

## BAB 3 PELAKSANAAN KERJA MAGANG

### 3.1 Kedudukan dan Organisasi

Kedudukan selama mengikuti kegiatan kerja magang di PT. Bank Rakyat Indonesia (Persero) Tbk adalah sebagai *frontend engineer intern* pada tim UMi Corner di bawah naungan divisi DDB. Pada tim UMi Corner yang menjadi Project Manager adalah Bapak Dian Charlo, dan sebagai *Team Lead Engineer* adalah Bapak Muhammad Hilman yang bertugas untuk melakukan pembagian pekerjaan kepada setiap anggota tim bagian *developer*. Selama pelaksanaan pengerjaan tugas kerja magang didampingi oleh Bapak Pradipta selaku *frontend engineer* pada tim UMi Corner. Koordinasi juga selalu dilakukan dengan *frontend engineer* tim untuk memastikan pengerjaan yang dilakukan sudah benar dan meminta arahan jika ada yang masih belum jelas.

Pada tim UMi Corner terdapat beberapa posisi dengan *job desk* yang berbeda dan saling berkaitan dalam proses pengembangan aplikasi UMi Corner, posisi tersebut adalah sebagai berikut :

1. *Project Manager*
2. *Team Lead Engineer*
3. *UI/UX Designer*
4. *Backend Engineer*
5. *Frontend Engineer*
6. *Quality Assurance*
7. *Dev Ops*

Teknis pengerjaan selama kegiatan magang menggunakan *git*, flow yang berjalan adalah melakukan *clone* dari *repository project* yang disediakan kemudian membuat *branch* baru untuk pengerjaan fitur yang dilakukan. Selanjutnya mulai dilakukan pengerjaan pada fitur notifikasi. Setelah pengerjaan selesai akan dilakukan demo aplikasi terlebih dulu bersama *frontend engineer* untuk memastikan pengerjaan yang dilakukan sudah benar, jika sudah sesuai maka dilakukan *commit* yang

diisi dengan pesan yang berisi apa saja yang dilakukan atau perubahan apa yang dilakukan. Setelah itu melakukan *push project* ke *repository* dan melakukan *pull request* ke branch yang sudah ditentukan agar bisa dilakukan pengetesan kembali oleh *Quality Assurance* (QA).

### 3.2 Tugas yang Dilakukan

Tugas yang dilakukan selama kegiatan kerja magang di PT. Bank Rakyat Indonesia (Persero) Tbk sebagai *frontend engineer intern* pada tim UMi Corner adalah melakukan pengembangan fitur notifikasi dari sisi *frontend* untuk aplikasi UMi Corner versi kedua. Target dari pengerjaan fitur notifikasi ini adalah dapat menampilkan daftar notifikasi yang berasal dari pengajuan yang dilakukan. Adapun informasi yang ditampilkan pada halaman notifikasi adalah sebagai berikut :

1. Nama nasabah.
2. Status pengajuan nasabah.  
Terdiri dari terkirim, diproses, sukses, batal, dan *expired*.
3. Produk yang diajukan nasabah.
4. Waktu notifikasi.

Proses pengembangan fitur dari sisi *frontend* ini menggunakan bahasa pemrograman *typescript* dengan *framework ReactJS*, dan untuk *styling* menggunakan *framework css tailwind*. Secara garis besar proses pengerjaan yang dilakukan dalam membuat fitur ini adalah sebagai berikut :

1. Melakukan implementasi desain yang diberikan oleh *UI/UX designer*.
2. Melakukan integrasi dengan *backend*.
3. Membuat fungsi pendukung.
4. Melakukan perbaikan jika ada temuan *bug* dari *Quality Assurance*

Pada proses pengerjaan fitur notifikasi terdapat pengerjaan fungsi pendukung. Fungsi yang dibuat untuk mendukung fitur notifikasi ini berjalan sesuai dengan *requirement* diantaranya adalah *grouping* berdasarkan tanggal, menentukan label dari pengajuan apakah masuk hari ini atau kemarin, serta fungsi untuk mengatur notifikasi menjadi terbaca dan diarahkan ke halaman status pengajuan.

### 3.3 Uraian Pelaksanaan Magang

Pelaksanaan kegiatan magang diawali dengan mengikuti kegiatan *on boarding* yang diadakan oleh divisi *Human Capital Development* (HCD) PT. Bank Rakyat Indonesia (Persero) Tbk selama tiga hari. Setelah kegiatan *on boarding* selanjutnya mengikuti pertemuan perdana dengan mentor magang untuk dilakukan pembagian tim yang berada di bawah naungan divisi DDB. Dalam kegiatan tersebut mendapat tugas untuk ditempatkan pada tim UMi Corner.

Setelah diberikan tugas untuk bergabung dengan tim UMi Corner, selanjutnya mulai mengikuti proses kerja yang ada pada tim tersebut. Setiap harinya dilakukan *daily meeting* untuk melaporkan apa saja yang sudah dikerjakan. Tugas yang diberikan oleh *team lead engineer* adalah membuat fitur notifikasi dari sisi *frontend*, maka informasi mengenai teknologi yang digunakan, *flowchart*, dan *screenshot* hasil pengerjaan yang dilakukan akan dijelaskan pada bab ini. Pelaksanaan kerja magang yang dilakukan diuraikan seperti pada Tabel 3.2.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Kegiatan <i>on boarding</i> dan masa pengenalan lingkungan kerja
2	- Penerimaan dan penempatan pada tim yang dituju - Pertemuan dengan <i>project manager</i> dan <i>team lead engineer</i>
3	- Melakukan <i>setup project</i> - Membaca dokumentasi <i>project</i> dan mempelajari <i>ReactJs</i>
4	Melakukan pengerjaan pembuatan halaman <i>forbidden access</i>
5	- Melakukan implementasi UI/UX halaman notifikasi - Melakukan integrasi dengan data <i>dummy</i>
6	- Melakukan pengerjaan fungsi pendukung pada fitur notifikasi - Melakukan pengerjaan pada halaman <i>not found</i>
7	- Membuat <i>service</i> baru untuk menghitung jumlah nasabah dan pengajuan dalam <i>service summaries</i> - Melakukan integrasi dengan data <i>dummy</i> dari <i>backend</i> - Membuat fungsi dan integrasi untuk <i>set read notification</i> - Melakukan perubahan agar notifikasi dapat berjalan pada semua halaman

Tabel 3.2. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
8	<ul style="list-style-type: none"> <li>- Membuat dokumen <i>system integration testing</i> fitur notifikasi</li> <li>- Melakukan <i>fixing</i> desain halaman notifikasi</li> <li>- Membuat halaman katalog produk</li> </ul>
9	<ul style="list-style-type: none"> <li>- Melakukan restruktur pada fungsi <i>determine status</i> dengan menambahkan <i>response failed</i></li> <li>- Menampilkan pesan saat pengajuan <i>error</i></li> </ul>
10	Melakukan perbaikan pada tampilan pesan <i>error</i> dan <i>thumbnail</i> nama <i>user</i>
11	<ul style="list-style-type: none"> <li>- Membuat modal jika tidak ada pengajuan yang direkomendasikan</li> <li>- Melakukan perbaikan pada halaman notifikasi</li> </ul>
12	Melakukan <i>debugging</i> pada fitur notifikasi sambil menunggu API fix yang akan digunakan
13	Melakukan <i>debugging</i> pada fitur notifikasi sambil menunggu API fix yang akan digunakan
14	<ul style="list-style-type: none"> <li>- Membuat komponen dan tampilan baru untuk fitur tabungan emas</li> <li>- Membuat komponen cara pembayaran</li> </ul>
15	- Melakukan pencegahan pada rekomendasi berdasarkan hasil <i>pen-test</i>
16	<ul style="list-style-type: none"> <li>- Menambah <i>value</i> baru ke dalam <i>body request</i></li> <li>- Integrasi data baru</li> </ul>
17	<ul style="list-style-type: none"> <li>- <i>Deployment</i> fitur notifikasi</li> <li>- Membuat tampilan <i>restricted product</i></li> </ul>

### 3.3.1 Perancangan Fitur Notifikasi

Fitur notifikasi ini akan berjalan pada semua halaman, ini dilakukan agar jika ada notifikasi baru dapat menampilkan tanda dan jika ada notifikasi yang belum dibaca oleh mitra UMi. Proses ini dilakukan dengan membuat satu komponen yang diimplementasikan pada semua halaman, komponen ini berisi pemanggilan atau *fetching* pada *end point* notifikasi. Proses ini dilakukan dengan interval waktu selama 30 detik.

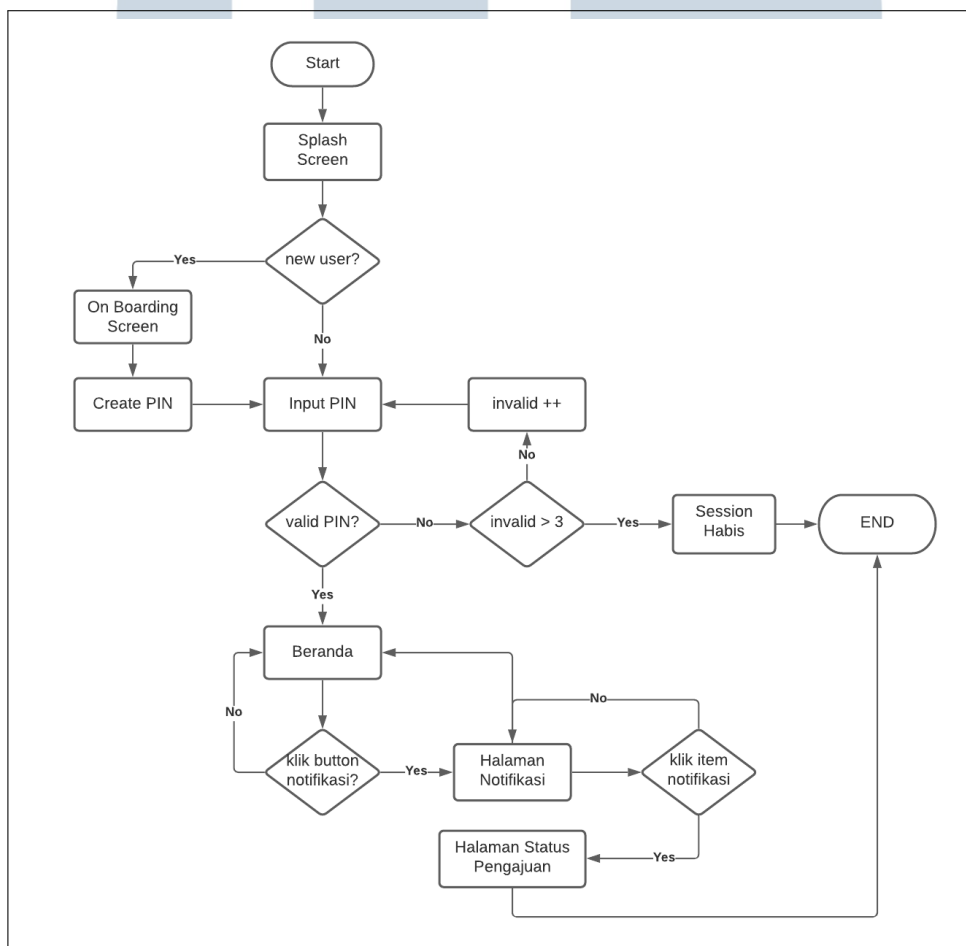
Dalam proses pengembangan fitur terdapat beberapa *tools* yang digunakan untuk mendukung proses pengembangan. Dalam hal manajemen *project* tim UMi Corner menggunakan Jira, kemudian untuk proses pengem-

bangan yang dilakukan oleh *developer* menggunakan *bitbucket* sebagai *version control system*, dan juga menggunakan *postman* untuk uji coba *REST API* yang sudah dibuat oleh *backend* dan juga *figma* yang digunakan untuk melihat desain dari tim UI/UX sebagai panduan saat implementasi UI

Berikut ini merupakan proses perancangan yang dilakukan dalam mengerjakan fitur notifikasi pada aplikasi UMi Corner.

### A. Flowchart

Gambar 3.1 merupakan *flowchart* dari fitur notifikasi



Gambar 3.1. *flowchart* fitur notifikasi

Proses yang akan dilalui oleh *user* dalam menggunakan fitur notifikasi akan dimulai saat menjalankan aplikasi masuk ke *splash screen* yang kemudian melakukan pengecekan apakah *user* yang masuk adalah *user* baru atau bukan, jika baru maka akan diarahkan ter-

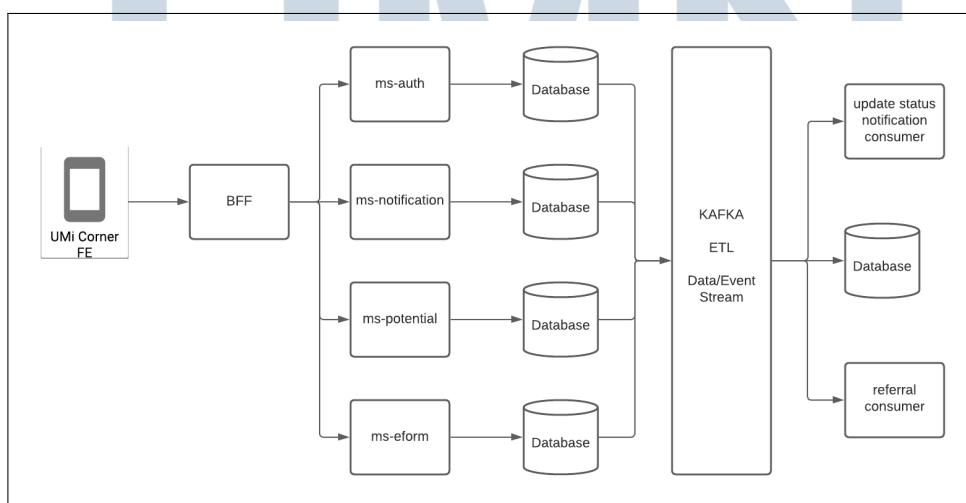
lebih dulu ke halaman *on boarding* yang kemudian baru akan masuk ke halaman untuk membuat PIN, dan selanjutnya akan diarahkan ke halaman input PIN. Sedangkan jika bukan *user* baru maka ketika menjalankan aplikasi setelah *splash screen* akan langsung diarahkan ke halaman input PIN.

Jika PIN yang dimasukkan valid akan diarahkan langsung ke halaman Beranda, jika terjadi kesalahan input PIN maka akan diarahkan kembali untuk melakukan input PIN. Jika terjadi kesalahan input sebanyak tiga kali maka sistem otomatis akan mengarahkan user ke halaman *session* habis dan user harus membuka kembali aplikasi.

Pada halaman Beranda terdapat *bottom navbar*, salah satu *button* pada *bottom navbar* tersebut adalah notifikasi. Jika user mengklik *button* notifikasi maka akan diarahkan ke halaman notifikasi. Pada halaman notifikasi akan ditampilkan data pengajuan nasabah. Jika user mengklik item notifikasi akan diarahkan pada halaman status pengajuan nasabah yang berisikan informasi pengajuan dari nasabah tersebut.

## B. Integrasi

Proses integrasi pada fitur notifikasi dengan *backend* dijelaskan pada gambar 3.2



Gambar 3.2. Proses integrasi dengan *backend*

Proses integrasi dengan *backend* pada fitur aplikasi ini di-

lakukan dengan melakukan pemanggilan *service* yang terkumpul pada BFF. BFF merupakan sebutan untuk *backend for frontend* dimana pada BFF terdapat banyak *service*, jadi dari sisi *frontend* cukup *hit* BFF saja. Selanjutnya BFF akan melakukan pemanggilan pada *service* yang dibutuhkan oleh *frontend*.

Dalam proses integrasi untuk pengerjaan fitur notifikasi ini *service* yang digunakan adalah *ms-notification*. *Service* tersebut terhubung dengan *database ms-notification* yang berisi *list* notifikasi yang akan ditampilkan berdasarkan pada pengajuan yang dilakukan. *Database* ini terhubung dengan kafka yang selanjutnya akan melakukan proses *update status notification consumer*.

### C. Debugging

Dalam proses perancangan fitur notifikasi terdapat proses *debugging*. Proses *debugging* dilakukan berdasarkan hasil temuan dari *quality assurance* pada saat melakukan pengetesan. Proses ini bertujuan untuk memastikan kembali hasil pengerjaan yang dilakukan oleh *developer* sudah sesuai dan tidak ada kesalahan yang dapat menyebabkan fitur tidak berjalan sesuai dengan ketentuan.

Selama proses pengerjaan fitur notifikasi yang dilakukan terdapat proses *debugging* pada hal berikut :

#### 1. Perbaiki *mapping status*

QA menemukan adanya kekeliruan saat melihat *response* pada status pengajuan SENT tetapi dari tampilan menunjukkan kalimat "Pengajuan Diproses". Proses *debugging* dilakukan dan ditemukan kesalahan saat *mapping status* untuk status SENT masih tergabung dengan status IN\_PROGRESS, maka selanjutnya dilakukan pemisahan untuk kasus dengan status pengajuan SENT dengan kalimat "Pengajuan Terkirim".

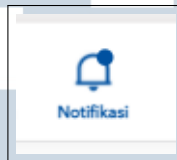
#### 2. Perbaiki urutan item notifikasi

QA menemukan kasus terdapat ketidaksesuaian urutan dari notifikasi yang masuk, setelah dilakukan penelusuran ditemukan bahwa terjadi kesalahan pengambilan *field response*. Perbaikan dilakukan dengan mengubah pengambilan *field*

untuk *grouping* tanggal yang sebelumnya pada *created\_at* menjadi *updated\_at*.

3. Data notifikasi kosong tetapi terdapat indikator pada *icon* notifikasi

Saat QA membuat akun baru dan masuk pada halaman beranda QA menemukan kasus bahwa terdapat indikator pada *icon* notifikasi (gambar 3.3) yang seharusnya menandakan adanya notifikasi, tetapi saat masuk ke halaman notifikasi dan cek pada response API tidak ada data notifikasi.



Gambar 3.3. Indikator notifikasi

Setelah dilakukan pengecekan ada indikasi ini terjadi karena efek dari penggunaan *redux* yang masih menyimpan data sebelumnya, maka perbaikan yang dilakukan adalah dengan menambahkan fungsi *clear state* saat *session* habis dan saat menjalankan *splash screen* yang menjadi halaman awal untuk *user* baru sehingga saat *user* baru masuk maka *default* statenya adalah kosong.

### 3.4 Hasil Pengerjaan

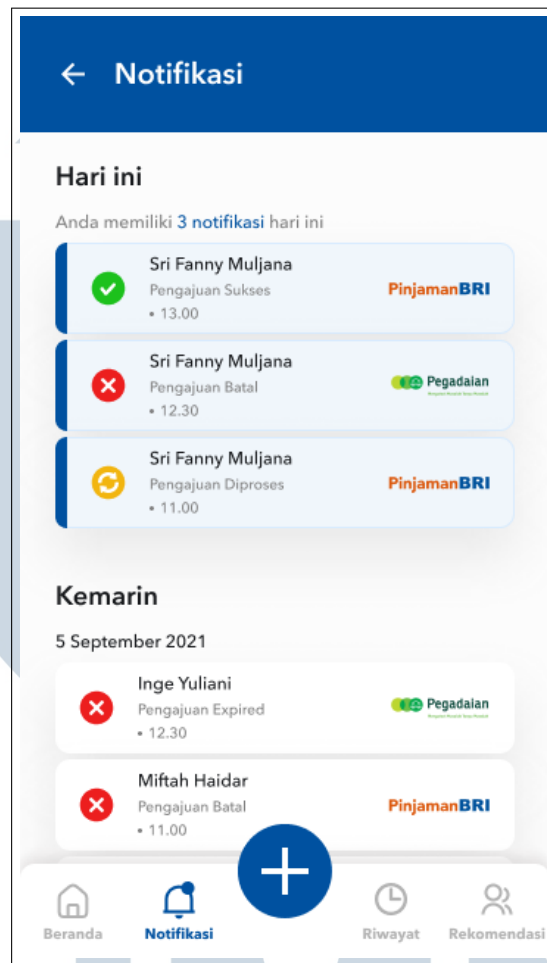
Hasil pengerjaan dari pembuatan fitur notifikasi yang dilakukan adalah sebagai berikut :

#### 3.4.1 Halaman Notifikasi

Berikut ini merupakan tangkapan layar dari halaman notifikasi yang dibuat berdasarkan pada desain yang diserahkan oleh tim UI/UX, pada halaman ini terdapat informasi yang ditampilkan, diantaranya adalah nama nasabah, status pengajuan nasabah, waktu, dan produk yang diajukan nasabah. Pada halaman ini juga terdapat pengelompokan atau *grouping* berdasarkan dari tanggal notifikasi tersebut masuk. Semua bagian pada halaman notifikasi ini akan dijelaskan secara lebih detail pada bagian di bawah



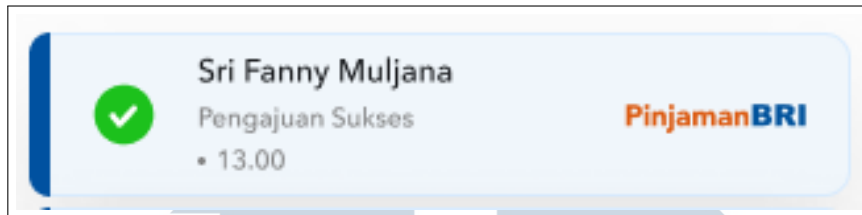
ini.



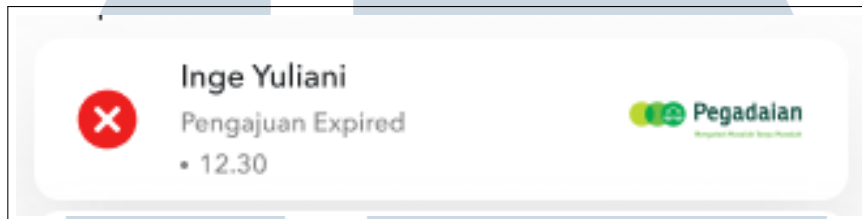
Gambar 3.4. Screenshot halaman notifikasi

### A. Komponen Item Notifikasi

Semua informasi yang ditampilkan dibuat dalam satu komponen yang dinamis, berikut ini merupakan komponen dari item notifikasi yang ditampilkan pada halaman notifikasi. Data yang ditampilkan pada komponen ini didapat dari API *get list notification*. Terdapat dua desain dalam komponen ini, pertama untuk notifikasi yang belum dibuka (gambar 3.5) dan satu lainnya adalah untuk item notifikasi yang sudah dibuka (gambar 3.6), hal ini bertujuan untuk memudahkan mitra umi selaku pengguna aplikasi ini melihat notifikasi mana yang belum mereka tindak lanjuti.



Gambar 3.5. Item notifikasi belum dibuka



Gambar 3.6. Item notifikasi sudah dibuka

Komponen ini menjadi sangat penting dikarenakan semua informasi ada pada komponen ini, informasi yang ditampilkan pada komponen ini adalah sebagai berikut :

1. Nama Nasabah

Nama nasabah didapat dari *field customer name*.

2. Status Pengajuan

Status pengajuan didapat dari *field to referral status* yang datanya akan terus diupdate berdasarkan status pengajuan nasabah dari sisi *backend* dan dari setiap perubahan statusnya akan ditampilkan dalam item baru.

3. Waktu

Informasi waktu pada item ini mengacu pada *field updated at* yang berdasarkan pada waktu perubahan status pengajuan nasabah.

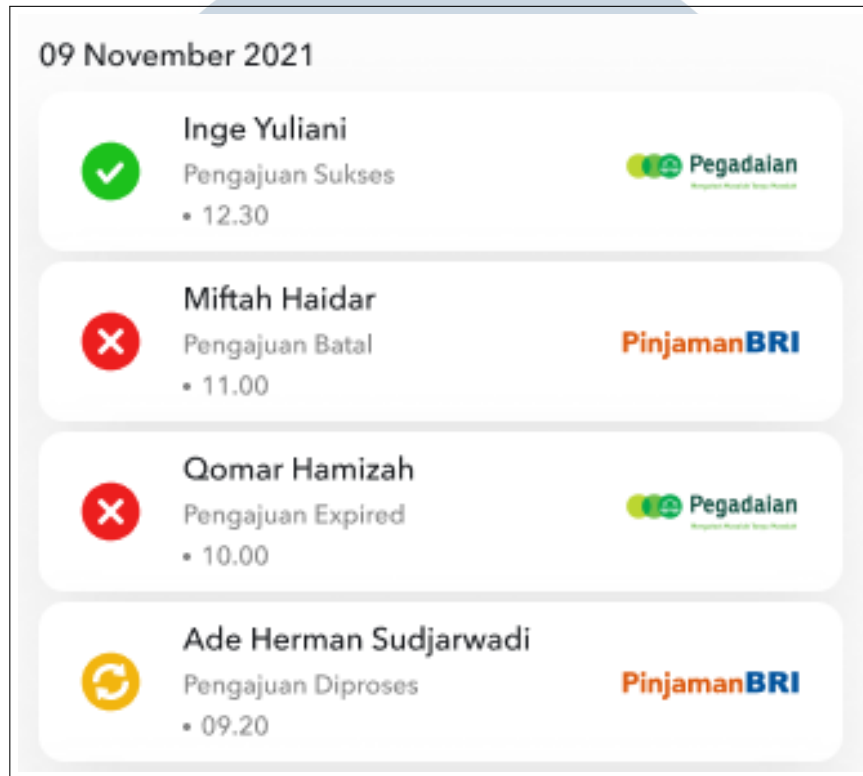
4. Produk yang diajukan

Produk yang diajukan oleh nasabah akan ditampilkan logo dari produknya, informasi ini didapat dari *field product id*.

## B. Grouping Tanggal

Penyusunan *list* pada notifikasi ini berdasarkan pada tanggal pengajuan dilakukan ataupun berdasarkan waktu dari perubahan status pengajuan nasabah yang diambil dari *field updated at*. *Grouping* tanggal pada fitur notifikasi ini dilakukan dengan menggunakan

fungsi *typescript* dimana pada awalnya akan dilakukan filter terlebih dulu untuk mengumpulkan data-data dengan tanggal yang sama, kemudian barulah akan ditampilkan seperti pada gambar 3.7.



Gambar 3.7. Item notifikasi sudah dibaca

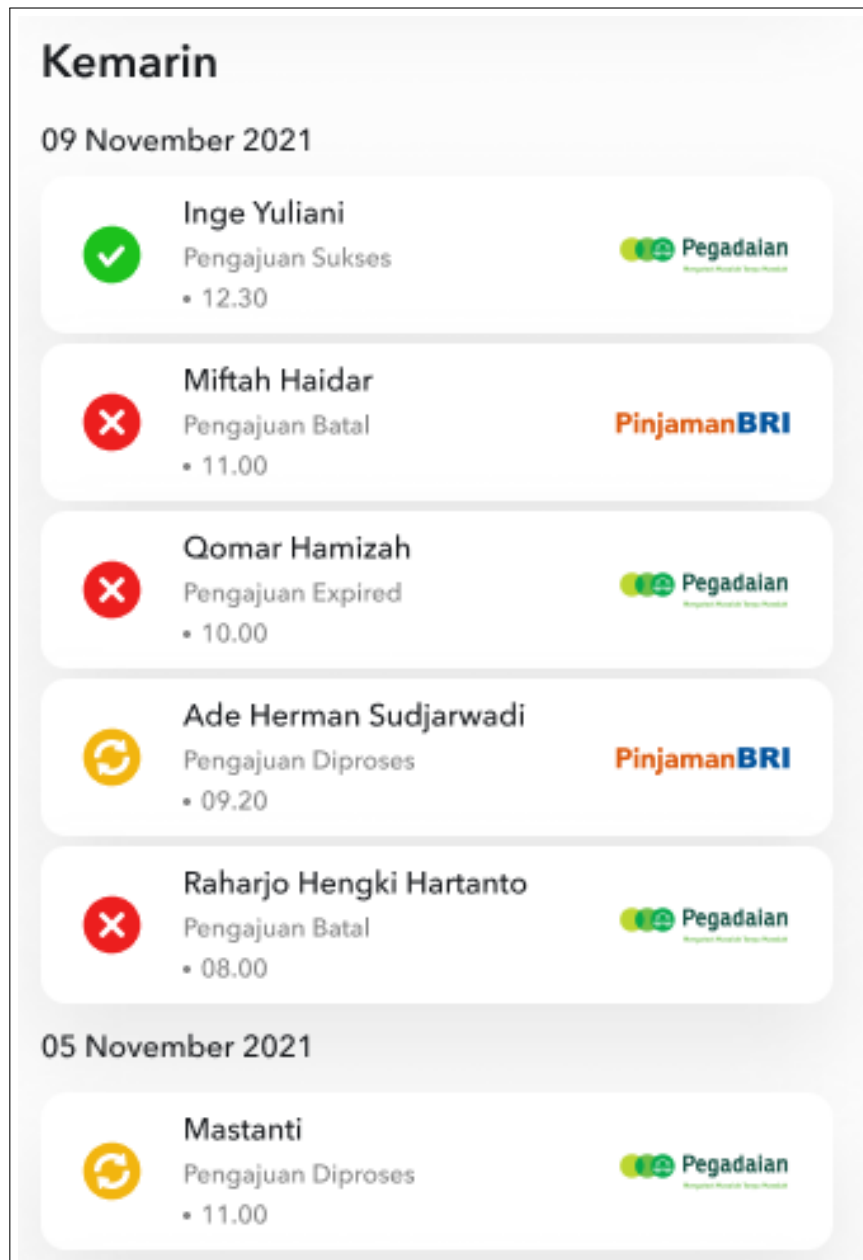
Pada fitur ini juga terdapat pengelompokkan terhadap notifikasi yang masuk pada hari ini (gambar 3.8) ataupun notifikasi yang masuk dari hari sebelumnya (3.9), untuk yang masuk pada hari ini akan diletakkan di atas dengan judul "Hari ini" dan sedangkan untuk yang sudah terlewat dari hari ini akan ditampilkan tepat di bawahnya dengan judul "Kemarin". Pada bagian kemarin ini akan menampilkan semua notifikasi yang masuk pada hari sebelumnya.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.8. Grouping notifikasi hari ini

UMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.9. Grouping notifikasi kemarin

### C. Icon Notifikasi

Icon notifikasi yang terdapat pada bottom tab bar navigation akan memunculkan tanda jika ada notifikasi yang belum dibaca, tanda tersebut akan terus ada sampai semua notifikasi sudah dibaca oleh mitra umi. Proses pengerjaan yang ada pada bagian ini adalah meletakkan sebuah fungsi pada komponen tab bar navigation yang akan membaca data dari *length* dari item notifikasi yang belum

dibaca.

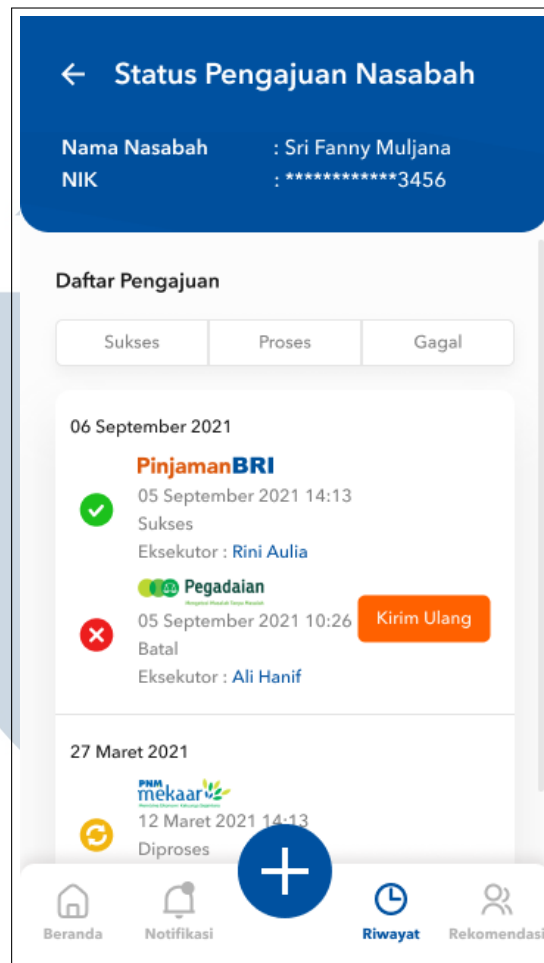


Gambar 3.10. Icon notifikasi

#### D. Redirect Page

Pada halaman notifikasi jika mitra umi men *tap* pada nasabah yang dipilih maka akan diarahkan pada halaman riwayat pengajuan dari nasabah tersebut. Pada halaman ini berisi informasi detail dari nasabah dan juga secara rinci riwayat dari pengajuan apa saja yang pernah diajukan oleh nasabah tersebut beserta statusnya. Tampilan dari halaman ini dapat dilihat pada gambar 3.11, halaman ini sudah ada sebelumnya jadi bukan bagian dari pengerjaan yang dilakukan dalam pembuatan fitur notifikasi. Kontribusi yang dilakukan pada bagian ini yaitu membuat fungsi agar ketika mitra umi men *tap* item notifikasi maka akan diarahkan ke halaman ini.

UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



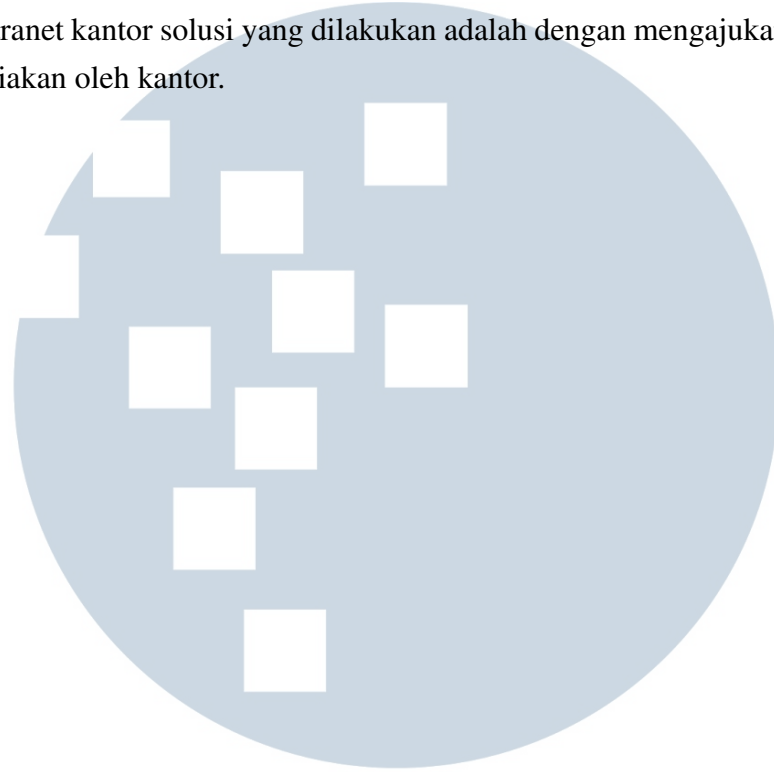
Gambar 3.11. Halaman riwayat pengajuan

### 3.5 Kendala dan Solusi yang Ditemukan

Selama melaksanakan kegiatan kerja magang di PT. Bank Rakyat Indonesia (Persero) Tbk terdapat beberapa kendala teknis yang ditemui, diantaranya adalah kurangnya pengetahuan terhadap *framework ReactJs* dengan bahasa pemrograman *typescript* serta tidak terbiasa dalam menggunakan *framework css tailwind*. Kendala lainnya yang ditemui adalah untuk menjalankan *project* aplikasi pada saat pengembangan harus menggunakan koneksi intranet kantor, ini menjadi kendala saat pengerjaan tugas dilakukan saat *work from home*.

Solusi yang dilakukan untuk mengatasi kendala selama kegiatan kerja magang berlangsung adalah membaca dokumentasi yang ada pada *website* ataupun membaca *blog* yang berkaitan dengan masalah yang dihadapi, solusi lain yang dilakukan adalah dengan bertanya pada pembimbing lapangan ataupun kepada teman

kampus yang memang memahami atau terbiasa dalam menggunakan *framework ReactJs*. Sedangkan dalam menjalankan *project* yang hanya bisa dijalankan dari koneksi intranet kantor solusi yang dilakukan adalah dengan mengajukan akses *vpn* yang disediakan oleh kantor.



# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA