

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Codelabs adalah salah satu divisi yang ada di Universitas Komputer Indonesia (UNIKOM). Codelabs memiliki banyak aplikasi yang sudah dibangun dan dikembangkan, yang mana salah satunya adalah Aplikasi Pembantu Meningkatkan Kualitas Tidur. Aplikasi Pembantu Meningkatkan Kualitas Tidur yang dimaksud adalah Sleepy. Sleepy adalah aplikasi untuk membantu pengguna agar mendapatkan kualitas tidur yang lebih baik. Namun setelah melakukan inspeksi *source code* pada aplikasi Sleepy ternyata ditemukan masalah bahwa *source code* yang ada saat ini tidak lagi dapat dikembangkan atau dilakukan pemeliharaan. Peneliti melakukan inspeksi awal terhadap *source code* aplikasi Sleepy ditemukan beberapa masalah yaitu arsitektur kode yang tidak baik, beberapa *class* yang memiliki *cyclomatic complexity* yang sangat tinggi, dan juga masih menggunakan *support library* pada bagian *platform* android. Penggunaan *support library* saat ini sudah jarang digunakan karena pengembangan *support library* sudah dihentikan dan dipindahkan menggunakan *androidx* [1]. Perihal tersebut mengakibatkan banyaknya *error* pada aplikasi Sleepy bagian android karena banyak *library-library* yang sudah tidak dapat digunakan. Adapun hasil inspeksi *source code* dapat dilihat pada **LAMPIRAN A HASIL INSPEKSI SOURCE CODE**.

Berdasarkan fenomena tersebut, dilakukan riset untuk meningkatkan kualitas dari aplikasi Sleepy. Menurut buku *Software Engineering* karangan dari Roger S. Pressman pada edisi ketujuh bahwa *Software Quality Assurance (SQA)* dibagi menjadi beberapa tujuan yaitu *Requirements Quality*, *Design Quality*, *Code Quality*, dan *Quality Control Effectiveness* yang mana setiap tujuan tersebut memiliki aspek penilaiannya masing-masing [2]. Sejalan dengan permasalahan yang ada saat ini pada *source code* aplikasi Sleepy maka penelitian ini akan

berfokus pada bagian *Code Quality*. Atribut yang dinilai pada *Code Quality* adalah *Complexity, Maintainability, Understandability, Reusability, Documentation* [2].

Berdasarkan masalah yang terjadi saat ini, *Reengineering* atau rekayasa ulang sistem merupakan solusi karena dapat memberikan peluang terhadap aplikasi dalam meningkatkan *code quality* dan membuat aplikasi lebih bisa terpelihara [3]. *Reengineering* aplikasi yang akan dilakukan adalah pembaharuan struktur aplikasi yang sudah dibangun dengan menerapkan *design pattern* dan *clean code*. Dan juga aplikasi Sleepy bagian android akan dilakukan migrasi *environment* menjadi *androidx* dan perubahan bahasa menjadi kotlin, hal tersebut dilakukan karena Google sendiri sebagai pemilik dan penyedia *resource* Android menyarankan untuk menggunakan *Androidx* daripada *support library* dan menggunakan kotlin untuk memudahkan pengujian dan pemeliharaan [4]. Dan juga Google sendiri menjelaskan pada Google I/O 2019 bahwa semua fitur, *library*, dan teknologi terbaru akan diprioritaskan menggunakan bahasa kotlin [5]. Hasil dari *Reengineering* diharapkan dapat meningkatkan *code quality* dan membuat aplikasi menjadi bisa dipelihara dengan baik.

## 1.2 Perumusan Masalah

Berdasarkan penjelasan dari latar belakang masalah maka perumusan masalah pada penelitian ini adalah apakah dengan menerapkan *clean code, design pattern* dan menerapkan *androidx* dapat meningkatkan *code quality*.

## 1.3 Maksud dan Tujuan

Maksud dari penelitian ini adalah untuk melakukan *reengineering* aplikasi Sleepy. Tujuan dari penelitian ini yaitu meningkatkan *code quality*.

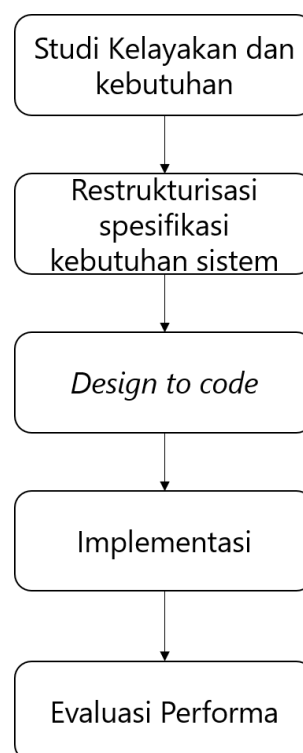
## 1.4 Batasan Masalah

Dalam penelitian ini, terdapat beberapa batasan masalah agar penelitian ini tidak menyimpang dari tujuan yang ingin dicapai. Berikut batasan masalah pada penelitian ini :

1. *Reengineering* yang dilakukan hanya berfokus pada *code quality*.
2. Data tidur didapatkan dari layanan dari S-Health.
3. Aplikasi web berfokus pada penyedia layanan data untuk aplikasi android.
4. Musik yang digunakan tergantung pada aplikasi Spotify, termasuk lisensi musik dan *media player*.
5. Aplikasi web menggunakan *framework* Laravel.
6. Pendekatan analisis perangkat lunak menggunakan *Object Oriented Analysis and Design (OOAD)*.

### 1.5 Metodologi Penelitian

Jenis penelitian pada penelitian ini adalah penelitian kualitatif *case study* [6]. Metodologi yang digunakan adalah *Enhanced Reengineering* yang mengacu pada paper “*Software Re-engineering: An Overview*”. Metodologi tersebut selanjutnya disesuaikan dengan kebutuhan penelitian. Langkah-langkah yang dilakukan pada penelitian ini dilihat pada **Gambar 1-1**.



**Gambar 1-1 Metodologi Penelitian**

### **1. Analisis Kelayakan dan Kebutuhan**

Pada tahap ini, analisis kelayakan dilakukan untuk memeriksa konfigurasi dan kompatibilitas pada aplikasi Sleepy. Setelah menyelesaikan studi kelayakan, maka dilakukan analisis kebutuhan sistem Sleepy.

### **2. Restrukturisasi Spesifikasi Kebutuhan Sistem**

Tahap ini menggambarkan secara detail proses spesifikasi yang direstrukturisasi pada aplikasi Sleepy. Pada tahap ini dilakukan perbandingan antara persyaratan sistem Sleepy yang sudah ada dengan mekanisme Sleepy yang baru.

### **3. *Design to Code***

Tahap ini melakukan perancangan arsitektur kode dengan menerapkan *design pattern* dan *clean code* pada aplikasi Sleepy. Hal tersebut dimaksudkan untuk memudahkan dalam perencanaan arsitektur kode dan menjadi mudah untuk proses pemeliharaan. Pada tahap ini juga dilakukan penyesuaian *design pattern* akan digunakan dengan kebutuhan aplikasi dan memastikan *design pattern* yang akan digunakan tepat guna.

### **4. Implementasi**

Tahap ini adalah tahap terakhir dari mekanisme tahap *reengineering* pada aplikasi Sleepy. Pada tahap implementasi, bagian – bagian tertentu diganti berdasarkan empat tahap sebelumnya [7].

### **5. Evaluasi Performa sistem baru dengan sistem yang sudah ada**

Tahap ini memberikan rincian tentang proses pengujian. Untuk pengujian, aplikasi Sleepy akan dibandingkan kinerja fungsionalitasnya dengan fungsionalitas aplikasi Sleepy yang baru.

## **1.6 Sistematika Penulisan**

Sistematika penulisan proposal ini disusun untuk memberikan gambaran umum mengenai penelitian yang dikerjakan. Sistematika penulisan dalam proposal skripsi ini adalah sebagai berikut:

### **BAB I PENDAHULUAN**

BAB I berisi tentang latar belakang permasalahan, merumuskan inti permasalahan, menentukan maksud dan tujuan, pembatasan masalah, metodologi penelitian, dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

BAB II berisi tentang bahan-bahan kajian, konsep dasar, dan teori dari para ahli yang berkaitan dengan penelitian. Meninjau permasalahan dan hal-hal yang berguna dari penelitian-penelitian dan menggunakannya sebagai acuan pemecahan masalah pada penelitian ini.

### **BAB III ANALISIS DAN PERANCANGAN**

BAB III berisi tentang tahapan analisis dari subjek penelitian untuk mengetahui hal atau masalah yang timbul dari penelitian yang dilakukan dan melakukan penentuan solusi yang tepat untuk masalah yang ditemukan.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM**

BAB IV menguraikan implementasi berdasarkan hasil analisis sebelumnya. Pada bagian ini juga akan ditentukan bagaimana model desain interaksi dibangun, diuji dan disesuaikan dengan hasil penelitian.

### **BAB V KESIMPULAN DAN SARAN**

BAB V berisi tentang kesimpulan dari hasil penelitian beserta saran untuk pengembangan selanjutnya.