**Dokumen 5 Tugas Besar IF3141**

**Rancangan Sistem Informasi**

**Beanfo**

**Brewspace**

**Disusun oleh:**

**Kelompok K01-G12**

Gagas Praharsa Bahar / 13520016

Bryan Bernigen / 13520034

Angelica Winasta Sinisuka / 13520097

Patrick Amadeus Irawan / 13520109

Owen Christian Wijaya / 13520124

Jevant Jedidia Augustine / 13520133

**Program Studi Teknik Informatika**

**Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung**

**Jl. Ganesha 10, Bandung 40132**

**2022**

| Ganesha (4) | **Program Studi**  **Teknik Informatika**  **STEI – ITB** | Nomor Dokumen | Jumlah Halaman |
| --- | --- | --- | --- |
| **SI-05/K01-G12** | **43** |

# Daftar Isi

[Daftar Isi](#_heading=h.htgwpows9n6n) 2

[Daftar Gambar](#_heading=h.1frjl1c7s93g) 3

[Daftar Tabel](#_heading=h.vdnnnziekbij) 4

[Bab I  
Pendahuluan](#_heading=h.ke30riia50c) 6

[1.1. Tujuan Dokumen](#_heading=h.pc1i7n19ex87) 6

[1.2. Definisi dan Istilah](#_heading=h.icbwpoa1btwg) 6

[Bab II  
Rancangan Desain Sistem Informasi](#_heading=h.j3s2wb3kk0iv) 7

[2.1. Pemodelan Proses](#_heading=h.t0ajacnbuuy5) 7

[2.2. Pemodelan Data](#_heading=h.9vx19d2p661z) 9

[2.3. Pemodelan Fungsional](#_heading=h.me4a5ih1030c) 16

[2.4. Pemodelan Struktur](#_heading=h.jwip9vpinbyf) 22

[2.5. Pemodelan Behavior](#_heading=h.xgrpgt21aofb) 26

[2.6. User Interface](#_heading=h.6rw57tbumooz) 31

[Bab III  
Rencana Pengujian Sistem Informasi](#_heading=h.x2itp967kzu0) 38

[Bab IV  
Kesimpulan dan Saran](#_heading=h.s1ruejsfk521) 43

[4.1. Kesimpulan](#_heading=h.mb1dkt73li5g) 43

[4.2. Saran](#_heading=h.cs7zoect2xjv) 43

# Daftar Gambar

| Gambar 2.1 STD Sistem Informasi | 8 |
| --- | --- |
| Gambar 2.2.1 DFD Level 0 | 9 |
| Gambar 2.2.2 DFD Level 1 | 1o |
| Gambar 2.2.3 DFD Level 2: Login dan Verifikasi | 11 |
| Gambar 2.2.4 DFD Level 2: Fitur Administrator | 12 |
| Gambar 2.2.5 ERD Sistem Informasi | 14 |
| Gambar 2.3 Use Case Diagram | 16 |
| Gambar 2.4 Pemodelan Struktur Sistem Informasi | 22 |
| Gambar 2.5.1 Sequence Diagram Login | 26 |
| Gambar 2.5.2 Sequence Diagram Menampilkan Daftar Pelanggan | 26 |
| Gambar 2.5.3 Sequence Diagram Menambahkan Daftar Pelanggan | 27 |
| Gambar 2.5.4 Sequence Diagram Menampilkan Daftar Promosi | 28 |
| Gambar 2.5.5 Sequence Diagram Menambahkan Daftar Promosi | 28 |
| Gambar 2.5.6 Sequence Diagram Menampilkan Daftar Transaksi | 29 |
| Gambar 2.5.7 Sequence Diagram Menambahkan Daftar Transaksi | 29 |
| Gambar 2.5.8 Sequence Diagram Cek Promo Customer | 30 |
| Gambar 2.5.9 Sequence Diagram Logout | 30 |

# Daftar Tabel

| Tabel 1.2 Definisi dan Istilah | 6 |
| --- | --- |
| Tabel 2.2.1 Kamus Data DFD | 12 |
| Tabel 2.2.2 Basis Data | 14 |
| Tabel 2.3.1 Deskripsi Use Case Login | 17 |
| Tabel 2.3.2 Deskripsi Use Case Buka Daftar Pelanggan | 17 |
| Tabel 2.3.3 Deskripsi Use Case Buka Daftar Promosi | 17 |
| Tabel 2.3.4 Deskripsi Use Case Buka Daftar Promosi Khusus | 18 |
| Tabel 2.3.5 Deskripsi Use Case Tambah Data Transaksi | 18 |
| Tabel 2.3.6 Deskripsi Use Case Tambah Data Pelanggan | 16 |
| Tabel 2.3.7 Deskripsi Use Case Tambah Data Promosi | 19 |
| Tabel 2.3.8 Deskripsi Use Case Tambah Data Promosi Khusus | 20 |
| Tabel 2.3.9 Deskripsi Use Case Buka Daftar Transaksi | 20 |
| Tabel 2.3.10 Deskripsi Use Case Logout | 21 |
| Tabel 2.4.1 Deskripsi Class Diagram Admin | 19 |
| Tabel 2.4.2 Deskripsi Class Diagram Customer | 22 |
| Tabel 2.4.3 Deskripsi Class Diagram Transaksi | 24 |
| Tabel 2.4.4 Deskripsi Class Diagram Promo | 24 |
| Tabel 2.6.1 *Screen Mockup* Layar *Login* | 31 |
| Tabel 2.6.2 *Screen Mockup* Layar Daftar Pelanggan | 31 |
| Tabel 2.6.3 *Screen Mockup* Layar Tambah Pelanggan | 32 |
| Tabel 2.6.4 *Screen Mockup* Layar Daftar Promosi | 32 |
| Tabel 2.6.5 *Screen Mockup* Layar Tambah Promosi | 33 |
| Tabel 2.6.6 *Screen Mockup* Layar Daftar Promosi Khusus | 34 |
| Tabel 2.6.7 *Screen Mockup* Layar Tambah Promosi Khusus | 35 |
| Tabel 2.6.8 *Screen Mockup* Layar Daftar Transaksi | 36 |
| Tabel 2.6.8 *Screen Mockup* Layar Tambah Transaksi | 36 |
| Tabel 2.6.9 Pemetaan Layar | 37 |

# Bab I Pendahuluan

## Tujuan Dokumen

Tujuan dari penyusunan dokumen ini adalah untuk memodelkan proses, data, fungsionalitas, struktur, dan behaviour sistem informasi melalui STD, DFD, ERD, use case diagram, class diagram, dan sequence diagram. Dokumen ini juga bertujuan untuk menampilkan antarmuka sementara. Dokumen ini juga bertujuan untuk menunjukkan rencana pengujian sistem melalui skenario-skenario uji yang mungkin terjadi.

## Definisi dan Istilah

Tabel 1.2 Definisi dan Istilah

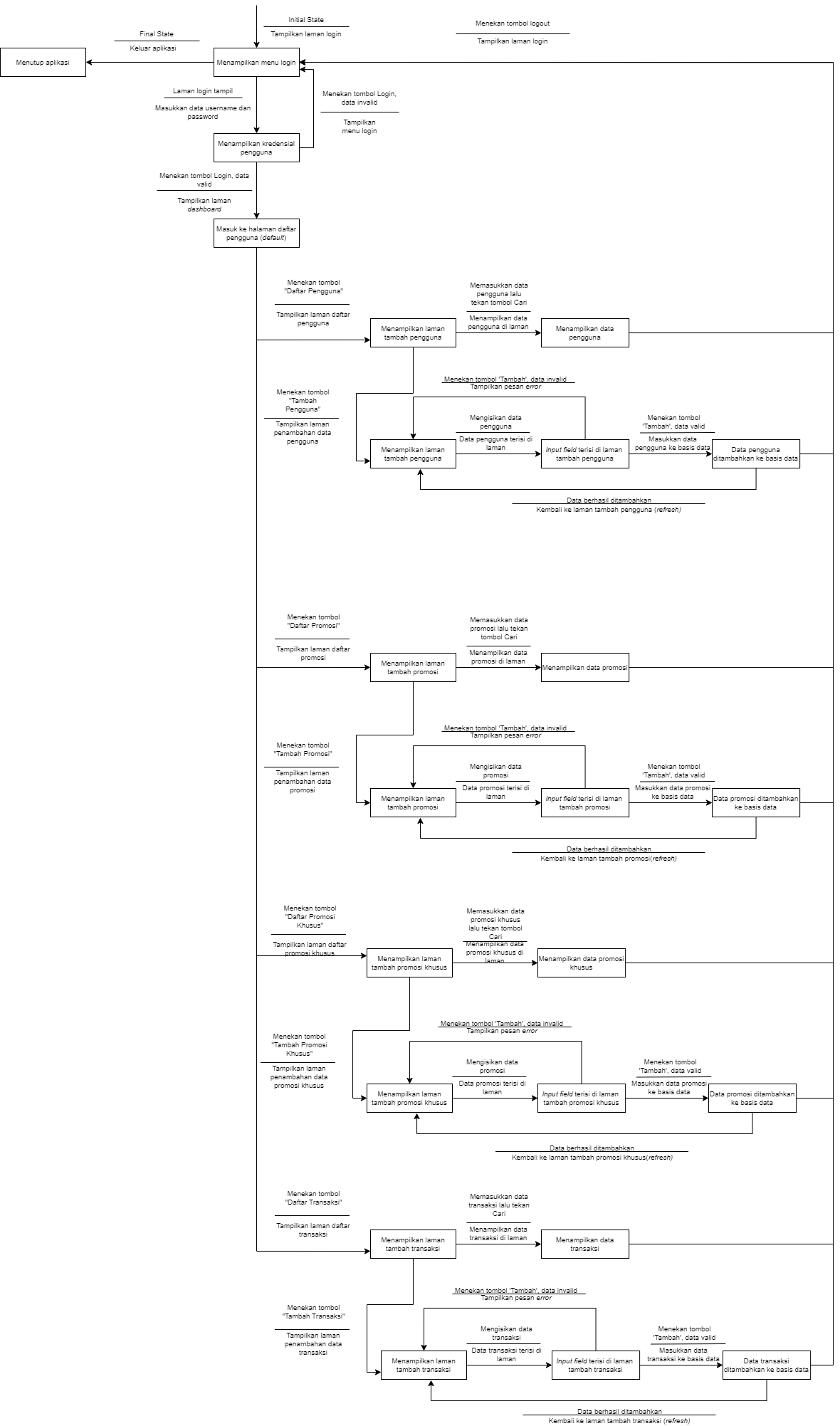
| **Istilah/Singkatan** | **Definisi** |
| --- | --- |
| DFD | *Data Flow Diagram,* sebuah diagram untuk mengidentifikasi aliran data pada perangkat lunak dan transformasi terkait aliran data tersebut |
| ERD | *Entity Relation Diagram,* sebuah diagram untuk mewakili hubungan antar entitas dalam sebuah basis data |
| STD | *State Transition Diagram*, sebuah diagram untuk menjelaskan perilaku dinamika sistem dan transisi status dari suatu entitas |
| SMS | *Short Message Service,* sebuah protokol komunikasi yang digunakan telepon seluler melalui jaringan 2G, 3G, 4G atau 5G |
| PoS | *Point of Sales* merupakan tempat dimana pelanggan membayar untuk produk atau servis |

# Bab II Rancangan Desain Sistem Informasi

## Pemodelan Proses

Sistem informasi yang dibuat akan mengikuti alur sebagai berikut:

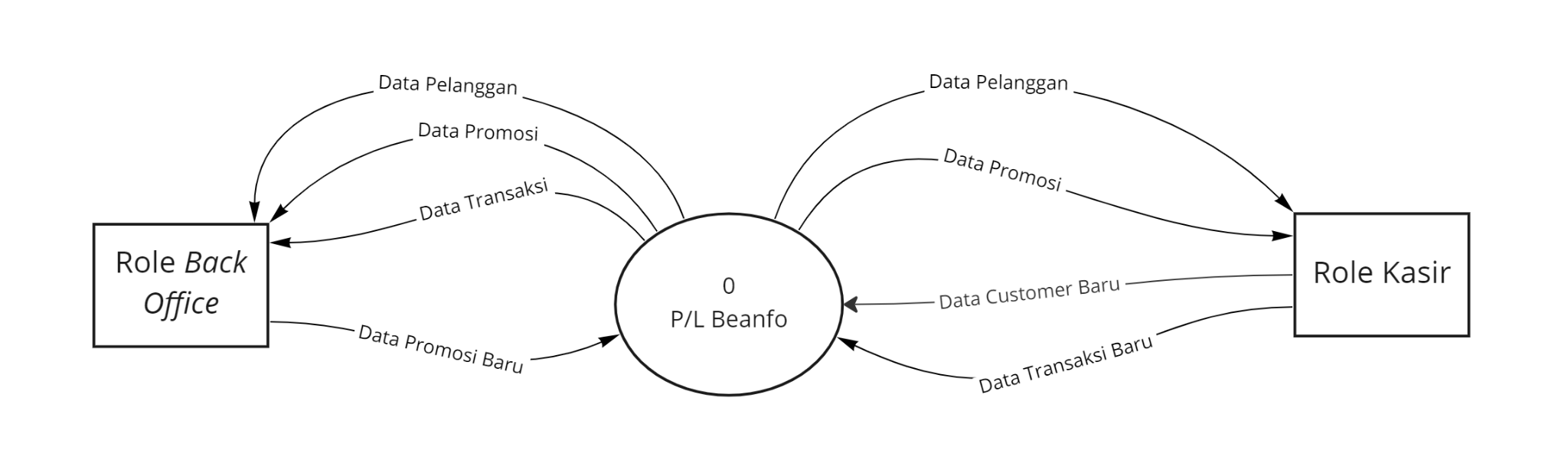
1. Alur dimulai saat administrator akan melakukan *login* dengan akun yang sudah dibuatkan sebelumnya. Akun yang dibuat akan berdasarkan *platform Odoo,* namun pada dokumen ini, akan dibuat halaman *login* sebagai visualisasi.
2. Administrator akan mempunyai dua buah hak, yaitu untuk membuka daftar dari data yang disimpan pada *database* dan menambahkan data baru ke dalam *database.* Data yang akan disimpan di dalam *database* adalah data terkait pelanggan yang terdaftar di dalam program loyalti, data terkait promosi-promosi umum yang hendak disebarluaskan ke pelanggan-pelanggan melalui media SMS / *e-mail*, dan data transaksi yang dilakukan oleh pelanggan terdaftar. Promosi khusus seorang pelanggan spesifik akan dikalkulasi secara otomatis oleh sistem, sehingga akan muncul apabila dilakukan pencarian / penampilan data pelanggan tersebut.
3. Pada halaman daftar data, administrator dapat melihat daftar lengkap dari data yang telah disimpan, serta mencari sebuah data spesifik.
   1. Untuk data pelanggan, data yang akan ditampilkan adalah nama pelanggan, SMS / e-mail pelanggan, serta jumlah poin yang telah dikumpulkan berdasarkan transaksi
   2. Untuk data promosi, data yang akan ditampilkan adalah nama promosi, deskripsi promosi (apabila ada), tanggal awal dan akhir waktu promosi aktif, serta kode promosi
   3. Untuk data transaksi, data yang akan ditampilkan adalah kode transaksi, nominal, serta pelanggan yang melakukan transaksi tersebut. Data transaksi hanya digunakan untuk akumulasi poin saja, sehingga tidak diperlukan perhitungan secara detail dan diasumsikan perhitungan sudah dilakukan oleh sistem *point of sales* di perusahaan
4. Pada halaman penambahan data, administrator dapat menambahkan data baru ke dalam sistem, dapat berupa data transaksi, pelanggan, ataupun promosi (lihat **bagian 2.2.2** untuk keterangan lebih lanjut)
5. Alur selesai saat administrator *logout* dari sistem

******

Gambar 2.1 STD Sistem Informasi

## Pemodelan Data

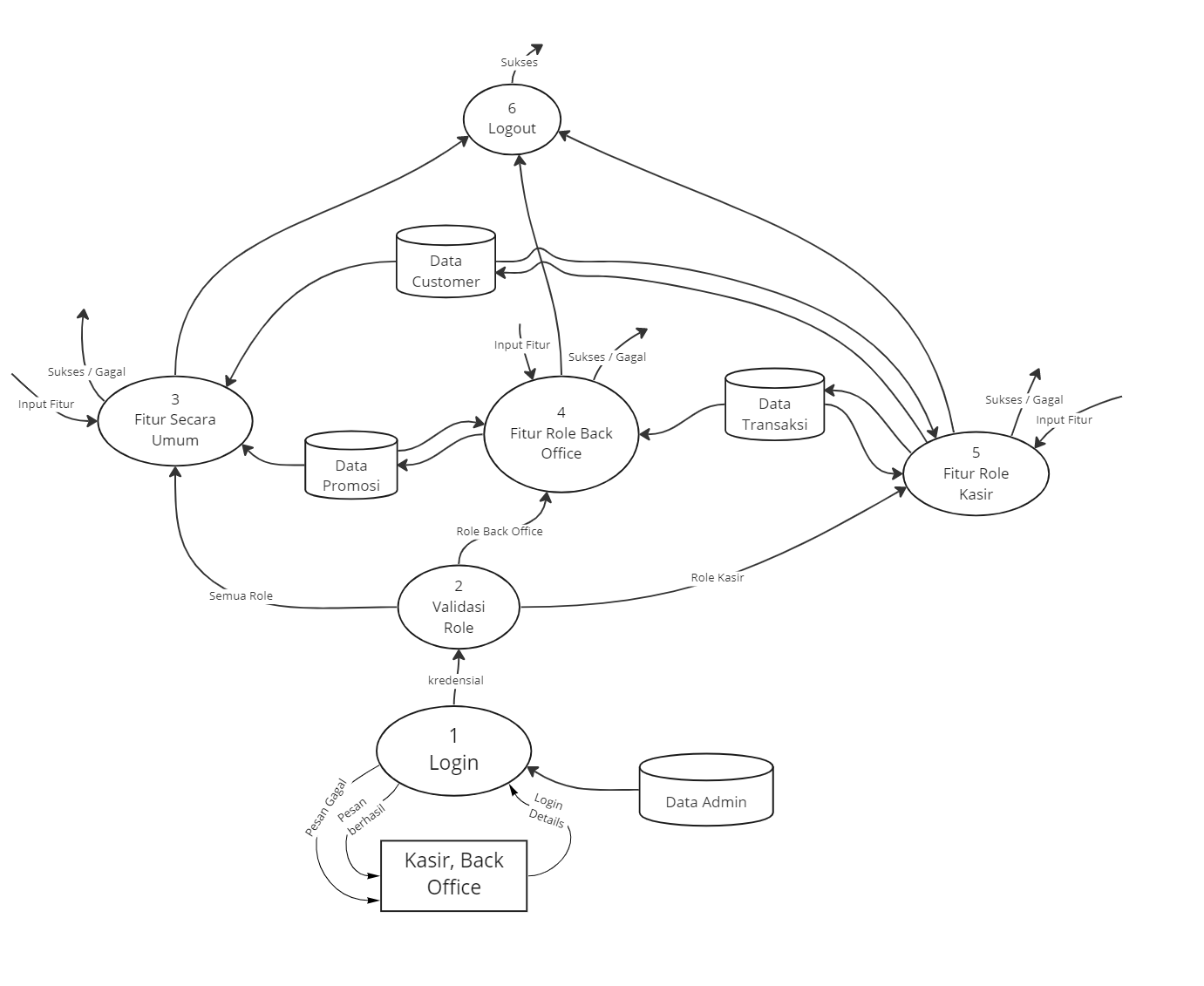
1. ***Data Flow Diagram***
2. **DFD Level 0**

****

Gambar 2.2.1 DFD Level 0

Diagram konteks (DFD Level 0) menjabarkan aliran data pada kedua *role* administrator yakni *back office* dan kasir. Pada *back office* aliran data dikhususkan untuk bagian promosi sedangkan pada kasir dikhususkan berkaitan dengan data *customer* dan transaksi. Penerimaan data secara garis besar serupa kecuali pada bagian transaksi yang secara eksklusif hanya dapat diakses oleh *role back office*.

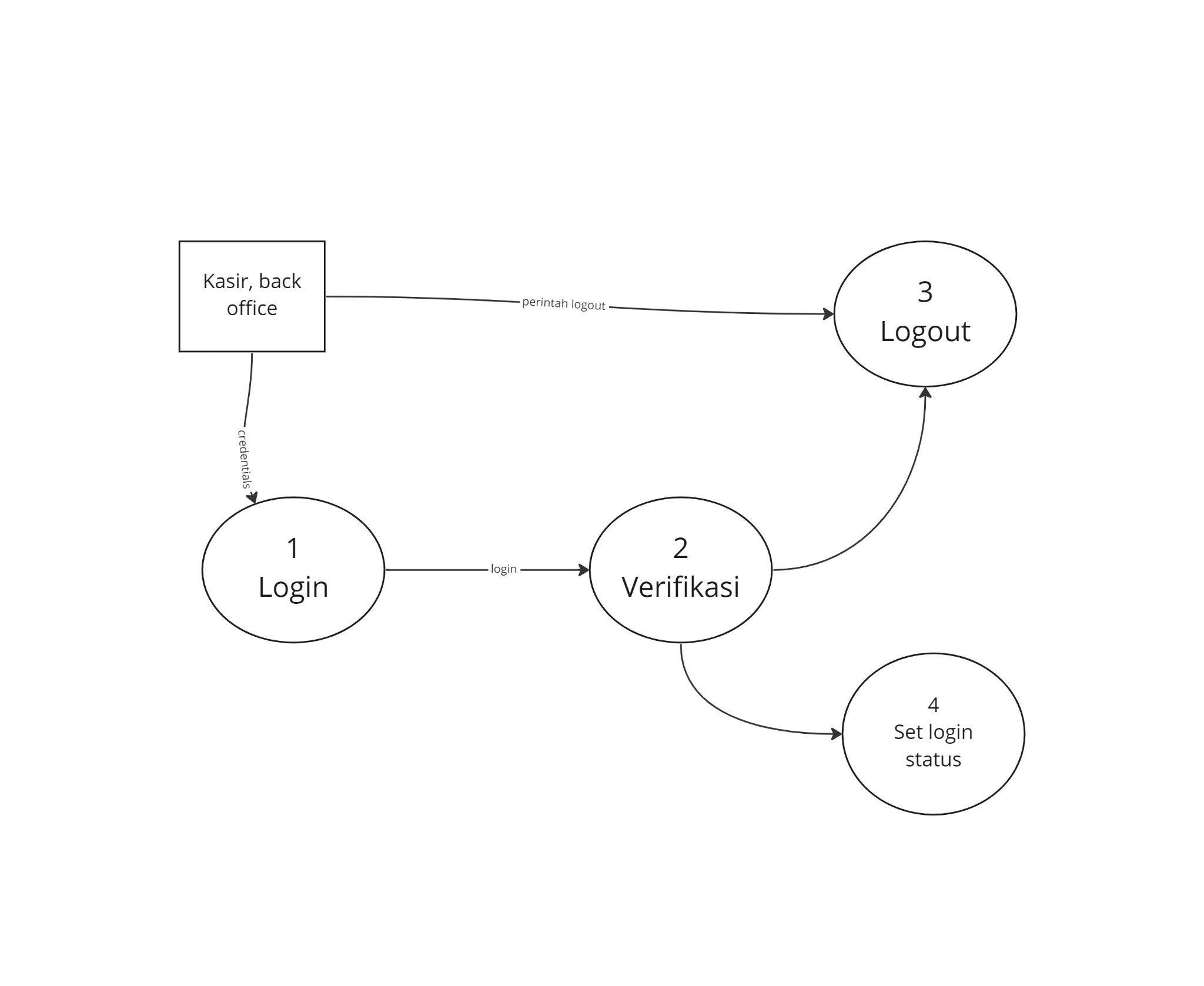
1. **DFD Level 1**

**

Gambar 2.2.2 DFD Level 1

Secara garis besar, aliran data yang diilustrasikan sesuai DFD Level 1 ini dimulai pada abstraksi fitur *login* hingga berakhir pada fitur *logout* dimana data yang dialirkan merupakan informasi terkait kredensial pengguna dan autentikasi. Di antara kedua fitur tersebut, terdapat abstraksi lebih lanjut mengenai penggunaan fitur sesuai dengan *role* yang terautentikasi yang dibagi menjadi *role back office* dan *role* kasir dengan aliran data berkaitan dengan data *input* yang berkorespondensi dengan fitur masing-masing serta perwujudan respon keluaran dari sistem terkait keberhasilan. Fitur-fitur ini disegmentasi menjadi 3 bagian berdasarkan ranah pengguna yang diperjelas pada DFD Level berikutnya.

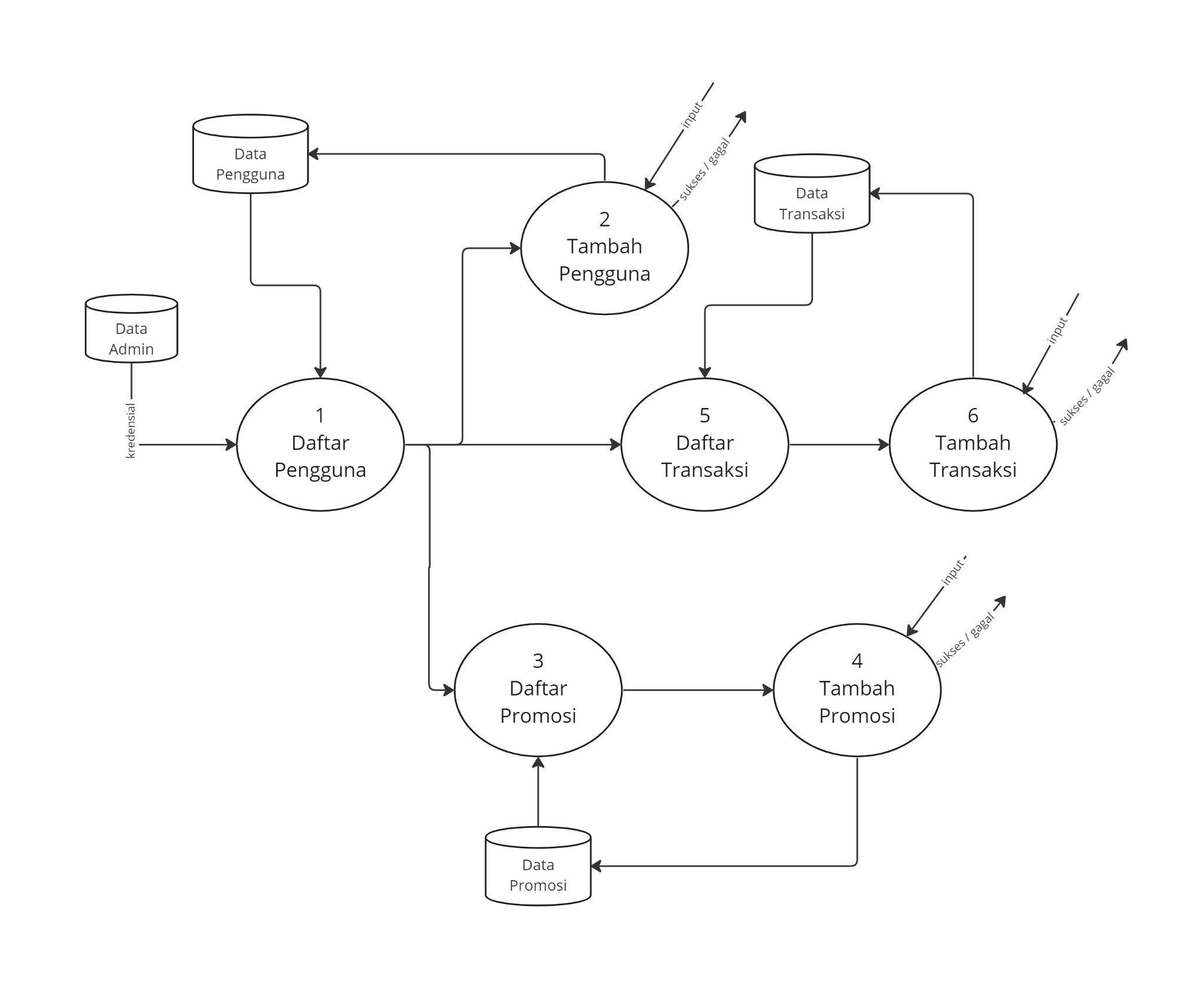
1. **DFD Level 2: Login dan Verifikasi**

****

Gambar 2.2.3 DFD Level 2: Login dan Verifikasi

DFD Level 2 bagian Login dan Verifikasi menunjukkan aliran data pada fitur login dengan pemberian kredensial dari pengguna (kasir dan back office), lalu pengguna dapat melakukan login untuk kemudian dilakukan verifikasi pada database. Setelah itu, status login dapat diatur oleh sistem, dan dimungkinkan untuk melakukan logout setelah login untuk keluar dari sistem.

1. **DFD Level 2: Fitur Administrator (Back Office dan Kasir)**



Gambar 2.2.4 DFD Level 2: Fitur Administrator

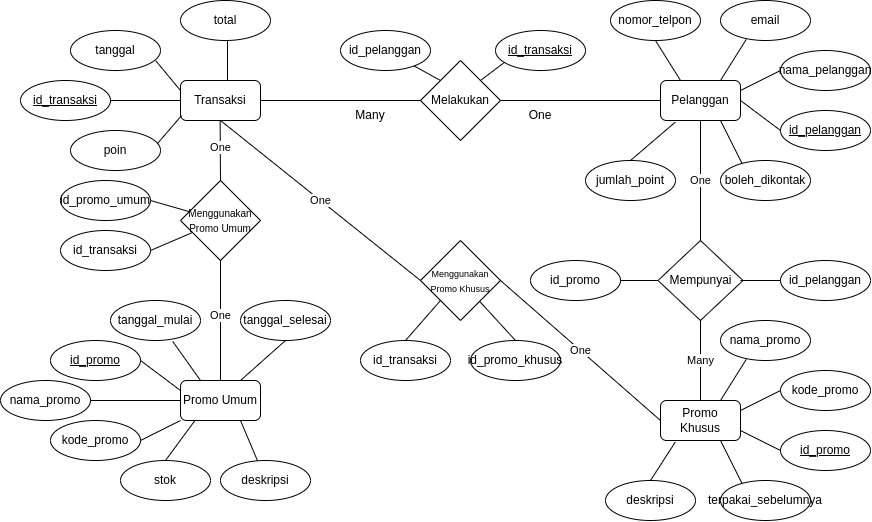
DFD Level 2 bagian Fitur Administrator menggambarkan aliran data pada fitur-fitur administrator yang sebelumnya dilakukan abstraksi pada DFD Level 1. Fitur untuk Kasir dan Back Office sama, dengan fitur-fitur utama melihat daftar pengguna, daftar transaksi, dan daftar promosi, serta dapat melakukan penambahan terhadap ketiga data tersebut.

1. ***Data Dictionary* DFD**

Tabel 2.2.1 Kamus Data DFD

| **ID Data *Flow*** | **Nama Data *Flow*** | **Deskripsi Data *Flow*** | ***Source*** | ***Destination*** | **Type** |
| --- | --- | --- | --- | --- | --- |
| DF-01 | Data pelanggan | Data pelanggan terdiri atas nama, email, nomor telepon, jumlah\_point, serta boleh\_dikontak | Kasir, Proses tambah pelanggan | Proses daftar pelanggan | Nama: varchar  email: varchar  Nomor telepon: varchar  jumlah\_point: integer  boleh\_dikontak: boolean |
| DF-02 | Data transaksi | Data transaksi yang terdiri atas id\_pelanggan, tanggal, total, poin, id promo umum dan id promo khusus | Kasir, Proses tambah transaksi | Proses daftar transaksi | ID pelanggan: integer  Tanggal: datetime  total: integer  poin: integer  ID promo umum: integer  ID promo khusus: integer |
| DF-03 | Data promosi umum | Data promosi yang terdiri atas nama promo, tanggal mulai, tanggal selesai, stok, dan deskripsi | Back Office, proses tambah promosi | Proses daftar promosi | Nama promo: varchar  Tanggal mulai: datetime  Tanggal selesai: datetime  Stok: integer  Deskripsi: varchar |
| DF-04 | Data promosi khusus | Data promosi yang terdiri atas nama promo, tanggal selesai, deskripsi, dan penunjuk telah terpakai sebelumnya. | Back office, proses tambah promosi | Proses daftar promosi | Nama promo: varchar  Tanggal selesai: datetime  Deskripsi: varchar  Terpakai sebelumnya: boolean |
| DF-05 | Data kredensial | Data kredensial yang terdiri atas username dan password | Kasir atau back office | Proses login | Username: varchar  Password: varchar |

1. ***Entity Relationship Diagram***

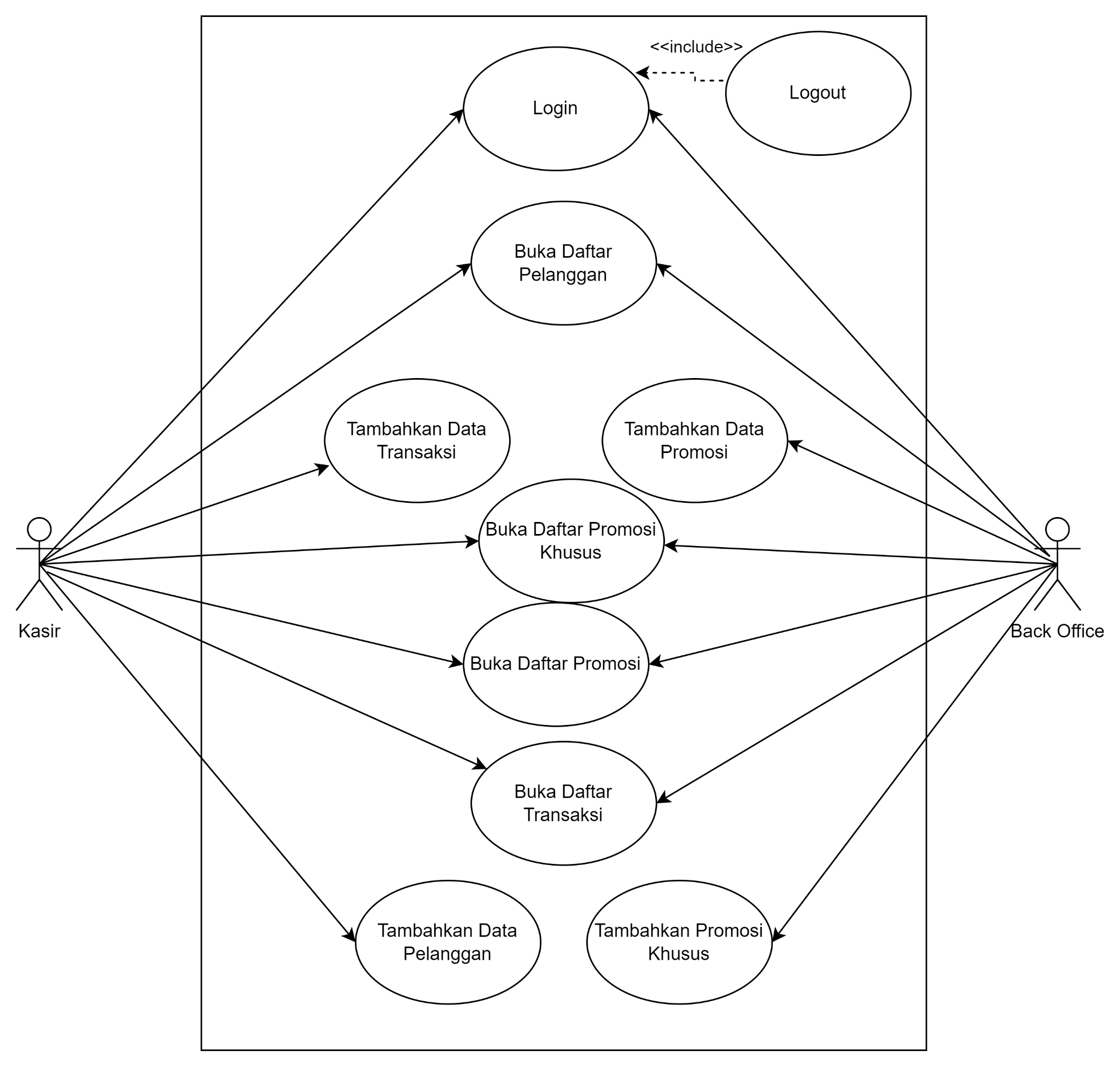
**

Gambar 2.2.5 ERD Sistem Informasi

Tabel 2.2.2 Basis Data

| **ID-Tabel** | **Nama Tabel** | ***Primary Key*** | **Deskripsi** |
| --- | --- | --- | --- |
| D-01 | Tabel Pelanggan | ID\_Pelanggan | Data pelanggan terdiri dari:   * id\_pelanggan, *primary key* untuk pelanggan * nama\_pelanggan, nama lengkap pelanggan terdaftar * e\_mail, untuk menyimpan informasi email pelanggan * nomor\_telpon, untuk menyimpan informasi nomor telpon pelanggan * jumlah\_point, untuk menyimpan total poin yang dimiliki pelanggan * boleh\_dikontak, khusus untuk pendaftar sistem loyalty, apakah ingin menerima SMS / email tentang informasi promosi |
| D-02 | Tabel Transaksi | ID\_Transaksi | Data transaksi terdiri dari:   * id\_transaksi, untuk mencatat id\_transaksi dari sistem PoS yang ada * tanggal, untuk mencatat tanggal transaksi * total, untuk mencatat nominal final transaksi dari sistem PoS |
| D-03 | Tabel Promo Umum | id\_promo | Data promo umum terdiri dari:   * id\_promo, untuk mencatat ID unik promosi * nama\_promo, untuk menyimpan nama promo * tanggal\_mulai, mencatat tanggal mulai promo * tanggal\_selesai, mencatat tanggal selesai promo * stok, mencatat berapa kali kode promo dapat di-redeem * deskripsi, menyimpan deskripsi promo |
| D-04 | Tabel Promo Khusus | id\_promo | Data promo khusus terdiri dari:   * id\_promo, untuk mencatat ID unik promosi * nama\_promo, untuk menyimpan nama promo * deskripsi, menyimpan deskripsi promo * terpakai\_Sebelumnya, untuk mengetahui apakah promo tersebut sudah pernah dipakai sebelumnya |

## Pemodelan Fungsional

******

Gambar 2.3 Use Case Diagram

Tabel 2.3.1 Deskripsi Use Case Login

| ***Name*** | Login |
| --- | --- |
| ***Actors*** | Kasir, Back Office |
| ***Entry Conditions*** | Actor sudah mempunyai akun masing-masing |
| ***Flow of Events*** | 1. Actor membuka sistem 2. Actor mengisikan kredensial yang dimiliki, lalu menekan tombol Masuk |
| ***Exit Conditions*** | Actor dapat mengakses sistem dengan hak-hak yang ia miliki |
| ***Special requirements*** | - |

Tabel 2.3.2 Deskripsi Use Case Buka Daftar Pelanggan

| ***Name*** | Buka Daftar Pelanggan |
| --- | --- |
| ***Actors*** | Kasir, Back Office |
| ***Entry Conditions*** | Data pelanggan telah tersedia |
| ***Flow of Events*** | 1. Actor melakukan login ke sistem 2. Actor memilih opsi Daftar Pelanggan 3. Layar Daftar Pelanggan akan tampil beserta baris berisi informasi tiap pelanggan di halaman tersebut 4. Actor dapat memilih untuk melihat informasi lebih lanjut tentang pelanggan |
| ***Exit Conditions*** | Actor mendapatkan informasi tentang pelanggan |
| ***Special requirements*** | - |

Tabel 2.3.3 Deskripsi Use Case Buka Daftar Promosi

| ***Name*** | Buka Daftar Promosi |
| --- | --- |
| ***Actors*** | Kasir, Back Office |
| ***Entry Conditions*** | Data promosi telah tersedia |
| ***Flow of Events*** | 1. Actor melakukan login ke sistem 2. Actor memilih opsi Daftar Promosi 3. Layar Daftar Promosi akan tampil beserta baris berisi informasi tiap promosi umum di halaman tersebut 4. Actor dapat memilih untuk melihat informasi lebih lanjut tentang promosi umum |
| ***Exit Conditions*** | Actor mendapatkan informasi tentang promosi umum |
| ***Special requirements*** | - |

Tabel 2.3.4 Deskripsi Use Case Buka Daftar Promosi Khusus

| ***Name*** | Buka Daftar Promosi Khusus |
| --- | --- |
| ***Actors*** | Kasir, Back Office |
| ***Entry Conditions*** | Data promosi telah tersedia |
| ***Flow of Events*** | 1. Actor melakukan login ke sistem 2. Actor memilih opsi Daftar Promosi Khusus 3. Layar Daftar Promosi Khusus akan tampil beserta baris berisi informasi tiap promosi umum di halaman tersebut 4. Actor dapat memilih untuk melihat informasi lebih lanjut tentang promosi umum |
| ***Exit Conditions*** | Actor mendapatkan informasi tentang promosi umum |
| ***Special requirements*** | - |

Tabel 2.3.5 Deskripsi Use Case Tambah Data Transaksi

| ***Name*** | Tambah Data Transaksi |
| --- | --- |
| ***Actors*** | Kasir |
| ***Entry Conditions*** | Actor sudah login dan data transaksi ada di sistem PoS |
| ***Flow of Events*** | 1. Actor melakukan login 2. Actor mendapatkan data dari sistem PoS 3. Actor memasukkan data transaksi dan subtotal akhir dari transaksi 4. Actor memasukkan kode promo yang ingin digunakan (apabila menggunakan promo) 5. Actor menekan tombol Tambahkan, kemudian data akan dimasukkan ke dalam sistem |
| ***Exit Conditions*** | Data transaksi seorang pelanggan yang terdaftar di program *loyalty* akan dimasukkan ke dalam sistem |
| ***Special requirements*** | - |

Tabel 2.3.6 Deskripsi Use Case Tambah Data Pelanggan

| ***Name*** | Tambah Data Pelanggan |
| --- | --- |
| ***Actors*** | Kasir |
| ***Entry Conditions*** | - |
| ***Flow of Events*** | 1. Actor melakukan login 2. Actor memilih opsi Tambahkan Pengguna 3. Actor memasukkan kredensial pengguna dan mencentang opsi Ingin Menerima Berita apabila pelanggan ingin menerima berita umum via SMS / email 4. Actor menekan tombol Tambahkan |
| ***Exit Conditions*** | Data pelanggan akan dimasukkan ke dalam sistem |
| ***Special requirements*** | - |

Tabel 2.3.7 Deskripsi Use Case Tambah Data Promosi

| ***Name*** | Tambah Data Promosi |
| --- | --- |
| ***Actors*** | Back Office |
| ***Entry Conditions*** | - |
| ***Flow of Events*** | 1. Actor melakukan login 2. Actor memilih opsi Tambahkan Promosi 3. Actor memasukkan informasi promosi 4. Actor menekan tombol Tambahkan |
| ***Exit Conditions*** | Data promosi akan dimasukkan ke dalam sistem |
| ***Special requirements*** | - |

Tabel 2.3.8 Deskripsi Use Case Tambah Data Promosi Khusus

| ***Name*** | Tambah Data Promosi Khusus |
| --- | --- |
| ***Actors*** | Back Office |
| ***Entry Conditions*** | - |
| ***Flow of Events*** | 1. Actor melakukan login 2. Actor memilih opsi Tambahkan Promosi Khusus 3. Actor memasukkan informasi promosi 4. Actor menekan tombol Tambahkan |
| ***Exit Conditions*** | Data promosi khusus akan dimasukkan ke dalam sistem |
| ***Special requirements*** | - |

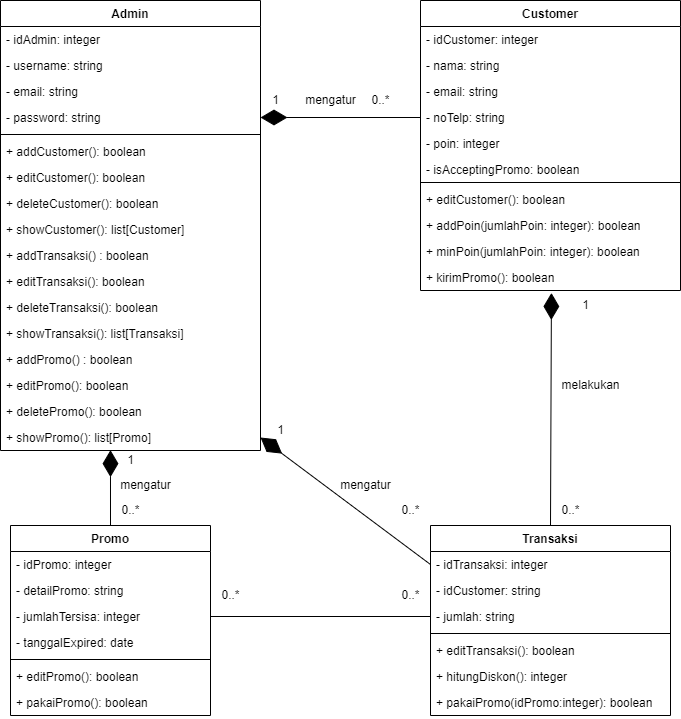
Tabel 2.3.9 Deskripsi Use Case Buka Daftar Transaksi

| ***Name*** | Buka Daftar Transaksi |
| --- | --- |
| ***Actors*** | Back Office |
| ***Entry Conditions*** | - |
| ***Flow of Events*** | 1. Actor melakukan login ke sistem 2. Actor memilih opsi Daftar Transaksi 3. Layar Daftar Transaksi akan tampil beserta baris berisi informasi tiap pelanggan di halaman tersebut 4. Actor dapat memilih untuk melihat informasi lebih lanjut tentang transaksi |
| ***Exit Conditions*** | Actor dapat mendapatkan informasi tentang transaksi |
| ***Special requirements*** | - |

Tabel 2.3.10 Deskripsi Use Case Logout

| ***Name*** | Logout |
| --- | --- |
| ***Actors*** | Kasir, *back office* |
| ***Entry Conditions*** | Actor sudah login ke dalam sistem |
| ***Flow of Events*** | 1. Actor memilih opsi logout |
| ***Exit Conditions*** | Actor dapat keluar dari sistem |
| ***Special requirements*** | - |

## Pemodelan Struktur

******

Gambar 2.4 Pemodelan Struktur Sistem Informasi

Tabel 2.4.1 Deskripsi Class Diagram Admin

| **Name** | Admin | |
| --- | --- | --- |
| **Description** | Kelas yang mengatur tentang pengelolaan customer, transaksi, dan promo | |
| **Attributes** | idAdmin, username, email, dan password | |
| **Methods** | addCustomer(): boolean | Digunakan untuk menambahkan data customer baru ke sistem |
| editCustomer(): boolean | Digunakan untuk mengedit data customer yang telah tersedia pada sistem |
| deleteCustomer(): boolean | Digunakan untuk menghapus data Customer yang telah tersedia pada sistem |
| showCustomer(): list[Customer] | Digunakan untuk menampilkan seluruh Customer yang telah tersedia pada sistem |
| addTransaksi(): boolean | Digunakan untuk menambahkan data Transaksi baru ke sistem |
| editTransaksi(): boolean | Digunakan untuk mengedit data Transaksi yang telah tersedia pada sistem |
| deleteTransaksi(): boolean | Digunakan untuk menghapus data Transaksi yang telah tersedia pada sistem |
| showTransaksi(): list[Transaksi] | Digunakan untuk menampilkan seluruh Transaksi yang telah tersedia pada sistem |
| addPromo(): boolean | Digunakan untuk menambahkan data Promo baru ke sistem |
| editPromo(): boolean | Digunakan untuk mengedit data Promo yang telah tersedia pada sistem |
| deletePromo(): boolean | Digunakan untuk menghapus data Promo yang telah tersedia pada sistem |
| showPromo(): list[Promo] | Digunakan untuk menampilkan seluruh Promo yang telah tersedia pada sistem |
| verifikasi() | Digunakan untuk verifikasi pengguna jika user terdaftar dalam database. Berbentuk prosedur yang akan set session dari sistem. |
| setLoggedIn() | Digunakan untuk *set* variabel jika user sudah *login* atau belum |
| setLogout() | Digunakan untuk *set* variabel kalau user *logout* dan mengakhiri *session* dari sistem. |
| isLoggedIn():Boolean | Digunakan untuk mengembalikan *condition*  jika user valid masuk atau belum ke dalam sistem |
| getCustomerDetail(id):String | Digunakan untuk mendapatkan *detail* dari *customer* berdasarkan id pembeli yang terdaftar |
| validasi():Boolean | Digunakan untuk memeriksa input dari user ke dalam *form* dari halaman |
| getPromoDetail(id): String | Digunakan untuk mendapatkan informasi mengenai promo tertentu yang diidentifikasi dari *id* promo |
| getTransaksiDetail(): String | Digunakan untuk mendapatkan informasi mengenai transaksi tertentu yang diidentifikasi dari *id* transaksi |

Tabel 2.4.2 Deskripsi Class Diagram Customer

| **Name** | Customer | |
| --- | --- | --- |
| **Description** | Kelas yang mengatur tentang pengelolaan Poin dan notifikasi promo seorang customer | |
| **Attributes** | idCustomer, nama, email, noTelp, poin, dan isAcceptingPromo | |
| **Methods** | editCustomer(): boolean | Digunakan untuk mengedit data customer yang telah tersedia pada sistem |
| addPoin(jumlahPoin: integer): boolean | menambah poin customer sebanyak input |
| minPoin(jumlahPoin: integer): boolean | mengurangi poin customer sebanyak input |
| kirimPromo(): boolean | mengirim promo pada noTelp atau email yang terdaftar jika isAcceptingPromo = true |

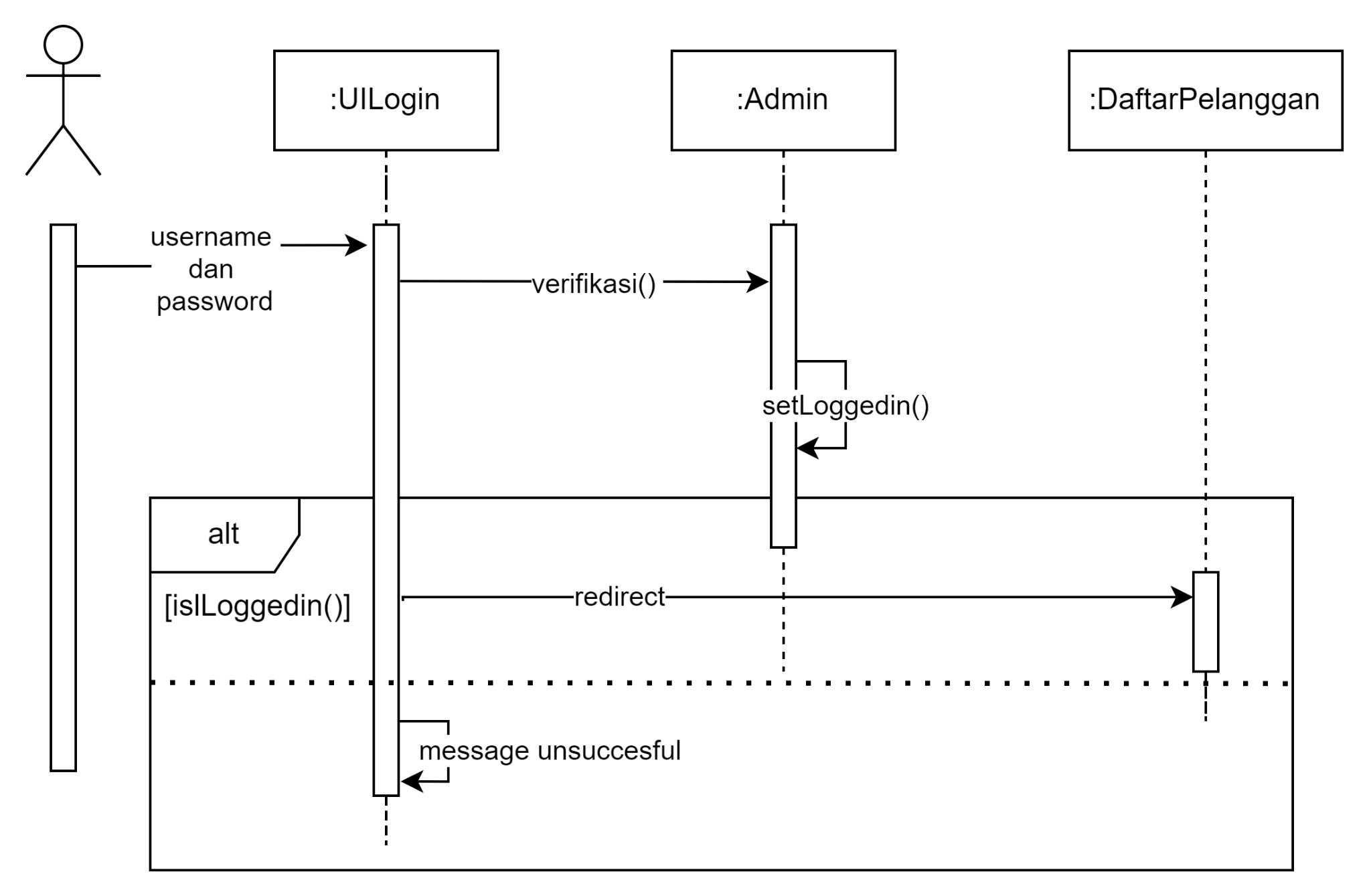
Tabel 2.4.3 Deskripsi Class Diagram Transaksi

| **Name** | Transaksi | |
| --- | --- | --- |
| **Description** | Kelas yang mengatur tentang pendataan transaksi dan perhitungan diskon dan harga akhir | |
| **Attributes** | idTransaksi, idCustomer, jumlah | |
| **Methods** | editTransaksi(): boolean | Digunakan untuk mengedit data Transaksi yang telah tersedia pada sistem |
| hitungDiskon(): integer | menghitung jumlah diskon dan menghasilkan harga akhir setelah diskon pada sebuah transaksi |
| pakaiPromo(idPromo: integer): boolean | menggunakan suatu promo pada suatu transaksi |
| validasi():Boolean | Digunakan untuk validasi *input user* dari halaman transaksi. |

Tabel 2.4.4 Deskripsi Class Diagram Promo

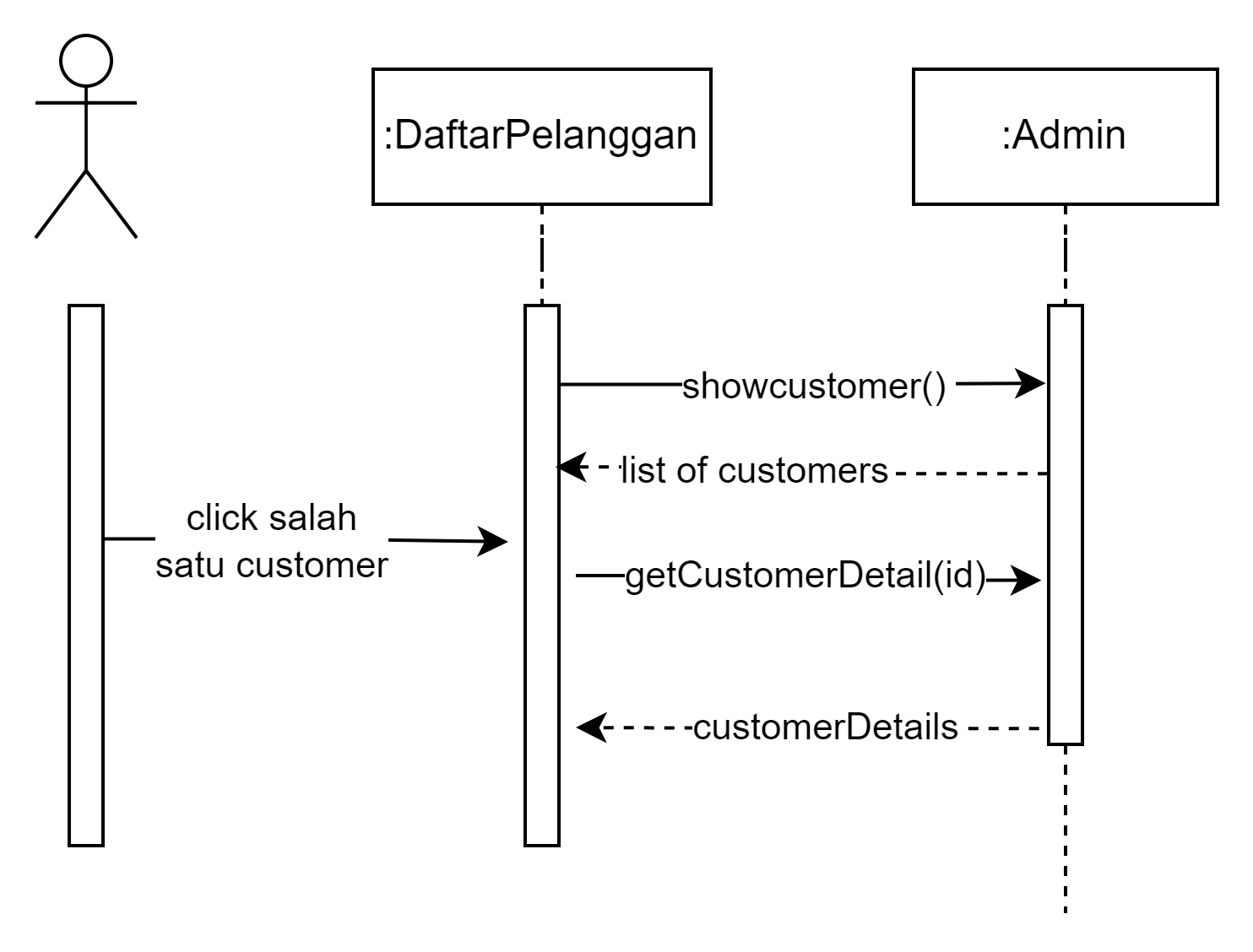
| **Name** | Promo | |
| --- | --- | --- |
| **Description** | Kelas yang mengatur tentang pendataan promo yang ada dan yang pernah ada | |
| **Attributes** | idPromo, detailPromo, jumlahTersisi, tanggalExpired | |
| **Methods** | editPromo(): boolean | Digunakan untuk mengedit data Promo yang telah tersedia pada sistem |
| pakaiPromo(): boolean | menggunakan promo pada sebuah transaksi sehingga mengurangi jumlah tersisa |

## Pemodelan Behavior



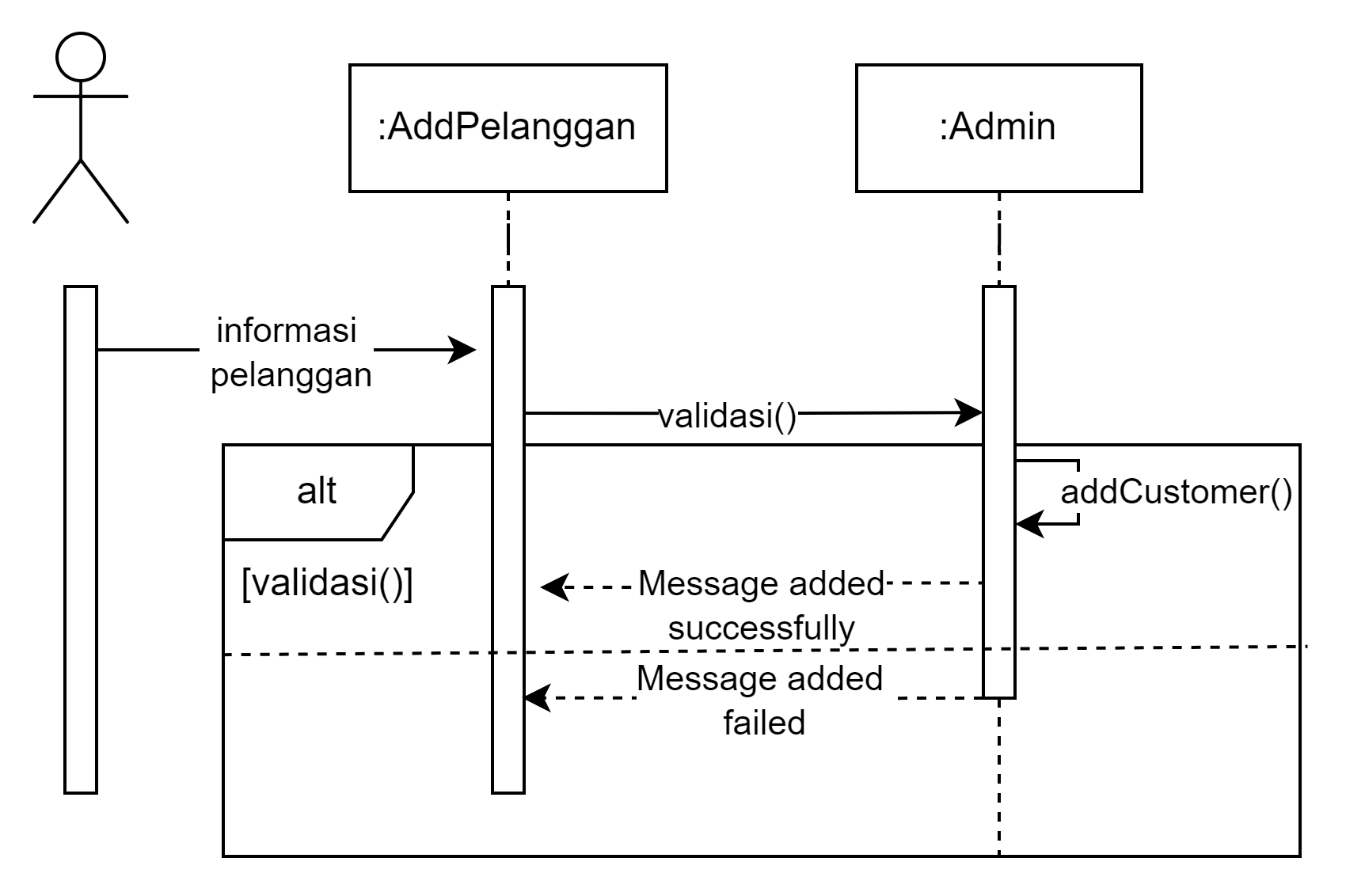
Gambar 2.5.1 Sequence Diagram Login

Terdapat 3 kelas yang terlibat untuk aksi ketika user login. Saat user input username dan password dan disubmit, maka kelas UILogin meminta fungsi validasi dari Admin. Hasil dari validasi akan mentrigger setloggedIn berupa *session* berdasarkan hasil dari validasi. Kemudian pada UILogin akan memeriksa jika bisa login menggunakan fungsi isLoggedin(). Apabila isLoggedin true, maka user akan di redirect ke halaman DaftarPelanggan. Apabila isLoggedin *false*, maka user akan menerima *message unsuccessful*.



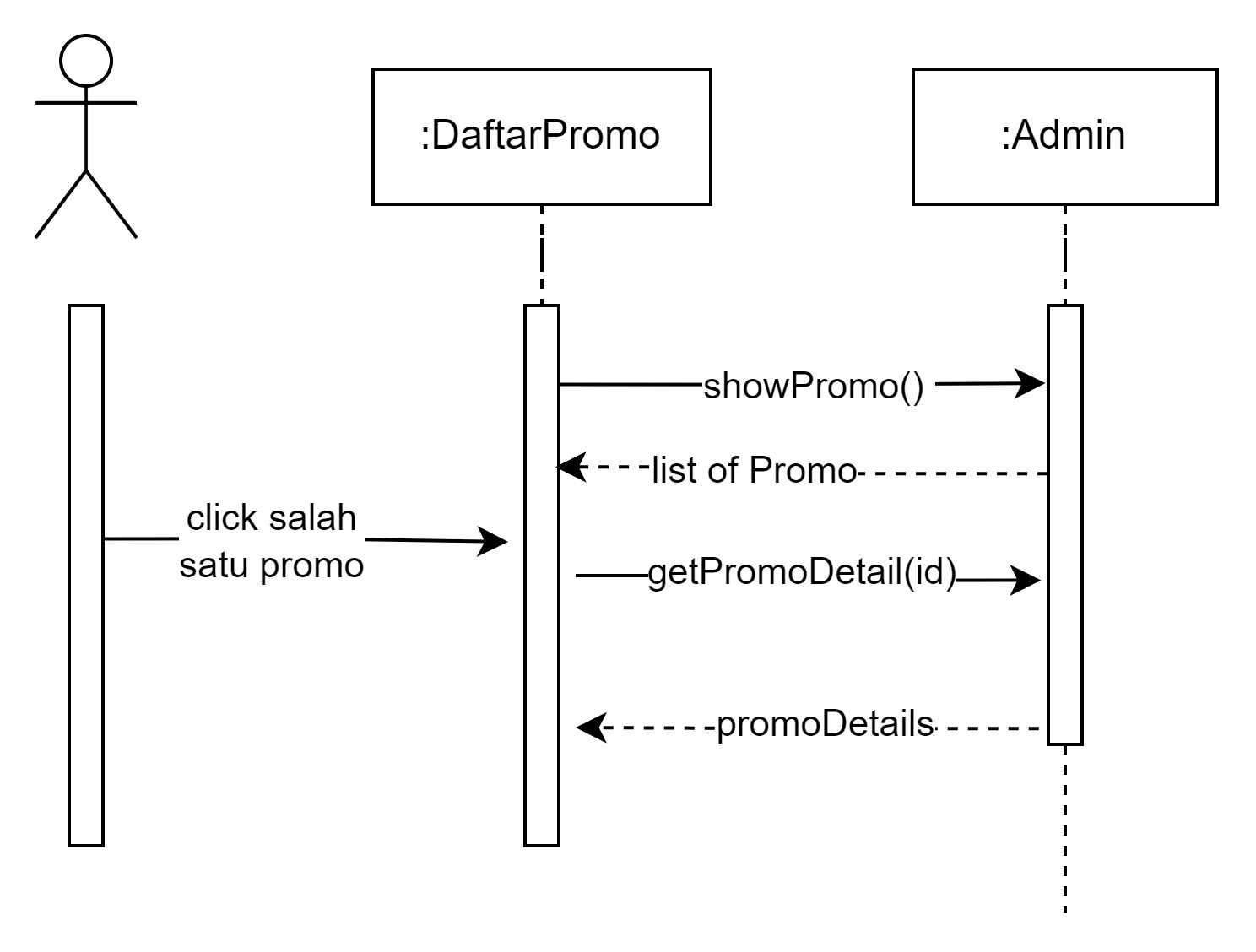
Gambar 2.5.2 Sequence Diagram Menampilkan Daftar Pelanggan

Pada halaman Daftar Pelanggan, akan dipanggil fungsi showcustomer untuk mengembalikan daftar pembeli agar ditunjukkan pada halaman Daftar Pelanggan. Apabila user menekan salah satu card dari pembeli pada halaman tersebut, maka akan mentrigger pemanggilan fungsi getCustomerDetail(id) ke Admin yang akan mengembalikan rincian dari pembeli yang ditekan (identifikasi menggunakan id pembeli).



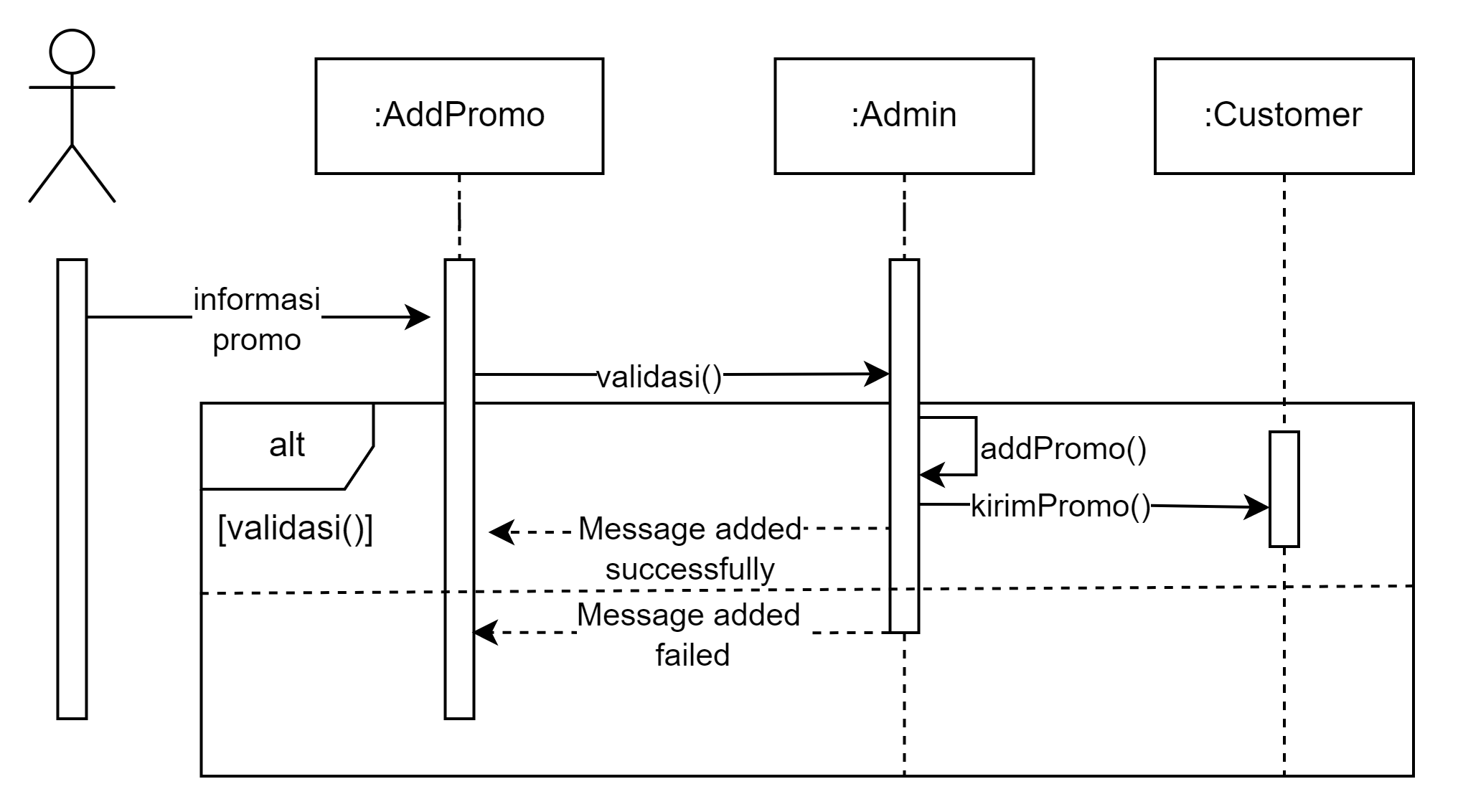
Gambar 2.5.3 Sequence Diagram Menambahkan Daftar Pelanggan

Pada AddPelanggan, akan diadakan form yang harus diisi oleh user berisi informasi pelanggan. Jika user submit, maka akan divalidasi terlebih dahulu inputnya. Apabila hasil validasi benar (tidak salah input) maka Admin memanggil fungsi addCustomer() pada dirinya yang bertugas untuk menambahkan data pembeli yang dimasukkan ke database lalu mengembalikan *message successful.* Apabila validasi salah maka akan mengembalikan *message unsuccessful.*



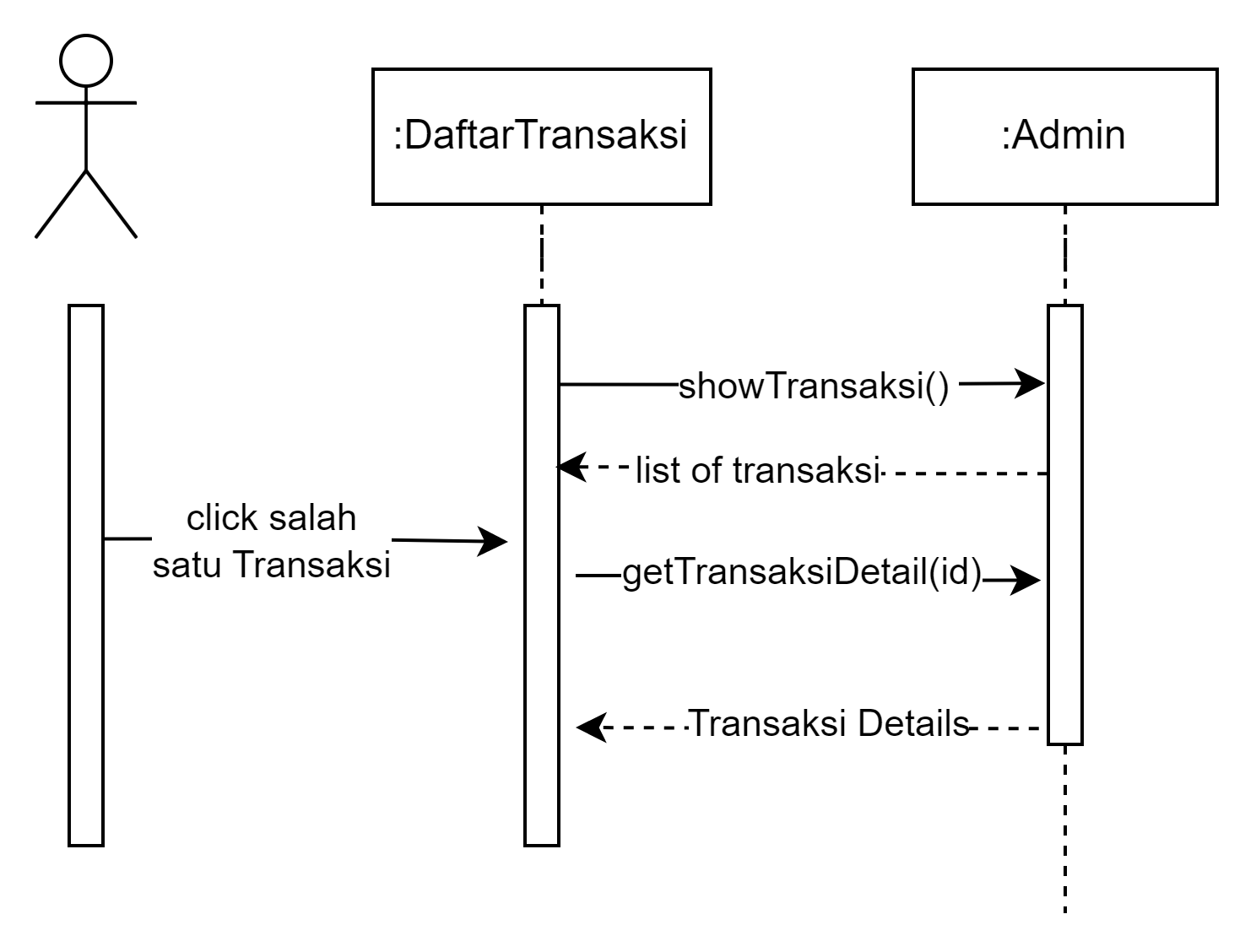
Gambar 2.5.4 Sequence Diagram Menampilkan Daftar Promosi

Pada halaman Daftar Promo, akan dipanggil fungsi showPromo() ke Admin. Kemudian admin mengembalikan daftar promo. Apabila user menekan salah satu promo dari *list*, maka halaman tersebut akan memanggil getPromoDetail(id) dengan id adalah id promo yang ditekan. Admin akan mengembalikan rincian dari promo tersebut.



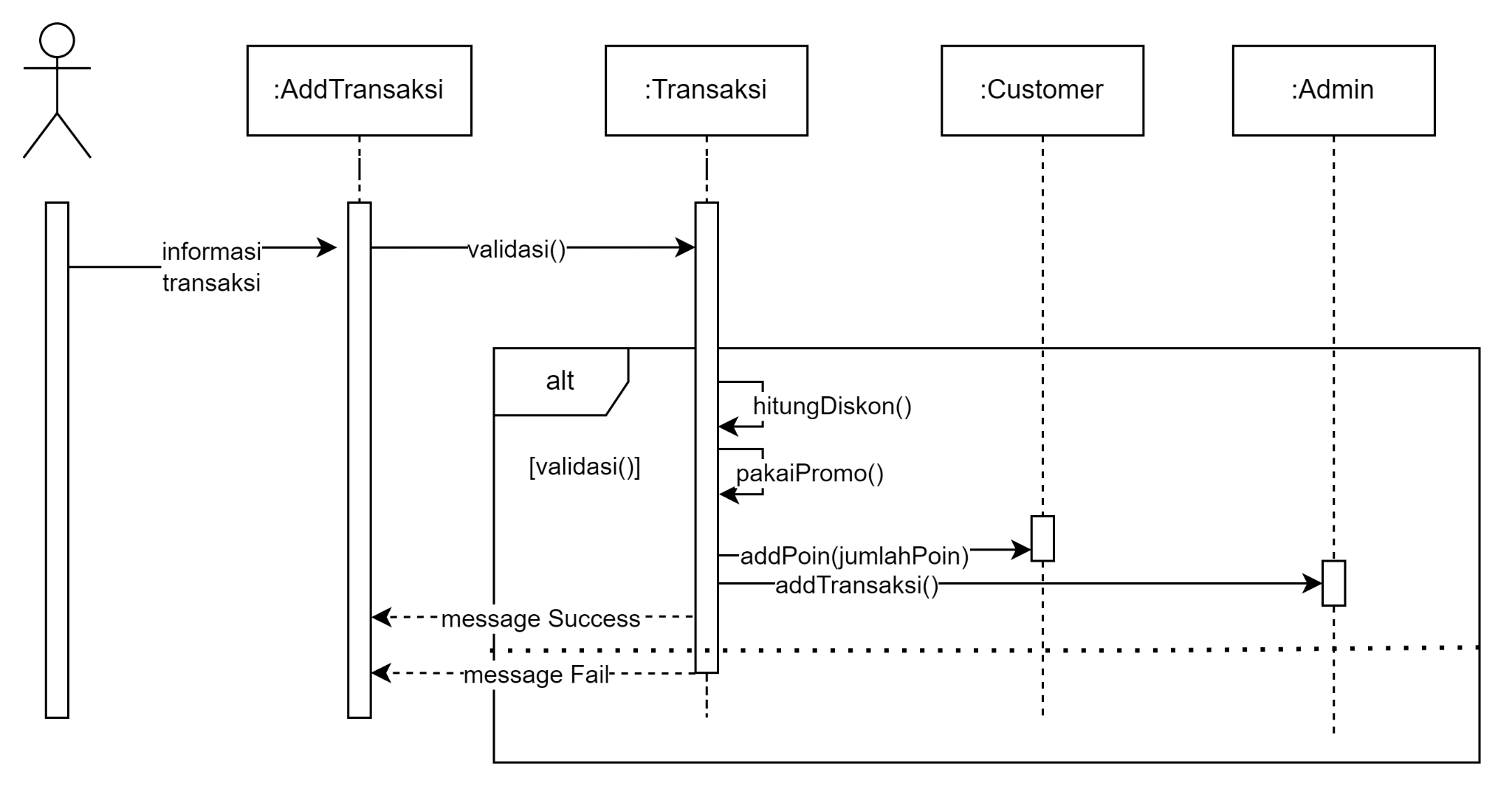
Gambar 2.5.5 Sequence Diagram Menambahkan Daftar Promosi

Pada halaman tambah promo, akan ada form yang perlu diisi oleh user. Apabila user telah mengisi informasi dan submit akan divalidasi terlebih dahulu inputnya. Apabila hasil validasi *true,* pada kelas Admin akan memanggil fungsi dirinya untuk menambahkan promo lalu akan mengirim promo jika pembeli menerima menggunakan fungsi kirimPromo dari kelas Customer. Diasumsikan penambahan selalu berhasil sehingga *message success* dikembalikan apabila validasi benar dan sebaliknya.



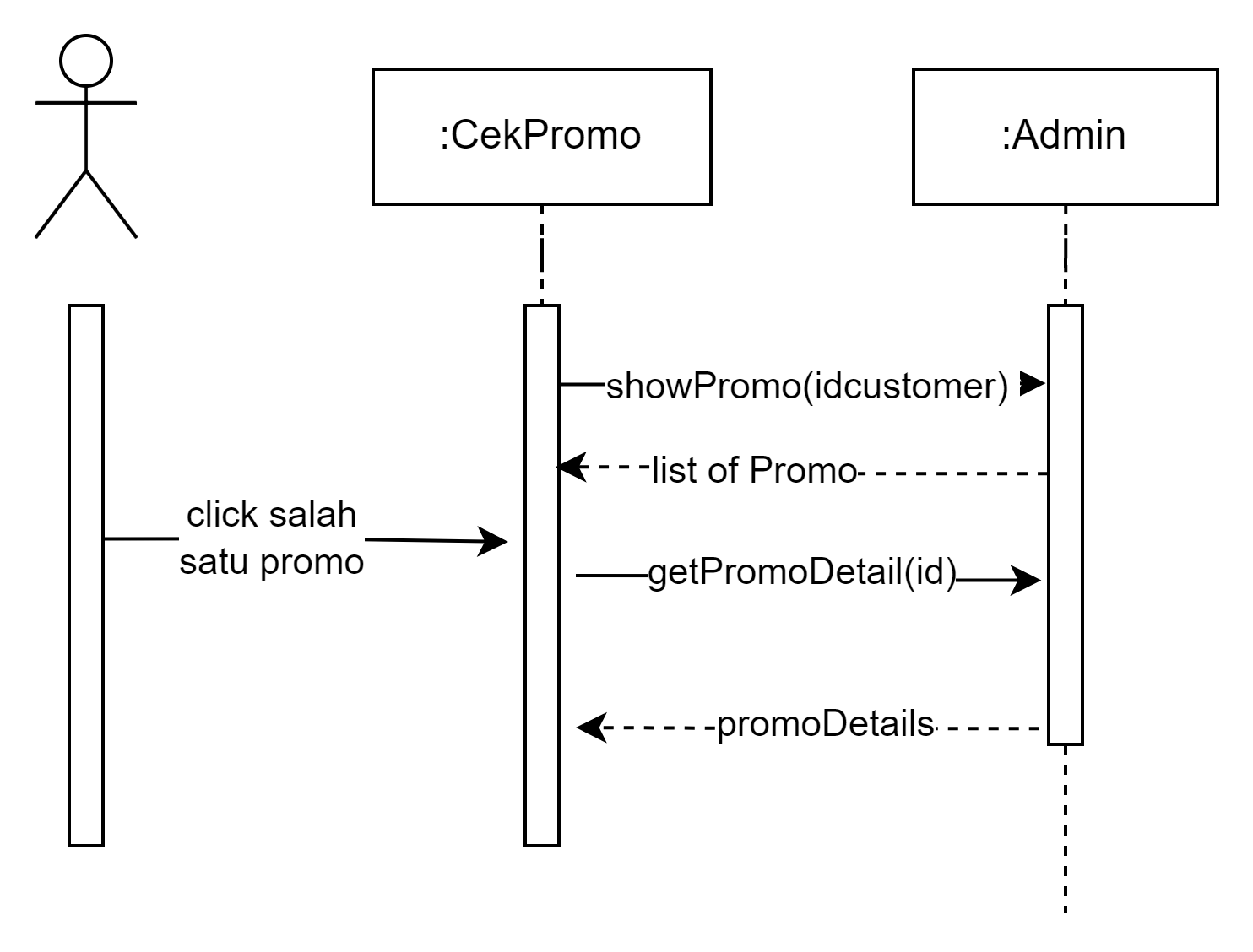
Gambar 2.5.6 Sequence Diagram Menampilkan Daftar Transaksi

Pada halaman transaksi, halaman ini akan memanggil fungsi showTransaksi ke Admin lalu mengembalikan list dari transaksi tersebut. Apabila user menekan salah satu transaksi pada *list*, maka fungsi getTransaksiDetail dengan parameter id transaksi akan dipanggil ke server untuk mengembalikan rincian dari transaksi yang ditekan.

******

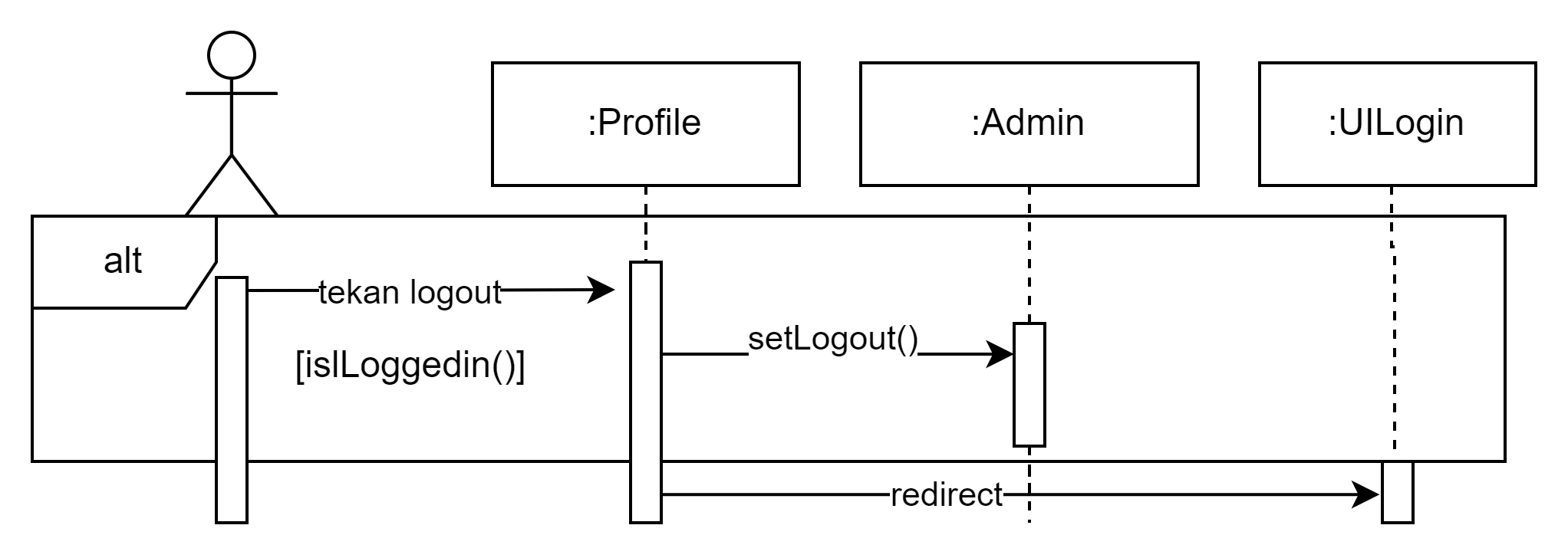
Gambar 2.5.7 Sequence Diagram Menambahkan Daftar Transaksi

Pada halaman tambah transaksi, akan ada form untuk menambahkan transaksi. apabila user mengisi informasi transaksi dan submit, akan divalidasi terlebih dahulu inputnya. Apabila hasil validasi benar, maka proses penambahan akan dilaksanakan. Data dari AddTransaksi diproses dengan menggunakan fungsi hitungDiskon, pakaiPromo, dan addPoin yang akan tetap jalan walau tidak ada input promo. Sehingga akan dihandle apabila ada dan tidak input promo. Setelah diproses datanya, maka akan dipanggil fungsi addTransaksi dari admin. Kemudian validasi pada kelas Transaksi akan mengembalikan pesan sukses apabila validasi benar dan pesan gagal apabila validasi salah.



Gambar 2.5.8 Sequence Diagram Cek Promo Customer

Pada halaman periksa Promo, akan dipanggil fungsi showPromo yang memiliki parameter dari id pembeli saat user memasukkan kode atau nama dari pelanggan. Hasil dari fungsi mengembalikan daftar promo. Apabila user menekan salah satu promo tersebut, maka akan dilakukan getPromoDetail dengan parameter id promo. Fungsi tersebut akan mengembalikan rincian dari promo yang ditekan user.



Gambar 2.5.9 Sequence Diagram Logout

Pada Profile apabila user menekan logout, akan dipanggil kelas Admin untuk prosedur setLogout untuk mengakhiri session lalu user akan diarahkan ke halaman Login.

## User Interface

1. ***Screen Mockup***

Tabel 2.6.1 *Screen Mockup* Layar *Login*

| ***Nama Layar*** | Layar *login* |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk memasukkan kredensial pelanggan |
| ***User*** | Kasir, *back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.2 *Screen Mockup* Layar Daftar Pelanggan

| ***Nama Layar*** | Layar daftar pelanggan |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk melihat daftar pelanggan beserta informasi penting lainnya, apabila mengklik tombol “Buat Pelanggan Baru” akan muncul halaman tambahkan pelanggan |
| ***User*** | Kasir, *back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.3 *Screen Mockup* Layar Tambah Pelanggan

| ***Nama Layar*** | Layar tambah pelanggan |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk menambahkan pelanggan baru ke dalam sistem |
| ***User*** | Kasir |
| ***Tampilan Layar*** |  |

Tabel 2.6.4 *Screen Mockup* Layar Daftar Promosi

| ***Nama Layar*** | Layar daftar promosi |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk melihat daftar promosi, apabila baris promosi di-klik akan muncul *modal* berisi informasi lebih lanjut terkait promosi, apabila mengklik tombol “Buat Promosi Baru” akan muncul halaman tambahkan promosi |
| ***User*** | Kasir, *back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.5 *Screen Mockup* Layar Tambah Promosi

| ***Nama Layar*** | Layar tambah promosi |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk menambahkan promosi baru ke dalam sistem |
| ***User*** | *Back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.6 *Screen Mockup* Layar Daftar Promosi Khusus

| ***Nama Layar*** | Layar daftar promosi khusus |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk melihat daftar promosi khusus, apabila baris promosi diklik akan muncul modal berisi informasi lebih lanjut, apabila mengklik tombol “Buat Promosi Baru” akan muncul halaman tambahkan promosi khusus |
| ***User*** | Kasir, *back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.7 *Screen Mockup* Layar Tambahkan Promosi Khusus

| ***Nama Layar*** | Layar tambahkan promosi khusus |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk menambahkan promosi khusus ke sistem |
| ***User*** | *Back office* |
| ***Tampilan Layar*** |  |

Tabel 2.6.8 *Screen Mockup* Layar Daftar Transaksi

| ***Nama Layar*** | Layar daftar transaksi |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk melihat daftar transaksi, apabila tombol Detail di-klik akan muncul *modal* berisi informasi lebih lanjut terkait transaksi, apabila mengklik tombol “Buat Transaksi Baru” akan muncul halaman tambahkan pelanggan |
| ***User*** | Kasir, *back office* |
| ***Tampilan Layar*** | *\* |

Tabel 2.6.9 *Screen Mockup* Layar Tambah Transaksi

| ***Nama Layar*** | Layar tambah transaksi |
| --- | --- |
| ***Deskripsi Layar*** | Layar dimanfaatkan untuk menambahkan transaksi baru ke dalam sistem |
| ***User*** | Kasir |
| ***Tampilan Layar*** |  |

1. **Pemetaan Layar**

Tabel 2.6.9 Pemetaan Layar

| ***No*** | ***Use Case / Activity*** | ***Layar*** |
| --- | --- | --- |
| 1 | Login | Layar *login* |
| 2 | Buka Daftar Pelanggan | Layar daftar pelanggan |
| 3 | Buka Daftar Promosi | Layar daftar promosi |
| 4 | Buka Daftar Promosi Khusus | Layar daftar promosi khusus |
| 5 | Tambah Data Transaksi | Layar tambah transaksi |
| 6 | Tambah Data Pelanggan | Layar tambah pelanggan |
| 7 | Tambah Data Promosi | Layar tambah promosi |
| 8 | Tambahkan Data Promosi Khusus | Layar tambah promosi khusus |
| 9 | Buka Daftar Transaksi | Layar daftar promosi |
| 10 | Logout | * Layar daftar pelanggan * Layar daftar promosi * Layar cek promosi * Layar tambah transaksi * Layar tambah pelanggan * Layar tambah promosi * Layar daftar promosi |

# Bab III Rencana Pengujian Sistem Informasi

1. **Skenario Uji Login**

SRS-ID : FR-01

Nama kebutuhan fungsional : Pengguna dapat melakukan *login* ke aplikasi

Deskripsi : Pengguna *login* ke aplikasi sehingga pengguna dapat mengakses fitur yang ada di aplikasi.

Data input : Data kredensial, yaitu username/e-mail dan password, dari pengguna.

Output yang diharapkan : Akun pengguna berhasil *login* ke aplikasi

Langkah Pengujian : Pada layar *login*, memasukkan username/e-mail dan password yang sesuai ke kolom yang bersangkutan kemudian menekan tombol “Masuk”.

1. **Skenario Uji Buka Daftar Pelanggan**

SRS-ID : FR-02

Nama kebutuhan fungsional : Pengguna dapat melihat daftar & detail pelanggan

Deskripsi : Pengguna dapat membuka dan melihat daftar dari pelanggan yang terdaftar ke dalam *membership* serta melihat detail dari setiap pelanggan yang terdaftar.

Data input : -

Output yang diharapkan : Ditampilkan daftar dari semua pelanggan yang terdaftar ke dalam *membership*. Pada tiap baris pelanggan, ditampilkan data pelanggan yang berisikan nama, nomor telepon, email, dan jumlah poin pelanggan terkait. Selain itu, ditampilkan detail dari pelanggan yang berisikan nama, nomor telepon, e-mail, jumlah poin, serta promo yang dimiliki pelanggan tersebut.

Langkah Pengujian : Masuk ke aplikasi, kemudian menekan tombol “Daftar Pelanggan” yang berada di navbar bagian kiri aplikasi. Kemudian salah satu baris pelanggan ditekan. Setelah pop-up “Detail Pelanggan” muncul, menekan tombol “Tutup”

1. **Skenario Uji Buka Daftar Promo**

SRS-ID : FR-03

Nama kebutuhan fungsional : Pengguna dapat melihat daftar & detail promo

Deskripsi : Pengguna dapat membuka dan melihat daftar dari promosi yang ada serta melihat detail dari promo yang ada.

Data input : -

Output yang diharapkan : Ditampilkan daftar dari semua promo yang ada saat itu. Pada tiap baris promo, ditampilkan data promo yang berisikan nama, tanggal rilis, dan tanggal selesai dari promo terkait. Selain itu, ditampilkan detail dari promo yang berisikan nama, masa berlaku, serta deskripsi dari promo tersebut.

Langkah Pengujian : Masuk ke aplikasi, kemudian menekan tombol “Daftar Promosi” yang berada di navbar bagian kiri aplikasi. Kemudian salah satu baris promo ditekan. Setelah pop-up “Detail Promosi” muncul, menekan tombol “Tutup”

1. **Skenario Uji Buka Daftar Promo Khusus**

SRS-ID : FR-04

Nama kebutuhan fungsional : Pengguna dapat melihat daftar & detail promokhusus

Deskripsi : Pengguna dapat membuka dan melihat daftar dari promosi yang ada serta melihat detail dari promo yang ada.

Data input : -

Output yang diharapkan : Ditampilkan daftar dari semua promo khusus yang ada saat itu. Pada tiap baris promo, ditampilkan data promo yang berisikan nama, kode, pemilik promo, deskripsi singkat

Langkah Pengujian : Masuk ke aplikasi, kemudian menekan tombol “Daftar Promosi” yang berada di navbar bagian kiri aplikasi. Kemudian salah satu baris promo ditekan. Setelah pop-up “Detail Promosi” muncul, menekan tombol “Tutup”

1. **Skenario Uji Tambah Data Transaksi**

SRS-ID : FR-05

Nama kebutuhan fungsional : Pengguna dapat menambahkan data transaksi

Deskripsi : Pengguna menambahkan data transaksi yang baru ke sistem

Data input : Kode transaksi, tanggal transaksi, pembeli, kode promo yang dipakai saat transaksi, nominal dari transaksi.

Output yang diharapkan : Transaksi beserta data yang dimasukkan muncul pada daftar transaksi. Data transaksi pada daftar transaksi sesuai dengan data masukan.

Langkah Pengujian : *Login* ke aplikasi sebagai pengguna dengan *role* kasir. Masuk ke halaman daftar transaksi kemudian menekan tombol “Tambahkan Transaksi” pada halaman. Memasukkan data yang sesuai ke kolom yang terkait kemudian menekan tombol “Tambahkan”. Menekan tombol “Daftar Transaksi” kemudian memeriksa transaksi yang baru saja dimasukkan.

1. **Skenario Uji Tambah Data Pelanggan**

SRS-ID : FR-06

Nama kebutuhan fungsional : Pengguna dapat menambahkan data pelanggan

Deskripsi : Pengguna menambahkan data pelanggan yang baru ke sistem

Data input : Nama, nomor telepon, dan e-mail dari pelanggan

Output yang diharapkan : Pelanggan beserta data yang dimasukkan muncul pada daftar pelanggan. Data pelanggan pada daftar pelanggan sesuai dengan masukan data.

Langkah Pengujian : *Login* ke aplikasi sebagai pengguna dengan *role* kasir. Membuka halaman daftar pelanggan kemudian menekan tombol “Tambahkan Pelanggan” pada halaman tersebut. Memasukkan data yang sesuai ke kolom yang terkait kemudian menekan tombol “Tambahkan”. Menekan tombol “Daftar Pelanggan” kemudian memeriksa pelanggan yang baru saja dimasukkan.

1. **Skenario Uji Tambah Data Promosi**

SRS-ID : FR-07

Nama kebutuhan fungsional : Pengguna dapat menambahkan data promosi

Deskripsi : Pengguna menambahkan data promosi yang baru ke sistem

Data input : Nama, kode, tanggal mulai, tanggal selesai, stok, dan deskripsi dari promo.

Output yang diharapkan : Promosi beserta data yang dimasukkan muncul pada daftar promosi. Data promosi pada daftar promosi sesuai dengan masukan data.

Langkah Pengujian : *Login* ke aplikasi sebagai pengguna dengan *role* *back-office*. Membuka halaman Daftar Promosi kemudian menekan tombol “Tambahkan Promosi” pada halaman. Memasukkan data yang sesuai ke kolom yang terkait kemudian menekan tombol “Tambahkan”. Menekan tombol “Daftar Promosi” kemudian memeriksa promosi yang baru saja dimasukkan.

1. **Skenario Uji Tambah Data Promosi Khusus**

SRS-ID : FR-07

Nama kebutuhan fungsional : Pengguna dapat menambahkan data promosi khusus

Deskripsi : Pengguna menambahkan data promosi yang baru ke sistem

Data input : Nama, kode, pemilik promosi, deskripsi

Output yang diharapkan : Promosi beserta data yang dimasukkan muncul pada daftar promosi. Data promosi pada daftar promosi khusus sesuai dengan masukan data.

Langkah Pengujian : *Login* ke aplikasi sebagai pengguna dengan *role* *back-office*. Masuk ke halaman “Daftar Promosi Khusus”. Menekan tombol “Tambahkan Promosi” pada halaman tersebut. Memasukkan data yang sesuai ke kolom yang terkait kemudian menekan tombol “Tambahkan”. Menekan tombol “Daftar Promosi Khusus” kemudian memeriksa promosi yang baru saja dimasukkan.

1. **Skenario Uji Buka Daftar Transaksi**

SRS-ID : FR-09

Nama kebutuhan fungsional : Pengguna dapat melihat daftar & detail transaksi

Deskripsi : Pengguna dapat membuka dan melihat daftar dari transaksi yang sudah terdata serta melihat detail dari transaksi yang sudah terdata.

Data input : -

Output yang diharapkan : Ditampilkan daftar dari semua transaksi yang terdata. Pada tiap baris transaksi, ditampilkan data transaksi yang berisikan nama transaksi, tanggal, dan pembeli dari transaksi terkait. Selain itu, ditampilkan detail dari transaksi yang berisikan kode, nama, tanggal, pembeli, kode promo yang digunakan, nominal, serta poin yang terakumulasi dari transaksi yang terkait.

Langkah Pengujian : *Login* ke aplikasi sebagai pengguna dengan *role* *back-office*. Kemudian menekan tombol “Daftar Transaksi” yang berada di navbar bagian kiri aplikasi. Kemudian tombol “Detail” yang berada pada salah satu baris transaksi ditekan. Setelah pop-up “Detail Transaksi” muncul, menekan tombol “Tutup”

1. **Skenario Uji Logout**

SRS-ID : FR-10

Nama kebutuhan fungsional : Pengguna dapat melakukan *logout* dari aplikasi

Deskripsi : Pengguna *logout* dari akun pengguna di aplikasi sehingga pengguna tidak memiliki akses ke fitur yang ada di aplikasi kecuali fitur *login.*

Data input : -

Output yang diharapkan : Akun pengguna di-*logout* dan pengguna dibawa ke layar *login*

Langkah Pengujian : Masuk ke aplikasi, kemudian pada layar apapun, kecuali layar *login*, tekan tombol *“*Keluar*”* yang berada di bagian kiri bawah layar.

# Bab IV Kesimpulan dan Saran

## Kesimpulan

Kebutuhan sistem informasi Brewspace secara garis besar dikategorisasi menjadi tiga, yakni hubungan perusahaan dengan pelanggan, sistem koordinasi *outlet*, dan keperluan migrasi sistem. Berdasarkan analisis masalah, disimpulkan bahwa hubungan perusahaan dengan pelanggan merupakan masalah yang masuk dalam kategori *urgent*, memiliki *gap* besar, dan berpeluang krusial untuk menjamin ketercapaian target perusahaan.

Beanfo merupakan usulan solusi berbasis *Customer Relationship Management* yang menawarkan solusi untuk meningkatkan loyalitas pelanggan. Secara garis besar, Beanfo mencakup kontrol pelanggan, promosi, dan transaksi yang kemudian dijadikan basis informasi personalisasi insentif sesuai dengan kategorisasi pelanggan yang telah ditentukan. Pengembangan berada pada level *local* dengan kemungkinan pengembangan pada *deployment* pada skala yang besar.

Perwujudan sistem informasi ini diharapkan menjadi solusi dari kebutuhan Brewspace untuk membangun hubungan antara pelanggan dan perusahaan yang lebih baik dan menghilangkan keberadaan *gap* terkait ketiadaan *software* untuk menjawab permasalahan terkait.

## Saran

Saran dari hasil analisis serta pembuatan dokumen-dokumen yang dibutuhkan untuk sistem informasi Brewspace ialah melakukan *interview* kembali untuk mengkonfirmasi kebutuhan atau *requirements* jika hasil analisis yang ditetapkan sudah tepat dan dibutuhkan. Kemudian, saat mengimplementasikan sistem informasi, dilakukan validasi checklist requirement yang telah dibuat agar tidak meleset dan sesuai dengan rencana (*controlling*) untuk kedepan.

Selain itu, ketika sudah mengimplementasikan sistem informasi nanti, karyawan dari Brewspace sendiri disarankan untuk diberikan pelatihan dari pembuat sistem informasi agar dapat digunakan dengan baik.

Terakhir, dalam pengerjaan dokumen, disarankan untuk merapikan terlebih dahulu *template* dokumen yang diberikan agar pengerjaan lebih mudah dan efisien.