

NYUMBANGIN: PLATFORM DONASI DIGITAL

LAPORAN PROYEK III

Diajukan untuk Memenuhi Kelulusan Matakuliah
Proyek 3 pada Program Studi DIV Teknik Informatika



Universitas Logistik & Bisnis Internasional

DISUSUN OLEH :

714230027 – Muhamad Haekal Syukur
714230060 – Muhammad Ferdy Leoza

**PROGRAM STUDI DIV TEKNIK INFORMATIKA
FAKULTAS SEKOLAH VOKASI
UNIVERSITAS LOGISTIK & BISNIS INTERNASIONAL
BANDUNG
2026**

LEMBAR PENGESAHAN
NYUMBANGIN: PLATFORM DONASI DIGITAL

Laporan Proyek 3 ini telah diperiksa, disetujui, dan disidangkan

Di Bandung,

Penguji Pendamping,

Penguji Utama,

Rolly Maulana Awangga, S.T., M.T.
(Simulasi)

Rolly Maulana Awangga, S.T., M.T.

NIK: 113.74.163

NIK: 1

Pembimbing,

Koordinator Proyek 3

Roni Habibi, S.Kom., M.T., SFPC

Rolly Maulana Awangga, S.T., M.T.

NIK: 117.86.219

NIK: 117.88.233

Menyetujui,

Ketua Program Studi D-IV Teknik Informatika,

Roni Andarsyah, S.T., M.Kom

NIK: 115.88.193

SURAT PERNYATAAN BEBAS PLAGIARISME

Nama : Muhamad Haekal Syukur
NPM : 714230027
Program Studi : DIV Teknik Informatika
Judul : NYUMBANGIN: PLATFORM DONASI DIGITAL

Menyatakan bahwa:

1. Proyek pemrograman aplikasi (PROYEK 3) ini adalah karya asli yang belum pernah diajukan untuk memenuhi kelulusan pada program studi DIV Teknik Informatika di Universitas Logistik & Bisnis Internasional maupun di perguruan tinggi lainnya.
2. Proyek ini merupakan hasil pemikiran, rumusan, dan penelitian saya sendiri, tanpa adanya bantuan dari pihak lain, kecuali arahan yang diberikan oleh pembimbing.
3. Dalam proyek ini, tidak terdapat karya atau pendapat yang ditulis atau dipublikasikan oleh orang lain, kecuali jika telah dicantumkan secara tertulis sebagai acuan dalam naskah.
4. Saya menyatakan bahwa pernyataan ini dibuat dengan sesungguhnya.

Bandung, Januari 2026
Yang membuat pernyataan,

Muhamad Haekal Syukur

NPM : 714230027

SURAT PERNYATAAN BEBAS PLAGIARISME

Nama : Muhammad Ferdy Leoza
NPM : 714230060
Program Studi : DIV Teknik Informatika
Judul : NYUMBANGIN: PLATFORM DONASI DIGITAL

Menyatakan bahwa:

1. Proyek pemrograman aplikasi (PROYEK 3) ini adalah karya asli yang belum pernah diajukan untuk memenuhi kelulusan pada program studi DIV Teknik Informatika di Universitas Logistik & Bisnis Internasional maupun di perguruan tinggi lainnya.
2. Proyek ini merupakan hasil pemikiran, rumusan, dan penelitian saya sendiri, tanpa adanya bantuan dari pihak lain, kecuali arahan yang diberikan oleh pembimbing.
3. Dalam proyek ini, tidak terdapat karya atau pendapat yang ditulis atau dipublikasikan oleh orang lain, kecuali jika telah dicantumkan secara tertulis sebagai acuan dalam naskah.
4. Saya menyatakan bahwa pernyataan ini dibuat dengan sesungguhnya.

Bandung, Januari 2026
Yang membuat pernyataan,

Muhammad Ferdy Leoza

NPM : 714230060

ABSTRAK

Aplikasi donasi digital kini banyak digunakan oleh kreator konten untuk memudahkan dukungan dari para pendukung. Laporan ini membahas pengembangan platform donasi dengan fitur utama seperti pengiriman donasi, notifikasi real-time melalui overlay, leaderboard pendukung, serta mekanisme pencairan dana bagi kreator. Pengembangan dilakukan melalui analisis kebutuhan, perancangan sistem, dan implementasi fitur sesuai alur donasi hingga pencairan dana. Hasil pengujian menunjukkan bahwa aplikasi dapat memproses donasi dengan baik, menampilkan notifikasi secara langsung dan menyediakan proses pencairan dana yang terstruktur untuk kreator. Secara keseluruhan, aplikasi yang dibangun telah memenuhi tujuan utama, yaitu menyediakan sarana donasi yang fungsional dan mudah digunakan, meskipun masih terdapat ruang pengembangan lebih lanjut untuk meningkatkan stabilitas dan cakupan fitur.

Kata Kunci: donasi digital, aplikasi web, kreator, notifikasi overlay, payout.

ABSTRACT

Digital donations applications are now widely used by content creators to make easier for supporters to contribute. This report covers the development of a donation platform with key features such as donation delivery, real-time notifications via overlays, supporter leaderboards, and fund disbursement for creators. The development process included needs analysis, system design, and feature implementation following the donations flow up to payout. Testing show the applications can process donations well, display notifications instantly, and provide a structured fund disbursement process for creators. Overall, the applications that was built has met its main objective, which is to provide a functional and easy-to-use donation tool, although there is still room for further development to improve stability and feature coverage.

Keywords: *digital donations, web applications, creator, overlay notifications, payouts.*

DAFTAR ISI

1 Pendahuluan	8
1.1 Latar Belakang	8
1.2 Identifikasi Masalah	9
1.3 Tujuan	9
1.4 Ruang Lingkup	10
1.4.1 Cakupan Fungsional	10
1.4.2 Cakupan Teknis	11
1.4.3 Cakupan Pengujian	11
1.4.4 Batasan Penilitian	12
1.4.5 Luaran Penilitian	12
1.5 Sistematika Penulisan	12
2 Landasan Teori	14
2.1 Platform Donasi Digital	14
2.2 Arsitektur Aplikasi Web Modern	15
2.2.1 Arsitektur Three-Tier/N-Tier	15
2.2.2 Konsep Client-Server	15
2.3 REST API dan Protokol HTTP	15
2.4 Autentikasi dan Otorisasi	16
2.4.1 OAuth 2.0	16
2.4.2 OpenID Connect	16
2.4.3 JWT	17
2.4.4 Skema Bearer Token	17
2.4.5 Keamanan API	17
2.5 Database NoSQL (MongoDB)	18
2.6 Flowchart Sistem	18
2.7 Unified Modeling Language (UML)	19
2.7.1 Use Case Diagram	19
2.7.2 Activity Diagram	19
2.7.3 Sequence Diagram	20

2.7.4	Class Diagram	20
2.8	Peran Fitur Social Sharing dalam Donasi Digital	20
3	Metode Penelitian	22
3.1	Metode Pengembangan Sistem	22
3.1.1	Konsep Agile Development	22
3.1.2	Alur Iterasi Pengembangan	23
3.1.3	Penerapan Agile pada Proyek Nyumbangin	23
3.2	Analisis Kebutuhan	24
3.2.1	Sumber Kebutuhan	24
3.2.2	Kebutuhan Fungsional	24
3.2.3	Kebutuhan Non-Fungsional	25
3.3	Perancangan Sistem	25
3.3.1	Arsitektur Logis	25
3.3.2	Arsitektur Fisik	25
3.3.3	Arsitektur Teknologi	26
3.3.4	Modul Utama	27
3.3.5	Strategi Desain	28
3.4	Pemodelan Sistem	28
3.4.1	Use Case	29
3.4.2	Activity Diagram	29
3.4.3	Activity Diagram Proses Payout	31
3.4.4	Activity Diagram Feedback	32
3.4.5	Activity Diagram Login	33
3.4.6	Activity Diagram Kelola Overlay	34
3.4.7	Activity Diagram Kelola Profil	35
3.4.8	Activity Diagram Request Payout	36
3.4.9	Activity Diagram Riwayat Donasi	37
3.4.10	Sequence Diagram	38
3.4.11	Model Entitas	45
3.5	Metode Perancangan Teknis	46
3.6	Metode Pengujian	47
3.6.1	Jenis Pengujian	47
3.6.2	Skenario Pengujian	48
3.6.3	Alat Pengujian	49
3.7	Evaluasi Keberhasilan	49
3.7.1	Cakupan Pengujian	49
3.7.2	Hasil Code Coverage	49

3.7.3 Interpretasi dan Evaluasi	50
3.7.4 Kesimpulan Evaluasi	51

DAFTAR GAMBAR

3.1	Arsitektur Teknologi	27
3.2	Use Case Diagram	29
3.3	Activity Diagram Donasi	30
3.4	Activity Diagram Payout	31
3.5	Activity Diagram Feedback	32
3.6	Activity Diagram Login	33
3.7	Activity Diagram Kelola Overlay	34
3.8	Activity Diagram Kelola Profil	35
3.9	Activity Diagram Request Payout	36
3.10	Activity Diagram Riwayat Donasi	37
3.11	Sequence Diagram Donasi	38
3.12	Sequence Diagram Leaderboard	39
3.13	Sequence Diagram Login	40
3.14	Sequence Diagram Feedback	41
3.15	Sequence Diagram Kelola Profil	42
3.16	Sequence Diagram Riwayat Donasi	43
3.17	Sequence Diagram Request Payout	44

DAFTAR TABEL

3.1 Code Coverage	50
-----------------------------	----

BAB 1

Pendahuluan

1.1 Latar Belakang

Perkembangan teknologi digital telah membawa perubahan yang signifikan dalam praktik penggalangan dana, khususnya melalui platform donasi digital berbasis crowdfunding. Secara global, donation-based crowdfunding berkembang pesat karena mampu menghubungkan donatur dan penerima manfaat secara efisien tanpa batasan geografis. Penelitian menunjukkan bahwa platform donasi digital berperan penting dalam meningkatkan akses pendanaan sosial serta mendorong partisipasi masyarakat melalui pemanfaatan teknologi informasi [48].

Dalam konteks donasi digital, berbagai studi menekankan bahwa faktor kepercayaan (*trust*) dan inovasi teknologi merupakan determinan utama dalam Keputusan berdonasi. Inovasi platform crowdfunding serta Tingkat kepercayaan pengguna berpengaruh positif terhadap Keputusan donasi online [60]. Selain itu, persepsi transparansi dan keamanan sistem turut membentuk keyakinan donator dalam menggunakan olat-platform donasi digital. Temuan ini di perkuat oleh penelitian [47] yang menunjukkan bahwa kepercayaan dan persepsi risiko memiliki pengaruh signifikan terhadap Keputusan donasi online, khususnya pada platform berbasis web.

Di Indonesia, perkembangan donasi digital sejalan dengan meningkatnya adopsi teknologi finansial. Integrasi sistem pembayaran digital, seperti QRIS, pada platform donasi terbukti mempermudah proses transaksi serta meningkatkan efisiensi dan aksesibilitas bagi Masyarakat. Studi [58] menunjukkan bahwa penggunaan QRIS pada platform donation-based crowdfunding mampu meningkatkan kenyamanan pengguna dan mempercepat proses donasi, meskipun tantangan terkait literasi digital dan kepercayaan terhadap platform masih menjadi perhatian.

Meskipun demikian, sejumlah penelitian mengindikasikan bahwa tantangan utama dalam donasi digital tidak hanya terletak pada aspek teknis pembayaran, tetapi juga ada upaya membangun kepercayaan pengguna secara berkelanjutan. Inovasi fitur pada platform donasi dinilai dapat berkontribusi dalam meningkatkan partisipasi donator apabila mampu memberikan pengalaman pengguna yang transparan dan interaktif [60, 47].

Berdasarkan kondisi tersebut, aplikasi Nyumbangin di kembangkan sebagai platform donasi digital berbasis web dan mengintegrasikan sistem pembayaran digital serta fitur pendukung seperti notifikasi real-time, leaderboard donator, dan media sharing. Fitur media sharing dirancang untuk mendorong keterlibatan sosial dan meningkatkan kepercayaan pengguna melalui mekanisme berbagi aktivitas donasi, yang berdasarkan penelitian sebelumnya terbukti dapat mempengaruhi partisipasi donator. Dengan pendekatan tersebut, aplikasi Nyumbangin diharapkan mampu memberikan pengalaman donasi digital yang transparan, interaktif, dan terpercaya.

1.2 Identifikasi Masalah

Meskipun platform donasi digital sudah banyak tersedia, sebagian besar memiliki kompleksitas fitur dan arsitektur yang cukup tinggi, sehingga kurang sesuai untuk dipelajari atau dijadikan dasar pengembangan mandiri. Selain itu, implementasi digital payment membutuhkan contoh sistem yang sederhana namun fungsional agar dapat dipahami dengan mudah.

Berdasarkan kondisi tersebut, kebutuhan yang muncul adalah:

1. Kebutuhan untuk memahami sekaligus mempraktikkan implementasi sistem Digital Payment melalui proyek nyata.
2. Kebutuhan akan platform donasi yang ringan dan sederhana, tanpa kompleksitas berlebih.
3. Kebutuhan akan sebuah media belajar dan inovasi, yang tetap memiliki potensi digunakan oleh publik.

1.3 Tujuan

Tujuan utama dari pengembangan platform Nyumbangin adalah membangun sistem donasi digital yang sederhana, fungsional, dan dapat menjadi dasar pengembangan lebih lanjut. Secara khusus, tujuan proyek ini adalah:

1. Merancang dan mengimplementasikan platform donasi sederhana sebagai penerapan teknologi, termasuk Digital Payment.
2. Menyediakan alternatif platform donasi yang ringan, fleksibel, dan mudah dikembangkan sesuai kebutuhan.
3. Membangun pondasi produk digital yang dapat diekspansi menjadi sistem yang lebih kompleks di masa depan sekaligus menjadi sarana pembelajaran dan inovasi.

1.4 Ruang Lingkup

Ruang lingkup pengembangan ini difokuskan pada perancangan, pembangunan, dan evaluasi prototipe Nyumbangin, yaitu platform web sederhana untuk penggalangan dukungan/donasi bagi kreator. Lingkup ini mencakup proses analisis kebutuhan, perancangan arsitektur dan model data, implementasi modul inti, serta pengejalan fungsional dalam lingkungan pengembangan.

1.4.1 Cakupan Fungsional

Penilitian ini mencakup pengembangan fitur inti sebagai berikut:

1. Manajemen Kreator

Melibati registrasi, autentikasi, dan donasi otorisasi kreator menggunakan dua metode:

- JSON Web Token (JWT) untuk autentikasi berbasis username/password.
- OAuth Google Sign-in sebagai metode login alternatif menggunakan akun Google.

2. Pengelolaan Donasi

Pencatatan transaksi donasi (nominal, waktu, dan identitas donor terbatas/anonym), penyimpanan data donasi, serta penyajian ringkasan donasi kepada kreator.

3. Dashboard Kreator

Penyajian metrik dasar seperti jumlah donasi, total nominal terkumpul, dan daftar 10 donatur terbaru.

4. Leaderboard Global

Agregasi donasi lintas kreator untuk menampilkan peringkat donasi secara

global melalui endpoint contoh GET /api/dashboard/leaderboard dengan verifikasi JWT tipe creator.

5. Antarmuka Pengguna Web

Halaman publik untuk menampilkan profile kreator dan melakukan donasi, serta halaman privat untuk dashboard kreator.

1.4.2 Cakupan Teknis

Secara teknis, penilitian ini mencakup:

1. Arsitektur Aplikasi

Pembangunan sistem berbasis Next.js dengan API Routes sebagai backend, Node.js runtime, dan NoSQL MongoDB menggunakan Mongoose.

2. Keamanan Dasar Sistem

Meliputi:

- Implementasi JWT untuk login tradisional
- Integrasi OAuth 2.0 Google Sign-In
- Validasi input dan sanitasi sederhana
- Penanganan akses endpoint privat berdasarkan token

3. Perancangan dan Pemodelan

Meliputi use case diagram, flowchart proses (alur donasi, autentikasi, leaderboard), dan rancangan model data (Creator, Donation).

4. Integrasi Modul Internal

Modul koneksi database, middleware verifikasi token (JWT & OAuth), utilitas hashing, serta pengelolaan data melalui shcema Creator dan Donation.

1.4.3 Cakupan Pengujian

Pengujian dilakukan meliputi:

1. Uji fungsional terhadap endpoint inti seperti autentikasi kreator, pencatatan/pengambilan donasi, dan pemuatan data leaderboard.
2. Uji integritas sederhana untuk memastikan alur donasi hingga dashboard berjalan end-to-end menggunakan data simulasi.

1.4.4 Batasan Penilitian

Penilitian ini dibatasi pada aspek-aspek berikut:

1. Fokus Pada Alur Donasi Dasar

Sistem hanya mencakup proses donasi sederhana tanpa fitur pendukung seperti manajemen kampanye, penjadwalan donasi, atau komisi.

2. Integrasi Pembayaran Bersifat Simulasi

Payment gateway digunakan dalam mode sandbox untuk tujuan pembelajaran dan pengujian; tidak mencakup transaksi finansial nyata, KYC/AML, atau kepatuhan regulasi.

3. Lingkup Pengembangan Berskala Proyek

Optimasi performa produksi, skalabilitas tinggi, dan security hardering tingkat lanjut tidak menjadi fokus utama.

1.4.5 Luaran Penilitian

Luaran yang dihasilkan meliputi:

1. Prototipe aplikasi web Nyumbangin yang dapat dijalankan pada lingkungan pengembangan.
2. Dokumen desain arsitektur, use case, flowchart, dan schema model data.
3. Spesifikasi endpoint inti, termasuk leaderboard global.
4. Hasil pengujian fungsional serta evaluasi ketercapaian kebutuhan.

1.5 Sistematika Penulisan

Sistematika penulisan analisis ini disusun dalam 5 bab dan bagian akhir terdapat daftar pustaka dan lampiran. Di mana pada setiap bab tersebut akan dibagi lagi menjadi sub-bab yang akan dibahas secara terperinci. Berikut merupakan sistematika penulisan dan keterangan singkatnya:

1. Bab I Pendahuluan

Berisi latar belakang, identifikasi masalah, tujuan, ruang lingkup, batasan, dan sistematika penulisan. Bab ini juga menjelaskan alasan pemilihan teknologi seperti Next.js, Node.js, MongoDB, JWT, serta integrasi Oauth Google.

2. Bab II Landasan Teori

Menguraikan teori dan konsep pendukung seperti arsitektur web modern, SSR/SPA, autentikasi JWT, OAuth 2.0 dan OpenID Connect (Google), API REST, basis data NoSQL, serta UML.

3. Bab III Metode Penelitian

Menjelaskan metode analisis kebutuhan, perancangan, pemodelan, dan pengujian.

4. Bab IV Hasil dan Pembahasan

Menyajikan hasil implementasi prototipe, struktur proyek, model data, endpoint, antarmuka pengguna, hasil pengujian, serta evaluasi sesuai ruang lingkup.

5. Bab V Kesimpulan dan Saran

Berisi kesimpulan akhir dan rekomendasi pengembangan lebih lanjut.

BAB 2

Landasan Teori

2.1 Platform Donasi Digital

Platform donasi digital merupakan pengembangan dari teknologi platform berbasis internet yang memfasilitasi interaksi antara berbagai pihak untuk tujuan sosial dan filantropi. Platform digital didefinisikan sebagai seperangkat komponen teknologi yang menyediakan fungsi inti bagi suatu sistem dan menjadi fondasi bagi pengembangan layanan pelengkap di atasnya [39, 62]. Secara konseptual, platform ini beroperasi sebagai *two-sided market* yang mempertemukan kelompok pengguna berbeda namun saling bergantung, seperti donatur dan penerima manfaat, di mana nilai platform tercipta dari interaksi antar pengguna tersebut [25].

Dalam konteks filantropi di Indonesia, platform digital digunakan sebagai alternatif lembaga amil konvensional dengan menawarkan kemudahan akses, transparansi, dan kecepatan distribusi dana untuk zakat, infaq, sedekah, dan wakaf [17, 49]. Perkembangan ini sejalan dengan meningkatnya kepercayaan masyarakat terhadap sistem donasi daring serta pergeseran perilaku filantropi ke media digital [50].

Dalam lingkup penggalangan dana massal, platform donation-based crowdfunding didefinisikan secara spesifik sebagai perantara teknologi yang digunakan oleh penggalang dana untuk mencocokkan donasi dengan tujuan para donatur. Tujuan yang dipertemukan dalam platform ini tidak bersifat moneter, melainkan pemenuhan kebutuhan psikologis donatur [14]. Dalam ekosistem ini, penyedia situs web berperan menyediakan layanan web dan membangun sistem yang mendukung pemilik proyek untuk mempresentasikan kampanye mereka kepada calon pendukung guna menggalang donasi [14, 40].

2.2 Arsitektur Aplikasi Web Modern

Bagian ini membahas konsep dasar arsitektur aplikasi web modern yang menjadi landasan dalam memahami cara kerja sistem berbasis web [18]. Pemahaman mengenai pola komunikasi serta pembagian lapisan dalam aplikasi diperlukan untuk menjelaskan bagaimana komponen penyusun sistem saling berinteraksi dan menjalankan fungsinya. Oleh karena itu, pembahasan berikut difokuskan pada model client-server dan arsitektur three-tier/N-tier sebagai struktur arsitektural yang umum digunakan dalam pengembangan aplikasi web masa kini.

2.2.1 Arsitektur Three-Tier/N-Tier

Arsitektur three-tier (tiga lapisan) atau N-tier merupakan sebuah model arsitektur perangkat lunak yang membagi fungsionalitas aplikasi menjadi tiga lapisan logis dan fisik yang berbeda untuk meningkatkan skalabilitas dan keandalan sistem [53, 56]. Tiga lapisan utama tersebut terdiri dari: lapisan presentasi (*presentation tier*), yang berinteraksi langsung dengan pengguna; lapisan aplikasi (*application tier* atau *business logic tier*), yang menangani pemrosesan data dan logika bisnis inti; dan lapisan data (*data tier*), yang bertanggung jawab atas penyimpanan dan manajemen basis data [53]. Pemisahan fungsionalitas ini memungkinkan setiap lapisan dikelola dan dikembangkan secara *independent*, menjadi arsitektur ini pilihan yang efektif untuk sistem yang memerlukan ketersediaan tinggi (*high availability*), seperti pada kasus penerapan LMS Moodle [51].

2.2.2 Konsep Client-Server

Client-server merupakan model perangkat lunak yang memungkinkan sumber daya dan permintaan layanan dipenuhi melalui jaringan, di mana komputer klien akan meminta layanan dan server akan menerima, memproses, serta memberikan respons yang sesuai [57, 19]. Komunikasi antara klien dan server difasilitasi melalui protokol standar seperti HTTP, FTP, dan SMTP [55].

2.3 REST API dan Protokol HTTP

REST API dipahami sebagai pendekatan arsitektur web service yang memanfaatkan prinsip *Representational State Transfer* [13, 37]. REST menekankan penggunaan URI standar untuk mengidentifikasi resource, komunikasi stateless, serta penerapan uniform interface. HTTP berperan sebagai protokol utama yang digunakan sebagai

standar komunikasi dengan metode GET, POST, PUT, dan DELETE [2, 37].

2.4 Autentikasi dan Otorisasi

Bagian ini membahas konsep dasar autentikasi dan otorisasi yang menjadi fondasi penting dalam pengamanan aplikasi berbasis web. Mekanisme pengenalan identitas pengguna dan pemberian hak akses harus dirancang secara tepat agar interaksi antar sistem tetap aman, terukur, dan sesuai dengan tingkat kewenangan yang dibutuhkan. Oleh karena itu, pembahasan berikut mencakup OAuth 2.0 sebagai protokol delegasi akses, OpenID Connect sebagai lapisan identitas, JSON Web Token (JWT) sebagai format token yang umum digunakan, skema Bearer Token yang banyak diadopsi dalam komunikasi API, serta prinsip-prinsip keamanan API yang memastikan perlindungan terhadap ancaman dan penyalahgunaan akses.

2.4.1 OAuth 2.0

OAuth 2.0 didefinisikan sebagai framework otorisasi yang memungkinkan suatu aplikasi memperoleh akses terbatas ke resource yang dilindungi tanpa harus menyimpan kredensial pengguna secara langsung [27]. OAuth 2.0 menyediakan seperangkat authorization server [31]. Framework ini dirancang untuk mendukung berbagai konteks-mulai dari aplikasi web, single-page apps, hingga aplikasi mobile-dengan cara memberikan fleksibilitas pada mekanisme autentikasi dan otorisasi yang aman di antara berbagai jenis klien[59].

2.4.2 OpenID Connect

OpenID Connect (OIDC) merupakan sebuah protokol yang mapan yang digunakan secara luas dalam manajemen identitas terfederasi (federated identity management). Protokol ini berfungsi sebagai dasar bagi otentikasi dan sistem Masuk Tunggal (Single Sign-On atau SSO), yang memungkinkan klien untuk memverifikasi identitas pengguna akhir berdasarkan otentikasi yang dilakukan oleh Penyedia Identitas (Identity Provider)[22, 45]. Dibangun di atas kerangka kerja otorisasi OAuth 2.0, kegunaan OIDC meluas hingga ke infrastruktur kompleks, seperti memfasilitasi akses Secure Shell (SSH) pada pengaturan terfederasi dengan menggunakan token akses OIDC untuk otentikasi pengguna pada server jarak jauh[21].

2.4.3 JWT

JSON Web Token (JWT) merupakan sebuah standar terbuka yang didasarkan pada RFC 7519, yang digunakan secara luas sebagai mekanisme standar untuk otentikasi dan otorisasi pengguna pada layanan web. Standar ini tidak hanya populer untuk mengamankan transmisi data dan otentikasi pada RESTful API, tetapi juga dapat diperluas untuk meningkatkan keamanan dengan menyimpan informasi historis perilaku pengguna, seperti konsistensi alamat IP dan jenis user agent [10, 33]. Sementara itu, JWT secara fundamental adalah format token yang memfasilitasi transmisi data yang ringkas dan aman antara pihak-pihak yang berkepentingan sebagai objek JSON, yang menjadikannya mekanisme otentikasi yang penting dalam implementasi berbagai aplikasi modern [29].

2.4.4 Skema Bearer Token

Skema Bearer Token merupakan mekanisme autentikasi pada OAuth 2.0 di mana klien cukup menyertakan token pada header (Authorization: Bearer <token>) untuk memperoleh akses ke resource yang dilindungi [27]. Karena token ini bersifat bearer, siapa pun yang memiliki token dapat menggunakannya tanpa verifikasi tambahan, sehingga membuat keamanan transport menjadi aspek kritis. Penelitian terbaru menyoroti bahwa risiko pencurian token dapat diminimalkan melalui penggunaan kalal terenskripsi, pembatasan masa hidup token, serta validasi ketat pada sisi server [9]. Selain itu, praktik modern juga menekankan pentingnya menghindari pengiriman token melalui URL dan memastikan proses otorisasi mengikuti pedoman keamanan OAuth 2.0 [54].

2.4.5 Keamanan API

Keamanan API merupakan aspek kritis karena API sering menjadi target serangan [3]. Banyak celah keamanan muncul akibat pengelolaan asset API yang lemah, API lama yang tidak terinventarisasi, serta kerentanan pada alur data dan logika bisnis [43]. Selain itu, meningkatnya kompleksitas arsitektur RESTful dan GraphQL memperluas permukaan serangan, termasuk risiko seperti information leakage, unauthorized access, dan eksploitasi validasi input yang tidak memadai [61]

Untuk mengatasi ancaman tersebut, mekanisme keamanan API membutuhkan pendekatan berlapis yang mencakup autentikasi kuas berbasis OAuth/JWT, penggunaan HTTPS/TLS untuk mengamankan transmisi data, serta manajemen hak akses yang detail guna mencegah penyalahgunaan kredensial [61]. Pentingnya teknik

seperti asset discovery, traffic auditing, dan analisis alur data untuk mengidentifikasi API tersembunyi dan aktivitas mencurigakan [43]. Di samping itu, penggunaan API Gateaway dapat membantu menerapkan pembatasan trafik, filtrasi permintaan, dan perlindungan terhadap serangan seperti DDoS, sehingga API tetap terawasi dan terlindungi secara konsisten.

2.5 Database NoSQL (MongoDB)

Basis data dokumen NoSQL (*Not Only SQL*) muncul sebagai alternatif yang signifikan terhadap basis data relasional tradisional yang sering memiliki batasan ketat pada struktur data dan relasi, sehingga kurang efisien untuk menangani volume data yang sangat besar (*huge database*) [11]. NoSQL document database mengatasi masalah ini dengan menyediakan kemampuan untuk menyimpan dan mengelola data dalam format dokumen, sehingga dapat menampung data yang tidak terstruktur, semi-struktur, maupun terstruktur [12]. Keunggulan utama NoSQL, khususnya jenis berorientasi dokumen seperti MongoDB, terletak pada fleksibilitas dan skalabilitas horizontal yang tinggi, menjadikannya pilihan esensial ketika skema data yang dinamis tidak sesuai dengan kebutuhan basis data realisonal [11].

Secara opsional, basis data dokumen NoSQL menyimpan dalam bentuk dokumen. Meskipun memiliki perbedaan dalam beberapa aspek, MongoDB, Couchbase, dan CouchDB adalah contoh utama dari basis data dokumen yang terkenal [12]. Sebagai contoh MongoDB merupakan basis data berorientasi dokumen, crossplatform, yang menawarkan kinjera tinggi, ketersediaan tinggi, dan skalabilitas yang sederhana. Basis data ini menggunakan MongoDB Query Language (MQL) yang dirancang untuk kemudahan penggunaan oleh pengembang [11]. Oleh karena itu, basis data dokumen NoSQL menjadi solusi penting untuk aplikasi padat data, memastikan penyimpanan big data dan kinerja kueri yang baik.

2.6 Flowchart Sistem

Flowchart merupakan salah satu model yang paling mendasar dan penting dalam perancangan sistem informasi, di mana ia berfungsi untuk mendesain dan menyusun alur dokumen serta memvisualisasikan prosedur atau tahapan proses secara sistematis. Secara umum, flowchart memiliki aplikasi yang luas di berbagai bidang seperti pengembangan perangkat lunak, desain teknik, dan eksperimen ilmiah [46]. Struktur data flowchart tradisional sering kali didasarkan pada *adjacency list*, *cross-linked list*, atau *adjacency matrix* dari struktur graf, yang didasari fakta bahwa setiap dua

node dapat memiliki hubungan koneksi [46]. Namun, terlepas dari kompleksitas penyimpanan datanya, flowchart tetap menjadi alat fundamental yang menyediakan representasi visual dari urutan dan hubungan logis dalam suatu sistem [36, 30].

2.7 Unified Modeling Language (UML)

Unified Modeling Language (UML) didefinisikan sebagai sebuah bahasa pemodelan standar yang digunakan untuk merancang dan mendokumentasikan sistem berorientasi objek. Sebagai bahasa standar, UML menyediakan seperangkat notasi grafis yang komprehensif untuk memvisualisasikan, memspesifikasikan, membangun, dan mendokumentasikan artefak dalam sistem perangkat. Tujuan utama penggunaan UML adalah untuk memperjelas model yang tidak konsisten dan mengurangi ambiguitas selama proses pengembangan perangkat lunak [7]. UML membantu memvisualkan, menspesifikasikan, dan mendokumentasikan desain sistem secara grafis [42].

Dengan menggunakan diagram-diagram yang berbeda, seperti Use Case Diagram dan Activity Diagram, UML membantu pengembang dalam memodelkan interaksi, struktur, dan perilaku sistem [4]. Penerapan UML sangat krusial dalam siklus hidup pengembangan sistem (*System Development Life Cycle* atau SDLC) karena membantu memastikan konsistensi model dan mempermudah komunikasi antara pihak-pihak yang terlibat dalam proyek [28].

2.7.1 Use Case Diagram

Use Case adalah suatu diagram fundamental yang umum diajarkan dalam ilmu komputer dan rekayasa perangkat lunak. Diagram ini berfungsi sebagai representasi visual dari fungsionalitas sistem dari sudut pandang pengguna. Meskipun definisinya tampak sederhana, penilaian terhadap diagram use case sering kali menjadi hambatan dalam proses pembelajaran, terutama karena dua masalah utama: masalah interpersonal (tidak adanya standar penilaian di antara para pengajar) [23] dan masalah intrapersonal (inkonsistensi seorang pengajar saat menilai banyak diagram) [16, 5, 44].

2.7.2 Activity Diagram

Activity Diagram adalah salah satu diagram perilaku yang tersedia dalam Unified Model Language (UML) yang digunakan untuk memodelkan alur kontrol dan alur data dalam suatu sistem [38]. Diagram ini secara visual merepresentasikan

Langkah-langkah, keputusan, dan urutan tindakan yang diperlukan untuk menyelesaikan suatu proses atau kegiatan bisnis tertentu [42]. Dalam konteks pemodelan sistem, Activity Diagram sangat berguna untuk memvisualisasikan bagaimana berbagai kegiatan saling terkait dan bergantung satu sama lain [24, 35].

2.7.3 Sequence Diagram

Sequence Diagram adalah diagram UML yang paling umum kedua, digunakan untuk merepresentasikan interaksi objek dan pertukaran pesan antar objek tersebut seiring berjalannya waktu [42]. Diagram ini secara visual menunjukkan bagaimana peristiwa atau aktivitas yang ada dalam sebuah use case dipetakan menjadi operasi-operasi dari kelas objek yang ada pada Class Diagram [1].

2.7.4 Class Diagram

Class Diagram merupakan salah satu diagram Unified Modeling Language (UML) yang paling umum digunakan dalam Pendidikan dan pengembangan perangkat lunak berorientasi objek [42]. Fungsi utama dari Class Diagram adalah untuk merepresentasikan kelas-kelas dalam sistem perangkat lunak dan hubungan yang terjalin antar kelas-kelas tersebut [16].

Entity-Relationship Diagram (ERD) digunakan untuk memodelkan struktur konseptual basis data [32]. ERD merepresentasikan entitas, atribut, dan relasi antar entitas [52, 6].

2.8 Peran Fitur Social Sharing dalam Donasi Digital

Kepercayaan merupakan faktor kunci dalam keberhasilan platform donasi digital, mengingat interaksi antara donatur dan penerima dilakukan sepenuhnya secara daring tanpa kontak langsung [60]. Tingkat kepercayaan ini sangat dipengaruhi oleh persepsi transparansi dan kredibilitas sistem yang disediakan oleh platform donasi digital [20]. Oleh karena itu, platform donasi perlu menyediakan mekanisme yang mampu memperkuat transparansi tersebut.

Salah satu mekanisme yang dapat diterapkan adalah fitur social sharing, yang memungkinkan aktivitas donasi dibagikan ke media sosial. Penerapan fitur social sharing berperan sebagai *social proof*, di mana visibilitas partisipasi pengguna lain dapat meningkatkan persepsi keandalan dan legitimasi platform [34]. Peningkatan kepercayaan dan persepsi positif tersebut pada akhirnya berdampak pada meningkatnya

niat dan partisipasi pengguna dalam kegiatan donasi digital [26].

Berdasarkan temuan tersebut, fitur social sharing pada aplikasi Nyumbangin dirancang sebagai fitur pendukung yang memungkinkan pengguna membagikan aktivitas donasi ke media sosial. Fitur ini diharapkan dapat memperkuat kepercayaan pengguna serta mendorong partisipasi donasi secara lebih luas.

BAB 3

Metode Penelitian

3.1 Metode Pengembangan Sistem

Pengembangan platform Nyumbangin menggunakan pendekatan Agile, yang menekankan proses pengembangan sistem secara bertahap, adaptif, dan berulang. Pendekatan Agile dipilih karena sesuai dengan karakteristik proyek berskala kecil hingga menengah, serta memungkinkan penyesuaian fitur berdasarkan hasil evaluasi pada setiap tahap pengembangan.

Pendekatan ini memungkinkan sistem dikembangkan secara inkremental, di mana setiap iterasi menghasilkan fungsionalitas yang dapat diuji dan dievaluasi sebelum melanjutkan ke iterasi berikutnya. Dengan demikian, risiko kesalahan desain dan implementasi dapat diminimalkan sejak tahap awal.

3.1.1 Konsep Agile Development

Agile Development merupakan pendekatan pengembangan perangkat lunak yang berfokus pada fleksibilitas, kolaborasi, dan kemampuan beradaptasi terhadap perubahan kebutuhan [8]. Berbeda dengan metode linear seperti waterfall, Agile memungkinkan perubahan kebutuhan terjadi selama proses pengembangan tanpa harus mengulang seluruh tahapan dari awal.

Dalam konteks pengembangan platform Nyumbangin, Agile digunakan sebagai kerangka kerja konseptual untuk mengelola proses pelaksanaan fitur secara bertahap, mulai dari analisis kebutuhan dasar, implementasi modul inti, hingga pengujian dan evaluasi sistem. Pendekatan ini mendukung pengembangan sistem yang responsif terhadap kebutuhan fungsional dan teknis yang berkembang selama proyek berlangsung.

3.1.2 Alur Iterasi Pengembangan

Proses pengembangan sistem dilakukan melalui beberapa siklus iterasi yang masing-masing mencakup tahapan berikut:

1. Perencanaan Iterasi

Penentuan fitur yang akan dikembangkan berdasarkan prioritas kebutuhan sistem.

2. Implementasi Fitur

Pengembangan modul atau fungsi tertentu sesuai dengan hasil perencanaan iterasi.

3. Pengujian Fungsional

Pengujian terhadap fitur yang telah dikembangkan untuk memastikan kesesuaian dengan kebutuhan.

4. Evaluasi dan Penyempurnaan

Evaluasi hasil iterasi dan perbaikan terhadap kurangnya sebelum melanjutkan ke iterasi berikutnya.

Setiap iterasi menghasilkan peningkatan fungsional sistem yang dapat langsung diuji, sehingga kemajuan proyek dapat dipantau secara berkelanjutan.

3.1.3 Penerapan Agile pada Proyek Nyumbangin

Penerapan pendekatan Agile pada proyek Nyumbangin dilakukan dengan membagi pengembangan sistem ke dalam beberapa iterasi utama. Iterasi awal difokuskan pada pembangunan fitur inti, seperti autentikasi kreator, pencatatan donasi, dan integrasi sistem pembayaran. Iterasi berikutnya mencakup pengembangan fitur pendukung, seperti notifikasi real-time melalui overlay, leaderboard donatur, serta mekanisme payout bagi kreator.

Pada setiap iterasi, fitur yang telah diimplementasikan langsung diuji menggunakan scenario pengujian fungsional untuk memastikan alur donasi, pembayaran, dan pencairan dana berjalan dengan benar. Hasil pengujian digunakan sebagai dasar evaluasi untuk menentukan perbaikan atau pengembangan fitur pada iterasi selanjutnya.

Pendekatan ini memungkinkan pengembangan aplikasi dilakukan secara terstruktur namun tetap fleksibel, sehingga sistem dapat berkembang secara bertahap hingga memenuhi kebutuhan fungsional yang telah ditetapkan.

3.2 Analisis Kebutuhan

3.2.1 Sumber Kebutuhan

Analisis kebutuhan sistem dilakukan melalui tiga pendekatan utama. Pertama, observasi terhadap platform donasi digital untuk mengidentifikasi pola umum, seperti kebutuhan transparansi transaksi, tampilan notifikasi real-time, dan mekanisme payout yang akuntabel. Kedua, studi pustaka terkait autentikasi modern (OAuth 2.0, JWT), keamanan API, serta pola desain sistem web yang relevan dengan karakteristik aplikasi donasi. Ketiga, analisis artefak kode dan struktur API yang telah dikembangkan, termasuk model basis data, alur donasi, integrasi Midtrans, serta skrip pemeliharaan yang digunakan untuk verifikasi dan pengarsipan data. Pendekatan ini memastikan kebutuhan sistem dirumuskan berdasarkan konteks teknis dan operasional yang aktual.

3.2.2 Kebutuhan Fungsional

Kebutuhan fungsional mendeskripsikan fitur yang wajib disediakan agar aplikasi donasi dapat berfungsi secara utuh. Fitur tersebut meliputi:

1. Autentikasi Pengguna menggunakan Google OAuth melalui NextAuth, serta validasi akses melalui JWT untuk endpoint sensitif.
2. Validasi Username Kreator, memastikan username unik dan dapat diverifikasi sebelum transaksi dilakukan.
3. Pengelolaan Donasi, mencakup pembuatan transaksi, integrasi Midtrans, pembaruan status melalui webhook, serta penyimpanan konten media share.
4. Notifikasi Real-Time, yang menampilkan donasi terbaru pada overlay kreator untuk keperluan siaran langsung.
5. Leaderboard, baik secara global maupun berdasarkan kreator, untuk menampilkan agregasi donasi.
6. Statistik Kreator, berupa ringkasan donasi berdasarkan periode.
7. Payout, mencakup permintaan penarikan dana, perhitungan biaya platform, dan persetujuan admin.
8. Pengelolaan Data Media Share dan Notifikasi, termasuk pengaturan masa berlaku (TTL) dan keterkaitan dengan transaksi donasi.

3.2.3 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional mencakup karakteristik kualitas sistem, yaitu:

- **Keamanan:** verifikasi token JWT, sanitasi input, pembatasan metode HTTP, dan pemisahan akses berdasarkan peran pengguna.
- **Performa:** optimasi query leaderboard melalui limit dan sorting.
- **Skalabilitas:** rencana pagination serta caching pada proses agregasi data.
- **Reliabilitas:** konsistensi penanganan kesalahan dengan kode status standar (401, 404, 500).
- **Integritas Data:** akurasi perhitungan saldo payout dan pemrosesan donasi berdasarkan statur valid (PAID).

3.3 Perancangan Sistem

3.3.1 Arsitektur Logis

Arsitektur logis sistem terdiri atas empat lapisan utama sebagai berikut:

1. **Lapisan Antarmuka Pengguna (UI Layer)**
Berisi halaman donasi, dashboard kreator, halaman overlay notifikasi, serta antarmuka admin.
2. **Lapisan API (Application Layer)**
Mengelola endpoint untuk donasi, leaderboard, overlay, autentikasi, payout, dan operasi admin melalui mekanisme API Routes di Next.js.
3. **Lapisan Layanan Utilitas (Service Layer)**
Meliputi modul koneksi database, pengelolaan token JWT, serta utilitas untuk validasi dan perhitungan internal.
4. **Lapisan Data (Data Layer)**
Terdiri atas model MongoDB seperti Donation, Creator, MediaShare, Payout, dan Notification.

3.3.2 Arsitektur Fisik

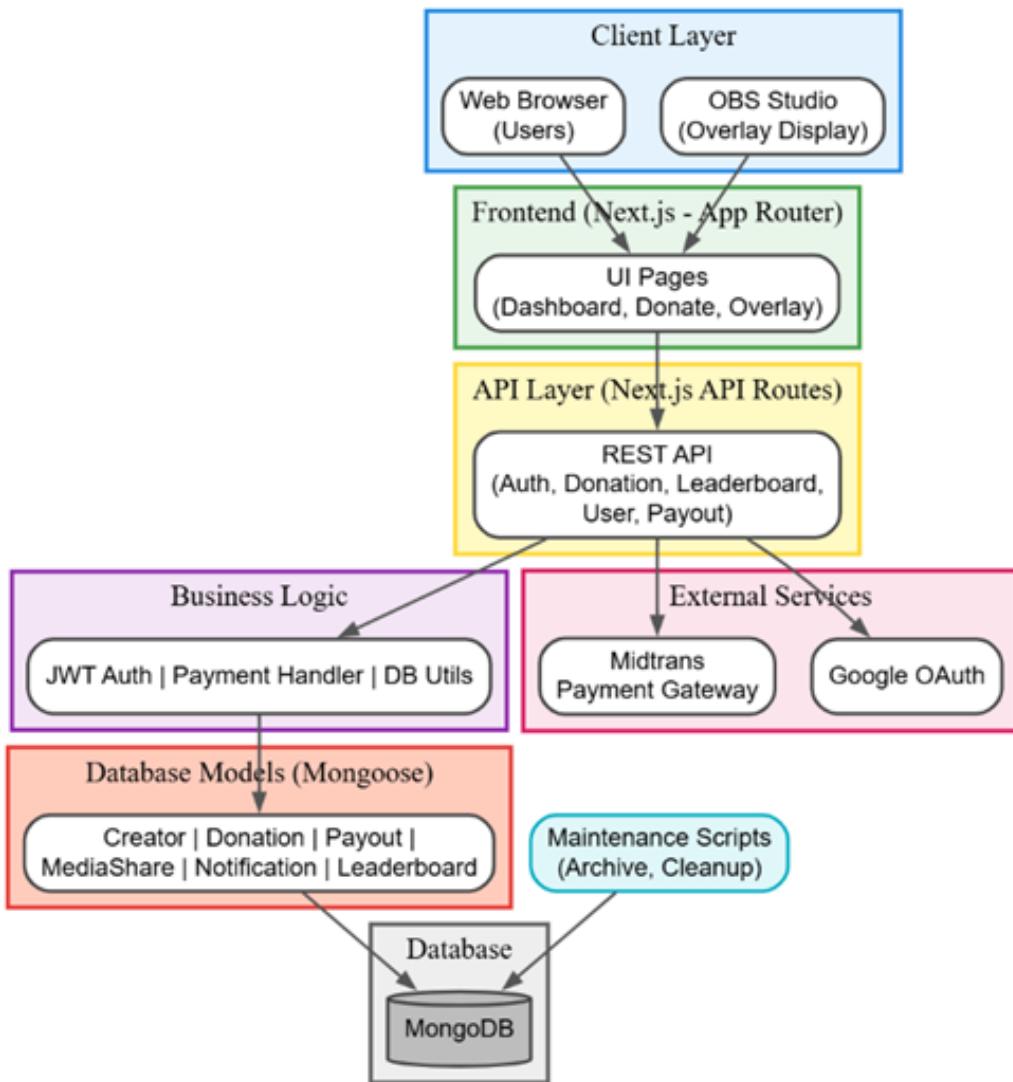
Arsitektur fisik sistem mengikuti pola aplikasi web modern sebagai berikut:

Browser / OBS Overlay → Next.js Runtime (Node.js) → MongoDB → Layanan Eksternal (Google OAuth, Midtrans)

- Next.js menangani logika antarmuka pengguna dan API dalam satu platform.
- MongoDB digunakan sebagai basis data dokumen.
- Midtrans mengelola proses pembayaran melalui mekanisme Snap Token dan webhook.
- Overlay digunakan secara mandiri melalui OBS atau *iframe* untuk menampilkan notifikasi donasi.

3.3.3 Arsitektur Teknologi

Arsitektur teknologi sistem mencakup penggunaan Next.js sebagai framework utama yang menjalankan frontend dan backend melalui API Routes, Node.js sebagai runtime server-side, serta MongoDB sebagai basis data dokumen. Sistem autentikasi menggunakan Google OAuth 2.0 melalui NextAuth dan JWT untuk otorisasi endpoint privat. Mekanisme pembayaran dilakukan dengan Midtrans melalui Snap Token dan Webhook. Selain itu, aplikasi menyediakan overlay web real-time untuk integrasi dengan OBS sebagai tampilan notifikasi donasi. Kombinasi teknologi ini menghasilkan sistem yang modern, responsive, serta siap diintegrasikan dengan berbagai layanan eksternal.



Gambar 3.1: Arsitektur Teknologi

3.3.4 Modul Utama

Sistem dibagi ke dalam beberapa modul utama sebagai berikut:

1. **Authentication Module** (OAuth + JWT)
2. **Donation Module** (pembuatan transaksi, webhook, media share)
3. **Leaderboard Module** (global dan per kreator)
4. **Payout Module** (request, approval, perhitungan fee)
5. **Notification Module** (TTL, event donasi)
6. **Maintenance Module** (arsip donasi, integritas data)

3.3.5 Strategi Desain

Strategi desain menerapkan pola penanganan API yang konsisten meliputi validasi metode HTTP, autentikasi, validasi input, eksekusi query database, dan pengembalian response dalam format JSON. Selain itu, prinsip pemisahan tanggung jawab diterapkan melalui pembagian endpoint berdasarkan peran dan fungsi. Pembatasan data seperti penggunaan limit dan sorting digunakan untuk menghindari overfetching, terutama pada proses pengambilan data leaderboard dan statistik.

3.4 Pemodelan Sistem

Pemodelan sistem dilakukan untuk memberikan representasi visual dan konseptual dari fungsi, alur kerja, serta struktur data yang digunakan dalam aplikasi Nyumbangin ini. Pemodelan ini bertujuan memastikan bahwa kebutuhan fungsional dan non-fungsional yang telah diidentifikasi dapat diterjemahkan ke dalam desain sistem yang jelas, terstruktur, dan mudah diimplementasikan. Diagram-diagram pada bagian ini mencakup model proses, interaksi, dan entitas yang saling berhubungan, sehingga dapat memberikan gambaran menyeluruh mengenai cara sistem beroperasi secara end-to-end.

3.4.1 Use Case



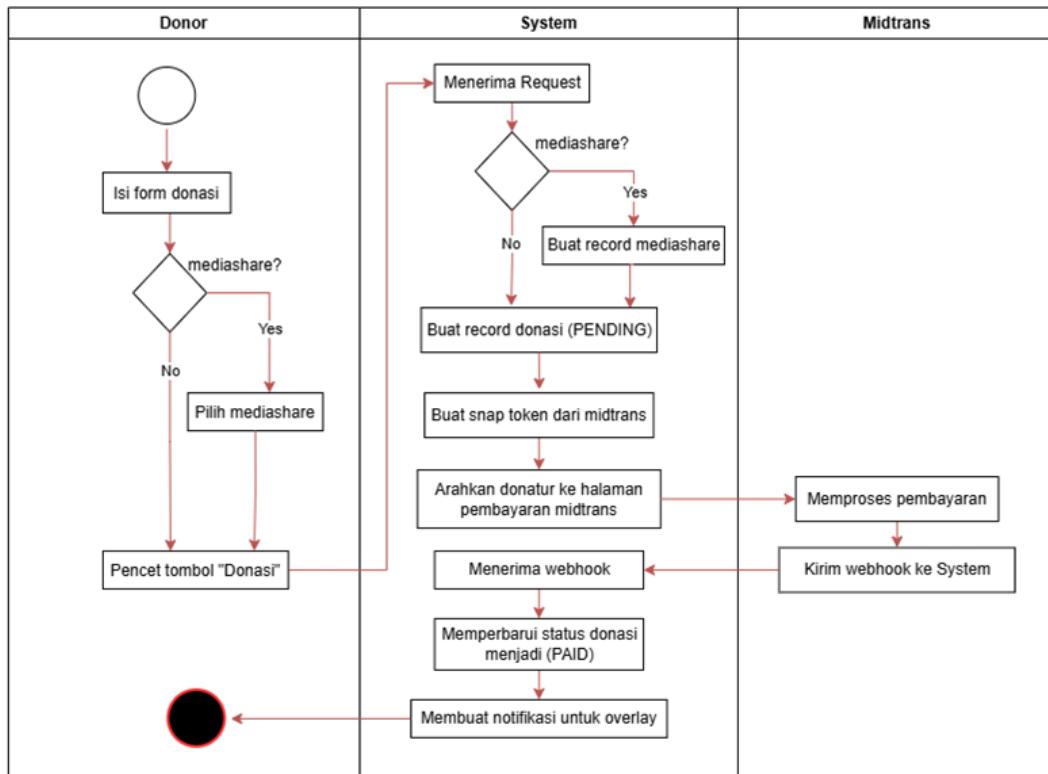
Gambar 3.2: Use Case Diagram

Use case menggambarkan interaksi antara aktor dengan sistem aplikasi Nyumbangin. Diagram ini bertujuan untuk menunjukkan fungsi-fungsi utama yang dapat dilakukan oleh masing-masing aktor sesuai dengan perannya.

Aktor yang terlibat dalam sistem ini yaitu Donatur, Kreator, dan Admin. Selain itu, sistem terintegrasi dengan layanan eksternal seperti Google OAuth untuk proses autentikasi dan Midtrans sebagai gateway pembayaran. Use case ini menggambarkan secara umum cakupan fungsi sistem tanpa menampilkan detail alur proses.

3.4.2 Activity Diagram

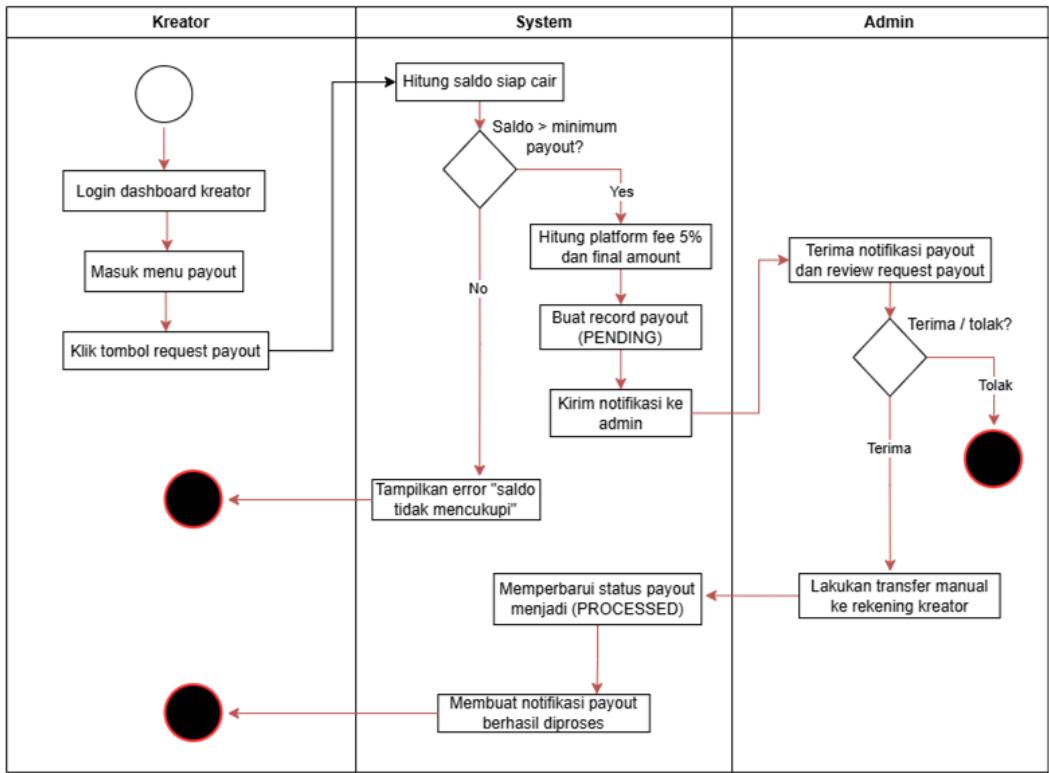
Activity Diagram proses Donasi



Gambar 3.3: Activity Diagram Donasi

Activity diagram proses donasi menggambarkan alur di mana Donor mengisi formulir donasi (dengan atau tanpa media share), kemudian sistem membuat record donasi dengan status PENDING, menghasilkan *snap token*, dan mengarahkan Donor ke Midtrans untuk proses pembayaran. Setelah Midtrans mengirimkan webhook, sistem memperbarui status donasi menjadi PAID dan membuat notifikasi donasi.

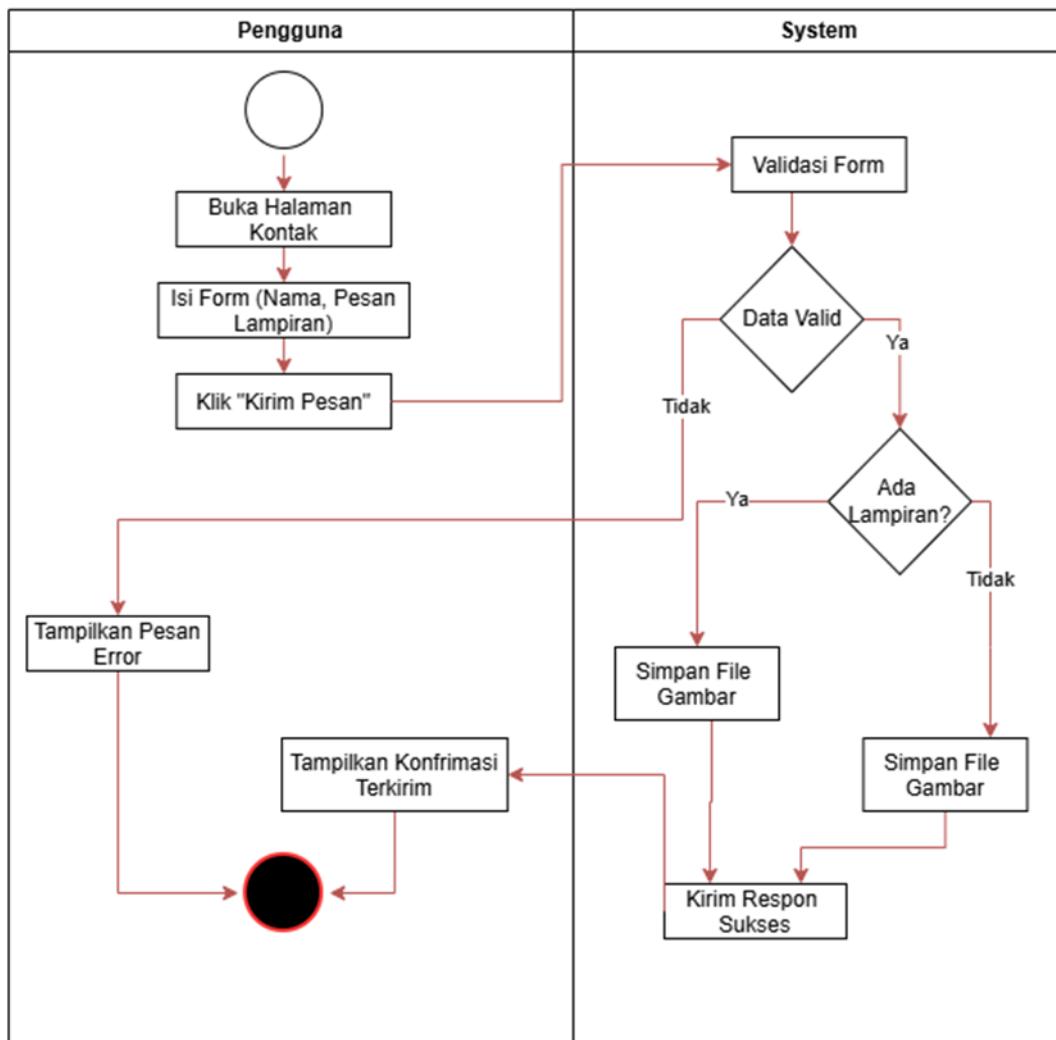
3.4.3 Activity Diagram Proses Payout



Gambar 3.4: Activity Diagram Payout

Activity diagram proses payout dimulai ketika Kreator mengajukan permintaan payout, di mana sistem melakukan pemeriksaan terhadap saldo minimal. Apabila saldo memenuhi syarat, sistem akan menghitung biaya layanan sebesar 5% dan mengubah status payout menjadi PENDING. Selanjutnya, Admin melakukan peninjauan dan transfer dana secara manual, setelah itu sistem memperbarui status payout menjadi PROCESSED.

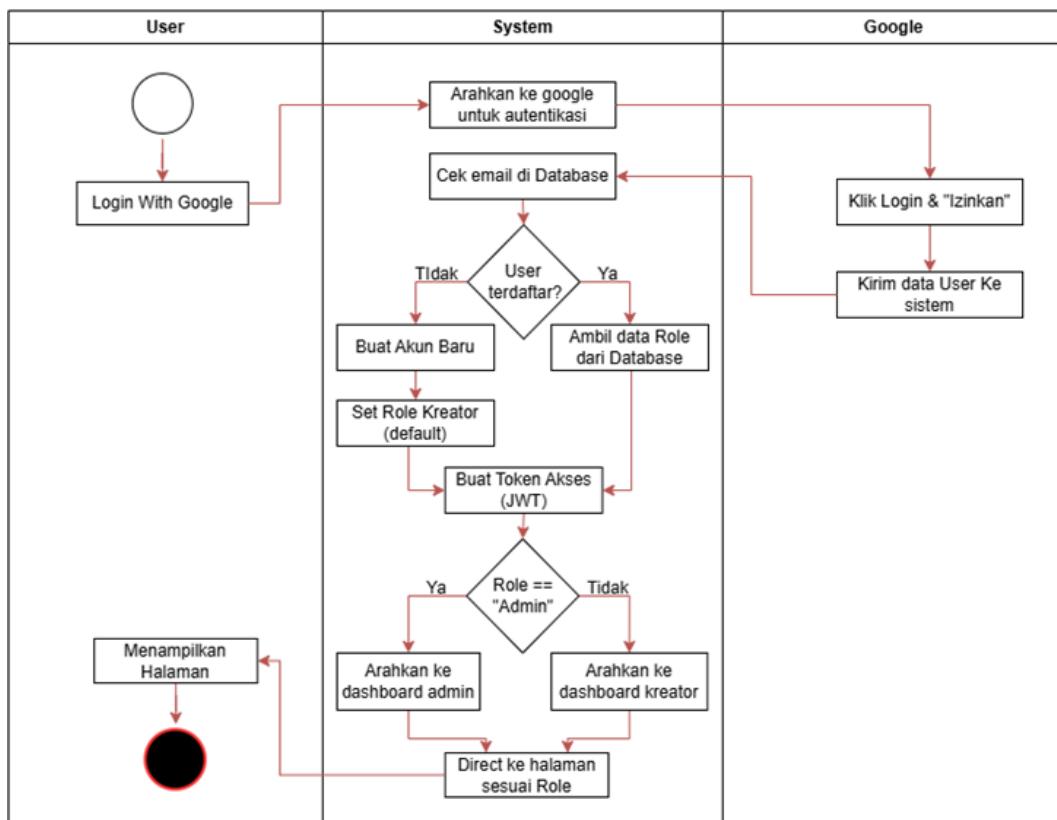
3.4.4 Activity Diagram Feedback



Gambar 3.5: Activity Diagram Feedback

Activity diagram feedback menggambarkan alur proses pengiriman feedback oleh pengguna. Proses dimulai dari pengisian formulir feedback, kemudian sistem melakukan validasi data. Apabila data dinyatakan valid, sistem akan menyimpan feedback dan menampilkan konfirmasi bahwa feedback berhasil dikirim. Sebaliknya, jika data tidak valid, sistem akan menampilkan pesan kesalahan.

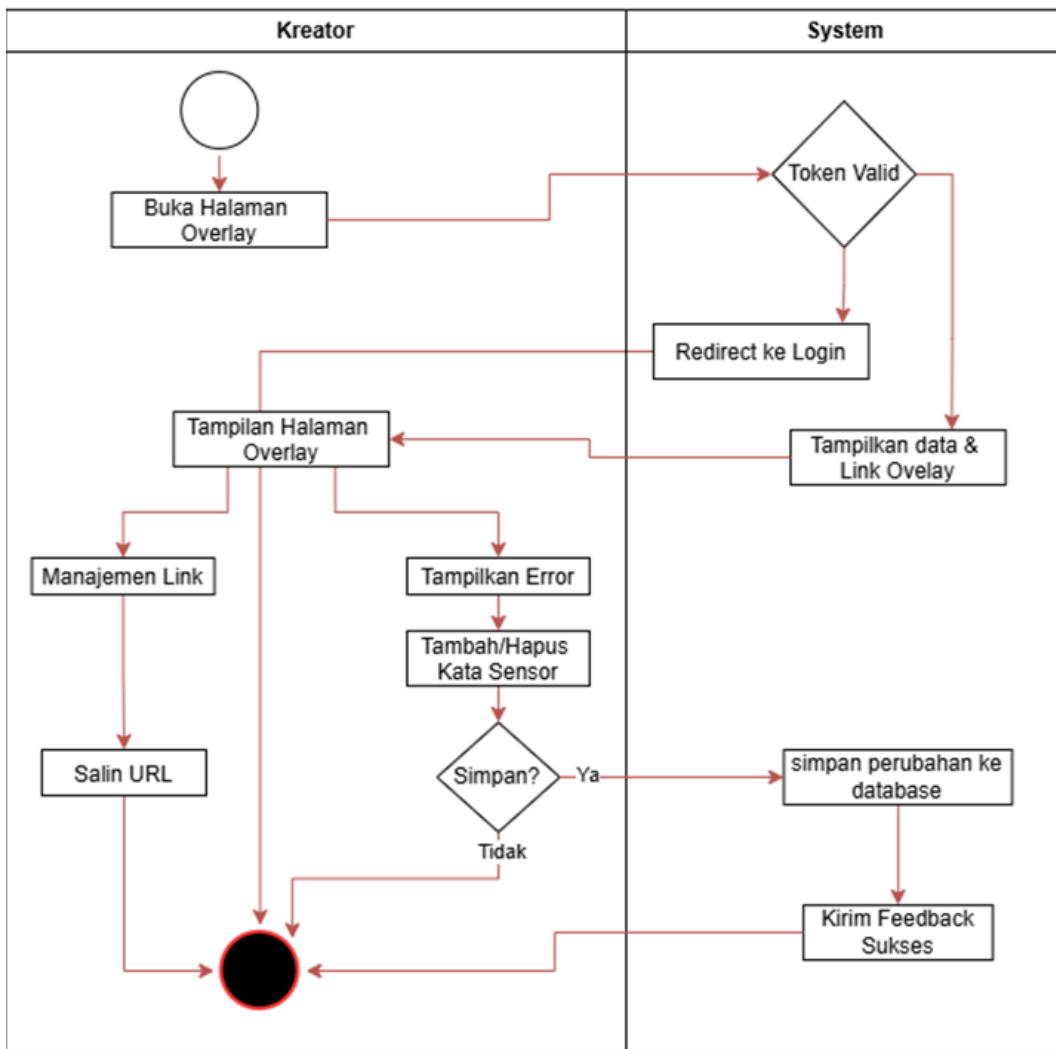
3.4.5 Activity Diagram Login



Gambar 3.6: Activity Diagram Login

Activity diagram login menggambarkan alur autentifikasi pengguna menggunakan Google OAuth. Proses dimulai dari login, autentifikasi ke Google, pengecekan data pengguna, pembuatan token akses, dan pengalihan ke halaman dashboard sesuai dengan peran pengguna.

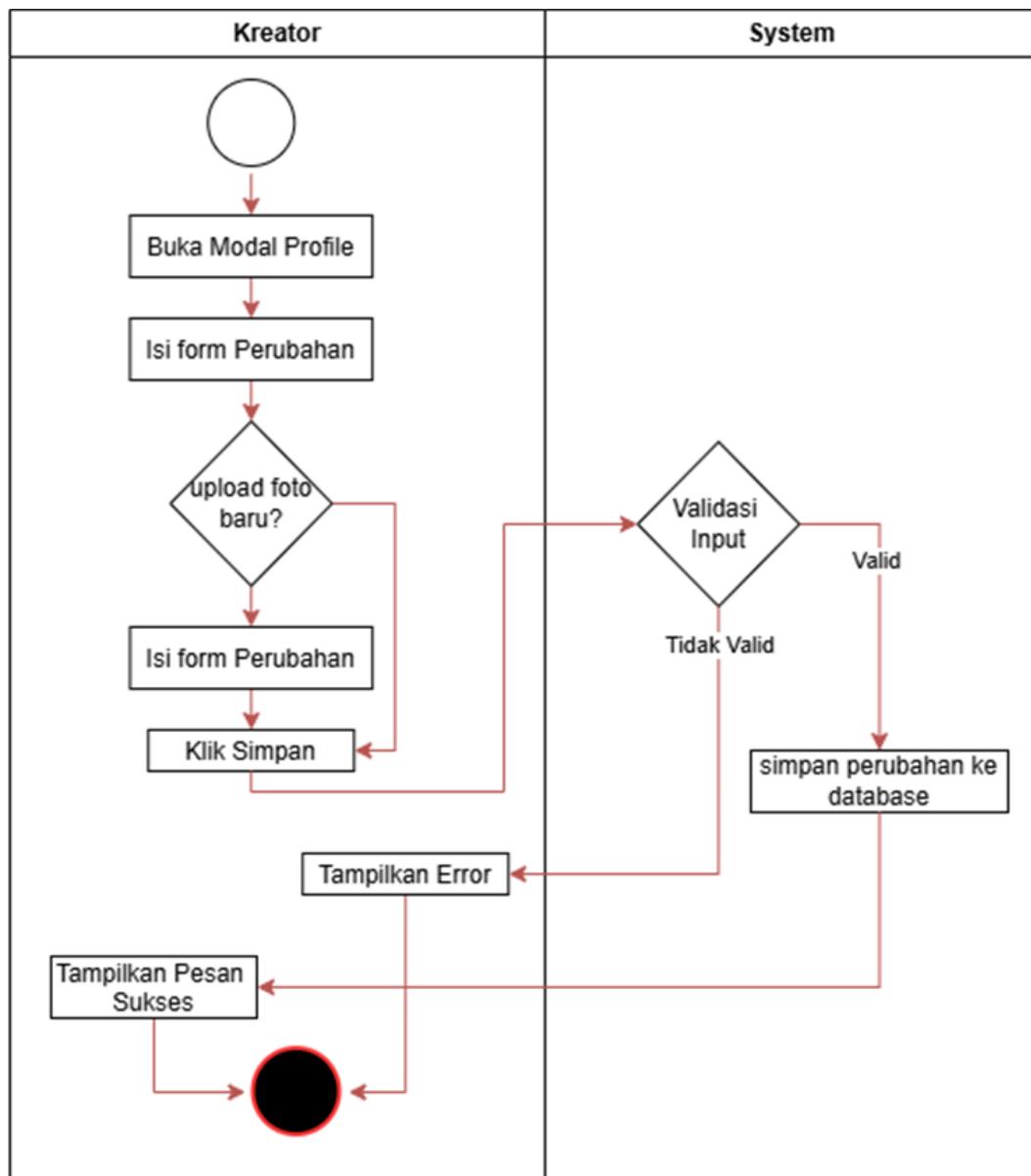
3.4.6 Activity Diagram Kelola Overlay



Gambar 3.7: Activity Diagram Kelola Overlay

Activity diagram kelola overlay menggambarkan alur pengelolaan overlay oleh kreator. Sistem melakukan pengecekan autentikasi sebelum menampilkan halaman overlay. Kreator dapat menambah, mengubah, atau menghapus data overlay. Sistem memvalidasi dan menyimpan perubahan, kemudian memberikan feedback keberhasilan kepada kreator.

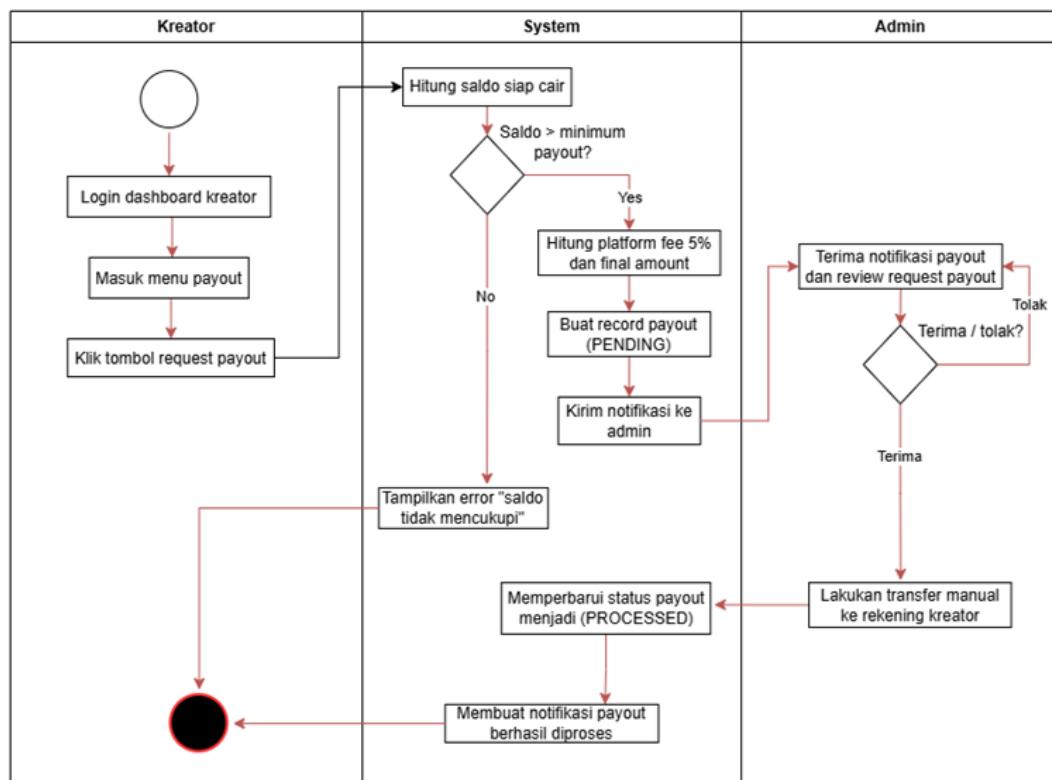
3.4.7 Activity Diagram Kelola Profil



Gambar 3.8: Activity Diagram Kelola Profil

Activity diagram kelola profil menggambarkan proses perubahan data profil oleh kreator. Kreator mengisi form perubahan data dan mengunggah foto profil jika diperlukan. Sistem melakukan validasi input dan menyimpan perubahan ke database. Jika proses berhasil, sistem menampilkan pesan sukses.

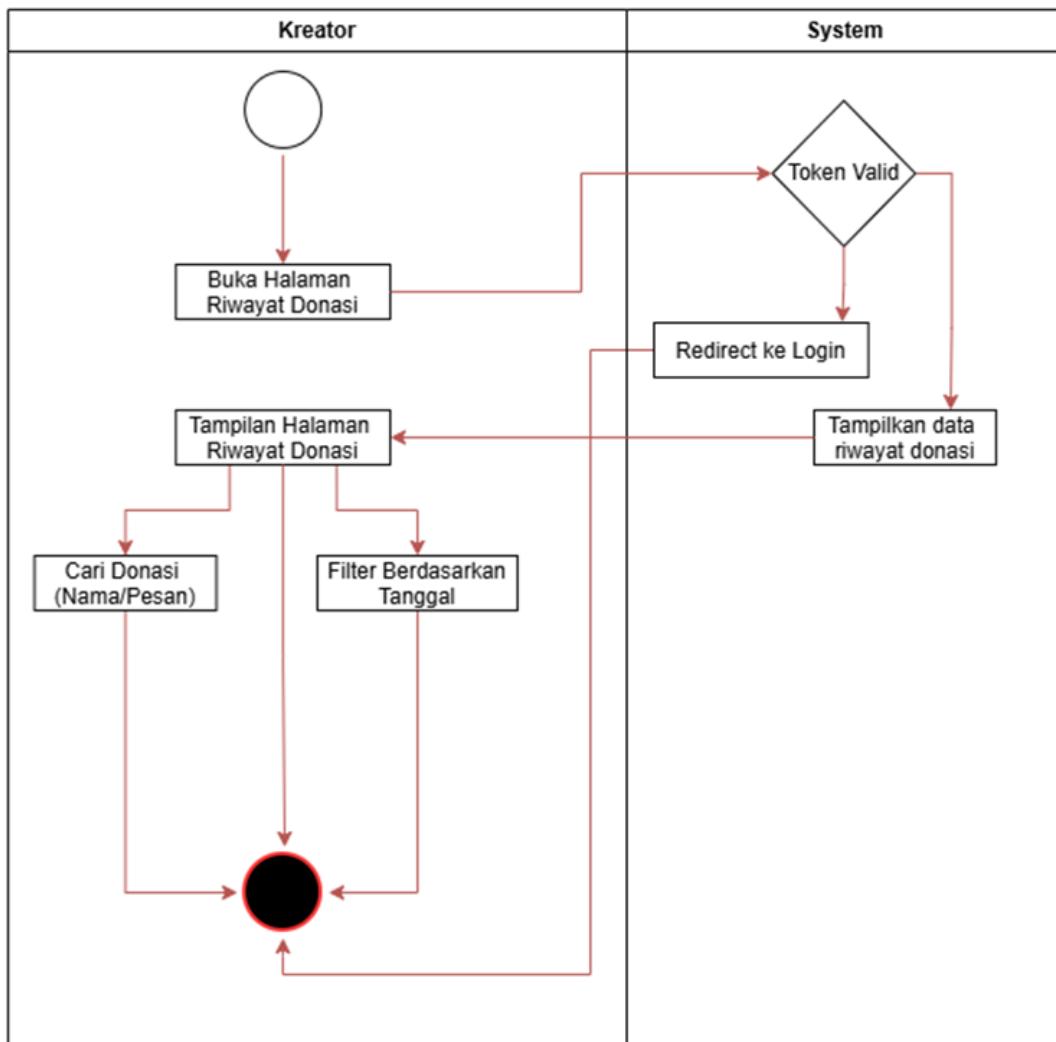
3.4.8 Activity Diagram Request Payout



Gambar 3.9: Activity Diagram Request Payout

Activity diagram request payout menggambarkan alur pencairan dana yang diajukan oleh kreator. Sistem memeriksa saldo dan minimum payout, menyimpan data payout, serta mengirimkan notifikasi ke admin. Admin melakukan proses review, dan sistem memperbarui status payout sesuai dengan keputusan admin.

3.4.9 Activity Diagram Riwayat Donasi

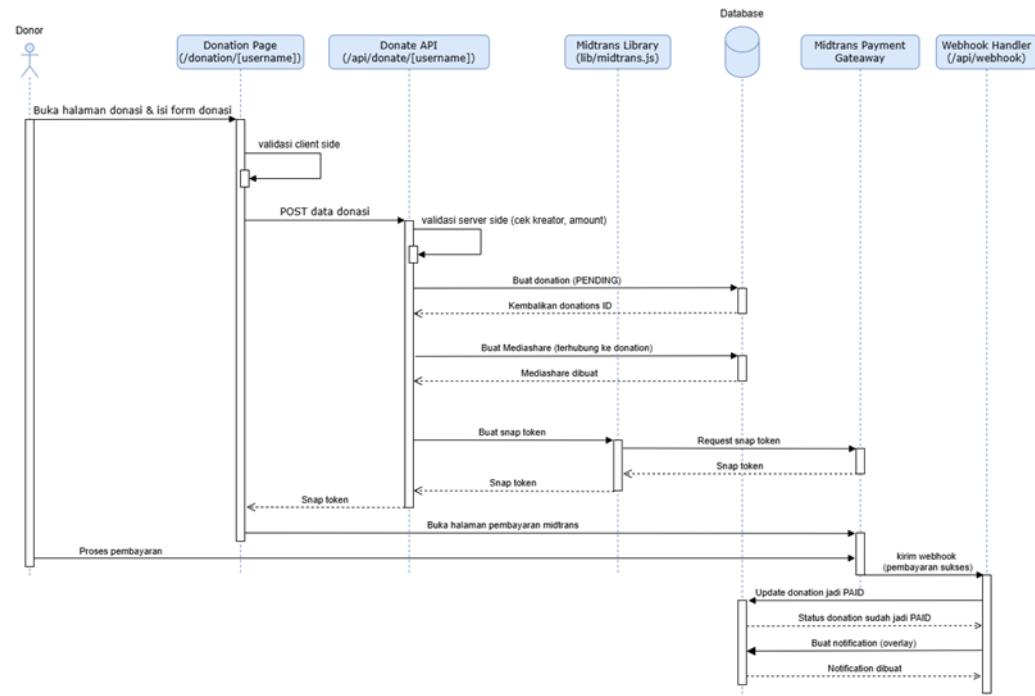


Gambar 3.10: Activity Diagram Riwayat Donasi

Activity diagram riwayat donasi menggambarkan proses penampilan data riwayat donasi. Kreator membuka halaman riwayat donasi, sistem menampilkan data donasi yang tersimpan, dan kreator dapat melakukan pencarian serta melakukan filter data berdasarkan kriteria tertentu.

3.4.10 Sequence Diagram

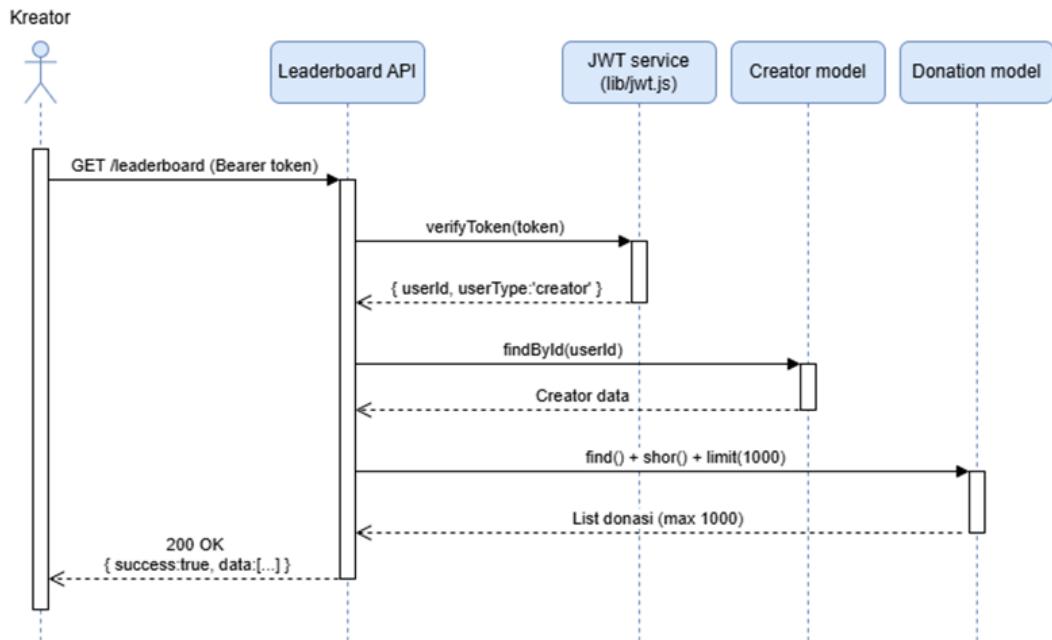
Sequence Diagram Donasi



Gambar 3.11: Sequence Diagram Donasi

Sequence diagram donasi menggambarkan alur proses donasi yang dimulai ketika donor mengisi formulir pada halaman donasi (*Donation Page*) yang kemudian divalidasi dan dikirim ke *Donate API*. Setelah dilakukan validasi di sisi server, data donasi disimpan ke dalam basis data, dan apabila terdapat *youtubeUrl*, sistem juga akan membuat data *MediaShare*. Selanjutnya, API meminta *Snap Token* ke Midtrans yang digunakan untuk membuka halaman pembayaran. Setelah donor menyelesaikan pembayaran dan transaksi dinyatakan berhasil, Midtrans mengirimkan webhook ke server sehingga status donasi diperbarui menjadi *PAID*, dan pada tahap akhir sistem membuat notifikasi overlay.

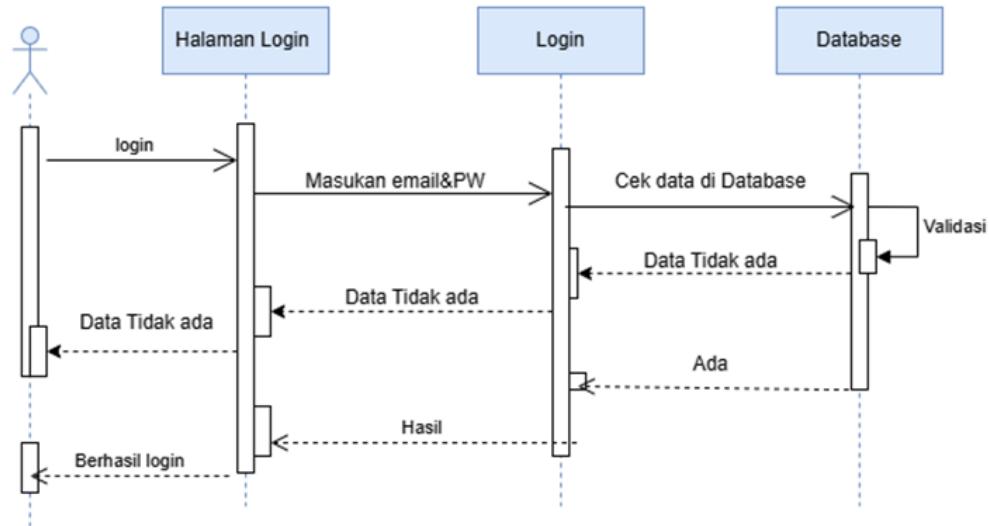
Sequence Diagram Leaderboard



Gambar 3.12: Sequence Diagram Leaderboard

Sequence diagram leaderboard menggambarkan alur pengambilan data leaderboard, di mana client (dashboard kreator) mengirimkan request ke *Leaderboard API* dengan menyertakan token. API melakukan pemeriksaan metode HTTP dan validasi token melalui layanan JWT, kemudian memastikan bahwa pengguna merupakan kreator dan data yang diminta tersedia. Selanjutnya, API mengambil data donasi terbaru dari basis data, memformat respons, dan mengirimkan hasilnya kembali ke client. Apabila terjadi kesalahan seperti token tidak valid, kreator tidak ditemukan, atau metode HTTP tidak sesuai, API akan mengembalikan kode error yang relevan.

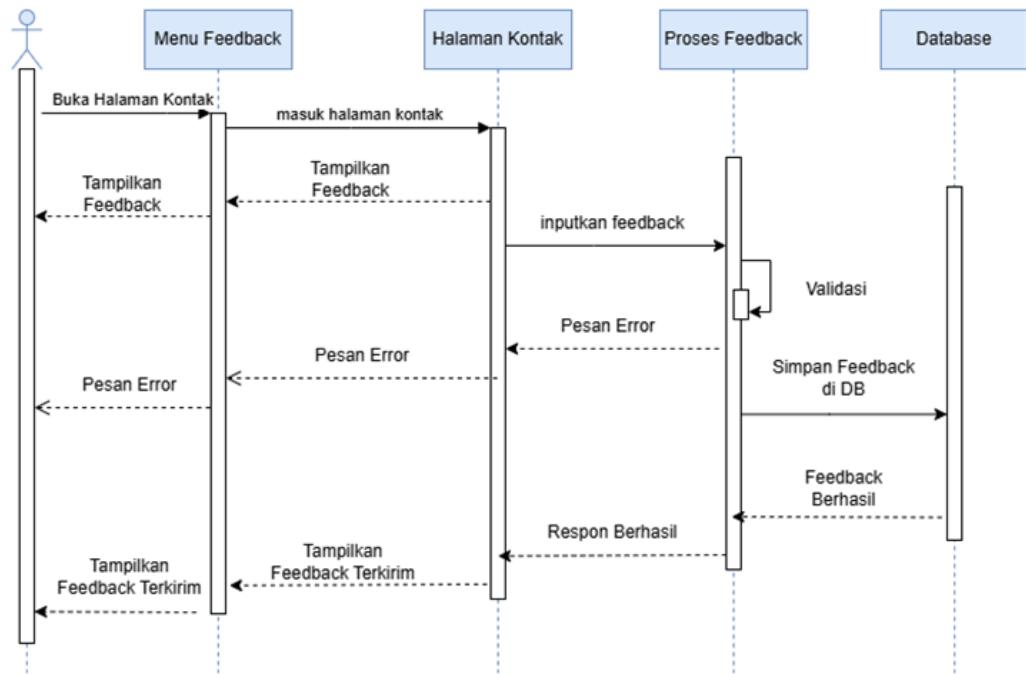
Sequence Diagram Login



Gambar 3.13: Sequence Diagram Login

Sequence diagram login menggambarkan urutan interaksi antara pengguna dan sistem dalam proses autentikasi menggunakan email dan kata sandi. Proses dimulai ketika pengguna membuka halaman login dan memasukkan data email serta kata sandi. Sistem kemudian melakukan pengecekan data ke basis data untuk memverifikasi kecocokan informasi yang dimasukkan. Apabila data tidak sesuai, sistem akan menampilkan pesan kesalahan kepada pengguna. Jika data sesuai, sistem memproses autentikasi dan mengarahkan pengguna ke halaman utama atau dashboard sesuai dengan hak akses yang dimiliki.

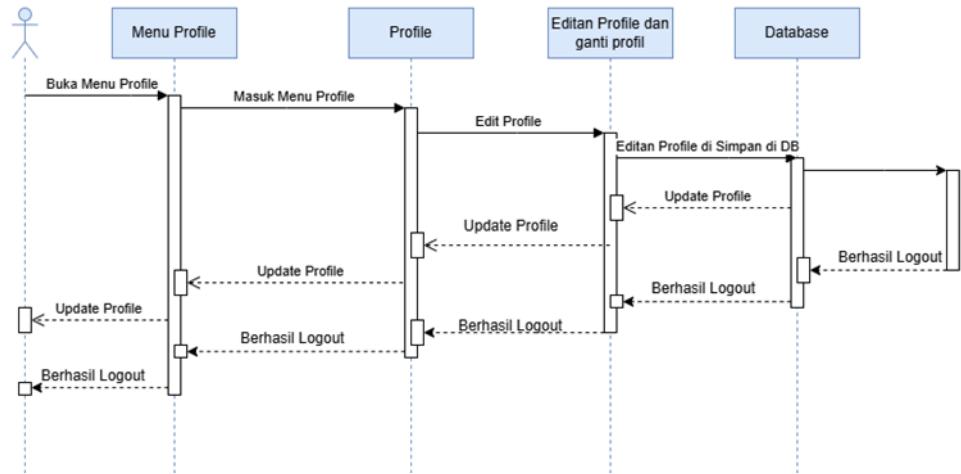
Sequence Diagram Feedback



Gambar 3.14: Sequence Diagram Feedback

Sequence diagram feedback menggambarkan alur pengiriman feedback dari pengguna ke sistem. Proses dimulai ketika pengguna membuka halaman kontak dan mengisi form feedback. Sistem melakukan validasi terhadap data yang dikirimkan. Jika data tidak valid, sistem akan mengirimkan pesan error. Jika data valid, sistem menyimpan feedback ke dalam basis data dan mengirimkan respons bahwa feedback berhasil dikirim.

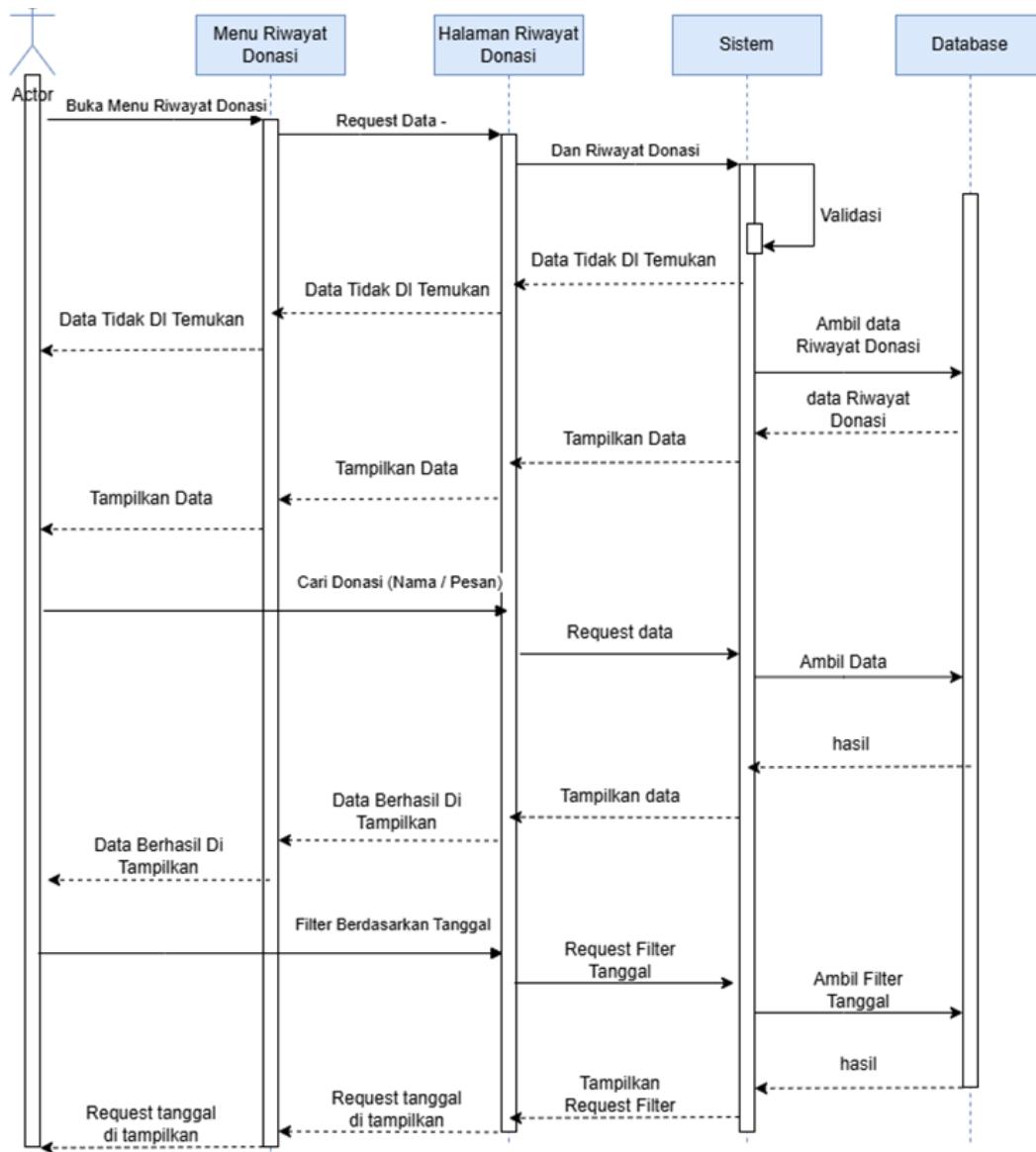
Sequence Diagram Kelola Profil



Gambar 3.15: Sequence Diagram Kelola Profil

Sequence diagram kelola profil menggambarkan interaksi antara kreator dan sistem dalam memproses pengelolaan data profil. Kreator membuka menu profil dan melakukan perubahan data. Sistem menerima perubahan, melakukan validasi, dan menyimpan perubahan ke basis data. Setelah proses berhasil, sistem mengirimkan respons bahwa profil berhasil diperbarui.

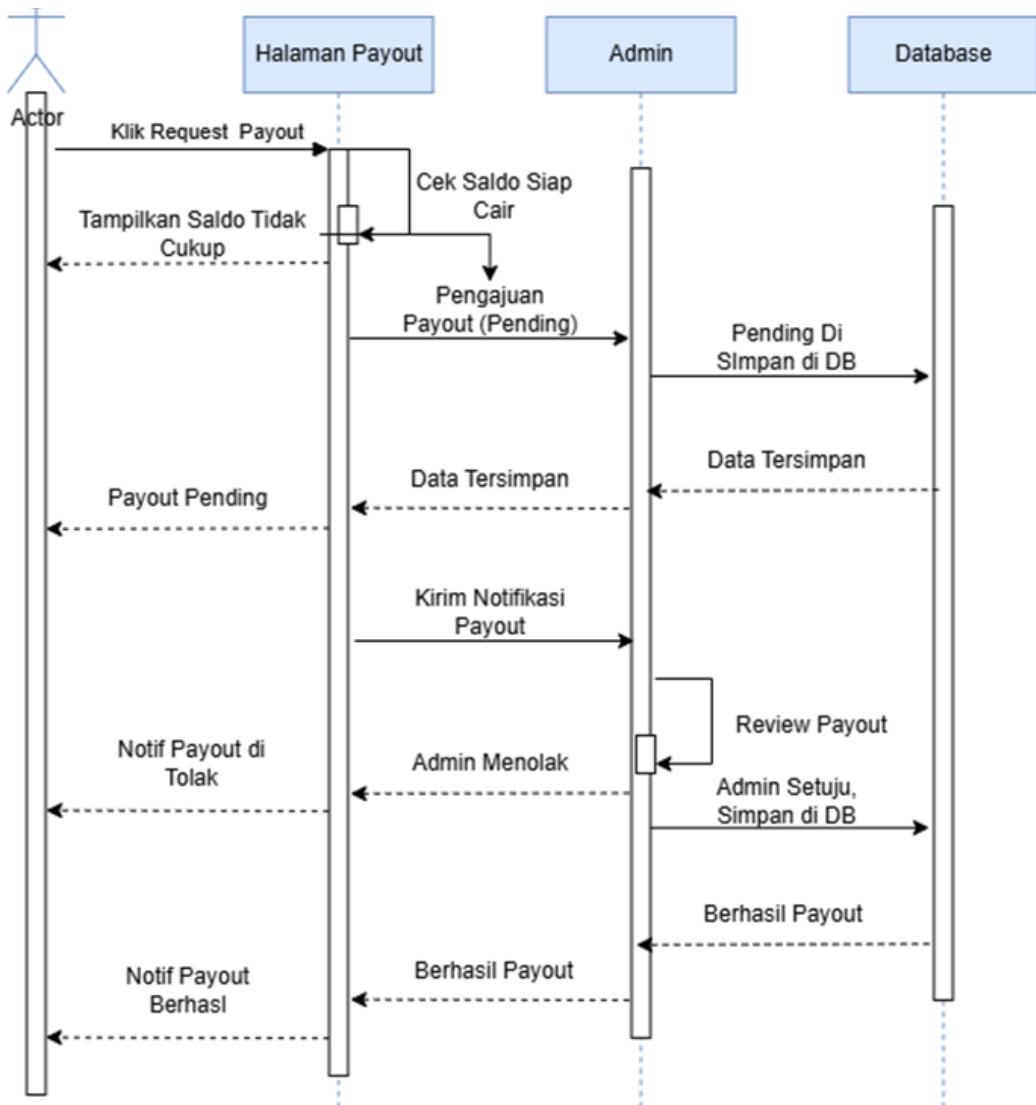
Sequence Diagram Riwayat Donasi



Gambar 3.16: Sequence Diagram Riwayat Donasi

Sequence diagram riwayat donasi menggambarkan proses pengambilan dan penampilan data riwayat donasi. Kreator mengakses halaman riwayat donasi, sistem melakukan validasi autentikasi, kemudian mengambil data donasi dari basis data. Kreator dapat melakukan pencarian dan filter data, dan sistem akan menampilkan hasil sesuai dengan permintaan.

Sequence Diagram Request Payout



Gambar 3.17: Sequence Diagram Request Payout

Sequence diagram request payout menggambarkan proses pengajuan pencairan dana oleh kreator. Kreator mengajukan permintaan payout melalui sistem. Sistem melakukan pengecekan saldo dan menyimpan data payout dengan status PENDING. Admin kemudian melakukan proses review terhadap permintaan tersebut dan memutuskan untuk menerima atau menolak. Apabila permintaan diterima, sistem memperbarui status payout dan mengirimkan notifikasi keberhasilan kepada kreator.

3.4.11 Model Entitas

Model entitas digunakan untuk merepresentasikan struktur data utama yang bekerja di dalam platform donasi. Setiap entitas dirancang untuk mendukung proses transaksi, pengelolaan kreator, penayangan media share di overlay, hingga alur pencairan dana. Secara umum, entitas yang digunakan dapat dikelompokkan menjadi entitas utama dan entitas pendukung.

Entitas utama platform meliputi:

- **Creator**

Menyimpan data kreator seperti nama, email, profil, serta informasi akun yang diperlukan untuk menerima donasi dan melakukan permintaan payout.

- **Donation**

Mencatat seluruh transaksi donasi, termasuk nominal, pesan, metode pembayaran, status (PENDING/PAID), serta relasi terhadap kreator yang menerima donasi.

- **MediaShare**

Entitas untuk menangani request media share (YouTube video) yang dikaitkan dengan donasi tertentu, termasuk durasi dan validasi media.

- **Payout**

Menyimpan informasi permintaan pencairan dana kreator, mencakup jumlah pencairan, fee platform, status (PENDING/APPROVED/PROCESSED), serta log aktivitas admin.

- **Notification**

Berfungsi untuk menampilkan data overlay secara real-time, seperti donasi terbaru atau media share yang harus ditayangkan oleh streamer atau kreator.

Selain itu, terdapat entitas pendukung yang digunakan untuk historisasi dan agregasi data, yaitu:

- **DonationHistory** – mencatat perubahan status donasi.

- **MonthlyLeaderboard** – menyimpan data peringkat donatur setiap bulan.

- **Contact** – mencatat umpan balik dari pengguna.

- **ProfileImage** – menyimpan data gambar untuk kebutuhan profil kreator.

- **Admin** – menyimpan kredensial admin yang bertugas memverifikasi payout dan melakukan manajemen sistem.

Seluruh entitas tersebut berperan dalam memastikan integritas data serta menghubungkan seluruh proses inti mulai dari transaksi donasi, pengelolaan kreator, hingga operasional sistem admin.

3.5 Metode Perancangan Teknis

Perancangan teknis pada platform donasi ini difokuskan pada penyusunan arsitektur layanan yang aman, efisien, dan mudah dipelihara. Pendekatan utama yang digunakan adalah pemisahan tanggung jawab antar modul serta penerapan pola penanganan API yang konsisten. Setiap endpoint dirancang mengikuti alur standar, yaitu validasi metode HTTP, autentikasi menggunakan JWT untuk endpoint privat, validasi input, eksekusi operasi basis data, dan pengembalian respons dalam format JSON. Pola yang seragam ini memudahkan proses debugging serta menjaga konsistensi perilaku lintas layanan.

Dari sisi keamanan, validasi token JWT diterapkan untuk memastikan setiap permintaan memiliki otorisasi yang benar, termasuk pengecekan masa berlaku token dan jenis pengguna (donor, kreator, atau admin). Seluruh input kritis seperti username, nominal donasi, dan URL media share divalidasi untuk mencegah data tidak sah masuk ke dalam sistem. Integrasi pembayaran dirancang agar hanya bergantung pada webhook resmi Midtrans, sehingga status transaksi tidak bergantung pada aktivitas pengguna di sisi klien.

Optimasi basis data dilakukan melalui penempatan indeks pada atribut yang sering digunakan dalam kueri, seperti `createdAt`, `creatorId`, dan `creatorUsername`. Operasi agregasi seperti leaderboard dan statistik kreator menggunakan pipeline agregasi MongoDB untuk mengurangi beban pada server aplikasi. Pembatasan kueri (*limit*) diterapkan untuk mencegah pengambilan data berlebihan pada endpoint yang memiliki potensi pertumbuhan data tinggi.

Selain itu, proses donasi dan penyajian notifikasi dipisahkan dari alur pembayaran utama. Server hanya membuat record donasi dan *Snap Token* pada permintaan awal, sementara pembaruan status dan pemicu notifikasi dilakukan ketika webhook diterima. Pemisahan ini meningkatkan stabilitas sistem dan memastikan overlay hanya menampilkan data yang telah tervalidasi.

3.6 Metode Pengujian

Metode pengujian pada platform donasi ini dirancang untuk memastikan bahwa seluruh fungsi sistem berjalan sesuai kebutuhan, aman digunakan, dan menghasilkan data yang konsisten. Pendekatan pengujian dilakukan melalui kombinasi pengujian unit, pengujian integrasi, serta pengujian fungsional terhadap endpoint API dan alur bisnis utama. Fokus utama pengujian meliputi keakuratan proses donasi, keandalan mekanisme payout, integritas data pada leaderboard serta statistik kreator, dan validitas proses autentikasi berbasis OAuth dan JWT.

Pengujian dilakukan menggunakan data uji terkontrol, simulasi webhook pembayaran, serta verifikasi hasil secara langsung melalui basis data. Seluruh skenario kritis seperti validasi input, autentikasi dan otorisasi, serta penanganan kesalahan diuji untuk memastikan sistem tetap stabil dalam berbagai kondisi operasional.

3.6.1 Jenis Pengujian

Pengujian sistem mencakup beberapa jenis pengujian sebagai berikut:

a. Pengujian Unit (Unit Testing)

Dilakukan pada fungsi atau modul kecil yang berdiri sendiri, seperti:

- Validasi token JWT (sign dan verify),
- Perhitungan *platformFee* dan *finalAmount* pada payout,
- Validasi input donasi (amount, username, media share).

b. Pengujian Integrasi (Integration Testing)

Berfokus pada alur yang melibatkan beberapa komponen, seperti:

- Proses donasi lengkap (create donation → Snap Token → pembayaran → webhook → update status),
- Permintaan payout oleh kreator dan proses approval oleh admin,
- Penampilan notifikasi overlay berdasarkan data terbaru.

c. Pengujian Fungsional (Functional Testing)

Dilakukan untuk mengevaluasi apakah setiap endpoint memenuhi kebutuhan fungsional yang telah ditetapkan, seperti:

- Validasi username saat login atau donasi,
- Pembatasan akses endpoint dashboard menggunakan JWT,

- Respons error 400, 401, 404, dan 500 yang konsisten.

d. Pengujian Keamanan (Security Testing)

Meliputi:

- Akses API tanpa token harus ditolak,
- Token kedaluwarsa atau *userType* tidak sesuai harus gagal,
- Percobaan pengiriman payload tidak valid harus tervalidasi.

e. Pengujian Kinerja (Performance Testing)

Difokuskan pada:

- Kecepatan query leaderboard (target < 300 ms),
- Stabilitas respons webhook,
- Performa overlay saat proses polling data.

3.6.2 Skenario Pengujian

Beberapa skenario pengujian utama yang digunakan meliputi:

1. Skenario 1 – Donasi Berhasil

Input valid → server membuat record PENDING → Snap Token sukses → pembayaran di Midtrans → webhook diterima → status menjadi PAID → overlay menampilkan donasi.

2. Skenario 2 – Donasi Gagal Validasi

Nominal di bawah minimum atau format URL salah → server mengembalikan status 400.

3. Skenario 3 – Akses Endpoint Tanpa Token

Mengakses leaderboard atau payout tanpa header *Authorization* → sistem mengembalikan 401 *Unauthorized*.

4. Skenario 4 – Request Payout oleh Kreator

Saldo cukup → request dicatat → status PENDING → admin review → APPROVED → status PROCESSED → saldo kreator berkurang sesuai *finalAmount*.

5. Skenario 5 – Token Salah atau Kedaluwarsa

Token tidak valid atau kedaluwarsa → akses ditolak dengan pesan error yang konsisten.

6. Skenario 6 – Webhook Simulasi Midtrans

Webhook dikirim manual dari Postman → status donasi berubah menjadi PAID → overlay menampilkan notifikasi.

3.6.3 Alat Pengujian

Alat yang digunakan dalam proses pengujian meliputi:

- **Postman** – untuk pengujian API (header, body, autentikasi).
- **MongoDB Compass** – untuk memverifikasi perubahan data secara langsung.
- **Logging Next.js (console)** – untuk memantau webhook, error, dan alur proses.

3.7 Evaluasi Keberhasilan

Evaluasi keberhasilan dilakukan untuk menilai sejauh mana implementasi sistem memenuhi kebutuhan fungsional, stabilitas operasional, serta ketepatan mekanisme kritis seperti autentikasi, pengelolaan sesi, dan pemrosesan pembayaran. Penilaian dilakukan melalui pengujian terstruktur dan analisis hasil *code coverage* yang dihasilkan dari proses unit testing pada modul-modul inti.

3.7.1 Cakupan Pengujian

Pengujian difokuskan pada komponen yang tergolong kritikal bagi kelangsungan sistem, meliputi:

- Autentikasi dan Otorisasi (JWT, session handling, route protection)
- Pemrosesan Pembayaran (Midtrans Snap dan webhook)
- Fungsi donasi (validasi input, perhitungan status)
- Fungsi payout (perhitungan *finalAmount*, status machine payout)

Komponen lain seperti UI Component dan utilitas pendukung tidak menjadi prioritas utama dalam pengujian ini karena tidak langsung berpengaruh pada keamanan maupun alur transaksi.

3.7.2 Hasil Code Coverage

Pengujian menghasilkan metrik *code coverage* sebagai berikut:

Tabel 3.1: Code Coverage

Metode	Cakupan
Statements	14.71%
Branch	15.56%
Functions	29.35%
Lines	14.21%

Meskipun angka coverage terlihat rendah secara keseluruhan, hal ini tidak mencerminkan kualitas fungsional sistem secara langsung karena coverage tidak mencakup seluruh file, melainkan hanya modul-modul yang dipilih berdasarkan kategori kritis. File non-kritis seperti UI, utilitas ringan, dan helper statis tidak disertakan dalam pengujian sehingga turut menurunkan total persentase.

Hasil tersebut menegaskan bahwa:

- Seluruh proses inti (autentikasi, sesi, validasi donasi, perhitungan payout, dan pembayaran) telah berhasil diuji.
- Jalur eksekusi utama (*main happy path*) telah tervalidasi.
- Beberapa jalur error belum sepenuhnya dicakup, sehingga membuka peluang peningkatan lebih lanjut.

3.7.3 Interpretasi dan Evaluasi

Berdasarkan hasil pengujian dan *code coverage* tersebut, dapat disimpulkan bahwa:

1. Stabilitas alur bisnis utama telah terverifikasi, terutama mekanisme donasi dan payout yang melibatkan transaksi dan webhook.
2. Keamanan dasar terkait autentikasi, JWT, dan sesi telah diuji dan berfungsi sesuai kebutuhan.
3. Konsistensi data pada proses pembayaran serta pencatatan notifikasi berhasil diuji melalui simulasi webhook.
4. Rendahnya nilai coverage lebih disebabkan oleh fokus pengujian pada modul kritis, bukan karena seluruh sistem tidak diuji.
5. Sistem dinilai layak digunakan, namun peningkatan cakupan pengujian tetap direkomendasikan untuk modul non-kritis seperti UI dan utilitas.

3.7.4 Kesimpulan Evaluasi

Secara keseluruhan, sistem telah memenuhi fungsi utamanya mulai dari pemrosesan donasi, pembayaran, hingga payout dan notifikasi. Hasil pengujian menunjukkan bahwa fitur inti berjalan stabil, meskipun pengujian yang lebih luas masih diperlukan pada tahap pengembangan selanjutnya.

DAFTAR PUSTAKA

- [1] Al-Fedaghi. Interaction modeling. *Systems Journal*, 2020.
- [2] Aprilliyani and Hamdana. Http methods. *Network Computing*, 2020.
- [3] Chandramouli and Butcher. Api security threats. *NIST*, 2020.
- [4] Marwah M. A. Dabdawb. Using uml diagrams to model system behavior. *Software Engineering Review*, 2024.
- [5] Abbott et al. Diagram consistency. *Modeling Journal*, 2025.
- [6] Afifah et al. Erd in database design. *Computer Science*, 2022.
- [7] Amani Bestari et al. Uml standards. *Software Modeling*, 2024.
- [8] Ariesta et al. Agile software development for adaptive systems. *Software Engineering Journal*, 2020.
- [9] Ball et al. Bearer token risks. *Security Review*, 2020.
- [10] Bucko et al. Jwt security. *Cybersecurity*, 2023.
- [11] Byali et al. Nosql databases. *Big Data Journal*, 2022.
- [12] Carvalho et al. Document databases. *Database Systems*, 2023.
- [13] Ehsan et al. Rest api principles. *Software Engineering*, 2022.
- [14] Fadzirul Kamarubahrin et al. Donation-based crowdfunding. *International Journal of Crowdfunding*, 2020.
- [15] Fauzan et al. Oop modeling. *Programming Education*, 2021.
- [16] Fauzan et al. Use case assessment. *Computer Education*, 2021.
- [17] Feibriandika et al. Digital philanthropy in indonesia. *Jurnal Sosial*, 2024.
- [18] Fraihat et al. Modern web architecture. *Web Systems Journal*, 2022.

- [19] Geofrey et al. Distributed systems. *Network Journal*, 2020.
- [20] Greselda Gosal et al. Transparency and credibility in online donation systems. *Digital Philanthropy Journal*, 2020.
- [21] Gudu et al. Oidc for ssh. *Network Security*, 2025.
- [22] Hammann et al. Openid connect. *Identity Journal*, 2020.
- [23] Jebli et al. Use case evaluation. *Education Computing*, 2024.
- [24] Jha et al. Business process modeling. *Information Systems*, 2023.
- [25] Jullien et al. Two sided digital platform theory. *Digital Market Journal*, 2021.
- [26] Kamarudin et al. Donation intention. *Digital Business*, 2023.
- [27] Lodderstedt et al. Oauth 2.0 security. *IETF*, 2025.
- [28] Marchezan et al. Uml in sdlc. *Software Engineering*, 2023.
- [29] Nashikhuddin et al. Jwt in web services. *Information Technology*, 2023.
- [30] Pan et al. Process visualization. *Engineering Design*, 2024.
- [31] Philippaerts et al. Authorization servers. *Security Journal*, 2022.
- [32] Pulungan et al. Erd design. *Database Journal*, 2020.
- [33] Rahman et al. Token authentication. *Web Security*, 2020.
- [34] Rahmayanti et al. Social proof in digital platforms. *Social Media Studies*, 2024.
- [35] R amdany et al. Uml activity diagram applications. *Software Engineering Review*, 2020.
- [36] Ratumurun et al. System flowchart. *Information Systems*, 2020.
- [37] Roziqin et al. Restful services. *Information Systems*, 2023.
- [38] Sandfreni et al. Activity diagram. *UML Journal*, 2021.
- [39] Shneor et al. Digital platform ecosystems. *Journal of Digital Business*, 2020.
- [40] Sirisawat et al. Crowdfunding platform design. *Technology Journal*, 2022.
- [41] Siska Narulita et al. Class diagram. *Object-Oriented Journal*, 2024.
- [42] Siska Narulita et al. Sequence diagram. *Software Design*, 2024.
- [43] Sun et al. Api attack surface. *Cyber Defense*, 2022.

- [44] Wang et al. Challenges in use case diagram evaluation. *Journal of Software Modeling*, 2025.
- [45] Yasuda et al. Federated identity. *Security Systems*, 2022.
- [46] Zhang et al. Flowchart modeling. *Software Modeling*, 2023.
- [47] Zikrinawati et al. Influence of trust and perceived risk on online donation decisions. *JIVA Journal*, 2025.
- [48] Halim. Platform donasi digital dan akses pendanaan sosial. *Jurnal Teknologi Informasi*, 2024.
- [49] Hidayatullah and Purbasari. Online zakat platform. *Islamic Finance Journal*, 2022.
- [50] Indriyani and Ibrahim. Digital donation behavior. *Journal of Social Tech*, 2024.
- [51] Ismail. Three tier lms implementation. *Education Technology*, 2023.
- [52] Jaimez-Gonzalez and Martinez-Samora. Conceptual data modeling. *Information Systems*, 2020.
- [53] Maruf and Ugli. Three tier architecture. *Software Architecture Journal*, 2020.
- [54] Neelan. Oauth best practices. *API Security*, 2022.
- [55] Nyabuto. Http protocol. *Internet Technology*, 2023.
- [56] Prabu and De Paul. Scalable web architecture. *Computer Science Review*, 2025.
- [57] Assistant Professor. Client server model. *Computer Basics*, 2020.
- [58] Saputra and Putri Nabila. Qris pada donation based crowdfunding. *Jurnal Fintech*, 2025.
- [59] Singh and Chaudhary. Oauth applications. *Mobile Computing*, 2023.
- [60] Tarigan. Trust dan inovasi pada donasi online. *Jurnal E-Commerce*, 2023.
- [61] Zhao. Rest api vulnerabilities. *Security Journal*, 2020.
- [62] Zhou and Inoue. Digital platform architecture. *Information Systems Journal*, 2025.