Menurut Sukamto & Shalahuddin (2018:50), “Entity Relationship Diagram (ERD) adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”.

Sedangkan menurut Fathansyah (2015:81), “Entity Relationship Diagram (ERD) adalah komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta di dunia nyata’.

Menurut Suprapto(2018a:49), “Entity Relationship Diagram (ERD) merupakan salah satu cara umum untuk menyatakan hasil analisis kebutuhan data(requirement analysis) pada fase awal perancangan database”

<https://repository.bsi.ac.id/index.php/unduh/item/269115/File_10-Bab-II-Landasan-Teori.pdf>

**DAFPUS**

1. **Sistem antrian**

**Penulis:** Sari, Y. W., & Rusydi, M. I. (2020)

**Isi:** mendefinisikan sistem antrian sebagai sistem yang digunakan untuk mengatur proses antrean dalam sebuah layanan pelayanan atau produksi. Sistem antrian ini berfungsi untuk mengelola jumlah pelanggan atau pengguna layanan yang datang dengan jumlah pelayanan atau produksi yang tersedia.

Selain itu, penulis juga menjelaskan bahwa sistem antrian memiliki beberapa elemen penting, seperti pelanggan atau pengguna layanan, pelayan atau petugas, proses pelayanan atau produksi, dan sistem pengaturan antrian yang meliputi metode pengaturan antrian dan alat yang digunakan untuk pengaturan antrian. Sistem antrian yang baik akan meminimalkan waktu tunggu pelanggan atau pengguna layanan serta meningkatkan efisiensi pelayanan atau produksi.

**Nama jurnal:** Evaluasi kinerja sistem antrian pada loket pelayanan umum menggunakan metode simulasi. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 4(12), 5345-5352.

Waktu ambil : 03/04/2023, 22.05

1. **Pelayanan**

pelayanan adalah setiap tindakan atau serangkaian tindakan yang dilakukan untuk memenuhi kebutuhan atau keinginan pelanggan. Pelayanan dapat berupa kegiatan yang dilakukan oleh manusia atau mesin dan mencakup berbagai jenis aktivitas, mulai dari interaksi tatap muka hingga layanan jarak jauh yang dilakukan secara online. Pelayanan juga dapat berlangsung dalam berbagai jenis lingkungan, seperti restoran, bank, rumah sakit, atau dalam bentuk transportasi."

**Daftar pustaka**

Fitzsimmons, J. A., & Fitzsimmons, M. J. (2019). Manajemen Pelayanan: Operasi, Strategi, Teknologi Informasi (edisi ke-9). (D. Harjanto & A. H. Mawardi, Trans.). Penerbit Salemba Empat.

1. **Pengertian aplikasi**

**Dafpus**

Hasugian, J. (2014). Pemrograman Visual dengan Delphi. Jakarta: Andi Offset.

1. **Konsep dasar pemrograman**

"Pemrograman adalah proses membuat, menguji, dan memelihara kode sumber dari program komputer. Tujuan utama dari pemrograman adalah untuk menemukan solusi untuk masalah tertentu melalui penulisan kode sumber. Proses pembuatan kode sumber melibatkan memahami kebutuhan penggunaan dari program, merancang solusi untuk masalah yang ditemukan, menulis kode sumber yang sesuai dengan bahasa pemrograman yang dipilih, menguji dan memperbaiki kesalahan yang terjadi, dan akhirnya memelihara kode sumber tersebut untuk memastikan kinerja yang optimal.

**Dafpus**

Wikipedia. (2023, 4 April). Pemrograman. Diakses pada 4 April 2023, dari [https://id.wikipedia.org/wiki/Pemrograman](https://id.wikipedia.org/wiki/Pemrograman" \t "https://chat.openai.com/_new)

1. **Metode pengembangan waterfall**

Metode waterfall menurut Roger S. Pressman adalah model pengembangan perangkat lunak yang mengikuti alur sequensial dan linear dari tahap awal hingga tahap akhir. Metode ini terdiri dari lima tahap utama, yaitu:

1. Analisis Kebutuhan (Requirements Analysis): Tahap ini merupakan tahap awal dalam pengembangan perangkat lunak. Pada tahap ini, kebutuhan pengguna ditentukan dan dianalisis untuk menghasilkan persyaratan sistem yang jelas dan terstruktur.
2. Perancangan (Design): Setelah kebutuhan pengguna dianalisis, selanjutnya adalah merancang solusi yang tepat untuk memenuhi kebutuhan tersebut. Tahap ini mencakup perancangan arsitektur sistem, perancangan modul, dan perancangan antarmuka pengguna.
3. Implementasi (Implementation): Tahap ini adalah proses pengembangan dan pembuatan kode program yang telah dirancang sebelumnya. Kode program ini kemudian diuji dan diuji kembali untuk memastikan kecocokan dengan persyaratan sistem.
4. Pengujian (Testing): Tahap ini mencakup pengujian keseluruhan sistem untuk memastikan bahwa semua persyaratan sistem terpenuhi. Pada tahap ini, tes fungsional, tes kesalahan, dan tes keamanan dilakukan untuk memastikan kualitas sistem.
5. Pemeliharaan (Maintenance): Tahap ini adalah proses pemeliharaan sistem setelah sistem telah diterapkan di lingkungan produksi. Hal ini mencakup perbaikan bug, peningkatan kinerja, dan perubahan persyaratan sistem.

Metode waterfall memiliki keuntungan karena alur pengembangan yang linear memudahkan pengendalian dan manajemen proyek, namun juga memiliki kelemahan karena sulitnya mengakomodasi perubahan kebutuhan pengguna yang muncul di tengah jalan.

**Dafpus**

Pressman, R. S. (2014). Rekayasa Perangkat Lunak: Pendekatan Praktisi. New York: McGraw-Hill Education.

1. **UML**

UML atau Unified Modeling Language adalah bahasa pemodelan visual yang digunakan untuk merancang dan memodelkan sistem perangkat lunak. Menurut Kroenke et al, UML adalah sebuah notasi standar yang digunakan untuk merepresentasikan desain sistem perangkat lunak secara visual. UML terdiri dari sejumlah diagram dan notasi yang dapat digunakan untuk menggambarkan berbagai aspek sistem, seperti struktur data, perilaku sistem, antarmuka pengguna, dan interaksi antar objek.

Beberapa diagram UML yang paling umum digunakan termasuk:

Diagram use case: Digunakan untuk menggambarkan hubungan antara pengguna dan sistem, serta fitur dan fungsionalitas yang disediakan oleh sistem.

Diagram kelas: Digunakan untuk menggambarkan struktur data dalam sistem dan hubungan antara kelas-kelas tersebut.

Diagram aktivitas: Digunakan untuk menggambarkan alur kerja dalam sistem dan aktivitas yang dilakukan oleh objek dalam sistem.

Diagram sekuen: Digunakan untuk menggambarkan interaksi antara objek dalam sistem.

Diagram komponen: Digunakan untuk menggambarkan komponen-komponen sistem dan hubungan antara komponen tersebut.

UML sangat penting dalam pengembangan sistem perangkat lunak karena dapat membantu dalam menggambarkan dan memahami kompleksitas sistem secara visual. Hal ini memudahkan pengembang untuk berkolaborasi dan berkomunikasi dengan tim pengembangan lainnya serta pemangku kepentingan lainnya.

1. **Sistem antrian**
2. **Sistem antrian**