

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

2.1.1. Pengertian Sistem Informasi

Sistem informasi yaitu sekumpulan elemen atau komponen yang saling berhubungan dan bertugas mengumpulkan, menyimpan, mengolah, dan berbagi informasi yang nantinya dapat digunakan untuk mengambil keputusan. (Nugraha & Munawar, 2022).

Komponen sistem informasi terdiri dari masukan berupa sumber data, pengolahan data menjadi informasi, keluaran berupa produk data. Sistem kontrol dan dukungan penyimpanan data.

Sistem informasi tersebut didukung oleh 5 pilar (Nugraha & Munawar, 2022), yaitu:

1. *Hardware Resources* (Perangkat Keras);
2. *Software Resources* (Perangkat Lunak);
3. *People Resource* (Pengguna);
4. *Data Resource* (Sumber Data);
5. *Network Resource* (Sumber Jaringan);

Sistem informasi terdiri dari komponen-komponen yang disebut Blok Bangunan (*Building Block*), Blok Masukan (*Input Block*), Blok Model (*Model Block*), Blok

Keluaran (*Output Block*), Teknologi (*Technology*), Blok Basis Data (*Database Block*), Blok Kendali (*Control Block*).

2.1.2. Pengertian Sistem Informasi Penjualan

Sistem informasi penjualan merupakan sebuah sistem berbasis komputer yang dapat membuat pencatatan, menghitung, membuat dokumen dan data, penjualan secara terkendali, untuk menghasilkan laporan yang cepat, relevan, dan mudah. (Risti, 2022).

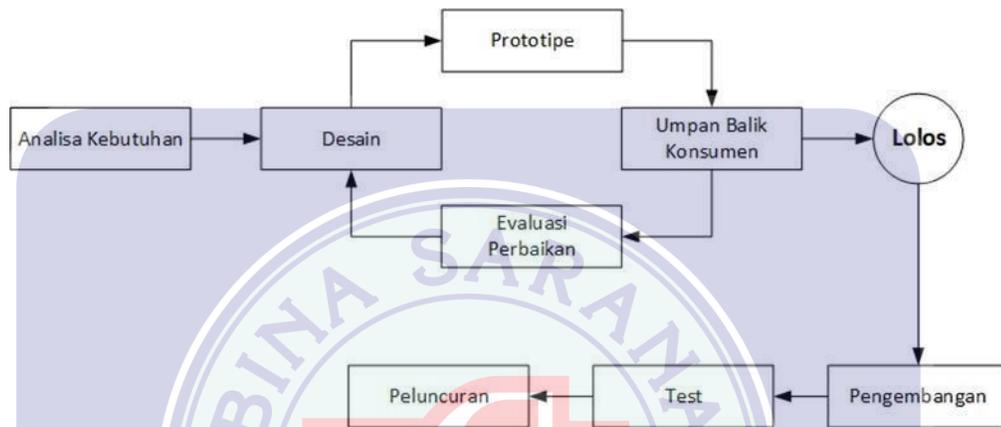
Sistem informasi penjualan adalah sistem komputerisasi itu membantu departemen penjualan memecahkan masalah informasi penjualan yang tidak efisien, akurat dan hemat data serta menghasilkan informasi penjualan lebih cepat daripada sistem manual.

Sistem informasi penjualan adalah sistem informasi terkomputerisasi berguna sebagai pengelola Pemesanan penjualan, yang memungkinkan pelaporan dan berbagi informasi penjualan harian.

2.1.3. Model Prototipe (*System Prototyping*)

Model Prototipe digunakan Ketika pengguna hanya menentukan tujuan umum perangkat lunak tanpa menentukan persyaratan input, pemrosesan, dan output. Selain itu, metodologi ini juga memiliki alternatif lain seperti *Throw-away Prototyping Model* metode ini mirip dengan metode berdasarkan *prototype*. Metode ini memungkinkan pengguna memberikan masukan dan saran untuk perbaikan sebelum sistem tersebut diimplementasikan secara penuh. Sistem prototyping juga memungkinkan pengembangan untuk mengidentifikasi masalah dan kesalahan dalam sistem sebelum diimplementasikan secara penuh.

Model Prototipe atau lebih dikenal dengan *System Prototyping*. Karena dengan metode ini dapat memudahkan dalam penyampaian ide baik secara visual ataupun rancangan (*blueprint*) kepada pembuat maupun pengguna (Sujono, Mayasari, & Koloniawan, 2019).



Gambar II.1 Diagram Sistem *Prototyping*

Sumber: (Sujono, Mayasari, & Koloniawan, 2019)

Dalam menggunakan metodologi prototipe ini pengembangan perangkat lunak terbagi menjadi beberapa tahapan, diantaranya:

6. Analisa Kebutuhan

Tahapan analisis kebutuhan merupakan tahapan paling awal dalam proses pengembangan sistem *prototyping*. Pada tahapan ini, pengembang harus memahami apa saja kebutuhan pengguna dan dapat pula memahami permasalahan yang nantinya dapat dipecahkan oleh sistem yang diproses. Selin itu pengembang juga diharuskan mampu mengidentifikasi berbagai kebutuhan diantaranya kebutuhan fungsional maupun kebutuhan non-fungsional dari sistem tersebut.

7. *Design Prototyping*

Setelah analisa kebutuhan pengguna dipahami, kemudian pengembang dapat merancang sebuah prototipe dari sistem yang diperlukan. Pada tahapan ini, pengembang harus dapat merancang antarmuka bagi pengguna, serta dapat mengidentifikasi fitur-fitur utama dari sistem yang dibuat, dan merancang arsitektur sistem yang dirancang sebelumnya.

8. Pengembangan

Setelah perancangan design prototipe selesai, kemudian pengembang dapat memulai mengembangkan prototipe dari sistem yang telah dibuat. Pada tahapan ini, pengembang harus dapat membangun sebuah sistem dengan menggunakan teknologi yang sesuai dengan permasalahan yang ada dan mengimplementasikan fitur-fitur utama dari sistem yang sudah dibuat.

9. Test

Kemudian setelah prototipe sistem selesai dikembangkan, maka pengembang harus menguji atau test sistem untuk dapat nantinya memastikan bahwa sistem yang digunakan dapat berfungsi dengan baik dan dapat memenuhi kebutuhan pengguna. Pengujian atau test prototipe juga dapat memungkinkan pengguna untuk dapat memberikan sebuah umpan balik dan saran untuk perbaikan sistem yang sudah dibuat.

10. Peluncuran

Kemudian setelah prototipe sistem telah diuji dan diperbaiki, maka pengembang dapat pula mengimplementasikan hasil dari sistem yang dibuat secara penuh. Pada tahapan ini, seorang pengembang harus dapat memastikan bahwa sistem dapat berfungsi dengan baik dan dapat memenuhi kebutuhan penggunanya.

2.2. Teori Pendukung

2.2.1. ERD (*Entity Relationship Diagram*)

ERD (Entity Relationship Diagram) adalah sebuah representasi grafis untuk mengilustrasikan konsep desain secara konseptual dari basis data yang akan di bangun. (Muharamani, Regawan, & Kurniawati, 2023).

2.2.2. LRS (*Logical Record Structure*)

LRS (Logical Record Structure) adalah uraian lebih rinci tentang rincian tentang structure record dalam tabel dari hasil antara kumpulan entitas. (Muharamani, Regawan, & Kurniawati, 2023).

2.2.3. UML (*Unified Modelling Language*)

Unified Modeling Language, atau *UML* merupakan bahasa visual untuk memodelkan juga mengkomunikasikan system menggunakan diagram dan skrip pendukung. (Elis & Voutama, 2022).

Unified Modeling Language, atau *UML* terbagi menjadi beberapa jenis dan memiliki fungsi spesifik yang berbeda, diantaranya:

1. Use Case Diagram

Pemodelan perilaku (*behavior*) dari sistem informasi yang akan diimplementasikan. Interaksi antara satu atau lebih aktor dalam sistem informasi yang akan dibuat dapat digambarkan dengan use case diagram.

2. Activity Diagram

Activity Diagram adalah gambaran alur atau operasi dari suatu sistem yang termasuk dalam program.

3. *Sequence Diagram*

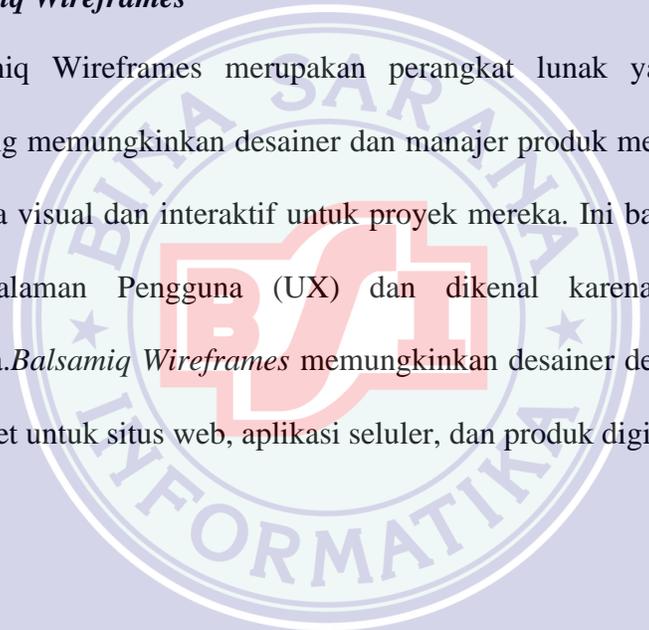
Sequence Diagram adalah objek dijelaskan dalam kasus penggunaan, menjelaskan pesan dikirim dan diterima antara objek dan umur objek.

4. **Class Diagram**

Class Diagram adalah struktur sistem yang mendefinisikan kelas-kelas yang dibuat untuk membangun sistem.

2.2.4. *Balsamiq Wireframes*

Balsamiq Wireframes merupakan perangkat lunak yang kuat dan mudah digunakan yang memungkinkan desainer dan manajer produk membuat wireframe yang menarik secara visual dan interaktif untuk proyek mereka. Ini banyak digunakan dalam Desain Pengalaman Pengguna (UX) dan dikenal karena kesederhanaan dan keefektifannya. *Balsamiq Wireframes* memungkinkan desainer dengan cepat dan mudah membuat maket untuk situs web, aplikasi seluler, dan produk digital lainnya.



UNIVERSITAS