# PENGESAHAN TUGAS AKHIR

**SISTEM INFORMASI PENGELOLAAN JASA PERBAIKAN ALAT ELEKTRONIK PADA ONEYA SOLUTIONS BERBASIS WEBSITE**

**Oleh:**

**I PUTU YOSUA YERIKHO (170030021)**

|  |  |  |
| --- | --- | --- |
| **Dosen Pembimbing** | **Tanda Tangan** | **Tanggal** |
| Rosalia Hadi, S.Kom., M.T | ........................... | ................... |
| I Gusti Ngurah Ady Kusuma, S.Kom., M.Kom | ........................... | ................... |

Denpasar, …………………..

Mengetahui,

Ketua Program Studi Sistem Informasi

Ricky Aurelius Nurtanto Diaz, S.Kom., M.T

***Halaman ini sengaja dikosongkan***

# KATA PENGANTAR

Berkat Rahmat Tuhan Yang Maha Esa, penulis dapat menyelesaikan Tugas Akhir yang berjudul “**SISTEM INFORMASI PENGELOLAAN JASA PERBAIKAN ALAT ELEKTRONIK PADA ONEYA SOLUTIONS BERBASIS WEBSITE”** sesuai dengan waktu yang telah direncanakan. Selanjutnya penulis menyampaikan Terima Kasih kepada:

1. Rektor ITB STIKOM Bali Bapak Dr. Dadang Hermawan.
2. Bapak Ida Bagus Suradarma, SE., M.Si, selaku Wakil Rektor I.
3. Ibu Ni Luh Putri Srinadi, SE., MM.Kom, selaku Wakil Rektor II.
4. Bapak I Made Sarjana, SE.,MM, selaku Wakil Rektor III.
5. Bapak Dian Pramana, S.Kom., M.Kom, selaku Dekan Fakultas Informatika dan Komputer ITB STIKOM Bali.
6. Bapak Ricky Aurelius Nurtanto Diaz, S.Kom., M.T, selaku Ketua Program Studi Sistem Informasi ITB STIKOM Bali.
7. Ibu Rosalia Hadi, S.Kom., M.T, selaku Dosen Pembimbing Utama yang telah membimbing penulis selama melaksanakan Tugas Akhir.
8. Bapak I Gusti Ngurah Ady Kusuma, S.Kom., M.Kom, selaku Dosen Pembimbing Pendamping yang juga turut membimbing penulis selama melaksanakan Tugas Akhir.
9. Semua teman dan berbagai pihak yang memberikan dukungan dan bantuan kepada penulis.

Semoga hasil penulisan Tugas Akhir ini bermanfaat bagi pihak yang berkepentingan.

Denpasar, Januari 2021

Penulis

***Halaman ini sengaja dikosongkan***

DAFTAR ISI

[PERSETUJUAN UJIAN PROPOSAL TUGAS AKHIR i](#_Toc59875334)

[DAFTAR ISI iii](#_Toc59875335)

[DAFTAR TABEL v](#_Toc59875336)

[DAFTAR GAMBAR vii](#_Toc59875337)

[BAB I PENDAHULUAN 1](#_Toc59875338)

[1.1 Latar Belakang 1](#_Toc59875339)

[1.2 Rumusan Masalah 3](#_Toc59875340)

[1.3 Tujuan Penelitian 3](#_Toc59875341)

[1.4 Manfaat Penelitian 3](#_Toc59875342)

[1.5 Ruang Lingkup Penelitian 4](#_Toc59875343)

[1.6 Sistematika Penulisan 6](#_Toc59875344)

[BAB II TINJAUAN PUSTAKA 7](#_Toc59875345)

[2.1 *State Of The Art* 7](#_Toc59875346)

[2.2 Oneya Solutions 9](#_Toc59875347)

[2.3 Sistem Informasi 9](#_Toc59875348)

[2.4 *Website* 9](#_Toc59875349)

[2.5 XAMPP 10](#_Toc59875350)

[2.6 *Hypertext Preprocessor* (PHP) 10](#_Toc59875351)

[2.7 *Data Flow Diagram* (DFD) 11](#_Toc59875352)

[2.8 MariaDB 12](#_Toc59875353)

[2.9 *Entity Relationship Diagram* (ERD) 12](#_Toc59875354)

[2.10 *Framework* Laravel 13](#_Toc59875355)

[2.11 *Model View Controller* 14](#_Toc59875356)

[2.12 Framework Vue.js 15](#_Toc59875357)

[2.13 REST API 15](#_Toc59875358)

[2.14 JSON 16](#_Toc59875359)

[2.15 *Black Box Testing* 16](#_Toc59875360)

[BAB III METODE PENELITIAN 17](#_Toc59875361)

[3.1 Pengumpulan Data 17](#_Toc59875362)

[3.2 Analisis Sistem 17](#_Toc59875363)

[3.3 Perancangan Sistem 18](#_Toc59875364)

[3.4 Implementasi Sistem 18](#_Toc59875365)

[3.5 Pengujian Sistem 18](#_Toc59875366)

[3.6 Penulisan Laporan 18](#_Toc59875367)

[BAB IV JADWAL KERJA 19](#_Toc59875368)

[DAFTAR PUSTAKA 21](#_Toc59875369)

# **DAFTAR TABEL**

[Tabel 2.1 *State Of The Ar*t 7](#_Toc59874797)

[Tabel 2.2 Simbol-Simbol DFD 11](#_Toc59874798)

[Tabel 2.3 Simbol-SimbolERD *Crow’s Foot* 13](#_Toc59874799)

[Tabel 4.1 Jadwal Kerja 19](#_Toc59874812)

***Halaman ini sengaja dikosongkan***

# **DAFTAR GAMBAR**

[Gambar 2.1 Pola MVC 15](#_Toc59875150)

***Halaman ini sengaja dikosongkan***

# BAB I

# PENDAHULUAN

## Latar Belakang

Alat-alat elektronik merupakan suatu hal yang sudah pasti ditemui dalam suatu rumah tangga. Bahkan, kegunaan alat elektronik mulai bergeser fungsi menjadi kebutuhan primer di beberapa pihak terutama dalam membantu memenuhi kebutuhan sehari-hari [1]. Berbagai perusahaan memproduksi dan menjual peralatan elektronik dan adapula perusahaan yang menyediakan jasa perbaikan peralatan elektronik. Banyak bengkel kecil, menengah, hingga perusahaan yang menyediakan jasa perbaikan peralatan elektronik. perusahaan yang menyediakan jasa perbaikan perlatan elektronik seperti kulkas, AC, dispenser dan lain-lain masih sangat jarang ada yang memanfaatkan media komputer untuk mengelola sistem informasi seputar perbaikan peralatan elektronik untuk meningkatkan pelayanan jasa diperusahaan tersebut [2].

Oneya Solutions merupakan salah satu usaha yang bergerak pada bidang yang menawarkan jasa perbaikan alat-alat elektronik seperti komputer, laptop, *smartphone, dan printer.* Jasa-jasa yang ditawarkan yaitu mulai dari pengantian *solid state drive* (SSD), pemasangan *random access memory* (RAM)*,* peningkatan *hard disk drive* (HDD)*,* penggantian *liquid crystal display* (LCD)*,* yang tentunya mendapatkan garansi hingga 30 hari lamanya. Oneya Solutions juga menerima jasa perakitan komputer, jasa *download* hingga jasa dalam membantu pembelian game seperti Steam, Origin, Unplay. Oneya Solutions telah berdiri sejak 16 Oktober 2018 dan beralamat di Jalan Kebo Iwa Selatan Nomor 19 B Padang Sambian Kaja, Kecamatan. Denpasar Barat, Kota Denpasar. Pada tahun pertama berdirinya Oneya Solutions hanya menerima jasa perbaikan melalui *software* dan *driver* dengan perantara *hard disk drive* dan *flash drive* dimana saat itu hanya mendapatkan satu kali pesanan setiap minggunya. Oneya Solution mulai mengembangkan usahanya dengan cara menjalin kerjasama dengan pihak-pihak penyedia perangkat-perangkat *hardware* yang kini sudah memiliki tiga pihak penyedia perangkat *hardware*. Berkembangnya Oneya Solutions ini hingga dikenal banyak orang dan meningkatnya jumlah permintaan baik jasa perbaikan maupun pemasangan dimana saat ini dalam sehari dapat melayani 6 hingga 7 orang, maka mengharuskan proses pelayanan jasa tersebut dilakukan secara efektif dan efisien. Pencatatan permintaan baik jasa perbaikan, pemasangan dan pembelian *game* yang masih dilakukan secara manual menimbulkan permasalahan baik penentuan waktu penyelesaian yang diharuskan tepat waktu sesuai permintaan pelanggan, kesesuaian biaya yang ditawarkan, kesalahan dari segi penulisan hingga proses pencarian data pesanan yang memakan waktu. Masalah lainnya muncul ketika nantinya terdapat banyak dokumen pesanan yang membutuhkan ruang untuk penyimpanan dan juga menghindari dokumen yang sewaktu-waktu dapat tercecer bahkan hilang.

Sebelumnya pada tahun 2018 pernah dilakukan penelitian terkait yang berjudul “Sistem Informasi Penyewaan Alat-Alat Fotografi Berbasis Web“ oleh I Gede Dwi Ari Mahardiana. Pada penelitian ini menghasilkan suatu sistem informasi yang membantu memberikan kemudahan penyewaan kepada para penyewa yang ingin menyewa dan meringankan pekerjaan karyawan serta memperkecil kesalahan saat membuat laporan. Kesimpulan dari penelitian ini menunjukan kemudahan dalam mencari alat-alat dan informasi harga sewa alat fotografi [3]. Penelitian dengan topik serupa lainnya juga pernah dilakukan sebelumnya oleh Yasir Aminudin pada tahun 2020. Penelitian tersebut berjudul “Sistem Informasi Manajemen Pengelolaan Data Siswa Pada Uptd Blk Industri Dan Pariwisata Provinsi Bali Berbasis Web” Hasil dari penelitian tersebut adalah sistem yang dapat mempermudah calon siswa dan pegawai, mulai dari mendaftar hingga mendapatkan informasi kelulusan. Kesimpulan dari penenlitian tersebut menggambarkan bahwa sistem informasi ini memiliki fitur yang dapat digunakan untuk mengelola data *users*, data periode, dan data pelatihan [4].

Berdasarkan latar belakang permasalahan yang terjadi dan mengacu pada penelitian terdahulu yang telah dijabarkan, maka dalam penelitian ini akan dibangun sebuah sistem informasi pengelolaan jasa perbaikan alat elektronik berbasis *web* pada Oneya Solutions yang dapat membantu dalam pencatatan pesanan yang masuk dari pelanggan*,* serta menangani pesanan sesuai dengan jadwal yang telah ditentukan. Sistem informasi pengelolaan jasa perbaikan alat elektronik berbasis *web* ini dapat diakses oleh admin, teknisi dan *user* dengan hak akses yang berbeda-beda. Admin merupakan pengguna yang memiliki hak akses penuh terhadap sistem baik dapat melakukan pengelolaan berupa penambahan, perubahan data pada jasa, teknisi, *spare part*, hingga melihat saran dan komplain dari pelanggan, juga dapat mencetak laporan keuangan bulanan dan formulir *service* pelanggan. Teknisi merupakan pengguna yang dapat melakukan perubahan status pada *service* yang dikerjakan, mengambil jasa sesuai keahlian teknisi, melihat komplain dari pelanggan dan melihat grafik keberhasilan perbaikan. *User* merupakan pengguna yang dapat melihat status *service* dari perangkatnya dengan cara memasukan *id* yang diberikan saat mengirimkan formulir *service*, mengirimkan saran dan komplain.

Sistem informasi pengelolaan jasa perbaikan alat elektronik ini akan dibangun berbasis *website* agar bisa memudahkan pengguna untuk melakukan proses pencarian atau pengecekan informasi mengenai pesanan yang dapat diakses darimana saja dan kapan saja. Sistem ini diharapkan dapat memberikan informasi tentang pesanan dari pelangganyang harus dipenuhi, meminimalisasi kesalahan dalam pencatatan penerimaan pesanan.

## Rumusan Masalah

Adapun rumusan masalah berdasarkan latar belakang yang telah dijabarkan adalah bagaimana merancang dan membangun sebuah sistem informasi pengelolaan jasa perbaikan alat elektronik berbasis *web* pada Oneya Solutions?

## Tujuan Penelitian

Adapun tujuan penelitian adalah untuk merancang dan membangun sistem informasi pengelolaan jasa perbaikan alat elektronik berbasis *web* pada Oneya Solutions.

## Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Memudahkan admin dalam mencatat pesanan yang diminta oleh pelanggan secara efektif dan efisien yang dapat dilakukan kapan saja dan darimana saja.
2. Memudahkan teknisi dalam mengambil pesanan yang sesuai dengan keahlian yang dimiliki serta merubah status perbaikan yang dikerjakan.
3. Membantu dalam memberikan informasi kepada pelanggan dalam melihat status *service* dari perangkatnya dengan cara memasukan *id* yang diberikan saat mengirimkan formulir *service*.
4. Membantu pelanggan dalam mengirimkan saran dan komplain atas kinerja yang telah dilakukan Oneya Solutions.
5. Membantu admin memantau kinerja atas teknisi nya berdasarkan saran dan komplain dari pelanggan sehingga meningkatkan kualitas pelayanan Oneya Solutions.

## Ruang Lingkup Penelitian

Ruang lingkup dari sistem informasi pengelolaan jasa perbaikan alat elektronik pada Oneya Solutions adalah sebagai berikut:

1. Objek penelitian dari sistem ini adalah Oneya Solutions, dimana pada proses pengelolaan jasa perbaikan alat elektronik merupakan fokus utamanya.
2. Pengguna Sistem

Adapun pengguna dari sistem informasi ini adalah sebagai berikut:

1. Admin

Admin dalam sistem yang akan dikerjakan ini dapat melakukan hak akses sebagai berikut:

* 1. Dapat menambah, menghapus, mengedit dan melihat jasa - jasa yang ada pada Oneya Solutions.
  2. Dapat menambah, mengedit, menghapus dan melihat teknisi – teknisi yang ada pada Oneya Solutions*.*
  3. Dapat menambah dan menghapus *user* yang terdaftar pada Sistem Informasi Oneya Solutions.
  4. Dapat menambah, menghapus, mengedit dan melihat *spare part* yang tersedia di gudang.
  5. Dapat mencetak laporan keuangan bulanan.
  6. Dapat melihat dan menyetujui hasil komplain yang dikirimkan oleh pelanggan.
  7. Dapat melihat, menghapus, dan membalas saran yang dikirimkan oleh pelanggan.
  8. Dapat mencetak formulir *service* yang akan diberikan ke pelanggan.
  9. Dapat mengubah biodata.

b. Teknisi

Teknisi pada sistem informasi yang akan dibuat ini dapat melakukan hak akses sebagai berikut:

1. Dapat mengubah biodata.
2. Dapat melihat dan mengambil jasa perbaikan sesuai dengan keahlian teknisi.
3. Dapat melihat *spare part* yang tersedia digudang.
4. Dapat mengubah status *service* yang sedang berjalan.
5. Dapat melihat dan mengubah status komplain yang dikirimkan oleh pelanggan.
6. Dapat melihat grafik keberhasilan perbaikan.

c. *User*

*User* dalam menggunakan sistem informasi ini dapat melakukan hak akses sebagai berikut:

1. Dapat melihat status *service* dari perangkatnya dengan cara memasukkan *id* yang diberikan saat mengirim formulir *service*.
2. Dapat mengirimkan saran kepada toko.
3. Dapat mengirimkan komplain kepada toko.
4. Dapat menerima dan menyetujui hasil komplain.
5. Dapat melihat informasi teknisi dan admin yang bekerja pada Oneya Solutions.
6. Sistem yang akan dirancang berbasis *website* menggunakan:
   1. Bahasa pemrograman yang digunakan adalah PHP.*.*
   2. *Database* yang digunakan pada sistem ini adalah MariaDB.
   3. Apache digunakan sebagai *web server* untuk mengelolanya.
   4. Perancangan sistemnya menggunakan *Data Flow Diagram* yang terdapat diagram konteks, DFD *Level* 0, DFD *Level* – n.
   5. Perancangan basis data menggunakan ERD, basis data konseptual dan struktur tabel.
   6. Pengujian sistem akan dilakukan dengan menggunakan metode *Blackbox Testing.*

## Sistematika Penulisan

Berikut ini merupakan sistematika penulisan dari laporan yang dikerjakan yang menjabarkan hal-hal yang ada pada setiap bab:

**BAB I PENDAHULUAN**

Dalam Bab ini secara umum menguraikan mengenai latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, ruang lingkup penelitian dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Dalam Bab ini berisi pembahasan mengenai hasil penelitian terdahulu *(state of the art)* yang memiliki keterkaitan topik yang serupa dengan penelitian yang akan dilakukan, seperti berbasis *web*, bahasa pemrograman PHP, DFD *(Data Flow Diagram),* ERD *(Entity Relationship Diagram),* dan *Black Box Testing*.

**BAB III METODE PENELITIAN**

Dalam Bab ini berisikan penjabaran tentang tahapan yang digunakan dalam pengerjaan penelitian seperti wawancara dan observasi untuk mempermudah dalam membangun sisitem informasi pengelolaan jasa perbaikian alat elektronik pada Oneya Solutions..

**BAB IV HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil dari penelitian yang dilakukan yaitu berupa implementasi sistem informasi pengelolaan jasa perbaikan alat elektronik pada Oneya Solution, serta pembahasan hasil penelitian tersebut sesuai dengan metodologi penelitian yang digunakan untuk memberikan jawaban atas hambatan yang telah ditentukan.

**BAB V PENUTUP**

Bab ini menjelaskan dan merangkum kesimpulan dari penelitian yang dibuat, serta memuat beberapa saran yang dapat digunakan untuk menjadi landasan dalam mengatasi hambatan-hambatan yang timbul dalam pengembangan dan perbaikan sistem ke depannya.

# BAB II

# TINJAUAN PUSTAKA

## State Of The Art

*State Of The Art* merupakan rujukan dari beberapa penelitian yang pernah dilakukan sebelumnya yang mengenai penelitian ini. Beberapa contoh penelitian sebelumnya diambil sebagai acuan untuk penelitian yang dilakukan. Berdasarkan pembahasan sebelumnya terdapat beberapa penelitian yang terkait dengan penelitian yang dilakukan yang terdapat pada tabel 2.1 sebagai berikut :

Tabel 2.1 State Of The Art

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Peneliti** | **Judul** | **Keterangan** |
| 1 | I Gede Dwi Ari Mahardiana | Sistem Informasi Penyewaan Alat-Alat Fotografi Berbasis Web (2018) | Pada penelitian ini menghasilkan suatu sistem informasi yang membantu memberikan kemudahan penyewaan kepada para penyewa yang ingin menyewa dan meringankan pekerjaan karyawan serta memperkecil kesalahan saat membuat laporan. Kesimpulan dari penelitian ini menunjukan kemudahan dalam mencari alat-alat dan informasi harga sewa alat fotografi [3]. |
| 2 | Yasir Aminudin | Sistem Informasi Manajemen Pengelolaan Data Siswa Pada Uptd Blk Industri Dan Pariwisata Provinsi Bali Berbasis Web (2020) | Hasil dari penelitian tersebut adalah sistem yang dapat mempermudah calon siswa dan pegawai, mulai dari mendaftar hingga mendapatkan informasi kelulusan. Kesimpulan dari penelitian tersebut menggambarkan bahwa sistem informasi ini memiliki fitur yang dapat digunakan untuk mengelola data *users*, data periode, dan data pelatihan [4]. |
| 3 | Ida Ayu Tri Mahayani | Sistem Pengelolaan Data Transaksi Penjualan Pada Toko Bangunan UD. Mirah Berbasis Web Responsive (2020) | Penelitian tersebut menghasilkan suatu sistem yang dapat membantu dalam pengelolaan data *supplier*, *customer*, *category*, unit, transaksi penjualan, transaksi pembelian serta *owner* dapat melakukan monitoring data transaksi penjualan [5]. |
| 4 | Pendi Mahendrata | Sistem Informasi Pengelolaan Arsip Pajak Klien Pada PT. Guna Artha Kencana Menggunakan Framework Laravel(2019) | Penelitian tersebut menghasilkan suatu yang memberikan kemudahan bagi pengguna/*staff* admin *officer* dalam perpajakan klien serta dapat memberikan laporan arsip perpajakan klien bagi pengguna/*staff* admin *officer* [6]*.* |
| 5 | I Made Liona Agus Agatama | Aplikasi Penjualan Dan Pengorderan Perhiasan Berbasis Web Pada Suteja Silver (2019) | Penelitian ini membantu pihak Suteja Silver agar lebih mudah melakukan pengelolaan produk dan penjualan serta memperluas jaringan dalam hal pemasaran secara nasional maupun internasional. Dengan aplikasi ini pula akan lebih memudahkan dalam melakukan promosi [7]. |

Berdasarkan rujukan dari beberapa penelitian yang pernah dilakukan sebelumnya yang terdapat pada Tabel 2.1 maka pada penelitian ini akan mengangkat judul Sistem Informasi Pengelolaan Jasa Perbaikan Alat Elektronik Berbasis Web dengan bahasa pemrograman PHP serta menggunakan *platform* *web*. Dalam penelitian ini juga akan menggunakan *Entity Relationship Diagram*, *Data Flow Diagram*, basis data konseptual dan struktur tabel serta diharapkan dengan sistem ini dapat membantu pihak Oneya Solutions dalam mengelola jasa perbaikan alat elektronik.

## Oneya Solutions

Oneya Solutions merupakan salah satu usaha yang bergerak pada bidang yang menawarkan jasa perbaikan alat-alat elektronik seperti komputer, laptop, *smartphone, dan printer.* Berdasarkan wawancara dengan Bapak Dandi selaku pemilik Oneya Solutions, Oneya Solutions telah berdiri sejak 16 Oktober 2018 dan beralamat di Jalan Kebo Iwa Selatan Nomor 19 B Padang Sambian Kaja, Kecamatan. Denpasar Barat, Kota Denpasar. Oneya Solutions saat ini telah memiliki 7 orang karyawan, jasa-jasa yang ditawarkan pun berbagai macam yaitu mulai dari pengantian *solid state drive* (SSD), pemasangan *random access memory* (RAM)*,* peningkatan *hard disk drive* (HDD)*,* penggantian *liquid crystal display* (LCD)*,* yang tentunya mendapatkan garansi hingga 30 hari lamanya.

Oneya Solutions juga menerima jasa perakitan komputer, jasa *download* hingga jasa dalam membantu pembelian game seperti Steam, Origin, Unplay. Pada tahun pertama berdirinya Oneya Solutions hanya menerima jasa perbaikan melalui *software* dan *driver* dengan perantara *hard disk drive* dan *flash drive* dimana saat itu hanya mendapatkan satu kali pesanan setiap minggunya. Oneya Solution mulai mengembangkan usahanya dengan cara menjalin kerjasama dengan pihak-pihak penyedia perangkat-perangkat *hardware* yang kini sudah memiliki tiga pihak penyedia perangkat *hardware*.

## Sistem Informasi

Sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi untuk mendukung operasi dan manajemen. Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*Building Box*), dimana masing-masing blok ini saling berintegrasi satu sama lainnya membentuk satu kesatuan untuk mecapai tujuan tertentu [8].

## Website

Situs *web* merupakan suatu layanan sajian informasi yang menggunakan konsep *hyperlink* yang memudahkan *surfer* yang melakukan penelusuran informasi di internet untuk medapatkan informasi dengan cukup mengklik suatu *link* berupa teks atau gambar yang akan ditampilkan secara lebih rinci.

*Website* dikategorikan menjadi dua yaitu *web* statis dan *web* dinamis. *Web* statis adalah *web* yang menampilkan informasi yang sifatnya statis atau tetap, sedangkan *web* dinamis menampilkan informasi serta dapat berinteraksi dengan *user* [9]*.* Pembuatan atau pembangunan *web* dinamis dibutuhkan kemampuan pemrograman *web.* Dalam pemrograman ada dua kategori, yakni:

1. *Server-side programming* yakni perintah-perintah program (*script*) yang dijalankan atau dibaca di *web server,* kemudian hasilnya dikirimkan ke *browser* dalam bentuk HTML (*Hypertext Markup Language*) biasa.
2. *Client-side programming* yakni perintah program yang dijalankan di *web* *browser* sehingga ketika *client* meminta dokumen yang mengandung *script*, maka *script* tersebut akan diunduh dari *server,* kemudian dijalankan di *browser* bersangkutan.

## XAMPP

XAMPP merupakan *tool* yang menyedikan paket perangkat lunak ke dalam satu buah paket. Dengan menginstal XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web* *server* Apache, PHP dan MySQL secara manual. XAMPP juga merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri *(localhost*), yang terdiri atas program Apache HTTP *Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl [10].

## Hypertext Preprocessor (PHP)

*Hypertext Preprocessor* merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu *website* dinamis. PHP menyatu dengan kode HTML yang digunakan sebagai pembangun atau pondasi dari kerangka layout *web*, sedangka PHP difungsikan sebagai proses yang ada di dalam *web* tersebut, sehingga dengan adanya PHP sebuah *web* akan mudah untuk di-*maintenance*.

PHP berjalan pada sisi *server* sehingga PHP disebut juga sebagai Bahasa *Server* *Side* *Scripting*, artinya bahwa dalam setiap menjalankan PHP, wajib ada *web* *server*. PHP bersifat *open* *source* sehingga dapat digunakan secara gratis dan mampu lintas *platform*, yaitu dapat berjalan pada sistem operasi Windows maupun Linux. PHP juga dibangun sebagai modul pada *web* *server* Apache sebagai *binary* yang dapat berjalan sebagai CGI[11].

Berikut ini adalah kelebihan-kelebihan yang dari PHP, yaitu:

1. Mudah dipelajari, alasan tersebut menjadi salah satu alasan utama untuk menggunakan PHP.
2. Mampu lintas *platform*, artinya PHP dapat diaplikasikan ke berbagai *Operating* *System* dan hampir semua *browser* juga mendukung PHP.
3. Gratis dan bersifat *open* *source*.
4. PHP memiliki tingkat akses yang cepat.
5. Didukung oleh beberapa macam *web* *server*, seperti Apache, IIS, Lighttpd, Xitami.

Mendukung beberapa *database*, baik yang gratis maupun berbayar seperti MySQL, PostgreSQL, mSQL, Information SQL *Server*, Oracle.

## Data Flow Diagram (DFD)

*Data Flow Diagram* merupakan serangkaian diagram yang menggambarkan kegiatan-kegiatan yang ada dalam suatu sistem. Teknik pembuatan DFD dimulai dengan menggambarkan sistem secara global dan dilanjutkan dengan melakukan analisis masing-masing bagian. Pada awalnya, digambarkan konteks diagram yang menggambarkan sebuah sistem secara menyeluruh yang akan diinvestasikan. Konteks diagram tersebut dapat dikatakan sebagai DFD *level* 0. Analisis sistem yang lebih detail selanjutnya dapat dilakukan dengan menggambarkan DFD *level* 1, 2 dan seterusnya. Kebanyakan investigasi akan dihentikan pada DFD *level* 2. *Data Flow Diagram (*DFD*)* secara umum harus dapat dimengerti oleh *programmer* karena akan menjadi panduan untuk pembuatan program [12].

Berikut gambar simbol-simbol dalam DFD menurut Gane / Sarson dan Yourdon / De Marco:

Tabel 2.2 Simbol-Simbol DFD

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Gane/Sarson** | **Yourdon/De Marco** | **Keterangan** |
| 1 | entitas1 | entitas2 | Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem. |
| 2 | proses1 | proses2 | Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi. |
| 3 | Aliran Data | Aliran Data | Aliran data dengan arah khusus dari sumber ke tujuan. |
| 4 | Data Store | Data Store | Penyimpanan data atau tempat data direfer oleh proses. |

## MariaDB

MariaDB termasuk salah satu RDBMS yang mulai digunakan di Indonesia diberbagai kalangan baik developer *Desktop* Aplikasi maupun *Web Developer*. Pada dasarnya, MariaDB adalah cloningan MySQL. Boleh dibilang bahwa MariaDB berisi MySQL yang diberi “merek” MariaDB serta penambahan fitur dan perbaikan performa.

MariaDB merupakan versi pengembangan terbuka dan mandiri dari MySQL. Sejak diakusisinya MySQL oleh Oracle pada September 2010, Monty Program sebagai penulis awal kode sumber MySQL memisahkan diri dari pengembangan dan membuat versi yang lebih mandiri yakni MariaDB [13].

## Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* merupakan suatu pemodelan data yang menggunakan *entity* dan relasi diantara *entity* tersebut. Bentley dan Whitten pada tahun (2007) mengatakan *entity* adalah segala sesuatu baik itu yang bersifat abstrak maupun yang *riil* yang ingin kita simpan [14].

*Entity* tidak sama dengan tabel, dimana tabel merupakan hasil pendeskripsian dari *entity* tersebut. Satu *entity* bisa saja dideskripsikan menjadi lebih dari satu tabel. Pemodelan di dalam ERD dinotasikan dengan simbol-simbol. Berikut ini simbol-simbol pada ERD adalah:

Tabel 2.3 Simbol-Simbol ERD Crow’s Foot

|  |  |  |
| --- | --- | --- |
| **No.** | **Simbol** | **Keterangan** |
| 1. |  | Entitas, merupakan objek yang dapat diidentifikasi dalam lingkungan pemakai. |
| 2. |  | Garis, merupakan penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut. |
| 3. |  | *One to One*, merupakan hubungan yang hanya melibatkan satu *record* di satu file dan satu *record* saja pada *file* lainnya. |
| 4. |  | *One to Many*, merupakan hubungan antara 2 tabel, yang dimana data tabel yang terdapat di sebelah kiri akan berhubungan dengan beberapa tabel di sebelah kanan, begitu juga sebaliknya. |
| 5. |  | *Many to Many*, merupakan hubungan antara 2 tabel yang dimana beberapa tabel yang ada pada sebelah kanan berhubungan dengan beberapa tabel di sebelah kanan, begitu juga sebaliknya. |

## Framework Laravel

*Framework* Laravel adalah salah satu jenis *framework* PHPyang dibuat oleh Taylor Otwell dan diluncurkan untuk pertama kali pada 22 Februari 2012. Laravel didesain untuk pengembangan dari teknik MVC (*Model* *View* *Controller*) untuk aplikasi berbasis *web* [15]*.* Laravel dirilis dibawah lisensi MIT dan *source code* dipublikasikan secara bebas di Github untuk dilakukan pengembangan lebih lanjut.

Menurut *survey* para pengembang hingga Desember 2013 tentang popularitas *framework* PHP*,* Laravel menempati posisi teratas sebagai *framework* PHPterpopuler yang diikuti oleh Phalcon, Symfony2, Codeigniter, dan lain sebagainya. Pada bulan Agustus 2014, Laravel menjadi proyek yang paling banyak dilihat di Github. Laravel mempunyai beberapa kelebihan yang tidak dimiliki oleh *framework* PHPlainnya yang membuat menjadi *framework* terpopuler antara lain:

1. *Composer* *Support* merupakan sebuah *dependency* *manager* untuk PHP dimana anda bisa meng-*install* suatu *library* melalui *composer* dan akan secara otomatis meng-*install* *library* lain yang dibutuhkan tanpa perlu mengunduh satu persatu.
2. *Blade* *templating* merupakan *templating* *system* yang disebut dengan *Blade*. Seperti namanya, *Blade* akan memotong-motong struktur pada *template* sehingga lebih mudah dalam pengembangan.
3. *Full* MVC (*Model* *View* *Controller*) merupakan suatu metode yang sangat berguna saat diterapkan dalam mengembangkan aplikasi dengan skala besar karena dengan konsep tersebut kode program yang dihasilkan akan lebih terstruktur. Tentu ini sangat berguna saat akan melakukan *update* dan *maintenance*.
4. Fitur *Eloquent* ORM merupakan fitur yang akan menyederhanakan proses *query* ke *database*.

## Model View Controller

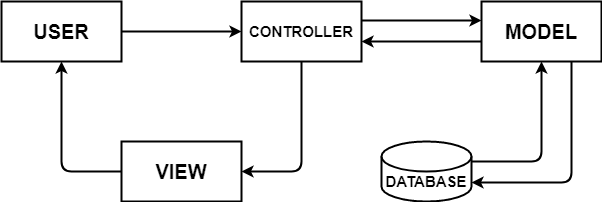
*Model View Controller (*MVC*)* merupakan nama dari suatu yang digunakan untuk merelasikan data dan *user-interface* aplikasi secara efisien. *Model View Controller* awalnya digunakan untuk rancang-bangun aplikasi *desktop*, khususnya untuk aplikasi-aplikasi yang dikembangkan menggunakan C++*,* Java*, dan* Smalltalk. Namun, saat ini arsitektur tersebut telah diadopsi untuk aplikasi berbasis *web*.

Hampir semua *framework* untuk pengembangan aplikasi *web* mengimplementasikan arsitektur MVC. Bukan hanya *framework* PHP, tapi *framework* untuk bahasa pemrograman lain seperti Python(Django*,* Flask*,* web2py, dll) dan Ruby *(*Ruby on Railsdan Sinatra) juga menerapkan konsep dasar yang sama, meskipun istilah yang digunakan sedikit berbeda [16].

Dalam pola *MVC*, komponen aplikasi dibagi menjadi tiga bagian, yaitu :

1. *Model*, yang merepresentasikan struktur data.
2. *View*, yang merupakan representasi keluaran *(output)* dari suatu model.
3. *Controller*, yaitu komponen yang bertugas mengambil masukan (*input*) dari *user* dan mengubahnya menjadi perintah untuk model dan/atau *view*.

Secara umum, pola MVC dapat digambarkan sebagai berikut :



Gambar 2.1 Pola MVC

## Framework Vue.js

Vue.js adalah framework Javascript progresif dan bersifat *open source* untuk membangun UI yang bertujuan agar dapat diadopsi secara bertahap. Vue.js difokuskan pada lapisan tampilan saja, dan mudah diambil dan diintegrasikan dengan proyek yang sudah ada [17].

Vue.js memiliki semua fitur yang harus dimiliki kerangka kerja untuk membangun *SPA*, seperti :

1. *Virtual* DOM adalah representasi pohon dalam memori yang ringan dari HTML asli, dan diperbarui tanpa mempengaruhi DOM asli.
2. Komponen digunakan untuk membuat elemen khusus yang dapat digunakan kembali di aplikasi Vue.js.
3. Template Vue.js menyediakan template berbasis HTML yang membutakan DOM dengan data *instance* Vue.

## REST API

Rest merupakan gaya arsitektur dalam mendesain sebuah *web service* di mana desain REST memiliki *resource* yang dapat diakses melalui sebuah alamat HTTP URL yang *unique*. REST juga memungkinkan klien dapat melakukan request melalui protokol HTTP dengan mudah menggunakan URI.

REST mengirimkan perintah yang akan dikerjakan oleh *server* menggunakan metode-metode HTTP *request method* yang disebut *verb* [19].

## JSON

JSON adalah sebuah format pesan balikan yang berukuran kecil yang mudah dibaca dan ditulis oleh manusia, dan juga mudah ditulis dan diurai oleh mesin. JSON merupakan salah satu pesan balikan yang dapat digunakan dalam sebuah REST API [19].

JSON terbagi dalam dua struktur, yaitu:

1. Gabungan *name/value*, atau yang biasa dikenal *object* atau *record* dalam bahasa pemrograman.
2. *List value*, atau yang biasa dikenal larik dalam bahasa pemrograman.

## Black Box Testing

Pengujian disisi pengembang dilakukan oleh *programmer* dari aplikasi bersangkutan atau mereka yang mengerti dan terlibat dalam pengembangan sistem tersebut. Pengujian disisi pengembang ini umum disebut dengan *Blackbox* *Testing*, penguji ini menekankan pada sejauh mana fungsionalitas sistem informasi berjalan dengan baik sesuai dengan yang diinginkan oleh pengguna dan pengembang.

Terdapat empat buah jenis pengujian pada pengujian disisi pengembang (*Blackbox*) ini antara lain [18]:

1. Pengujian *interface* (antarmuka) aplikasi sistem bertujuan untuk mengetahui fungsionalitas dari setiap elemen *interface* yang ada di setiap halaman aplikasi sistem informasi.
2. Pengujian fungsi dasar sistem bertujuan untuk mengetahui sejauh mana kinerja dari setiap fungsi dasar sistem yang ada di dalam aplikasi sistem informasi.
3. Pengujian *form handle* sistem bertujuan mengetahui seperti apa dan sejauh mana respon oleh sistem informasi terhadap *input* yang diberikan oleh pengguna.
4. Pengujian keamanan sistem bertujuan untuk mengetahui sejauh mana tingkat keamanan yang dimiliki oleh sistem informasi untuk dapat memberikan kenyamanan kepada pengguna.

# BAB III

# METODE PENELITIAN

### Pengumpulan Data

Pengumpulan data ini dilakukan bertujuan untuk mengetahui proses kerja yang selama ini berjalan dan masalah yang terjadi karenanya untuk diselesaikan dengan mengumpulkan data tentang sistem yang akan dibuat. Adapun teknik yang digunakan dalam proses pengumpulan data ini adalah sebagai berikut:

* + 1. Observasi

Observasi merupakan metode pengumpulan data yang dilakukan dengan cara melakukan pengamatan secara langsung serta melakukan pencatatan mengenai informasi yang telah didapat terkait dengan Sistem Informasi Pengelolaan Jasa Perbaikan Alat Elektronik Berbasis *Web* Pada Oneya Solution.

1. Wawancara

Wawancara merupakan metode pengumpulan data yang dilakukan dengan cara melakukan tanya jawab antara narasumber dan pewawancara. Tujuan dari wawancara ini yaitu agar dapat mengetahui data-data berserta informasi mengenai kebutuhan sistem yang akan digunakan nanti, serta untuk mengetahui masalah maupun kendala yang dihadapi.

1. Studi Literatur

Studi Literatur merupakan suatu metode pengumpulan data dan informasi dengan cara menggali pengetahuan atau ilmu dari beberapa sumber yang terkait dengan penelitian yang dibuat yaitu dengan cara mempelajari buku-buku mengenai bahasa pemrograman PHP, teknik-teknik pembuatan *web*, metode pada *Framework* Laravel dan Vue.js, serta buku terkait pemrograman lainnya.

### Analisis Sistem

Tahapan pada analisis sistem ini merupakan tahap mengidentifikasi dan mengevaluasi permasalahan, kelemahan, dan hambatan yang terjadi pada proses kerja sebelumnya yang diharapkan dapat diselesaikan dengan sistem ini. Berdasarkan analisis, proses pencatatan pesanan yang masuk, jadwal penyesaian pesanan, status perbaikan dan penerimaan saran dan komplain dari pelanggan perlu dilakukan dengan cepat dan tepat. Berdasarkan hal itu, perlu dibuatkan Sistem Informasi Pengelolaan Jasa Perbaikan Alat Elektronik Berbasis *Web* Pada Oneya Solutions yang dapat mengatasi kendala maupun masalah tersebut, sehingga dapat lebih mudah penyelesaian kendala dan hambatan yang dikerjakan tersebut.

### Perancangan Sistem

Tahapan perancangan sistem ini dilakukan berdasarkan hasil analisa yang diperoleh. Desain sistem terdiri dari DFD *(Data Flow Diagram)*, ERD(*Entity Relationship Diagram*), basis data konseptual, struktur tabel, dan desain antarmuka Sistem Informasi Pengelolaan Jasa Perbaikan Alat Elektronik Berbasis *Web* Pada Oneya Solutions.

### Implementasi Sistem

Tahapan implementasi sistem merupakan proses merealisasikan hasil analisa dan rancangan sistem yang sudah dibuat menjadi sistem yang bisa dipakai. Proses implementasi sistem ini menggunakan paket aplikasi XAMPP yang terdiri dari Apache sebagai *web* *server*, MariaDB sebagai *database* yang diakses melalui phpMyAdmin sebagai DBMS(*Database* *Management* *System*). Bahasa pemrograman yang digunakan adalah PHP yang menjadi basis dari *Framework* Laravel dan juga menggunakan Framework Vue.js.

### Pengujian Sistem

Tahapan pengujian sistem ini bertujuan untuk memastikan bahwa sistem yang sudah dibuat sesuai dengan rancangan dan berjalan dengan baik. Metode pengujian yang digunakan adalah *black box testing*, yaitu pengujian yang berfokus pada *input* dari pengguna ke sistem dengan mencoba semua fungsi yang terdapat pada sistem dan hasil *output* yang dihasilkan oleh sistem untuk dievaluasi apakah sudah sesuai dengan spesifikasi yang dibutuhkan.

### Penulisan Laporan

Tahap penulisan laporan adalah tahap akhir dari penelitian ini dan sebagai tolak ukur besarnya proses pembuatan sistem dilakukan. Seluruh aktivitas dalam penelitian ini akan dicatat dalam bentuk dokumen berupa laporan tugas akhir. Laporan akan digunakan untuk memantau aktivitas penelitian yang telah dilakukan.

# BAB IV

# HASIL DAN PEMBAHASAN

### Hasil Analisa

Pada tahap analisa sistem, dilakukan beberapa analisis yang bertujuan untuk menganalisa hambatan dan kendala yang mungkin terjadi pada sistem yang akan digunakan dalam pembuatan aplikasi Sistem Informasi Pengelolaan Jasa Perbaikan Alat Elektronik Pada Oneya Solutions Berbasis Website. Tahapan analisis ini terdiri dari 3 (tiga) bagian sebagai berikut :

1. Analisa Pengguna

Tahapan analisa pengguna ini melibatkan 3 (tiga) pengguna yang dapat mengakses sistem berdasarkan hak akses yang diberikan yaitu Admin, Teknisi dan *User*. Hak akses sistem tersebut akan dijabarkan pada Tabel 4.1. beserta keterangannya.

Tabel 4.1 Analisa Pengguna

|  |  |  |
| --- | --- | --- |
| **No** | **Pengguna** | **Keterangan** |
| 1 | Admin | Admin sebagai pengguna yang dapat mengelola data sparepart, mengelola data users, melihat dan membalas saran, melihat komplain, mengelola data *order*, mencetak formulir service, melihat grafik keuangan, serta mencetak laporan. |
| 2 | Teknisi | Teknisi sebagai pengguna yang dapat mengambil *order*, melihat *order*, mengubah status *order*, melihat komplain, mengubah status komplain serta memilih sparepart yang akan digunakan. |
| 3 | *User* | Pada sistem ini user hanya memiliki beberapa fitur yang terbatas, antara lain dapat melihat *order* nya, membuat komplain serta mengirim saran. |

1. Analisa Data

Pada tahap analisa data, pengguna baik admin, teknisi dan user dapat mengakses sistem dengan hak yang berbeda – beda. Data – data yang dapat diakses oleh admin, teknisi dan user tersebut akan dijabarkan pada Tabel 4.2.

Tabel 4.2 Analisa Data

|  |  |  |
| --- | --- | --- |
| **No** | **Data** | **Keterangan** |
| 1 | Data *Users* | Berisikan data – data dari pengguna yang dapat mengakses sistem. |
| 2 | Data Biodata | Berisikan data – data biodata dari pengguna yang terdaftar dalam sistem. |
| 3 | Data Spare Part | Data spare part berisikan data dari setiap spare part yang tersedia digudang seperti nama spare part, deskripsi spare part, tipe spare part, stok sparepart, harga asli serta harga jual. |
| 4 | Data Foto Spare Part | Data foto spare part berisikan data foto dari setiap sparepart seperti nama foto. |
| 5 | Data Jasa | Data jasa berisikan daftar – daftar jasa yang ditawarkan oleh Oneya Solutions seperti nama jasa, deskripsi jasa dan jenis jasa. |
| 6 | Data *Orders* | Data *orders* berisikan data dari setiap *order* seperti *unique id* order, nama pelanggan, alamat pelanggan, nama pelanggan, keluhan, jenis perangkat, merk serta status service. |
| 7 | Data *Orders* Spare Part | Data *orders* spare part berisikan data – data spare part yang diperlukan untuk menyelesaikan *order* seperti nama spare part, jumlah sparepart yang diperlukan, harga asli serta harga jual. |
| 8 | Data Pengaduan | Data pengaduan berisikan data – data dari saran dan komplain yang diterima dari pengguna seperti isi, balasan serta tipe pengaduan. |

1. Analisa Proses

Dalam tahap analisa proses ini, pengguna dapat menjalankan proses yang berkaitan dengan sistem yang akan dijabarkan pada Tabel 4.3 beserta keterangannya.

Tabel 4.3 Analisa Proses

|  |  |  |  |
| --- | --- | --- | --- |
| No | Proses | Keterangan | Pengguna |
| 1 | Login | Agar dapat mengakses sistem, pengguna baik Admin, Teknisi dan User harus melakukan proses login dengan cara memasukkan email dan password terlebih dahulu untuk mendapatkan hak akses sesuai dengan hak akses pengguna. | Admin, Teknisi, User |
| 2 | Kelola data | Dalam kelola data master dilakukan pengelolaan data – data yang tersimpan pada sistem seperti pengelolaan data sparepart, pengelolaan data order, pengelolaan data saran dan komplain, pengelolaan data pengguna serta dapat menampilkan grafik pendapatan serta menampilakn grafik keberhasilan perbaikan. | Admin, Teknisi |
| 3 | Cetak Laporan | Proses cetak laporan hanya dapat dilakukan oleh Admin. Admin dapat mencetak laporan pendapatan bulanan. | Admin |

1. Gambaran Umum Sistem

Berikut ini merupakan gambaran umum sistem yang dapat dilihat pada Gambar 4.1.



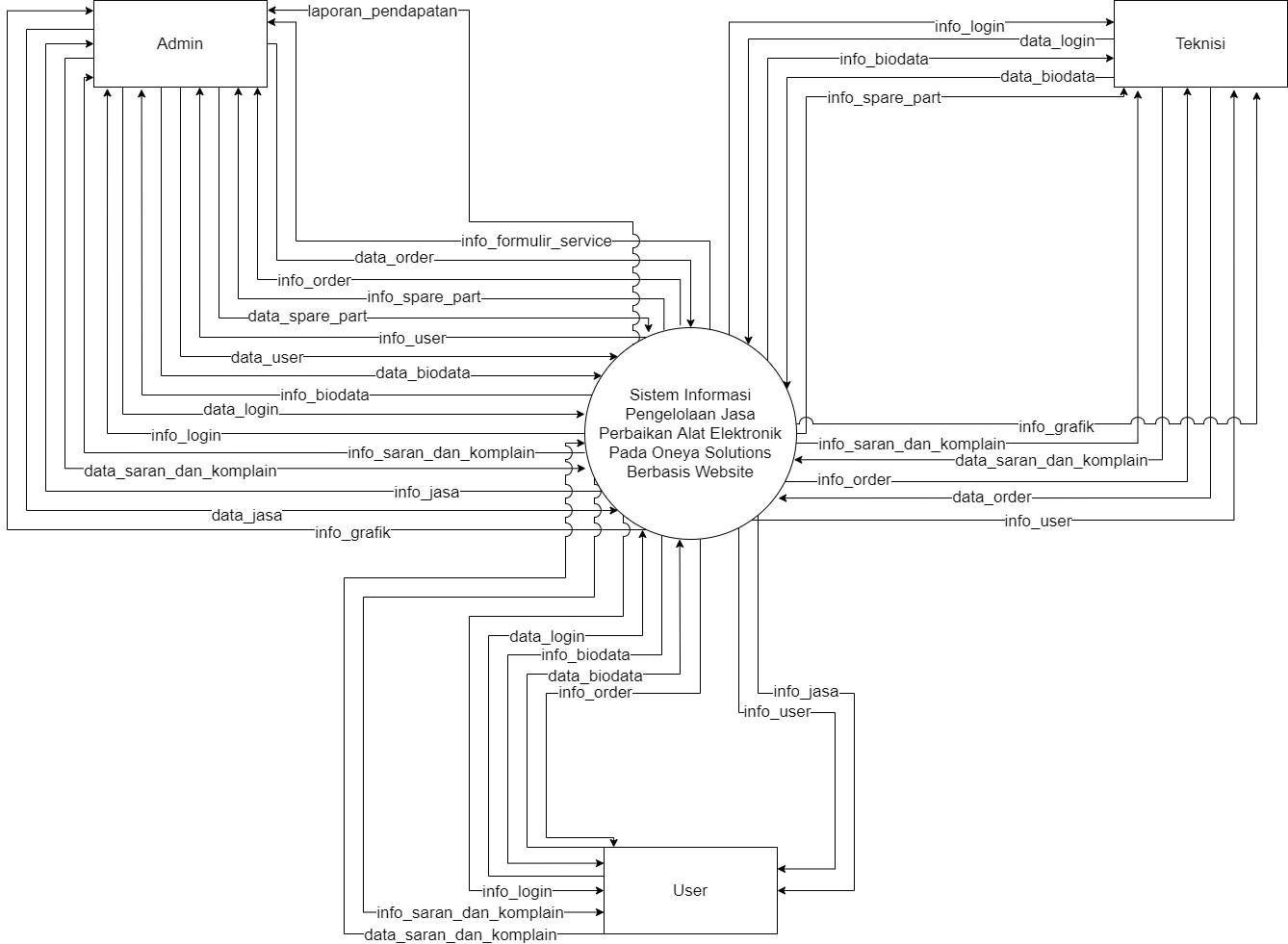
Gambar 4.1 Gambaran Umum Sistem

Gambar 4.1 menjelaskan mengenai proses – proses yang dapat dilakukan pengguna dalam mengakses sistem. Dalam gambar tersebut menjelaskan bahwa sebelum mengakses sistem, pengguna harus melakukan *login* terlebih dahulu. Admin harus *login* sebagai Admin agar dapat melakukan proses penambahan atau pengubahan data – data seperti data users, data sparepart, data saran dan komplain, dan data *order*. Teknisi dapat *login* sebagai teknisi untuk dapat melakukan proses penambahan atau pengubahan data - data seperti data komplain, data *order*, dan pengubahan status *service*. User harus *login* sebagai *User* untuk dapat melihat status *service* nya serta mengirim saran dan komplain kepada toko.

### Perancangan Sistem

Pada tahap perancangan sistem ini dilakukan berdasarkan hasil analisa yang didapatkan sebelumnya. Selanjutnya akan dilakukan proses pembuatan Diagram Konteks, Data Flow Diagram (DFD) Level 0 dan Data Flow Diagram (DFD) Level 1.

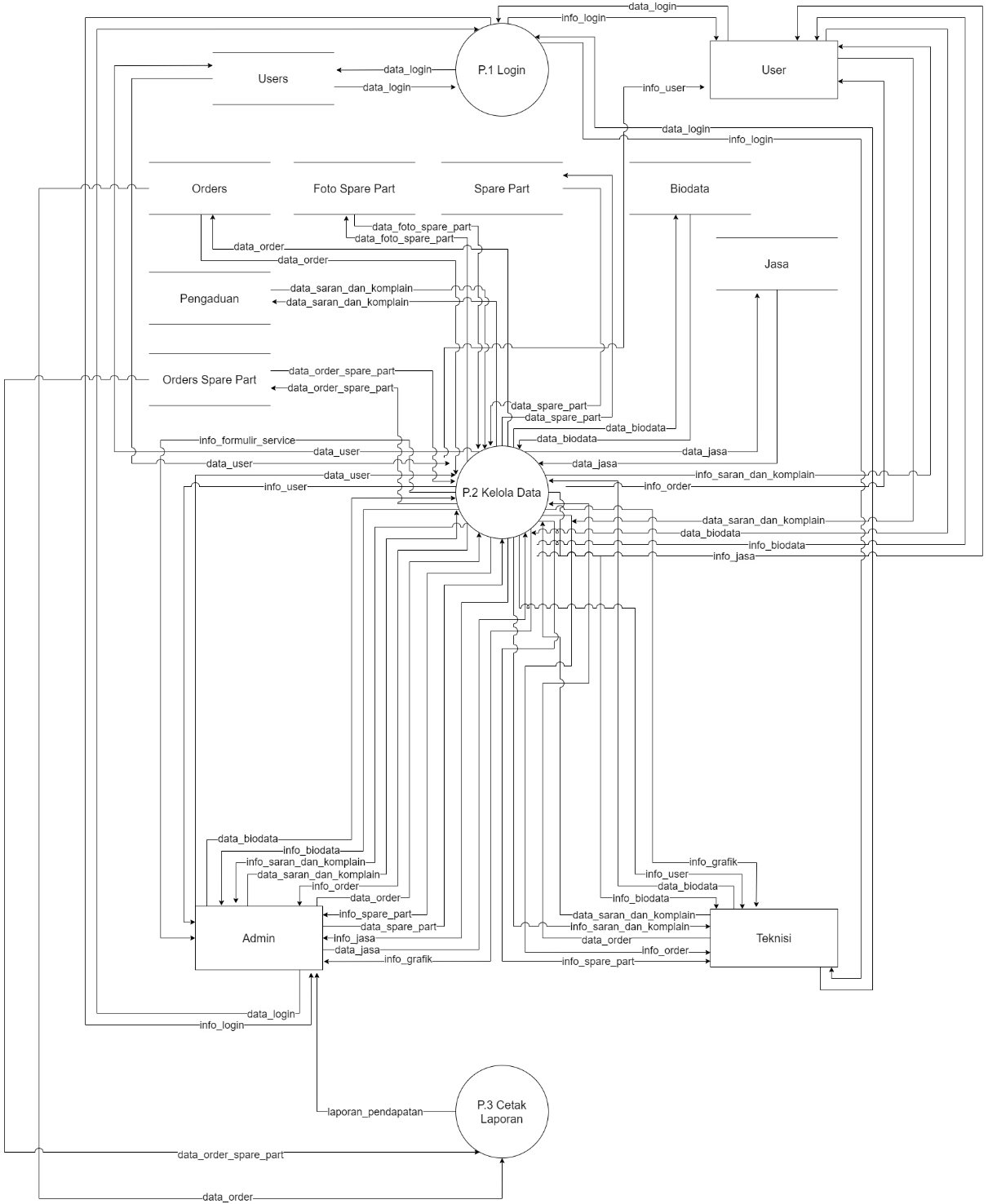
1. Diagram Konteks

Diagram Konteks adalah sebuah diagram yang menggambarkan ruang lingkup sebuah sistem yang secara umum menjelaskan bagaimana sebuah sistem melakukan proses – proses di dalamnya. Terdapat 3 (tiga) buah entitas yang dapat mengakses sistem ini yang terdiri dari Admin, Teknisi dan User. Berikut merupakan gambaran dari Diagram Konteks yang terdapat pada Gambar 4.2.

Gambar 4.2 Diagram Konteks

1. *DFD (Data Flow Diagram) Level* 0

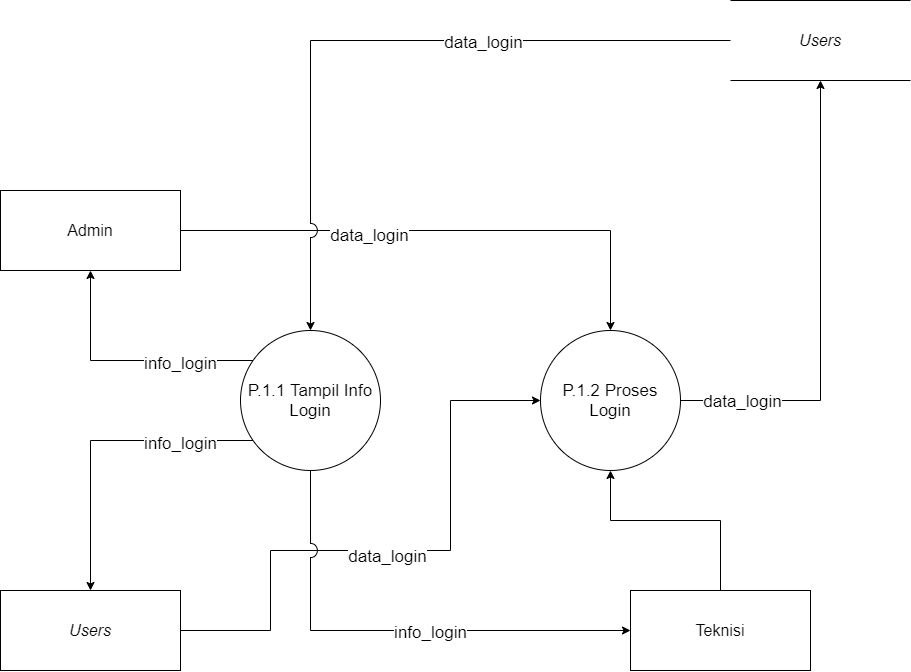
*DFD (Data Flow Diagram) Level* 0 merupakan bagian kecil dari pemecahan Diagram Konteks yang menjelaskan alur sistem lebih detail. Pada *DFD (Data Flow Diagram) Level* 0 ini terdiri dari 3 (tiga) buah proses yaitu proses *login*, kelola data dan mencetak laporan. Selain itu, terdapat 8 (delapan) *data* *store* yang terdiri dari *users*, biodata, sparepart, foto sparepart, jasa, *orders*, *orders* *sparepart* dan pengaduan. Berikut adalah gambaran dari *DFD (Data Flow Diagram) Level* 0 yang terdapat pada Gambar 4.3.



Gambar 4.3 *DFD (Data Flow Diagram) Level* 0

1. *DFD (Data Flow Diagram) Level* 1 Proses Login

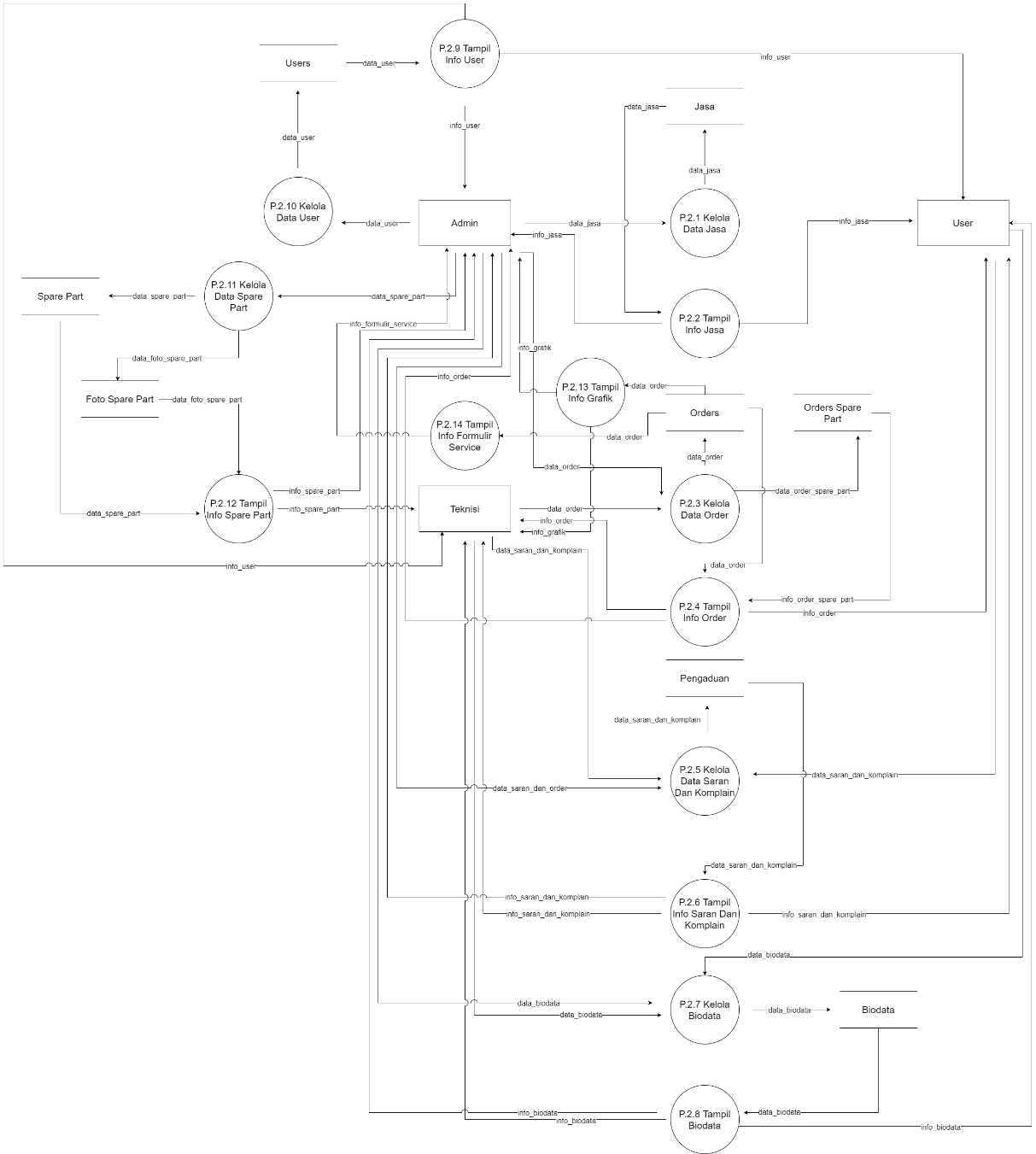
*DFD (Data Flow Diagram) Level* 1 menjelaskan Proses *Login* yang terdiri dari 3 buah proses yaitu *login* pengguna, cek pengguna, tampil *dashboard* serta terdapat 1 (satu) buah data *store* yaitu data *store users*. Agar dapat mengakses sistem, pengguna diharuskan login terlebih dahulu untuk mendapatkan hak akses pengguna. Berikut adalah gambaran dari *DFD (Data Flow Diagram) Level* 1 Proses *Login* yang terdapat pada Gambar 4.4.



Gambar 4.4 *DFD (Data Flow Diagram) Level* 1 Proses Login

1. *DFD (Data Flow Diagram) Level* 1 Kelola Data

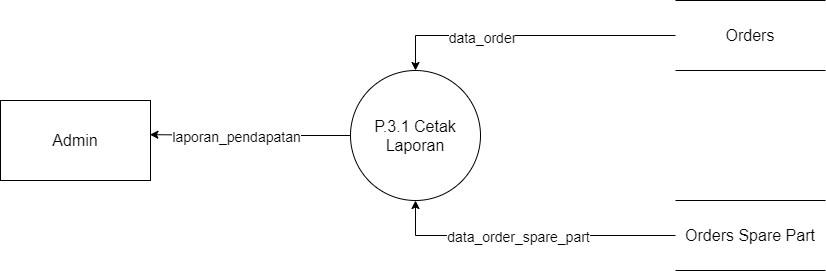
Dalam Proses Kelola Data terdiri dari 14 (empat belas) dimana kelola data hanya dapat dilakukan oleh pengguna yang memiliki hak akses admin dan teknisi, sedangkan pengguna yang memiliki hak akses user hanya dapat menampilkan info proses, kelola data biodata serta kelola data pengaduan, adapun proses – proses tersebut terdiri dari kelola kelola data *users*, tampil info *users*, kelola data spare part, tampil info spare part, kelola data biodata, tampil info biodata, kelola data *orders*, tampil info *orders*, kelola data jasa, tampil info jasa, kelola data pengaduan, tampil info pengaduan serta tampil info grafik. Admin hanya dapat melakukan semua proses namun teknisi hanya dapat melakukan proses kelola data spare part, tampil info spare part, kelola data biodata, tampil info biodata, kelola data *orders*, tampill info *orders*, kelola data pengaduan, tampil info pengaduan serta tampil info grafik. User hanya dapat melakukan proses info proses, kelola data biodata serta kelola data pengaduan. Gambaran dari *DFD (Data Flow Diagram) Level 1* Proses Kelola Data dapat dilihat pada Gambar 4.5.



Gambar 4.5 *DFD (Data Flow Diagram) Level* 1 Proses Kelola Data

1. *DFD (Data Flow Diagram) Level* 1 Proses Cetak Laporan

Dalam Proses Cetak Laporan terdiri dari 1 buah proses yaitu cetak laporan serta terdapat 2 (dua) buah data store yaitu *orders* dan *orders spare part*. Berikut adalah gambaran dari *DFD (Data Flow Diagram) Level* 1 Proses Cetak Laporan pada Gambar 4.6.

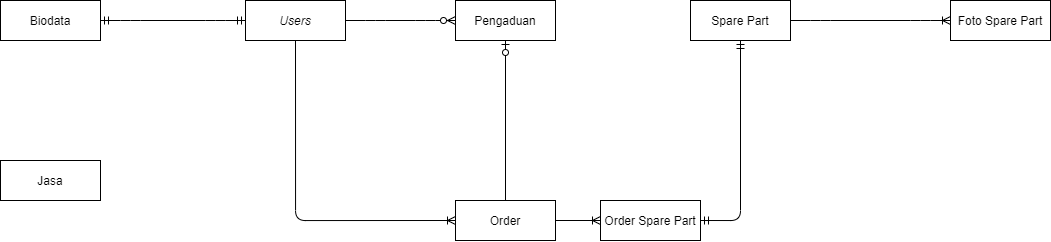


Gambar 4.6 *DFD (Data Flow Diagram) Level* 1 Proses Cetak Laporan

### Perancangan Basis Data

1. ERD (Entity Relationship Diagram)

Pada sistem ini terdapat ERD (Entity Relationship Diagram)yang terdiri dari 8 (delapan) buah entitas yaitu users, biodata, jasa, pengaduan, order, order sparepart, sparepart serta foto sparepart. Berikut adalah gambaran dari ERD (Entity Relationship Diagram) dalam sistem ini yang terdapat pada Gambar 4.7.



Gambar 4.7 ERD (Entity Relationship Diagram)

Adapun atribut – atribut yang terdapat pada entitas pada ERD (Entity Relationship Diagram) yaitu sebagai berikut :

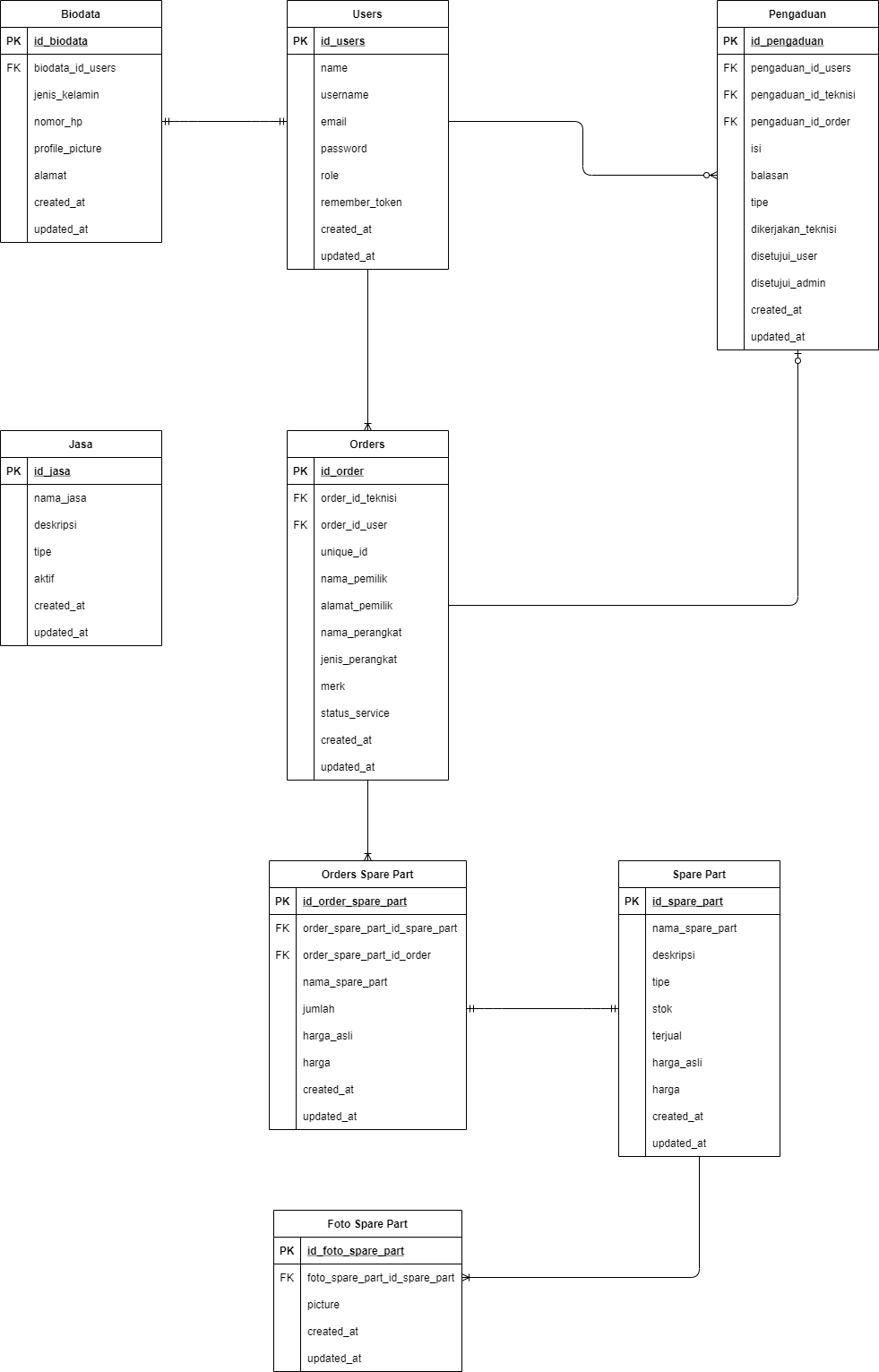
Tabel 4.4 ERD

|  |  |  |
| --- | --- | --- |
| **No** | **Entitas** | **Atribut** |
| 1 | *Users* | id\_users, name, username, email, password, role, remember\_token, created\_at, updated\_at |
| 2 | Biodata | id\_biodata, biodata\_id\_users, jenis\_kelamin, nomor\_hp, profile\_picture, alamat, created\_at, updated\_at |
| 3 | Jasa | id\_jasa, nama\_jasa, deskripsi, tipe, aktif, created\_at, updated\_at |
| 4 | Orders | id\_order, order\_id\_teknisi, order\_id\_user, unique\_id, nama\_pelanggan, alamat\_pelanggan, nama\_perangkat, keluhan, jenis\_perangkat, merk, status\_service, created\_at, updated\_at |
| 5 | Orders Sparepart | id\_order\_spare\_part, order\_spare\_part\_id\_spare\_part, order\_spare\_part\_id\_order, nama\_spare\_part, jumlah, harga\_asli, harga, created\_at, updated\_at |
| 6 | Sparepart | id\_spare\_part, nama\_spare\_part, deskripsi, tipe, stok, terjual, harga\_asli, harga, created\_at, updated\_at |
| 7 | Foto Sparepart | id\_foto\_spare\_part, foto\_spare\_part\_id\_spare\_part, picture, created\_at, updated\_at |
| 8 | Pengaduan | id\_pengaduan, pengaduan\_id\_users, pengaduan\_id\_teknisi, pengaduan\_id\_service, isi, balasan, tipe, dikerjakan\_teknisi, disetujui\_user, disetujui\_admin, created\_at, updated\_at |

1. Konseptual Basis Data

Konseptual Basis Data adalah atribut – atribut dan relasi pada setiap entitas yang dijabarkan dalam sebuah sistem. Berikut adalah gambaran dari Konseptual Basis Data yang ada dalam sistem yang terdapat pada Gambar 4.8.

Gambar 4.8 Konseptual Basis Data



1. Struktur Tabel

Struktur Tabel yang terdapat pada aplikasi ini meliputi tabel users, tabel biodata, tabel spare part, tabel foto spare part, tabel jasa, tabel orders, tabel order spare part, dan tabel pengaduan.

1. Tabel *Users*

Tabel *Users* merupakan tabel penyimpanan data pengguna dimana di dalamnya terdapat hak akses yang berbeda – beda untuk dapat mengakses sistem, dan dapat dilihat pada Tabel 4.5.

Tabel 4.5 Tabel *Users*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_users | bigint(20) | *Primary Key* |
| 2 | name | varchar(60) |  |
| 3 | username | varchar(40) |  |
| 4 | email | varchar(255) |  |
| 5 | password | varchar(255) |  |
| 6 | role | enum(‘admin’, ‘teknisi’, ‘’user) |  |
| 7 | remember\_me | varchar(100) |  |
| 8 | created\_at | timestamp |  |
| 9 | updated\_at | timestamp |  |

1. Tabel Biodata

Tabel Biodata merupakan tabel yang digunakan untuk menyimpan data biodata dari setiap *users* yang terdaftar pada sistem, dimana terdapat relasi antara tabel biodata dan table *users*, dan dapat dilihat pada Tabel 4.6.

Tabel 4.6 Tabel Biodata

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_biodata | bigint(20) | *Primary Key* |
| 2 | biodata\_id\_users | bigint(20) | *Foreign Key* |
| 3 | jenis\_kelamin | enum(‘Laki - laki’, ‘Perempuan’) |  |
| 4 | nomor\_hp | varchar(17) |  |
| 5 | profile\_picture | varchar(255) |  |
| 6 | alamat | varchar(100) |  |
| 7 | created\_at | timestamp |  |
| 8 | updated\_at | timestamp |  |

1. Tabel Spare Part

Tabel Spare Part yaitu tabel yang digunakan untuk menyimpan setiap data spare part, dan dapat dilihat pada Tabel 4.7.

Tabel 4.7 Tabel Spare Part

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_spare\_part | bigint(20) | *Primary Key* |
| 2 | nama\_spare\_part | varchar(100) |  |
| 3 | deskripsi | text |  |
| 4 | tipe | enum(‘hp’, ‘pc/komputer’, ‘printer’) |  |
| 5 | stok | smallint(5) |  |
| 6 | terjual | int(10) |  |
| 7 | harga\_asli | int(20) |  |
| 8 | harga | int(10) |  |
| 9 | created\_at | timestamp |  |
| 10 | updated\_at | timestamp |  |

1. Tabel Foto Spare Part

Tabel Foto Spare Part yaitu tabel yang digunakan untuk menyimpan setiap foto yang dimiliki oleh spare part dimana terdapat relasi antara tabel Spare Part dan tabel Foto Spare Part, dan dapat dilihat pada Tabel 4.8.

Tabel 4.8 Tabel Foto Spare Part

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_foto\_spare\_part | bigint(20) | *Primary Key* |
| 2 | foto\_spare\_part\_id\_spare\_part | bigint(20) | *Foreign Key* |
| 3 | picture | varchar(255) |  |
| 4 | created\_at | timestamp |  |
| 5 | updated\_at | timestamp |  |

1. Tabel *Orders*

Tabel *Orders* merupakan tabel yang digunakan untuk menyimpan data dari setiap *order* yang di terima oleh toko dimana setiap *order* akan menerima *id* yang bersifat unik sehingga tidak terdapat *order* *id* yang sama , dan dapat dilihat pada Tabel 4.9.

Tabel 4.9 Tabel *Orders*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_order | bigint(20) | *Primary Key* |
| 2 | order\_id\_teknisi | bigint(20) | *Foreign Key* |
| 3 | order\_id\_user | bigint(20) | *Foreign Key* |
| 4 | *unique\_id* | varchar(255) |  |
| 5 | nama\_pelanggan | varchar(255) |  |
| 6 | alamat\_pelanggan | varchar(255) |  |
| 7 | nama\_perangkat | varchar(50) |  |
| 8 | keluhan | text |  |
| 9 | jenis\_perangkat | enum(‘hp’, ‘pc/komputer’, ‘printer’) |  |
| 10 | merk | varchar(70) |  |
| 11 | status\_service | enum('menunggu', 'dicek', 'perbaikan', 'selesai', 'pembayaran', 'terima') |  |
| 12 | created\_at | timestamp |  |
| 13 | updated\_at | timestamp |  |

1. Tabel *Orders* Spare Part

Tabel *Orders* Spare Part merupakan tabel yang digunakan untuk menyimpan data sparepart beserta jumlah sparepart yang dibutuhkan saat menyelesaikan order, dan dapat dilihat pada Tabel 4.10.

Tabel 4.10 Tabel *Orders* Spare Part

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_order\_spare\_part | bigint(20) | *Primary Key* |
| 2 | order\_spare\_part\_id\_spare\_part | bigint(20) | *Foreign Key* |
| 3 | order\_spare\_part\_id\_order | bigint(20) | *Foreign Key* |
| 4 | nama\_spare\_part | varchar(120) |  |
| 5 | jumlah | smallint(5) |  |
| 6 | harga\_asli | bigint(20) |  |
| 7 | harga | bigint(10) |  |
| 8 | created\_at | timestamp |  |
| 9 | updated\_at | timestamp |  |

1. Tabel Jasa

Tabel Jasa merupakan tabel yang digunakan untuk menyimpan data jasa yang ditawarkan oleh Oneya Solutions dimana jasa yang ditawarkan dapat dilihat oleh pengguna yang memiliki hak akses *user*, struktur tabel jasa dapat dilihat pada Tabel 4.11.

Tabel 4.11 Tabel Jasa

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_jasa | bigint(20) | *Primary Key* |
| 2 | nama\_jasa | bigint(30) |  |
| 3 | deskripsi | varchar(255) |  |
| 4 | tipe | enum(‘hp’, ‘pc/komputer’, ‘printer’) |  |
| 5 | aktif | tinyint(1) |  |
| 6 | created\_at | timestamp |  |
| 7 | updated\_at | timestamp |  |

1. Tabel Pengaduan

Tabel Pengaduan merupakan tabel yang digunakan untuk menyimpan data pengaduan dari pengguna seperti ‘saran’ dan ‘komplain’, struktur tabel pengaduan dapat dilihat pada Tabel 4.12.

Tabel 4.12 Tabel Pengaduan

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Field** | **Tipe Data** | **Key Constraint** |
| 1 | id\_pengaduan | bigint(20) | Primary Key |
| 2 | pengaduan\_id\_users | bigint(20) | Foreign Key |
| 3 | pengaduan\_id\_teknisi | bigint(20) | Foreign Key |
| 4 | pengaduan\_id\_order | bigint(20) | Foreign Key |
| 5 | isi | text |  |
| 6 | balasan | text |  |
| 7 | tipe | enum(‘saran’, ‘komplain’) |  |
| 8 | dikerjakan\_teknisi | tinyint(1) |  |
| 9 | disetujui\_user | tinyint(1) |  |
| 10 | disetujui\_admin | tinyint(1) |  |
| 11 | created\_at | timestamp |  |
| 12 | updated\_at | timestamp |  |

### Perancangan Desain Antar Muka

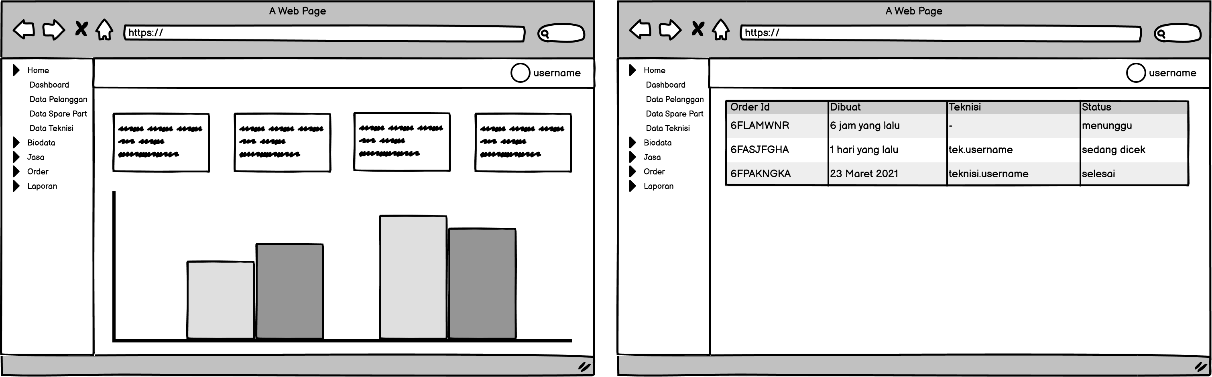
Perancangan desain antar muka merupakan bagian yang paling penting dalam merancang sebuah sistem. Dalam perancangan antar muka akan dijabarkan mengenai desain antar muka yang akan digunakan dalam implementasi sebuah sistem yang akan dibuat.

1. **Desain Antarmuka Halaman Admin**

Berikut merupakan rancangan desain antar muka untuk halaman yang dapat diakses oleh pengguna yang memiliki hak akses *admin*. Berikut merupakan rancangan antar muka tersebut:

1. Halaman Dashboard

Halaman Dashboard untuk hak akses admin memiliki beberapa perbedaan tampilan dengan hak ases lain, diantaranya adalah perbedaan tampilan grafik dan tampilan kotak informasi. Pada hak akses admin, admin dapat mengakses menu home, mengakses menu users, menu *spare part*, menu jasa, menu biodata, menu *orders*, menu pengaduan yang berada di pojok kanan atas serta dapat mengakses menu pembuatan laporan, dan dapat dilihat pada Gambar 4.9.

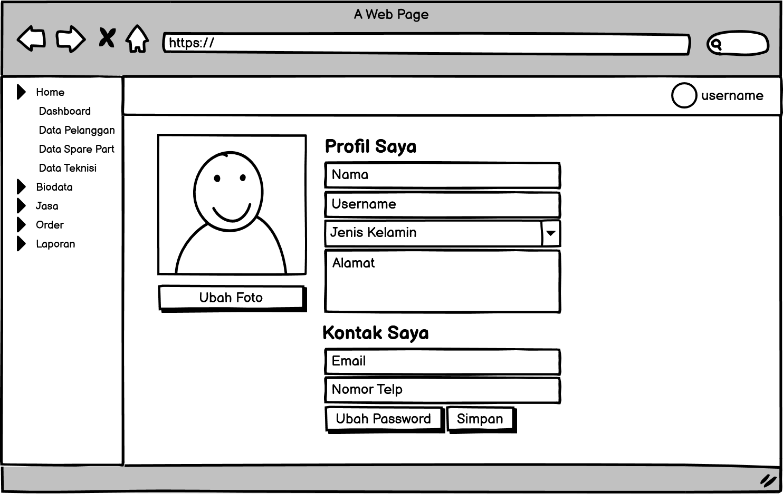


Gambar 4.9 Halaman Dashboard

1. Halaman Biodata

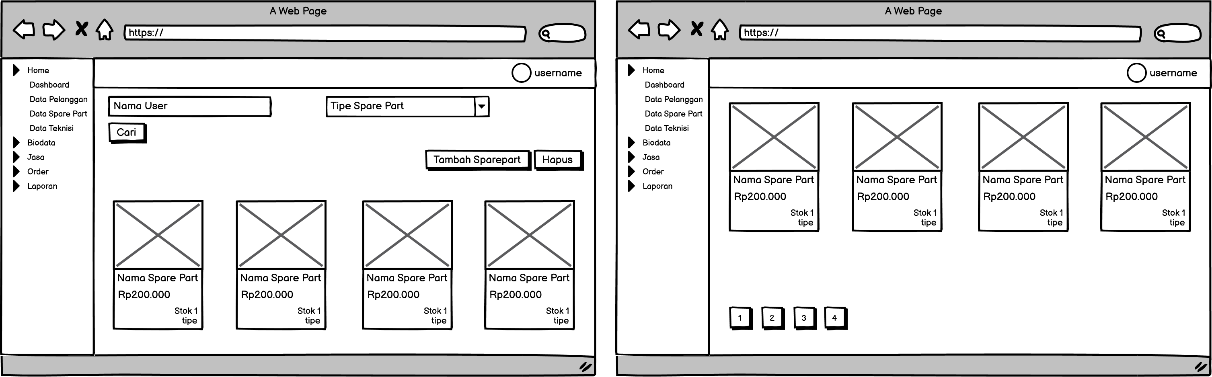
Halaman Biodata berisikan data – data biodata dari admin, pada halaman biodata, admin juga dapat melakukan perubahan terhadap biodata dirinya seperti biodata profil serta biodata kontak, dan dapat dilihat pada Gambar 4.10.

Gambar 4.10 Halaman Biodata



1. Halaman *Spare Part*

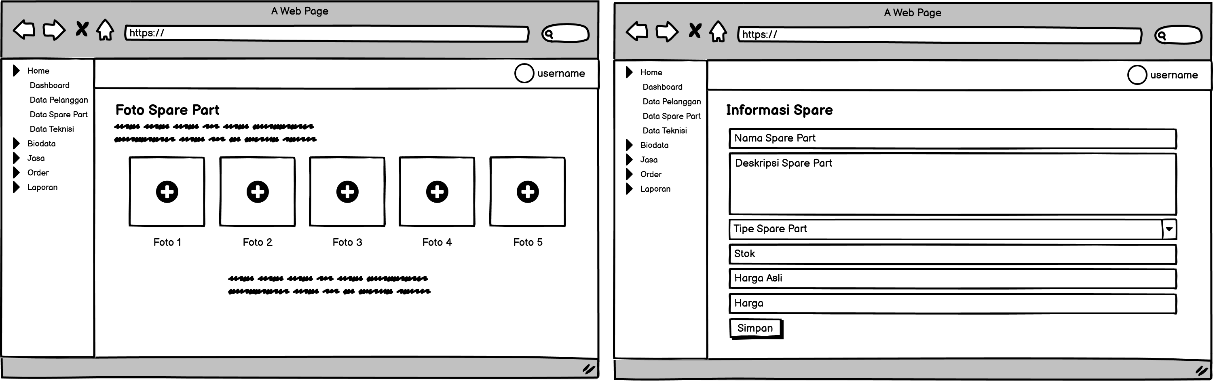
Halaman *Spare Part* berisikan *list spare part* yang tersedia digudang beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.11.



Gambar 4.11 Halaman *Spare Part*

1. Halaman Tambah dan Edit *Spare Part*

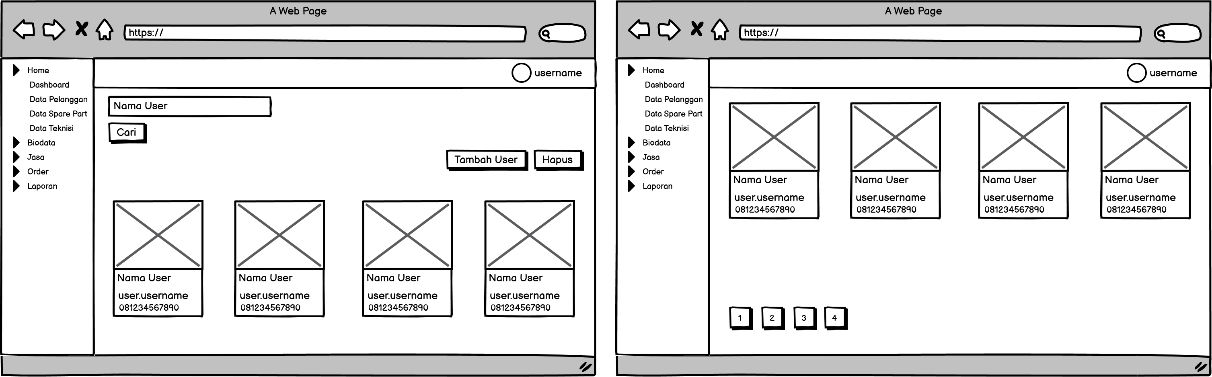
Halaman Tambah dan Edit Spare Part berisikan kolom – kolom mengenai data spare part yang akan di tambahkan maupun diubah, tidak ada perbedaan yang signifikan antara kedua halaman tersebut, halaman tersebut dapat dilihat pada Gambar 4.12.



Gambar 4.12 Halaman Tambah dan Edit Spare Part

1. Halaman *List* Pelanggan

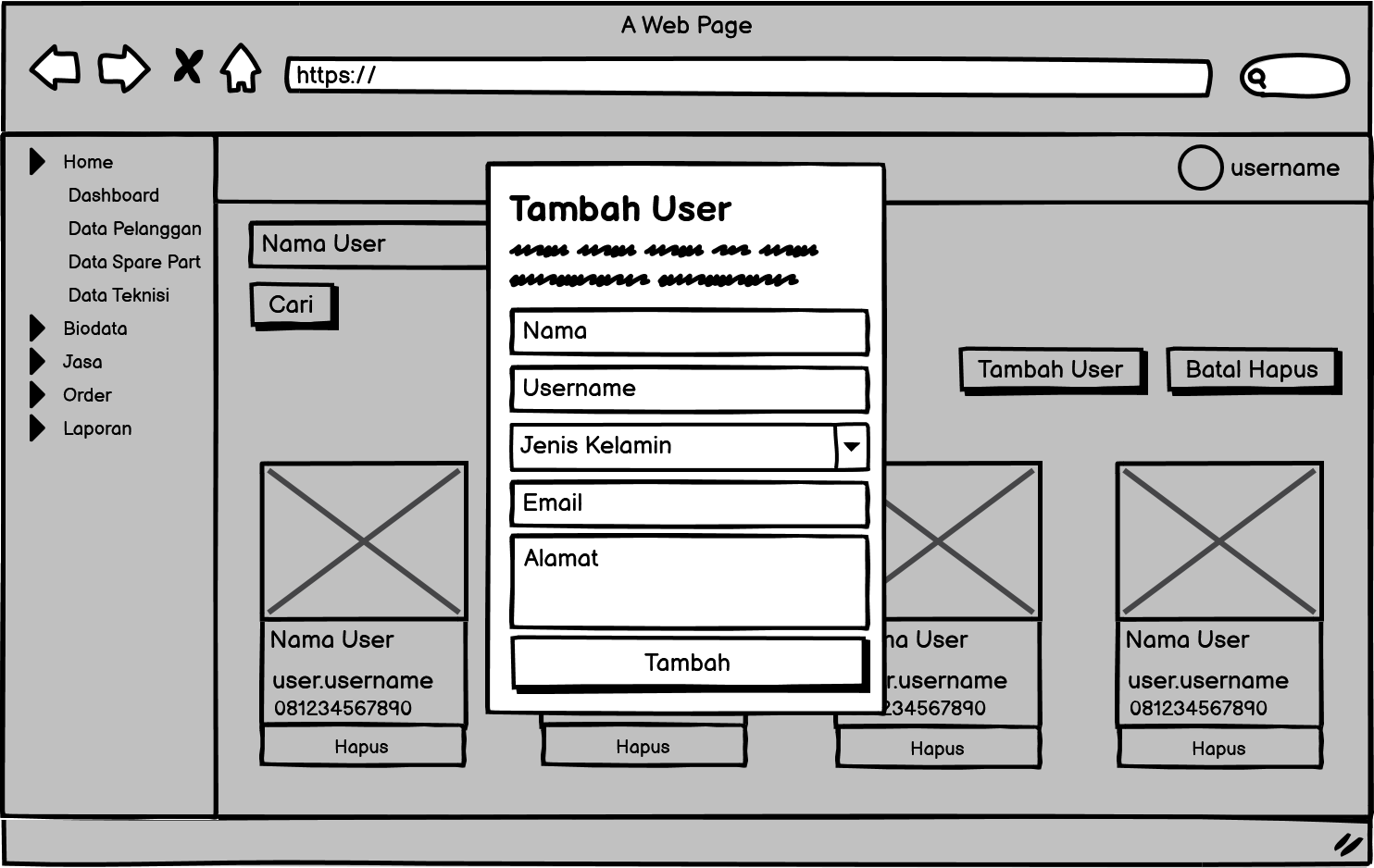
Halaman *List* Pelanggan berisikan *list* pelanggan yang terdaftar pada sistem beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.13.



Gambar 4.14 Halaman Pelanggan

1. Halaman Tambah Pelanggan

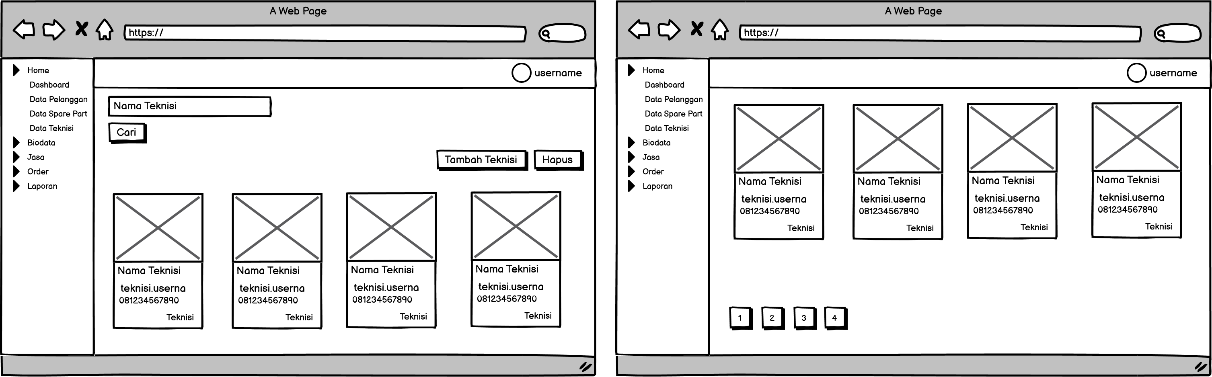
Halaman Tambah Pelanggan berisikan kolom – kolom mengenai data pelanggan, dan dapat dilihat pada Gambar 4.15.



Gambar 4.15 Halaman Tambah Pelanggan

1. Halaman Teknisi

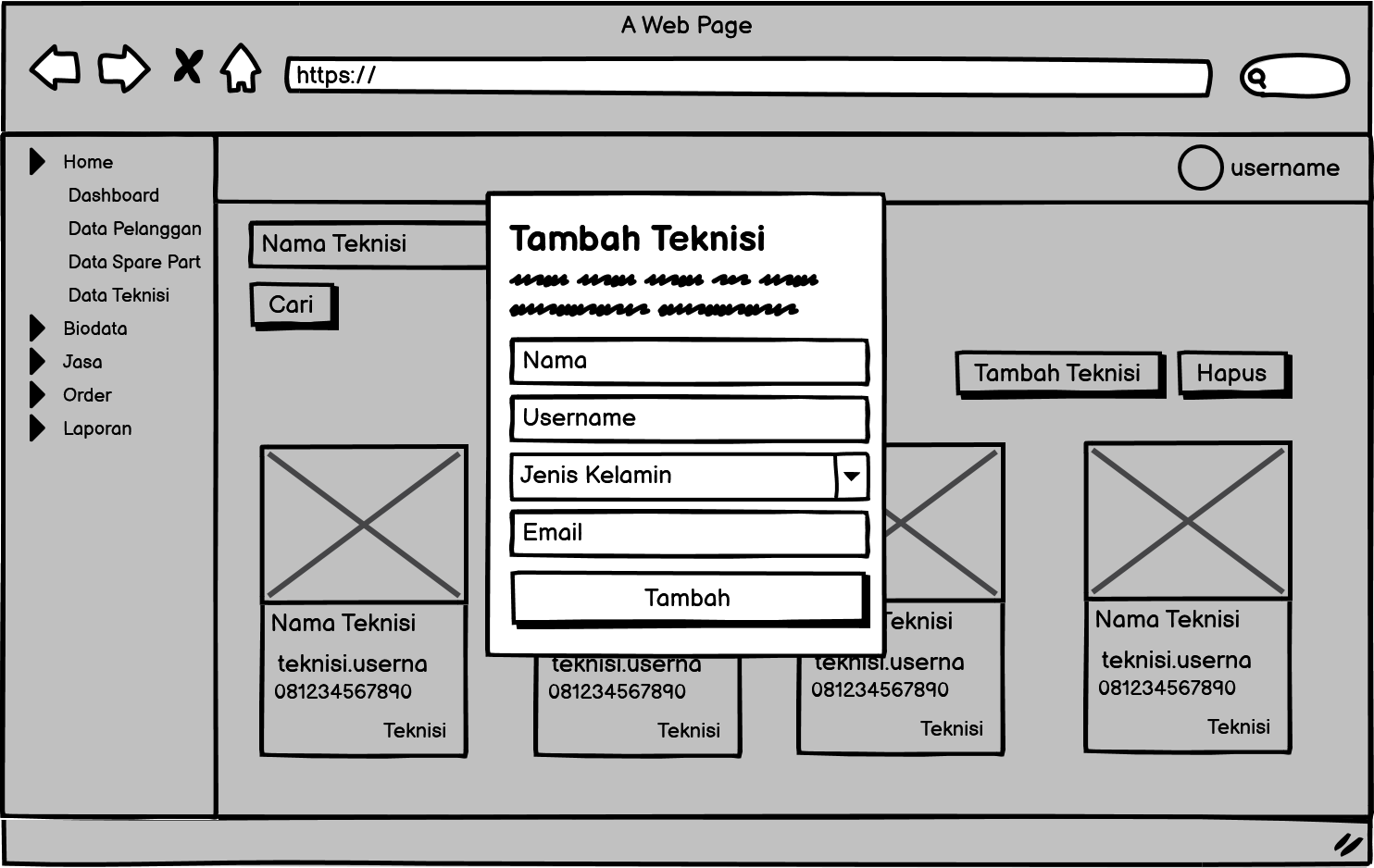
Halaman Teknisi berisikan *list* teknisi yang bekerja pada Oneya Solutions beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.13.



Gambar 4.14 Halaman Teknisi

1. Halaman Tambah Teknisi

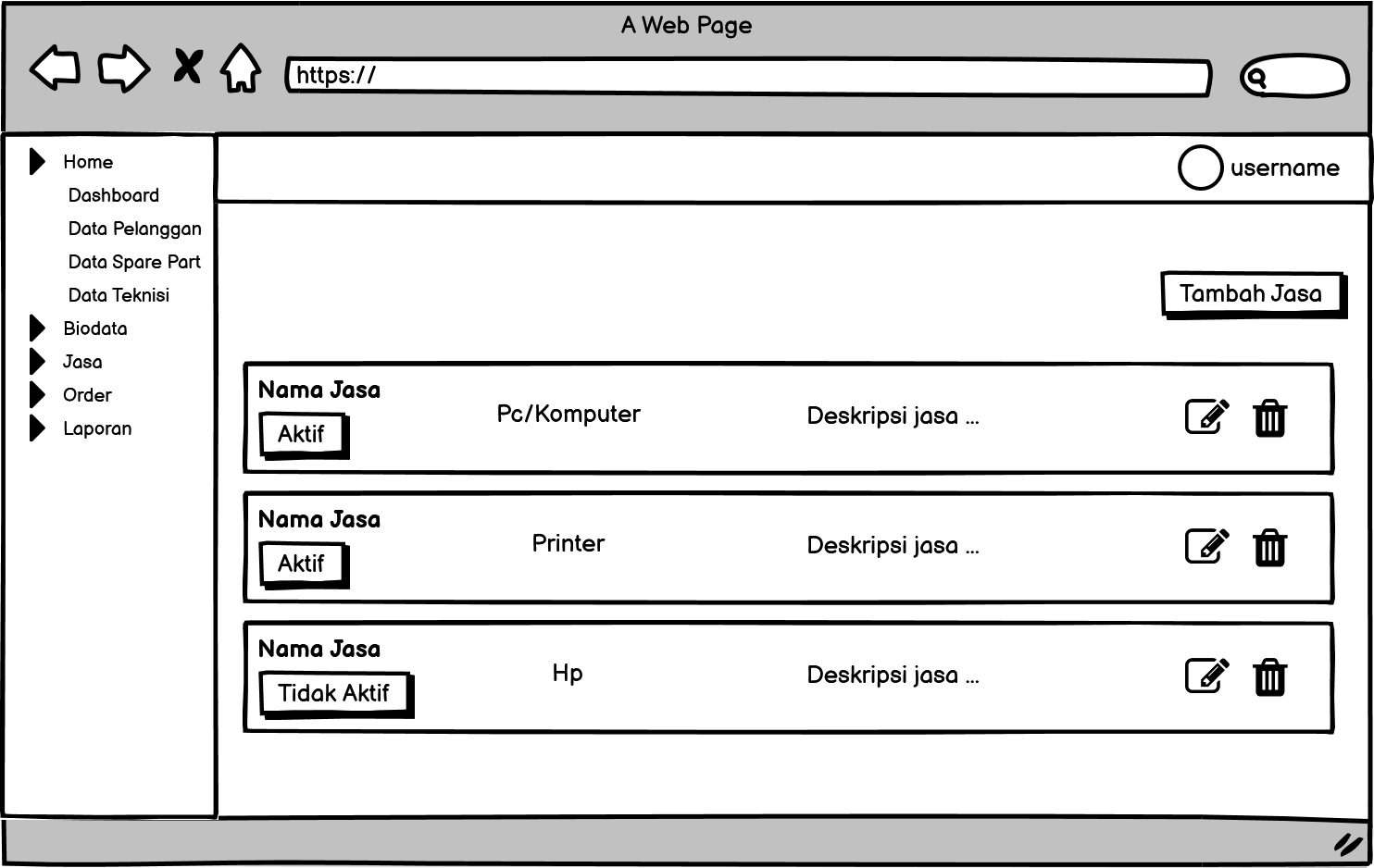
Halaman Tambah Teknisi berisikan kolom – kolom mengenai data teknisi, dan dapat dilihat pada Gambar 4.15.



Gambar 4.16 Halaman Tambah Teknisi

1. Halaman Jasa

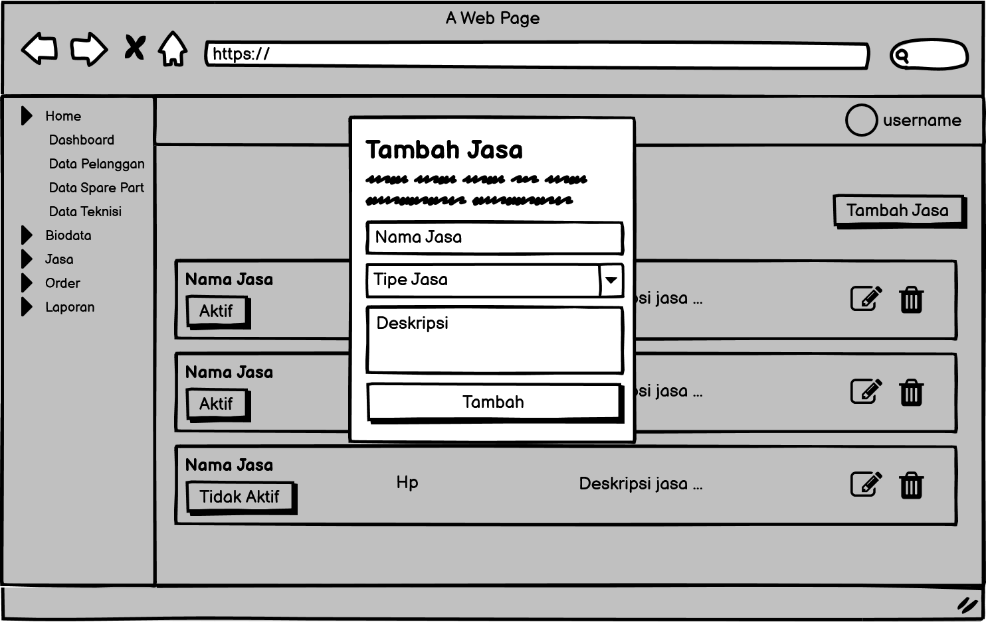
Halaman Jasa berisikan jasa – jasa yang dapat dilakukan oleh Oneya Solutions, dan dapat dilihat pada Gambar 4.17.



Gambar 4.16 Halaman Tambah Jasa

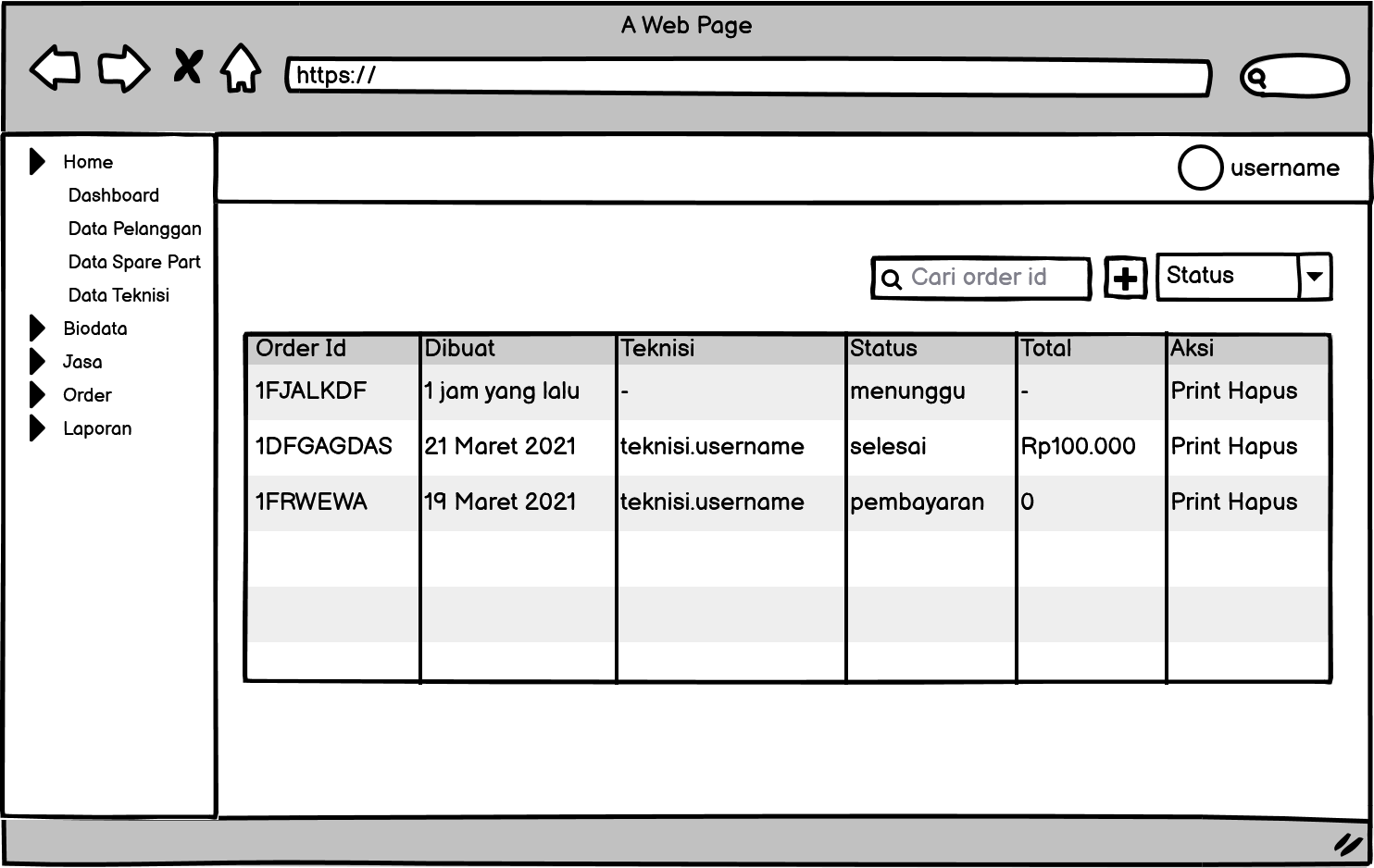
1. Halaman Tambah Jasa

Halaman Tambah Jasa berisikan kolom – kolom mengenai data jasa, dan dapat dilihat pada Gambar 4.17.



Gambar 4.17 Halaman Tambah Jasa

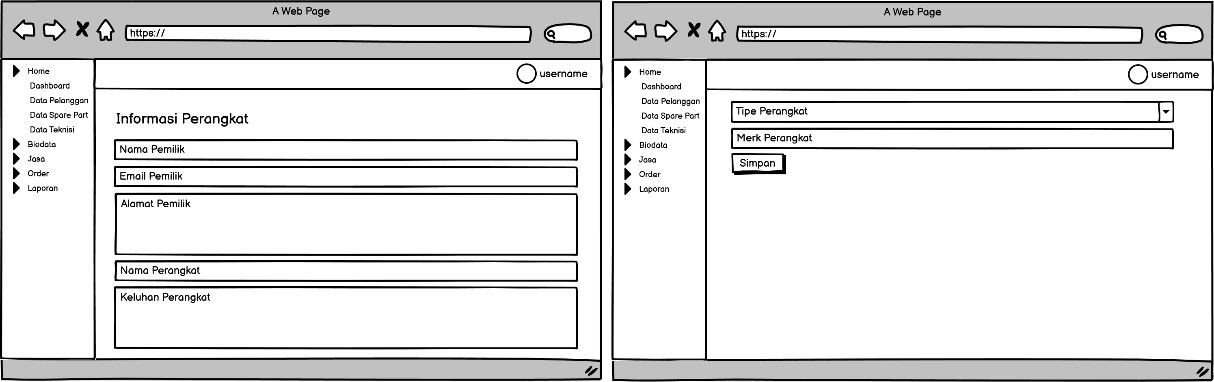
1. Halaman *Orders*

Halaman *Orders* berisikan *list* *orders* yang dikerjakan oleh teknisi pada Oneya Solutions beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.18.

Gambar 4.18 Halaman *Orders*

1. Halaman Tambah *Orders*

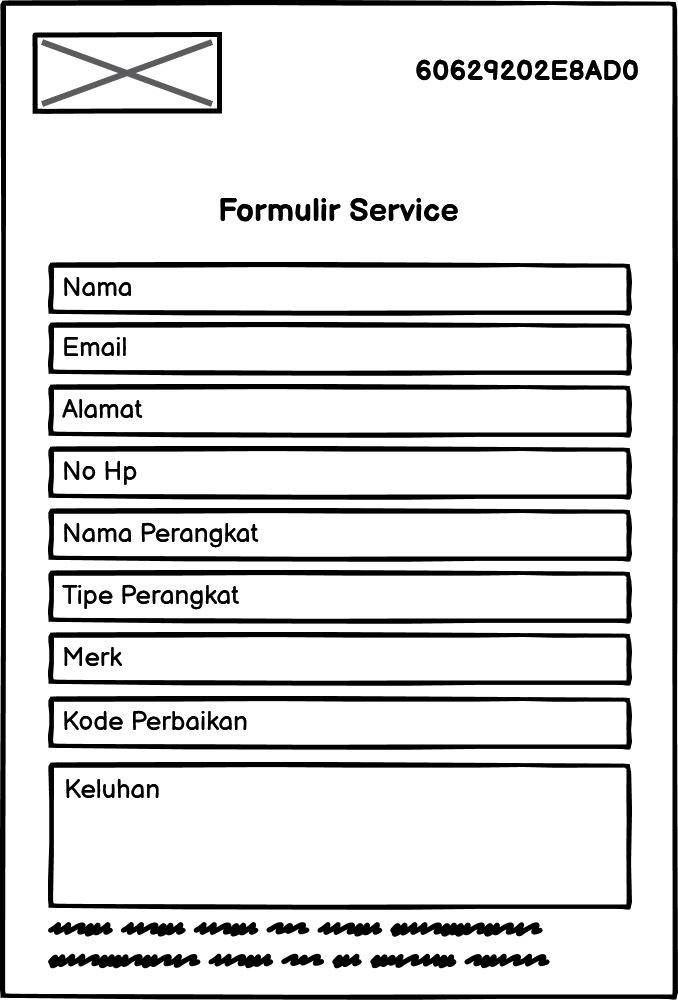
Halaman Tambah *Orders* berisikan kolom – kolom mengenai data *order*, dan dapat dilihat pada Gambar 4.19.



Gambar 4.19 Halaman Tambah *Orders*

1. Halaman *Print* *Order*

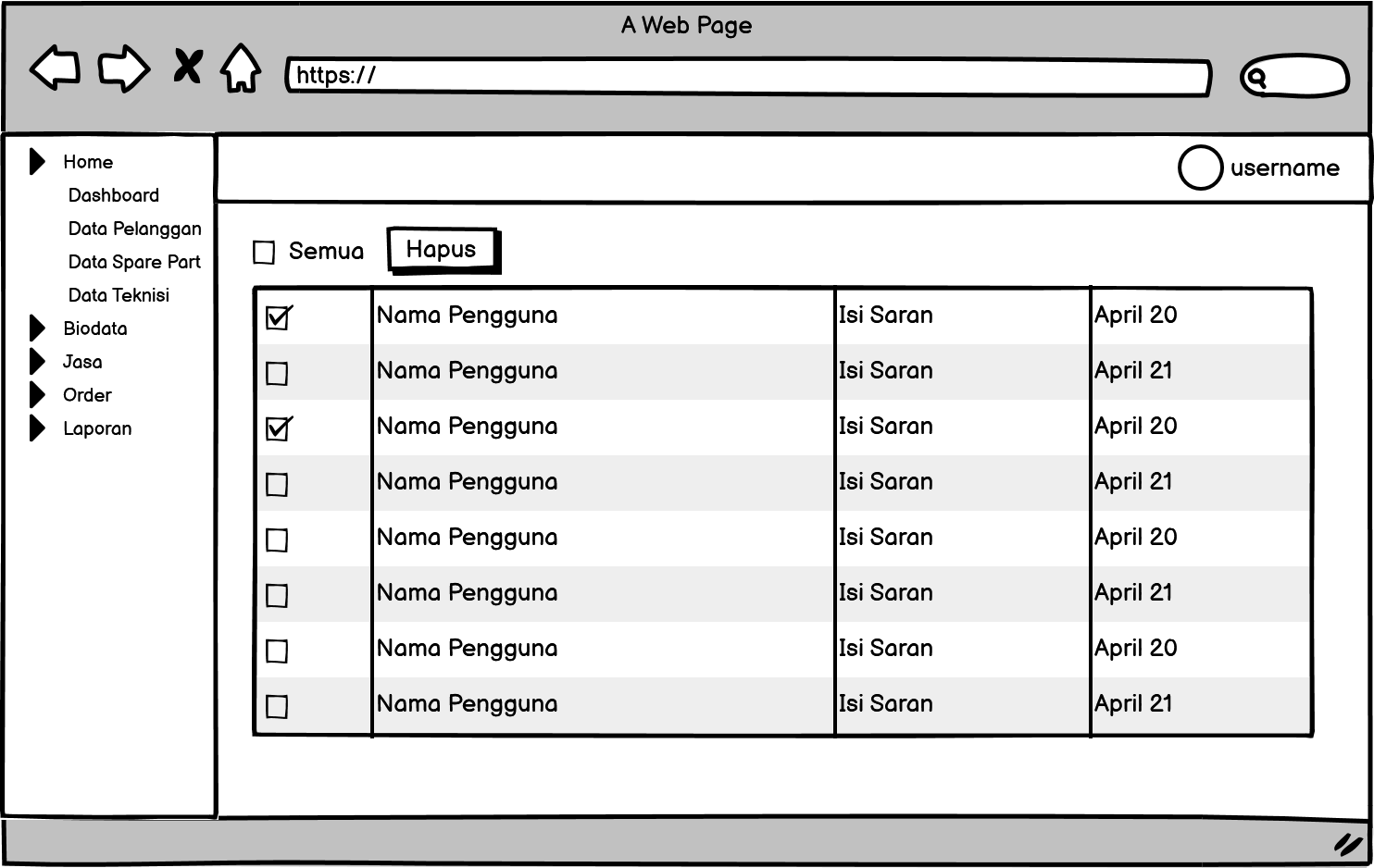
Halaman *Print Order* berisikah informasi mengenai detail order yang akan di print sebagai formulir *service*, dan dapat dilihat pada Gambar 4.20.



Gambar 4.20 Halaman *Print Order*

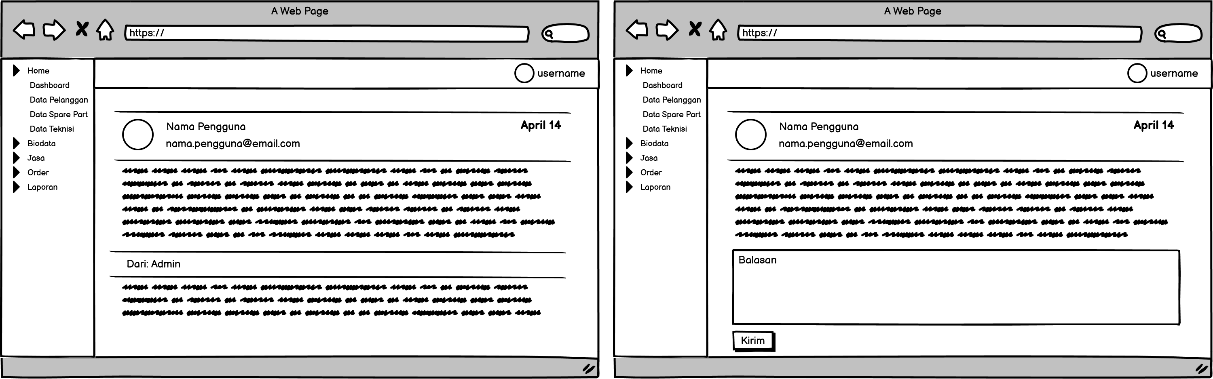
1. Halaman Saran

Halaman Saran berisikan saran – saran yang dikirimkan oleh pelanggan, dan dapat dilihat pada Gambar 4.21.



Gambar 4.21 Halaman Saran

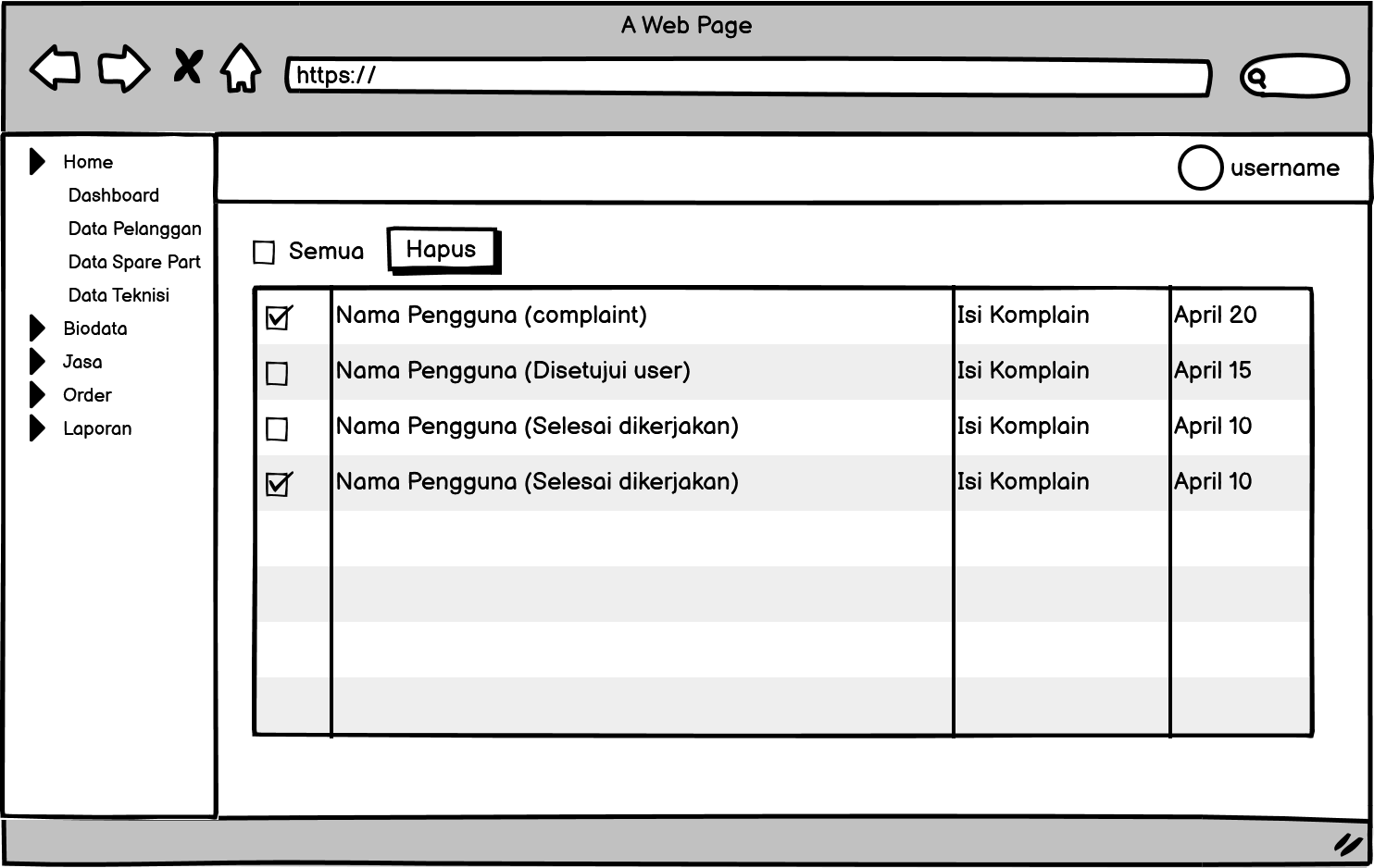
1. Halaman Balas Saran

Halaman Balas Saran merupakan halaman yang memungkinkan admin untuk membalas saran yang dikirimkan pelanggan, dan dapat dilihat pada Gambar 4.22.

Gambar 4.22 Halaman Balas Saran

1. Halaman Komplain

Halaman Komplain berisikan data mengenai komplain yang dilakukan oleh pelanggan, dan dapat dilihat pada Gambar 4.23.

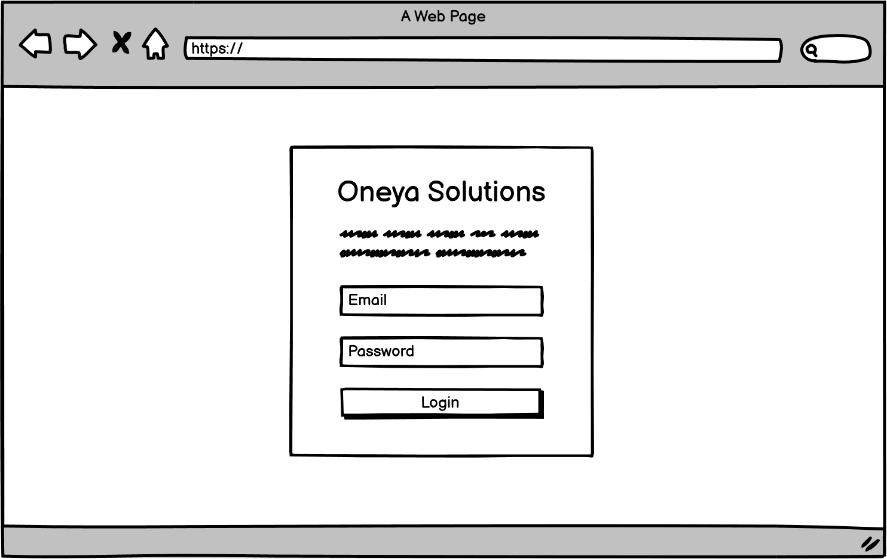


Gambar 4.23 Halaman Komplain

1. Halaman Login

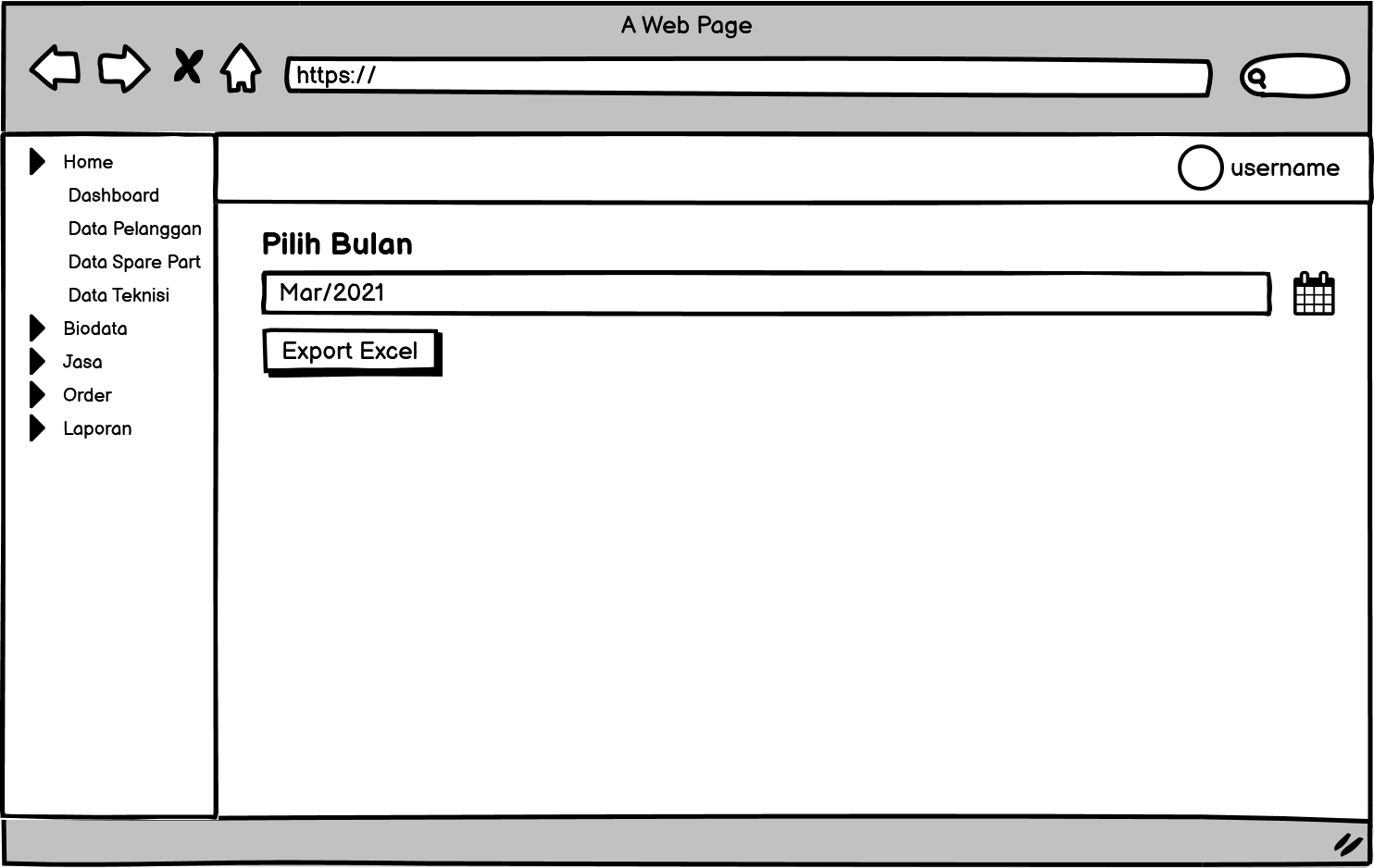
Halaman Login merupakan halaman pertama yang harus dilalui sebelum mengakses sistem, dan dapat dilihat pada Gambar 4.24.

Gambar 4.24 Halaman Login



1. Halaman Cetak Laporan

Halaman Cetak Laporan merupakan halaman yang digunakan untuk mencetak laporan keuangan, dan dapat dilihat pada Gambar 4.25.



Gambar 4.25 Halaman Cetak Laporan

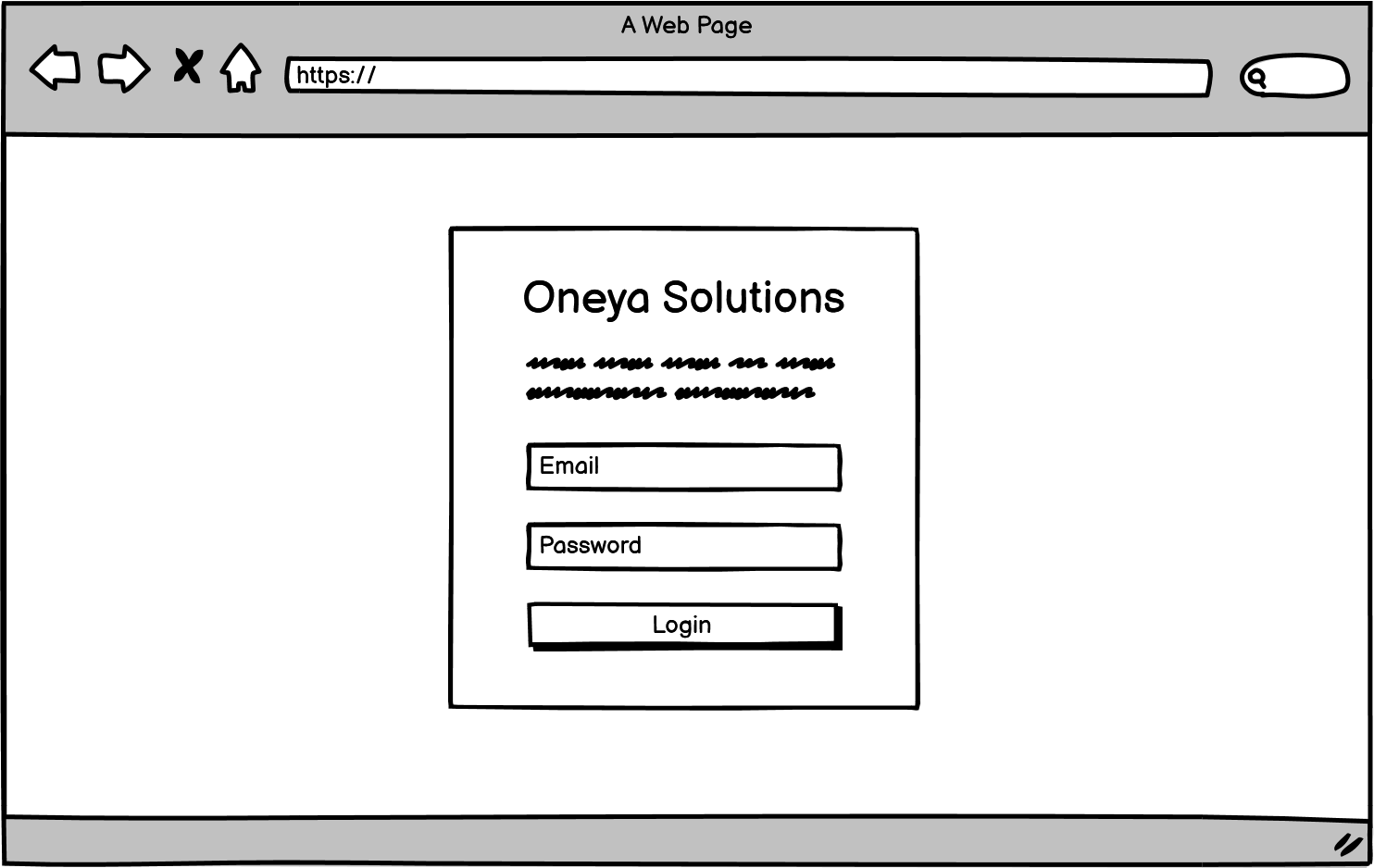
1. **Desain Antarmuka Halaman Teknisi**

Berikut merupakan rancangan desain antar muka untuk halaman yang dapat diakses oleh pengguna yang memiliki hak akses teknisi. Berikut merupakan rancangan antar muka tersebut:

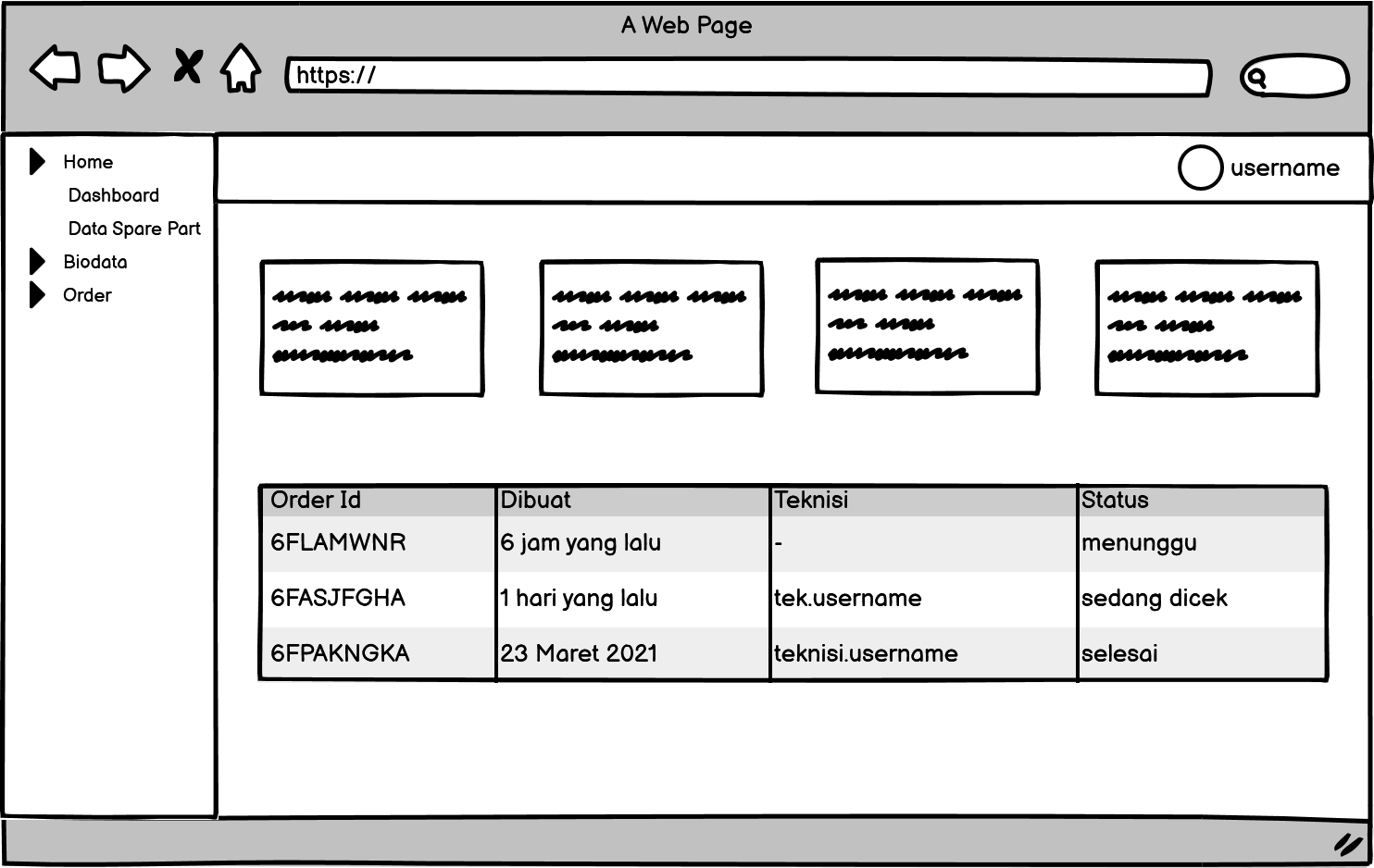
1. Halaman Login

Halaman Login merupakan halaman pertama yang harus dilalui sebelum mengakses sistem, dan dapat dilihat pada Gambar 4.24.

Gambar 4.24 Halaman Login



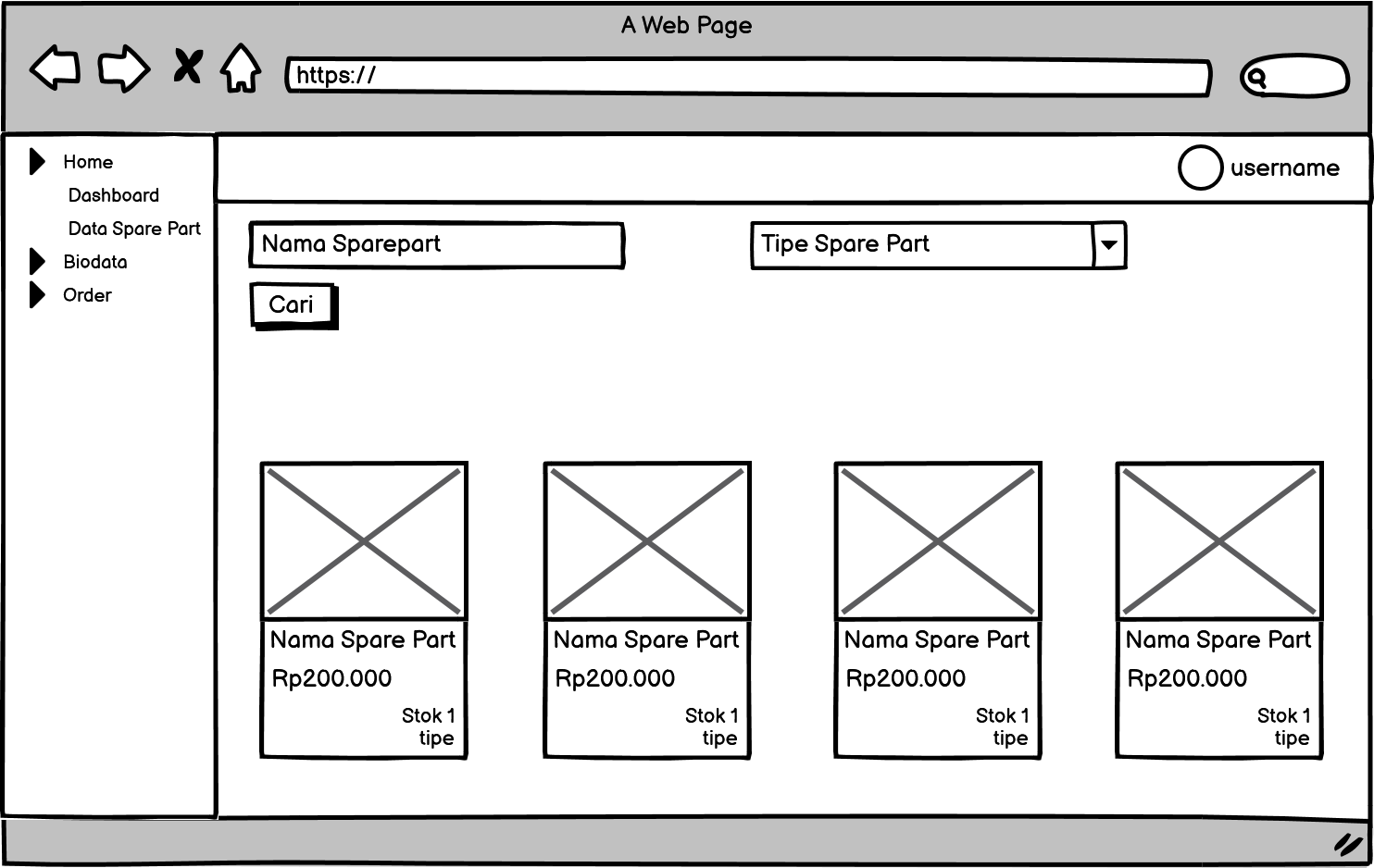
1. Halaman Dashboard

Halaman Dashboard admin dan teknisi tidak jauh berbeda, perbedaan utama dari dashboard admin dan teknisi terletak pada grafik yang ada, dan dapat dilihat pada Gambar 4.25.

Gambar 4.25 Halaman Dashboard

1. Halaman *Spare Part*

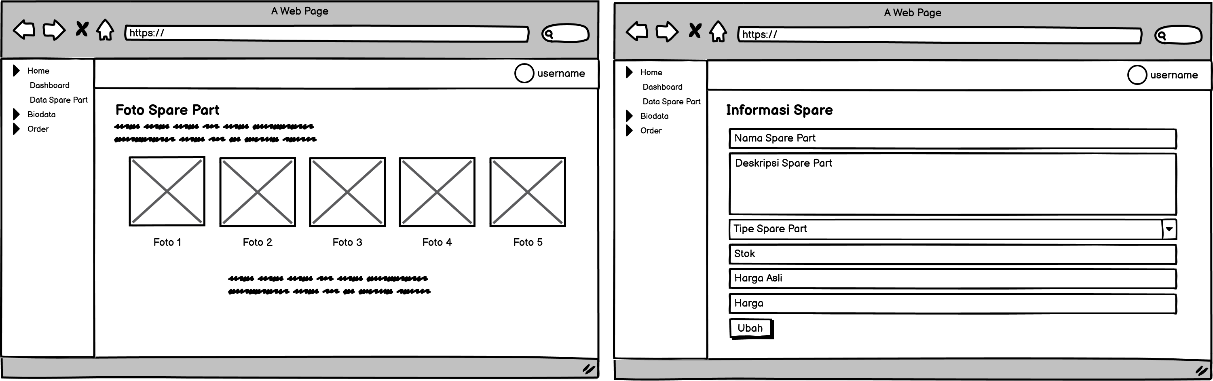
Halaman *Spare Part* berisikan *list spare part* yang tersedia digudang beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.26.



Gambar 4.26 Halaman *Spare Part*

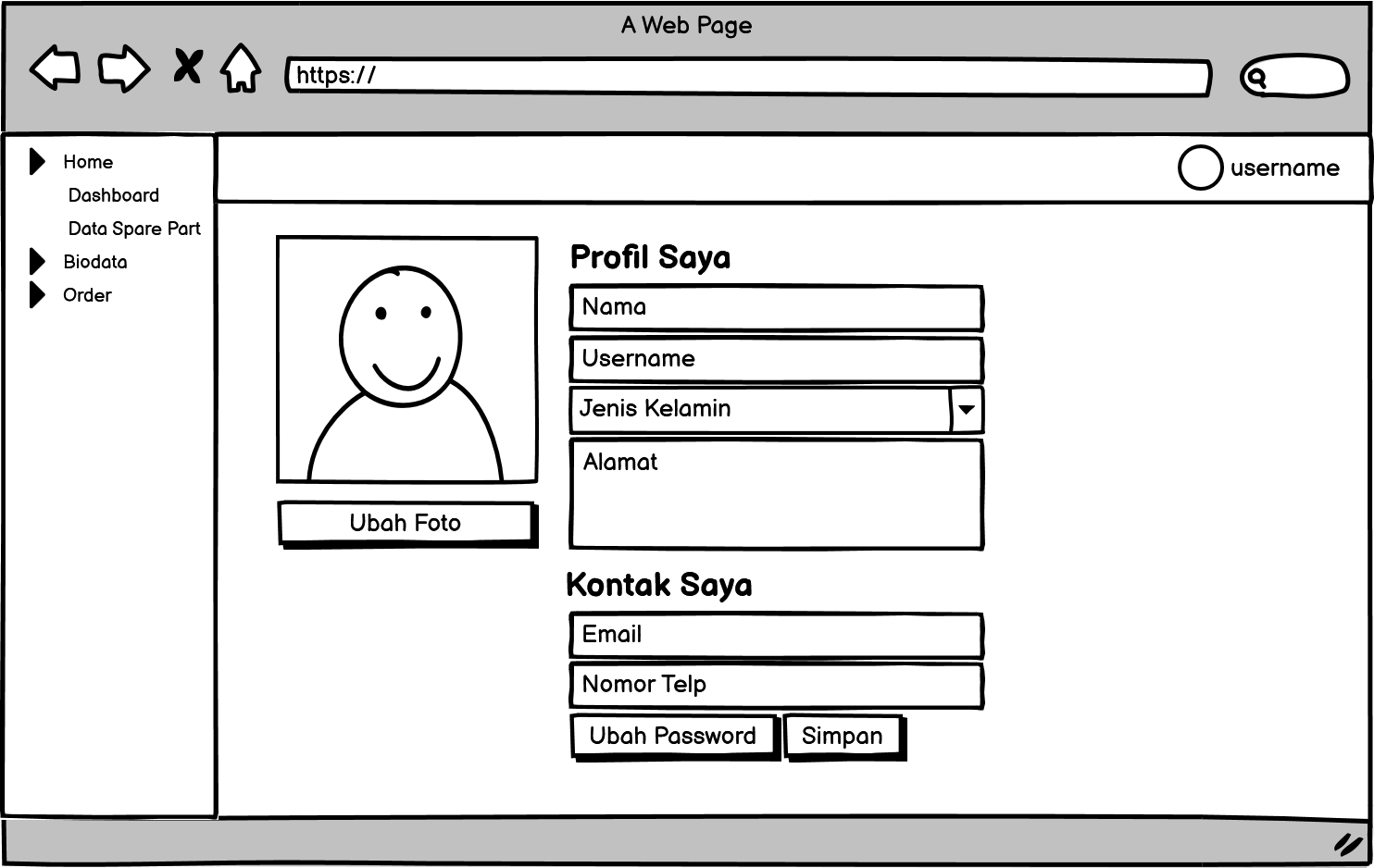
1. Halaman Detail *Spare Part*

Halaman Detail *Spare Part* berisikan tentang detail dari *spare part*, dan dapat dilihat pada Gambar 4.27.



Gambar 4.26 Halaman Detail *Spare Part*

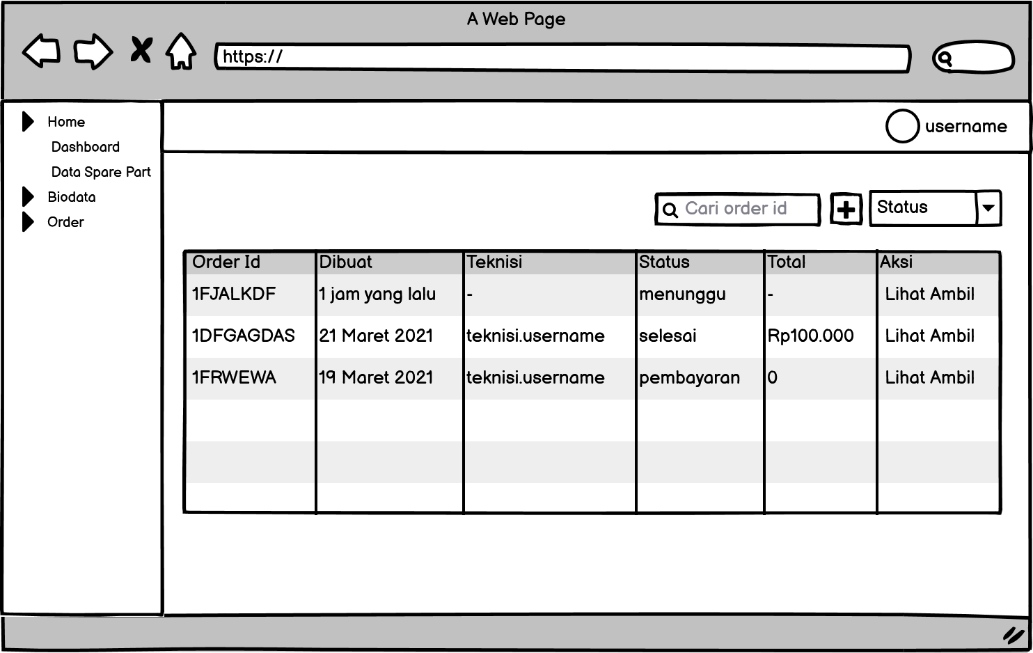
1. Halaman Biodata

Halaman Biodata berisikan data – data biodata dari teknisi, pada halaman biodata, teknisi juga dapat melakukan perubahan terhadap biodata dirinya seperti biodata profil serta biodata kontak, dan dapat dilihat pada Gambar 4.27.

Gambar 4.27 Halaman Biodata

1. Halaman *Orders*

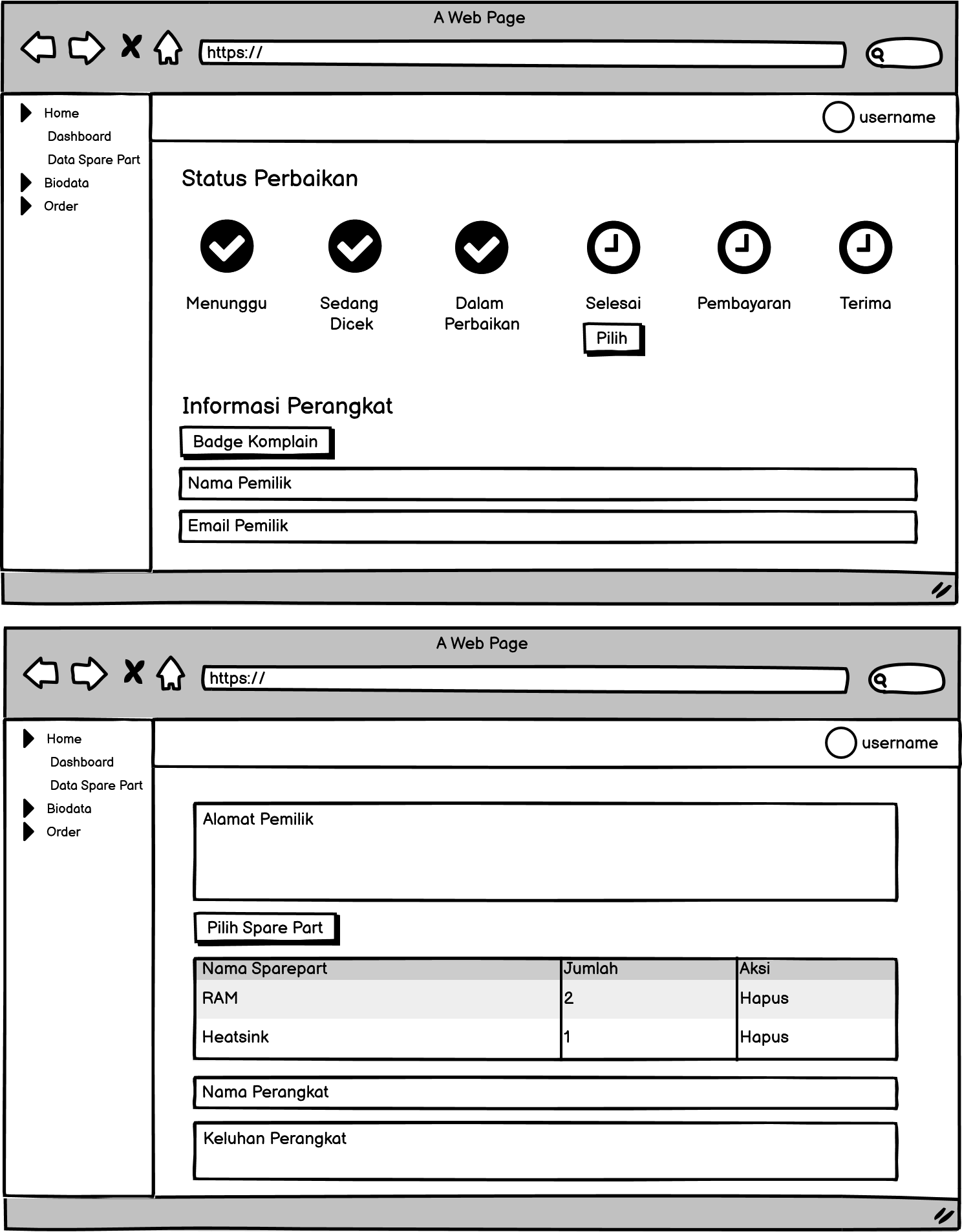
Halaman *Orders* berisikan *list* *orders* yang tersedia maupun dikerjakan oleh teknisi beserta beberapa informasinya, dan dapat dilihat pada Gambar 4.28.



Gambar 4.29 Halaman *Orders*

1. Halaman Detail *Orders*

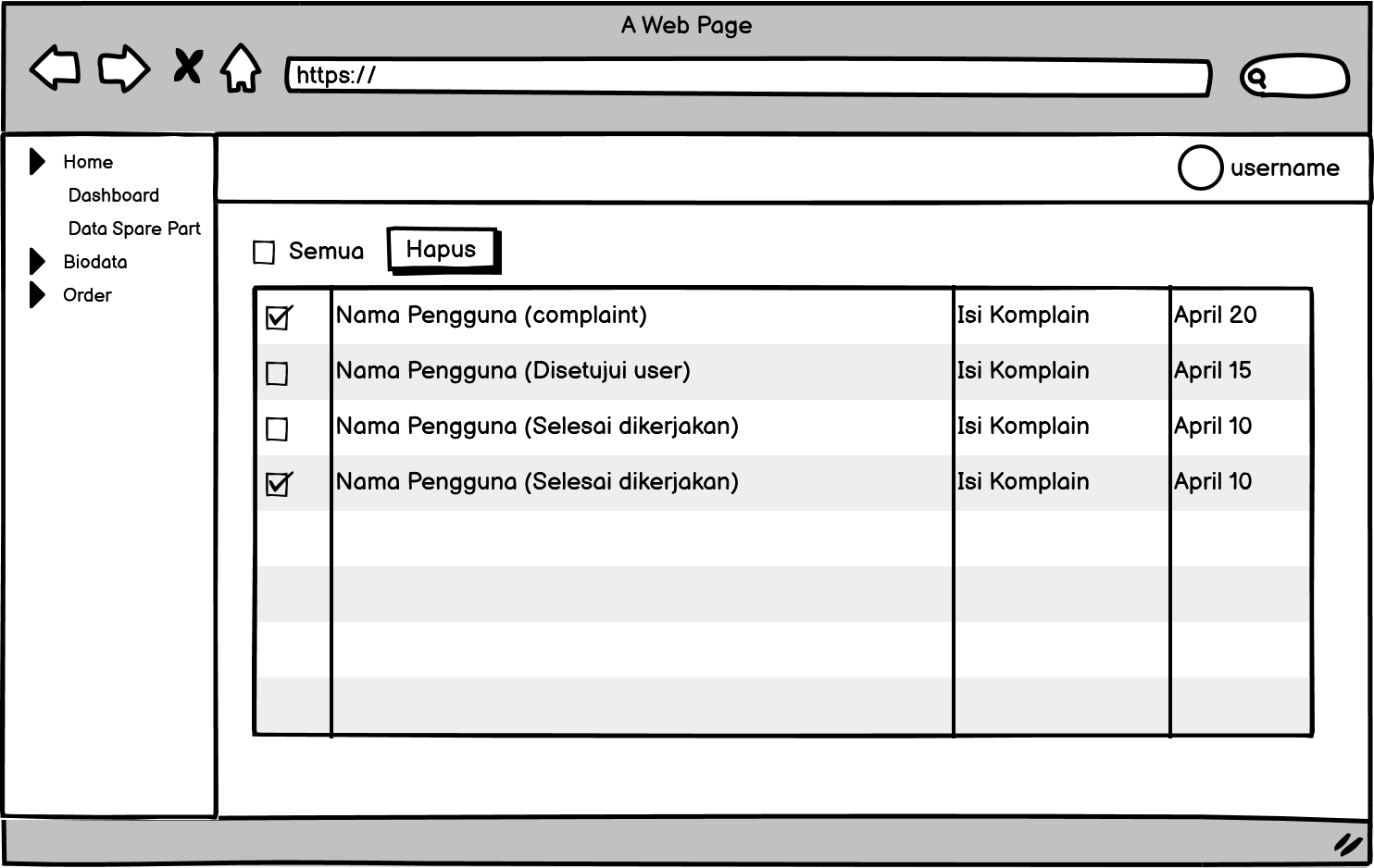
Pada Halaman Detail *Orders* teknisi dapat mengubah status order dan menambah maupun mengubah spare part yang digunakan dalam proses perbaikan, dan dapat dilihat pada Gambar 4.30.



Gambar 4.30 Halaman Detail *Orders*

1. Halaman Komplain

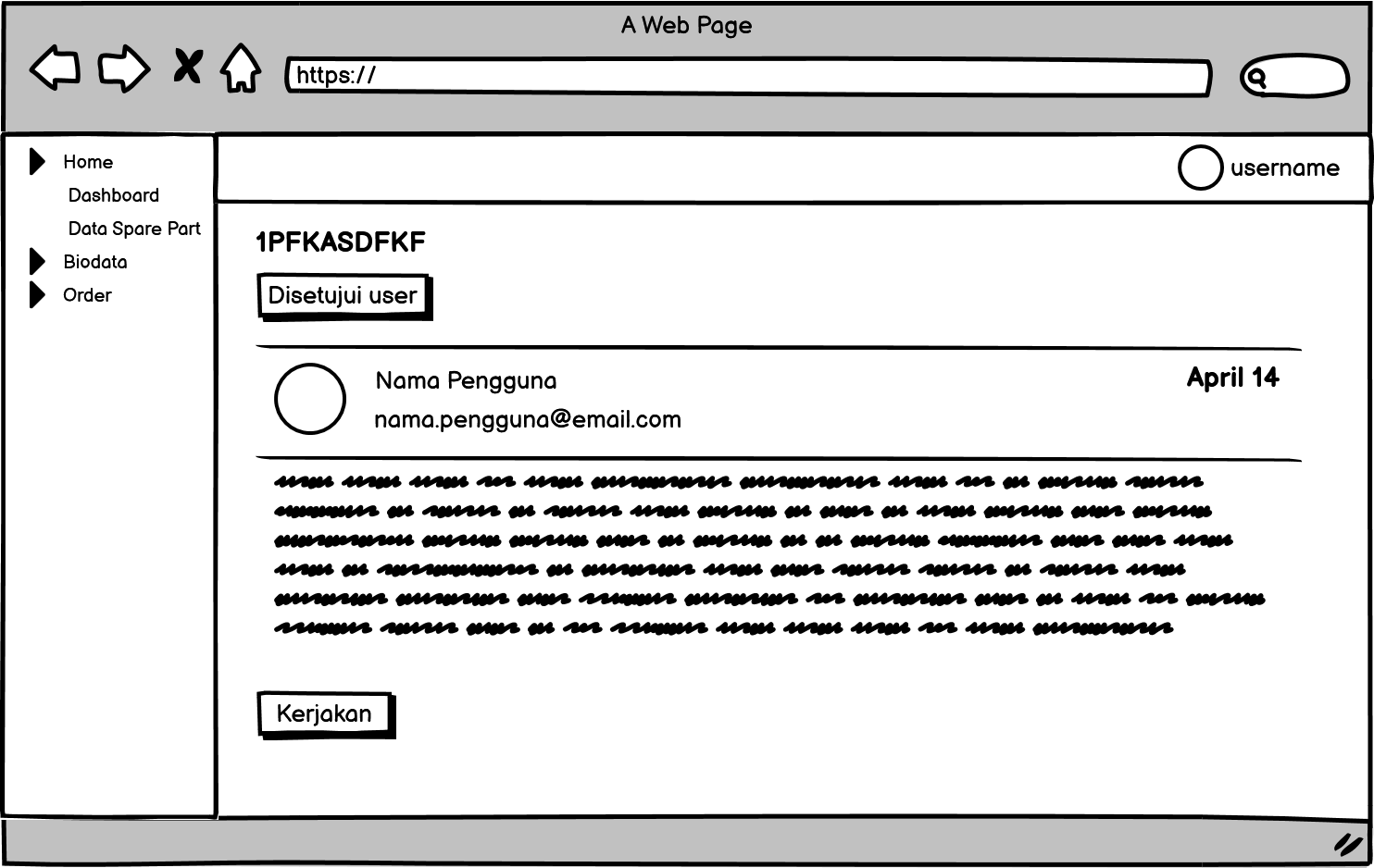
Halaman Komplain berisikan komplain – komplain dari pelanggan serta harus dikerjakan teknisi, dan dapat dilihat pada Gambar 4.31.



Gambar 4.31 Halaman Komplain

1. Halaman Detail Komplain

Halaman Detail Komplain berisikan data komplain dari pelanggan serta teknisi dapat mengubah status komplain, dan dapat dilihat pada Gambar 4.32.



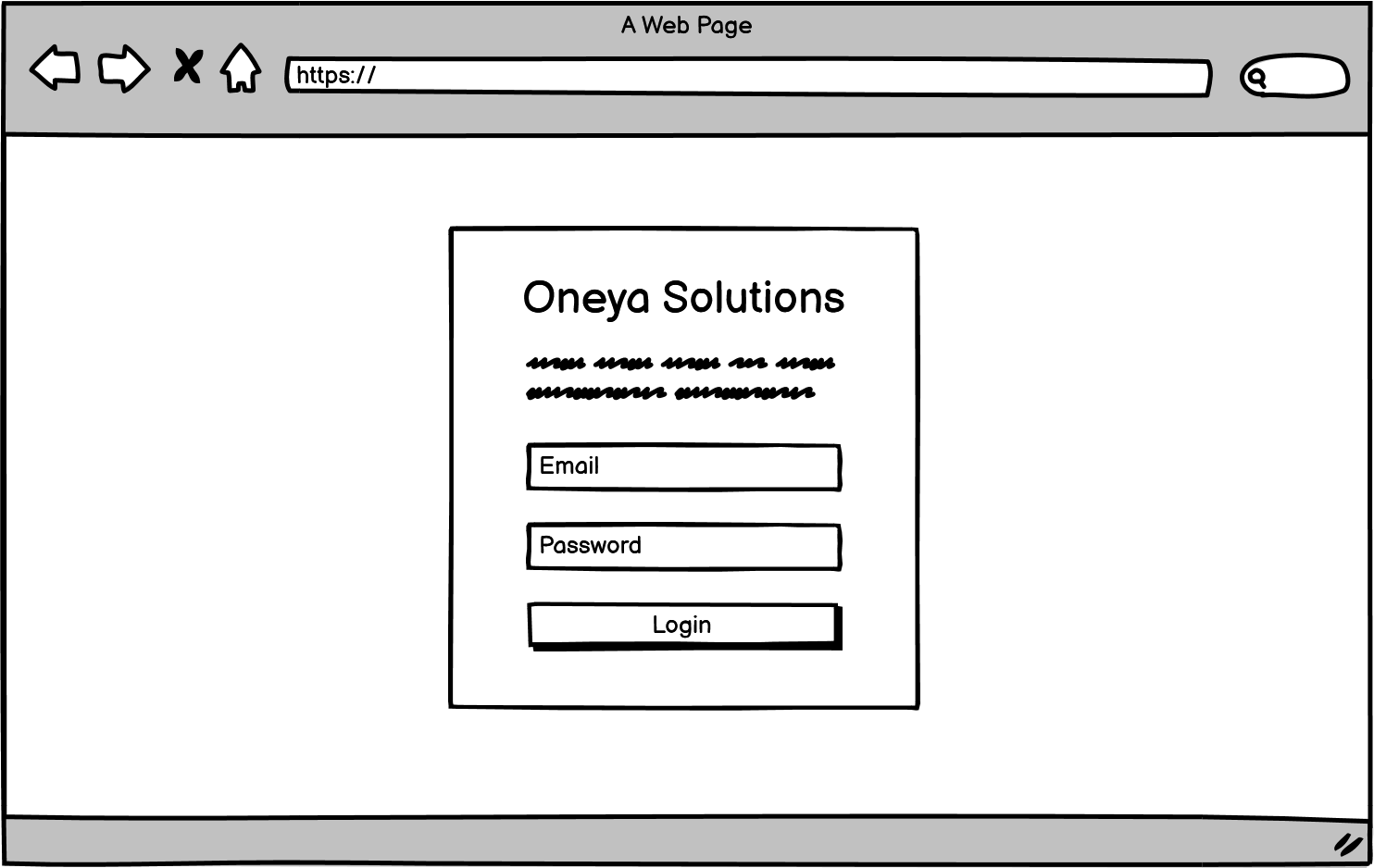
Gambar 4.32 Halaman Detail Komplain

1. **Desain Antarmuka Halaman User**

Berikut merupakan rancangan desain antar muka untuk halaman yang dapat diakses oleh pengguna yang memiliki hak akses user. Berikut merupakan rancangan antar muka tersebut:

1. Halaman Login

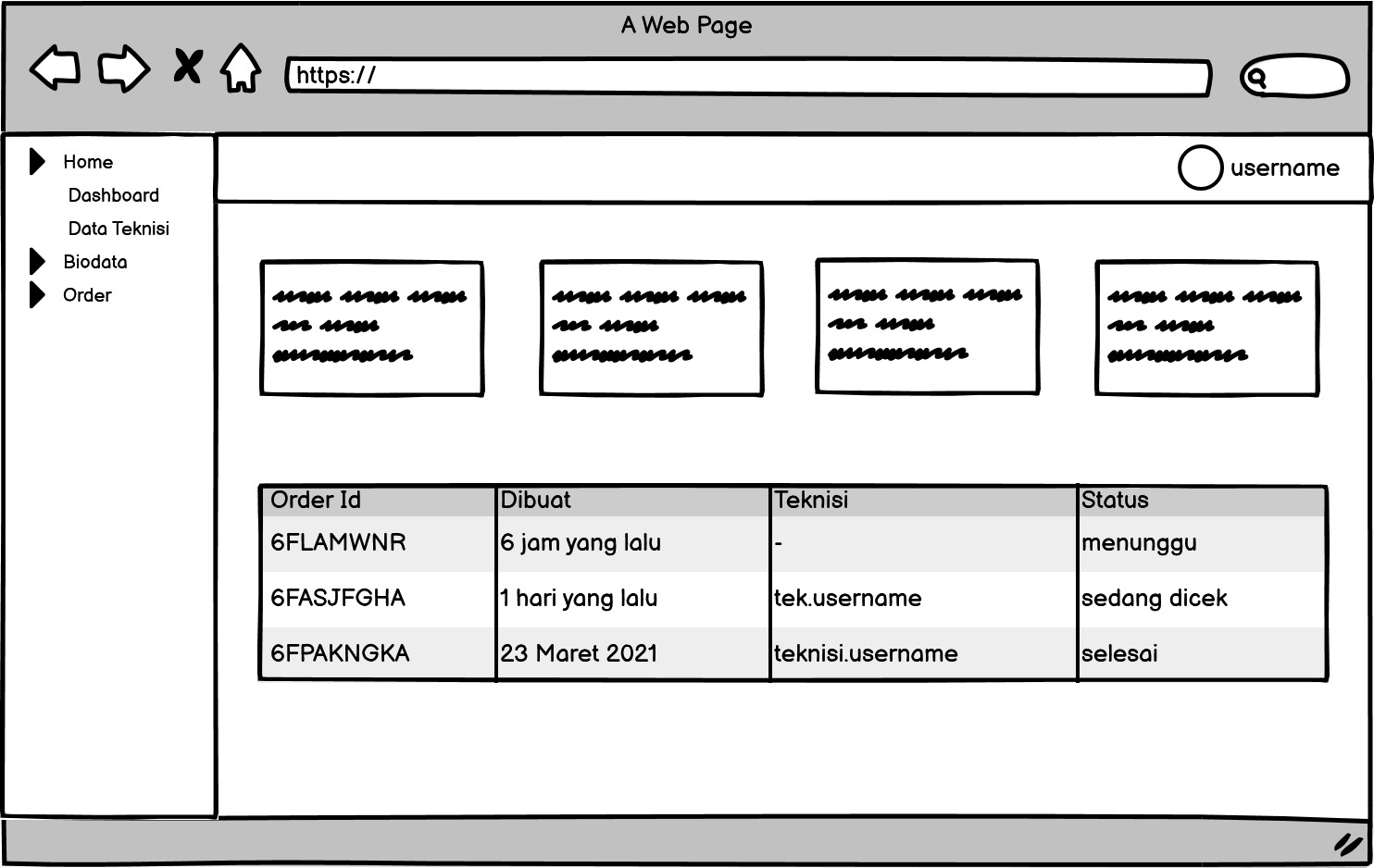
Halaman Login merupakan halaman pertama yang harus dilalui sebelum mengakses sistem, dan dapat dilihat pada Gambar 4.33.



Gambar 4.33 Halaman Login

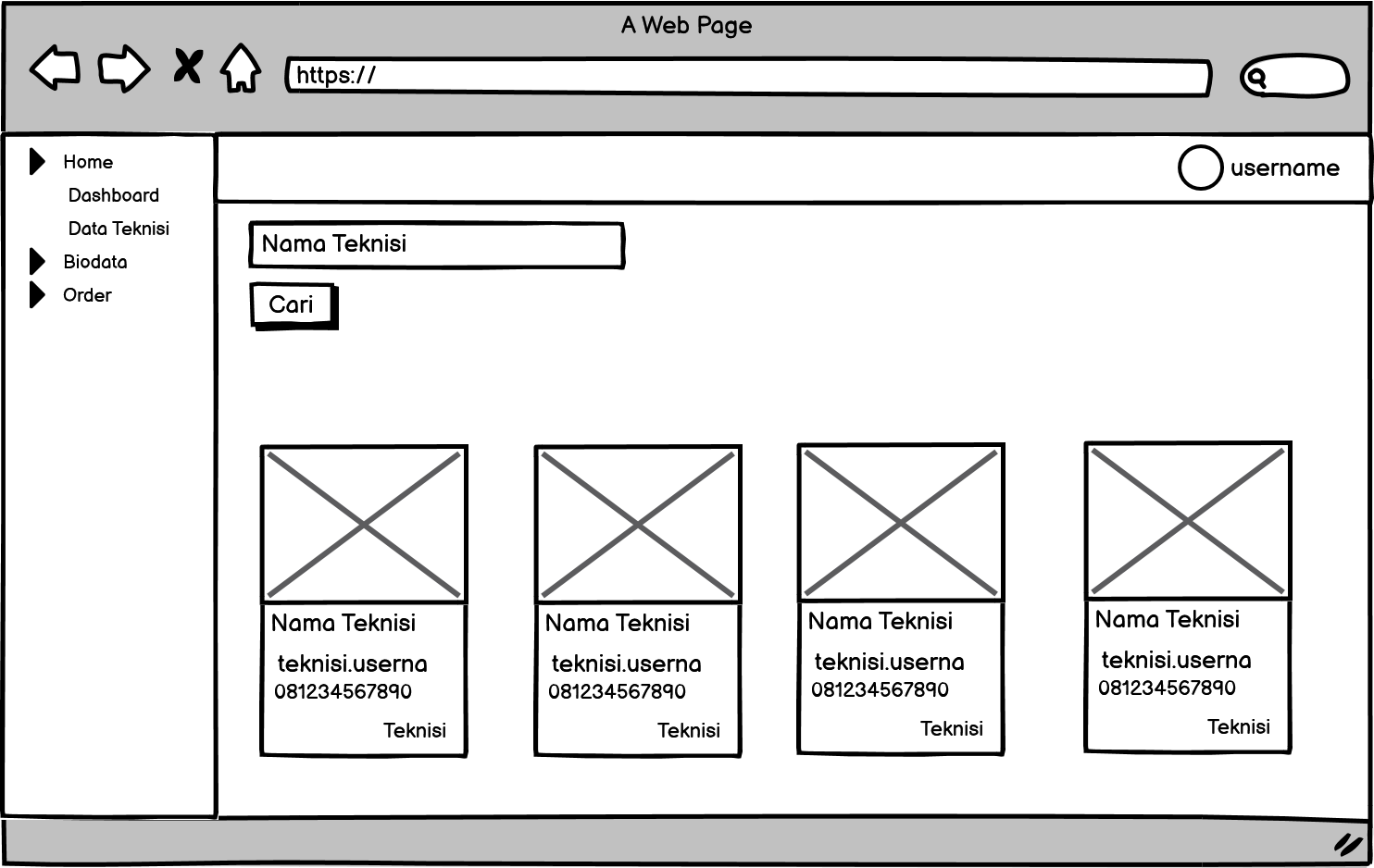
1. Halaman Dashboard

Halaman Dashboard pada hak akses user hanya menampilkan daftar *order* serta jasa yang tersedia di Oneya Solutions, dan dapat dilihat pada Gambar 4.34.



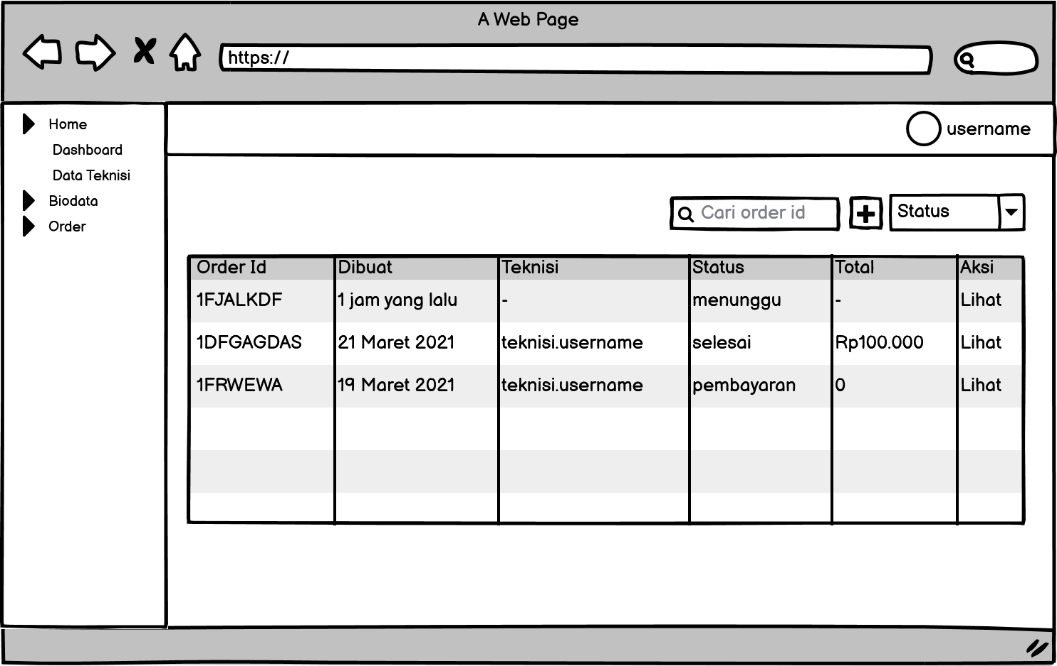
Gambar 4.34 Halaman Dashboard

1. Halaman Teknisi

Halaman Teknisi merupakan halaman yang menampilkan *list* teknisi yang bekerja di Oneya Solutions, hak akses user hanya dapat melihat teknisi tanpa melakukan aksi yang lain, dan dapat dilihat pada Gambar 4.35.

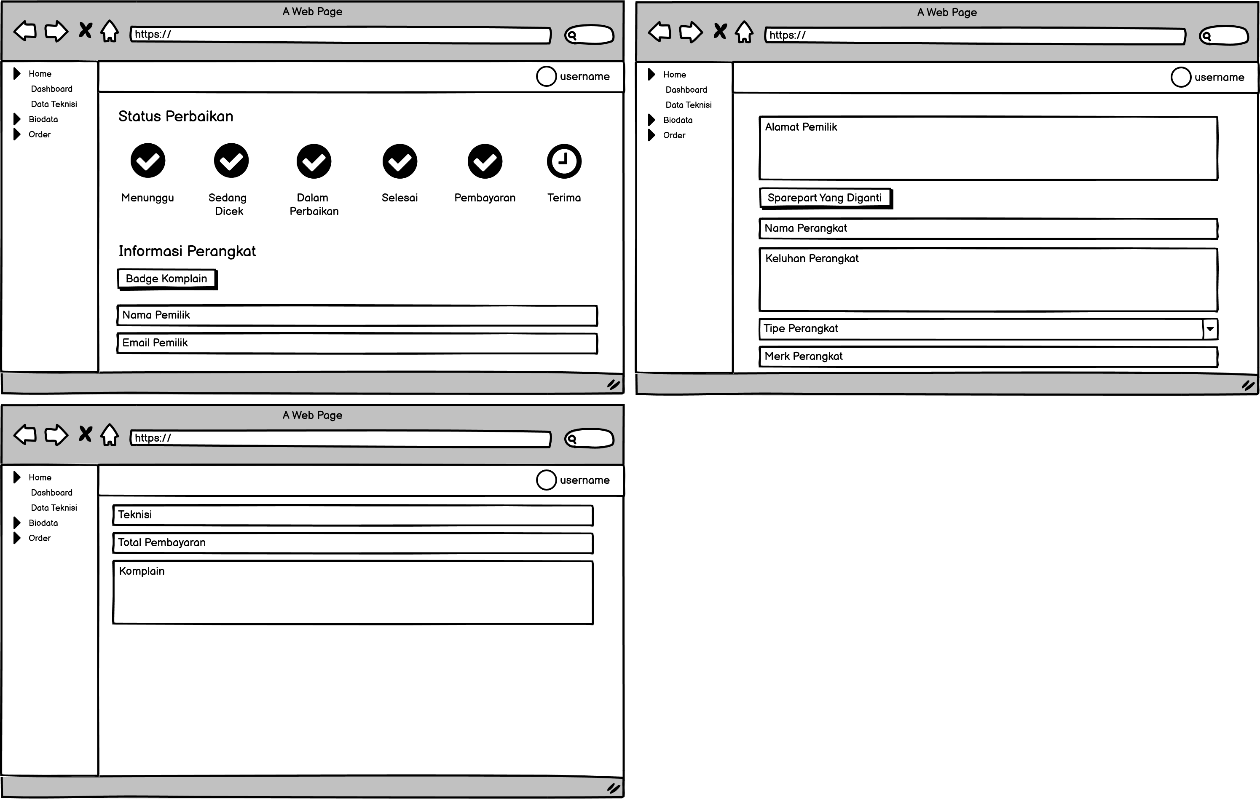
Gambar 4.35 Halaman Teknisi

1. Halaman *Orders*

Halaman Orders menampilkan daftar order yang dimiliki oleh *user*, dan dapat dilihat pada Gambar 4.36.

Gambar 4.36 Halaman *Orders*

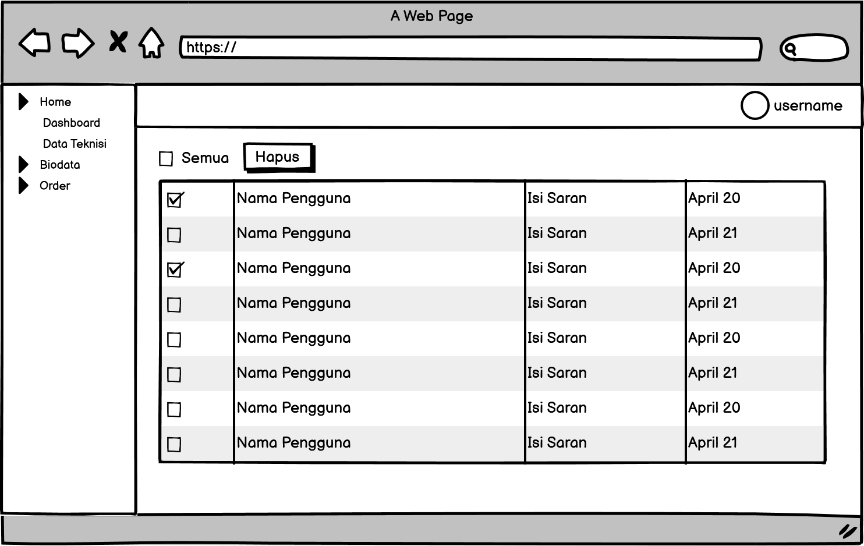
1. Halaman Detail *Orders*

Halaman Detail *Orders* berisikan data mengenai detail *order*, dan dapat dilihat pada Gambar 4.37.

Gambar 4.37 Halaman Detail *Orders*

1. Halaman Saran

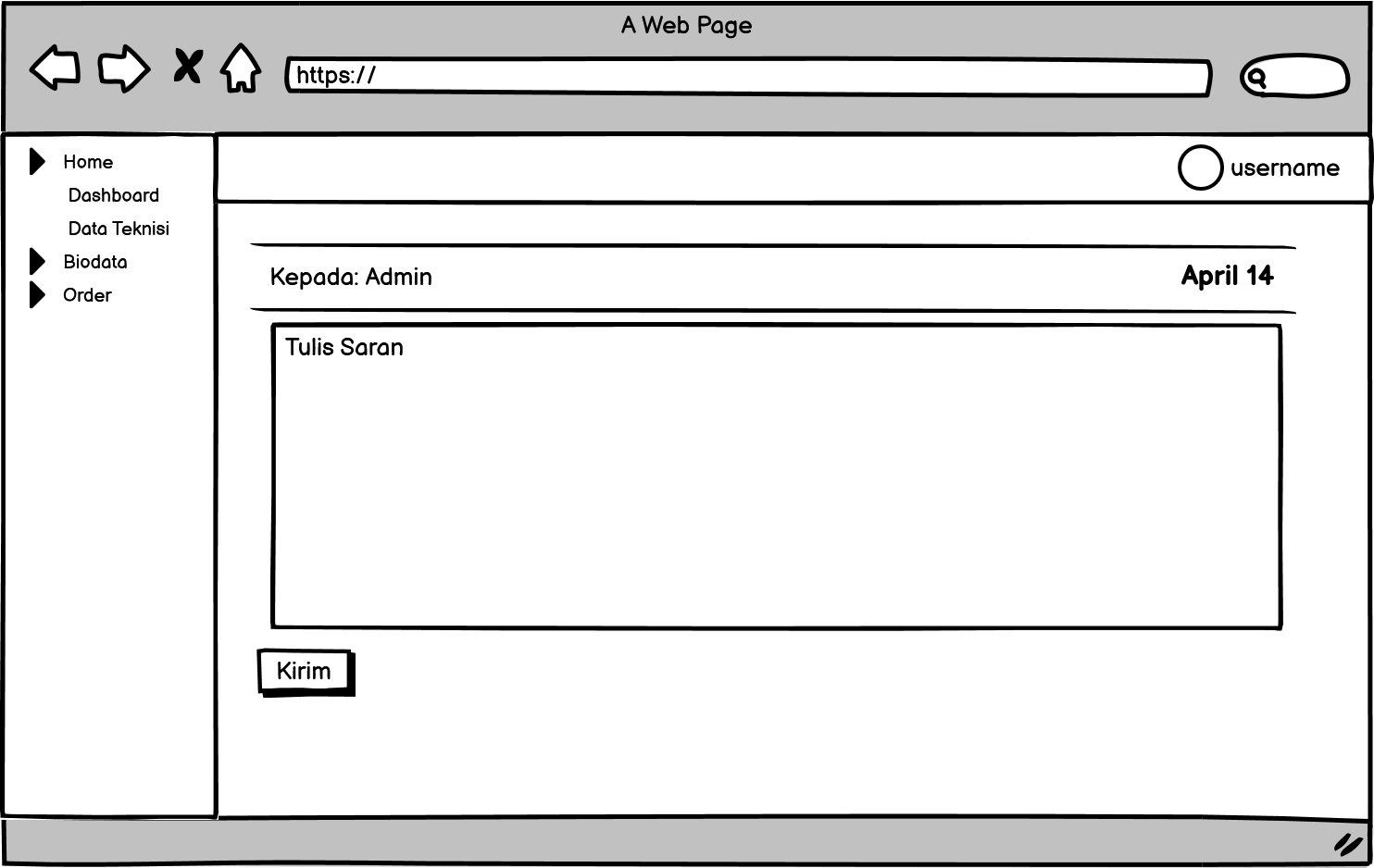
Halaman Saran berisikan saran – saran yang dikirimkan oleh user, dan dapat dilihat pada Gambar 4.38.



Gambar 4.38 Halaman Saran

1. Halaman Kirim Saran

Halaman Kirim Saran merupakan halaman yang diperuntukkan kepada user untuk mengirim saran, dan dapat dilihat pada Gambar 4.39.



Gambar 4.39 Halaman Kirim Saran

1. Halaman Komplain

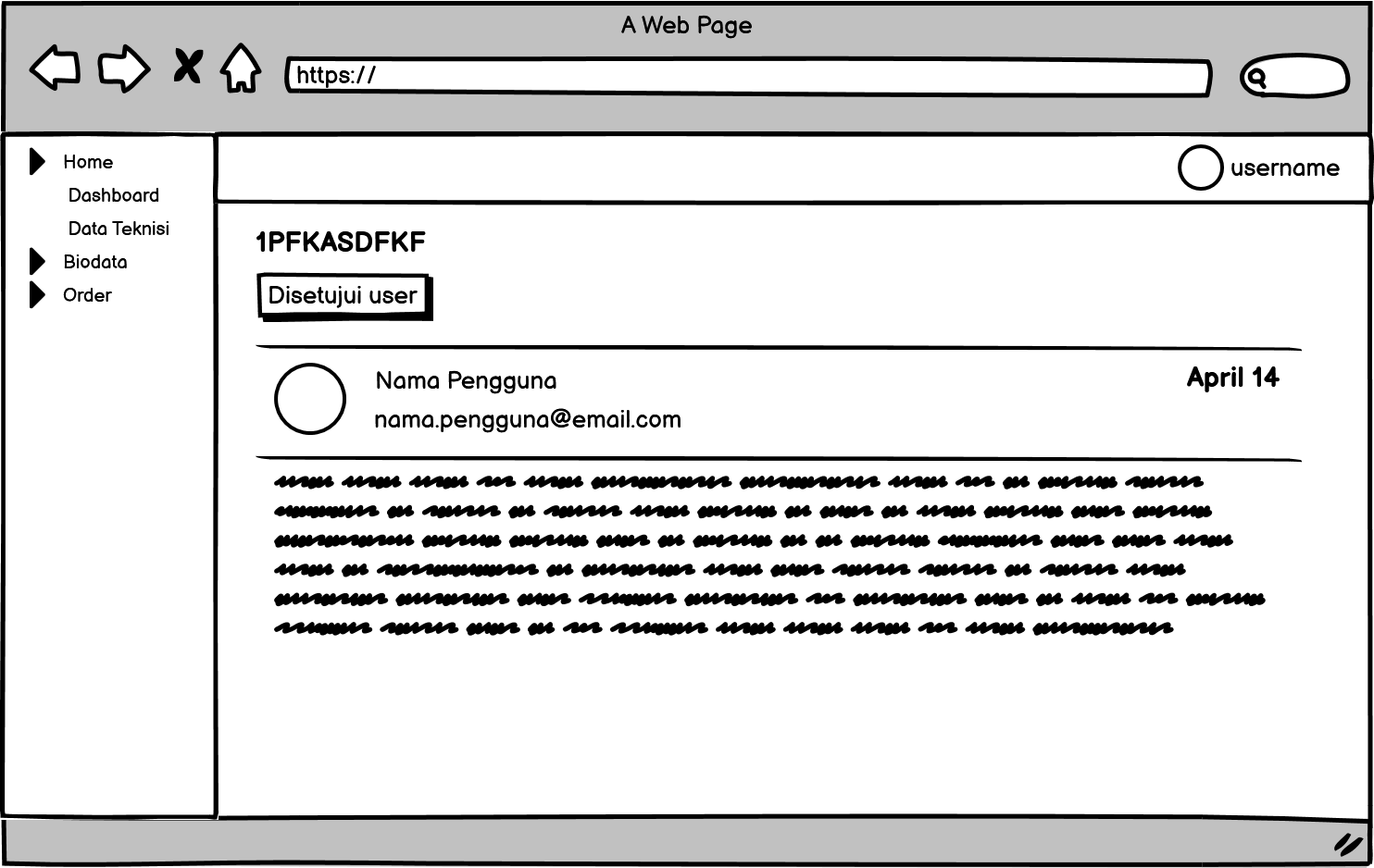
Halaman Komplain berisikan komplain yang telah dikirimkan user, dan dapat dilihat pada Gambar 4.40.



Gambar 4.40 Halaman Komplain

1. Halaman Detail Komplain

Halaman Detail Komplain berisikan detail komplain yang dikirimkan user, dan dapat dilihat pada Gambar 4.41.



Gambar 4.41 Halaman Detail Komplain

### Implementasi Sistem

Implementasi Sistem adalah penjabaran hasil yang telah dikerjakan dalam tahapan sebelumnya dimulai dari tahap analisa hingga perancangan antarmuka. Dalam sistem ini melibatkan 3 pengguna yaitu Admin, Teknisi dan *User*.

* + 1. **Halaman Admin**

Berikut merupakan implementasi sistem untuk halaman – halaman yang dapat diakses oleh pengguna dengan hak akses Admin. Implementasi tersebut diantaranya :

1. Halaman *Login*

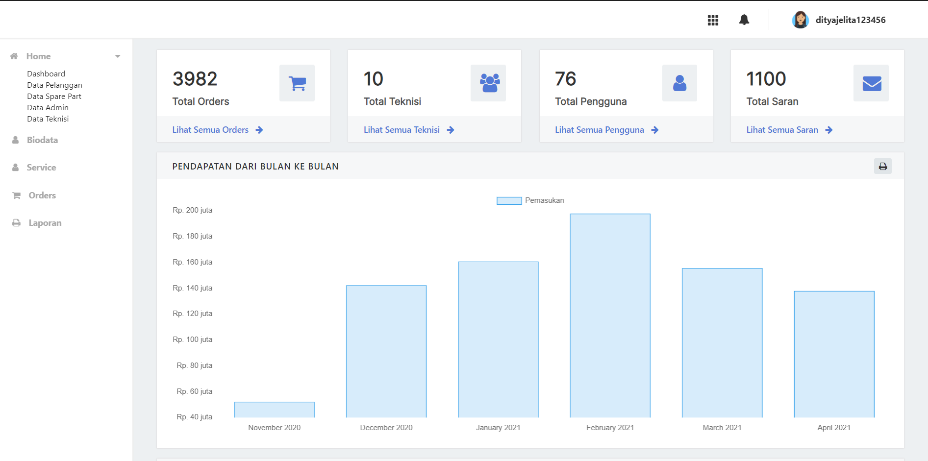
Pada halaman *login* terdiri atas 2 (dua) masukan yaitu *email* dan *password* serta sebuah tombol masuk. Pada halaman *login* ini pengguna yang akan mengakses sistem diharuskan memasukkan *email* dan *password*.



Gambar 4.42 Halaman *Login*

1. Halaman *Dashboard*

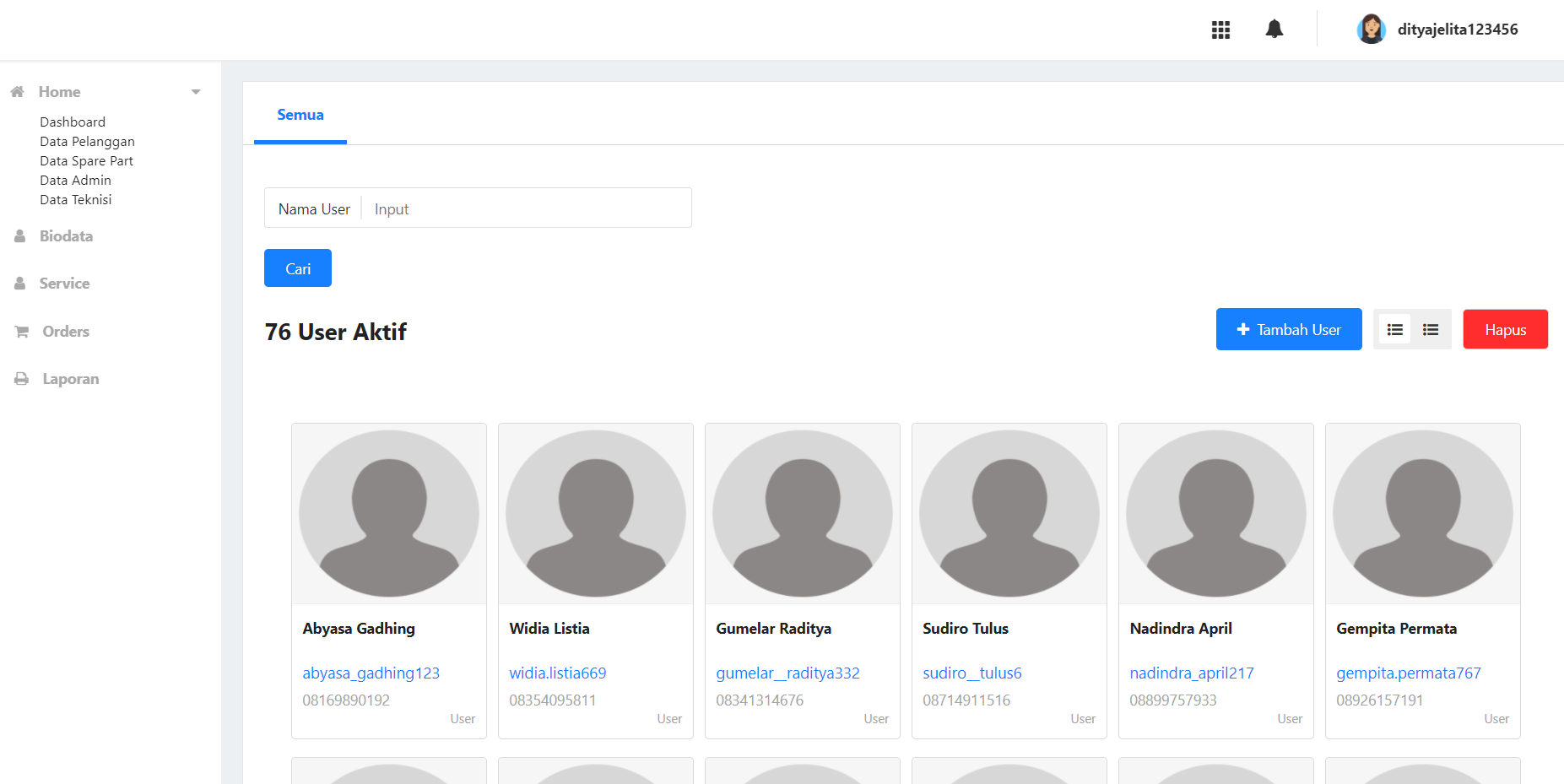
Halaman *Dashboard* merupakan halaman utama setelah melakukan *login*. Pada halaman dashboard, admin dapat melihat grafik pendapatan, jumlah total *orders*, jumlah total teknisi, jumlah total pengguna, jumlah total saran, serta dapat mengakses menu – menu yang tersedia di dalamnya antara lain *sub* *menu* *dashboard*, *sub* *menu* *data* pelanggan, *sub* *menu* *data* *spare* *part*, *sub* *menu* *data* admin, *sub* *menu* *data* teknisi, menu biodata, *menu* *service*, *menu* *orders* serta *menu* laporan dan dapat dilihat pada Gambar 4.43.



Gambar 4.43 Halaman *Dashboard*

1. HalamanPelanggan

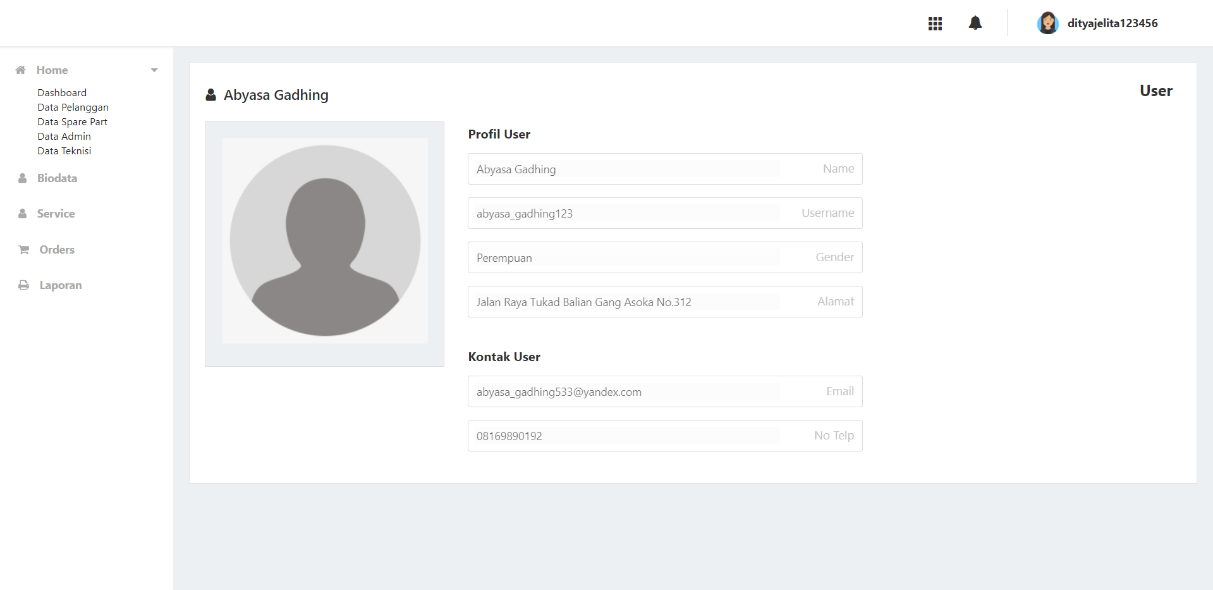
Halaman Pelanggan merupakan halaman yang berisikan daftar pelanggan yang terdata dalam sistem beserta kontaknya dan dapat dilihat pada Gambar 4.44 Halaman Pelanggan.



Gambar 4.44 Halaman Pelanggan

1. Halaman DetailPelanggan

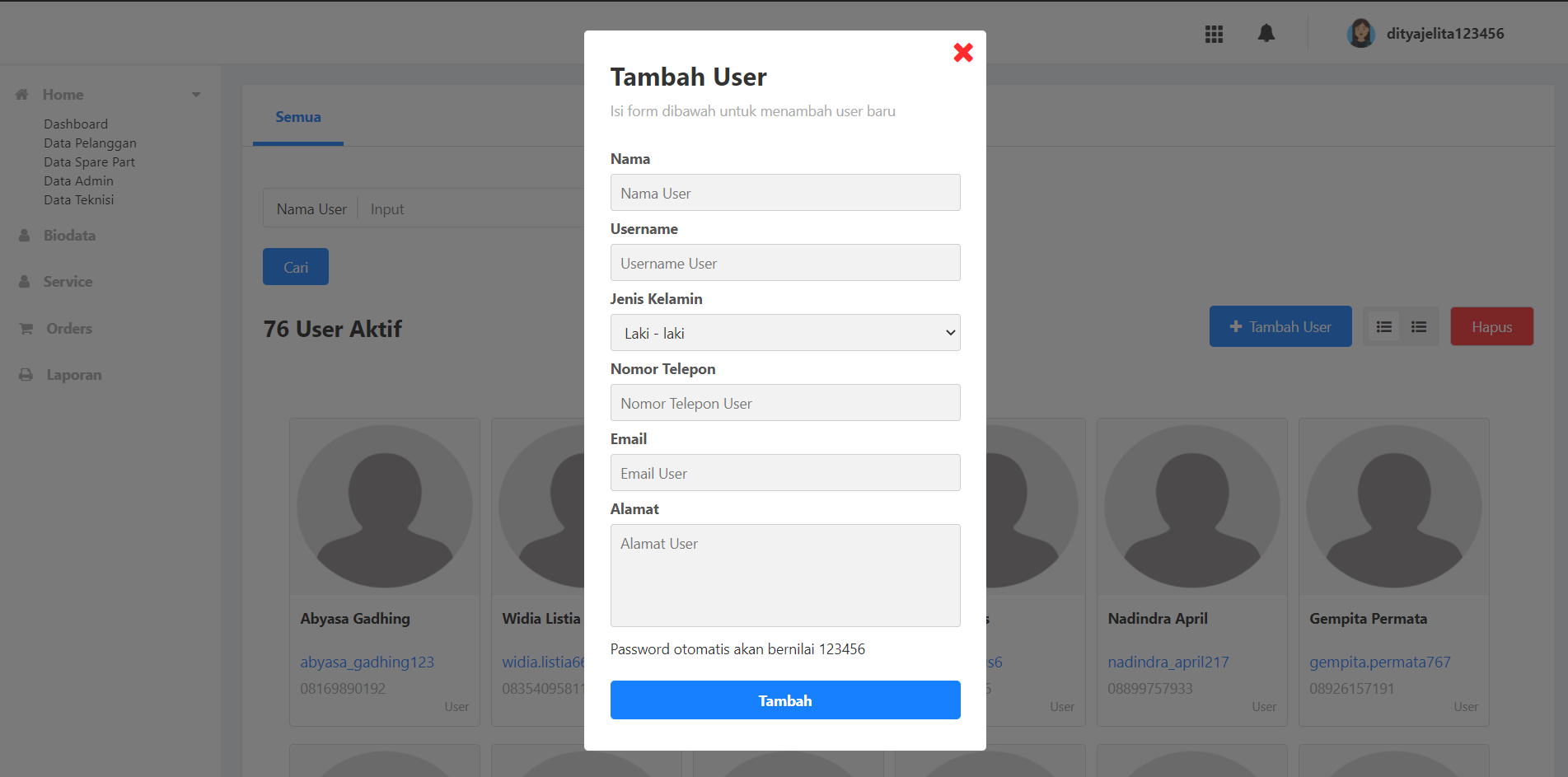
Halaman Detail Pelanggan merupakan halaman yang menampilkan data lengkap dari pelanggan seperti nama, *username*, jenis kelamin, alamat, email serta nomor telepon dan dapat dilihat pada Gambar 4.45.



Gambar 4.46 Halaman Lihat Pelanggan

1. Form Tambah *User*

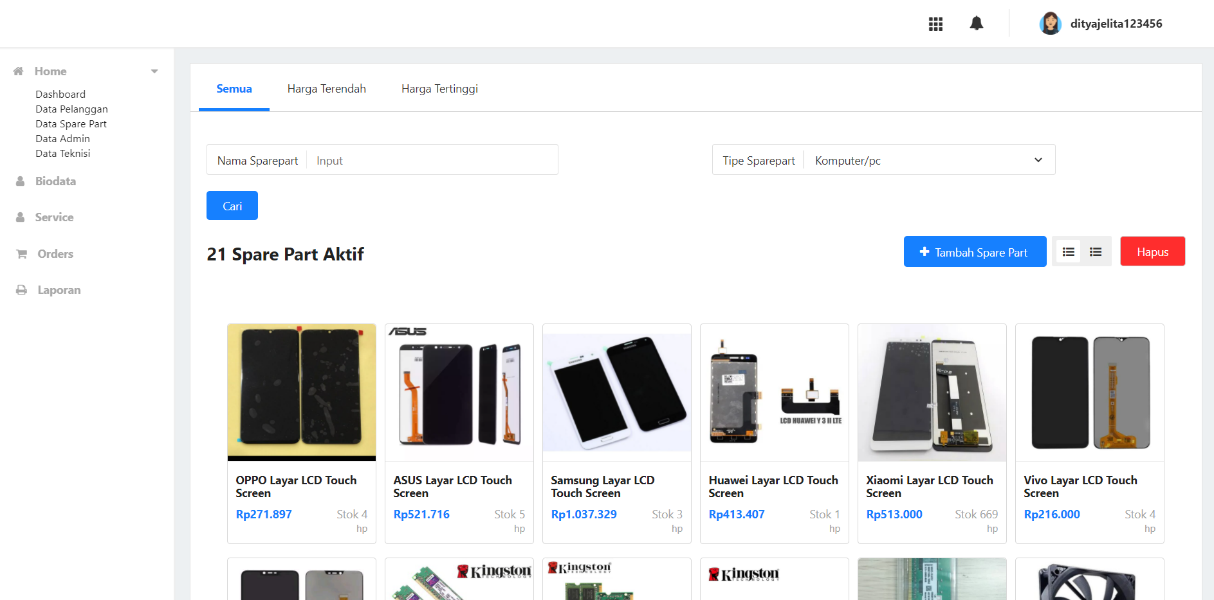
Form Tambah *User* berisi kolom – kolom mengenai data dari pelanggan sepert nama, *username*, jenis kelamin, nomor telepon email serta alamat pelanggan dan dapat dilihat pada Gambar 4.47 Form Tambah *User*.



Gambar 4.47 Form Tambah *User*

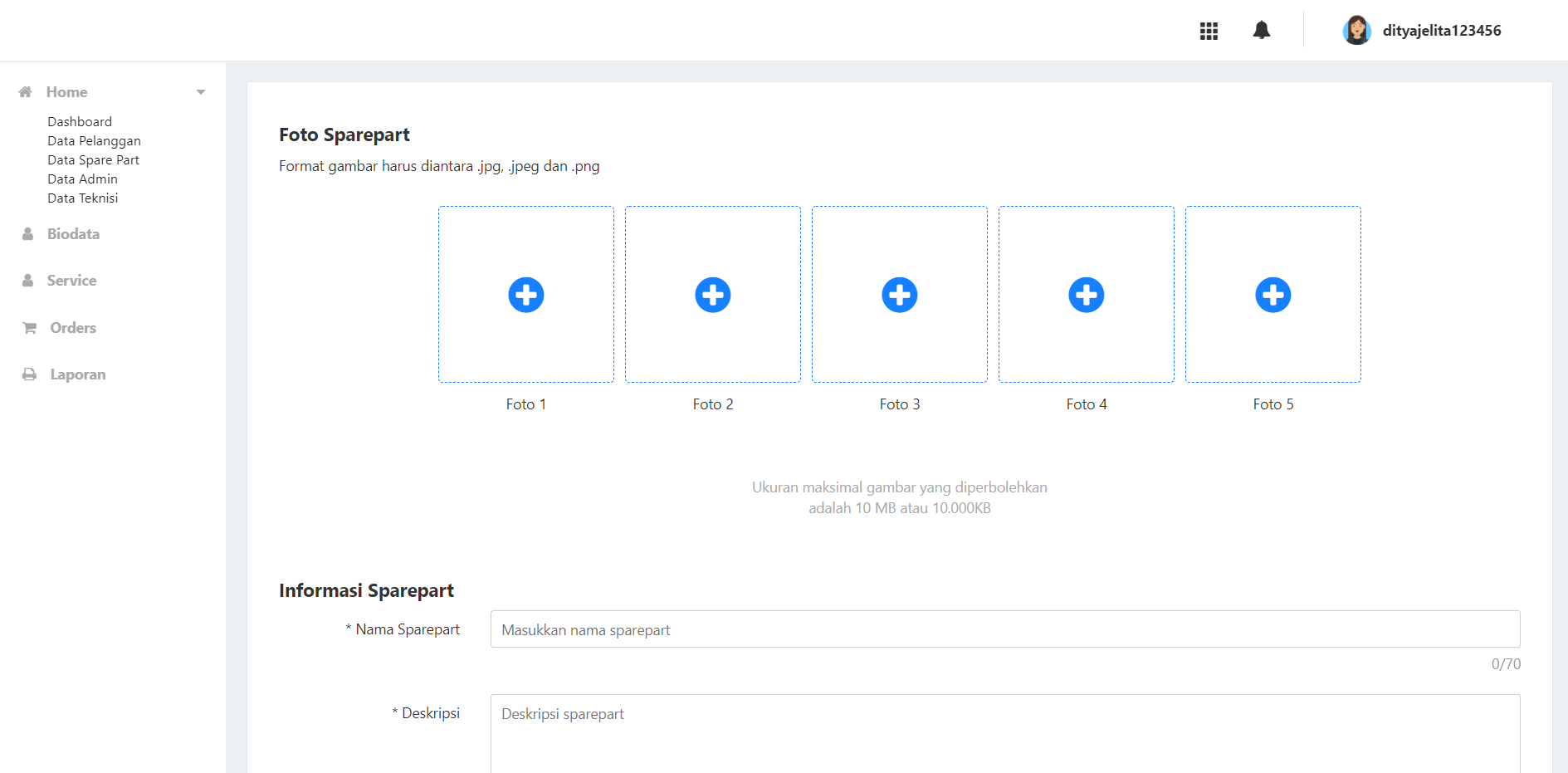
1. Halaman *Spare Part*

Halaman *Spare Part* merupakan halaman yang berisikan daftar dari *Spare Part* yang tersedia digudang serta beberapa informasinya seperti nama *spare part*, harga *spare part*, stok *spare part*, dan tipe *spare part*. Halaman *Spare Part* dapat dilihat pada Gambar 4.49.



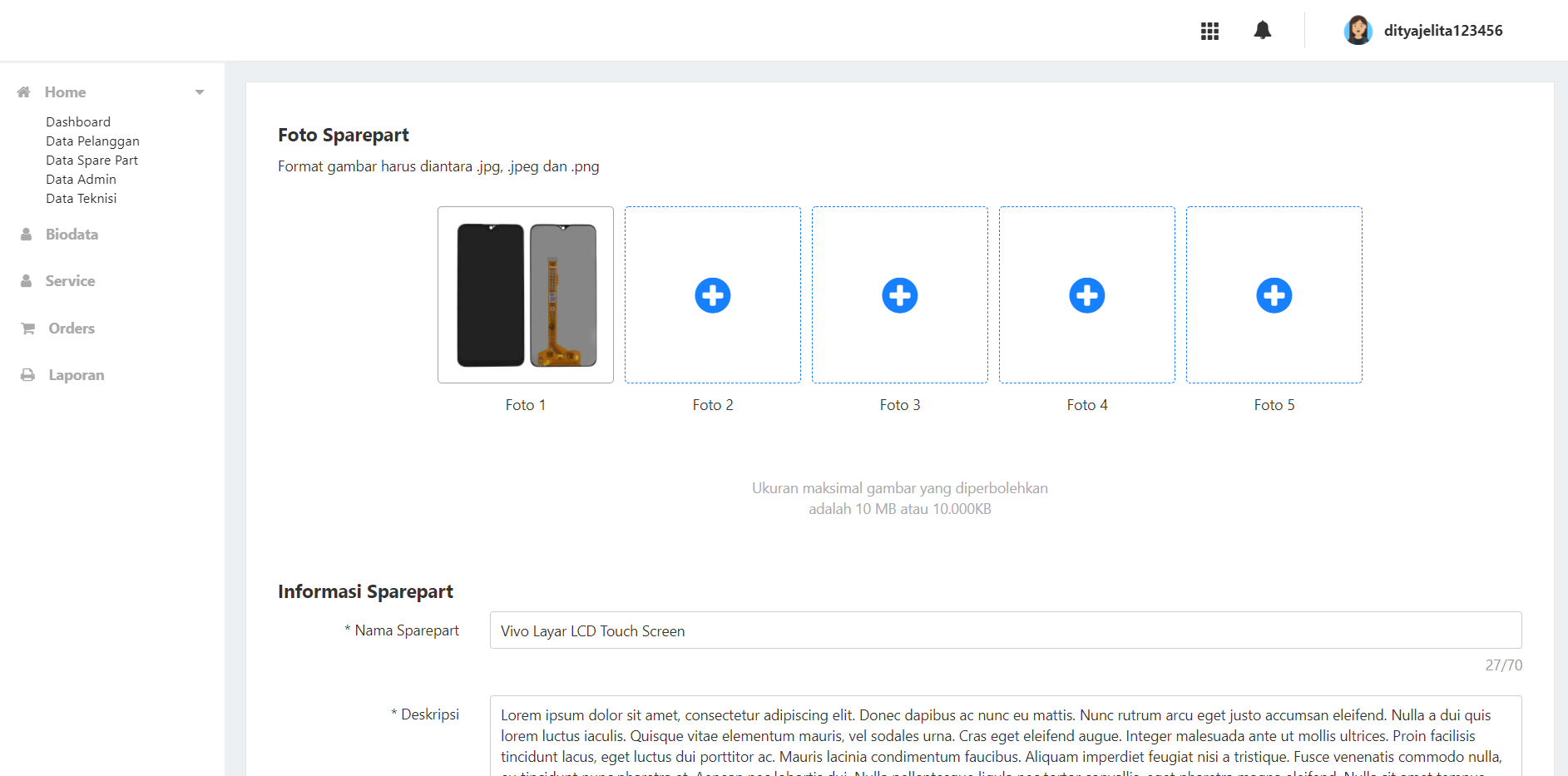
Gambar 4.49 Halaman *Spare Part*

1. Halaman Tambah *Spare Part*

Halaman Tambah Spare Part berisikan kolom – kolom mengenai data *spare part* yang akan ditambahkan seperti foto *spare part* yang maksimal berjumlah 5 (lima), ukuran foto maksimal 10 (sepuluh) MB, nama *spare part*, deskripsi *spare part*, tipe *spare part*, stok *spare part* yang tersedia di gudang, harga asli *spare part* saat pembelian, serta harga *spare part* saat penjualan dan dapat dilihat pada Gambar 4.50.

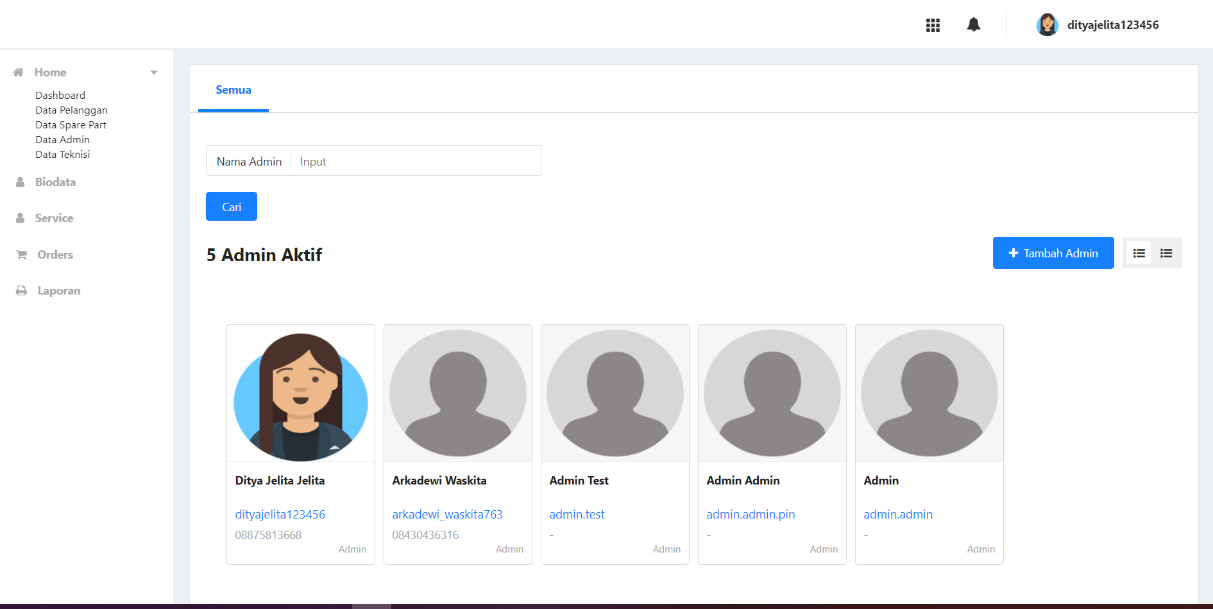
Gambar 4.50 Halaman Tambah *Spare Part*

1. Halaman Ubah *Spare Part*

Halaman Ubah Spare Part berisikan kolom – kolom mengenai data spare part yang akan diubah seperti foto *spare part* yang maksimal berjumlah 5 (lima), ukuran foto maksimal 10 (sepuluh) MB, nama *spare part*, deskripsi *spare part*, tipe *spare part*, stok *spare part* yang tersedia di gudang, harga asli *spare part* saat pembelian, serta harga *spare part* saat penjualan dan dapat dilihat pada Gambar 4.51.

Gambar 4.51 Halaman Ubah Spare Part

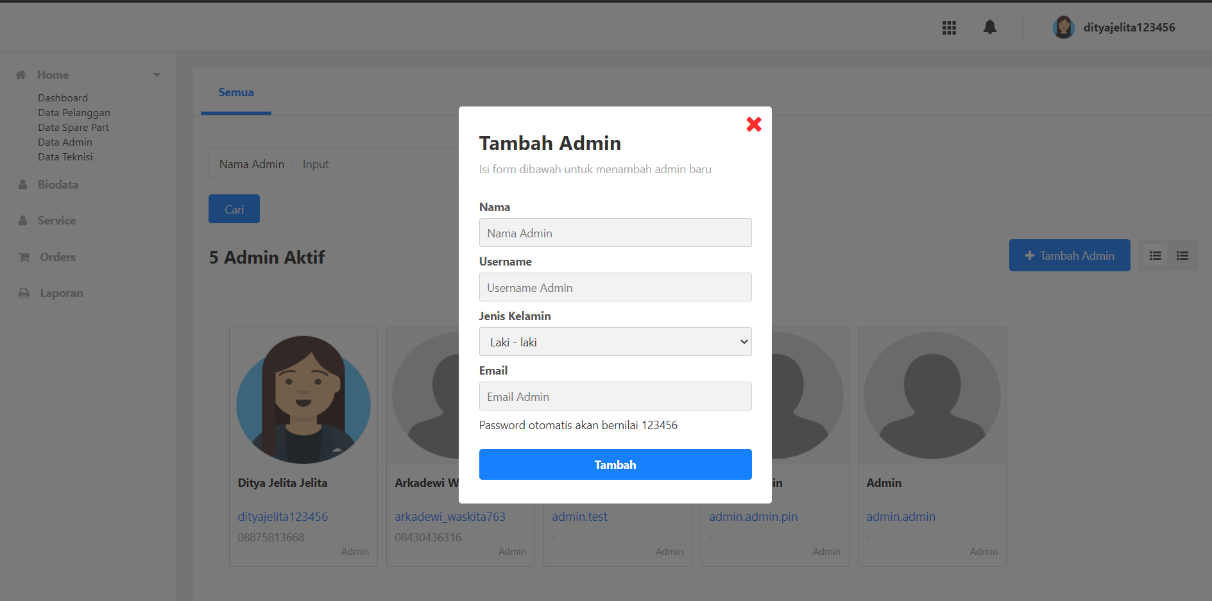
1. Halaman Admin

Halaman Admin merupakan halaman yang menampilkan daftar dari admin yang ada pada Oneya Solutions beserta kontaknya dan dapa dilihat pada Gambar 4.52.

Gambar 4.52 Halaman Admin

1. Form Tambah Admin

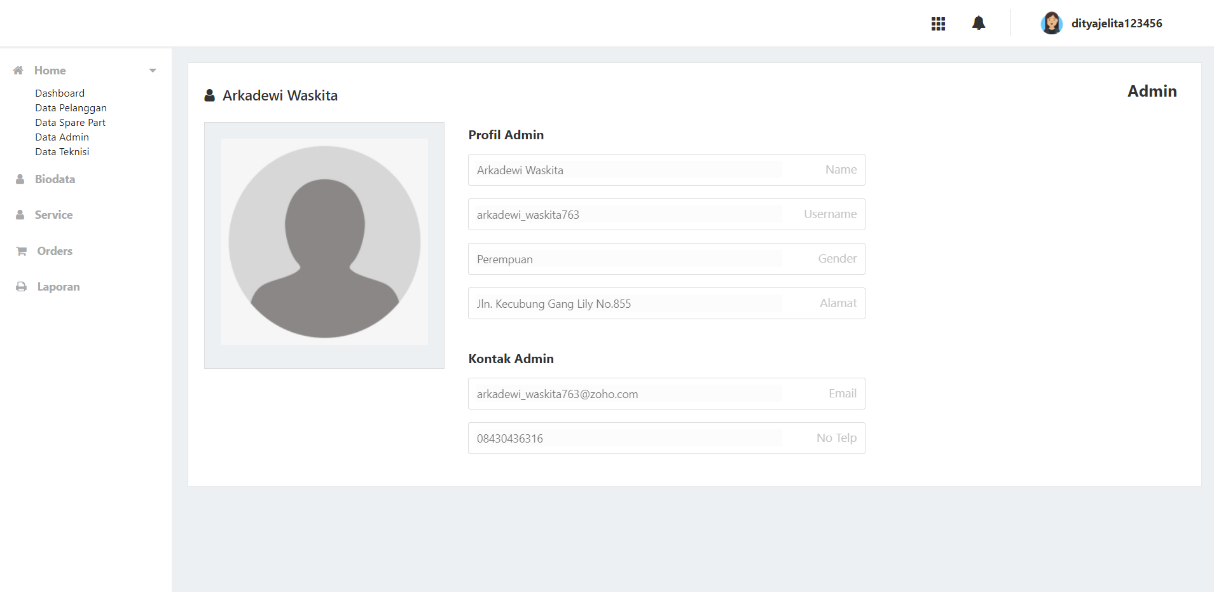
Form Tambah Admin berisikan kolom – kolom mengenai data dari admin seperti nama admin, username admin, jenis kelamin admin, serta email admin dan dapat dilihat pada Gambar 4.53.



Gambar 4.53 Form Tambah Admin

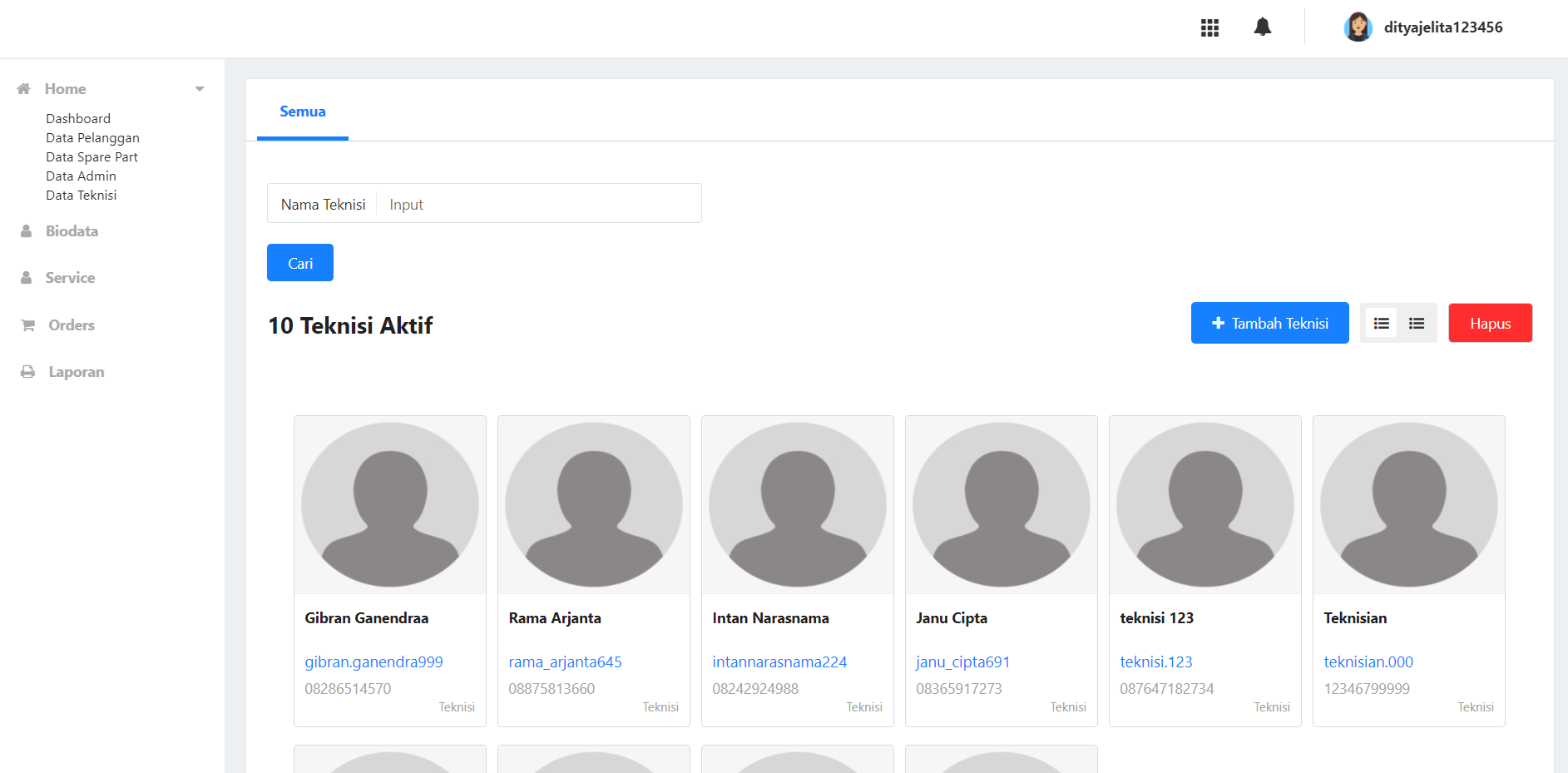
1. Halaman Detail Admin

Halaman Detail Admin merupakan halaman yang menampilkan data lengkap dari admin seperti nama, *username*, jenis kelamin, alamat, email serta nomor telepon dan dapat dilihat pada Gambar 4.54.



Gambar 4.54 Halaman Detail Admin

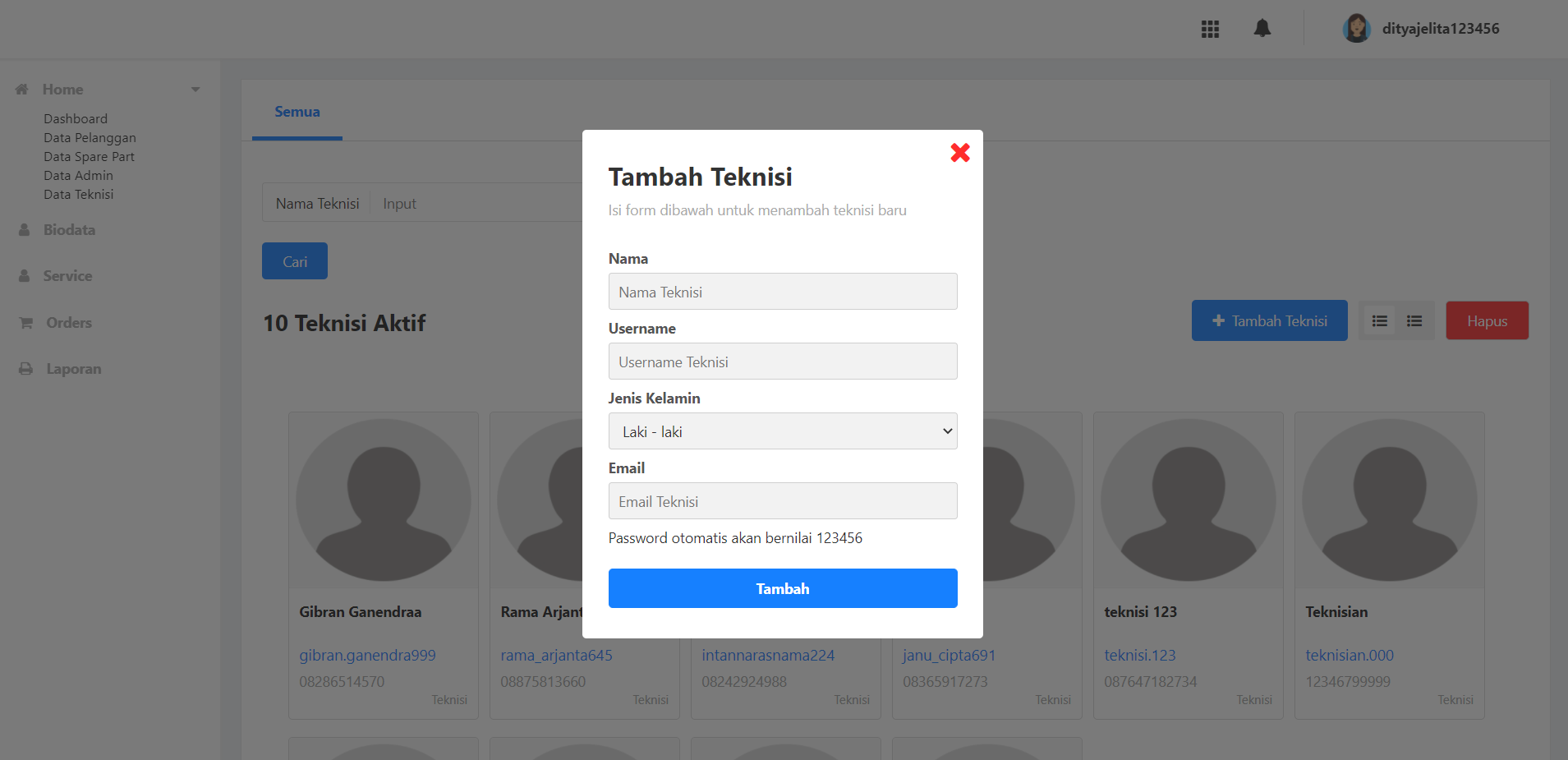
1. Halaman Teknisi

Halaman Teknisi merupakan halaman yang menampilkan daftar teknisi yang terdata dalam sistem beserta beberapa kontaknya dan dapat dilihat pada Gambar 4.55.

Gambar 4.55 Halaman Teknisi

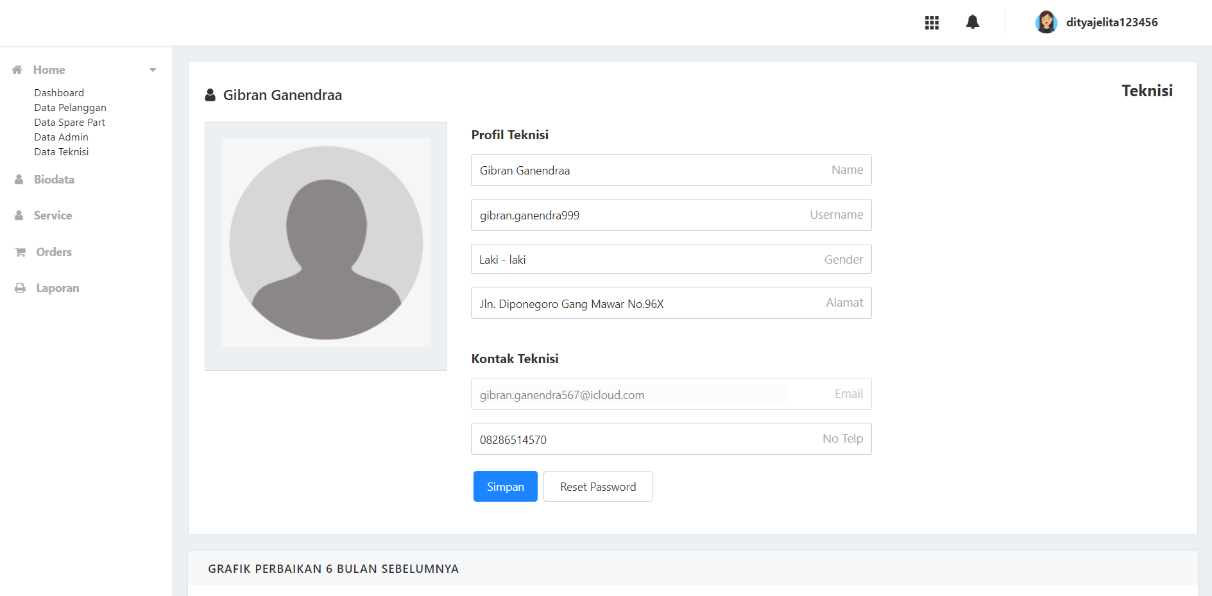
1. Form Tambah Teknisi

Form Tambah Teknisi berisikan kolom – kolom mengenai data dari teknisi seperti nama teknisi, username teknisi, jenis kelamin teknisi, serta email teknisi dan dapat dilihat pada Gambar 4.56.



Gambar 4.56 Form Tambah Teknisi

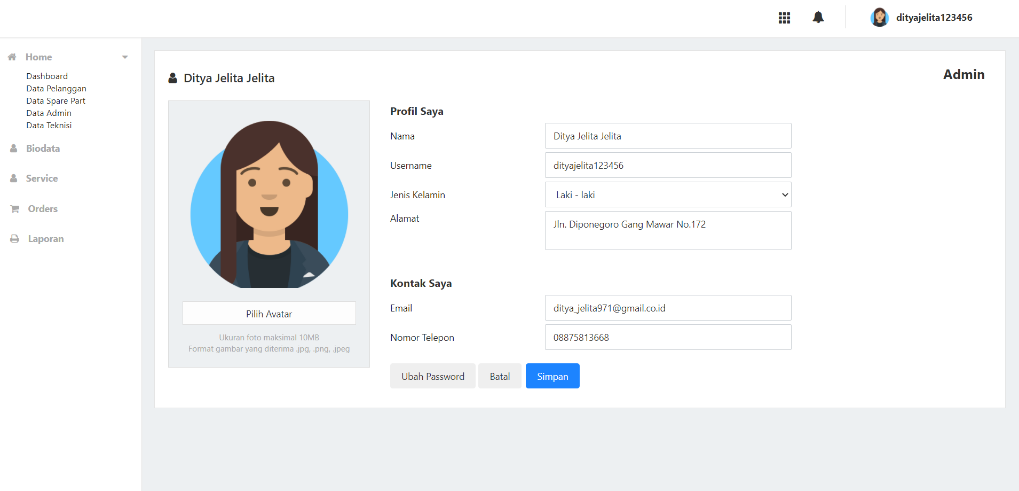
1. Halaman Detail Teknisi

Halaman Detail Teknisi merupakah halaman yang menampilkan data lengkap dari teknisi seperti nama, *username*, jenis kelamin, alamat, email serta nomor telepon. Admin dapat melakukan perubahan data teknisi, mereset password teknisi, serta dapat melihat grafik keberhasilan perbaikan teknisi dan dapat dilihat pada Gambar 4.57.

Gambar 4.57 Halaman Detail Teknisi

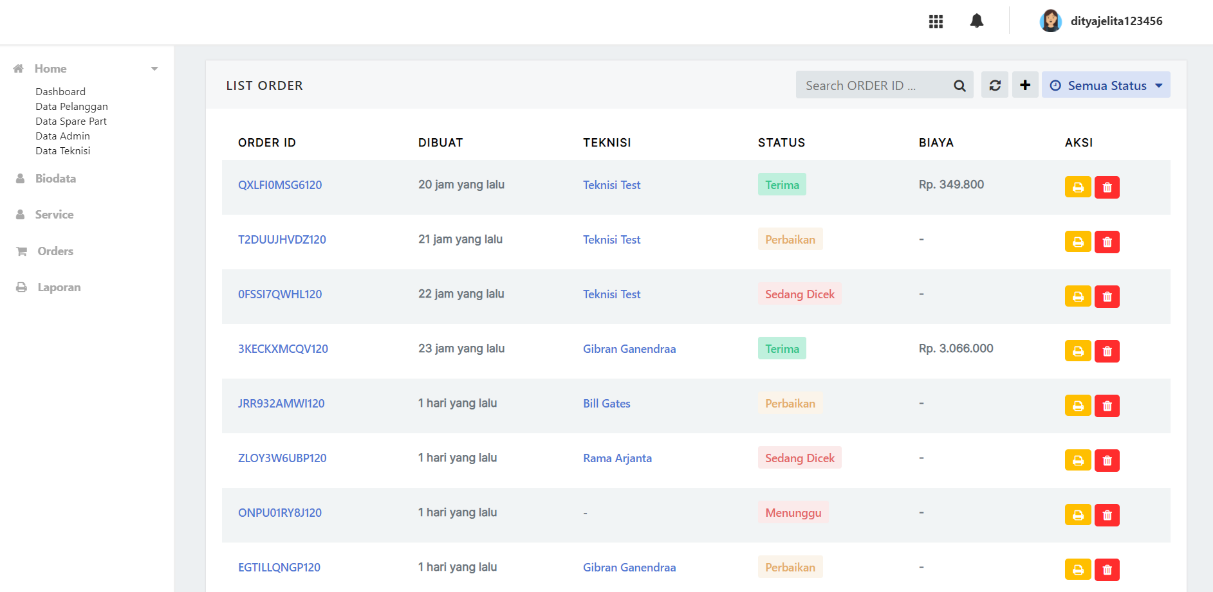
1. Halaman Ubah Biodata

Halaman Ubah Biodata berisikan kolom – kolom profil dan kontak yang dapat diubah seperti nama, username, jenis kelamin, alamat, email serta nomor telepon. Dalam halaman biodata, admin juga dapat mengubah passwordnya serta dapat mengubah foto profilnya dan dapat dilihat pada Gambar 4.58.



Gambar 4.59 Halaman Ubah Biodata

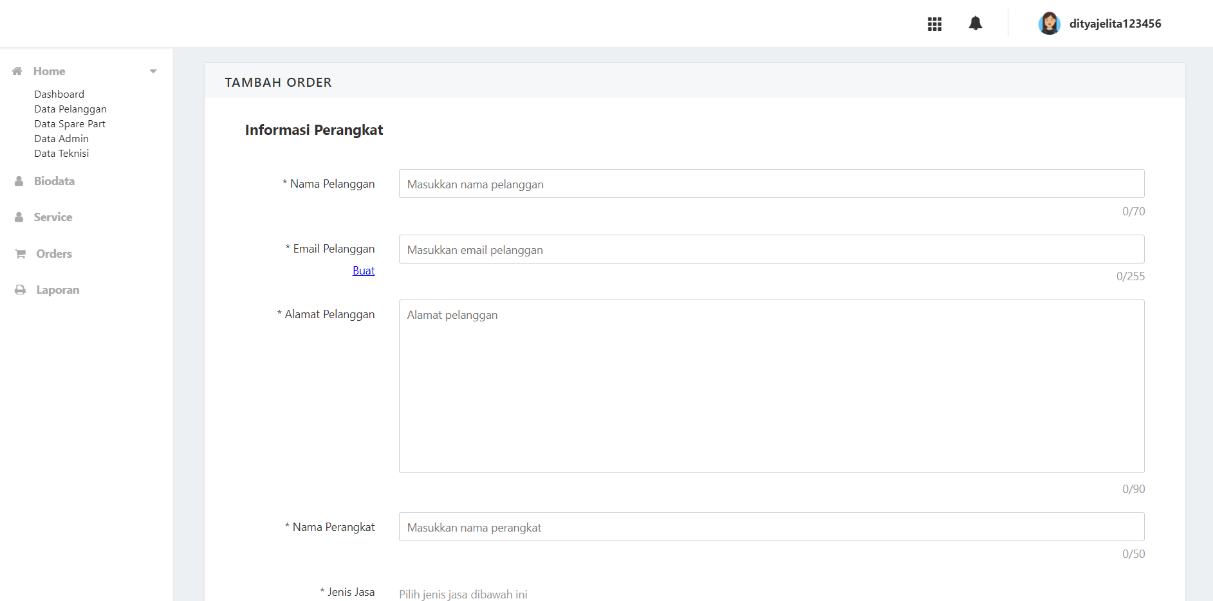
1. Halaman *Order*

Halaman *Order* berisikan daftar mengenai *order* yang masuk kedalam sistem beserta beberapa informasinya seperti *id* *order*, waktu dibuat, nama teknisi yang mengambil *order*, *status* *order*, total biaya untuk *order* serta aksi yang dapat dilakukan admin terhadap *order* tersebut dan dapat dilihat pada Gambar 4.60.

Gambar 4.60 Halaman *Order*

1. Halaman Tambah *Order*

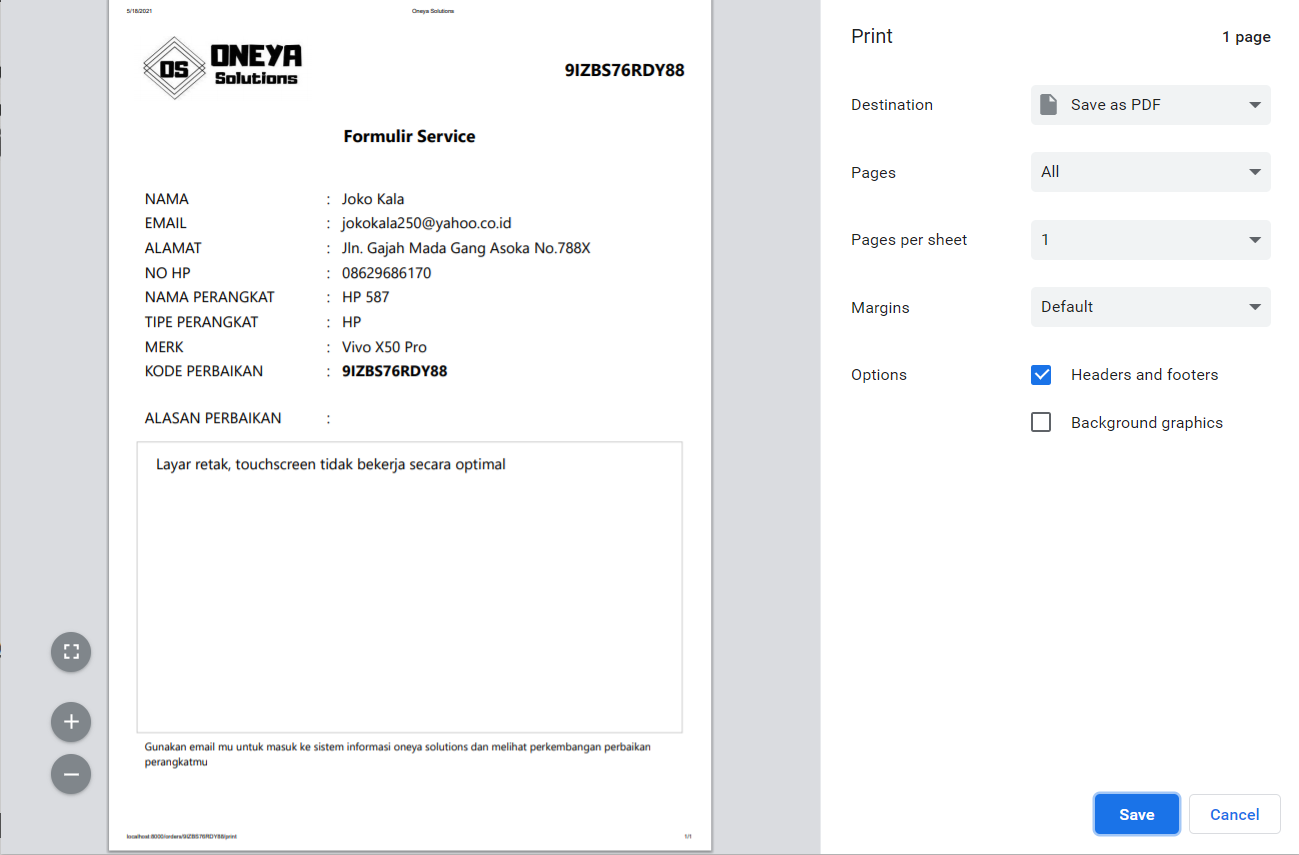
Halaman Tambah *Order* merupakan halaman yang digunakan untuk menambah order dan berisikan kolom – kolom mengenai data order seperti nama pelanggan, email pelanggan, alamat pelanggan, nama perangkat, jenis jasa, keluhan perangkat, tipe perangkat, serta merek perangkat dan dapat dilihat pada Gambar 4.61.



Gambar 4.61 Halaman Tambah Order

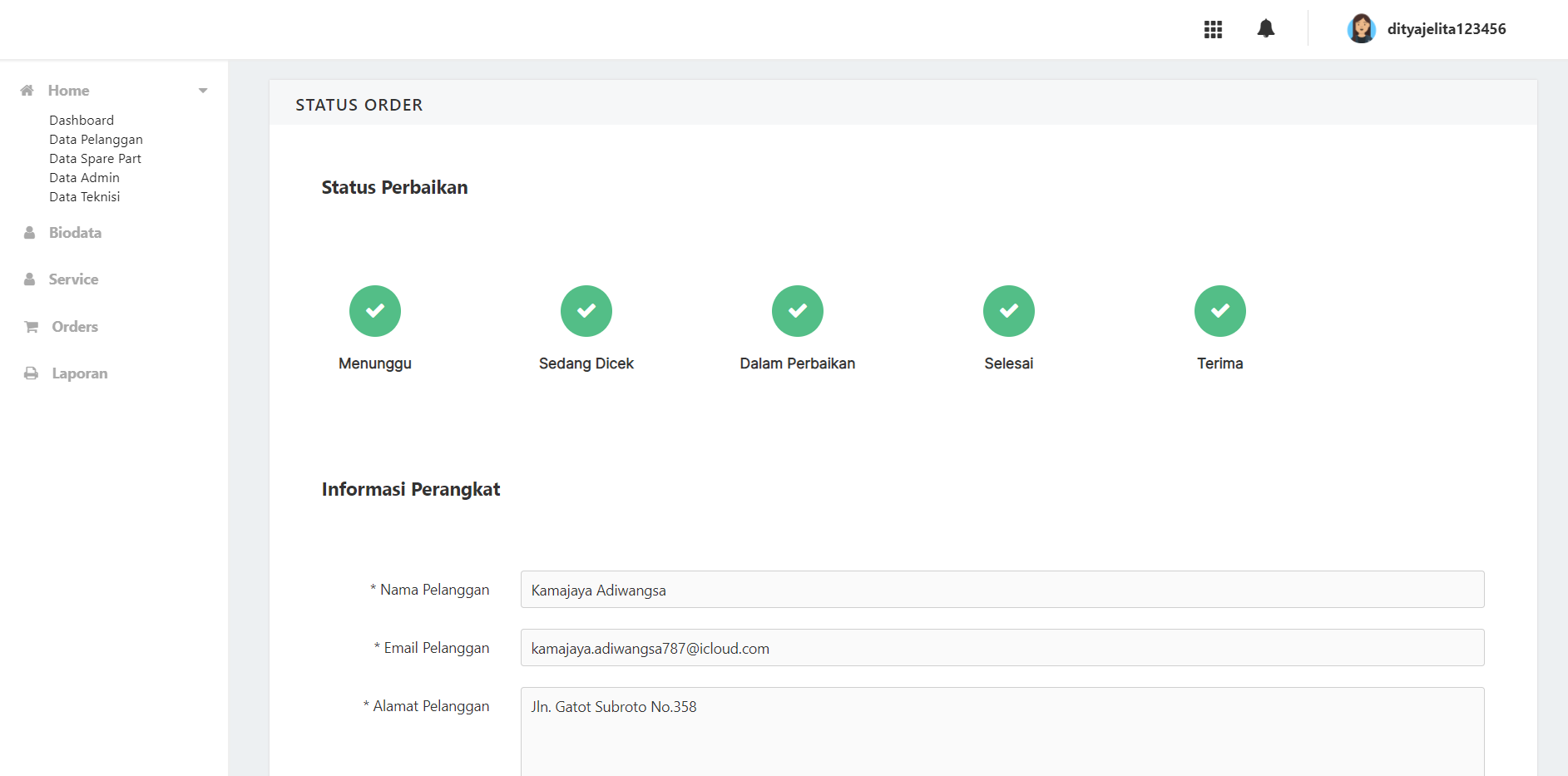
1. Halaman *Print* *Order*

Halaman *Print Order* merupakan halaman yang menampilkan informasi *order* serta informasi pelanggan untuk dicetak dan dapat dilihat pada Gambar 4.62.



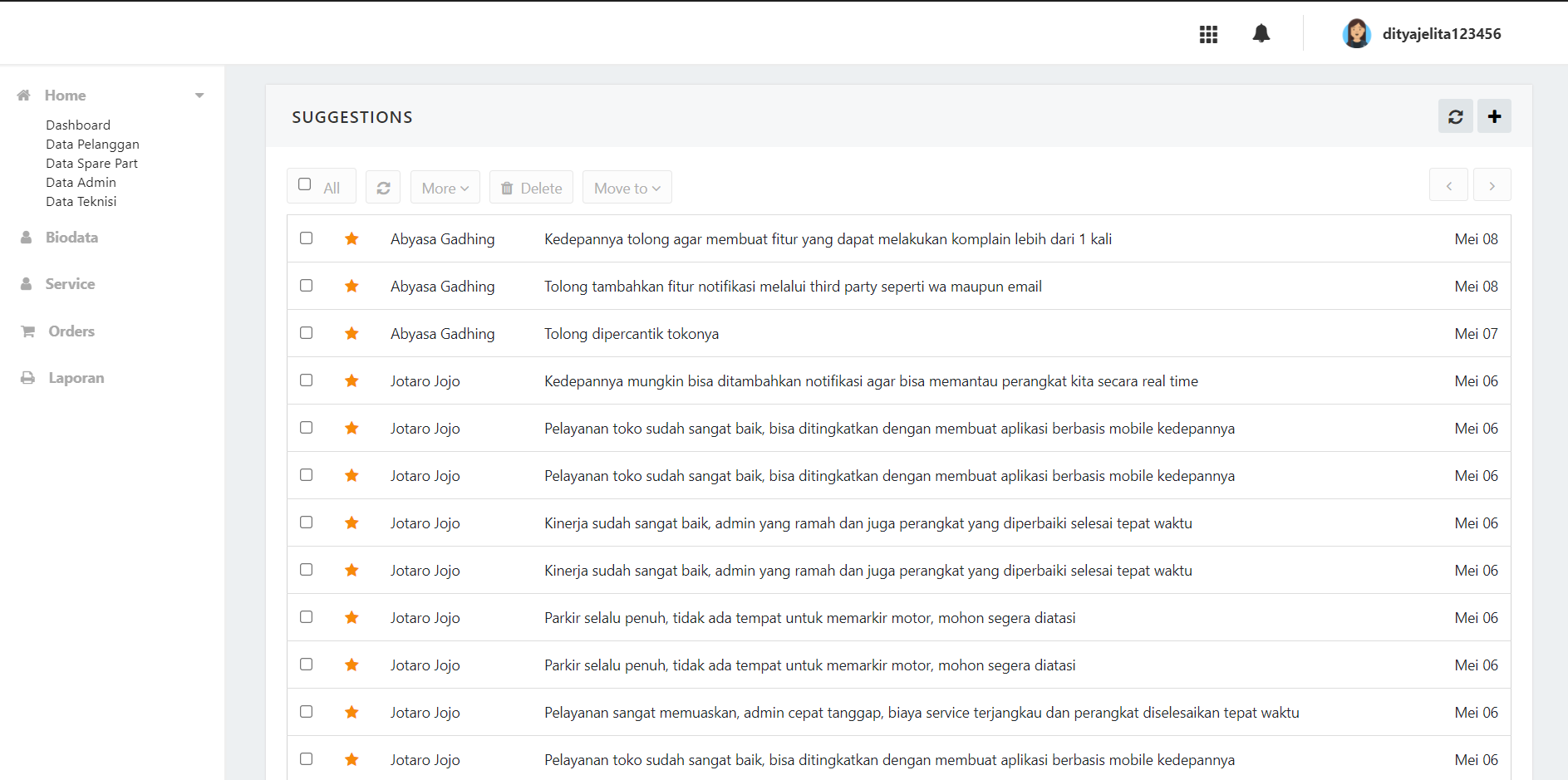
Gambar 4.62 Halaman *Print Order*

1. Halaman Lihat *Order*

Halaman Lihat *Order* merupakan halaman yang menampilkan data *order* seperti status *order*, nama pelanggan, *email* pelanggan, alamat pelanggan, *spare part* yang diganti, nama perangkat, keluhan perangkat, tipe perangkat, merek perangkat, jenis jasa, teknisi yang mengerjakan, serta total yang harus dibayar dan dapat dilihat pada Gambar 4.63.

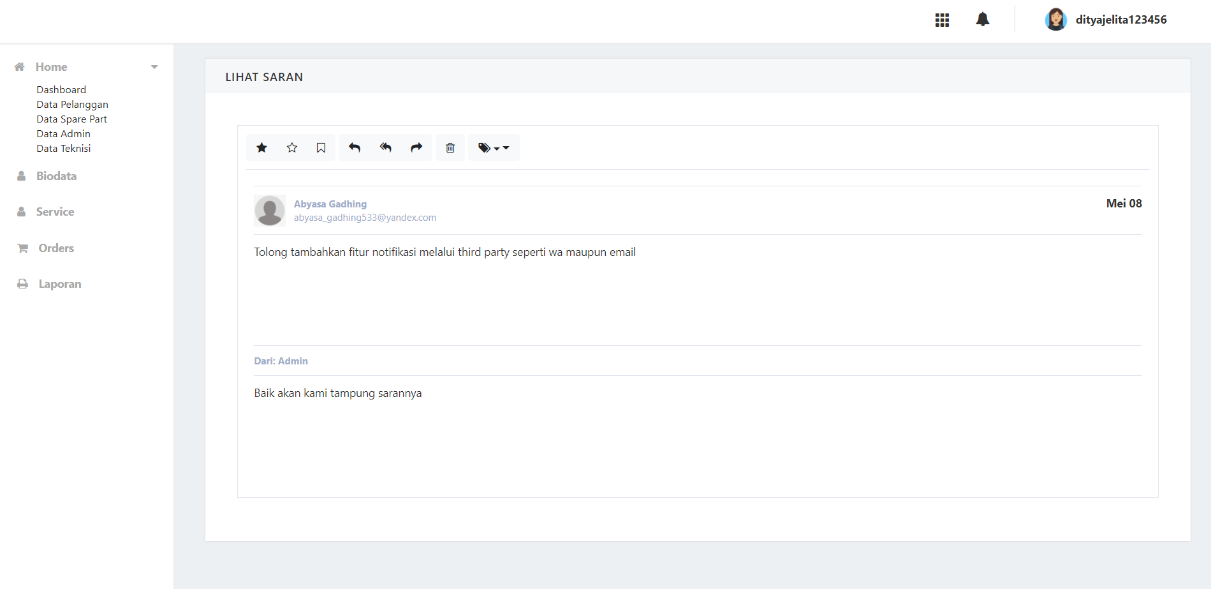
Gambar 4.63 Halaman Lihat *Order*

1. Halaman Saran

Halaman Saran merupakan halaman yang menampilkan daftar saran yang dikirimkan pelanggan kepada toko dan diurutkan dari tanggal terbaru dan dapat dilihat pada Gambar 4.64.

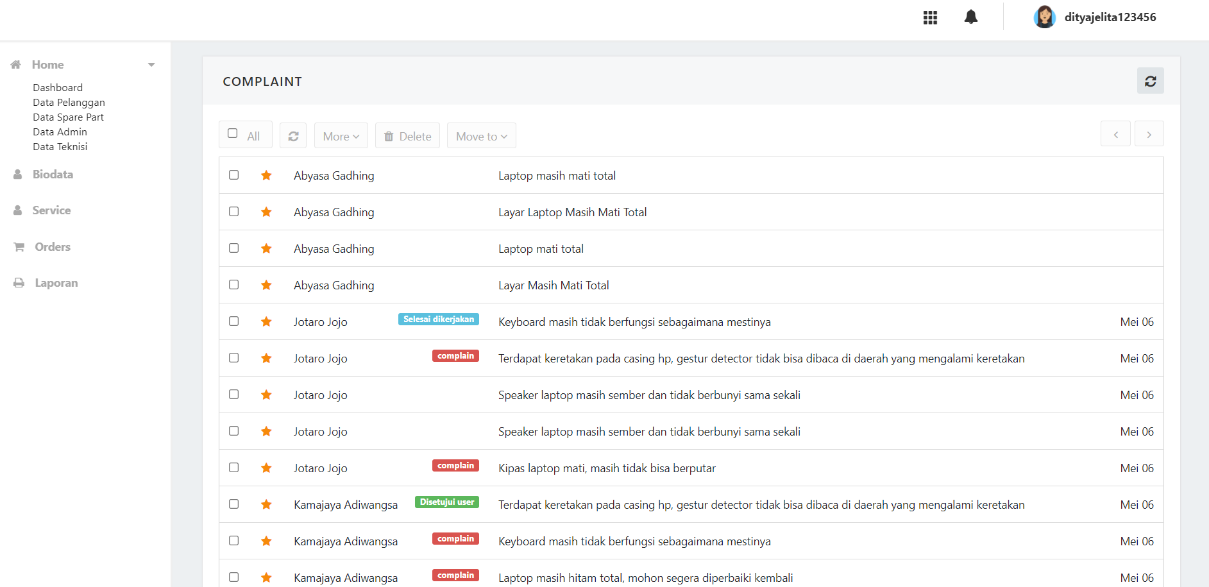
Gambar 4.64 Halaman Saran

1. Halaman Lihat Saran

Halaman Lihat Saran merupakan halaman yang menampilkan data mengenai saran yang dikirimkan kepada toko seperti nama pelanggan yang mengirim, email pelanggan, tanggal dikirim, isi saran serta balasan admin terhadap saran tersebut dan dapat dilihat pada Gambar 4.65.

Gambar 4.65 Halaman Lihat Saran

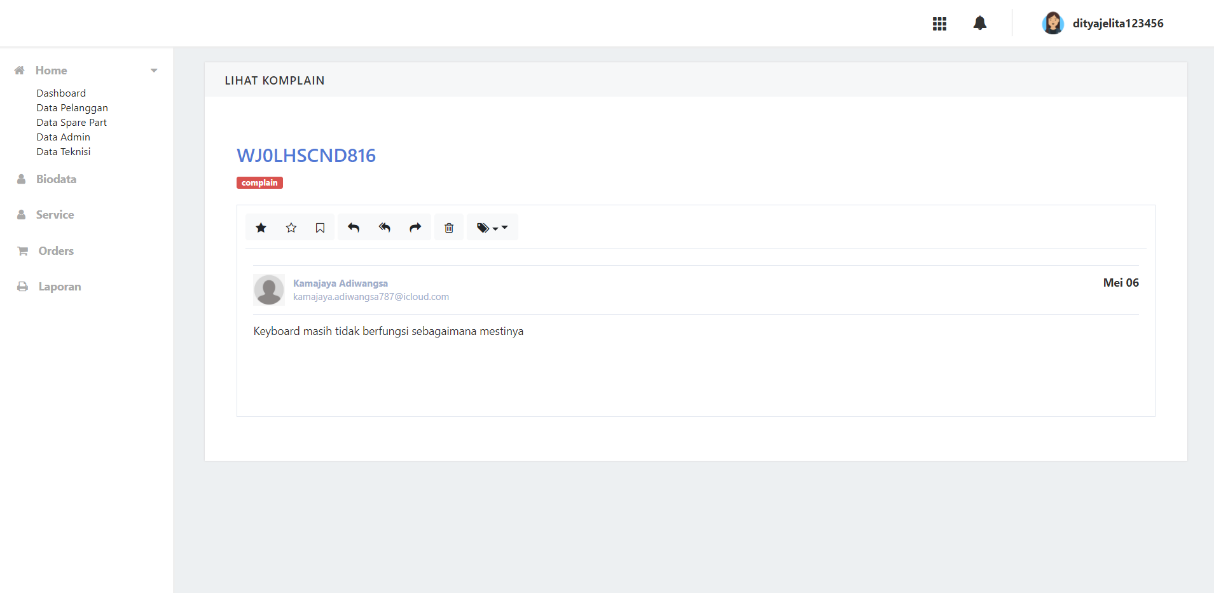
1. Halaman Komplain

Halaman Komplain merupakan halaman yang menampilkan daftar komplain yang dibuat oleh pelanggan serta terdapat *bagde* status dari komplain tersebut dan dapat dilihat pada Gambar 4.67.

Gambar 4.67 Halaman Komplain

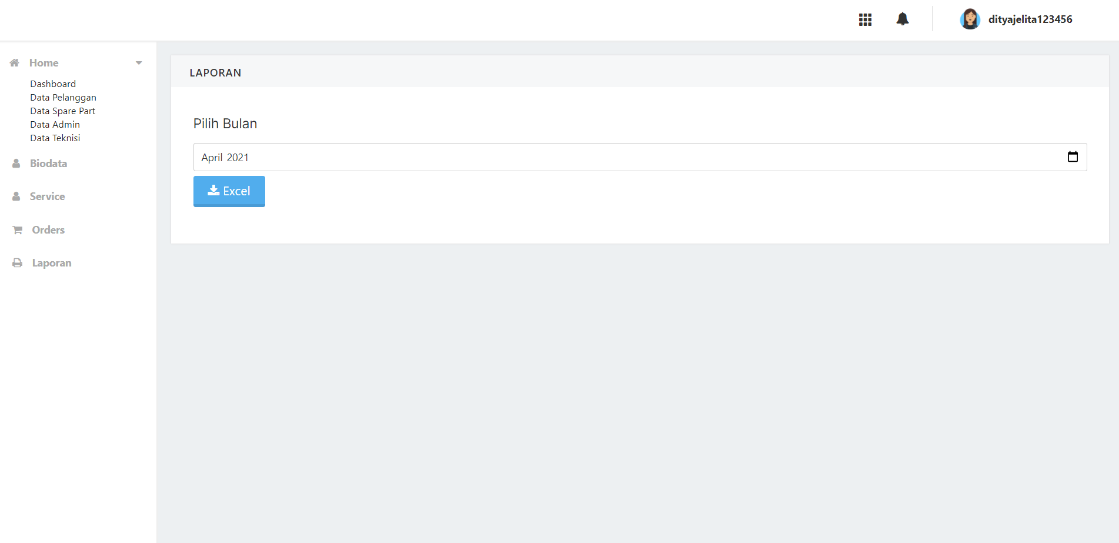
1. Halaman Lihat Komplain

Halaman Lihat Komplain merupakan halaman yang menampilkan data komplain yang dibuat oleh pelanggan beserta informasinya seperti nama pelanggan, *email* pelanggan, tanggal dibuat, serta *id order* yang dapat di klik untuk menampilkan informasi *order* dan dapat dilihat pada Gambar 4.68.



Gambar 4.68 Halaman Lihat Komplain

1. Halaman Cetak Laporan

Halaman Cetak Laporan merupakan halaman untuk mencetak laporan pendapatan bulanan ke dalam bentuk excel sesuai dengan pilihan bulan yang dimasukkan oleh admin dan dapat dilihat pada Gambar 4.69.

Gambar 4.69 Halaman Cetak Laporan

* + 1. **Halaman Teknisi**

Berikut merupakan implementasi sistem untuk halaman – halaman yang dapat diakses oleh pengguna dengan hak akses Teknisi. Implementasi tersebut diantaranya :

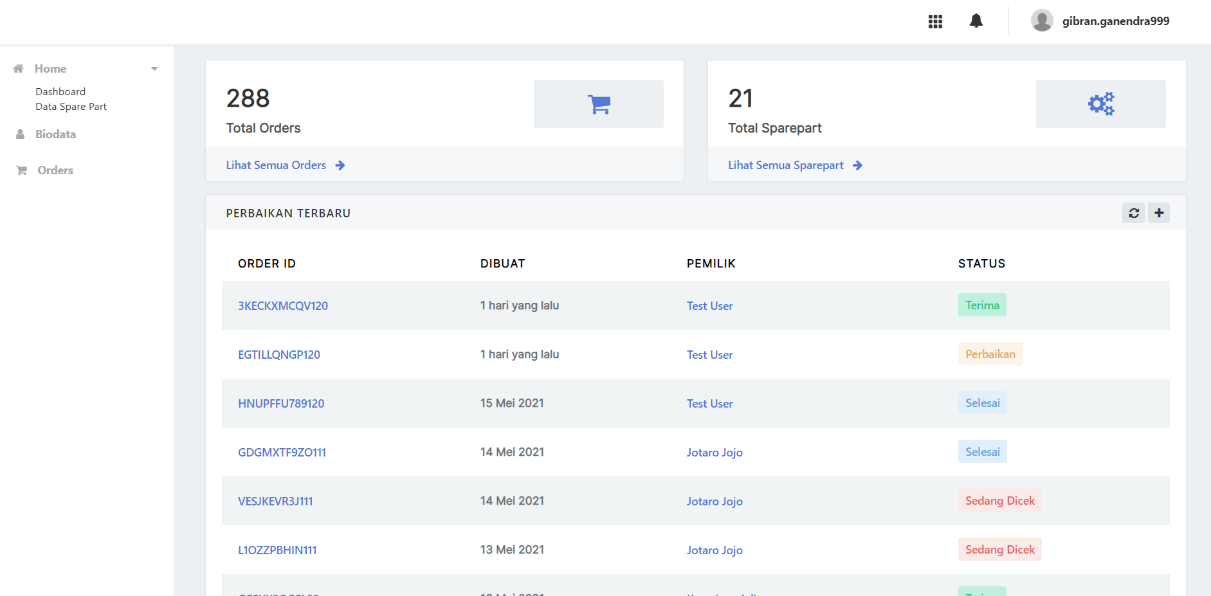
1. Halaman *Login*

Pada halaman *login* terdiri atas 2 (dua) masukan yaitu *email* dan *password* serta sebuah tombol masuk. Pada halaman *login* ini pengguna yang akan mengakses sistem diharuskan memasukkan *email* dan *password* dan dapat dilihat pada Gambar 4.70.

Gambar 4.70 Halaman *Login*

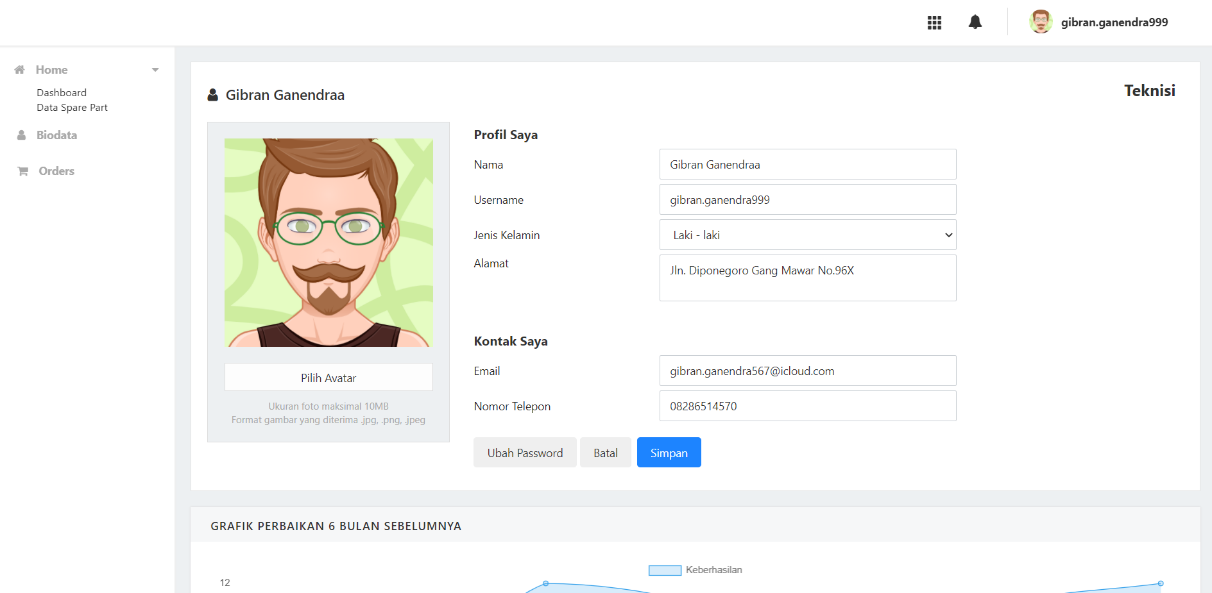
1. Halaman *Dashboard*

Halaman *Dashboard* merupakan halaman utama setelah melakukan *login*. Pada halaman dashboard, teknisi dapat melihat jumlah *order* dan daftar *order* yang telah diambil olehnya, jumlah total *spare part* yang tersedia, serta dapat mengakses menu – menu yang tersedia di dalamnya antara lain *sub* *menu* *dashboard*, *sub* *menu* *data* *spare* *part*, menu biodata, serta *menu* *orders* dan dapat dilihat pada Gambar 4.71.



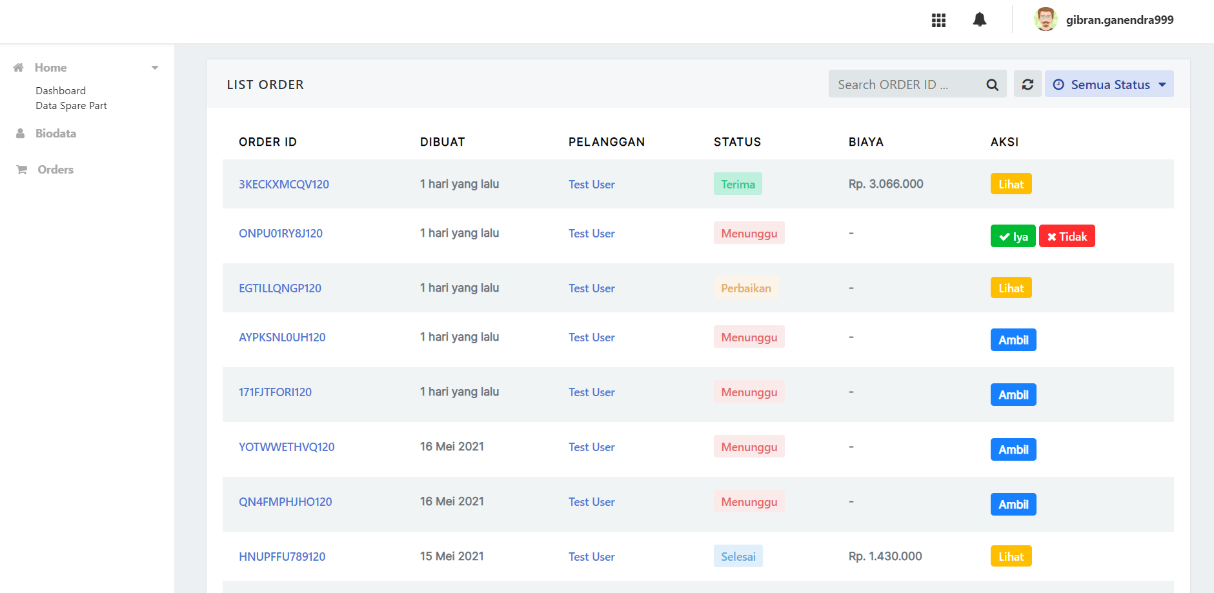
Gambar 4.71 Halaman Dashboard

1. Halaman Ubah Biodata

Halaman Ubah Biodata berisikan kolom – kolom profil dan kontak yang dapat diubah seperti nama, username, jenis kelamin, alamat, email serta nomor telepon. Dalam halaman biodata, teknisi juga dapat mengubah passwordnya, melihat grafik keberhasilan perbaikan serta dapat mengubah foto profilnya dan dapat dilihat pada Gambar 4.72.

Gambar 4.73 Halaman Ubah Biodata

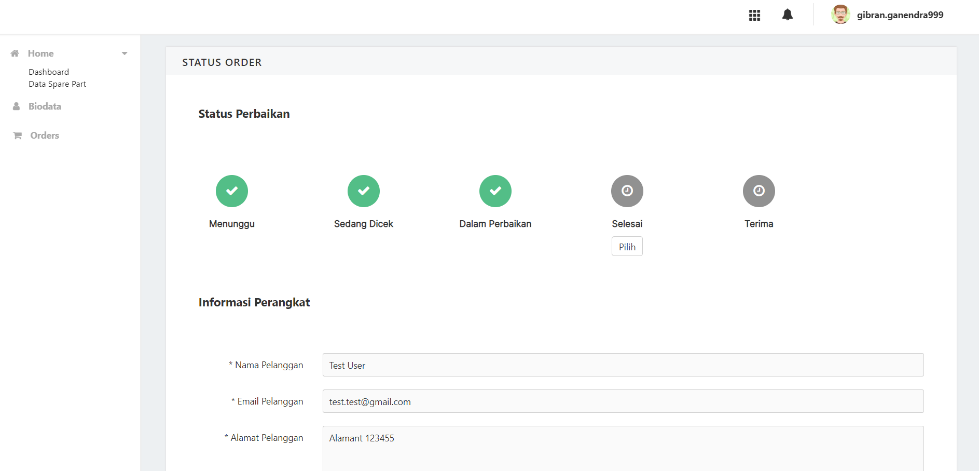
1. Halaman *Order*

Halaman *Order* berisikan daftar mengenai *order* yang masuk kedalam sistem beserta beberapa informasinya seperti *id* *order*, waktu dibuat, nama pelanggan *status* *order*, total biaya *order* serta aksi yang dapat dilakukan teknisi terhadap *order* tersebut seperti ambil order maupun lihat order dan dapat dilihat pada Gambar 4.74.

Gambar 4.74 Halaman *Order*

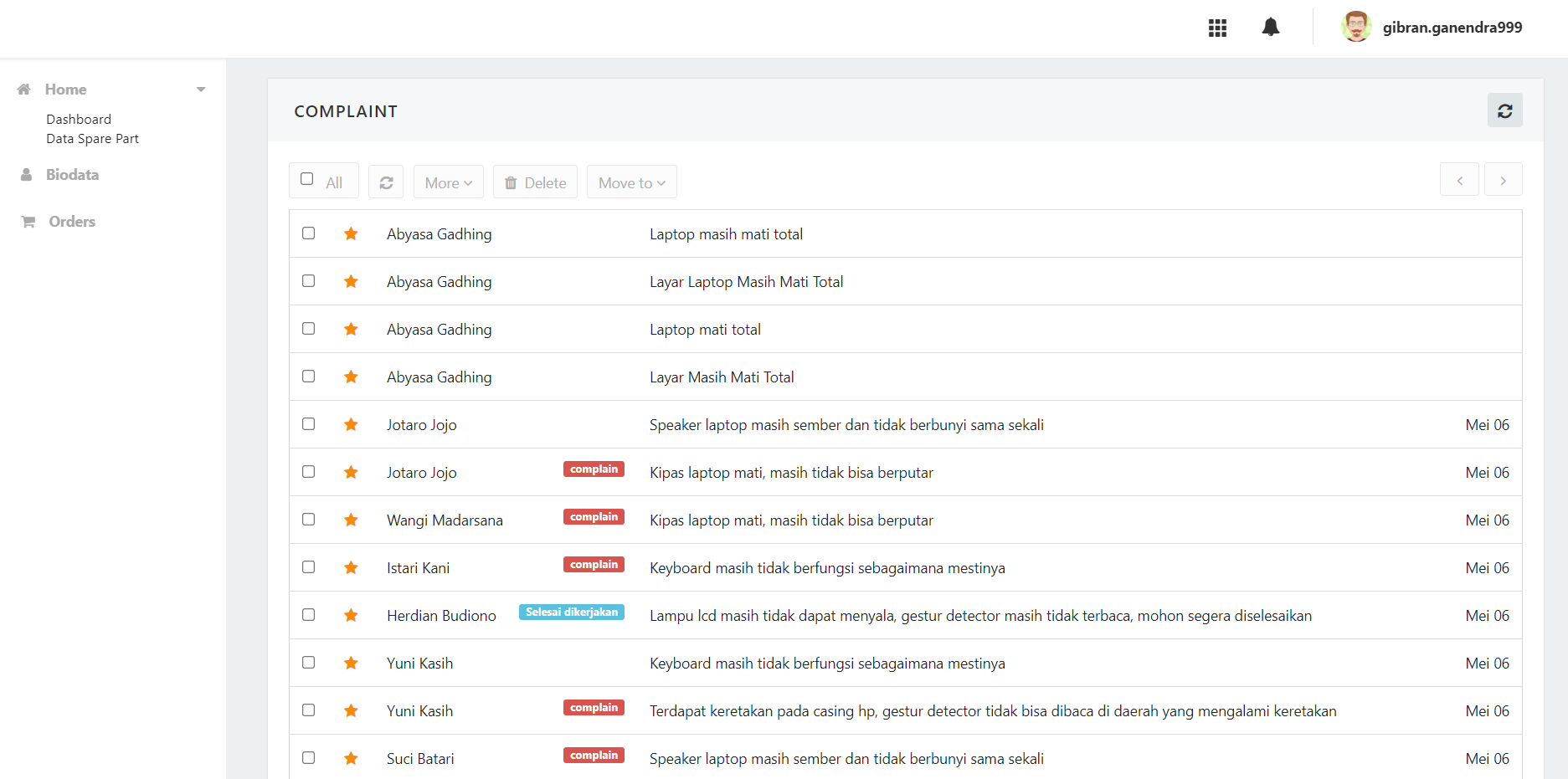
1. Halaman Lihat *Order*

Halaman Lihat *Order* merupakan halaman yang menampilkan data *order* seperti status *order*, nama pelanggan, *email* pelanggan, alamat pelanggan, *spare part* yang diganti, nama perangkat, keluhan perangkat, tipe perangkat, merek perangkat, jenis jasa, teknisi yang mengerjakan, serta total yang harus dibayar. Pada halaman Lihat *Order*, teknisi dapat mengubah status order serta dapat memilih spare part yang diganti dan dapat dilihat pada Gambar Gambar 4.75.



Gambar 4.76 Halaman Lihat *Order*

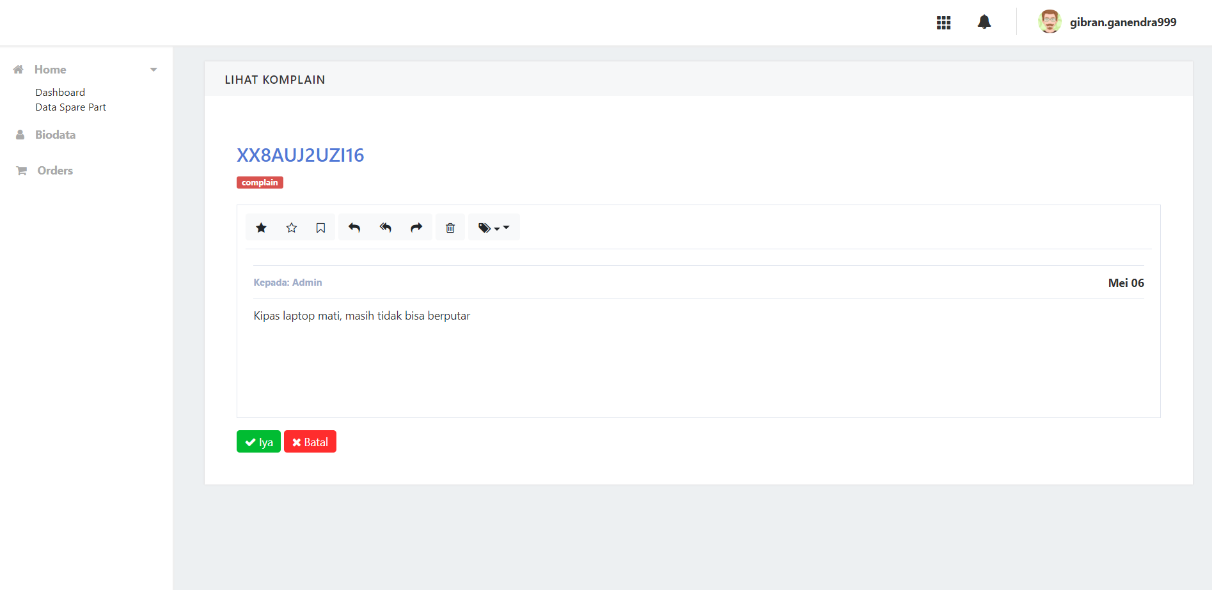
1. Halaman Komplain

Halaman Komplain merupakan halaman yang menampilkan daftar komplain milik teknisi yang dibuat oleh pelanggan serta terdapat *bagde* status dari komplain tersebut dan dapat dilihat pada Gambar 4.77.

Gambar 4.78 Halaman Komplain

1. Halaman Lihat Komplain

Halaman Lihat Komplain merupakan halaman yang menampilkan data komplain yang dibuat oleh pelanggan beserta informasinya seperti nama pelanggan, *email* pelanggan, tanggal dibuat, serta *id order* yang dapat di klik untuk menampilkan informasi *order*. Pada halaman lihat komplain, teknisi dapat mengubah status komplain menjadi “sudah dikerjakan” dengan menekan tombol “kerjakan” dan dapat dilihat pada Gambar 4.79.



Gambar 4.79 Halaman Lihat Komplain

* + 1. **Halaman Pelanggan**

Berikut merupakan implementasi sistem untuk halaman – halaman yang dapat diakses oleh pengguna dengan hak akses Pelanggan. Implementasi tersebut diantaranya :

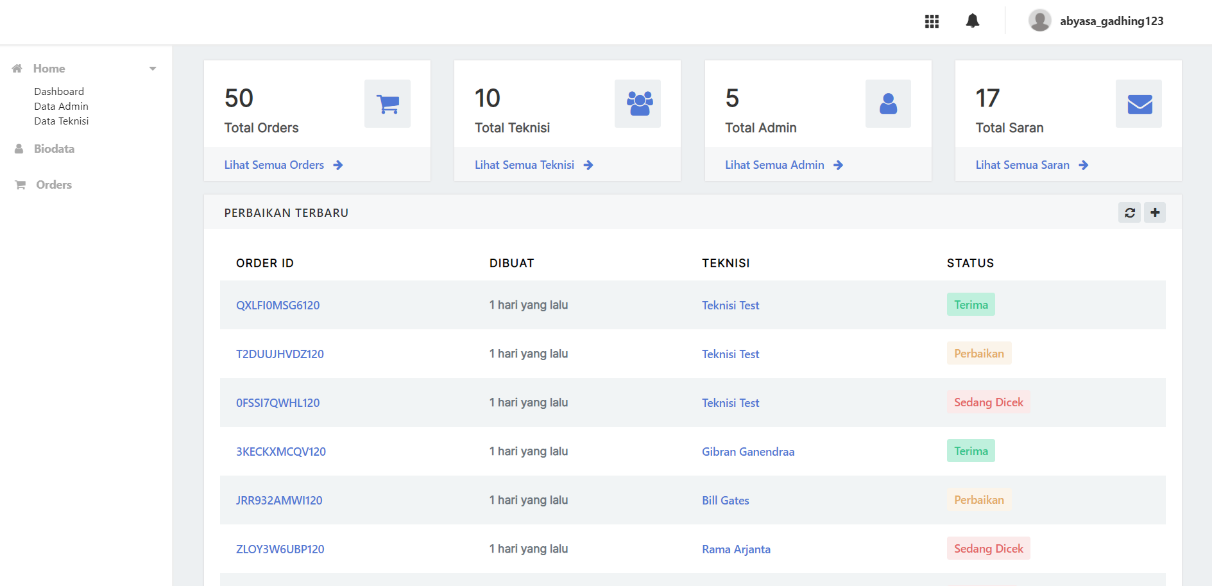
1. Halaman *Login*

Pada halaman *login* terdiri atas 2 (dua) masukan yaitu *email* dan *password* serta sebuah tombol masuk. Pada halaman *login* ini pengguna yang akan mengakses sistem diharuskan memasukkan *email* dan *password* dan dapat dilihat pada Gambar 4.80.

Gambar 4.80 Halaman *Login*

1. Halaman *Dashboard*

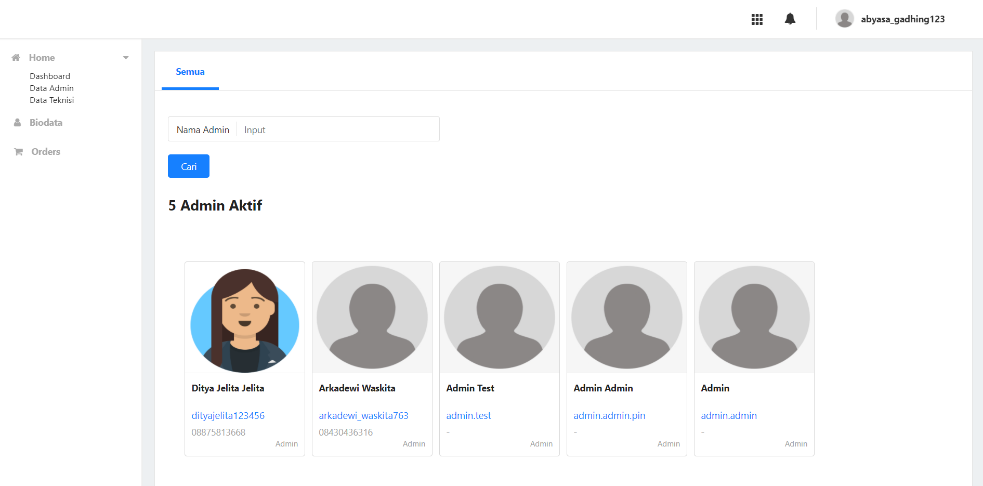
Halaman *Dashboard* merupakan halaman utama setelah melakukan *login*. Pada halaman dashboard, pelanggan dapat melihat jumlah *order*, total teknisi, total admin, total saran yang dikirimkannya dan daftar *order* yang diambil. Dalam halaman *dashboard*, pelanggan juga dapat mengakses menu – menu yang tersedia di dalamnya antara lain *sub* *menu* *dashboard*, *sub* *menu* *admin*, *sub menu* teknisi, *menu* biodata, serta *menu* *orders* dan dapat dilihat pada Gambar 4.81.



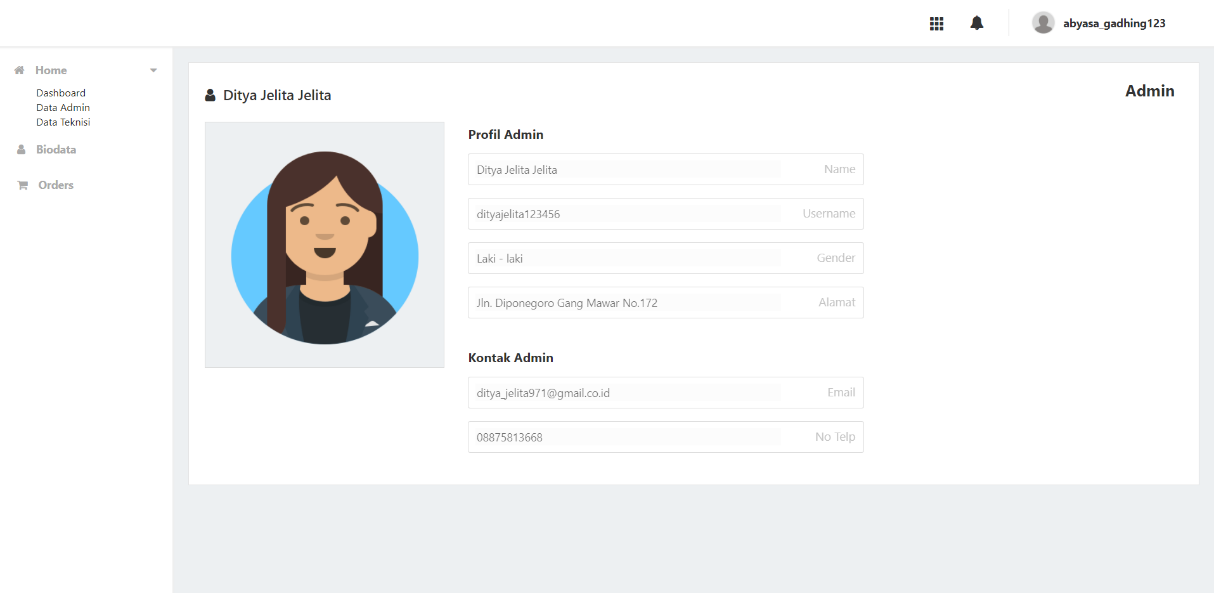
Gambar 4.81 Halaman *Dashboard*

1. Halaman *Admin*

Halaman Admin merupakan halaman yang menampilkan daftar dari admin yang ada pada Oneya Solutions beserta kontaknya dan dapat dilihat pada Gambar 4.82.

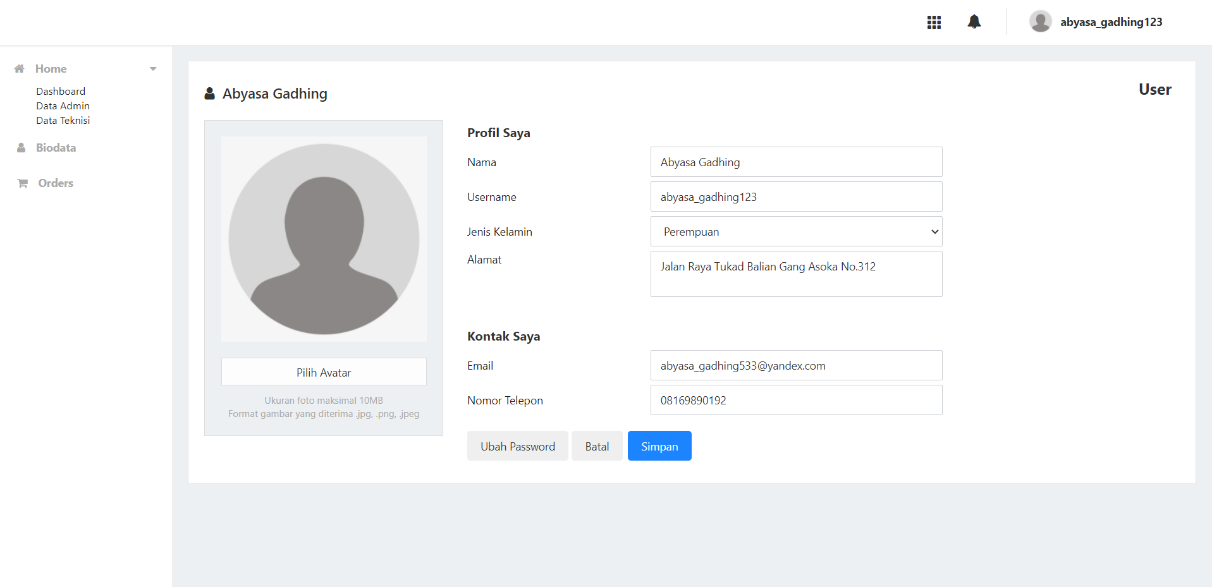


1. Halaman *Detail Admin*

Halaman *Detail Admin* merupakan halaman yang menampilkan data lengkap dari admin seperti nama, *username*, jenis kelamin, alamat, *email* serta nomor telepon dan dapat dilihat pada Gambar 4.83.

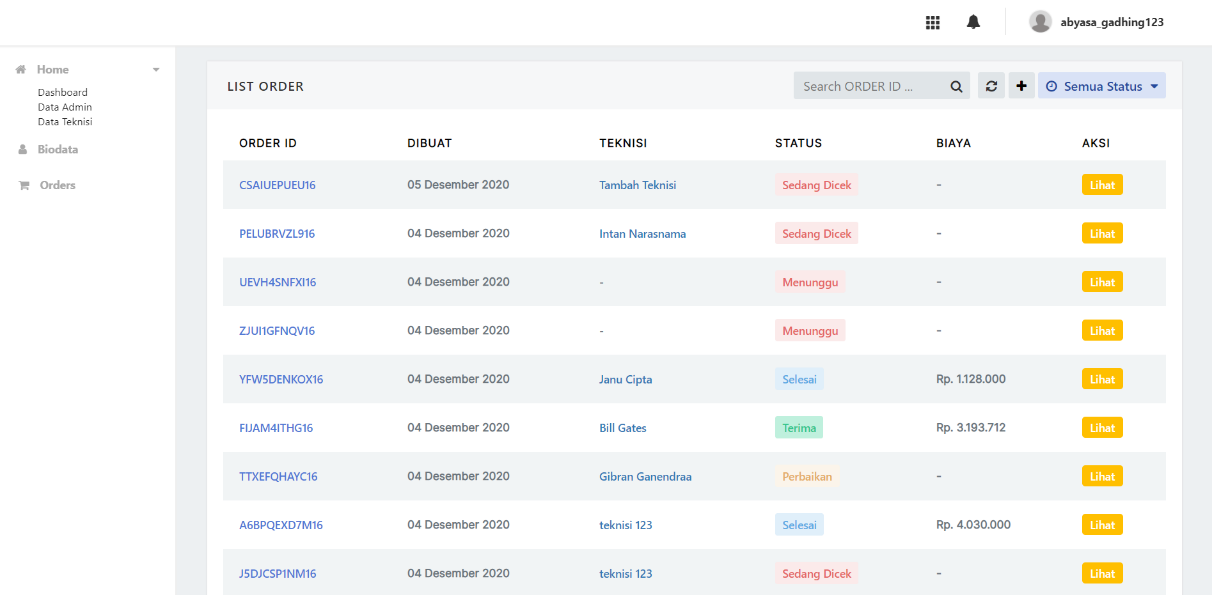
Gambar 4.83 Halaman *Detail* *Admin*

1. Halaman Ubah Biodata

Halaman Ubah Biodata berisikan kolom – kolom profil dan kontak yang dapat diubah seperti nama, username, jenis kelamin, alamat, email serta nomor telepon. Dalam halaman biodata, teknisi juga dapat mengubah passwordnya, melihat grafik keberhasilan perbaikan serta dapat mengubah foto profilnya dan dapat dilihat pada Gambar 4.84.

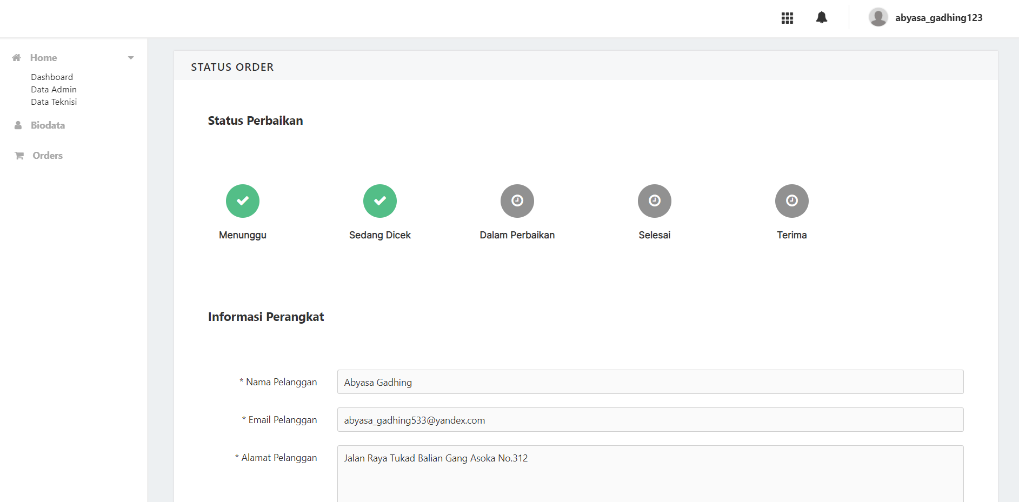
Gambar 4.84 Halaman Ubah Biodata

1. Halaman *Order*

Halaman *Order* berisikan daftar mengenai *order* yang dimiliki oleh pelanggan beserta beberapa informasinya seperti *id* *order*, waktu dibuat, nama teknisi yang mengambil *order*, *status* *order*, serta total biaya untuk *order* dilihat pada Gambar 4.85.

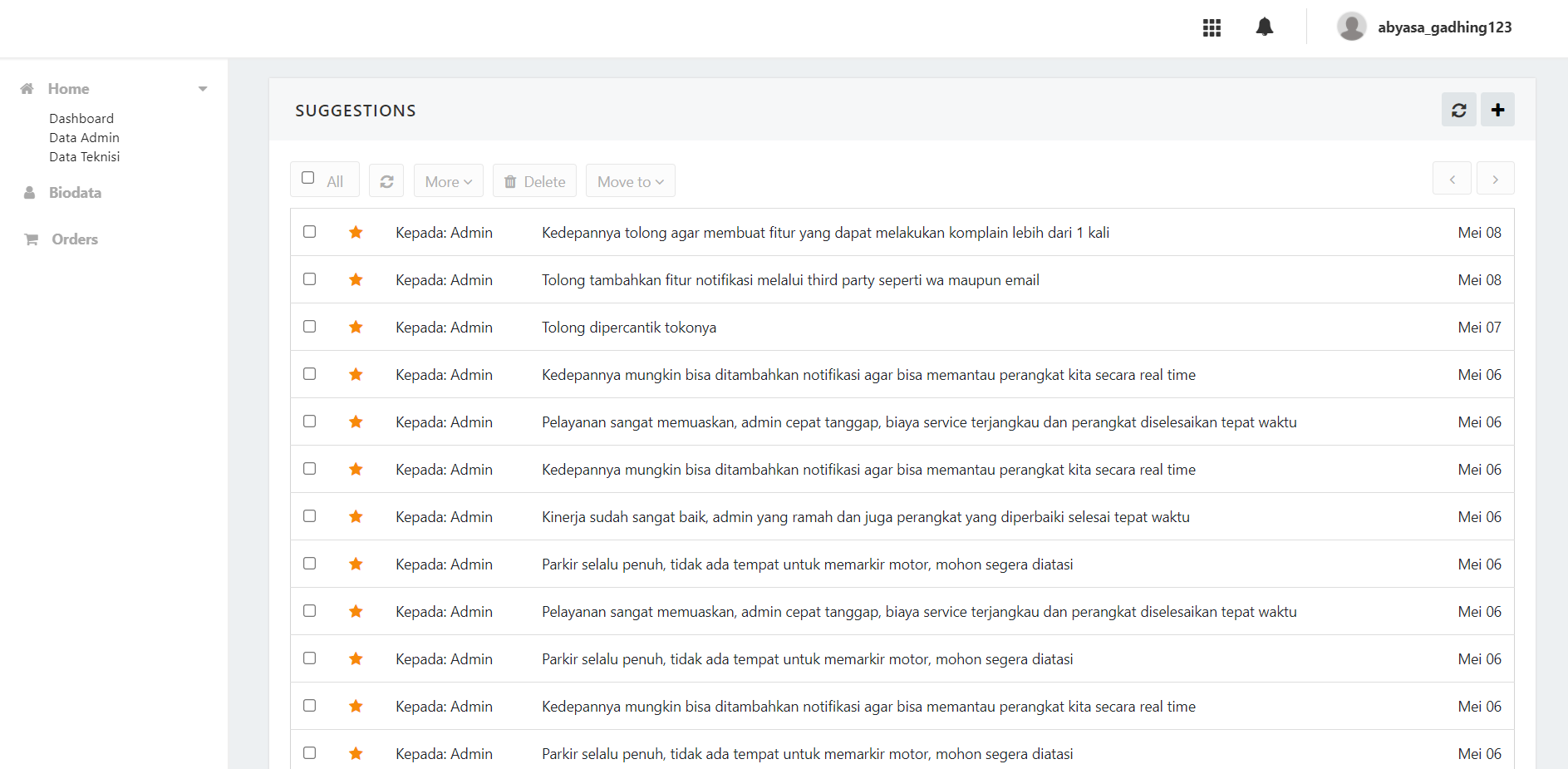
Gambar 4.85 Halaman *Order*

1. Halaman Lihat *Order*

Halaman Lihat *Order* merupakan halaman yang menampilkan data *order* milih pelanggan seperti status *order*, nama pelanggan, *email* pelanggan, alamat pelanggan, *spare part* yang diganti, nama perangkat, keluhan perangkat, tipe perangkat, merek perangkat, jenis jasa, teknisi yang mengerjakan, komplain yang dibuat, serta total yang harus dibayar dan dapat dilihat pada Gambar 4.86.

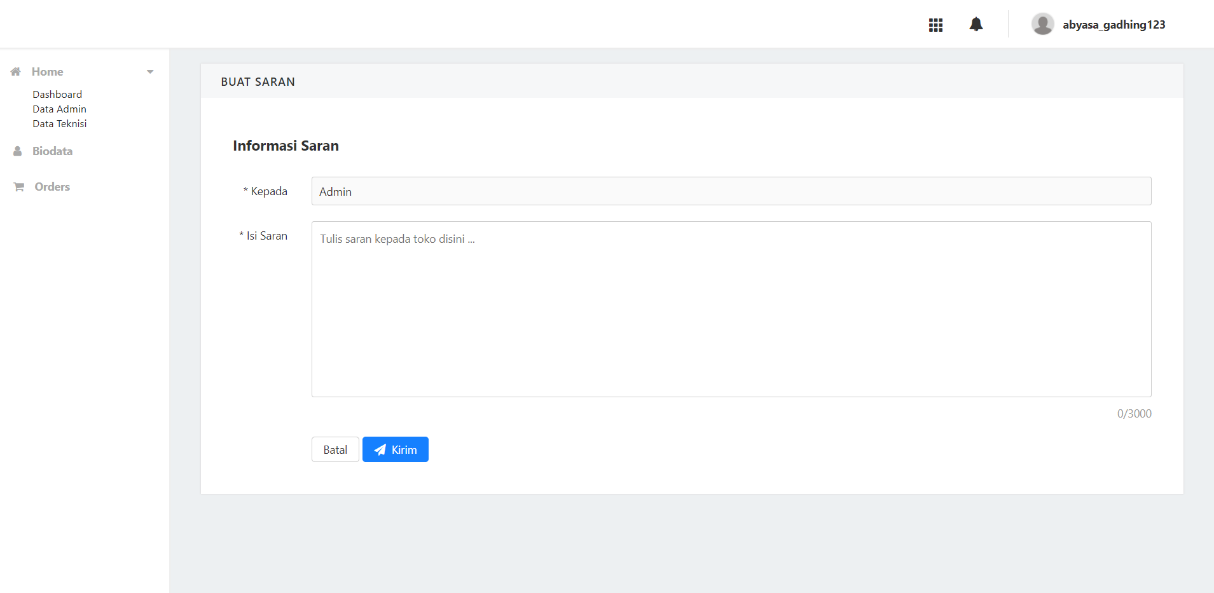
Gambar 4.86 Halaman Lihat *Order*

1. Halaman Saran

Halaman Saran merupakan halaman yang menampilkan daftar saran yang dikirimkan pelanggan kepada toko dan diurutkan dari tanggal terbaru dan dapat dilihat pada Gambar 4.86.

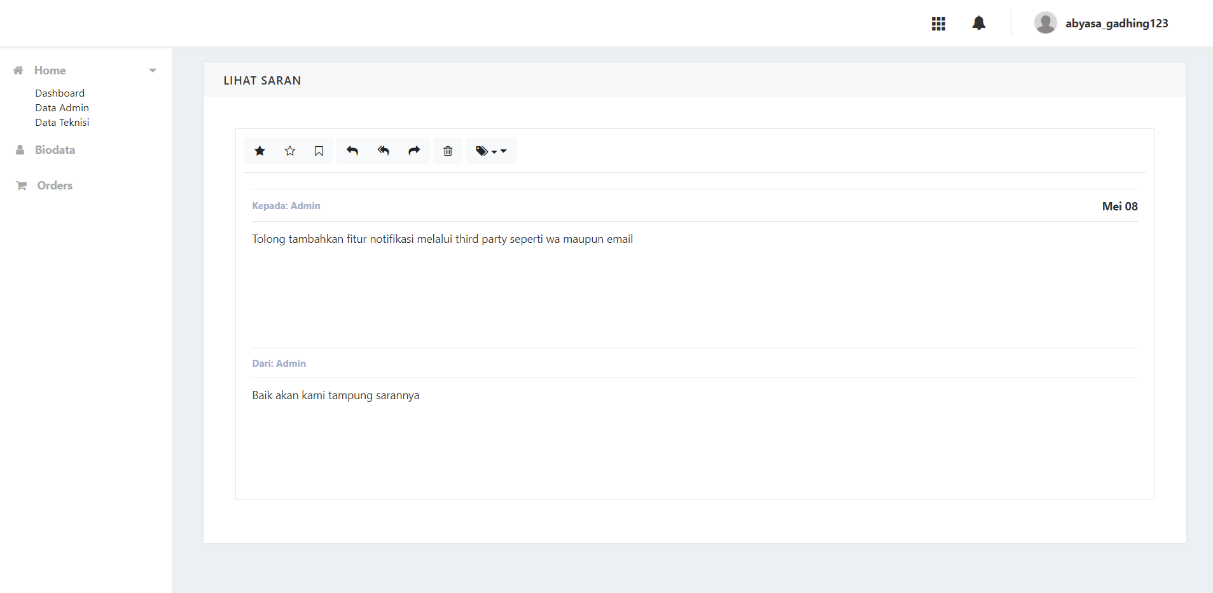
Gambar 4.86 Halaman Saran

1. Halaman Kirim Saran

Halaman Kirim Saran merupakan halaman yang berisi kolom isi saran yang akan dikirimkan kepada toko dan dapat dilihat pada Gambar 4.87.

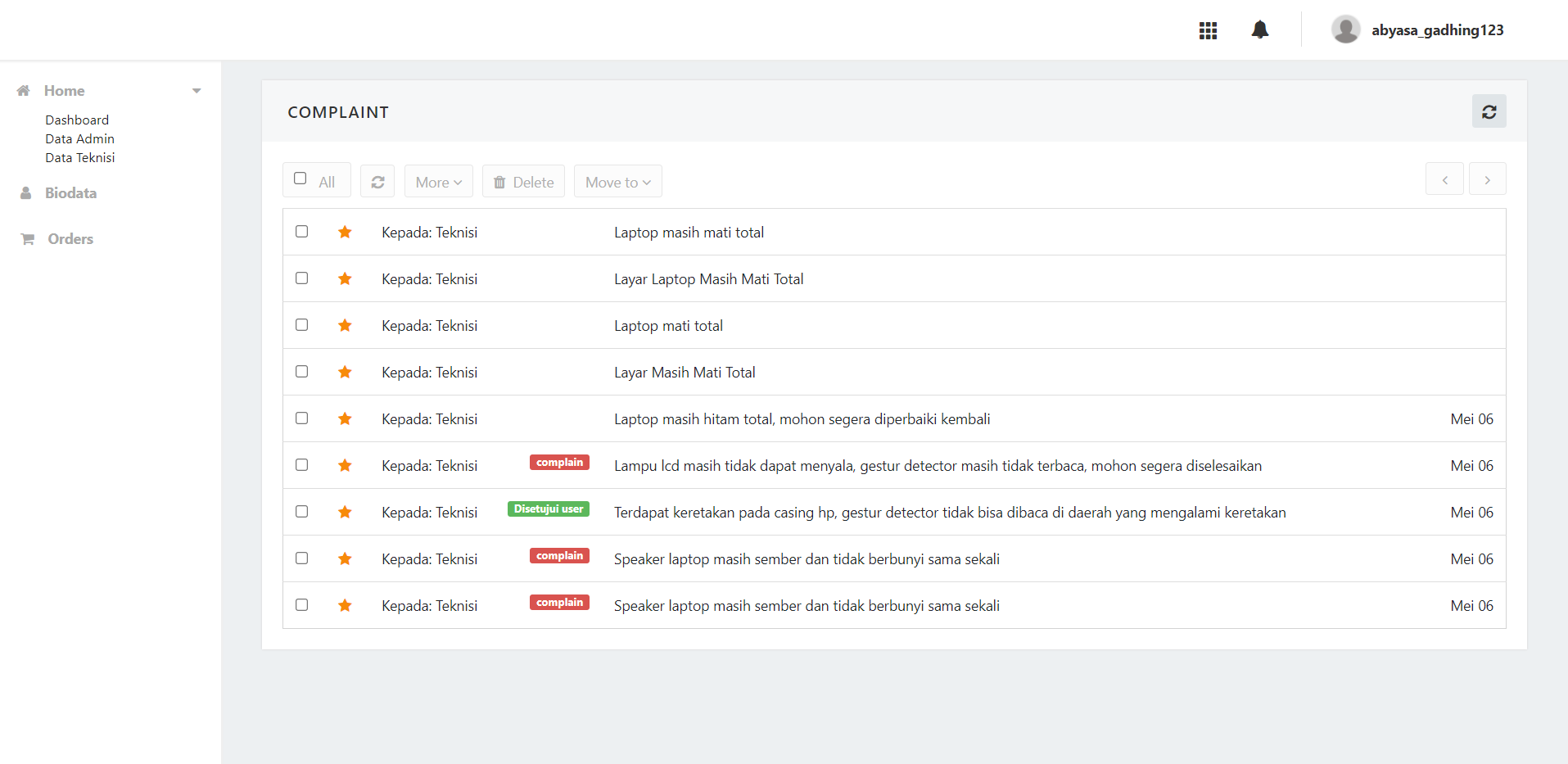
Gambar 4.87 Halaman Kirim Saran

1. Halaman Lihat Saran

Halaman Lihat Saran merupakan halaman yang menampilkan data mengenai saran yang dikirimkan kepada toko seperti nama pelanggan, email pelanggan, tanggal dikirim, isi saran serta balasan admin terhadap saran tersebut dan dapat dilihat pada Gambar 4.88.

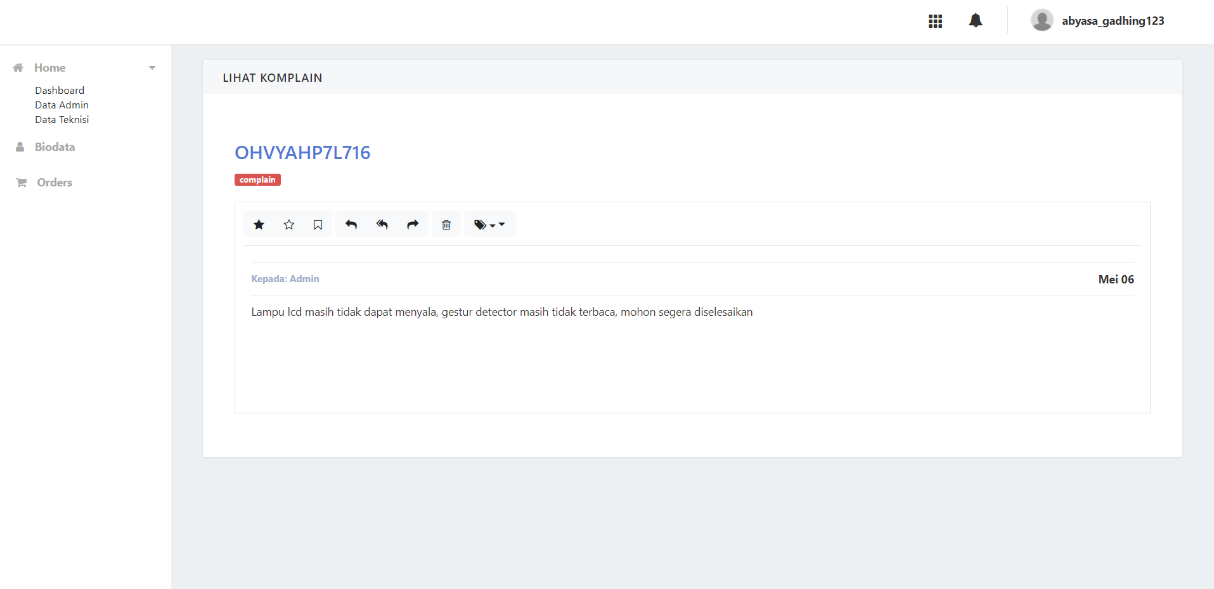
Gambar 4.88 Halaman Lihat Saran

1. Halaman Komplain

Halaman Komplain merupakan halaman yang menampilkan daftar komplain yang dibuat oleh pelanggan serta terdapat *bagde* status dari komplain tersebut dan dapat dilihat pada Gambar 4.89.

Gambar 4.89 Halaman Komplain

1. Halaman Lihat Komplain

Halaman Lihat Komplain merupakan halaman yang menampilkan data komplain yang dibuat oleh pelanggan beserta informasinya seperti nama pelanggan, *email* pelanggan, tanggal dibuat, serta *id order* yang dapat di klik untuk menampilkan informasi *order*. Pada halaman lihat komplain, pelanggan dapat mengubah status komplain menjadi “sudah disetujui” dengan menekan tombol “setujui” dan dapat dilihat pada Gambar 4.90.

Gambar 4.90 Halaman Lihat Komplain

1. **Hasil Pengujian *Black Box Testing***

Hasil Pengujian *Black Box Testing* yang telah dibuat ini digunakan untuk menunjukkan bahwa fungsi – fungsi yang ada pada aplikasi berjalan sesuai dengan yang diinginkan dan juga bertujuan untuk mengetahui kesalahan – kesalahan yang ada pada aplikasi tersebut.

* + 1. **Hasil Pengujian Sebagai Admin**

Berikut merupakan hasil pengujian pada fungsi – fungsi aplikasi apabila akan melakukan *login* sebagai Admin :

1. Hasil Pengujian Halaman *Login*

Berikut merupakan hasil pengujian sistem yang dilakukan pada halaman *login* dan dapat dilihat pada Tabel 4.13.

Tabel 4.13 Hasil Pengujian Halaman *Login*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Tampil Halaman Login | Aplikasi pertama kali menampilkan halaman *login* sebelum mengakses aplikasi | Aplikasi berhasil menampilkan Halaman *Login* saat pertama kali diakses | Sesuai |
| 2 | Input E-mail:  Salah Atau Kosong  Input Password:  Salah Atau Kosong | Data *login* pengguna ditolak dan muncul pesan peringatan *input* salah atau kosong | Aplikasi berhasil menolak data *login* pengguna dan berhasil memunculkan peringatan bahwa *input* salah atau kosong | Sesuai |
| 3 | Input E-mail:  Benar  Input Password:  Salah Atau Kosong | Data *login* pengguna ditolak dan muncul pesan peringatan *input* salah atau kosong | Aplikasi berhasil menolak data *login* pengguna dan berhasil memunculkan peringatan bahwa *input* salah atau kosong | Sesuai |
| 4 | Input E-mail:  Salah Atau Kosong  Input Password:  Benar | Data *login* pengguna ditolak dan muncul pesan peringatan *input* salah atau kosong | Aplikasi berhasil menolak data *login* pengguna dan berhasil memunculkan peringatan bahwa *input* salah atau kosong | Sesuai |
| 5 | Input E-mail:  Benar  Input Password:  Benar | Aplikasi akan menerima *data login* dan meneruskannya ke halaman *Dashboard* | Aplikasi berhasil menerima data login pengguna dan meneruskannya ke halaman Dashboard | Sesuai |

1. Hasil Pengujian Halaman *Dashboard*

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman *Dashboard* dan dapat dilihat pada Tabel 4.14.

Tabel 4.14 Hasil Pengujian Halaman *Dashboard*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Klik Sub Menu Data Pelanggan | Aplikasi menampilkan Halaman Data Pelanggan | Aplikasi berhasil menampilkan Halaman Data Pelanggan | Sesuai |
| 2 | Klik Sub Menu Spare Part | Aplikasi menampilkan Halaman Data Spare Part | Aplikasi berhasil menampilkan Halaman Data Spare Part | Sesuai |
| 3 | Klik Sub Menu Admin | Aplikasi menampilkan Halaman Data Admin | Aplikasi berhasil menampilkan Halaman Data Admin | Sesuai |
| 4 | Klik Sub Menu Teknisi | Aplikasi menampilkan Halaman Data Teknisi | Aplikasi berhasil menampilkan Halaman Data Teknisi | Sesuai |
| 5 | Klik Menu Biodata | Aplikasi menampilkan Halaman Data Teknisi | Aplikasi berhasil menampilkan Halaman Data Teknisi | Sesuai |
| 6 | Klik Menu Jasa | Aplikasi menampilkan Halaman Jasa | Aplikasi berhasil menampilkan Halaman Jasa | Sesuai |
| 7 | Klik Menu Order | Aplikasi menampilkan Halaman Order | Aplikasi berhasil menampilkan Halaman Order | Sesuai |
| 8 | Klik Menu Laporan | Aplikasi menampilkan Halaman Laporan | Aplikasi berhasil menampilkan Halaman Laporan | Sesuai |

1. Hasil Pengujian Halaman Pelanggan

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Pelanggan dan dapat dilihat pada Tabel 4.15.

Tabel 4.15 Hasil Pengujian Halaman Pelanggan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Klik tombol Tambah Pelanggan | Aplikasi menampilkan Form Tambah Pelanggan | Aplikasi berhasil menampilkan Form Tambah Pelanggan | Sesuai |
| 2 | Klik tombol Hapus Pelanggan | Aplikasi menampilkan *modal* Hapus Pelanggan | Aplikasi berhasil menampilkan *modal* Hapus Pelanggan | Sesuai |
| 3 | *Input* nama pelanggan pada kolom pencarian lalu klik tombol cari | Aplikasi menampilkan pelanggan yang dicari berdasarkan input yang diberikan | Aplikasi berhasil menampilkan pelanggan yang dicari berdasarkan *input* yang diberikan | Sesuai |
| 4 | Klik salah satu daftar pelanggan | Aplikasi menampilkan Halaman Lihat Pelanggan | Aplikasi berhasil menampilkan Halaman Lihat Pelanggan | Sesuai |

1. Hasil Pengujian Halaman Lihat Data Pelanggan

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Lihat Data Pelanggan dan dapat dilihat pada Tabel 4.16.

Tabel 4.16 Hasil Pengujian Halaman Lihat Data Pelanggan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Tampil data Profil Pelanggan dan Kontak Pelanggan | Aplikasi menampilkan data Profil Pelanggan dan Kontak Pelanggan | Aplikasi berhasil menampilkan data Profil Pelanggan dan Kontak Pelanggan | Sesuai |

1. Hasil Pengujian Form Tambah Pelanggan

Berikut merupakan hasil pengujian sistem yang dilakukan pada Form Tambah Pelanggan dan dapat dilihat pada Tabel 4.17.

Tabel 4.17 Form Tambah Pelanggan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Input nama kosong | Data pelanggan ditolak dan muncul pesan peringatan nama harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nama harus diisi | Sesuai |
| 2 | Input nama memiliki karakter kurang dari 4 (empat) | Data pelanggan ditolak dan muncul peringatan nama minimal memiliki 4 (empat) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nama minimal memiliki 4 (empat) karakter | Sesuai |
| 3 | Input nama memiliki karakter lebih dari 60 (enam puluh) | Data pelanggan ditolak dan muncul peringatan nama maksimal memiliki 60 (enam puluh) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nama maksimal memiliki 60 (enam puluh) karakter | Sesuai |
| 4 | Input username kosong | Data pelanggan ditolak dan muncul pesan peringatan username harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa username harus diisi | Sesuai |
| 5 | Input username memiliki karakter kurang dari 6 (enam) | Data pelanggan ditolak dan muncul peringatan username minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa username minimal memiliki 6 (enam) karakter | Sesuai |
| 6 | Input username memiliki karakter lebih dari 40 (empat puluh) | Data pelanggan ditolak dan muncul peringatan username maksimal memiliki 40 (empat puluh) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa username maksimal memiliki 40 (empat puluh) karakter | Sesuai |
| 7 | Input username memiliki spasi | Data pelanggan ditolak dan muncul peringatan bahwa username tidak boleh memiliki spasi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki spasi | Sesuai |
| 8 | Input username memiliki huruf besar | Data pelanggan ditolak dan muncul peringatan bahwa username tidak boleh memiliki huruf besar | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki huruf besar | Sesuai |
| 9 | Input *email* kosong | Data pelanggan ditolak dan muncul pesan peringatan *email* harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa *email* harus diisi | Sesuai |
| 10 | Input *email* memiliki karakter kurang dari 6 (enam) | Data pelanggan ditolak dan muncul peringatan *email* minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa *email* minimal memiliki 6 (enam) karakter | Sesuai |
| 11 | Input *email* memiliki karakter lebih dari 255 (dua ratus lima puluh lima) | Data pelanggan ditolak dan muncul peringatan *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Sesuai |
| 12 | Input jenis kelamin kosong | Data pelanggan ditolak dan muncul pesan peringatan jenis kelamin harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa jenis kelamin harus diisi | Sesuai |
| 13 | Input nomor telepon kosong | Data pelanggan ditolak dan muncul pesan peringatan nomor telepon harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nomor telepon harus diisi | Sesuai |
| 14 | Input nomor telepon memiliki karakter kurang dari 7 (tujuh) | Data pelanggan ditolak dan muncul peringatan input nomor telepon minimal memiliki 7 (tujuh) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nomor telepon minimal memiliki 7 (tujuh) karakter | Sesuai |
| 15 | Input nomor telepon memiliki karakter lebih dari 17 (tujuh belas) | Data pelanggan ditolak dan muncul peringatan nomor telepon maksimal memiliki 17 (tujuh belas) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa nomor telepon maksimal memiliki 17 (tujuh belas) karakter | Sesuai |
| 16 | Input alamat kosong | Data pelanggan ditolak dan muncul pesan peringatan alamat harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa alamat harus diisi | Sesuai |
| 17 | Input alamat memiliki karakter kurang dari 8 (delapan) | Data pelanggan ditolak dan muncul peringatan input alamat minimal memiliki 8 (delapan) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa input alamat minimal memiliki 8 (delapan) karakter | Sesuai |
| 18 | Input alamat memiliki karakter lebih dari 100 (seratus) | Data pelanggan ditolak dan muncul peringatan alamat maksimal memiliki 100 (seratus) karakter | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa alamat maksimal memiliki 100 (seratus) karakter | Sesuai |
| 19 | Input nama, username, jenis kelamin, nomor telepon, email, alamat: Benar | Aplikasi menyimpan data pelanggan | Aplikasi berhasil menyimpan data pelanggan | Sesuai |

1. Hasil Pengujian Halaman *Spare Part*

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Spare Part dan dapat dilihat pada Tabel 4.18.

Tabel 4.18 Form Tambah Pelanggan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Klik Tombol Tambah Spare Part | Aplikasi menampilkan halaman Tambah Spare Part | Aplikasi berhasil menampilkan halaman Tambah Spare Part | Sesuai |
| 2 | Klik Tombol Hapus Spare Part | Aplikasi menampilkan *modal* Hapus Spare Part | Aplikasi berhasil menampilkan *modal* Hapus Spare Part | Sesuai |
| 3 | *Input* nama *spare part* pada kolom pencarian lalu klik tombol cari | Aplikasi menampilkan *spare part* yang dicari berdasarkan input yang diberikan | Aplikasi berhasil menampilkan *spare part* yang dicari berdasarkan *input* yang diberikan | Sesuai |
| 4 | Klik salah satu daftar *spare part* | Aplikasi menampilkan Halaman Ubah *Spare Part* | Aplikasi berhasil menampilkan Halaman Ubah *Spare Part* | Sesuai |

1. Hasil Pengujian Halaman Ubah *Spare Part*

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Ubah Spare Part dan dapat dilihat pada Tabel 4.19.

Tabel 4.19 Hasil Pengujian Halaman Ubah *Spare Part*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Foto Spare Part Kosong | Data spare part ditolak dan muncul pesan peringatan bahwa foto harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa foto *spare part* harus diisi | Sesuai |
| 2 | Input nama *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan bahwa nama *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* harus diisi | Sesuai |
| 3 | Input nama spare part memiliki karakter kurang dari 7 (tujuh) | Data *spare part* ditolak dan muncul peringatan bahwa nama *spare part* minimal memiliki 7 (tujuh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* minimal memiliki 7 (tujuh) karakter | Sesuai |
| 4 | Input nama *spare part* memiliki karakter lebih dari 70 (tujuh puluh) | Data *spare part* ditolak dan muncul peringatan bahwa nama *spare part* maksimal memiliki 70 (tujuh puluh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* maksimal memiliki 70 (tujuh puluh) karakter | Sesuai |
| 5 | Input deskripsi *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan bahwa deskripsi *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* harus diisi | Sesuai |
| 6 | Input deskripsi *spare part* memiliki karakter kurang dari 10 (sepuluh) | Data *spare part* ditolak dan muncul peringatan deskripsi *spare part* minimal memiliki 10 (sepuluh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* minimal memiliki 10 (sepuluh) karakter | Sesuai |
| 7 | Input deskripsi *spare part* memiliki karakter lebih dari 3000 (tiga ribu) | Data *spare part* ditolak dan muncul peringatan deskripsi *spare part* maksimal memiliki 3000 (tiga ribu) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* maksimal memiliki 3000 (tiga ribu) karakter | Sesuai |
| 8 | Input stok *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan stok *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* harus diisi | Sesuai |
| 9 | Input stok *spare part* tidak berupa angka | Data *spare part* ditolak dan muncul pesan peringatan stok *spare part* harus berupa angka | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* harus berupa angka | Sesuai |
| 10 | Input stok *spare part* memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan stok *spare part* minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* minimal memiliki angka 1 (satu) | Sesuai |
| 11 | Input stok *spare part* memiliki angka lebih dari 65000 (enam puluh lima ribu) | Data *spare part* ditolak dan muncul peringatan stok *spare part* maksimal memiliki angka 65000 (enam puluh lima ribu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* maksimal memiliki angka 65000 (enam puluh lima ribu) | Sesuai |
| 12 | Input harga asli kosong | Data *spare part* ditolak dan muncul pesan peringatan input harga asli harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli harus diisi | Sesuai |
| 13 | Input harga asli memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan harga asli minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli minimal memiliki angka 1 (satu) | Sesuai |
| 14 | Input harga asli memiliki angka lebih dari 300000000 (tiga ratus juta) | Data *spare part* ditolak dan muncul peringatan harga asli maksimal memiliki angka 300000000 (tiga ratus juta) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli maksimal memiliki angka 300000000 (tiga ratus juta) | Sesuai |
| 15 | Input harga kosong | Data *spare part* ditolak dan muncul pesan peringatan input harga harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga harus diisi | Sesuai |
| 16 | Input harga memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan harga minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga minimal memiliki angka 1 (satu) | Sesuai |
| 17 | Input harga memiliki angka lebih dari 300000000 (tiga ratus juta) | Data *spare part* ditolak dan muncul peringatan harga maksimal memiliki angka 300000000 (tiga ratus juta) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga maksimal memiliki angka 300000000 (tiga ratus juta) | Sesuai |
| 18 | Input foto, nama, tipe, stok, harga, harga asli:  Benar | Aplikasi menyimpan data *spare part* | Aplikasi berhasil menyimpan data *spare part* | Sesuai |
| 19 | Klik tombol Kembali | Aplikasi menampilkan halaman ke halaman sebelumnya | Aplikasi berhasil menampilkan halaman ke halaman sebelumnya | Sesuai |

1. Hasil Pengujian Halaman Tambah *Spare Part*

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Tanbah *Spare Part* dan dapat dilihat pada Tabel 4.20.

Tabel 4.20 Hasil Pengujian Halaman Tambah *Spare Part*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Foto Spare Part Kosong | Data spare part ditolak dan muncul pesan peringatan bahwa foto harus diisi | Aplikasi berhasil menolak data pelanggan dan berhasil memunculkan peringatan bahwa foto *spare part* harus diisi | Sesuai |
| 2 | Input nama *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan bahwa nama *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* harus diisi | Sesuai |
| 3 | Input nama spare part memiliki karakter kurang dari 7 (tujuh) | Data *spare part* ditolak dan muncul peringatan bahwa nama *spare part* minimal memiliki 7 (tujuh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* minimal memiliki 7 (tujuh) karakter | Sesuai |
| 4 | Input nama *spare part* memiliki karakter lebih dari 70 (tujuh puluh) | Data *spare part* ditolak dan muncul peringatan bahwa nama *spare part* maksimal memiliki 70 (tujuh puluh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa nama *spare part* maksimal memiliki 70 (tujuh puluh) karakter | Sesuai |
| 5 | Input deskripsi *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan bahwa deskripsi *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* harus diisi | Sesuai |
| 6 | Input deskripsi *spare part* memiliki karakter kurang dari 10 (sepuluh) | Data *spare part* ditolak dan muncul peringatan deskripsi *spare part* minimal memiliki 10 (sepuluh) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* minimal memiliki 10 (sepuluh) karakter | Sesuai |
| 7 | Input deskripsi *spare part* memiliki karakter lebih dari 3000 (tiga ribu) | Data *spare part* ditolak dan muncul peringatan deskripsi *spare part* maksimal memiliki 3000 (tiga ribu) karakter | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa deskripsi *spare part* maksimal memiliki 3000 (tiga ribu) karakter | Sesuai |
| 8 | Input stok *spare part* kosong | Data *spare part* ditolak dan muncul pesan peringatan stok *spare part* harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* harus diisi | Sesuai |
| 9 | Input stok *spare part* tidak berupa angka | Data *spare part* ditolak dan muncul pesan peringatan stok *spare part* harus berupa angka | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* harus berupa angka | Sesuai |
| 10 | Input stok *spare part* memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan stok *spare part* minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* minimal memiliki angka 1 (satu) | Sesuai |
| 11 | Input stok *spare part* memiliki angka lebih dari 65000 (enam puluh lima ribu) | Data *spare part* ditolak dan muncul peringatan stok *spare part* maksimal memiliki angka 65000 (enam puluh lima ribu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa stok *spare part* maksimal memiliki angka 65000 (enam puluh lima ribu) | Sesuai |
| 12 | Input harga asli kosong | Data *spare part* ditolak dan muncul pesan peringatan input harga asli harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli harus diisi | Sesuai |
| 13 | Input harga asli memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan harga asli minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli minimal memiliki angka 1 (satu) | Sesuai |
| 14 | Input harga asli memiliki angka lebih dari 300000000 (tiga ratus juta) | Data *spare part* ditolak dan muncul peringatan harga asli maksimal memiliki angka 300000000 (tiga ratus juta) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga asli maksimal memiliki angka 300000000 (tiga ratus juta) | Sesuai |
| 15 | Input harga kosong | Data *spare part* ditolak dan muncul pesan peringatan input harga harus diisi | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga harus diisi | Sesuai |
| 16 | Input harga memiliki angka kurang dari 1 (satu) | Data *spare part* ditolak dan muncul peringatan harga minimal memiliki angka 1 (satu) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga minimal memiliki angka 1 (satu) | Sesuai |
| 17 | Input harga memiliki angka lebih dari 300000000 (tiga ratus juta) | Data *spare part* ditolak dan muncul peringatan harga maksimal memiliki angka 300000000 (tiga ratus juta) | Aplikasi berhasil menolak data *spare part* dan berhasil memunculkan peringatan bahwa harga maksimal memiliki angka 300000000 (tiga ratus juta) | Sesuai |
| 18 | Input foto, nama, tipe, stok, harga, harga asli:  Benar | Aplikasi menyimpan data *spare part* | Aplikasi berhasil menyimpan data *spare part* | Sesuai |
| 19 | Klik tombol Kembali | Aplikasi menampilkan halaman ke halaman sebelumnya | Aplikasi berhasil menampilkan halaman ke halaman sebelumnya | Sesuai |

1. Hasil Pengujian Halaman Admin

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Admin dan dapat dilihat pada Tabel 4.21.

Tabel 4.21 Hasil Pengujian Halaman Pelanggan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Klik tombol Tambah Admin | Aplikasi menampilkan Form Tambah Admin | Aplikasi berhasil menampilkan Form Tambah Admin | Sesuai |
| 2 | Klik tombol Hapus Admin | Aplikasi menampilkan *modal* Hapus Admin | Aplikasi berhasil menampilkan *modal* Hapus Admin | Sesuai |
| 3 | *Input* nama admin pada kolom pencarian lalu klik tombol cari | Aplikasi menampilkan admin yang dicari berdasarkan input yang diberikan | Aplikasi berhasil menampilkan admin yang dicari berdasarkan *input* yang diberikan | Sesuai |
| 4 | Klik salah satu daftar admin | Aplikasi menampilkan Halaman Lihat Admin | Aplikasi berhasil menampilkan Halaman Lihat Admin | Sesuai |

1. Hasil Pengujian Halaman Lihat Data Admin

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Lihat Data Admin dan dapat dilihat pada Tabel 4.22.

Tabel 4.22 Hasil Pengujian Halaman Lihat Data Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Tampil data Profil Admin dan Kontak Admin | Aplikasi menampilkan data Profil Admin dan Kontak Admin | Aplikasi berhasil menampilkan data Profil Admin dan Kontak Admin | Sesuai |

1. Hasil Pengujian Form Tambah Admin

Berikut merupakan hasil pengujian sistem yang dilakukan pada Form Tambah Admin dan dapat dilihat pada Tabel 4.23.

Tabel 4.23 Form Tambah Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Input nama kosong | Data admin ditolak dan muncul pesan peringatan nama harus diisi | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa nama harus diisi | Sesuai |
| 2 | Input nama memiliki karakter kurang dari 4 (empat) | Data admin ditolak dan muncul peringatan nama minimal memiliki 4 (empat) karakter | Aplikasi berhasil menolak data adnub dan berhasil memunculkan peringatan bahwa nama minimal memiliki 4 (empat) karakter | Sesuai |
| 3 | Input nama memiliki karakter lebih dari 60 (enam puluh) | Data admin ditolak dan muncul peringatan nama maksimal memiliki 60 (enam puluh) karakter | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa nama maksimal memiliki 60 (enam puluh) karakter | Sesuai |
| 4 | Input username kosong | Data admin ditolak dan muncul pesan peringatan username harus diisi | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa username harus diisi | Sesuai |
| 5 | Input username memiliki karakter kurang dari 6 (enam) | Data admin ditolak dan muncul peringatan username minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa username minimal memiliki 6 (enam) karakter | Sesuai |
| 6 | Input username memiliki karakter lebih dari 40 (empat puluh) | Data admin ditolak dan muncul peringatan username maksimal memiliki 40 (empat puluh) karakter | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa username maksimal memiliki 40 (empat puluh) karakter | Sesuai |
| 7 | Input username memiliki spasi | Data admin ditolak dan muncul peringatan bahwa username tidak boleh memiliki spasi | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki spasi | Sesuai |
| 8 | Input username memiliki huruf besar | Data admin ditolak dan muncul peringatan bahwa username tidak boleh memiliki huruf besar | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki huruf besar | Sesuai |
| 9 | Input *email* kosong | Data admin ditolak dan muncul pesan peringatan *email* harus diisi | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa *email* harus diisi | Sesuai |
| 10 | Input *email* memiliki karakter kurang dari 6 (enam) | Data admin ditolak dan muncul peringatan *email* minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa *email* minimal memiliki 6 (enam) karakter | Sesuai |
| 11 | Input *email* memiliki karakter lebih dari 255 (dua ratus lima puluh lima) | Data admin ditolak dan muncul peringatan *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Sesuai |
| 12 | Input jenis kelamin kosong | Data admin ditolak dan muncul pesan peringatan jenis kelamin harus diisi | Aplikasi berhasil menolak data admin dan berhasil memunculkan peringatan bahwa jenis kelamin harus diisi | Sesuai |
| 13 | Input nama, username, jenis kelamin, email: Benar | Aplikasi menyimpan data admin | Aplikasi berhasil menyimpan data admin | Sesuai |

1. Hasil Pengujian Halaman Teknisi

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Teknisi dan dapat dilihat pada Tabel 4.24.

Tabel 4.24 Hasil Pengujian Halaman Teknisi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Klik tombol Tambah Teknisi | Aplikasi menampilkan Form Tambah Teknisi | Aplikasi berhasil menampilkan Form Tambah Teknisi | Sesuai |
| 2 | Klik tombol Hapus Teknisi | Aplikasi menampilkan *modal* Hapus Teknisi | Aplikasi berhasil menampilkan *modal* Hapus Teknisi | Sesuai |
| 3 | *Input* nama teknisi pada kolom pencarian lalu klik tombol cari | Aplikasi menampilkan teknisi yang dicari berdasarkan input yang diberikan | Aplikasi berhasil menampilkan teknisi yang dicari berdasarkan *input* yang diberikan | Sesuai |
| 4 | Klik salah satu daftar teknisi | Aplikasi menampilkan Halaman Lihat Teknisi | Aplikasi berhasil menampilkan Halaman Lihat Teknisi | Sesuai |

1. Hasil Pengujian Halaman Lihat Data Teknisi

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Lihat Data Teknisi dan dapat dilihat pada Tabel 4.25.

Tabel 4.25 Hasil Pengujian Halaman Lihat Data Teknisi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Tampil data Profil Teknisi dan Kontak Teknisi | Aplikasi menampilkan data Profil Teknisi dan Kontak Teknisi | Aplikasi berhasil menampilkan data Profil Teknisi dan Kontak Teknisi | Sesuai |
| 2 | Tampil Grafik Keberhasilan Perbaikan | Aplikasi menampilkan Grafik Keberhasilan Perbaikan | Aplikasi berhasil menampilkan Grafik Keberhasilan Perbaikan Teknisi | Sesuai |
| 3 | Klik Tombol *Reset Password* | Aplikasi *mereset password* teknisi | Aplikasi berhasil *mereset password* teknisi | Sesuai |

1. Hasil Pengujian Halaman Ubah Biodata

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Ubah Biodata dan dapat dilihat pada Tabel 4.25.

Tabel 4.25 Hasil Pengujian Halaman Ubah Biodata

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Kelas Uji | Hasil Yang Diharapkan | Hasil Pengujian | Keterangan |
| 1 | Input nama kosong | Data biodata ditolak dan muncul pesan peringatan nama harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa nama harus diisi | Sesuai |
| 2 | Input nama memiliki karakter kurang dari 6 (enam) | Data biodata ditolak dan muncul peringatan nama minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa nama minimal memiliki 6 (enam) karakter | Sesuai |
| 3 | Input nama memiliki karakter lebih dari 60 (enam puluh) | Data biodata ditolak dan muncul peringatan nama maksimal memiliki 60 (enam puluh) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa nama maksimal memiliki 60 (enam puluh) karakter | Sesuai |
| 4 | Input username kosong | Data biodata ditolak dan muncul pesan peringatan username harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa username harus diisi | Sesuai |
| 5 | Input username memiliki karakter kurang dari 6 (enam) | Data biodata ditolak dan muncul peringatan username minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa username minimal memiliki 6 (enam) karakter | Sesuai |
| 6 | Input username memiliki karakter lebih dari 40 (empat puluh) | Data biodata ditolak dan muncul peringatan username maksimal memiliki 40 (empat puluh) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa username maksimal memiliki 40 (empat puluh) karakter | Sesuai |
| 7 | Input username memiliki spasi | Data biodata ditolak dan muncul peringatan bahwa username tidak boleh memiliki spasi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki spasi | Sesuai |
| 8 | Input username memiliki huruf besar | Data biodata ditolak dan muncul peringatan bahwa username tidak boleh memiliki huruf besar | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa username tidak boleh memiliki huruf besar | Sesuai |
| 9 | Input *email* kosong | Data biodata ditolak dan muncul pesan peringatan *email* harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa *email* harus diisi | Sesuai |
| 10 | Input *email* memiliki karakter kurang dari 6 (enam) | Data biodata ditolak dan muncul peringatan *email* minimal memiliki 6 (enam) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa *email* minimal memiliki 6 (enam) karakter | Sesuai |
| 11 | Input *email* memiliki karakter lebih dari 255 (dua ratus lima puluh lima) | Data biodata ditolak dan muncul peringatan *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa *email* maksimal memiliki 255 (dua ratus lima puluh lima) karakter | Sesuai |
| 12 | Input jenis kelamin kosong | Data biodata ditolak dan muncul pesan peringatan jenis kelamin harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa jenis kelamin harus diisi | Sesuai |
| 13 | Input nomor telepon kosong | Data biodata ditolak dan muncul pesan peringatan nomor telepon harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa nomor telepon harus diisi | Sesuai |
| 14 | Input nomor telepon memiliki karakter kurang dari 7 (tujuh) atau lebih dari 17 (tujuh belas) | Data biodata ditolak dan muncul peringatan input nomor telepon minimal memiliki 7 (tujuh) karakter atau maksimal memiliki 17 (tujuh belas) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa nomor telepon minimal memiliki 7 (tujuh) karakter atau maksimal memiliki 17 (tujuh belas) karakter | Sesuai |
| 16 | Input alamat kosong | Data biodata ditolak dan muncul pesan peringatan alamat harus diisi | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa alamat harus diisi | Sesuai |
| 17 | Input alamat memiliki karakter kurang dari 8 (delapan) | Data biodata ditolak dan muncul peringatan input alamat minimal memiliki 8 (delapan) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa input alamat minimal memiliki 8 (delapan) karakter | Sesuai |
| 18 | Input alamat memiliki karakter lebih dari 100 (seratus) | Data biodata ditolak dan muncul peringatan alamat maksimal memiliki 100 (seratus) karakter | Aplikasi berhasil menolak data biodata dan berhasil memunculkan peringatan bahwa alamat maksimal memiliki 100 (seratus) karakter | Sesuai |
| 19 | Input nama, username, jenis kelamin, nomor telepon, email, alamat: Benar | Aplikasi menyimpan data biodata | Aplikasi berhasil menyimpan data biodata | Sesuai |
| 20 | Klik Tombol Pilih Avatar | Aplikasi menampilkan kotak *file explorer* untuk memilih foto yang akan diupload | Aplikasi berhasil menampilkan kotak *file explorer* untuk memilih foto yang akan diupload | Sesuai |
| 21 | Klik Tombol Ubah *Password* | Aplikasi menampilkan *Form* Ubah *Password* | Aplikasi berhasil menampilkan *Form* Ubah *Password* | Sesuai |

1. Hasil Pengujian Halaman Jasa

Berikut merupakan hasil pengujian sistem yang dilakukan pada Halaman Jasa dan dapat dilihat pada Tabel 4.26.

Tabel 4.26 Hasil Pengujian Halaman Jasa

# DAFTAR PUSTAKA

[1] N. Luh, I. Arini, B. Priyambadha, and H. Nurwasito, “Pengembangan Sistem Manajemen Layanan Jasa Servis Alat Elektronik Rumah Tangga ( Studi Kasus : UKM Bali Tekindo Jaya ),” vol. 3, no. 8, 2019.

[2] A. Prastomo, “Sistem Informasi Pelayanan Jasa Perbaikan Peralatan Elektronik CV Sumber Teknik Cool,” *Fakt. Exata*, vol. 7, no. 4, pp. 305–316, 2014.

[3] I. G. D. A. Mahardiana, *SISTEM INFORMASI PENYEWAAN ALAT-ALAT FOTOGRAFI BERBASIS WEB*. Denpasar: STMIK STIKOM Bali, 2018.

[4] Y. Aminudin, *SISTEM INFORMASI MANAJEMEN PENGELOLAAN DATA SISWA PADA UPTD BLK INDUSTRI DAN PARIWISATA PROVINSI BALI BERBASIS WEB*. Denpasar: ITB Stikom Bali, 2020.

[5] I. A. T. Mahayani, *Sistem Pengelolaan Data Transaksi Penjualan Pada Toko Bangunan UD. Mirah Berbasis Web Responsive*. Denpasar: ITB Stikom Bali, 2020.

[6] P. Mahendrata, *Sistem Informasi Pengelolaan Arsip Pajak Klien Pada PT. Guna Artha Kencana Menggunakan Framework Laravel*. Denpasar: ITB Stikom Bali, 2019.

[7] I. M. L. A. Agatama, *Aplikasi Penjualan Dan Pengorderan Perhiasan Berbasis Web Pada Suteja Silver*. Denpasar: ITB Stikom Bali, 2019.

[8] A. Kadir, *Pengenalan Sistem Informasi*. Yogyakarta: Andi, 2014.

[9] M. R. Nahado, *Bumbu Bumbu Membuat Website*. Jakarta: Bang Roy Han, 2015.

[10] and D. R. Haviluddin, A. T. Haryono, *Aplikasi program php dan Mysql*. 10.1017/CBO9781107415324.004, 2016.

[11] E. W. Ali Z, Smitdev C, *Pemrograman Web Berbasis HTML5, PHP & Javascript*. Jakarta: PT Elex Media Komputindo, 2014.

[12] I. Y. Supardi, *Mudah dan Cepat Membuat Skripsi dengan VB 2012*. Jakarta: PT Elex Media Komputindo, 2015.

[13] S. T. Randi Adrika Putra, *Belajar Otodidak Bahasa Pemrograman SQL Menggunakan MariaDB*. Yogyakarta: Penerbit Gava Media, 2020.

[14] Aryanto, *Soal Latihan dan Jawaban Pengolahan Database MySQL Tingkat Dasar/Pemula*. Yogyakarta: Deepublish Publisher, 2016.

[15] A. Rahmat, *Seminggu Belajar Laravel*. Jakarta: Learnpub, 2014.

[16] B. Raharjo, *Belajar Otodidak Framework CodeIgniter*. Bandung: Informatika Bandung, 2018.

[17] C. Rojas, *Building Progressive Web Applications with Vue.js: Reliable, Fast, and Engaging Apps with Vue.js*. Medellin, Colombia: Apress Media LLC, 2020.

[18] E. Pratama, *Sistem Informasi dan Implementasi*. Bandung: Informatika Bandung, 2014.

[19] M. I. Perkasa and E. B. Setiawan, “Pembangunan Web Service Data Masyarakat Menggunakan REST API dengan Access Token,” *J. Ultim. Comput.*, vol. 10, no. 1, pp. 19–26, 2018, doi: 10.31937/sk.v10i1.838.