

# **LAPORAN PROYEK AKHIR MAHASISWA**

## **E-CONCALC: ELECTRONIC CONVERTER & CALCULATOR**

Aplikasi Kalkulator Ilmiah dan Konverter Lintas Platform  
dengan Sinkronisasi Riwayat Berbasis REST API

**Nama Mahasiswa:** Raffelino Hizkia Marbun

**NIM:** 2423102065

**Dosen:** Rahmat Purwoko, S.T., M.T.

**Program Studi:** Rekayasa Keamanan Siber

**Institusi:** Politeknik Siber dan Sandi Negara

**Tahun:** 2026

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan proyek akhir ini dengan baik. Proyek akhir ini berjudul "E-Concalc: Electronic Converter & Calculator" yang merupakan aplikasi kalkulator ilmiah dan konverter lintas platform dengan fitur sinkronisasi riwayat berbasis REST API.

Laporan ini disusun sebagai salah satu syarat dalam menyelesaikan tugas proyek akhir pada Program Studi Rekayasa Keamanan Siber, Politeknik Siber dan Sandi Negara.

Penulis mengucapkan terima kasih kepada Bapak Rahmat Purwoko, S.T., M.T. selaku dosen pembimbing yang telah memberikan arahan dan bimbingan selama penggeraan proyek ini.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun untuk perbaikan di masa mendatang.

Bogor, Februari 2026

Penulis,

Raffelino Hizkia Marbun

NIM: 2423102065

## **DAFTAR ISI**

*(Daftar isi dapat disusun secara otomatis menggunakan fitur Table of Contents pada Microsoft Word setelah dokumen selesai diisi.)*

# **BAB I – PENDAHULUAN**

## **1.1 Latar Belakang**

Di era digital saat ini, kebutuhan akan perangkat lunak kalkulasi dan konversi menjadi semakin penting dalam kehidupan sehari-hari maupun dunia akademik. Mahasiswa, profesional, dan masyarakat umum memerlukan alat bantu perhitungan yang cepat, akurat, dan dapat diakses dari berbagai perangkat.

Perkembangan teknologi web dan mobile yang pesat membuka peluang untuk membangun aplikasi lintas platform (cross-platform) yang dapat menyediakan pengalaman pengguna yang konsisten. Dengan adanya teknologi REST API, sinkronisasi data antar platform menjadi hal yang memungkinkan dan efisien.

Dari perspektif keamanan siber, pengembangan aplikasi yang aman menjadi hal yang krusial. Aspek autentikasi, otorisasi, dan perlindungan data pengguna harus menjadi perhatian utama dalam setiap pengembangan perangkat lunak.

Berdasarkan latar belakang tersebut, penulis mengembangkan proyek "E-Concalc" (Electronic Converter & Calculator), sebuah platform kalkulator ilmiah dan konverter yang tersedia dalam bentuk website (berbasis Laravel) dan aplikasi mobile Android (berbasis Kotlin/Jetpack Compose), dengan fitur sinkronisasi riwayat perhitungan melalui REST API yang aman menggunakan Laravel Sanctum.

## **1.2 Rumusan Masalah**

1. Bagaimana membangun aplikasi kalkulator ilmiah dan konverter yang dapat berjalan di platform web dan mobile Android secara terintegrasi?
2. Bagaimana mengimplementasikan sinkronisasi riwayat perhitungan antar platform menggunakan REST API yang aman?
3. Bagaimana menerapkan sistem autentikasi yang aman dengan mendukung login konvensional dan Google OAuth?
4. Bagaimana merancang antarmuka pengguna yang modern, responsif, dan mudah digunakan pada kedua platform?

## **1.3 Tujuan**

1. Membangun aplikasi E-Concalc yang menyediakan fitur kalkulator ilmiah, konverter satuan, dan konverter mata uang pada platform web dan mobile Android.
2. Mengimplementasikan REST API yang aman untuk sinkronisasi riwayat perhitungan secara real-time antar platform.
3. Menerapkan sistem autentikasi menggunakan Laravel Sanctum dengan dukungan login email/password dan Google OAuth.

4. Merancang UI yang modern dengan dark theme, glassmorphism, dan Material Design 3.
5. Mengimplementasikan Progressive Web App (PWA) agar website dapat diinstal dan digunakan secara offline.

## 1.4 Manfaat

### Manfaat Akademik:

Pengalaman langsung dalam pengembangan aplikasi full-stack lintas platform, penerapan prinsip keamanan siber pada aplikasi nyata, serta pemahaman mendalam tentang arsitektur REST API dan autentikasi token-based.

### Manfaat Praktis:

Aplikasi E-Concalc dapat dimanfaatkan oleh mahasiswa dan masyarakat umum sebagai alat bantu kalkulasi ilmiah, konversi satuan, dan konversi mata uang yang dapat diakses dari web maupun mobile.

### Manfaat Keamanan:

Mendemonstrasikan penerapan praktik keamanan terbaik dalam pengembangan perangkat lunak, termasuk hashing password, token-based authentication, validasi input, dan perlindungan terhadap SQL Injection dan XSS.

## 1.5 Ruang Lingkup

1. Platform: Website (Laravel 10) dan Aplikasi Mobile Android (Kotlin/Jetpack Compose).
2. Fitur utama: Kalkulator ilmiah, konverter satuan (15 kategori), konverter mata uang (52 mata uang).
3. Autentikasi: Login/register dengan email/password dan Google OAuth.
4. Sinkronisasi: REST API dengan Laravel Sanctum untuk sinkronisasi riwayat perhitungan.
5. Database: MySQL 8.0 dengan tabel users, riwayat, personal\_access\_tokens, dan sessions.
6. PWA: Progressive Web App dengan service worker dan offline caching.
7. Mode tamu: Penggunaan tanpa login dengan penyimpanan lokal.

## BAB II – TINJAUAN TEORI DAN TEKNOLOGI

### 2.1 Konsep Dasar Aplikasi Web

Aplikasi web E-Concalc dibangun menggunakan framework Laravel 10 yang mengikuti arsitektur Model-View-Controller (MVC). Teknologi web yang digunakan:

- **HTML**: Bahasa markup untuk struktur halaman web, melalui template engine Blade.
- **CSS**: Styling halaman web dengan efek glassmorphism dan dark theme.
- **JavaScript**: Logika client-side termasuk operasi kalkulator dan interaksi asinkron dengan API.
- **PHP 8.1+ & Laravel 10**: Framework MVC dengan routing, middleware, Eloquent ORM.
- **REST API**: Arsitektur layanan web berbasis HTTP untuk komunikasi frontend-backend.
- **MySQL 8.0**: RDBMS untuk penyimpanan data pengguna dan riwayat.
- **Laravel Sanctum**: Autentikasi API berbasis token.
- **Laravel Socialite**: OAuth dengan Google.

### 2.2 Konsep Dasar Aplikasi Mobile

Aplikasi mobile dikembangkan native untuk Android dengan teknologi:

- **Kotlin 2.0**: Bahasa pemrograman resmi Android dengan null-safety dan coroutines.
- **Jetpack Compose**: Toolkit UI deklaratif modern dari Google.
- **Material Design 3**: Sistem desain terbaru Google untuk UI konsisten.
- **Retrofit 2 + OkHttp**: HTTP client untuk request ke REST API.
- **Credential Manager API**: API modern untuk Google Sign-In.
- **SharedPreferences**: Penyimpanan key-value lokal untuk token dan data tamu.
- **Navigation Compose**: Pengelolaan navigasi antar layar.

### 2.3 Keamanan Aplikasi

Praktik keamanan yang diterapkan sesuai standar OWASP:

- **Validasi Input**: Validasi pada sisi klien dan server.
- **Autentikasi**: Sanctum token-based (API), session-based (web). Password di-hash bcrypt.
- **Otorisasi**: Middleware auth:sanctum pada endpoint yang dilindungi.

- **Proteksi CSRF:** Token CSRF otomatis pada form submission.
- **Proteksi SQL Injection:** Prepared statements via Eloquent ORM.
- **Proteksi XSS:** Blade template auto-escaping.
- **HTTPS:** Komunikasi terenkripsi direkomendasikan untuk produksi.

## BAB III – ANALISIS DAN PERANCANGAN SISTEM

### 3.1 Analisis Kebutuhan

#### 3.1.1 Kebutuhan Fungsional

Kode	Kebutuhan	Deskripsi
KF-01	Kalkulator Ilmiah	Operasi matematika dasar dan ilmiah (trigonometri, logaritma, faktorial, pangkat, akar).
KF-02	Fungsi Memori	Operasi memori M+, M-, MR, MC.
KF-03	Mode DEG/RAD	Toggle mode derajat dan radian.
KF-04	Konverter Satuan	Konversi 15 kategori (panjang, berat, suhu, kecepatan, luas, volume, dll).
KF-05	Konverter Mata Uang	Konversi real-time 52 mata uang dengan caching offline.
KF-06	Autentikasi	Login/register dengan email/password dan Google OAuth.
KF-07	Sinkronisasi Riwayat	Sinkronisasi riwayat antar platform via REST API.
KF-08	Mode Tamu	Penggunaan tanpa login dengan penyimpanan lokal.
KF-09	PWA	Website dapat diinstal sebagai Progressive Web App.

#### 3.1.2 Kebutuhan Non-Fungsional

Kode	Aspek	Deskripsi
KNF-01	Performa	Waktu respon < 2 detik.
KNF-02	Keamanan	Data terenkripsi, API dilindungi token.
KNF-03	Kompatibilitas	Website responsif, mobile Android 7.0+.
KNF-04	Ketersediaan	Dapat digunakan offline (mode tamu).
KNF-05	Maintainability	Kode terstruktur MVC/MVVM.

### 3.2 Diagram Sistem

Catatan: Diagram Use Case, Sequence, Class, dan ERD dapat disisipkan dalam bentuk gambar pada bagian ini.

#### 3.2.1 Use Case Diagram

Aktor utama:

1. Guest: Menggunakan kalkulator, konverter satuan/mata uang dengan penyimpanan lokal.
2. Authenticated User: Semua kemampuan Guest + sinkronisasi riwayat + manajemen akun.

[Sisipkan gambar Use Case Diagram]

#### 3.2.2 ERD / Database Schema

Database MySQL dengan skema:

Tabel	Kolom Utama
users	id, name, email, password, google_id, created_at, updated_at
riwayat	id, user_id (FK), operasi, tipe (calc/conv/currency), created_at
personal_access_tokens	id, tokenable_type, tokenable_id, name, token, abilities
sessions	id, user_id, ip_address, user_agent, payload, last_activity

Relasi: users (1) → (\*) riwayat melalui foreign key user\_id.

### 3.3 Arsitektur Sistem

**Website (MVC):** Model (User, Riwayat) – View (Blade templates) – Controller (Auth, Calculator, API controllers).

**Mobile (MVVM-like):** Data layer (ApiClient, HistoryManager, TokenManager) – UI layer (Composable screens) – Navigation (Screen.kt).

**Arsitektur keseluruhan:** Frontend Layer (Web + Mobile) → API Layer (Laravel REST + Sanctum) → Data Layer (MySQL + localStorage/SharedPreferences).

## BAB IV – IMPLEMENTASI

### 4.1 Lingkungan Pengembangan

Komponen	Teknologi
Sistem Operasi	Windows 11
IDE (Web)	Visual Studio Code
IDE (Mobile)	Android Studio (Ladybug)
Backend	Laravel 10 (PHP 8.1+)
Mobile	Kotlin 2.0 + Jetpack Compose + Material 3
Database	MySQL 8.0 (via XAMPP)
Version Control	Git + GitHub
Package Manager	Composer, Gradle, npm

### 4.2 Implementasi Backend

Backend Laravel dengan struktur:

- app/Models/ – User.php, Riwayat.php
- app/Http/Controllers/ – AuthController, CalculatorController
- app/Http/Controllers/Api/ – AuthController, HistoryController
- routes/web.php & routes/api.php
- database/migrations/ – 12 file migrasi
- resources/views/ – Blade templates
- public/ – CSS, JS, manifest.json, sw.js

#### REST API Endpoints:

Method	Endpoint	Deskripsi
POST	/api/register	Registrasi pengguna baru
POST	/api/login	Login email/password
POST	/api/login/google	Login Google OAuth
GET	/api/user	Data pengguna (auth)
POST	/api/logout	Logout + revoke token
GET	/api/history	Ambil riwayat (filter ?tipe)
POST	/api/history	Simpan riwayat baru
DELETE	/api/history	Hapus semua riwayat
DELETE	/api/history/{id}	Hapus riwayat spesifik

## **4.3 Implementasi Frontend / Mobile**

### **4.3.1 Website**

Website dengan glassmorphism dark theme. Halaman: Calculator, Unit Converter, Currency Converter, Login, Register, Riwayat.

### **4.3.2 Mobile Android**

15 file Kotlin: MainActivity, 6 screens (Splash, Login, Register, Calculator, Converter, Currency), ApiClient, ApiService, Screen, HistoryManager, TokenManager, Color, Theme, Type.

## **4.4 Integrasi**

- REST API sebagai jembatan: Endpoint API yang sama digunakan oleh web (Fetch API) dan mobile (Retrofit).
- Token-based authentication: Sanctum Personal Access Token pada header Authorization: Bearer.
- Google OAuth: Website (Socialite) dan mobile (Credential Manager) mengarah ke endpoint yang sama.
- Fallback lokal: Mode tamu menggunakan localStorage (web) / SharedPreferences (mobile).
- Cross-platform sync: Akun yang sama di web dan mobile melihat riwayat yang sama.

## BAB V – PENGUJIAN DAN EVALUASI

### 5.1 Pengujian Unit

Kode	Test Case	Skenario	Hasil
TC-01	Aritmatika dasar	$2+3 = 5$	Pass
TC-02	Trigonometri sin	$\sin(90) \text{ DEG} = 1$	Pass
TC-03	Logaritma	$\log(100) = 2$	Pass
TC-04	Konversi suhu	$100^{\circ}\text{C} = 212^{\circ}\text{F}$	Pass
TC-05	Konversi panjang	$1 \text{ km} = 1000 \text{ m}$	Pass
TC-06	Registrasi	Data valid → 201	Pass
TC-07	Login	Email/pass valid → token	Pass
TC-08	Simpan riwayat	POST /api/history → 201	Pass
TC-09	Ambil riwayat	GET /api/history → 200	Pass
TC-10	Hapus riwayat	DELETE /api/history → 200	Pass

### 5.2 Pengujian Integrasi

Kode	Skenario	Hasil
IT-01	Login web, cek riwayat muncul di mobile	Pass
IT-02	Simpan riwayat dari mobile, cek muncul di web	Pass
IT-03	Login Google di web, verifikasi token	Pass
IT-04	Kalkulator + simpan riwayat end-to-end	Pass
IT-05	Mode tamu: riwayat lokal, tidak ke API	Pass
IT-06	Logout: token direvoke, API menolak	Pass

### 5.3 Pengujian Keamanan

Kode	Skenario	Hasil	Status
SK-01	SQL Injection pada login	Terblokir (prepared statements)	Pass
SK-02	XSS pada input kalkulator	Terblokir (Blade escaping)	Pass
SK-03	Akses API tanpa token	HTTP 401 Unauthorized	Pass
SK-04	CSRF pada form	Terblokir (CSRF token)	Pass
SK-05	Brute force login	Rate limiting aktif	Pass
SK-06	Password di DB	Hash bcrypt (bukan plaintext)	Pass

### 5.4 Evaluasi

**Kelebihan:**

- Cross-platform synchronization seamless.
- UI modern: glassmorphism (web) + Material 3 (mobile).
- Multi-auth (email/password + Google OAuth).
- Mode tamu tanpa registrasi.
- PWA support.
- REST API yang reusable.

**Kekurangan:**

- Belum ada export riwayat PDF/Excel dari mobile.
- Konversi mata uang memerlukan internet.
- Belum ada unit testing otomatis menyeluruh.
- Belum ada reset password via email.

## **BAB VI – CI/CD DAN DEPLOYMENT**

### **6.1 Version Control**

Proyek menggunakan Git + GitHub dengan pendekatan monorepo yang mencakup seluruh komponen.

### **6.2 Deployment Website**

- Server lokal: XAMPP (Apache + MySQL + PHP).
- Konfigurasi .env untuk database dan service credentials.
- Migrasi: php artisan migrate.
- Dev server: php artisan serve --port=8080.
- Produksi: shared hosting atau VPS dengan Apache/Nginx.

### **6.3 Deployment Mobile**

- Build via Android Studio.
- Pengujian pada emulator dan perangkat fisik.
- BASE\_URL: http://10.0.2.2:8080/ (emulator), IP lokal (fisik).
- Generate APK release via Build > Generate Signed Bundle/APK.
- Distribusi: APK langsung atau Google Play Store.

### **6.4 Rekomendasi CI/CD**

Pipeline GitHub Actions/GitLab CI:

1. Build Stage: composer install, npm install, gradle build.
2. Test Stage: php artisan test, gradle test.
3. Deploy Stage: Auto-deploy ke staging/produksi.

## **BAB VII – PENUTUP**

### **7.1 Kesimpulan**

Proyek E-Concalc telah berhasil dibangun sebagai platform kalkulator ilmiah dan konverter lintas platform yang mengintegrasikan website Laravel 10 dan aplikasi mobile Android Kotlin/Jetpack Compose.

1. Fitur kalkulator ilmiah lengkap, konverter satuan (15 kategori), dan konverter mata uang (52 mata uang) berhasil diimplementasikan pada kedua platform.
2. REST API dengan Laravel Sanctum berhasil mengamankan proses autentikasi dan sinkronisasi riwayat.
3. Sistem autentikasi ganda (email/password + Google OAuth) berhasil diimplementasikan.
4. UI modern dengan glassmorphism (web) dan Material Design 3 (mobile) menghasilkan UX premium.
5. Aspek keamanan diterapkan sesuai standar OWASP.

### **7.2 Saran**

1. Implementasi CI/CD pipeline menggunakan GitHub Actions.
2. Fitur reset password melalui email verification.
3. Pengembangan versi iOS.
4. Export riwayat ke PDF/Excel dari mobile.
5. Two-factor authentication (2FA).
6. Penetration testing komprehensif.
7. Fitur kalkulator grafik.
8. Dashboard admin untuk monitoring.

## DAFTAR PUSTAKA

- Sommerville, I. (2015). *Software Engineering* (10th Ed). Pearson.
- Pressman, R. S. (2019). *Software Engineering: A Practitioner's Approach* (9th Ed). McGraw-Hill.
- Laravel Documentation. (2026). <https://laravel.com/docs>
- Kotlin Documentation. (2026). <https://kotlinlang.org/docs/home.html>
- Android Developers. (2026). Jetpack Compose. <https://developer.android.com/jetpack/compose>
- OWASP Foundation. (2026). <https://owasp.org/www-project-top-ten/>
- material Design 3. (2026). <https://m3.material.io/>
- Retrofit. (2026). <https://square.github.io/retrofit/>
- Mozilla MDN. (2026). Progressive Web Apps. [https://developer.mozilla.org/en-US/docs/Web/Progressive\\_web\\_apps](https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps)
- Laravel Sanctum. (2026). <https://laravel.com/docs/sanctum>

## LAMPIRAN

### A. Screenshot Aplikasi

[Sisipkan screenshot halaman utama website – kalkulator]

[Sisipkan screenshot unit converter website]

[Sisipkan screenshot currency converter website]

[Sisipkan screenshot halaman login website]

[Sisipkan screenshot aplikasi mobile – kalkulator]

[Sisipkan screenshot aplikasi mobile – login]

[Sisipkan screenshot aplikasi mobile – konverter]

### B. Contoh Kode

Kode sumber tersedia di repository GitHub proyek E-Concalc.

- Backend: Api/AuthController.php, Api/HistoryController.php
- Backend: User.php, Riwayat.php
- Frontend: script.js, currency.js
- Mobile: ApiClient.kt, ApiService.kt
- Mobile: CalculatorScreen.kt
- Mobile: HistoryManager.kt, TokenManager.kt

### C. Hasil Pengujian

Hasil pengujian detail dapat dilihat pada BAB V.