

# UTS 2020/2021

## Deskripsi Kasus

Di masa pandemi ini, PT Maju Terus ingin memulai bisnis sebagai perantara yang menghubungkan para ibu yang kesulitan belanja, warung-warung sayur, dan para pengemudi ojol yang mulai kekurangan penumpang. Untuk itu, developer software diminta bantuannya untuk membangun perangkat lunak pendukungnya. Perangkat lunak dinamakan Titip Belanja. Karena PT Maju Terus belum punya pengalaman membangun software, pihak developer diminta untuk mendefinisikan sendiri perangkat lunak yang akan dibangun dengan melihat berbagai contoh perangkat lunak pendukung marketplace yang sudah ada, seperti Tokopedia, Bukalapak, Shopee, atau Lazada. Dari hasil eksplorasi terhadap aplikasi sejenis, developer diminta untuk menyesuaikan dengan kebutuhan PT Maju Terus.

## Deskripsi Perangkat Lunak Titip Belanja

Titip belanja adalah sebuah perangkat lunak yang memungkinkan para ibu (pembeli) untuk meminta bantuan (titip) kepada mang belanja (agen) yang dipercaya untuk membelikan bahan pokok sehari-hari (belanjaan) yang diperlukan ke warung-warung yang dipilih. Setelah membeli titipan belanja sesuai order, agen menginformasikan status belanjanya kepada pembeli yang titip.

## User Story

- Pembeli memilih/menentukan agen yang akan dititipi belanjaan
- Pembeli memilih warung tempat belanja yang akan dibeli
- Warung (penjual) mempromosikan (menampilkan) dagangannya
- Pembeli memilih bahan-bahan pokok (dagangan) yang akan dibeli (termasuk jumlahnya) ke warung yang dipilih
- Pembeli menyampaikannya rincian titipan belanjanya ke agen yang dipilih
- Agen menerima rincian order yang harus dibeli ke warung sesuai pilihan pembeli
- Agen menginformasikan order yang sudah dibelinya kepada pembeli yang titip belanja
- Pengelola Sistem (Admin) mendata agen yang bergabung dalam sistem Titip Belanja
- Pengelola Sistem (Admin) mendata warung yang bergabung dalam sistem Titip Belanja
- Pembeli dapat melihat riwayat belanjanya
- Warung dapat melihat riwayat penjualannya

## Batasan

- Stok dagangan dianggap selalu tersedia
- Urusan pembayaran tidak dicakup oleh sistem

## Pertanyaan

1. Model pengembangan apa yang cocok diterapkan untuk studi kasus diatas ? Jelaskan tentang model pengembangan tersebut dan jelaskan mengapa model pengembangan tersebut cocok!
2. Identifikasi apa kebutuhan fungsional dan non-fungsional yang diperlukan pada studi kasus diatas!
3. Sebutkan external entity pada studi kasus diatas!
4. Buatlah DFD untuk studi kasus diatas!
5. Buatlah data store dan kamus data untuk DFD yang sudah kamu buat di no 4 !
6. Buatlah structure chart (diagram terstruktur) dari studi kasus diatas!
7. Buatlah diagram status dari studi kasus diatas!

1) Model pengembangan waterfall model karena sudah disediakan definisi sistem beserta apa saja yang dapat sistem tersebut lakukan sesuai dari pandangan user. Waterfall model ini dapat digunakan apabila penjelasan sudah rinci dan domain-domain sudah jelas, maka dari itu kasus ini cocok untuk menggunakan waterfall model. Kasus ini juga lebih mudah untuk dilakukan per fase.

2) Kebutuhan fungsional

Kode	Pernyataan Kebutuhan	Penjelasan Tambahan
F01	P/L memiliki fungsi dimana pembeli dapat memilih agen yang akan dititipi belanjaan	-
F02	P/L memiliki fungsi dimana pembeli dapat memilih warung tempat belanja yang akan dibeli	-
F03	P/L memiliki fungsi dimana warung dapat mempromosikan dagangannya	-
F04	P/L memiliki fungsi dimana pembeli dapat memilih dagangan yang akan dibeli (termasuk jumlahnya)	Harus sudah memilih warungnya
F05	P/L memiliki fungsi dimana pembeli menyampaikan rincian belanjaan kepada agen	-
F06	P/L memiliki fungsi dimana agen menerima rincian order yang harus dibeli ke warung sesuai pilihan pembeli	-
F07	P/L memiliki fungsi dimana agen menginformasikan order yang sudah dibelinya kepada pembeli yang titip belanja	-
F08	P/L memiliki fungsi dimana admin mendaftarkan agen yang bergabung dalam sistem Titip Belanja	-
F09	P/L memiliki fungsi dimana admin mendaftarkan warung yang bergabung dalam sistem Titip Belanja	-

F10	P/L memiliki fungsi dimana pembeli dapat melihat riwayat belanjanya	-
F11	P/L memiliki fungsi dimana warung dapat melihat riwayat penjualannya	-

Kebutuhan non fungsional

Kode	Parameter	Pernyataan Kebutuhan
NF01	Reliability	Sistem tidak mencakup pembayaran

### 3) Entitas eksternal

Entitas Eksternal	Penjelasan
Pembeli	Pelayan tidak memiliki akses sepenuhnya pada P/L. User jenis pelayan ini hanya bisa menampilkan menu, mencatat pesanan, memberi status pesanan, dan mencetak invoice.
Penjual (warung)	Koki tidak memiliki akses sepenuhnya pada P/L. User jenis koki ini hanya bisa menerima pesanan yang dicatat pelayan dan memberi status pesanan.
Agen (ojol)	Kasir tidak memiliki akses sepenuhnya pada P/L. User jenis kasir ini hanya bisa mencatatkan pembayaran sesuai invoice ke catatan keuangan dan menerbitkan slip pembayaran.
Admin	Manajer memiliki akses sepenuhnya pada P/L. User jenis manajer ini bisa membuat laporan penjualan dan mengupdate menu makanan. Dalam membuat laporan penjualan, user ini harus mengakses data penjualan alias data pesanan dengan status sudah disajikan dan catatan keuangan.

### 4) DFD

### 5) Data store

6)

7)

8)

9)

10) Fungsional

## UTS 2018/2019

### Bagian I. (Bobot: 40%)

- 1) Sebutkan minimal dua (2) karakteristik penulisan spesifikasi kebutuhan P/L yang baik. Jelaskan karakteristik tersebut, serta berikan contohnya.
- 2) Apa yang digambarkan pada Diagram Konteks, dan apa manfaatnya? Uraikan dengan singkat dan jelas.
- 3) Uraikan dengan singkat dan jelas konsep perancangan berikut:
  - a. Abstraksi
  - b. Information Hiding
  - c. Refactoring
- 4) Rancangan antarmuka yang baik seharusnya menempatkan pengguna sebagai pemegang kendali. Jelaskan maksud pernyataan tersebut.

#### 1. Penulisan SKPL yang baik :

- Jelas/Lengkap → Mencakup semua aspek, hanya satu kebutuhan, mudah dibaca
- Konsisten → Tiap kebutuhan tidak ada yg konflik
- Tepat → Akurat dalam identifikasi situasi dan keterbatasannya. Kebutuhan mungkin benar di satu masa, dan tidak benar di masa lain
- Mudah diubah → Pengelompokan dan strukturisasi rapi
- Terurut → Sesuai prioritas. Makin kompleks, makin susah pengurutannya. Salah urut, inefficient development
- Dapat diuji → Dinilai secara kuantitatif, definisikan sistem yang mudah digunakan (cth: "sistem memiliki S/O Windows", "sistem memiliki fasilitas...")
- Dapat ditelusuri → Spesifikasi diberi nomor identifikasi unik, konsisten, struktur logis
- Tidak ambigu → Tidak ada interpretasi ganda, jangan pakai struktur bahasa yang jelek (cth: kata "dan", "atau", "seminimal mungkin", "user-friendly")

#### 2. Yang digambarkan pada diagram konteks adalah keterkaitan sistem dengan sistem lain atau aktor luar. Diagram ini menjelaskan fungsi sistem dan mendefinisikan jangkauan dari sistem. **Hanya ada satu lingkaran proses** dan minimal satu entitas eksternal/terminator dengan minimal ada satu input dan satu output. Semua terminator terhubung minimal satu aliran input atau output. Tidak ada pertukaran data antara terminator.

Aliran data → untuk menunjukkan data apa yang ditransfer entah itu dari entitas eksternal ke sistem ataupun sebaliknya

Proses → menggambarkan sistemnya

Entitas eksternal → menggambarkan pihak luar yang berperan menggunakan proses, baik itu aktor maupun sistem lain

3. Abstraksi → Cara untuk mengatur kompleksitas pada suatu sistem komputer karena setiap level memiliki kompleksitas yang berbeda-beda

Information hiding → Disebut juga data hiding atau encapsulation. Fungsinya untuk mengurangi kemungkinan 'efek samping'. Bila menggunakan information hiding, tidak terjadi dampak global bila dilakukan perubahan lokal, menekankan komunikasi melalui manajemen interface, mengurangi pemakaian data global, dan mengarah ke ide encapsulation (atribut perancangan yang berkualitas bagus).

Refactoring → Proses mengubah sistem PL dimana tidak mengubah perilaku eksternal kode, namun memperbaiki struktur internal. Refactoring dilakukan jika ada duplikat, kode terlalu panjang, coupling tinggi atau kohesi rendah, ada loop dalam loop, terlalu banyak parameter, perubahan satu tempat mempengaruhi tempat lain, level abstraksi tidak konsisten, dll

4. Arti dari pernyataan tersebut adalah:

- Interaksi tidak memaksa pengguna melakukan aksi yang tidak diinginkan
- Interaksi pengguna dapat di-interrupt/dibatalkan
- Interaksi dibuat fleksibel, khususnya saat kemampuan makin meningkat
- Hindari pengguna terbiasa mengerti dengan masalah teknis

## Bagian II. Studi Kasus [Bobot: 60%]

Sebuah warung makan khas daerah menyajikan makanan yang dipersiapkan dan dimasak terlebih dahulu oleh koki berdasarkan pesanan tamunya. Karena sangat lezat makanannya, warung tersebut semakin banyak pelanggannya. Agar pelayanan tetap baik, **manajer** warung berupaya menjaga layanannya membangun software untuk mendukung pelayanan di warung. Untuk memudahkan pengoperasian, software akan dijalankan pada **perangkat mobile/tablet dengan tampilan yang simple**. Software yang dibangun dipersyaratkan **tidak mengubah alur dan proses kerja layanan penjualan yang saat ini berlaku**.

Setiap ada pelanggan datang, **pelayan** akan **menunjukkan menu makanan** yang tersedia. Selanjutnya pelayan akan **mencatat makanan yang dipesan oleh pelanggan**. Catatan pesanan tersebut, selanjutnya **diserahkan kepada koki untuk dipersiapkan/dimasak makanannya**.

Setelah masak, **koki** **menandai (memberi status) pesanan makanan yang sudah siap**. Selanjutnya pelayan akan **menyajikan pesanan makanan** tersebut kepada pemesannya dan **menandai (memberi status) bahwa pesanan telah disajikan**.

Setelah selesai menyantap makanan, atau pesanan makanan akan dibawa pulang, **diterbitkan invoice** berdasarkan pesanan yang telah disajikan. Pelayan menyerahkan invoice kepada pelanggan untuk dilakukan pembayaran ke kasir. Setelah menerima pembayaran sesuai jumlah invoice, **kasir** akan **mencatatkan pembayaran** tersebut ke buku catatan keuangan dan **menerbitkan slip pembayaran** untuk diserahkan ke pelanggan melalui pelayan.

Setiap hari, manajer warung akan **membuat laporan penjualan** berdasarkan data pesanan dan catatan keuangan. Selain dari itu, manajer juga akan **mengupdate daftar makanan** jika terjadi perubahan.

- 1) Tuliskan spesifikasi kebutuhan fungsional perangkat lunak dan berikan penjelasan secukupnya
- 2) Tuliskan spesifikasi kebutuhan non fungsional perangkat lunak dan berikan penjelasan secukupnya
- 3) Identifikasi dan jelaskan entitas eksternalnya
- 4) Gambarkan Diagram Konteks untuk P/L tersebut,
- 5) Gambarkan DFD (level 0, hingga ke level berikutnya).  
Beri penjelasan singkat untuk tiap proses (tidak perlu membuat PSPEC)
- 6) Buatlah kamus data untuk data utamanya
- 7) Transformasikan DFD yang telah dibuat menjadi diagram hirarki modul atau *structured chart*

### 1) Kebutuhan fungsional

Kode	Pernyataan Kebutuhan	Penjelasan Tambahan
F01	P/L memiliki fungsi dimana pelayan dapat menampilkan menu makanan kepada pelanggan	-
F02	P/L memiliki fungsi dimana pelayan dapat mencatat pesanan pelanggan	-
F03	P/L memiliki fungsi dimana pelayan dapat menyerahkan informasi pesanan kepada koki	-
F04	P/L memiliki fungsi dimana koki dapat	Makanan yang sudah selesai

	memberi status (menandai) pesanan yang sudah siap disajikan	dimasak artinya adalah pesanan yang sudah siap untuk disajikan kepada pelanggan
F05	P/L memiliki fungsi dimana pelayan dapat memberi status (menandai) pesanan yang sudah disajikan	Pesanan sudah diantar/disajikan ke pelanggan
F06	P/L memiliki fungsi dimana pelayan dapat menerbitkan invoice untuk pelanggan	Invoice diterbitkan setelah pelanggan selesai makan atau saat makanan dibawa pulang
F07	P/L memiliki fungsi dimana kasir mencatatkan pembayaran sesuai invoice ke buku catatan keuangan	-
F08	P/L memiliki fungsi dimana kasir dapat menerbitkan slip pembayaran	-
F09	P/L memiliki fungsi dimana manajer dapat membuat laporan penjualan berdasarkan data pesanan dan catatan keuangan	Dilakukan setiap hari
F10	P/L memiliki fungsi dimana manajer dapat mengupdate menu/daftar makanan	-

## 2) Kebutuhan non fungsional

Kode	Parameter	Pernyataan Kebutuhan
NF01	Portability	Sistem dapat dijalankan di perangkat mobile/tablet
NF02	Reliability	Sistem tidak boleh mengubah alur dan proses penjualan yang sedang berlaku

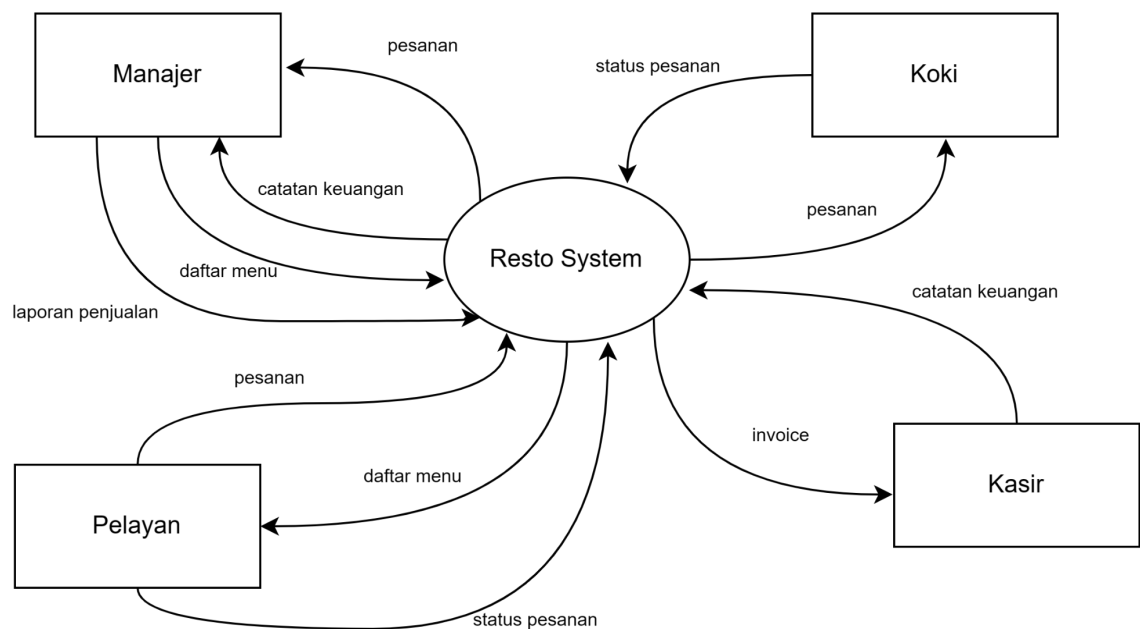
## 3) Entitas eksternal

Entitas Eksternal	Penjelasan
Pelayan	Pelayan tidak memiliki akses sepenuhnya pada P/L. User jenis pelayan ini hanya bisa menampilkan menu, mencatat pesanan, memberi status pesanan, dan mencetak invoice.



Koki	Koki tidak memiliki akses sepenuhnya pada P/L. User jenis koki ini hanya bisa menerima pesanan yang dicatat pelayan dan memberi status pesanan.
Kasir	Kasir tidak memiliki akses sepenuhnya pada P/L. User jenis kasir ini hanya bisa mencatatkan pembayaran sesuai invoice ke catatan keuangan dan menerbitkan slip pembayaran.
Manajer	Manajer memiliki akses sepenuhnya pada P/L. User jenis manajer ini bisa membuat laporan penjualan dan mengupdate menu makanan. Dalam membuat laporan penjualan, user ini harus mengakses data penjualan alias data pesanan dengan status sudah disajikan dan catatan keuangan.

#### 4) Diagram konteks



#### 5) DFD

#### 6) Kamus data

#### 7) Structured chart