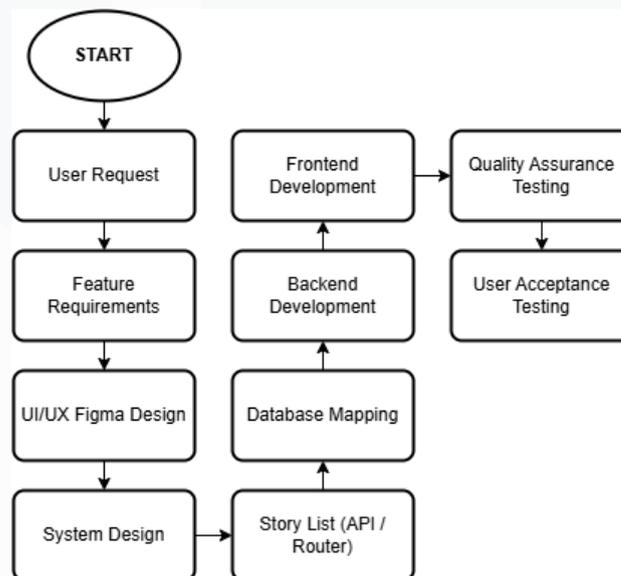


BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

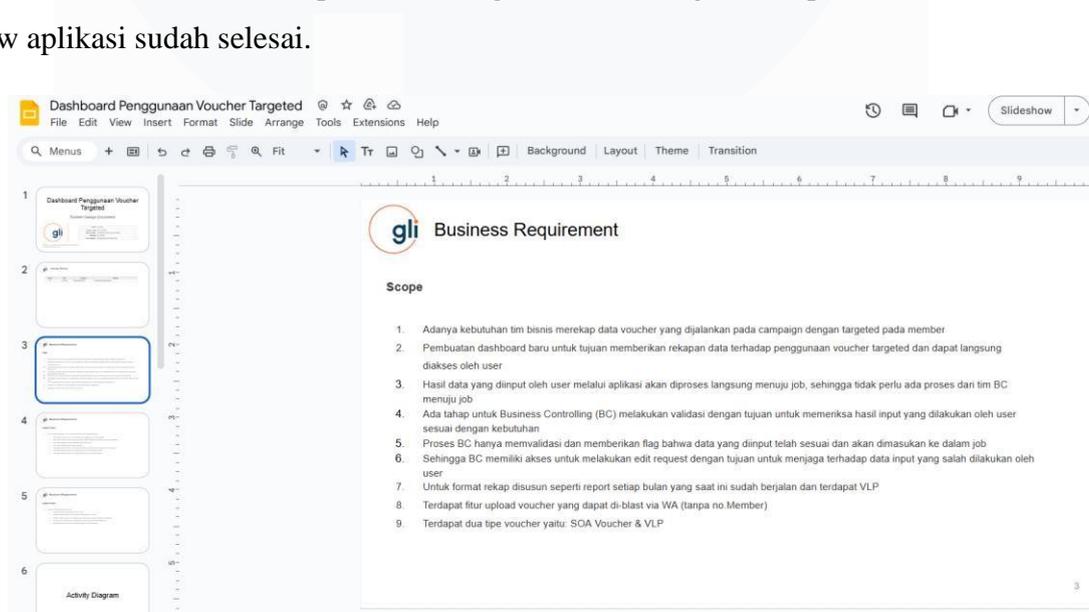
Posisi yang ditempati selama bekerja di PT. Global Loyalty Indonesia adalah Intern IT Corporate dan berada dalam departemen IT Corporate. Dalam IT Corporate, tugas utama adalah sebagai Backend Engineer untuk mengembangkan website internal perusahaan dari sisi API dan database. Selama pelaksanaan magang, Bapak Farhan Purdi selaku supervisor yang membantu dalam membimbing dan mengkoordinasikan tugas apa saja yang harus dikerjakan.



Gambar 3.1 Alur Kerja & Koordinasi.

Gambar 3.1 menunjukkan bagaimana alur sebuah project diterima dan dikerjakan hingga selesai. Dimulai dari permintaan user yang mencakup fitur-fitur dari web atau aplikasi. Kemudian tim UI/UX mulai mengerjakan desain aplikasi sesuai permintaan user. Apabila desain sudah sesuai, maka project manager mulai membuat *System Design*, *Story List*, dan *Database Mapping* yang diberikan kepada Backend dan Frontend Engineer. Gambar 3.2 menunjukkan *System Design* yang dibuat dengan Google Slides. *System Design* meliputi Scope, Business Rules, dan Activity Diagram. Gambar 3.3 merupakan beberapa *Story List* yang dibuat

dengan Google Spreadsheet, sebagai panduan apa saja yang harus dikerjakan. Story List disini dicampur antara Backend (API) dan Frontend (FE) yang dilengkapi juga dengan Business Requirements atau kegunaannya, Priority, Status Dev, Dev Size, QA Size, dan Note. Gambar 3.4 merupakan Database Mapping yang juga dibuat dengan Google Spreadsheet sebagai panduan table database, nama column, tipe data, dan atribut tertentu. Selama pengerjaan project, koordinasi dilakukan dengan menghampiri langsung pihak yang bersangkutan atau mengontak melalui media WhatsApp. Apabila pengerjaan oleh Backend sudah selesai, maka Frontend mulai mengerjakan bagiannya. Backend tetap mengawasi apabila Frontend mengalami masalah dan perlu diperbaiki dari sisi Backend. Setelah Backend dan Frontend sudah selesai, maka aplikasi dilakukan testing oleh QA (Quality Assurance) untuk memastikan tidak ada bug atau error. Jika tidak ada bug atau error lagi dari QA, dilakukan UAT (User Acceptance Testing) untuk menunjukkan kepada user bahwa flow aplikasi sudah selesai.



Gambar 3.2 Contoh System Design.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Platform	User Akses	Business Requirements	Priority	Sprint	Assignee	Status Dev	Dev Size	QA Size	Note
FE		Menampilkan status saat ini				S			
FE		Tersedia button persetujuan untuk validasi				S			
FE		Tersedia button penolakan				S			
FE		Tersedia kolom keterangan untuk alasan penolakan				S			
Validator Master Setting Group Program & Sub Program									
FE		Menampilkan Tabel Group Program & Sub Program yang telah tersedia				On Pr	S		
FE		Menampilkan button untuk menambah Group Program & Sub Program				On Pr	S		
API		User dapat membuat Group Program baru				Done	M		
API		User dapat membuat Sub Program baru di bawah Group Program yang ada				Done	M		
API		User dapat mengelola (menambah, mengubah, dan non-aktif) Group Program				Done	M		
API		User dapat mengelola (menambah, mengubah, dan non-aktif) Sub Program				Done	M		
Validator Master Setting Department									
FE		Menampilkan Tabel Department yang telah tersedia				To Do	S		
FE		Menampilkan button untuk menambah Department				To Do	S		
API		User dapat membuat Department baru				Done	M		
API		User dapat mengelola (menambah, mengubah, dan non-aktif) Department				Done	M		
Validator Master Setting Remark Blast									
FE		Menampilkan Tabel Remark Blast yang telah tersedia				To Do	S		
FE		Menampilkan button untuk menambah Remark Blast				To Do	S		

Gambar 3.3 Beberapa Contoh Story List.

Parameter	Column	Type	Restriction	Not Null
Tanggal Mulai	mvtr_periode_awal	date		TRUE
Tanggal Selesai	mvtr_periode_akhir	date		TRUE
No Juklak VLP / Reward SO	mvtr_id_program	string	_/JKK/GLI-MED/ / _ atau 5 digit mso id	TRUE
Group Program	mvtr_group_program	string	dependent dropdown	TRUE
Sub Program	mvtr_sub_program	string		
Sub Program 1	mvtr_sub_program1	string		FALSE
Sub Program 2	mvtr_sub_program2	string		FALSE
Sub Program 3	mvtr_sub_program3	string		FALSE
Sub Program 4	mvtr_sub_program4	string		FALSE
Dept	mvtr_dept	string	dropdown	TRUE
Mekanisme	mvtr_mekanisme	string		FALSE
Nominal Potongan	mvtr_vcr_amount	integer		FALSE
Min. Belanja	mvtr_min_amount	integer		FALSE
Total Blast	mvtr_total_blast	integer		TRUE
Remark Blast	mvtr_remark_blast	string	dropdown	TRUE

Gambar 3.4 Beberapa Contoh Database Mapping.

3.2 Tugas dan Uraian Kerja Magang

Tabel 3.1 Company Internship Period

Minggu Ke-	Pekerjaan yang dilakukan
1 - 2	Mempelajari FastAPI, template backend perusahaan, dan setup software dan hardware.
3	Mulai mengerjakan project Voucher Targeted dengan mempelajari dan menerapkan Clean Architecture.
4	Mengerjakan project Voucher Targeted untuk router GET dan POST data voucher.

5	Mengerjakan project Voucher Targeted untuk router PUT agar bisa update data voucher yang sudah di POST. Serta router DELETE data voucher.
6	Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Group Program.
7	Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Sub Program.
8	Mengerjakan project Voucher Targeted untuk router GET Departemen dan Remark Blast.
9	Mengerjakan project Voucher Targeted untuk router GET dan POST file.
10	Mengerjakan project Voucher Targeted untuk router GET dan PUT approval status. Setiap data voucher yang di approve atau reject oleh validator maka akan tercatat riwayat statusnya.
11	Mengerjakan project Voucher Targeted untuk implementasi SSO V3 milik perusahaan dan validasi dan decode token.
12	Mengerjakan project Voucher Targeted memperbaiki beberapa bug setelah testing oleh QA dan Frontend.
13	Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Remark Blast.
14	Mengerjakan project Voucher Targeted untuk router GET dan PUT tableau link.
15	Mengerjakan project Voucher implementasi email service. Sistem akan mengirim email ke yang bersangkutan saat ada data voucher baru dibuat, approve atau reject, dan link tableau muncul.

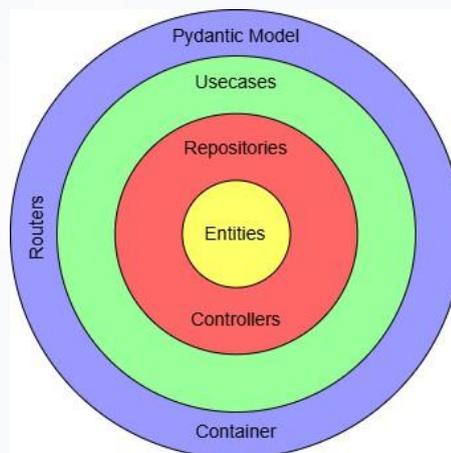
- Mempelajari FastAPI, template backend perusahaan, dan setup software dan hardware.

Pertama-tama, sebelum memulai pengerjaan proyek, dilakukan persiapan software yang diperlukan untuk menyesuaikan kebutuhan perusahaan. Perusahaan mewajibkan setiap perangkat laptop terdapat aplikasi anti-virus bernama eScan Protection Center sesuai SOP-nya. 2 Software utama yang digunakan adalah Visual Studio Code dan DBeaver.

Pada minggu pertama, Project tidak langsung diberikan dan dikerjakan karena masih menunggu konfirmasi dari tim UI/UX dan User. Sembari menunggu, penulis memanfaatkan waktu dengan mempelajari template backend yang sudah disediakan perusahaan. Framework yang digunakan adalah FastAPI, dengan bahasa pemrograman Python. Untuk database menggunakan MySQL dan PostgreSQL. Penulis juga mempersiapkan software DBeaver untuk koneksi database dan mempelajari bagaimana cara kerjanya.

- Mulai mengerjakan project Voucher Targeted dengan mempelajari dan menerapkan Clean Architecture.

Pada minggu ketiga, baru diberikan project pertama bernama Dashboard Voucher Targeted atau singkatnya Voucher Targeted. Project ini mulai dikerjakan pada bulan Agustus hingga bulan November. Database ditentukan oleh supervisor dan akan dirancang seiring waktu menyesuaikan kebutuhan User dan Developer.



Gambar 3.5 Gambaran Umum Clean Architecture [4].

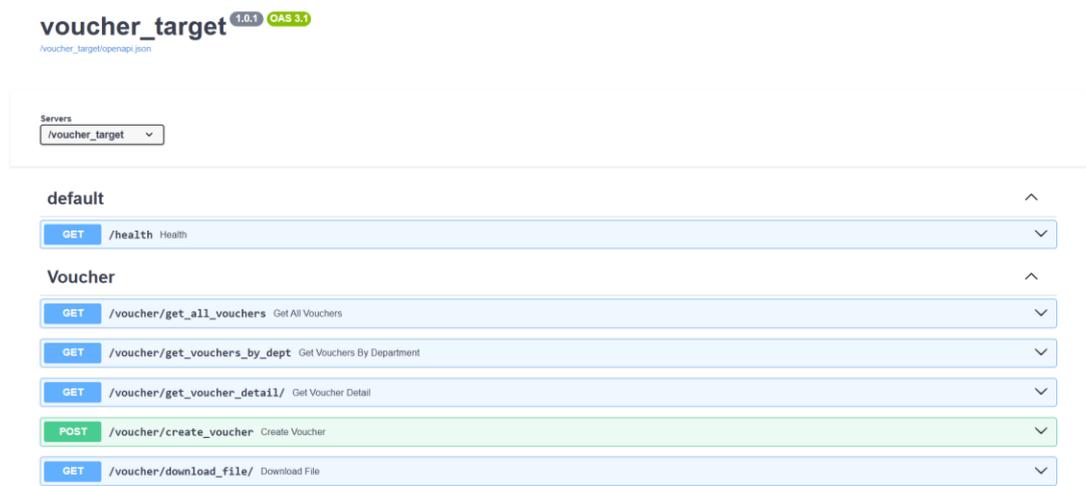
Pengerjaan project ini menerapkan prinsip Clean Architecture yang dimodifikasi sesuai standar perusahaan. Berdasarkan gambar 3.5 dan standar perusahaan, dapat ditarik beberapa komponen penting Clean Architecture yang terdiri atas berikut:

- Entities: Representasi dari table database dalam code yang dibuat.

- Repositories: Tempat query atau logika dilakukan. Bisa dengan menggunakan ORM (Object Relational Mapping) atau query langsung ke table database.
- Usecases: Tempat repositories dipanggil. 1 Usecase dapat memiliki banyak repositories.
- Container: Membungkus repositories dan usecases agar dapat digunakan.
- Pydantic Model: Model yang merepresentasikan bagaimana memberi respond dari dan ke server.
- Routers: Endpoint yang akan dikonsumsi oleh Front-End. Tempat Pydantic Model, Container, Usecases, Repositories, dan Entities digunakan.

Kelebihan dari Clean Architecture adalah adaptif, sehingga mampu untuk dimodifikasi secara fleksibel tanpa harus terpaksa akan aturan tertentu. Selain itu penggunaannya cukup luas untuk project yang memiliki beban workload besar.

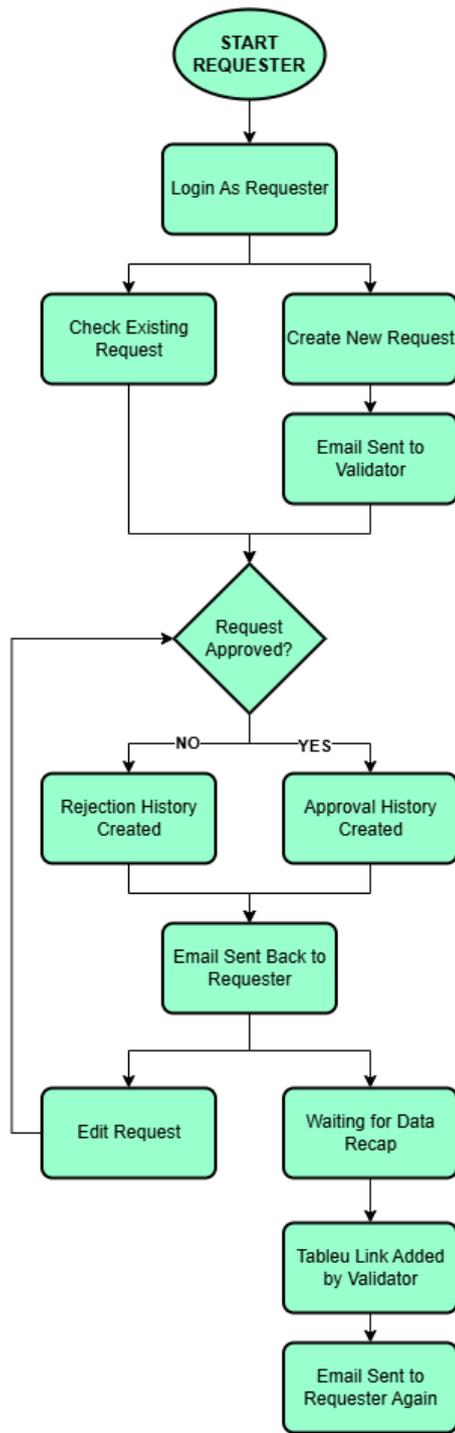
Penulis memulai dengan membuat virtual environment. Virtual environment berfungsi agar hardware atau laptop tidak terbebani dengan package library yang banyak. Untuk mengaktifkan virtual environment adalah dengan command `“./venv/scripts/activate”` pada terminal. Setelah itu dilakukan pip install dalam virtual environment tersebut untuk package library yang dibutuhkan sesuai kebutuhan project. Untuk koneksi ke database, digunakan file `“.env.dev”` untuk menyimpan host, username, password, dan port. Untuk run aplikasi backend dilakukan dengan command `“uvicorn main:app --reload”`. URL Local dari FastAPI adalah <http://127.0.0.1:8000/>. FastAPI menyediakan fitur Swagger UI yang dimana untuk mengaksesnya maka perlu ditambah nama aplikasi dan kata `“docs”`, sehingga URL menjadi http://127.0.0.1:8000/voucher_target/docs. Gambar 3.6 merupakan tampilan Swagger UI sebagai tempat untuk testing apakah router yang dibuat sudah berhasil dan dapat di consume oleh Front-End.



Gambar 3.6 Tampilan Swagger UI.

- Usecase Requester Voucher Targeted

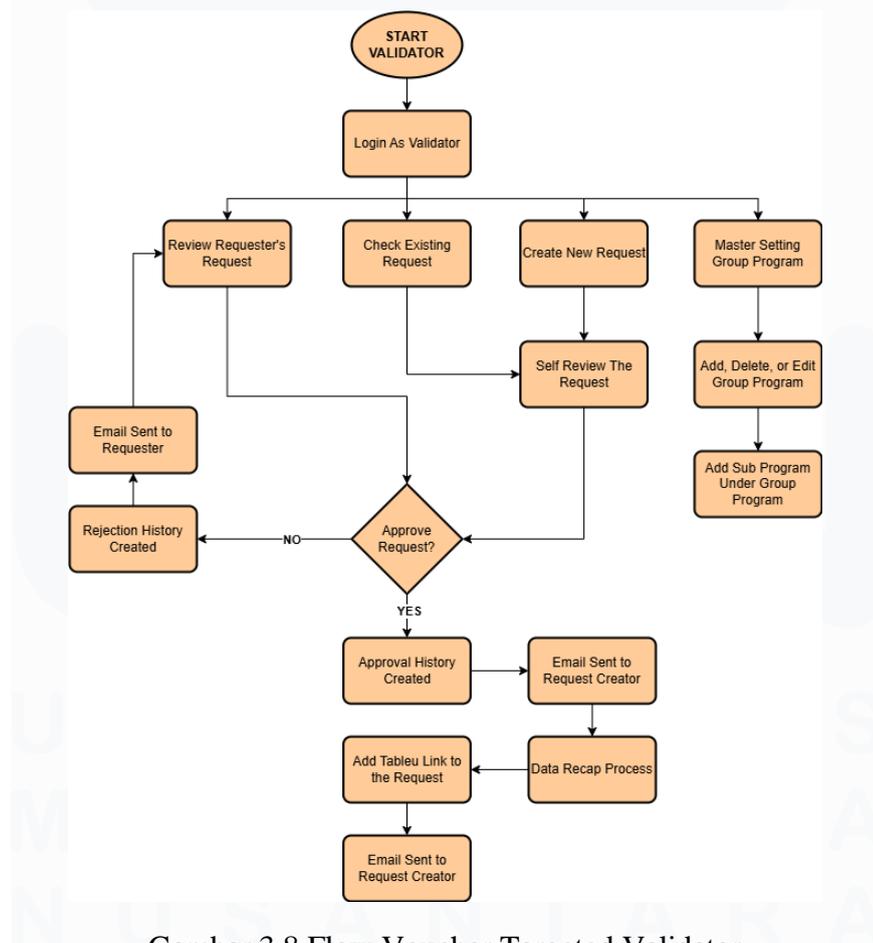
Voucher Targeted memiliki 2 user, yaitu requester dan validator. Gambar 3.7 adalah alur untuk user yang berperan sebagai requester. Fitur sebagai requester adalah membuat request data voucher dan mengecek apakah request sudah di approve atau reject. Apabila requester membuat request baru, maka akan mengirim email ke tim Business Controlling atau validator sebagai tanda bahwa ada request yang perlu diperiksa. Status awal dari request adalah “Menunggu Validasi Businiess Controlling”. Requester dapat memodifikasi data apabila belum di approve atau di reject oleh validator. Apabila request di approve ataupun di reject, keduanya akan mengirim email kembali ke requester dan tercatat history terhadap request tersebut. Dalam kasus request di approve, status akan berubah menjadi “Proses Rekap Data” dan requester akan mendapat notifikasi email jika ada link tableu dari validator.



Gambar 3.7 Flow Voucher Targeted Requester.

- Usecase Validator Voucher Targeted

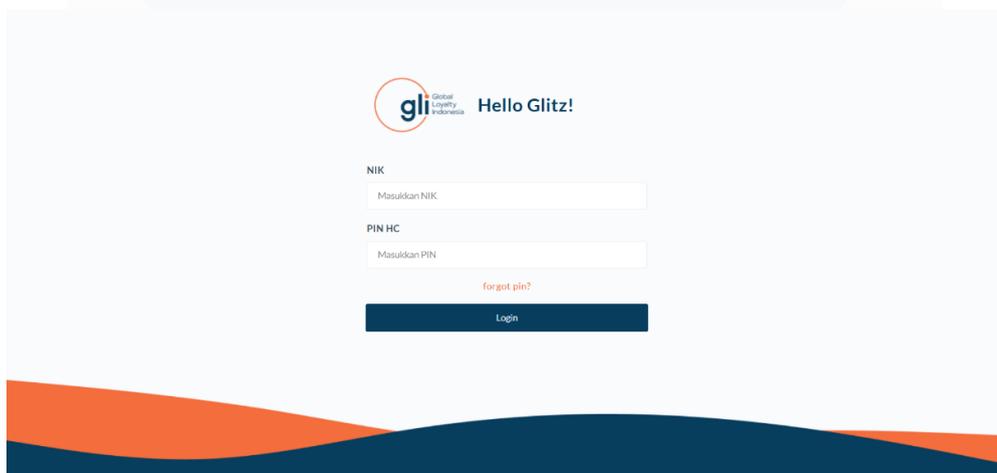
Gambar 3.8 adalah alur untuk yang berperan sebagai validator. Validator memiliki fitur lebih banyak. Selain bisa membuat request baru dan mengecek untuk dirinya sendiri, validator dapat melakukan approve atau reject ke data requester dan bahkan melakukan perubahan pada data itu sendiri. Validator dapat menaruh link menuju tableau sebagai visualisasi data yang sudah disiapkan oleh tim Data Engineer. Fitur lain dari validator adalah Master Setting Group Program, dimana validator dapat menambah, menghapus, atau mengubah Group Program. Group Program adalah opsi yang dapat requester pilih saat akan membuat request baru. Di bawah Group Program ada Sub Program yang juga dapat ditambah, dihapus, atau diubah oleh validator. Sehingga beberapa opsi saat membuat request akan bergantung pada validator.



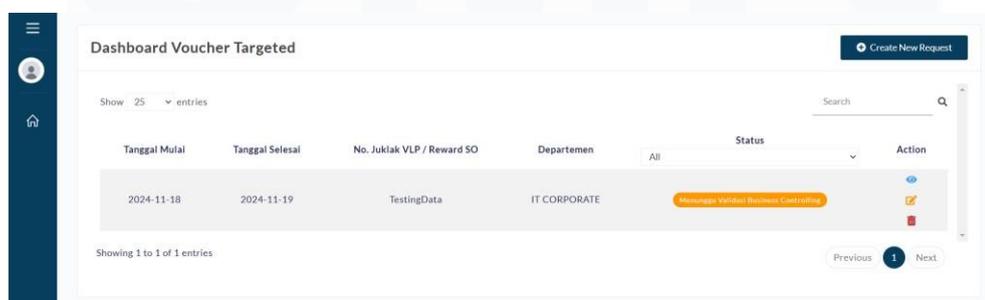
Gambar 3.8 Flow Voucher Targeted Validator.

- Mengerjakan project Voucher Targeted untuk router GET dan POST data voucher.

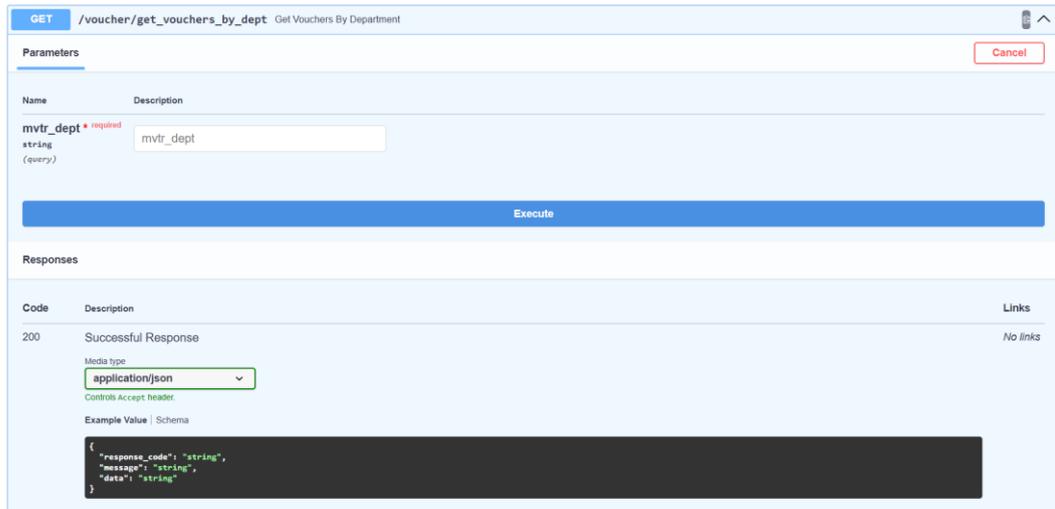
Penulis mulai mengerjakan bagian pertama, yaitu router agar user dapat membuat request data voucher baru dan dapat melihat data yang sudah dibuat. Ada 2 jenis router yang digunakan di halaman utama. Router pertama adalah “Get Vouchers By Department” untuk requester, karena setiap requester hanya dapat melihat data yang sesuai dengan departemennya masing-masing, seperti misalnya IT Corporate hanya dapat melihat sesama IT Corporate. Router kedua adalah “Get All Vouchers” untuk validator agar dapat melihat semua data untuk di approve atau reject. Untuk membedakan antara requester dan validator adalah melalui NIK yang digunakan saat di halaman login. Terdapat juga halaman untuk melihat data voucher yang sudah dibuat secara lebih detail, seperti pada gambar 3.13.



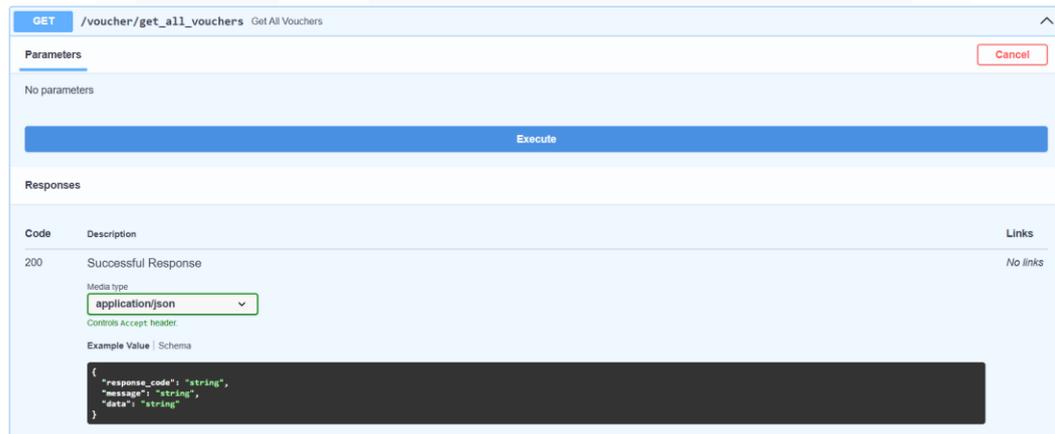
Gambar 3.9 Halaman Login Voucher Targeted.



Gambar 3.10 Halaman Utama Voucher Targeted.



Gambar 3.11 Router Get Vouchers By Department Untuk Requester.

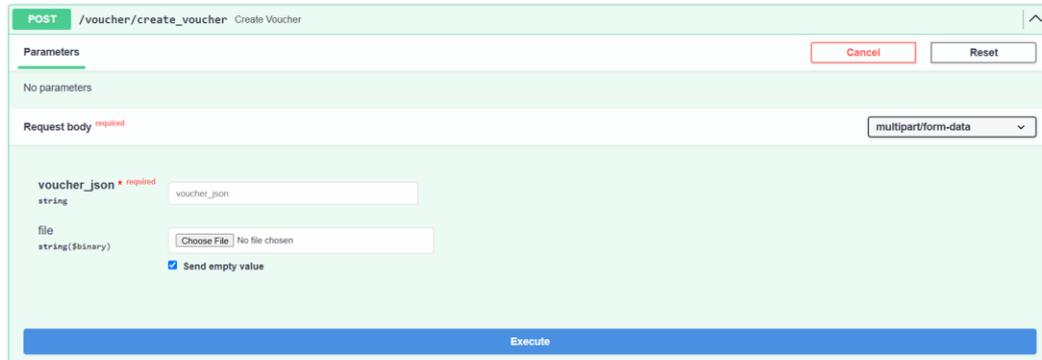


Gambar 3.12 Router Get All Vouchers Untuk Validator.

Gambar 3.13 Halaman Detail Voucher.

Gambar 3.14 adalah halaman create new request dimana user dapat mengisi form sesuai ketentuan. No. Juklak VLP / Reward SO bertindak sebagai Primary Key, sehingga value-nya tidak boleh duplicate dalam database. Untuk departemen, group program, dan sub program adalah opsi dropdown yang pilihannya sudah tersedia dalam database. Sub program dapat ditambah sebanyak 5 dan bersifat opsional. Sementara untuk Remark Blast hanya terdapat 2 opsi, yaitu “ALFAGIFT” dan “NON ALFAGIFT” sehingga dilakukan hardcoded.

Gambar 3.14 Halaman Create New Request.

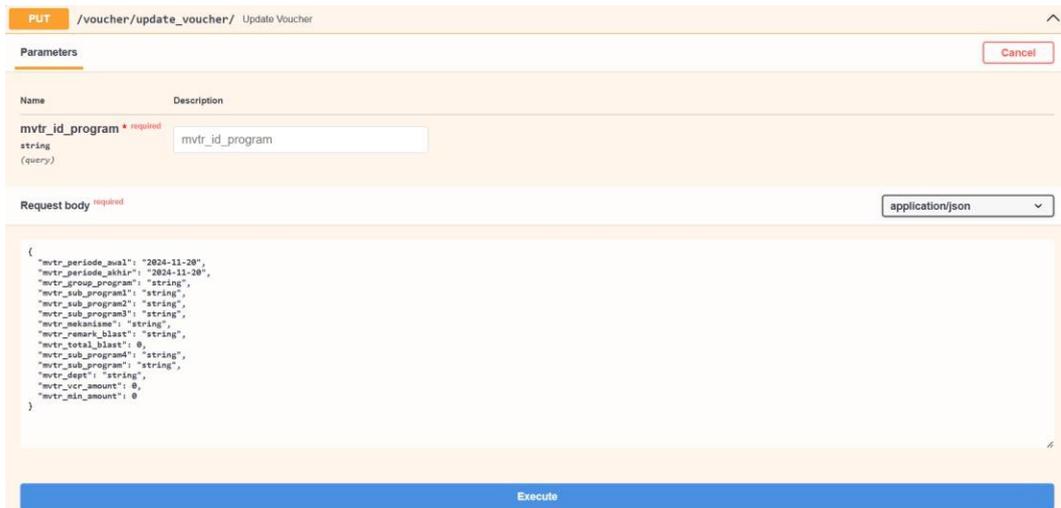


Gambar 3.15 Router Create Voucher.

- Mengerjakan project Voucher Targeted untuk router PUT agar bisa update data voucher yang sudah di POST. Serta router DELETE data voucher.

Penulis kemudian mengerjakan router “Update Voucher” yang memungkinkan user mengubah data voucher. Dengan syarat status request data voucher masih dalam “Menunggu Validasi Business Controlling” atau belum disetujui. Jika request di reject, user masih dapat melakukan perubahan. Apabila user melakukan perubahan pada data yang direject, maka statusnya akan kembali menjadi “Menunggu Validasi Business Controlling”.

Gambar 3.16 Halaman Update Data Voucher



Gambar 3.17 Router Update Voucher.

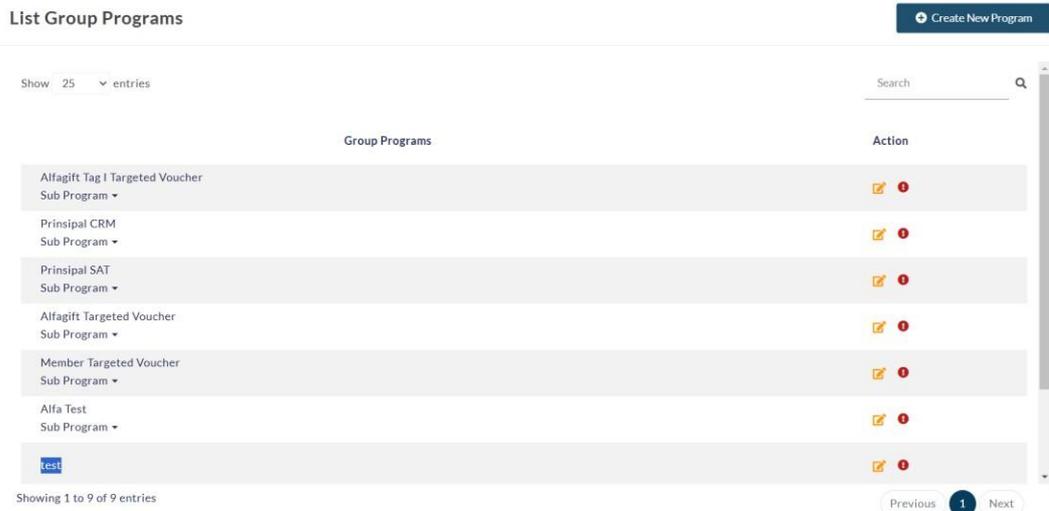
Kemudian juga terdapat router “Delete Voucher” yang penggunaannya sederhana, yaitu untuk menghapus data voucher yang telah dibuat. Dengan syarat data belum diapprove juga.



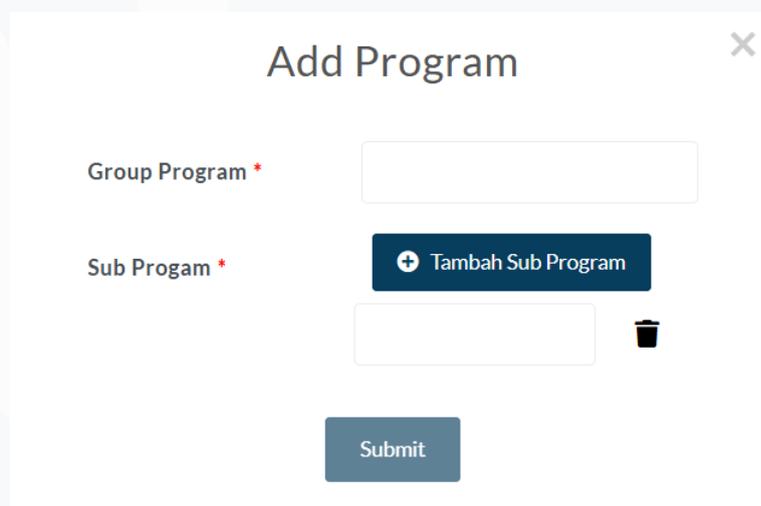
Gambar 3.18 Router Delete Voucher.

- Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Group Program.

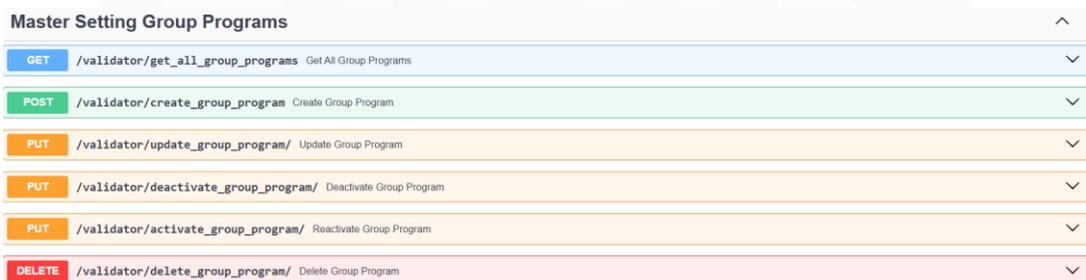
Setelah itu, penulis mengerjakan bagian bernama “Master Setting Group Program”. Ini merupakan bagian khusus untuk validator agar dapat mengatur daftar group program dan sub program yang user pilih saat membuat request data voucher. Setiap group program dapat memiliki banyak sub program. Validator dapat melakukan aksi seperti menambah, mengubah nama, menghapus, dan deactivate program sementara.



Gambar 3.19 Halaman Master Setting Group Program.



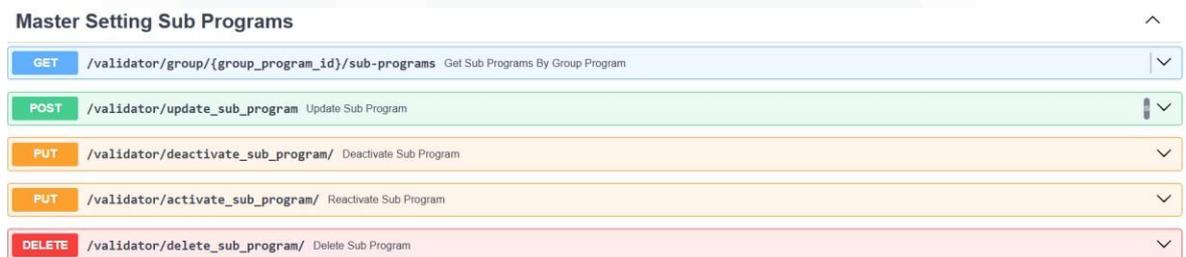
Gambar 3.20 Tampilan Create New Program.



Gambar 3.21 Router Group Program.

- Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Sub Program.

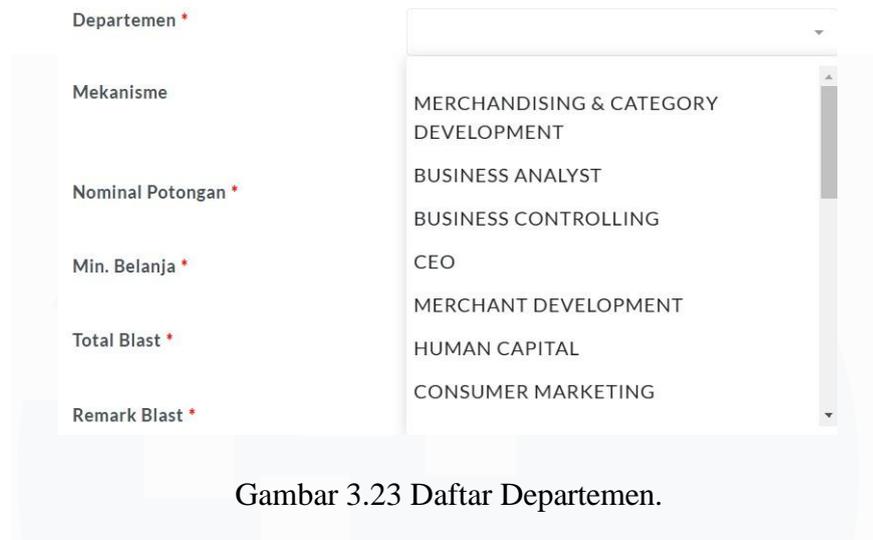
Masih dalam bagian “Master Setting Group Program”, terdapat sub program yang berada di bawah group program tertentu. Validator masih dapat melakukan hal yang sama seperti sebelumnya, yaitu CRUD terhadap sub program.



Gambar 3.22 Router Sub Program.

- Mengerjakan project Voucher Targeted untuk router GET Departemen dan Remark Blast.

Kemudian penulis mengerjakan router “Get All Departemen” agar user dapat memilih departemen saat membuat request data voucher. Namun karena data-data departemen harus diambil dari database yang berbeda, penulis harus membuat koneksi database baru, yaitu database PostgreSQL. Data departemen ini bersifat universal, karena selalu digunakan di hampir setiap aplikasi internal perusahaan. Sementara untuk Remark Blast tidak memiliki table database, karena hanya memiliki 2 value, yaitu “ALFAGIFT” dan “NON ALFAGIFT”. Untuk value Remark Blast dilakukan hardcoded.



Gambar 3.23 Daftar Departemen.



Gambar 3.24 Router Get All Departemen.



Gambar 3.25 Router Get Remark Blast Values.

- Mengerjakan project Voucher Targeted untuk router GET dan POST file.

Fitur berikutnya adalah saat membuat request data voucher, user dapat melakukan upload file Excel. Penulis memodifikasi router “Create Voucher” dengan memanfaatkan library pandas, agar file dapat diupload dan tersimpan di database bersamaan dengan data voucher. Request body yang awalnya berupa object sekarang berubah menjadi string semua. File yang telah di upload juga dapat diubah menggunakan router “Update File” oleh user selama request belum diapprove. File dapat diunduh oleh user dengan router “Download File”.

Upload Voucher

 No file chosen

*Upload file jika voucher diblast melalui
Whatsapp

[Download Template Excel](#)

Gambar 3.26 Fitur Upload File.

POST /voucher/create_voucher Create Voucher

Parameters

No parameters

Request body required

voucher_json * required
string

file
string(\$binary) No file chosen

Send empty value

Gambar 3.27 Router Create Voucher Dimodifikasi.

GET /voucher/download_file/ Download File

Parameters Cancel

Name	Description
vtri_id_program <small>* required</small> string (query)	<input type="text" value="vtri_id_program"/>

Execute

Gambar 3.28 Router Download File.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.29 Router Update File.

- Mengerjakan project Voucher Targeted untuk router GET dan PUT approval status. Setiap data voucher yang di approve atau reject oleh validator maka akan tercatat riwayat statusnya.

Kemudian yang dikerjakan adalah fitur untuk melihat history approval. Setiap data voucher yang di approve atau reject akan tercatat tanggal dilakukannya, NIK validator, dan status accepted atau rejected. Terdapat juga tulisan yang memberi tahu kepada validator kapan request tersebut dibuat. Terdapat 3 router untuk fitur ini, yaitu “Approve And Update Status”, “Reject And Update Status”, dan “Get Approvals”. Jika request diapprove maka status akan menjadi “Proses Rekap Data”.

History Approval
(request dibuat tanggal 22-11-2024)

Tanggal	Approval	Status
No data available in table		

Back Reject Accept

Gambar 3.30 Fitur History Approval.

History Approval
(request dibuat tanggal 15-11-2024)

Tanggal	Approval	Status
2024-11-15	1023053630	Accepted
2024-11-15	1023053630	Rejected

Gambar 3.31 Jika Approve Atau Reject Data.

PUT /voucher/approve_and_update_status Approve And Update Status

Parameters Cancel

Name	Description
mvtr-id-program required	mvtr-id-program

string (header)

Request body required application/json

```
{
  "vtra_nsk": "string",
  "vtra_status": 3,
  "status_update": {
    "mvtr_status_proses": "string"
  }
}
```

Execute

Gambar 3.32 Router Approve And Update Status.

PUT /voucher/reject_and_update_status Reject And Update Status

Parameters Cancel

Name	Description
mvtr-id-program required	mvtr-id-program

string (header)

Request body required application/json

```
{
  "vtra_nsk": "string",
  "vtra_status": 4,
  "status_update": {
    "mvtr_status_proses": "string"
  }
}
```

Execute

Gambar 3.33 Router Reject And Update Status.

GET /voucher/get_approvals Get Approvals

Parameters Cancel

Name	Description
vtra_id_program required	vtra_id_program

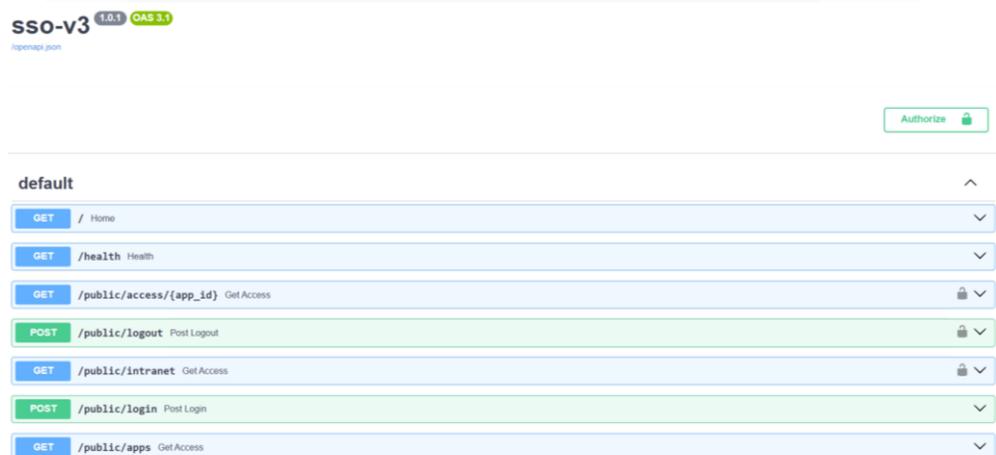
string (query)

Execute

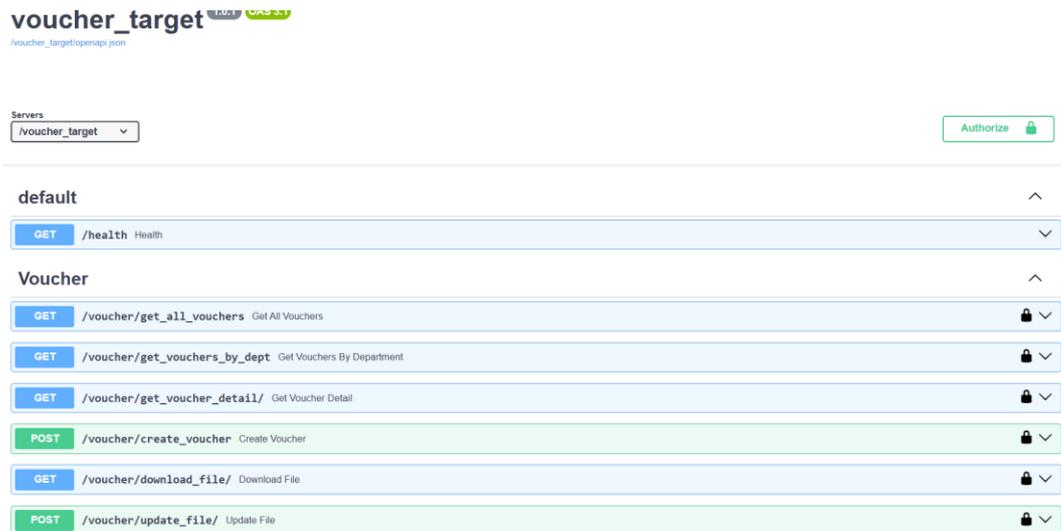
Gambar 3.34 Router Get Approvals.

- Mengerjakan project Voucher Targeted untuk implementasi SSO versi 3 milik perusahaan dan validasi dan decode token

PT. Global Loyalty Indonesia memiliki Single Sign-On (SSO) versinya sendiri di setiap aplikasi web internal, agar tidak bebas diakses pihak asing. Sekarang sudah digunakan versi ketiga atau SSO V3. Aplikasi Voucher Targeted juga harus menggunakan SSO V3 dengan cara melakukan hit ke API SSO V3 yang sudah dibuat oleh developer backend senior. SSO V3 ini menggunakan NIK dan PIN karyawan. Setelah berhasil log in, user akan mendapat auto-generated token dari sistem dan token tersebut digunakan sebagai authorization untuk dapat mengakses tiap aplikasi. Setelah mengimplementasikan SSO, maka akan muncul icon lock di bagian kanan router yang menandakan bahwa router tersebut membutuhkan token untuk dapat diakses.



Gambar 3.35 SSO V3 Perusahaan.



Gambar 3.39 Router Yang Memerlukan Token.

- Mengerjakan project Voucher Targeted memperbaiki beberapa bug setelah testing oleh QA dan Frontend.

Setelah mengerjakan semua router yang dibutuhkan, router tersebut diberikan ke frontend untuk dikonsumsi. Setelah frontend menyelesaikan bagiannya, aplikasi diberikan kepada QA untuk dilakukan testing. Selama menunggu testing, penulis memanfaatkan waktu untuk mempelajari gambaran project baru yang akan dikerjakan secara paralel, bernama “E-Ticketing”. Terdapat beberapa bug yang cukup signifikan di berbagai bagian aplikasi, seperti opsi departemen tidak ada, list data voucher tidak urut sesuai tanggal pembuatan, tidak bisa menghapus request yang direject, history data tidak muncul, dan banyak lainnya. Gambar 3.40 – 3.43 menunjukkan daftar bug dari backend.

Platform	User Access	Modul / Field	Description	Screenshot
BE	Validator	Create Request	Tidak ada opsi department BUSINESS CONTROLLING	Bugs 3.png
BE	All User	Create Request	Tidak bisa Create Voucher jika memasukkan nominal Min. Belanja yang lebih dari batas yang ada	Bugs 4.png
BE	Validator	Create Program	Tidak bisa membuat Group Program dan Sub Program baru	Bugs 5.mp4
BE	Validator	Edit Program	Tidak bisa menambahkan Sub Program di Group Program yang sudah ada	Bugs 6.png

Gambar 3.40 List Bug 1.

BE	All User	Home	List Voucher tidak sort by waktu pembuatan Request Voucher	Bugs 15.mp4
BE	Validator	Report	Tidak bisa buka file Excel Voucher	Bugs 16.mp4
BE	Requester	Home	Ketika tidak ada data, muncul error di console	
BE	All User	Detail Request	Muncul error di network apabila tidak ada data approval	Bugs 17.mp4

Gambar 3.41 List Bug 2.

BE	Validator	Approve	Tidak bisa Approve dan Accept Request	Bugs 23.mp4
BE	Validator	Home Request & Approval	List tidak sort by waktu pembuatan Request Voucher	Bugs 24.png
BE	Validator	Approve	Request berhasil di approve/reject namun tidak muncul di History Approval & bisa dilakukan berulang kali	Bugs 25.mp4
BE	All User	Detail Request	Data Approval / Reject tidak muncul di History Approval	Bugs 26.mp4
BE	All User	Create Request	Muncul error gagal Create Request Voucher di halaman Create Request tapi saat user klik submit, request berhasil tersimpan ke List Request Voucher dan database	Bugs 28.png

Gambar 3.42 List Bug 3.

BE	Validator	Detail Request	Muncul error ketika masuk ke halaman Detail Request	Bugs 39.mp4
BE	Validator	Detail Request	Data Approval / Reject tidak muncul di History Approval	Bugs 40.mp4
BE	Validator	Detail Request	Request berhasil di approve/reject namun tidak muncul di History Approval & bisa dilakukan berulang kali	Bugs 41.mp4
BE	Validator	Home	Muncul error karena tidak ada department BUSINESS CONTROLLING	Bugs 42.png
BE	All User	Submit Link	Status request belum berubah menjadi DONE meskipun Validator sudah menaiki link Tableau	Bugs 43.mp4

Gambar 3.43 List Bug 4.

- Mengerjakan project Voucher Targeted untuk router GET, POST, PUT, dan DELETE Remark Blast.

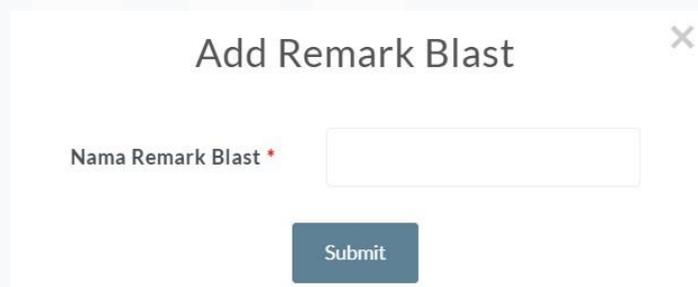
Terdapat permintaan baru dari user, yaitu Master Setting Remark Blast. Sama seperti group program dan sub program, user ingin agar validator dapat mengatur remark blast, seperti menambah, mengubah, dan menghapus. Sehingga opsi remark blast tidak hanya “ALFAGIFT” dan “NON-ALFAGIFT” saja.

Master Setting Remark Blast	
GET	/remark_blast/get_all_remark_blast Get All Remark Blast
POST	/remark_blast/create_remark Create Remark
PUT	/remark_blast/update_remark Update Remark
DELETE	/remark_blast/delete_remark Delete Remark

Gambar 3.44 Router Master Setting Remark Blast.



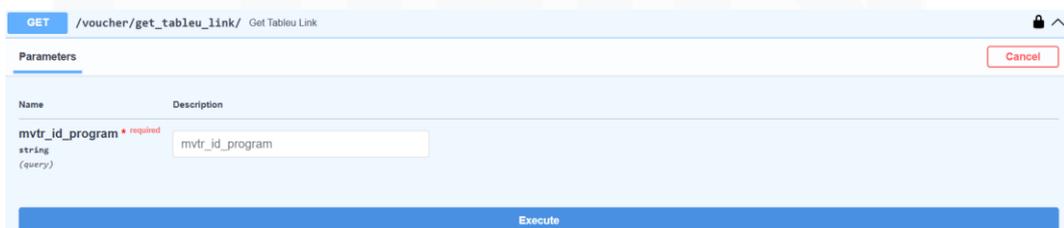
Gambar 3.45 Halaman Master Setting Remark Blast.



Gambar 3.46 Menambah Remark Blast.

- Mengerjakan project Voucher Targeted untuk router GET dan PUT tableau link.

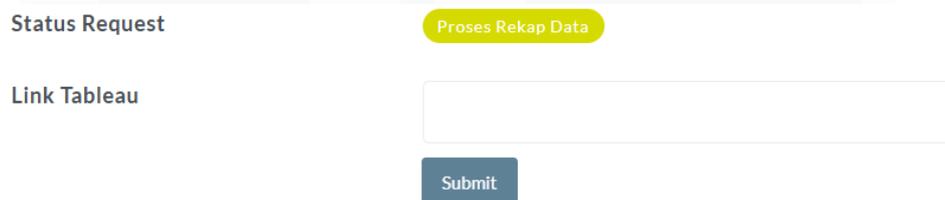
Kemudian terdapat fitur baru lagi yang harus dikerjakan, yaitu validator dapat memberikan link menuju tableau untuk visualisasi data voucher. Link hanya bisa diberikan jika status request berada di “Proses Rekap Data”. Secara bersamaan, diberikannya link tableau ini menandakan bahwa request telah berhasil dan status menjadi “Selesai”. Untuk fitur ini digunakan 2 router, yaitu “Get Tableau Link” dan “Update Tableau Link”.



Gambar 3.47 Router Get Tableau Link.



Gambar 3.48 Router Update Tableau Link.



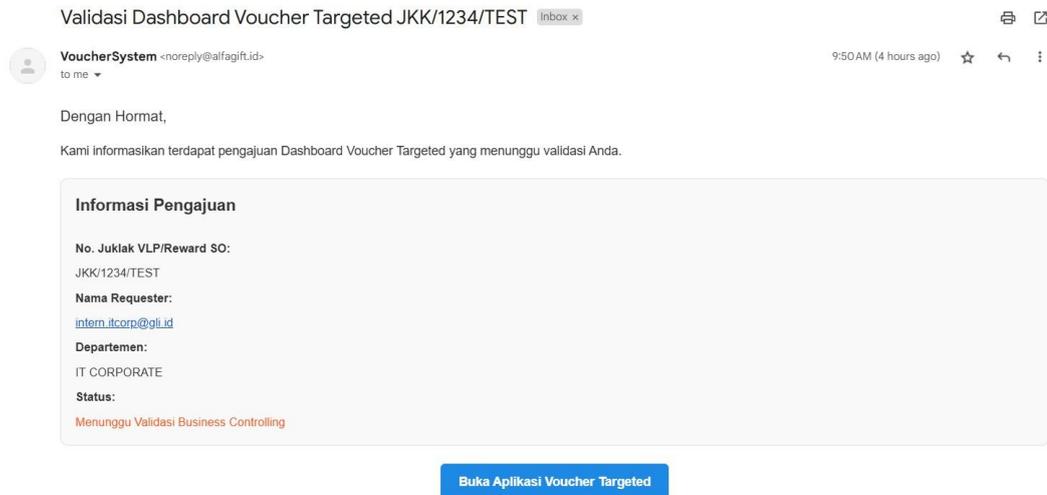
Gambar 3.49 Tempat Submit Link Tableau.



Gambar 3.50 Sudah Submit Link Tableau.

- Mengerjakan project Voucher implementasi email service.

Fitur terakhir yang dikerjakan sebelum aplikasi ini masuk ke tahap UAT adalah notifikasi email. Aplikasi ini akan mengirim email ke yang bersangkutan saat ada data voucher baru dibuat, data voucher diupdate, approve atau reject, link tableau muncul, dan link tableau diupdate. Implementasi fitur ini dilakukan dengan cara hit ke API email service yang sudah dibuat oleh developer backend senior. Selama proses development digunakan email dummy untuk testing apakah fitur email sudah berhasil. Function email yang melakukan hit ke API ini dipanggil oleh tiap router yang memerlukan.



Gambar 3.51 Contoh Notifikasi Email.

3.3 Kendala yang Ditemukan

- Pengembangan website menggunakan template perusahaan, konsep clean architecture, dan framework FastAPI yang belum pernah dipelajari sebelumnya.
- Belum familiar dengan aplikasi DBeaver dan PostgreSQL untuk database.

3.4 Solusi atas Kendala yang Ditemukan

- Membaca dan memahami dokumentasi mengenai FastAPI yang dimana dokumentasinya cukup banyak, karena basisnya menggunakan python.
- Menanyakan dan mendapat tutorial dari supervisor dan developer senior.
- Menerapkan metode “Learn by Doing”, karena harus memahami sendiri bagaimana arsitektur template perusahaan.