

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Denyut nadi merupakan frekuensi irama denyut/detak jantung yang dapat diukur dengan cara meraba permukaan kulit pada tempat-tempat tertentu. Kecepatan denyut nadi dapat bergantung pada bermacam macam kondisi seperti pada waktu sedang istirahat, bekerja, olahraga, emosi, dan lain-lain [1]. Pada dunia kedokteran, denyut nadi dapat digunakan untuk menganalisa atau mengindikasikan adanya kelainan dari jantung. Selanjutnya, dokter akan mencatat hasil dari pengukuran denyut nadi berikut dengan data data yang menyangkut dengan identitas pasien mulai dari pemeriksaan, hasil diagnose, riwayat penyakit, dan pengobatan serta tindakan dan pelayanan lain kepada pasien selama mendapatkan perawatan dari pihak penyedia layanan baik rawat jalan maupun rawat inap pada dokumen pasien atau yang sering disebut dengan rekam medik (*medical record*). Bentuk rekam medik yang umum ditemui berupa berkas kertas beserta lampiran-lampiran dokumen yang tidak sederhana. Dokumen rekam medik tersebut dapat membantu dokter untuk memberikan penanganan yang tepat jika terjadi keadaan darurat yang disebabkan oleh serangan jantung. Namun, hal ini akan memberikan suatu permasalahan yang terjadi apabila pasien sedang berada di luar kota dan tidak membawa dokumen rekam medik sedangkan pasien membutuhkan tindakan medis dengan cepat. Kesulitan akses terhadap dokumen rekam medik ini akan menyulitkan dokter sehingga dapat meningkatkan resiko keterlambatan penanganan.

Atas dasar permasalahan tersebut, muncul sebuah ide untuk membuat sebuah sistem aplikasi yang dapat membantu dokter melihat denyut nadi dari alat *wearable device* dan melihat data rekam medik yang sudah tersimpan pada database sistem.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang diuraikan sebelumnya, maka dapat disimpulkan masalah yang terjadi sebagai berikut :

1. Bagaimana membangun sebuah sistem *monitoring* denyut nadi dan rekam medik yang dapat diakses baik secara *offline* maupun *online*?
2. Bagaimana mengakses dan mengolah denyut nadi pasien dari *wearable device* yang tersimpan didalam *database* untuk ditampilkan pada aplikasi?
3. Bagaimana data rekam medik dapat dilihat oleh pihak terkait sesuai dengan hak akses masing-masing?

1.3 Tujuan

Adapun tujuan dari proyek akhir ini adalah sebagai berikut :

1. Merancang sistem aplikasi yang dapat diakses melalui jaringan internet agar sistem dapat tersinkron dengan *database* pusat secara periodik.
2. Menampilkan dan menyimpan data rekam medik dan data denyut nadi yang sebelumnya diperoleh dari *wearable device* pada aplikasi desktop.
3. Membuat sebuah *user interface* yang dapat menampilkan informasi sesuai dengan hak akses pengguna.

1.4 Batasan Masalah

Adapun batasan masalah dari proyek akhir ini adalah sebagai berikut :

1. Input aplikasi *desktop* ini berasal dari *database*.
2. Aplikasi menampilkan rincian data denyut nadi, rata rata denyut nadi dalam periode tertentu (harian).
3. Aplikasi hanya menampilkan, menambahkan dan mengubah data rekam medik.
4. Data rekam medik hanya dapat dilihat dan diubah melalui aplikasi.
5. Proyek akhir ini tidak membahas mengenai sistem keamanan pada *database*.

1.5 Definisi Operasional

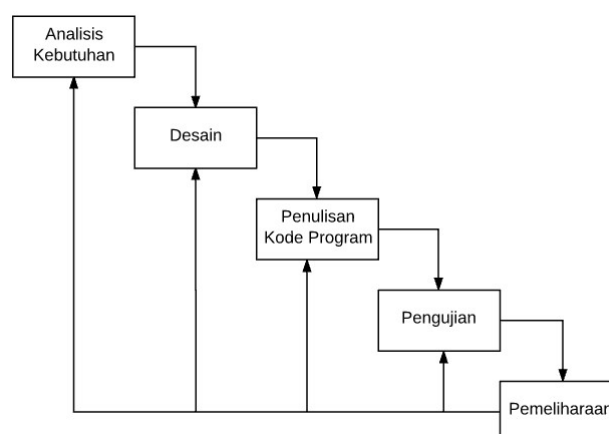
Denyut nadi merupakan frekuensi irama denyut/detak jantung yang dapat diukur dengan cara meraba permukaan kulit pada tempat-tempat tertentu. Kecepatan denyut nadi dapat bergantung pada bermacam macam kondisi seperti pada waktu sedang istirahat, bekerja, olahraga, emosi, dan lain-lain. Pada sistem yang akan dibangun, denyut nadi berasal dari perangkat *wearable device* yang telah dibuat pada penelitian sebelumnya.

Pembuatan sistem monitoring denyut nadi bertujuan agar pihak terkait dapat melihat rekap data denyut nadi pasien dan digunakan untuk menganalisa apabila terdapat kelainan pada jantung. Selain itu, terdapat data rekam medik pasien yang dapat dilihat, diubah maupun dihapus. Hal ini dapat meminimalisir keterlambatan penanganan pasien apabila data rekam medik tertinggal ataupun hilang.

Sistem ini dibangun menggunakan Aplikasi Microsoft Visual Studio dengan Bahasa pemrograman *c#*. Untuk pengolahan data pada sistem ini menggunakan Aplikasi *Mysql* dan sedangkan untuk inputan data denyut nadi berasal dari *database* yang diperoleh dari Aplikasi *Firebase*.

1.6 Metode Pengerjaan

Metode *waterfall* merupakan metode yang sering digunakan oleh penganalisa sistem pada umumnya. Inti dari metode *waterfall* adalah pengerjaan dari suatu



Gambar 1 - 1 Alur Pengerjaan Metode Waterfall

sistem dilakukan secara berurutan atau secara linear. Jadi jika langkah ke-1 belum dikerjakan, maka langkah ke-2 tidak dapat dikerjakan. Jika langkah ke-2 belum dikerjakan maka langkah ke-3 juga tidak dapat dikerjakan, begitu seterusnya. Secara otomatis langkah ke-3 akan dapat dilakukan jika langkah ke-1 dan ke-2 sudah dilakukan. Secara garis besar metode *waterfall* mempunyai langkah-langkah Analisa, Desain, Penulisan, Pengujian dan Penerapan serta Pemeliharaan [2].

1.7 Jadwal Pengerjaan

Adapun beberapa struktur jadwal pengerjaan yang akan dilakukan untuk pengerjaan proyek akhir ini adalah sebagai berikut :

1.7.1 Studi Literatur

Mempelajari tentang teori yang berhubungan dengan proyek akhir yang sedang dikerjakan seperti konsep *Database*, *Aplikasi Desktop* dan lain-lain.

1.7.2 Pembuatan Proposal

Tahapan ini merupakan tahapan untuk mereancang dan membuat proposal

1.7.3 Perancangan dan pengolahan data

Tahapan ini bertujuan untuk merancang gambaran topologi dan pengolahan data yang dibutuhkan dalam proyek akhir ini.

1.7.4 Uji coba tampilan dari data

Proses penerapan sistem yang akan dibangun dan yang untuk diujikan.

1.7.5 Pengapikasian dan pemeliharaan

Penerapan sistem saat selesai dibangun dan pemeliharaan sistem

1.7.6 Dokumentasi dan pelaporan

Pada tahapan ini semua yang telah dikumpulkan dan dilakukan pengujian dapat dibuat menjadi dokumentasi berupa laporan proyek akhir atau PA.

Tabel 1 - 1 Jadwal Pengerjaan Proyek Akhir

No	Kegiatan	Pengerjaan Proyek Akhir 2017-2018							
		Okt	Nov	Des	Jan	Feb	Mar	Apr	Mei
1	Studi Literatur								
2	Pembuatan Proposal								
3	Perancangan dan pengolahan data								
4	Uji coba tampilan dari data								
5	Pengaplikasian dan pemeliharaan								
6	Dokumentasi dan pelaporan								