisa dianggap kurang dari *framework-framework* yang lain. Python flask dibangun sebagai *micro-framework* yang dapat diperluas dari bawah hingga ke atas. Itu berarti flask memiliki inti yang solid pada layanan dasarnya, sementara ekstensi yang dimilikinya menyediakan sisanya.

Kita juga dapat memilih paket ekstensi yang kita inginkan, karena dapat memilih paket ekstensi yang kita inginkan. Maka kita tidak akan memiliki kelebihan ukuran fail yang akan menambah berat ukuran fail yang kita miliki.

Menurut Grinberg python flask mempunyai beberapa dependensi utama untuk mendukungnya yaitu *Routing*, *debugging* dan *Web Server Gateway Interface* (WSGI). *Subsystems* python flask berasal dari "Werkzeug".

Support untuk *template* yang dimiliki python flask didukung oleh “Jinja2” dan untuk *command-line integration* berasal dari “Click”. Inilah semua dependensi yang dimiliki oleh python flask yang semua dependensi utamanya juga dimiliki oleh Armin Ronacher selaku dari pencipta python flask. Flask tidak memiliki dependensi utama atau bawaan pada sistem *database*-nya, pada validasi *web-forms*, *user authenticating*, dan *high-level tasks* lainnya. Layanan utama ini dan lainnya merupakan pelayanan utama yang biasanya terintegrasi dengan dependensi utama pada *framework* umum lainnya. Sebagai pengembang, pastinya kita memiliki rasa bimbang dalam memilih ekstensi yang cocok atau tepat dalam proyek yang kita bangun atau bahkan mungkin membuat sendiri apabila diperlukan. Ini tentu berbanding terbalik dengan *framework- framework* lainnya yang dapat dikatakan besar, dimana sebagai besar pilihan yang telah kita buat terkadang sulit atau tidak mungkin untuk diubah lagi (Darmawan & Iswari, 2022).

2.3.9. MySQL

MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak ber pemilik yang penggunaannya terbatas). MySQL salah satu *database* gratis dengan GNU sebagai lisensi-nya dan masuk ke jenis *Relational Database Management System* (RDBMS), dengan tabel, kolom, dan baris sebagai istilah untuk pemakaiannya (Solahudin, 2021).

RDBMS merupakan program yang membolehkan pengguna database buat membuat, mengelola serta memakai informasi pada sesuatu model relational. Sehingga, tabel-tabel yang terdapat pada database mempunyai kedekatan antara satu tabel dengan tabel yang lain.

Salah satu keunggulan MySQL adalah lebih cepat 3 hingga 4 kali dari pada *database* server komersial yang tersebar dikala ini, mudah diatur serta tidak membutuhkan seorang yang pakar buat mengendalikan administrasi pemasangan MySQL. Selain itu, MySQL juga sanggup membuat tabel berdimensi sangat besar. Dimensi optimal dari tiap tabel yang bisa terbuat dengan MySQL merupakan 4 GB hingga dengan dimensi file yang bisa ditangani oleh sistem pembedahan yang dipakai (Yulianti, 2021).

2.3.10. UML

UML (Unified Modeling Language) merupakan alat bantu, bahasa pemodelan yang dapat digunakan buat rancang bangun berorientasi objek. UML bisa digunakan guna spesifikasi, visualisasi serta dokumentasi sistem pada fase pengembangan (Yulianti, 2021).

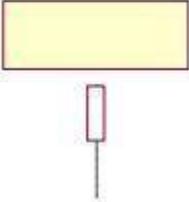
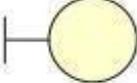
Jenis-jenis dari UML antara lain :

1. *Usecase diagram*

Menurut Tohari dalam Tabrani dan Aghniya menyimpulkan bahwa, “*usecase* adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor” (Kurniawan et al., 2020).

Menurut Pratama, “*Usecase diagram* adalah gambaran grafis dari beberapa atau semua aktor, *usecase*, dan interaksi diantaranya yang memperkenalkan suatu sistem. *Usecase diagram* tidak menjelaskan secara detail tentang pelanggan *usecase*, tetapi hanya memberi gambaran singkat hubungan antara *usecase*, aktor, dan sistem. Di dalam *usecase* ini akan diketahui fungsi-fungsi apa saja yang berada pada sistem yang dibuat” (Kurniawan et al., 2020). Disajikan pada Tabel 2.1 Simbol *usecase diagram*.

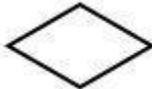
Tabel 2. 1. *Use case Diagram*

No.	Simbol	Nama	Keterangan
1.		Aktor	Aktor digunakan untuk merepresentasikan entitas yang berada di luar sistem dan berinteraksi dengan sistem
2.		<i>Lifeline</i>	<i>Lifeline</i> digunakan untuk menghubungkan objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya).
3.		<i>General</i>	<i>General</i> digunakan untuk merepresentasikan entitas tunggal dalam <i>sequence diagram</i> .
4.		<i>Boundary</i>	<i>Boundary</i> yaitu berupa tepi dari sistem, seperti <i>user interface</i> atau suatu alat yang berinteraksi dengan sistem yang lain.
5.		<i>Control</i>	<i>Control</i> adalah element mengatur aliran dari informasi untuk sebuah skenario. Objek ini umumnya mengatur perilaku dan perilaku bisnis.
6.		Entitas	Entitas adalah elemen yang bertanggung jawab menyimpan data atau informasi. Ini dapat berupa <i>beans</i> atau model <i>object</i> .
7.		<i>Activation</i>	<i>Activation</i> adalah suatu titik dimana sebuah objek mulai berpartisipasi di dalam sebuah <i>sequence</i> yang menunjukkan kapan sebuah objek

2. Activity diagram

Activity diagram ialah sesuatu yang menjelaskan tentang alir kegiatan dalam program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi, dan bagaimana sistem akan berakhir. Disajikan pada Tabel 2.2. Simbol-simbol *activity diagram*.

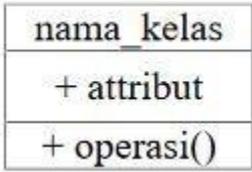
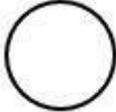
Tabel 2. 2. *Activity Diagram*

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / <i>Decision</i>	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / <i>Join</i>	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	<i>Swimlane</i>	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

3. Class diagram

Class diagram adalah visual dari struktur sistem program pada jenis-jenis yang di bentuk. *Class diagram* merupakan alur jalannya database pada sebuah sistem. Disajikan pada Tabel 2.3. Simbol-simbol dari *class diagram*.

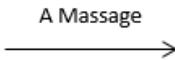
Tabel 2. 3. Simbol *Class Diagram*

Simbol	Nama	Keterangan
	Kelas	Kelas pada struktur sistem.
	Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
	Association	Relasi antarclass dengan arti umum, asosiasi biasanya juga disertai dengan Multiplicity.
	Directed Association	Relasi antarkelas dengan makna kelas yang atau digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
	Dependency	Relasi antarkelas dengan makna kebergantungan antarkelas'
	Aggregation	Relasi antarkelas dengan makna semua-bagian (whole-part)

4. Sequence diagram

Menurut Tohari dalam Tabrani dan Aghniya, menyimpulkan bahwa, “*sequence diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu”. *Sequence Diagram* adalah salah satu dari diagram-diagram yang ada pada UML, *Sequence Diagram* ini adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem (Kurniawan et al., 2020). Disajikan pada Tabel 2.4. Simbol-simbol dari *sequence diagram*.

Tabel 2. 4. *Sequence Diagram*

No	Simbol	Keterangan
1		<i>ACTOR</i> Menggambarkan orang yang sedang berinteraksi dengan sistem
2		<i>ENTITY CLASS</i> Menggambarkan hubungan kegiatan yang akan dilakukan
3		<i>BOUNDARY CLASS</i> Menggambarkan sebuah penggambaran dari form
4		<i>CONTROL CLASS</i> Menggambarkan penghubung antara boundary dengan tabel
5		<i>A FOCUS OF CONTROL & A LIFE LINE</i> Menggambarkan tempat mulai dan berakhirnya sebuah message
6		<i>A MESSAGE</i> Menggambarkan pengiriman pesan

2.3.11. Visual Studi Code

Menurut penelitian yang dilakukan oleh Agustini dan Wahyu Joni Kurniawan pengertian *Visual Studio Code* adalah *code editor* sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux* dan *macOS*. Ini termasuk dukungan untuk *debugging*, kontrol *git* yang tertanam dan *GitHub*, penyorotan sintaksis, penyelesaian kode cerdas, *snippet*, dan *refactoring* kode (Joni, 2019). Berikut ini beberapa fitur *Visual Code Studio* yang menjadikannya sebagai *software* editor paling banyak digunakan saat ini:

1) *Basic Editing*

Sesuai fungsinya sebagai *code editor*, kemampuan *Visual Code Studio* untuk coding tidak perlu diragukan lagi. *Software* ini mempunyai semua yang dibutuhkan *programmer*. Mulai dari *Keyboard Shortcuts*, *Multiple Selection*, hingga *Column Selection*. Bahkan, *Visual Code* juga menambahkan fitur *Auto Save* dan *Hot Exit*. Keduanya berfungsi untuk menyimpan file secara otomatis. Hal ini berguna untuk mencegah hal-hal yang tidak diinginkan, seperti lupa menyimpan file.

2) *IntelliSense*

IntelliSense adalah fitur *Visual Studio* yang dapat memudahkan proses coding. Cara kerjanya mirip dengan *Autocomplete*, yakni menyarankan keseluruhan kata berdasarkan apa yang *programmer* ketik. *IntelliSense* tersedia secara default untuk bahasa pemrograman JavaScript, TypeScript, JSON, HTML, dan CSS. Dan dapat menambahkan *IntelliSense* untuk bahasa pemrograman lain lewat *Extension Marketplace*.

3) *Debugging*

Fitur kunci lainnya di *Visual Code* adalah *Debugging*. Fitur ini berguna untuk memudahkan dalam mengedit, meng-compile, dan mengeksekusi kode berulang kali (looping). Secara default, *Visual Studio* menyediakan fitur *Debugging* untuk Node.js. Namun, *debugging* untuk bahasa pemrograman lain tetap dapat *programmer* unduh melalui *Extension Marketplace*.

4) *Extension Marketplace*

Extension Marketplace adalah fitur yang membuat *Visual Code Studio* unggul jauh dari para pesaing nya. Dengan *Extension*, Anda bisa dengan mudah menginstall *tools*, *debuggers*, bahkan bahasa pemrograman tambahan. Dengan begitu, proses *Website development* programmer jadi lebih cepat. Menariknya lagi, *Visual Code* memungkinkan para pembuat *Extension* berkontribusi secara fungsional, guna menghindari masalah kompatibilitas.

5) *GitHub Integration*

Fitur unggulan lain dari *Visual Code* adalah integrasi *GitHub*, *platform* manajemen *project* terpopuler di dunia. Di sini, programmer dapat berbagi code dan berkolaborasi dengan rekan kerja, tanpa perlu berpindah *software*. Cara menggunakannya juga sangat mudah. Programmer hanya perlu menyiapkan akun *GitHub*, kemudian menginstall *GitHub Pull Request and Issues* melalui *Extension Marketplace*.

2.3.12. Metode Pengujian

Pada proses perancangan perangkat lunak, perancangan pertama-tama berusaha membangun perangkat lunak dari konsep abstrak ke implementasi yang dapat dilihat, baru kemudian dilakukan pengujian.

Terdapat sejumlah aturan yang berfungsi sebagai sasaran pengujian.

- a. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
- b. *Test case* yang baik adalah *test case* yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
- c. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

1) *Black Box Testing*

Black Box Testing adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian *Black Box testing* harus dibuat dengan kasus benar dan kasus salah (Wulandari, 2020).

Black Box Testing untuk menguji suatu aplikasi, penguji tidak perlu memiliki pengetahuan terkait bahasa pemrograman tertentu. Penguji aplikasi tidak perlu memeriksa kode program. Memungkinkan penguji dan developer bekerja sama secara independen tanpa mengganggu proses kerja satu sama lain. Penguji mampu memahami dan melihat dari perspektif pengguna. Selain itu, *Black Box testing* memungkinkan penguji menganalisis kekurangan dari tahap awal pengujian secara cepat.