

BINUS University

Academic Career: <i>Undergraduate / Master / Doctoral *)</i>		Class Program: <i>International/Regular/Smart Program/Global Class*)</i>	
<input type="checkbox"/> Mid Exam <input checked="" type="checkbox"/> Final Exam <input type="checkbox"/> Short Term Exam <input type="checkbox"/> Others Exam : _____		Term : Odd/Even/Short *)	
<input checked="" type="checkbox"/> Kemanggisan <input checked="" type="checkbox"/> Alam Sutera <input type="checkbox"/> Bekasi <input type="checkbox"/> Senayan <input type="checkbox"/> Bandung <input type="checkbox"/> Malang		Academic Year : 2020 / 2021	
Faculty / Dept. : School of Computer Science		Deadline	Day / Date : Rabu / 10 Feb 2021 Time : 13.00
Code - Course : COMP6056 - Program Design Methods		Class : All Classes	
Lecturer : Team		Exam Type : Online	
*) <i>Strikethrough the unnecessary items</i>			
<i>The penalty for CHEATING is DROP OUT!!!</i>			

I. Esai (30%)

1. a. [5%] Jelaskan **persamaan** dan **perbedaan** antara obyek (*object*) dan kelas (*class*)! Jawaban harap dilengkapi dengan referensi pendukung minimal satu.
 b. [5%] Berikan **satu contoh obyek** dan **satu contoh kelas**!
2. a. [5%] Apakah setiap *State Diagram* harus memiliki *self-transition*? Berikan alasan yang mendukung jawaban Anda!
 b. [5%] Di dalam *State Diagram* memiliki *Event* dan *Guard*, berikan **satu contoh Event** yang memiliki **dua Guard** yang menghasilkan **hasil State yang berbeda**!
3. [10%] Di dalam *Implementation Diagram* terdapat dua diagram, yaitu: *Component Diagram* dan *Deployment Diagram*. Apakah peranan *Component Diagram* dan *Deployment Diagram* dalam perancangan sistem berbasis objek (*Object Oriented Design*)?

II. Kasus (70%)

Kondisi pandemi COVID-19 menyebabkan pelanggan yang datang ke *mini-mart* atau toko kelontong berkurang. Namun tidak semua toko kelontong atau *mini-mart* mau membuka toko *online*. Pelanggan pun terkadang enggan untuk datang ke *mini-mart* langsung karena kondisi pandemi dan ada juga pelanggan yang sedang sakit atau tidak memungkinkan untuk keluar rumah untuk membeli keperluan sehari-hari. Prajna melihat peluang bisnis untuk menghubungkan keinginan pelanggan dibelikan barang sehari-hari dengan kebutuhan toko atau *mini-mart* untuk menjual barang dagangannya. Prajna ingin membuat sebuah aplikasi **E-AstShop** yaitu sebuah aplikasi *personal assistant* yang dapat membelikan barang kebutuhan pelanggan dengan cepat.

Verified by,

Kenny Jingga, S.Kom., M.T. (D6426) and sent to Program on Jan 18, 2021

Dengan ide bisnis ini juga dapat membuka peluang pekerjaan bagi driver yang menjadi *personal assistant*.

Adapun kebutuhan Prajna untuk aplikasi **E-AstShop** adalah sebagai berikut :

1. Setiap *mini-mart* atau toko yang ingin menjadi mitra toko dari aplikasi **E-AstShop** ini harus mendaftar terlebih dahulu. Adapun data yang wajib dimasukkan adalah nomor *handphone* dan *email*, nama toko dan alamat toko. Jika nomor *handphone* atau *email* pernah terdaftar, maka tidak bisa mendaftar kembali. Untuk pendaftaran toko atau *minimart* dapat dilakukan secara *group* sebagai contoh, BinusMart memiliki 10 cabang di Jakarta, maka mitra toko dapat mendaftar sekaligus secara kelompok.
2. Setiap pelanggan juga perlu mendaftar sebagai pengguna **E-AstShop** dengan memasukan nomor *handphone*, *email*, nama dan alamat. Jika nomor *handphone* atau *email* sudah pernah terdaftar, maka tidak bisa mendaftar kembali.
3. Data *driver* akan ditambahkan oleh admin aplikasi **E-AstShop** sebagai *personal assistant*. Data *driver* yang ditambahkan berupa nama *driver*, nomor *handphone*, *email*, dan nomor kendaraan serta jenis kendaraan yang dimiliki. Admin juga yang dapat melakukan blokir maupun buka blokir akun *personal assistant*.
4. Pelanggan dapat mencari produk yang diinginkan atau mencari toko atau *mini-mart* yang diinginkan. Dari produk yang dicari akan ditampilkan semua toko atau *mini-mart* yang menyediakan produk tersebut termasuk cabangnya.
5. Pada saat mau melakukan *order*, pelanggan dapat memilih toko atau *mini-mart* yang diinginkan kemudian baru memilih barang yang dibutuhkan. Selanjutnya menuliskan alamat tujuan barang yang akan diantarkan oleh *personal assistant*. Aplikasi akan menampilkan rincian harga dari barang yang dibeli dan ongkos antar barang ke tempat tujuan. *Member* dapat memilih kupon promo yang tersedia oleh aplikasi untuk mendapatkan potongan harga dari mitra toko. Kemudian memilih metode pembayaran dan melakukan pembayaran. Metode pembayaran tidak ada yang tunai, semuanya melalui *Fintech*. Selanjutnya, *order* akan diberikan pada *personal assistant* yang menerima *order* tersebut. Jika *order* sudah diterima oleh *personal assistant* maka pelanggan tidak dapat membatalkan *order*.
6. *Personal assistant* akan membelikan sesuai pesanan dari pelanggan dan mengantarkan *order* ke alamat tujuan diantarkan. Jika ada barang yang tidak tersedia, maka *personal assistant* akan melakukan konfirmasi dengan pelanggan apakah mau diganti barang lain atau dibatalkan. Nantinya mitra toko yang akan melakukan perubahan pada pesanan pelanggan.
7. Pelanggan dapat memberikan *rating* dan *review* pada mitra toko maupun mitra *personal assistant*. *Rating* dimulai dari terendah bintang satu menjadi tertinggi bintang lima.

Verified by,

Kenny Jingga, S.Kom., M.T. (D6426) and sent to Program on Jan 18, 2021

8. Mitra toko dapat melakukan penambahan produk baru di **E-AstShop** maupun melakukan perubahan harga atau jumlah barang. Setiap produk memiliki kategori masing-masing agar mudah untuk dilakukan pencarian. Setiap mitra toko dapat menyediakan berbagai macam kategori produk.
9. Admin dapat melakukan blokir maupun membuka blokir akun pelanggan maupun akun mitra toko yang bermasalah.

Anda sebagai konsultan Sistem Analis, diminta bantuan oleh Prajna untuk merancang aplikasi **E-AstShop**. Agar Prajna dapat memiliki gambaran secara visual pada sistem aplikasi **E-AstShop**, maka Anda diminta mengambarkan rancangan sistem dengan UML.

Tugas Anda:

- a. **[20%]** Rancang dan gambarkan *Use Case Diagram* untuk **E-AstShop** yang dapat mudah dipahami Prajna. Anda dapat menggunakan relasi *Include* / *Extend* jika memang diperlukan.
- b. **[20%]** Rancang dan gambarkan *Class Diagram*. Anda dapat menambahkan atribut lain untuk informasi yang belum dijelaskan pada kasus di atas.
- c. **[15%]** Rancang dan gambarkan dua *Activity Diagram* untuk dua *Use Case* dari jawaban (a).
Catatan: Gunakan *swimlane*.
- d. **[15%]** Rancang dan gambarkan dua *Sequence Diagram* untuk dua *Use Case* dari jawaban (a).
Catatan: *Use Case* yang digambarkan sama dengan *Use Case* untuk *Activity Diagram*.

Tambahan: Silahkan menambahkan asumsi sendiri untuk mempermudah ataupun memperjelas rancangan sistem dengan UML. Harap dituliskan asumsi yang Anda gunakan.

-- Selamat Mengerjakan --

Verified by,

Kenny Jingga, S.Kom., M.T. (D6426) and sent to Program on Jan 18, 2021