

## BAB 3

### METODOLOGI PENELITIAN

Pada penelitian pembangunan sistem *medical reimburse* berbasis Odoo, metode yang digunakan merupakan SDLC (*Software Development Life Cycle*). *Software Development Life Cycle* merupakan sebuah kerangka kerja yang digunakan untuk menyusun, merencanakan, dan mengendalikan proses pengembangan sistem informasi [28]. Salah satu Model yang digunakan adalah *Agile Software Development* yaitu *Scrum*. *Scrum* merupakan *framework* dari *agile methodology* yang menyediakan fleksibilitas untuk melakukan kontrol dan mengelola seluruh persyaratan sama baiknya dengan sebuah *software development*. Metode *scrum* dirancang untuk meningkatkan kemampuan produksi dari proses pengembangan, menyelaraskan keputusan organisasi dan individu, mendefinisikan kultur yang berfokus pada resital, dukungan konstruksi nilai pemegang saham, untuk memiliki pesan yang baik tentang kinerja di semua tingkatan, dan meningkatkan pengembangan dan kelas kehidupan [16]. Tahapan metode *scrum* meliputi *product backlog*, *sprint backlog*, *daily scrum*, *sprint review*, dan *sprint restropective* [17]. Berikut langkah-langkah yang dilakukan dalam penelitian antara lain :

#### 3.1 Analisis Kebutuhan

Pada sistem *medical reimburse* yang akan dibangun, terdapat *requirements* yang dibutuhkan untuk menjadi tujuan dari sistem yang akan dibuat. Kebutuhan sistem *medical reimburse* yang diminta berhubungan dengan kebutuhan setiap aktor yang menggunakan sistem *medical reimburse* Kompas Gramedia. Terdapat tiga buah aktor yang bersangkutan yaitu karyawan, *Human Resources*, dan admin. Berikut kebutuhan yang dimiliki oleh masing-masing aktor:

##### 1. Admin

- Admin dapat melihat seluruh *history* pengajuan yang dibuat oleh karyawan.
- Admin dapat melakukan CRUD terhadap data tarif produk pengobatan yang akan menjadi tarif pergantian biaya kesehatan.
- Admin dapat melakukan CRUD data pagu karyawan yang menjadi saldo pagu karyawan saat melakukan proses *medical reimburse*.
- Admin dapat melakukan CRUD data kategori pengajuan.

- Admin dapat melakukan CRUD data dokter spesialis dan produk dokter spesialis.

## 2. Human Resources

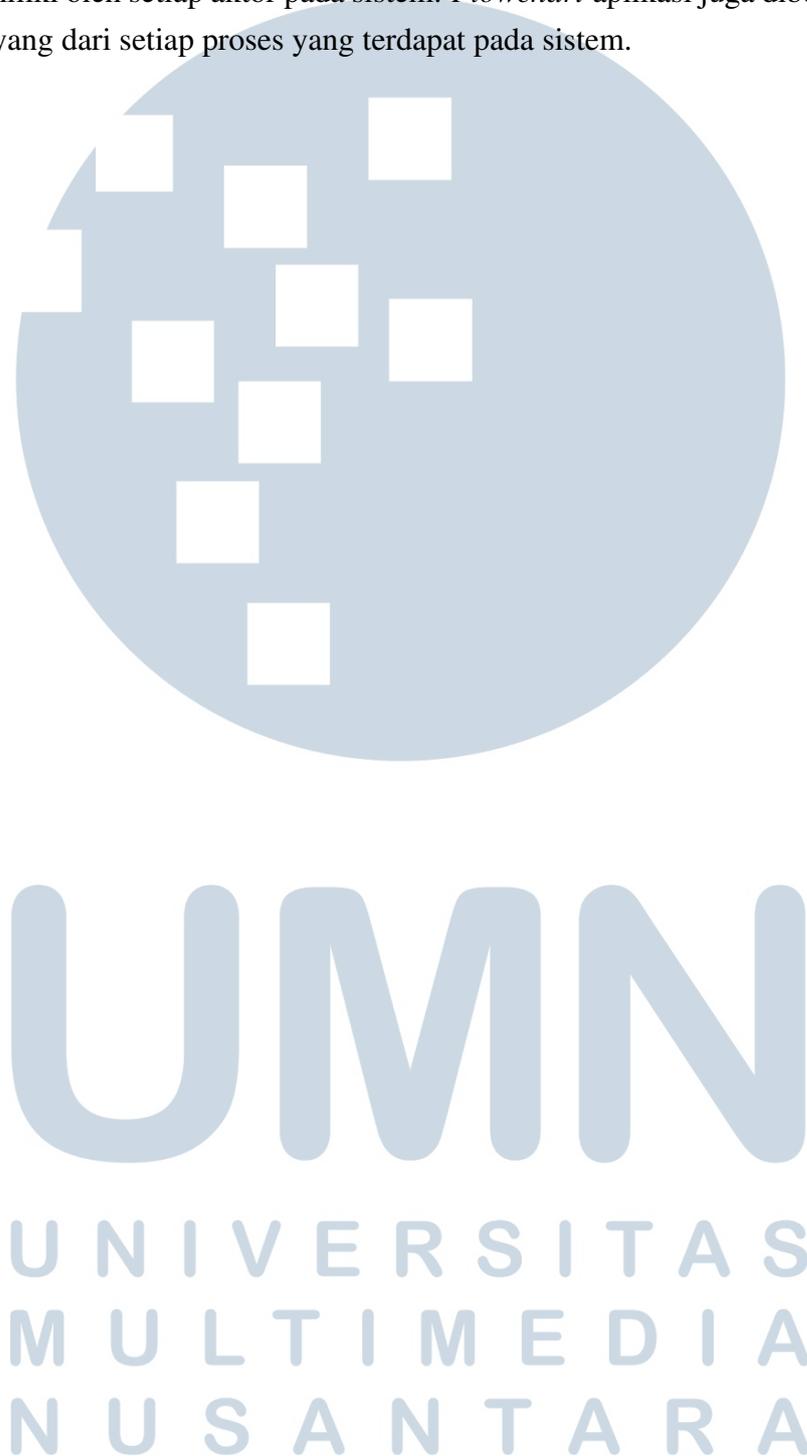
- HR dapat melihat pengajuan *medical reimburse* karyawan yang sesuai dengan jabatan pers admin yang dimiliki oleh HR.
- HR dapat melakukan persetujuan terhadap pengajuan *medical reimburse* karyawan.
- HR dapat melakukan penolakan dan memberikan alasan penolakan terhadap pengajuan *medical reimburse* karyawan.
- HR dapat membuat dan melihat bukti kas pengajuan *medical reimburse* karyawan.
- HR dapat membuat dan melihat lampiran detail pengajuan *medical reimburse* karyawan.
- HR dapat menerima notifikasi *email* dan *whatsapp* bila terdapat pengajuan *medical reimburse* oleh karyawan.

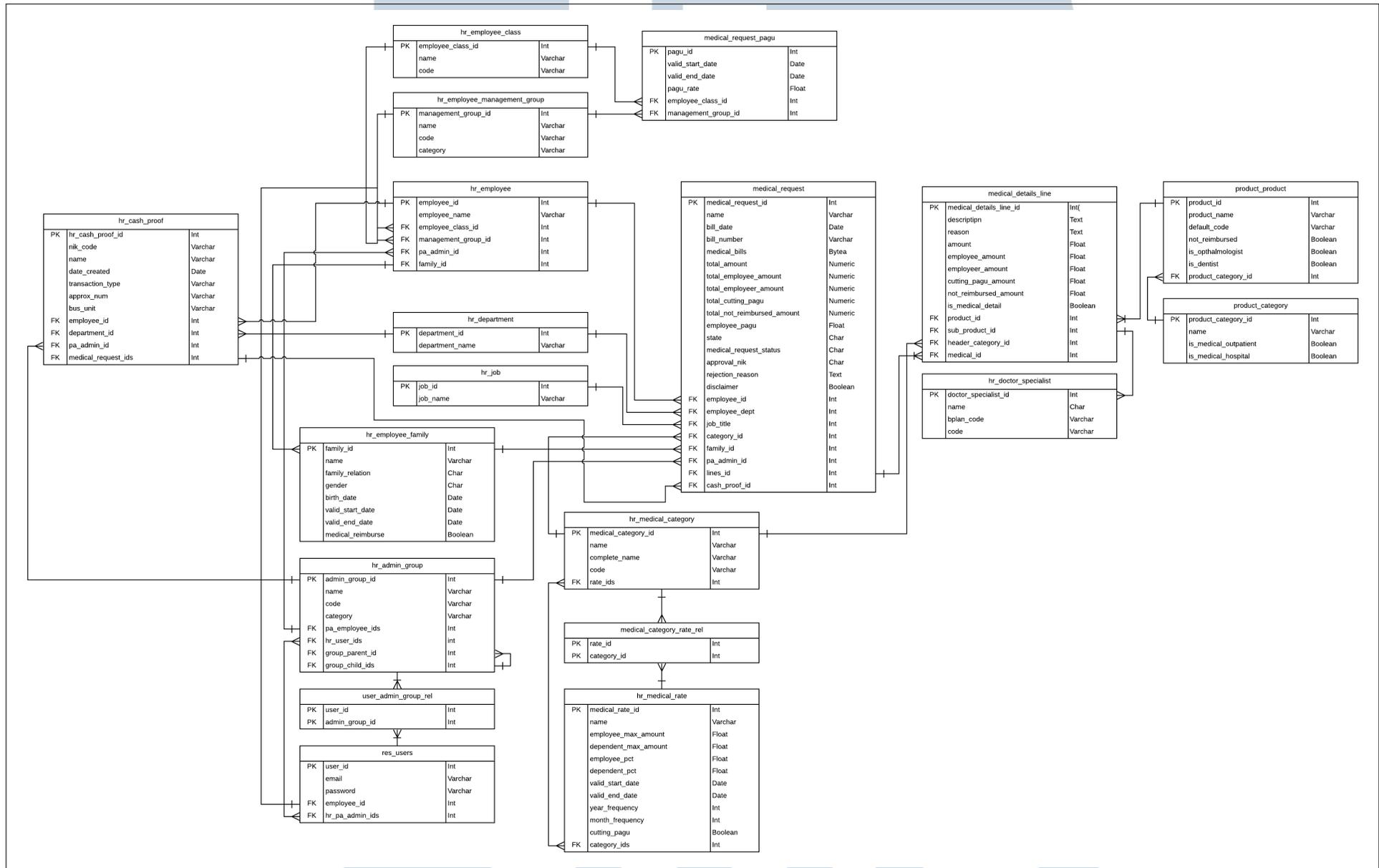
## 3. Karyawan

- Karyawan dapat melakukan pengajuan *medical reimburse* kepada sistem.
- Karyawan dapat melakukan pengajuan *medical reimburse* keluarga kepada sistem.
- Karyawan dapat melihat *history* pengajuan *medical reimburse* milik karyawan itu sendiri.
- Karyawan mendapatkan notifikasi *email* atau *whatsapp* bila pengajuan *medical reimburse* diterima atau ditolak.
- Karyawan dapat melihat status *approval* terhadap pengajuan *medical reimburse* yang dilakukan.
- Karyawan dapat melihat detail pengajuan *medical reimburse* yang telah dilakukan.
- Karyawan dapat melihat total biaya pergantian perusahaan.
- Karyawan dapat melihat total biaya yang ditanggung sendiri oleh karyawan.
- Karyawan dapat melihat total biaya yang memotong saldo pagu karyawan.
- Karyawan dapat melihat total biaya yang tidak ditanggung oleh perusahaan.
- Karyawan dapat melihat sisa saldo pagu karyawan yang telah dipakai pada pengajuan *medical reimburse* selanjutnya.
- Karyawan dapat melakukan pengajuan *medical reimburse* terhadap produk kesehatan yang tidak memotong pagu bila saldo pagu karyawan sudah habis.

### 3.2 Perancangan Sistem

Proses rancangan sistem diawali dengan membentuk struktur basis data dari sistem *medical reimburse*. Selanjutnya, dibentuk DFD (*Data Flow Diagram*) untuk menentukan alur informasi yang dimiliki oleh setiap aktor pada sistem. *Flowchart* aplikasi juga dibentuk untuk melihat alur kerja yang dari setiap proses yang terdapat pada sistem.





Gambar 3.1. Struktur basis data sistem *medical reimburse*

### 3.2.1 Struktur Basis Data

Gambar 3.1 merupakan struktur dan relasi tabel yang dimiliki oleh sistem *medical reimburse*. Terdapat 19 tabel dengan fungsi dan relasi yang saling berkaitan untuk menyimpan data-data yang dimiliki oleh sistem. Tabel-tabel tersebut antara lain, *medical\_request*, *medical\_details\_line*, *hr\_medical\_rate*, *hr\_medical\_category*, *medical\_request\_pagu*, *medical\_category\_rate\_rel* yang merupakan tabel penghubung dari relasi *many to many* tabel *hr\_medical\_category* dan *hr\_medical\_rate*, *product\_product*, *product\_category*, *hr\_doctor\_specialist*, *hr\_employee*, *hr\_employee\_class*, *hr\_employee\_management\_group*, *hr\_department*, *hr\_job*, *hr\_employee\_family*, *hr\_admin\_group*, *res\_users*, *user\_admin\_group\_rel* yang merupakan tabel penghubung dari relasi *many to many* tabel *hr\_admin\_group* dan *res\_users*, dan *hr\_cash\_proof*. Berikut penjelasan dari masing-masing tabel beserta data yang disimpan.

#### A Tabel Medical Request

Tabel *medical\_request* merupakan tabel utama yang menyimpan data dan informasi yang dibutuhkan karyawan untuk mengajukan pergantian kesehatan. Tabel *medical\_request* memiliki satu buah *primary key* yaitu *medical\_request\_id*. Tujuh buah *foreign key*, dengan relasi *many to one* yang terdiri dari *employee\_id* dengan tabel *hr\_employee*, *employee\_dept* dengan tabel *hr\_department*, *job\_title* dengan tabel *hr\_job*, *category\_id* dengan tabel *hr\_medical\_category*, *family\_id* dengan tabel *hr\_employee\_family*, *pa\_admin\_id* dengan tabel *hr\_admin\_group*, *cash\_proof\_id* dengan tabel *hr\_cash\_proof*. Satu buah *foreign key* dengan relasi *one to many* yaitu *lines\_id* dengan tabel *medical\_details\_line*. Tabel 3.1 menunjukkan struktur data yang terdapat di dalam tabel *medical\_request*.

Tabel 3.1. Medical Request

Key	Nama Kolom	Tipe Data	Keterangan
PK	<i>medical_request_id</i>	Integer	<i>Primary key</i> tabel <i>medical request</i>
	<i>name</i>	Varchar	Nomor <i>sequence</i> untuk setiap pengajuan
	<i>bill_date</i>	Date	Tanggal kwitansi

Tabel 3.1. Medical Request (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
	bill_number	Varchar	Nomor kwitansi
	medical_bills	Bytea	Foto kwitansi
	total_amount	Numeric	Total biaya pengajuan
	total_employee_amount	Numeric	Total biaya ditanggung karyawan
	total_employer_amount	Numeric	Total biaya ditanggung perusahaan
	total_cutting_pagu	Numeric	Total biaya pengajuan yang memotong pagu
	total_not_reimbursed_amount	Numeric	Total biaya yang tidak ditanggung perusahaan
	employee_pagu	Float	Saldo pagu karyawan
	state	Char	Tahap <i>form</i> pengajuan kesehatan
	medical_request_status	Char	Status <i>approval form</i> pengajuan kesehatan
	approval_nik	Char	NIK karyawan yang melakukan <i>approval</i>
	rejection_reason	Text	Alasan penolakan pengajuan kesehatan
	disclaimer	Boolean	Ketentuan dan kondisi pengajuan kesehatan
FK	employee_id	Integer	Karyawan yang melakukan pengajuan kesehatan
FK	employee_dept	Integer	Departemen karyawan
FK	job_title	Integer	Kedudukan pekerjaan karyawan
FK	category_id	Integer	Kategori pengajuan kesehatan
FK	family_id	Integer	Keluarga karyawan
FK	pa_admin_id	Integer	Admin grup yang dimiliki karyawan
FK	lines_id	Integer	Produk-produk yang diajukan

Tabel 3.1. Medical Request (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
FK	cash_proof_id	Integer	ID bukti kas

## B Tabel Medical Details Line

Tabel `medical_details_line` merupakan tabel yang menyimpan seluruh data detail dari produk yang akan diajukan pada proses pengajuan kesehatan. Tabel ini memiliki satu *primary key* yaitu `medical_details_line_id`. Tiga buah *foreign key* dengan relasi *many to one* yaitu `product_id` dengan tabel `product_product`, `header_category_id` dengan tabel `hr_medical_category`, `medical_id` dengan tabel `medical_request`. Satu buah *foreign key* dengan relasi *one to many* yaitu `sub_product_id` dengan tabel `hr_doctor_specialist`. Tabel 3.2 menunjukkan data yang terdapat pada tabel `medical_details_line`.

Tabel 3.2. Medical Details Line

Key	Nama Kolom	Tipe Data	Keterangan
PK	<code>medical_details_line_id</code>	Integer	<i>Primary key</i> untuk tabel <code>medical_details_line</code>
	<code>description</code>	Text	Deskripsi produk yang diajukan
	<code>reason</code>	Text	Alasan pengajuan pada produk
	<code>amount</code>	Float	Jumlah biaya pengajuan pada produk
	<code>employee_amount</code>	Float	Biaya pergantian produk ditanggung karyawan
	<code>employeer_amount</code>	Float	Biaya pergantian produk ditanggung perusahaan
	<code>cutting_pagu_amount</code>	Float	Jumlah biaya yang memotong pagu karyawan
	<code>not_reimbursed_amount</code>	Float	Biaya produk yang tidak digantikan

Tabel 3.2. Medical Details Line (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
	is_medical_detail	Boolean	Kondisi untuk pemilihan produk dokter spesialis
FK	product_id	Integer	Produk-produk kesehatan yang dapat diajukan
FK	sub_product_id	Integer	Produk kesehatan untuk dokter spesialis
FK	header_category_id	Integer	Kategori pengajuan kesehatan
FK	medical_id	Integer	ID <i>form header</i> pengajuan kesehatan

### C Tabel HR Medical Rate

Tabel *hr\_medical\_rate* merupakan tabel yang menyimpan data tarif pergantian untuk keseluruhan produk kesehatan yang dibedakan berdasarkan golongan dan manajemen grup. Tabel tersebut memiliki satu buah *primary key* yaitu *medical\_rate\_id* dan satu buah *foreign key* dengan relasi *many to many* pada tabel *hr\_medical\_category*. Tabel 3.3 menunjukkan data yang terdapat pada tabel *hr\_medical\_rate*.

Tabel 3.3. Medical Rate

Key	Nama Kolom	Tipe Data	Keterangan
PK	medical_rate_id	Integer	<i>Primary key</i> untuk tabel <i>hr_medical_rate</i>
	name	Varchar	Nomor sequence untuk setiap tarif produk
	employee_max_amount	Float	Tarif maksimal pergantian untuk karyawan
	dependent_max_amount	Float	Tarif maksimal pergantian untuk keluarga karyawan

Tabel 3.3. Medical Rate (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
	employee_pct	Float	Persentase pergantian untuk karyawan
	dependent_pct	Float	Persentase pergantian untuk keluarga karyawan
	valid_start_date	Date	Tanggal awal tarif
	valid_end_date	Date	Tanggal akhir tarif
	year_frequency	Integer	Frekuensi tahunan untuk produk kesehatan
	month_frequency	Integer	Frekuensi bulanan untuk produk kesehatan
	cutting_pagu	Boolean	Produk potong pagu atau tidak
FK	category_ids	Integer	Kategori kesehatan yang dimiliki oleh setiap tarif produk

#### D Tabel HR Medical Category

Tabel *hr\_medical\_category* merupakan tabel yang menyimpan nama dan kode kategori pengajuan kesehatan beserta tarif-tarif produk yang dimiliki oleh setiap kategori. Tabel tersebut memiliki satu *primary key* yaitu *medical\_category\_id* dan satu buah *foreign key* yaitu *rate\_ids* dengan relasi *many to many* yang berhubungan pada tabel *hr\_medical\_rate*. Tabel 3.4 menunjukkan data yang terdapat pada tabel *hr\_medical\_category*.

Tabel 3.4. Medical Category

Key	Nama Kolom	Tipe Data	Keterangan
PK	medical_category_id	Integer	Primary key untuk tabel <i>hr_medical_category</i>
	name	Varchar	Nama unik kategori

Tabel 3.4. Medical Category (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
	complete_name	Varchar	Nama lengkap kategori
	code	Varchar	Kode kategori
FK	rate_ids	Integer	Tarif-tarif produk yang dimiliki setiap kategori

#### E Tabel Medical Category Rate Rel

Tabel `medical_category_rate_rel` merupakan tabel yang menyimpan data id tarif dan id kategori yang dibentuk dari relasi *many to many* tabel `hr_medical_category` dan `hr_medical_rate`. Tabel ini memiliki dua buah *primary key* yang diambil dari dua buah tabel sebelum yang memiliki hubungan. *Primary key* tersebut yaitu `rate_id` dan `category_id`. Tabel 3.5 menunjukkan data yang terdapat pada tabel `medical_category_rate_rel`.

Tabel 3.5. Medical Category Rate Rel

Key	Nama Kolom	Tipe Data	Keterangan
PK	rate_id	Integer	ID <i>rate</i> dari tabel <i>hr_medical_rate</i>
PK	category_id	Integer	ID kategori dari tabel <i>hr_medical_category</i>

#### F Tabel HR Doctor Specialist

Tabel `hr_doctor_specialist` merupakan tabel yang menyimpan nama dan kode dokter spesialis beserta produk kesehatan yang berhubungan dengan dokter spesialis. Tabel ini memiliki satu buah *primary key* yaitu `doctor_specialist_id`. Tabel 3.6 menunjukkan data yang terdapat pada tabel `hr_doctor_specialist`.

Tabel 3.6. HR Doctor Specialist

Key	Nama Kolom	Tipe Data	Keterangan
PK	doctor_specialist_id	Integer	<i>Primary key</i> untuk tabel <i>hr_doctor_specialist</i>
	name	Varchar	Nama dokter spesialis atau produk dokter spesialis
	bplan_code	Varchar	Kode produk kesehatan dokter spesialis
	code	Varchar	Kode dokter spesialis atau produk dokter spesialis

### G Tabel Medical Request Pagu

Tabel *medical\_request\_pagu* merupakan tabel yang menyimpan tarif pagu karyawan yang dibedakan berdasarkan golongan dan manajemen grup. Tabel ini memiliki satu buah *primary key* yaitu *pagu\_id*. Dua buah *foreign key* dengan relasi *many to one* yaitu *employee\_class\_id* pada tabel *hr\_employee\_class* dan *management\_group\_id* pada tabel *hr\_employee\_management\_group*. Tabel 3.7 menunjukkan data yang terdapat pada tabel *medical\_request\_pagu*.

Tabel 3.7. Medical Request Pagu

Key	Nama Kolom	Tipe Data	Keterangan
PK	pagu_id	Integer	<i>Primary key</i> untuk tabel <i>medical_request_pagu</i>
	valid_start_date	Date	Tanggal awal tarif pagu
	valid_end_date	Date	Tanggal akhir tarif pagu
	pagu_rate	Float	Tarif pagu karyawan
FK	employee_class_id	Integer	Golongan karyawan
FK	management_group_id	Integer	Manajemen grup karyawan

## H Tabel Product Product

Tabel `product_product` merupakan tabel yang menyimpan data produk-produk kesehatan yang ditampilkan pada *form* pengajuan. Tabel ini memiliki satu buah *primary key* yaitu `product_id` dan satu buah *foreign key* yang memiliki relasi *many to one* dengan tabel `product_category`. Tabel 3.8 menunjukkan data yang terdapat pada tabel `product_product`.

Tabel 3.8. Product Product

Key	Nama Kolom	Tipe Data	Keterangan
PK	<code>product_id</code>	Integer	<i>Primary key</i> untuk tabel <code>product_product</code>
	<code>product_name</code>	Varchar	Nama produk kesehatan
	<code>default_code</code>	Varchar	Kode produk kesehatan
	<code>not_reimbursed</code>	Boolean	Produk kesehatan yang digantikan atau tidak
	<code>is_opthalmologist</code>	Boolean	Produk kesehatan dokter spesialis mata atau tidak
	<code>is_dentist</code>	Boolean	Produk kesehatan dokter gigi atau tidak
FK	<code>product_category</code>	Integer	Kategori umum dari produk kesehatan

## I Tabel Product Category

Tabel `product_category` merupakan tabel yang menyimpan data kategori untuk produk kesehatan, namun kategori yang dimaksud merupakan kategori umum. Tabel ini diperlukan karena tabel bawaan yang dimiliki oleh Odoo untuk menyimpan kategori produk. Tabel `hr_medical_category` merupakan tabel tambahan yang dibuat untuk membuat kategori baru yang akan digunakan pada pengajuan kesehatan. Sehingga kategori yang dimiliki oleh tabel ini merupakan kategori umum yang diimplementasikan pada semua data produk kesehatan. Tabel `product_category` memiliki satu buah *primary key* yaitu `product_category_id`. Tabel 3.9 menunjukkan data yang terdapat pada tabel `product_category`.

Tabel 3.9. Product Category

Key	Nama Kolom	Tipe Data	Keterangan
PK	product_category_id	Integer	Primary key untuk tabel <i>product_category</i>
	name	Varchar	Nama kategori
	is_medical_outpatient	Boolean	Produk rawat jalan atau tidak
	is_medical_hospital	Boolean	Produk rawat inap atau tidak

## J Tabel HR Employee

Tabel *hr\_employee* merupakan tabel yang menyimpan data karyawan yang diperlukan untuk melakukan pengajuan kesehatan. Tabel ini memiliki satu buah *primary key* yaitu *employee\_id*. Tiga buah *foreign key* yang memiliki relasi *many to one* yaitu *employee\_class\_id* pada tabel *hr\_employee\_class*, *management\_group\_id* pada tabel *hr\_employee\_management\_group*, dan *pa\_admin\_id* pada tabel *hr\_admin\_group*. Satu buah *foreign key* dengan relasi *one to many* pada tabel *hr\_employee\_family* yaitu *family\_id*. Tabel 3.10 menunjukkan data yang terdapat pada tabel *hr\_employee*.

Tabel 3.10. HR Employee

Key	Nama Kolom	Tipe Data	Keterangan
PK	employee_id	Integer	Primary key untuk tabel <i>hr_employee</i>
	employee_name	Varchar	Nama karyawan
FK	employee_class_id	Integer	Golongan karyawan
FK	management_group_id	Integer	Manajemen grup karyawan
FK	pa_admin_id	Integer	HR pers admin yang dimiliki karyawan
FK	family_id	Integer	Keluarga karyawan

## K Tabel HR Employee Family

Tabel `hr_employee_family` merupakan tabel yang menyimpan seluruh data yang berhubungan dengan keluarga yang dimiliki karyawan. Tabel ini memiliki satu buah *primary key* yaitu `family_id`. Tabel 3.11 menunjukkan data yang terdapat pada tabel `hr_employee_family`.

Tabel 3.11. HR Employee Family

Key	Nama Kolom	Tipe Data	Keterangan
PK	<code>family_id</code>	Integer	<i>Primary key</i> untuk tabel <code>hr_employee_family</code>
	<code>name</code>	Varchar	Nama keluarga karyawan
	<code>family_relation</code>	Char	Status hubungan keluarga
	<code>gender</code>	Char	Jenis kelamin
	<code>birth_date</code>	Date	Tanggal lahir
	<code>valid_start_date</code>	Date	Tanggal awal keluarga dapat melakukan pengajuan
	<code>valid_end_date</code>	Date	Tanggal akhir batas keluarga melakukan pengajuan
	<code>medical_reimburse</code>	Boolean	Boleh melakukan pengajuan kesehatan atau tidak

## L Tabel HR Employee Class

Tabel `hr_employee_class` merupakan tabel yang menyimpan nama dan kode golongan yang dimiliki oleh karyawan. Tabel ini memiliki satu buah *primary key* yaitu `employee_class_id`. Tabel 3.12 menunjukkan data yang terdapat pada tabel `hr_employee_class`.

Tabel 3.12. HR Employee Class

Key	Nama Kolom	Tipe Data	Keterangan
PK	employee_class_id	Integer	<i>Primary key</i> untuk tabel <i>hr_employee_class</i>
	name	Varchar	Nama golongan
	code	Varchar	Kode golongan

### M Tabel HR Employee Management Group

Tabel *hr\_employee\_management\_group* merupakan tabel yang menyimpan nama dan kode manajemen grup yang dimiliki oleh karyawan. Tabel ini memiliki satu buah *primary key* yaitu *management\_group\_id*. Tabel 3.13 menunjukkan data yang terdapat pada tabel *hr\_employee\_management\_group*.

Tabel 3.13. HR Employee Management Group

Key	Nama Kolom	Tipe Data	Keterangan
PK	management_group_id	Integer	<i>Primary key</i> untuk tabel <i>hr_employee_management_group</i>
	name	Varchar	Nama manajemen grup
	code	Varchar	Kode manajemen grup
	category	Varchar	Kategori manajemen grup

### N Tabel HR Department

Tabel *hr\_department* merupakan tabel yang menyimpan data-data departemen tempat karyawan bekerja. Tabel ini memiliki satu buah *primary key* yaitu *department\_id*. Tabel 3.14 menunjukkan data yang terdapat pada tabel *hr\_department*.

Tabel 3.14. HR Department

Key	Nama Kolom	Tipe Data	Keterangan
PK	department_id	Integer	<i>Primary key</i> untuk tabel <i>hr_department</i>
	department_name	Varchar	Nama departemen

### O Tabel HR Job

Tabel *hr\_job* merupakan tabel yang menyimpan data jabatan pekerjaan yang dapat dimiliki oleh karyawan. Tabel ini memiliki satu buah *primary key* yaitu *job\_id*. Tabel 3.15 menunjukkan data yang terdapat pada tabel *hr\_job*.

Tabel 3.15. HR Job

Key	Nama Kolom	Tipe Data	Keterangan
PK	job_id	Integer	<i>Primary key</i> untuk tabel <i>hr_job</i>
	job_name	Varchar	Jabatan pekerjaan

### P Tabel HR Admin Group

Tabel *hr\_admin\_group* merupakan tabel yang menyimpan data *pers admin* yang bertanggung jawab memegang suatu unit bisnis. Hal ini diperlukan untuk keperluan *approval* pengajuan kesehatan yang dilakukan oleh karyawan. Tabel ini memiliki satu buah *primary key* yaitu *admin\_group\_id*. Satu buah *foreign key* dengan relasi *one to many* pada tabel *hr\_employee*. Satu buah *foreign key* dengan relasi *many to many* pada tabel *res\_users*. Dua buah *foreign key* yang saling memiliki relasi pada tabel *hr\_admin\_group*, yaitu *group\_parent\_id* dan *group\_child\_ids*. Kedua *key* tersebut dibuat karena pada satu *pers admin* dapat memegang beberapa unit bisnis di bawahnya yang menjadi anak dari suatu unit bisnis tertentu. Tabel 3.16 menunjukkan data yang terdapat pada tabel *hr\_admin\_group*.

Tabel 3.16. HR Admin Group

Key	Nama Kolom	Tipe Data	Keterangan
PK	admin_group_id	Integer	Primary key untuk tabel <i>hr_admin_group</i>
	name	Varchar	Nama pers admin
	code	Varchar	Kode pers admin
	category	Varchar	Kategori pers admin
FK	pa_employee_ids	Integer	Karyawan yang masuk kedalam pers admin
FK	group_parent_id	Integer	Parent pers admin
FK	group_child_ids	Integer	Child yang menjadi bagian dari parent pers admin
FK	hr_user_ids	Integer	User yang mendapat akses menjadi pers admin

### Q Tabel Res Users

Tabel *res\_users* merupakan tabel yang menyimpan data pengguna yang melakukan *login* ke dalam Odoo, seperti *email* dan *password*. Tabel ini memiliki satu buah *primary key* yaitu *user\_id*. Satu buah *foreign key* dengan relasi *one to one* pada tabel *hr\_employee* yaitu *employee\_id*. Satu buah *foreign key* dengan relasi *many to many* pada tabel *hr\_admin\_group* yaitu *hr\_pa\_admin\_ids*. Tabel 3.17 menunjukkan data yang terdapat pada tabel *res\_users*.

Tabel 3.17. Res Users

Key	Nama Kolom	Tipe Data	Keterangan
PK	user_id	Integer	Primary key untuk tabel <i>res_users</i>
	email	Varchar	Email pengguna
	password	Varchar	Password pengguna
FK	employee_id	Integer	Karyawan yang memiliki akun pengguna

Tabel 3.17. Res Users (lanjutan)

Key	Nama Kolom	Tipe Data	Keterangan
FK	hr_pa_admin_ids	Integer	Pers admin yang dapat dimiliki oleh pengguna

## R Tabel User Admin Group Rel

Tabel `user_admin_group_rel` merupakan tabel yang menyimpan data id pengguna dan id pers admin yang dibentuk dari relasi *many to many* tabel `hr_admin_group` dan `res_users`. Tabel ini memiliki dua buah *primary key* yang diambil dari dua buah tabel sebelum yang memiliki hubungan. *Primary key* tersebut yaitu `user_id` dan `admin_group_id`. Tabel 3.18 menunjukkan data yang terdapat pada tabel `user_admin_group_rel`.

Tabel 3.18. User Admin Group Rel

Key	Nama Kolom	Tipe Data	Keterangan
PK	<code>user_id</code>	Integer	ID pengguna dari tabel <i>res_users</i>
PK	<code>admin_group_id</code>	Integer	ID pers admin dari tabel <i>hr_admin_group</i>

## S Tabel HR Cash Proof

Tabel `hr_cash_proof` merupakan tabel yang menyimpan data bukti kas pembayaran pengajuan kesehatan yang sudah di *approve* oleh HR dan siap dilanjutkan ke tahap pembayaran pada kasir. Tabel `hr_cash_proof` memiliki satu buah *primary key* yaitu `hr_cash_proof_id`. Tiga buah *foreign key* dengan relasi *many to one* yaitu, `employee_id` pada tabel `hr_employee`, `department_id` pada tabel `hr_department`, dan `pa_admin_id` pada tabel `hr_admin_group`. Satu buah *foreign key* dengan relasi *one to many* pada tabel `medical_request` yaitu `medical_request_ids`. Tabel 3.19 menunjukkan data yang terdapat pada tabel `hr_cash_proof`.

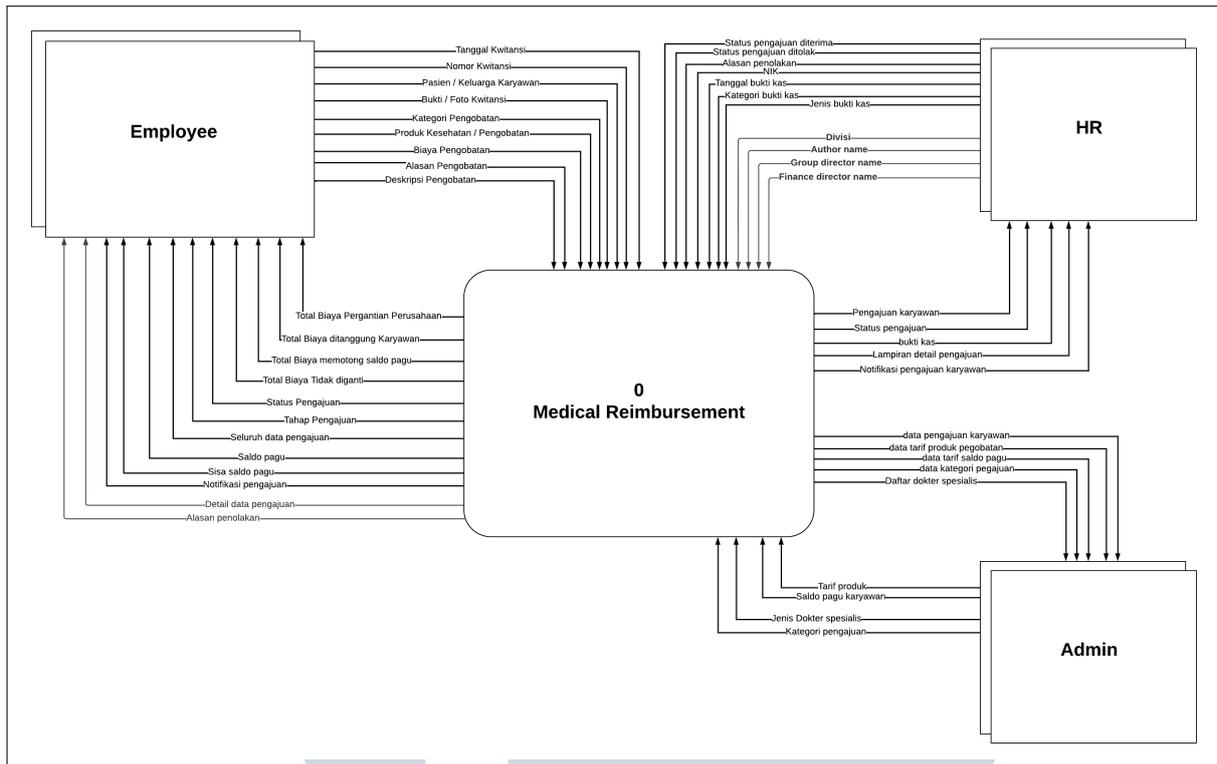
Tabel 3.19. HR Cash Proof

Key	Nama Kolom	Tipe Data	Keterangan
PK	hr_cash_proof_id	Integer	Primary key untuk tabel hr_cash_proof
	nik_code	Varchar	Kode unik bukti kas
	name	Varchar	Nomor penyusun
	date_created	Date	Tanggal bukti kas
	transaction_type	Varchar	Tipe bukti kas
	approx_num	Varchar	Approx number
	bus_unit	Varchar	Business Unit
FK	employee_id	Integer	Karyawan pembuat bukti kas
FK	department_id	Integer	Departemen pembuat bukti kas
FK	pa_admin_id	Integer	Pers admin pembuat bukti kas
FK	medical_request_ids	Integer	Pegajuan yang dijadikan bukti kas

### 3.2.2 Data Flow Diagram (DFD)

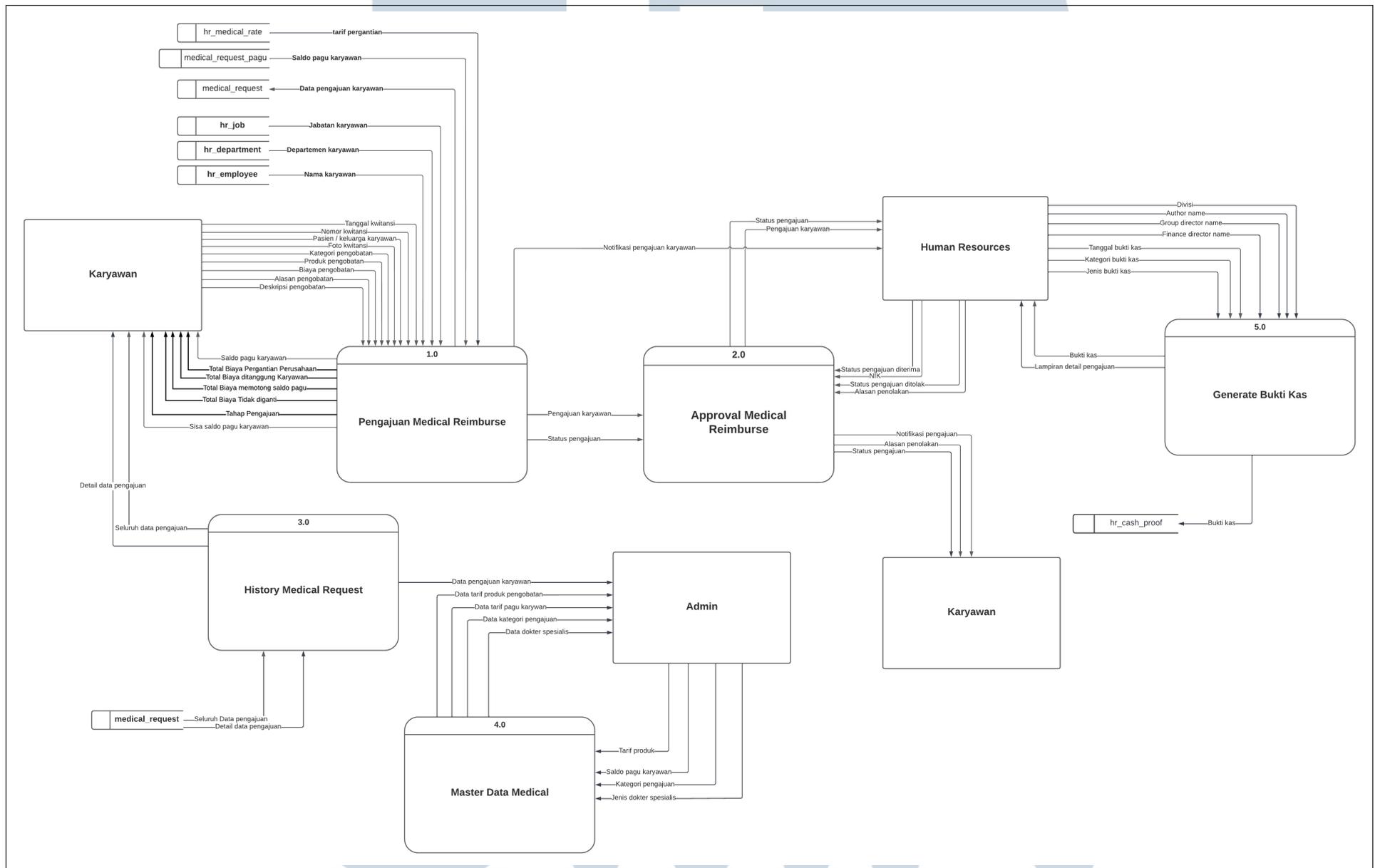
Pada bagian ini akan dijelaskan *data flow diagram* (DFD) yang dibuat untuk menjelaskan alur informasi proses yang dimiliki oleh sistem. Aliran informasi tersebut berupa data *input* dan *output* yang dimiliki oleh entitas yang menjalankan sistem. Terdapat tiga buah level *data flow diagram* (DFD) yang dibangun yaitu diagram konteks, *data flow diagram* level 1, dan *data flow diagram* level 2.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.2. Data flow diagram level 0

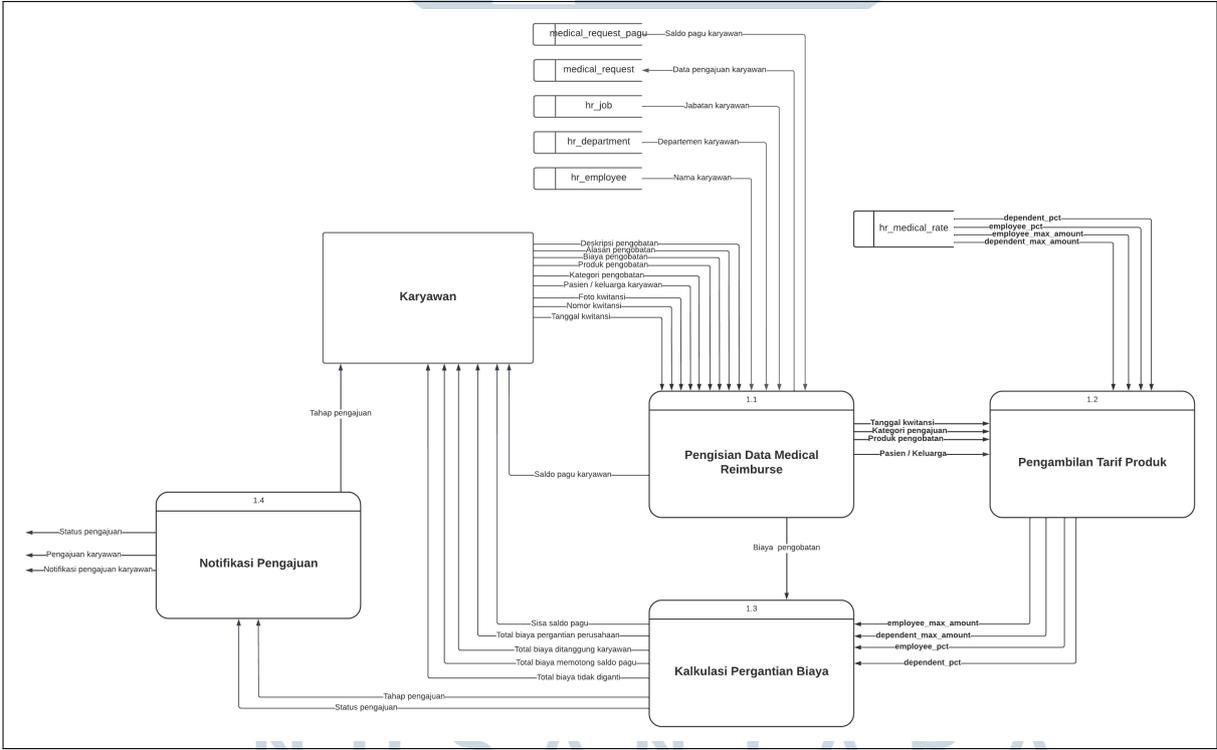
Gambar 3.2 merupakan diagram konteks yang dimiliki oleh sistem medical reimburse. Terdapat tiga buah aktor yang menerima dan memberikan informasi yang dihasilkan dan dibutuhkan oleh sistem. DFD level 0 menggambarkan gambaran besar dari sistem *medical reimburse* yang dibangun, sehingga proses yang dimiliki hanya satu proses yaitu sistem medical reimburse itu sendiri. Pada entitas karyawan alur informasi yang terjadi merupakan data-data yang dibutuhkan untuk melakukan pengajuan kesehatan. Data-data tersebut seperti nama pasien, tanggal dan bukti kwitansi pengobatan, biaya pengobatan, produk pengobatan, total pergantian biaya, status pengajuan. Pada entitas HR alur informasi yang terjadi merupakan data pengajuan kesehatan yang dilakukan karyawan, status approval yang diberikan HR, bukti kas, dan lampiran. Pada entitas admin, alur informasi yang terjadi merupakan data-data master yang diperlukan untuk melakukan pengajuan kesehatan seperti, data tarif pergantian biaya, kategori pengajuan, produk kesehatan, saldo pagu karyawan, dan seluruh data pengajuan yang telah dilakukan karyawan.



Gambar 3.3. Data flow diagram level 1 sistem *medical reimburse*

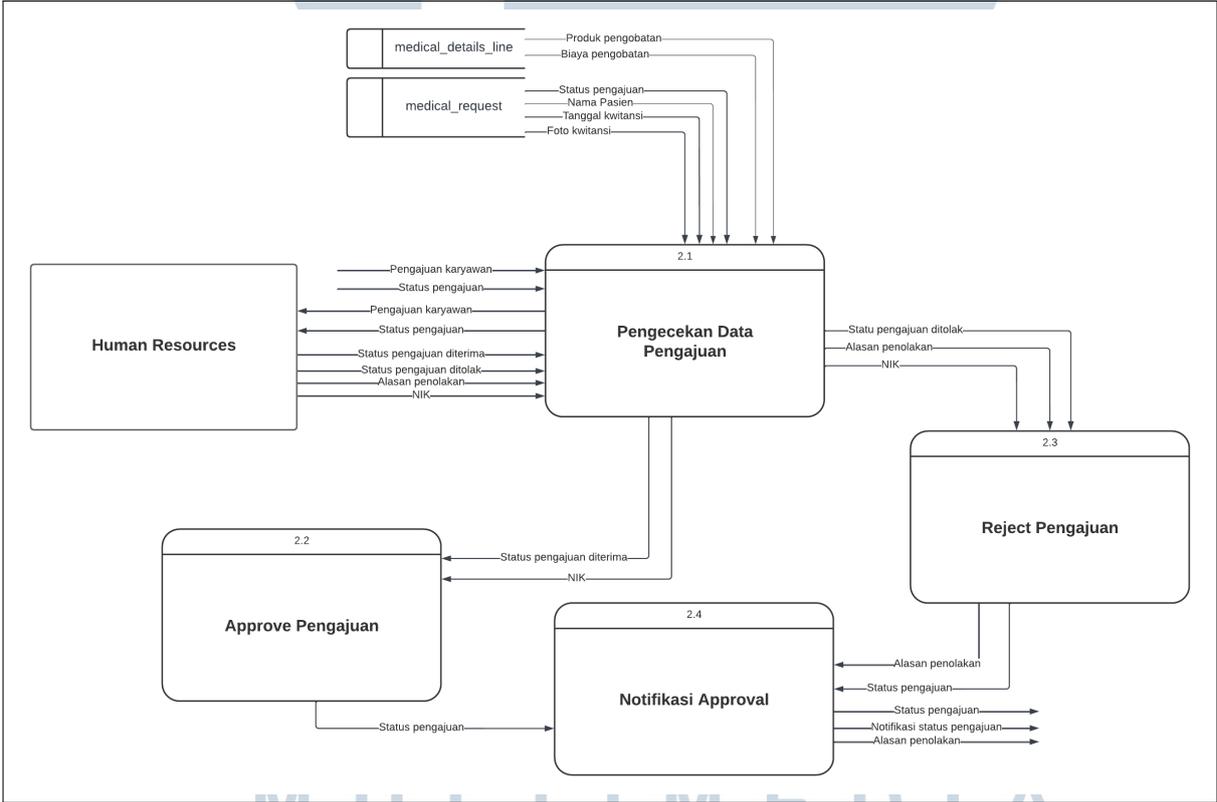
Pada Gambar 3.3, *data flow diagram* level 1 menunjukkan lima buah proses yang dibuat untuk menggambarkan secara lebih detail alur informasi terhadap proses yang dimiliki oleh sistem *medical reimburse* yang dibangun. Kelima proses tersebut yaitu proses pengajuan kesehatan, proses *approval* pengajuan kesehatan, proses *reject* pengajuan kesehatan, proses histori pengajuan, dan proses *generate* bukti kas. Proses pengajuan kesehatan merupakan proses karyawan melakukan pengajuan pergantian biaya pengobatan terhadap sistem dengan mengisi data-data yang dibutuhkan untuk melakukan pengajuan. Proses *approval* dan *reject* pengajuan merupakan proses yang dilakukan oleh HR untuk memberikan status persetujuan terhadap pengajuan yang dilakukan karyawan. Proses akan menerima data pengajuan yang dilakukan oleh karyawan, selanjutnya HR akan memeriksa data pengajuan yang dibuat dan memberikan status persetujuan kepada pengajuan yang dibuat.

Proses histori pengajuan digunakan untuk menampilkan seluruh pengajuan yang telah dilakukan karyawan. Proses *generate* bukti kas merupakan proses untuk membuat bukti kas terhadap pengajuan yang telah diterima untuk dilakukan pergantian biaya kepada karyawan. Selain bukti kas, terdapat lampiran yang digunakan untuk memberikan detail produk pengobatan yang diajukan untuk diberikan pergantian biaya. Pada *data flow diagram* level 1, terdapat pula tujuh buah tabel yang digunakan untuk menyimpan dan mengambil data yang dibutuhkan pada sistem.



Gambar 3.4. *Data flow diagram* level 2 pengajuan kesehatan

Pada Gambar 3.4 merupakan *data flow diagram* level 2 untuk proses pengajuan kesehatan karyawan. Terdapat empat buah proses yang dilakukan yaitu proses pengisian data pengajuan, proses pengambilan tarif pergantian biaya, dan proses kalkulasi pergantian biaya, dan proses hasil dan notifikasi. Proses pengisian data pengajuan merupakan proses karyawan memberi data masukan yang dibutuhkan untuk melakukan pengajuan kesehatan seperti nama pasien, bukti dan tanggal kwitansi, kategori pengajuan, biaya pengobatan, dan produk pengobatan. Selanjutnya produk pengobatan, pasien / keluarga, dan tanggal kwitansi yang diterima akan digunakan untuk mengambil tarif pergantian biaya untuk produk yang diajukan. Tarif pergantian biaya yang diterima akan digunakan pada proses kalkulasi untuk menghitung total pergantian biaya yang diberikan. Pada proses kalkulasi akan dilakukan perhitungan menggunakan biaya pengobatan dan tarif pergantian yang telah diterima. Proses ini akan menghasilkan total pergantian biaya yang diganti perusahaan, total biaya yang ditanggung karyawan, total biaya yang tidak digantikan, total biaya yang tidak memotong saldo karyawan, dan sisa saldo karyawan. Selanjutnya dilakukan proses memperlihatkan hasil total pergantian biaya kepada karyawan dan notifikasi pengajuan karyawan kepada HR.



Gambar 3.5. *Data flow diagram* level 2 persetujuan dan penolakan pengajuan

Pada Gambar 3.5 digambarkan *data flow diagram* level 2 untuk proses *approval* dan *reject* pengajuan kesehatan. Terdapat empat buah proses yang dilakukan yaitu proses

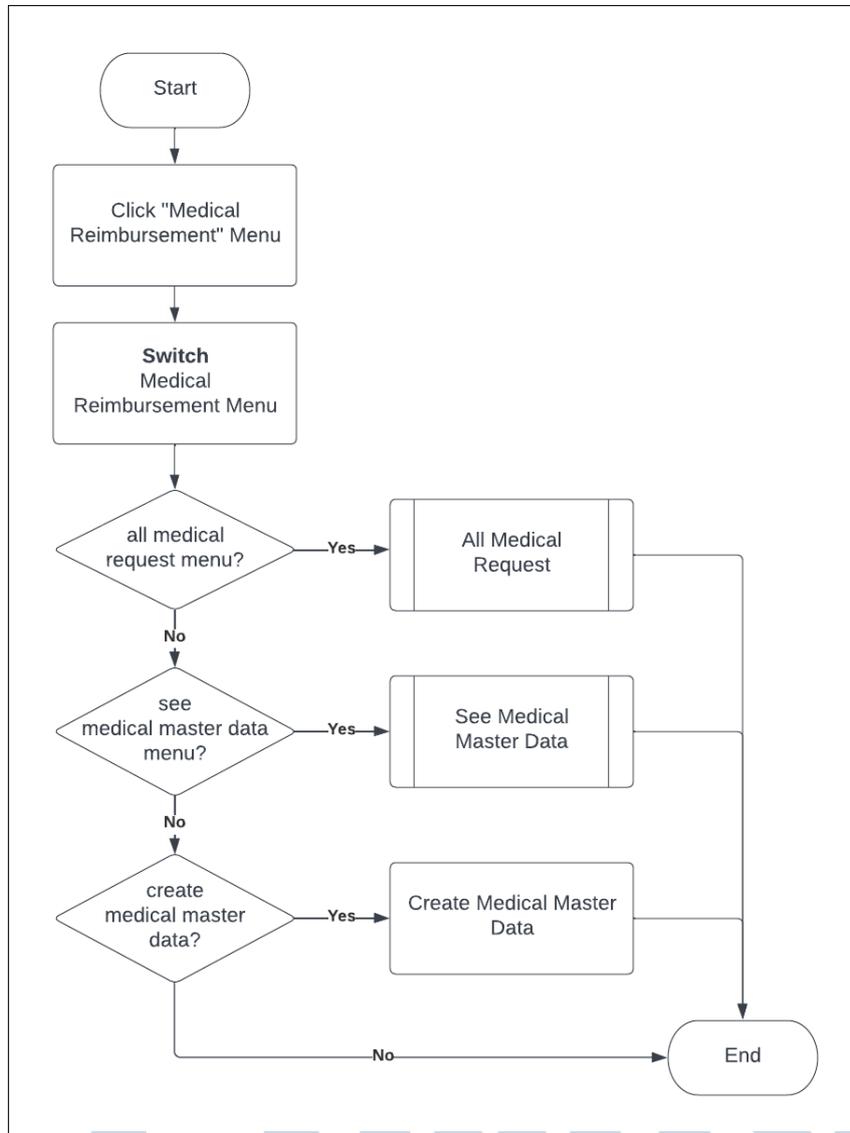
pengecekan data pengajuan dan proses persetujuan pengajuan, proses penolakan pengajuan, dan proses notifikasi status pengajuan. Pengecekan data pengajuan akan mengambil data-data seperti tanggal dan bukti kwitansi, biaya pengobatan, produk pengobatan, nama pasien, yang selanjutnya dilakukan pencocokan antara data tersebut untuk mengecek kebenaran pengajuan kesehatan yang dilakukan. Selanjutnya bila data yang dicek valid maka HR akan memberikan status pengajuan diterima dan NIK (Nomor Induk Karyawan) HR yang melakukan penerimaan pengajuan, namun bila data yang dicek tidak valid maka HR akan memberikan status pengajuan ditolak, alasan penolakan, dan NIK (Nomor Induk Karyawan) HR yang melakukan penolakan. Setelah memberikan status *approval* terhadap data pengajuan yang diterima, selanjutnya dilakukan proses pengiriman notifikasi status pengajuan kepada karyawan terhadap data pengajuan yang diajukan.

### 3.2.3 Diagram Alir

Pada bagian ini akan dijelaskan diagram alir yang telah dibuat untuk menggambarkan alur kerja dari sistem *medical reimburse* yang dibangun. Diagram alir yang dibangun mencakup diagram alir utama, *history* pengajuan kesehatan, proses pengajuan kesehatan, proses persetujuan pengajuan kesehatan, dan proses pembuatan bukti kas / lampiran.

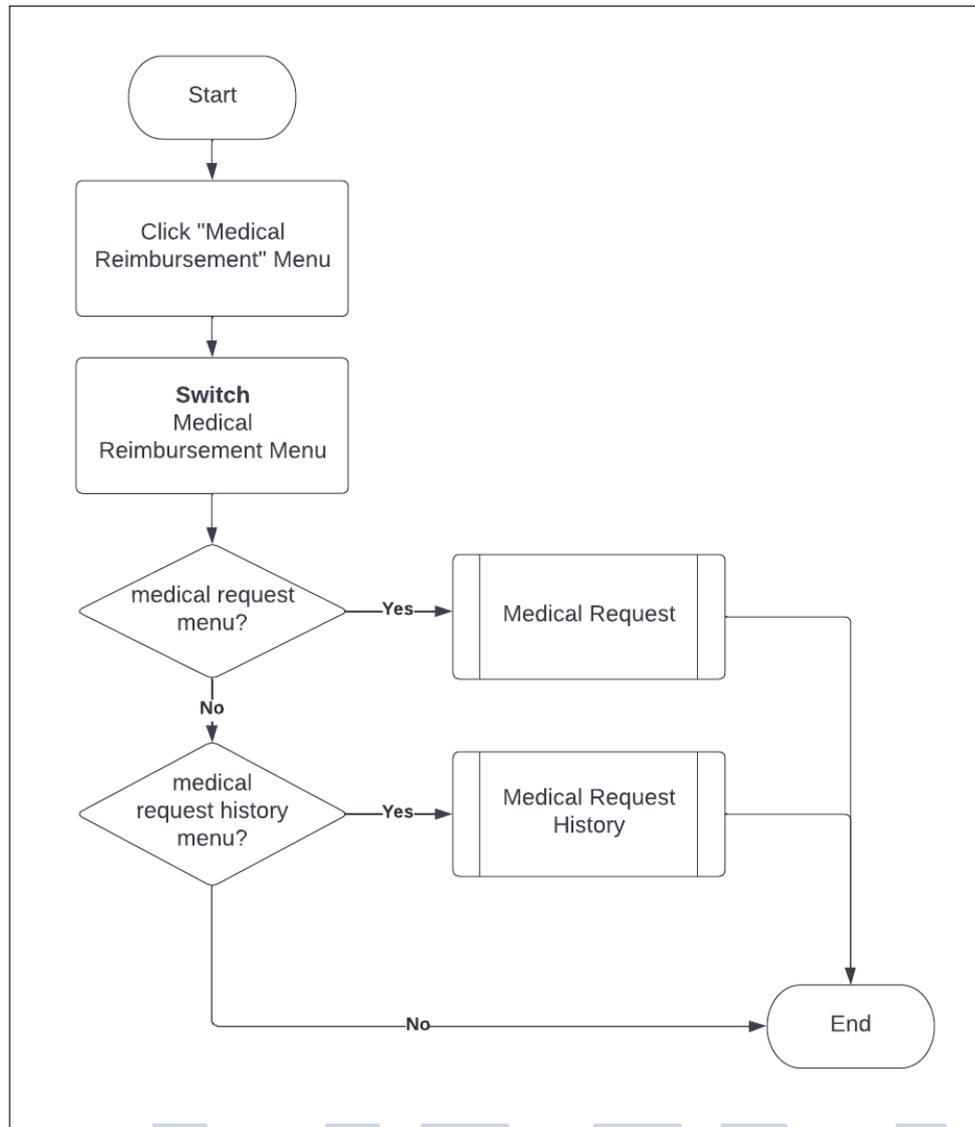


## A Diagram Alir Utama Sistem



Gambar 3.6. Diagram alir utama admin

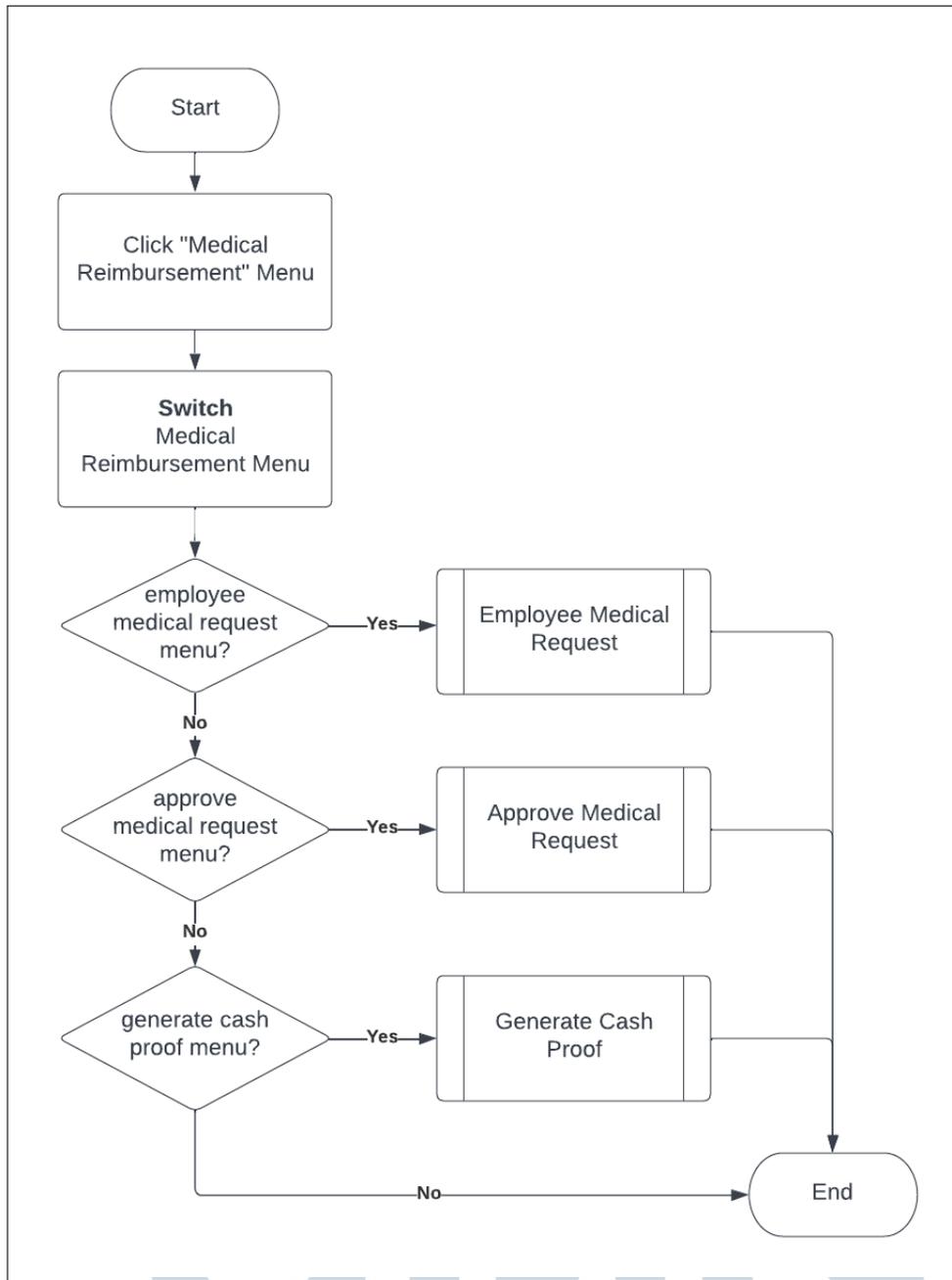
Gambar 3.6 menggambarkan diagram alir utama untuk admin. Terdapat tiga buah proses utama yang dilakukan admin pada sistem *medical reimburse* yang dibangun yaitu *history* pengajuan kesehatan seluruh karyawan dan membuat *master data medical*, dan melihat *master data medical*. Proses pembuatan *master data medical* merupakan proses admin menyiapkan data-data yang dibutuhkan oleh sistem pada saat karyawan akan melakukan pengajuan kesehatan. Data - data tersebut berupa daftar saldo pagu karyawan, daftar tarif pengobatan, daftar kategori pengajuan kesehatan, dan daftar dokter spesialis.



Gambar 3.7. Diagram alir utama karyawan

Gambar 3.7 menunjukkan diagram alir utama untuk karyawan. Karyawan memiliki dua buah proses utama dalam sistem *medical reimburse* yaitu proses pengajuan kesehatan, melihat *history* pengajuan kesehatan. Proses pengajuan kesehatan merupakan diagram alir yang menjelaskan alur kerja karyawan saat akan melakukan pengajuan kesehatan pada sistem. Proses *history* pengajuan merupakan proses karyawan dapat melihat daftar-daftar pengajuan yang telah diajukan sebelumnya.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



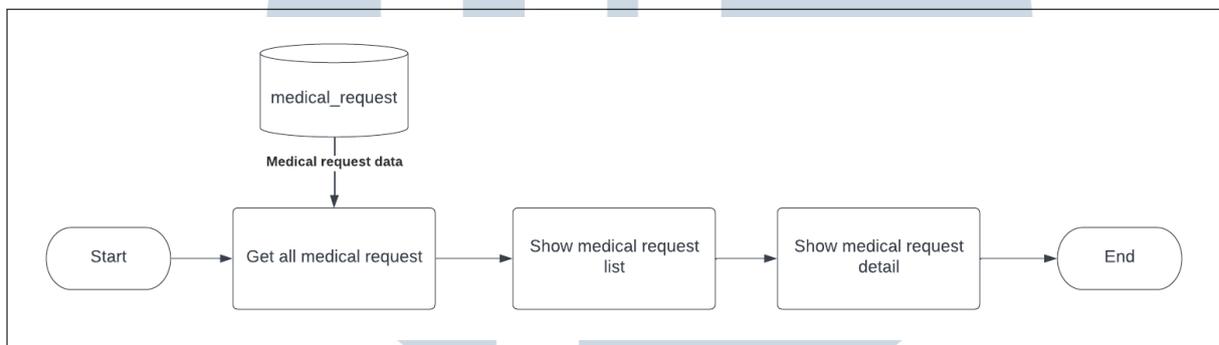
Gambar 3.8. Diagram alir utama HR

Gambar 3.8 menunjukkan diagram alir utama untuk HR. HR memiliki 3 buah proses utama dalam sistem *medical reimburse* yaitu melakukan *approve* pada pengajuan kesehatan karyawan, melihat *history* pengajuan kesehatan karyawan berdasarkan PA admin HR, dan membuat bukti kas / lampiran. Proses *approve* pengajuan kesehatan merupakan proses HR melakukan validasi terhadap data pengajuan dengan bukti kwitansi yang tertera pada pengajuan kesehatan karyawan. Proses *history* pengajuan kesehatan karyawan merupakan proses HR melihat seluruh daftar pengajuan kesehatan karyawan berdasarkan jabatan pers

admin yang dimiliki oleh HR. Proses pembuatan bukti kas dan lampiran merupakan tahap dimana pengajuan - pengajuan karyawan yang sudah disetujui oleh HR akan dilanjutkan ke tahap pergantian biaya dengan menunjukkan bukti keuangan kepada kasir.

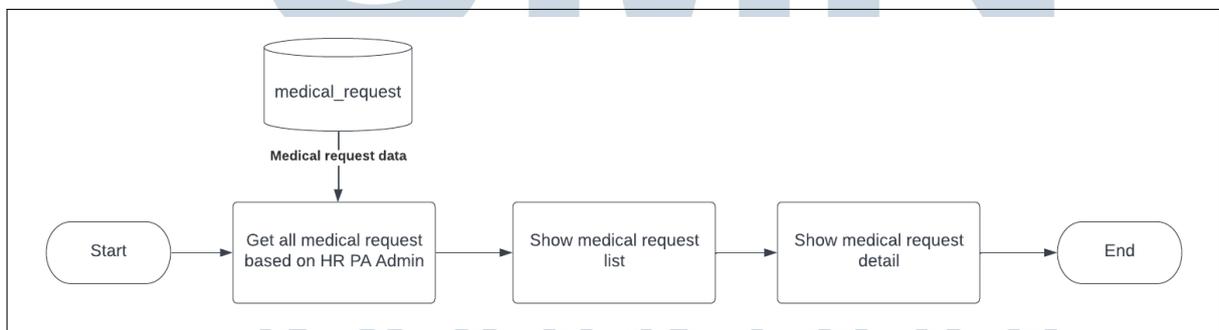
## B Diagram Alir History Pengajuan Kesehatan

Pada bagian ini terdapat tiga buah diagram alir yang menunjukkan proses melihat daftar pengajuan kesehatan yang telah diajukan yang dipisahkan berdasarkan aktor yang melihat *history* tersebut. Ketiga diagram alir tersebut antara lain diagram alir *history* pengajuan admin, diagram alir *history* pengajuan HR, dan diagram alir *history* pengajuan karyawan.



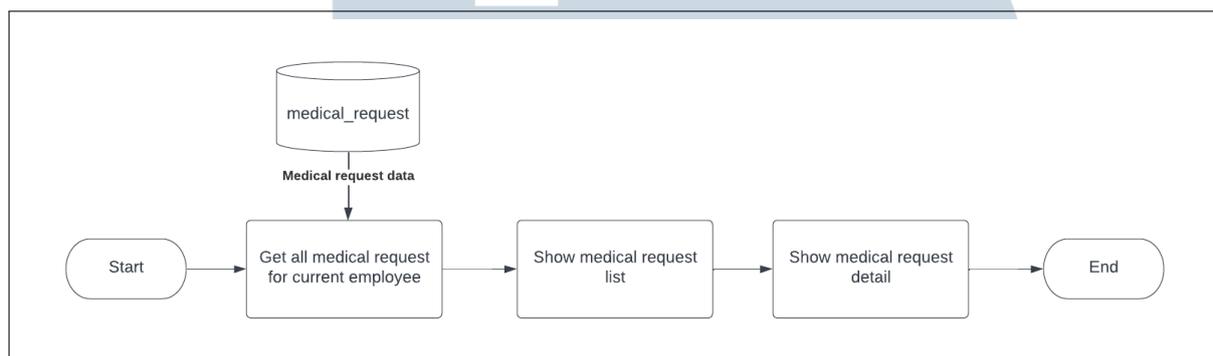
Gambar 3.9. Diagram alir *all medical request*

Gambar 3.9 menunjukkan alur proses saat admin melihat *history* pengajuan kesehatan yang dilakukan karyawan. Pertama saat admin membuka sistem *medical reimburse* terdapat menu All Medical Request. Menu tersebut merupakan menu bagi admin untuk melihat seluruh pengajuan kesehatan yang dilakukan karyawan. Sistem akan mengambil seluruh data pengajuan karyawan yang telah diajukan dari tabel *medical\_request* dan kemudian akan ditampilkan pada menu dalam bentuk *list*. Admin juga dapat membuka detail pengajuan kesehatan karyawan dengan melakukan klik pada salah satu pengajuan yang ingin dilihat.



Gambar 3.10. Diagram alir *employee medical request*

Gambar 3.10 menggambarkan alur proses HR saat akan melihat *history* pengajuan kesehatan karyawan yang di bawah oleh HR berdasarkan jabatan PA admin yang dimiliki oleh HR. Pertama saat HR membuka sistem akan terdapat menu Request To Approve. Menu tersebut merupakan menu untuk HR melihat seluruh pengajuan karyawan yang di bawah oleh HR. Sistem akan mengambil seluruh data pengajuan karyawan yang memiliki pers admin yang sama dengan jabatan pers admin yang dimiliki oleh HR dari tabel *medical\_request*. Selanjutnya sistem akan menampilkan seluruh daftar pengajuan tersebut dalam bentuk *list*. HR juga dapat membuka detail pengajuan kesehatan karyawan dengan melakukan klik pada salah satu pengajuan yang ingin dilihat.

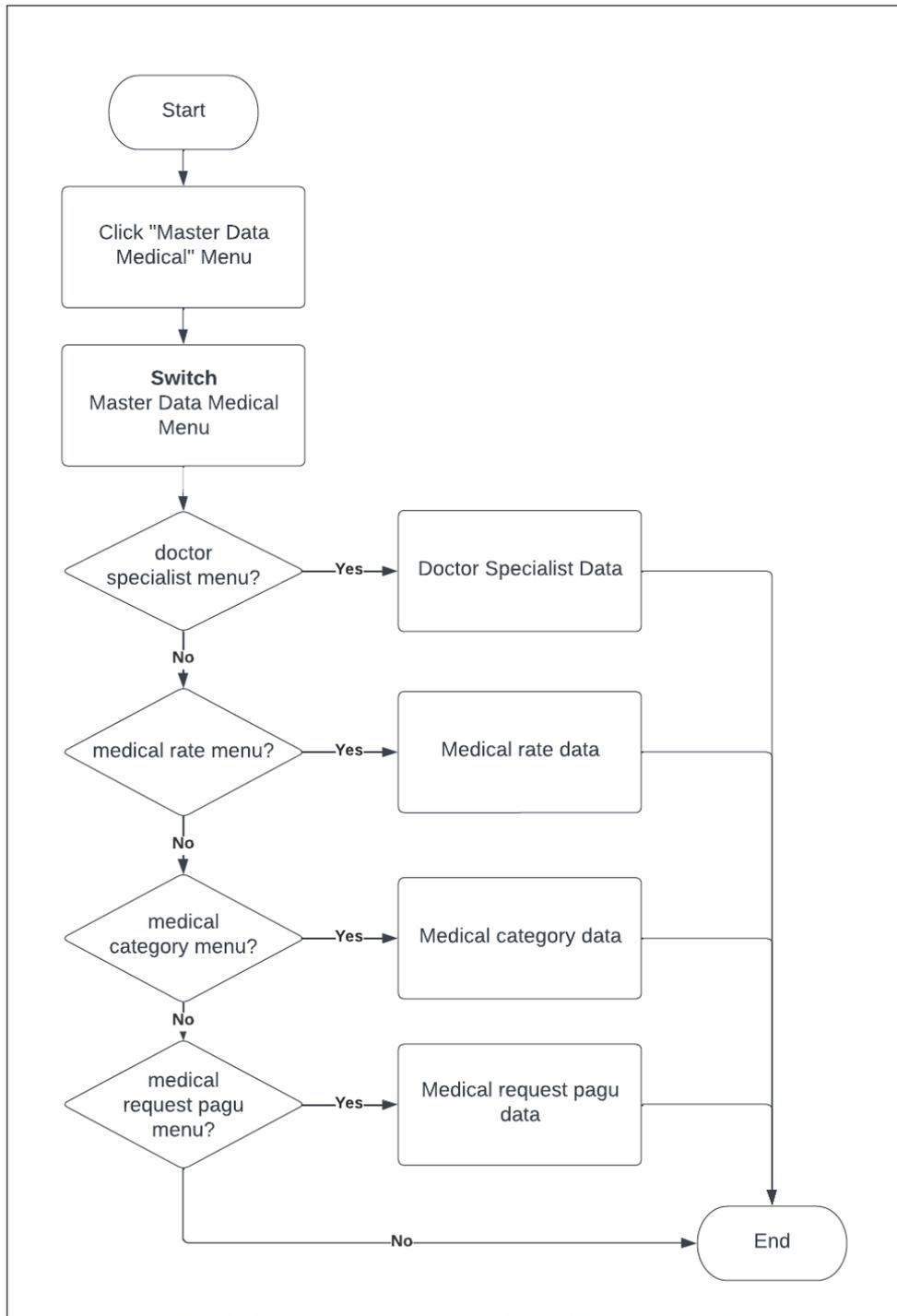


Gambar 3.11. Diagram alir *medical request history*

Gambar 3.11 menunjukkan alur proses karyawan saat akan melihat *history* pengajuan kesehatan yang telah diajukan kepada HR. Saat karyawan membuka sistem terdapat menu Medical Request, sistem akan mengambil seluruh data pengajuan kesehatan karyawan yang sedang masuk ke dalam sistem. Sistem akan melakukan pengecekan terhadap *employee\_id* karyawan yang masuk ke dalam sistem yang sama dengan *employee\_id* pada pengajuan karyawan. Sistem akan menampilkan seluruh data pengajuan dalam bentuk *list*. Karyawan juga dapat membuka detail pengajuan kesehatan dengan melakukan klik pada salah satu pengajuan yang ingin dilihat.

### C Diagram Alir Master Data Medical

Pada bagian ini dijelaskan alur kerja admin dalam melihat *master data medical* yang digunakan sebagai data-data yang dibutuhkan dalam proses *medical reimburse*.



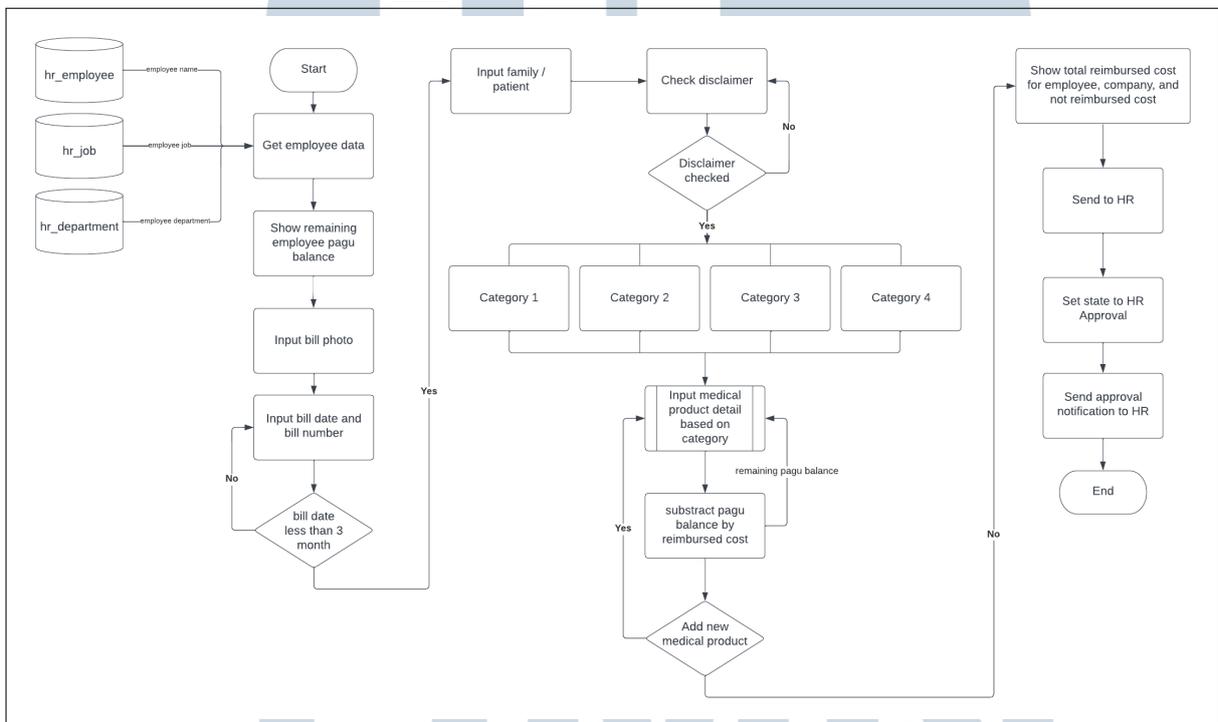
Gambar 3.12. Diagram alir *see medical master data*

Gambar 3.12 menggambarkan proses admin untuk melihat informasi *master data medical* yang dibutuhkan untuk keperluan pengajuan kesehatan. Terdapat empat master data yang dapat dilihat oleh admin yaitu saldo pagu karyawan, kategori pengajuan kesehatan, tarif pengobatan, daftar dokter spesialis. Saldo pagu karyawan merupakan daftar saldo pagu

yang dikelompokkan berdasarkan golongan dan manajemen grup. Data kategori pengajuan kesehatan merupakan daftar kategori yang dapat diajukan untuk digantikan pergantian biaya, terdapat empat kategori pengajuan. Data tarif pengobatan merupakan daftar tarif pergantian biaya seluruh produk pengobatan yang dapat diajukan. Data dokter spesialis merupakan daftar dokter spesialis yang dapat diajukan untuk digantikan pergantian biaya.

#### D Diagram Alir Pengajuan Kesehatan

Pada bagian ini dijelaskan diagram alir proses *medical reimburse*. Diagram alir ini menjelaskan tahapan karyawan saat melakukan pengajuan kesehatan kepada sistem *medical reimburse* yang dibangun.

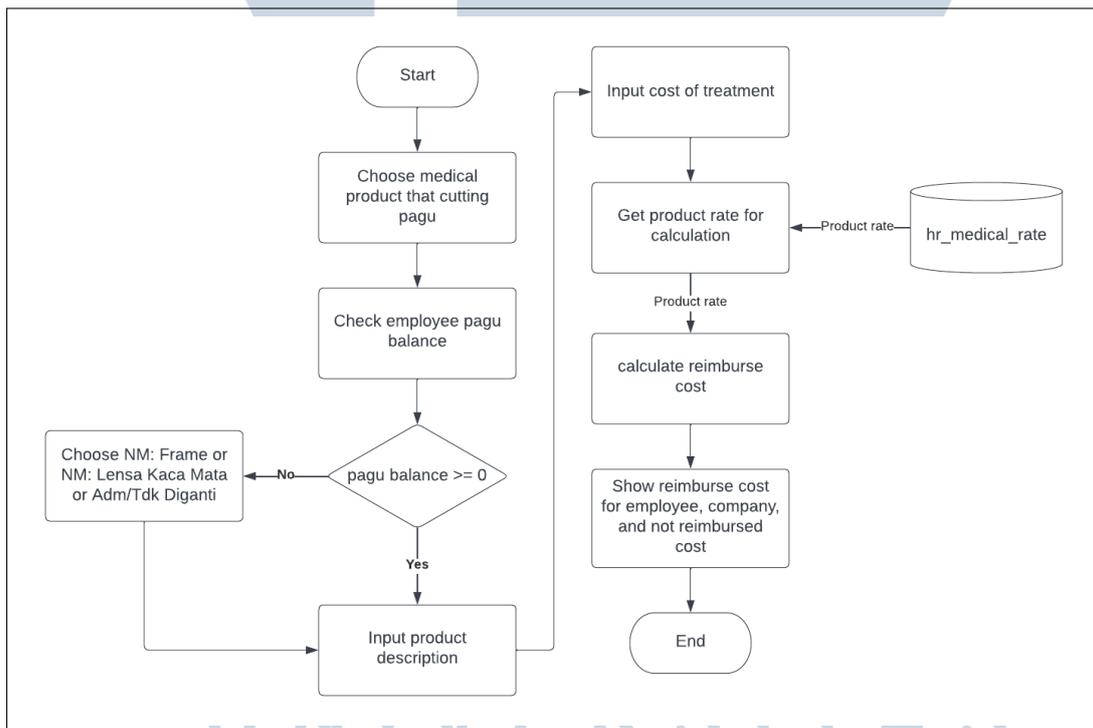


Gambar 3.13. Diagram alir *medical request*

Gambar 3.13 merupakan diagram alir proses pengajuan kesehatan yang digunakan untuk menjelaskan alur proses pengajuan pergantian biaya kesehatan yang dilakukan karyawan. Pertama saat *form* pengajuan kesehatan dibuka, sistem akan mengambil data karyawan yang akan melakukan pengajuan berupa nama, jabatan, dan departemen karyawan. Data tersebut akan ditampilkan pada *form* pengajuan kesehatan. Sistem juga menampilkan sisa saldo pagu kesehatan yang dimiliki karyawan. Selanjutnya, karyawan diminta untuk memasukkan bukti kwitansi pengobatan dalam bentuk foto. Setelah bukti foto kwitansi berhasil diunggah, karyawan diminta mengisi nomor dan tanggal kwitansi sesuai dengan bukti

yang ada. Sistem kemudian akan melakukan validasi pada tanggal kwitansi, batas tanggal kwitansi yang boleh diajukan paling lambat 3 bulan sebelum dari tanggal karyawan melakukan pengajuan kesehatan. Bila melebihi 3 bulan maka tanggal akan ditolak dan karyawan diminta mengisi tanggal kwitansi ulang.

Setelah mengisi data kwitansi, karyawan mengisi nama pasien / keluarga yang akan diajukan pergantian biaya pengobatannya, namun bila tidak diisi maka pengajuan yang dilakukan untuk karyawan itu sendiri. Selanjutnya karyawan harus menyetujui *disclaimer* yang terdapat pada *form* pengajuan, bila karyawan tidak menyetujui *disclaimer* yang ada maka karyawan tidak dapat mengisi detail pengobatan yang akan diajukan. Bila *disclaimer* sudah disetujui, karyawan dapat memilih kategori pengobatan yang sesuai dengan bukti kwitansi pengobatan yang ada. Terdapat 4 kategori pengobatan dimana kategori yang dipilih akan menjadi acuan untuk menampilkan produk pengobatan yang ada pada saat karyawan akan mengisi detail produk pengobatan yang akan mereka ajukan. Karyawan kemudian akan mengisi detail produk pengobatan yang akan diajukan yang digambarkan dalam modul proses pada Gambar 3.13 yang akan dijabarkan pada Gambar 3.14.



Gambar 3.14. Diagram alir *detail medical request*

Gambar 3.14 merupakan diagram alir proses *input* detail produk pengobatan yang diajukan karyawan beserta proses kalkulasi pergantian biaya. Pertama karyawan akan memilih produk pengobatan yang akan diajukan untuk dilakukan pergantian biaya. Produk - produk

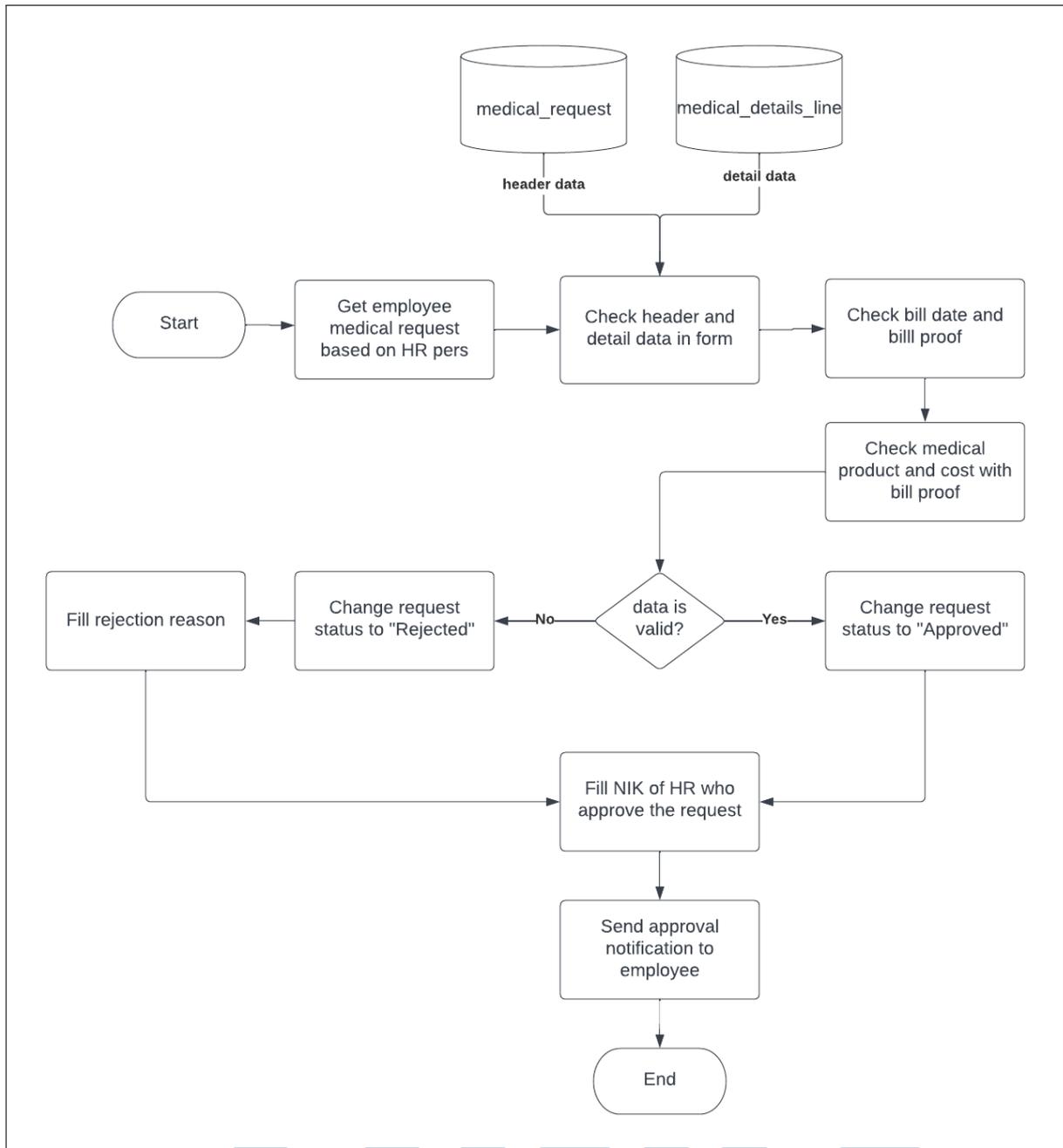
pengobatan yang ditampilkan merupakan produk pengobatan yang berada pada kategori yang telah dipilih karyawan sebelumnya. Semua produk yang ditampilkan merupakan produk yang memotong saldo pagu karyawan, produk yang tidak memotong pagu hanya tiga produk yaitu NM: Lensa Kacamata, NM: Frame, dan Adm/Tidak diganti yang berada pada kategori dua. Setelah memilih produk yang akan diajukan, sistem akan melakukan cek pada sisa saldo pagu karyawan, bila saldo pagu karyawan kosong maka karyawan hanya dapat melakukan pengajuan pada produk pengobatan yang tidak memotong pagu. Bila masih memiliki saldo maka produk dapat dipilih dan kemudian karyawan mengisi deskripsi dari produk pengobatan yang diajukan.

Selanjutnya karyawan akan mengisikan nominal biaya pengobatan pada produk yang dipilih. Setelah data detail produk pengobatan diisi, sistem akan melakukan proses kalkulasi untuk menentukan total pergantian biaya kepada produk yang dipilih. Proses kalkulasi akan mengambil tarif pergantian biaya pada produk terkait dari tabel *hr\_medical\_rate*, selanjutnya biaya pengobatan yang diisi akan dikalkulasikan dengan tarif biaya yang telah didapatkan. Kalkulasi yang dilakukan akan menghasilkan biaya pergantian pada produk yang ditanggung perusahaan, karyawan, dan biaya yang tidak digantikan. Sistem akan menampilkan ketiga biaya tersebut kepada karyawan sehingga karyawan dapat melihat biaya pergantian yang didapatkan.

Kembali pada diagram alir Gambar 3.13, setelah karyawan mengisi produk-produk pengobatan yang diajukan, sistem akan menampilkan total pergantian biaya untuk keseluruhan produk pengobatan yang diajukan. Total pergantian biaya yang ditampilkan merupakan total biaya pergantian yang diganti perusahaan, total biaya yang ditanggung karyawan, dan total biaya yang tidak diganti. Karyawan kemudian menekan tombol *Send To HR* untuk mengajukan pergantian biaya pengobatan kepada HR karyawan yang bersangkutan. *State* pengajuan akan berubah menjadi *HR Approval* dan sistem akan mengirim notifikasi kepada HR untuk meminta *approval* pada pengajuan yang dibuat karyawan. Karyawan selanjutnya menunggu HR karyawan yang bersangkutan untuk melakukan *approve* pada pengajuan yang dilakukan.

## **E Diagram Alir Persetujuan Pengajuan Kesehatan**

Pada bagian ini dijelaskan diagram alir *approval medical reimburse* karyawan oleh HR. Diagram alir ini menjelaskan tahapan yang dilakukan HR saat melakukan persetujuan atau penolakan kepada pengajuan kesehatan yang diajukan karyawan kepada perusahaan.



Gambar 3.15. Diagram alir *approve medical request*

Gambar 3.15 menunjukkan proses kerja *approval* dan *reject* yang dilakukan HR pada pengajuan kesehatan yang dilakukan karyawan. Pertama HR akan membuka daftar pengajuan - pengajuan yang dilakukan karyawan berdasarkan pers admin karyawan yang dimiliki oleh HR. Saat membuka *form* pengajuan kesehatan karyawan, sistem akan mengambil data *header* dan detail pengajuan yang dilakukan karyawan. Data *header* dan detail diambil dari tabel *medical\_request* untuk data *header* dan *medical\_details\_line* untuk data

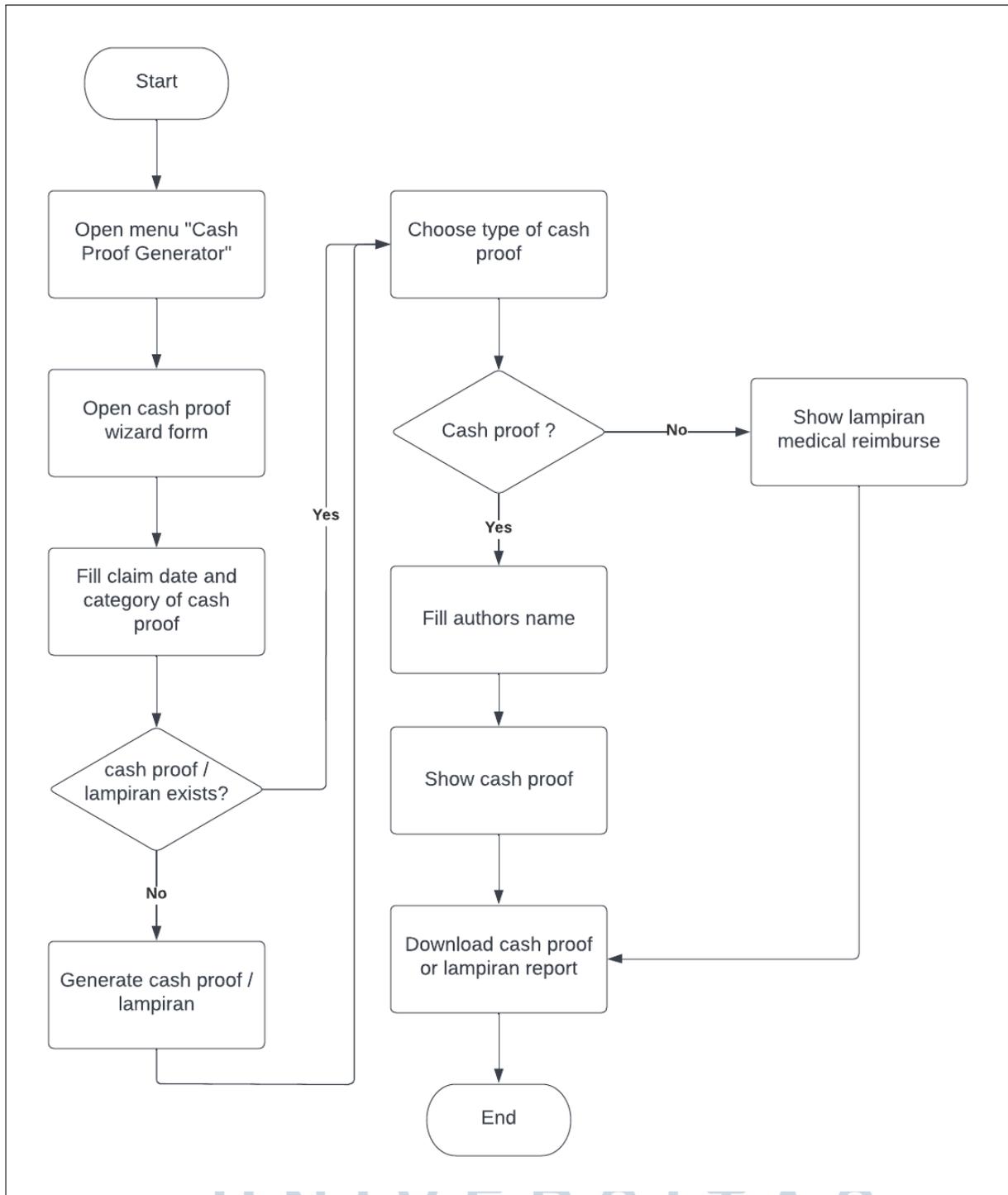
detail pengobatan. Selanjutnya HR akan melakukan validasi pada bukti dan tanggal kwitansi untuk memastikan keaslian bukti yang dimasukan karyawan. HR kemudian melakukan pengecekan pada detail produk pengobatan yang diajukan oleh karyawan dengan bukti kwitansi. Pengecekan dilakukan dengan melihat produk dan biaya pengobatan yang diajukan karyawan dengan bukti kwitansi, hal ini dilakukan untuk mengecek apakah produk dan biaya pengobatan yang diajukan sesuai dengan nominal yang tercantum pada bukti kwitansi.

Data yang sudah divalidasi oleh HR akan menjadi acuan bagi HR dalam menerima dan menolak pengajuan kesehatan yang dilakukan oleh karyawan. Bila data pengajuan valid, HR akan memberikan persetujuan pada pengajuan dan status pengajuan karyawan menjadi *approved*. Namun bila data pengajuan tidak valid, HR akan mengubah status pengajuan menjadi *rejected* dan memberikan alasan penolakan pengajuan pada *form* pengajuan karyawan, sehingga karyawan dapat mengetahui alasan pengajuan nya ditolak oleh HR. HR juga memasukan NIK (Nomor Induk Karyawan) pada form pengajuan yang disetujui untuk memberikan info kepada karyawan akan siapa HR yang melakukan *approval*. Setelah semua data untuk melakukan persetujuan diisi, sistem akan memberikan notifikasi kepada karyawan bahwa pengajuan kesehatan yang diminta telah dicek oleh HR dan memberikan informasi mengenai status pengajuan yang didapatkan oleh karyawan.

## **F Generate Bukti Kas dan Lampiran**

Pada bagian ini dijelaskan diagram alir proses *generate* bukti kas dan lampiran untuk sistem *medical reimburse*. Diagram alir ini menjelaskan tahapan HR ketika membuat bukti kas untuk sarana pergantian biaya pengobatan karyawan dan lampiran untuk detail produk yang diajukan karyawan.





Gambar 3.16. Diagram alir *generate cash proof*

Bukti kas merupakan bukti total penggantian biaya pengobatan karyawan yang telah disetujui oleh HR untuk meminta penggantian biaya kepada kasir. Lampiran merupakan detail pengajuan pengobatan karyawan yang terdiri dari nama, NIK, produk pengobatan, biaya, dan tanggal kwitansi. Pada Gambar 3.16 menunjukkan alur proses yang dilakukan HR saat

membuat bukti kas dan lampiran dari pengajuan - pengajuan yang telah disetujui. Pada saat HR membuka menu *Medical Reimburse*, akan terdapat menu *Generate Cash Proof* untuk membuat bukti kas dan lampiran. Saat HR menekan menu tersebut sistem akan membuka sebuah *form*, setelah *form* terbuka HR diminta mengisi tanggal dan kategori bukti kas. Sistem kemudian akan melakukan pengecekan pada tanggal bukti kas yang diisi oleh HR, bila pada tanggal tersebut bukti kas dan lampiran sudah pernah dibuat maka sistem tidak akan membuat bukti kas baru melainkan mengambil data bukti kas yang telah dibuat pada tanggal tersebut. Namun bila bukti kas belum pernah dibuat pada tanggal tersebut maka sistem akan membuat bukti kas dan lampiran untuk keseluruhan pengajuan kesehatan karyawan yang belum pernah dibuat bukti kas sampai pada tanggal bukti kas yang diisi oleh HR.

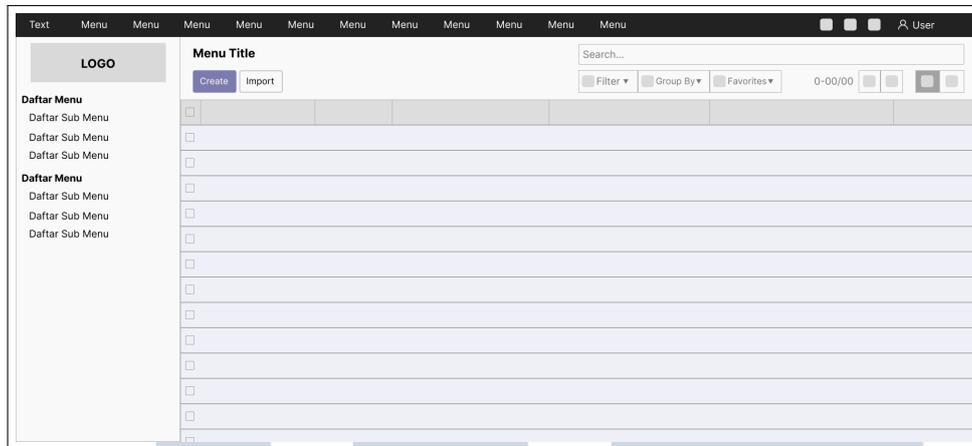
Setelah bukti kas berhasil dibuat, HR dapat memilih jenis bukti kas yang ingin dilihat, terdapat dua tipe yaitu bukti kas dan lampiran. Bila memilih bukti kas maka HR akan diminta mengisi data penanggung jawab dari bukti kas yang dibuat berupa divisi, direktur keuangan, direktur grup, dan nama HR yang membuat bukti kas. Data yang diisi akan ditampilkan dalam bukti kas sebagai bukti kepada kasir. Namun bila HR memilih lampiran maka sistem akan langsung menampilkan lampiran yang sudah dibuat yang berisikan detail pengobatan dari masing-masing pengajuan karyawan yang telah disetujui dan masuk ke dalam bukti kas. Selanjutnya HR dapat melakukan *download* pada bukti kas atau lampiran yang ada untuk dijadikan bukti *report* dan pergantian biaya kepada kasir.

### 3.3 Desain Antar Muka

Pada proses perancangan sistem *medical reimburse*, dibuatkan desain tampilan antar muka atau sering disebut sebagai *mock up* untuk memberikan gambaran tampilan dari sistem yang akan dibangun. Desain antar muka yang ada dibuat menggunakan aplikasi *Figma* dan desain yang dibuat berupa *wireframe* dari tampilan yang akan dibangun. Desain tersebut dibuat untuk memudahkan saat proses pembangunan sistem *medical reimburse* yang akan digunakan oleh karyawan. Berikut merupakan desain antar muka yang telah dibangun.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

### 3.3.1 Desain Tampilan Menu Medical Reimbursement

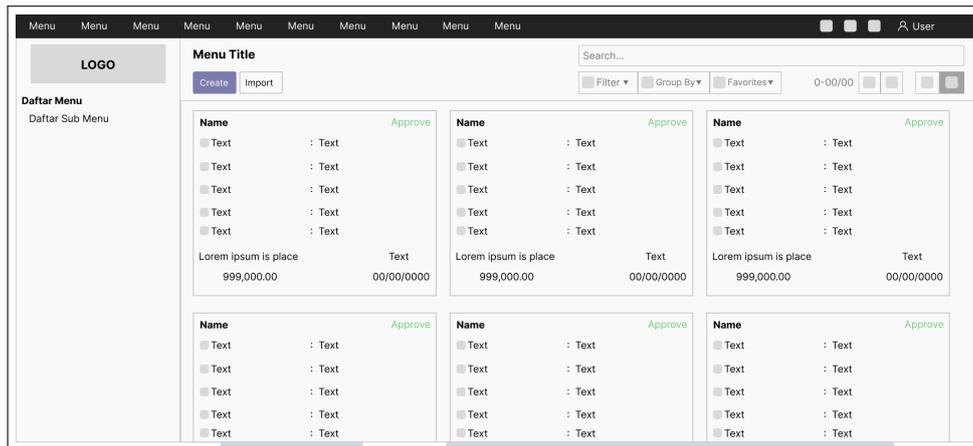


Gambar 3.17. Desain tampilan menu *medical reimburse*

Gambar 3.17 menggambarkan desain tampilan menu *medical reimburse* yang akan dibangun pada sistem. Desain tampilan tersebut diimplementasikan pada keseluruhan *sub-menu* Medical Reimbursement pada sistem yang dibangun. Pada menu ini terdapat dua buah bagian yang terlihat seperti gambar di atas yaitu *side-bar* dan bagian utama dari menu. Pada bagian *side-bar* akan berisikan menu-menu yang terdapat pada menu utama Medical Reimbursement. Menu - menu tersebut akan berbeda tergantung dari *role* pengguna yang masuk kedalam sistem.

Bagian utama dari menu merupakan isi dari menu yang ada pada *side-bar*. Pada bagian *header* dari tampilan terdapat judul dari menu, lalu dibawah judul terdapat dua buah tombol *Create* dan *Import*. Pada bagian kanan *header* terdapat sebuah kotak pencarian untuk melakukan *searching* pada data yang tersedia dalam menu. Selanjutnya di bawah kotak pencarian terdapat tombol untuk melakukan *filter*, *grouping*, dan *favorites* dan disebelah kanal ketig tombol tersebut terdapat tombol untuk mengganti bentuk tampilan pada menu.

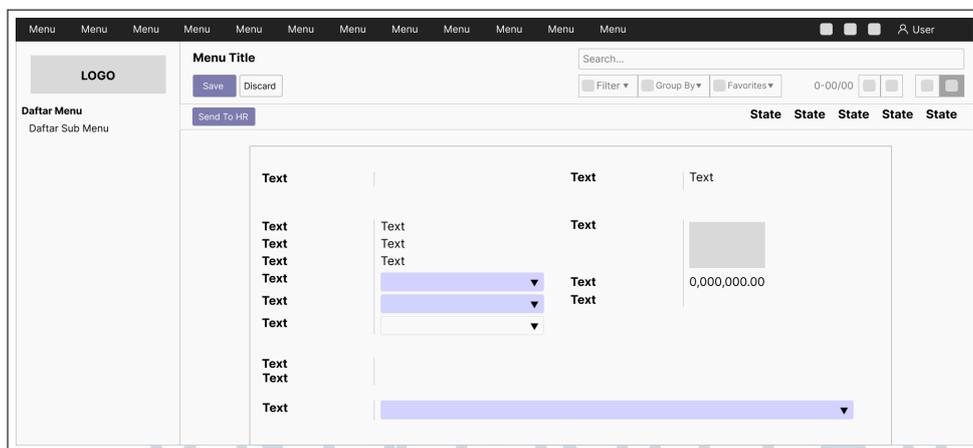
Bagian isi dari menu merupakan data-data yang dimiliki dan ditampilkan pada menu. Pada desain yang dibuat bentuk tampilan untuk menampilkan data adalah dalam bentuk *list* atau tabel. Selain bentuk *list* terdapat desain bentuk *kanban* atau kartu yang dibuat, namun desain tersebut hanya diimplementasikan pada menu Medical Request untuk menampilkan data pengajuan pergantian biaya kesehatan karyawan dalam bentuk *list* dan *kanban*. Gambar 3.18 memperlihatkan desain yang dimaksud.



Gambar 3.18. Desain tampilan menu *medical request* dalam bentuk kanban

Pada Gambar di atas dalam bagian isi terdapat bentuk desain kartu yang dibuat untuk menampilkan data pengajuan pergantian biaya kesehatan yang sudah dibuat dalam bentuk kartu. Data yang ditampilkan dalam kartu merupakan status pengajuan, nama karyawan yang berada pada bagian atas kartu. Selanjutnya terdapat total pergantian biaya, nama pasien, dan tanggal kwitansi yang ditampilkan pada bagian isi dari kartu. Desain *kanban* ini dibangun untuk menampilkan data pengajuan pada mode *mobile-app* sehingga tampilan data akan lebih mudah dilihat.

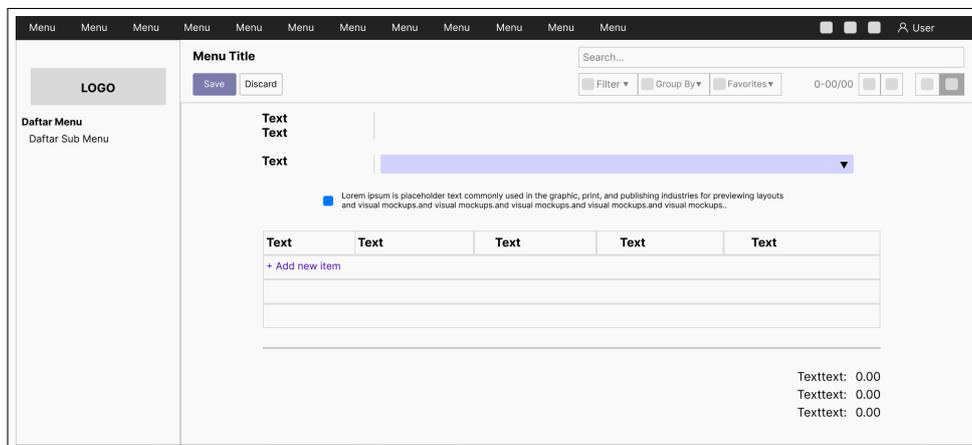
### 3.3.2 Desain Tampilan Form Medical Request



Gambar 3.19. Desain tampilan *form medical request*

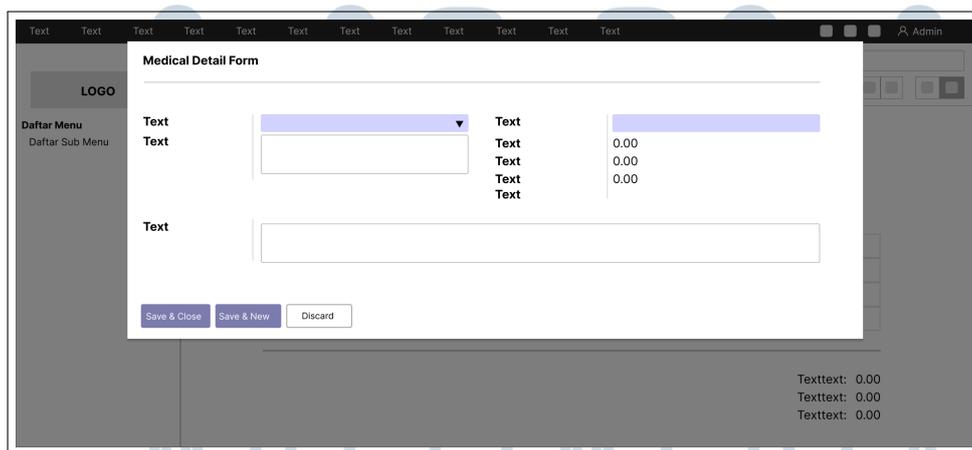
Gambar 3.19 memperlihatkan desain tampilan *form medical request* untuk melakukan pengajuan pergantian biaya kesehatan kepada sistem. Pada bagian *header form* terdapat satu buah tombol *Send To HR* untuk mengirimkan data pengajuan yang telah dibuat kepada

HR melalui sistem. Selanjutnya pada bagian kanan terdapat *state* yang menunjukkan tahap pengajuan *form* yang dibuat. Bagian isi dari *form* berisikan *field-field* yang perlu diisi dan informasi yang diberikan untuk mengajukan pergantian biaya kepada sistem.



Gambar 3.20. Desain tampilan *form medical request*

Masih menjadi bagian dari form, pada Gambar 3.20 memperlihatkan bagian selanjutnya dari desain tampilan *form medical request*. Kotak *dropdown* yang berwarna ungu berisikan jenis kategori pengajuan yang dapat dipilih. Selanjutnya di bawah kategori pengajuan terdapat *disclaimer* yang harus dicentang oleh pengguna. Pada *form* juga terdapat tampilan *list* yang menampilkan data detail produk pengobatan yang akan diajukan serta sebuah tombol untuk menambahkan produk pengobatan yang baru. Bagian paling bawah pada *form* akan menunjukkan total pergantian biaya produk pengobatan yang ditampilkan kepada pengguna.

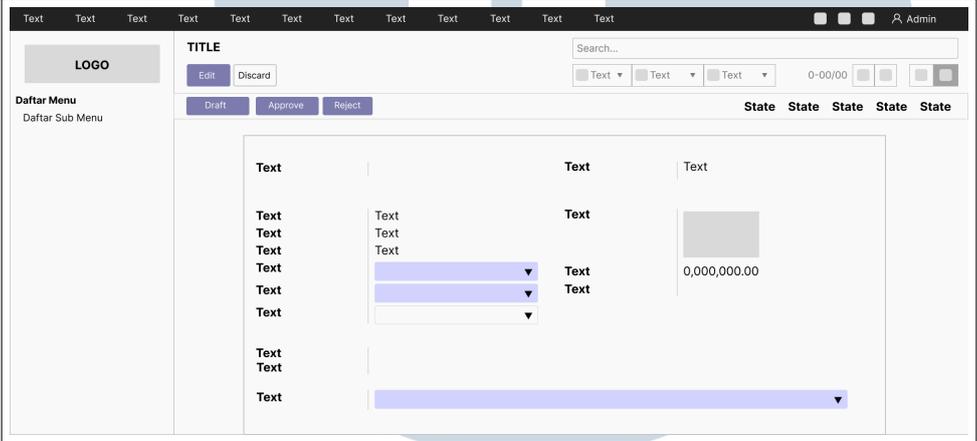


Gambar 3.21. Desain tampilan *form medical request*

Saat tombol *Add new item* ditekan maka sistem akan membuka tampilan *form* dalam bentuk modal seperti pada Gambar 3.21. *Form* tersebut digunakan untuk menambahkan

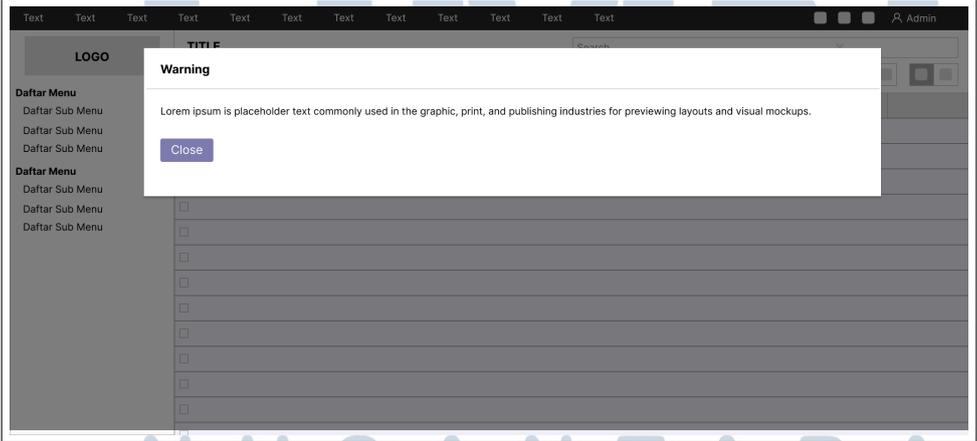
produk pengobatan yang akan diajukan. Bagian isi dari form akan menampilkan *field-field* yang dibutuhkan untuk mengisi data detail produk pengobatan. Bagian *footer* dari form terdapat tiga buah tombol yaitu Save & Close, Save & New, dan Discard.

Selain desain tampilan *form medical request*, juga dibentuk desain tampilan *form* untuk HR saat akan melakukan *approve* atau *reject* pada *form* pengajuan karyawan. Desain tampilan yang dimiliki sama seperti pada desain tampilan *form* karyawan, yang membedakan hanya button yang ditampilkan pada bagian *header form*. Pada *form approval* HR, terdapat tiga buah tombol yaitu Draft, Approve, dan Reject. Ketiga tombol tersebut digunakan untuk keperluan *approval* kepada data pengajuan karyawan. Gambar 3.22 memperlihatkan desain tampilan yang dibuat.



Gambar 3.22. Desain tampilan *form medical request*

### 3.3.3 Desain Error Message

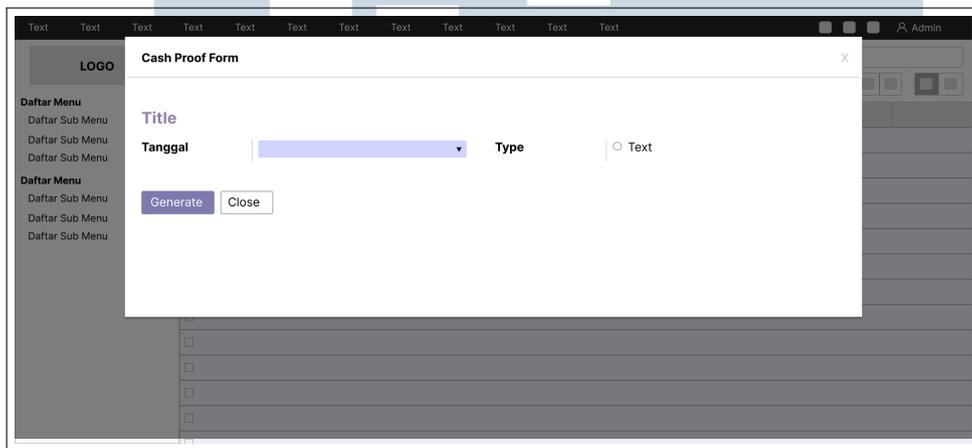


Gambar 3.23. Desain tampilan *error message*

Gambar 3.23 menunjukkan desain tampilan *error message* pada sistem. *Error message* akan muncul bila terjadi kesalahan yang dilakukan pengguna pada sistem yang dibangun. Bagian *header* akan menampilkan tulisan *warning* sebagai tanda peringatan. Bagian isi akan menampilkan isi pesan yang disampaikan sistem kepada pengguna. Bagian *footer* terdapat sebuah tombol *Close* untuk menutup modal yang ditampilkan.

### 3.3.4 Desain Tampilan Bukti Kas dan Lampiran

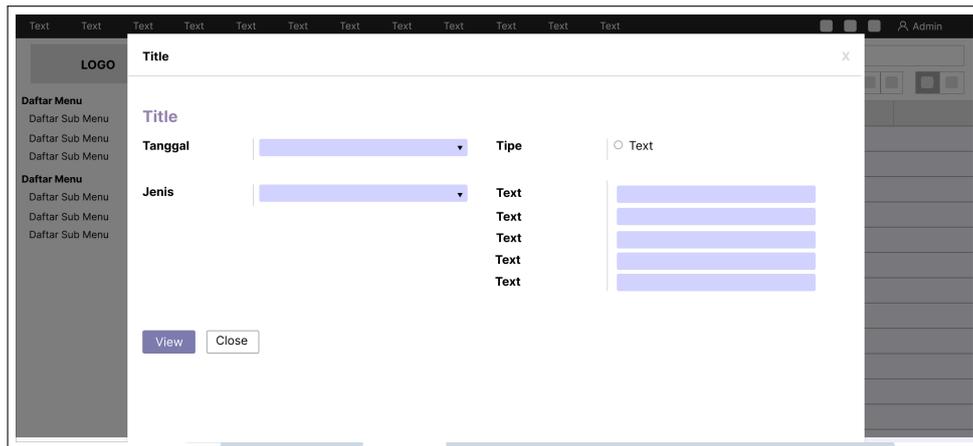
Pada tampilan bukti kas dan lampiran terdapat dua buah desain yang dibuat, pertama merupakan desain tampilan *form* untuk melakukan *generate* bukti kas. Kedua merupakan desain *report* dalam format *.mrt* untuk bukti kas dan lampiran yang dibentuk menggunakan aplikasi *stimulsoft*. Berikut merupakan desain tampilan yang dibuat.



Gambar 3.24. Desain tampilan *form* bukti kas

Gambar 3.24 memperlihatkan desain tampilan *form* untuk melakukan *generate* data bukti kas dan lampiran yang baru. Pada desain tampilan *form* ini terdapat dua buah *field* yang ditempatkan pada form untuk diisi oleh HR saat akan membuat data bukti kas baru yaitu tanggal dan tipe bukti kas. Selanjutnya *form* juga memiliki dua buah tombol yaitu *Generate* untuk membuat data bukti kas baru dan *Close* untuk membatalkan dan menutup *form*.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

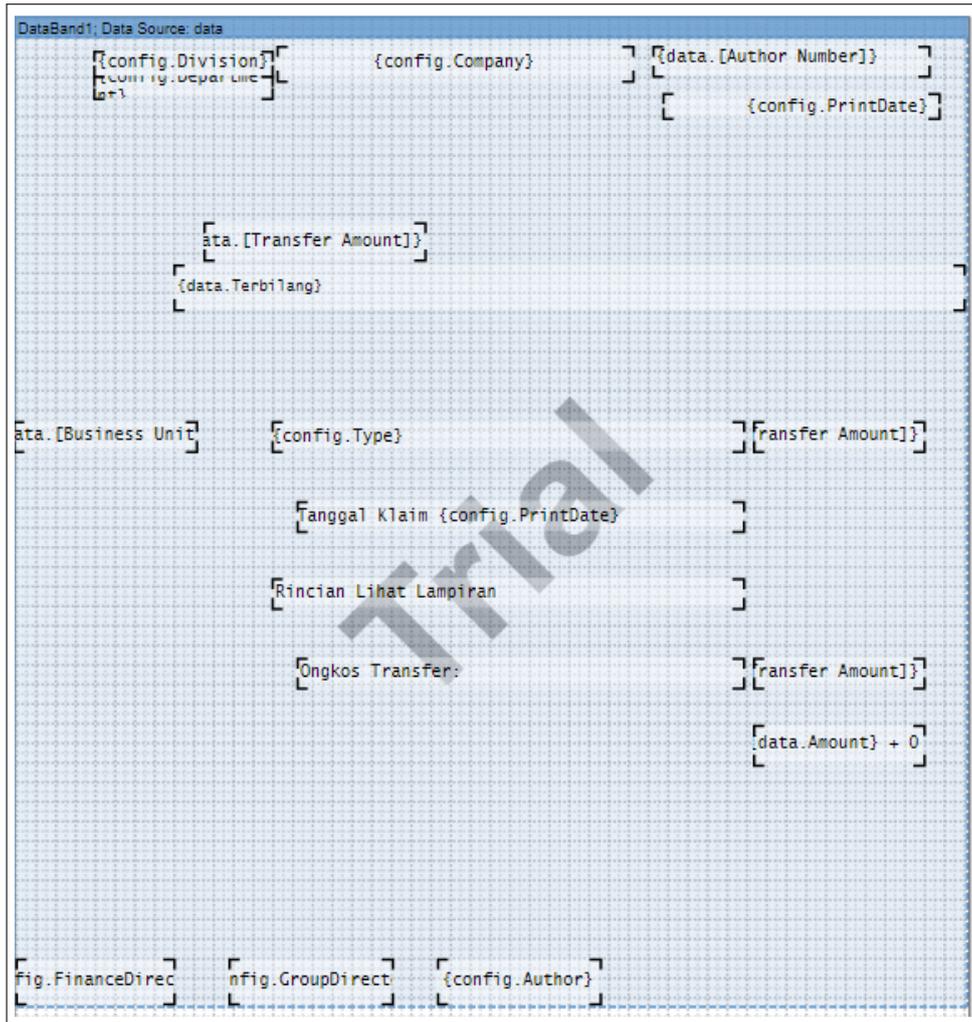


Gambar 3.25. Desain tampilan *form view* bukti kas

Setelah tombol *Generate* ditekan, maka sistem akan menampilkan tampilan form seperti pada Gambar 3.25. *Form* tersebut digunakan untuk menampilkan tampilan bukti kas atau lampiran yang sudah berhasil dibuat. Untuk melihat data yang sudah dibuat HR dapat memilih jenis bukti kas yang akan dilihat, apakah bukti kas atau lampiran. Setelah memilih jenis HR selanjutnya dapat mengisi data-data yang dibutuhkan untuk melihat bukti kas atau lampiran. Untuk melihat hasil dari bukti kas atau lampiran terdapat tombol *View* pada bagian bawah untuk melihat hasil bukti kas yang sudah dibuat.

Gambar selanjutnya menunjukkan desain tampilan *report* dari bukti kas dan lampiran dalam format *.mrt*. Gambar 3.26 merupakan desain *report* yang dibentuk untuk menampilkan data bukti kas dimana data-data yang disusun dalam desain tersebut disesuaikan dengan kerangka kertas biru yang dimiliki oleh perusahaan. Kertas biru merupakan kertas untuk mencetak bukti kas pada perusahaan Kompas Gramedia. Gambar 3.27 merupakan desain *report* yang dibentuk untuk menampilkan data detail lampiran dari pengajuan kesehatan yang telah dilakukan dan kertas yang digunakan merupakan kertas putih dengan ukuran *custom* yang dicetak secara manual.





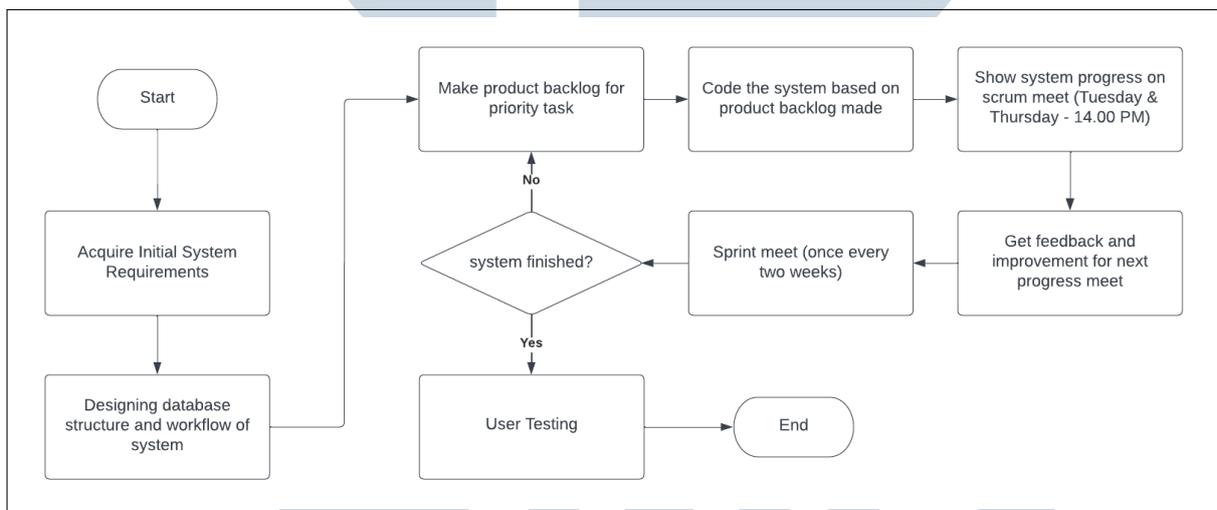
Gambar 3.26. Tampilan desain *report* bukti kas

UMMN  
 UNIVERSITAS  
 MULTIMEDIA  
 NUSANTARA

PageHeaderBand1							
{config.Company} Lampiran Bukti Kas Detil							
Tanggal: {config.PrintDate} Tipe : {config.Type}							
GroupHeaderBand2: Condition: {data.Business_Unit}							
GroupHeaderBand1: Condition: {data.Department}							
Headerdata							
NIK	Nama	Kategori	Nama Pasien	Tgl. Kwitansi	Biaya Kwitansi	Biaya Pergantian	
Datadata: Data Source: data							
{data.NIK}	{data>Nama}	{data.Bplan}	{data.[Nama Pasien]}	{data.[Tgl Kwitansi]}	{data.[Biaya Kwitansi]}	{data.[Biaya Pergantian]}	
GroupFooterBand1							
* Departemen {data.Department}					Biaya Kwitansi}}	Biaya Pergantian}}	
GroupFooterBand2							
** Unit Usaha {data.[Business Unit]}					Biaya Kwitansi}}	Biaya Pergantian}}	

Gambar 3.27. Tampilan desain report lampiran

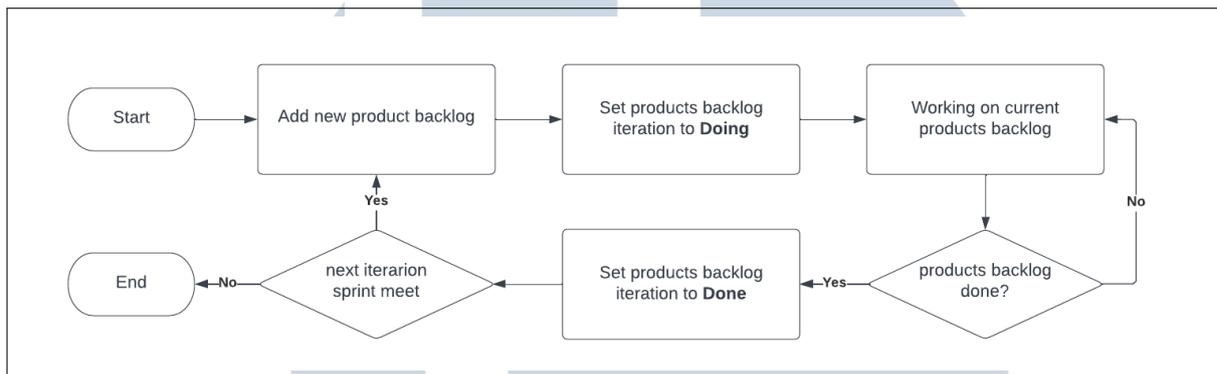
### 3.4 Pembangunan Sistem



Gambar 3.28. Diagram alir proses pembangunan dengan metode *scrum*

Proses pembangunan sistem dilakukan menggunakan ERP (Odo) dengan bahasa pemrograman Python untuk membangun kode, XML untuk membuat tampilan dari sistem, dan PostgreSQL sebagai sistem basis data untuk menyimpan seluruh data sistem. Gambar 3.28 menggambarkan alur proses pembangunan sistem *medical reimburse* menggunakan metode *scrum*. Tahap pembangunan diawali dengan proses penerimaan *requirements* awal sistem yang diberikan oleh Tim HRIS. *Requirements* tersebut berupa fitur-fitur utama yang dimiliki

oleh sistem dan perencanaan alur kerja dari sistem yang akan dibangun. *Requirements* tersebut akan terus bertambah dan dimodifikasi sesuai dengan kebutuhan pengguna pada setiap iterasi *sprint* yang dilakukan. Setelah menerima kebutuhan awal sistem, selanjutnya dilakukan proses rancangan terhadap sistem. Rancangan yang dibangun yaitu struktur basis data dengan membentuk ERD (*Entity Relationship Diagram*), DFD (*Data Flow Diagram*) untuk menggambarkan alur informasi dari proses - proses yang terdapat pada sistem, dan Diagram alir untuk menggambarkan alur kerja dari sistem yang dibangun.



Gambar 3.29. Diagram alir proses pembuatan *product backlog*

Setelah mendapatkan *requirements* yang dibutuhkan, kebutuhan sistem yang akan dikerjakan akan dibagi ke dalam bentuk *product backlog*. *Product backlog* yang merupakan prioritas pekerjaan yang akan dikerjakan selama satu proses iterasi *sprint* berlangsung. Gambar 3.29 menunjukkan proses pembuatan *product backlog* yang menjadi prioritas pekerjaan pada satu iterasi *sprint*. Pertama proses dimulai dengan menambahkan *product backlog* baru dengan mengisikan judul, deskripsi, iterasi *sprint* dan orang yang bertanggung jawab untuk mengerjakan *product backlog*. Selanjutnya *product backlog* yang telah dibuat akan diatur *state* menjadi *Doing*, hal tersebut menandakan bahwa *backlog* ini sedang dikerjakan. Selanjutnya *products backlog* yang telah dibuat akan dikerjakan selama proses *sprint* berlangsung sampai pekerjaan tersebut selesai. Setelah *product backlog* selesai dikerjakan, maka *state backlog* akan diatur menjadi *Done*, hal tersebut menandakan bahwa *product backlog* selesai dikerjakan. *Products backlog* yang ada dibuat menggunakan aplikasi *microsoft azure*.

Saat proses pengerjaan *products backlog* berlangsung akan diadakan dua kali *scrum meet* untuk menunjukkan perkembangan pekerjaan yang sedang dilakukan pada sistem yang dibangun. *Scrum meet* dilakukan setiap hari Selasa dan Kamis pada pukul 14.00 WIB. Pada *scrum meet* tim HRIS akan memberikan masukan dan saran perbaikan kepada perkembangan sistem yang sedang berlangsung, dan *feedback* tersebut akan menjadi saran perbaikan untuk *scrum meet* selanjutnya. Masa iterasi untuk satu buah *sprint* berlangsung selama dua minggu,

dimana selama proses tersebut akan diadakan empat kali *scrum meet* untuk menunjukkan perkembangan sistem kepada tim HRIS. *Sprint meet* akan diadakan pada hari Jumat pada minggu kedua dari iterasi, pada proses ini seluruh divisi akan memberikan perkembangan dari *product backlogs* yang sudah dibuat dan dikerjakan selama masa *sprint* berlangsung. *Sprint master* lalu memberikan *feedback* terhadap pekerjaan yang dilakukan dan memberikan keputusan apakah *backlog* yang dikerjakan sudah sesuai atau masih harus dilanjutkan pada iterasi selanjutnya.

Siklus *sprint* akan terus berulang sampai sistem yang dibangun selesai dikerjakan dan memenuhi kriteria kebutuhan yang diinginkan oleh pengguna. Selanjutnya sistem yang telah selesai dibangun akan memasuki tahap pengujian kepada pengguna sistem. Pada penelitian ini, sistem *medical reimburse* akan diujikan kepada karyawan GoRP (*Group Retail and Publishing*) Kompas Gramedia. Karyawan yang menggunakan sistem akan memberikan *feedback* dengan mengisi survei yang diberikan kepada karyawan untuk mengukur tingkat EUCS (*End User Computing Satisfaction*) dari sistem yang telah berhasil dibangun.

