[POPC] Request Mutation Rotation Employee

Date	@September 6, 2022
Summary	Feature Request Mutation or Rotation Employee
Tags	
In Charge	
created	@September 6, 2022 10:12 AM
created by	
Team	Enterprise
Swimlane	
state	modifying
↗ application	
↗ RFC relation	
Url	
PRD	https://www.notion.so/Pengajuan-Rotasi-Mutasi-7d56048f9e4b435fa56e775d33e10225

▼ ***Related Document***

Document Name	Document URL

▼ ***Table of Content***

Version
1. Summary
2. Problem and Motivation
3. Detailed Design
    Flowchart
    API Contract

Design UI/UX
4. Dependencies
5. Milestone/Deployment Strategy
6. Data Result
7. Drawbacks/Risks/Possible Failures
8. Alternatives
9. Unresolved/Future Possibilities

Need Review

@Mention person	<set Status or description>

Version

date updated	changes summary

1. Summary

> 💡 *TLDR lo mau ngajuin apa, masalah lo apa dan bentuknya apa, pitch paragraph gitu.*

People Culture (OD&P) sulit me-manage pengajuan mutasi rotasi yang diterima melalui email. Sehingga diperlukan aplikasi yang dapat menampung request dari managers, mudah untuk diproses review dan approval oleh People Culture.

2. Problem and Motivation

> 💡 *Latar belakang dan why problem, What's the current status of real condition, refer ke Problem Definition juga bisa.*

Pain points: penerimaan request via email sering tertimbun dan lack of attention, approval manual via email, pencatatan manual, tidak adanya visibility atas status pengajuan.

Automate proses Pengajuan Mutasi Rotasi yang memerlukan review dan approval People Culture

Value for users:
1. Faster process handling request and give feedback after review to requestor.
2. Better visibility of request status

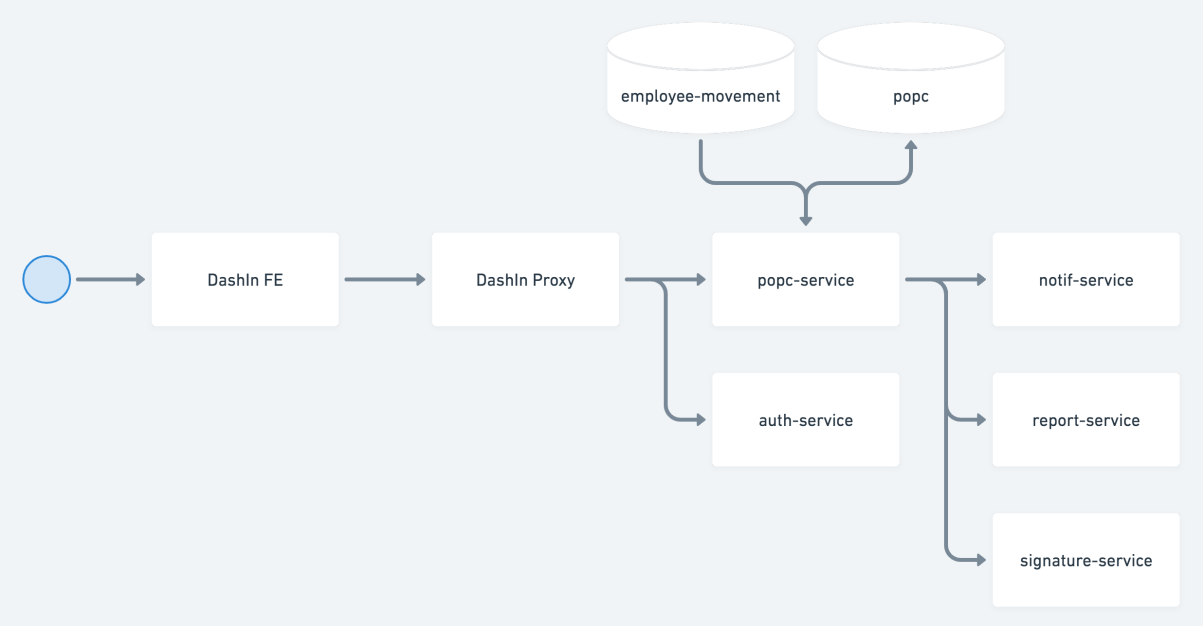
Value for business:
1. Improve efficiency: less manual work (reduce human error, less hassle, more secure)
2. Proper container for storing data request mutasi rotasi

3. Detailed Design

> 💡 *Bisa pasang sketching, C4 architecture, pseudocode, flow/algorithm, rancangan kode modul, cara penggunaan, constraint, ide teknis, dll yang spesifik teknis.*

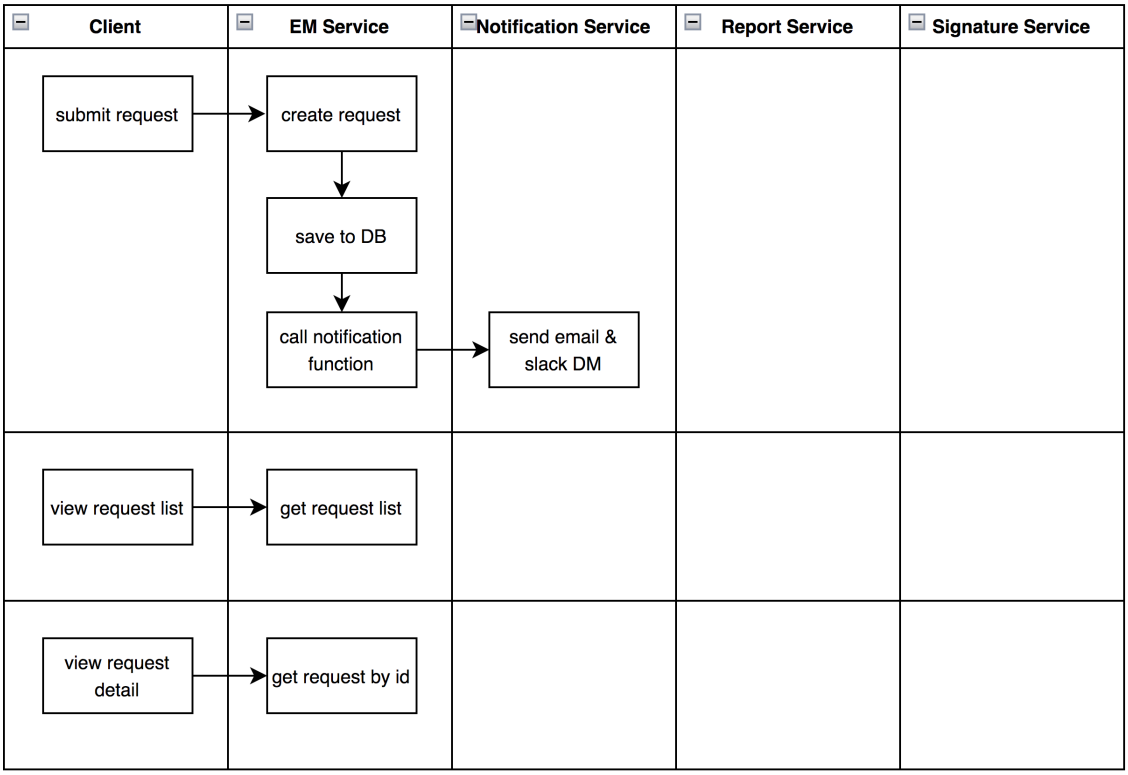
▼ Flowchart

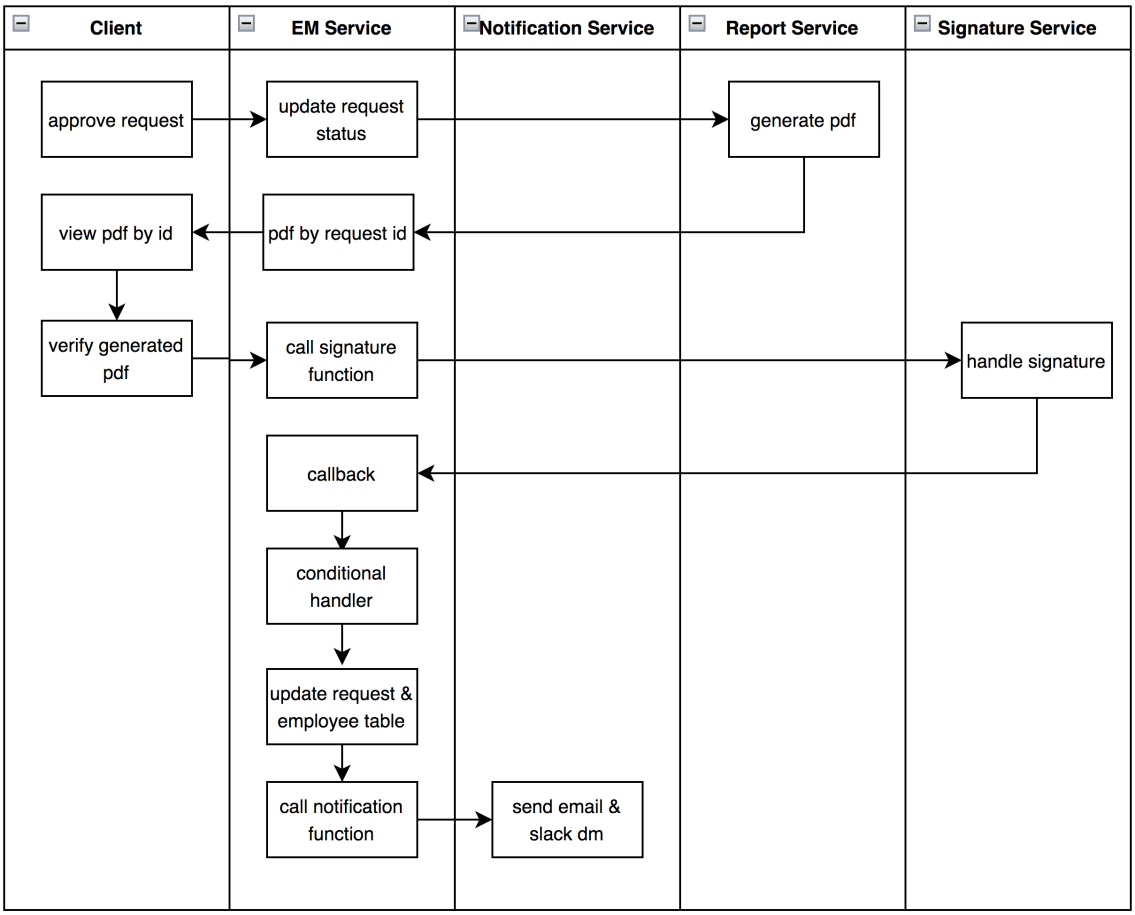
https://whimsical.com/popc-request-mutasi-atau-rotasi-employee-NbnZPFZh2VdAuYYUgt5Uwv@2Ux7TurymN1FMXHPy7Lx



▼ Swimlane Diagram

https://t.ly/pvBe (link swimlane diagram.net)


Client
EM Service
Notification Service
Report Service
Signature Service
submit request
create request
save to DB
call notification function
send email & slack DM
view request list
get request list
view request detail
get request by id



▼ Sequence Diagram

▼ ***Submit***

```
sequenceDiagram
    Client (Requestor)->>+Mutation Rotation: Submit New Request
    Mutation Rotation->>+Database: Create New Row in DB
    Database->>-Mutation Rotation: DB Trx Response
    Mutation Rotation->>+Notification Service: http request
    Notification Service->>-Mutation Rotation: response
    Mutation Rotation->>-Client (Requestor): response
```

▼ ***GetListRequested***

```
sequenceDiagram
    Client (Requestor & P&C)->>+Mutation Rotation: Get Request List
    Mutation Rotation->>+Database: Find List in DB
    Database->>-Mutation Rotation: DB Trx Response
    Mutation Rotation->>-Client (Requestor & P&C): Response
```

▼ ***Update Status***

```
sequenceDiagram
    Client (P&C)->>+Mutation Rotation: P&C Update Status
    Mutation Rotation->>+Database: Update Row in DB
    Database->>-Mutation Rotation: DB Trx Response
    Mutation Rotation->>+Notiffication Service: http request
    Notiffication Service->>-Mutation Rotation: response
    Mutation Rotation->>-Client (P&C): response
```

▼ ***Details***

```
sequenceDiagram
    Client (Requestor & P&C)->>+Mutation Rotation: Get Request Details
    Mutation Rotation->>+Database: Find Request by uuid in DB
    Database->>-Mutation Rotation: DB Trx Response
    Mutation Rotation->>-Client (Requestor & P&C): Response
```

▼ ***Revision***

```
sequenceDiagram
    Client (Requestor) ->>+Mutation Rotation: Requestor Revise Request
    Mutation Rotation->>+Database: Update Row in DB
    Database->>-Mutation Rotation: DB Trx Response
    Mutation Rotation->>-Client (Requestor): Response
```

▼ API Contract

▼ Phase 1

▼ **V1**

General

- Headers
  - X-App-Token / Authorization: string token
  - request-id: string uuid
- Format Response Success

```
{
  success: boolean,
  data: object[],
  message: string
}

# Example
{
  success: true,
  data: [
    {
      id: "123-asd",
      ...
    },
    ...
  ],
  message: "Berhasil Request"
}
```

- Format Response Error

```
{
  success: boolean,
  data: object[],
  message: string
}

# Example
{
  success: false,
  data: [],
  message: "Gagal Request"
}
```

Spesifik

- Health Check

```
Method: "GET",
URL: "/api/v1/mutation-rotation/ping",
Response Success on Message Format: {
  "Pong!"
}
```

- Get List Requested

```
Method: "GET",
URL: "/api/v1/mutation-rotation/requested",
Params: {
  email: "kirun@efishery.com",
  status: "approved",
  limit: 10,
  offset: 0
},
Response Success on Data Format: {
  requestor: [{...}],
  pagination: {
    start_data: 1,
    end_data: 10,
    total_data: 130
  }
}
```

- Get List Requested By Searching

```
Method: "GET",
URL: "/api/v1/mutation-rotation/requested",
Params: {
  search: "kirun",
  limit: 10,
  offset: 0
}
Response Success on Data Format: {
  requestor: [{...}],
  pagination: {
    start_data: 1,
    end_data: 10,
    total_data: 130
  }
}
```

- Submit Request

```
Method: "POST",
URL: "/api/v1/mutation-rotation/submit",
Body: {
  field: "semua field yg ada di form",
  …
}
```

- Update Request

```
Method: "PUT",
URL: "/api/v1/mutation-rotation/update",
Body: {
```

```
  field: "semua yang mau di update",
  …
}
```

- Detil Request

```
Method: "GET",
URL: "/api/v1/mutation-rotation/details",
Params: {
  id: "ini-uuid-ceritanya-yah-heheh"
},
Response Success on Data Format: {
  requestor: {...},
}
```

- Drop Request (Soft Delete)

```
Method: "PUT",
URL: "/api/v1/mutation-rotation/delete",
Params: {
  id: "ini-uuid-ceritanya-yah-heheh"
},
Response Success on Data Format: {
  requestor: {...},
}
```

▼ **V2**

General

- Headers
  - X-App-Token / Authorization: string token
  - request-id: string uuid
- Format Response Success

```
{
  success: boolean,
  message: string,
  data: object[]
}

# Example
{
  success: true,
  message: "Berhasil Request",
  data: [
    {
```

```
      id: "123-asd",
      ...
    },
    ...
  ]
}
```

- Format Response Error

```
{
  success: boolean,
  message: string,
  data: object[]
}

# Example
{
  success: false,
  message: "Gagal Request",
  data: []
}
```

Spesifik

- Health Check

```
Method: "GET",
URL: "/api/v1/mutation-rotation/ping",
Response Success on Message Format:
{
  "success": true,
  "message": "Pinging All Database",
  "data": {
        "PostgreSQLStatusOK": true,
        "RedisStatusOK": true
    }
}
```

- Get List Requested

```
Method: "GET",
URL: "/api/v1/mutation-rotation/requested",
Params: {
  email: "kirun@efishery.com",
  status: "approved",
  limit: 10,
  offset: 0
},
```

```
Response Success on Data Format:
{
    "success": true,
    "message": "Success Get List Move Request",
    "data": {
        "requestor": [
            {...},
            {...}
        ],
        "pagination": {
            "start_data": 1,
            "end_data": 10,
            "total_data": 130
        }
    }
}
```

- Get List Requested By Searching

```
Method: "GET",
URL: "/api/v1/mutation-rotation/requested",
Params: {
  search: "kirun",
  limit: 10,
  offset: 0
}
Response Success on Data Format:
{
    "success": true,
    "message": "Success Get List Move Request",
    "data": {
        "requestor": [
            {...},
            {...}
        ],
        "pagination": {
            "start_data": 1,
            "end_data": 10,
            "total_data": 130
        }
    }
}
```

- Submit Request

```
Method: "POST",
URL: "/api/v1/mutation-rotation/submit",
Body: {
  "field": "semua field yg ada di form",
  …
}
```

- Update Request

```
Method: "PUT",
URL: "/api/v1/mutation-rotation/update",
Body: {
  "field": "semua yang mau di update",
  …
}
```

- Detail Request

```
Method: "GET",
URL: "/api/v1/mutation-rotation/:uuid",
Response Success on Data Format:
{
  "success": true,
  "message": "Success Get Detail",
  "data": {...}
}
```

▼ Phase 2

- Generate Document

```
Method: "POST",
URL: "/api/v1/mutation-rotation/generate-document",
FormData: {
  Coming Soon . . .
}
```

- Generate DocuSign

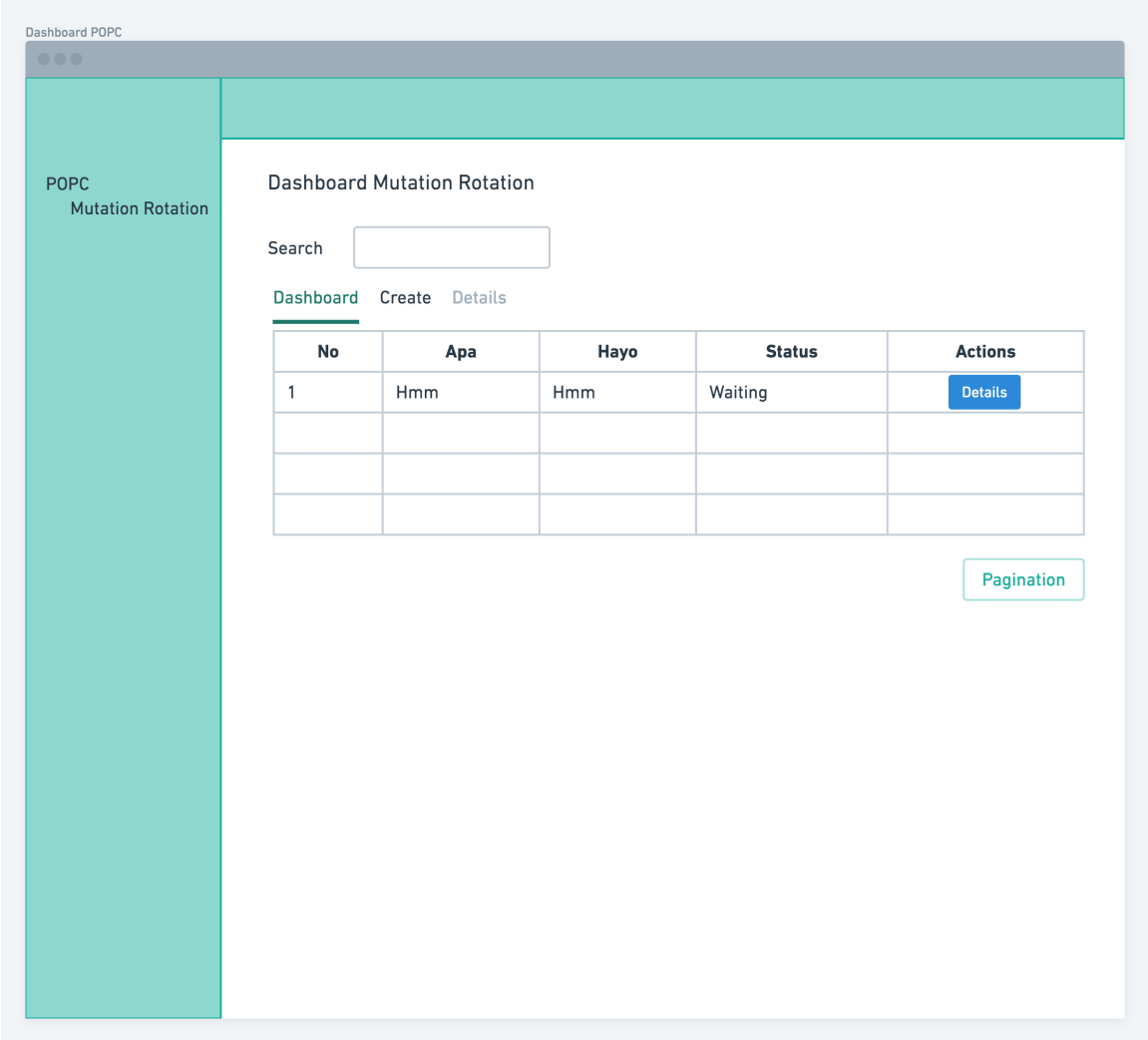
```
Method: "POST",
URL: "/api/v1/mutation-rotation/generate-docusign",
FormData: {
  Coming Soon . . .
}
```

▼ Phase 3

▼ Design UI/UX

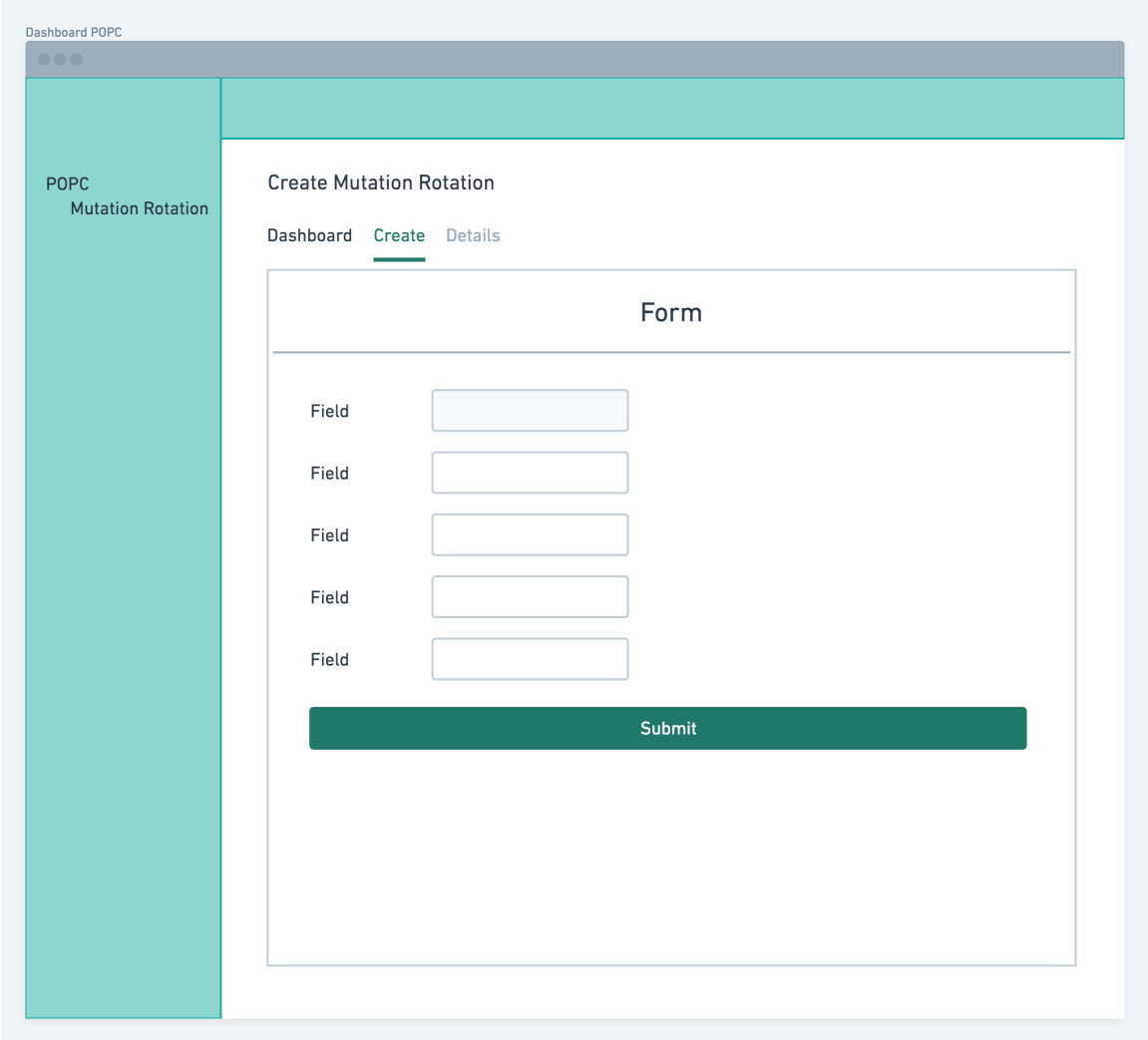
▼ Phase 1

https://whimsical.com/popc-request-mutasi-rotasi-employee-BSmHLcgHc9wnkEzHFeKw2h@7YNFXnKbYqP6zN2sESBBy

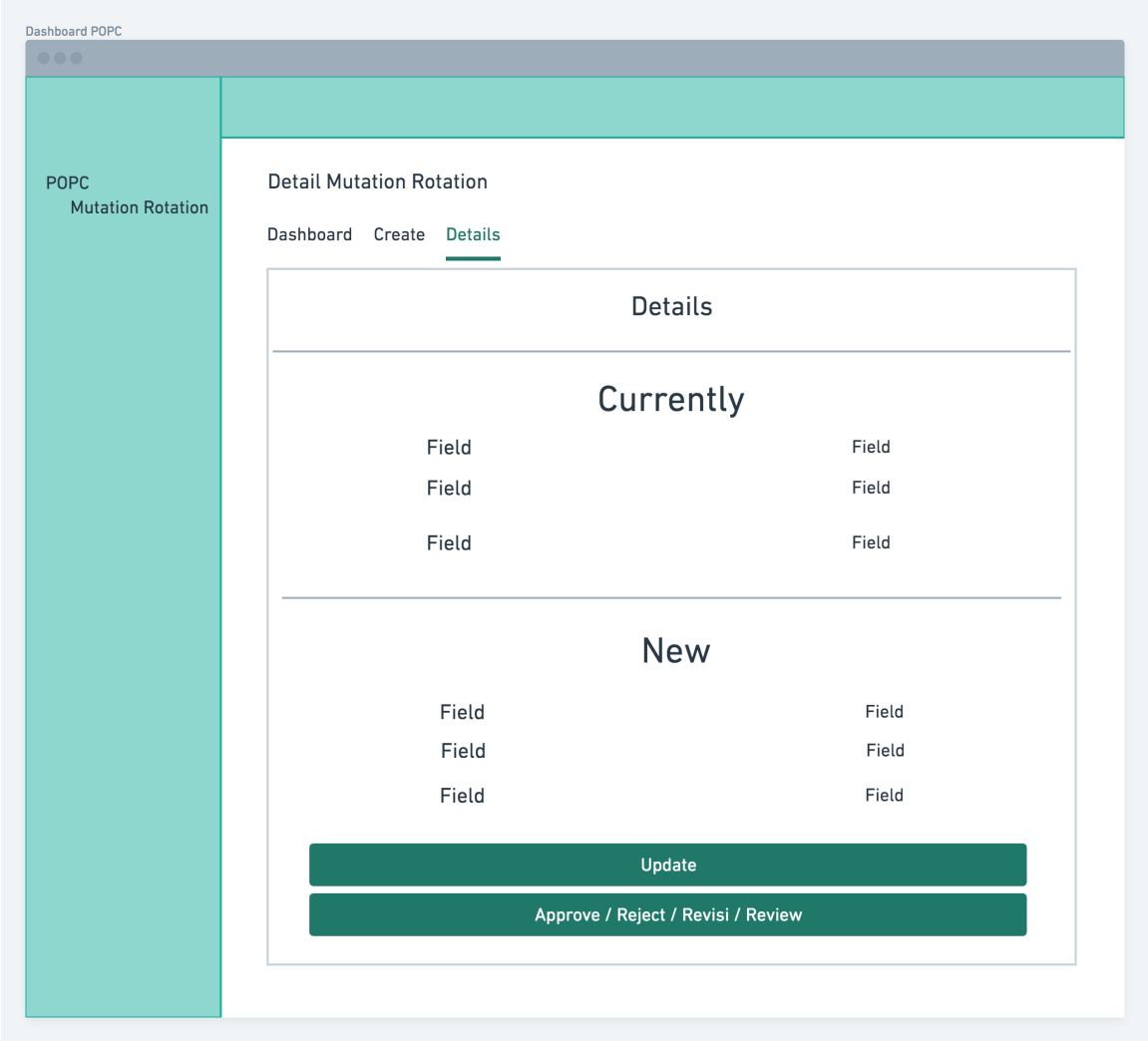


- Design based on DashIn. Jadi kita mengikuti atau adaptasi layout dari existing Dashin yg mana dia memiliki sidebar, header dan content.
- View di pisah berdasarkan 2 jenis role yaitu Manager/Supervisor dan POPC
- Perbedaan view terdapat dari query atau parameter ketika user login
  Contoh: User POPC login, maka parameter yg kita berikan ke service yaitu email user tersebut
- Terdapat search untuk memudahkan user mencari data yg di inginkan (no request, email atau nama atau bisa bertambah)

- Berbentuk tabular atau table dan setiap row memiliki button action (detail) tentunya dengan pagination



- Component Create dan Update/Edit disamakan
- Bedanya hanya di action dan copywrite saja, secara layout sama
  Contoh: Ketika Create, button menjadi `Submit` sedangkan ketika Update, button menjadi `Update`
- Kemudian ada beberapa field yg akan autofill ketika memilih nama user yang ingin di Mutasi Rotasi dan ada beberapa field yg tidak dapat di ubah

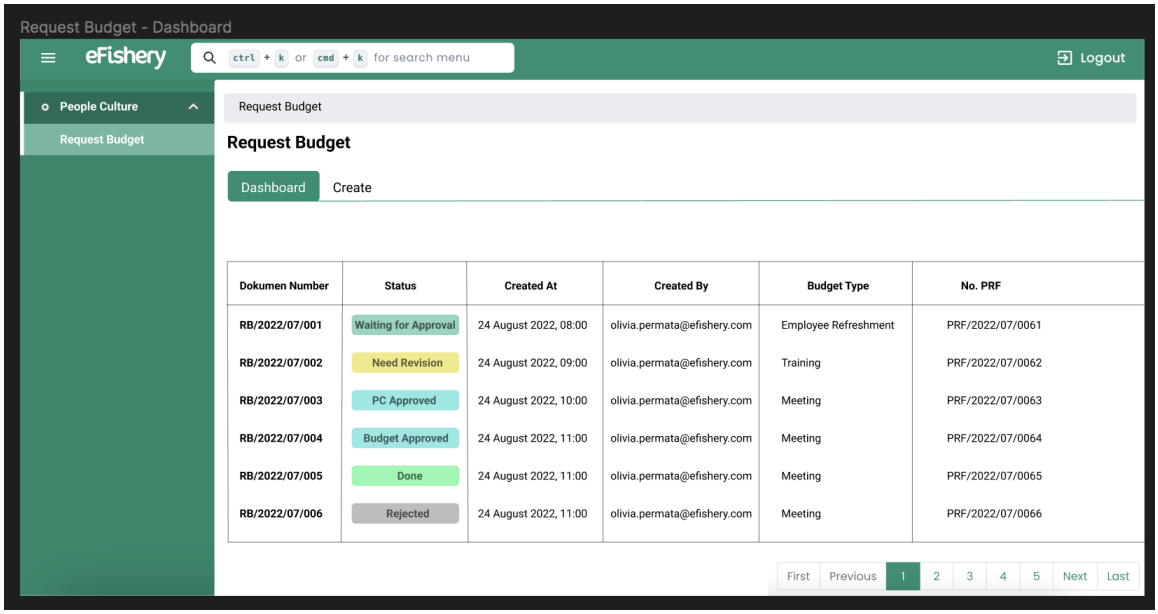


- Terdiri dari informasi Sekarang atau pun Terbaru/Akan di request
- Button akan dinamis berubah sesuai dengan status dan siapa yg view
  Contoh: Manager view dan status Revision, maka button akan muncul `Update`

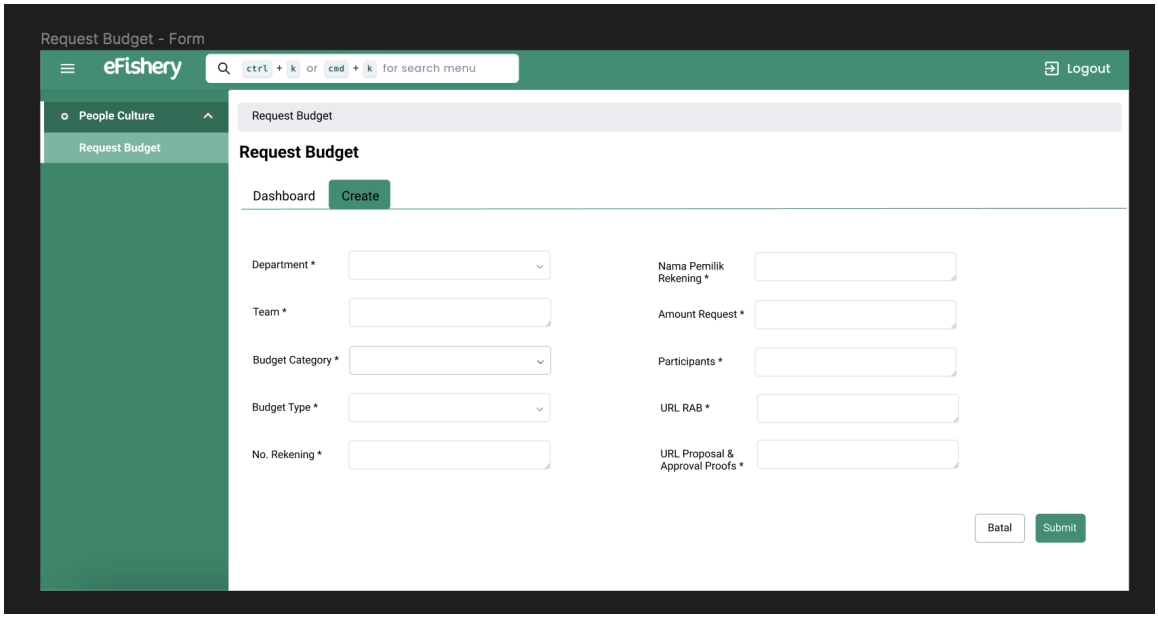
Updates!

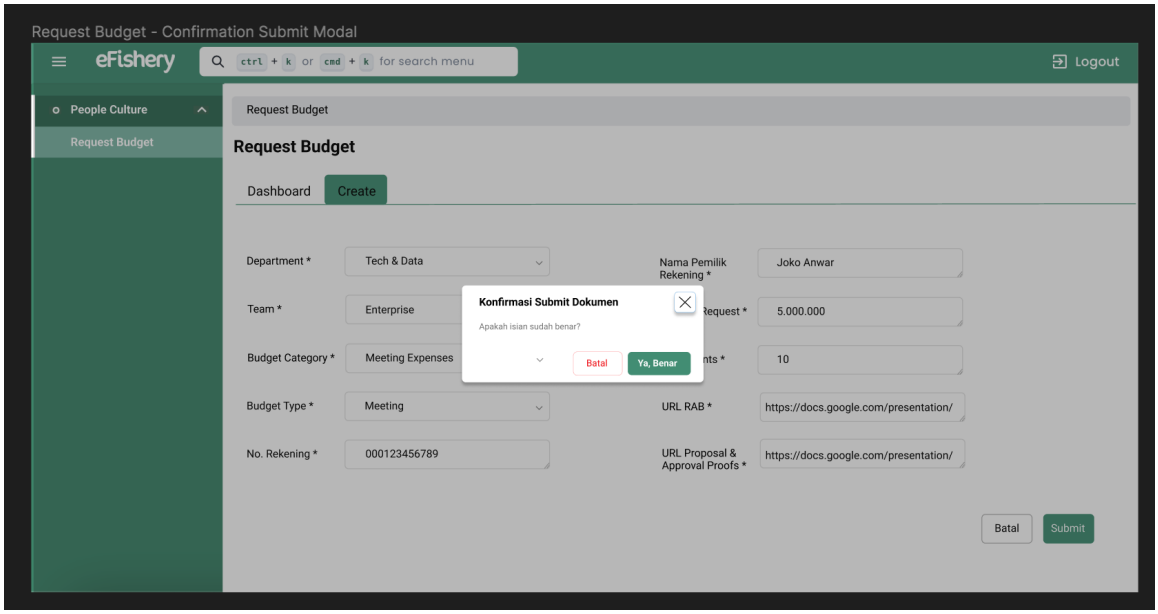
https://www.figma.com/file/TahuoYEu4z7uwtepA9NcP2/Request-Budget-and-Realization?node-id=0%3A1

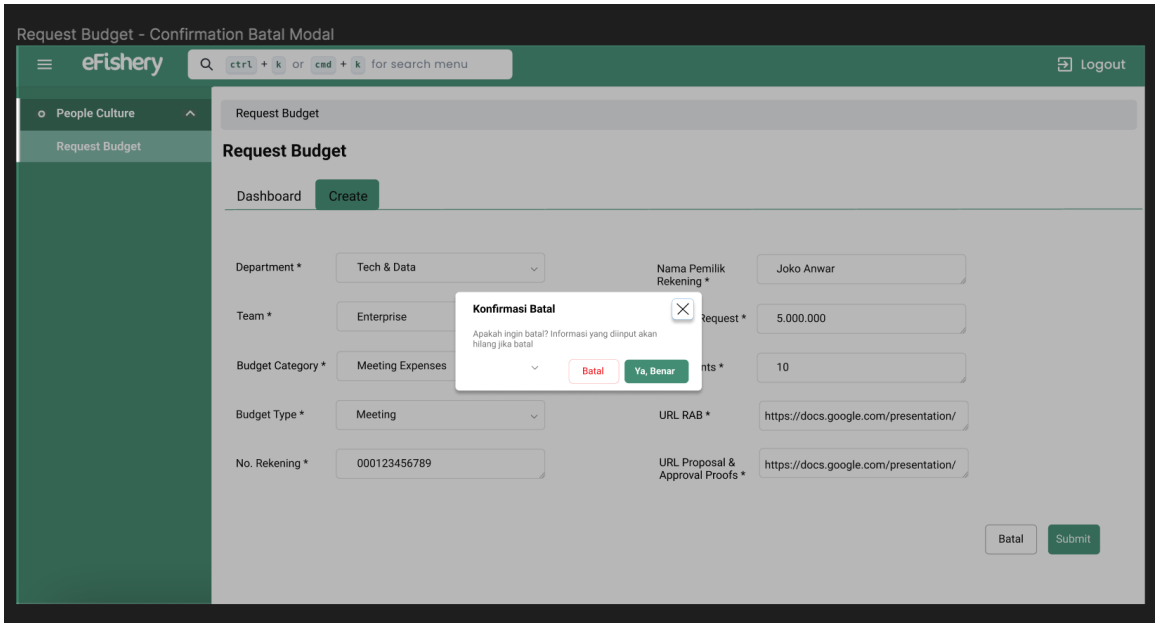
Request Budget - Dashboard



Request Budget - Form







▼ Phase 2

▼ Phase 3

4. Dependencies

> 💡 *Kalau ada kemungkinan dependensi ke sistem apa atau bikin dependensi untuk sistem apa.*

- auth-service
- notif-service
- report-service
- signature-service
- master employee database

5. Milestone/Deployment Strategy

> 💡 *Ajuan cara adopsi, milestone development risetnya atau pemasangannya di production level atau saat implementasi riilnya.*

1. Phase 1 || 05 - 16 Sept 2022
    a. 05 - 09 Sept 2022
        i. Grooming
        ii. Create RFC
        iii. Set timeline and task
    b. 12 - 14 Sept 2022 || Create, Read dan Update Request Mutation or Rotation (FE-BE)
    c. 15 Sept 2022 || Testing on Staging (FE-BE)
    d. 16 Sept 2022 || Testing on UAT (FE-BE)
2. Phase 2 || 19 - 30 Sept 2022
    a. Coming soon . . .
3. Phase 3 || 03 - 14 Okt 2022
    a. Coming soon . . .

POPC Request Mutation or Rotation

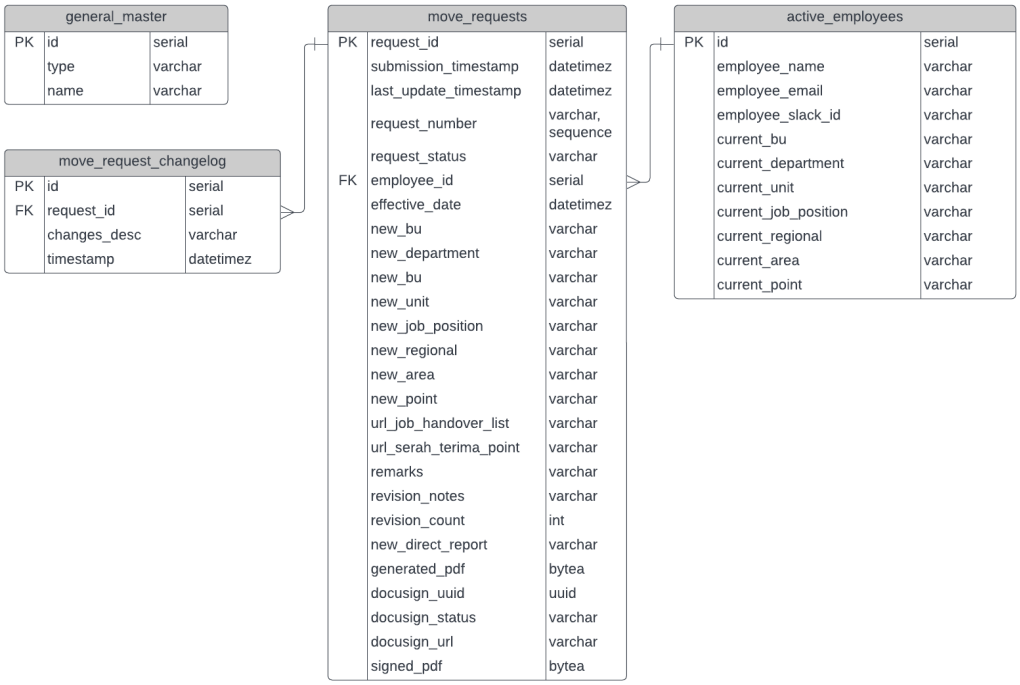
Aa Name	Assign	Date	Status
CRU Request Mutation or Rotation		@September 12, 2022 → September 16, 2022	
Docusign		@September 19, 2022 → September 30, 2022	
Generate Surat Keterangan Mutasi Rotasi		@October 3, 2022 → October 14, 2022	

6. Data Result

💡 *Silakan isi dengan hasil data production-nya. Bisa diisi dengan Data Flow Diagram, Entity and Relationship Diagram, dan atau Data Model Documentation-nya.*

▼ **ERD V1**

https://lucid.app/lucidchart/7dc4cde4-d652-465c-99a7-8f1cb376468a/edit?invitationId=inv_15377951-4d1c-4d95-bf09-9e723e66cdee&page=0_0#



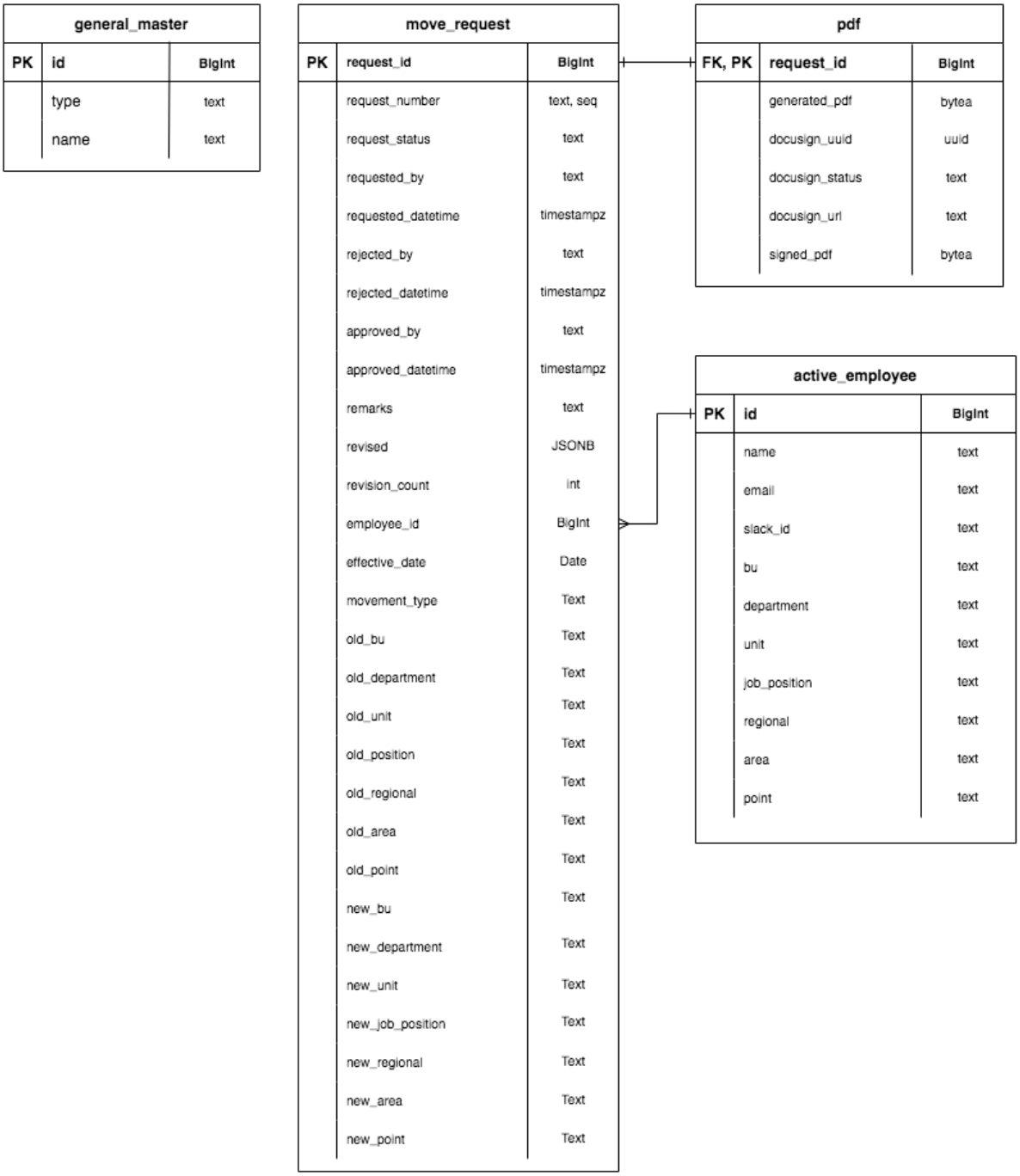
▼ **Metadata Needs**

Metadata Needs

No	Properties	Type	Example	Notes
1	requested_by	varchar	kirun@efishery.com	Siapa yang melakukan Request
2	requested_datetime	timestamp	1662075243	Kapan melakukan Request
3	revised	jsonb	[{by: “kirun@efishery.com”, datetime: 1662075243, notes: “apa hayo”}]	Siapa, Kapan dan Mengapa melakukan Revisi
4	rejected_by	varchar	kirun@efishery.com	Siapa yang melakukan Reject
5	rejected_datetime	timestamp	1662075243	Kapan melakukan Reject
6	approved_by	varchar	kirun@efishery.com	Siapa yang melakukan Approve
7	approved_datetime	timestamp	1662075243	Kapan melakukan Approve

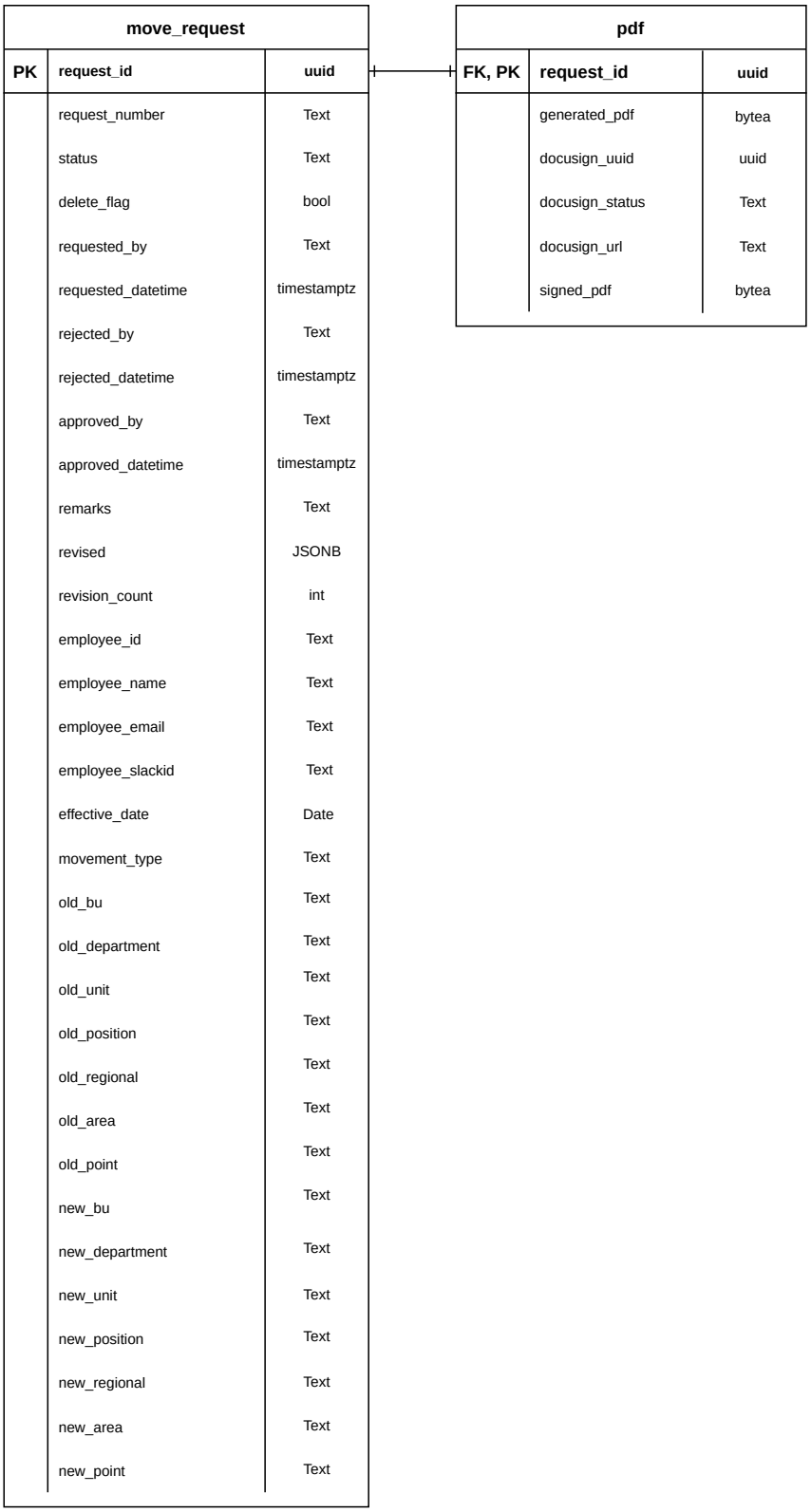
▼ **ERD V2**

https://t.ly/OT7n



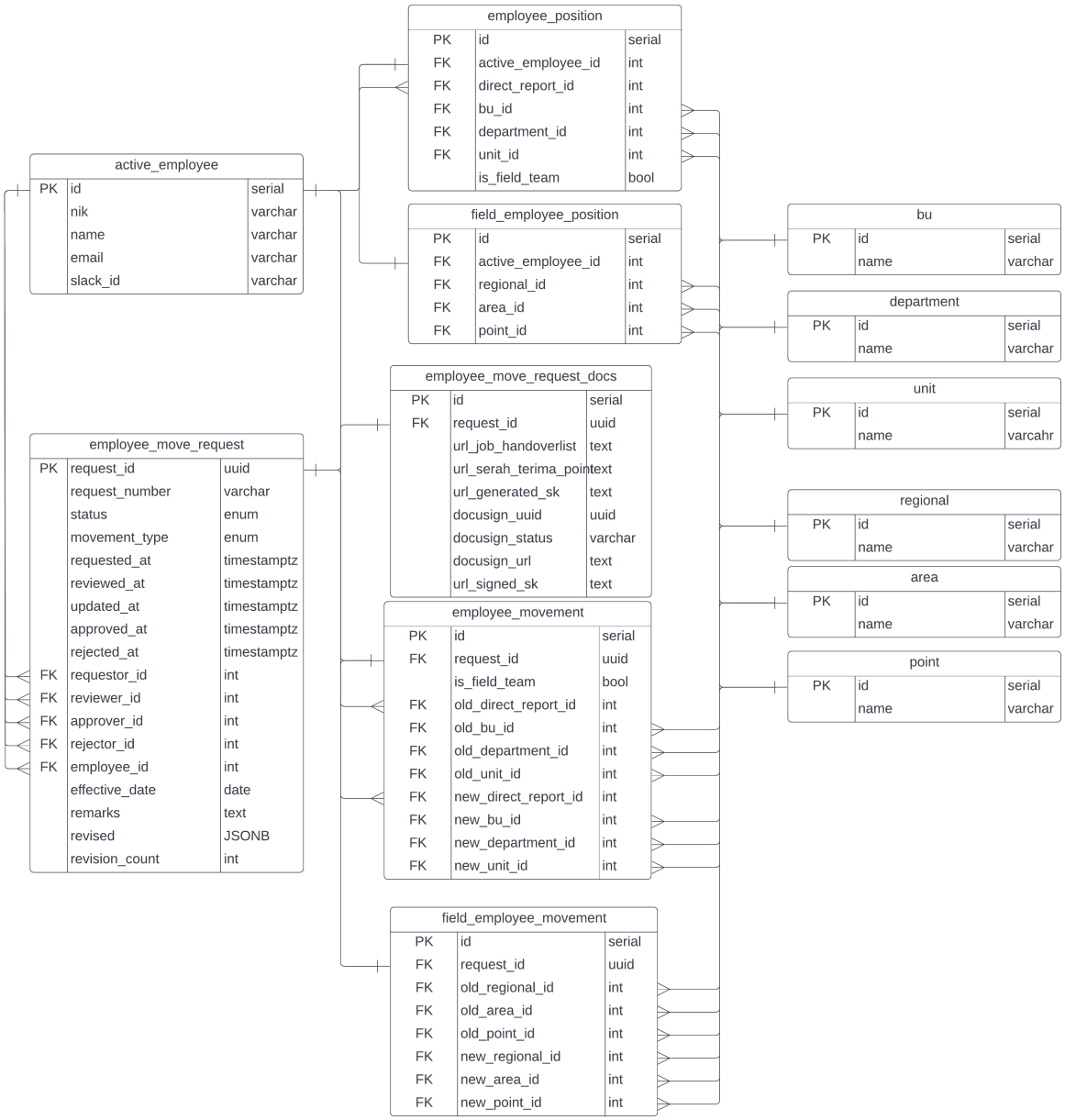
▼ **ERD V3**

https://t.ly/bqYM


move_request
PK
request_id
uuid
request_number
Text
status
Text
delete_flag
bool
requested_by
Text
requested_datetime
timestamptz
rejected_by
Text
rejected_datetime
timestamptz
approved_by
Text
approved_datetime
timestamptz
remarks
Text
revised
JSONB
revision_count
int
employee_id
Text
employee_name
Text
employee_email
Text
employee_slackid
Text
effective_date
Date
movement_type
Text
old_bu
Text
old_department
Text
old_unit
Text
old_position
Text
old_regional
Text
old_area
Text
old_point
Text
new_bu
Text
new_department
Text
new_unit
Text
new_position
Text
new_regional
Text
new_area
Text
new_point
Text
pdf
FK, PK
request_id
uuid
generated_pdf
bytea
docusign_uuid
uuid
docusign_status
Text
docusign_url
Text
signed_pdf
bytea

▼ ERD V4

https://lucid.app/publicSegments/view/b0ba017c-0a72-4197-ae77-6030126a1e29/image.png



ℹ️ **table** `employee_move_request`

Name	Description	Type	Required	key
request_id	request_uuid, generated from backend	uuid	yes	primary
request_number	autogenerate sequence (MR/year/month/xx)	text	yes	-
status	status pengajuan	enum	yes	
delete_flag	soft delete flag	bool	yes	
requested_by	requestor email	text	yes	-
requested_datetime	submit timestamp	timestamptz	yes	-
rejected_by	p&c rejector email	text	no	-
rejected_datetime	reject timestamp	varchar	no	-
approved_by	p&c approver email	text	no	-
approved_datetime	approve timestamp	timestamptz	no	-
remarks	approve/reject remarks	text	no	-
revised	array of object, revision detail. [{by: “kirun@efishery.com”, datetime: 1662075243, notes: “apa hayo”}]	JSONB	no	-
revision_count	increment on every revision	int	no	-
employee_id	nik	text	yes	-
employee_name	nama karyawan	text	yes	-
employee_email	email karyawan	text	yes	-
employee_slackid	slack id karyawan	text	yes	-
effective_date	tanggal efektif mutasi/rotasi	date	no	-
movement_type	selection, mutasi/rotasi	enum	yes	-
old_bu		text	no	-
old_department		text	no	-
old_unit		text	no	-
old_position		text	no	-
old_regional		text	no	-
old_area		text	no	-

Name	Description	Type	Required	key
old_point		text	no	-
new_bu		text	no	-
new_departmenet		text	no	-
new_unit		text	no	-
new_position		text	no	-
new_regional		text	no	-
new_area		text	no	-
new_point		text	no	-

ℹ️ **table** `employee_move_request_docs`

Name	Description	Type	Required	key
request_id	primary key references `move_request`	uuid	yes	primary, foreign
generated_pdf	generated pdf from report service	bytea	no	-
docusign_uuid	docusign uuid from signature service	uuid	no	-
docusign_status	docusign status	text	no	-
docusign_url	docusign s3 url from signature service	text	no	-
signed_pdf	generated pdf from signature service	bytea	no	-

7. Drawbacks/Risks/Possible Failures

💡 *Kalau ada kemungkinan cons atau kekurangan dengan design ini, apa saja. Kemungkinan gagal dll-nya.*

…

8. Alternatives

> 💡 *Diluar ide ajuan ini ada apa lagi? Bisa masukkin RFC lain atau link services lain atau link artikel lain.*

…

9. Unresolved/Future Possibilities

> 💡 *Listing sesuatu yang masih abu-abu dan ga jelas, perlu riset lanjut, kemungkinan ke depannya.*

…