

BAB II

TINJAUAN PUSTAKA

2.1 Tuberkulosis

Menurut Sejati (2015), tuberkulosis adalah salah satu penyakit menular, kronik yang diakibatkan oleh kuman *Mycobacterium tuberculosis*, yang disertai dengan adanya jaringan granulasi nekrotik (perkijauan) sebagai respons terhadap kuman tersebut. Penularan penyakit ini sangat cepat dan mudah, terutama bagi orang yang rentan dan daya tahan tubuh lemah. Tuberkulosis merupakan penyakit yang menginfeksi organ paru, yang membutuhkan jangka waktu pengobatan cukup panjang dalam mencapai kesembuhan.

Masih menurut Sejati (2015), Pasien TB menjalani rangkaian pengobatan selama 6 bulan secara rutin dan teratur. Jika pasien TB berhenti dalam menjalani rangkaian pengobatan maka jangka waktu pengobatan akan bertambah menjadi 2 tahun, karena pasien TB akan mengalami TB-MDR (Tuberkulosis Multi Medicine Resistance) yaitu sebuah situasi penyakit tersebut menjadi lebih sulit untuk diobati. Kasus Tuberkulosis umumnya terjadi pada kelompok masyarakat dengan golongan sosial ekonomi rendah.

2.2 Health Monitoring System

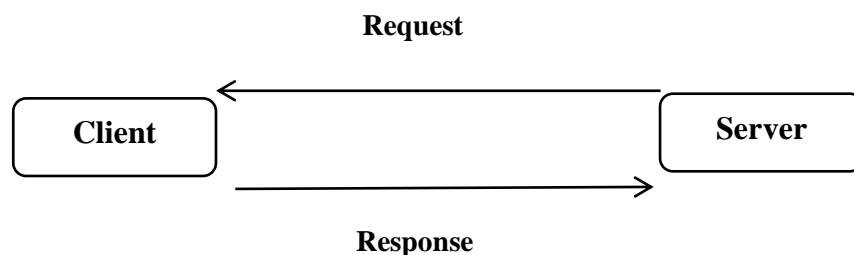
Skripsi ini membangun sistem monitoring kepatuhan pasien TB terhadap tata laksana pengobatan, dengan kata lain sistem ini disebut juga health monitoring system (HMS). Pada penelitian (Taryudi, 2019) mendefinisikan sebagai sebuah cara yang digunakan untuk menganalisa informasi terhadap implementasi suatu program kesehatan yang ditujukan kepada pasien TB. Pada kegiatan ini health monitoring system bertujuan untuk memonitor pasien penderita TB sehingga dapat mengetahui aktifitas menjalani pengobatan.

2.3 Kontroling

Hal utama dalam skripsi ini merupakan kontrolling (pengawasan) terhadap pasien TB. Pada tahun 2019, Nainggolan menjabarkan kontrolling (pengawasan) merupakan aktifitas manajemen yang berhubungan dengan pemeriksaan untuk bertujuan apakah perencanaannya telah dilakukan sesuai dengan perencanaan, telah sampai sejauh mana perkembangan yang sudah dilakukan, dan perencanaan yang belum terlaksanakan sesuai rencana.

2.4 Metode Client Server

Rancang bangun aplikasi sistem pengingat dan penjadwalan pasien TB menggunakan metode client server. Pada tahun 2014, Soepomo mendefinisikan metode client server merupakan salah satu metode dalam struktur aplikasi dimana sitem aplikasi menjalankan request dan action melalui layanan dari suatu penyedia layanan. Client adalah aplikasi yang bergerak pada komputer client serta melayani interaksi berupa request ke server. Algoritma metode client server ditunjukkan oleh gambar 1 berikut :



Gambar 2.1 Algoritma metode client server

2.5 Sistem Android

Dalam perancangan perangkat lunak sistem kontrol pasien TB menggunakan sistem android sebagai platform perancangan perangkat lunak. Pada tahun 2018, Mulyadi dan Dale menjelaskan bahwa sistem android merupakan sistem operasi yang pengembangannya dipimpin oleh Google untuk perangkat mobile.

Distributor sistem operasi Android di dunia terbagi menjadi dua jenis, yaitu Open Handset Distribution (OHD) yang bebas dan tanpa dukungan langsung Google dan Google Mail Services (GMS) yang didukung penuh oleh Google.

2.6 Firebase

Dalam perancangan perangkat lunak sistem kontrol pasien TB menggunakan tools Firebase sebagai server. Pada tahun 2017, M.Ilhami menjelaskan Firebase Realtime Database merupakan cloud database yang data disimpan dalam format JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika membangun aplikasi hybrid lintas platform, seperti Android dan iOS maka semua klien berbagi satu instance Realtime Database dan secara otomatis menerima pembaruan dengan data tertentu. Firebase Realtime Database adalah basis data NoSQL dan karena itu memiliki optimalisasi dan fungsionalitas yang berbeda dibandingkan dengan basis data relasional. Membuat database Firebase bisa melalui import file JSON ke konsol Firebase, atau dapat juga dibuat langsung melalui halaman konsol Realtime Database secara manual.

2.7 Android

Jenis platform yang digunakan dalam pembuatan perangkat lunak sistem kontrol pasien TB menggunakan platform android. Pada tahun 2019, Sari dan Ali menjelaskan bahwa Android adalah sistem operasi untuk telepon seluler dan tablet. Sistem operasi dapat digunakan sebagai penghubung antara piranti dan pengguna, sehingga pengguna dapat berinteraksi dengan piranti (*device*) dan dapat menjalankan aplikasi-aplikasi yang tersedia pada *device*.

Dari uraian di atas dapat di simpulkan bahwa Android merupakan sistem operasi yang di kembangkan untuk perangkat mobile seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh berbagai macam piranti bergerak.

2.8 Android Studio

Android Studio adalah sebuah IDE yang bisa digunakan untuk pengembangan aplikasi Android, dan dikembangkan oleh google. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android Studio Di rencanakan untuk menggantikan Eclipse ke depannya sebagai IDE resmi untuk pengembangan aplikasi Android (Makiolor dkk, 2017).

2.9 Android SDK (Software Development Kit)

Android SDK adalah tools API (Aplication Programming Interface) yang diperlukan untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Beberapa fitur-fitur Android yang paling penting adalah mesin Virtual Dalvik yang dioptimalkan untuk perangkat mobile, integrated browser berdasarkan engine open source WebKit, Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1.0 (Opsional akselerasi perangkat keras), kemudian SQLite untuk penyimpanan data (database). Fitur-fitur android lainnya termasuk media yang mendukung audio, video, dan gambar, juga ada fitur bluetooth, EDGE, 3G dan WiFi, dengan fitur kamera, GPS, dan kompas.(Ryanda, dkk. 2015).

2.10 Java

Penggunaan bahasa pemrograman dalam pembuatan perangkat lunak sistem pengingat dan penjadwalan pasien TB menggunakan bahasa pemrograman java sebagai tahap pengkodean perangkat lunak. Pada tahun 2018, Sibarani mendefinisikan java adalah bahasa pemrograman yang populer, dikembangkan oleh Sun Microsystems. Salah satu penggunaan terbesar Java adalah dalam pembuatan aplikasi native untuk android. Bahasa pemrograman ini bersifat multiplatform yakni bahasa ini dapat digunakan di berbagai platform, seperti desktop, android dan bahkan untuk sistem operasi Linux. Beberapa ciri dari bahasa pemrograman ini adalah sebagai berikut :

1. Object oriented language
2. Multithreading
3. Garbage collector support
4. Statically Typed
5. Multiplatform

2.11 Push Notification

Perancangan perangkat lunak sistem pengingat dan penjadwalan pasien TB memiliki fitur berupa push notifikasi sebagai pengingat untuk pasien TB. Pada tahun 2018, Sidik dan Nasution mendefinisikan menjelaskan push notification merupakan sebuah layanan yang banyak digunakan untuk keperluan pemberitahuan melalui pesan pendek yang ada di smartphone. Dengan adanya layanan Push Notification tersebut, pengguna dapat terbantu dalam hal yang bersipat pemberitahuan secara singkat. Pada implementasinya Push Notification dapat dimanfaatkan dalam berbagai keperluan sehari-hari misalnya untuk monitoring absensi, update berita terbaru, dan sebagainya.

Aplikasi yang akan dirancang adalah sebuah aplikasi yang dapat mengirim Push Notification yang nantinya akan dapat dikembangkan di berbagai bidang sesuai dengan kebutuhan pengguna. Kurangnya pengetahuan dalam memberikan informasi secara real time, berakibat informasi yang tersampaikan tidak terbaru, sehingga dalam berbagai situasi dan kondisi informasi yang diberikan telah usang.

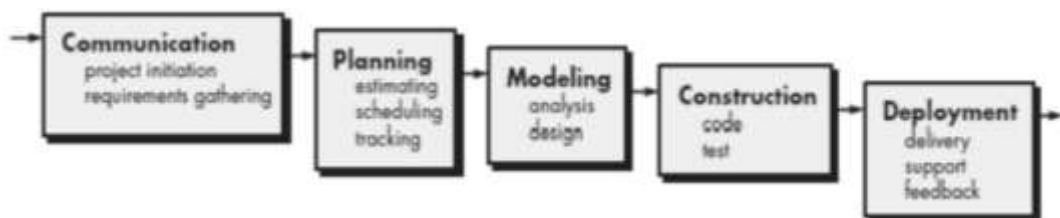
Push Notification salah satu layanan yang dapat menjawab masalah tersebut sehingga tidak ada lagi informasi yang terbaru tidak tersampaikan, dengan penggunaan layanan ini setiap terjadi update informasi maka akan langsung terkirim sebagai pesan notification, sehingga informasi yang terbaru tidak akan terlewatkan. Layanan Push Notification umumnya banyak diterapkan pada aplikasi mobile seperti Android dan IOS. Untuk penggunaan Sistem Operasi mobile terbesar berdasarkan statcounter untuk tahun 2012 sampai 2016 di Indonesia dikuasi oleh Sistem Operasi

2.12 Metode Pengembangan Perangkat Lunak

Metode pengembangan sistem Informasi adalah metode yang digunakan dalam mengembangkan sistem berdasarkan dengan proses-proses yang berjalan secara runtut hingga didapatkan sistem yang sesuai.

2.12.1 Metode Waterfall

Pada penelitian ini, metode perencanaan aplikasi yang digunakan adalah *Waterfall*. Model *Waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*, dimana proses pengerjaannya bertahap dan harus menunggu tahap sebelumnya selesai dilaksanakan kemudian memulai tahap selanjutnya (Tujni dan Hutrianto, 2020). Berikut adalah tahapan-tahapan dari metode *Waterfall* menurut (Pressman, 2012).



Gambar 2.2 Metode Waterfall

a. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan perangkat lunak, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan objek penelitian, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

b. *Planning*

Proses planning merupakan lanjutan dari proses communication (analysis requirement). Tahapan ini akan menghasilkan dokumen perencanaan proses

pengembangan perangkat lunak atau bisa dikatakan sebagai data yang berhubungan dengan proses pengerjaan.

c. Modeling

Proses modeling ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dilakukan pengkodean perangkat lunak. Proses ini berfokus pada rancangan struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural.

d. Construction

Construction merupakan proses membuat coding atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan kebutuhan ke dalam bentuk antarmuka perangkat lunak melalui pengkodean. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu perangkat lunak, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan pengujian terhadap sistem yang telah dibuat. Tujuan pengujian adalah menemukan kesalahankesalahan terhadap perangkat lunak untuk kemudian bisa diperbaiki.

e. Deployment

Tahapan ini bisa dikatakan final dalam pembuatan sebuah perangkat lunak. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh user. Kemudian perangkat lunak yang telah dibuat harus dilakukan pemeliharaan secara berkala.

2.13 Unified Modeling Language (UML)

Dalam pembuatan perangkat lunak sistem kontrol pasien TB memerlukan sebuah desain *Unified Modelling Language (UML)* sebagai proses yang dibutuhkan dalam membangun perangkat lunak. Pada tahun 2011, Havaluddin menjelaskan bahwa Unified Modelling Language (UML) adalah sebuah “bahasa” yg telah

menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah system. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (Object Modeling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering).


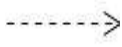
2.13.1 Use Case Diagram

Diagram yang menggambarkan actor, use case dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah use case digambarkan sebagai elips horizontal dalam suatu diagram UML use case. Use Case memiliki dua istilah




1. System use case : interaksi dengan sistem.
2. Business use case : interaksi bisnis dengan konsumen atau kejadian nyata.

Penjelasan simbol Use Case dapat dilihat pada tabel 2.1

Tabel 2.1 *Use Case Diagram.*

No	Gambar	Nama	Keterangan
1		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		Include	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).


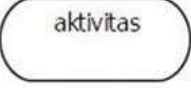
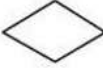
Tabel 2.1 *Use Case Diagram* (lanjutan)

3		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
5		Use Case	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.



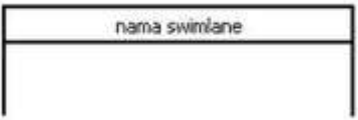
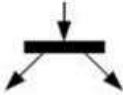

2.13.2 Activity Diagram

Activity diagram menggambarkan rangkaian aktifitas-aktifitas, objek, state, transisi state dan event. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas. *Activity diagram* digambarkan dengan simbol-simbol yang setiap simbolnya memiliki makna dan tujuan (Haviluddin, 2011). Untuk keterangan simbol *activitydiagram* dapat dilihat pada tabel 2.2.

Tabel 2.2 *Activity Diagram*.

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.







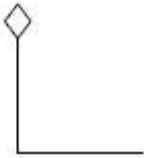
Tabel 2.2 *Activity Diagram* (lanjutan)

Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
<i>Fork</i> 	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel.
<i>Join</i> 	Digunakan untuk menunjukkan kegiatan yang digabungkan.

2.13.3 Class Diagram

Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. Class diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, class diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat (Haviluddin, 2011). Keterangan simbol-simbol dapat dilihat pada tabel 2.3 :


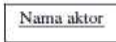

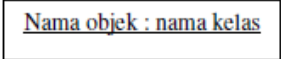

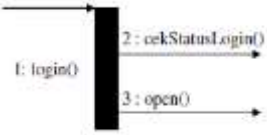
Tabel 2.3 *Class Diagram*.

No.	Simbol	Deskripsi
1	<p>Kelas</p> 	Kelas pada struktur sistem.
2	<p>Antarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	<p>Asosiasi/<i>association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	<p>Asosiasi berarah/<i>directed association</i></p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umumkhusus).
6	<p>Kebergantungan/<i>dependensi</i></p> 	Relasi antar kelas dengan makna kebergantungan antar kelas
7	<p>Aggrgasi/<i>aggregation</i></p> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

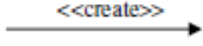
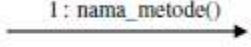
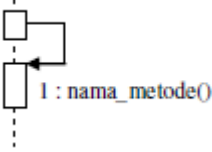

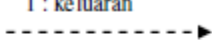
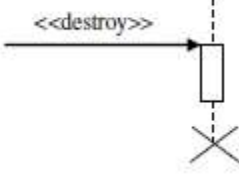
2.13.4 Sequence Diagram

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya sequence diagram adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan use case diagram (Haviluddin, 2011). Simbol-simbol yang ada pada diagram sekuen dapat dilihat pada tabel 2.4 :

Tabel 2.4 *Sequence Diagram.*

No	Simbol	Deskripsi
1	<p>Aktor</p>  <p>Atau</p>  <p>Tanpa waktu aktif</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.
2	<p>Garis hidup/<i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.
3	<p>Objek</p> 	Menyatakan objek yang berinteraksi pesan.
4	<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semuanya yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya :</p>  <p>Maka cekStatusLogin() dan open() dilakukan didalam metode login(). Aktor tidak memiliki waktu aktif.</p>

Tabel 2.4 *Sequence Diagram* (lanjutan)

5	<p>Pesan tipe <i>create</i></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.</p>
6	<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <p>Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.</p>
7	<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
8	<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.</p>
9	<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i></p>

2.14 Pengujian *Black Box*

Black-Box Testing merupakan Teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Blackbox Testing bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan pada informasi domain. Blackbox Testing memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program (Jaya, 2018).

2.15 Penelitian terkait

Pada tahun 2019, Nainggolan, J., Iswar, F. J., & Abraham, A menghasilkan penelitian yang berjudul KONTROL OPTIMAL PADA PEYEBARAN TUBERKULOSIS DENGAN EXOGENOUS REINFECTION. Penelitian ini membahas tentang upaya pengawasan kasus penyebaran penyakit tuberkulosis sehingga dapat mengetahui grafik naik dan turun kasus penyebaran penyakit tuberkulosis. Tetapi penelitian ini belum menggunakan media teknologi informasi sebagai sarana alat kontrol pasien TB.

Sedangkan Setiyadi (2019) menghasilkan penelitian yang berjudul PENERAPAN APLIKASI SISTEM INFORMASI SPASIAL UNTUK PENGELOLAAN PROGRAM TUBERKULOSIS DI PUSKESMAS GATAK DAN KARTASURA KABUPATEN SUKOHARJO. Pada sistem ini bertujuan sebagai alat bantu dalam Pengelolaan dalam Penyajian data kasus tuberkulosis per puskesmas Secara Akurat dan berbasis Spasial. Sistem informasi ini belum dapat menyelesaikan masalah pasien penderita TB dalam hal pengobatan penyakit tuberkulosis.

Pada tahun 2017, Enggi Ledy Asri, K menghasilkan penelitian yang berjudul ANALISA DAN PERANCANGAN SISTEM INFORMASI DIAGNOSA PENYAKIT PARU-PARU MENGGUNAKAN APLIKASI VB 6.0. Sistem ini mempermudah pasien dalam mengetahui jenis penyakit paru yang di derita.

Namun sistem informasi ini belum dapat menyelesaikan masalah tentang solusi pengobatan penyakit sebagai alat bantu.

Hayurani, H., & Hartanti, F. D (2016) menghasilkan penelitian tentang Sistem monitoring dan kontrolling pasien tuberkulosis (TB) berbasis web merupakan sebuah sistem yang digunakan dalam memantau pengawasan pasien TB. Perancangan sistem monitoring dan kontrolling pasien tuberkulosis mampu menghasilkan informasi mengenai TB, mengenai penyebaran pasien dan fasilitas layanan kesehatan menggunakan media peta lokasi, melihat grafik visualisasi pasien penderita TB juga laporan pendataan lengkap seperti data kader dan data penderita tuberkulosis. Tetapi pada sistem ini belum dapat menyelesaikan masalah tentang alat bantu pengobatan pasien TB.

Sedangkan Musa, M. I. (2017) menghasilkan penelitian yang berjudul APLIKASI MONITORING UNTUK PASIEN TBC DEWASA BERBASIS ANDROID. Sistem aplikasi ini dapat memantau kondisi pasien TB serta sebagai media kontrol untuk minum obat. Kelemahan sistem ini adalah belum adanya alat berupa notifikasi sebagai pengingat ketika minum obat, serta belum adanya fitur jadwal untuk bertemu dengan dokter.