

**PEMBUATAN APLIKASI PENGOLAHAN
DATA DI AMANAH LAUNDRY
BATURAN COLOMADU**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Mencapai Gelar Sarjana Komputer di Program Studi
Informatika, Fakultas Sains, Teknologi dan Kesehatan
Universitas Sahid Surakarta



Disusun Oleh:

Rynaldi Christiano
NIM. 2018061014

**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS, TEKNOLOGI DAN KESEHATAN
UNIVERSITAS SAHID SURAKARTA
2024**

LEMBAR PERSETUJUAN

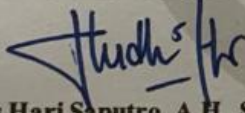
**PEMBUATAN APLIKASI PENGOLAHAN
DATA DI AMANAH LAUNDRY
BATURAN COLOMADU**

Disusun Oleh:

**Rynaldi Christiano
NIM. 2018061014**

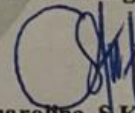
Tugas Akhir ini telah disetujui untuk dipertahankan
Dihadapan Dewan Penguji
Pada Tanggal 13 Februari 2024

Pembimbing I



Firdhaus Hari Saputro, A.H., ST., M. Eng.
NIDN. 0614068201

Pembimbing II



Astri Charolma, S.Kom., M.Cs
NIDN. 0627088301

Mengetahui,

Ketua Program Studi Informatika



Hardika Khusnuliawati, S.Kom., M.Kom.
NIDN. 0631089201

LEMBAR PENGESAHAN

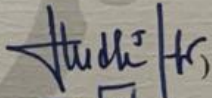
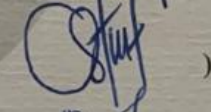
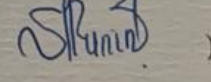
PEMBUATAN APLIKASI PENGOLAHAN
DATA LAUNDRY DI AMANAH LAUNDRY
BATURAN COLOMADU

Disusun Oleh:

Rynaldi Christiano
NIM. 2018061014

Tugas Akhir ini telah disetujui dan disahkan
Oleh dewan penguji Tugas Akhir
Program Studi Informatika
Universitas Sahid Surakarta
Pada hari Kamis tanggal 15 Februari 2024

Dewan Penguji

1. Penguji I Firdhaus Hari Saputro, A.H., ST., M, Eng(
NIDN. 0614068201
2. Penguji II Astri Charolina, S.Kom., M.Cs ()
NIDN. 0627088301
3. Penguji III Sri Huning Anwariningsih, S.T., M.Kom ()
NIDN. 0017067901

Mengetahui,

Ketua Program Studi
Informatika


Hardika Khasnuliawati, S.Kom., M.Kom.
NIDN. 0631089201

Dekan Fakultas
Sains, Teknologi Dan Kesehatan


Ahwan, S.Farm., M.Sc.
NIDN.0605078703

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa telah melimpahkan Rahmat dan Kasih-Nya, sehingga penulis mampu menyelesaikan tugas akhir yang berjudul “ Pembuatan Aplikasi Pengolah Data *Laundry* di Amanah *Laundry* Baturan Colomadu ” dapat diselesaikan dengan baik.

Terima kasih juga disampaikan atas segala bentuk bantuan, dukungan, semangat, dan doa dari berbagai pihak selama pelaksanaan tugas akhir:

- 1) Ibu Sri Huning Anwariningsih, S.T., M.Kom, selaku Rektor Universitas Sahid Surakarta.
- 2) Bapak Apt.Ahwan, S.Farm., M.Sc. selaku Dekan Fakultas Sains, Teknologi, dan Kesehatan Universitas Sahid Surakarta.
- 3) Ibu Hardika Khusnuliawati, S.Kom., M.Kom, selaku Ketua Program Studi Universitas Sahid Surakarta .
- 4) Bapak Firdhaus Hari Saputro Al Haris, S.T., M.Eng, selaku Dosen Pembimbing I.
- 5) Ibu Astri Charolina, S.Kom., M.Cs, selaku Dosen Pembimbing II.
- 6) Ibu Sri Huning Anwariningsih, S.T., M.Kom selaku Dosen Penguji.
- 7) Seluruh anggota keluarga yang telah memberikan dukungan, baik secara moral maupun material.
- 8) Teman-teman mahasiswa Informatika Angkatan 2018 dan semua pihak yang tidak bisa disebutkan satu-satu yang telah membantu dalam menyelesaikan tugas akhir ini.

Segala bentuk kritik dan saran sangat diharapkan untuk menyempurnakan laporan tugas akhir ini sehingga dapat memberikan manfaat dan wawasan tambahan untuk pembaca.

Surakarta, Februari 2024

Penulis

DAFTAR ISI

| | |
|---|-----|
| LEMBAR PERSETUJUAN | ii |
| LEMBAR PENGESAHAN..... | iii |
| KATA PENGANTAR..... | iv |
| DAFTAR ISI..... | v |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 2 |
| 1.2 Perumusan Masalah..... | 2 |
| 1.3 Batasan Masalah..... | 2 |
| 1.4 Tujuan Penelitian..... | 2 |
| 1.5 Manfaat Penelitian..... | 2 |
| BAB II LANDASAN TEORI | 4 |
| 2.1 Tinjauan Pustaka | 4 |
| 2.2 Website..... | 5 |
| 2.3 HTML..... | 5 |
| 2.4 CSS..... | 5 |
| 2.5 PHP..... | 5 |
| 2.6 Bootstrap..... | 6 |
| 2.7 MySQL..... | 6 |
| 2.8 UML(<i>Unnified Modeling Language</i>)..... | 6 |
| 2.9 Metode Penelitian..... | 12 |
| BAB III ANALISIS DAN PERANCANGAN SISTEM..... | 14 |
| 3.1 Perancangan Sistem..... | 14 |
| 3.2 Perancangan Antar Muka..... | 23 |
| BAB IV IMPLEMENTASI DAN ANALISIS HASIL..... | 28 |
| 4.1 Implementasi Halaman Pelanggan | 28 |
| 4.2 Implementasi Halaman Admin | 29 |
| 4.3 Pengujian Sistem..... | 35 |
| BAB V SIMPULAN DAN SARAN | 38 |
| 5.1 Simpulan..... | 38 |

| | | |
|-----|---------------------|----|
| 5.2 | Saran..... | 38 |
| | DAFTAR PUSTAKA..... | 39 |
| | LAMPIRAN..... | 41 |



BABI

PENDAHULUAN

1.1 Latar Belakang

Di era yang semakin modern, usaha jasa *laundry* berkembang sangat pesat. Karena teknologi menggeser pola pikir manusia ke hal yang bersifat instan. Usaha *laundry* merupakan sebuah bisnis yang berkaitan dengan pelayanan jasa cuci pakaian dengan mesin cuci maupun mesin pengering otomatis. Usaha ini cukup menjamur di kota-kota besar terlebih di sekitar kampus yang terdapat banyak kost atau kontrakan yang dihuni mahasiswa yang tidak sempat mencuci atau setrika baju sendiri karena kesibukannya.

Amanah *Laundry* merupakan salah satu dari sekian usaha jasa *laundry* yang telah berdiri sejak tahun 2017 di daerah Baturan, Colomadu. Fokus utama pelayanan di Amanah *Laundry* tidak hanya cuci setrika pakaian saja, melainkan juga menyediakan jasa cuci karpet, boneka, sepatu, Jas, Beskap, dan masih banyak lagi. Hal ini tentu menjadi sebuah keunggulan Amanah *Laundry* dibanding dengan jasa *laundry* yang lain.

Sejak awal berdirinya usaha ini, Amanah *Laundry Kiloan & Dry Clean* belum menerapkan sistem yang terkomputerisasi, sehingga dalam pemrosesan transaksinya masih dilakukan secara manual dimana setiap aktivitas pencatatan di simpan dalam buku. Hal itu membuat Ibu Titik Sumiarti selaku pemilik *laundry* ini sering mengalami kendala saat memeriksa data yang dicatat sebelumnya, seperti salah mencatat jenis paket *laundry*, perhitungannya yang masih menggunakan alat bantu kalkulator, lambatnya pencarian data dan rekap laporan, hingga kehilangan buku transaksinya. Selain itu, Amanah *Laundry* juga merasa perlu memiliki sebuah *website* sebagai media informasi dan profil usaha untuk memperkenalkan keunggulan Amanah *Laundry* kepada calon pelanggan.

Berdasarkan permasalahan tersebut, tugas akhir ini bertujuan untuk menyelesaikan masalah yang ada dengan membuat aplikasi pengolahan data *laundry* yang nantinya aplikasi tersebut akan berguna untuk mempermudah dan membantu mengurangi terjadinya kesalahan dalam pemrosesan pendataan di Amanah *Laundry*.

1.2 Perumusan Masalah

Berdasarkan uraian diatas, dapat dirumuskan masalah yang dihadapi adalah “Bagaimana membuat aplikasi pengolahan data *laundry* di Amanah *Laundry* Kiloan & *Dry Clean*?”.

1.3 Batasan Masalah

Batasan masalah pada aplikasi Amanah *Laundry* sebagai berikut:

- a) Sistem ini dapat di akses oleh pemilik yang ada di Amanah *Laundry*.
- b) Sistem yang di bangun menggunakan Bahasa pemrograman PHP 8.0 dan menggunakan *database* MySQL 8.0 .
- c) Pengembangan aplikasi tidak sampai mengakomodir fitur notifikasi kepada pelanggan.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai ini adalah menghasilkan aplikasi pengolahan data *laundry* yang dapat membantu Amanah *Laundry* Kiloan & *Dry Clean* dalam pemrosesan pendataan.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat-manfaat sebagai berikut:

- a) Bagi Peneliti
Penelitian ini sebagai penambah pengetahuan dan pengalaman di bidang pembuatan sistem berbasis Aplikasi dalam penerapannya di lapangan dan mencoba mengukur seberapa jauh kemampuan peneliti dalam membuat progam sistem.
- b) Bagi Amanah *Laundry*
Sebagai bahan masukan untuk memperbaiki pemrosesan transaksi di Amanah *Laundry* dengan menerapkan sistem yang terkomputerisasi.
- c) Bagi Universitas
Universitas dapat mengetahui kemampuan mahasiswa dalam menerapkan ilmu yang telah diperoleh selama perkuliahan sebagai bahan evaluasi sistem perkuliahan yang ada saat ini.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Menurut Oetario Putro (2021) dengan judul Pengelolaan Data Informasi Pelayanan Jasa *Laundry* Shinwash, sistem pelayanan jasa *laundry* shinwash yang berjalan saat ini masih menggunakan form manual, sehingga sering mengalami ketidaksesuaian data, kehilangan data transaksi pelanggan serta kurang baik dalam penyajian laporannya. Sehingga perlu dilakukan pembuatan aplikasi yang dapat membantu perusahaan jasa *laundry* agar lebih mudah mengelola data dan lebih cepat, selain itu juga mempermudah kasir dalam melayani customer dan mempermudah juga bagi *owner* untuk mengecek secara cepat dan lebih efisien tanpa harus menulis di laporan buku jurnal, dengan adanya aplikasi kasir dan *owner* dengan mudah melihat data customer. Tujuan penelitian ini untuk menganalisa dan merancang suatu sistem informasi pelayanan jasa *laundry* berbasis *web*. Peneliti menggunakan metode waterfall dan pengembangan dengan UML (*Unified Modeling Language*) dengan menggunakan bahasa pemrograman PHP dan *database* MySQL. Hasil yang didapat adalah merancang sistem informasi pelayanan jasa *laundry* berbasis *web* yang akan mendukung kegiatan pelayanan jasa *laundry*, transaksi kiloan, transaksi satuan serta pembuatan kwitansi maupun laporan per periode. Sistem informasi ini menggunakan penyimpanan di dalam 4 *database*, agar admin maupun user dapat mengelola dan melihat informasi pelayanan jasa *laundry* secara akurat dan lengkap. Adanya sistem informasi pelayanan jasa *laundry* berbasis *web* dapat memudahkan user dan admin untuk mengelola pelayanan jasa *laundry* dan laporan.

Menurut Wuyo (2023) dengan judul Rancang Bangun Sistem Pengelolaan Data dan Pelayanan Jasa Pada Berkah *Laundry* Karangasem Berbasis *Website*, diketahui bahwa Berkah *Laundry* Karangasem belum menerapkan sistem yang terkomputerisasi dalam proses mengelola data pelanggan, dan data laporan. Sehingga masih dilakukan secara manual dimana setiap aktivitas pencatatan disimpan dalam buku. Hal itu membuat pemilik *laundry* mengalami kendala ketika memeriksa data yang dicatat sebelumnya. Berkah *Laundry* Karangasem dengan mengedepankan efektivitas dan efisiensi dari segi tenaga dan biaya serta

kebutuhan baik sekarang maupun dimasa pengembangan mendatang. Hasil penelitian menunjukkan sistem pengelolaan data dan jasa pada *laundry* mudah digunakan, mulai dari mengelola data pelanggan, mengelola data admin, mengelola data pemesanan, mengelola data nota, hingga melihat rekap data pemasukan dan melihat rekap data pemesanan.

Pada penelitian Khoirunnisa(2021) merancang aplikasi yang dapat manajemen data sebuah jasa *laundry*. Platform yang dipakai adalah *platform web*. Bahasa pemrograman yang digunakan dalam membangun aplikasi ini adalah bahasa PHP. Manajemen datanya dibuat dalam basis data MySQL. Metode yang diterapkan untuk membangun aplikasi ini adalah metode *Rapid Application 8 Development* (RAD). Aplikasi yang dibuat dapat mengolah data transaksi, mengelola data pengeluaran dan melihat laporan keuntungan atau kerugian dari sebuah *laundry*.

Dari penelitian tersebut memiliki persamaan dengan tujuan penelitian penulis yaitu membuat aplikasi pengolahan data *laundry* berbasis *web* pada jasa *laundry* untuk mempermudah aktivitas pemrosesan data *laundry* menjadi lebih efektif dan efisien. Tinjauan pustaka ini bertujuan sebagai bahan referensi dan rujukan terhadap hasil penelitian sebelumnya yang berhubungan dengan penelitian yang akan dilakukan

2.2 Website

Website adalah keseluruhan halaman *web* yang terdapat dari sebuah domain yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan antara satu halaman *web* dengan halaman yang lainnya disebut dengan *hyperlink* sedangkan teks yang dijadikan media penghubung disebut hypertext (Putra & Wardani, 2019). Secara umum, terdapat dua jenis *website* berdasarkan sifatnya, *website* statis dan *website* dinamis. *Website* Statis adalah *web* yang mempunyai halaman tidak berubah. Sehingga melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengedit *code* yang menjadi struktur dari *website* tersebut. *Website* Dinamis merupakan *website* yang secara struktur diperuntukan untuk update sesering mungkin. Biasanya selain yang bisa diakses oleh user pada umumnya, disediakan halaman backend untuk mengedit konten dari *website* (Putra, A. R., & Wardani, T. I., 2019).

2.3 HTML

HTML (*Hyper Text Markup Language*) adalah sebuah bahasa pemrograman atau file teks yang berisi tag-tag markup yang berguna untuk memberitahukan *BROWSER* bagaimana harus menampilkan sebuah halaman *web* (Pangestu & Afuan, 2021).

Sebuah file HTML harus memiliki ekstensi htm atau html. HTML merupakan bahasa standar yang digunakan oleh *BROWSER* internet untuk membuat halaman dan dokumen pada sebuah *web* yang kemudian dapat diakses dan dibaca layaknya sebuah artikel. HTML juga dapat digunakan sebagai link antara file-file dalam situs atau dalam komputer dengan menggunakan *localhost*, atau *link* yang menghubungkan antar situs dalam dunia internet (Pangestu & Afuan, 2021).

2.4 CSS

CSS (*Cascading Style Sheet*) adalah suatu teknologi bahasa pemrograman *web* yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* akan lebih indah, rapi, terstruktur, dan seragam (Pangestu & Afuan, 2021).

CSS dipakai untuk memformat tampilan *web* seperti *style heading*, *border*, *navbar*, *body text*, *footer*, *images*, *sidebar*, dan *style* lainnya dengan tujuan supaya tampilan halaman *web* lebih bagus untuk dilihat, yang dibuat menggunakan bahasa HTML dan XHTML untuk dapat digunakan bersama-sama di dalam beberapa file (berkas) (Solahudin, 2021).

2.5 PHP

PHP (*Hypertext Preprocessor*) adalah salah satu kode atau bahasa pemrograman, yang diciptakan oleh Rasmus Lerdorf pada tahun 1995, dengan nama awal Form Interpreted (FI), dan hanya memiliki fungsi untuk mengolah data formulir dengan bentuk skrip-nya (Solahudin, 2021).

PHP biasanya menyatu dengan HTML dan dijalankan pada *server side*, sehingga semua sintaks yang kita berikan akan sepenuhnya dijalankan pada *server*, sedangkan yang dikirimkan ke *BROWSER* hanya hasil dari pengolahan sintaks yang telah diolah di sisi *server* (Pangestu & Afuan, 2021).

2.6 Bootstrap

Bootstrap adalah *framework* atau *tools* yang digunakan untuk membuat aplikasi *web* ataupun situs *web* responsive secara cepat, mudah, dan gratis. *Bootstrap* terdiri dari CSS dan HTML untuk menghasilkan *grid*, *layout*, *typography*, *table*, *form*, *navigation*, dan lain-lain. Didalam *bootstrap* juga sudah terdapat *query plugins* untuk menghasilkan komponen user interface yang cantik seperti *transitions*, *modal*, *dropdown*, *scrollspy*, *tooltip*, *tab*, *popover*, *alert*, *button*, *carousel*, dan lain-lain. Kegunaan *Bootstrap* adalah membuat *responsive website* dengan cepat, mudah dan dapat berjalan pada *web browser* umum seperti Chrome, Firefox, Safari Opera, dan Internet Explorer (Wahyudi dkk., 2021).

2.7 MySQL

MySQL adalah DBMS yang open *source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). MySQL salah satu *database* gratis dengan GNU sebagai lisensi-nya dan masuk ke jenis *Relational Database Management System* (RDBMS), dengan tabel, kolom, dan baris sebagai istilah untuk pemakaiannya (Solahudin, 2021).

2.8 UML (*Unnified Modeling Language*)




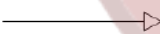
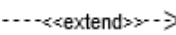

Diagram UML merupakan sebuah alat bantu bagi seorang pengembang aplikasi untuk mendesain sistem. *Unified Modeling Language* (UML) sudah diterima luas sebagai standar untuk pemodelan objek. UML merupakan sebuah bahasa untuk memodelkan sebuah sistem yang akan dirancang melalui notasi - notasi yang sudah disepakati.

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1) *Use Case Diagram*

Use Case Diagram merupakan pemodelan untuk kebutuhan sebuah sistem fungsional, setiap *Use Case* digambarkan sebagai kunci dari suatu skenario yang dilakukan oleh aktor dan diringkas dalam sebuah batas sistem, setiap *Use Case* dihubungkan dengan sebuah garis notasi (Aliman, 2021). Simbol-simbol yang digunakan dalam *Use Case Diagram* dapat dilihat pada Tabel 1.

Tabel 1. *Use Case Diagram* (Sumber : Aliman, 2021)




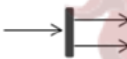

| No | Simbol | Keterangan |
|----|--|---|
| 1 | Actor  | Orang proses, atau sistem lain yang berinteraksi dengan sistem <i>informasi</i> yang akan dibuat di luar sistem <i>informasi</i> yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i> . |
| 2 | Use Case  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau <i>actor</i> biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> |
| 3 | Asosiasi  | Komunikasi antara <i>actor</i> dan <i>Use Case</i> yang berpartisipasi pada <i>Use Case</i> atau <i>Use Case</i> memiliki interaksi dengan <i>actor</i> |
| 4 | Generalisasi  | Hubungan Generalisasi dan sosialisasi antara dua buah <i>Use Case</i> di mana fungsi yang satunya adalah fungsi yang lebih umum dari lainnya. |
| 5 | Extend  | Relasi <i>Use Case</i> tambahan ke sebuah <i>use case</i> |
| 6 | Include  | Relasi tambahan ke sebuah <i>use case</i> di mana <i>Use Case</i> yang di tambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan |

2) Activity Diagram

Activity Diagram merupakan pemodelan yang menggambarkan sebuah sistem kerja dari sebuah objek atau sebuah sistem, sebuah *Activity Diagram*

digambarkan dengan sebuah alur secara terstruktur proses kerja dari *Use Case* yang sedang diproses dari titik awal sampai titik akhir, setiap aktivitas digambarkan dengan notasi - notasi sesuai fungsinya (Aliman, 2021). Simbol-simbol yang digunakan dalam *Activity Diagram* dapat dilihat pada Tabel 2.

Tabel 2. Activity Diagram (Sumber : Aliman, 2021)

| No | Simbol | Keterangan |
|----|---|---|
| 1 | Status Awal  | Status awal aktivitas sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status awal. |
| 2 | Activity  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3 | Decision  | Asosiasi percabangan di mana lebih dari satu aktivitas dijadikan satu. |
| 4 | Join  | Asosiasi penggabungan di mana lebih dari satu aktivitas lebih dari satu. |
| 5 | Final  | Status akhir yang dilakukan sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status akhir. |




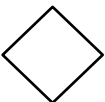
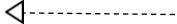
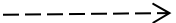
3) Class Diagram

Class diagram adalah salah satu pemodelan yang cukup penting dalam UML, fungsinya adalah untuk membuat sebuah logical models dari sebuah sistem. Sebuah *Class diagram* akan menunjukkan bagaimana skema dari arsitektur sebuah sistem yang sedang dirancang. *Class diagram* digambarkan dengan *Class* yang berisi atribut dan method, setiap *Class* akan dihubungkan dengan sebuah garis disebut Asosiasi.

Class Diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi : Kelas (*Class*), Relasi *Associations*,

Generalitation dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Aliman, 2021). Simbol-simbol yang digunakan dalam *Class Diagram* dapat dilihat pada Tabel 3.

Tabel 3. *Class Diagram* (Sumber : Aliman, 2021)






| No | Simbol | Keterangan |
|----|--|--|
| 1 | Class  | Operasi kelas pada struktur sistem. |
| 2 | Asosiasi  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |
| 3 | Generalisasi  | Relasi antar kelas dengan makna generalisasi spesialisasi (umum khusus) |
| 4 | Nary Association  | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek |
| 5 | Realization  | Operasi yang benar-benar dilakukan oleh suatu objek |
| 6 | Dependency  | Relasi antar kelas dengan makna kebergantungan antar kelas. |

4) Sequence Diagram

Sequence diagram merupakan *diagram* yang digunakan untuk menggambarkan pemrosesan yang ada dalam *Use Case* scenario. Dalam praktiknya, *sequence diagram* sering digunakan dalam proses desain sistem untuk memperoleh interaksi, hubungan, dan berbagai metode objek-objek dalam sistem. Setiap satu *sequence diagram* yang dibuat dalam rangkaian UML, harus merujuk pada satu *activity diagram*.

Banyaknya *diagram sequence* yang harus digambar adalah minimal sebanyak pendefinisian *Use Case* yang memiliki proses sendiri atau yang penting semua *Use Case* yang telah didefinisikan interaksinya pesan sudah dicakup dalam *diagram sequence* sehingga semakin banyak *Use Case* yang didefinisikan maka *diagram* sekuen yang harus dibuat juga semakin banyak (Kartika & Priyadi, 2020). Simbol-simbol yang digunakan dalam *Sequence Diagram* dapat dilihat pada Tabel 4.



Tabel 4. Sequence Diagram (Sumber : Kartika & Priyadi, 2020)

| No | Simbol | Keterangan |
|----|---|--|
| 1 | Actor  | Orang proses, atau sistem lain yang berinteraksi dengan sistem <i>informasi</i> yang akan dibuat di luar sistem <i>informasi</i> yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i> . |
| 2 | Life Line  | Menyatakan Kehidupan suatu objek |
| 3 | Objek  | Menyatakan objek yang berinteraksi pesan |
| 4 | Waktu aktif  | Menyatakan objek dalam keadaan aktif dan berinteraksi. |
| 5 | Pesan  | Menyatakan suatu objek memanggil operasi yang ada pada objek lain atau dirinya sendiri |

5) *Component Diagram*

Component Diagram digunakan untuk memodelkan desain komponen *software* yang digunakan dalam implementasi aplikasi proses bisnis yang telah dibuat. *Diagram* ini bersifat statis yang memperlihatkan organisasi serta kebergantungan pada komponen-komponen yang telah ada sebelumnya (Kartika & Priyadi, 2020). Simbol-simbol yang digunakan dalam *Component Diagram* dapat dilihat pada Tabel 5.

Tabel 5. *Component Diagram* (Sumber : Aliman, 2021)

| No | Simbol | Keterangan |
|----|--|---------------------------------|
| 1 | <p>Component</p>  | komponen sistem |
| 2 | <p>Link</p>  | Menggambarkan relasi antar node |

2.9 Metode Penelitian

1. Metode Pengumpulan Data

Metode pengumpulan data dilakukan untuk menambah pengetahuan dan mencari referensi. Adapun untuk mendapatkan data yang benar-benar akurat, maka diharuskan mengumpulkan data dengan cara berikut ini:

a) Studi Pustaka

Studi pustaka dilakukan dengan cara mengumpulkan data dari berbagai buku, jurnal, literatur, dan *website* yang berkaitan dengan sistem pemrosesan transaksi berbasis *website*.

b) Observasi

Observasi dilakukan secara langsung di Amanah *Laundry* yang beralamat di Jl. Duren, Rt 01/I, Griyan, Baturan, Kec. Colomadu, Kabupaten Karanganyar dengan tujuan untuk memperoleh data dan informasi yang dibutuhkan dalam pembuatan sistem.

c) Wawancara

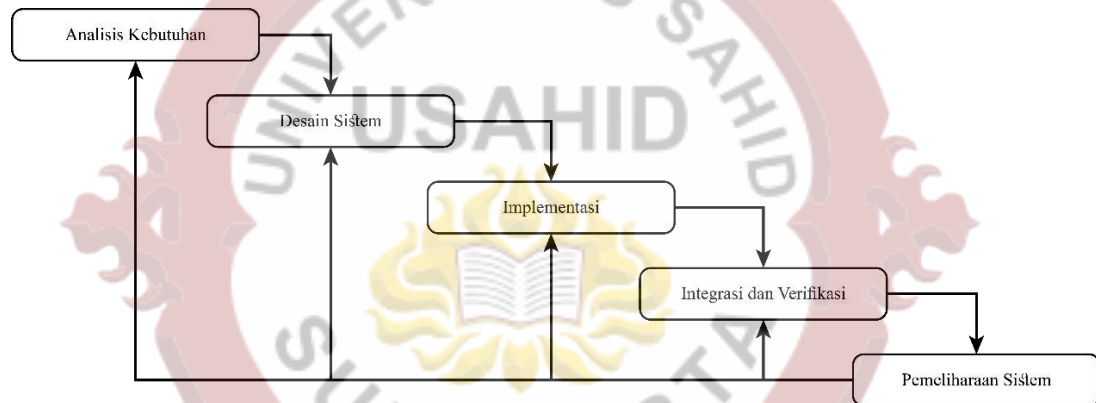
Wawancara dilakukan secara langsung dengan narasumber Ibu Titi Sumarti selaku pemilik Amanah *Laundry*.

2. Metode Pengembangan Aplikasi

Metode pengembangan aplikasi menggunakan metode *Waterfall*, metode air terjun / *Waterfall* yang termasuk dalam kategori metode SDLC (*Software Development Life Cycle*) dalam sebuah pengembangan perangkat lunak dengan pendekatan yang runtut dimulai dengan menganalisis, mendesain, pengkodean, pengujian, dan pendukung. Menurut Febri Kuncoro (2022) metode *Waterfall* dapat diklasifikasikan menjadi beberapa tahapan, yaitu:

- a) Tahap Analisis Sistem, pada tahap analisis sistem ini dilakukan dengan identifikasi permasalahan yang ada pada sistem yang berjalan saat ini dan mendeskripsikan rancangan awal sebagai solusi dari permasalahan tersebut.
- b) Tahap Desain Sistem, pada tahap desain sistem dilakukan pembuatan rancangan dari suatu sistem berdasarkan hasil analisis sistem sebelumnya meliputi alur sistem, skema basis, perancangan bisnis proses, perancangan fungsionalitas pengguna hingga desain antarmuka pengguna.

- c) Tahap Pengkodean Sistem, pada tahap pengkodean sistem ini dilakukan penerjemahan dari desain sistem yang ada kedalam bahasa pemrograman *Hypertext Preprocessor* (PHP) dengan *framework* HTML dengan bantuan aplikasi Visual Studio Code sebagai teks editor.
- d) Tahap Pengujian Sistem, pada tahap pengujian ini dilakukan dengan mengimplementasikan aplikasi secara langsung dan menguji fitur aplikasi berdasarkan tahapan sebelumnya.
- e) Tahap Pendukung Sistem, pada tahap ini dilakukan pengembangan dan pemeliharaan aplikasi agar dapat tetap digunakan dalam beberapa tahun ke depan.



Gambar 1. Metode Waterfall (Sumber : Menurut Febri Kuncoro, 2022)