

BAB II

TINJAUAN PUSTAKA

2.1. Teori Terkait

Rancang Bangun Aplikasi Kependudukan Berbasis *Web* Di Desa Kedungrejo Waru-Sidowarjo. Jurnal ini buat oleh Efi Widyawati, Ari Kurniawan Jurusan Informatika, Fakultas Teknik, Universitas Negri Surabaya. Sistem Informasi Kependudukan adalah bagian dari sistem informasi administrasi kependudukan negara yang memiliki cakupan lebih luas. Saat ini, sistem kependudukan di Desa Kedungrejo Waru-Sidowarjo masih dilakukan secara manual, sehingga kurang efisien dalam pengelolaan informasi. Proses *input* data oleh pihak desa yang masih *manual* serta penyimpanan berkas yang terus bertambah menyebabkan ruang penyimpanan semakin sempit dan meningkatkan risiko hilang atau rusaknya berkas lama. Tahapan penelitian meliputi perancangan sistem, analisis sistem, desain sistem, pemrograman, uji coba *program*, implementasi sistem, dan pemeliharaan sistem. Dari hasil pembuatan *website* desa kependudukan di desa Kedugrejo Waru-Sidowarjo, dapat menyediakan informasi dengan cepat dan akurat. Aplikasi kependudukan ini akan membantu dan mempermudah pencarian informasi kependudukan, selain itu juga mencegah penumpukan berkas, mengurangi kebutuhan ruang penyimpanan, meminimalkan kesalahan yang mungkin terjadi, dan memudahkan pembuatan laporan kependudukan.[2]

Rancang Bangun Sisten Informasi Pelayanan Masyarakat Berbasis *Web* Di Desa Bojong. Jurnal ini dibuat oleh Racmad Destriana, Hengking Rusdianto, Rahmat Fauzan Ardi Gustama, Program studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Tangerang. Pemerintah berperan sebagai pelayan masyarakat, khususnya dalam melaksanakan tugas-tugas umum melalui pembentukan sistem administrasi. Di tingkat desa, proses layanan administrasi kependudukan kepada masyarakat masih dilakukan secara manual. Masyarakat harus datang ke kantor desa untuk mengetahui prosedur layanan, sehingga waktu yang dibutuhkan untuk pelayanan seringkali melebihi standar waktu maksimal yang telah ditetapkan. Penelitian ini dilakukan untuk mengetahui sejauh mana pengaruh sistem surat-menyurat di Balai Desa Bojong.[3]

Rancang Bangun Sistem Informasi Pelayanan Masyarakat Berbasis *Web* Pada Desa Sambeng Kulon Kabupaten Banyumas. Jurnal ini disusun oleh Supriatiningsih, Mahmud Safudin, dan Eko Yulianto, mahasiswa Universitas Bina Sarana Informatika. Jurnal ini membahas sistem pelayanan masyarakat di Desa Sambeng Kulon, Kecamatan Kembaran, Kabupaten Banyumas, yang merupakan instansi pemerintah yang menangani data kependudukan. Dalam prosesnya, penduduk harus datang langsung ke desa dengan membawa semua persyaratan yang diperlukan. Setelah semua persyaratan lengkap, staf desa mengisi formulir yang dibutuhkan secara manual. Hal ini dapat memperlambat proses pelayanan dan meningkatkan risiko kesalahan dalam penulisan. Selain itu, banyak masyarakat

mengeluhan lamanya proses yang mempengaruhi waktu menunggu. Dengan demikian artinya bahwa kualitas pelayanan dikantor Kelurahan Klapagading masih belum begitu memuaskan, artinya masih sangat perlu upaya-upaya peningkatan kualitas pelayanan tersebut. Sehingga tingkat kepuasan masyarakat yang dilayani dapat dicapai. Hasil dari Penelitannya adalah membuat sistem informasi masyarakat berbasis *web* dirancang sebagai solusi bagi pihak desa sambang kulon untuk mengelolah data secara cepat dan tepat sehingga mempercepat proses pengolahan data memaksimalkan hasil kerja.[4]

2.2. Landasan Teoritis

2.2.1. Website

Website, yang juga dikenal sebagai situs, situs *web*, atau portal, adalah kumpulan halaman *web* yang saling berhubungan. Halaman pertama dari sebuah *website* disebut *home page*, sedangkan halaman-halaman berikutnya secara individu disebut *web page*. Dengan kata lain, *website* adalah situs yang dapat diakses dan dilihat oleh pengguna *internet* di seluruh dunia.[5]

2.2.2. PHP



Gambar 2.1. Logo *PHP*

PHP, atau *Personal Home Page*, adalah bahasa pemrograman *web* atau *scripting language* yang dirancang khusus untuk pengembangan *web*. *PHP* pertama kali diciptakan oleh *Rasmus Lerdorf* dengan tujuan awal menghitung jumlah pengunjung di halaman *website*. Bahasa pemrograman *PHP* dapat digabungkan dengan *HTML*, dengan sintaks khusus yang dimulai dengan tanda *tag* pembuka diikuti tanda tanya (<?) dan diakhiri dengan tanda tanya diikuti *tag* penutup (?>).[6]

2.2.3. MySQL



Gambar 2.2. Logo *MySQL*

MySQL adalah salah satu jenis *database server* yang sangat populer, karena menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database*. *MySQL* bersifat *Open Source* dan dilengkapi

dengan *source code* (kode yang digunakan untuk membuat *MySQL*).[7]

2.2.4. Apache

Apache HTTP Server merupakan perangkat lunak yang berjalan pada berbagai *platform* sistem operasi (OS) yang *mendukung multi-tasking*, dan menyediakan layanan bagi aplikasi lain yang terhubung ke dalamnya, seperti *web browser (web server)*. Awalnya, *Apache* dikembangkan untuk beroperasi dengan sistem operasi *Linux/Unix*, namun kemudian diadaptasi untuk berfungsi di bawah berbagai sistem operasi lainnya, termasuk *Windows* dan *Mac*. [8]

2.2.5. Domain dan Hosting

Domain adalah sebuah nama unik yang digunakan untuk mengidentifikasi *server hosting* dan memudahkan dalam pemanggilan atau penyebutan informasi serta data dalam suatu *server*. *Posting* merupakan media untuk menyimpan *file-file* di dalam sebuah situs. Dalam analogi, *domain* dapat diibaratkan sebagai alamat, sementara *hosting* adalah rumah dari alamat tersebut. [9]

2.2.6. Database

Database atau basis data adalah kumpulan informasi yang tersimpan secara teratur di dalam komputer, memungkinkan penggunaan *program* komputer untuk memeriksa dan mengakses informasi dari basis data tersebut. Secara konseptual, basis data merupakan kumpulan data yang saling terkait, membentuk *file* yang berhubungan satu sama lain dengan cara tertentu untuk menghasilkan data baru atau informasi. Atau dapat dijelaskan sebagai kumpulan data yang terkait satu sama lain, diorganisir berdasarkan skema atau struktur tertentu. Basis data disimpan dalam perangkat keras penyimpanan komputer dan dikelola dengan menggunakan perangkat lunak khusus. Hubungan atau relasi antar data sering kali diindikasikan dengan menggunakan kunci (*key*) dari setiap *file* yang ada. [10]

2.2.7. Visual Studio Code



Gambar 2.3. Logo *Visual Studio Code*

Visual Studio Code merupakan *editor* kode sumber yang dibuat oleh *Microsoft* untuk digunakan pada sistem operasi *Windows*, *Linux*, dan *macOS*. *Editor* ini memiliki fitur dukungan *debugging*, integrasi

kontrol *git* yang telah disematkan, penyorotan sintaksis, penyelesaian kode cerdas, *snippet*, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi, dan *menginstal* ekstensi yang menambah fungsionalitas tambahan.[11]

2.2.8. Administrasi Kependudukan

Administrasi Kependudukan merujuk pada serangkaian tindakan pengorganisasian dan pengaturan dalam pengeluaran dokumen dan informasi kependudukan melalui proses pendaftaran penduduk, pencatatan sipil, pengelolaan data administrasi kependudukan, serta pemanfaatan hasilnya untuk pelayanan masyarakat dan pembangunan sektor lainnya.[12]

2.2.9. XAMPP



Gambar 2.4. Logo *XAMPP*

XAMPP adalah sebuah aplikasi yang memungkinkan komputer kita berfungsi sebagai *server*. Manfaat dari *XAMPP* adalah untuk menciptakan jaringan lokal sendiri di mana kita dapat membuat *website* secara *offline* di komputer sendiri untuk uji coba. Dengan *XAMPP*, komputer kita dapat berperan sebagai *server website* yang

memungkinkan kita untuk menggunakannya. Istilah "*server*" digunakan karena komputer kita harus memberikan layanan untuk mengakses situs *web*, sehingga komputer dapat menjadi *server*. Dapat disimpulkan *xampp* adalah aplikasi *tools* untuk menyediakan paket lunak yang berisi konfigurasi *Web Server, Apache, PHP, MySQL* untuk membantu kita dalam proses pembuatan aplikasi *web* yang menyatu menjadi satu sehingga memudahkan kita dalam membuat *program web*. [13]

2.2.10. Disdukcapil

Pengertian dinas adalah bagian kantor pemerintah yang mengurus pekerjaan tertentu atau segala sesuatu yang bersangkutan dengan jabatan (pemerintah). Catatan Sipil adalah pencatatan peristiwa penting yang dialami oleh seseorang dalam *register* pencatatan sipil pada instansi pelaksana. (UU Republik Indonesia No. 23 tahun 2006). Jadi yang dimaksud dengan dinas catatan sipil adalah bagian kantor yang mengurus pekerjaan pencatatan peristiwa yang dialami oleh seseorang dalam *register* pencatatan sipil pada instansi pelaksana. [14]

2.2.12. UML

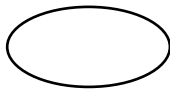
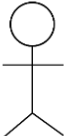


Unified Modeling Language merupakan salah satu metode pemodelan *visual* yang digunakan dalam perancangan dan pembuatan

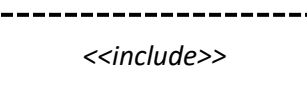
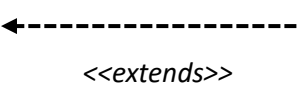
sebuah *software* yang berorientasikan pada objek. UML merupakan sebuah standar penulisan dimana didalamnya terdapat sebuah proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik.[15] Terdapat beberapa diagram UML yang sering digunakan dalam pengembangan sebuah sistem, yaitu :

A. Use Case Diagram

Use case merupakan gambaran dari fitur atau fungsionalitas dari sebuah sistem, dan menggambarkan interaksi antara aktor dengan sistem tersebut. Di dalam *use case*, terdapat aktor atau sistem yang melakukan tugas di dalam sistem tersebut.

Tabel 2.1. Simbol *Use Case* Diagram




No	Simbol	Keterangan
1		<i>Use Case</i> : Abstraksi dan interaksi antara sistem dan aktor.
2		<i>Actor</i> : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i> .
3		Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung.
4		<i>Generalisation</i> , Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila


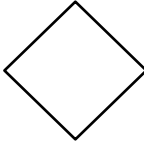
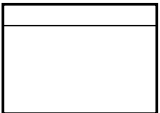
No	Simbol	Keterangan
		aktor berinteraksi secara pasif dengan sistem.
5		<i>Include</i> , Menunjukkan spesialisasi <i>actor</i> untuk dapat berpartisipasi dengan <i>use case</i> .
6		<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

B. Activity Diagram

Merupakan gambaran alir dari aktivitas-aktivitas didalam sistem yang berjalan. Digunakan untuk menggambarkan struktur statis dari sistem, termasuk kelas-kelas yang ada, atribut-atributnya, serta hubungan antara kelas-kelas.

Tabel 2.2. Simbol *Activity Diagram*



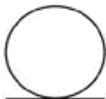
No	Simbol	Keterangan
1		<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
2		<i>End Point</i> , akhir aktivitas.
3		<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.

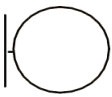


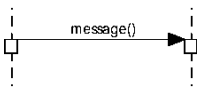

No	Simbol	Keterangan
4		<i>Fork</i> /penggabungan dimana yang lebih dari satu kegiatan digabungkan menjadi satu.
5		<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
6		<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa dan melakukan apa.

C. Sequence Diagram

Menggambarkan interaksi antar objek didalam dan di sekitar sistem yang berupa *message* yang digambarkan terhadap waktu.

Tabel 2.3. Simbol *Sequence Diagram*

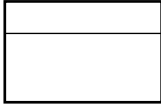






No	Simbol	Nama	Keterangan
1		Actor	Representasi entitas yang berada di luar sistem dan berinteraksi dengan sistem.
2		Lifeline	Menghubungkan objek selama <i>Sequence</i> (<i>message</i> dikirim, diterima, dan aktivitasnya).
3		<i>Entity Class</i>	Gambaran sistem sebagai landasan dalam menyusun basis data

No	Simbol	Nama	Keterangan
4		Boundary	Berupa <i>user interface</i> yang berinteraksi dengan sistem yang lain.
5		Control	Bertanggung jawab terhadap kelas – kelas terhadap objek yang berisi logika
6		Activation	Titik dimana sebuah objek mulai berpartisipasi didalam sebuah <i>Sequence</i>
7		Message	Komunikasi antar objek untuk menggambarkan aksi yang akan dilakukan.
8		Message Entry	Menggambarkan pesan/hubungan antar objek untuk menunjukkan urutan kejadian yang terjadi.
9		Self Message	Sebuah pesan yang mendefinisikan komunikasi tertentu antara Lifeline.

D. Class diagram

Merupakan gambaran struktur dan deskripsi dari *class*, *package*, dan objek yang saling berhubungan seperti diantaranya pewarisan, asosiasi dan lainnya.

Tabel 2.4. Simbol *Class Diagram*

No.	Simbol	Deskripsi
1		Class, Kelas pada struktur sistem
2	 Nama_interface	Antarmuka / <i>Interface</i> , Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3		Asosiasi / <i>Association</i> , Relasi antar kelas dengan makna umum, biasanya juga disertai dengan <i>multiplicity</i>
4		Asosiasi berarah / <i>Directed Association</i> , Relasi antarkelas dengan kelas yang digunakan oleh kelas yang lain.
5		Generalisasi, Relasi antarkelas dengan generalisasi-spesialisasi
6		Kebergantungan/ <i>Dependency</i> Relasi antar kelas dengan makna kebergantungan antarkelas
7		Agregasi / <i>Aggregation</i> , Relasi antarkelas semua-bagian (<i>whole-part</i>)

2.2.13. Pengujian Sistem

Pengujian dalam perangkat lunak dibagi menjadi dua jenis, yaitu *white box* dan *black box testing*. *White-box testing* adalah metode pengujian sistem yang melibatkan analisis terhadap kode *program* untuk menemukan kesalahan atau ketidaksesuaian.

Sementara *black-box testing* adalah strategi pengujian sistem yang berfokus pada fitur-fitur yang disediakan, seperti fungsi-fungsi dan tampilan pada sistem, serta memeriksa kesesuaian alur fungsi dengan sistem kerja yang diinginkan oleh perancang. Pengujian ini sangat penting dilakukan sebelum sistem digunakan untuk memastikan kelayakan sistem. Perangkat lunak memerlukan pengendalian, pengujian, dan pengelolaan untuk menjamin kualitasnya. Pada penelitian ini yang digunakan dalam pengujian sistem adalah metode pengujian *black-box testing*. [16]

2.2.14. Black-box Testing

Black-Box Testing adalah metode pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Dalam teknik pengujian ini, pengujian dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsionalitas dari perangkat lunak. Pendekatan *Black-Box Testing* memungkinkan pengembang perangkat lunak untuk membuat serangkaian kondisi input yang mencakup semua syarat fungsional dari *program* tersebut.[17]

2.2.15. White-box Testing

White-box testing adalah metode pengujian yang didasarkan pada kode *program*. Penguji dalam *white-box testing* memiliki

pemahaman tentang struktur dan penulisan kode serta mampu menyusun kasus uji dengan parameter yang tepat. Fokus utama *white-box testing* adalah pada aliran kontrol dan aliran data dalam program. Teknik-teknik yang umum digunakan dalam *white-box testing* antara lain pengujian berstruktur, pengujian *transparent box*, pengujian berbasis logika, dan pengujian berbasis kode. Pengujian ini menguji keputusan logis yang dapat menghasilkan kondisi benar atau salah di semua situasi. Pengujian *White-Box Testing* telah berkembang sebuah analisa sistem yang dapat menjelaskan secara spesifik tentang kode sumber pemrograman, algoritma sistem, dan jalur.[18]