

## BAB III

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Kerja magang yang dilakukan oleh mahasiswa di Onero Solution sebagai *back-end developer intern* mengerjakan beberapa tugas yang di percaya dan di beri tanggung jawab atas tugas yang diberikan. Tugas yang diberikan mahasiswa membuat fitur yang dibutuhkan dalam sebuah project. Selain itu, melakukan uji coba sebuah fitur dan memperbaiki jika terdapat *error* atau *bug*.

Pengerjaan sebuah project yang dilakukan oleh mahasiswa secara langsung di awasi dan dibimbing oleh Filbert Limena selaku *project manager* dan Hanif selaku *assistant project manager*. Mahasiswa mengerjakan project Bersama tim dari *front-end* dan *back-end*. Mahasiswa juga melakukan koordinasi dengan sesama *back-end developer* untuk bertanya tentang *package/function* yang telah dibuat dan juga berkoordinasi dengan tim *front-end* tentang input dan output API apa yang dibutuhkan.

Alur kerja yang dilakukan oleh mahasiswa cukup sederhana, dimulai dari briefing pada saat rapat yang dilakukan setiap hari nya pada pukul 09.00 hingga selesai. *Briefing* pada rapat mencakupi laporan progress yang telah dikerjakan dan progress yang akan di kerjakan. Progres yang akan di kerjakan telah diberikan oleh *assistant project manager* berupa *to-do list*. Setelah pekerjaan di selesaikan, mahasiswa melakukan push ke gitlab dan di upload ke *server* oleh ketua tim *back-end* yang nanti nya akan digunakan oleh tim *front-end*. Setelah itu, akan ada rapat

untuk melakukan *review progress* yang telah dikerjakan. Jika terdapat *error* atau *bug* atau salah format atau salah algoritma maka assistant project manager akan meminta mahasiswa untuk memperbaikinya. Semua penugasan yang diberikan melalui *zoom meeting*, *direct message lark*, dan sistem CRM Onero.

### 3.2 Tugas yang Dilakukan

Tabel di bawah merupakan timeline kerja dari mahasiswa secara garis besar selama kerja magang di onero solutions. Setiap pekerjaan memiliki pekerjaan rincian sebagai berikut ini:

*Tabel 3. 1 Timeline Kerja Magang Mahasiswa*

MINGGU KE	JENIS PEKERJAAN YANG DILAKUKAN MAHASISWA
1 (14 Juni – 18 Juni)	Membuat <i>website weather</i> dengan menggunakan Laravel dan <i>review</i> .
2 (21 Juni – 25 Juni)	Mempelajari Lumen dengan membuat Web API Artikel Sederhana dan <i>review</i> .
3 (28 Juni – 2 Juli)	Membuat Web API E-Commerce sederhana dengan laravel.
4 (5 Juli - 9 Juli)	Menyelesaikan Web API E-Commerce dan <i>review</i> .
5 (12 Juli – 16 Juli)	Memulai membuat project AXL Perfume, Implementasi SSO pada login admin dan super admin, membuat Api untuk mengatur admin, <i>change password</i> , dan <i>logout</i> .
6 (19 Juli – 23 Juli)	Membuat API untuk mengatur <i>File Library</i> , <i>Image Library</i> , <i>Article Category</i> , dan <i>Article</i> .

7 (26 Juli – 30 Juli)	Membuat API untuk Mengatur <i>Article Content</i> , <i>Article Comment Moderation</i> , dan <i>Payment Method</i> .
8 (2 Agustus – 6 Agustus)	Membuat API untuk mengatur <i>Bank Account</i> , <i>Courier</i> , <i>Courier Note</i> , dan <i>Expedition Note</i> .
9 (9 Agustus – 13 Agustus)	Membuat API untuk mengatur <i>Expedition Export Data</i> dan <i>Pickup Note</i> .
10 (16 Agustus – 20 Agustus)	Membuat API untuk mengatur <i>import shipping expedition</i> .
11 (23 Agustus – 27 Agustus)	Membuat API untuk mengatur <i>Expedition Coverage</i> , <i>FAQ Subcategory</i> , <i>FAQ Category</i> , dan <i>FAQ</i> .
12 (30 Agustus – 3 september)	Membuat API untuk mengatur <i>Contact Address</i> , <i>Social Media Link</i> , dan <i>Register Popup</i> .

### 3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang yang dilakukan di PT Satu Resolusi Optima yang mengerjakan untuk membuat API admin dan super-admin untuk AXL Parfume yang dilakukan secara *Work From Home* terdapat beberapa rincian sebagai berikut ini:

### **3.3.1 Rincian Pengerjaan Minggu ke-1**

#### **3.3.1.1 Membuat *Website Weather* dengan Menggunakan Laravel**

Pada minggu pertama, mahasiswa diberikan tugas untuk membuat website secara bebas. Mahasiswa memutuskan untuk membuat website melihat cuaca. Hal ini dilakukan untuk melihat kemampuan mahasiswa dalam menggunakan teknologi laravel. Mahasiswa mendapatkan data cuaca dari *website* Openweathermap.org dengan memanggil API yang dimana akan menampilkan nama kota atau negara, temperatur cuaca, arah angin, kekuatan angin, dan pukul jam. Mahasiswa juga membuat fitur menyimpan, memperbarui, dan menghapus default lokasi atau lokasi favorit di dalam database sehingga ketika pengguna membuka website tidak perlu mencari ulang negara atau kota.

#### **3.3.1.2 Review**

Hari Jumat pada tanggal 18 Juni 2021, pembimbing lapangan melakukan *review* terhadap *website* dan *code* yang dikerjakan oleh mahasiswa. Hasil dari review bahwa mahasiswa sudah memahami dasar laravel.

### **3.3.2 Rincian Pengerjaan Minggu ke-2**

#### **3.3.2.1 Mempelajari Lumen dengan Membuat Web API Artikel Sederhana**

Pada minggu ke dua, mahasiswa diberikan tugas oleh pembimbing lapangan untuk mempelajari teknologi lumen dan membuat Web API Artikel sederhana dengan konsep MVC. Tugas yang diberikan yaitu mahasiswa dapat melakukan CRUD (*Create, Read, Update, Delete*) pada artikel, membuat register, login, logout, implementasi bearer token atau otorisasi untuk melakukan penambahan, memperbarui, dan hapus artikel, dan mahasiswa dapat implementasi AWS S3 untuk menyimpan foto. Mahasiswa juga mempelajari penggunaan gitlab dan AWS EC2 untuk upload API ke server.

#### **3.3.2.2 Review**

Pada tanggal 25 Juni bertepatan hari Jumat, pembimbing lapangan melakukan review terhadap API yang dibuat oleh mahasiswa. Dari hasil review, mahasiswa dinyatakan bahwa sudah memahami konsep MVC, CRUD, AWS, Gitlab, dan Otorisasi.

### **3.3.3 Membuat Web API E-Commerce sederhana dengan Laravel**

Pada minggu ke tiga, setelah mempelajari lumen dan memahami konsep MVC dan API. Mahasiswa diberikan tugas berkelompok dengan

orang magang selain mahasiswa. Mahasiswa bekerja dengan total dua orang termasuk mahasiswa untuk membuat Web API E-Commerce menggunakan Laravel dengan konsep MVC. Dari hasil pembelajaran mahasiswa pada minggu ke dua diharapkan mahasiswa dapat melakukan pengerjaan Web API E-Commerce dengan baik. Tugas pada minggu ketiga ini mahasiswa membuat API yang dapat *login* dan *register* untuk pengguna mengatur akun bank, mengatur profile pada halaman pengguna dan halaman admin, akun bank, dan dapat mengatur alamat untuk admin dan pengguna yang dimana admin dapat melakukan CRUD sedangkan pengguna dapat melakukan RU serta mahasiswa juga membuat API transaksi untuk admin dan pengguna.

### **3.3.4 Rincian Pengerjaan pada Minggu ke-4**

#### **3.3.4.1 Menyelesaikan Web API E-Commerce**

Pada minggu ke empat mahasiswa menyelesaikan API E-Commerce yang belum diselesaikan pada minggu ke tiga, mahasiswa memperbaiki rute url API pada alamat, menambahkan beberapa otorisasi yang hanya dapat di akses untuk admin, melakukan perubahan *output json* pada API alamat, menambah beberapa validasi yang salah dan kurang serta memberikan *status code* pada *output json*.

### **3.3.4.2 Review**

Pada tanggal 9 Juli 2021, pembimbing lapangan melakukan *review* terhadap API yang telah dibuat dan menyatakan bahwa mahasiswa telah menyelesaikan tugas yang diberikan.

### **3.3.5 Rincian Pengerjaan pada Minggu ke-5**

#### **3.3.5.1 Implementasi SSO pada Login Admin dan Super Admin**

Setelah memahami dan mempelajari laravel dengan baik pada minggu sebelumnya, mahasiswa mengerjakan proyek sesungguhnya dari klien AXL Parfume. Tugas pertama yang diberikan oleh pembimbing lapangan untuk mahasiswa yaitu menerapkan sistem SSO dari onero solution ke AXL Parfume. Sistem SSO sudah tersedia oleh onero solution dan menghubungkan halaman login admin dan super-admin AXL Parfume ke SSO onero solutions yang nantinya dari api sso akan memberikan kredensial untuk AXL Parfume jika berhasil login. Berikut merupakan contoh *output* dari SSO API login di super-admin bahwa berhasil login akan mendapatkan userToken, userId, dan roleCode seperti pada gambar 3.2.

```
{
  "userToken": [
    "VD8nZVdAZ41TRFDnzFry"
  ],
  "userId": [
    "A161019002"
  ],
  "roleCode": [
    "SA"
  ]
}
```

Gambar 3.1 Output SSO Login

### 3.3.5.2 Membuat API untuk Mengatur Admin

API yang dapat mengatur admin seperti membuat akun admin, edit informasi akun admin, menampilkan akun admin dalam bentuk *list* dan *detail* serta dapat menghapus akun admin.

```
{
  "id": "A60f14a56335eb",
  "role": "Admin",
  "email": "",
  "name": "Admin Onero",
  "username": "onerotest123",
  "updatedAt": "2021-07-16 16:27:03",
  "active": 1
}
```

Gambar 3.2 List Akun Admin

### 3.3.5.3 Membuat API untuk *Change Password*

Pada gambar 3.3 menunjukkan *field* yang dibutuhkan untuk melakukan pergantian password dengan mengisi password sekarang dan password baru.



KEY	VALUE
<input checked="" type="checkbox"/> password	admin
<input checked="" type="checkbox"/> newPassword	onerolevel100
<input checked="" type="checkbox"/> confPassword	onerolevel100
Key	Value

*Gambar 3. 3 Contoh Change Password*

### 3.3.5.4 Membuat API untuk *Logout*

Dalam pembuatan fungsi API *logout* yang akan digunakan untuk admin dan super admin dengan cara menghapus token yang tersedia. Apabila token admin atau super admin terhapus maka admin dan super admin tidak dapat mengakses halaman tersebut dan butuh login kembali.

```
function logout(Request $request){
    $token = (string) $request->bearerToken();
    $this->Token->removeToken($token);
    return response()->json();
}
```

*Gambar 3. 4 Contoh Logout*

## 3.3.6 Rincian Pengerjaan pada Minggu ke-6

### 3.3.6.1 Membuat API untuk Mengatur *File Library*

Pada pembuatan API untuk menyimpan file dapat dilakukan di halaman admin dan super admin dengan fungsi yang sama akan tetapi perbedaan otoritas seperti pada otoritas admin hanya dapat menambah dan menghapus file sedangkan otoritas super admin dapat melakukan

apapun yang dilakukan admin ditambah dapat *update* file. Pada gambar 3.5 merupakan contoh dari *list* file.

```
1 {
2   "id": "F1611f1a433b826",
3   "title": "janji",
4   "url": "https://research-onoero.s3.ap-southeast-1.amazonaws.com/axl-parfume-remake/file/28f6e3b764e4f26231ff0cc5b8b68865.jpg",
5   "lastUpdate": "2021-08-28 09:58:11",
6   "show": "1"
7 }
```

Gambar 3. 5 List File Library

### 3.3.6.2 Membuat API untuk Mengatur *Image Library*

Pada gambar 3.6 merupakan API yang dapat mengatur foto dalam sebuah library. Otorisasi admin hanya dapat melihat seluruh foto, menambahkan foto dan menghapus foto sedangkan super admin dapat melakukan semua hal yang dilakukan oleh admin ditambah dapat untuk update informasi pada foto.

```
1 {
2   "data": [
3     {
4       "id": "IMG611f17a92a88c",
5       "title": "aaaaaa",
6       "url": "https://research-onoero.s3.ap-southeast-1.amazonaws.com/axl-parfume-remake/image/image/gelas.jpg",
7       "lastUpdate": "2021-08-28 09:47:05",
8       "show": "1"
9     },
10    {
11      "id": "IMG6183af94f8ca6",
12      "title": "Killua",
13      "url": "https://research-onoero.s3.ap-southeast-1.amazonaws.com/axl-parfume-remake/image/image/wanted-killua.jpg",
14      "lastUpdate": "2021-08-05 10:56:07",
15      "show": "1"
16    }
17  ],
18 }
```

Gambar 3. 6 List Image Library

### 3.3.6.3 Membuat API untuk Mengatur *Article Category*

Sebelum melakukan pembuatan API untuk artikel mahasiswa mengerjakan pembuatan API untuk kategori artikel yang nanti dapat digunakan di API artikel. Admin dapat melihat *list*,

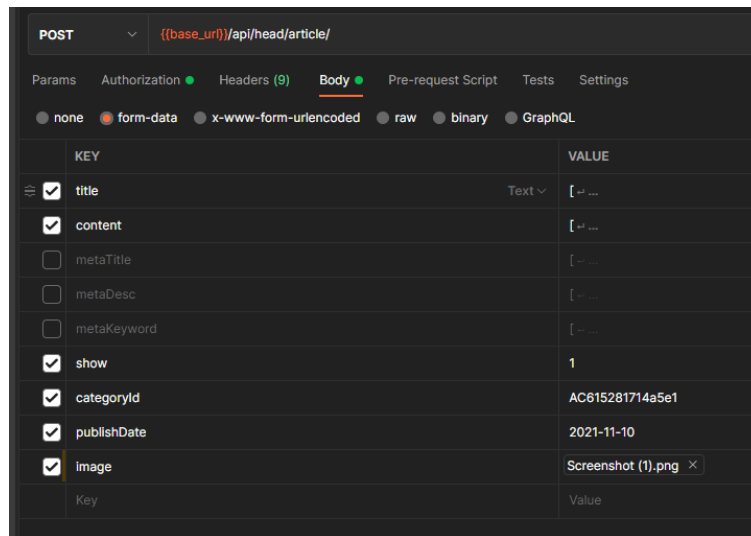
menambahkan, memperbarui, menghapus, menyembunyikan atau menampilkan kategori artikel serta dapat membuat *default* kategori untuk artikel. Pada gambar 3.7 merupakan contoh dari *list* kategori artikel.

```
{
  "id": "AC615281714a5e1",
  "title": "Tips",
  "permalink": "tips",
  "lastUpdate": "2021-09-28 09:44:01",
  "show": "1"
}
```

Gambar 3. 7 List Kategori Artikel

#### 3.3.6.4 Membuat API untuk Mengatur *Article*

Setelah mahasiswa menyelesaikan API untuk kategori artikel maka selanjutnya mahasiswa melanjutkan pembuatan artikel. Admin dapat melihat list, menambahkan, memperbarui, menghapus, menyembunyikan atau menampilkan artikel. Pada gambar 3.8 merupakan contoh untuk melakukan penambahan artikel yang dimana terdapat beberapa field yang *required* dan tidak *required* serta sudah mendukung SEO dan *MultiLanguage* yang terdapat pada *field title, content, meta title, meta description, dan meta keyword*.



Gambar 3. 8 Contoh API Menambahkan Artikel

Pada gambar 3.9 merupakan contoh *list* dari artikel yang dapat dilihat oleh admin.

```

"data": [
  {
    "id": "A6156d34b0c767",
    "title": "Deodorant Terbaik dan Berkualitas Hanya Di AXL Parfume",
    "categoryId": "AC615281714a5e1",
    "categoryName": "Tips",
    "permalink": "deodorant-terbaik-dan-berkualitas-hanya-di-axl-parfume",
    "publishDate": "2021-10-01 16:20:43",
    "lastUpdate": "2021-10-01 16:22:19",
    "show": "1"
  },

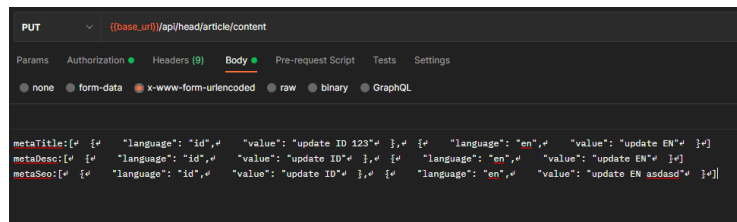
```

Gambar 3. 9 List Artikel

### 3.3.7 Rincian Pengerjaan pada Minggu ke-7

#### 3.3.7.1 Membuat API untuk Mengatur *Article Content*

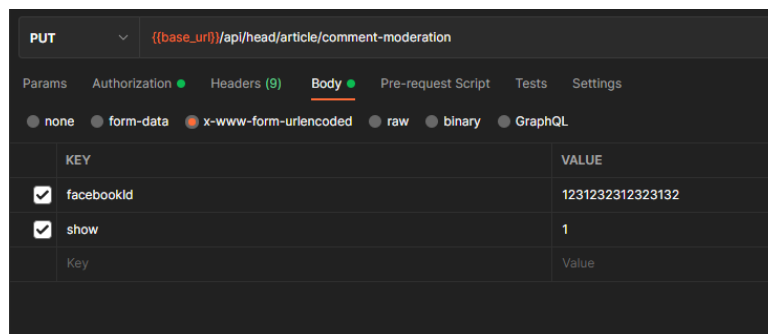
Pada gambar 3.10 merupakan contoh dari konten artikel yang dimana admin hanya dapat memperbarui konten artikel. Konten artikel mendukung MultiLanguage berbahasa inggris dan indonesia yang diperlukan untuk optimasi SEO pada konten artikel.



Gambar 3. 10 Contoh Konten Artikel

### 3.3.7.2 Membuat API untuk Mengatur *Article Commnet Moderation*

Dan modul terakhir yang digunakan untuk keperluan arikel adalah moderasi komen. Moderasi komen menggunakan facebook apps id seperti pada gambar 3.11.



Gambar 3. 11 Contoh Article Comment Moderation

### 3.3.7.3 Membuat API untuk Mengatur *Payment Method*

Dalam pembuatan API untuk metode pembayaran menggunakan pihak ketiga yaitu ipay88 yang berfungsi menjembatani dan verifikasi pembayaran yang dilakukan untuk pembelian barang dan *deposit point* secara otomatis tanpa perlu verifikasi pembayaran secara manual oleh admin. Metode pembayaran yang digunakan yaitu metode pembayaran otomatis seperti kartu kredit, virtual account, ovo, gopay,

dan lain-lainnya. Dalam otorisasi hanya super admin yang dapat melakukan penambahan, memperbarui, melihat, dan menghapus metode pembayaran yang tersedia sedangkan admin hanya dapat melakukan pembaruan informasi metode pembayaran dan menampilkan atau sembunyikan metode pembayaran. Pada gambar 3.12 merupakan contoh dari metode pembayaran dan terdapat beberapa *field* yang dibutuhkan oleh metode pembayaran seperti *name*, *fee active*, *fee*, *order*, *deposit*, dan lain-lainnya.

```
{
  "id": "PM611a3dd59b8f3",
  "name": "BNI Virtual Account",
  "active": 1,
  "feeType": 0,
  "feeLabel": "50.00%",
  "fee": "50.00",
  "order": 1,
  "deposit": 1,
  "lastUpdate": "2021-09-20 13:11:28",
  "show": 1
},
```

Gambar 3. 12 Contoh Metode Pembayaran

### 3.3.8 Rincian Pengerjaan pada Minggu ke-8

#### 3.3.8.1 Membuat API untuk Mengatur *Bank Account*

Dalam pembuatan API ini digunakan untuk bank transfer yang tidak didukung oleh ipay88 dan menghemat dapat *fee* karena tidak menggunakan pihak ketiga. Pembayaran bank transfer dilakukan secara manual dari upload bukti hingga verifikasi pembayaran oleh admin.

Pada gambar 3.13 merupakan contoh dari akun bank yang terdapat informasi yang akan digunakan untuk melakukan pembayaran.

```
{
  "id": "B4610772e1a67cf",
  "bankName": "ALEXANDER SATRYO WIBOWO",
  "bankAccountName": "BCA",
  "bankAccountNumber": "5320102788",
  "imageUrl": "https://research-onoero.s3.ap-southeast-1.amazonaws.com/axl-parfume-zenake/image/bank/6128abdd6595e-bankaccount-image.png",
  "imageResizedUrl": "https://research-onoero.s3.ap-southeast-1.amazonaws.com/axl-parfume-zenake/image/bank/6128abdd6595e-bankaccount-image_thumb.png",
  "lastUpdate": "2021-09-17 09:13:26",
  "show": 1
}
```

*Gambar 3.13 Contoh Akun Bank*

### **3.3.8.2 Membuat API untuk Mengatur Courier**

Pada API kurir ada beberapa modul yang akan dikerjakan. Akan dibahas mulai dari super admin yang dimana super admin hanya dapat mengatur jenis pengiriman yang tersedia seperti tambah, hapus, perbarui, lihat, dan mengatur jangkauan kurir seperti pada gambar 3.14 karena kasus AXL Parfume menjual parfum yang mudah terbakar oleh karena itu tidak semua kurir mengirimkan barang tersebut ke daerah yang cukup jauh dengan melihat resikonya sedangkan admin hanya dapat melihat jenis pengiriman dan jangkauannya. Pada gambar 3.15 merupakan contoh kurir yang digunakan oleh AXL Parfume yang terdapat *id*, *name*, *imageUrl*, *lastupdate*, dan, *show*.

```
"data": [  
  {  
    "provinceId": "P19072300001",  
    "name": "Bali",  
    "show": 1  
  },  
  {  
    "provinceId": "P19072300002",  
    "name": "Bangka Belitung",  
    "show": 0  
  },  
]
```

Gambar 3. 14 Contoh Jangkauan Kurir

```
"data": [  
  {  
    "id": "C611f77014ac6c",  
    "name": "Jalur Nugraha Ekakurir (JNE) edit",  
    "imageUrl": "https://research-onero.s3.ap-sout",  
    "lastUpdate": "2021-08-27",  
    "show": 0  
  },  
]
```

Gambar 3. 15 Contoh Kurir

### 3.3.8.3 Membuat API untuk Mengatur *Courier Note*

Setelah pembuatan API untuk kurir selanjutnya mahasiswa membuat API catatan pengiriman yang berguna untuk pelanggan mengetahui informasi atau catatan dari pengiriman kurir. Pada gambar 3.16 merupakan contoh dari catatan kurir yang sudah mendukung MultiLanguage berbahasa indonesia dan inggris.



```

"title": [
  {
    "language": "ID",
    "value": "Pilih jasa pengiriman untuk pulau JAWA dan luar pulau JAWA"
  },
  {
    "language": "EN",
    "value": "Using Logistic J&T dan MAHANA only for shipment to java island & Using JNE for shipment to the outside of java island to get a bigger ratio shipped to the destination"
  }
],
"note": [
  {
    "language": "ID",
    "value": "Jasa Logistic J&T dan MAHANA hanya untuk pengiriman pulau JAWA.\nGunakan kurir JNE REGULAR untuk pengiriman luar pulau jawa rasio sampai lebih besar"
  },
  {
    "language": "EN",
    "value": "Using Logistic J&T dan MAHANA only for shipment to java island \nUsing JNE for shipment to the outside of java island to get a bigger ratio shipped to the destination"
  }
],

```

*Gambar 3. 16 Contoh Catatan Kurir*

#### **3.3.8.4 Membuat API untuk Mengatur Expedition Note**

Pada gambar 3.17 merupakan contoh API dari catatan ekspedisi yang berbeda dari catatan kurir yang dimana catatan ekspedisi memberikan informasi pengiriman yang menggunakan ekspedisi perusahaan AXL Parfume yang terdapat minimal berat.

```

"note": [
  {
    "language": "ID",
    "value": "Kami Akan Menginformasikan Mengenai Biaya Pengiriman Melalui Telepon.\n\n6 lusin.\n- Daerah lain seperti Sumatera, Kalimantan, Sulawesi dengan minima
  },
  {
    "language": "EN",
    "value": "We will inform the information about shipping price by phone call.\n\ndozen.\n- Other regions such as Sumatera, Kalimantan, Sulawesi with a minimum
  }
],
"masterId": "ACLT610a2ea4232af",
"id": "EN610a2ea3efc42",
"langId": "L6101110a29fb7",
"setting": "1",
"weight": "1",
"updatedAt": "2021-10-08 15:39:11"

```

Gambar 3. 17 Contoh Catatan Ekspedisi

### 3.3.9 Rincian Pengerjaan pada Minggu ke-9

#### 3.3.9.1 Membuat API untuk Mengatur *Expedition Export Data*

Pada pengerjaan API ekspor ekspedisi data yang merupakan daftar harga dan wilayah. Hasil ekspor akan berbentuk xlsx seperti pada gambar 3.18.

1	Provinsi	Kota	Kecamatan	Harga
2	Bali			10000
3	Bali	Badung		0
4	Bali	Badung	Abiansem	0
5	Bali	Badung	Kuta	0
6	Bali	Badung	Kuta Selat	0
7	Bali	Badung	Kuta Utara	0
8	Bali	Badung	Mengwi	0
9	Bali	Badung	Patang	0

Gambar 3. 18 Contoh Hasil Ekspor Data Ekspedisi

### 3.3.9.2 Membuat API untuk Mengatur *Pickup Note*

Pada gambar 3.19 merupakan contoh dari API catatan pengambilan barang yang berguna untuk memberi informasi pada pelanggan jam kurir mengambil dan mengirimkan pesanan. Pada catatan pengambilan barang sudah mendukung MultiLanguage berbahasa indonesia dan inggris.

```

[15]
  "note": [
    {
      "language": "ID",
      "value": "Untuk pengambilan pesanan, bisa diambil di alamat berikut:"
    },
    {
      "language": "EN",
      "value": "Orders can be picked up at the following address:"
    }
  ],
  "masterId": "ACL611228b16b160",
  "id": "PN611228b1405fd",
  "langId": "L6101110a29fb7",
  "setting": 0,
  "updatedAt": "2021-10-11 21:40:12"
[16]

```

Gambar 3. 19 Contoh API *Pickup Note*

### 3.3.10 Rincian Pengerjaan pada Minggu ke-10

#### 3.3.10.1 Membuat API untuk Mengatur *Import Shipping Expedition*

Pada API impor data pengiriman yang berisi wilayah dan harga melalui file excel seperti pada gambar 3.20 yang dimana harga

mengikuti parent seperti tidak ada harga untuk kecamatan maka mengikuti harga kota dan provinsi.

Bali	Buleleng		120000
Bali	Buleleng	Banjar	0
Bali	Buleleng	Buleleng	0
Bali	Buleleng	Busungbiu	0
Bali	Buleleng	Gerokgak	0
Bali	Buleleng	Kubutamba	0
Bali	Buleleng	Sawan	0
Bali	Buleleng	Seririt	0
Bali	Buleleng	Sukawada	0

Gambar 3. 20 Contoh Import Shipping Expedition

### 3.3.11 Rincian Pengerjaan pada Minggu ke-11

#### 3.3.11.1 Membuat API untuk Mengatur *Expedition Coverage*

Pada gambar 3.21 menunjukkan API dari jangkauan ekspedisi perusahaan.

```
{
  "provinceId": "P19072300004",
  "name": "Bengkulu",
  "show": 0
},
{
  "provinceId": "P19072300005",
  "name": "DI Yogyakarta",
  "show": 0
},
```

Gambar 3. 21 Contoh Jangkauan Ekspedisi

### 3.3.11.2 Membuat API untuk Mengatur *FAQ Subcategory*

Pembuatan API untuk FAQ subkategori dilakukan duluan agar dapat digunakan di *parent* nya yaitu FAQ kategori dan FAQ. Pada gambar 3.22 merupakan contoh *list* pada FAQ Subkategori yang dimana terdapat *id*, *title*, *lastUpdate*, dan *show*. FAQ Subkategori dapat dilakukan oleh admin yang dimana admin dapat melihat *list*, detil, melakukan pengurutan, menambah, memperbarui, menghapus, dan menampilkan atau menyembunyikan FAQ Subkategori.

```
{
  "data": [
    {
      "id": "FAQSC614c1b484e3ac",
      "title": "Baju Kemeja ID 31 ",
      "lastUpdate": "2021-09-23 13:14:32",
      "show": 0
    }
  ],
  "pagination": {
    "currentPage": "1",
    "limit": "5",
    "totalData": 1,
    "links": [
      {
        "type": "number",
        "text": 1,
        "active": 1
      }
    ]
  }
}
```

Gambar 3. 22 Contoh List *FAQ SubCategory*

### 3.3.11.3 Membuat API untuk Mengatur *FAQ Category*

Pada gambar 3.23 merupakan contoh dari list FAQ kategori yang dimana memiliki FAQ Subkategori. FAQ Kategori dapat dilakukan oleh admin yang dimana admin dapat melihat list, detil, melakukan pengurutan, menambah, memperbarui, menghapus, dan menampilkan atau menyembunyikan FAQ kategori.

```
"data": [  
  {  
    "id": "FAQC614c1af99cc54",  
    "title": "Baju ID 34 ",  
    "lastUpdate": "2021-09-23 13:13:13",  
    "show": "1"  
  },  
  {  
    "id": "FAQC614d49683c878",  
    "title": "Kemeja",  
    "lastUpdate": "2021-09-24 10:43:36",  
    "show": "1"  
  }  
]
```

*Gambar 3. 23 Contoh List FAQ Category*

### 3.3.11.4 Membuat API untuk Mengatur *FAQ*

Pada gambar 3.24 merupakan contoh dari list FAQ kategori yang dimana memiliki FAQ Kategori dan Subkategori. FAQ dapat dilakukan oleh admin yang dimana admin dapat melihat list, detil, melakukan pengurutan, menambah, memperbarui, menghapus, dan menampilkan atau menyembunyikan FAQ kategori.

```
"data": [  
  {  
    "id": "FQ614c2189a2331",  
    "title": "hnm ID Coba gnti",  
    "lastUpdate": "2021-09-24 15:32:04",  
    "show": 1  
  },  
  {  
    "id": "FQ614c1c6e342f5",  
    "title": "Uniqlo ID 21 ",  
    "lastUpdate": "2021-09-24 10:49:17",  
    "show": 1  
  }  
]
```

*Gambar 3. 24 Contoh FAQ*

### 3.3.12 Rincian Pengerjaan pada Minggu ke-12

#### 3.3.12.1 Membuat API untuk Mengatur *Contact Address*

Pada gambar 3.25 menunjukkan informasi lengkap seperti title, address, email, dan phone agar pengguna dapat mengetahui informasi dasar perusahaan.

```
"id": "CA612c93f3d3fa9",  
"title": "AXL Parfum 3",  
"address": "Jl. Malaka No.29B, RT.6/RW.2, Roa Malaka, Kec.  
"email": "axlparfume@gmail.com",  
"phone": "082178978918",  
"updatedAt": "2021-10-11 21:46:52"
```

*Gambar 3. 25 Contoh API Contact Address*

### 3.3.12.2 Membuat API untuk Mengatur *Social Media Link*

Social media link digunakan untuk memberikan informasi social media perusahaan. Pada API ini mahasiswa membuat dapat mengatur social media link seperti menambah, memperbarui, menghapus, melihat dan menampilkan atau menyembukan. Social media link hanya dapat menggunakan favicon atau image. Tidak bisa menggunakan 2 sekaligus yang dimana *icon 1* merupakan menggunakan *favicon* yang tertera pada gambar 2.26 sedangkan *icon 0* menggunakan *image* yang tertera pada gambar 3.27.

```
"id": "SM614b05031fdbb",  
"link": "faceboook.id",  
"icon": "1",  
"faicon": "ICON FACE",  
"imageUrl": null,  
"imageResizeUrl": null,  
"order": "5",  
"show": 1
```

Gambar 3. 26 Contoh API Social Media Link Menggunakan Favicon



```
"id": "SM612dfcf3aa583",
"link": "instagram.com",
"icon": "1",
"faicon": null,
"imageUrl": "https://research-onero.s3.ap-s
"imageResizeUrl": "https://research-onero.s
"order": "1",
"show": 1
```

Gambar 3. 27 Contoh API Social Media Link Menggunakan Image

### 3.3.12.3 Membuat API untuk Mengatur Register Popup

Pada gambar 3.28 menunjukkan sebuah pesan yang ditampilkan di halaman pengguna ketika pengguna mendaftar akun. Admin dapat mengatur isi dari konten *register popup*.

```
"content": [
  {
    "language": "ID",
    "value": "Kamu sudah jadi member AXL Parfume. Silahkan
  },
  {
    "language": "EN",
    "value": "You have already become AXL Parfume Member.
  }
],
"masterId": "RPLT612c90bbc7d52",
"id": "RP612c90bb9c825",
"langId": "L6101110a29fb7",
"updatedAt": "2021-08-30 15:03:07"
```

Gambar 3. 28 Contoh Register Popup

### **3.4 Kendala yang Dihadapi**

#### **3.4.1 Kendala yang Dihadapi**

Selama melaksanakan kerja magang, mahasiswa beberapa kali menghadapi kendala pada saat *development*, antara lain:

1. Kendala dalam menggunakan produk AWS (*Amazon web services*) sehingga memakan waktu untuk belajar dan mencari solusi untuk *error*.
2. Kesulitan implementasi *multiple language* pada sistem sehingga terdapat banyak kesalahan pada pertama kali karena belum pernah mengimplementasikan sebelumnya.
3. Belum menyesuaikan pembuatan *code* yang sesuai standard yang ditetapkan oleh onero seperti input dan output JSON dan penulisan *code*.

#### **3.4.2 Solusi atas Kendala**

Berdasarkan masalah yang telah dijabarkan diatas terdapat beberapa solusi diantara lainnya yaitu:

1. Mempelajari serta mengimplementasi produk AWS dengan cara melihat dokumentasi yang telah disediakan dan juga dapat bertanya kepada *senior programmer*.
2. Dapat untuk bertanya kepada *senior programmer* untuk meminta penjelasan yang lebih detil dan jelas.
3. Belajar untuk menyesuaikan *format coding* atau beradaptasi yang sesuai dengan standar perusahaan.