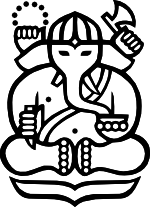
Laporan Tugas *Web Service* Pemrograman Integratif

*Customer Relationship Management*



Dipersiapkan oleh:

Aliyah Sausan Huwel

18215040

Program Studi Sistem dan Teknologi Informasi

STEI - ITB

Jl. Ganesha 10, Bandung 40132

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi**  **Teknik Informatika**  **STEI – ITB** | Nama Mata Kuliah | | Halaman |
| Pemrograman Integratif | | 21 |
| Dosen | I Gusti Baskara Nugraha S.T.,M.T.,Ph.D | Tgl: 14/12/2017 |

# Daftar Isi

[Daftar Isi 2](#_Toc501051057)

[Daftar Tabel 3](#_Toc501051058)

[Daftar Gambar 4](#_Toc501051059)

[1. Pendahuluan 5](#_Toc501051060)

[1.1 Tujuan Penulisan Dokumen 5](#_Toc501051061)

[1.2 Definisi, Istilah dan Singkatan 5](#_Toc501051062)

[1.3 Aturan Penomoran 5](#_Toc501051063)

[1.4 Referensi 5](#_Toc501051064)

[1.5 Deskripsi umum Dokumen (Ikhtisar) 5](#_Toc501051065)

[2 Deskripsi Umum Web Service 7](#_Toc501051066)

[2.1 Deskripsi Umum Web Service 7](#_Toc501051067)

[2.2 Deskripsi API 7](#_Toc501051068)

[2.3 Ruang Lingkup 7](#_Toc501051069)

[2.4 Batasan 8](#_Toc501051070)

[3 Deskripsi Kebutuhan 9](#_Toc501051071)

[3.1 Kebutuhan Fungsional 9](#_Toc501051072)

[3.2 Kebutuhan Non Fungsional 9](#_Toc501051073)

[3.3 Kebutuhan Software-Hardware 10](#_Toc501051074)

[3.4 Batasan Perancangan 10](#_Toc501051075)

[4 Deskripsi Perancangan Web Service 11](#_Toc501051076)

[4.1 Lingkungan Implementasi 11](#_Toc501051077)

[4.2 Use Case 11](#_Toc501051078)

[4.3 Deskripsi Fungsi 11](#_Toc501051079)

[4.4 Matriks Kerunutan 14](#_Toc501051080)

[5 Pengembangan Web Service 14](#_Toc501051081)

[6 Deskripsi dan Hasil Uji 17](#_Toc501051082)

# Daftar Tabel

[Tabel 1: Tabel Daftar Istilah dan Singkatan 5](#_Toc501049933)

[Tabel 2: Tabel Aturan Penomoran 5](#_Toc501049934)

[Tabel 3: Tabel Deskripsi API 7](#_Toc501049935)

[Tabel 4: Kebutuhan Fungsional 9](#_Toc501049936)

[Tabel 5: Kebutuhan Non-Fungsional 9](#_Toc501049937)

[Tabel 6: Tabel Lingkungan Implementasi 11](#_Toc501049938)

[Tabel 7: Matriks Kerunutan 14](#_Toc501049939)

# Daftar Gambar

[Gambar 1: Skema Web Service CRM 7](file:///C:\Users\sausan\Desktop\AutoRecovery%20save%20of%20Laporan%20Tugas%20Web%20Service%20Pemrograman%20Integratif.asd.docx#_Toc501049954)

[Gambar 2: Pengembangan dengan framework Beego 14](#_Toc501049955)

[Gambar 3: Pengembangan web service dengan framework Bee API 15](#_Toc501049956)

[Gambar 4: Pengembangan web service 15](#_Toc501049957)

[Gambar 5: Hasil Pengujian 1 18](file:///C:\Users\sausan\Desktop\AutoRecovery%20save%20of%20Laporan%20Tugas%20Web%20Service%20Pemrograman%20Integratif.asd.docx#_Toc501049958)

[Gambar 6: Hasil Pengujian 2 19](file:///C:\Users\sausan\Desktop\AutoRecovery%20save%20of%20Laporan%20Tugas%20Web%20Service%20Pemrograman%20Integratif.asd.docx#_Toc501049959)

[Gambar 7: Hasil Pengujian 3 20](file:///C:\Users\sausan\Desktop\AutoRecovery%20save%20of%20Laporan%20Tugas%20Web%20Service%20Pemrograman%20Integratif.asd.docx#_Toc501049960)

[Gambar 8: Hasil Pengujian 4 21](file:///C:\Users\sausan\Desktop\AutoRecovery%20save%20of%20Laporan%20Tugas%20Web%20Service%20Pemrograman%20Integratif.asd.docx#_Toc501049961)

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen Laporan Tugas *Web Service* ini dibuat untuk menjabarkan spesifikasi *web service* dalam bentuk laporan mengenai proses pengembangan pembuatan *web service*. Dokumen ini juga digunakan untuk menguji keseluruhan sistem. Dokumen ini akan diberikan kepada Pak Baskara selakudosen Pemrograman Integratif.

## Definisi, Istilah dan Singkatan

Berikut ini semua definisi dan singkatan yang digunakan dalam dokumen ini.

Tabel 1: Tabel Daftar Istilah dan Singkatan

|  |  |
| --- | --- |
| Istilah | Definisi |
| FR | Functional Requirement  Kebutuhan fungsional dalam pengembangan pernagkat lunak ini |
| Database | Kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut |
| Sistem operasi | Perangkat lunak komputer atau software yang bertugas untuk melakukan kontrol dan manajemen perangkat keras dan juga operasi-operasi dasar sistem, termasuk menjalankan software aplikasi seperti program-program pengolah data yang bisa digunakan untuk mempermudah kegiatan manusia. |
| Server | Server adalah sebuah sistem komputer yang menyediakan jenis layanan (service) tertentu dalam sebuah jaringan komputer |
| DBMS | Database Management System |
| Client | PC yang bergabung dalam jaringan sebagai penerima data |

## Aturan Penomoran

Tabel 2: Tabel Aturan Penomoran

|  |  |
| --- | --- |
| Kode | Arti |
| FR | *Functional requirement* |
| NFR | *Nonfunctional Requirement* |
| SP | Skenario Pengujian |

## Referensi

Dokumen ini dibuat berdasarkan template yang diunduh di s.id/tubesRPL yang dibuat oleh asisten mata kuliah Rekayasa Perangkat Lunak.

## Deskripsi umum Dokumen (Ikhtisar)

Dokumen SKPL ini dibagi menjadi sembilan bagian utama. Bagian pertama berisi penjelasan mengenai tujuan pembuatan, definisi, istilah, singkatan, aturan penomoran, referensi, dan deskripsi umum.

Bagian kedua berisi penjelasan web service yang akan dikembangkan secara umum. Pada bagian ini akan dijabarkan pula deskripsi umum web service, ruang lingkup, batasan, dan lingkungan operasi web service.

Bagian ketiga berisi penjabaran deskripsi kebutuhan yang dibutuhkan dalam pengembangan web service, dimulai dari kebutuhan fungsional, kebutuhan non-fungsional, dan batasan perancangan.

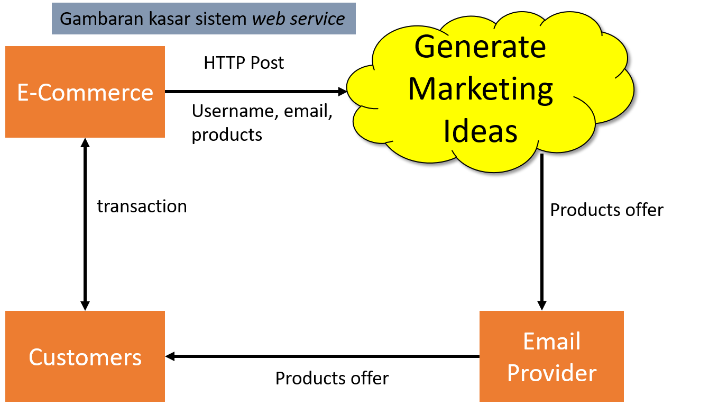
Bagian keempat berisi penjabaran deskripsi perancangan global, dimulai dari lingkungan rancangan implementasi, use case, deskripsi fungsi, dan matriks kerunutan.

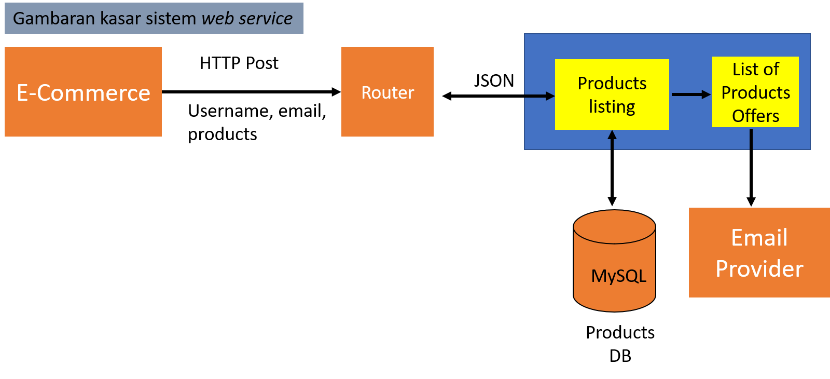
Bagian kelima berisi penjabaran pengembangan web service, bagian ini menjelaskan bagaimana web service dikembangkan, kesalahan apa saja yang ditemukan, bagaimana menanganinya, serta pelajaran apa saja yang didapatkan dari pengembangan web service.

Bagian keenam berisi penjabaran deskripsi dan hasil pengujian yang telah dilakukan.

# Deskripsi Umum Web Service

## Deskripsi Umum Web Service





Gambar 1: Skema Web Service CRM

*Customer Relationship Management* adalah *web service* yang berfungsi untuk memastikan bahwa pelanggan dari suatu sistem *online shopping* akan kembali menggunakan barang/jasa dalam sistem sehingga *customer retention* atau kemampuan dari suatu perusahaan dalam memastikan *customers-*nya setia dapat diterapkan dengan baik. *Web Service* ini akan menerima data pelanggan berupa username, email, produk yang dibeli, dan kategori dari barang yang dibeli. *Web service* ini akan mengelola daftar produk dan promosi yang dapat ditawarkan kepada pelanggan tersebut berdasarkan transaksi pelanggan sebelumnya. Setelah mendapatkan daftar produk yang dapat ditawarkan, *web service* akan mengirimkan pesan ucapan terima kasih dan daftar produk yang dapat ditawarkan kepada pelanggan.

## Deskripsi API

Berikut adalah tabel yang berisi fungsi dari API yang dapat digunakan oleh pengembang lain.

Tabel 3: Tabel Deskripsi API

|  |  |  |  |
| --- | --- | --- | --- |
| **API** | **Method** | **URL** | **Deskripsi** |
| UPDATELIST | POST | 167.205.67.248:12345/UPDATELIST | Mengubah data stock produk dalam *database,* membuat pesan berdasarkan data yang sesuai dengan kategori produk, mengirimkan pesan |

## Ruang Lingkup

Berikut ini merupakan ruang lingkup yang ada dalam *Customer Relationship Management*.

1. CRM hanya bekerja saat transaksi berjalan, dimana CRM akan mengelola hasil transaksi dan menghasilkan daftar produk dan promosu jika kategori produk yang dibeli pelanggan masih tersedia di database.
2. CRM akan mengirimkan email berdasarkan ada tidaknya produk yang bisa ditawarkan atau tidak. Jika tidak terdapat penawaran produk, CRM akan mengirim email ucapan terima kasih ke pelanggan. Jika terdapat penawaran produk, CRM akan mengirim email ucapan terima kasih dan penawaran produk kepada pelanggan.
3. Pengembangan CRM dalam bahasa Go

## Batasan

*Customer Relationship Management*(CRM)memiliki batasan-batasan berikut.

1. CRM hanya dapat mengelola satu pelanggan membeli satu barang pada satu waktu
2. CRM hanya berguna pada *online shopping* yang menggunakan *database* produk dengan sistem kategori barang

# Deskripsi Kebutuhan

## Kebutuhan Fungsional

Berikut adalah kebutuhan fungsional dari sistem layanan lokasi gedung dan ruangan.

Tabel 4: Kebutuhan Fungsional

|  |  |  |
| --- | --- | --- |
| **ID** | **Kebutuhan** | **Deskripsi** |
| FR-01 | Sistem dapat menerima data username pelanggan, email pelanggan, kategori barang dan barang yang dibeli dari suatu transaksi | Data username pelanggan, email pelanggan, kategori barang, dan barang yang dibeli dapat ditampilkan |
| FR-02 | Sistem dapat mengupdate database | Sistem dapat mengurangi jumlah barang yang ada dalam *database* berdasarkan masukan barang yang dibeli pelanggan |
| FR-03 | Sistem dapat melakukan pencarian barang-barang dan promosinya dari jenis barang yang dibeli dalam transaksi di database produk | Sistem akan menghasilkan daftar barang dan promosi dihasilkan berdasarkan pencarian dari kategori barang yang dibeli pelanggan di transaksi sebelumnya |
| FR-04 | Sistem mengirimkan email kepada pelanggan | Sistem dapat mengirim email ucapan terima kasih telah membeli barang kepada pelanggan. Apabila terdapat produk yang ditawarkan, maka email juga berisi tawaran produk dan promosi kepada pelanggan |

## Kebutuhan Non Fungsional

Berikut ini merupakan tabel kebutuhan non fungsional dari aplikasi Super Gaji.

Tabel 5: Kebutuhan Non-Fungsional

| **ID** | **Parameter** | **Kebutuhan** |
| --- | --- | --- |
| NFR-001 | Availability | Sistem harus dapat beroperasi terus menerus 5 hari per minggu, selama minimal 12 jam per hari |
| NFR-002 | *Reliability* | Sistem harus selalu dapat mengirim email kepada pelanggan |
| NFR-003 | *Ergonomy* | Sistem harus mudah dipahami dan diimplementasikan ke website |
| NFR-004 | *Response time* | Sistem harus dapat mengirim email setiap satu transaksi kurang dari 5 detik |
| NFR-005 | *Security* | Sistem tidak akan mengakses database user dari website, hanya mendapatkan *username* dan *email* pelanggan |

## Kebutuhan Software-Hardware

Berikut ini merupakan kebutuhan software:

1. Sistem dibuat menggunakan bahasa pemrograman GO
2. Sistem memiliki prosesor minimal 2GHz
3. Sistem memiliki RAM minimal 2GB
4. Sistem dikembangkan melalui ubuntu
5. Client mengakses sistem dengan *web browser* selama terhubung dengan jaringan ITB atau melalui OpenVPN ITB.

## Batasan Perancangan

Terdapat beberapa batasan yang diterapkan saat melakukan perancangan *Customer Relationship Management*. Sistem hanya mengurus *username* dan *email* pelanggan tanpa mengakses database pelanggan dalam *website*. Selain itu, CRM juga berdasar pada asumsi bahwa tidak ada *updating database* dalam suatu website setiap kali terjadi transaksi sehingga CRM juga akan mengatur fungsional *updating database.*

# Deskripsi Perancangan Web Service

## Lingkungan Implementasi

Berikut adalah rincian informasi mengenai lingkungan implementasi dari *web service*

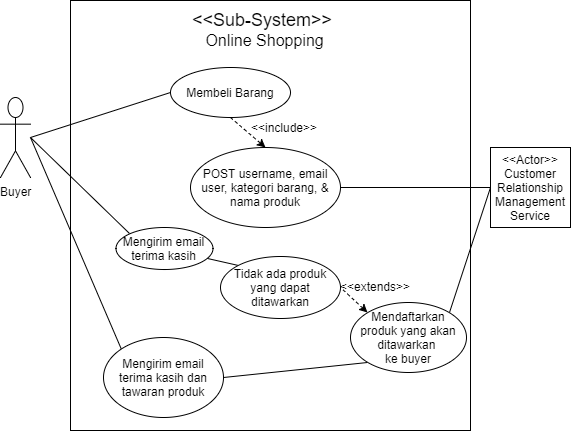
ITB Room Locator:

Tabel 6: Tabel Lingkungan Implementasi

|  |  |
| --- | --- |
| **Sistem Operasi** | Ubuntu |
| **DBMS** | MySQL |
| **Server** | HTTP Server, Gmail |
| **Development Tools** | Sublime Text, Bash, Postman |
| **Bahasa Pemrograman** | GoLang |
| **Package** | Net/http, net/smtp, strings |
| **Format Data** | JSON |

## *Use* Case

Berikut ini merupakan *use case* dari *web service Customer Relationship Management.*



## Deskripsi Fungsi

Berikut ini adalah fungsi-fungsi yang diterapkan dalam Customer Relationship Management.

1. Fungsi Prods

Fungsi Prods merupakan fungsi yang akan digunakan saat *service requestor* mengakses *web service* melalui POST.

**Initial State (IS)**:

Database Produk tersedia

**Final State (FS)**:

Fungsi web service untuk melakukan listing products dan mengirim email ke pelanggan

**Spesifikasi Proses/algoritma**:

1. Memanggil fungsi remove user
2. Memanggi fungsi Post Transaction
3. Memanggil fungsi Update Database Products
4. Mengambil data user pertama dari array user
5. Memanggil fungsi Customer Preferences
6. Memanggil fungsi send email
7. Fungsi Post Transaction

Fungsi Post Transaction merupakan fungsi yang akan akan melakukan pembacaan pada body dari request HTML dan menghasilkan suatu user baru yang terdiri atas username, email, produk yang dibeli, dan kategori produk.

**Initial State (IS)**:

User tidak ada

**Final State (FS)**:

User tersedia

**Spesifikasi Proses/algoritma**:

1. Membaca body request menggunakan json
2. Menggabungkan data user ke array user
3. Melakukan encode data array user
4. Fungsi Customer Preferences

Fungsi Customer Preferences merupakan fungsi yang digunakan untuk menghasilkan string, yang akan dipakai sebagai isi pesan dari email.

**Initial State (IS)**:

Tersedia user

**Final State (FS)**:

String berisi isi pesan dari email

**Spesifikasi Proses/algoritma**:

1. Memanggil fungsi Get Selected Products
2. Membangun string dari array produk-promosi menggunakan join string
3. Fungsi Remove User

Fungsi Remove User merupakan fungsi yang digunakan untuk menghapus data user dari array user.

1. Fungsi Get Selected Product

**Initial State (IS)**:

Tersedia kategori barang dalam bentuk string

**Final State (FS)**:

Menghasilkan array produk berikut promosi

**Spesifikasi Proses/algoritma**:

1. Membuka database mysql produk
2. Melakukan *query* : select products\_name, promotion from product where category = kategori barang and stocks > 0
3. Pembuatan variable daftar produk
4. Pembacaan setiap baris

> Ditampung terlebih dahulu ke variable string a dan b, pembuatan

> Pembuatan variable produk, setting nama dan promosi dari variable produk tersebut dengan a dan b

> Menyambungkan produk yang ada ke daftar produk

1. Fungsi Update Database Products

Fungsi Update Database Products adalah fungsi yang digunakan untuk melakukan update database produk berdasarkan nama pembelian barang.

**Initial State (IS)**:

Tersedia user

**Final State (FS)**:

Jumlah barang yang dibelikan berkurang 1 dari jumlah sebelumnya

**Spesifikasi Proses/algoritma**:

1. Membuka database mysql produk
2. Melakukan *query* : update product set stocks=stocks-1 where category = kategori barang dari user and products\_name = nama produk yang dibeli user and stocks > 0
3. Fungsi Send

Fungsi Send adalah fungsi yang digunakan untuk melakukan pengiriman email ke email user.

**Initial State (IS)**:

Tersedia pesan email dalam bentuk string, dan tujuan pengiriman email dalam bentuk string

**Final State (FS)**:

Pesan terkirim

**Spesifikasi Proses/algoritma**:

1. Melakukan setting akun sumber pengiriman email dan passwordnya
2. Melakukan setting message berisi data dari, untuk, subjek, cc, dan isi pesan
3. Menggunakan fungsi send mail lewat smtp dengan smtp gmail

## Matriks Kerunutan

Tabel 7: Matriks Kerunutan

| **ID Kebutuhan** | **Fungsi** | **Keterangan** |
| --- | --- | --- |
| FR-01 | Fungsi Prods dan Post Transaction |  |
| FR-02 | Fungsi Prods dan Update Database Products |  |
| FR-03 | Fungsi Prods dan Get Selected Products |  |
| FR-04 | Fungsi Prods, Customer Preferences, Get Selected Products, dan Send Mail |  |

# Pengembangan Web Service

Pengembangan *Web Service* dilaksanakan dalam 3 langkah utama.

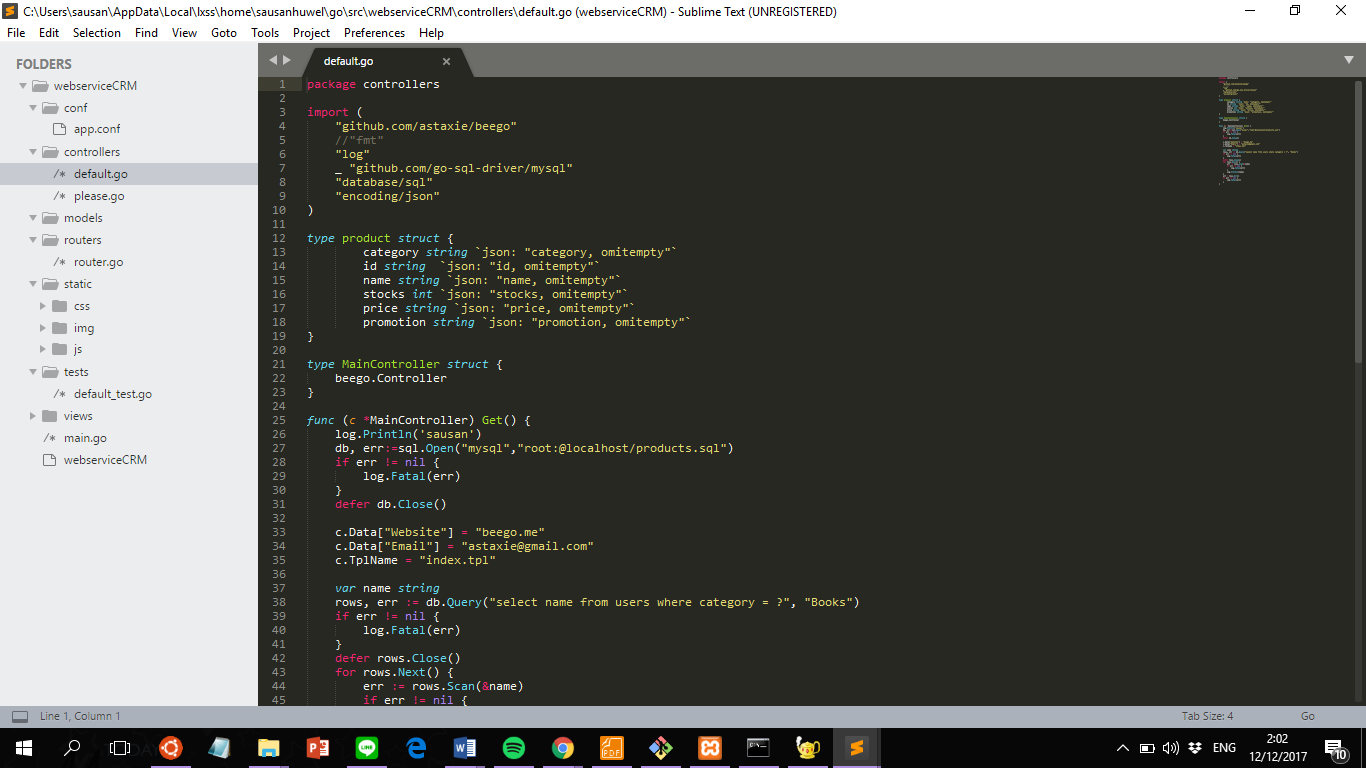
1. Langkah Pengembangan Database

* Membuat database dalam excel
* Mengubah file excel menjadi csv
* Import file csv ke database

1. Pengembangan web service, method POST

Dalam pengembangan *web service*, penulis melakukan beberapa kali perubahan.

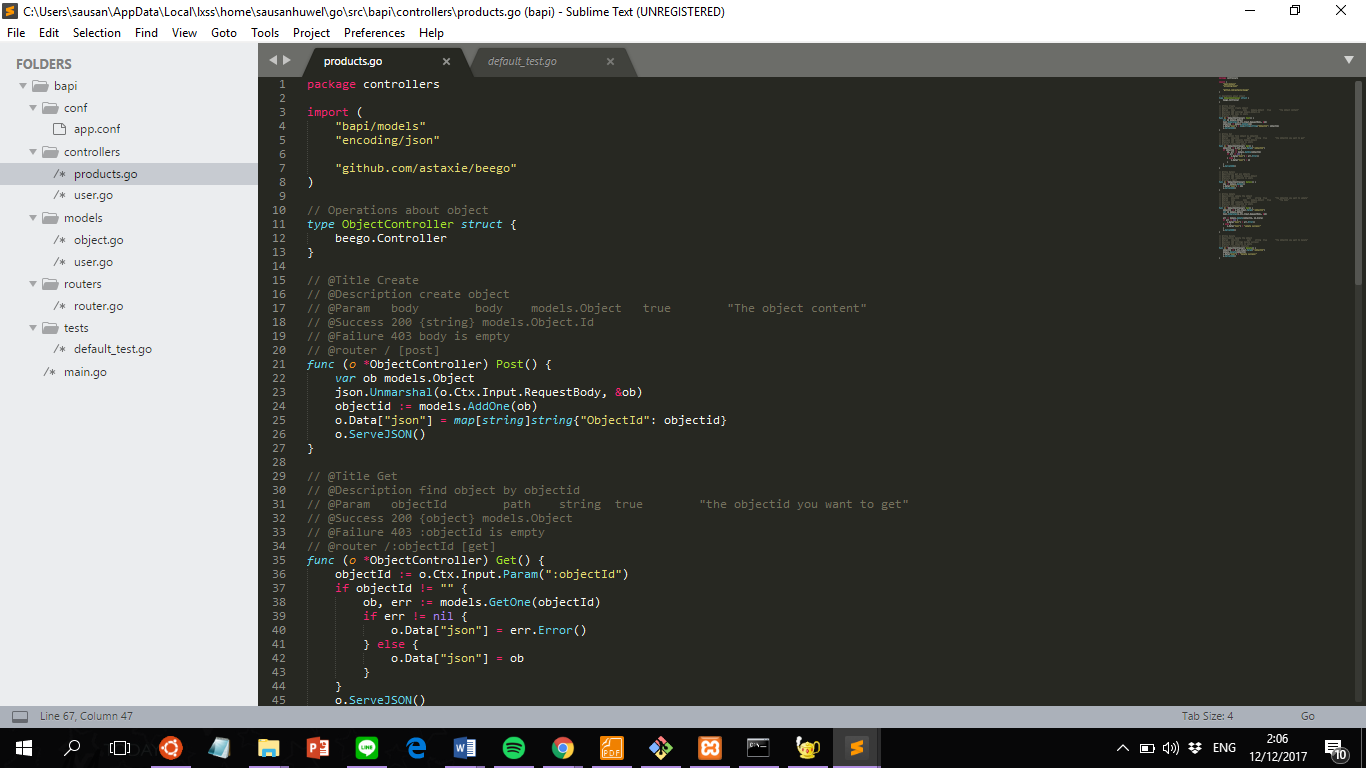
1. Penulis mengembangkan web service menggunakan *framework* Beego



Gambar 2: Pengembangan dengan framework Beego

Pengembangan web service dengan menggunakan *framework* beego tidak menyediakan *serve json* sehingga penggunaan serve json dilaksanakan secara paksa, sementara method yang akan digunakan dalam CRM adalah method POST. Selain itu, saat penulis mencari dokumentasi dari beego, terdapat penjelasan bahwa beego memiliki *framework* tersendiri untuk pembuatan *web service* berneama Bee API sehingga penulis memutuskan untuk tidak memakai *framework* Beego.

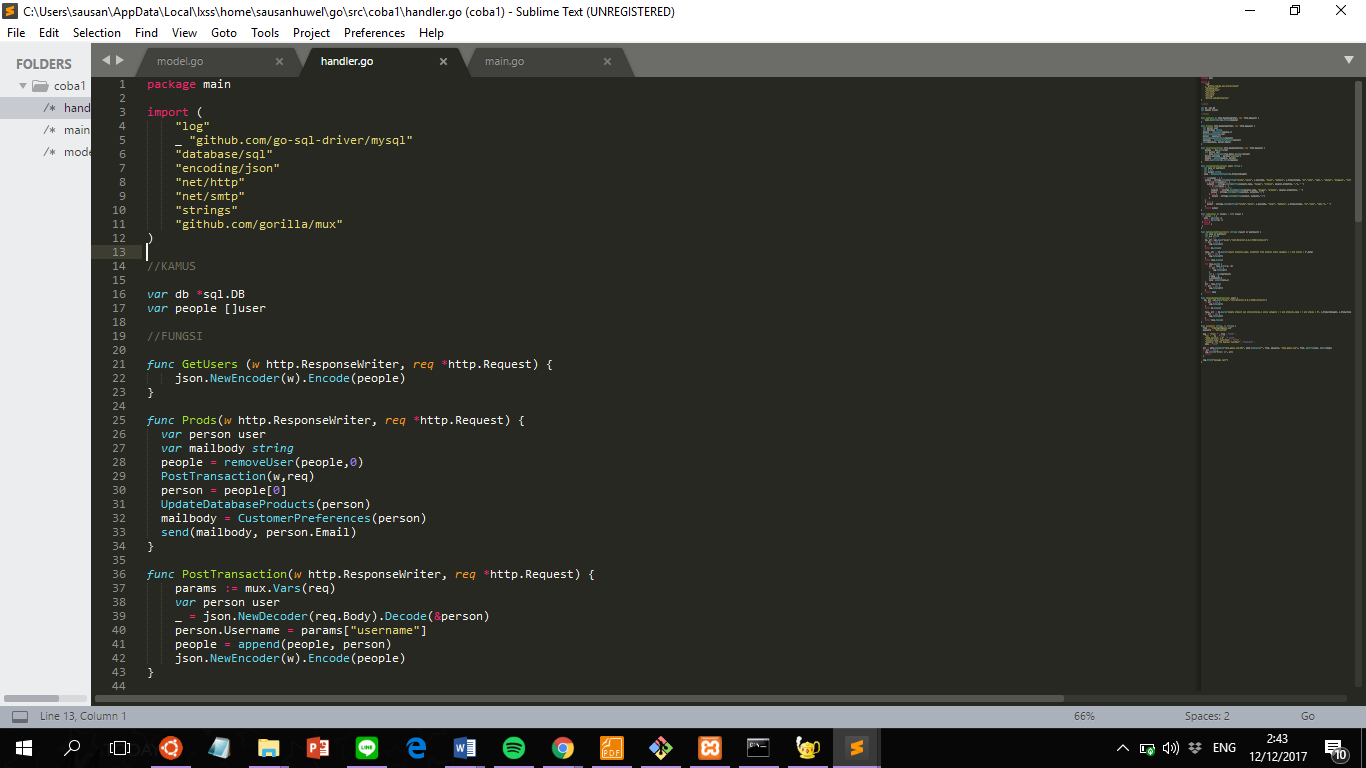
1. Penulis mengembangkan *web service* dengan *framework* Bee API



Gambar 3: Pengembangan web service dengan framework Bee API

Penulis mengembangkan *web service* dalam *framework* Bee API. Dalam pengembangannya, penulis menemukan bahwa *framework* Bee API memiliki standar pasti untuk setiap metode pembuatan *web service*. Karena penulis ingin mempelajari bagaimana pembuatan *web service* dan kurangnya dokumentasi *framework* Bee API, maka penulis memutuskan untuk tidak menggunakan *framework* Bee API.

1. Penulis mengembangkan web service dengan gorilla mux dan net/http



Gambar 4: Pengembangan web service

Penulis memutuskan untuk mengembangkan *web service* menggunakan *gorilla mux* sebagai *package* *router* dan *dispatcher*, dan package net/http.

1. Implementasi Mail
2. Implementasi Email menggunakan API Gmail

Penulis melakukan implementasi mail menggunakan API Gmail, namun penggunaan API Gmail mengharuskan adanya authentication setiap kali adanya pengiriman email sehingga API ini tidak sesuai dengan *web service* CRM.

1. Implementasi Email menggunakan SMTP *Package*

Penulis memutuskan untuk memanfaatkan pengiriman email menggunakan fungsi SendMail dari  *package* SMTP melalui server gmail.

# Deskripsi dan Hasil Uji

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Kesimpulan** |
| SP-01 | Pengujian menerima data *username, email,* produk yang dibeli, dan kategori produk | * Dalam Postman, melakukan request POST ke 167.205.67.248:12345/UPDATELIST | Memasukan data *username*, *email*, produk yang dibeli, dan kategori produk pada body request | Data berhasil diterima | Data muncul di layer dengan status 200 OK | Diterima |
| Hasil yang Didapat | Gambar 5: Hasil Pengujian 1 | | | | | |
| SP-02 | Pengujian *updating database* dimana jumlah produk dari data produk yang dibeli berkurang satu | * Dalam Postman, melakukan request POST ke 167.205.67.248:12345/UPDATELIST | Memasukan data *username*, *email*, produk yang dibeli, dan kategori produk pada body request | Jumlah kuantitas barang berkurang satu dalam database | Jumlah kuantitas barang berkurang satu dalam database | Diterima |
| Hasil yang Didapat | Sebelum Request  Setelah Request  Gambar 6: Hasil Pengujian 2 | | | | | |
| SP-03 | Pengujian mengirimkan email kepada pelanggan dimana tawaran produk pasti ada | * Dalam Postman, melakukan request POST ke 167.205.67.248:12345/UPDATELIST | Memasukan data *username*, *email*, produk yang dibeli, dan kategori produk pada body request | Terdapat pesan ucapan terima kasih dan tawaran produk di email *customer* | Terdapat pesan ucapan terima kasih dan tawaran produk di email *customer* | Diterima |
| Hasil yang Didapat | Gambar 7: Hasil Pengujian 3 | | | | | |
| SP-04 | Pengujian mengirimkan email kepada pelanggan dimana tawaran produk tidak ada | * Dalam Postman, melakukan request POST ke 167.205.67.248:12345/UPDATELIST | Memasukan data *username*, *email*, produk yang dibeli, dan kategori produk pada body request | Terdapat pesan ucapan terima kasih di email *customer* | Terdapat pesan ucapan terima kasih di email *customer* | Diterima |
| Hasil yang Didapat | Gambar 8: Hasil Pengujian 4 | | | | | |