

BAB III METODOLOGI PENELITIAN

3.1 Objek Penelitian

3.1.1 Profil Bank BTN KCP Sepatan

Objek penelitian pada penelitian ini ialah BTN KCP Sepatan yang beralamatkan di JL. Raya Mauk, Km 11 No. 88, Sepatan, Pisangan Jaya, Tangerang, Kabupaten Tangerang, Banten 15520. PT. Bank Tabungan Negara, (persero), atau dikenal dengan Bank BTN khususnya Bank BTN KCP Sepatan merupakan Badan perjuangan milik negara (BUMN) yang bergerak di bidang jasa perbankan. Bank ini mempunyai komitmen untuk mendukung serta memberikan pelayanan pembiayaan untuk sector perumahan atau yang dikenal dengan layanan Kredit Kepemilikan Rumah (KPR). Bank BTN KCP Sepatan mempunyai peranan penting dalam pelayanan.

Metode yg dipergunakan berupa metode deskriptif kualitatif, yaitu dengan mengumpulkan data yg dilakukan melalui wawancara serta pengamatan terhadap proses pekerjaan yang dilakukan. Data dikumpulkan diawali memakai pengamatan atau observasi langsung diwilayah penelitian serta dilakukan wawancara. Metode penelitian merupakan mekanisme penyelesaian masalah penelitian serta menjelaskan metode yang digunakan selama penelitian ini. Untuk proses bisnis nya seperti :

Konfirmasi setoran uang dari outlet ke cabang jika melebihi batas limitasi outlet.

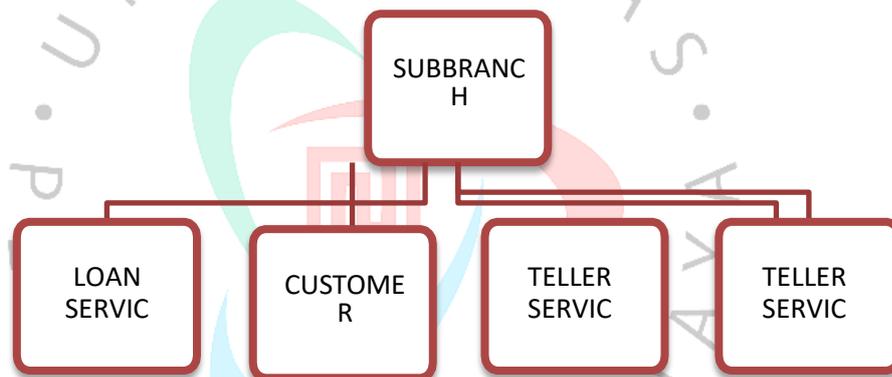
- a. *Teller* konfirmasi ke kas besar.
- b. Kas besar memberi konfirmasi ke *head teller*.
- c. *Head Teler* melakukan *approval*.
- d. Kas besar konfirmasi ke vendor.

- e. Vendor menjemput uang setoran.

Konfirmasi bon uang dari cabang ke outlet jika kas di outlet tidak cukup untuk transaksi.

- a. *Teller* konfirmasi ke kas besar.
- b. Kas besar menginfokan ada atau tidaknya jumlah uang yang akan dibon.
- c. Jika ada, *Head teller* melakukan *approval*.
- d. Kas besar konfirmasi ke vendor.
- e. Vendor mengantar uang yang dibon ke outlet.

3.1.2 struktur Organisasi Bank BTN KCP Sepatan



Gambar 3.1 struktur Organisasi Bank BTN KCP Sepatan

Berdasarkan gambar struktur organisasi yang ada di gambar 3.1, dapat dipandang bahwa berasal holistik rantai organisasi antar departemen saling berhubungan satu sama lain dan tidak terdapat yang terputus. Hal itu berarti PT. Bank Tabungan Negara, Tbk sudah mempunyai seni manajemen manajerial yang baik di organisasinya.

buat detail akan dijelaskan tentang susunan struktur jabatan dan *job description* secara lebih kentara :

Sub Branch Head

- *Monitoring.*
- *Controlling.*
- Menjalin hubungan dengan mitra.

- Menyetujui pelaksanaan akad kredit.
- *Mereview* dan menandatangani perjanjian kerjasama (PKS).

Loan Service

- Pelayanan keluhan.
- Pengajuan kredit.

Customer Service

- Pelayanan pembukaan rekening tabungan, giro, deposito.

Teller Service

- Melakukan input jumlah transaksi,, memastikan penerimaan *transfer*, no rekening lalu bank yg dituju.
- menuntaskan transaksi normal penyetoran dana lalu penarikan dana (tanpa memerlukan *approval* berasal atasan).
- Setoran transaksi harian.
- Melakukan pembayaran surat perintah kerja (SPK) tunai asal unit lain.

3.1.3 pengumpulan Data

Metode pengumpulan data yang penulis gunakan pada proposal ini terdapat tiga metode yaitu;

1. Observasi, gunakan cara mengamati, mengumpulkan, menyelidiki serta mencatat tertentu, dan dapat menghasilkan informasi akurat.
2. Wawancara, buat mendapat isu secara lengkap maka penulis melakukan metode tanya jawab pribadi.
3. Studi Pustaka, mengumpulkan teori-teori yg bersumber asal kitab juga jurnal-jurnal yg mampu mendukung penelitian ini. (Larasati & Masripah,2017:13)

3.1.4 Pendekatan Pengembangan Sistem

Metode pengembangan sistem yg digunakan dalam merancang sistem software e-CIT berbasis web pada PT. BTN KCP Sepatan gunakan metode SDLC (*Sistem Development Life Cycle*), dengan model Waterfall. “SDLC atau *Software Development LifeCycle* ialah proses berbagi atau mengganti Suatu sistem perangkat lunak memakai model contoh-contoh lalu metodologi yang dipergunakan orang buat menyebarkan sistem-sistem aplikasi sebelumnya”.)“Air terjun (*Waterfall*) tak jarang disebut juga model sekuensial linier (*sequential linear*) atau alur hayati klasik.

3.1.5 Perancangan Sistem

Pada tahap perancangan sistem, nantinya penulis akan merancang untuk mendeskripsikan sistem usulan secara akal, di penulisan ini memakai UML Diagram (*Unified Modelling Language Diagram*) yang fungsinya buat mempermudah pemahaman wacana sistem yang berjalan. Adapun UML diagram dari usulan prosedur sistem yang baru terdiri asal Use Case Diagram, Activity Diagram, Sequence Diagram serta class Diagram.

Desain Diagram Use Case

diagram Use Case ialah salah satu jenis diagram yang dipergunakan pada analisis dan desain perangkat lunak. Ini membantu dalam memodelkan interaksi antara sistem dan pengguna atau aktor-aktor lainnya yang terlibat dalam sistem. Diagram Use Case menyajikan fungsi-fungsi atau fitur-fitur yang bisa dilakukan oleh sistem dan bagaimana aktor-aktor ini berinteraksi dengan sistem tersebut.

Pada dasarnya, Diagram Use Case menggambarkan skenario-skenario yang berbeda di mana aktor-aktor menggunakan sistem untuk mencapai tujuan

tertentu. Aktor dapat menjadi pengguna nyata seperti pengguna akhir atau pengguna administratif, atau entitas yang berinteraksi dengan sistem seperti perangkat keras eksternal atau sistem lain.

Elemen utama dalam Diagram Use Case adalah sebagai berikut:

1. Aktor: Representasi dari entitas yang berinteraksi dengan sistem. Ini bisa berupa pengguna nyata atau entitas lain di luar sistem yang berinteraksi dengannya.
2. Use Case: Representasi dari suatu fungsi atau fitur yg dapat dilakukan sang sistem. Use case mendeskripsikan tindakan-tindakan yg dapat dilakukan oleh aktor dalam hubungannya dengan sistem.
3. Hubungan: Hubungan antara aktor dan use case menunjukkan bagaimana aktor terlibat dalam penggunaan fitur-fitur sistem. Hubungan ini dapat berupa asosiasi, inklusi, atau generalisasi/spesialisasi.

Diagram Use Case membantu dalam memahami kebutuhan pengguna, fungsionalitas yang harus diimplementasikan dalam sistem, serta interaksi antara aktor serta sistem. Ini juga dapat digunakan sebagai dasar untuk merancang antarmuka pengguna yang intuitif dan responsif.

Dengan menggunakan Diagram Use Case, tim pengembang perangkat lunak dapat berkomunikasi dengan lebih efektif dan memastikan bahwa semua kebutuhan pengguna terpenuhi. vakan digambarkan activity diagramnya.

Diagram Aktivitas (Activity Diagram) ialah galat satu jenis diagram yang dipergunakan dalam analisis dan desain perangkat lunak untuk memodelkan aliran aktivitas atau tindakan dalam suatu proses atau sistem. Diagram ini membantu dalam memvisualisasikan serangkaian langkah atau aktivitas yang terjadi dalam suatu proses secara urut.

Berikut adalah elemen utama dalam Diagram Aktivitas:

1. Aktivitas (Activity): Merepresentasikan tindakan atau langkah dalam proses. Aktivitas ini dapat berupa tindakan konkret seperti "mengirim email" atau tindakan abstrak seperti "verifikasi data". Setiap aktivitas direpresentasikan oleh sebuah persegi panjang dengan nama aktivitas di dalamnya.

2. Arah Aliran (Flow Arrow): Menunjukkan aliran atau urutan langkah-langkah dalam proses. Arah aliran digunakan untuk menghubungkan aktivitas-aktivitas dalam diagram. Aliran bisa berupa panah yang menunjuk ke aktivitas berikutnya atau kondisi yang mempengaruhi aliran.

3. Kondisi (Condition): Jika ada kondisi yang mempengaruhi aliran atau pengambilan keputusan dalam proses, kondisi dapat ditambahkan ke diagram untuk memodelkan percabangan atau perulangan. Kondisi ini direpresentasikan oleh rombongan dengan label yang menjelaskan kondisi tersebut.

4. Fork dan Join: Fork digunakan untuk membagi aliran menjadi beberapa jalur yang dapat dilakukan secara paralel, sedangkan join digunakan untuk menggabungkan kembali jalur-jalur tersebut. Fork direpresentasikan oleh tanda panah dengan dua garis paralel, sedangkan join direpresentasikan oleh tanda panah dengan dua garis yang berpotongan.

5. Awal (Initial Node) dan Akhir (Final Node): Awal menunjukkan titik awal dari proses atau aktivitas, sedangkan akhir menunjukkan titik akhir atau hasil dari proses. Awal direpresentasikan oleh lingkaran dengan tanda panah, sedangkan akhir direpresentasikan oleh lingkaran yang diisi dengan titik atau dengan tanda panah.

Diagram Aktivitas membantu dalam memahami aliran logika atau urutan

langkah-langkah dalam suatu proses atau sistem. Ini memudahkan tim pengembang perangkat lunak untuk merencanakan, menganalisis, dan merancang implementasi dari proses tersebut.

Dalam desain sistem, Diagram Aktivitas dapat digunakan untuk merancang proses bisnis, alur kerja aplikasi, atau logika tindakan dalam sistem. Diagram ini juga membantu dalam mengidentifikasi dan memahami tanggung jawab serta tugas-tugas yang wajib dilakukan pada sistem yang sedang dirancang.

Sequence Diagram (Diagram Urutan) ialah jenis diagram interaksi yang menggambarkan urutan pesan atau pemanggilan metode antara objek-objek dalam sistem. Diagram ini digunakan untuk memodelkan aliran komunikasi atau interaksi antara objek-objek dalam konteks skenario tertentu.

Berikut adalah elemen utama dalam Sequence Diagram:

1. Objek (Object): Objek merupakan instance dari suatu kelas atau entitas yang terlibat dalam interaksi. Setiap objek direpresentasikan oleh sebuah kotak vertikal dengan nama objek di dalamnya. Objek-objek ini saling berinteraksi dan saling mengirim pesan dalam urutan yang ditentukan.
2. Pesan (Message): Pesan mewakili komunikasi atau pemanggilan metode antara objek-objek. Ada dua jenis pesan utama dalam Sequence Diagram:

- Pesan Synchronous: Pesan synchronous adalah pesan di mana pengirim menunggu respons dari penerima sebelum melanjutkan. Ini direpresentasikan oleh garis panah lurus dengan label yang menunjukkan nama pesan.

- Pesan Asynchronous: Pesan asynchronous adalah pesan di mana pengirim tidak menunggu respons langsung dari penerima dan dapat melanjutkan tugasnya. Ini direpresentasikan oleh garis putus-putus dengan

label yang menunjukkan nama pesan.

3. Fokus (Lifeline): Fokus menunjukkan waktu hidup atau interval waktu di mana objek aktif dalam interaksi. Fokus direpresentasikan oleh garis vertikal yang terhubung ke objek dan menunjukkan interval waktu di mana objek terlibat dalam interaksi.

4. Pesan Return (Return Message): Pesan return menggambarkan respons atau nilai yang dikembalikan dari pemanggilan metode. Ini ditunjukkan oleh panah dengan label yang menunjukkan nama pesan return.

Sequence Diagram membantu dalam memodelkan urutan langkah-langkah dan interaksi antara objek-objek dalam sistem. Diagram ini memberikan pemahaman visual tentang bagaimana pesan dikirim dan diproses antara objek-objek, urutan yang diikuti, dan pesan-pesan yang dikembalikan. Sequence Diagram juga membantu dalam mengidentifikasi kebutuhan dan tanggung jawab objek-objek dalam sistem serta membantu dalam memahami alur eksekusi dalam suatu skenario atau proses.

Sequence Diagram sering digunakan dalam analisis dan desain perangkat lunak untuk memodelkan aliran logika atau interaksi antara objek-objek dalam sistem, membantu dalam mengidentifikasi kebutuhan fungsionalitas, dan merencanakan implementasi yang tepat.

3.1.6 Implementasi dan Pengujian Sistem

Implementasi sistem artinya tahapan penerapan perangkat lunak yang telah dilaksanakan, diterapkan serta didesain untuk lalu dijalankan sepenuhnya. Termin ini artinya tahap dimana sistem siap untuk dioperasikan, penulis menggunakan bahasa pemrograman PHP juga basis data nya MYSQL Pengujian sistem dilakukan memakai metode pengujian black box, Black box testing adalah pengujian yang dilakukan hanya mengamati dampak data uji serta menyelidiki fungsional dari

software. Pengujian black box, mengevaluasi hanya berasal tampilan luarnya (interface-nya), fungsionalitasnya, tidak perlu mengetahui bagaimana proses detilnya, yg diperlukan hanya mengetahui input juga output nya saja. Implementasi sistem ialah tahapan penerapan aplikasi yang telah dilaksanakan, diterapkan dan dirancang/dibuat buat lalu dijalankan sepenuhnya. Termin ini adalah termin dimana sistem siap buat dioperasikan, penulis menggunakan bahasa pemrograman PHP dan basis data nya MYSQL Pengujian sistem dilakukan menggunakan metode pengujian black box, Black box testing merupakan pengujian yg dilakukan hanya mengamati hasil hukuman melalui data uji serta menyidik fungsional asal perangkat lunak. Pengujian black box, mengevaluasi hanya berasal tampilan luarnya (interface-nya), fungsionalitasnya, tidak perlu mengetahui bagaimana proses detilnya, yg diperlukan hanya mengetahui input serta output nya saja.

3.2 Analisis Sistem Yang Berjalan

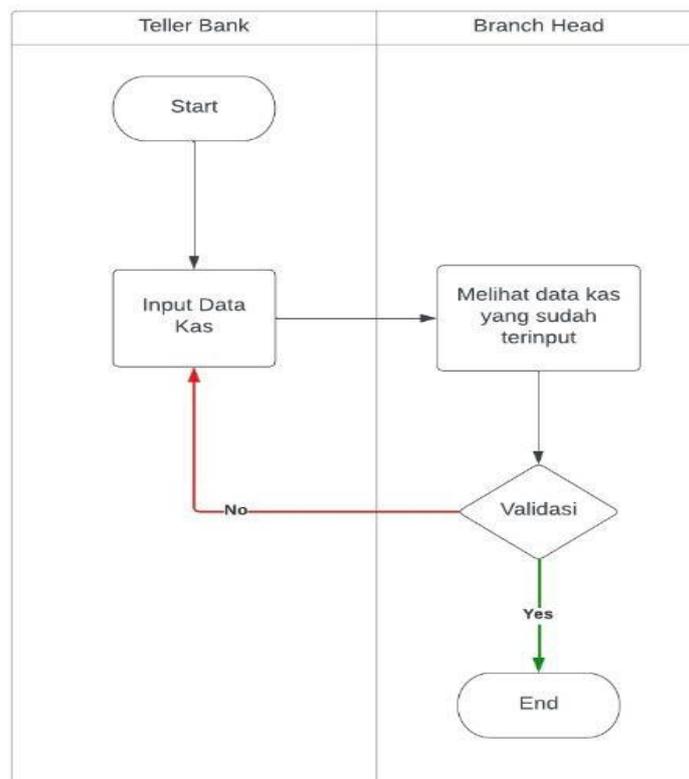
Bank BTN KCP Sepatan adalah satlah satu bank yang cukup ramai nasabahnya. Bank BTN ini terletak di jl. Raya Mauk, Km 11 No. 88, Sepatan, Kab. Tangerang. Setiap hari senin sampai dengan jumat bank BTN ini banyak dikunjungi oleh nasabah dengan kebutuhan seperti setoran tunai, penarikan, pembayaran jasa, serta transaksi lain nya, maka tak heran jika kas yang ada di KCP Sepatan cukup banyak jika dilihat dari transaksi setoran tunai setiap hari nya. Setiap akhir hari teller wajib untuk melakukan laporan posisi kas akhir nya kepada head teller dan juga kas besar , jika kas akhir yang ada tidak melebihi batas limitasi kas makan uang yang ada dapat dikeep di brankas KCP Sepatan, jika kas akhir hari melebihi batas limitasi outlet maka teller pun wajib untuk konfirmasi setoran ke cabang utama dengan head teller dan juga kas besar.

Proses konfirmasi setoran saat ini masih manual, maksudnya dengan mengirimkan pesan teks digrup yang beranggotakan para teller, head teller, dan juga kas besar. Dalam proses konfirmasi yang masih menerapkan system manual seperti ini seringkali membuat lama memberi respon untuk pendataan kas,serta membuat laporan tidak terdokumentasi dengan rapi sehingga sulit mencari data yang sudah terlampau lama.

Permasalahan tidak hanya terjadi di bagian Teller, tetapi di bagian kas besar dengan Head Teller yang terkadang keliru menginfokan terkait pengantaraan uang atau penyetorang uang melalui vendor.

3.2.6 Flowchart Transaksi Cash In Transit

Proses Transaksi Cash in Transit dilakukan pada saat kas akhir hari yang ada di outlet melebihi limit transaksi, maka Teller outlet harus melakukan konfirmasi setoran ke kantor cabang utama. Setelah melakukan konfirmasi setoran ke cabang utama, teller menyiapkan uang yang akan disetor dan uang yang akan disetor dijemput oleh mobil dinas ataupun vendor keesokan harinya untuk disetorkan ke bagian kas besar di cabang utama. Berikut flowchart untuk transaksi cash in transit:

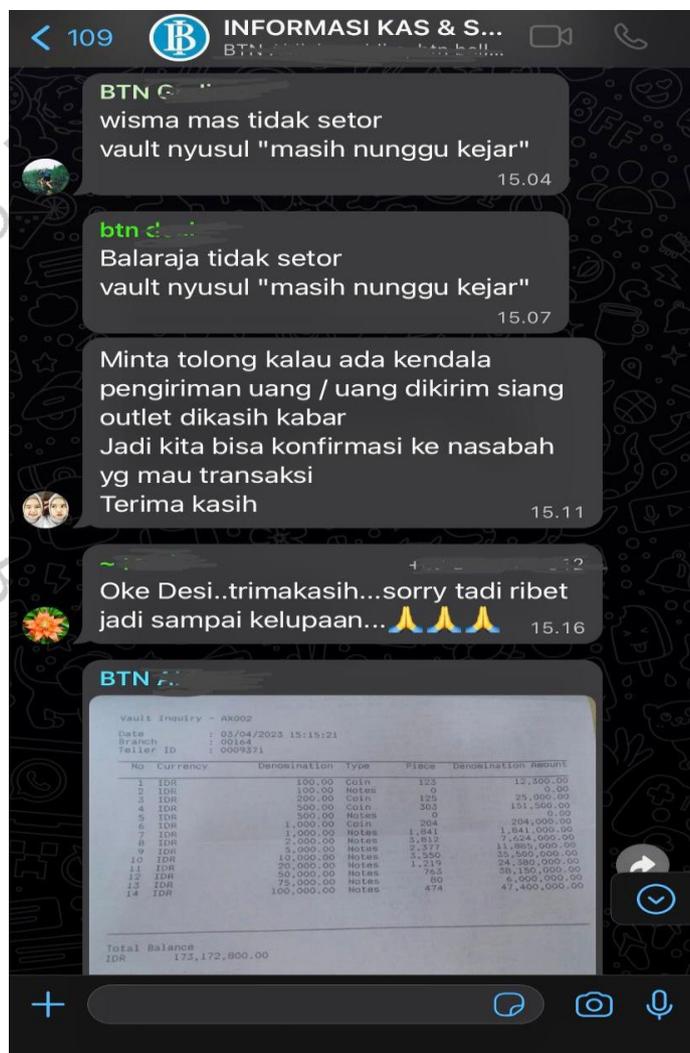


Gambar 3.2 Flow Chart CIT

Berdasarkan flowchart transaksi cash in transit, Teller akan menginput kas yang akan disetorkan ke cabang dan Branch Head akan melakukan pengecekan transaksi dan melakukan approval agar dapat tervalidasi.

3.1.6 Analisis Permasalahan

Berdasarkan hasil dari proses bisnis sistem yang berjalan terkait penyetoran dan pengantaran uang ke cabang, teridentifikasi beberapa permasalahan yang terjadi, salah satunya saat teller konfirmasi setor ataupun bon melalui grup chat terjadi kekeliruan karna disebabkan konfirmasi dari beberapa Teller yang tergabung di satu forum grup, adanya keliru informasi dari kas besar ke vendor yang mengakibatkan keterlambatan pengiriman atau pengantaran uang ke outlet outlet.



Gambar 3.3 Room Chat Teller

Berikut adalah salah satu contoh permasalahan yang terjadi di room chat teller dan juga bagian kas besar yang menyebabkan miss komunikasi antara teller dan juga bagian kas besar.

3.2.3 Analisis Kebutuhan

Dalam pembuatan program berbasis website untuk Cash in Transit (CIT), analisis kebutuhan akan menjadi langkah penting dalam merancang dan mengembangkan sistem tersebut. Berikut adalah uraian mengenai analisis kebutuhan yang dapat dilakukan dalam konteks pembuatan program berbasis website untuk CIT:

1. Identifikasi Kebutuhan Fungsional:

- Identifikasi fungsi-fungsi utama yang harus dilakukan oleh program CIT berbasis website, seperti pengelolaan pengguna, pengambilan uang, pengangkutan uang, pengiriman uang, dan pelaporan.
- Definisikan langkah-langkah atau aktivitas yang harus dilakukan dalam setiap fungsi tersebut, seperti proses pendaftaran pengguna, input data pengambilan uang, atau tampilan laporan keuangan.
- Tentukan persyaratan input dan output untuk setiap fungsi, misalnya formulir pendaftaran pengguna, halaman pengisian data pengambilan uang, atau tampilan grafik atau tabel untuk laporan.

2. Identifikasi Kebutuhan Non-Fungsional:

- Tentukan atribut-atribut penting yang harus dipenuhi oleh program CIT berbasis website, seperti keamanan data, responsifitas, skalabilitas, kompatibilitas dengan berbagai perangkat dan browser, dan tampilan antarmuka yang intuitif.
- Identifikasi batasan atau konstrain yang ada, misalnya batasan waktu untuk proses pengiriman uang atau persyaratan keamanan tertentu yang harus dipatuhi.

3. Pengumpulan Informasi:

- Lakukan wawancara atau diskusi dengan pihak terkait, seperti perusahaan pengelola uang atau layanan CIT yang ada, untuk memahami proses dan persyaratan yang sudah ada.

- Kumpulkan dokumen-dokumen terkait, seperti kebijakan atau pedoman yang berkaitan dengan CIT, aturan keamanan, atau kebutuhan kompatibilitas dengan sistem lain.

- Pelajari literatur dan penelitian terkait dalam domain CIT dan pembuatan aplikasi berbasis website buat mendapatkan pemahaman yg lebih baik ihwal kebutuhan yang harus dipenuhi.

4. Analisis dan Pemodelan:

- Gunakan teknik pemodelan, seperti use case diagram atau diagram alir data, untuk menggambarkan fungsi-fungsi utama program CIT dan hubungan antara fungsi-fungsi tersebut.

- Identifikasi pengguna yang akan menggunakan program, seperti administrator, petugas pengambilan uang, atau petugas pengiriman.

- Definisikan skenario penggunaan yang menggambarkan interaksi antara pengguna dan program CIT berbasis website, termasuk langkah-langkah yang harus dilakukan dan hasil yang diharapkan.

5. Verifikasi Kebutuhan:

- Diskusikan hasil analisis kebutuhan dengan tim pengembang atau pihak terkait lainnya untuk mendapatkan umpan balik dan memastikan bahwa kebutuhan yang diidentifikasi sudah tepat dan komprehensif.

- Lakukan review ulang terhadap kebutuhan yang telah diidentifikasi untuk memastikan kesesuaian dengan tujuan pembuatan program dan kebutuhan pengguna.

Analisis kebutuhan yang komprehensif akan memberikan dasar yang kuat untuk merancang dan mengimplementasikan program CIT berbasis website yang efektif dan sesuai dengan kebutuhan pengguna. Selama proses analisis kebutuhan, penting untuk terus berkomunikasi dengan pihak terkait dan memperhatikan perubahan kebutuhan yang mungkin terjadi selama pengembangan.