

SISTEM INFORMASI E-ARSIP KELURAHAN CAKRANEGARA UTARA

(E-Archive Information Systems Of Northern Cakranegara Village)

Ida Ayu Vigi Meidhyana Putri^[1], I Gde Putu Wirarama^[1], I Made Suta^[2]

^[1]Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok, NTB, INDONESIA

^[2]Kelurahan Cakranegara Utara
Jl. Gora No. 25, Cakranegara, Lombok, NTB, INDONESIA

Email: idaayuvigi03@gmail.com, wirarama@unram.ac.id, imadesuta@gmail.com

Abstrak

Hingga saat ini pengelolaan arsip pada Kelurahan Cakranegara Utara masih dilakukan secara manual atau konvensional sehingga menyebabkan sering terjadinya kesalahan dalam penyimpanan dan pencarian data arsip. Padahal arsip merupakan dokumen penting yang disimpan dengan tujuan apabila dibutuhkan kembali akan mudah untuk menemukannya, maka sangat diharapkan proses pengelolaan arsip dilakukan dengan baik, cepat, dan mudah dengan memanfaatkan teknologi yang ada saat ini. Oleh karena itu sistem informasi e-arsip berbasis website menggunakan framework CodeIgniter dan Bootstrap ini dibuat dengan tujuan untuk mengubah cara pengelolaan dan pengarsipan data arsip yang semula hard copy menjadi soft copy. Metode yang digunakan dalam pembuatan sistem yaitu metode waterfall dengan urutan tahapan dari analisis kebutuhan, desain sistem, coding dan testing, penerapan program, serta pemeliharaan. Pengujian sistem dilakukan dengan menggunakan user acceptance testing yang di dalamnya terdapat pengujian black box dan kuisioner yang kemudian didapatkan hasil bahwa responden 100% setuju bahwa sistem informasi yang dibuat dapat membantu pegawai dan telah berjalan dengan baik.

Keywords: E-Arsip, CodeIgniter, Bootstrap, Arsip, Metode Waterfall, User Acceptance Testing.

1. PENDAHULUAN

1.1. Latar Belakang

Di era globalisasi ini perkembangan teknologi dari waktu ke waktu begitu pesat sehingga membuat pekerjaan manusia dipermudah dengan adanya teknologi saat ini, yang pada umumnya akan diselesaikan dengan cepat. Teknologi merupakan salah satu alat bantu yang sering digunakan pada aktivitas manusia, yang mana perannya begitu penting seperti halnya dalam mendapatkan suatu informasi dengan cepat maupun mendata suatu informasi [1]. Teknologi mampu mengolah suatu data dengan efisien dan terstruktur yang dapat dimanfaatkan oleh setiap instansi atau organisasi untuk meningkatkan produktivitas, waktu, dan biaya.

Kantor Kelurahan Cakranegara Utara adalah salah satu kantor pelayanan kependudukan yang bertanggung jawab untuk memberikan layanan kepada masyarakat. Selama ini, kantor kelurahan Cakranegara Utara telah melayani masyarakat di wilayah Cakranegara bagian utara untuk membantu masyarakat mendapatkan kebutuhan-kebutuhan seperti Surat Keterangan Catatan Kepolisian (SKCK), Surat Pengantar Kartu Tanda Penduduk (KTP), dan berbagai macam jenis surat lainnya. Surat memiliki fungsi yang sangat penting dan dapat digunakan sebagai dokumen karena memiliki bukti tanda tangan [2]. Tetapi pada saat ini, kantor Kelurahan Cakranegara Utara belum memiliki sistem atau basis data informasi yang digunakan untuk melakukan pengelolaan arsip data surat sehingga masih dilakukan secara manual, dimana hal ini akan memperlambat kinerja pegawai kantor Kelurahan Cakranegara Utara. Sebagai kantor pelayanan masyarakat, Kelurahan Cakranegara Utara berharap dapat memiliki sistem yang baik yang digunakan untuk memberikan pelayanan yang mudah, cepat, dan tepat.

Oleh karena itu penulis mencoba mencari solusinya agar kantor Kelurahan Cakranegara Utara dapat menyediakan dan melakukan pengelolaan data surat-surat dengan mudah, cepat, dan tepat dengan membuat sistem pengarsipan elektronik berbasis website dengan judul "Sistem Informasi E-Arsip Kelurahan Cakranegara Utara Berbasis Website". Dengan adanya pengarsipan data yang terkomputerisasi pada kantor Kelurahan Cakranegara Utara diharapkan dapat memberikan pelayanan dengan mudah, cepat, dan tepat, serta mempermudah pegawai kantor untuk melakukan pengelolaan data surat-surat.

2. TINJAUAN PUSTAKA

Sistem informasi adalah kumpulan komponen yang saling terhubung dalam mengumpulkan, memproses, menyimpan, menyediakan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi [3]. Pengertian lain mengatakan bahwa sebuah sistem informasi merupakan sekumpulan dari perangkat keras dan perangkat lunak. Selain itu data juga memegang peranan penting dalam sistem informasi. Data yang akan dimasukkan dalam sebuah sistem informasi yaitu dapat berupa formulir-formulir, prosedur-prosedur, dan bentuk data lainnya [4].

Database atau basis data merupakan sekumpulan data yang saling berintegrasi antara satu sama lain dan terorganisasi berdasarkan sebuah skema atau struktur tertentu dan tersimpan pada sebuah komputer. *Database* terdiri dari beberapa tabel atau lebih dari satu tabel yang saling terorganisir untuk menyimpan data dan terdiri dari baris dan kolom. Data yang tersimpan pada tabel diantaranya dapat ditampilkan, dimodifikasi dan dihapus. Setiap pemakai (*user*) diberi wewenang untuk dapat melakukan akses data tersebut [5].

World Wide Web atau yang biasa disebut *website* dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk mempublikasikan informasi berupa teks, gambar, dan program multimedia lainnya berupa animasi suara dan atau gabungan dari semua itu baik yang bersifat statis maupun dinamis yang saling terkait antara satu halaman dengan halaman yang sering disebut dengan *hyperlink* [6]. Pada *website* statis komunikasi berjalan secara satu arah dari *server* ke *client* sedangkan *website* dinamis komunikasinya berjalan secara dua arah dimana *user* bisa memasukkan data tertentu ke halaman *web* untuk kemudian diproses oleh server dan menghasilkan *output* yang diinginkan oleh *user* [7].

Unified Modelling Language (UML) merupakan alat perancangan sistem yang berorientasi pada objek yang memiliki tujuan utama untuk membantu tim pengembangan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau membuat program [8]. UML adalah salah satu standar Bahasa yang banyak digunakan di dunia industry untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [9].

ERD (*Entity Relationship Diagram*) merupakan sebuah diagram yang digunakan untuk merancang hubungan antar tabel-tabel dalam basis data [10]. ERD (*Entity Relation Diagram*) berguna untuk menggambarkan gambaran dari dunia nyata yang akan diterapkan pada suatu *database* sebuah sistem. ERD melihat objek nyata dapat sebagai sebuah entitas – entitas yang memiliki relasi antara entitas yang satu ataupun yang lain

PHP merupakan *software* digunakan sebagai Bahasa *script server-side* dalam pengembangan *web* yang disisipkan pada dokumen HTML [11]. PHP (*Hypertext Preprocessor*) disebut sebagai Bahasa pemrograman *server side* karena diproses pada komputer server saja. Dengan menggunakan PHP, pembuatan *web* dimudahkan dalam membuat konten di dalam sebuah halaman karena lebih efektif. Misalnya, untuk melakukan pengulangan secara lebih efektif, meng-*input database*, dan sebagainya [12].

MySQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan oleh para *programmer* aplikasi *web*. Sampai saat ini MySQL masih menjadi DBMS paling banyak digunakan di dunia. MySQL terkenal dengan kecepatan dalam pengaksesan datanya [13].

CodeIgniter merupakan sebuah *framework* PHP yang bersifat *open source* dan menggunakan metode MVC (*Model, View, Controller*) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis *web* tanpa harus membuatnya dari awal. Sedangkan *framework* adalah kumpulan instruksi-instruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan developer dalam memanggilnya tanpa harus menulis *syntax* program yang sama berulang-ulang serta dapat menghemat waktu [14].

XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem informasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache, HTTP *server*, MySQL *database*, dan penerjemah Bahasa yang ditulis dengan Bahasa pemrograman PHP dan Perl [13].

Bootstrap merupakan *framework* untuk membangun desain *web* secara responsive. Artinya, tampilan *web* yang dibuat oleh *bootstrap* akan menyesuaikan ukuran *layer* dan *browser* yang kita gunakan baik di desktop, *tablet* ataupun *mobile device*. Dengan *bootstrap* kita juga bisa membangun *web* dinamis ataupun statis [14].

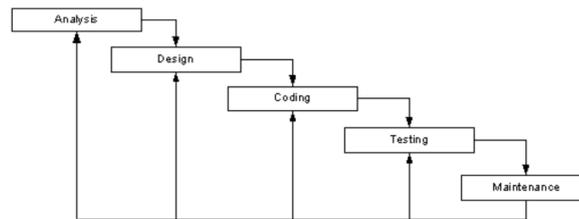
3. METODE PENGABDIAN MASYARAKAT

3.1. Metode Perancangan Sistem Untuk Masyarakat Pada Kantor Kelurahan Cakranegara Utara

Metode yang digunakan dalam perancangan sistem informasi pengarsipan surat pada Kelurahan Cakranegara Utara ini adalah metode *waterfall*. Alasan penggunaan metode *waterfall* dalam pembuatan sistem informasi e-arsip ini dikarenakan sistem yang dibuat tidak terlalu besar dan lingkup pengembangannya tidak begitu luas sehingga

akan lebih memudahkan jika menggunakan metode *waterfall* yang pendekatannya dilakukan secara sistematis dan berurutan.

Secara garis besar, metode pengembangan perangkat lunak dengan metode *waterfall* dibagi menjadi lima tahapan utama, yaitu analisa kebutuhan masyarakat yang bertujuan untuk menganalisis kebutuhan yang dibutuhkan oleh masyarakat keluarahan cakranegara utara, baik berupa dokumen maupun sumber lain yang dapat membantu dalam menentukan solusi permasalahan yang ada, perancangan sistem untuk masyarakat (*design*) untuk menyediakan sistem informasi yang terorganisir dengan baik yang menggunakan beberapa diagram serta *mockup* sistem, pembuatan kode program (*coding*) sesuai dengan analisis kebutuhan untuk membuat sistem berbasis *website* dengan Bahasa pemrograman PHP, pengujian (*testing*), serta pendukung (*support*) dan pemeliharaan (*maintanance*) [16]. Berikut gambaran dari tahapan-tahapan pada metode *waterfall*:

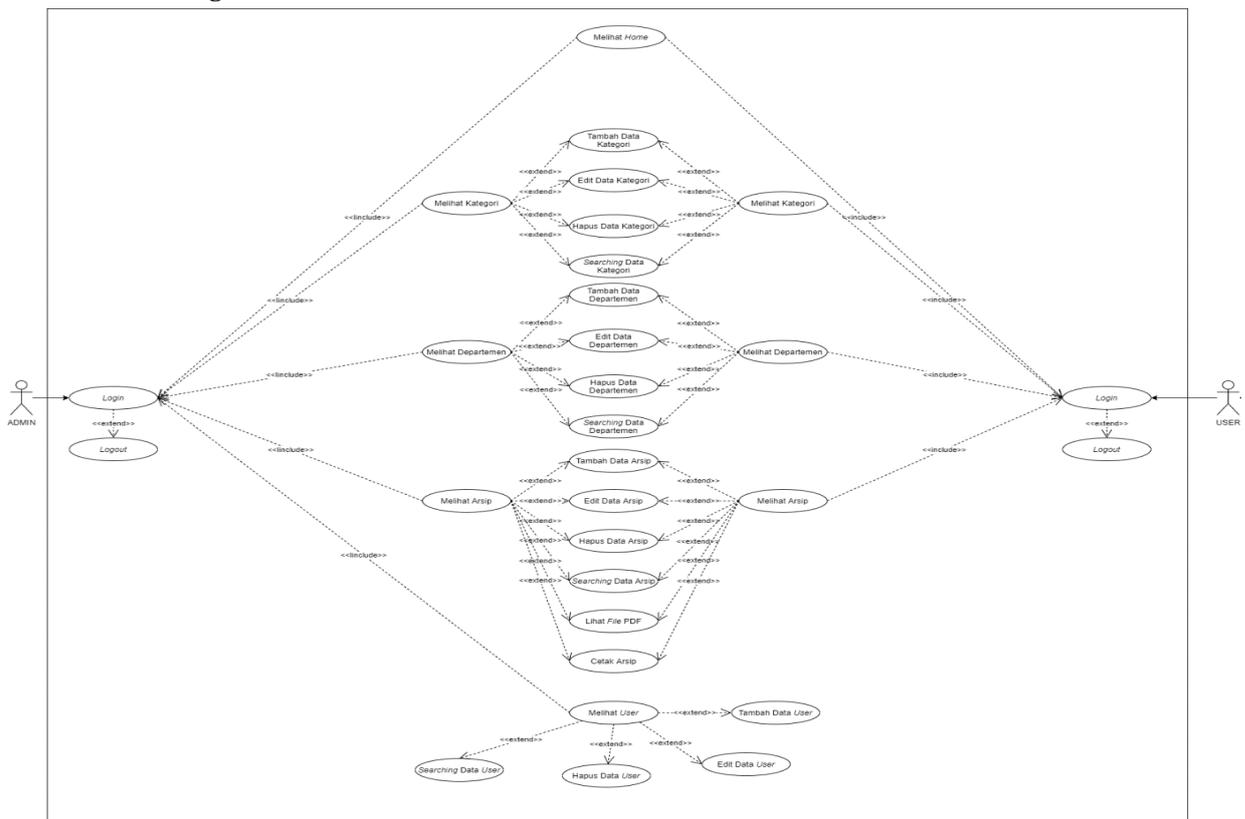


Gambar 1. Metode *waterfall*

3.2. Desain Sistem

Sistem Informasi E-Arsip Kelurahan Cakranegara Utara memiliki beberapa diagram seperti *Use Case Diagram*, *Class Diagram*, *Sequence Diagram*, *Activity Diagram*, dan *Entity Relationship Diagram*.

3.2.1. Use Case Diagram



Gambar 2. Use case diagram e-arsip

Pada Gambar 2 dapat dilihat bahwa dalam *use case diagram* tersebut terdapat dua aktor yaitu administrator dan *user*. Adapun tugas administrator dan *user* dijabarkan sebagai berikut:

1. Administrator

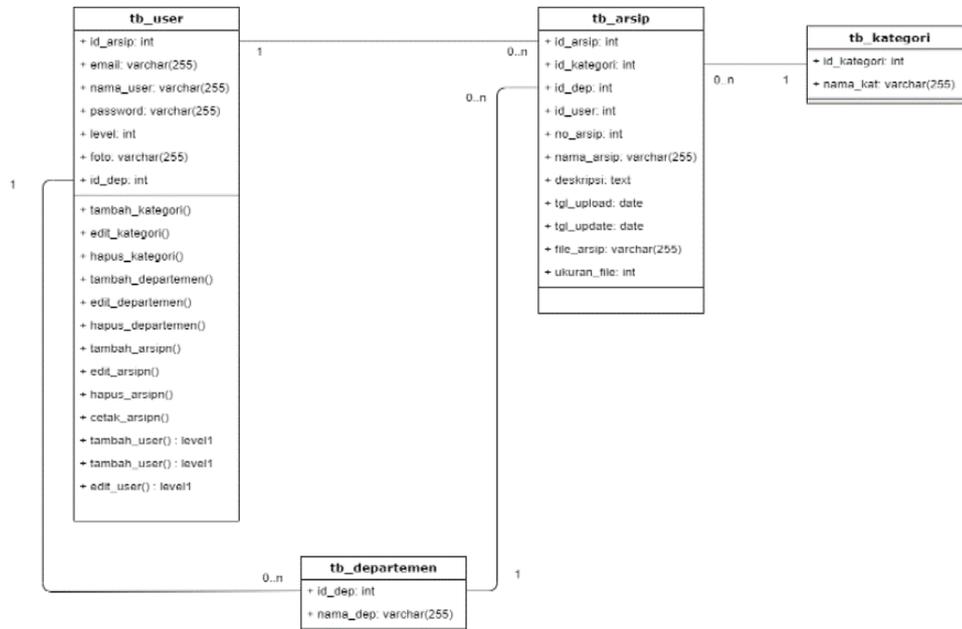
Dapat melihat *home*, melihat dan mengedit menu kategori, melihat dan mengedit menu departemen (melakukan tambah data, edit data, hapus data, dan *searching data*), melihat dan mengedit menu, serta melihat dan mengedit menu *user*.

2. *User*

Dapat melihat *home*, melihat dan mengedit menu kategori, melihat dan mengedit menu departemen, serta melihat dan mengedit menu.

3.2.2. **Class Diagram**

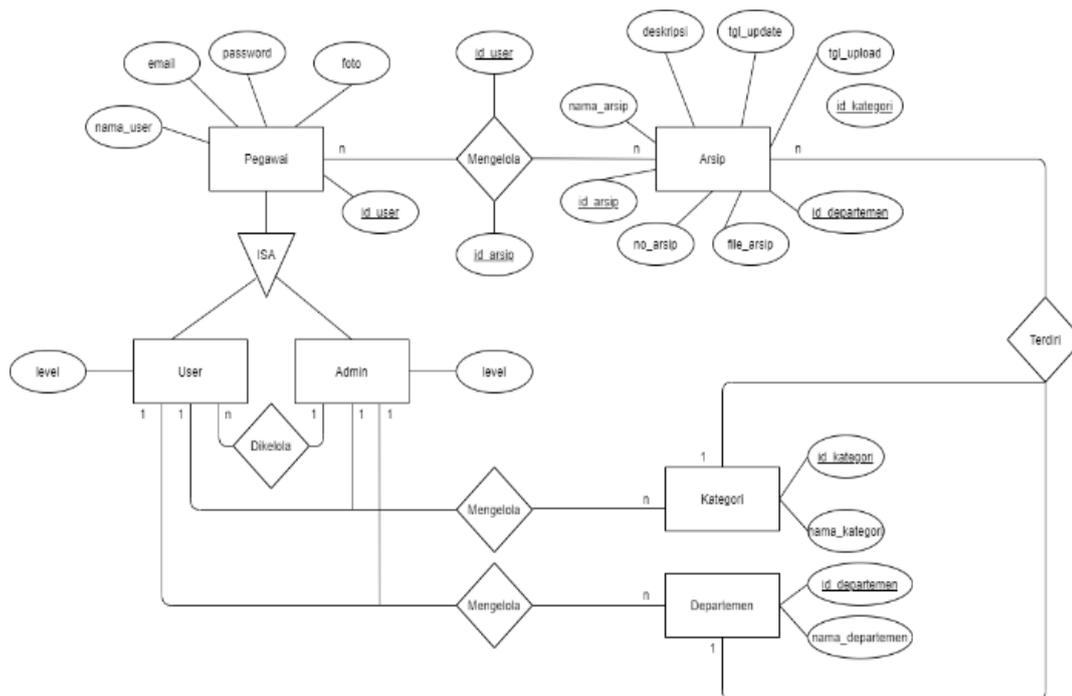
Class diagram dari sistem informasi e-arsip pada Kelurahan Cakranegara Utara adalah sebagai berikut:



Gambar 3. *Class diagram* Sistem Informasi E-Arsip Kelurahan Cakranegara Utara

3.2.3. **Entity Relationship Diagram**

Entity relationship diagram Sistem Informasi E-arsip Kelurahan Cakranegara Utara adalah sebagai berikut:



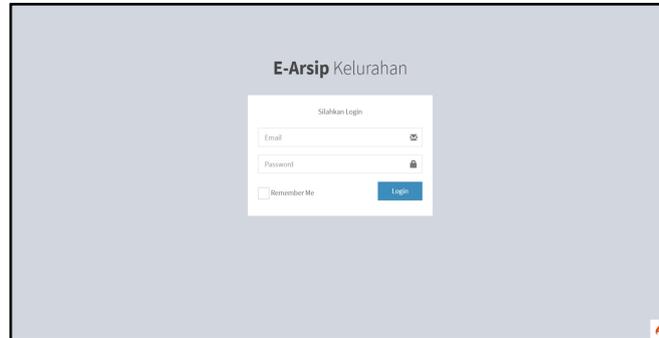
Gambar 4. *Entity relationship diagram* Sistem Informasi E-Arsip Kelurahan Cakranegara Utara

4. HASIL DAN PEMBAHASAN

4.1. Implementasi Sistem

Implementasi sistem merupakan tahap penerapan serta pengujian untuk sistem berdasarkan hasil analisa dan perancangan yang dilakukan. Implementasi sistem telah disesuaikan dengan perancangan pada sub-sub bab sebelumnya. Berikut merupakan implementasi sistem dari Sistem Informasi E-arsip Kelurahan Cakranegara Utara:

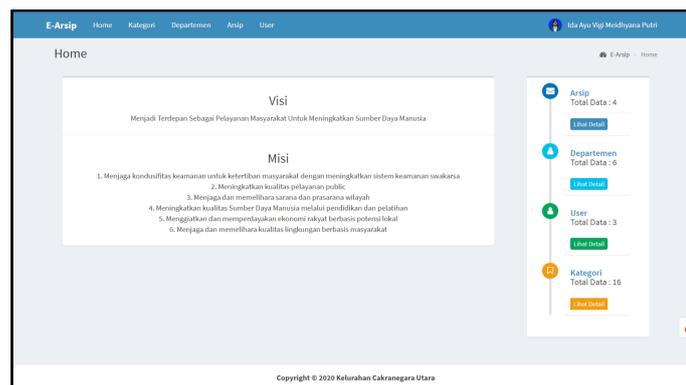
1. Halaman *Login*



Gambar 5. Implementasi tampilan halaman *login*

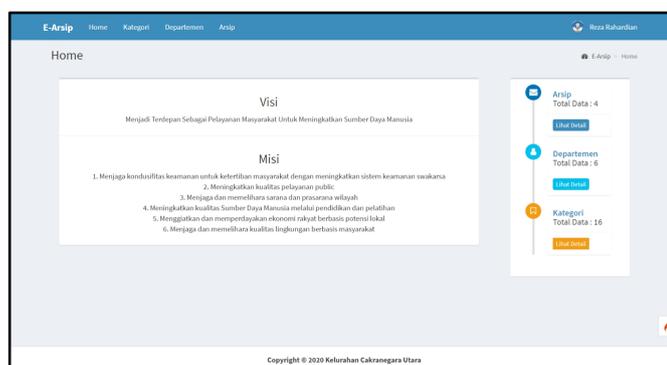
Pada Gambar 5 merupakan implementasi tampilan halaman *login* ketika pegawai pertama kali membuka *website e-arsip*. Pada halaman ini pegawai terlebih dahulu harus memasukkan *email* dan *password* untuk dapat menggunakan *website*.

2. Halaman *Home*



Gambar 6. Implementasi tampilan halaman *home* admin

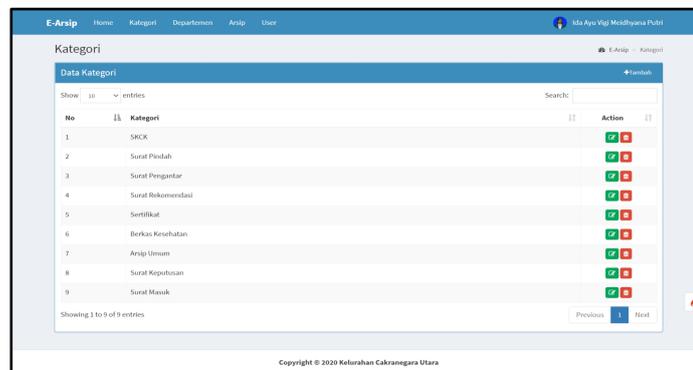
Pada Gambar 6 merupakan implementasi tampilan *home* ketika pegawai sebagai admin berhasil melakukan *login* pada *website*. Pada halaman ini terdapat informasi “Visi dan Misi” dari Kelurahan Cakranegara Utara, dan terdapat informasi singkat mengenai total jumlah data dari tiap menu yang ada, seperti total data: “Arsip”, “Departemen”, “Kategori”, dan “User”. Kemudian pada navbar terdapat menu “Kategori”, “Departemen”, “Arsip”, “User” dan informasi mengenai akun pegawai yang digunakan untuk *login* seperti menampilkan foto dan nama pegawai tersebut. Pada bagian *footer* terdapat keterangan “Copyright 2020 Kelurahan Cakranegara Utara”.



Gambar 7. Implementasi tampilan halaman *home* user

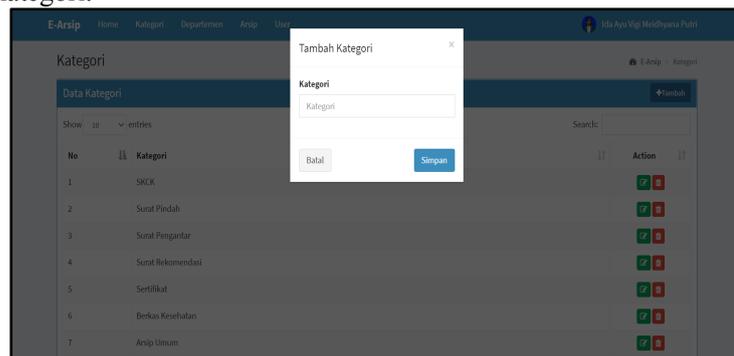
Pada Gambar 7 merupakan implementasi tampilan *home* ketika pegawai sebagai *user* berhasil melakukan *login* pada *website*. Pada halaman ini sama seperti tampilan *home* pada *home* admin, perbedaannya adalah pada tampilan halaman ini menu “*User*” tidak ada karena menu “*User*” hanya dapat diakses oleh admin saja.

3. Halaman Kategori



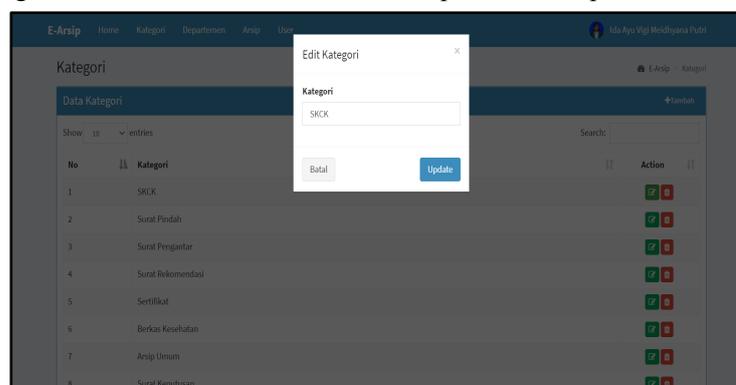
Gambar 8. Implementasi tampilan kategori

Pada Gambar 8 merupakan implementasi tampilan menu kategori. Pada halaman ini terdapat tabel Data Kategori dengan kolom “No”, “Kategori”, dan “Action”. Pada pojok kanan atas tabel terdapat tombol yang dapat digunakan untuk menambahkan data kategori. Kemudian terdapat kolom pencarian yang digunakan oleh pegawai untuk melakukan pencarian sesuai data yang diinginkan. Pada kolom “Action” terdapat tombol edit berwarna hijau yang digunakan untuk melakukan perubahan data kategori, dan tombol hapus berwarna merah yang digunakan untuk menghapus data kategori.



Gambar 9. Implementasi tampilan tambah kategori

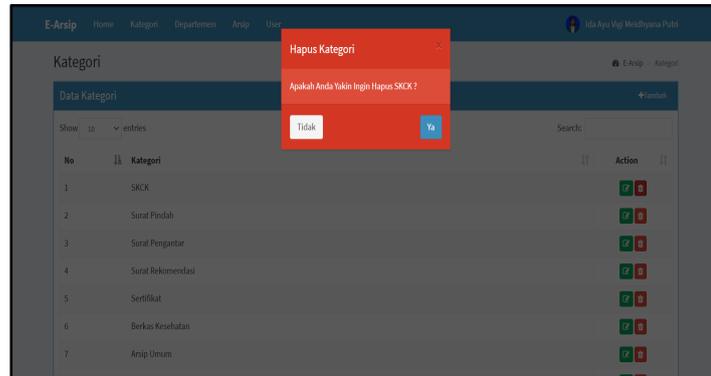
Pada Gambar 9 merupakan tampilan tambah data kategori. Ketika *user* mengklik tombol tambah pada tabel Data Kategori maka akan muncul *window* untuk menambahkan data kategori. Pada tampilan ini terdapat tombol simpan yang digunakan ketika *user* ingin melakukan penambahan data sesuai yang dimasukkan pada kolom *input* dan tombol batal yang digunakan ketika *user* batal melakukan penambahan pada data kategori.



Gambar 10. Implementasi tampilan edit kategori

Pada Gambar 10 merupakan implementasi tampilan edit data kategori. Ketika *user* mengklik tombol edit berwarna hijau pada kolom *action* maka akan muncul *window* untuk melakukan edit data kategori. Pada tampilan

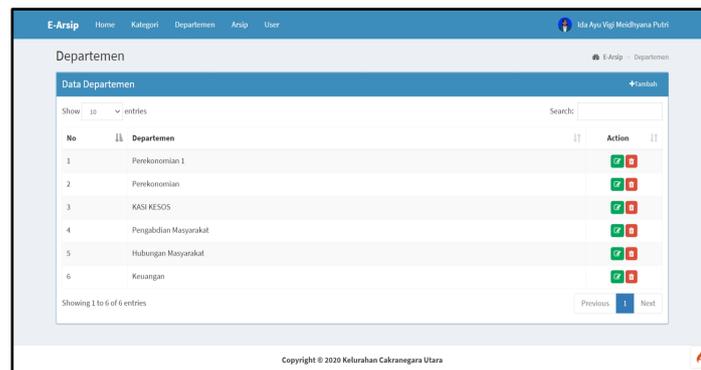
ini terdapat tombol *update* yang digunakan ketika *user* ingin melakukan perubahan data sesuai yang dimasukkan pada kolom *input* dan tombol batal yang digunakan ketika *user* batal melakukan perubahan pada data kategori.



Gambar 11. Implementasi tampilan hapus kategori

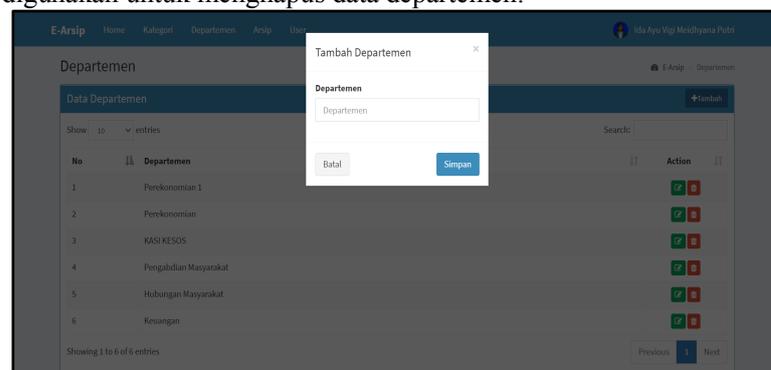
Pada Gambar 11 merupakan implementasi tampilan hapus data kategori. Ketika *user* mengklik tombol hapus berwarna merah pada kolom *action* maka akan muncul *window* untuk melakukan hapus data kategori. Pada tampilan ini terdapat tombol “Ya” yang digunakan ketika *user* yakin akan menghapus data kategori dan tombol “Tidak” yang digunakan ketika *user* batal melakukan penghapusan data kategori.

4. Halaman Departemen



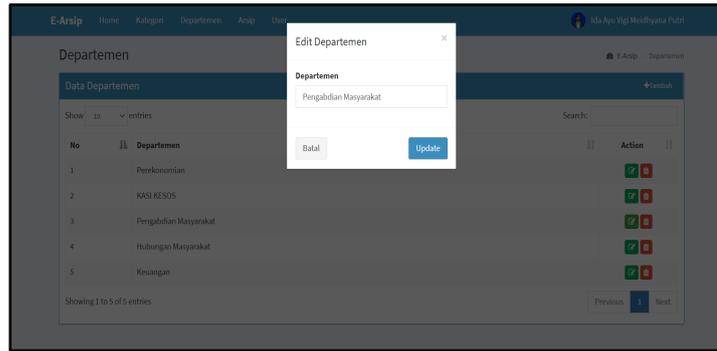
Gambar 12. Implementasi tampilan menu departemen

Pada Gambar 12 merupakan implementasi tampilan menu kategori. Pada halaman ini terdapat tabel Data Departemen dengan kolom “No”, “Departemen”, dan “Action”. Pada pojok kanan atas tabel terdapat tombol yang dapat digunakan untuk menambahkan data departemen. Kemudian terdapat kolom pencarian yang digunakan oleh pegawai untuk melakukan pencarian sesuai data untuk melakukan perubahan pada departemen, dan tombol hapus berwarna merah yang digunakan untuk menghapus data departemen.



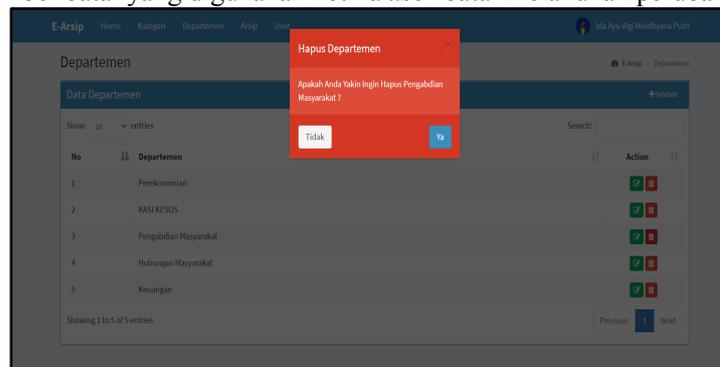
Gambar 13. Implementasi tampilan tambah departemen

Pada Gambar 13 merupakan implementasi tampilan tambah data departemen. Ketika *user* mengklik tombol tambah pada tabel Data Departemen maka akan muncul *window* untuk menambahkan data departemen. Pada tampilan ini terdapat tombol simpan yang digunakan ketika *user* ingin melakukan penambahan data sesuai yang dimasukkan pada kolom *input* dan tombol batal yang digunakan ketika *user* batal melakukan penambahan data departemen.



Gambar 14. Implementasi tampilan edit departemen

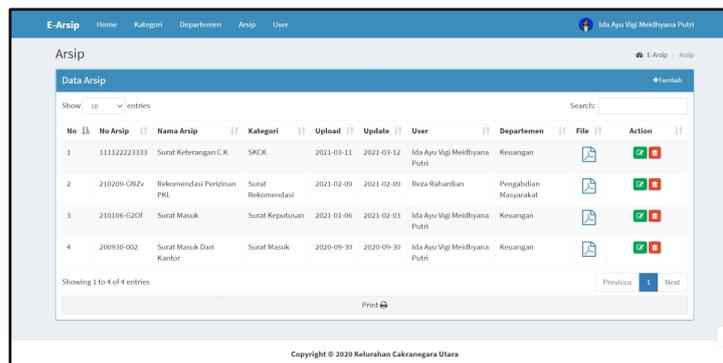
Pada Gambar 14 merupakan implementasi tampilan edit data departemen. Ketika *user* mengklik tombol edit berwarna hijau pada kolom *action* maka akan muncul *window* untuk melakukan edit data departemen. Pada tampilan ini terdapat tombol *update* yang digunakan ketika *user* ingin melakukan perubahan data sesuai yang dimasukkan pada kolom *input* dan tombol *batal* yang digunakan ketika *user* batal melakukan perubahan pada data departemen.



Gambar 15. Implementasi tampilan hapus departemen

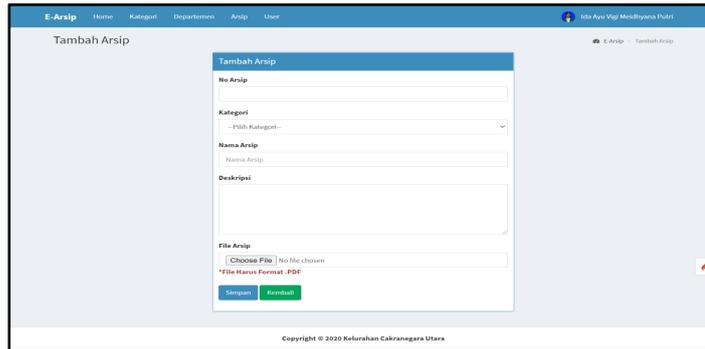
Pada Gambar 15 merupakan implementasi tampilan hapus data departemen. Ketika *user* mengklik tombol hapus berwarna merah pada kolom *action* maka akan muncul *window* untuk melakukan hapus data departemen. Pada tampilan ini terdapat tombol “Ya” yang digunakan ketika *user* yakin akan menghapus data departemen dan tombol “Tidak” yang digunakan ketika *user* batal melakukan penghapusan pada data departemen.

5. Halaman Arsip



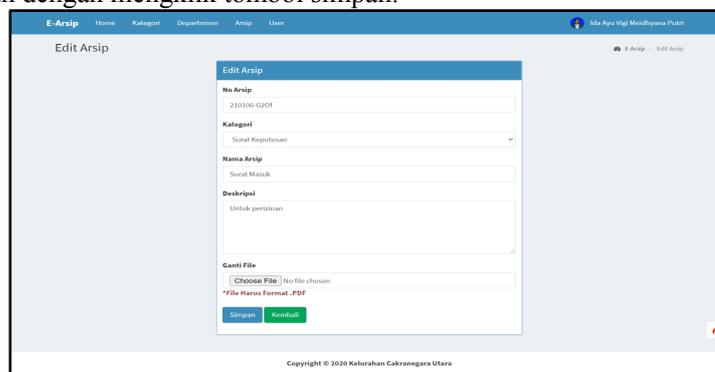
Gambar 16. Implementasi tampilan menu arsip

Pada Gambar 16 merupakan implementasi tampilan menu arsip. Pada halaman ini terdapat tabel Data Arsip dengan kolom “No”, “No Arsip”, “Kategori”, “Upload”, “Update”, “User”, “Departemen”, “File”, dan “Action”. Pada pojok kanan atas tabel terdapat tombol yang dapat digunakan untuk menambahkan data arsip. Kemudian terdapat kolom pencarian yang digunakan oleh pegawai untuk melakukan pencarian sesuai data yang diinginkan. Pada kolom “Action” terdapat tombol edit berwarna hijau yang digunakan untuk melakukan perubahan pada arsip dan tombol hapus berwarna merah yang digunakan untuk menghapus data arsip.



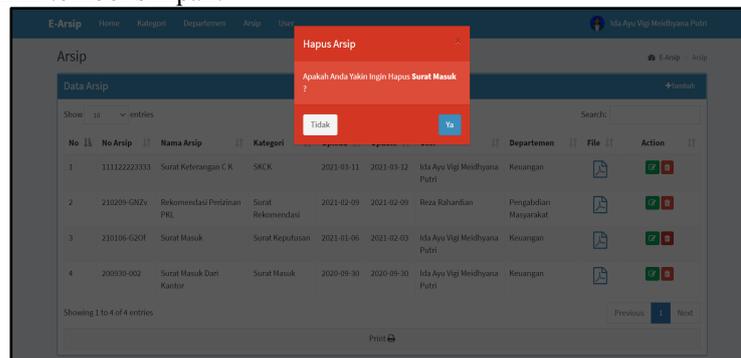
Gambar 17. Implementasi tampilan tambah arsip

Pada Gambar 17 merupakan implementasi tampilan dari halaman tambah data arsip yang akan diisi oleh pegawai ketika mengklik tombol tambah pada tabel Data Arsip di halaman menu Arsip. Pada halaman ini pegawai diminta untuk memasukkan “No Arsip”, “Kategori”, “Nama Arsip”, “Deskripsi”, dan “File Arsip”. Data arsip yang sudah terisi bisa disimpan dengan mengklik tombol simpan.



Gambar 18. Implementasi tampilan edit arsip

Pada Gambar 18 merupakan implementasi tampilan dari halaman edit data arsip yang akan diubah oleh pegawai ketika mengklik tombol edit pada kolom *action* di tabel Data Arsip. Data arsip yang sudah diedit bisa disimpan dengan mengklik tombol simpan.



Gambar 19. Implementasi tampilan hapus arsip

Pada Gambar 19 merupakan implementasi tampilan hapus data arsip. Ketika *user* mengklik tombol hapus berwarna merah pada kolom *action* maka akan muncul *window* untuk melakukan hapus data arsip. Pada tampilan ini terdapat tombol “Ya” yang digunakan ketika *user* yakin akan menghapus data arsip dan tombol “Tidak” yang digunakan ketika *user* batal melakukan penghapusan pada data arsip.

4.2. Pengujian Sistem

Setelah dilakukan implementasi, maka kemudian dilakukan pengujian sistem yang telah dirancang dan dibuat dengan menggunakan pengujian *black box*. Pengujian *black box* merupakan metode pengujian yang dilakukan dengan menguji fungsi-fungsi dari sistem yang dibuat untuk mengetahui fungsi-fungsi yang dibuat sudah terlaksana dengan baik atau tidak [15]. Pengujian ini lebih mengutamakan pengujian pada luar sistem (*interface*) agar mudah digunakan oleh *client*. Berdasarkan pengujian *black box* yang dilakukan dalam menguji seluruh fitur dari sistem

informasi e-arsip dengan memasukkan berbagai data yang diperlukan, maka dapat diambil kesimpulan bahwa fitur-fitur yang terdapat dalam sistem informasi e-arsip pada Kelurahan Cakranegara Utara telah berjalan dengan baik.

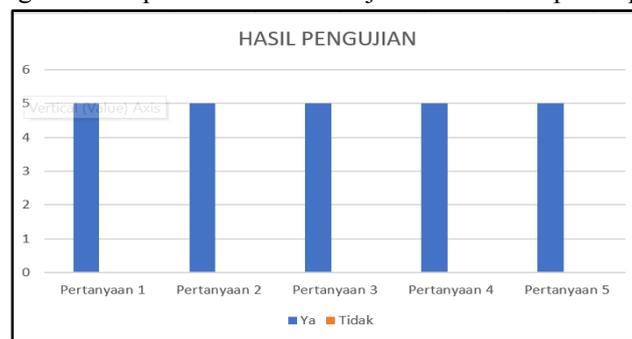
Selanjutnya dilakukan pengujian sistem dengan pengisian kuisioner untuk menanyakan beberapa responden terkait sistem yang dibuat. Parameter yang digunakan dalam metode ini yaitu:

1. Apakah sistem informasi e-arsip Kelurahan Cakranegara Utara yang dibuat sudah memenuhi kebutuhan?
2. Apakah sistem yang dibuat dapat memudahkan pegawai dalam mengelola data arsip?
3. Apakah sistem yang dibuat mudah digunakan?
4. Apakah sistem sudah bekerja sesuai dengan kegunaannya?
5. Apakah sistem informasi e-arsip yang dibuatkan memiliki tampilan yang sesuai kebutuhan?

Responden akan diberikan pilihan jawaban dari pertanyaan sebelumnya diantaranya:

1. Ya
2. Tidak

Hasil keseluruhan jawaban dari lima responden dilakukan rekapitulasi berdasarkan setiap pertanyaan yang diajukan. Berikut merupakan grafik dari persentase rata-rata jawaban dari empat responden:



Gambar 20. Persentase hasil pengujian

Gambar 20 merupakan hasil pengujian kuisioner yang dilakukan dengan memberikan pertanyaan kepada lima responden terkait. Hasil tersebut menunjukkan bahwa ke-lima responden menjawab semua pertanyaan dengan “Ya” atau setuju. Sehingga, dapat disimpulkan bahwa dari hasil pengujian tersebut kepuasan reponden terhadap sistem informasi e-arsip Kelurahan Cakranegara Utara yang dirancang dan dibuat 100% setuju sesuai dengan kebutuhan, fungsi, kinerja dan informasi yang diperlukan.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari kegiatan yang telah dilakukan, didapatkan kesimpulan yaitu :

1. Sistem informasi e-arsip yang dibuat dapat digunakan untuk melakukan pendataan terhadap data-data arsip serta dapat mencetak seluruh data arsip yang telah disimpan di dalam *website* dengan memiliki 2 pengguna (administrator dan *user*).
2. Hasil pengujian yang dilakukan dengan mengetahui kepuasan pengguna terhadap sistem yang dibuat yaitu 100% sesuai dengan kebutuhan, fungsi, kinerja, dan informasi yang dihasilkan.

5.2. Saran

Berdasarkan sistem informasi e-arsip yang dibuat, dapat diberikan saran yaitu dengan melakukan perbaikan tampilan serta menambahkan beberapa fitur baru seperti dapat menerima surat elektronik melalui *website* tersebut untuk dapat menunjang kinerja dari sistem dan dapat memudahkan administrator dan user dalam mengelola data-data arsip.

DAFTAR PUSTAKA

- [1] B.K. Williams, S. C. Sawyer, *Using Information Technology (Terjemahan Indonesia)*, Yogyakarta: Andi Offset, 2007.
- [2] Darmono, “Sistem Informasi Pelayanan Arsip Pada Dinas Perpustakaan dan Kearsipan Kabupaten Seruyan Menggunakan Borland Delphi 7.0 dan Interbase,” *Jurnal Penelitian Dosen Fikom (UNDA)*, Vol. 10, No. 2, November 2019, ISSN : 2088-3595.
- [3] A. Kadir, *Pengenalan Sistem Informasi*, Yogyakarta: CV.Andi Offset, 2014.

- [4] J. Hartono, *Analisis dan Desain Sistem Informasi*, Yogyakarta: Andi Offset, 2005.
- [5] A. Kristanto, *Perancangan SI dan Aplikasinya*, Yogyakarta: Penerbit Gava Media, 2003.
- [6] M. R. Arief, *Pemrograman Basis Data Menggunakan Transact-SQL dengan Microsoft SQL Server 2000*, Yogyakarta: Andi, 2006.
- [7] F. Marisa, *Web Programming untuk Membangun Portal*, Yogyakarta: CV Budi Utama, 2017.
- [8] H. Hartono, *Pengertian Website dan Fungsinya*, Ilmu Teknologi Informasi, pp.2-7, 2014.
- [9] F. T. Yuniko dan F. K. Putra, "Penerapan Teknologi Informasi *Web Programming* untuk Meningkatkan Pelayanan Publik Dalam Bidang Kebijakan Administrasi Kekematianan," *JOISIE (Journal Of Information Systems And Informatics Engineering)*, 1(1), 13, 2019.
- [10] Havaluddin, "Memahami Penggunaan UML (*Unified Modelling Language*)," *Jurnal Informatika Mulawarman*, vol. 6, No. 1, Februari 2011, pp. 1-15.
- [11] R. A. Sukanto dan M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Informatika, 2014.
- [12] Prijambodo, *Out of The Boc: Koperasi Tantangan Perubahan Kini dan Masa Depan*, Yogyakarta: Phoenix Publisher, 2018.
- [13] Sutarman, *Membangun Aplikasi Web dengan PHP dan MySQL*, Yogyakarta: Penerbit Graha Ilmu, 2003.
- [14] M. R. Arief, *Pemrograman Web Dinamis Menggunakan PHP & MySQL*, Yogyakarta: Andi, 2011.