

SKPL-Friperie

## **SPESIFIKASI KEBUTUHAN SISTEM**

**Friperie**

**(E-Commerce Thrift Shop Friperie)**

untuk:

Universitas Telkom

Dipersiapkan oleh:

Kelompok 9

Muhammad Alfi Syahri Nasution / 1301190324

Alifia Shafira / 1301194102

Serly Setyani / 1301190225

Laura Imanuela Mustamu / 1301194027

Rania Az Zahra / 1301194349

Program Studi S1 Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

<b>Program Studi S1 Teknik Informatika - Fakultas Informatika</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
	<i><b>SKPL-Friperie</b></i>		<b>26</b>
	<b>Revisi</b>	<i><b>1</b></i>	<i><b>Tgl: 19/12/2020</b></i>

# Daftar Perubahan

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX	-	A	B	C	D	E	F	G
TGL								
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
18	B B B C D		

# Daftar Isi

<b>1. Pendahuluan</b>	<b>7</b>
1.1 Tujuan Penulisan Dokumen	7
1.2 Konvensi Dokumen	7
1.3 Cakupan Produk	8
1.4 Referensi	8
<b>2. Overall Description</b>	<b>8</b>
2.1 Perspektif Produk	8
2.2 Fungsi Produk	8
2.3 Kelas dan Karakteristik Pengguna	9
2.4 Lingkungan Operasi	10
2.5 Batasan Perancangan dan Implementasi	11
2.6 Dokumentasi Pengguna	11
2.7 Asumsi dan Dependensi	11
<b>3. Requirements Antarmuka Eksternal</b>	<b>11</b>
3.1 Antarmuka Pengguna	11
3.2 Antarmuka Perangkat Keras	11
3.3 Antarmuka Perangkat Lunak	11
3.4 Antarmuka Komunikasi	12
<b>4. Fitur Sistem</b>	<b>12</b>
4.1 Registrasi dan login	12
4.1.1 Deskripsi:	12
4.1.2 Trigger	12
4.1.3 Input	12
4.1.4 Output	12
4.1.5 Skenario Utama	13
4.1.5.1 Prakondisi	
4.1.5.2 Pascakondisi	
4.1.5.3 Langkah-langkah	13
4.1.6 Skenario eksepsional 1	13
4.1.6.1 Prakondis	13
4.1.6.2 Pascakondisi	13
4.1.6.3 Langkah-langkah	13
4.2 Shopping Cart	13
4.2.1 Deskripsi	13
4.2.2 Trigger	14
4.2.3 Input	14

4.2.4 Output	14
4.2.5 Skenario Utama	14
4.3 Wishlist	14
4.3.1 Deskripsi	14
4.3.2 Trigger	15
4.3.3 Input	15
4.3.4 Output	15
4.3.5 Skenario Utama	15
4.4 Kolom Pencarian	15
4.4.1 Deskripsi	15
4.4.2 Trigger	15
4.4.3 Input	15
4.4.4 Output	16
4.4.5 Skenario Utama	16
4.5 Payment Gateway	16
4.5.1 Deskripsi	16
4.5.2 Trigger	16
4.5.3 Input	16
4.5.4 Output	17
4.5.5 Skenario Utama	17
4.6 Informasi Produk Baru	17
4.6.1 Deskripsi	17
4.6.2 Trigger	17
4.6.3 Input	17
4.6.4 Output	18
4.6.5 Skenario Utama	18
4.7 Halaman Promo	18
4.7.1 Deskripsi	18
4.7.2 Trigger	18
4.7.3 Input	18
4.7.4 Output	19
4.7.5 Skenario Utama	19
4.8 Related Items	19
4.8.1 Deskripsi	19
4.8.2 Trigger	19
4.8.3 Input	20
4.8.4 Output	20
4.8.5 Skenario Utama	20
4.9 Kontak antara Seller dan Buyer	20
4.9.1 Deskripsi	20

4.9.2 Trigger	20
4.9.3 Input	20
4.9.4 Output	21
4.9.5 Skenario Utama	21
4.10 Review Produk	21
4.10.1 Deskripsi	21
4.10.2 Trigger	21
4.10.3 Input	21
4.10.4 Output	22
4.10.5 Skenario Utama	22
4.11 Pengembalian Barang	22
4.11.1 Deskripsi	22
4.11.2 Trigger	22
4.11.3 Input	22
4.11.4 Output	22
4.11.5 Skenario Utama	23
4.12 Pembatalan Pesanan	23
4.12.1 Deskripsi	23
4.12.2 Trigger	23
4.12.3 Input	23
4.12.4 Output	24
4.12.5 Skenario Utama	24
<b>5. Requirements Non Fungsional</b>	<b>24</b>
5.1 Atribut Kualitas	24
5.1.1 Pemeliharaan (Maintainability)	24
5.1.2 Keamanan (Security)	24
5.1.3 Dapat diandalkan (Dependability)	24
5.2 Requirements Legal	25
<b>Lampiran A: Daftar Kata-Kata Sukar</b>	<b>25</b>
<b>Lampiran B: Analysis Models</b>	<b>25</b>

# 1. Pendahuluan

## 1.1 Tujuan Penulisan Dokumen

Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan sebuah dokumen yang berisi spesifikasi lengkap dari apa yang dapat dilakukan oleh perangkat lunak, tanpa menjelaskan bagaimana hal tersebut dikerjakan oleh perangkat lunak. Dokumen SKPL ini dibuat untuk menjelaskan spesifikasi kebutuhan perangkat lunak pada aplikasi Friperie.

E-commerce berasal dari bahasa Inggris, yang merupakan electronic commerce atau dalam bahasa Indonesia disebut perdagangan elektronik. Dan sebagaimana perdagangan yang dilakukan secara langsung atau face to face tetapi pada e-commerce tidak perlu dilakukan dengan face to face. Dalam e commerce juga meliputi proses promosi, pembelian, dan pemasaran produk. Yang berbeda adalah pada sistem berdagang yang digunakan, yaitu melalui media elektronik atau internet.

Thrift shop berasal dari kata thrift yang artinya penghematan atau cara dalam penggunaan uang dengan menghindari pemborosan dan store berarti toko. Thrift shop dapat diartikan sebagai toko penghematan. Toko yang menjual barang-barang bekas dari luar maupun dalam negeri. Sejak tahun 2013, perdagangan barang bekas mulai masuk ke Indonesia, dimulai dari barang langka hingga barang dengan brand yang terkenal. Akan tetapi, dalam bisnis thrift shop, apapun barang serta nama brand nya, selama barang bekas tersebut ber kondisi baik dan layak untuk dipakai dapat diperjualbelikan.

Pada saat ini, toko online yang menjual barang bekas atau biasa disebut thrift shop mulai terkenal. Akan tetapi, sangat sulit dalam mencari thrift shop yang terjamin dan juga memiliki barang yang bagus. Ada Pula thrift shop yang abal-abal bahkan menipu.

## 1.2 Konvensi Dokumen

1. SKPL adalah Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggris-nya sering juga disebut sebagai software requirements specification (SRS), dan merupakan spesifikasi perangkat lunak yang akan dikembangkan.
2. SKPL-FRIPERIE.K-xxx adalah kode yang digunakan untuk merepresentasikan kebutuhan (requirement) pada FRIPERIE, dengan FRIPERIE merupakan kode perangkat lunak, FRIPERIE adalah kode fase, dan xxx adalah digit/nomor kebutuhan (requirement).
3. DFD adalah data flow program, diagram dan notasi yang digunakan untuk menunjukkan aliran data pada perangkat lunak.

## 1.3 Cakupan Produk

FRIPERIE merupakan aplikasi e-commerce yang khusus menjual barang-barang bekas terutama barang branded yang masih layak pakai dengan harga yang murah. Tujuan dari aplikasi FRIPERIE ini yaitu sebagai sarana jual-beli barang bekas layak pakai. FRIPERIE kami angkat dari bahasa Perancis yang artinya toko barang bekas. FRIPERIE memberikan kemudahan pada pengguna. Pada buyer, buyer dapat dengan mudah mencari barang yang dicari ataupun toko yang dicari dengan mudah dan toko yang ada terpercaya.

Aplikasi ini memberikan beberapa fitur yang sangat memudahkan *seller* untuk menjual barang bekasnya dan *buyer* untuk membeli barang yang tersedia di aplikasi FRIPERIE yaitu fitur chat antara *seller* dan *buyer*, fitur pembatalan pesanan dengan persetujuan dari *seller*, fitur kebijakan pengembalian barang dengan persetujuan dari *seller*, dan sebagainya. FRIPERIE juga sering mengadakan promo yang terletak pada halaman promo.

## 1.4 Referensi

Dokumen ini memerlukan beberapa referensi yaitu berasal dari:

1. Survei berupa *questionnaire* di Google Form
2. Musliman, Septiana (2020). BISNIS THRIFT STORE, PELUANG BISNIS YANG BESAR?. <https://www.jurnal.id/id/blog/bisnis-thrift-store/> (Akses 19 Desember 2020)

## 2. Overall Description

### 2.1 Perspektif Produk

Friperie ini merupakan perangkat lunak yang menghubungkan sistem jual beli *online* dan *thrift shop* yang saat ini sedang marak. Perangkat lunak ini adalah *thrift shop* secara *online* yang terpercaya dan juga berkualitas, sehingga pengguna akan lebih mudah jika ingin membeli sesuatu yang berkualitas dan juga bermerek tetapi dengan harga yang tidak mahal.

### 2.2 Fungsi Produk

Adapun fungsi-fungsi dari perangkat lunak ini adalah :

1. Registrasi/*Sign-up*. Pengguna yang baru akan menggunakan aplikasi dapat membuat akun. (SKPL-Friperie.K-001)
2. Login. Pengguna dapat masuk kedalam aplikasi setelah registrasi dengan menggunakan *username* dan *password*. (SKPL-FRIPERIE.K-002)
3. Membeli barang. *Buyer* dapat membeli barang yang dijual oleh *seller* pada



- aplikasi.(SKPL-FRIPERIE.K-003)
4. Menjual barang. *Seller* dapat menjual barang yang ingin dijual pada aplikasi.(SKPL-FRIPERIE.K-004)
  5. Mencari barang. *Buyer* dapat mencari barang yang akan dibeli pada kolom pencarian yang disediakan.(SKPL-FRIPERIE.K-005)
  6. Melakukan transaksi. Setelah membeli barang, *buyer* dapat membayar/melakukan transaksi dengan layanan yang sudah disediakan oleh sistem aplikasi kepada *seller*.(SKPL-FRIPERIE.K-006)
  7. Kontak antara *buyer* dan *seller*. Dengan adanya fitur untuk mengirim pesan, maka *seller* maupun *buyer* dapat saling menghubungi.(SKPL-FRIPERIE.K-007)
  8. Memberi *feedback*. *Buyer* yang sudah membeli barang dapat memberikan *feedback* kepada toko mengenai barang yang dibeli dengan memberi komentar maupun rating. (SKPL-FRIPERIE.K-008)
  9. Mengupdate mengenai barang. Mengupdate mengenai barang yang dijual seperti jumlah barang yang tersedia. (SKPL-FRIPERIE.K-009)
  10. Memberi keluhan mengenai aplikasi. *User* seperti *buyer* dan *seller* dapat memberi keluhan mengenai aplikasi. (SKPL-FRIPERIE.K-0010)
  11. Membatalkan pesanan. *Buyer* dapat membatalkan pesanan dengan alasan yang jelas. (SKPL-FRIPERIE.K-0011)
  12. Mengembalikan barang. *Buyer* dapat mengembalikan barang yang sudah diterima jika tidak sesuai dengan apa yang dipesan. (SKPL-FRIPERIE.K-0012)

## 2.3 Kelas dan Karakteristik Pengguna

Perangkat lunak FRIPERIE ini merupakan perangkat lunak yang digunakan untuk jual beli. Sistem ini berkaitan dengan beberapa entitas luar, yaitu admin, pembeli(*buyer*), dan penjual(*seller*). Hal – hal yang dilakukan oleh entitas – entitas tersebut adalah :

Kategori Pengguna	Tugas	Hak Akses ke Aplikasi
<i>Buyer</i>	Melihat barang, melakukan pembelian pada aplikasi.	(SKPL-FRIPERIE.K-001) (SKPL-FRIPERIE.K-002) (SKPL-FRIPERIE.K-003) (SKPL-FRIPERIE.K-005)

		(SKPL-FRIPERIE.K-006) (SKPL-FRIPERIE.K-007) (SKPL-FRIPERIE.K-008) (SKPL-FRIPERIE.K-010) (SKPL-FRIPERIE.K-011) (SKPL-FRIPERIE.K-012)
<i>Seller</i>	Menjual barang, melihat dan menanggapi pesanan dari <i>buyer</i> .	(SKPL-FRIPERIE.K-001) (SKPL-FRIPERIE.K-002) (SKPL-FRIPERIE.K-004) (SKPL-FRIPERIE.K-005) (SKPL-FRIPERIE.K-006) (SKPL-FRIPERIE.K-007) (SKPL-FRIPERIE.K-009) (SKPL-FRIPERIE.K-010)
<i>Administrator</i>	Memantau dan memaintenance aplikasi.	(SKPL-FRIPERIE.K-009)

## 2.4 Lingkungan Operasi

Sistem Operasi : Microsoft Windows 7/8/8.1/10, Android

## 2.5 Batasan Perancangan dan Implementasi

Adapun batasan-batasan dalam menggunakan aplikasi ini adalah :

1. Aplikasi dapat dijalankan saat ada terhubung dengan internet
2. Aplikasi hanya dapat dijalankan di sistem operasi windows dan android
3. Kapasitas memori yang dibutuhkan lumayan besar
4. Antarmuka berupa tampilan yang menarik

## 2.6 Dokumentasi Pengguna

1. Akan dilakukan promosi mengenai aplikasi agar di lebih dikenali oleh pengguna
2. Terdapat fitur untuk bertanya maupun memberikan keluhan mengenai aplikasi

## 2.7 Asumsi dan Dependensi

Untuk mengakses aplikasi pengguna harus terhubung ke internet dan juga memiliki *email* sehingga bisa membuat akun pada aplikasi.

# 3. Requirements Antarmuka Eksternal

## 3.1 Antarmuka Pengguna

Sistem Friperie ini menggunakan antarmuka berbasis windows dan android serta pengguna menggunakan keyboard dan mouse. Pengguna akan berinteraksi menggunakan keyboard dan mouse dengan memberi sebuah perintah kemudian keluaran dari perintah tersebut dapat dilihat langsung di monitor *PC/Smartphone*.

## 3.2 Antarmuka Perangkat Keras

Kebutuhan perangkat keras yang dibutuhkan adalah seperti *PC, mouse, keyboard, laptop*, dan *smartphone*.

## 3.3 Antarmuka Perangkat Lunak

Untuk mengakses aplikasi Friperie ini dapat menggunakan Windows dan Android.

### 3.4 Antarmuka Komunikasi

Sistem Friperie merupakan sistem yang terhubung ke jaringan komputer dengan menggunakan desktop, sehingga pelanggan bisa dilayani oleh lebih dari satu pegawai dengan menggunakan database yang sama. Dan pihak yang bertugas(admin) bisa memonitor system lewat jaringan komputer. Dengan demikian aliran informasi menjadi lebih lancar.

## 4. Fitur Sistem

### 4.1 Registrasi dan login

#### 4.1.1 *Deskripsi:*

*User* atau pengguna dapat melakukan registrasi jika belum memiliki akun, dan pengguna yang sudah memiliki akun atau sudah pernah melakukan registrasi bisa langsung untuk login dan masuk ke dalam aplikasi.

#### 4.1.2 *Trigger:*

Saat registrasi, pengguna akan mengisi data sesuai dengan yang diminta oleh sistem, kemudian memilih “*Create an account*” dan langsung akun akan dibuat. Kemudian saat login, pengguna akan mengisi *username* dan *password* kemudian memilih “*sign-in*” dan langsung masuk ke dalam aplikasi.

#### 4.1.3 *Input:*

Data yang dimasukan untuk registrasi adalah nama, *email*, *username*, dan *password*. Kemudian data yang dimasukan untuk login adalah *username* dan *password*.

#### 4.1.4 *Output:*

*Output* yang dihasilkan oleh fitur ini adalah pengguna akan langsung masuk ke dalam aplikasi yang akan langsung terlihat di monitor PC maupun *smartphone*.

#### **4.1.5 Skenario Utama:**

*4.1.5.1 Prakondisi:* Akan muncul kolom-kolom untuk mengisi nama, *email*, *username*, dan *password* yang harus diisi oleh pengguna.

*4.1.5.2 Pascakondisi:* Setelah mengisi kolom-kolom tersebut dengan data pengguna dan menekan “*create an account*”, maka akun baru pengguna akan langsung dapat digunakan dan bisa langsung masuk ke aplikasi

##### **4.1.5.3 Langkah-langkah:**

- o Mengisi data yang diminta dengan benar
- o Mencentang syarat dan ketentuan dari aplikasi
- o Memilih pilihan “*create an account*”
- o Masuk ke aplikasi

#### **4.1.6 Skenario eksepsional 1:**

*4.1.6.1 Prakondisi:* Akan muncul kolom-kolom untuk mengisi *username*, dan *password* yang harus diisi oleh pengguna.

*4.1.6.2 Pascakondisi:* Setelah mengisi kolom-kolom tersebut dengan data pengguna dan menekan “*sign in*”, maka akan langsung masuk ke aplikasi.

##### **4.1.6.3 Langkah-langkah:**

- o Mengisi data yang diminta dengan benar
- o Memilih pilihan “*sign in*”
- o Masuk ke aplikasi

## **4.2 Shopping Cart**

### **4.2.1 Deskripsi:**

*Buyer* atau pembeli dapat memilih atau menyimpan barang yang diinginkan atau akan dibeli tetapi tidak pada saat itu juga.

#### 4.2.2 **Trigger:**

Pembeli akan melihat barang kemudian memilih barang tersebut dan memilih pilihan yang tersedia yaitu “*add to shopping cart*”.

#### 4.2.3 **Input:**

*Input* akan diberikan oleh pengguna (pembeli) berupa klik pada pilihan “*add to shopping cart*” dengan *mouse* maupun *touch screen* melalui monitor *smartphone*.

#### 4.2.4 **Output:**

*Output* yang akan dikeluarkan setelah diberi perintah adalah pemberitahuan “*added to shopping cart*” dan langsung bisa di lihat kembali di list *shopping cart* yang ada.

#### 4.2.5 **Skenario Utama:**

4.2.5.1 *Prakondisi*: Pembeli melihat barang yang diinginkan dan memilih pilihan “*add to shopping cart*”.

4.2.5.2 *Pascakondisi*: Akan muncul pemberitahuan di layar yaitu “*added to shopping cart*”.

4.2.5.3 *Langkah-langkah*:

- o Memilih barang yang ingin dimasukan ke *shopping cart*
- o Memilih pilihan “*add to shopping cart*”

### 4.3 **Wishlist**

#### 4.3.1 **Deskripsi:**

*Buyer* bisa memasukkan barang yang mereka sukai ke dalam *wishlist* agar jika sewaktu-waktu *buyer* ingin membelinya mereka bisa dengan mudah menemukan produk tersebut jika sudah dimasukkan ke dalam *wishlist* dan kemudian siap untuk dimasukkan ke *shopping cart*.

#### 4.3.2 **Trigger:**

Pembeli akan melihat barang kemudian memilih barang tersebut dan memilih pilihan yang tersedia yaitu “*add to wishlist*”.

#### 4.3.3 **Input:**

Input akan diberikan oleh *buyer* (pembeli) berupa klik pada pilihan “add to wishlist” menggunakan tombol kiri mouse ataupun dengan sentuhan layar (*touchscreen*) pada smartphone.

#### 4.3.4 **Output:**

Output yang akan dikeluarkan yaitu pemberitahuan di layar berupa “*added to wishlist*” dan barang sudah berada di list *wishlist*.

#### 4.3.5 **Skenario Utama:**

4.3.5.1 *Prakondisi:* Pembeli melihat barang yang diinginkan dan memilih pilihan “*add to wishlist*”.

4.3.5.2 *Pascakondisi:* muncul pemberitahuan di layar berupa “*added to wishlist*” 4.3.5.3 *Langkah-langkah:*

- o Memilih barang yang ingin dimasukan ke *wishlist*
- o Memilih pilihan “*add to wishlist*”

### 4.4 Kolom Pencarian

#### 4.4.1 **Deskripsi:**

Pada fitur kolom pencarian ini, *user/buyer* dapat melakukan pencarian mengenai produk yang sedang mereka cari maupun toko yang sedang mereka cari.

#### 4.4.2 **Trigger:**

Pembeli ingin mencari suatu barang dan akan mencari dengan fitur kolom pencarian ini.

#### 4.4.3 **Input:**

Input yang akan diberikan oleh *user* yaitu berupa klik pada bagian atas pada kolom yang bertuliskan “cari” kemudian *user* menginputkan berupa ketik barang atau toko yang mereka inginkan kemudian klik tombol “cari” pada sebelah kanan kolom.

#### 4.4.4 **Output:**

output yang akan dikeluarkan oleh sistem yaitu laman yang berisi foto barang/toko terkait yang sedang dicari oleh *user*.

#### 4.4.5 **Skenario Utama:**

4.4.5.1 *Prakondisi:* *User* akan menginputkan nama barang/toko yang akan

dicari 4.4.5.2 *Pascakondisi:* laman berisi foto barang/toko terkait yang

sedang dicari *user* 4.4.5.3 *Langkah-langkah:*

- o klik kolom pada bagian atas yang bertuliskan “cari”
- o ketik nama barang atau toko yang ingin dicari
- o Memilih tombol “cari” pada bagian kanan kolom

### 4.5 **Payment Gateway**

#### 4.5.1 **Deskripsi:**

Setelah melakukan proses *checkout*, *buyer* akan melakukan proses pembayaran pada *payment gateway*. Pada aplikasi ini kami menyediakan metode pembayaran berupa *e-wallet* yaitu Link-Aja, dan bank-bank terkait yaitu ada bank BNI, BRI, BCA, dan Mandiri.

#### 4.5.2 **Trigger:**

Setelah melakukan *checkout*, maka pembeli akan melakukan proses pembayaran dengan fitur yang disediakan ini.



#### 4.5.3 **Input:**

Input akan diberikan oleh buyer setelah melakukan semua rangkaian pembelian dan konfirmasi produk

#### 4.5.4 **Output:**

Output yang dikeluarkan sistem adalah nomor virtual account untuk melakukan pembayaran oleh buyer

#### 4.5.5 **Skenario Utama:**

*4.5.5.1 Prakondisi:* Sistem akan mengeluarkan nomor virtual account untuk pembayaran

*4.5.5.2 Pascakondisi:* Setelah melakukan pembayaran sistem akan mengeluarkan rincian pembelian lengkap dengan waktu transaksi.

*4.5.5.3 Langkah-langkah:*

- Pilih barang yang akan di beli pada menu wishlist atau shopping cart,lalu klik tombol beli sekarang.
- Lakukan pengecekan ulang pada produk yang akan dibayarkan.
- Klik “Bayar Sekarang”
- Sistem akan menampilkan nomor virtual account untuk proses pembayaran .

### 4.6 **Informasi Produk Baru**

#### 4.6.1 **Deskripsi:**

*Buyer* dapat mengetahui informasi produk baru dari toko yang telah diikuti.

#### 4.6.2 **Trigger:**

Jika ada produk baru yang dijual pada aplikasi, maka *user* dapat melihat produk tersebut pada halaman produk baru ini.

#### 4.6.3 ***Input:***

Input yang akan diberikan oleh *user* yaitu berupa klik pada bagian bawah pada fitur yang bertuliskan “Feed”, setelah itu akan muncul informasi produk baru yang di post oleh toko yang telah user ikuti.

#### 4.6.4 ***Output:***

Output yang dikeluarkan berupa postingan terbaru dari toko yang user ikuti.

#### 4.6.5 ***Skenario Utama:***

4.6.5.1 *Prakondisi:* Halaman belum di geser ke bawah (*Refresh*) sehingga belum menampilkan produk terbaru

4.6.5.2 *Pascakondisi:* Halaman akan menampilkan produk terbaru.

4.6.5.3 *Langkah-langkah:*

- o Pastikan user telah mengikuti minimal 1 toko pada aplikasi FRIPERIE.
- o Klik icon “Feed” yang terletak pada bagian bawah.
- o Lalu, geser ke bawah (*Refresh*).

### 4.7 **Halaman Promo**

#### 4.7.1 ***Deskripsi:***

Pada halaman ini, *buyer* bisa mengetahui barang apa saja yang sedang diskon.

#### 4.7.2 ***Trigger:***

Jika ada promo yang diadakan baik oleh toko/*seller* maupun dari aplikasi, maka produk-produk yang sedang promo tersebut akan terlihat pada halaman ini.

#### **4.7.3 Input:**

Input yang diberikan user hanya mengklik fitur “Beranda” pada bagian bawah aplikasi.

#### **4.7.4 Output:**

Output yang dikeluarkan berupa postingan produk yang sedang promo.

#### **4.7.5 Skenario Utama:**

4.7.5.1 *Prakondisi*: halaman utama friperie

4.7.5.2 *Pascakondisi*: halaman produk yang sedang promo

4.7.5.3 *Langkah-langkah*:

- Setelah ada pada halaman beranda dapat terlihat beberapa produk yang sedang promo.
- Klik “Lihat Lainnya” untuk melihat lebih banyak barang yang sedang promo

### **4.8 Related Items**

#### **4.8.1 Deskripsi:**

Setelah *buyer* mengisi kolom pencarian mengenai barang yang ingin dicari, maka selanjutnya akan muncul beberapa produk terkait yang mungkin merupakan barang yang sedang dicari.

#### **4.8.2 Trigger:**

Barang/produk yang sering dicari oleh *buyer* akan ter-program dengan algoritma yang dibuat sehingga produk yang berkaitan tersebut akan terlihat pada halaman *related items* ini.

#### 4.8.3 **Input:**

Input yang diberikan user hanya mengklik fitur “Beranda” pada bagian bawah aplikasi.

#### 4.8.4 **Output:**

Output yang dikeluarkan adalah munculnya produk yang berkaitan dengan pencarian *user* sebelumnya.

#### 4.8.5 **Skenario Utama:**

4.8.5.1 *Prakondisi*: Masuk ke dalam beranda, ada menu “*Related Items*”

4.8.5.2 *Pascakondisi*: Klik “*See More*” untuk melihat barang atau produk

yang lain 4.8.5.3 *Langkah-langkah*:

- Setelah ada pada halaman beranda dapat terlihat beberapa produk yang berkaitan dengan yang dicari *user* sebelumnya.
- Klik “Lihat Lainnya” untuk melihat lebih banyak barang yang berkaitan

### 4.9 Kontak antara **Seller** dan **Buyer**

#### 4.9.1 **Deskripsi:**

*Buyer* dan *seller* dapat saling menghubungi dengan melakukan kontak via *messenger* yaitu fitur yang disediakan oleh aplikasi ini.

#### 4.9.2 **Trigger:**

Jika salah satu *seller* dan *buyer* ingin mengontak satu sama lain

#### 4.9.3 **Input:**

Input yang diberikan *user* adalah mengirim pesan kepada satu sama lain.

#### 4.9.4 **Output:**

Output yang dikeluarkan berupa pesan yang masuk ke pihak *buyer* maupun *seller*.

#### 4.9.5 **Skenario Utama:**

4.9.5.1 *Prakondisi*: Ada fitur untuk mengirim pesan antara *buyer* dan *seller*.

4.9.5.2 *Pascakondisi*: Jika ingin mengontak satu sama lain mengenai pesanan atau produk, maka mengklik “*Chat*” yang ada pada halaman toko dan mengirim pesan.

4.9.5.3 *Langkah-langkah*:

- Masuk ke dalam halaman toko yang ingin dihubungi
- Mengklik pilihan “*Chat*” dan kirim pesan kepada *seller*
- *Seller* yang menerima pesan dapat langsung membalasnya

### 4.10 **Review Produk**

#### 4.10.1 **Deskripsi:**

Setelah barang yang dipesan oleh *buyer* sampai tujuan, *buyer* bisa melakukan review produk/barang tersebut agar *buyer* yang lain mengetahui review-review yang ada pada barang tersebut.

#### 4.10.2 **Trigger:**

Setelah *buyer* menerima pesanan, maka dia dapat memberikan *review* kepada toko tersebut mengenai barang yang sudah diterima.

#### 4.10.3 **Input:**

Input yang diberikan *buyer* hanya mengklik fitur “Review Produk” yang akan muncul ketika produk sudah sampai di tangan *buyer*.

#### 4.10.4 **Output:**

Output yang dikeluarkan berupa postingan komentar dan *review* dari *buyer* yang langsung dapat dilihat pada komentar produk.

#### 4.10.5 **Skenario Utama:**

*4.10.5.1Prakondisi:* *Buyer* memberikan *review* berupa teks,gambar, dan

bintang. *4.10.5.2Pascakondisi:* Sistem akan memunculkan komentar pada

halaman produk. *4.10.5.3Langkah-langkah:*

- *Buyer* akan menerima notifikasi jika barang sudah sampai.
- Klik notifikasi tersebut.
- Lalu terdapat fitur “Review Produk”.

### 4.11 Pengembalian Barang

#### 4.11.1 **Deskripsi:**

Pada fitur ini, *buyer* dapat mengembalikan barang yang sudah dibeli dengan alasan yang jelas.

#### 4.11.2 **Trigger:**

Jika barang yang sudah sampai ke tangan *buyer* tidak sesuai dengan apa yang diinginkan, maka barang tersebut dapat dikembalikan dengan alasan yang jelas menggunakan fitur ini.

#### 4.11.3 **Input:**

Input yang diberikan user hanya mengklik fitur “Pengembalian barang” dan mengisi alasan nya.

#### 4.11.4 **Output:**

Output yang dikeluarkan berupa pemberitahuan detail mengenai pengembalian barang.

#### 4.11.5 **Skenario Utama:**

4.11.5.1 *Prakondisi:* Barang ingin dikembalikan

4.11.5.2 *Pascakondisi:* Mengklik fitur untuk mengembalikan barang, mengisi alasan kenapa barang ingin dikembalikan, barang dapat dikembalikan

4.11.5.3 *Langkah-langkah:*

- Mengklik “Kembalikan Barang”
- Memilih alasan mengapa barang ingin dikembalikan
- Menunggu konfirmasi dari *seller*
- Menerima konfirmasi dari *seller* dan pengembalian uang
- Barang dapat dikembalikan

## 4.12 **Pembatalan Pesanan**

### 4.12.1 **Deskripsi:**

*Buyer* dapat melakukan pembatalan pesanan sebelum *seller* mengirimkan barang dan harus disetujui oleh *seller*.

### 4.12.2 **Trigger:**

*Buyer* melakukan pembatalan pesanan dengan klik “batalkan pesanan” dan setelah pembatalan disetujui oleh *seller*, maka sistem akan otomatis membatalkan pesanan *buyer*.

### 4.12.3 **Input:**

Input yang diberikan user yaitu klik “batalkan pesanan” pada laman pesanan.

#### 4.12.4 **Output:**

Output yang diberikan berupa konfirmasi pembatalan pesanan oleh aplikasi.

#### 4.12.5 **Skenario Utama:**

4.12.5.1 *Prakondisi:* Pesanan ingin dibatalkan

4.12.5.2 *Pascakondisi:* Buyer dapat membatalkan pesanan dengan memakai fitur pembatalan pesanan

4.12.5.3 *Langkah-langkah:*

- Memilih produk yang akan dibatalkan pesanannya
- Mengklik “Batalkan Pesanan”
- Memilih alasan mengapa pesanan dibatalkan
- Pesanan berhasil dibatalkan

## 5. Requirements Non Fungsional

### 5.1 Atribut Kualitas

#### 5.1.1 **Pemeliharaan (*Maintainability*)**

Akan dilakukan *maintenance* secara berkala dan rutin untuk menghindari gangguan sistem, memperbaiki fitur atau menambahkan fitur, memperbaiki *bug*, dan sebagainya.

#### 5.1.2 **Keamanan (*Security*)**

Data pengguna terjamin aman, hanya orang-orang khusus saja (*Administrator*) yang dapat mengubah isi data dan program.

#### 5.1.3 **Dapat diandalkan (*Dependability*)**

Program dapat memenuhi layanan dan kebutuhan yang diajukan pelanggan.



## 5.2 Requirements Legal

*User* atau pengguna harus menggunakan aplikasi ini dengan sebaik-baiknya dan tidak melanggar hukum dan aturan yang berlaku. Jika pengguna melanggar aturan dalam penggunaan aplikasi ini maka akan dikenakan hukuman sesuai dengan aturan yang berlaku.

## Lampiran A: Daftar Kata-Kata Sukar

*E-commerce* : Perdagangan online

*Thrift shop* : Toko penghematan

Marak : Populer / naik daun

Entitas : Objek, orang

PC : *Personal Computer* / komputer pribadi

*Chat* : Percakapan

*Input* : Masukan

*Output* : Keluaran

## Lampiran B: Analysis Models



