# Api E-Commerce

Buat api dan tampilan CMS untuk admin saja dengan ketentuan sebagai berikut:

**# Customer:**

inheritance dari class member silverstripe bukan admin tidak bisa login ke cms / admin

tambahan field:

* PhotoProfile
* Has Many Cart

**# Merchant:**

inheritance dari class member silverstripe bukan admin tidak bisa login ke cms / admin

tambahan field:

* PhotoProfile
* Has One Category
* Has Many Product

**# Merchant Category:**

Inheritance dari class data object silverstripe dengan field:

* Title

**# Product:**

Inheritance dari class data object silverstripe dengan field:

* Title
* Price
* Has Many Custom Image (Inheritance dari class image silverstripe)

Fitur dari aplikasi yang akan dibuat adalah:

* Customer dapat melakukan login dan registrasi serta forgot password
* Setelah melakukan registrasi Customer perlu melakukan validasi email sebelum dapat login
* Customer dapat melakukan edit profile serta upload photo profile
* Customer dapat melakukan pencarian merchant berdasarkan nama merchant dan atau nama product merchant. (Referensi grab/go food)
* Customer dapat melihat detail merchant dan product merchant (available/unavailable)
* Customer mempunyai sebuah cart dan product dari merchant dapat ditambahkan kedalam cart tersebut
* Customer juga dapat mengedit jumlah product pada cart dan menghapus cart
* Setelah selesai pada cartnya, customer dapat membuat order
* Merchant akan menerima email disetiap order yang berhasil dibuat
* Merchant dapat melakukan login dan registrasi serta forgot password
* Setelah melakukan registrasi Merchant perlu melakukan validasi email sebelum dapat login
* Merchant perlu diapprove oleh admin sebelum dapat mengakses fitur dibawah
* Merchant dapat menambah, mengedit dan menghapus product
* Merchant mempunyai status (open/close) untuk menandakan bahwa merchant sedang buka/tutup
* Jika buka, product dari merchant dapat dimasukkan cart dan dapat diorder oleh customer
* Sebaliknya jika tidak, product dari merchant tidak dapat dimasukkan cart dan diorder oleh customer
* Product juga mempunyai sebuah status (available/unavailable) untuk menandakan bahwa productnya dapat diorder atau tidak
* Merchant dapat melihat order yang telah dibuat oleh customer kepadanya
* Merchant dapat menerima atau menolak order
* Customer akan menerima email penerimaan atau penolakan terhadap order mereka

Fitur ini juga memerlukan halaman CMS untuk admin memonitor data. Semua data akan ditampilkan pada halaman CMS Admin dan hanya admin yang dapat login.

**# Job:**

* Buat planning waktu yang dibutuhkan dikelompokan berdasarkan fitur. (Gantt Chart / Google Spreedshet)
* Buat API untuk memenuhi fitur yang telah disebutkan
* Buat halaman CMS untuk admin
* Buat dokumentasi API pada Postman

**# Ketentuan API:**

* Setiap api yang diakses memerlukan token keamanan seperti ClientID
* Beberapa API (yang diakses setelah login) memerlukan token keamanan seperti AccessToken
* Setiap token keamanan wajib diletakkan pada header
* Penggunakan method (GET / POST) pada situasi tertentu

**# Note:**

* Struktur data diatas hanyalah struktur data sementara, Anda dapat menambahkan table atau field baru jika diperlukan

Api end point

/api -> main end point

# Customers

POST /customers/register

POST /customers/login

POST /customers/forgot-password

POST /customers/logout

# Profile Customer

GET /customers/profile/ [get customer]

PUT /customers/profile [edit profile]

PUT /customers/profile/picture [edit profile picture]

#Carts

GET /customers/carts [get customer carts]

POST /customers/carts [add Customer carts]

DELETE /customers/carts/ {product id} [delete product in carts with id]

PUT /customers/carts/ {product id} [edit total product in carts with id]

# Merchants

POST /merchants/register

POST /merchants/login

POST /merchants/forgot-password

GET /merchants/logout

GET /merchants/ {merchant id} [get merchants detail]

PUT /merchants/profile [edit profile merchants]

POST /merchants/profile/picture [edit profile picture merchant]

GET /merchants/ {merchant id}/products [get all merchant products]

PUT /merchants/profile/status [edit status merchant true: open/false: close]

# Products

GET /products

GET /products/ {product id} [get product by id]

POST /products

POST /products/ {product id} [edit product by id]

DELETE /products/ {product id} [delete product]

# Orders

POST /orders [create order]

PUT /orders [edit status order] (int 2 => accept / 3 => rejected)