

### 3. ANALISIS DAN DESAIN SISTEM

#### 3.1 Analisis Permasalahan

Motif atau faktor pelanggan dalam membeli atau menjual emas berbeda-beda. Ada pelanggan yang ingin membeli emas sebagai alat perhiasan, namun ada juga pelanggan yang melakukan jual beli emas sebagai alat investasi. Dalam melakukan penjualan atau pembelian emas, harga beli dan harga jual emas selalu berubah-ubah setiap harinya sehingga bisa dikatakan bahwa harga emas bersifat fluktuatif. Dikarenakan harga emas yang selalu naik turun dan mengalami perubahan setiap harinya, hal ini mempengaruhi keputusan pelanggan dan pihak toko dalam menentukan kapan waktu yang tepat untuk menjual atau membeli emas. Pelanggan dan pihak toko terkadang berandai-andai sekiranya harga emas beberapa hari ke depan mengalami kenaikan atau penurunan. Prediksi atau perkiraan baik pelanggan ataupun pihak toko bisa saja salah atau tidak akurat. Jika terdapat kesalahan prediksi harga emas, maka bisa menyebabkan kerugian pada saat jual/beli emas dikarenakan perkiraan harga yang tidak sesuai dengan harga kenyataan di kemudian hari. Pelanggan toko biasanya juga bertanya mengenai harga emas yang dimana kebutuhan pelanggan akan informasi terhadap harga emas berbeda-beda, misalnya harga emas 1 gram, 5 gram, 10 gram, harga emas merek ANTAM, UBS, dan lain sebagainya. Dikarenakan permintaan informasi harga emas yang beranekaragam dari pelanggan, pihak toko harus selalu memantau dan mengetahui daftar harga emas yang terbaru setiap harinya agar dapat memenuhi permintaan pelanggan dan tidak memberikan informasi yang salah.

Selain harga emas, pelanggan toko juga terkadang ingin melihat stok emas perhiasan apa saja yang sedang tersedia di toko tanpa harus datang ke toko. Dengan kata lain, pelanggan berharap dapat melihat stok emas perhiasan yang sedang tersedia di toko dari mana saja dan kapan saja. Oleh karena itu, dalam memenuhi kebutuhan pelanggan, biasanya pihak toko mengambil foto untuk berbagai stok emas perhiasan yang ingin dilihat pelanggan. Hal ini tentunya merepotkan pihak toko dan memakan waktu untuk mengirim foto dan data stok emas satu per satu secara manual, misalnya melalui *WhatsApp* ke pelanggan dikarenakan Toko Emas XYZ belum memiliki *platform*, seperti *website* khusus untuk toko. Masalah lain yang dihadapi Toko Emas XYZ adalah terkadang pencatatan data toko, baik itu data penjualan, pembelian, pemesanan, ataupun data keuangan toko tidak tercatat atau tersimpan dengan baik dan

terkadang datanya kurang lengkap sehingga dapat menyebabkan kebingungan apabila ingin melihat kembali riwayat penjualan, pembelian, pemesanan, ataupun keuangan toko.

### **3.2 Analisis Kebutuhan**

Berdasarkan analisis permasalahan yang telah dijabarkan, maka dibutuhkan suatu sistem yang dapat membantu Toko Emas XYZ dalam mengatasi masalah, diantaranya:

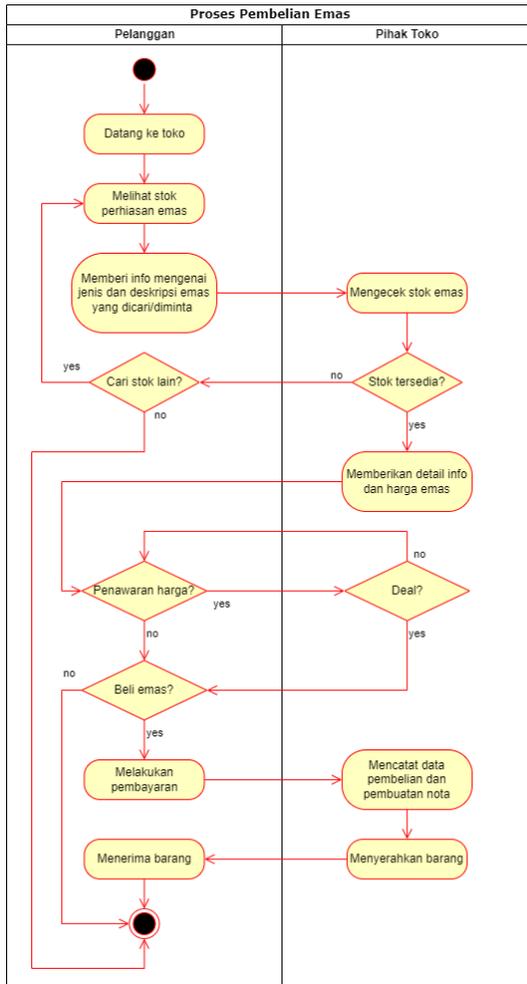
- Program yang bisa memprediksi harga emas atau menganalisis trend pergerakan harga emas di masa yang akan datang berdasarkan daftar riwayat harga emas di masa lampau.
- Program yang bisa menampilkan informasi mengenai daftar harga emas secara terkini (*up to date*).
- Program semacam katalog digital yang bisa menampilkan daftar stok emas yang sedang tersedia di toko beserta informasi terperinci dari emas tersebut, seperti jenis emas, kadar, berat, pabrik, dan deskripsi item.
- Program yang bisa melakukan pencatatan dan penyimpanan data toko, seperti data penjualan, pembelian, pemesanan, serta laporan keuangan toko secara terorganisir.

### **3.3 Analisis Proses Bisnis**

Proses bisnis Toko Emas XYZ terbagi menjadi beberapa bagian, yaitu:

#### **3.3.1 Proses Pembelian**

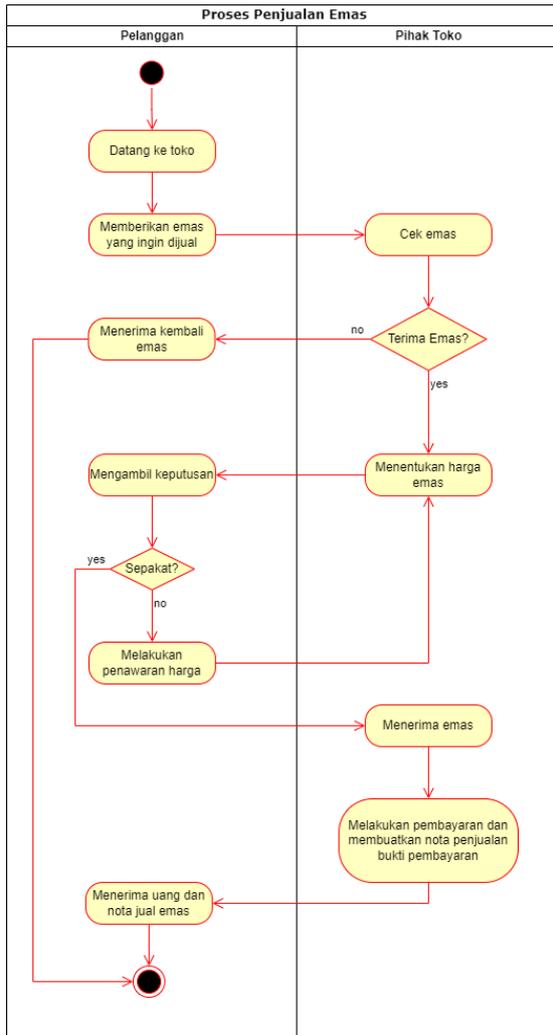
Proses pembelian emas yang dilakukan antara pelanggan toko dengan pihak toko emas xyz dimulai pada saat pelanggan datang ke toko, lalu melihat berbagai stok perhiasan emas yang tersedia. Pelanggan biasanya memberikan informasi mengenai jenis dan detail emas yang dicari, seperti berat emas, kadar kemurnian emas, jenis emas, dan lain sebagainya. Pihak toko akan mengecek apakah emas yang diinginkan pelanggan sedang tersedia. Jika emas yang dicari/diinginkan pelanggan tersedia, maka pihak toko akan mengeluarkan emas tersebut, lalu memberikan informasi dan harga dari emas tersebut. Pada tahapan ini, pelanggan bisa melakukan penawaran harga dengan pihak toko hingga mencapai kesepakatan antar dua belah pihak. Apabila pelanggan tertarik untuk membeli emas tersebut, maka pelanggan akan melakukan pembayaran. Setelah itu, pihak toko akan mencatat data pembelian serta membuat nota pembelian emas dan pelanggan bisa menerima barang yang dibeli. Proses pembelian emas ini dapat dilihat pada Gambar 3.1.



Gambar 3.1 Proses Pembelian

### 3.3.2 Proses Penjualan

Proses penjualan emas yang dilakukan antara pelanggan toko dengan pihak toko emas xyz dimulai pada saat pelanggan datang ke toko dan memberikan emas yang ingin dijual ke pihak toko. Pihak toko akan mengecek emas milik pelanggan apakah emas yang ingin dijual merupakan emas asli atau palsu dan meminta nota bukti pembelian emas. Jika tidak memenuhi syarat tersebut, maka emas akan dikembalikan ke pelanggan. Sedangkan apabila pihak toko menerima emas yang ingin dijual oleh pelanggan, pihak toko akan memberikan penawaran harga di hari tersebut. Pada tahap ini, pelanggan bisa melakukan perundingan mengenai harga emas yang ingin dijual dengan pihak toko. Jika sepakat, maka pihak toko akan menerima emas tersebut dan melakukan pembayaran ke pelanggan serta membuatkan nota penjualan bukti pembayaran. Proses penjualan emas ini dapat dilihat pada Gambar 3.2.

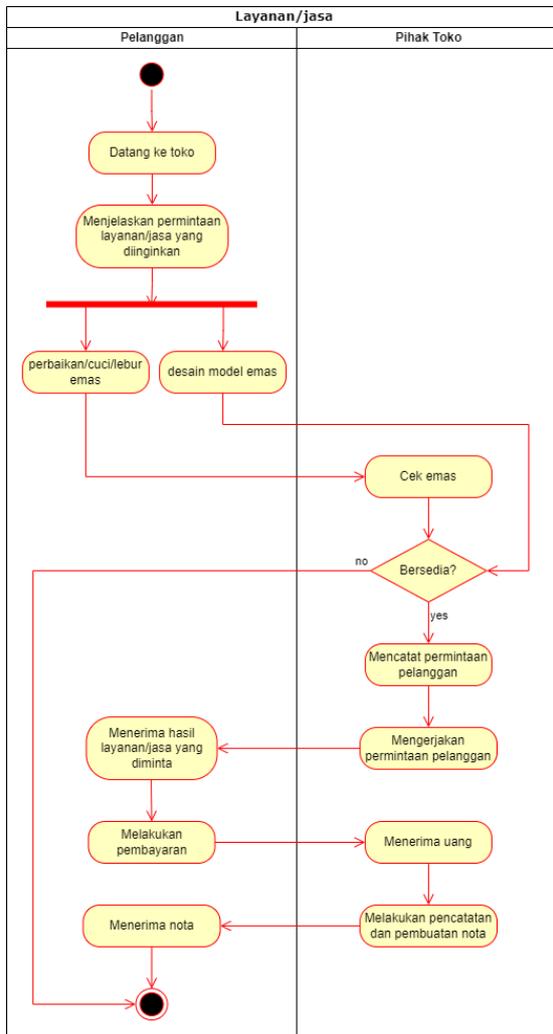


Gambar 3.2 Proses Penjualan

### 3.3.3 Proses Pemesanan Layanan/Jasa

Proses pemesanan layanan/jasa emas yang dilakukan antara pelanggan toko dengan pihak toko emas xyz dimulai pada saat pelanggan datang ke toko, lalu menjelaskan permintaan layanan/jasa yang diinginkan ke pihak toko, misalnya jasa perbaikan emas yang rusak, cuci emas, lebur emas, atau layanan desain model emas perhiasan. Pihak toko akan mengecek/melihat emas dan mempertimbangkan permintaan pelanggan, apakah bersedia dan sanggup atau tidak. Jika bersedia, maka pihak toko akan mencatat data permintaan pesanan pelanggan, lalu mengerjakan permintaan sesuai keinginan pelanggan. Apabila pengerjaan sudah selesai, maka pelanggan akan menerima hasil dari layanan/jasa yang diminta. Setelah itu, pelanggan akan melakukan pembayaran terhadap jasa/layanan tersebut, lalu pihak toko akan menerima uang dan melakukan pencatatan serta pembuatan nota. Proses ini diakhiri ketika pelanggan

menerima nota pemesanan layanan/jasa emas. Proses pemesanan layanan/jasa ini dapat dilihat pada Gambar 3.3.



Gambar 3.3 Proses Pemesanan Layanan/Jasa

### 3.4 Desain Sistem

Pada bagian ini akan dijabarkan atau dijelaskan mengenai rancangan mengenai desain sistem terhadap aplikasi yang akan dibuat. Bagian ini akan menjelaskan *use case diagram* dan *activity diagram* dalam membangun aplikasi.

#### 3.4.1 Use Case Diagram



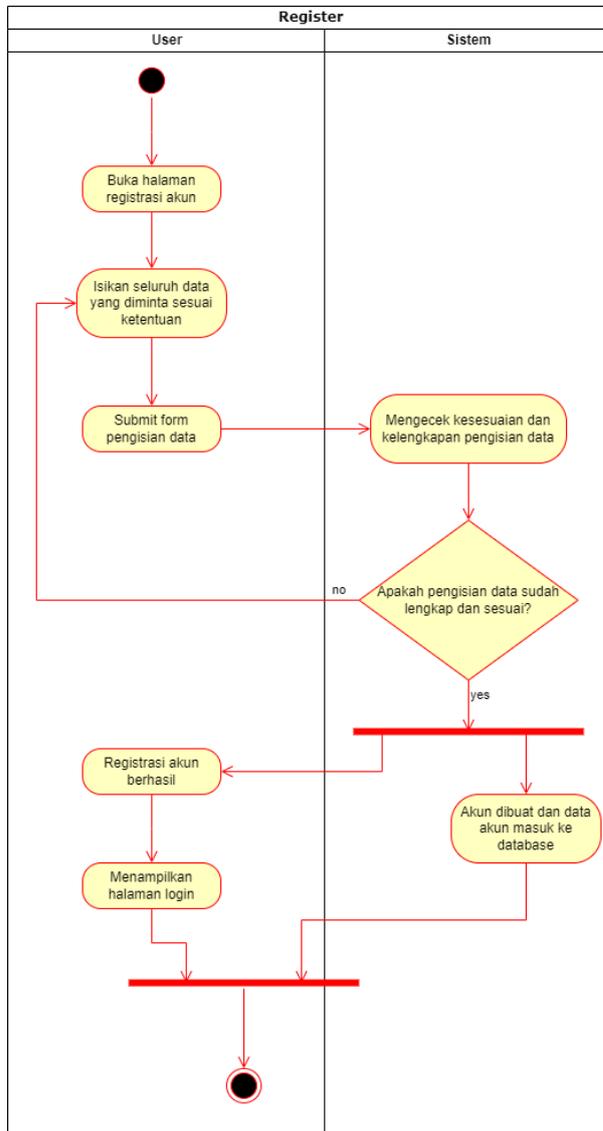
Gambar 3.4 Use Case Diagram

Gambar 3.4 di atas merupakan *use case* diagram dari sistem yang akan dibuat untuk Toko Emas XYZ. Terdapat dua aktor yang bisa melakukan interaksi dengan sistem, yaitu pihak toko XYZ yang berperan sebagai *admin* dan pelanggan toko yang berperan sebagai *user*. *Admin* bisa menggunakan beberapa fitur, seperti *login*, manajemen akun, manajemen inventaris untuk mengelola stok perhiasan emas, verifikasi pesanan, memberikan *appointment status*, *input* data jual dan beli emas, *input* data pesanan layanan/jasa emas, laporan keuangan, dan menampilkan daftar harga emas hari ini, prediksi harga emas, dan memberikan kisaran harga emas kepada pelanggan apabila pelanggan ingin menjual emas. Sedangkan, *user* bisa menggunakan fitur-fitur seperti *register*, *login*, manajemen akun, katalog digital untuk melihat berbagai stok perhiasan emas yang tersedia, *filter* stok perhiasan emas, melakukan pemesanan layanan/jasa emas, misalnya jasa perbaikan emas yang rusak, cuci emas, lebur emas, atau layanan desain model emas perhiasan. Selain itu, *user* juga bisa menggunakan fitur *set appointment* apabila pelanggan ingin mengatur waktu dengan pihak toko untuk jual beli emas atau keperluan lainnya. *User* juga

bisa menggunakan fitur untuk cek harga emas saat ini dan mendapatkan informasi mengenai kisaran harga emas dari emas yang ingin dijual oleh *user*.

### 3.4.2 Activity Diagram

#### 3.4.2.1 Activity Diagram Register

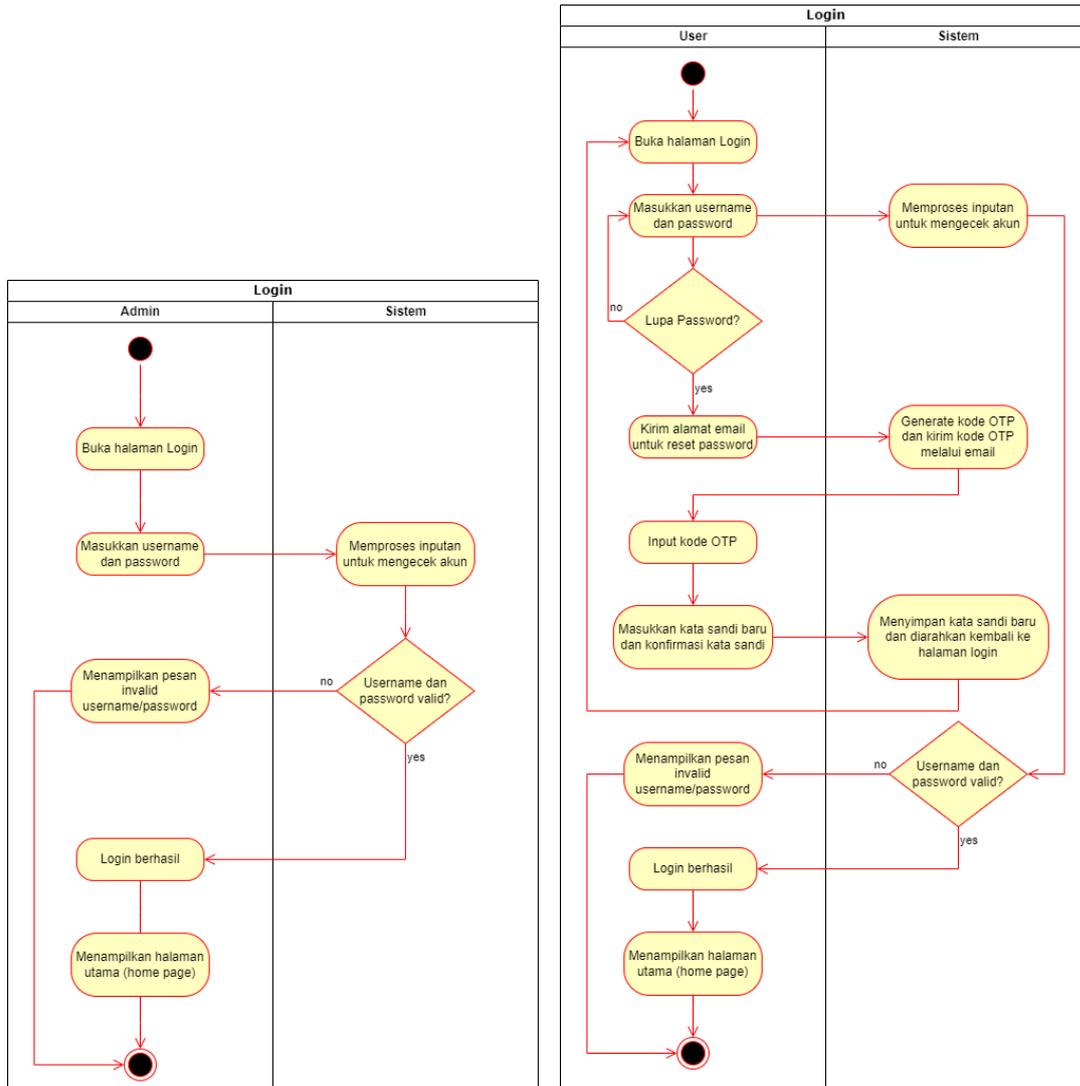


Gambar 3.5 Activity Diagram Register

Gambar 3.5 di atas merupakan *activity diagram* dari proses *register* untuk aktor *user* (pelanggan toko) atau pengguna aplikasi *mobile* Toko Emas XYZ. Proses *register* (registrasi akun) diawali ketika *user* membuka halaman registrasi akun. Lalu, *user* diminta untuk mengisi seluruh data yang diperlukan untuk membuat sebuah akun. Setelah *user* melakukan *submit form* pengisian data, maka sistem akan mengecek apakah data sudah lengkap dan seluruh data yang dimasukkan sudah sesuai dengan ketentuan atau tidak. Jika data yang dimasukkan sudah

lengkap dan sesuai, maka registrasi akun berhasil dan data akun yang dibuat akan masuk ke dalam *database*. Kemudian, *user* akan diarahkan ke halaman *login*. Namun, jika data yang dimasukkan tidak lengkap atau tidak sesuai dengan ketentuan, maka *user* diminta untuk mengisi ulang data.

### 3.4.2.2 Activity Diagram Login



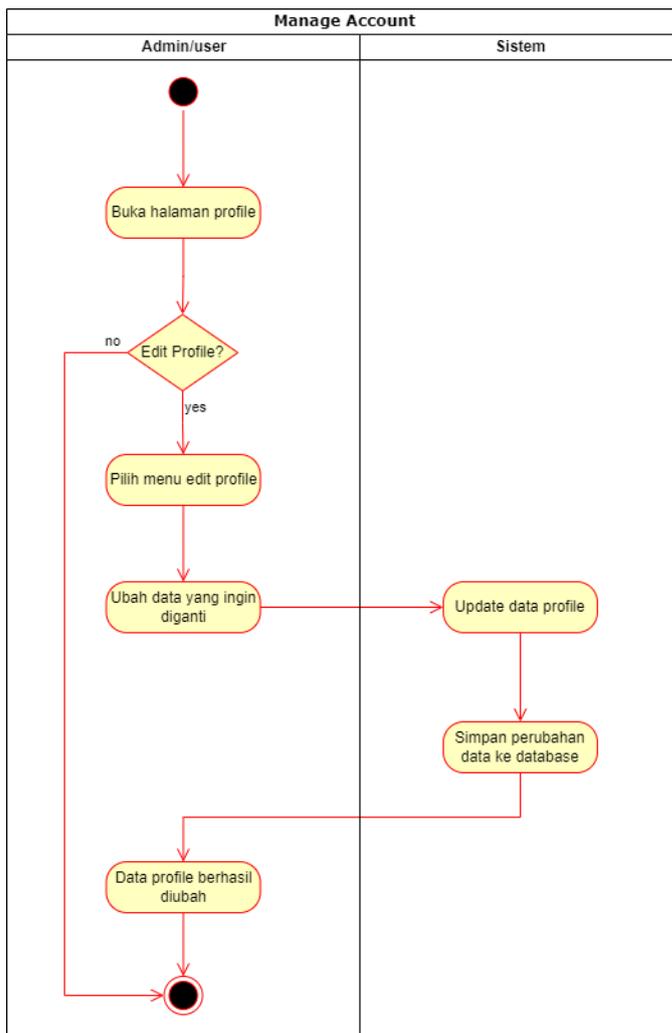
Gambar 3.6 Activity Diagram Login

Gambar 3.6 sebelah kiri merupakan *activity diagram* dari proses *login* untuk aktor *admin* (pihak toko) pada aplikasi versi *website*. Proses *login* diawali ketika *admin* membuka halaman *login*. Lalu, *admin* akan diminta untuk memasukkan *email* dan *password*. Kemudian, sistem akan mengecek *email* dan *password* yang telah dimasukkan. Jika *email* dan *password* yang dimasukkan *valid*, maka *login* berhasil dan *admin* akan diarahkan ke halaman utama (*home*

page). Namun, jika *email* dan *password* yang dimasukkan tidak *valid*, maka sistem akan menampilkan pesan “*invalid email/password*”.

Gambar 3.6 sebelah kanan merupakan *activity* diagram dari proses *login* untuk aktor *user* (pelanggan) pada aplikasi versi *mobile*. Proses *login* diawali ketika *user* membuka halaman *login*. Lalu, *user* akan diminta untuk memasukkan *email* dan *password*. Jika pengguna lupa kata sandi pada akun, maka pengguna bisa menggunakan fitur *forgot password* yang dimana pengguna dapat mengatur ulang kata sandi. Setelah berhasil mengatur ulang kata sandi, pengguna akan diarahkan kembali ke halaman *login* dan pengguna bisa memasukkan *email* dan *password*. Kemudian, sistem akan mengecek *email* dan *password* yang telah dimasukkan. Jika *email* dan *password* yang dimasukkan *valid*, maka *login* berhasil dan *admin* akan diarahkan ke halaman utama (*home page*). Namun, jika *email* dan *password* yang dimasukkan tidak *valid*, maka sistem akan menampilkan pesan “*invalid email/password*”.

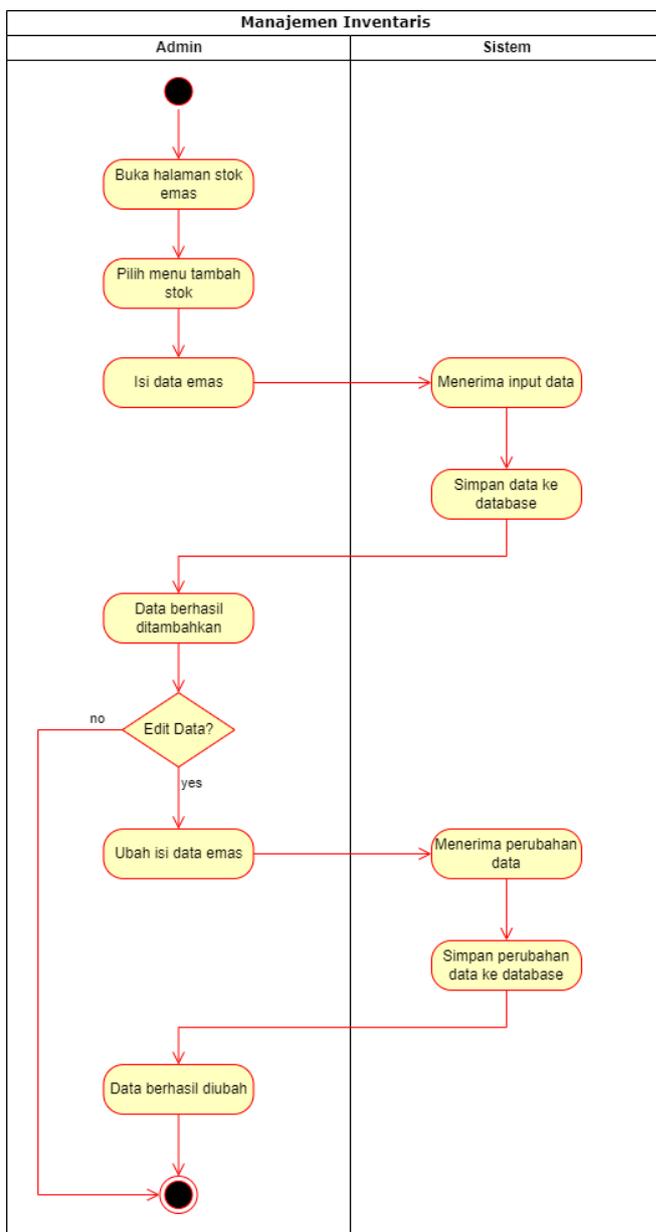
### 3.4.2.3 Activity Diagram Manage Account



Gambar 3.7 Activity Diagram Manage Account

Gambar 3.7 di atas merupakan *activity* diagram dari proses manajemen akun untuk aktor *user* (pelanggan toko) dan *admin* (pihak toko). Proses manajemen akun diawali ketika *admin/user* membuka halaman *profile*. Lalu, jika *admin/user* ingin mengubah data pada akun, maka *admin/user* bisa membuka menu *edit profile* dan kemudian ubah data yang ingin diganti. Setelah melakukan perubahan data akun, maka sistem akan melakukan *update* data *profile* dan menyimpan perubahan data ke dalam *database*.

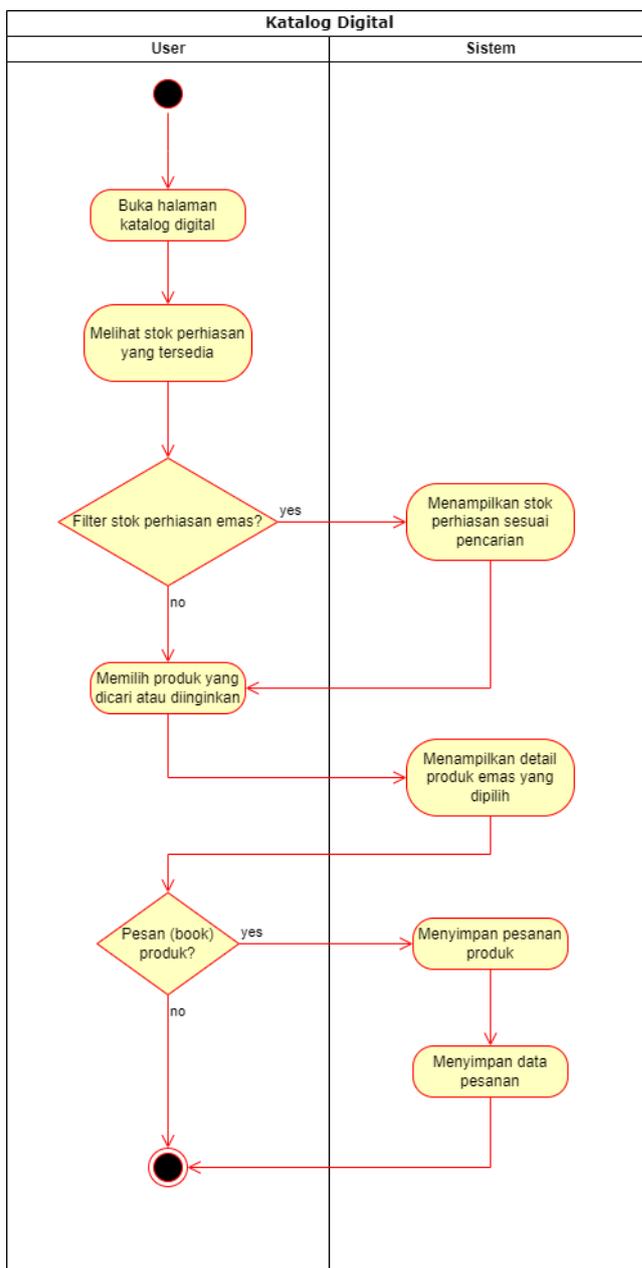
### 3.4.2.4 Activity Diagram Manajemen Inventaris



Gambar 3.8 Activity Diagram Manajemen Inventaris

Gambar 3.8 di atas merupakan *activity* diagram dari proses manajemen inventaris untuk aktor *admin* (pihak toko). Proses manajemen inventaris diawali ketika *admin* membuka halaman stok emas. Lalu, *admin* bisa memilih menu tambah stok untuk memasukkan stok perhiasan emas dengan mengisi data emas yang ingin ditambahkan. Sistem akan menerima *input* data dan menyimpan data ke dalam *database* ketika data berhasil ditambahkan. Jika *admin* ingin *edit* data, maka *admin* bisa mengubah isi data emas yang ingin diganti datanya, lalu sistem akan menerima dan menyimpan perubahan data ke dalam *database* ketika data berhasil diubah.

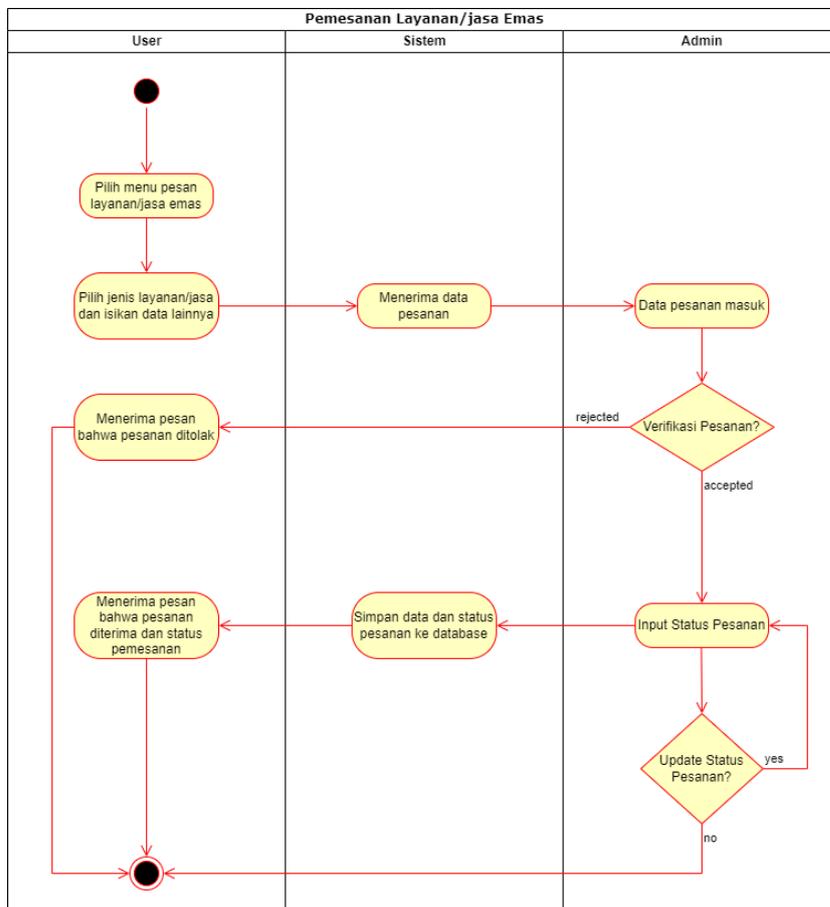
### 3.4.2.5 Activity Diagram Katalog Digital



Gambar 3.9 Activity Diagram Katalog Digital

Gambar 3.9 di atas merupakan *activity* diagram dari proses katalog digital untuk aktor *user* (pelanggan toko). Proses katalog digital diawali ketika *user* membuka halaman katalog digital. Setelah membuka halaman tersebut, *user* bisa melihat berbagai stok emas yang tersedia. Jika *user* ingin melakukan *filter* stok emas, maka sistem hanya akan menampilkan daftar stok emas sesuai dengan pencarian *user*. Jika *user* mengklik atau memilih salah satu stok emas, maka sistem akan mengarahkan *user* ke halaman detail item yang berisi informasi terperinci mengenai stok emas yang dipilih.

### 3.4.2.6 Activity Diagram Pemesanan Layanan/jasa Emas

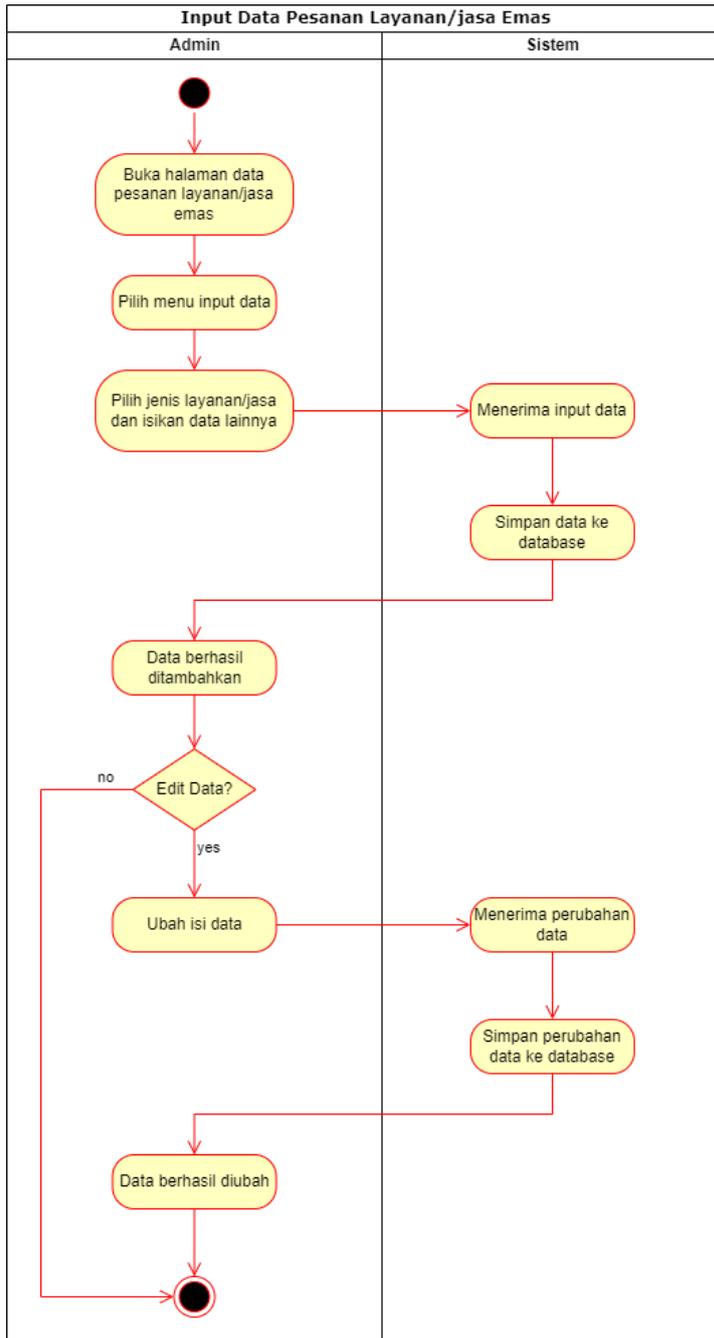


Gambar 3.10 Activity Diagram Pemesanan Layanan/jasa Emas

Gambar 3.10 di atas merupakan *activity* diagram dari proses pemesanan layanan/jasa emas antara aktor *user* (pelanggan toko) dengan *admin* (pihak toko). Proses pemesanan layanan/jasa emas diawali ketika *user* memilih menu pesan layanan/jasa emas, lalu menentukan jenis layanan/jasa yang diinginkan serta mengisi data pemesanan. Sistem akan menerima data pesanan dari *user* dan *default status* pada data pesanan yang dibuat adalah “pending”. Lalu,

data status pesanan tersebut masuk ke aplikasi versi *website* dan status bisa di-*update* oleh *admin*. Apabila terdapat perubahan status pesanan, maka status pesanan akan tersimpan di *database*. *User* akan menerima pesan bahwa pesanan diterima dan bisa melihat status pemesanan saat ini. Jika *admin* ingin melakukan *update* pada status pemesanan, maka *admin* bisa *input* ulang status pesanan, misalnya pengerjaan pemesanan layanan/jasa telah selesai, maka status pesanan yang awalnya “dalam pengerjaan” bisa diubah menjadi “selesai”. Namun, jika *admin* menolak pesanan dari *user*, maka verifikasi pesanan ditolak dengan keterangan “tolak”. *User* dapat melihat perubahan status pada data pesanan.

#### **3.4.2.7 Activity Diagram Input Data Pesanan Layanan/jasa Emas**

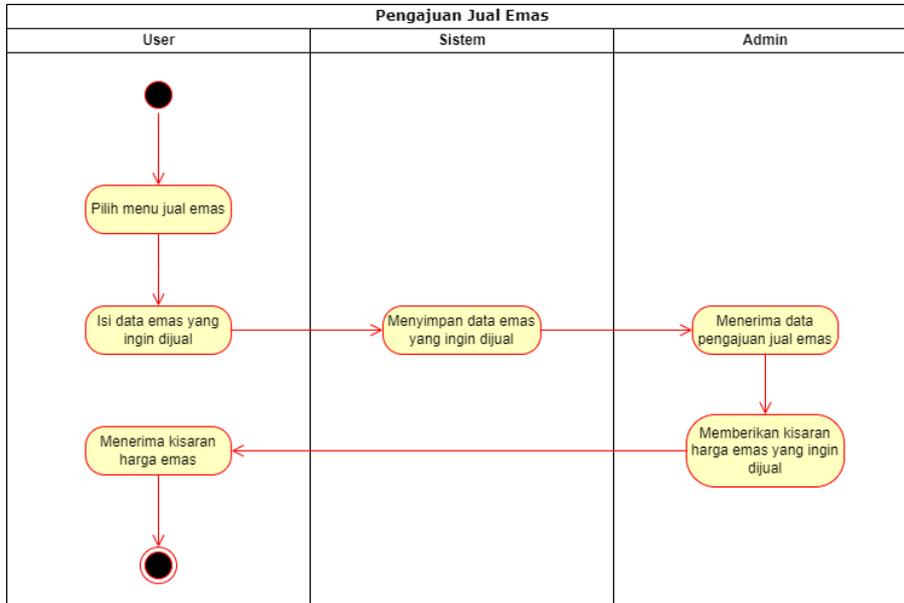


Gambar 3.11 Activity Diagram *Input Data Pesanan Layanan/jasa Emas*

Gambar 3.11 di atas merupakan *activity diagram* dari proses *input data* pesanan layanan/jasa emas untuk aktor *admin* (pihak toko). Proses *input data* pesanan layanan/jasa emas diawali ketika *admin* membuka halaman data pesanan layanan/jasa emas, lalu memilih menu *input data* dan memilih jenis layanan/jasa emas serta mengisi data pemesanan. Sistem akan menerima *input data* dan menyimpan data tersebut ke *database* ketika data berhasil ditambahkan. Jika *admin* ingin melakukan *edit data*, maka ubah isi data yang ingin diganti.

Kemudian, sistem akan menerima perubahan data dan menyimpan perubahan data ke dalam *database* ketika data berhasil diubah.

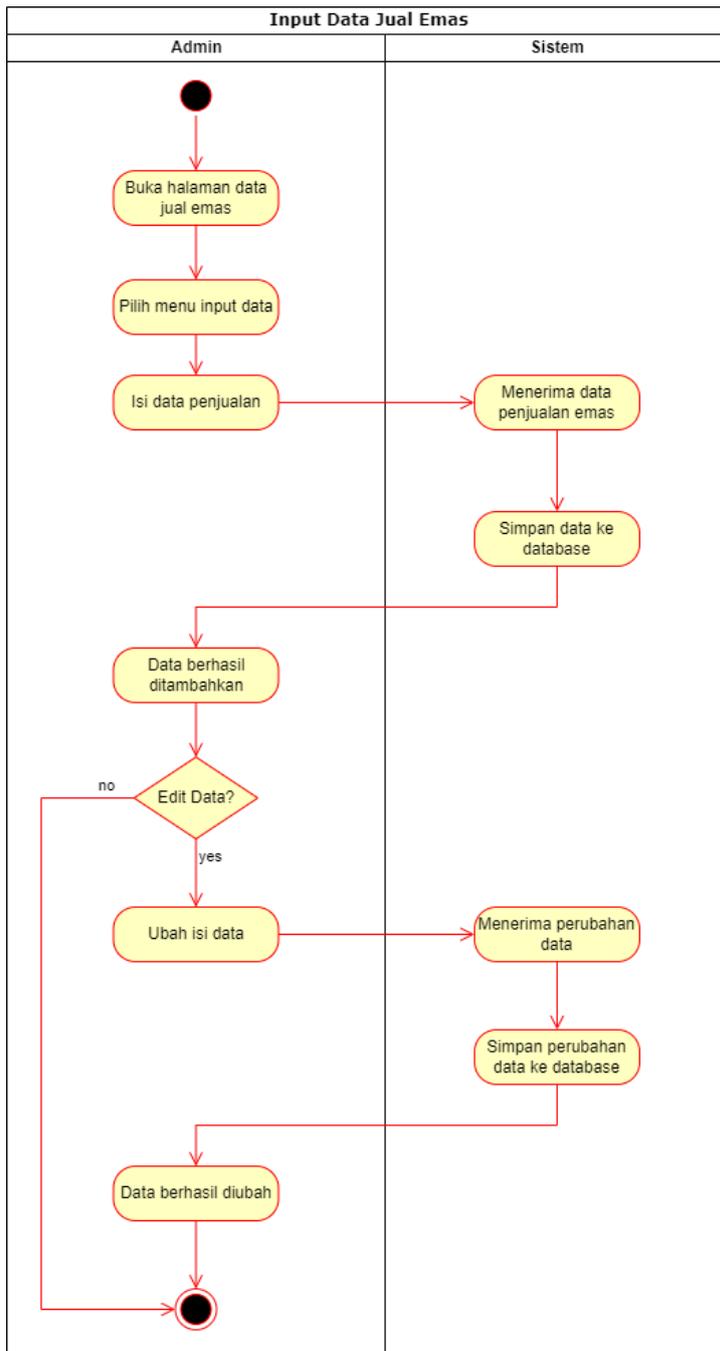
### 3.4.2.8 Activity Diagram Pengajuan Jual Emas



Gambar 3.12 Activity Diagram Pengajuan Jual Emas

Gambar 3.12 di atas merupakan *activity* diagram dari proses pengajuan jual emas antara aktor *user* (pelanggan toko) dengan *admin* (pihak toko). Proses pengajuan jual emas diawali ketika *user* memilih menu jual emas dan mengisi data emas yang ingin dijual. Sistem akan menyimpan data emas yang ingin dijual, lalu *admin* akan menerima data pengajuan jual emas dari *user*. Setelah itu, *admin* bisa memberikan kisaran harga emas yang ingin dijual *user* dan *user* akan menerima informasi berupa kisaran harga emas tersebut.

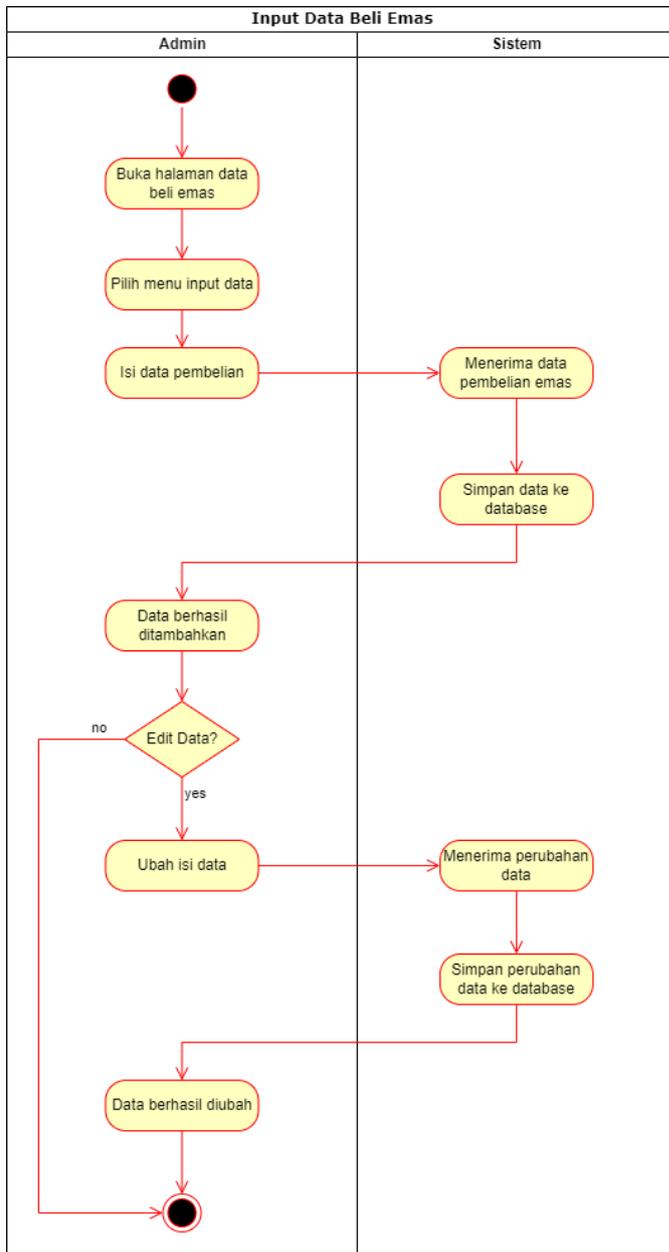
### 3.4.2.9 Activity Diagram Input Data Jual Emas



Gambar 3.13 Activity Diagram *Input* Data Jual Emas

Gambar 3.13 di atas merupakan *activity* diagram dari proses *input* data jual emas untuk aktor *admin* (pihak toko). Proses *input* data jual emas diawali ketika *admin* membuka halaman data jual emas, lalu memilih menu *input* data dan mengisi data penjualan. Sistem akan menerima data penjualan emas dan menyimpan data tersebut ke *database* ketika data berhasil ditambahkan. Jika *admin* ingin melakukan *edit* data, maka ubah isi data yang ingin diganti. Kemudian, sistem akan menerima perubahan data dan menyimpan perubahan data ke dalam *database* ketika data berhasil diubah.

### 3.4.2.10 Activity Diagram *Input Data Beli Emas*

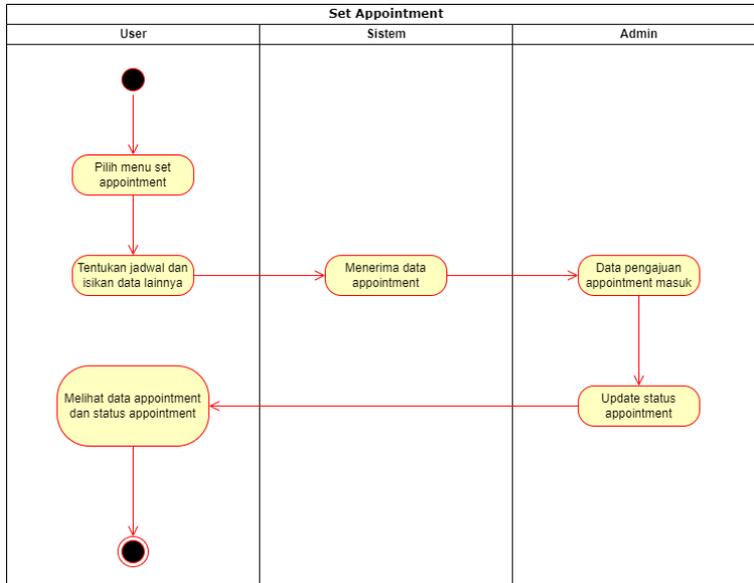


Gambar 3.14 Activity Diagram *Input Data Beli Emas*

Gambar 3.14 di atas merupakan *activity* diagram dari proses *input* data beli emas untuk aktor *admin* (pihak toko). Proses *input* data beli emas diawali ketika *admin* membuka halaman data beli emas, lalu memilih menu *input* data dan mengisi data pembelian. Sistem akan menerima data pembelian emas dan menyimpan data tersebut ke *database* ketika data berhasil ditambahkan. Jika *admin* ingin melakukan *edit* data, maka ubah isi data yang ingin diganti.

Kemudian, sistem akan menerima perubahan data dan menyimpan perubahan data ke dalam *database* ketika data berhasil diubah.

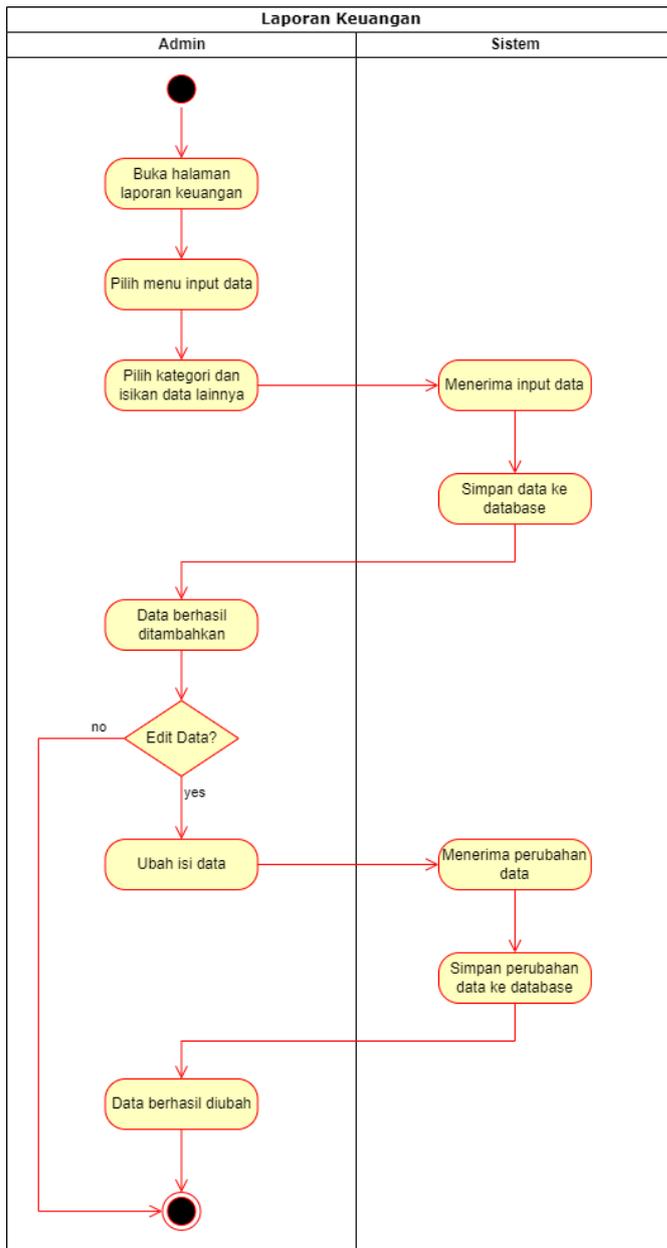
### 3.4.2.11 Activity Diagram Set Appointment



Gambar 3.15 Activity Diagram Set Appointment

Gambar 3.15 di atas merupakan *activity* diagram dari proses *set appointment* antara aktor *user* (pelanggan toko) dengan *admin* (pihak toko). Proses *set appointment* diawali ketika *user* memilih menu *set appointment*, lalu menentukan jadwal *appointment* dan mengisi data yang diperlukan. Sistem akan menerima data *appointment* dan data pengajuan *appointment* akan masuk ke akun *admin*. Jika *admin* menyetujui *appointment*, maka *admin* bisa *update* status *appointment*, misalnya status menjadi "terima". Kemudian, sistem akan menyimpan data *appointment* ke *database* dan *user* dapat melihat perubahan status *appointment*.

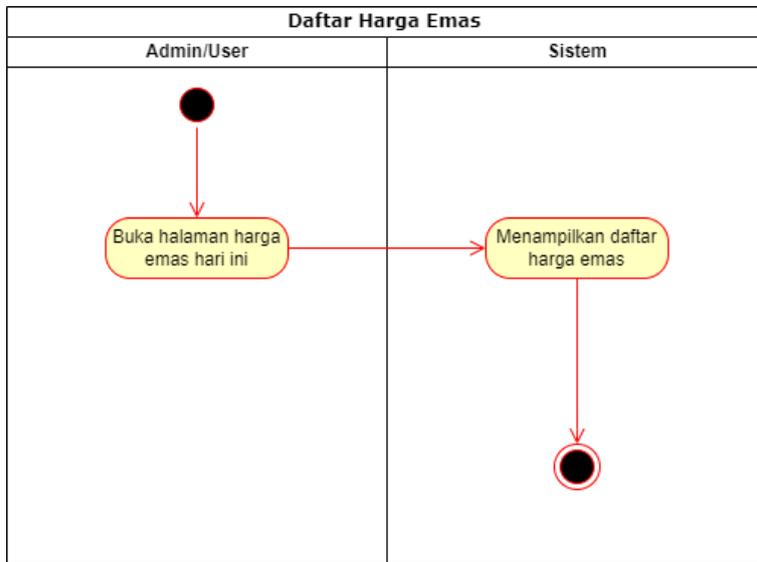
### 3.4.2.12 Activity Diagram Laporan Keuangan



Gambar 3.16 Activity Diagram Laporan Keuangan

Gambar 3.16 di atas merupakan *activity* diagram dari proses laporan keuangan untuk aktor *admin* (pihak toko). Proses laporan keuangan diawali ketika *admin* membuka halaman laporan keuangan, lalu memilih menu *input* data dan menentukan kategori, apakah termasuk data pengeluaran atau pemasukan serta mengisi data keuangan. Sistem akan menerima *input* data dan menyimpan data tersebut ke *database* ketika data berhasil ditambahkan. Jika *admin* ingin melakukan *edit* data, maka ubah isi data yang ingin diganti. Kemudian, sistem akan menerima perubahan data dan menyimpan perubahan data ke dalam *database* ketika data berhasil diubah.

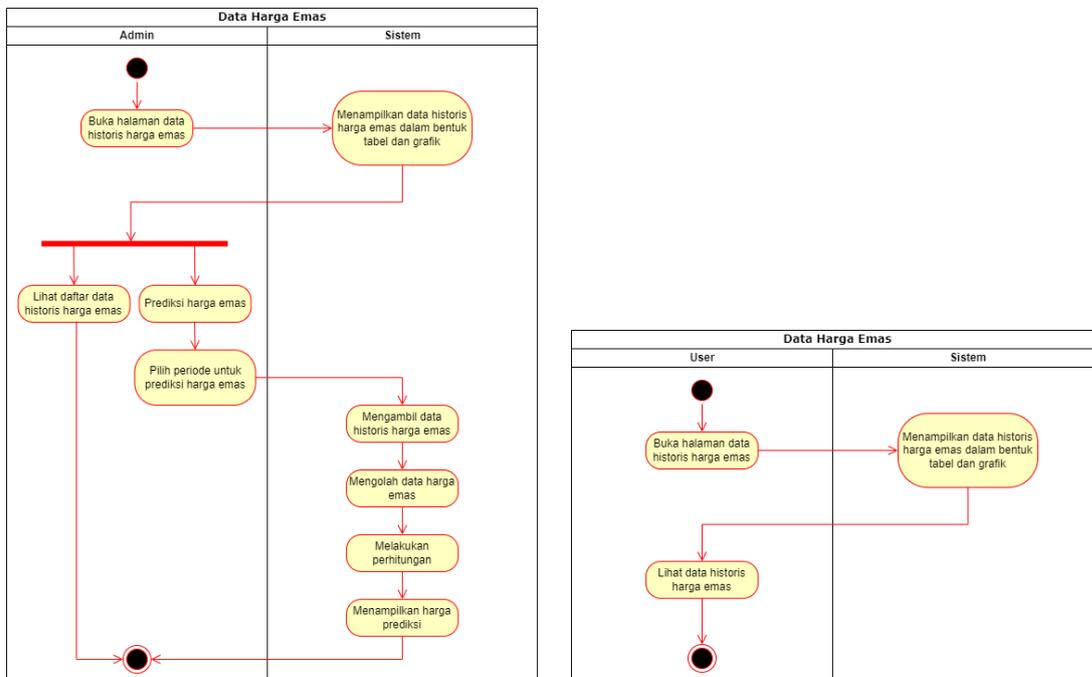
### 3.4.2.13 Activity Diagram Cek Harga Emas



Gambar 3.17 Activity Diagram Cek Harga Emas

Gambar 3.17 di atas merupakan *activity* diagram dari cek harga emas. Baik aktor *admin* maupun *user*, daftar harga emas hari ini bisa diakses secara langsung ketika pengguna membuka halaman daftar harga emas hari ini pada aplikasi versi *website* maupun aplikasi versi *mobile*.

### 3.4.2.14 Activity Diagram Data Harga Emas



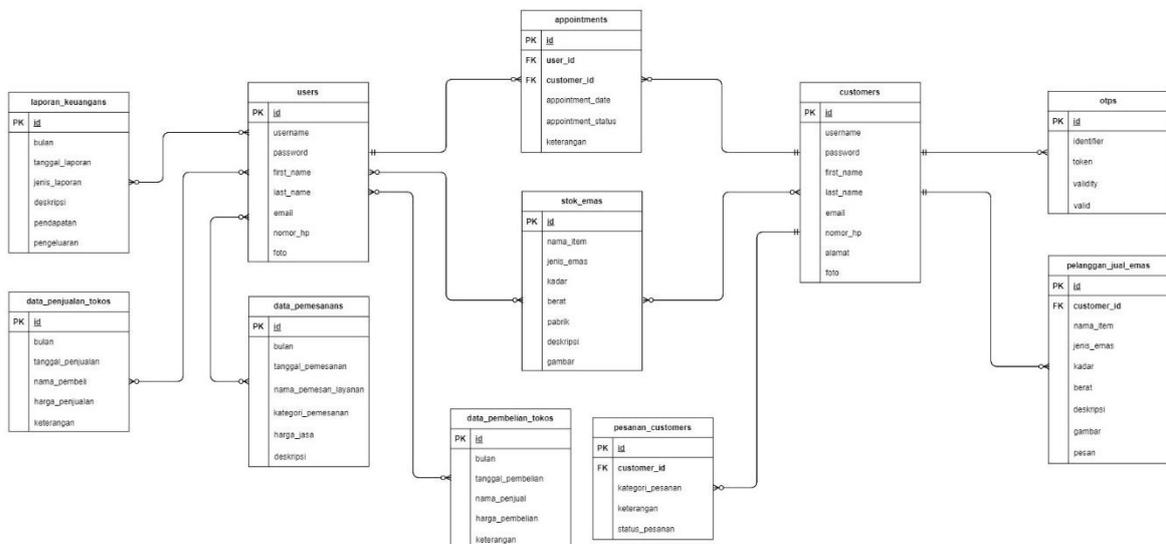
Gambar 3.18 Activity Diagram Data Harga Emas

Gambar 3.18 di atas merupakan *activity* diagram dari proses data historis harga emas. Untuk aktor *user*, *user* bisa melihat data historis harga emas dalam bentuk tabel dan grafik pada halaman data historis harga emas pada aplikasi versi *mobile*. Sedangkan untuk aktor *admin*, pada aplikasi versi *website* tidak hanya terdapat kumpulan data historis harga emas yang ditampilkan dalam bentuk tabel dan grafik, tapi juga terdapat fitur untuk memprediksi harga emas berdasarkan data historis harga emas tersebut. Jika ingin melakukan prediksi harga emas, *admin* harus input periode waktu terlebih dahulu, lalu sistem akan mengolah data dan melakukan perhitungan. Setelah itu, sistem akan menampilkan daftar harga prediksi dalam bentuk grafik pada aplikasi versi *website*.

### 3.5 Desain Database

Pada bagian ini terdapat *Entity Relationship* Diagram (ERD) dan tabel-tabel pada *database*.

#### 3.5.1 Entity Relationship Diagram



Gambar 3.19 *Entity Relationship* Diagram dengan notasi *Crow's Foot*

Gambar 3.19 di atas merupakan *entity relationship* diagram dengan notasi *crow's foot* yang berisi tabel-tabel yang berperan dalam penyimpanan data untuk sistem toko emas XYZ. Terdapat beberapa tabel, seperti tabel *customers*, *users*, *appointments*, *laporan\_keuangans*, *stok\_emas*, *data\_penjualan\_tokos*, *pelanggan\_jual\_emas*, *data\_pembelian\_tokos*, *pesanan\_customers*, *data\_pemesanans*, dan tabel *otps*.

#### 3.5.2 Tabel Database

### 3.5.2.1 Tabel Customer

Tabel *customers* menyimpan data pengguna aplikasi *mobile* Toko Emas XYZ dengan atribut *id*, *username*, *password*, *first\_name*, *last\_name*, *email*, *nomor\_hp*, *alamat*, dan *foto*. Detail informasi mengenai tabel *customers* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Tabel Customer

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id pengguna dan merupakan <i>primary key</i> dari tabel <i>customers</i>
username	varchar(255)	Username dari pengguna
password	varchar(255)	Kata sandi dari akun pengguna
first_name	varchar(255)	Atribut yang menyimpan nama depan pengguna
last_name	varchar(255)	Atribut yang menyimpan nama belakang pengguna
email	varchar(255)	Email pengguna
nomor_hp	varchar(255)	Nomor telepon pengguna
alamat	varchar(255)	Alamat tempat tinggal pengguna
foto	varchar(255)	Foto profil pengguna

### 3.5.2.2 Tabel User

Tabel *users* menyimpan data pengguna *website* Toko Emas XYZ, yaitu admin (pihak toko) dengan atribut *id*, *username*, *password*, *first\_name*, *last\_name*, *email*, *nomor\_hp*, dan *foto*. Detail informasi mengenai tabel *users* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Tabel User

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id pengguna dan merupakan <i>primary key</i> dari tabel <i>users</i>
username	varchar(255)	Username dari pengguna
password	varchar(255)	Kata sandi dari akun pengguna
first_name	varchar(255)	Atribut yang menyimpan nama depan pengguna
last_name	varchar(255)	Atribut yang menyimpan nama belakang pengguna
email	varchar(255)	Email pengguna
nomor_hp	varchar(255)	Nomor telepon pengguna
foto	varchar(255)	Foto profil pengguna

### 3.5.2.3 Tabel Appointment

Tabel *appointments* menyimpan data janji temu dengan atribut *id*, *user\_id*, *customer\_id*, *appointment\_date*, *appointment\_status*, dan keterangan. Detail informasi mengenai tabel *appointments* dapat dilihat pada Tabel 3.3.

Tabel 3.3 Tabel Appointment

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id appointment dan merupakan <i>primary key</i> dari tabel <i>appointments</i>
user_id	bigint(20) (FK)	Id pengguna <i>website</i> (pihak toko) yang memiliki relasi dengan tabel <i>users</i> dan merupakan <i>foreign key</i> dari tabel <i>appointments</i>
customer_id	bigint(20) (FK)	Id pengguna aplikasi <i>mobile</i> (pelanggan) yang memiliki relasi dengan tabel <i>customers</i> dan merupakan <i>foreign key</i> dari tabel <i>appointments</i>
appointment_date	datetime	Tanggal dan waktu janji temu yang diajukan
appointment_status	varchar(255)	Status dari pengajuan janji temu
keterangan	text	Keterangan atau deskripsi dari janji temu

### 3.5.2.4 Tabel Laporan Keuangan

Tabel *laporan\_keuangans* menyimpan data laporan keuangan toko dengan atribut *id*, *bulan*, *tanggal\_laporan*, *jenis\_laporan*, *deskripsi*, *pendapatan*, dan *pengeluaran*. Detail informasi mengenai tabel *laporan\_keuangans* dapat dilihat pada Tabel 3.4.

Tabel 3.4 Tabel Laporan Keuangan

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id laporan keuangan dan merupakan <i>primary key</i> dari tabel <i>laporan_keuangans</i>
bulan	varchar(255)	Atribut yang menyimpan bulan pembuatan laporan
tanggal_laporan	date	Atribut yang menyimpan tanggal pembuatan laporan

jenis_laporan	int(11)	Atribut yang menyimpan jenis laporan yang dibuat, misalnya pembelian, penjualan, dan lain sebagainya
deskripsi	text	Keterangan atau penjelasan dari laporan yang dibuat
pendapatan	int(11)	Pendapatan yang didapatkan toko
pengeluaran	int(11)	Pengeluaran yang dikeluarkan toko

### 3.5.2.5 Tabel Stok Emas

Tabel *stok\_emas* menyimpan data stok emas dengan atribut *id*, *nama\_item*, *jenis\_emas*, *kadar*, *berat*, *pabrik*, *deskripsi*, dan *gambar*. Detail informasi mengenai tabel *stok\_emas* dapat dilihat pada Tabel 3.5.

Tabel 3.5 Tabel Stok Emas

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id barang dan merupakan <i>primary key</i> dari tabel <i>stok_emas</i>
nama_item	varchar(255)	Nama barang
jenis_emas	int(11)	Jenis atau kategori emas, misalnya kalung, cincin, gelang, anting, liontin, dan sebagainya
kadar	varchar(255)	Kandungan kemurnian emas dengan satuan karat
berat	varchar(255)	Berat dari emas dengan satuan gram
pabrik	varchar(255)	Perusahaan atau pabrik pembuat emas, misalnya UBS, Antam, dan sebagainya
deskripsi	text	Penjelasan atau ringkasan singkat dari barang
gambar	varchar(255)	Gambar/foto dari barang

### 3.5.2.6 Tabel Data Penjualan Toko

Tabel *data\_penjualan\_tokos* menyimpan data penjualan toko dengan atribut *id*, *bulan*, *tanggal\_penjualan*, *nama\_pembeli*, *harga\_penjualan*, dan *keterangan*. Detail informasi mengenai tabel *data\_penjualan\_tokos* dapat dilihat pada Tabel 3.6.

Tabel 3.6 Tabel Data Penjualan Toko

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id data penjualan toko dan merupakan <i>primary key</i> dari tabel <i>data_penjualan_tokos</i>
bulan	varchar(255)	Atribut yang menyimpan data bulan pada penjualan toko
tanggal_penjualan	date	Atribut yang menyimpan data tanggal penjualan toko
nama_pembeli	varchar(255)	Nama pembeli
harga_penjualan	int(11)	Harga penjualan barang emas
keterangan	text	Keterangan atau deskripsi dari data penjualan toko

### 3.5.2.7 Tabel Pelanggan Jual Emas

Tabel *pelanggan\_jual\_emas* menyimpan data pengajuan jual emas yang dibuat oleh pengguna di aplikasi *mobile* Toko Emas XYZ dengan atribut *id*, *customer\_id*, *nama\_item*, *jenis\_emas*, *kadar*, *berat*, *deskripsi*, *gambar*, dan *pesan*. Detail informasi mengenai tabel *pelanggan\_jual\_emas* dapat dilihat pada Tabel 3.7.

Tabel 3.7 Tabel Pelanggan Jual Emas

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id data pengajuan jual emas oleh pengguna dari aplikasi <i>mobile</i> dan merupakan <i>primary key</i> dari tabel <i>pelanggan_jual_emas</i>
customer_id	bigint(20) (FK)	Id pengguna dari aplikasi <i>mobile</i> yang memiliki relasi dengan tabel <i>customers</i> dan merupakan <i>foreign key</i> dari tabel <i>pelanggan_jual_emas</i>
nama_item	varchar(255)	Nama barang
jenis_emas	varchar(255)	Jenis atau kategori emas, misalnya kalung, cincin, gelang, anting, liontin, dan sebagainya
kadar	varchar(255)	Kandungan kemurnian emas dengan satuan karat
berat	varchar(255)	Berat dari emas dengan satuan gram
deskripsi	text	Penjelasan atau ringkasan singkat dari barang
gambar	varchar(255)	Gambar/foto dari barang

pesan	text	Atribut yang menyimpan pesan balasan dari pihak toko ke pengguna
-------	------	------------------------------------------------------------------

### 3.5.2.8 Tabel Data Pembelian Toko

Tabel *data\_pembelian\_tokos* menyimpan data pembelian toko dengan atribut *id*, *bulan*, *tanggal\_pembelian*, *nama\_penjual*, *harga\_pembelian*, dan *keterangan*. Detail informasi mengenai tabel *data\_pembelian\_tokos* dapat dilihat pada Tabel 3.8.

Tabel 3.8 Tabel Data Pembelian Toko

Nama Atribut	Type Data	Keterangan
id	bigint(20) (PK)	Id data pembelian toko dan merupakan <i>primary key</i> dari tabel <i>data_pembelian_tokos</i>
bulan	varchar(255)	Atribut yang menyimpan data bulan pada pembelian toko
tanggal_pembelian	date	Atribut yang menyimpan data tanggal pembelian toko
nama_penjual	varchar(255)	Nama penjual
harga_pembelian	int(11)	Harga pembelian barang emas
keterangan	text	Keterangan atau deskripsi dari data pembelian toko

### 3.5.2.9 Tabel Pesanan Customer

Tabel *pesanan\_customers* menyimpan data pesanan layanan/jasa pelanggan dengan atribut *id*, *customer\_id*, *kategori\_pesanan*, *keterangan*, dan *status\_pesanan*. Detail informasi mengenai tabel *pesanan\_customers* dapat dilihat pada Tabel 3.9.

Tabel 3.9 Tabel Pesanan Customer

Nama Atribut	Type Data	Keterangan
id	bigint(20) (PK)	Id data pesanan pelanggan dan merupakan <i>primary key</i> dari tabel <i>pesanan_customers</i>
customer_id	bigint(20) (FK)	Id pengguna aplikasi <i>mobile</i> yang memiliki relasi dengan tabel <i>customers</i> dan merupakan <i>foreign key</i> dari tabel <i>pesanan_customers</i>

kategori_pesanan	varchar(255)	Kategori pesanan yang dipesan, misalnya cuci emas, lebur emas, perbaikan emas, atau jasa desain model emas
keterangan	text	Keterangan atau deskripsi dari pesanan
status_pesanan	varchar(255)	Status dari pesanan

### 3.5.2.10 Tabel Data Pemesanan

Tabel *data\_pemesanans* menyimpan data pemesanan layanan/jasa emas dengan atribut *id*, *bulan*, *tanggal\_pemesanan*, *nama\_pemesan\_layanan*, *kategori\_pemesanan*, *harga\_jasa*, dan *deskripsi*. Detail informasi mengenai tabel *data\_pemesanans* dapat dilihat pada Tabel 3.10.

Tabel 3.10 Tabel Data Pemesanan

Nama Atribut	Tipe Data	Keterangan
id	bigint(20) (PK)	Id data pemesanan dan merupakan <i>primary key</i> dari tabel <i>data_pemesanans</i>
bulan	varchar(255)	Atribut yang menyimpan data bulan pada data pemesanan
tanggal_pemesanan	date	Atribut yang menyimpan data tanggal pemesanan
nama_pemesan_layanan	varchar(255)	Nama pelanggan yang memesan layanan
kategori_pemesanan	int(11)	Jenis atau kategori pemesanan, misalnya cuci emas, lebur emas, perbaikan emas, atau jasa desain model emas
harga_jasa	int(11)	Biaya dari layanan/jasa yang dipesan
deskripsi	text	Deskripsi atau keterangan mengenai data pemesanan

### 3.5.2.11 Tabel OTP

Tabel *otps* menyimpan kode OTP (*one time password*) yang di-*generate* oleh sistem ketika pengguna ingin mengganti kata sandi mereka. Tabel ini menyimpan atribut *id*, *identifier*, *token*, *validity*, dan *valid*. Detail informasi mengenai tabel *otps* dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel OTP

Nama Atribut	Tipe Data	Keterangan
--------------	-----------	------------

id	int(10) (PK)	Id otp dan merupakan <i>primary key</i> dari tabel <i>otps</i>
identifier	varchar(255)	Atribut yang menyimpan email pengguna
token	varchar(255)	Atribut yang menyimpan kode OTP yang digunakan untuk <i>reset password</i>
validity	int(11)	Atribut yang menyimpan durasi berlakunya kode OTP yang di- <i>generate</i> sebelum <i>expired</i>
valid	tinyint(1)	Atribut yang menandakan apakah otp <i>valid</i> atau tidak

### 3.6 Desain User Interface

Pada bagian ini akan dijelaskan mengenai beberapa rancangan awal atau sketsa dari tampilan *user interface* sebagai gambaran dari tampilan aplikasi yang akan dibuat. Desain *user interface* terbagi menjadi dua, yaitu desain *user interface* versi *mobile app* untuk pelanggan atau pengguna aplikasi *mobile* Toko Emas XYZ dan desain *user interface* versi *web app* untuk pihak toko atau pengguna *website* Toko Emas XYZ.

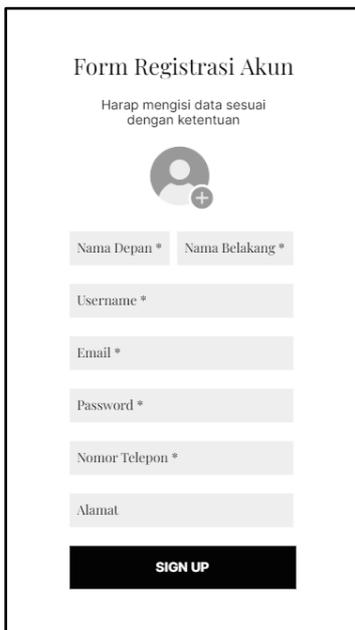
#### 3.6.1 Desain User Interface Versi Mobile App

##### 3.6.1.1 Desain User Interface Versi Mobile App Halaman Login

Gambar 3. 20 Desain *User Interface* Versi *Mobile* Halaman *Login*

Gambar 3.20 merupakan desain *user interface* versi *mobile* untuk halaman *login*. Jika *user* belum memiliki akun, maka *user* bisa mengklik tulisan “Daftar di sini” dan akan diarahkan ke halaman *register* untuk registrasi akun. Setelah berhasil mendaftar akun, *user* bisa mengisi atau memasukkan *email* dan *password*, lalu klik tombol *log in*. Jika *user* lupa kata sandi, maka *user* bisa klik tulisan “Lupa kata sandi?”, jika data yang dimasukkan *valid*, maka *user* akan diarahkan ke halaman utama.

### 3.6.1.2 Desain User Interface Versi Mobile App Halaman Register



Form Registrasi Akun

Harap mengisi data sesuai dengan ketentuan



Nama Depan \*    Nama Belakang \*

Username \*

Email \*

Password \*

Nomor Telepon \*

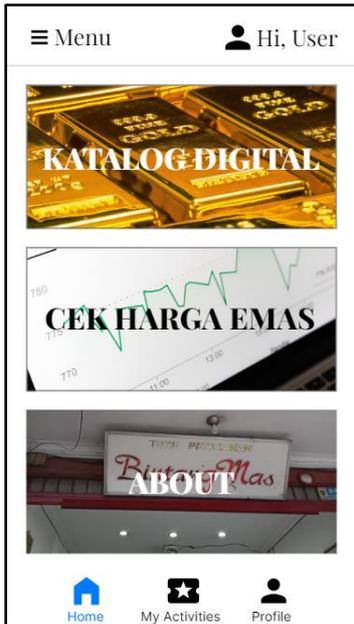
Alamat

**SIGN UP**

Gambar 3.21 Desain *User Interface* Versi *Mobile* Halaman *Register*

Gambar 3.21 merupakan desain *user interface* versi *mobile* untuk halaman *register*. *User* bisa mengisi data sesuai dengan ketentuan yang ada, lalu klik tombol *sign up*. Jika berhasil, maka akun berhasil didaftarkan dan langsung diarahkan ke halaman *login*.

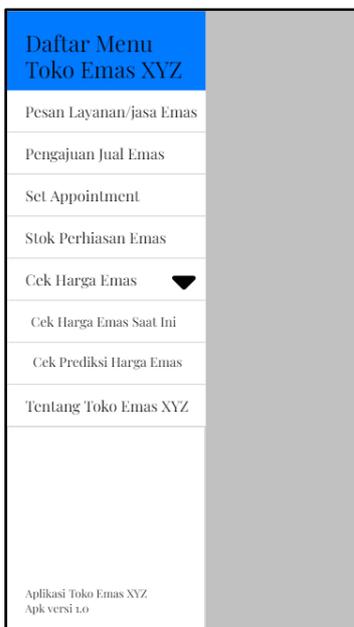
### 3.6.1.3 Desain User Interface Versi Mobile App Halaman Utama



Gambar 3.22 Desain *User Interface* Versi *Mobile* Halaman Utama (*Home Screen*)

Gambar 3.22 merupakan desain *user interface* versi *mobile* untuk halaman utama (*home screen*). Pada halaman ini terdapat banyak menu atau tombol-tombol yang bisa mengarahkan *user* ke halaman lain sesuai dengan pilihan *user*.

### 3.6.1.4 Desain *User Interface* Versi *Mobile App* Halaman *Side Navigation Menu*

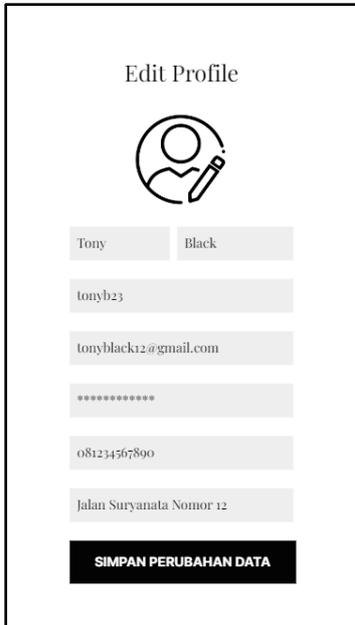


Gambar 3.23 Desain *User Interface* Versi *Mobile* Halaman *Side Navigation Menu*

Gambar 3.23 merupakan desain *user interface* versi *mobile app* untuk halaman *side navigation menu*. Tampilan halaman ini akan muncul ketika *user* mengklik *icon* garis tiga atau

*icon* menu pada halaman utama (*home screen*). Terdapat beberapa daftar menu yang ketika *user* klik menu tersebut, maka akan diarahkan ke halaman sesuai pilihan *user*.

### 3.6.1.5 Desain User Interface Versi Mobile App Halaman Edit Profile

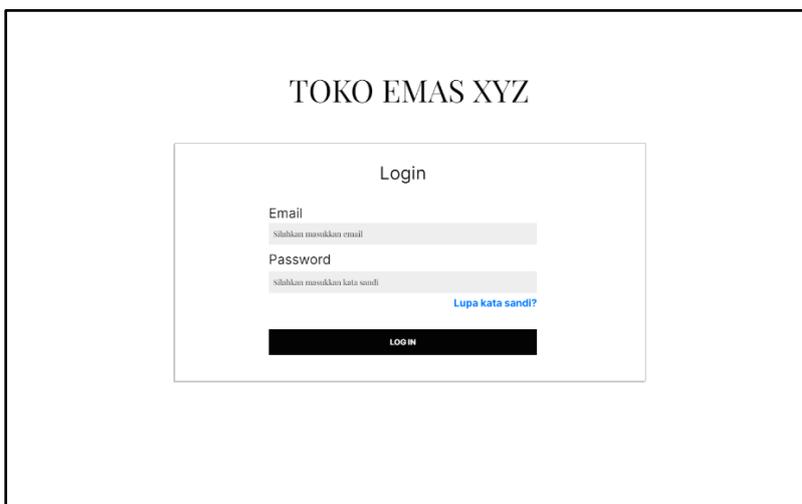


Gambar 3.24 Desain *User Interface* Versi *Mobile* Halaman *Edit Profile*

Gambar 3.24 merupakan desain *user interface* versi *mobile* untuk halaman *edit profile*. Jika *user* ingin mengubah data pada akun, maka *user* bisa mengganti isi data yang ingin diubah, lalu klik tombol “simpan perubahan data” dan setelah itu data akan diubah.

### 3.6.2 Desain User Interface Versi Web App

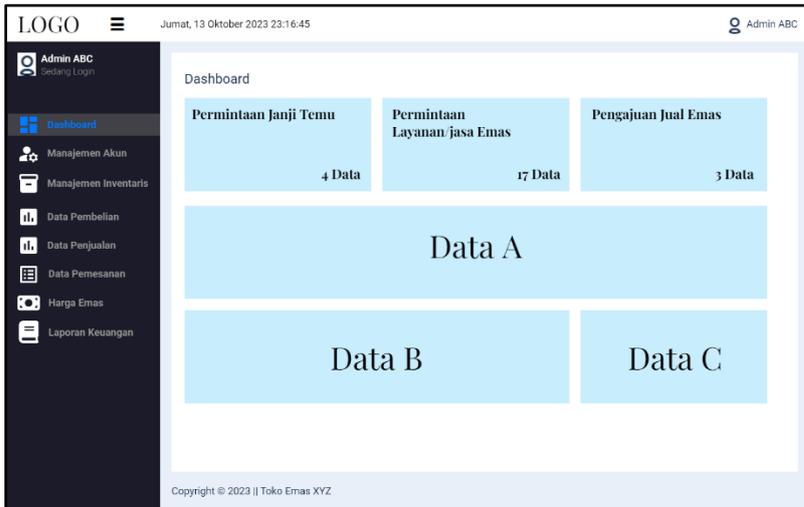
#### 3.6.2.1 Desain User Interface Versi Web App Halaman Login



Gambar 3.25 Desain *User Interface* Versi *Web* Halaman *Login*

Gambar 3.25 merupakan desain *user interface* versi *web* untuk halaman *login*. *Admin* bisa memasukkan *email* dan *password*, lalu klik tombol *log in*. Jika *admin* lupa kata sandi, maka *admin* bisa klik tulisan “Lupa kata sandi?”, jika data yang dimasukkan *valid*, maka *admin* akan diarahkan ke halaman *dashboard* pada *website*.

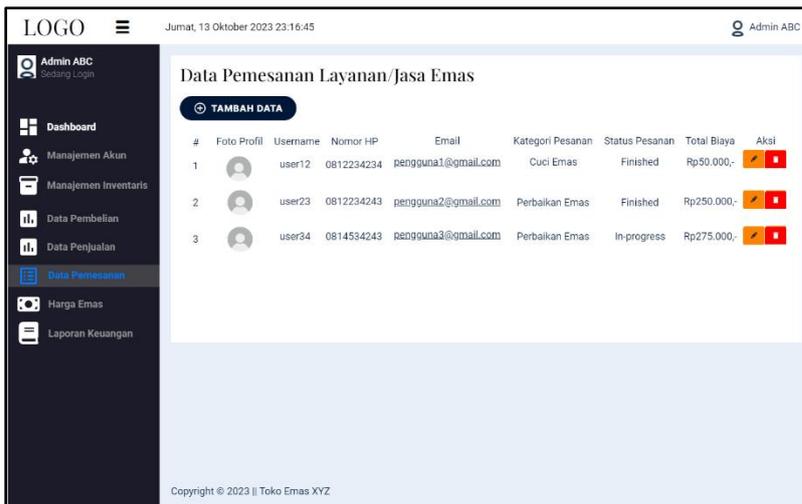
### 3.6.2.2 Desain *User Interface* Versi *Web App* Halaman *Dashboard*



Gambar 3.26 Desain *User Interface* Versi *Web* Halaman *Dashboard*

Gambar 3.26 merupakan desain *user interface* versi *web* untuk halaman *dashboard*. Pada bagian kiri *website* terdapat *sidebar* yang berisi menu-menu yang jika diklik akan mengarah ke masing-masing halaman sesuai dengan menu yang dipilih. Pada halaman *dashboard* berisi informasi yang cukup penting, misalnya data pemesanan, permintaan janji temu, permintaan layanan/jasa emas, pengajuan jual emas oleh pelanggan, dan lain sebagainya.

### 3.6.2.3 Desain *User Interface* Versi *Web App* Halaman *Data Pemesanan*



Gambar 3.27 Desain *User Interface* Versi *Web* Halaman Data Pemesanan

Gambar 3.27 merupakan desain *user interface* versi *web* untuk halaman data pemesanan. Gambar 3.26 juga merupakan salah satu contoh desain untuk tampilan *website* toko emas XYZ. Halaman ini menampilkan daftar data pelanggan yang melakukan pemesanan jasa/layanan emas. *Admin* bisa menambahkan data dengan mengklik tombol “tambah data” dan jika ingin mengubah atau menghapus data, maka *admin* bisa mengklik tombol pada kolom “aksi” sesuai dengan kebutuhan.