

BAB III METODE PENELITIAN

3.1 Metode Pengembangan Sistem

Metode yang digunakan pada penelitian ini adalah *waterfall*. *Waterfall* merupakan sebuah metode pengembangan produk yang menerapkan prinsip seperti air terjun. Tidak ada fase yang dapat dilewati jika dokumentasi dan pengembangan pada fase tersebut sudah selesai dilakukan. Ketika terdapat perubahan kebutuhan pada produk yang sedang dikembangkan, proses pengembangan harus kembali ke fase sebelumnya (Schach, 2011). Setelah itu, fase selanjutnya yang sudah diselesaikan wajib untuk dilakukan kembali efek dari perubahan kebutuhan pada produk. Satu hal penting yang wajib dilakukan pada setiap tahap adalah pengujian pada akhir tahap pengembangan. Hal tersebut dilakukan untuk meminimalisir kesalahan yang mungkin terjadi pada hasil akhir sistem.

3.2 Tahap Penelitian

Penelitian yang dilakukan akan melewati tiga proses, yaitu studi literatur, pengembangan sistem, dan analisis dan kesimpulan. Studi literatur dilakukan untuk mengembangkan wawasan dan pengetahuan mengenai segala macam teknologi yang digunakan dan diterapkan pada sistem ini sehingga penggunaan teknologi tersebut dapat dilakukan secara tepat. Pengembangan sistem dilakukan dengan menggunakan metode *waterfall*. Penggunaan metode ini akan melibatkan seluruh rangkaian tahapan di dalamnya supaya pengembangan sistem dapat sesuai dengan tujuan penelitian yang sudah ditentukan. Setelah sistem sudah selesai dibuat, sistem akan dianalisis untuk dapat menghasilkan kekurangan dan kelebihan yang ada pada sistem ini. Lalu, kesimpulan akan dibuat dari analisis yang telah dilakukan.

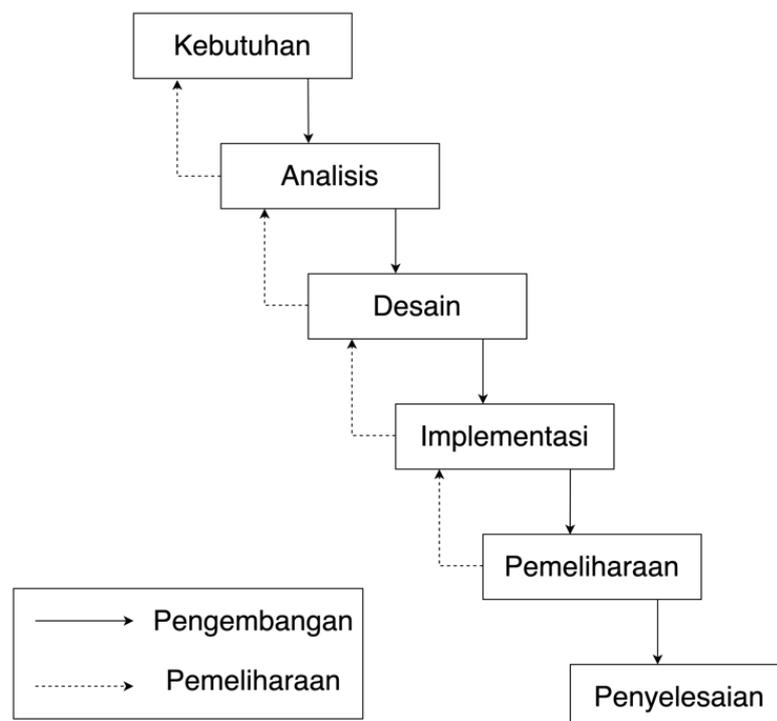
3.2.1 Studi Literatur

Studi literatur dilakukan dengan cara mencari referensi dan dasar pengetahuan mengenai teknologi yang digunakan pada penelitian ini. Referensi tersebut dapat berupa artikel ilmiah, jurnal, buku, dan situs resmi dari perusahaan pengembang

teknologi tersebut. Dasar pengetahuan yang berhubungan dengan teori dan metode pendukung bersumber dari artikel ilmiah, jurnal, dan buku. Sedangkan, teknis pengembangan yang berhubungan langsung dengan sistem bersumber dari situs resmi dari perusahaan pengembang yang menyediakan dokumentasi mengenai teknologi tersebut.

3.2.2 Pengembangan Sistem

Pengembangan sistem kelas terpadu ini menggunakan metode *waterfall*. Pada Gambar 3.1, terdapat enam tahap yang akan dilalui pada penelitian ini. Setiap tahap memiliki tujuan yang berbeda sehingga dibutuhkan pemahaman penting akan hasil dari setiap tahap yang akan dijalankan. Hasil dari tiap tahap akan digunakan sebagai awal dari pengembangan di tahap selanjutnya. Maka dari itu, tahap satu dengan tahap yang lainnya akan saling berkaitan.



Gambar 3.1. Diagram *Waterfall*.

a. Kebutuhan Sistem

Metode Waterfall bergantung pada dokumen spesifikasi aplikasi yang hendak dibuat. Ketika terdapat beberapa perubahan pada dokumen spesifikasi yang

diakibatkan adanya penambahan fitur atau perubahan fitur, tahap pengembangan harus kembali ke tahap sebelumnya di mana proses pengembangan harus melewati proses penentuan *requirements* dan seterusnya. Pada tahap ini, seluruh kebutuhan yang akan digunakan dalam pengembangan aplikasi perlu untuk dibuat dalam sebuah dokumentasi. Dokumentasi tersebut akan digunakan pada tahap selanjutnya hingga proses pengembangan selesai.

Sistem ini akan menggunakan identitas pengguna, berupa kartu, sebagai alat untuk identifikasi otoritas pengguna dalam memasuki kelas. Selain fitur utama tersebut, terdapat fitur lain yang perlu diterapkan pada sistem ini guna mendukung keberjalanan sistem ini agar dapat digunakan di lingkungan DPTE FPTK UPI. Fitur lain tersebut mencakup tentang penggunaan data rekam kelas yang dapat digunakan sebagai perekap presensi secara otomatis berdasarkan kehadiran mahasiswa selama kelas berlangsung. Selain itu, kelas pun dapat diatur secara mudah agar dapat disesuaikan dengan jadwal mata kuliah yang ada dan dosen yang tersedia. Fitur-fitur tersebut terdapat pada sistem ini sebagai salah satu tujuan yang akan dicapai dari perancangan dan pengembangan sistem kelas terpadu ini.

Terdapat beberapa perangkat keras yang dipakai untuk membuat peranti yang akan digunakan pada kelas sebagai pemindai RFID sebelum memasuki kelas. Tabel 3.1 merupakan data beberapa perangkat keras yang digunakan untuk membuat peranti sistem kelas terpadu dalam tahap perancangan.

Tabel 3.1. Perangkat keras yang digunakan pada peranti sistem kelas terpadu.

No	Nama Perangkat Keras	Keterangan
1	NodeMCU (ESP8266)	Mikrokontroler pengendali utama peranti
2	Sensor RFID (MFRC-522)	Sensor pendeteksi nomor identifikasi dari kartu
3	LCD OLED 128x64	Layar OLED penampil informasi
4	PCF8574	Penambah pin I/O
5	KEYPAD 4x4	Modul sarana input data angka
6	Kartu RFID 13,56 MHz	Kartu berteknologi frekuensi radio yang menyimpan nomor identifikasi
7	LED	Indikator pengganti solenoid <i>door lock</i>
8	<i>Bread Board</i>	Media penghubung antar komponen

Lumban Toruan Ebin Elbentor, 2023

RANCANG BANGUN KELAS TERPADU BERBASIS RADIO FREQUENCY IDENTIFICATION
Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

9	Kabel <i>Jumper</i>	Penghubung antar komponen pada <i>breadboard</i>
---	---------------------	--------------------------------------------------

Sistem kelas terpadu ini pun memiliki aplikasi yang dapat digunakan sebagai penghubung antara peranti dengan pengguna sehingga pengguna dapat lebih mudah dalam menggunakan data yang dihasilkan dari peranti sistem kelas terpadu ini. Dalam pembuatannya, terdapat beberapa perangkat lunak yang digunakan untuk mengembangkan aplikasi sistem kelas terpadu ini. Tabel 3.2 merupakan beberapa perangkat lunak yang digunakan.

Tabel 3.2. Perangkat lunak yang digunakan pada aplikasi sistem kelas terpadu.

No	Nama Perangkat Lunak	Keterangan
1	Golang	Bahasa pemrograman utama sisi belakang (<i>backend side</i>)
2	PostgreSQL	<i>Relational database</i> yang menggunakan bahasa pemrograman sequel
3	Redis	<i>Nonrelational database</i> yang digunakan pada struktur dalam memori sebagai <i>cache</i>
4	Javascript	Bahasa pemrograman utama sisi depan (<i>frontend side</i>)
5	React	Kerangka kerja yang dibangun di atas bahasa javascript yang digunakan untuk membangun sisi depan (<i>frontend side</i>)
6	TablePlus	Antarmuka pengguna berbasis grafis untuk manajemen <i>relational database</i>
7	Insomnia	Penguji REST API
8	Visual Studio Code	Penyunting kode

b. Analisis Sistem

Ketika kebutuhan pengembangan aplikasi sudah ada dalam sebuah dokumentasi, analisis terhadap dokumentasi tersebut akan dilakukan. Analisis ini akan menghasilkan penjelasan lebih rinci dari *requirement* yang sudah didokumentasikan. Hasil analisis dapat berupa fungsional dan nonfungsional. Analisis fungsional berhubungan dengan maksud, tujuan, ruang lingkup, fungsi, atribut software, kebutuhan *database*, dan lain-lain. Analisis fungsional berkaitan

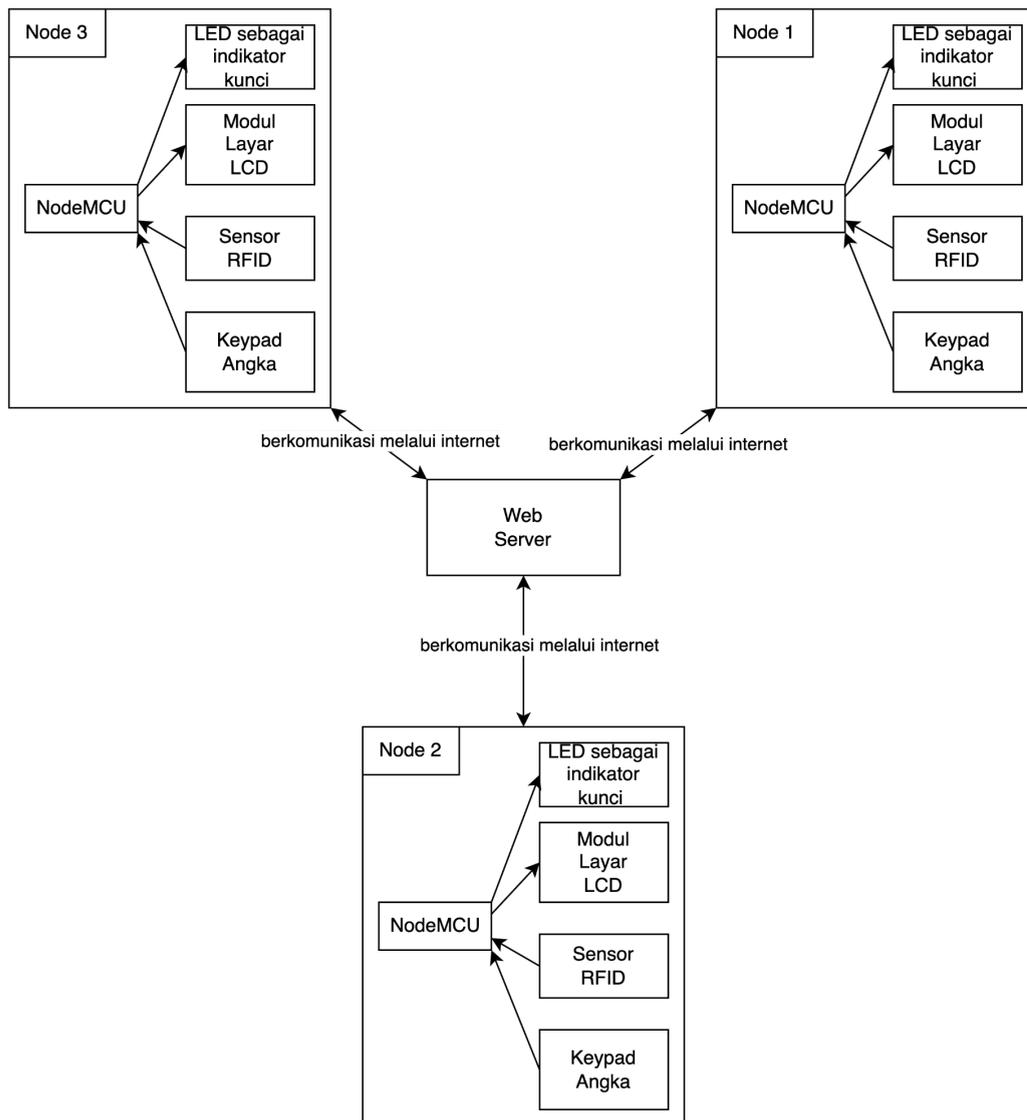
erat dengan pengembangan aplikasi secara penulisan kode dan secara keteknikan. Pada analisis nonfungsional, hasilnya akan berhubungan dengan berbagai kriteria, kesepakatan, aturan, batasan, dan lain-lain. Analisis nonfungsional dimaksudkan untuk menjelaskan properti terkait keandalan, ketahanan, kemampuan, pemeliharaan, dan skalabilitas.

Kebutuhan sistem yang sudah ditentukan pada awal perencanaan dan pengembangan sistem dianalisis dan hasilnya dibuat dalam bentuk dokumen tertentu. Pada penelitian ini, dokumen tersebut bersifat tidak baku sehingga spesifikasi sistem yang akan dibuat akan disesuaikan dengan kebutuhan fitur-fitur sistem kelas terpadu ini.

Hasil akhir dari sistem ini akan berupa peranti untuk merekap kehadiran pada kelas sesuai dengan identifikasi kartu RFID dan sebuah aplikasi yang menjadi pengguna antarmuka yang memudahkan staf dalam mengatur jadwal dan rekap presensi. Peranti yang dibuat akan berupa prototipe yang dibuat menggunakan *breadboard* dan kabel *jumper*. Selain itu, aplikasi yang digunakan sebagai pengguna antarmuka tidak perlu di-*deploy* pada *production environment*, seperti AWS atau *cloud service* lainnya. Oleh karena itu, aplikasi tersebut hanya akan digunakan pada komputer lokal pada pengembangan dan pengujiannya. Analisis ini dilakukan berdasarkan riset yang dilakukan dari dokumentasi resmi pengembang perangkat keras dan perangkat lunak yang digunakan untuk membuat sistem kelas terpadu ini.

c. Desain Sistem

Fase ini menentukan gambar besar tentang bagaimana tujuan dari aplikasi tersebut dapat dicapai. Desain aplikasi dibuat berdasarkan perancangan arsitektur, algoritma, alur kerja, konsep database, dan desain grafis sebagai *user interface*. Proses ini akan menghasilkan perancangan dan pemecahan masalah untuk sebuah perangkat keras dan perangkat lunak.



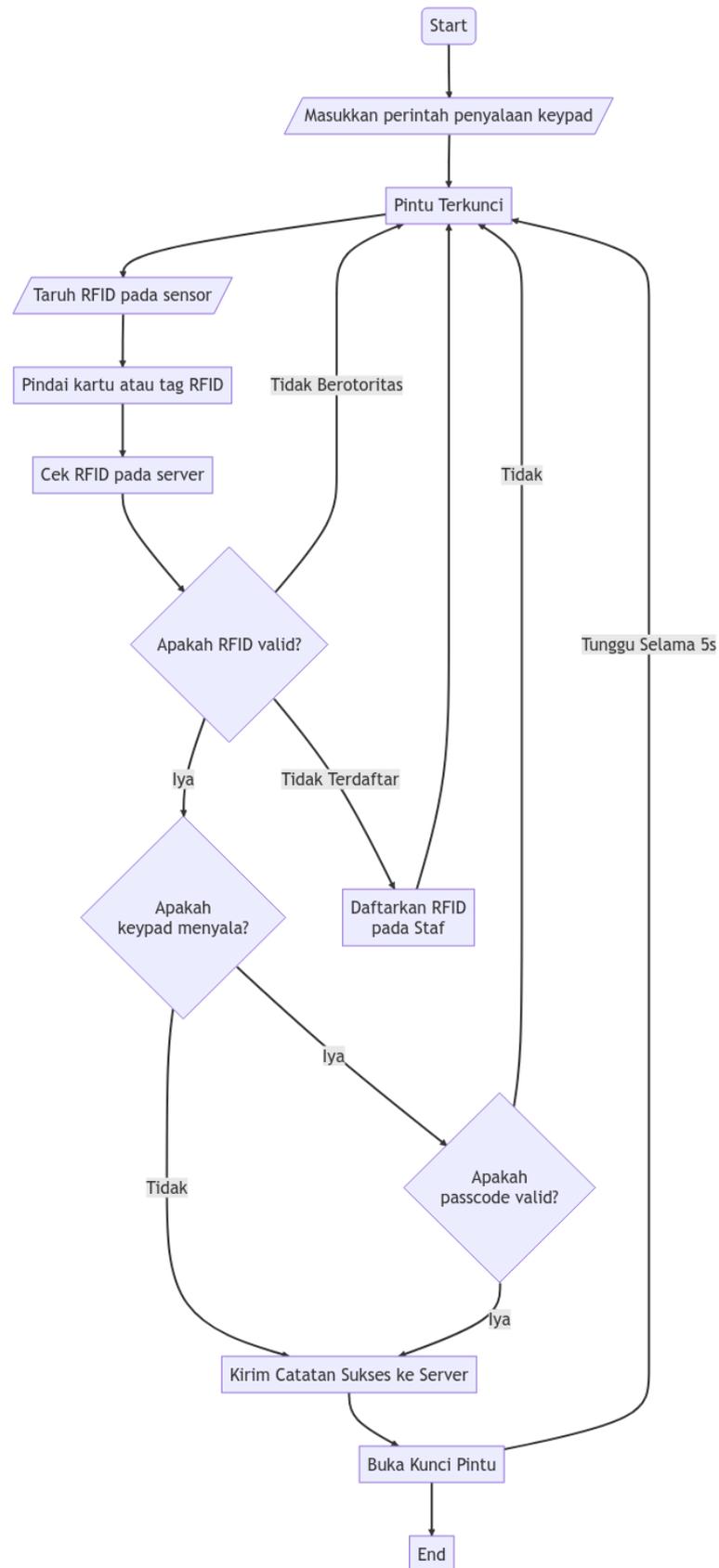
Gambar 3.2. Desain peranti sistem kelas terpadu.

Pembuatan sistem ini akan melibatkan pengintegrasian antara aplikasi, yang diantaranya merupakan *server*, dengan peranti yang dirancang untuk merekap kehadiran mahasiswa pada kelas yang dipasang peranti tersebut. Pada Gambar 3.2, tiap node memiliki komponen perangkat keras yang serupa dengan rangkaian yang sama. Seluruh komponen tersebut terhubung dan diatur menggunakan mikrokontroler NodeMCU. NodeMCU akan menerima data dari RFID *sensor* dan *number keypad*. Data tersebut kemudian akan disusun dan dikirimkan menuju *server* untuk proses validasi. Setelah data tersebut tervalidasi, responsnya kemudian akan dikirim kembali pada NodeMCU. Respons tersebut

akan ditampilkan pada layar LCD. NodeMCU pun akan mengirimkan sinyal terhadap LED sebagai indikator dari *solenoid door lock* sesuai dengan respons yang dikirimkan dari *server*.

Gambar 3.3 menjelaskan bagaimana alur dari sistem tersebut dari sisi peranti menuju *server*. Sebelum peranti dapat bekerja, peranti akan meminta konfirmasi penggunaan *keypad*. Kemudian, peranti akan dalam keadaan terjaga untuk selalu siap dalam mendeteksi setiap kartu RFID. Kartu RFID akan memiliki nomor identifikasi unik yang kemudian akan divalidasi terlebih dahulu. Validasi tersebut tidak hanya dilakukan untuk mengecek status pendaftaran dari kartu RFID tersebut pada jadwal yang sedang berlangsung. Akan tetapi, validasi tersebut juga akan menjalankan pengecekan terhadap ada atau tidaknya jadwal pada kelas tersebut. Ketika kartu RFID sudah tervalidasi, *server* akan mengirimkan response sukses dengan menyisipkan enam kode kombinasi angka (*passcode*) yang terdapat pada *database*. Jika peranti diatur untuk tidak menggunakan *keypad* dalam pemakaiannya, maka *passcode* tidak akan digunakan. Sedangkan jika peranti diatur untuk menggunakan *keypad*, maka pengguna akan diminta untuk memasukkan *passcode* tersebut. *Passcode* dimaksudkan untuk validasi kepemilikan kartu RFID pengguna. NodeMCU akan menerima masukan kode tersebut dan memvalidasinya pada mikrokontroler tanpa perlu mengirimkan permintaan pada *server*. Apabila *passcode* yang dimasukkan pengguna tepat, NodeMCU akan mengirimkan sinyal pada LED sebagai indikator bahwa pintu akan terbuka dan NodeMCU akan mengirimkan permintaan untuk menyimpan data kehadiran menuju *server*.

Proses-proses tersebut terjadi untuk setiap pengguna yang hendak masuk ke dalam kelas. Ketika pengguna selanjutnya hendak masuk ke dalam kelas yang sama, pengguna tersebut harus menunggu selama 5 detik hingga peranti kelas terpadu siap untuk memindai kartu RFID. Selama 5 detik tersebut, peranti kelas terpadu akan tetap membuka kunci pintu solenoid yang ditandai oleh nyalanya LED.



Gambar 3.3. Diagram alir peranti sistem kelas terpadu.

d. Implementasi Sistem

Tahap implementasi merupakan tahap perwujudan dari aplikasi yang hendak dibuat. Di tahap ini, penulisan kode dibuat dan dikompilasi menjadi sebuah aplikasi yang dapat digunakan. Dengan kata lain, ini merupakan tahap konversi dari *requirements* yang sudah dibuat menjadi komponen aplikasi dan program yang dapat dieksekusi.

Pada tahap ini, kode program untuk peranti yang menggunakan NodeMCU dan kode program pada aplikasi, bagian depan (*frontend* side) dan bagian belakang (*backend* side), sudah selesai dibuat. Selain itu, pada tahap ini pun sudah dilakukan pengujian pada aplikasi menggunakan *unit test* dan *integration test*. *Unit test* dilakukan pada aplikasi bagian *backend* dengan cukup mudah karena bahasa pemrograman Go sudah menyediakan dokumentasi untuk pengujian unit. Sedangkan untuk *integration test*, perangkat lunak yang digunakan adalah Insomnia. Selain itu, pengujian secara manual melalui aplikasi bagian *frontend* pun dilakukan di tahap ini.

e. Pemeliharaan Sistem

Pemeliharaan dilakukan pada sebuah aplikasi dengan maksud untuk menjaga keandalan agar aplikasi dapat terus berjalan dengan semestinya. Bentuk pemeliharaan yang dilakukan dapat berupa perubahan kode, pengintegrasian teknologi, perbaikan galat, dan peningkatan performa. Perlu diketahui bahwa pengujian dilakukan tidak di tahap ini saja. Pengujian dilakukan pada setiap tahap pengembangan, khususnya dari awal penulisan kode. Hal tersebut dilakukan untuk meminimalisir kesalahan yang dapat terjadi pada alur kerja aplikasi.

f. Penyelesaian Sistem

Tahap akhir dari penyelesaian sistem pada metode *waterfall* adalah dengan membiarkan sistem tersebut berjalan sendiri tanpa perlu mengadakan peningkatan. Selain itu, pada tahap ini juga akan dilakukan peniadaan

komponen-komponen yang sudah tidak diperlukan dari sistem yang sudah dibuat, baik pada peranti maupun pada kode aplikasi.