

From Classroom to Boardroom:
***SOLVING REAL-
WORLD
PROBLEMS***



WHO



Gilang Kusuma Jati

Education

- University of Indonesia (Master's Degree, Computer Science)
- Institut Teknologi Telkom (Bachelor of Informatics Engineering)

Experience

- **CIMB Niaga**
 - Digital Enablement Development Group Head **(Jul 2025 - Present)**
 - Mobile & Internet Banking - Application Development Division Head **(May 2024 - Jul 2025)**
 - Mobile Banking - Application Development Division Head **(Sep 2023 - May 2024)**
- **Tokopedia**
 - Head of Engineering **(Sep 2020 - Jun 2023)**
 - Engineering Manager **(Mar 2019 - Sep 2020)**
- **Moka POS**
 - Engineering Manager **(Jan 2019 - Mar 2019)**
 - Lead Software Engineer **(Mar 2018 - Jan 2019)**
 - Senior Software Engineer **(Mar 2017 - Mar 2018)**
- **Samsung R&D Institute Indonesia (SRIN)**
 - Lead Engineer **(May 2015 - Mar 2017)**
- **University of Indonesia**
 - Lecturer **(Jan 2013 - Dec 2014)**
 - Teaching Assistant **(Aug 2011 - Jan 2013)**

01.

Cardless Cash Withdrawal



CASE STUDY 1: Cardless Cash

Withdrawal

Problem Statement

Nasabah perbankan kini semakin *mobile* dan menginginkan kemudahan dalam **mengakses layanan perbankan tanpa harus selalu membawa kartu fisik**. Dalam situasi sehari-hari—seperti saat lupa membawa dompet, atau saat keadaan darurat—nasabah sering kali membutuhkan cara untuk melakukan tarik tunai langsung di ATM tanpa kartu.

Fitur Tarik Tunai Tanpa Kartu (*Cardless Cash Withdrawal*) menawarkan solusi modern melalui aplikasi mobile banking. Nasabah cukup membuat token atau QR di aplikasi, lalu menggunakannya untuk menarik uang tunai di ATM. Namun, dari perspektif nasabah, terdapat sejumlah kebutuhan dan kendala nyata yang perlu dijawab oleh sistem.

Kebutuhan Nasabah

- 1. Akses fleksibel**
Nasabah ingin tetap bisa menarik tunai meskipun tidak membawa kartu fisik, misalnya saat bepergian ringan atau ketika kartu tertinggal di rumah/kantor.
- 2. Kemudahan berbagi uang tunai**
Nasabah menginginkan opsi untuk mengirim token tarik tunai kepada anggota keluarga/kerabat tanpa perlu memberikan kartu ATM atau PIN.
- 3. Keamanan tinggi**
Karena transaksi melibatkan uang tunai, nasabah menuntut perlindungan maksimal (misalnya OTP, biometrik, validitas token singkat) agar tidak ada penyalahgunaan.

CASE STUDY 1: Cardless Cash

Withdrawal

Ekspektasi Nasabah

- **Praktis:** cukup beberapa langkah di aplikasi untuk menghasilkan token/QR tarik tunai.
- **Cepat:** proses pembuatan token harus instan (< 3 detik) agar tidak menghambat.
- **Aman:** token/QR hanya berlaku singkat (≤ 5 menit), dilindungi OTP/biometrik, dan memiliki audit trail jelas.
- **Transparan:** adanya notifikasi real-time saat token dibuat, digunakan, atau kedaluwarsa.

Implikasi Studi Kasus

Dengan kebutuhan dan ekspektasi nasabah tersebut, tim pengembang diminta merancang fitur Cardless Cash di aplikasi mobile banking dengan memperhatikan aspek berikut:

Integrasi

- Konektivitas ke core banking serta switch jaringan ATM bank agar token/QR dapat divalidasi secara real-time.

Keamanan

- Penerapan OTP/biometrik, validitas token ≤ 5 menit, enkripsi, serta audit trail untuk pencegahan fraud.

Non-Fungsional

- Token harus dibuat ≤ 3 detik pada beban puncak (misalnya 500 TPS).
- Reliability tinggi saat penggunaan token di ATM.
- Error handling jelas (contoh: token expired, saldo tidak cukup).

Kepatuhan

- Pencatatan lengkap untuk audit & anti-fraud, serta perlindungan data pribadi sesuai regulasi privasi.

CASE STUDY 1

Pertanyaan

1. Susun komposisi tim Scrum dan tugasnya (PO, SM, Dev—FE/BE, QA, Sec/Compliance, DevOps).
2. Tulis ≥ 8 user stories berformat As a ... I want ... so that ... untuk fitur inti (buat token, verifikasi biometrik/OTP, lihat riwayat, pembatalan token, dsb.) + minimal 3 acceptance criteria untuk 2 user story terpenting.
3. Bentuk Product Backlog (10–14 item) dan bagi ke 2 Sprint (2 minggu/ sprint) dalam tabel: Backlog Item, Estimasi (story point), Ketergantungan, Sprint. Sertakan Definition of Done ringkas.
4. Jelaskan bagaimana Daily Stand-up meningkatkan komunikasi & transparansi. Beri 1 contoh impediment (mis. sertifikat mTLS expired) dan cara Anda mengeskalasinya. (4)
5. Bagaimana pendekatan Agile membantu tim beradaptasi dengan perubahan dalam ruang lingkup proyek, misalnya perubahan requirement dari awalnya menggunakan kode token dan nominal tarik tunai, menjadi nomor HP + kode token.



02.

Digital Account Opening

 **CIMB NIAGA**



CASE STUDY 2: Digital Account Opening

Problem Statement

Nasabah perbankan saat ini semakin menuntut kecepatan, kemudahan, dan fleksibilitas dalam mengakses layanan keuangan. Mereka ingin dapat membuka rekening baru tanpa harus datang ke cabang, mengantri, atau membawa dokumen fisik. Dalam situasi sehari-hari—seperti generasi muda yang ingin segera memiliki rekening untuk transaksi digital, pekerja yang sibuk, hingga masyarakat di daerah terpencil—proses pembukaan rekening konvensional di cabang sering kali dianggap lambat dan merepotkan.

Fitur Digital Account Opening melalui aplikasi mobile banking menawarkan solusi modern: nasabah cukup memotret e-KTP, melakukan verifikasi wajah (liveness detection), tervalidasi ke Dukcapil secara real-time, lalu menandatangani perjanjian secara elektronik. Setelah semua langkah terpenuhi, rekening langsung aktif secara otomatis tanpa harus ke cabang.

Namun, dari perspektif nasabah, ada sejumlah kebutuhan, kendala, dan ekspektasi nyata yang perlu dijawab agar pengalaman benar-benar seamless.

Kebutuhan Nasabah

- 1. Akses cepat dan tanpa batasan lokasi**
Nasabah ingin bisa membuka rekening dari mana saja, kapan saja, tanpa perlu hadir ke cabang bank.
- 2. Proses mudah tanpa dokumen fisik**
Nasabah menginginkan proses berbasis digital penuh—cukup dengan e-KTP dan kamera ponsel, tanpa perlu membawa fotokopi dokumen.
- 3. Kepastian keamanan dan validasi identitas**
Nasabah berharap data pribadinya aman, serta verifikasi dilakukan dengan sistem resmi (Dukcapil, tanda tangan elektronik tersertifikasi).
- 4. Aktivasi instan**
Setelah semua langkah selesai, rekening diharapkan langsung aktif untuk dipakai transaksi (transfer, top-up, pembayaran).

CASE STUDY 2: Digital Account Opening

Ekspektasi Nasabah

- **Praktis:** cukup foto e-KTP, selfie, dan tanda tangan digital langsung di aplikasi.
- **Cepat:** seluruh proses pembukaan rekening selesai dalam hitungan menit (< 5 menit).
- **Aman:** data terenkripsi, verifikasi wajah (liveness) memastikan identitas asli, dan ada notifikasi real-time.
- **Transparan:** nasabah mendapat kepastian status setiap tahap (dokumen diverifikasi, Dukcapil terhubung, rekening aktif).



Implikasi Studi Kasus

Dengan kebutuhan dan ekspektasi tersebut, tim pengembang diminta merancang Digital Account Opening dengan memperhatikan aspek berikut:

Integrasi

- Koneksi ke Dukcapil untuk verifikasi identitas resmi.
- Integrasi ke core banking agar rekening otomatis dibuat dan langsung aktif.
- Integrasi penyedia tanda tangan elektronik yang sudah tersertifikasi.

Keamanan

- OCR e-KTP dengan validasi checksum.
- Liveness detection untuk mencegah spoofing (foto/video palsu).
- Enkripsi end-to-end data pribadi.
- Audit trail digital untuk anti-fraud dan kepatuhan.

Non-Fungsional

- Proses pembukaan rekening harus selesai ≤ 5 menit dengan beban puncak ribuan transaksi per detik.
- Tingkat keberhasilan OCR & liveness $> 95\%$.
- Error handling jelas (contoh: gagal verifikasi wajah, e-KTP buram, jaringan Dukcapil down).

Kepatuhan

- Pencatatan lengkap sesuai regulasi OJK dan BI.
- Perlindungan data pribadi sesuai UU PDP.
- Tanda tangan elektronik harus sah secara hukum (UU ITE).

CASE STUDY 2



Problem

Bank ingin meningkatkan pembukaan rekening digital di aplikasi dengan menggunakan OCR e-KTP, liveness detection, verifikasi Dukcapil, tanda tangan elektronik, dan pembuatan rekening otomatis.

Kendala & Batasan

1. Alur bisnis end-to-end: input data → verifikasi identitas → persetujuan → pembukaan rekening → aktivasi.
2. Kepatuhan: 2FA/OTP, consent, penyimpanan log persetujuan.
3. UX: progres langkah, resume later, error copy jelas.



CASE STUDY 2



Pertanyaan

1. Buat Activity Diagram alur “Buka Rekening Digital” dari registrasi hingga rekening aktif (sertakan percabangan: kondisi jika liveness gagal, data Dukcapil mismatch, retry).
2. Buatlah minimal satu mockup tampilan utama dari alur yang dibuat di poin 1, lalu berikan penjelasan terkait mockup tersebut



03.

Scheduled BI-FAST Transfer



CASE STUDY 3: Scheduled BI-FAST

Transfer **Problem Statement**

Nasabah semakin rutin melakukan transfer antar bank untuk kebutuhan pribadi maupun bisnis—mulai dari gaji mingguan, uang saku keluarga, iuran rutin, hingga pembayaran vendor. Meski BI-FAST sudah menawarkan biaya rendah dan proses cepat, banyak nasabah masih harus mengulang input setiap kali jatuh tempo, rawan kelupaan, dan cemas soal keamanan transaksi bernilai besar.

Fitur Transfer Terjadwal BI-FAST menjawab kebutuhan tersebut: nasabah dapat menjadwalkan transfer harian/mingguan, menyimpan penerima tepercaya, dan menerima notifikasi status. Untuk keamanan, sistem menerapkan OTP saat penjadwalan (mencegah pembuatan instruksi tanpa sepengetahuan nasabah) serta OTP saat eksekusi untuk transaksi bernilai besar (di atas Rp10.000.000), sehingga kontrol ganda tetap terjaga.

Namun, agar benar-benar berguna, solusi harus mengatasi kebutuhan, kendala, dan ekspektasi nyata dari perspektif nasabah.

Kebutuhan Nasabah

- 1. Penjadwalan yang fleksibel**
Nasabah ingin mengatur frekuensi (harian/mingguan), hari tertentu, tanggal mulai/akhir, dan opsi skip jika jatuh pada hari libur.
- 2. Otomasi yang bisa dipercaya**
Instruksi berjalan tanpa input ulang, dengan riwayat jelas, serta opsi pause/ubah/batalkan kapan saja.
- 3. Keamanan transaksi bernilai tinggi**
Nasabah menghendaki lapisan otorisasi tambahan (OTP/biometrik) khusus untuk nominal besar serta pemberitahuan real-time.
- 4. Transparansi biaya & status**
Sebelum konfirmasi, nasabah melihat biaya BI-FAST, estimasi waktu, dan status eksekusi (berhasil/gagal/diulang).

CASE STUDY 3: Scheduled BI-FAST

Transfer

Ekspektasi Nasabah

- **Praktis:** pilih penerima, nominal, frekuensi (harian/mingguan), hari eksekusi, dan tanggal mulai/akhir.
- **Cepat:** penjadwalan tersimpan \leq 3 detik, eksekusi BI-FAST real-time.
- **Aman:** OTP saat penjadwalan, dan OTP saat eksekusi untuk $>$ Rp10 juta (atau sesuai limit kustom nasabah).
- **Transparan:** notifikasi pembuatan jadwal, pengingat H-x, hasil eksekusi, dan alasan kegagalan (saldo kurang, rekening tidak valid, limit terlampaui).



Implikasi Studi Kasus

Dengan kebutuhan & ekspektasi tersebut, tim diminta merancang Transfer Terjadwal BI-FAST dengan fokus pada:

Integrasi

- Core banking: penahanan (reservation) saldo saat eksekusi atau pre-check saldo; posting debit saat transaksi sukses.
- BI-FAST rails: validasi penerima (name/alias), pengiriman instruksi kredit, penerimaan status akhir (ACCP/RJCT).
- Notification service: push/in-app/SMS/email untuk pengingat & hasil eksekusi.
- Scheduler service: engine penjadwalan dengan dukungan RRULE (daily/weekly), timezone Asia/Jakarta, dan holiday calendar.

Keamanan

- OTP/biometrik saat penjadwalan (mencegah pembuatan jadwal oleh pihak tidak sah).
- OTP kedua saat eksekusi untuk $>$ Rp10 juta (nilai threshold dapat dikonfigurasi per kebijakan).
- Device binding & risk scoring (deteksi perangkat baru, IP anomali, perubahan pola).
- Idempotency key per eksekusi untuk mencegah double send saat retry.
- Audit trail menyeluruh (siapa/apa/kapan/dari perangkat mana), *immutable*.

CASE STUDY 3

Problem

Bank menambah fitur transfer antar bank via **BI-FAST dengan opsi terjadwal** (harian/mingguan) **dan menggunakan OTP** (one time password) saat penjadwalan & saat eksekusi transfer dengan nominal tertentu (nilai di atas threshold 10 juta rupiah).

Kendala & Batasan

1. Integrasi dengan BI-FAST
2. Auth/OTP
3. Core Banking
4. Audit/Monitoring; notifikasi.

CASE STUDY 3

Pertanyaan

1. Jelaskan setiap tahapan dan output yang dihasilkan, jika tim pengembangan perangkat lunak menggunakan Software Development Life Cycle (SDLC) untuk membangun fitur transfer antar bank via BI-FAST?



04.

***TOP UP E-MONEY
CARD VIA
MOBILE BANKING***

 **CIMB NIAGA**



CASE STUDY 4: Top-up Prepaid Card with NFC

Problem Statement

Di era digital, nasabah semakin mengandalkan **uang elektronik** (e-money) untuk kebutuhan sehari-hari: membayar tol, transportasi umum, parkir, hingga belanja di merchant offline. Namun, proses pengisian saldo (top up) masih sering dianggap **merepotkan**.

Saat ini, banyak nasabah harus mencari mesin EDC atau ATM tertentu untuk memperbarui saldo setelah melakukan top up. Hal ini mengurangi kenyamanan, apalagi ketika mereka sedang terburu-buru atau berada di lokasi yang jauh dari mesin update saldo.

Fitur **Top Up e-Money lewat Mobile Banking dengan NFC** menghadirkan solusi praktis: nasabah memilih nominal top up di aplikasi, saldo dipotong dari rekening, lalu kartu e-money ditempelkan ke ponsel (tap NFC) untuk langsung memperbarui saldo.

Namun, dari perspektif nasabah, masih ada sejumlah kebutuhan, kendala, dan ekspektasi yang harus dijawab oleh sistem agar fitur ini benar-benar bermanfaat.

Kebutuhan Nasabah

- 1. Kemudahan isi saldo kapan saja**
Nasabah ingin bisa mengisi e-money tanpa harus mencari ATM/EDC, cukup lewat ponsel dengan NFC.
- 2. Proses instan**
Nasabah mengharapkan saldo e-money bertambah seketika, tanpa perlu langkah tambahan.
- 3. Ketersediaan universal**
Fitur harus bisa digunakan di berbagai tipe ponsel dengan NFC yang kompatibel, tidak terbatas pada merek tertentu.
- 4. Keamanan transaksi**
Karena melibatkan uang, nasabah menuntut perlindungan maksimal (OTP, biometrik, notifikasi real-time).

CASE STUDY 4: Top-up Prepaid Card with NFC

Ekspektasi Nasabah

- **Praktis:** cukup pilih nominal, bayar dari rekening, dan tempelkan kartu ke ponsel.
- **Cepat:** keseluruhan proses ≤ 5 detik dari pembayaran hingga saldo kartu ter-update.
- **Aman:** dilindungi OTP/biometrik, notifikasi real-time, serta riwayat transaksi yang jelas.
- **Transparan:** aplikasi memberi status yang jelas (berhasil, gagal, saldo terbaru).

Implikasi Studi Kasus

Dengan kebutuhan dan ekspektasi tersebut, tim pengembang diminta merancang fitur **Top Up e-Money via Mobile Banking** dengan memperhatikan aspek berikut:

Integrasi

- Koneksi ke core banking untuk debit saldo rekening.
- Integrasi dengan sistem penerbit e-money (switch/payment gateway).
- Pemanfaatan modul NFC pada ponsel untuk menulis saldo ke chip e-money.

Keamanan

- OTP/PIN/biometrik untuk otorisasi transaksi.
- Mekanisme rollback jika saldo rekening terpotong tetapi NFC gagal update.
- Audit trail untuk investigasi transaksi gagal/dobel.

Non-Fungsional

- Proses top up & update saldo ≤ 5 detik.
- Tingkat keberhasilan NFC $> 98\%$ pada beban puncak (misalnya 300 TPS).
- Error handling jelas (contoh: NFC tidak terbaca, saldo rekening tidak cukup, kartu tidak valid).

Kepatuhan

- Pencatatan transaksi sesuai regulasi Bank Indonesia untuk uang elektronik.
- Perlindungan data pribadi sesuai UU PDP.
- Mekanisme anti-fraud, termasuk deteksi duplikasi update saldo.

CASE STUDY 4



Problem

Bank ingin menambah fitur **Top Up kartu e-money** (saldo tersimpan di chip) **lewat aplikasi mobile banking**. Nasabah memilih nominal, bayar dari rekening, lalu update saldo ke kartu via NFC (tap kartu ke ponsel).

Kendala & Batasan

1. Nominal top up: Rp20.000–Rp1.000.000 (kelipatan Rp10.000).
2. Masa berlaku sesi NFC 60 detik.
3. Integrasi cukup ke Payment Service bank & Issuer e-money (dianggap 1 API).
4. Logging dasar & push notification.



CASE STUDY 4



Pertanyaan

1. Buat Use Case Diagram sederhana untuk aktor Nasabah dan Sistem Mobile Banking (use case minimal: pilih nominal, bayar, update saldo via NFC, lihat riwayat).
2. Tulis ≥ 5 user story format As a ... I want ... so that ... (contoh: pilih nominal, bayar, update NFC, gagal NFC \rightarrow retry, lihat riwayat). Pilih 2 yang paling penting dan buat Acceptance Criteria singkat (Given/When/Then).
3. Buat tabel backlog (8–10 item) berisi ID, Deskripsi, Estimasi (SP), Prioritas. Bagi ke 1 Sprint (2 minggu). Tambahkan DoD ringkas (mis. Unit testing $\geq 70\%$, happy path QA lulus, log transaksi tersimpan).



05.

Top-up Pulsa & Paket Data

 CIMB NIAGA



CASE STUDY 5: Top-up Pulsa & Paket Data

Problem Statement

Nasabah semakin mengandalkan ponsel untuk bekerja, belajar, dan hiburan. Ketika kuota data habis atau pulsa menipis, mereka membutuhkan cara **isi ulang yang cepat, mudah, dan andal** tanpa harus berpindah aplikasi atau datang ke gerai fisik.

Fitur **Isi Pulsa/Paket Data** di aplikasi mobile banking menjanjikan kemudahan: nasabah memilih operator & denominasi, membayar dari rekening, lalu bank meneruskan order ke Agregator Telco dan menunggu callback sukses/gagal. Jika gagal, sistem otomatis refund ke saldo nasabah.

Agar benar-benar memuaskan, solusi harus menjawab kebutuhan, kendala, dan ekspektasi nyata dari perspektif nasabah.

Kebutuhan Nasabah

1. **Kemudahan lintas operator**
Satu aplikasi untuk semua operator tanpa perlu berpindah platform.
2. **Kecepatan dan kepastian**
Top up harus diproses dalam hitungan detik dengan status yang jelas (berhasil/gagal), termasuk nomor referensi.
3. **Pilihan denominasi & paket yang relevan**
Tersedia berbagai nominal pulsa/paket data (harian, mingguan, bulanan), termasuk promo/bonus bila ada.
4. **Keamanan & transparansi biaya**
Otorisasi kuat (OTP/biometrik), informasi harga all-in (termasuk pajak/biaya), dan notifikasi real-time.

CASE STUDY 5: Top-up Pulsa & Paket Data

Ekspektasi Nasabah

- **Praktis:** pilih operator & denominasi → konfirmasi → bayar → terima notifikasi hasil.
- **Cepat:** konfirmasi status ≤ 10 detik sejak pembayaran.
- **Aman:** OTP/biometrik, riwayat transaksi, dan bukti digital (invoice/e-receipt).
- **Transparan:** jika gagal, refund otomatis & instan ke saldo, dengan penjelasan alasan kegagalan.

Implikasi Studi Kasus

Dengan kebutuhan & ekspektasi tersebut, tim diminta merancang fitur Isi Pulsa/Paket Data dengan fokus pada:

Integrasi

- Core banking: pendebitan rekening & pengelolaan reversal/refund.

Aggregator Telco:

- API order (create top-up) + callback/webhook hasil (success/failed/unknown).
- Katalog (denominasi/paket) sinkron berkala, termasuk harga & ketersediaan.
- Health check dan pengalihan ke jalur/aggregator cadangan (jika ada).
- Idempotency pada order (idempotency-key) untuk mencegah double debit saat retry.

Keamanan

- Otorisasi via OTP/biometrik; pembatasan harian/transaction limit.
- Validasi MSISDN (format, operator detection).
- Audit trail end-to-end (request, response, callback, refund).

UX & Transparansi

- Validasi real-time nomor & operator; autofill dari kontak (opsional, dengan izin).
- Ketersediaan paket selalu terbaru; tampilkan ETA pemrosesan & status live.
- E-receipt berisi no. referensi bank & aggregator, waktu, denominasi, biaya, dan kebijakan *refund*.

Pertanyaan

1. Susun komposisi Tim dengan pendekatan **Agile** beserta peran dan tanggung jawabnya (Product Owner, Scrum Master, Developer (backend, mobile, web), Quality Assurance, Security/Compliance, DevOps, etc.)
2. Buat *end-to-end flow* dalam format **Swimlane** untuk menggambarkan proses dari studi kasus yang dipilih dan menjawab kebutuhan user.
3. Tulis minimum **5 user stories** berformat **As a ... I want ... so that...** dengan minimum **acceptance criteria** untuk masing-masing *user story*.
4. Indikator apa saja yang perlu digunakan untuk mengukur tingkat keberhasilan setelah studi kasus berhasil diimplementasikan?
5. Identifikasi **risiko** (potensi kerugian) dari flow proses yang sudah dibuat, dan berikan *risk mitigation plan* nya.

CASE STUDY 5

Problem

Bank ingin **menambah menu isi pulsa/paket data ke semua operator**. Aplikasi menampilkan denominasi, nasabah bayar dari rekening, bank meneruskan ke Agregator Telco dan menunggu callback sukses/gagal. Jika gagal, sistem otomatis refund ke saldo nasabah.

Kendala & Batasan

1. Denominasi disediakan aggregator (10k–200k).
2. SLA respons aggregator ≤ 15 detik; jika time-out \rightarrow status pending dan diselesaikan via callback.
3. Push notification + email.
4. Rate limit 3 transaksi/menit per nasabah.

CASE STUDY 5

Pertanyaan

1. Activity Diagram alur dasar: pilih operator → pilih nominal → bayar → request ke aggregator → (sukses/gagal/timeout) → notifikasi → (jika gagal) refund. Sertakan 1 percabangan timeout.
2. Tulis min. 8 kebutuhan, pisah Fungsional (pilih operator, pilih nominal, bayar, terima callback, tampilkan status, refund otomatis) dan Non-Fungsional (SLA, rate limit, audit log sederhana, validasi MSISDN). (6)

CASE STUDY 5

Pertanyaan

1. Activity Diagram alur dasar: pilih operator → pilih nominal → bayar → request ke aggregator → (sukses/gagal/timeout) → notifikasi → (jika gagal) refund. Sertakan 1 percabangan timeout.
2. Tulis min. 8 kebutuhan, pisah Fungsional (pilih operator, pilih nominal, bayar, terima callback, tampilkan status, refund otomatis) dan Non-Fungsional (SLA, rate limit, audit log sederhana, validasi MSISDN). (6)

THANKS!

DO YOU HAVE ANY QUESTIONS?



CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**

