

**RANCANGAN SIPDES DENGAN MENGGUNAKAN
METODE *SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC)***

(Studi Kasus : Desa Sumber Agung, Lampung Selatan)

SKRIPSI

Oleh :

NADIA ARIYATI

20421039



**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BANDAR LAMPUNG**

2024

**RANCANGAN SIPDES DENGAN MENGGUAKAN
*METODE AGILE DEVELOPMENT***

(Studi Kasus : Desa Sumber Agung, Lampung Selatan)

SKRIPSI

(Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana S-1

Pada Program Studi Informatika Fakultas Ilmu Komputer
Universitas Bandar Lampung)

Oleh :

NADIA ARIYATI

20421039



**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BANDAR LAMPUNG**

2024

HALAMAN PENGESAHAN

Judul Penelitian : **"Rancangan SIPDes Dengan Menggunakan Metode
Agile Development (Studi Kasus : Desa Sumber
Agung,, Lampung Selatan)"**

Nama Mahasiswa : Nadia Ariyati

No. Pokok Mahasiswa : 20421039

Program Studi : Informatika

Fakultas : Ilmu Komputer

Bandarlampung, Februari 2024

Dosen Pembimbing

Ketua Program Studi Informatika

Yuthsi Aprilinda, Skom. M.Kom

Dr. Maria Shusanti F, S.Kom., M.Kom

ABSTRAK

Desa Sumber Agung merupakan unit pemerintah yang memberikan pelayanan kepada masyarakat. Pelayanan yang diberikan pemerintah desa yaitu layanan permohonan pembuatan surat dan Informasi mengenai Desa. Selama ini proses pelayanan dilakukan masih secara manual, warga desa harus datang ke Kantor Desa untuk mengajukan permohonan surat serta menyebarkan informasi terkait desa kurang efektif. Proses pelayanan yang berjalan saat ini memerlukan waktu yang lama serta kurang efektif dan efisien. Pendataan yang dilakukan masih menggunakan cara manual menyebabkan banyak resiko kesalahan penulisan, memerlukan lebih banyak ruang penyimpanan dan data tidak terorganisir.

Tujuan dari penelitian ini adalah untuk membangun sistem informasi pelayanan publik desa di Desa Sumber Agung untuk mempermudah dalam proses pelayanan. Metode yang digunakan dalam mengembangkan sistem ini adalah metode *SDLC*, tahapannya meliputi Analisis, Desain Sistem, Implementasi, dan pengujian sistem. Sistem dibangun dengan menggunakan *framework Quasar*. Pengujian sistem menggunakan metode *black box* dan *white box testing* yang menyatakan bahwa sistem ini dapat berjalan sesuai dengan fungsionalitasnya dan metode perhitungan Skala Likert dengan skor 82,76% sehingga dapat disimpulkan bahwa sistem ini layak untuk digunakan. Sistem Informasi Layanan publik ini, memberikan banyak manfaat dan kemudahan kepada petugas dan warga dalam pelayanan publik di kantor desa.

Kata kunci : Layanan Publik, Sistem Informasi, *SDLC*

ABSTRACT

Sumber Agung Village is a government unit that provides services to the community. The services provided by the village government are letter request services and information about the village. So far, the service process is still carried out manually, village residents have to come to the Village Office to submit letter requests and disseminate information related to the village which is less effective. The current service process takes a long time and is less effective and efficient. Data collection that is still carried out using manual methods causes a lot of risk of writing errors, requires more storage space and the data is disorganized.

The aim of this research is to build a village public service information system in Sumber Agung Village to simplify the service process. The method used in developing this system is the SDLC method, the stages include Analysis, System Design, Implementation and system testing. The system was built using the Quasar framework. System testing uses black box and white box testing methods which state that this system can run according to its functionality and the Likert Scale calculation method with a score of 82.76% so it can be concluded that this system is suitable for use. This public service information system provides many benefits and conveniences to officers and residents in public services at the village office.

Keywords: Public Services, Information Systems, SDLC

RIWAYAT HIDUP

Penulis memiliki nama lengkap Nadia Ariyati yang lahir di Provinsi Lampung tepatnya di Desa Tanjung Ratu, Kecamatan Katibung, Kabupaten Lampung Selatan, lahir pada tanggal 15 Mei 2001 merupakan anak perempuan Ke-9 dari sepuluh bersaudara, penulis memiliki seorang Ayah Kandung yang bernama Bapak Muhamad Arifi dan Ibu Kandung Ibu Marti.

Pendidikan yang telah ditempuh Penulis adalah:

1. Madrasah Ibtidaiyah Mathla'ul Anwar, Lampung Selatan
2. Madrasah Tsanawiyah Mathla'ul Anwar, Lampung Selatan
3. Sekolah Menengah Kejuruan Negeri 1 Katibung, Lampung Selatan

Pada tahun 2020 penulis melanjutkan Pendidikan ke jenjang S1 Pada Fakultas Ilmu Komputer Program Studi Informatika di Universitas Bandar Lampung.

MOTTO

“Lemah Bukan Berarti Tidak Mampu”

(NADIA ARIYATI)

PERSEMBAHAN

Alhamdulillah, segala puji syukur bagi Allah SWT atas limpahan rahmat, taufiq, hidayah dan inayah-Nya kepada penulis beserta keluarga dan saudara lainnya sehingga penulis dapat menyelesaikan Tugas Akhir/Skripsi ini tepat pada waktunya. Tiada lembar paling indah dalam Laporan Tugas Akhir/Skripsi ini kecuali lembar persembahan untuk orang-orang tercinta yaitu :

1. Kepada Allah SWT, dengan rasa syukur atas kehendak-Mu yang telah memberikan petunjuk, kemudahan dan kesabaran dalam segala hal, sehingga penulis dapat menyelesaikan Laporan Tugas Akhir/Skripsi ini.
2. Cinta pertama dan panutanku, Ayahanda Muhamad Arifi, Beliau memang tidak sempat melanjutkan bangku pendidikannya karena adanya suatu halangan, namun beliau mampu mendidik penulis, memberikan semangat dan motivasi tiada henti hingga penulis dapat menyelesaikan pendidikannya sampai sarjana strata-1.
3. Pintu surgaku, Ibunda Marti, terima kasih sebesar-besarnya penulis berikan kepada beliau atas segala bentuk bantuan, semangat dan doa yang diberikan selama ini. Terima kasih atas nasihat yang selalu diberikan meski terkadang pikiran kita tidak sejalan, terima kasih atas kesabaran dan kebesaran hati menghadapi penulis yang keras kepala. Ibu menjadi penguat dan pengingat paling hebat. Terima kasih, sudah menjadi tempatku pulang, bu.
4. Kakak adik ku, terima kasih sudah ikut serta dalam proses penulis menempuh Pendidikan selama ini, terima kasih atas semangat, doa, dan cinta yang selalu diberikan kepada penulis.
5. Untuk jodoh penulis kelak kamu adalah salah satu alasan penulis untuk menyelesaikan skripsi ini, meskipun saat ini penulis tidak tahu keberadaanmu entah di bumi bagian mana dan menggenggam tangan siapa. Tapi penulis selalu ingat petuah seperti kata Bj Habibie "Kalau memang dia dilahirkan untuk saya, kamu jungkir balik pun saya yang dapat"
6. Dan yang terakhir, terima kasih kepada diri sendiri. Hebat bisa tetap berdiri tegap menghadapi segala tantangan hidup walau kadang jenuh dan ingin berhenti. Kamu keren dan hebat, nadia

KATA PENGANTAR

Puji syukur beserta rahmatnya kepada Tuhan Yang Maha ESA karena atas berkah dan rezeki-nya untuk peneliti melaksanakan Penulisan Tugas Akhir ini yang berjudul Rancangan SIPDes Dengan Menggunakan Metode *SDLC Software Developmen life Cycle* (Studi Kasus: Desa Sumber Agung, Lam-Sel), Untuk Meningkatkan Efisiensi pelayanan desa pada masyarakat. Tugas Akhir ini bertujuan untuk menyelesaikan mata kuliah wajib kuliah sebelum melaksanakan tugas akhir dan skripsi. Tugas Akhir ini tidak mungkin dapat dilaksanakan terlepas dari dukungan seluruh pihak maupun bimbingan, fasilitas, dukungan dan masih banyak lagi akhirnya peneliti dapat melakukan penelitian dengan baik dan benar. Dengan seluruh rasa syukur penelitian mengucapkan terima kasih kepada:

1. Ibu Dra. Hj. Sri Hayati Barusman, Selaku dewan pembina Yayasan Administrasi Lampung .
2. Bapak Dr. andala rama putra barusman, SE., MA.Ec selaku Ketua Yayasan Administrasi Lampung
3. Bapak prof. Dr. Ir. M. Yusuf Sulfarano Barusman, MBA, selaku Rektor Universitas bandar Lampung
4. Bapak Marzuki, S.Kom., M.Kom Selaku Dekan Fakultas Ilmu Komputer , Universitas Bandar Lampung
5. Ibu. Dr. Maria Shusanti Febrianti, S.Kom., M.Kom selaku ketua program Studi Informatika.
6. Ibu Yuthsi Aprilinda, S.Kom. M.Kom. Selaku dosen pembimbing yang telah memberikan arahan dan bimbingan dalam proses menyelesaikan laporan penulisan ilmiah ini.
7. Bapak Ismail, S.Pd.I Selaku Kepala Desa Sumber Agung dan para staff kantor desa Sumber Agung yang telah membantu dalam penyelesaian penelitian Tugas Akhir.
8. Seluruh Dosen dan Staff pengajar fakultas ilmu Komputer Universitas Bandar Lampung.

9. Kedua Orang Tua, kakak, adik yang ada di rumah karena telah memberikan semangat, dukungan, serta doa.
10. Teman-teman Angkatan 2020 Fakultas Ilmu Komputer Universitas Bandar Lampung.
11. Serta seluruh pihak yang tidak bisa saya sebutkan satu persatu, yang telah membantu dalam penyelesaian tugas akhir ini.

Dengan ini peneliti akan melakukan Tugas Akhir dengan maksimal dan berharap dengan penelitian ini, ilmu komputasi yang telah dipelajari dapat berpengaruh bagi seluruh kalangan baik orang-orang atau peneliti sendiri dan dengan variasi perspektif dari seluruh pihak, peneliti dengan terbuka hati menerima kritik dan saran dalam perkembangan Penulisan Tugas Akhir ini.

Bandarlampung, Februari 2024

Penulis

DAFTAR ISI

| | |
|-----------------------------------|------|
| HALAMAN PENGESAHAN..... | iii |
| ABSTRAK..... | iv |
| <i>ABSTRACT</i> | v |
| RIWAYAT HIDUP..... | vi |
| MOTTO | vii |
| PERSEMBAHAN | viii |
| KATA PENGANTAR | 1 |
| DAFTAR ISI..... | 3 |
| DAFTAR GAMBAR | 6 |
| DAFTAR TABEL..... | 9 |
| BAB I PENDAHULUAN..... | 11 |
| 1.1 Latar Belakang | 11 |
| 1.2 Identifikasi Masalah | 13 |
| 1.3 Batasan Masalah..... | 13 |
| 1.4 Rumusan Masalah | 13 |
| 1.6 Kerangka Pemikiran | 15 |
| 1.7 Sistematika Penulisan..... | 16 |
| BAB II LANDASAN TEORI | 18 |
| 2.1 Tinjauan Pustaka | 18 |
| 2.2 Tinjauan Studi | 24 |

| | | |
|--------------------------------|--|----|
| 2.2.1 | Manajemen Proyek | 24 |
| 2.2.2 | SDLC (System Development Life Cycle) | 24 |
| 2.2.3 | User Interface..... | 26 |
| 2.2.4 | Design Process | 26 |
| 2.2.5 | Tools User Interface & User Experience | 27 |
| 2.2.6 | Database | 30 |
| 2.2.7 | UML (unified Modeling Language)..... | 32 |
| 2.2.8 | Use Case Diagram | 33 |
| 2.2.9 | Sequence Diagram | 35 |
| 2.2.10 | Activity Diagram | 36 |
| 2.2.11 | Class Diagram | 38 |
| 2.2.12 | Flowchart Diagram..... | 39 |
| 2.2.13 | Nodejs | 41 |
| 2.2.14 | Javascript | 41 |
| 2.2.15 | Visual Studio Code | 41 |
| 2.2.16 | Framework Qausar..... | 42 |
| 2.2.17 | White Box Testing | 42 |
| 2.2.18 | Blackbox Testing..... | 44 |
| 2.2.19 | Skala Likert | 46 |
| BAB III METODE PENELITIAN..... | | 47 |
| 3.1 | Metode Penelitian..... | 47 |
| 3.2 | Metode Pengumpulan Data | 48 |
| 3.3 | Analisa Kebutuhan | 50 |
| 3.3.1 | Analisa Kebutuhan Fungsional | 50 |
| 3.3.2 | Kebutuhan Non Fungsional..... | 52 |
| 3.4 | Perancangan Sistem..... | 52 |
| 3.5 | Use Case Diagram | 53 |
| 3.6 | Sequence Diagram..... | 53 |
| 3.7 | Activity Diagram..... | 59 |
| 3.8 | Flowchart Diagram..... | 62 |

| | | |
|-----------------------------------|---|-----|
| 3.9 | <i>Class Diagram</i> | 62 |
| 3.10 | Perancangan Basis Data | 63 |
| 3.11 | Rancangan Sistem Perangkat Lunak | 68 |
| 3.12 | Jadwal Penelitian | 79 |
| BAB IV HASIL DAN PEMBAHASAN | | 80 |
| 4.1 | HASIL | 80 |
| 4.1.1 | Desain Sistem / Perancangan Sistem..... | 81 |
| 4.1.2 | Implementasi Sistem | 81 |
| 4.2 | PEMBAHASAN | 97 |
| 4.2.1 | Pengujian <i>Black Box</i> | 98 |
| 4.2.2 | Pengujian WhiteBox..... | 102 |
| 4.2.3 | Uji Skala Likert | 112 |
| BAB V KESIMPULAN DAN SARAN..... | | 116 |
| 5.1 | Kesimpulan..... | 116 |
| 5.2 | Saran | 116 |
| DAFTAR PUSTAKA | | 117 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 1. 1 kerangka pemikiran..... | 15 |
| Gambar 2. 1 Tahapan SDLC..... | 25 |
| Gambar 2. 2 wireframe | 27 |
| Gambar 2. 3 icon google..... | 28 |
| Gambar 2. 4 Envanto | 29 |
| Gambar 2. 5 Adobe Xd..... | 29 |
| Gambar 2. 6 Figma | 30 |
| Gambar 2. 7 Invision..... | 30 |
| Gambar 2. 8 Gambar Framework Quasar | 42 |
| Gambar 2. 9 white box..... | 43 |
| Gambar 2. 10 black box | 44 |
| Gambar 3. 1 Tahap Pengembangan Software Dengan Metode SDLC | 48 |
| Gambar 3. 2 Use Case Diagram..... | 53 |
| Gambar 3. 3 sequence Diagram Registrasi | 53 |
| Gambar 3. 4 sequence diagram login..... | 54 |
| Gambar 3. 5 sequence diagram halaman utama..... | 55 |
| Gambar 3. 6 sequence diagram profile desa | 55 |
| Gambar 3. 7 sequence diagram pengajuan surat..... | 56 |
| Gambar 3. 8 sequence diagram peta pembangunan | 57 |
| Gambar 3. 9 sequence diagram pengaduan | 57 |
| Gambar 3. 10 sequence jadwal posyandu | 58 |
| Gambar 3. 11 activity diagram registrasi dan login | 59 |
| Gambar 3. 12 activity diagram profile desa..... | 60 |
| Gambar 3. 13 activity diagram pengajuan surat | 60 |
| Gambar 3. 14 activity peta pembngunan | 61 |

| | |
|--|----|
| Gambar 3. 15 activity diagram pengaduan | 61 |
| Gambar 3. 16 activity diagram jadwal posyandu..... | 62 |
| Gambar 3. 17 flowchart diagram aplikasi SIPDes | 62 |
| Gambar 3. 18 class diagram aplikasi SIPDes | 63 |
| Gambar 3. 19 basis data aplikasi SIPDes..... | 63 |
| Gambar 3. 20 wireframe halaman Login | 69 |
| Gambar 3. 21 wireframe halaman home..... | 70 |
| Gambar 3. 22 wireframe halaman menu utama | 71 |
| Gambar 3. 23 wireframe halaman profile | 72 |
| Gambar 3. 24 wireframe halaman pengajuan surat..... | 73 |
| Gambar 3. 25 wireframe halaman permohonan berhasil dikirim | 74 |
| Gambar 3. 26 wireframe halaman riwayat surat | 75 |
| Gambar 3. 27 wireframe halaman pembangunan | 76 |
| Gambar 3. 28 wireframe halaman pengaduan | 77 |
| Gambar 3. 29 wireframe halaman posyandu | 78 |
| Gambar 4. 1 Gambar Flowchart Perancangan aplikasi SIPDes | 81 |
| Gambar 4. 2 Halaman Login..... | 82 |
| Gambar 4. 3 Halaman Registrasi | 83 |
| Gambar 4. 4 Tampilan Halaman Utama | 84 |
| Gambar 4. 5 Tampilan Profile Desa..... | 85 |
| Gambar 4. 6 Tampilan Layanan Adminitrasi..... | 86 |
| Gambar 4. 7 Tampilan Halaman Permintaan Terkirim..... | 87 |
| Gambar 4. 8 Tampilan Halaman Riwayat Surat | 88 |
| Gambar 4. 9 Tampilan Halaman Unduh Surat..... | 89 |
| Gambar 4. 10 Tampilan Halaman Peta Bangunan | 90 |
| Gambar 4. 11 Tampilan Halaman Pengaduan | 91 |

| | |
|--|-----|
| Gambar 4. 12 Tampilan Hal Input Pengaduan..... | 92 |
| Gambar 4. 13 Tampilan Hal Pengaduan Berhasil Dikirim | 93 |
| Gambar 4. 14 Tamplan Hal Riwayat Pengaduan..... | 94 |
| Gambar 4. 15 Tampilan Halaman Posyandu | 95 |
| Gambar 4. 16 Halaman Admin | 96 |
| Gambar 4. 17 Flowgraph Login & Registrasi..... | 102 |
| Gambar 4. 18 Flowgraph Halaman Utama | 104 |
| Gambar 4. 19 Flowgraph Halaman Layanan Desa | 105 |
| Gambar 4. 20 Flowgraph Peta Pembangunan..... | 107 |
| Gambar 4. 21 Flowgraph Halaman Pengaduan | 108 |
| Gambar 4. 22 Flowgraph Halaman Posyandu | 110 |
| Gambar 4. 23 Diagram Usability | 114 |
| Gambar 4. 24 Diagram Information Quality..... | 115 |

DAFTAR TABEL

| | |
|--|-----|
| Tabel 2. 1 Use Case Diagram..... | 33 |
| Tabel 2. 2 Sequence Diagram | 35 |
| Tabel 2. 3 Activity Diagram | 37 |
| Tabel 2. 4 Class Diagram..... | 38 |
| Tabel 2. 5 Flowchart Diagram | 39 |
| Tabel 2. 6 perbandingan..... | 46 |
| Tabel 3. 1 wawancara | 49 |
| Tabel 3. 2 data user | 64 |
| Tabel 3. 3 data admin..... | 65 |
| Tabel 3. 4 data profile desa | 65 |
| Tabel 3. 5 data surat | 66 |
| Tabel 3. 6 data proyek..... | 66 |
| Tabel 3. 7 data pengaduan | 67 |
| Tabel 3. 8 data posyandu | 68 |
| Tabel 3. 9 jadwal penelitian | 79 |
| tabel 4. 1 pengujian blackbox registrasi dan login..... | 98 |
| tabel 4. 2 pengujian blackbox halaman utama | 99 |
| tabel 4. 3 pengujian blackbox halaman profile desa | 99 |
| tabel 4. 4 pengujian blackbox halaman profile desa | 99 |
| tabel 4. 5 pengujian blackbox halaman layanan desa | 99 |
| tabel 4. 6 pengujian blackbox halaman layanan desa | 99 |
| tabel 4. 7 pengujian blackbox hal peta pembangunan..... | 100 |
| tabel 4. 8 pengujian blackbox hal peta pembangunan..... | 100 |
| tabel 4. 9 pengujian blackbox hal pengaduan | 101 |
| tabel 4. 10 pengujian blackbox hal pengaduan | 101 |

| | |
|---|-----|
| tabel 4. 11 pengujian blackbox hal posyandu | 101 |
| tabel 4. 12 Cyclomatic Complexity Login & Registrasi | 103 |
| tabel 4. 13 Cyclomatic Complexity..... | 104 |
| tabel 4. 14 Cyclomatic Complexity..... | 106 |
| tabel 4. 15 Cyclomatic Complexity Halaman Peta Pembangunan..... | 107 |
| tabel 4. 16 Cyclomatic Complexity Halaman Pengaduan..... | 109 |
| tabel 4. 17 Cyclomatic Complexity Halaman Posyandu..... | 110 |
| tabel 4. 18 Tabel Skala Likert | 112 |
| tabel 4. 19 Penilaian Interval..... | 113 |
| tabel 4. 20 pengujian interval..... | 113 |
| tabel 4. 21 Dimensi Usability..... | 114 |
| tabel 4. 22 Information Quality | 115 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemerintahan desa adalah sistem pemerintahan yang berlaku di tingkat desa atau kelurahan. Tujuan dari pemerintahan ini adalah untuk mengurus tugas-tugas administratif, memajukan pembangunan, dan memberikan layanan kepada masyarakat pada tingkat lokal. Pemerintahan desa memiliki struktur organisasi dan mekanisme pengambilan keputusan yang dirancang untuk memenuhi kebutuhan dan kepentingan masyarakat di tingkat desa. Elemen-elemen penting dalam pemerintahan desa melibatkan kepala desa (atau lurah), perangkat desa, dan badan permusyawaratan desa (BPD) sebagai lembaga legislatif. Kepala desa memimpin pemerintahan desa dan bertanggung jawab atas pengelolaan administrasi, pembangunan, dan pelayanan kepada warganya. Perangkat desa, seperti sekretaris desa, membantu dalam operasional sehari-hari pemerintahan desa. BPD, yang terdiri dari perwakilan masyarakat desa, memiliki peran dalam membuat peraturan desa dan mengawasi kebijakan kepala desa. Pemerintahan desa juga melibatkan partisipasi aktif masyarakat melalui forum musyawarah desa, di mana keputusan-keputusan penting dibahas dan diambil. Pemerintahan desa di Indonesia diatur oleh undang-undang tertentu, seperti Undang-Undang Nomor 6 Tahun 2014 tentang Desa. (Ali & Saputra, 2020).

Desa Sumber Agung merupakan salah satu desa yang terletak di Kecamatan Way Sulan, Kabupaten Lampung Selatan, Provinsi Lampung. Jumlah penduduk Desa Sumber Agung adalah 5.377 jiwa. Kantor Desa Sumber Agung bertugas menyelenggarakan urusan politik melalui pemerintahan desa dan badan-badan yang terlibat dalam organisasi dan pengelolaan kepentingan masyarakat desa setempat. Dalam menjalankan misinya mengelola kepentingan masyarakat, Kantor Desa Sumber Agung memberikan pelayanan publik seperti informasi desa, administrasi, mengelola komunikasi, dan menangani berbagai proses komunikasi, antara lain: 1. Surat keterangan pindah, 2. Surat keterangan tidak mampu, 3. Surat keterangan

usaha, 4. Surat kematian, 5. Catatan penduduk dll. Dengan adanya Pelayanan publik yang diberikan oleh pemerintah desa sebagai lembaga layanan publik masih dianggap kurang baik oleh masyarakat setempat. Penyebabnya adalah

pelayanan yang masih manual, pelayanan yang lambat, proses administrasi yang memakan waktu lama, pelayanan yang diterima masyarakat kurang optimal, dan penyediaan informasi mengenai desa sangat terbatas.

Kemajuan teknologi informasi dan komunikasi telah memberikan solusi bervariasi untuk meningkatkan efisiensi layanan, terutama dalam ranah pelayanan publik yang berbasis good governance. Memberikan sumber daya manusia, regulasi, sarana, dan prasarana adalah aspek penting yang harus dipertimbangkan dalam penyelenggaraan pemerintahan. Salah satu tujuan utama dalam menerapkan good governance adalah memberikan kemampuan kepada lembaga pemerintah untuk menyediakan layanan publik yang lebih unggul. Oleh karena itu, memulai inovasi dalam pelayanan publik memerlukan komitmen yang kuat dari pemerintah. (Junirianto & Wita, 2020).

Sistem Pelayanan Publik Desa (SIPDes) merupakan platform pengelolaan desa yang menyediakan berbagai layanan, antara lain: Informasi mengenai pembangunan desa, pemerintahan, kependudukan, pelayanan publik, dan berbagai pelayanan lainnya. Sistem ini akan memudahkan pemerintah desa dalam mengelola data dan informasi desa secara terpusat dan efektif. Aplikasi ini menjadi solusi untuk mempercepat pelayanan publik di desa.

SDLC adalah pendekatan yang dipilih oleh penulis. *SDLC*, atau Siklus Hidup Pengembangan Perangkat Lunak, sering disebut juga sebagai Siklus Hidup Pengembangan Sistem. Metode Ini merupakan proses yang digunakan untuk mengembangkan atau memodifikasi suatu sistem perangkat lunak dengan menggunakan berbagai bentuk metodologi yang telah diterapkan sebelumnya dalam pengembangan sistem perangkat lunak. *SDLC* juga merupakan model dan metodologi yang digunakan untuk memudahkan proses pengembangan rekayasa perangkat lunak. (ROSA, 2022).

Dalam menuju transformasi digital masyarakat menjadi pilar utama untuk pembangunan akses digital, Sehingga waktu yang dihasilkan sangat efisien. Oleh karena itu penulis ingin merancang sebuah "Rancangan SIPDes Menggunakan *Metode Software Development Life Cycle SDLC* (Studi Kasus : Desa Sumber Agung, Lampung Selatan)". Sehingga SIPDes sebagai fasilitas pelayan publik, dapat memudahkan dalam mengoptimalkan pelayanan dan kinerja yang lebih efektif dan efisien.

1.2 Identifikasi Masalah

Berdasarkan konteks permasalahan yang telah di jelaskan, maka identifikasi masalah dalam penelitian ini ialah :

- a. Pelayanan publik pada Pemerintahan Desa Sumber Agung masih manual, sehingga proses pelayanan membutuhkan waktu yang lama.
- b. Proses pelayanan publik tidak efektif sehingga mempengaruhi kinerja menjadi lambat dan akses yang terbatas mengenai informasi desa.

1.3 Batasan Masalah

Dengan adanya identifikasi masalah diatas, batasan masalah pada penelitian ini penulis merancang aplikasi SIPDes pada bagian pelayanan publik diantaranya profile desa, pelayanan administrasi, pengaduan, jadwal posyandu, cek bansos, dan peta pembangunan. Aplikasi ini digunakan hanya untuk warga desa Sumber Agung dan staff kantor desa Sumber Agung.

1.4 Rumusan Masalah

Rumusan masalah yang diperoleh dari analisis masalah diatas ialah, "Bagaimana implementasi penerapan SIPDes dengan menggunakan metode *SDLC* (*Software Development life cycle*)"

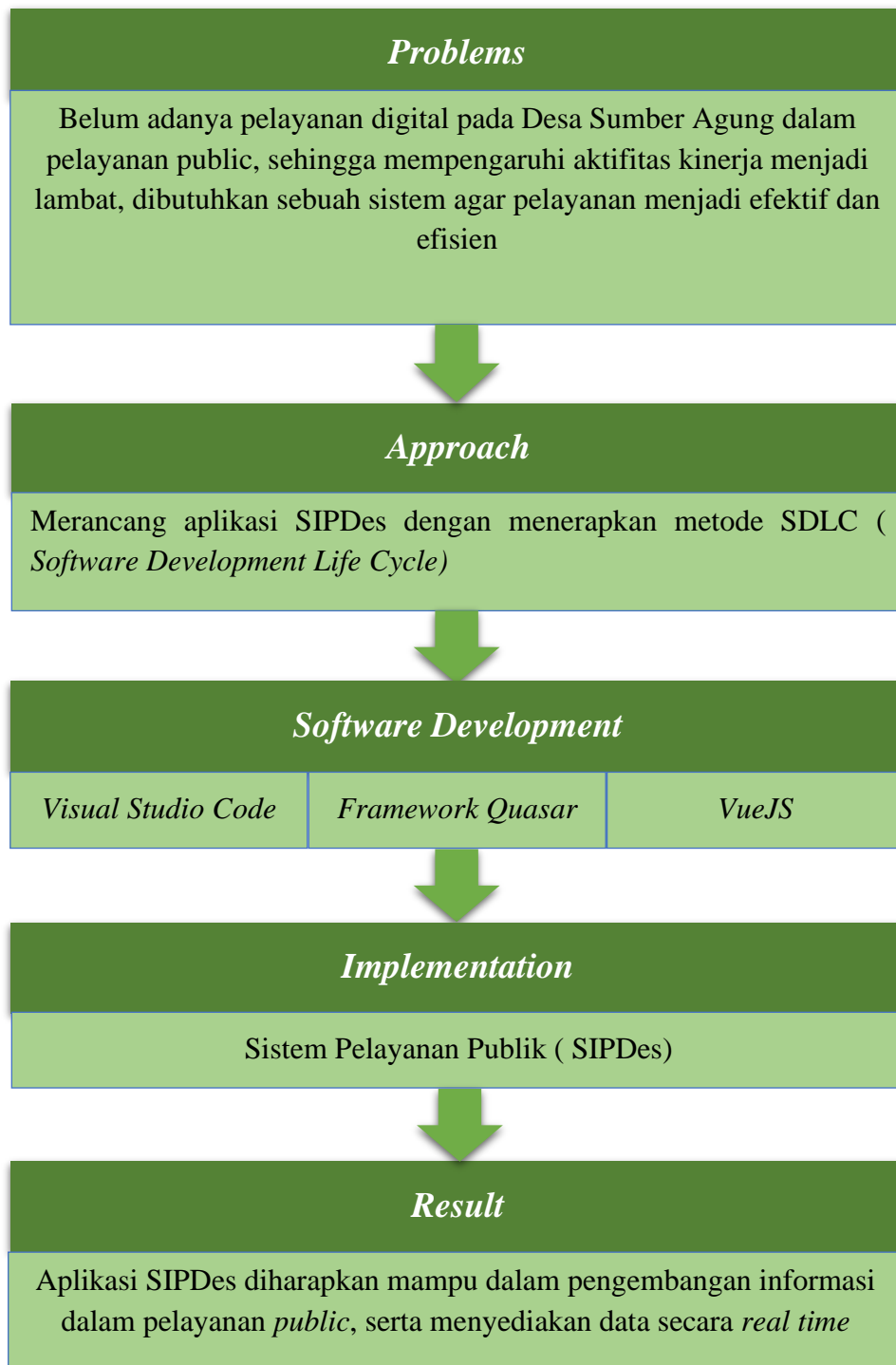
Berikut ini merupakan tujuan dari penelitian yang dilakukan oleh penulis yaitu :

- a. Merancang SIPDes pada Desa Sumber Agung, Lampung Selatan, yang efektif dan efisien sehingga dapat mengoptimalkan kinerja dalam pelayanan publik.
- b. Meningkatkan kualitas layanan publik pada Desa Sumber Agung dengan memberikan kemudahan atau kesediaan untuk mengakses atau menggunakan suatu layanan publik dan informasi desa.

Manfaat penelitian ini bagi penulis yaitu :

1. Secara Teoritis:
 - a. Pada Penelitian ini memberikan kontribusi terhadap pengetahuan dibidang sistem informasi pelayanan desa dengan menggunakan metode *Software Development Life Cycle SDLC*.
2. Secara praktis:
 - a. Penelitian ini dapat membantu pemerintah desa untuk meningkatkan efisiensi dan kualitas pelayanan kepada warga desa, termasuk dalam hal administrasi, dan layanan lainnya.
 - b. penelitian ini akan memberikan data dan informasi yang lebih baik kepada pemerintah desa sehingga memungkinkan pemerintah desa mengambil keputusan yang lebih baik mengenai investasi di bidang teknologi informasi dan pengembangan sistem.

1.6 Kerangka Pemikiran



Gambar 1. 1 kerangka pemikiran

1.7 Sistematika Penulisan

Di bawah ini adalah struktur penulisan yang disusun oleh penulis terbagi menjadi 5 bab, yakni sebagai berikut :

I. PENDAHULUAN

Dalam bab ini, terdapat penjelasan mengenai proses identifikasi permasalahan sebagai dasar penelitian. Pendekatan ini melibatkan beberapa langkah yang terdokumentasi untuk menguraikan permasalahan, seperti menyusun ruang lingkup atau batasan masalah, merumuskan masalah, menentukan tujuan dan manfaat penelitian, serta merinci ruang lingkup kerangka pemikiran.

II. TEORI DASAR

Mengidentifikasi dan menganalisis tentang studi literatur yang digunakan dan teori sebagai landasan dalam menentukan langkah untuk penelitian yang dilakukan oleh penulis seperti jurnal, buku dan media yang digunakan sebagai acuan.

III. METODE PENELITIAN

Memaparkan dan menjelaskan teknik penelitian yang diterapkan oleh penulis dalam penyusunan tugas akhir serta dalam pelaksanaan penelitian. Ini merupakan langkah dalam perancangan yang disusun dan dikembangkan oleh penulis, mencakup analisis kebutuhan, perancangan penelitian, teknik analisis, dan jadwal penelitian yang diimplementasikan.

IV. HASIL DAN PEMBAHASAN

Dalam bab ini, dipaparkan hasil analisis yang telah dilakukan oleh penulis sebagai konsekuensi dari pelaksanaan penelitian dan riset. Selain itu, diterangkan pula implementasi dari temuan hasil penelitian yang telah diperoleh.

V. KESIMPULAN DAN SARAN

Bagian bab ini berisi kesimpulan yang diperoleh dari hasil penelitian yang dilakukan oleh penulis, beserta rekomendasi yang diajukan oleh penulis berdasarkan perancangan penelitian yang telah disusun oleh penulis.

DAFTAR PUSTAKA

LAMPIRAN

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Untuk mendukung penulisan dan penelitian ini, penulis merujuk pada beberapa jurnal yang relevan dan terkait dengan penelitian serta analisis yang dilakukan. Berikut adalah jurnal-jurnal yang digunakan sebagai dasar teori dan acuan dalam penelitian ini:

2.1.1 Penerapan *Smart Village* Dalam Peningkatan Pelayanan Masyarakat Menggunakan Metode *Web Engineering* (Studi Kasus: Desa Sukanegeri Jaya)

Konsep *smart village* merupakan gagasan membangun desa cerdas dengan memanfaatkan *smart Government*, *smart community*, dan *smart environment* dalam penerapan teknologi informasi. Hingga saat ini, *Smart Village* belum tersebar secara merata, terutama di wilayah pedesaan seperti Desa Sukanegeri di Kecamatan Tanggamus, dengan jumlah penduduk 1.016 jiwa dan 226 kepala keluarga. Desa ini termasuk dalam program implementasi Desa Cerdas Gubernur Lampung. Permasalahan muncul karena pengolahan data kependudukan masih dilakukan secara manual oleh lembaga pemberitaan, dicatat dalam buku dan media cetak. Dampaknya meliputi proses pengolahan data yang lambat, kerusakan data karena format kertas, dan risiko kehilangan atau perubahan data. Bertujuan untuk mewujudkan konsep desa pintar dalam pengelolaan informasi seni dan budaya, permintaan surat, dan produk desa, penelitian ini mengadopsi metode pengembangan sistem berbasis rekayasa *web*. Hasil penelitian ini akan memudahkan masyarakat dalam melihat produk olahan di desa dan menyampaikan surat secara online. (Agung Saputra & Rahman Isnain, 2021).

2.1.2 Peningkatan Pelayanan Publik Pemerintah Desa Melalui Sistem Informasi Kependudukan Berbasis Web

Digitalisasi layanan administrasi pemerintahan agar lebih cepat dan mudah dimungkinkan oleh kemajuan teknologi yang pesat. Penggunaan teknologi memiliki potensi untuk meningkatkan pelayanan publik, khususnya di tingkat desa. Tujuan dari penelitian ini adalah untuk menyediakan sistem informasi kependudukan digital berbasis web bagi pemerintah desa. Penelitian ini menggunakan metodologi eksplorasi dan pengembangan. Empat kategori pengguna yang berbeda dilibatkan dalam perancangan sistem ini: *administrator*, petugas, penduduk, dan kepala desa. Kemampuan situs web, yang mencakup permintaan surat hingga layanan pengaduan, diimplementasikan dengan cara ini. *Stoner Acceptance Test dan Blackbox Testing* adalah metode yang digunakan untuk pengujian sistem. Berdasarkan hasil pengujian, dapat dikatakan bahwa sistem informasi kependudukan yang dibuat memiliki kualitas yang cukup baik untuk digunakan di kalangan

2.1.3 Sistem Informasi Desa Dalam Meningkatkan Pelayanan Publik Berbasis Web

Di zaman globalisasi, perkembangan teknologi informasi mengalami kemajuan yang sangat pesat, dan Internet telah menjadi salah satu elemen yang sangat penting dalam memperoleh berbagai informasi yang diperlukan. Setiap orang dapat mengakses atau menerima data yang dibagikan melalui jaringan terhubung di Internet. Data dan informasi juga sangat diperlukan untuk memenuhi kebutuhan informasi dalam pelayanan publik di Desa Lesung Batu, sebuah kantor pemerintahan di Kabupaten Lahat, Sumatera Selatan. Desa Lesung Batu saat ini masih menggunakan sistem manual dalam memberikan pelayanan dan mengkomunikasikan informasi kepada masyarakat, sehingga diperlukan peningkatan kecepatan dan keakuratan informasi. Penelitian ini bertujuan supaya dapat meningkatkan efektifitas pelayanan publik di tingkat lokal di Desa Lesung Batu melalui pengembangan sistem informasi desa.

Sistem ini memudahkan penyampaian informasi dan memungkinkan pelayanan yang lebih sederhana dan cepat. Dibangun dengan menggunakan bahasa pemrograman *Hypertext Processor* dengan *database MySQL*, *Laravel*, dan *Bootstrap*, sistem informasi desa ini merupakan sistem berbasis web yang meliputi tahap komunikasi, perencanaan, pemodelan, pembangunan, dan pengembangan. Penelitian ini menggunakan *Web Engineering Systems Development*.

Metodologi, model rekayasa perangkat lunak untuk mengembangkan aplikasi.. Penelitian ini diharapkan akan menghasilkan sistem informasi desa berbasis *web* yang bersifat *mobile*, memungkinkan masyarakat untuk dengan mudah menemukan informasi yang dibutuhkan dan meningkatkan efektivitas, kecepatan, dan akurasi pelayanan yang diberikan oleh Pemerintah Desa Lesung Batu.. (Mur et al., 2019).

2.1.4 Transaformasi Digital Melalui Sistem Informasi Desa (SID) : Meningkatkan Kualitas Pelayanan Publik dan Kesejahteraan Masyarakat.

Sistem Informasi Desa merupakan platform yang bertujuan untuk mengumpulkan, melakukan pengelolaan dan menyebarkan informasi terkait pelayanan publik desa yang menggunakan teknologi informasi dan komunikasi yang memungkinkan desa untuk mengintegrasikan data dan memproses pengelolaan desa ke dalam sistem terpusat, sehingga informasi lebih mudah diakses dan diproses. Salah satu keuntungan utama menggunakan SID adalah dapat meningkatkan efektifitas dalam pengelolaan desa. Dengan SID, integrasi data administrasi seperti kependudukan, keuangan dan pelayanan dapat dilakukan secara elektronik, sehingga dapat mengurangi proses manual yang memakan waktu dan rentan terhadap kesalahan. SID memungkinkan pemrosesan data yang cepat dan akurat, meningkatkan efisiensi penggunaan waktu dan sumber daya. Penelitian ini memiliki tujuan untuk mengajak masyarakat dalam partisipasi aktif untuk pengambilan keputusan dan pengawasan dalam penggunaan sumber daya desa, khususnya di Desa Sumamun. Selain itu, implementasi SID diharapkan dapat meningkatkan kualitas pelayanan publik di Desa Sumamun dan memberikan

pelayanan publik yang lebih baik kepada masyarakat sekitar. Melalui SID, masyarakat umum dapat dengan mudah dan cepat mendapatkan informasi terkait seperti peraturan daerah, layanan publik, dan kegiatan desa. Hal ini akan meningkatkan transparansi, keamanan dan aksesibilitas layanan desa dan meningkatkan kesadaran masyarakat terhadap layanan yang ditawarkan. (Mardinata et al., 2023).

2.1.5 Perancangan Sistem Informasi Pelayanan Warga Berbasis Web Dikelurahan SindangSari Kabupaten Tangerang.

Implementasi sistem informasi pelayanan masyarakat berbasis web menjadi langkah inovatif untuk meningkatkan kepuasan warga terhadap kinerja pemerintahan kecamatan dan memajukan teknologi informasi dan komunikasi. Dengan memanfaatkan teknologi ini, staf di kabupaten dapat memberikan informasi terkait pengelolaan kependudukan dengan lebih efisien serta melakukan pengajuan surat kepada pemerintah kota. Berbagai peraturan dan ketentuan mulai dari tingkat RT, RW, hingga kelurahan dapat diakses melalui teknologi ini, menjadi dasar untuk mengatasi tantangan pengelolaan kependudukan di Desa Sindang Sari melalui pengembangan aplikasi sistem informasi ayanan masyarakat berbasis web. Metodologi penelitian yang digunakan adalah *Agile Scrum*. Sistem ini memproses permintaan berbagai produk pengelolaan kependudukan secara langsung dan online, mempermudah petugas kecamatan dalam memasukkan data kependudukan yang sudah tersimpan di dalam *database*. Dengan kemampuannya dalam mengolah data kependudukan, mencatat catatan penduduk, dan fungsi lainnya, sistem ini dapat menyimpan dokumen yang telah diproses dan menghasilkan laporan yang dapat dicetak. Secara keseluruhan, sistem ini dapat membantu mengurangi berbagai permasalahan yang terkait dengan pengelolaan kependudukan di Desa Sindang Sari. (Wannahar et al., 2023).

2.1.6 Comparative Analysis of Software Development Lifecycle Methods in Software Development: A Systematic Literature Review

alam beberapa dekade terakhir, berbagai model *Software Development Lifecycle* (SDLC) telah dikembangkan untuk memenuhi kebutuhan dan tantangan yang berbeda dalam industri perangkat lunak. Tujuan dari penelitian ini adalah untuk menganalisis dan membandingkan beberapa metode *SDLC* yang paling umum. Setelah proses pemilihan dan evaluasi selesai, dilakukan tinjauan pustaka dengan mengumpulkan artikel, buku, dan sumber-sumber lain yang berkaitan dengan metode *SDLC*. Beberapa metode *SDLC* utama dipilih untuk dianalisis secara menyeluruh. Waterfall, Agile, dan Scrum adalah beberapa metode tersebut. Faktor-faktor penting seperti fleksibilitas, kecepatan pengembangan, kemampuan untuk beradaptasi dengan perubahan kebutuhan, dan risiko proyek dievaluasi. Hasil analisis menunjukkan bahwa setiap metode *SDLC* memiliki kekuatan dan kelemahan, dan sesuai untuk berbagai situasi, sementara metode *Agile* dan Scrum menekankan pada fleksibilitas dan kerja sama tim, metode *Waterfall* memberikan struktur yang lebih besar dan kejelasan pada rencana. Penelitian ini bertujuan untuk menentukan metode proses terbaik untuk pengembangan perangkat lunak. Tinjauan literatur ini memberikan pemahaman yang mendalam tentang fitur, kekuatan, dan kelemahan dari berbagai metode *SDLC* yang ada. Dengan pemahaman yang lebih baik mengenai metode-metode tersebut, organisasi dapat memilih metode *SDLC* yang paling sesuai dengan kebutuhan proyek mereka, sehingga dapat meningkatkan efisiensi dan efektivitas pengembangan perangkat lunak. Penelitian ini menghasilkan sebuah proses

2.1.7 Analysis of website-based village population administration system development methods

Sistem informasi administrasi kependudukan desa adalah sebuah fasilitas dalam sistem komputer yang telah dilengkapi dengan fitur-fitur dan dirancang sesuai kebutuhan untuk menginput data-data tertentu, dengan tujuan memudahkan, mempercepat, dan meningkatkan akurasi data yang diolah. Metode yang digunakan dalam tinjauan sistematis ini mengacu pada konsep-konsep terkait sistem

administrasi kependudukan desa. Literatur dicari melalui *Google Scholar* pada bulan Agustus 2021 dengan menggunakan bahasa Indonesia. Penelitian ini menggunakan metode studi literatur untuk mencari referensi dari jurnal, dengan 10 jurnal yang direview. Berdasarkan literatur jurnal yang diterbitkan antara tahun 2017-2021, metode pengembangan sistem informasi administrasi kependudukan desa berbasis website yang paling sering digunakan adalah metode *waterfall*.

2.2 Tinjauan Studi

Berikut ini uraian dan penjelasan teori yang terakait dalam penelitian yang dilakukan :

2.2.1 *Manajemen Proyek*

Manajemen proyek adalah proses pengelolaan, perencanaan, pengembangan sumber daya perusahaan dan pengorganisasian untuk mendapatkan hasil tujuan yang telah ditentukan. Pertumbuhan tim manajemen proyek terjadi sebagai respons terhadap kebutuhan untuk mengidentifikasi pendekatan manajemen yang tepat yang sesuai dengan tujuan dan karakteristik proyek. Proyek dipandang sebagai kegiatan yang menyenangkan yang berbeda dari rutinitas operasional sehari-hari. (Rani, 2016). Pentingnya manajemen proyek terlihat dari masalah-masalah yang sering muncul dalam proyek tanpa adanya solusi yang efektif, seperti penganggaran proyek dan manajemen waktu yang tidak sesuai dengan rencana. (ROSA, 2022). Dalam manajemen proyek memiliki beberapa aktivitas secara umum yang dilakukan diantaranya :

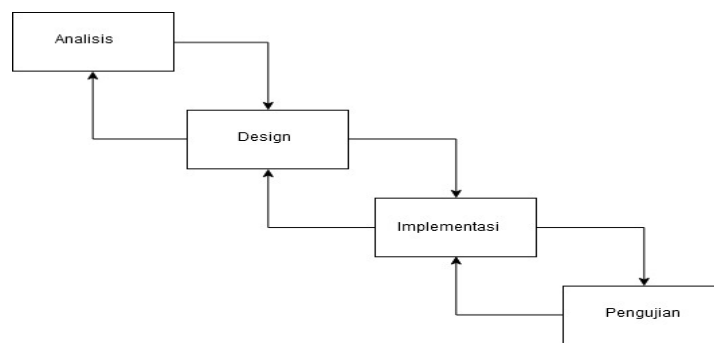
1. Manajemen aktivitas proyek
2. Perencanaan proyek
3. Manajemen resiko

2.2.2 *SDLC (System Development Life Cycle)*

Penulis menggunakan metode *Software Development Life Cycle (SDLC)* sebagai pendekatan. *Software Development Life Cycle*, atau *SDLC*, adalah sebuah proses yang menggunakan berbagai pendekatan yang telah digunakan sebelumnya untuk membuat atau memodifikasi sistem perangkat lunak. Siklus hidup pengembangan perangkat lunak (*SDLC*) mencakup pembuatan atau pengembangan perangkat lunak, atau sistem informasi, yang pada setiap tahapnya juga melalui sebagian dari siklus pengembangan. Untuk memastikan program akhir adalah yang terbaik, hal ini juga melibatkan pengujian sistem. Pada tahap implementasi proyek, yang

didasarkan pada Piagam Proyek dan Rencana Manajemen Proyek, proses *SDLC* diimplementasikan.(ROSA, 2022).

Dalam rekayasa sistem dan perangkat lunak, istilah "*SDLC*" (Siklus Hidup Pengembangan Sistem) mengacu pada proses yang mencakup pendekatan pemodelan dan pengembangan yang digunakan untuk membuat dan memodifikasi sistem. Secara umum, komputer atau sistem informasi terlibat dalam gagasan ini. Siklus hidup pengembangan perangkat lunak (*SDLC*) juga mencakup proses yang digunakan untuk membuat perangkat lunak, yang dibagi menjadi beberapa fase: analisis, desain, implementasi, dan pengujian..



Gambar 2. 1 Tahapan SDLC

Tahapan metode SDLC :

1. Analisa : Analisis: Untuk mengidentifikasi dan menilai masalah, keterbatasan, dan kebutuhan yang diantisipasi serta memberikan rekomendasi untuk perubahan, sistem informasi yang komprehensif harus dibedah ke dalam bagian-bagian komponennya.
2. Desain: Desain berkaitan dengan bagaimana sistem akan dibangun untuk memenuhi persyaratan yang ditemukan selama tahap analisis. Tujuan dari desain sistem adalah untuk memberikan gambaran desain yang komprehensif kepada para programmer untuk digunakan sebagai panduan ketika mereka membuat aplikasi.

3. Implementasi: Semua desain yang telah dirancang sebelumnya diubah menjadi kode program selama fase ini. Kode akhir masih berupa modul-modul yang harus disatukan pada tahap berikutnya.
4. Pengujian: Pada tahap ini, komponen-komponen yang telah dikembangkan sebelumnya diintegrasikan dan diuji untuk memastikan

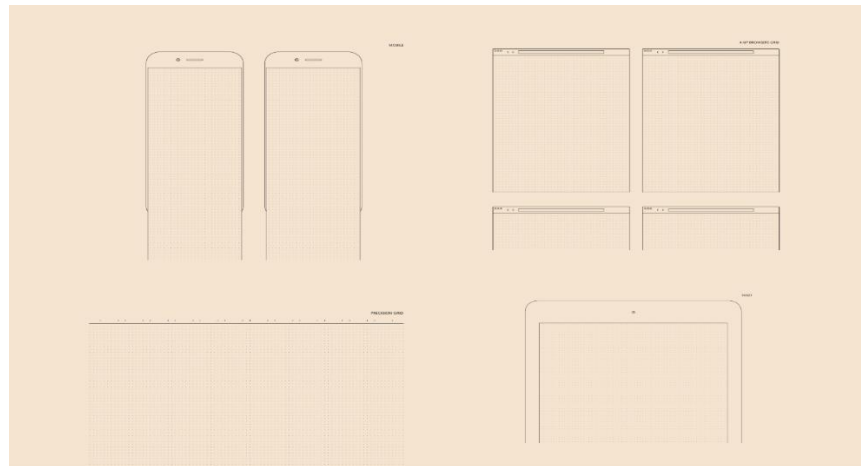
2.2.3 User Interface

UI atau User Interface adalah sistem komunikasi yang memungkinkan interaksi antara pengguna dan suatu sistem dalam sebuah program, seperti situs *web*, perangkat seluler, atau aplikasi perangkat lunak. Mekanisme ini dapat disesuaikan dengan kebutuhan pengguna yang sedang dikembangkan dalam program. Lingkup antarmuka pengguna melibatkan aspek tampilan fisik, pilihan warna, elemen animasi, dan pola interaksi antara program dan pengguna. Desain UI bertujuan untuk menciptakan antarmuka pengguna yang menarik secara visual dan mudah dipahami. (Iverson & Dervan, n.d.).

2.2.4 Design Process

a. Wireframe

Wireframe adalah representasi visual sederhana dari struktur dan tata letak elemen-elemen utama dalam desain sebuah halaman *web*, aplikasi, atau produk digital lainnya. *Wireframe* biasanya digunakan dalam tahap awal perancangan (desain) untuk menggambarkan bagaimana elemen-elemen akan diatur dan berinteraksi satu sama lain tanpa terlalu banyak detail visual. Tujuan utama dari pembuatan *wireframe* adalah untuk fokus pada organisasi informasi, hierarki konten, dan navigasi. *Wireframe* membantu tim desain, pengembangan, dan pemangku kepentingan lainnya untuk memahami dan mengkaji struktur dasar dari desain sebelum memasuki tahap implementasi yang lebih detail.. (Hartawan, 2022) .



Gambar 2. 2 wireframe

b. *Design*

Arti desain adalah mewujudkan suatu ide dalam bentuk model (gambaran aktual desain produk). Berbagai alat desain dapat membantu Anda mendesain produk Anda. Saat mendesain, selalu ikuti prinsip desain yang ada dengan mengikuti pedoman yang diberikan. Kebijakan bersifat spesifik (tergantung *platform*, seperti *iOS*, *Android*, atau *Web*), dan ada pula yang bersifat global. Yang kedua seperti rambu lalu lintas yang jika mengabaikannya dapat mengakibatkan bencana desain. Tujuan utamanya selalu kegunaan (seberapa mudah suatu desain dapat diakses oleh orang yang berbeda dalam kondisi yang berbeda) dan aksesibilitas (seberapa mudah suatu desain digunakan dan seberapa lancar dan cepat tugas yang diberikan padanya dapat diselesaikan). Maka setelah dua hal ini terpenuhi, maka dapat berekspresi sekreatif mungkin dalam sisi *aesthetic* dan *pleasure*.

2.2.5 *Tools User Interface & User Experience*

User Interface (UI) dan pengalaman pengguna (*UX*) adalah aspek penting dalam desain dan pengembangan produk dan aplikasi digital. Fokus dari alat ini adalah untuk menciptakan antarmuka yang menarik secara visual, intuitif, dan ramah pengguna yang meningkatkan pengalaman pengguna secara keseluruhan. Di sini kami memberikan gambaran umum tentang konsep dan beberapa alat yang umum

digunakan dalam desain *UI* dan *UX*. Antarmuka pengguna mengacu pada elemen visual dan tata letak produk digital. Ini mencakup segala sesuatu yang berinteraksi dengan pengguna di layar, seperti tombol, menu, tipografi, warna, dan gambar. Tujuan dari desain UI adalah membuat antarmuka pengguna menarik secara visual dan mudah dipahami. (Kridalukmana & Windasari, 2023).

Prinsip desain secara umum meliputi:

1. **Konsistensi:** Memastikan desain yang konsisten di semua elemen antarmuka untuk menghindari kebingungan.
2. **Hierarki:** Mengatur elemen dengan cara yang memandu perhatian pengguna dan membantu mereka memahami aliran konten.
3. **Umpan Balik:** Memberikan isyarat visual dan umpan balik kepada pengguna saat mereka berinteraksi dengan elemen.
4. **Kesederhanaan:** Menjaga desain tetap bersih dan sederhana untuk mencegah pengguna kewalahan dengan elemen yang tidak perlu.

Beberapa *tools* yang digunakan peneliti dalam penerapan *user interface* sebagai berikut :

a. Google Icon



Gambar 2. 3 icon google

Dalam penggunaan pembuatan aplikasi SIPDes, peneliti menggunakan *icon-icon di google icon*, karena *google icon* memiliki variasi yang cukup interaktif bagi pengguna.

b. *Envanto*



Gambar 2. 4 Envanto

Envanto merupakan *software* desain grafis yang memiliki ragam jenis templet maupun asset yang dapat digunakan dengan membayar *royalty*. Hal yang dapat digunakan seperti stok font, stok video, foto, audio dan templet grafis.

Istilah "antarmuka pengguna" (*UI*) menggabungkan kata "pengguna", yang mengacu pada pengguna, dan kata "antarmuka", yang mengacu pada antarmuka, permukaan, atau tampilan depan suatu objek atau media. *UI* adalah representasi visual dari media atau produk yang bertindak sebagai penghubung antara sistem dan pengguna. *User Interface* memiliki peran sentral dalam fasilitasi interaksi dengan pengguna. Pentingnya *UI* terletak pada kenyataan bahwa ini merupakan tampilan pertama yang dipersepsikan oleh pengguna saat berinteraksi dengan aplikasi, situs web, dan berbagai media lainnya seperti media sosial. (Angela & Erandaru, 2022).

Berikut ini beberapa yang populer:

1. Adobe XD



Gambar 2. 5 Adobe Xd

Adobe XD (Experience Design) merupakan suatu perangkat lunak desain yang dikembangkan oleh *Adobe Systems*. Ini adalah alat yang dirancang khusus untuk mendukung desainer dalam membuat dan mengembangkan desain antarmuka pengguna (*UI/UX*) yang menarik dan interaktif untuk berbagai jenis aplikasi digital seperti aplikasi mobile, *website*, dan lainnya.

2. Figma



Gambar 2. 6 Figma

Alat desain kolaboratif yang memungkinkan kolaborasi waktu nyata pada proyek UI dan UX.

3. InVisio



Gambar 2. 7 Invision

Platform untuk membuat prototipe interaktif dan melakukan pengujian pengguna.

2.2.6 Database

Data adalah fakta yang menggambarkan peristiwa aktual pada suatu titik waktu tertentu. Data tersebut berasal dari peristiwa yang benar-benar terjadi, seperti transaksi penjualan dan pembelian. Datanya sama dengan kuitansi transaksi dalam perusahaan, seperti: Contoh: kwitansi, faktur, formulir, dll. Data yang diolah dapat digunakan untuk membuat informasi dalam format laporan seperti laporan keuangan, laporan penjualan, dll.(Kurniati et al., 2015) .

a. Basis data

Basis data adalah kelompok informasi yang disimpan dalam jangka waktu yang lama, bahkan sampai bertahun-tahun. Keberadaan database sangat penting bagi banyak jenis usaha, baik skala besar maupun kecil. Semua perusahaan memelihara *database* yang berisi informasi penting tentang operasi bisnis mereka. Efektivitas suatu basis data sangat bergantung pada pengetahuan dan teknologi yang diterapkan dan diintegrasikan ke dalam software khusus yang disebut sistem manajemen basis data.

b. Database Management System (DBMS)

Penciptaan kerangka kerja untuk penyimpanan informasi dan penyediaan prosedur untuk data ini merupakan bagian dari sistem basis data, atau *DBMS*. Lebih jauh lagi, sistem basis data memastikan bahwa data tersebut aman bahkan jika terjadi kerusakan atau upaya akses yang tidak sah. Ketika data dibagikan ke beberapa pengguna, data tersebut harus menghilangkan potensi hasil yang tidak konsisten. Diharapkan *DBMS* memiliki fitur-fitur berikut ini: Buat *database* baru dan tentukan skema database oleh pengguna.

1. Memberi pengguna kemampuan untuk membuat kueri dan mengubah data menggunakan bahasa kueri.
2. Mendukung data yang berkapasitas besar dan penyimpanan jangka panjang.
3. Menyediakan akses yang efisien ke data untuk kueri dan modifikasi basis data.
4. Memberikan dukungan untuk pemulihan basis data jika terjadi kerusakan, dan beberapa kesalahan penyalahgunaan yang disengaja.
5. Mengontrol akses data oleh banyak pengguna secara bersamaan. Faktanya, berbagai perangkat lunak *DBMS* biasa digunakan untuk mengelola database perusahaan.

c. Bahasa Basis Data

Basis data menyediakan bahasa definisi data, yang merinci skema basis data dan bahasa manipulasi data, yang menyatakan kueri dan pembaruan dari basis data.

1. *Data-Definition Language (DDL)*

Administrator database sering menggunakan serangkaian perintah *DDL*. Baris-baris ini mencakup implementasi skema data spesifik yang biasanya tidak dapat diakses oleh kriteria integritas tertentu, seperti larangan akun departemen negatif, harus diikuti oleh nilai-nilai data yang terkandung dalam database. Anda dapat menerapkan batasan ini dengan menggunakan *DDL*. Sistem database akan memverifikasi keberadaan batasan-batasan ini setiap kali database diubah. Anda dapat memodifikasi struktur tabel dan menambahkan indeks baru menggunakan bahasa ini. Kamus data yang berisi informasi tentang keadaan sebenarnya dihasilkan dengan menggabungkan prosedur-prosedur *DDL* ini (*create, change, rename, drop*).

2. *Data-Manipulation Language (DML)*

Pengguna dapat mengakses dan memodifikasi data menggunakan bahasa *DML* sesuai dengan model data yang relevan. *DML* juga digunakan untuk menjalankan tindakan penyisipan, pembaruan, dan penghapusan data dalam tabel. *DML* berisi perintah-perintah berikut: menyisipkan, memperbarui, menghapus, dan memilih. Data

3. *Control Language (DCL)*

DCL adalah bahasa *SQL* untuk memanipulasi dan mengatur izin pengguna. Perintah-perintah yang termasuk dalam *DCL* melibatkan (*Grant, dan revoke*).

2.2.7 UML (unified Modeling Language)

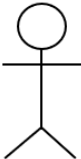

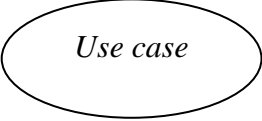
UML (Unified Modeling Language) ialah bahasa pemodelan yang digunakan dalam pengembangan perangkat lunak. *UML* menyajikan notasi visual dan metode untuk mendeskripsikan dan merencanakan desain sistem perangkat lunak. *UML* memungkinkan pengembang perangkat lunak untuk memvisualisasikan dengan lebih baik struktur dan perilaku sistem yang mereka buat. (Julianti et al., 2019).


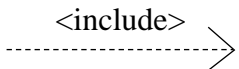
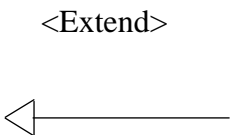


2.2.8 Use Case Diagram


Use case Diagram yang merepresentasikan aktor, *use case*, dan hubungan antar aktor sebagai rangkaian tindakan yang memberikan nilai terukur kepada aktor tersebut. Kasus penggunaan direpresentasikan dalam diagram kasus penggunaan UML sebagai *elips horizontal*.(Haviluddin, 2011). *Use Case* memiliki dua istilah :

1. Sistem *use case*; interaksi dengan system.
2. *Business use case*; interaksi perusahaan dengan konsumen atau kejadian nyata

Tabel 2. 1 Use Case Diagram

| Simbol | Nama | Keterangan |
|---|-----------------|---|
|  | Actor | Penentuan peran-peran yang dilakukan oleh pengguna ketika terlibat dalam Kasus Penggunaan. |
|  | Association | Relasi atau koneksi antar sebuah objek dengan objek yang lainnya . |
|  | <i>Use case</i> | Penjabaran mengenai rangakain urutan atau langkah-langkah sistem dalam mendapatkan hasil yang dapat dikuru bagi seoarng pelaku atau actor |


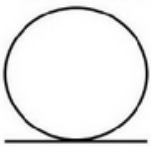
| | | |
|---|----------------|--|
|  | Generalization | Relasi di mana objek turunan mengambil sifat dan struktur data objek yang berada di hierarki di atasnya. |
|  | Include | Menguraikan use case sumber tersebut secara tegas atau terus terang. |
|  | Extend | Menentukan bahwa use case target mengembangkan perilaku dari kasus penggunaan sumber pada titik tertentu. |
|  | System | Menyatakan dengan jelas paket yang membatasi tampilan sistem |
|  | Collaboration | Interaksi antara elemen-elemen yang saling berkolaborasi untuk menciptakan perilaku yang lebih kompleks daripada yang dapat dihasilkan oleh setiap elemen secara individu. |

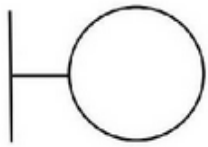
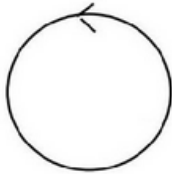
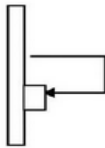


| | | |
|---|---------|---|
|  | Catatan | Komponen real-time yang aktif saat program berjalan dan mencerminkan sumber daya komputasi. |
|---|---------|---|

2.2.9 Sequence Diagram

Diagram urutan menjelaskan bagaimana objek berinteraksi melalui waktu dan diurutkan berdasarkan urutan waktu. Secara sederhana, diagram urutan memberikan gambaran langkah-langkah secara berurutan, termasuk kronologi perubahan logis yang terjadi untuk mencapai suatu hal sesuai dengan diagram penggunaan. (Haviluddin, 2011).

Tabel 2. 2 Sequence Diagram

| Simbol | Nama | Keterangan |
|---|--------------|---|
|  | Aktor | Komponen ini mencerminkan pengguna yang melakukan interaksi dengan sistem, baik dari dalam maupun dari luar sistem. |
|  | Entity Class | Cerminan sistem sebagai landasan dalam menyusun <i>database</i> . |


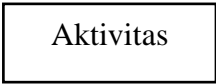
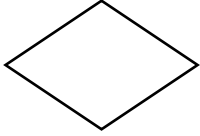


| | | |
|---|-----------------------|---|
|  | <i>Boundary Class</i> | Melayani komunikasi antar lingkungan sistem. |
|  | <i>Control Class</i> | Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika. |
|  | <i>Recursive</i> | Pesan untuk diri sendiri |
|  | <i>Activation</i> | Mewakili proses durasi aktivasi sebuah operasi. |
|  | <i>Life line</i> | Untuk menerangkan urutan kejadian yang terjadi pada suatu objek selama proses desain grafis |

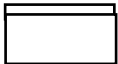
2.2.10 Activity Diagram

Diagram aktivitas menggambarkan urutan kegiatan dalam sistem atau perangkat lunak. Ini mencakup alur proses, pengambilan keputusan yang mungkin

terjadi, dan terminasi sistem. Rencana aktivitas ini dikonstruksi berdasarkan satu atau beberapa skenario dalam diagram kasus pengguna. (Ariesta et al., 2021).

Tabel 2. 3 Activity Diagram

| Simbol | Nama | Keterangan |
|---|----------------------|--|
|  | Status Awal | Diagram aktivitas memuat kondisi permulaan atau status awal. |
|  | Aktivitas | Umumnya, aktivitas diawali oleh sistem dengan memanfaatkan kata kerja. |
|  | Percabangan/Decision | Percabangan adalah kondisi di mana terdapat beberapa opsi aktivitas yang dapat dipilih. |
|  | Penggabungan | Penggabungan adalah proses menggabungkan dua atau lebih aktivitas menjadi satu kesatuan. |
|  | Status Akhir | Diagram aktivitas memiliki titik akhir yang mengindikasikan kondisi akhir yang diperoleh oleh sistem setelah melaksanakan serangkaian aktivitas. |


| | | |
|---|-----------------|--|
|  | <i>Swimlane</i> | Pemisahan entitas bisnis yang bertanggung jawab terhadap berbagai kegiatan yang berlangsung. |
|---|-----------------|--|




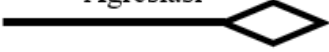

2.2.11 Class Diagram

Class Diagram menggambarkan kondisi saat ini dari struktur kelas sistem dan memberikan informasi tentang fitur, interaksi, dan peran masing-masing kelas. Salah satu jenis diagram populer yang digunakan untuk membantu memvisualisasikan struktur sistem adalah diagram kelas. Diagram kelas membantu dalam memahami arsitektur sistem yang tidak lengkap dengan menyediakan struktur setiap kelas di seluruh fase desain..(Haviluddin, 2011). *Class* memiliki tiga area pokok :

- a. Nama (dan stereotype)
- b. Atribut
- c. Metod

Tabel 2. 4 Class Diagram

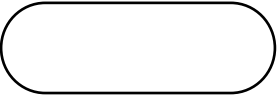
| Simbol | Keterangan |
|---|--|
| Kelas <div data-bbox="517 1503 737 1630"> <div>Nama_kelas</div> <div>+ atribut</div> <div>+ operasi()</div> </div> | Entitas dalam struktur sistem. |
| Antarmuka / interfaces <div data-bbox="563 1731 632 1798">  </div> nama interfaces | Serupa dengan ide konsep antarmuka dalam pemrograman berorientasi objek. |

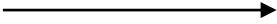


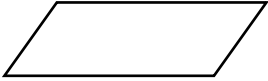

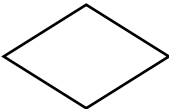
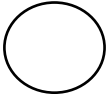
| | |
|--|---|
| <p style="text-align: center;"><u>Association/Asosiasi</u></p>  | <p>Hubungan antara kelas dengan pengertian umum, umumnya asosiasi juga memiliki keterangan multiplicitas.</p> |
| <p style="text-align: center;">Asosiasi berarah</p>  | <p>Relasi antara kelas yang memiliki signifikansi yang dimanfaatkan oleh kelas lain..</p> |
| <p style="text-align: center;">Generalisasi</p>  | <p>Hubungan antara kelas dengan hierarki generalisasi dari yang umum hingga yang spesifik.</p> |
| <p style="text-align: center;">Agresiasi</p>  | <p>Hubungan antar kelas dengan makna semua bagian.</p> |
| <p style="text-align: center;"><i>Depedency / Kebergantungan</i></p>  | <p>Relasi antar kelas dengan makna kebergantungan antar kelas.</p> |

2.2.12 Flowchart Diagram

Diagram alur merupakan representasi visual dari langkah-langkah dan proses dalam suatu program. Diagram alur membantu analis memecah masalah menjadi bagian-bagian yang lebih rinci dan mengevaluasi pilihan-pilihan alternatif untuk implementasi. Simbol dalam diagram alur antara lain:

Tabel 2. 5 Flowchart Diagram

| Simbol | Nama | Fungsi |
|---|------------|--|
|  | Terminator | Tanda <i>flowchart</i> oval mewakili awal atau akhir dari suatu proses atau operasi. |

| | | |
|---|--|--|
|  | Garis Alir (<i>Flow Line</i>) | Simbol yang menunjukkan arah aliran dalam suatu sistem |
|  | <i>Preparation</i> | Proses pemberian nilai awal atau inisialisasi |
|  | Proses | Proses pengolahan data atau proses perhitungan |
|  | <i>Input/Output Data</i> | Proses input / output data, parameter, informasi. |
|  | <i>Predefined Process</i> (Sub Program) | Proses menjalankan sub program atau proses permulaan sub program. |
|  | <i>Decission</i> | Perbandingan pernyataan, pemilihan data yang memberikan opsi untuk tindakan selanjutnya. |
|  | <i>On Page Connector</i> | Membandingkan bagian-bagian <i>flowchart</i> yang berbeda dalam satu halaman. |

| | | |
|---|---------------------------|--|
|  | <i>Off Page Connector</i> | Menghubungkan elemen-elemen <i>flowchart</i> yang berbeda pada halaman-halaman yang berbeda. |
|---|---------------------------|--|

2.2.13 Nodejs

Nodejs lahir dari mesin *JavaScript* milik Google yang digunakan untuk browser *Chrome* yang dikenal dengan sebutan V8, dengan *Nodejs* proses pengembangan aplikasi dapat dilakukan dengan menggunakan bahasa *JavaScript* baik pada sisi klien maupun server (Maulana, 2022).

2.2.14 Javacript

JavaScript merupakan bahasa skrip yang umum digunakan dalam domain teknologi, terutama di dunia internet. Bahasa ini memiliki kemampuan untuk beroperasi pada sebagian besar peramban web, termasuk *Internet Explorer (IE)*, *Netscape*, *Opera*, *Mozilla Firefox*, dan berbagai peramban web lainnya. (Sari et al., 2019).

2.2.15 Visual Studio Code

Editor sumber terbuka dan gratis yang disebut *Visual Studio Code (VS Code)* dibuat khusus untuk membuat dan men-debug aplikasi cloud dan online kontemporer. Sistem operasi *Windows*, *Linux*, dan *OS X* dapat digunakan dengan editor ini. Sesuai dengan rincian yang diberikan dalam buku ini, *Visual Studio Code* kompatibel dengan lebih dari tiga puluh bahasa pemrograman, basis data, dan markup, seperti *JavaScript*, *C#*, *C++*, *PHP*, *Java*, *HTML*, dan beberapa lainnya..(Kahlert & Giza, 2016).

2.2.16 Framework Quasar

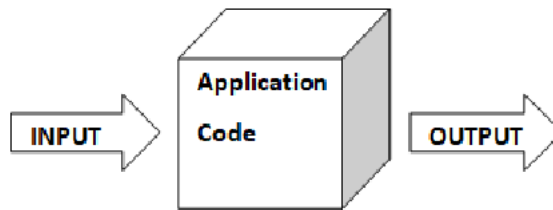


Gambar 2. 8 Gambar Framework Quasar

Quasar Framework adalah sebuah kerangka kerja *Javascript* untuk pengembangan *UI* bersifat *open source* yang dibangun oleh *MIT* yang memungkinkan programmer web dengan kemampuan *HTML*, *javascript*, dan *CSS* untuk membangun dan menyebarkan aplikasi mereka ke berbagai platform perangkat yaitu *browser*, *desktop* dan *seluler*. Dalam lingkungan *web*, *Quasar framework* mendukung: *Single Page Application* (SPA), *Server Side Renderer* (SSR), *Progressive Web Application* (PWA) *mode deployment* dengan memanfaatkan satu set kode. *Quasar framework* ditulis berdasarkan kerangka kerja *UI VueJS* (Murad et al., 2019) .

2.2.17 White Box Testing

Pengujian *white box* adalah suatu metode untuk menguji perangkat lunak yang berfokus pada struktur aplikasi. Tujuan utama dari pengujian *white box* adalah memeriksa logika internal aplikasi, struktur kode, dan alur kontrol. Cara ini juga dikenal dengan banyak istilah lain, seperti: Uji *white box*, uji kotak terbuka, uji kotak kaca, uji kotak transparan, uji basis kode atau uji struktur. Dalam pengujian *white box*, seorang pengujian harus memiliki pengetahuan mendalam tentang bahasa pemrograman yang digunakan (kode sumber). Pengujian *white box* berfokus pada pengujian internal aplikasi dan membutuhkan pemahaman rinci tentang struktur aplikasi. Pengujian *white box* melibatkan pemeriksaan semua jalur kode sumber untuk memastikan kelancaran dan kebenaran fungsionalitas aplikasi. Penguji dalam pengujian *white box* perlu memiliki keterampilan pemrograman yang solid. (Verma et al., 2017) . *White Box Testing* membutuhkan sumber daya yang sangat terampil dan pengetahuan implementasi .



Gambar 2. 9 white box

1. Teknik Pengujian White Box

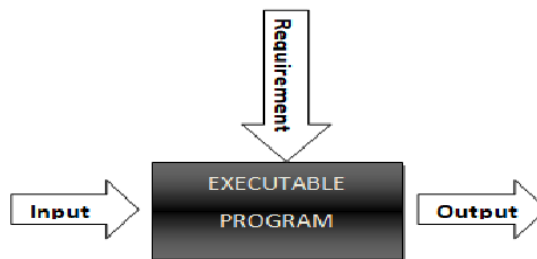
- a. *Control Flow testing* : Ini adalah teknik pengujian struktural. Ini memeriksa aliran kontrol program. Ini berisi keduanya jalur yang lebih sederhana dan rumit. Pengujian ini diterapkan pada semua perangkat lunak untuk menguji aliran kontrol. Ini adalah teknik mendasar. Dia berlaku pada program kecil dan subbagian besar program.
- b. *Path Testing* : Dalam Pengujian Jalur, ia menguji semua kemungkinan jalur program. Ini adalah teknik yang komprehensif. Ini menjamin bahwa semua jalur dilalui setidaknya satu kali. Cakupan jalur tekniknya lebih baik daripada cakupan Cabang. Itu bagus untuk menguji program yang kompleks.
- c. *Data Flow Testings* : Pengujian Aliran Data berfokus pada titik di mana variabel menerima nilai dan di mana nilai berada digunakan. Ini memiliki beberapa masalah seperti:
 - 1) Variabel yang dideklarasikan tetapi tidak pernah digunakan dalam program.
 - 2) Variabel yang tidak pernah dideklarasikan tetapi digunakan dalam program.
 - 3) Deklarasi beberapa variabel sebelum digunakan

Teknik pengujian *White Box* dapat diterapkan pada tingkat pengujian perangkat lunak berikut:

- a. Pengujian Satuan
- b. Pengujian Sistem
- c. Pengujian Integrasi

2.2.18 *Blackbox Testing*

Pengujian *black box* adalah jenis pengujian perangkat lunak yang tujuan utamanya adalah untuk mengevaluasi fungsi aplikasi. Masukan yang diberikan dan penerapan hasil yang diantisipasi untuk setiap nilai masukan adalah fokus utama dari pengujian ini. Spesifikasi dan kebutuhan program menjadi dasar metodologi pengujian ini. Metode ini termasuk dalam kategori pengujian berbasis perilaku, di mana penguji tidak mengetahui cara kerja internal item yang diuji. Selain itu, pengujian berbasis spesifikasi adalah istilah lain yang sering digunakan untuk menggambarkan pengujian *black box*. Alasan mengapa disebut sebagai "*black box*" adalah karena penguji tidak harus mengetahui cara kerja internal dari implementasi aplikasi. Input yang valid dan salah ditangani oleh pengujian kotak hitam sesuai dengan spesifikasi klien.(Verma et al., 2017) .



Gambar 2. 10 *black box*

1. Teknik Pengujian *Black Box*

- a. *Equivalence partitioning* : adalah pendekatan desain kasus uji. Metode ini membagi semua nilai input menjadi beberapa bagian yang terpisah. Baik partisi yang sah maupun tidak sah. Untuk menemukan kekurangannya, kasus uji dibuat dari setiap bagian.

- b. *Boundary Value Analysis* : Kasus uji dibuat dengan menggunakan metode ini untuk menemukan kesalahan. Untuk data uji, pendekatan ini menggunakan nilai batas domain input atau nilai batas terdekat. Kasus uji dibuat untuk nilai batas yang valid dan masalah. Satu Kasus Uji dipilih untuk setiap nilai batas.
- c. *Cause Effect Graphing* : Teknik desain pengujian perangkat lunak ini berkonsentrasi pada perilaku eksternal sistem. Operasi Boolean digunakan untuk memastikan
- d. *Decision Table Based Testing* : Metode ini bekerja dengan baik untuk memproses banyak input dan menghasilkan output yang sesuai. Kelengkapan tabel keputusan adalah salah satu sifatnya. Ini mencakup setiap nilai yang mungkin diberikan pada nilai kondisi. Hal ini sangat membantu untuk menyederhanakan skenario pengujian bisnis yang rumit.
- e. *Error Guessing* : Ini adalah metode untuk membuat asumsi dan tebakan. Penguji yang berpengalaman akan menemukan kekurangannya. Keahlian penguji adalah kunci dari keefektifan teknik ini; penguji yang terampil dapat mengidentifikasi jenis dan lokasi kekurangan.

Dalam teknik ini dibutuhkan semua nilai masukan dan terbagi menjadi Pengujian *Black Box* dilakukan dalam level pengujian perangkat lunak:

- a. Pengujian Integrasi
- b. Pengujian Sistem
- c. Pengujian Penerima

2. Perbandingan *Black Box Testing* dan *White Box Testing*

Tabel 2. 6 perbandingan

| <i>Black Box Testing</i> | <i>White Box Testing</i> |
|---|---|
| disebut Berbasis Spesifikasi Teknik. | disebut Struktural Teknik Pengujian. |
| Struktur internal dan pengetahuan pengkodean tidak dibutuhkan | struktur internal dan pengetahuan pengkodean dibutuhkan |
| Fokus utama pada fungsionalitas sistem. | Konsentrasi utama pada kode struktur, cabang, <i>loop</i> , kondisi, dll. |
| Penerapan pengetahuan tidak diperlukan. | Penerapan diperlukan pengetahuan. |

2.2.19 *Skala Likert*

Teknik yang digunakan untuk mengukur anggapan, sikap, atau pandangan seseorang atau kelompok terhadap suatu kejadian atau fenomena sosial adalah *skala Likert*. Ada dua kategori pertanyaan dalam *Skala Likert*: pertanyaan positif dan pertanyaan negatif. Pertanyaan negatif diberi skor 1, 2, 3, 4, dan 5, sedangkan pertanyaan positif diberi skor 5, 4, 3, 2, dan 1. (Viktor et al., 2019).

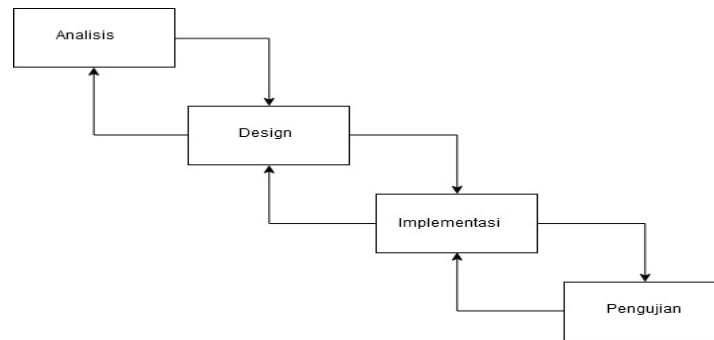
BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian merupakan langkah-langkah yang penulis ikuti dalam melakukan proses penelitian dengan menggunakan metode penelitian terapan. Penelitian terapan merupakan jenis penelitian yang mempunyai tujuan praktis dan keinginan untuk memperbaiki sesuatu, meningkatkan efisiensi, dan meningkatkan efektivitas. Penelitian terapan didasarkan pada fakta-fakta praktis tentang penerapan dan pengembangan pengetahuan yang diperoleh dari pengetahuan dasar. Penelitian ini berfokus pada situasi dunia nyata dan bertujuan untuk menemukan solusi konkrit terhadap permasalahan tertentu. Tujuan utama penelitian terapan adalah untuk memberikan solusi yang bermanfaat bagi masyarakat, baik bagi individu maupun kelompok, dan untuk kepentingan industri atau politik, serta untuk tujuan pengetahuan ilmiah. (Endang Mulyatiningsih, 2011).

Penulis menggunakan metodologi *SDLC*. Metodologi *Software Development Life Cycle (SDLC)* adalah proses mengembangkan atau memodifikasi sebuah sistem perangkat lunak dengan menggunakan model metadis yang telah digunakan untuk mengembangkan sistem perangkat lunak lain di masa lalu. Dalam rekayasa sistem dan perangkat lunak, istilah "*SDLC*" (Siklus Hidup Pengembangan Sistem) mengacu pada proses yang mencakup pendekatan pemodelan dan pengembangan yang digunakan untuk membuat dan memodifikasi sistem. Secara umum, komputer atau sistem informasi terlibat dalam gagasan ini. Siklus hidup pengembangan perangkat lunak (*SDLC*) juga mencakup proses yang digunakan untuk membuat perangkat lunak, yang dibagi menjadi beberapa fase: analisis, desain, implementasi, dan pengujian.



Gambar 3. 1 Tahap Pengembangan Software Dengan Metode SDLC

SDLC yang digunakan dalam penelitian ini melibatkan tahapan-tahapan seperti yang ditunjukkan pada gambar 3.1. Metode ini dipilih karena dalam setiap tahapannya dapat menerima perubahan dan pengembangan konsep dari tahap sebelumnya. Hal ini mendukung tujuan penelitian untuk menciptakan sistem yang mampu beradaptasi dengan perkembangan proses bisnis di masa depan.

3.2 Metode Pengumpulan Data

Metode pengumpulan data merupakan proses dimana data informasi dikumpulkan dan disimpan untuk tujuan analisis. Pengumpulan data pada penelitian yang dilakukan penulis yaitu dengan melakukan observasi dan wawancara kepada petugas/staff kantor desa Sumber Agung terkait sistem informasi pelayanan publik desa. Kemudian penulis melakukan pengamatan dan analisa dari hasil wawancara yang telah dilakukan dengan studi literatur seperti, buku, jurnal, artikel, serta referensi yang bersumber dari *website*. Berikut merupakan metode pengumpulan data yang dilakukan :

1. Observasi

Pada tahap observasi peneliti melakukan observasi terkait alur pelayanan publik dan penyampaian informasi di desa Sumber Agung, Lampung Selatan.

2. Wawancara

Peneliti mengumpulkan informasi mengenai karakteristik pengguna, tujuan pengguna dan tugas pengguna, Peneliti melakukan wawancara kepada beberapa pihak yang terlibat dibidang proses penanganan pelayanan publik pada desa Sumber Agung, Lampung Selatan.

Berikut merupakan hasil dari wawancara yang telah dilakukan oleh penulis kepada kepala desa Sumber Agung Lampung Selatan :

Tabel 3. 1 wawancara

| No | Pertanyaan | Hasil |
|----|--|---|
| 1 | Apakah sistem kinerja pada kantor Desa Sumber Agung, Lampung Selatan sudah terdigitalisasi ? | Belum ada sistem digital dalam pelayanan publik desa |
| 2 | Bagaimana mekanisme pelayanan publik Desa Sumber Agung kepada masyarakat ? | Mekanisme dalam pelayanan publik desa masih terhitung manual, seperti datang secara langsung ke kantor desa, dan harus melakukan penandaanganan secara langsung kepada kepala desa dan RT. |
| 3 | Kendala apa saja yang dialami oleh masyarakat pada pelayanan publik Desa? | <ol style="list-style-type: none">1. Kurang nya informasi terkait persyaratan dalam pengajuan layanan dan kegiatan desa2. Waktu yang terbatas3. Alur pelayanan yang membutuhkan waktu lama. |

| | | |
|---|---|---|
| 4 | Apakah pelayanan publik selama ini berjalan dengan efektif dan efisien ? | Masih banyak kekurangan dan kendala yang dihadapi seperti berbelitnya alur dalam pelayanan, persyaratan yang begitu banyak, dan tertinggalnya informasi mengenai desa. |
| 5 | Bagaiman cara penyampaian terkait informasi desa kepada masyarakat desa ? | Dalam penyampaian informasi desa kepada masyarakat masih menggunakan cara manual seperti penyebaran surat, mengumpulkan ketua RT, dan kemudian di sampaikan kepada masyarakat desa. |

3.3 Analisa Kebutuhan

Menganalisis kebutuhan perangkat lunak memerlukan mencari tahu langkah-langkah apa yang harus dilakukan sistem untuk memenuhi semua fungsinya dan menemukan kendala implementasi dan operasional. Salah satu cara untuk mengkonseptualisasikan proses analisis kebutuhan perangkat lunak adalah sebagai upaya untuk memahami data, model, dan spesifikasi sistem yang dibutuhkan pengguna. Dua komponen utama dari analisis kebutuhan sistem adalah kebutuhan fungsional dan non-fungsional.

3.3.1 Analisa Kebutuhan Fungsional

Berikut adalah kebutuhan fungsional dalam perancangan SIPDes di Desa Sumber Agung Lampung Selatan yang telah disusun :

1. Petugas Kelurahan

- a. Admin petugas kelurahan dapat melakukan proses login untuk mengakses data masyarakat, melihat rincian data masyarakat, mengedit data masyarakat, dan menghapus data masyarakat.
 - b. Pegawai kelurahan memiliki kemampuan untuk mengelola informasi, termasuk menginput, mengedit, dan menghapus informasi
2. Masyarakat
- a. Masyarakat dapat dengan mudah melihat dan mengakses berbagai layanan yang tersedia dalam aplikasi.
 - b. Masyarakat dapat melakukan pencarian layanan cek bansos dengan memasukkan NIK pada kolom pencarian.
 - c. Proses pengajuan layanan desa oleh masyarakat melibatkan input jenis surat, jenis kelamin, nama lengkap, NIK, nomor KTP, dan nomor telepon.
 - d. Masyarakat dapat memonitor status pengerjaan layanan surat.
 - e. Masyarakat berhak mengunduh hasil pengajuan layanan surat yang telah diajukan.
3. Sistem
- a. Sistem dapat melakukan pembuatan akun untuk pengguna baru dengan menginput nik, username, email, nomor telepon, dan password
 - b. Sistem mendukung proses login melalui akun yang sudah dibuat sebelumnya dengan memasukkan email yang terdaftar dan kata sandi.
 - c. Sistem memiliki kemampuan mengirimkan kode verifikasi kepada pengguna melalui email yang terdaftar
 - d. Halaman utama sistem menampilkan berbagai layanan dan informasi terkait desa.
 - e. Pilihan pelayanan desa ditampilkan dengan jelas di halaman utama.
 - f. Sistem dapat menyimpan data personal pengguna.
 - g. Sistem dapat menyimpan riwayat aktivitas pengguna, termasuk riwayat pengajuan layanan desa yang pernah diajukan sebelumnya.

3.3.2 Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional mempersempit diskusi tentang fungsionalitas sistem atau penurunan layanan. Persyaratan non-fungsional mengaburkan elemen-elemen seperti perangkat lunak dan perangkat keras yang digunakan. Sebagai komponen pengembangan SIPDes, analisis kebutuhan non-fungsional pada saat ini berfokus pada perangkat keras, seperti laptop yang mengembangkan sistem dengan spesifikasi sebagai berikut:

- a. *Processor Intel Corei3 (Ram 8GB)*
- b. Memory 2048 RAM DDR3
- c. Koneksi internet kecepatan 2Mbps
- d. Hardisk 500 GB
- e. Mouse
- f. Smartphone Oppo A15

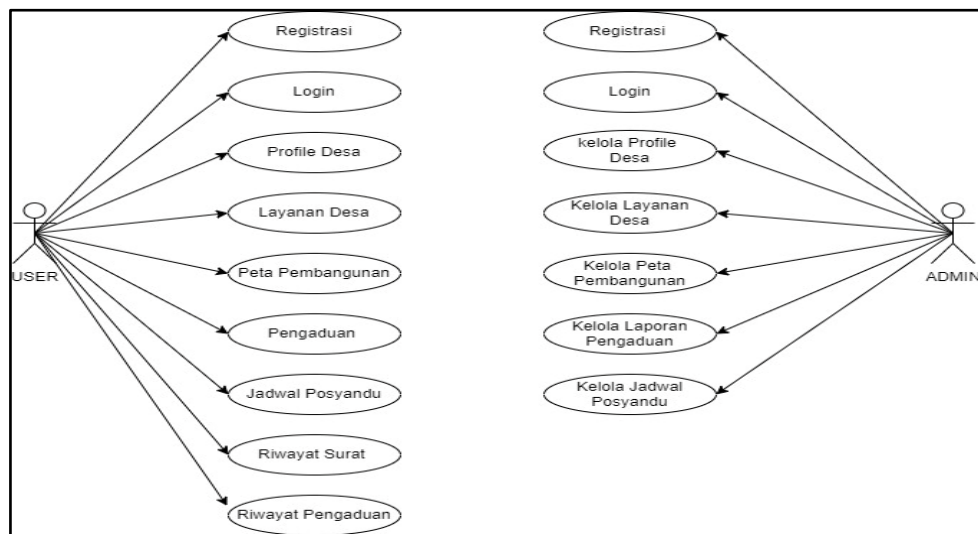
Software Spesifikasi Perangkat lunak untuk pembangunan sistem yaitu :

- a. Bahasa *script* | *PHP , HTML, CSS*.
- b. Database | *MogoDB*.
- c. Editor | *Photoshop*.
- d. *Web Broser / Mozilla Firefox, Internet Explorer dan Google Crome*.

3.4 Perancangan Sistem

Setelah melakukan analisis kebutuhan, kebutuhan yang telah diidentifikasi dianalisis lebih lanjut untuk mempermudah proses pembangunan sistem. Pada penelitian ini, perancangan sistem melibatkan penggunaan beberapa diagram, termasuk *use case diagram*, *sequence diagram*, *Activity diagram*, *Flowchart diagram*, dan *Class diagram*. Model operasional sistem informasi yang akan dikembangkan direpresentasikan melalui diagram-diagram tersebut. Flowchart, sebagai contoh, menggambarkan interaksi antara satu atau lebih aktor dengan sistem informasi yang sedang dibangun. Dengan menggunakan diagram ini, dijelaskan secara visual kegiatan-kegiatan yang dilakukan oleh sistem.

3.5 Use Case Diagram

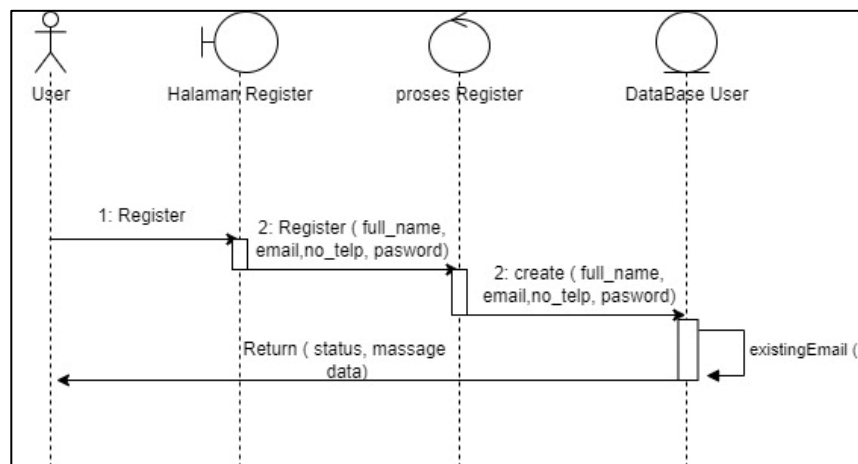


Gambar 3. 2 Use Case Diagram

3.6 Sequence Diagram

Dibawah ini merupakan *Sequence Diagram* pada sistem aplikasi pelayanan Desa (SIPDes).

3.6.1 Sequence diagram Registrasi



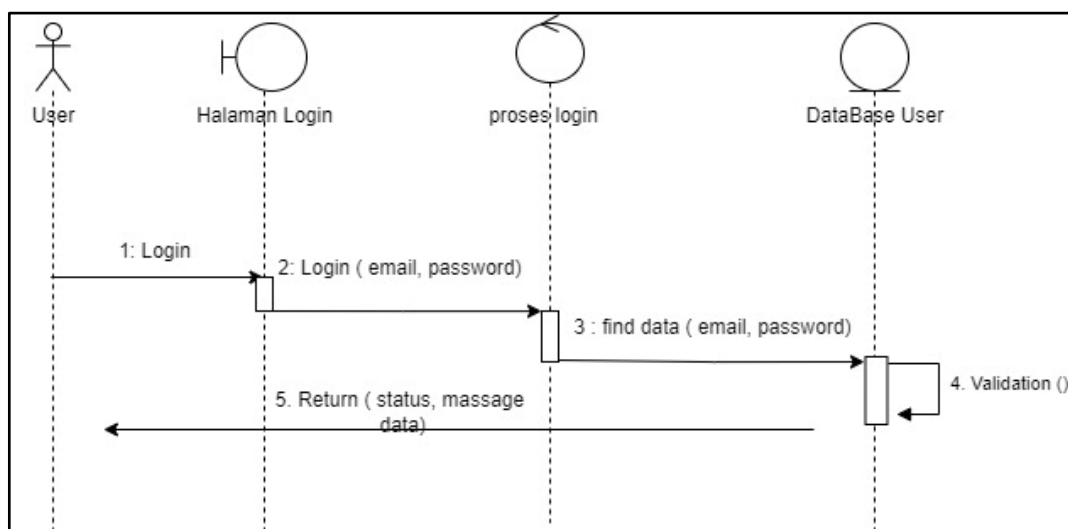
Gambar 3. 3 sequence Diagram Registrasi

Penjelasan :

1. User masuk halaman registrasi.

2. User mengisi form registrasi seperti, full_name, email, no_telp dan password.
3. Database menyimpan data yang telah diisi oleh user pada table user.
4. Database akan memvalidasi apakah email sudah pernah terdaftar atau belum oleh user pada SIPDes.
5. Apabila user berhasil mendaftarkan akun, sistem memberikan pesan kepada user jika berhasil registrasi.

3.6.2 Sequence diagram login

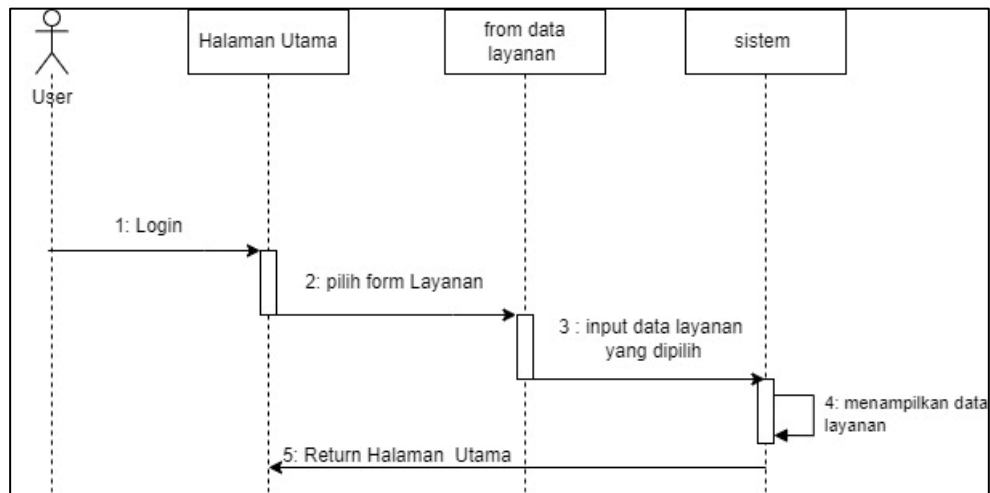


Gambar 3. 4 sequence diagram login

Penjelasan :

1. Setelah user berhasil registrasi sistem akan menampilkan halaman login.
2. User mengisi form registrasi dengan memasukan email dan password yang telah didaftarkan oleh.
3. Controller mengirim request data untuk menemukan data pada table user
4. Kemudian table user akan memvalidasi apakah email dan password yang dimasukan benar atau salah.
5. Kemudian sistem menampilkan status atau pesan berhasil login atau tidak

3.6.3 Sequence Diagram Halaman Utama

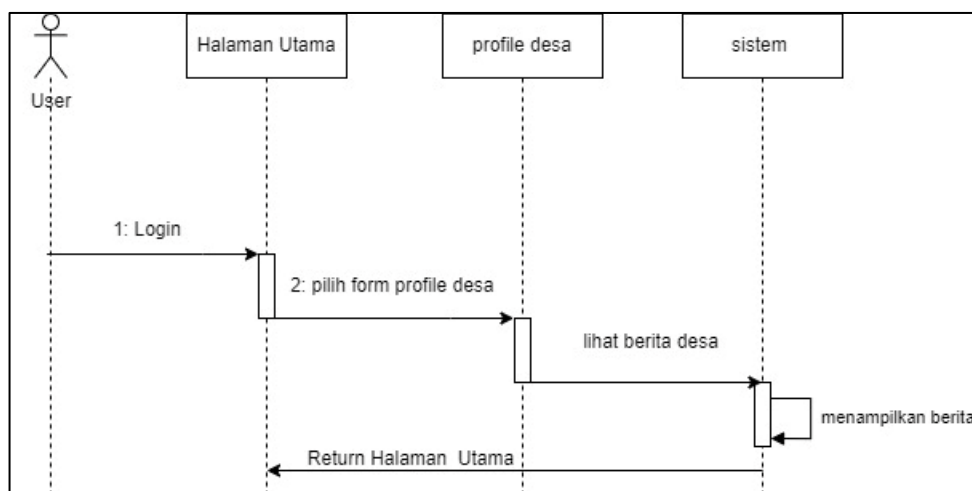


Gambar 3. 5 sequence diagram halaman utama

Penjelasan :

1. Setelah user berhasil login user masuk halaman utama
2. Kemudian user memilih menu form layanan pada sistem
3. Sistem akan menampilkan menu form layanan
4. Kemudian user klik menu form yang dipilih
5. Sistem akan menampilkan data layanan

3.6.4 Sequence Diagram Layanan Profile Desa

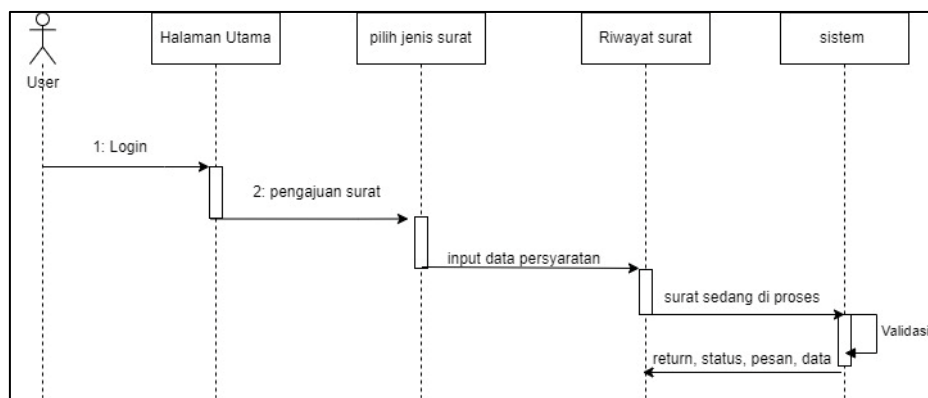


Gambar 3. 6 sequence diagram profile desa

Penjelasan :

1. User masuk halaman utama.
2. Kemudian user pilih form layanan profile desa
3. Sistem akan menampilkan profile desa
4. Kemudian User lihat berita desa
5. Sistem akan menampilkan berita desa.

3.6.5 Sequence diagram pengajuan surat

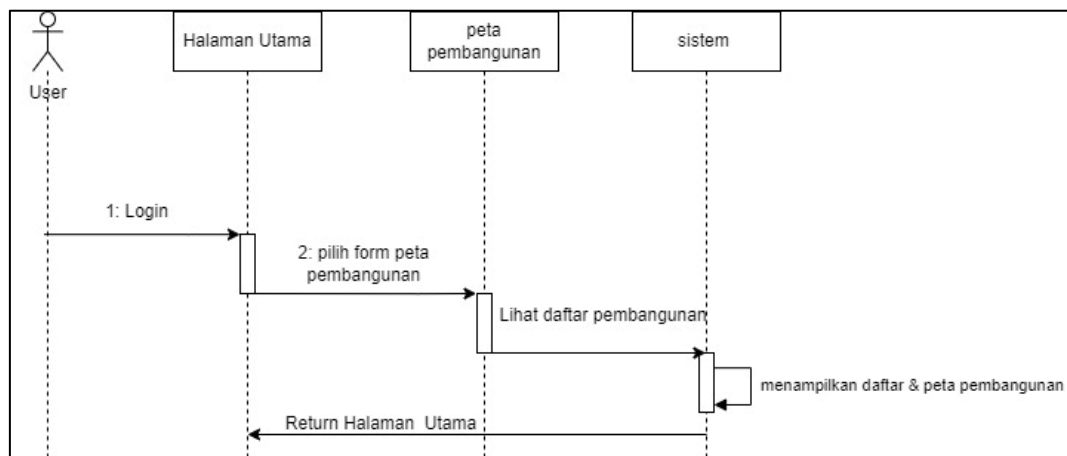


Gambar 3. 7 sequence diagram pengajuan surat

Penjelasan :

1. User masuk halaman utama
2. User pilih layanan pengajuan surat
3. Kemudian user pilih jenis surat yang akan diajukan seperti (SKTM, SKU, surat Domisili)
4. User input data persyaratan pengajuan surat.
5. sistem akan menampilkan riwayat surat yang sudah terkirim
6. sistem akan memproses surat kemudian sistem akan memberikan pesan berhasil atau tidak .
7. user mengunduh hasil pengajuan surat berupa file pdf.

3.6.6 Sequence Diagram Layanan Peta Pembangunan

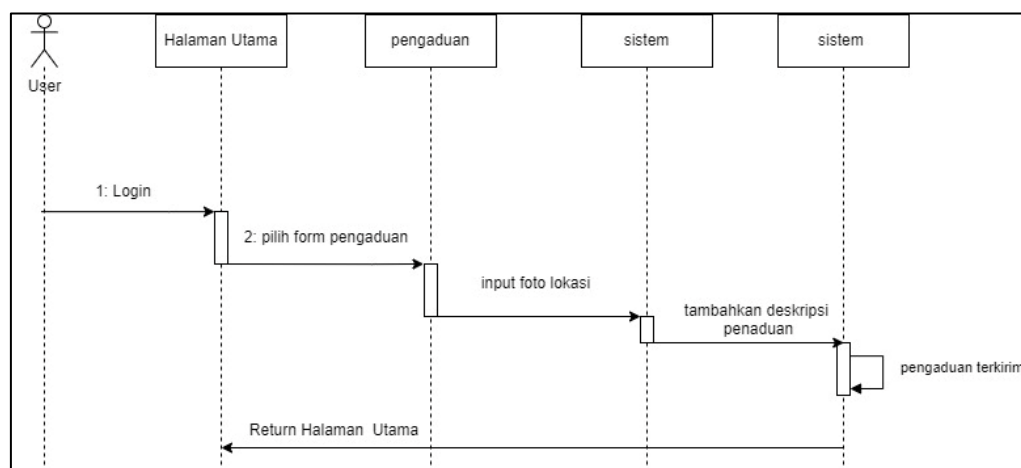


Gambar 3. 8 sequence diagram peta pembangunan

Penjelasan :

1. User masuk pada halaman utama
2. Kemudian user memilih from pelayanan peta pembangunan
3. Sistem akan menampilkan menu from pelayanan peta pembangunan
4. User mencari dan lihat daftar pembangunan
5. Sistem akan menampilkan daftar peta pembangunan yang sedang diproses/dikerjakan

3.6.7 Sequence Diagram Layanan pengaduan

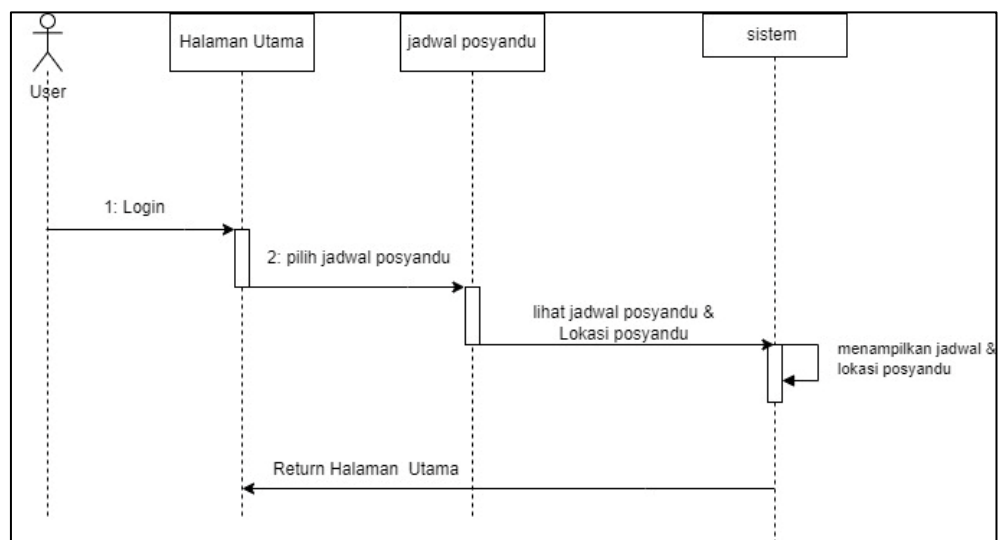


Gambar 3. 9 sequence diagram pengaduan

Penjelasan :

1. User masuk pada halaman utama
2. Kemudian user memilih form layanan pengaduan
3. Sistem akan menampilkan menu form layanan pengaduan
4. Kemudian user input foto lokasi
5. Sistem akan menampilkan kamera/memilih foto pada gallery
6. User diminta untuk menambahkan deskripsi pengaduan
7. Sistem akan memproses pengaduan berhasil terkirim
8. User kembali pada halaman utama

3.6.8 Sequence Diagram pada layanan jadwal posyandu



Gambar 3. 10 sequence jadwal posyandu

Penejelasan :

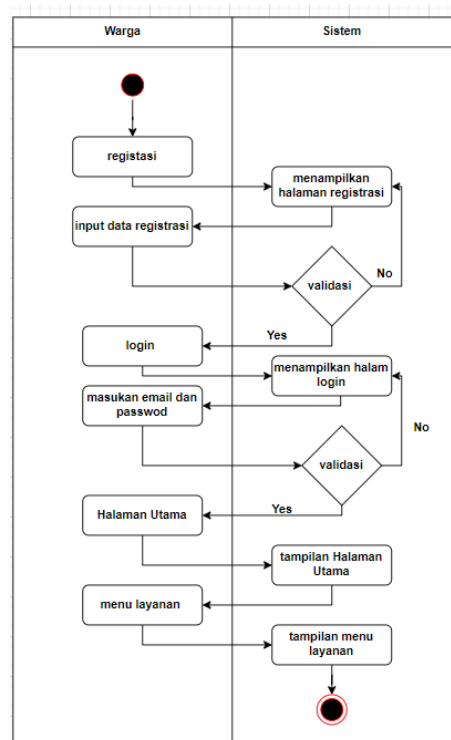
1. User masuk pada halaman utama
2. Kemudian user memilih form pelayanan jadwal posyandu
3. Setelah user memilih form layanan posyandu sistem akan menampilkan menu pada layanan jadwal posyandu
4. User melihat jadwal posyandu pada data daftar jadwal posyandu yang tersedia

5. Sistem akan menampilkan data layanan posyandu seperti (jadwal, Lokasi Dan nama dokter)
6. Setelah melakukan akses pada layanan jadwal posyandu User dapat kembali pada halaman utama.

3.7 Activity Diagram

Diagram aktivitas menunjukkan bagaimana sebuah sistem atau proses bekerja dan apa saja aktivitasnya. Diagram aktivitas SIPDes ditunjukkan di bawah ini.

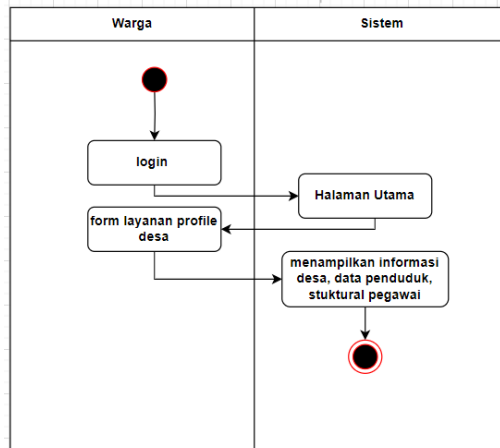
3.7.1 Activity Diagram Login Dan Registrasi



Gambar 3. 11 activity diagram registrasi dan login

3.7.2 Activity Diagram Layanan Profile Desa

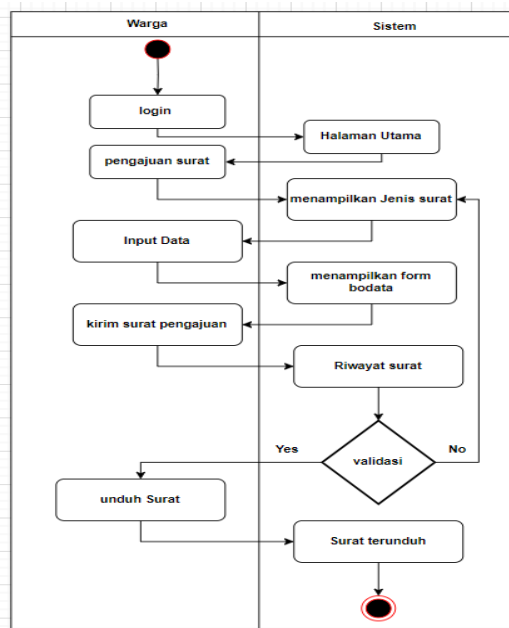
Berikut merupakan *activity diagram* halaman layanan profile desa pada aplikasi SIPDes.



Gambar 3. 12 activity diagram profile desa

3.7.3 Activity Diagram Pengajuan surat

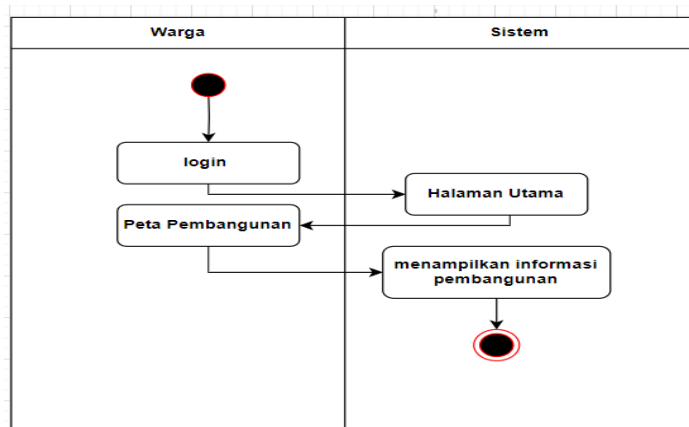
Berikut merupakan *activity diagram* pengajuan surat pada aplikasi SIPDes



Gambar 3. 13 activity diagram pengajuan surat

3.7.4 Activity Diagram Peta Pembangunan

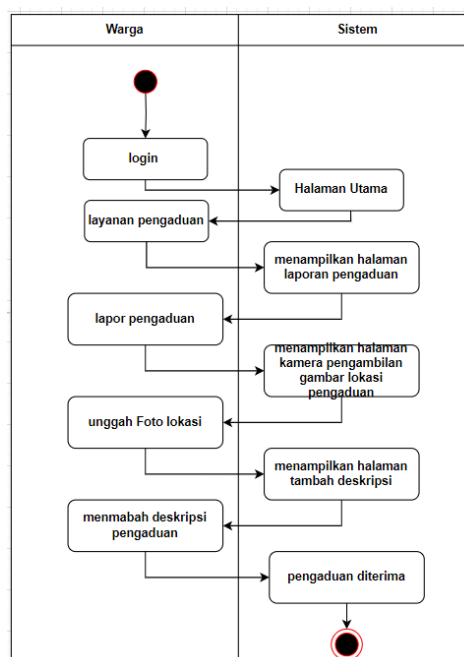
Berikut ini *activity diagram* menu peta pembangunan pada aplikasi SIPDes.



Gambar 3. 14 activity peta pembngunan

3.7.5 Activity Diagram Pengaduan

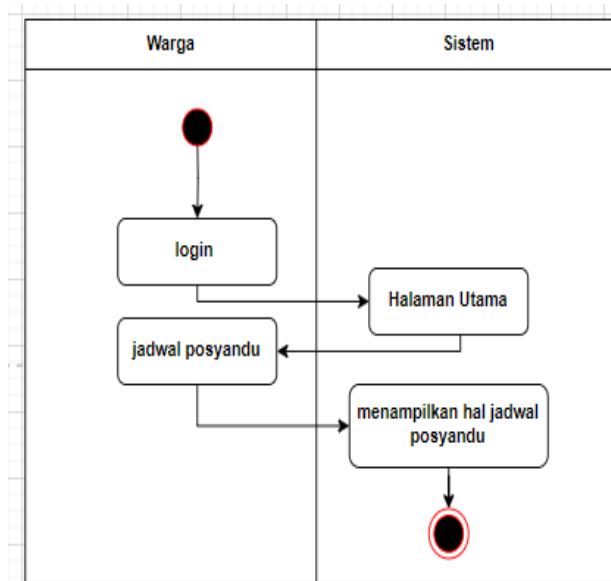
Berikut ini merupakan *activity diagram* menu layanan pengaduan pada aplikasi SIPDes.



Gambar 3. 15 activity diagram pengaduan

3.7.6 Activity Diagram Jadwal Posyandu

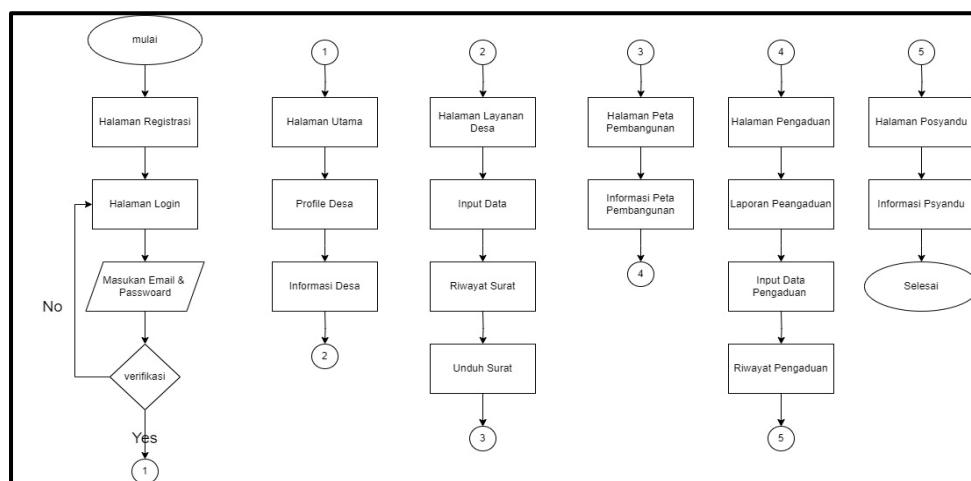
Berikut ini merupakan *Activity diagram* menu halaman jadwal posyandu pada aplikasi SIPDes.



Gambar 3. 16 activity diagram jadwal posyandu

3.8 Flowchart Diagram

Berikut ini merupakan *flowchart* diagram pada sistem aplikasi pelayanan Desa (SIPDes) pada User :

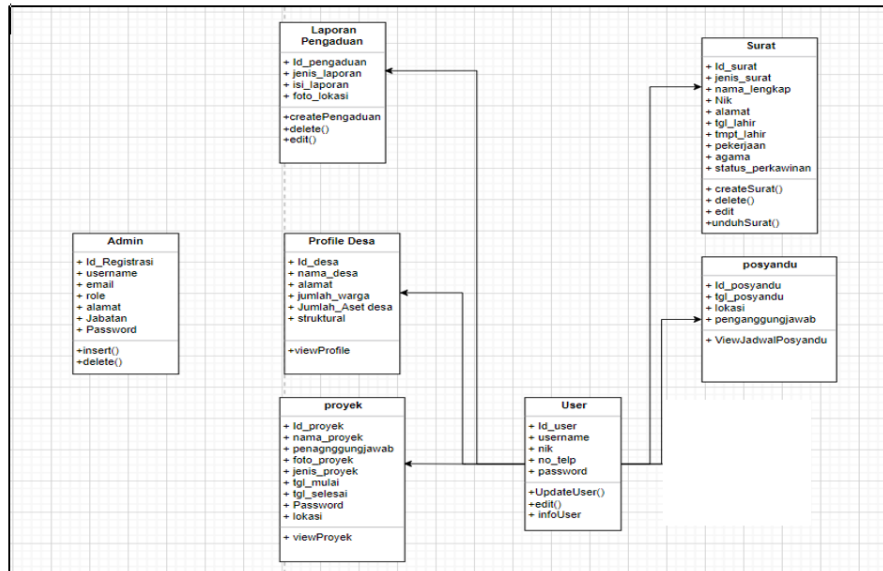


Gambar 3. 17 flowchart diagram aplikasi SIPDes

3.9 Class Diagram

Di bawah ini adalah *class diagram* aplikasi SIPDes. *Class Diagram* menggambarkan struktur kelas statis aplikasi SIPDes dan menjelaskan atribut,

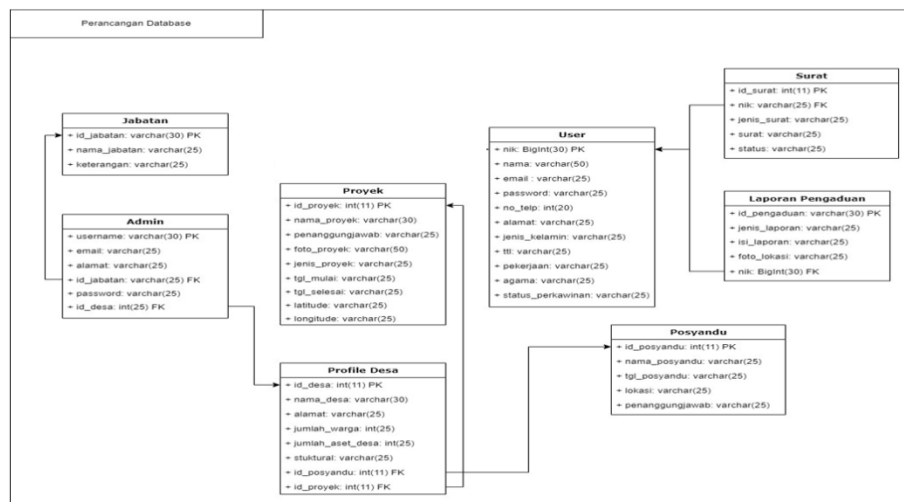
operasi, dan hubungan antar kelas. Diagram kelas ini membantu SIPD memvisualisasikan struktur kelas dalam sistem.



Gambar 3. 18 class diagram aplikasi SIPDes

3.10 Perancangan Basis Data

Perencanaan basis data pada SIPDes ini bertujuan untuk menentukan entitas yang diperlukan oleh sistem. Berikut ini merupakan model perancangan basis data pada SIPDes:



Gambar 3. 19 basis data aplikasi SIPDes

1. Kamus Data (*Data Disctionry*)

a) Tabel user

Primary key : Nik

Jumlah field : 9

Tabel 3. 2 data user

| No | Nama field | Type | Panjang data | keterangan |
|----|-------------------|---------|--------------|----------------------|
| 1 | Nama | Varchar | 50 | Nama Lengkap |
| 2 | No_telp | Varchar | 20 | Nomor Telepon |
| 3 | Password | Varchar | 25 | Kata Sandi |
| 4 | Alamat | Varchar | 25 | Alamat Lengkap |
| 5 | Jenis_kelamin | Varchar | 25 | Jenis Kelamin |
| 6 | Ttl | Varchar | 25 | Tempat Tanggal Lahir |
| 7 | Pekerjaan | Varchar | 25 | Pekerjaan |
| 8 | Agama | Varchar | 25 | Agama |
| 9 | Status perkawinan | Varchar | 25 | Status perkawinan |

b) Tabel data admin

Primary Key : Username

Forign Key : Id_desa

Jumlah field : 5

Tabel 3. 3 data admin

| No | Nama field | Type | Panjang data | keterangan |
|----|------------|---------|--------------|----------------|
| 1 | username | Varchar | 30 | Nama lengkap |
| 2 | Email | Varchar | 25 | Email |
| 3 | Alamat | Varchar | 25 | Alamat lengkap |
| 4 | Id_jabatan | Varchar | 25 | Id jabatan |
| 5 | Password | varchar | 25 | Kata sandi |

c) Tabel data profile desa

Primary Key : Id_desa

Forign Key : Id_posyandu

Jumlah Field : 3

Tabel 3. 4 data profile desa

| No | Nama field | Type | Panjang data | keterangan |
|----|------------|---------|--------------|------------------|
| 1 | Nama_desa | Varchar | 30 | Nama Desa |
| 2 | alamat | Varchar | 25 | Alamat Lengkap |
| 3 | struktural | Varchar | 25 | Struktur Jabatan |

d) Tabel data surat

Primary Key : Id_surat

Forign Key : Nik

Jumlah field : 4

Tabel 3. 5 data surat

| No | Nama field | Type | Panjang data | keterangan |
|----|-------------|---------|--------------|--------------------------|
| 1 | Nik | Varchar | 25 | Nomor induk kependudukan |
| 2 | Jenis_surat | Varchar | 25 | Jenis surat |
| 3 | Surat | Varchar | 25 | Surat |
| 4 | Status | Varchar | 25 | Status permohonan |

e) Tabel data proyek

Primary Key : Id_proyek

Forign Key : -

Jumlah field : 8

Tabel 3. 6 data proyek

| No | Nama field | Type | Panjang data | keterangan |
|----|------------------|---------|--------------|------------------------|
| 1 | Nama_proyek | Varchar | 30 | Nama proyek |
| 2 | penanggungjawab | Varchar | 25 | Penanggung jawab |
| 3 | Foto_proyek | Varchar | 50 | Foto proyek |
| 4 | Jenis_proyek | Varchar | 25 | Jenis proyek |
| 5 | Tgl_mulai | Varchar | 25 | Tanggal mulai proyek |
| 6 | Tgl_selesai | Varchar | 25 | Tanggal selesai proyek |
| 7 | <i>latitutde</i> | Varchar | 25 | Penentuan jarak lokasi |

| | | | | |
|---|------------------|---------|----|---------------------------------|
| 8 | <i>longitude</i> | Varchar | 25 | Penentuan titik kordinat lokasi |
|---|------------------|---------|----|---------------------------------|

f) Tabel data pengaduan

Primary Key : id_pengaduan

Forign Key : Nik

Jumlah field : 4

Tabel 3. 7 data pengaduan

| No | Nama field | Type | Panjang data | keterangan |
|----|---------------|---------|--------------|-----------------------|
| 1 | Id_pengaduan | Varchar | 25 | Id pengaduan |
| 2 | Jenis_laporan | Varchar | 25 | Jenis pengaduan |
| 3 | Isi_laporan | Varchar | 25 | Isi laporan |
| 4 | Foto_lokasi | Varchar | 25 | Foto lokasi pengaduan |

g) Tabel data posyandu

Primary Key : Id_posyandu

Forign Key : -

Jumlah Field : 4

Tabel 3. 8 data posyandu

| No | Nama field | Type | Panjang data | keterangan |
|----|-----------------|---------|--------------|------------------|
| 1 | Nama_posyandu | Varchar | 25 | Nama posyandu |
| 2 | Jenis_posyandu | Varchar | 25 | Tanggal posyandu |
| 3 | Lokasi | Varchar | 25 | Lokasi posyandu |
| 4 | Penanggungjawab | Varchar | 25 | Penanggung jawab |

3.11 Rancangan Sistem Perangkat Lunak

Perancangan *wireframe* adalah proses menentukan struktur dan halaman dasar dari sebuah situs web atau aplikasi. *Wireframe* adalah sebuah diagram atau skema yang menggambarkan bagaimana konten, navigasi, dan interaksi berfungsi di dalam sebuah aplikasi atau situs *website*. *Wireframe* dapat berupa gambar, desain, dan teks yang menjelaskan bagaimana halaman *website* atau aplikasi akan terlihat, terasa, dan berfungsi.

Wireframe menunjukkan gambaran awal dalam membuat suatu sistem sebelum sistem tersebut divisualkan menjadi desain *user Interface*. Pada penelitian ini peneliti mendesain *wireframe* SIPDes sebagai konsep dasar pengambilan keputusan dari data yang telah di Analisa

3.11.1 Tampilan Halaman Login

Berikut ini merupakan halaman login user pada aplikasi SIPDes.

Gambar Desa

Selamat Datang

Masuk

Email

Password

☐ Ingatkan Saya

Masuk

Belum punya akun Daftar Sekarang !

Gambar 3. 20 *wireframe* halaman *Login*

3.11.2 Tampilan Halaman Home Aplikasi SIPDes

Berikut merupakan tampilan halaman home pada aplikasi SIPDes



Gambar 3. 21 *wireframe* halaman *home*

3.11.3 Tampilan Halaman Menu Utama / *Dassbord*

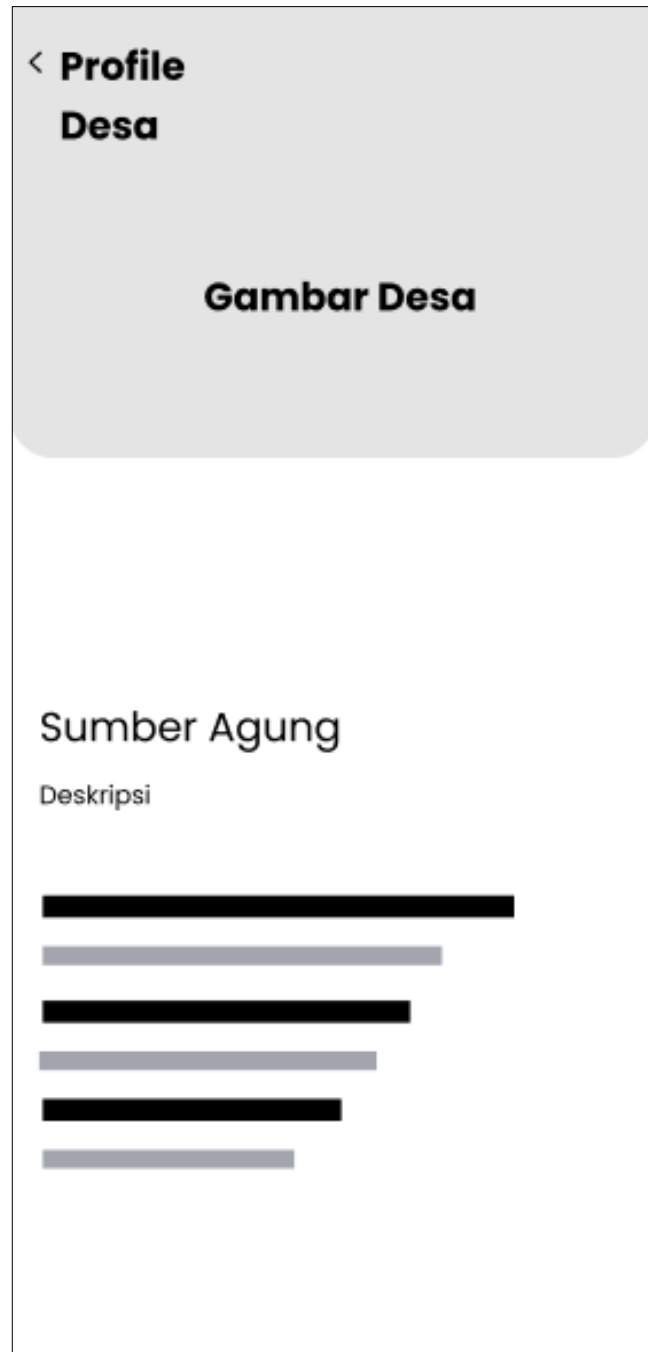
Berikut merupakan tampiln halaman utama pada aplikasi SIPDes.



Gambar 3. 22 *wireframe* halaman menu utama

3.11.4 Tampilan Halaman profile User Pada Aplikasi SIPDes

Berikut ini merupakan tampilan halaman profile desa pada aplikasi SIPDes.



Gambar 3. 23 *wireframe* halaman profile

3.11.5 Tampilan Menu Layanan Pengajuan Pemohonan surat

Berikut ini merupakan tampilan menu layanan pengajuan permohonan surat pada aplikasi SIPDes.

The wireframe shows a mobile application screen for applying for a 'Surat Keterangan Domisili' (SKD). The screen has a light gray header with a back arrow and the title 'Surat Keterangan Domisili'. Below the header, there is a greeting 'Halo...' followed by a horizontal line. The form consists of several input fields with labels: 'Pilih Jenis Surat' (a dropdown menu), 'Nama Lengkap', 'Alamat', 'Jenis Kelamin', 'Tempat Lahir', 'Tanggal Lahir', 'Agama', and 'Status Perkawinan'. At the bottom of the form is a large, rounded button labeled 'Kirim'.

<

Surat Keterangan Domisili

Halo...

Pilih Jenis Surat

Nama Lengkap

Alamat

Jenis Kelamin

Tempat Lahir

Tanggal Lahir

Agama

Status Perkawinan

Kirim

Gambar 3. 24 wireframe halaman pengajuan surat

3.11.6 Tampilan Halaman permintaan berhasil terkirim

Berikut ini merupakan tampilan pesan permintaan berhasil terkirim

<

Surat Keterangan Domisili

Halo...

Pilih Jenis Surat

Nama Lengkap

Tanggal Lahir

Agama

Status Perkawinan

Kirim

Peringatan

Berhasil Input Pengajuan

Ok

Gambar 3. 25 *wireframe* halaman permohonan berhasil dikirim

3.11.7 Tampilan Halaman Riwayat pada Aplikasi SIPDes

Berikut merupakan tampilan hal riwayat surat pada aplikasi SIPDes

The wireframe shows a mobile application interface for 'Layanan' (Services). At the top, there is a back arrow and the title 'Layanan'. Below this is a section titled 'Daftar Riwayat Surat' (Document History List). This section contains a search bar with the placeholder text 'Cari Surat' and a magnifying glass icon. Below the search bar, there are three list items, each representing a document history entry. Each entry consists of a title and a redacted area (black bar). The titles are 'Formulir Permohonan KTP', 'Formulir Permohonan tidak mampu', and 'Formulir Permohonan Domisili'. At the bottom of the page, there is a large, rounded button labeled 'Kirim' (Send).

Gambar 3. 26 *wireframe* halaman riwayat surat

3.11.8 Tampilan Halaman pembangunan

Berikut merupakan tampilan halaman peta pembangunan pada aplikasi SIPDes



Gambar 3. 27 *wireframe* halaman pembangunan

3.11.9 Tampilan Halaman pengaduan

Berikut merupakan halaman pengaduan pada aplikasi SIPDes.

The wireframe shows two side-by-side panels representing the complaint page in the SIPDes application.

Left Panel (Form):

- Header: A grey bar with a back arrow and the text "Pengaduan".
- Form Fields:
 - Judul Pengaduan**: A text input field.
 - Gambar Pengaduan**: A placeholder for an image, represented by a rectangle with a diagonal 'X'.
 - Deskripsi**: A text input field.
 - Jenis Kelamin**: A dropdown menu.
 - Nomor Hanphone**: A text input field.
 - Alamat Lengkap**: A text input field.
 - Nama Lengkap**: A text input field.
- Button: A grey button labeled "Kirim" at the bottom.

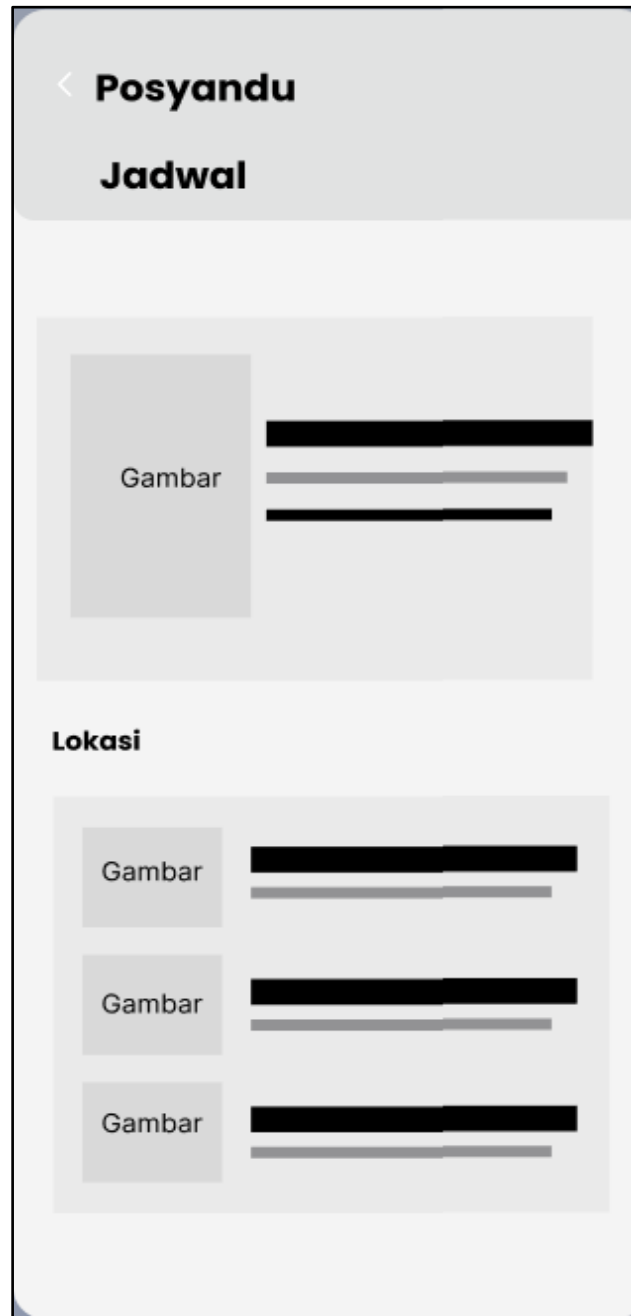
Right Panel (Report):

- Header: A grey bar with a back arrow and the text "Pengaduan".
- Image: A placeholder for a profile picture or logo, represented by a rectangle with a diagonal 'X'.
- Section: **Laporan Pengaduan**, followed by five horizontal grey bars representing lines of text.

Gambar 3. 28 *wireframe* halaman pengaduan

3.11.10 Tampilan menu halaman jadwal posyandu

Berikut ini merupakan tampilan halaman jadwal posyandu pada aplikasi SIPDes



Gambar 3. 29 *wireframe* halaman posyandu

3.12 Jadwal Penelitian

Berikut ini merupakan jadwal penelitian yang telah dilakukan penulis dalam perancang aplikasi SIPDes.

Tabel 3. 9 jadwal penelitian

| No | Kegiatan | Des | | | | Jan | | | | Feb | | | | Mar | | | | April | | | | Mei | | | | jun | | | | Jul | | | | agt | | | |
|----|----------------------------|-----|---|---|---|-----|---|---|---|-----|---|---|---|-----|---|---|---|-------|---|---|---|-----|---|---|---|-----|---|---|---|-----|---|---|---|-----|---|---|---|
| | | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | wawancara | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Identifikasi Masalah | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | Analisa Data & Studi Kasus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | Perancangan UML Diagram | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | Konsep & Desain Sistem | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | Evaluasi Desain Sistem | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | pembuatan Program | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | Menyusun Laporan | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | Seminar Hasil Akhir | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

BAB IV

HASIL DAN PEMBAHASAN

4.1 HASIL

Hasil perancangan dan pengembangan sistem yang telah dilakukan oleh penulis pada Aplikasi SIPDes dengan menggunakan media *visual studio code* untuk membuat aplikasi *mobile* pelayanan publik Desa pada Desa Sumber Agung, Lampung Selatan. Uji coba sistem dari aspek pengembangan menjadi suatu keharusan untuk mengenali dan mengurangi potensi masalah yang mungkin timbul, serta untuk mengetahui apakah fungsi sistem mampu berjalan dengan efisien pada lingkungan sebelum akhirnya diperkenalkan kepada pengguna.

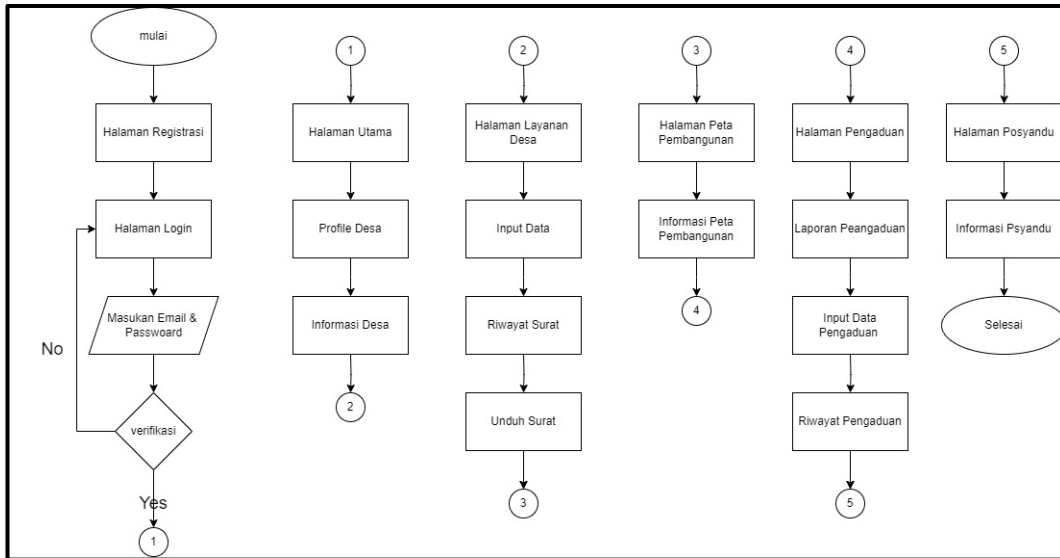
Dalam hasil implementasi ini, penulis berhasil mengintegrasikan semua elemen desain yang telah dipersiapkan sebelumnya ke dalam *source code*. Hal ini mencakup pembuatan tampilan halaman login, halaman registrasi, dan berbagai halaman lainnya. Pengguna yang mengakses aplikasi akan pertama kali disambut dengan halaman login. Di sini, mereka diharuskan untuk memasukkan alamat email dan kata sandi yang telah terdaftar sebelumnya. Jika proses login berhasil, pengguna akan diarahkan ke halaman *dashboard* pengguna.

Selanjutnya, untuk pengguna yang belum memiliki akun terdaftar, mereka dapat melakukannya melalui halaman registrasi. Di halaman ini, data diri yang diperlukan dapat diisi untuk membuat akun baru. Setelah berhasil mendaftar, pengguna akan kembali ke halaman login untuk masuk ke dalam akun yang baru dibuat.

Penelitian ini menerapkan model SDLC di dalam proses pengembangan sistem pelayanan Desa berbasis *android* untuk pengolahan layanan publik di desa Sumber Agung. Tahapan-tahapan yang dilakukan pada penelitian ini yaitu analisis kebutuhan, desain sistem, implementasi, dan pengujian.

4.1.1 Desain Sistem / Perancangan Sistem

Berdasarkan identifikasi kebutuhan, perancangan sistem melibatkan dua aktor yaitu admin dan warga. Kedua aktor tersebut berinteraksi di dalam sistem sebagai *eksternal interactor* dalam diagram *flowchart* seperti yang diperlihatkan. Berikut merupakan bentuk *flowchart* dari perancangan sistem yang telah dibuat :



Gambar 4. 1 Gambar Flowchart Perancangan aplikasi SIPDes

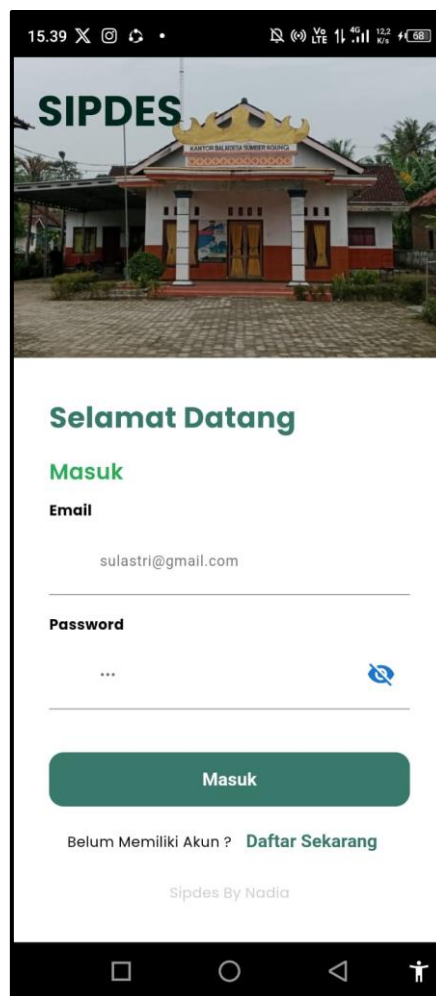
4.1.2 Implementasi Sistem

Hasil akhir dari tahap implementasi ini adalah tampilan visual yang menggambarkan keseluruhan aplikasi SIPDes yang telah diimplementasikan dalam *source code*. Proses ini tidak hanya melibatkan aplikasi *Visual Studio Code* sebagai alat utama, tetapi juga memanfaatkan *framework quasar* dalam menciptakan antarmuka yang interaktif, efisien, dan responsive.

4.2.1.1 Tampilan Halaman Login dan Registrasi User

1. Login

Saat awal memasuki aplikasi, pengguna akan ditampilkan halaman login dan perlu untuk mengisi email dan password yang telah didaftarkan. Pada tampilan ini, pengguna dapat memasukkan alamat email dan kata sandi yang telah terdaftar sebelumnya. Jika alamat email dan kata sandi yang dimasukkan cocok, maka pengguna akan diizinkan untuk masuk ke halaman *dasbord* aplikasi. Namun, jika alamat email dan kata sandi yang dimasukkan tidak sesuai dengan yang telah terdaftar sebelumnya, pengguna akan diminta untuk melakukan pendaftaran terlebih dahulu melalui menu registrasi.



Gambar 4. 2 Halaman Login

2. Registrasi

Pada tampilan registrasi, jika pengguna belum memiliki akun terdaftar untuk melakukan login, pengguna dapat melakukan pendaftaran terlebih dahulu melalui halaman ini. Pada halaman ini, pengguna diminta untuk mengisi data pribadi, termasuk nama lengkap, alamat email, Nik, nomor telepon, dan *password*. Jika pendaftaran berhasil, data pengguna akan disimpan dan pengguna akan diarahkan ke halaman masuk (login).

The image displays two screenshots of a mobile application interface for SIPDES. The left screenshot shows the 'Buat Akun Baru' (Create New Account) registration form. It features a header image of a building and a title 'Buat Akun Baru'. Below the title is a green message: 'Mari buat akun sipdes untuk pembuatan surat pelayanan publik dengan satu kali klik !!!'. The form includes five input fields: 'NIK' (with the value 32564848454684), 'NamaLengkap' (with the value sulastri), 'Email' (with the value sulastri@gmail.com), 'No. Telp' (with the value 085464848767), and 'Password' (with a masked value ***). A blue eye icon is visible next to the password field. The right screenshot shows the 'Selamat Datang' (Welcome) login screen. It features the same header image and title 'SIPDES'. Below the title is a green message: 'Selamat Datang'. The screen includes two input fields: 'Email' (with the placeholder text 'Masukan Email') and 'Password' (with the placeholder text 'Masukkan Password' and a blue eye icon). A green 'Masuk' (Login) button is located below the password field. At the bottom, there is a link 'Daftar Sekarang' (Register Now) and the text 'Sipdes By Nadia'.

Gambar 4. 3 Halaman Registrasi

4.2.1.2 Tampilan Halaman User

Berikut ini merupakan tampilan halaman user yang ditampilkan setelah melakukan login dengan menggunakan akun user.

1. Halaman Utama User

Setelah User berhasil login dan masuk ke halaman utama aplikasi, pengguna dapat melihat menu pada aplikasi yang telah tersedia. Pengguna juga dapat mengakses dan menggunakan semua fitur layanan yang telah tersedia. Sebagai contoh di dalam gambar terlihat ada 6 fitur pada aplikasi yang dapat digunakan.



Gambar 4. 4 Tampilan Halaman Utama

2. Tampilan Halaman Profile Desa

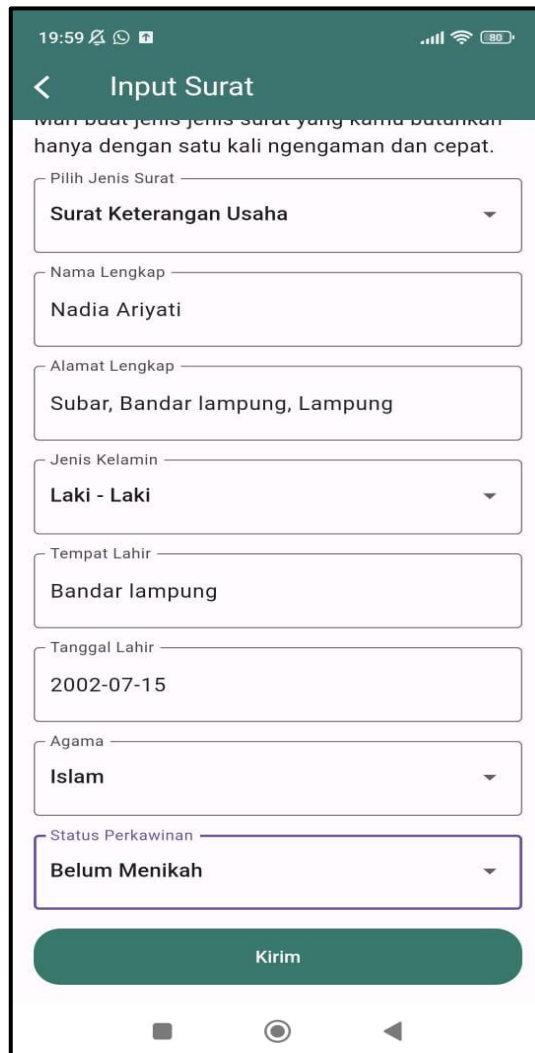
Pada halaman ini user dapat mengakses dan melihat berbagai informasi secara detail mengenai sejarah Desa Sumber Agung seperti, tahun berdirinya desa Sumber Agung, Jumlah Penduduk, serta destinasi wisata yang terletak pada Desa Sumber Agung.



Gambar 4. 5 Tampilan Profile Desa

3. Tampilan Halaman Layanan Adminitrasi Desa

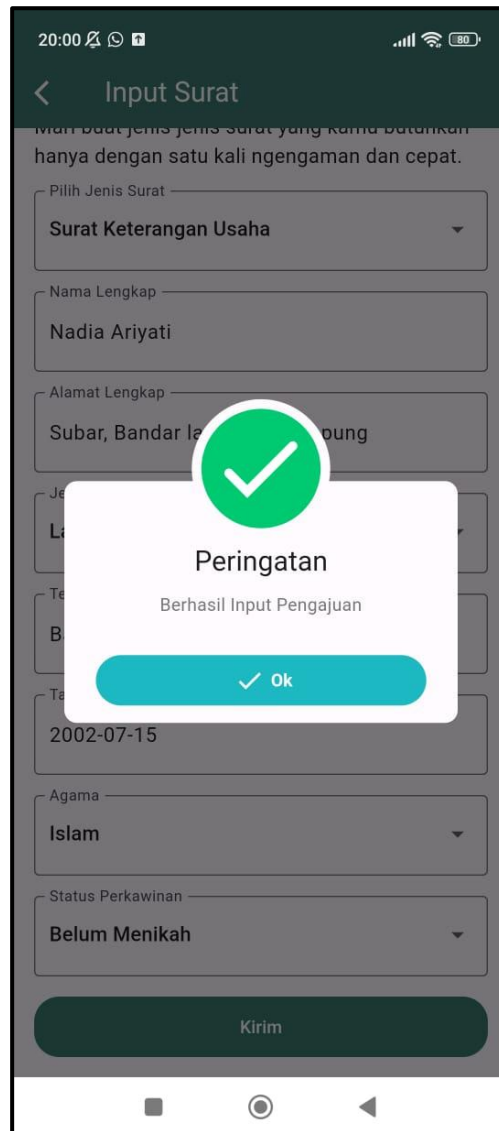
Pada halaman ini user dapat megakses dan mengajukan layanan adminitrasi seperti, surat keterangan tidak mampu, surat keterangan usaha, surat keterangan domisili. User dapat memilih salah satu pengajuan surat sesuai dengan yang dibutuhkan. Kemudian user dapat menginput biodata yang dibutuhkan dalam pengajuan layanan surat seperti, jenis surat, nama lengkap, alamat, jenis kelamin, tempat lahir, tanggal lahir, agama, status perkawinan, pekerjaan. Setelah user selesai mengisi biodata kemudian klik kirim.



Gambar 4. 6 Tampilan Layanan Adminitrasi

4. Tampilan Halaman Berhasil Dikirim

Setelah melakukan pengiriman data, pada tampilan halaman ini user dapat melihat pemberitahuan permintaan pengajuan berhasil terkirim ke desa, dan user dapat menunggu proses pengerjaan surat.



Gambar 4. 7 Tampilan Halaman Permintaan Terkirim

5. Tampilan Halaman Riwayat Surat

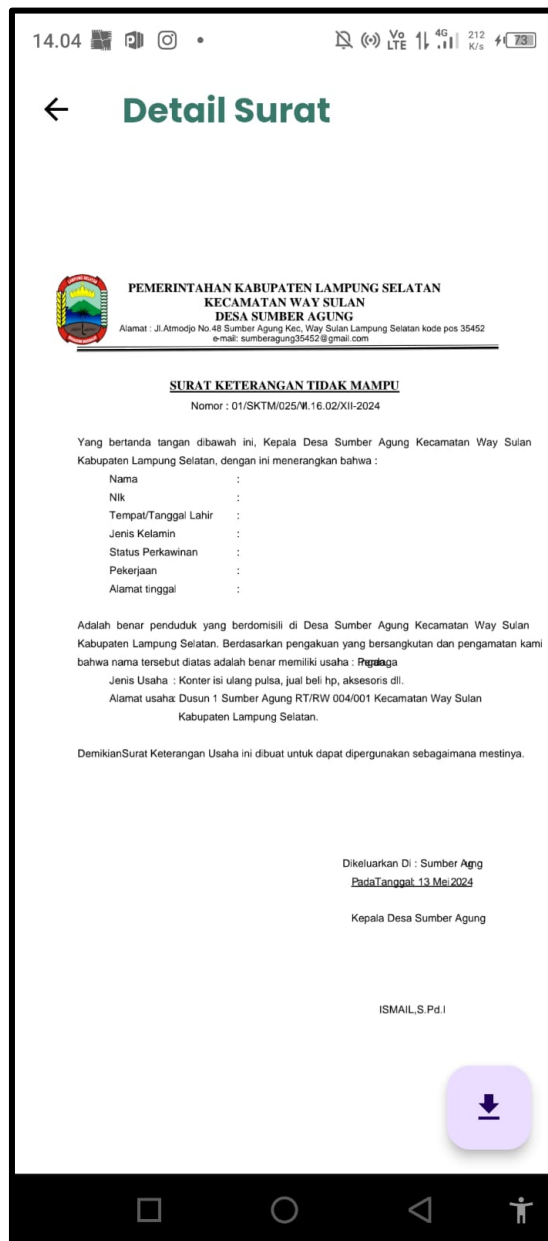
Setelah berhasil input pengajuan, Pada tampilan halaman ini merupakan riwayat surat, dimana user dapat melihat riwayat surat dengan status surat sudah berhasil atau masih dalam proses .



Gambar 4. 8 Tampilan Halaman Riwayat Surat

6. Tampilan Halaman Unduh Surat

Setelah status pada halaman riwayat surat telah berhasil, Pada tampilan halaman ini user dapat mengunduh surat yang telah disetujui oleh petugas desa. Kemudian user dapat melihat surat yang telah di unduh pada folder download pada perangkat user.



Gambar 4. 9 Tampilan Halaman Unduh Surat

7. Tampilan Halaman Peta Pembangunan

Pada tampilan halaman ini user dapat mengakses dan melihat daftar program kerja peta pembangunan yang sedang dibangun di Desa Dumber Agung, Lampung Selatan.



Gambar 4. 10 Tampilan Halaman Peta Bangunan

8. Tampilan Halaman Pengaduan

Pada tampilan halaman ini user dapat mengakses laporan pengaduan dan riwayat pengaduan dimana tampilan ini merupakan tampilan dari halaman pengaduan. dan melaporkan pengaduan kepada Desa dengan mengunggah bukti foto lokasi dan memberikan keterangan laporan pengaduan terkait dengan permasalahan yang diadukan.



Gambar 4. 11 Tampilan Halaman Pengaduan

9. tampilan Halaman Penginputan data pengaduan

Pada tampilan halaman ini user dapat menginput data pengaduan yang akan dilaporkan, dimana tampilan ini merupakan tampilan dari halaman pengaduan. Setelah melakukan penginputan klik tombol kirim

The image displays two screenshots of a mobile application interface for reporting a community complaint. Both screenshots show a screen titled 'Pengaduan' (Complaint) with a back arrow on the left.

Left Screenshot (Form Input):

- Form Pengaduan Masyarakat**
- Judul Pengaduan**: Input field containing 'pungli sekolah'.
- Gambar Pengaduan**: Input field containing a placeholder path and a camera icon.
- Deskripsi**: Input field containing 'ada pungli di sekolah SDN Al azhar'.
- No Handphone**: Input field containing '085663356'.
- Alamat Lengkap**: Input field containing 'SDN Al Azhar Lampung'.

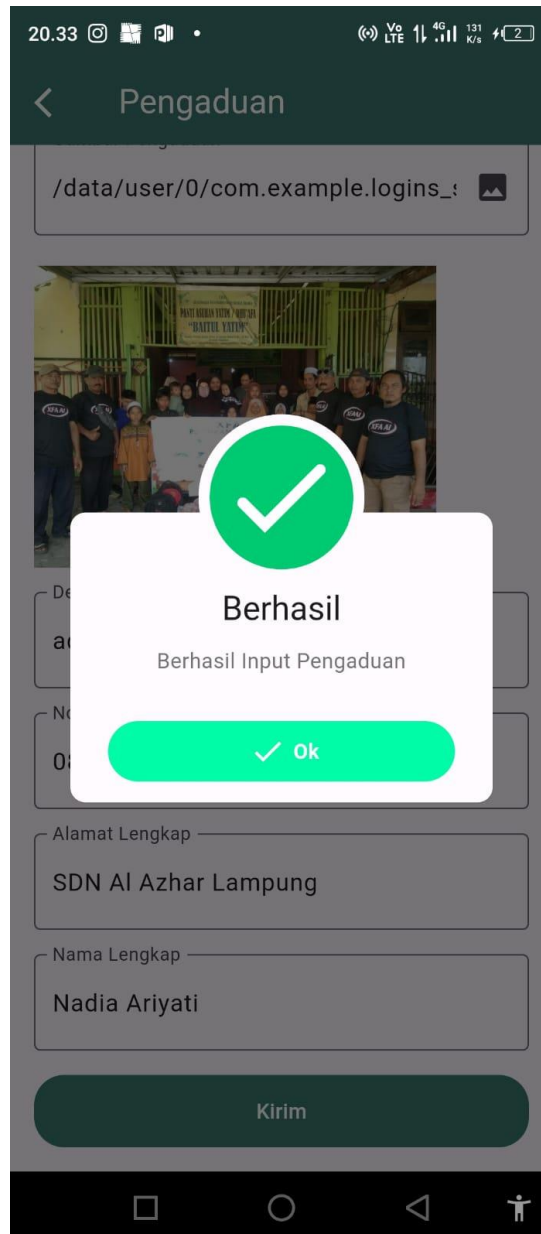
Right Screenshot (Form Preview):

- Gambar Pengaduan**: Input field containing a placeholder path and a camera icon.
- Deskripsi**: Input field containing 'ada pungli di sekolah SDN Al azhar'.
- No Handphone**: Input field containing '085663356'.
- Alamat Lengkap**: Input field containing 'SDN Al Azhar Lampung'.
- Nama Lengkap**: Input field containing 'Nadia Ariyati'.
- Kirim**: A green button at the bottom right.

Gambar 4. 12 Tampilan Hal Input Pengaduan

10. Tampilan Halaman pengaduan berhasil dikirim

Setelah melakukan penginputan data pengaduan dan telah melakukan pengiriman maka sistem akan menampilkan peringatan berhasil input pengaduan, dimana pada halaman ini merupakan tampilan dari halaman penduan yang telah berhasil dikirim.



Gambar 4. 13 Tampilan Hal Pengaduan Berhasil Dikirim

11. Tampilan Halaman Riwayat Pengaduan

User dapat melihat riwayat pengaduan yang sudah pernah dikirim sebelumnya, dimana tampilan halaman ini merupakan tampilan dari halaman riwayat pengaduan.



Gambar 4. 14 Tamplan Hal Riwayat Pengaduan

12. Tampilan Halaman Posyandu

Pada tampilan halaman ini user dapat mengakses dan melihat jadwal poyandu yang telah ditentukan oleh Pemerintahan Desa Sumber Agung, Lampung Selatan.



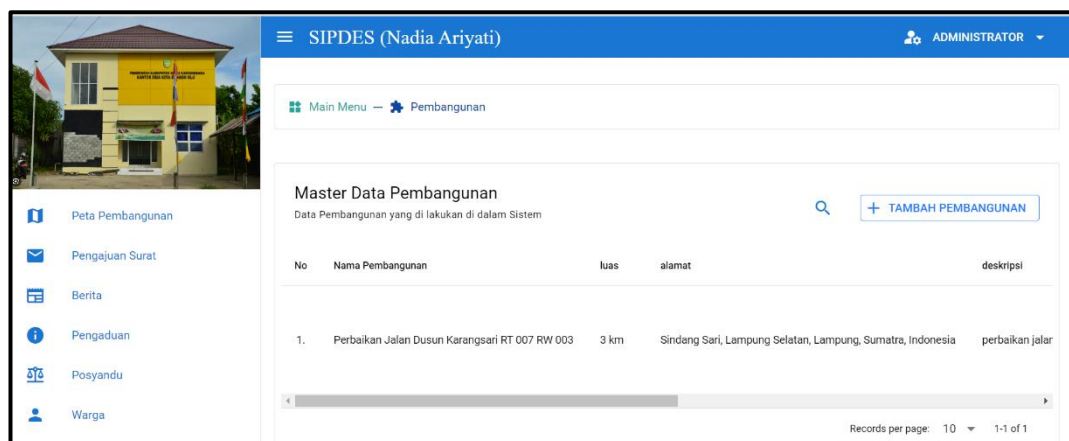
Gambar 4. 15 Tampilan Halaman Posyandu

4.2.1.3 Tampilan Halaman Admin

Berikut adalah tampilan halaman admin yang ditampilkan setelah melakukan login dengan menggunakan akun admin.

1. Dashboard Admin

Saat awal memasuki *website*, pengguna akan ditampilkan halaman *login* dan perlu untuk mengisi *email* dan *password* yang telah didaftarkan. Pengguna dapat memasukkan alamat *email* dan kata sandi yang telah terdaftar sebelumnya. Setelah admin berhasil login dan masuk ke halaman dashboard website, pengguna dapat melihat data jumlah user, dapat menginput berita desa, menambah data pembangunan, input data posyandu, input surat dan melihat laporan pengaduan.



Gambar 4. 16 Halaman Admin

4.2 PEMBAHASAN

Setelah desain diimplementasikan, pengguna akan melihat tampilan yang telah dirancang untuk memenuhi kebutuhan pengguna. Ketika pengguna pertama kali mengunjungi aplikasi, pengguna akan dihadapkan dengan tampilan halaman login. Di sini, pengguna diminta untuk memasukkan alamat email dan password yang telah pengguna daftarkan sebelumnya. Jika detail yang dimasukkan sesuai, pengguna akan diarahkan ke halaman *dashboard* sesuai dengan perannya. Jika seorang pengguna baru yang belum memiliki akun mengunjungi aplikasi, pengguna akan dapat memilih halaman registrasi. Di halaman ini, pengguna diminta untuk mengisi informasi pribadi seperti nik, nama lengkap, email, nomor telepon, dan password. Setelah data tersebut diisi dengan benar, pengguna akan dapat membuat akun dan diarahkan kembali ke halaman login.

Setelah login, *user* akan diarahkan ke halaman *dashboard*. Di sini, user bisa melihat layanan desa yang tersedia. User juga memiliki akses ke fitur seperti profile desa, layanan desa, peta pembangunan, dan jadwal posyandu. Halaman ini memungkinkan pengguna untuk melihat dan mengelola layanan yang telah disediakan. Pada halaman ini, pengguna dapat melihat secara lengkap tentang informasi desa. Selain itu, di halaman ini, pengguna juga memiliki akses untuk mengajukan permohonan surat serta laporan pengaduan. Halaman ini memungkinkan pengguna untuk melihat dan mengelola data pribadi.

Setelah berhasil mengimplementasikan tampilan depan (*front-end*) aplikasi SIPDes sesuai desain yang telah dirancang, tahap selanjutnya adalah melakukan pengujian untuk memastikan bahwa aplikasi berjalan dengan baik dan sesuai dengan kebutuhan yang telah ditetapkan. Pengujian ini bertujuan untuk mengidentifikasi potensi masalah atau kesalahan yang mungkin muncul selama penggunaan perangkat lunak.

Pentingnya melakukan uji coba sistem program, untuk mengetahui sejauh mana sistem dapat berjalan melakukan fungsinya dengan benar sesuai dengan

perintah program yang dibuat. Dalam pengujian ini, dua jenis pengujian utama yang dilakukan yaitu pengujian *black box* dan pengujian *white box*.

4.2.1 Pengujian *Black Box*

Pengujian *Black box* digunakan untuk memverifikasi apakah seluruh aspek fungsi perangkat lunak beroperasi dengan tepat sesuai dengan kebutuhan fungsional yang telah ditetapkan. Dalam proses ini, perangkat lunak diuji tanpa memerhatikan implementasi internalnya. sebaliknya, fokus diberikan pada pengujian input dan verifikasi bahwa output yang dihasilkan sejalan dengan desain yang telah ditetapkan. Berikut merupakan pengujian *black box* yang telah dilakukan oleh salah satu ahli pakar di bidangnya.

1. Pengujian blackbox registrasi dan login

tabel 4. 1 pengujian blackbox registrasi dan login

| No | Skenario Kondisi | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|----------------------------------|---|--|--------|
| 1 | Melakukan Proses Login | Dapat melakukan proses autentikasi pada form login dan diarahkan ke halaman menu utama. | Berhasil melakukan proses autentikasi dan user berhasil diarahkan halaman utama. | Sukses |
| 2 | User melakukan proses registrasi | Sistem menyimpan ke database dan memberi pesan Pendaftaran Berhasil | Data registrasi sementara user berhasil disimpan ke dalam database dan memberi pesan pendaftaran berhasil. | Sukses |

2. Pengujian blackbox halaman utama

tabel 4. 2 pengujian blackbox halaman utama

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|-----------|---|--|--------|
| 1 | Tampilan | menu utama layanan dapat ditampilkan beserta informasi. | Berhasil ditampilkan dan user dapat mengakses fitur layanan yang tersedia. | Sukses |

3. Pengujian Blackbox Halaman Profile Desa

tabel 4. 3 pengujian blackbox halaman profile desa

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|-----------|---|--|--------|
| 1 | Tampilan | Informasi mengenai profile desa yang sudah di buat dapat ditampilkan. | Berhasil ditampilkan dan user dapat mengakses dan melihat mengenai latar belakang profile desa | Sukses |

4. Pengujian Blackbox Halaman Layanan Desa

tabel 4. 5 pengujian blackbox halaman layanan desa

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|---|--|--|--------|
| 1 | Tampilan penginputan data pengajuan surat | Dapat menampilkan tampilan penginputan data. | Berhasil ditampilkan dan user dapat menginput data pengajuan surat yang dibutuhkan | Sukses |

| | | | | |
|---|---------------------------|---|---|--------|
| 2 | Tampilan berhasil dikirim | Dapat menampilkan tampilan pengajuan berhasil terkirim | Setelah melakukan pengiriman pengajuan, tampilan terkirim berhasil ditampilkan. | Sukses |
| 3 | Tampilan Riwayat Surat | Dapat melihat tampilan riwayat surat yang telah di ajukan. | Berhasil ditampilkan, user dapat melihat status riwayat surat. | Sukses |
| 4 | Tampilan unduh surat | Dapat menampilkan tampilan unduh surat, dan dapat melakukan pengunduhan surat | Berhasil ditampilkan, user dapat mendownload surat yang telah berhasil. | sukses |

5. Pengujian halaman peta pembangunan

tabel 4. 7 pengujian blackbox hal peta pembangunan

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|-----------|--|--|--------|
| 1 | Tampilan | Informasi mengenai peta pembangunan dapat ditampilkan. | Berhasil ditampilkan, user dapat melihat informasi pembangunan secara akurat | Sukses |

6. Pengujian Halaman Pengaduan

tabel 4. 9 pengujian blackbox hal pengaduan

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|-------------------------------------|--|--|--------|
| 1 | Tampilan penginputan data pengaduan | Dapat menampilkan tampilan penginputan data pengaduan. | Berhasil ditampilkan dan user dapat menginput data pengaduan sesuai dengan yang dilaporkan. | Sukses |
| 2 | Tampilan berhasil dikirim | Dapat menampilkan tampilan berhasil input pengaduan. | Setelah melakukan pengiriman pengaduan, tampilan berhasil input pengaduan berhasil ditampilkan | Sukses |
| 3 | Tampilan Riwayat Pengaduan | Dapat melihat tampilan riwayat pengaduan yang telah di ajukan. | Berhasil ditampilkan, user dapat melihat tampilan riwayat pengaduan. | Sukses |

7. Pengujian Halaman Posyandu

tabel 4. 11 pengujian blackbox hal posyandu

| No | Pengujian | Hasil yang diharapkan | Hasil yang didapat | Status |
|----|-----------|---|---|--------|
| 1 | Tampilan | Informasi mengenai jadwal posyandu dapat ditampilkan. | Berhasil ditampilkan, user dapat melihat informasi jadwal posyandu. | Sukses |

4.2.2 Pengujian WhiteBox

Pengujian *white box* digunakan untuk menguji aplikasi atau perangkat lunak dengan menganalisis aspek yang terkait dengan logika dan struktur kode. Dalam metode pengujian *white box*, penulis menggunakan teknik basis *path testing*. Teknik ini melibatkan beberapa langkah, termasuk *flow graph*, *cyclomatic complexity*, dan *independent paths*. *cyclomatic complexity* merupakan Metrik yang mengukur kompleksitas kode dengan rumus $V(G)=E-N+2P$.

Keterangan: $V(G) = Cyclomatic Complexity$ $E = Path / jalur$,

$P = Predicate$

$N = Node$

Keterangan:

$N = Node$ (Lingkaran yang menggambarkan aktivitas dalam flowgraph)

$E = Edge$ (Garis panah yang menghubungkan node dalam flowgraph)

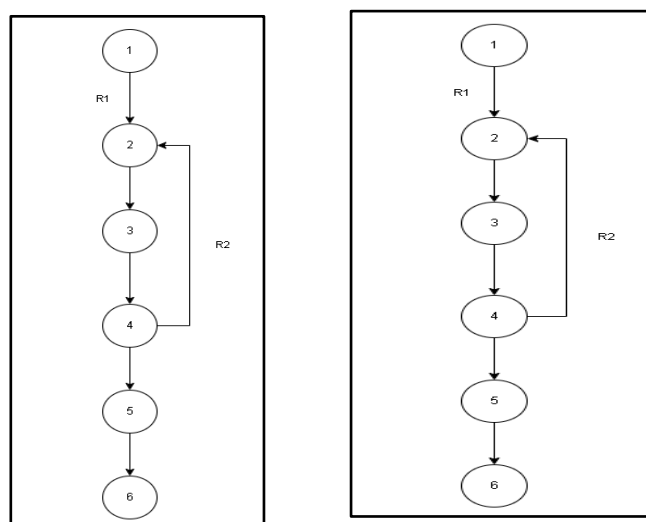
$P = Predicate$ (node yang memiliki cabang didalam flowgraph)

$R = Region$ (Area yang terdapat didalam flowgraph)

4.2.2.1 White Box Testing Login dan Registrasi

Berikut ini merupakan bentuk flowgraph dari Login register user yang digunakan sesuai alur flowchart diagram:

a) Flow Graph Login & Registrasi



Gambar 4. 17 Flowgraph Login & Registrasi

Berdasarkan flowgraph Registrasi diatas diketahui bahwa jumlah :

$$E \text{ (Edge)} = 6$$

$$N \text{ (Node)} = 6$$

$$P \text{ (Predicate)} = 1$$

$$R \text{ (Region)} = 2$$

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 12 Cyclomatic Complexity Login & Registrasi

| | | |
|--------------------|----------------|------------|
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
| $V(G) = 6 - 6 + 2$ | $V(G) = 1 + 1$ | $V(G) = 2$ |
| $V(G) = 2$ | $V(G) = 2$ | |

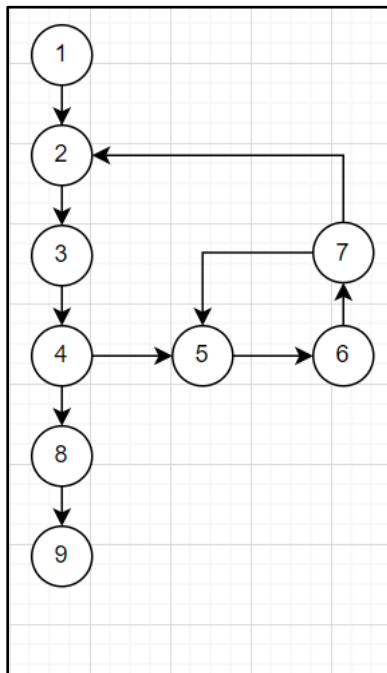
Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk *halaman utama user* yaitu dengan 3 jalur *independent path* diantaranya yaitu :

a) Jalur 1 = 1 2 3 4 5 6

b) Jalur 2 = 1 2 3 4 2 3 4 5 6

4.2.2.2 White Box Testing Halaman Utama

Berikut ini merupakan bentuk flowgraph dari Halaman Utama user yang digunakan sesuai alur *flowchart diagram*.



Gambar 4. 18 Flowgraph Halaman Utama

Berdasarkan *flowgrap* diatas diketahui bahwa jumlah :

$$E \text{ (Edge)} = 10$$

$$N \text{ (Node)} = 9$$

$$P \text{ (Predicate)} = 2$$

$$R \text{ (Region)} = 2$$

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 13 Cyclomatic Complexity

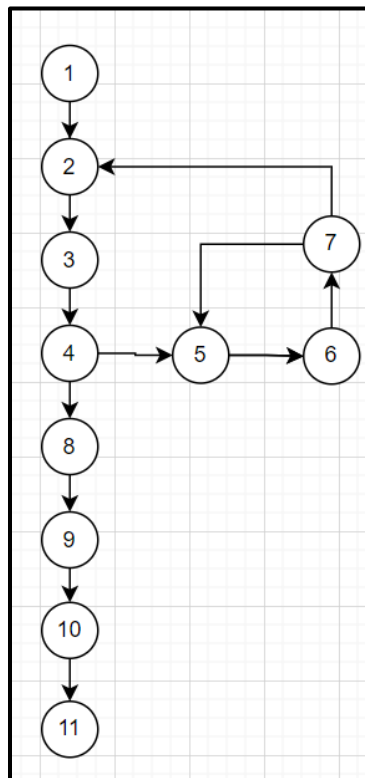
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
|---------------------|----------------|------------|
| $V(G) = 10 - 9 + 2$ | $V(G) = 2 + 1$ | $V(G) = 2$ |
| $V(G) = 3$ | $V(G) = 3$ | |

Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk *halaman utama user* yaitu dengan 3 jalur *independedent path* diantaranya yaitu :

- a) Jalur 1 = 1 2 3 4 8 9
- b) Jalur 2 = 1 2 3 4 5 6 7 2 3 4 8 9
- c) Jalur 3 = 1 2 3 4 5 6 7 6 7 2 3 4 8 9

4.2.2.3 Testing White Box Halaman Layanan Desa

Berikut ini merupakan bentuk flowgraph dari Halaman Layanan user yang digunakan sesuai alur *flowchart diagram*.



Gambar 4. 19 Flowgraph Halaman Layanan Desa

Berdasarkan *flowwgrap* diatas diketahui bahwa jumlah :

E (Edge) = 12

N (Node) = 11

$$P \text{ (Predicate)} = 2$$

$$R \text{ (Region)} = 2$$

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 14 *Cyclomatic Complexity*

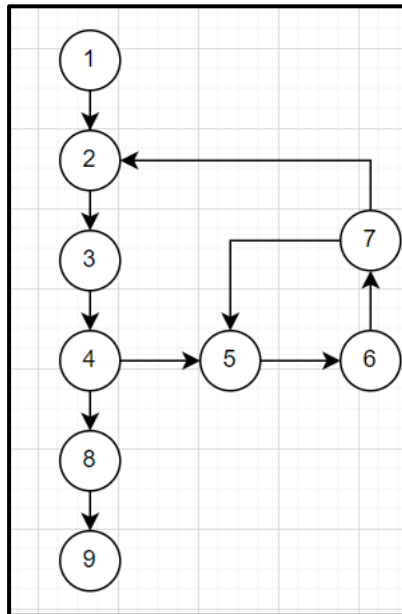
| | | |
|----------------------|----------------|------------|
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
| $V(G) = 12 - 11 + 2$ | $V(G) = 2 + 1$ | $V(G) = 2$ |
| $V(G) = 3$ | $V(G) = 3$ | |

Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk halaman layanan desa yaitu dengan 2 jalur *independedent path* diantaranya yaitu :

- a) Jalur 1 = 1 2 3 4 8 9 10 11
- b) Jalur 2 = 1 2 3 4 5 6 7 2 3 4 8 9 10 11
- c) Jalur 3 = 1 2 3 4 5 6 7 5 6 7 2 3 4 8 9 10 11

4.2.2.4 Testing White Box Halaman Peta Pembangunan

Berikut ini merupakan bentuk flowgraph dari Halaman peta pembangunan user yang digunakan sesuai alur *flowchart diagram*.



Gambar 4. 20 Flowgraph Peta Pembangunan

Berdasarkan *flowwgrap* diatas diketahui bahwa jumlah :

E (*Edge*) = 10

N (*Node*) = 9

P (*Predicate*) = 2

R (*Region*) = 2

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 15 *Cyclomatic Complexity* Halaman Peta Pembangunan

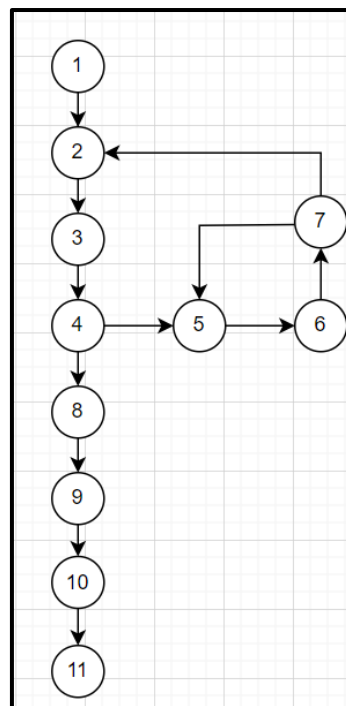
| | | |
|---------------------|----------------|------------|
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
| $V(G) = 10 - 9 + 2$ | $V(G) = 2 + 1$ | $V(G) = 2$ |
| $V(G) = 3$ | $V(G) = 3$ | |

Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk halaman layanan desa yaitu dengan 2 jalur *independedent path* diantaranya yaitu :

- a) Jalur 1 = 1 2 3 4 8 9
- b) Jalur 2 = 1 2 3 4 5 6 7 2 3 4 8 9
- c) Jalur 3 = 1 2 3 4 5 6 7 5 6 7 2 3 4 8 9

4.2.2.5 Testing White Box Halaman Pengaduan

Berikut ini merupakan bentuk flowgraph dari Halaman pengaduan user yang digunakan sesuai alur *flowchart diagram*.



Gambar 4. 21 Flowgraph Halaman Pengaduan

Berdasarkan *florwgrap* diatas diketahui bahwa jumlah :

E (Edge) = 12

N (Node) = 11

P (Predicate) = 2

$$R \text{ (Region)} = 2$$

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 16 Cyclomatic Complexity Halaman Pengaduan

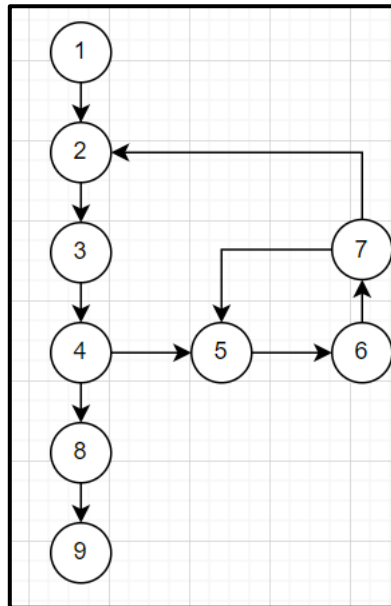
| | | |
|----------------------|----------------|------------|
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
| $V(G) = 12 - 11 + 2$ | $V(G) = 2 + 1$ | $V(G) = 2$ |
| $V(G) = 3$ | $V(G) = 3$ | |

Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk halaman layanan desa yaitu dengan 2 jalur *independedent path* diantaranya yaitu :

- a) Jalur 1 = 1 2 3 4 8 9 10 11
- b) Jalur 2 = 1 2 3 4 5 6 7 2 3 4 8 9 10 11
- c) Jalur 3 = 1 2 3 4 5 6 7 5 6 7 2 3 4 8 9 10 11

4.2.2.6 Testing White Box Halaman Posyandu

Berikut ini merupakan bentuk flowgraph dari Halaman posyandu user yang digunakan sesuai alur *flowchart diagram*.



Gambar 4. 22 Flowgraph Halaman Posyandu

Berdasarkan *flowgrap* diatas diketahui bahwa jumlah :

$$E \text{ (Edge)} = 10$$

$$N \text{ (Node)} = 9$$

$$P \text{ (Predicate)} = 2$$

$$R \text{ (Region)} = 2$$

Jika dihitung dengan menggunakan rumus *Cyclomatic Complexity* akan menghasilkan perhitungan seperti berikut :

tabel 4. 17 Cyclomatic Complexity Halaman Posyandu

| | | |
|---------------------|----------------|------------|
| $V(G) = E - N + 2$ | $V(G) = P + 1$ | $V(G) = R$ |
| $V(G) = 10 - 9 + 2$ | $V(G) = 2 + 1$ | $V(G) = 2$ |
| $V(G) = 3$ | $V(G) = 3$ | |

Dari hasil perhitungan berdasarkan *cyclomatic complexity* pada tabel diatas didapatkan hasil *independent path* untuk halaman layanan desa yaitu dengan 2 jalur *independedent path* diantaranya yaitu :

- a) Jalur 1 = 1 2 3 4 8 9
- b) Jalur 2 = 1 2 3 4 5 6 7 2 3 4 8 9
- c) Jalur 3 = 1 2 3 4 5 6 7 5 6 7 2 3 4 8 9

4.2.3 Uji Skala Likert

Pengujian dengan memanfaatkan skala *Likert* yang berfungsi sebagai alat untuk mengukur persepsi, sikap, atau tanggapan seseorang. Pada skala *Likert*, digunakan nilai-nilai berikut:

tabel 4. 18 Tabel *Skala Likert*

| SKOR SKALA | |
|------------|---------------------|
| SKOR SKALA | KET |
| 5 | sangat setuju |
| 4 | setuju |
| 3 | netral |
| 2 | tidak setuju |
| 1 | sangat tidak setuju |

Setelah didapat data dari responden yang sudah mengisi kuisioner, kemudian dilakukan pengujian skala *Likert*. Sebelum menguji skala *Likert* penulis terlebih dahulu menentukan nilai interpretasi skor nilai dan nilai interval sebagai acuan persentase kategori, dengan menggunakan rumus sebagai berikut :

Interpretasi skor

Jumlah responden = 104

Nilai max = skor maksimal x jumlah responden

Nilai min = skor minimal x jumlah responden

Nilai max = $5 \times 104 = 520$

Nilai min = $1 \times 104 = 104$

Sedangkan untuk perhitungan rumus Interval adalah :

$I = 100/\text{jumlah skor likert}$

Maka = $100/5 = 20$

Hasil nilai interval yang didapat adalah 20.

Berdasarkan perhitungan skor nilai interval diatas, diperoleh hasil nilai interval, yaitu:

tabel 4. 19 Penilaian Interval

| INTERVAL | KRITERIA |
|---------------|--------------|
| 0 % - 19,99 % | Sangat buruk |
| 20% - 39,99 % | Buruk |
| 40% - 59,99% | Cukup |
| 60% - 79,99% | Baik |
| 80% - 100% | Sangat Baik |

Setelah ditentukan nilai interval kemudian akan dilakukan pengujian indeks pada setiap pertanyaan yang dibuat oleh penulis dan sudah diisi oleh responden. Rumus pengujian pertanyaan adalah jumlah responden x jumlah skor Likert sedangkan untuk rumus pengujian indeks adalah total skor / Y x 100, berikut adalah hasil pertanyaan dan pengujian indeks untuk 10 pertanyaan yang sudah penulis buat:

1. Pengujian interval

tabel 4. 20 pengujian interval

| No | Pertanyaan | Persentase | rata-rata | Kategori |
|----|---|------------|-----------|-------------|
| 1 | Seberapa Mudah Anda Mengakses Aplikasi Ini ? | 83,46% | 82,72% | Sangat Baik |
| 2 | Seberapa Jelas Tampilan Aplikasi Ini? | 83,84% | | |
| 3 | Seberapa sering Anda mengalami kesalahan atau masalah teknis saat menggunakan aplikasi ini? | 78,46% | | |
| 4 | Apakah Petunjuk Dalam Aplikasi Ini Jelas Dan Mudah Dipahami? | 82,30% | | |
| 5 | Seberapa puas Anda dengan waktu respons layanan atau permintaan yang diajukan melalui aplikasi? | 84,23% | | |
| 6 | Bagaimana penilaian Anda terhadap kualitas informasi yang diberikan melalui aplikasi ini? | 81,92% | | |
| 7 | Apakah Anda merasa aplikasi ini meningkatkan keefisienan waktu dalam pelayanan desa? | 83,46% | | |
| 8 | Apakah Informasi Yang Ada Pada Aplikasi Cukup Memuaskan ? | 82,88% | | |
| 9 | Waktu Respon Aplikasi Dalam Memproses Permintaan Atau Laporan Cukup Cepat ? | 81,53% | | |
| 10 | Apakah Aplikasi ini Bermanfaat bagi anda? | 85,15% | | |

Dari hasil pengujian index dari setiap pertanyaan yang dibuat oleh penulis. Dapat disimpulkan bahwa nilai interval yang di dapat dengan skor rata-rata mencapai 82,72% dimana pada penilaian interval masuk dalam kategori (Sangat Baik).

2. Diagram Hasil Uji Berdasarkan Dimensi Variabel

Pada diagram hasil uji dibawah ini didapatkan berdasarkan total jawaban pada masing- masing dimensi.

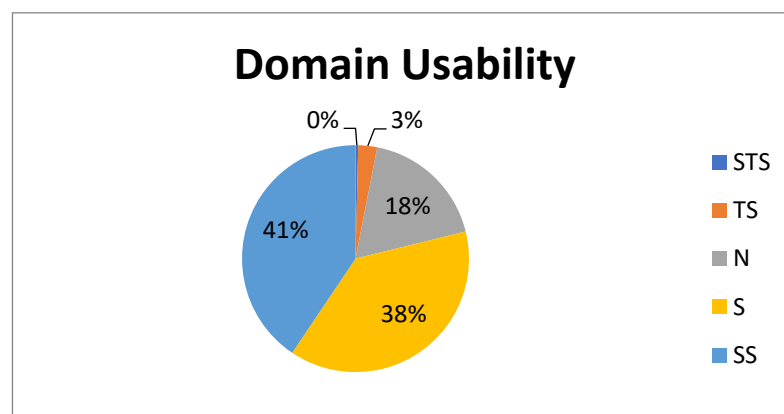
1) Diagram Usability

Diketahui hasil tanggapan kuesioner dari partisipan terkait dimensi Usability sebagai berikut:

tabel 4. 21 Dimensi Usability

| Pertanyaan | Hasil |
|---------------------------|-------|
| Sangat Tidak Setuju (STS) | 2 |
| Tidak Setuju (TS) | 14 |
| Netral (N) | 94 |
| Setuju (S) | 199 |
| Sangat Setuju (ST) | 211 |

Hasil dari respon diatas apabila dibuat menjadi diagram akan seperti gambar di bawah ini



Gambar 4. 23 Diagram Usability

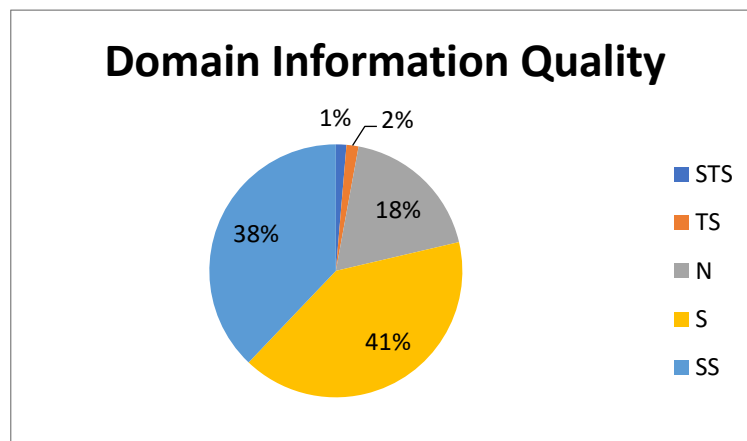
2) Diagram Information Quality

Diketahui hasil tanggapan kuesioner dari partisipan terkait dimensi Information Quality sebagai berikut:

tabel 4. 22 Information Quality

| Pertanyaan | Hasil |
|---------------------------|-------|
| Sangat Tidak Setuju (STS) | 7 |
| Tidak Setuju (TS) | 8 |
| Netral (N) | 96 |
| Setuju (S) | 212 |
| Sangat Setuju (ST) | 197 |

Hasil dari respon diatas apabila dibuat menjadi diagram akan seperti gambar di bawah ini.



Gambar 4. 24 Diagram Information Quality

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini dapat menghasilkan sebuah Sistem Pelayanan Desa (SIPDes) Informasi mengenai Desa Sumber Agung yang dapat membantu masyarakat khususnya Desa Sumber Agung Kabupaten Lampung Selatan yang dibuat menggunakan *framework quasar* dan VueJs sebagai bahasa pemrograman. Pada aplikasi ini terdapat beberapa informasi yang dapat diakses diantaranya berita desa, profil desa, pengajuan surat, peta pembangunan desa, jadwal posyandu, dan laporan pengaduan. Dengan adanya sistem SIPDes ini akses informasi akan semakin mudah, cepat dan akurat dan meningkatkan pelayanan kepada masyarakat akan lebih baik lagi.

Kualitas pelayanan Publik di Kantor Desa Sumber Agung, Kabupaten Lampung Selatan dapat dikategorikan Sangat Baik, hal ini dapat terlihat dari jawaban responden bahwa menyatakan Sangat Baik dengan jumlah 520 dengan persentase 82,72% . Hal ini dapat dilihat dari jawaban responden dan berdasarkan hasil riset, pelayanan yang diterima oleh masyarakat dalam pembuatan Sistem Pelayanan Desa (SIPDes) di Desa Sumber Agung Kabupaten Lampung Selatan dinilai Sangat Baik.

5.2 Saran

Berdasarkan hasil penelitian dan pembahasan pada bab sebelumnya terdapat beberapa saran yang dapat diberikan kepada penulis untuk penelitian berikutnya :

1. Sistem pelayanan desa ini dapat berjalan dengan baik, namun masih memerlukan pengembangan agar sistem pelayanan desa serta tampilan bisa lebih sempurna lagi.
2. Penelitian ini dapat dijadikan sebagai bahan referensi untuk penelitian selanjutnya.

DAFTAR PUSTAKA

- Angela, L., & Erandaru, E. (2022). STUDI PERBANDINGAN TEORI DAN PRAKTEK PROSES PERANCANGAN UI/UX di ARYANNA. *Jurnal DKV Adiwarna*, 1, 10.
- Ariesta, A., Dewi, Y. N., Sariasih, F. A., & Fibriany, F. W. (2021). Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada Pt Xyz. *Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer Dan Teknologi Informasi*, 7(1), 38. <https://doi.org/10.24014/coreit.v7i1.12635>
- Endang Mulyatiningsih. (2011). *Metode Penelitian Terapan Bidang Pendidikan* (Nuryanto Apri (ed.)). UNY Press.
- Hartawan, M. S. (2022). Penerapan User Centered Design (Ucd) Pada Wireframe Desain User Interface Dan User Experience Aplikasi Sinopsis Film. *Jeis: Jurnal Elektro Dan Informatika Swadharma*, 2(1), 43–47. <https://doi.org/10.56486/jeis.vol2no1.161>
- Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Memahami Penggunaan UML (Unified Modelling Language)*, 6(1), 1–15. <https://informatikamulawarman.files.wordpress.com/2011/10/01-jurnal-informatika-mulawarman-feb-2011.pdf>
- Kridalukmana, R., & Windasari, I. P. (2023). PERANCANGAN ULANG USER INTERFACE DAN USER EXPERIENCE (UI / UX) WEBSITE PT SUBUR MAKMUR MIGAS PRATAMA SEBAGAI MEDIA MONITORING ADMINISTRASI MENGGUNAKAN METODE HUMAN CENTERED-DESIGN (HCD). *Jurnal Teknik Komputer*, 2(1), 53–66. <https://doi.org/10.14710/jtk.v2i1.38090>
- Kurniati, I. D., Setiawan, R., Rohmani, A., Lahdji, A., Tajally, A., Ratnaningrum, K., Basuki, R., Reviewer, S., & Wahab, Z. (2015). *Buku Ajar*.
- Mardinata, E., Cahyono, T. D., & Muhammad Rizqi, R. (2023). Transformasi Digital Desa Melalui Sistem Informasi Desa (SID): Meningkatkan Kualitas Pelayanan Publik dan Kesejahteraan Masyarakat. *Parta: Jurnal Pengabdian Kepada Masyarakat*, 4(1), 73–81. <https://doi.org/10.38043/parta.v4i1.4402>
- Mur, M. M., Lia, L., Hafiz, A., Informatika, J. M., Dian, A., Cendikia, C., & Lampung, B. (2019). *Metode Extreme Programming Dalam Membangun Aplikasi Kos-Kosan Di Kota Bandar Lampung Berbasis Web*. XVIII(2013), 377–383.
- Murad, D. F., Widjaya, W., Noviani, D. R., Fitriyyah, N., & Saputri, L. M. (2019). Hybrid Mobile Executive Information (m-EIS) System Using Quasar Framework for Indonesia Financial Service Authority. *Journal of Information Systems Engineering and Business Intelligence*, 5(2), 195.

<https://doi.org/10.20473/jisebi.5.2.195-207>

ROSA, A. . (2022). *Analisa dan Desain Perangkat Lunak*. Informatika Bandung.



Verma, A., Khatana, A., & Chaudhary, S. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304.
<https://doi.org/10.26438/ijcse/v5i12.301304>

Viktor, Widiatry, Ressa, & Putubagus. (2019). View of Penerapan Skala Likert dan Skala Dikotomi Pada Kuesioner Online. In *Jurnal Sains dan Informatika* (Vol. 5, Issue 2, pp. 128–137).
<https://jsi.politala.ac.id/index.php/JSI/article/view/185/106>

Wannahar, J., Septiana Windyasari, V., & Anam, M. K. (2023). Perancangan Sistem Informasi Pelayanan Warga Berbasis Web Di Kelurahan Sindangsari Kabupaten Tangerang. *Jurnal Teknik Informatika Unis*, 11(1), 2252–5351.

LAMPIRAN

1. Bukti Surat Penelitian

| | | | | | | | | | | | |
|--|---|------|-----------------|-----|------------|---------------|---------------|------|------------------------------|---------|------------------|
|  | <p>PROGRAM STUDI INFORMATIKA FAKULTAS ILMU KOMPUTER UNIVERSITAS BANDAR LAMPUNG Jl. ZA Pagar Alam No.89, Gedong Meneng, Rajabasa Bandar Lampung 35142, Telp (0721) 774626, Fax (0721) 701467</p> | | | | | | | | | | |
| <hr/> | | | | | | | | | | | |
| <p style="text-align: right;">Bandar Lampung, 22 Januari 2024</p> | | | | | | | | | | | |
| <p>Nomor : 004/SIP/IF/FIK/UBL/I/2024 Perihal : Permohonan Izin Penelitian</p> | | | | | | | | | | | |
| <p>Kepada Yth,</p> <p style="padding-left: 40px;">Kepala Desa Sumber Agung, Lampung Selatan</p> <p>Di,</p> <p style="padding-left: 40px;">Talang Way Sulan, Kec. Way Sulan, Kab. Lampung Selatan, Lampung 35452</p> | | | | | | | | | | | |
| <p>Dengan Hormat, Dengan ini kami menerangkan bahwa :</p> | | | | | | | | | | | |
| <table border="0"><tr><td>Nama</td><td>: Nadia Ariyati</td></tr><tr><td>NPM</td><td>: 20421039</td></tr><tr><td>Program Studi</td><td>: Informatika</td></tr><tr><td>Asal</td><td>: Universitas Bandar Lampung</td></tr><tr><td>No. Tlp</td><td>: 0812-7151-5542</td></tr></table> | | Nama | : Nadia Ariyati | NPM | : 20421039 | Program Studi | : Informatika | Asal | : Universitas Bandar Lampung | No. Tlp | : 0812-7151-5542 |
| Nama | : Nadia Ariyati | | | | | | | | | | |
| NPM | : 20421039 | | | | | | | | | | |
| Program Studi | : Informatika | | | | | | | | | | |
| Asal | : Universitas Bandar Lampung | | | | | | | | | | |
| No. Tlp | : 0812-7151-5542 | | | | | | | | | | |
| <p>Adalah mahasiswa kami yang akan mengadakan Penelitian dalam rangka menyusun Tugas Akhir. Maka dengan ini kami memohon bantuan Bapak / Ibu untuk memberikan izin mahasiswa tersebut melakukan penelitian di Kantor Kepala Desa Sumber Agung mulai tanggal 23 Januari 2024 s/d 30 Juli 2024.</p> <p>Demikian harapan kami atas perhatian, bantuan dan kerjasamanya kami ucapkan terima kasih.</p> | | | | | | | | | | | |
| <p style="text-align: right;">Ketua Program Studi Informatika,</p> <div style="text-align: center;"> Dr. Maria Shusanti F, S.Kom., M.Kom</div> | | | | | | | | | | | |
| <p style="text-align: right;"><small>Dipada dengan Cap Coornet</small></p> | | | | | | | | | | | |

2. Bukti Wawancara



3. Bukti Pengujian Aplikasi





4. Program Aplikasi User

```

File Edit Selection View Go Run ...
sidxs_aplikasi

EXPLORER
  SIPDES...
  android
  gradle.properties
  gradlew
  gradlew.bat
  local.properties
  settings.gradle
  assets
  icons
  images
  vectors
  build
  ios
  lib
    RestApi.dart
  Components
  pages
  access
  login.dart
  register.dart
  BarBottom
  splashscreen
  OUTLINE
  TIMELINE
  DEPENDENCIES

lib > API > RestApi.dart > ...
1 var baseUrl = 'https://nadiaapp-production.up.railway.app';
2 // var baseUrl = 'http://192.168.196.3000';
3 // users
4 var signIn = Uri.parse("$baseUrl/users/login");
5 var signUp = Uri.parse("$baseUrl/users/registrasi");
6 var addSurat = Uri.parse("$baseUrl/pengajuan_surat/inputdata");
7
8 //
9 var dataBarangRes = Uri.parse("$baseUrl/pengajuan_surat/get-all-barang");
10 var dataPembangunanRes = Uri.parse("$baseUrl/pembangunan/get-all-barang");
11 var dataPosyandu = Uri.parse("$baseUrl/posyandu/get-all-barang");
12
13 //pengaduan
14 var uploadPengaduan = Uri.parse("$baseUrl/pengaduan/input");
15
16 //keranjang
17 var inputKeranjangRes = Uri.parse("$baseUrl/keranjang/input-keranjang");
18 var getAllKeranjangRes = "$baseUrl/keranjang/get-all-keranjang";
19 var updateKeranjangRes = "$baseUrl/keranjang/update-keranjang";
20 var hapusKeranjangRes = "$baseUrl/keranjang/delete-keranjang";
21
22 //transaksi
23 var transaksiInput = Uri.parse("$baseUrl/transaksi/input-transaksi");
24 var transaksiUpdate = Uri.parse("$baseUrl/transaksi/update");
25 var getTransaksi = Uri.parse("$baseUrl/transaksi/get-transaksi-by-idUser");
26
  
```

Activate Windows
Go to Settings to activate Windows.

Ln 4, Col 48 Spaces: 2 UTF-8 CRLF () Dart No Device

20:57 30/07/2024

5. Program Aplikasi Admin

```

1  /* eslint-disable */
2  import { boot } from "quasar/wrappers";
3  import axios from "axios";
4
5  const baseURL = "https://nadiapp-production.up.railway.app";
6  const api = axios.create({ baseURL: baseURL });
7
8  export default boot(({ app, router }) => {
9    app.config.globalProperties.$axios = axios;
10
11    app.config.globalProperties.$ingUrl = baseURL;
12    app.config.globalProperties.$api = api;
13  });
14
15  export { api, axios };

```

6. Program BackEnd

```

1  const express = require("express");
2  const app = express();
3  const path = require("path");
4  const mongoose = require("mongoose");
5  const dbConfig = require("../config/dbConfig");
6  const cors = require("cors");
7
8  app.use(express.json());
9  app.use(cors());
10 app.options("*", cors());
11 app.use(
12   express.urlencoded({
13     enableTypes: ["json", "form"],
14     extended: true,
15   })
16 );
17 app.use(
18   express.json({
19     extended: true,
20   })
21 );
22
23 app.get("/", function (req, res) {
24   res.send("Selamat Datang");
25 });
26
27 app.use("/users", require("../routes/userRoutes"));
28 app.use("/kategori", require("../routes/kategoriRoutes"));
29 app.use("/pembangunan", require("../routes/petaRoutes"));

```

```

30 app.use("/pengaduan", require("../routes/pengaduan"));
31 app.use("/pengajuan_surat", require("../routes/pengajuan_surat"));
32 app.use("/keranjang", require("../routes/keranjangRoutes"));
33 app.use("/transaksi", require("../routes/transaksiRoutes"));
34 app.use("/berita", require("../routes/berita"));
35 app.use("/posyandu", require("../routes/posyandu"));
36
37 app.use("/gambar-barang", express.static("public/images"));
38
39 mongoose
40   .connect(dbConfig.mongoUrl, {
41     useUnifiedTopology: true,
42     useNewUrlParser: true,
43   })
44   .then(() => {
45     console.log("Berhasil Konek ke mongodb");
46     const PORT = process.env.PORT || 4000;
47     app.listen(PORT, () => {
48       console.log("Server Berjalan di Port");
49     });
50   })
51   .catch((err) => {
52     console.log(err);
53   });
54

```