

**SISTEM PENDETEKSI BAKTERI MYCOBACTERIUM  
TUBERCULOSIS DAN PENENTU TINGKAT  
PENYAKIT TBC**



N a m a : Ridho Imam Pratama  
NIM : 15523021

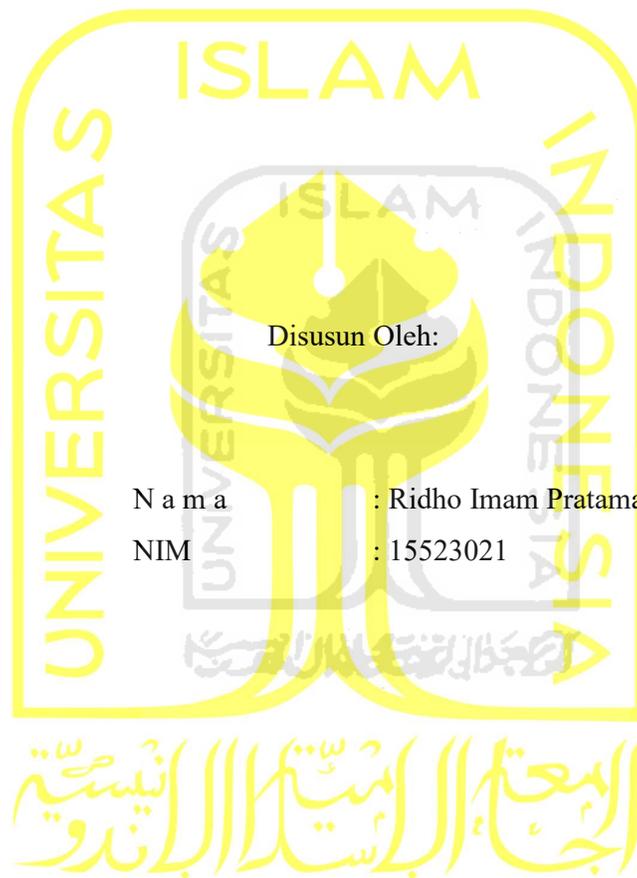
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA**

**2020**

**HALAMAN PENGESAHAN DOSEN PEMBIMBING**

**SISTEM PENDETEKSI BAKTERI MYCOBACTERIUM  
TUBERCULOSIS DAN PENENTU TINGKAT  
PENYAKIT TBC**

**TUGAS AKHIR**



Yogyakarta, 26 Oktober 2020

Pembimbing 1,

Pembimbing 2,

( Izzati Muhimmah, S.T., M.SC., Ph.D. )

( Arrie Kurnawardhani, S.SI., M.Kom. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

**SISTEM PENDETEKSI BAKTERI MYCOBACTERIUM  
TUBERCULOSIS DAN PENENTU TINGKAT  
PENYAKIT TBC**

**TUGAS AKHIR**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk  
memperoleh gelar Sarjana Komputer dari Program Studi Informatika  
di Fakultas Teknologi Industri Universitas Islam Indonesia  
Yogyakarta, 26 Oktober 2020

Tim Penguji

Izzati Muhimmah, S.T., M.SC., Ph.D.

**Anggota 1**

Aridhanyati Arifin, S.T., M.CS.

**Anggota 2**

Septia Rani, S.T., M.CS.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



( Dr. Raden Teduh Dirgahayu, S.T., M.Sc. )

**HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR**

Yang bertanda tangan di bawah ini:

Nama : Ridho Imam Pratama

NIM : 15523021

Tugas akhir dengan judul:

**SISTEM BAKTERI MYCOBACTERIUM  
TUBERCULOSIS DAN PENENTU TINGKAT  
PENYAKIT TBC**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 26 Oktober 2020



( Ridho Imam Pratama )

## HALAMAN PERSEMBAHAN

Tugas akhir ini penulis persembahkan kepada seluruh keluarga serta kerabat yang telah memberikan bantuan, dukungan serta nasihat kepada penulis, sehingga penulis mampu untuk menyelesaikan tugas akhir ini.



## HALAMAN MOTO

*“Pengalaman merupakan seorang guru terbaik jika kamu mau belajar darinya.” –*

Penulis

*“Walaupun tidak ada satu hal pun yang sempurna di dunia, tetaplah membawakan yang terbaik, untuk dirimu dan orang sekitarmu.” - Penulis*



## KATA PENGANTAR

*Assalamu 'alaikumwarahmatullahi wabaraktuh*

Dengan ini, penulis ingin menyampaikan syukur yang sebesar-besarnya kehadiran Allah SWT yang telah memberikan rahmat, hidayah dan karunia-Nya sehingga penulis mampu menyelesaikan laporan penelitian tugas akhir setelah berbagai kesempatan yang diberikan-Nya. Shalawat dan salam yang juga tidak lupa untuk dipanjatkan kepada panutan dan junjungan seluruh umat muslim di dunia, Nabi Muhammad SAW, yang melalui perjuangan dan kerja kerasnya, umatnya mampu untuk keluar dari masa-masa kelam jahiliah menuju masa yang penuh harapan.

Penulisan tugas akhir oleh penulis ini bertujuan untuk menyelesaikan studi S1 di Jurusan Informatika, yang menjadi syarat kelulusan yang harus dipenuhi. Melalui berbagai keterbatasan dan beberapa rintangan, *alhamdulillah* laporan tugas akhir ini mampu terselesaikan. Tidak lupa penulis juga ingin menyampaikan terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah membantu penulis, baik itu secara langsung atau tidak, sehingga akhirnya penulis mampu merampungkan laporan ini.

Biarpun setelah semua yang dilalui dan diusahakan, penulis secara penuh sadar bahwa tidak ada hal yang luput dari kesalahan. Masih banyak kesalahan dan kekurangan yang terjadi baik itu selama proses pengerjaan maupun yang tertuang dalam penulisan tugas akhir ini. Itu semua tidak lepas dari masih kurangnya kemampuan dan pengalaman yang dimiliki oleh penulis. Tetapi, terlepas dari segala kekurangan tersebut, penulis berharap dengan terselesainya tugas akhir ini, banyak pihak yang mampu untuk mengambil pelajaran, pengetahuan dan pengalaman sehingga dapat bermanfaat lebih untuk semua pihak.

*Wassalamu 'alaikum warahmatullahi wabaraktuh*

Yogyakarta, 26 Oktober 2020



( Ridho Imam Pratama )

## SARI

Merupakan salah satu penyebab kematian terbanyak di seluruh dunia, penyakit Tuberkulosis menjadi salah satu fokus penting dalam meningkatkan kualitas Kesehatan di dunia. Berbagai upaya dilakukan oleh seluruh pihak, baik itu melalui program yang dijalankan oleh pemerintah atau pengembangan di dunia medis dengan memberikan terapi pada penderitanya. Walaupun dengan segala usaha yang telah dilakukan, tingkat penderita di Indonesia masih terbilang cukup tinggi, berada pada angka 420.994 kasus pada tahun 2017. Penyakit tuberkulosis sendiri disebabkan oleh bakteri *mycobacterium tuberculosis* yang menginfeksi organ pernafasan penderitanya. Bakteri tersebut merupakan bakteri tahan asam (BTA) dengan ukuran yang cukup besar dan bersifat *nonmotile*.

Salah satu cara dalam melakukan diagnosis terhadap penyakit ini adalah dengan melakukan pengujian terhadap spesimen *sputum* dari pasien dan melihatnya di bawah mikroskop. Spesimen diambil sebanyak tiga kali dengan kurun waktu yang berbeda, yaitu pada saat awal kunjungan, satu hari setelah kunjungan dan sore hari pada hari kedua. Melalui gambar dari mikroskop ini nantinya laboran diminta untuk menentukan diagnosis dengan mencari keberadaan bakteri pada sampel yang diberikan. Pada tahapan inilah diharapkan sistem mampu membantu proses diagnosis dari sampel yang diberikan.

Pengolahan citra dapat dimanfaatkan dalam membantu sistem menyelesaikan tugasnya. Dengan melakukan ekstraksi informasi terhadap citra yang didapat melalui mikroskop, sistem diharapkan mampu mendeteksi keberadaan bakteri pada citra sampel yang diberikan. Kemudian, dengan melihat seberapa banyak bakteri yang terdeteksi pada keseluruhan citra sampel, sistem akan menentukan seberapa parah penyakit yang diderita oleh pasien. Dengan begitu, proses diagnosis dapat dilakukan melalui citra mikroskop yang diolah dengan menggunakan bantuan komputer.

Melalui penelitian ini, pendeteksian bakteri menggunakan komputasi terhadap citra digital sangat memungkinkan. Dari pengujian yang dilakukan pada sistem, didapatkan nilai F1 dari pendeteksian bakteri sebesar 74.25%. Nilai ini didapat dari tingkat *precision* sebesar 86.49% dan tingkat *recall* sebesar 65.04%. Sementara itu, ketepatan kesimpulan yang diambil akan sangat terpengaruh oleh seberapa baik sistem mampu menemukan serta menentukan bakteri dengan baik.

Kata kunci: Tuberkulosis, *mycobacterium tuberculosis*, *nonmotile*, pengolahan citra.

## GLOSARIUM

<i>Mycobacterium Tuberculosis</i>	Bakteri yang menginfeksi manusia sehingga menyebabkan penyakit Tuberkulosis.
<i>Colour Deconvolution</i>	Algoritma yang digunakan untuk mencari nilai kepekatan suatu warna pada sebuah citra.
<i>Nonmotile</i>	Sifat yang dimiliki organisme yang tidak mampu untuk bergerak.
<i>Sputum</i>	Dahak



## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING .....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI .....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR.....	vii
SARI .....	viii
GLOSARIUM .....	ix
DAFTAR ISI .....	x
DAFTAR TABEL .....	xii
DAFTAR GAMBAR.....	xiii
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
1.6 Metodologi Penelitian .....	3
1.7 Sistematika Penelitian .....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>5</b>
2.1 Penelitian Sejenis .....	5
2.2 Penyakit Tuberkulosis .....	6
2.3 Pengolahan Citra .....	7
2.4 Colour Deconvolution .....	8
<b>BAB III ANALISIS DAN PERANCANGAN SISTEM.....</b>	<b>10</b>
3.1 Analisis Sistem.....	10
3.1.1 Kebutuhan Masukan .....	10
3.1.2 Kebutuhan Proses .....	10
3.1.3 Kebutuhan Keluaran .....	11
3.1.4 Kebutuhan Antarmuka .....	11
3.2 Pengumpulan Data .....	12
3.3 Perancangan Sistem .....	13
3.3.1 Use Case Diagram .....	14
3.3.2 Activity Diagram .....	15
3.3.3 Perancangan Basisdata .....	21
3.3.4 Perancangan Algoritma Diagnosis .....	24
3.3.5 Perancangan Antarmuka .....	31
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>40</b>
4.1 Implementasi Sistem .....	40
4.1.1 Halaman Login .....	40
4.1.2 Halaman Utama .....	41
4.1.3 Halaman Manajemen Puskesmas .....	42
4.1.4 Halaman Manajemen Sampel.....	44
4.1.5 Halaman Diagnosis.....	46
4.1.6 Halaman Hasil .....	50

4.2 Pengujian Sistem.....	51
BAB V KESIMPULAN DAN SARAN.....	60
5.1 Kesimpulan .....	60
5.2 Saran.....	60
DAFTAR PUSTAKA.....	61
LAMPIRAN .....	62



**DAFTAR TABEL**

Tabel 2.1 Panduan Diagnosa IUATLD.....	7
Tabel 3.1 Tabel Pengguna.....	22
Tabel 3.2 Tabel Puskesmas.....	23
Tabel 3.3 Tabel Sampel .....	23
Tabel 3.4 Tabel Citra .....	24
Tabel 4.1 Tabel <i>Confussion Matrix</i> .....	52



## DAFTAR GAMBAR

Gambar 2.1 Alur <i>Colour Deconvolution</i> .....	9
Gambar 3.1 Use Case Diagram.....	14
Gambar 3.2 Activity Diagram Login .....	16
Gambar 3.3 Activity Diagram Manajemen Pengguna .....	17
Gambar 3.4 Activity Diagram Manajemen Puskesmas .....	18
Gambar 3.5 Activity Diagram Manajemen Sampel.....	19
Gambar 3.6 Activity Diagram Lihat Detil .....	20
Gambar 3.7 Activity Diagram Diagnosis Citra.....	21
Gambar 3.8 Tabel Relasi.....	22
Gambar 3.9 Alur proses diagnosis sampel.....	25
Gambar 3.10 Alur <i>pre-process</i> .....	26
Gambar 3.11 Alur deteksi bakteri.....	27
Gambar 3.12 Alur pemilihan dan pelabelan objek .....	28
Gambar 3.13 Alur pengambilan kesimpulan .....	30
Gambar 3.14 Rancangan antarmuka Login .....	31
Gambar 3.15 Rancangan antarmuka Halaman Utama 1 .....	32
Gambar 3.16 Rancangan antarmuka Halaman Utama 2 .....	33
Gambar 3.17 Rancangan antarmuka Puskesmas 1 .....	34
Gambar 3.18 Rancangan antarmuka Puskesmas 2 .....	34
Gambar 3.19 Rancangan antarmuka Sampel 1 .....	36
Gambar 3.20 Rancangan antarmuka Sampel 2 .....	36
Gambar 3.21 Rancangan antarmuka Sampel 3 .....	37
Gambar 3.22 Rancangan antarmuka Diagnosis .....	38
Gambar 3.23 Rancangan antarmuka Hasil.....	39
Gambar 4.1 Antarmuka Login .....	41
Gambar 4.2 Antarmuka Halaman Utama.....	42
Gambar 4.3 Antarmuka Manajemen Puskesmas .....	43
Gambar 4.4 Antarmuka Tambah Puskesmas .....	43
Gambar 4.5 Antarmuka Manajemen Sampel.....	45
Gambar 4.6 Antarmuka Detil Sampel.....	45
Gambar 4.7 Antarmuka Ubah Detil .....	46

Gambar 4.8 Antarmuka Diagnosis.....	47
Gambar 4.9 Potongan kode <i>pre-processing</i> .....	48
Gambar 4.10 Potongan kode algoritma <i>colour deconvolution</i> .....	48
Gambar 4.11 Potongan kode pemilihan objek.....	49
Gambar 4.12 Potongan kode pengambilan kesimpulan.....	50
Gambar 4.13 Antarmuka Hasil .....	51
Gambar 4.14 Perbandingan gambar sistem (atas) dengan ahli (bawah) 1 .....	54
Gambar 4.15 Perbandingan gambar sistem (atas) dengan ahli (bawah) 2 .....	55
Gambar 4.16 Perbandingan gambar sistem (atas) dengan ahli (bawah) 3 .....	56
Gambar 4.17 Perbandingan gambar sistem (atas) dengan ahli (bawah) 4 .....	57
Gambar 4.18 Perbandingan gambar sistem (atas) dengan ahli (bawah) 5 .....	58



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Penyakit Tuberkulosis merupakan salah satu dari sepuluh penyebab kematian tertinggi di seluruh dunia, Tuberkulosis terus berada dalam jajaran sepuluh penyebab kematian dalam rentang tahun 2000-2016. Sementara di Asia Tenggara, yang menyumbang 45% kasus Tuberkulosis di seluruh dunia pada tahun 2016, menempati posisi kelima sebagai penyebab kematian terbanyak, di bawah jantung koroner dan stroke mulai dari tahun 2000 hingga yang terakhir pada tahun 2016 (WHO, 2018). Di Indonesia khususnya, kasus Tuberkulosis yang terjadi pada tahun 2017 mencapai 420.994 kasus atau 254 kasus per 100.000 penduduk (Kementerian Kesehatan Republik Indonesia, 2018).

Penyakit Tuberkulosis sendiri diakibatkan dari infeksi oleh *Mycobacterium Tuberculosis*, yang merupakan Bakteri Tahan Asam (BTA). *Mycobacterium Tuberculosis*, yang menjadi penyebab penyakit Tuberkulosis ini, memiliki ukuran yang cukup besar untuk sebuah bakteri, dengan sifat *nonmotile*. Bentuknya menyerupai batang dengan panjang sekitar 2-4 mikrometer dan lebar 0,2-0,5 mikrometer dan memiliki waktu regenerasi yang lambat. Sifat regenerasinya yang lambat ini lah yang menjadi salah satu pengaruh besar terhadap keganasan dari penyakit ini.

Sebagaimana penyakit pada umumnya, Tuberkulosis terbagi ke dalam beberapa kelompok, yaitu Tuberkulosis paru BTA (+), Tuberkulosis paru BTA (-) dan Tuberkulosis ekstraparu. Pembagian Tuberkulosis paru BTA (+) dan Tuberkulosis paru BTA (-) didasari oleh pengujian terhadap spesimen sputum (dahak) pasien, di mana untuk Tuberkulosis paru BTA (+) pengujian terhadap *sputum* menghasilkan positif BTA bersamaan dengan hasil radiologi positif Tuberkulosis aktif dan atau biakan positif. Sementara untuk Tuberkulosis paru BTA (-) merupakan hasil radiologi positif Tuberkulosis aktif dengan hasil pengujian *sputum* BTA negatif. Sedangkan Tuberkulosis ekstraparu, yang merupakan Tuberkulosis yang menginfeksi organ tubuh selain paru, diperlukan pengujian berupa pembiakan atau patologi anatomi terhadap lokasi lesi.

Melalui sedikit uraian sebelumnya, setidaknya terdapat tiga cara dalam memastikan pasien terinfeksi *Mycobacterium Tuberculosis*. Pertama adalah melalui kultur bakteri

(pembiakan), baik melalui media *Egg-based* atau *Agar-based*, lalu dilihat laju pertumbuhan dari bakteri yang berada pada bahan uji. Selanjutnya adalah melalui citra radiologi dari organ yang diduga terinfeksi, dan pemeriksaan bakteriologik terhadap *sputum* atau bahan lain. Sampel yang digunakan dalam pemeriksaan bakteriologik dapat berasal dari berbagai macam sumber, seperti *sputum* (dahak), cairan *pleura*, urin hingga *faeces*. Pengambilan sampel dilakukan sebanyak tiga kali, yang dapat dilakukan dengan cara SPS atau pengambilan sampel selama tiga hari berturut-turut.

Berdasarkan prosedur pemeriksaan bakteriologik di atas, jika terjadi sebuah kasus yang mana pasien dikhawatirkan terinfeksi Tuberkulosis, diperlukan sekurang-kurangnya tiga sampel untuk dilakukan pemeriksaan bakteriologik. Jika melihat data sebelumnya, di mana kasus Tuberkulosis yang terjadi di Indonesia pada tahun 2017, maka akan terdapat satu juta lebih sampel yang perlu dilakukan pemeriksaan pada tahun itu. Jumlah tersebut belum mencakup pasien yang telah diperiksa namun memberikan hasil negatif Tuberkulosis, yang tentunya jika ikut disertakan, jumlah sampel yang perlu dilakukan pemeriksaan akan semakin bertambah. Hal ini dapat membebani kinerja spesialis yang menangani pemeriksaan tersebut. Ditambah lagi dengan kemungkinan persebaran laboratorium di Indonesia yang kurang merata, dapat menghambat proses pemeriksaan dan pastinya akan mempengaruhi pemberian vonis terhadap pasien oleh dokter. Atas dasar ini, maka diperlukan suatu cara untuk mempermudah proses tersebut.

Salah satu solusi yang dapat digunakan adalah dengan menggunakan komputer dalam melakukan pemeriksaan terhadap citra mikroskopik dari sampel terkait. Hal ini dapat dicapai melalui teknik pengolahan citra oleh komputer. Pengaplikasian teknik pengolahan citra pun bisa dibidang tidak lah sedikit, salah satu contohnya adalah penggunaan e-tilang yang belakangan ini banyak diperbincangkan. Dalam bidang medis khususnya, pengaplikasian teknik ini sangat besar, sebagai contohnya adalah alat MRI, CT scan atau *ultrasound*. Dengan demikian, pemeriksaan citra mikroskopik dari sampel pasien sangat memungkinkan untuk dicapai. Melalui berbagai tahapan, seperti *denoising*, *preprocessing*, *segmentasi* hingga pengambilan fitur citra dapat membantu dalam memenuhi tujuan sistem.

## 1.2 Rumusan Masalah

Didasari atas latar belakang yang telah diuraikan di atas, berikut rumusan masalah yang akan disimpulkan :

- a. Bagaimana cara mengolah citra mikroskopik sampel, sehingga mampu menemukan bakteri yang terdapat dalam sampel?:
- b. Bagaimana menentukan tingkat keparahan Tuberkulosis berdasarkan hasil penemuan bakteri dari sampel?

### 1.3 Batasan Masalah

Batasan masalah yang ditentukan untuk penelitian ini adalah sebagai berikut :

- a. Hanya mengakomodasi Tuberkulosis paru.
- b. Penelitian didasari atas pemeriksaan bakteriologik sampel untuk menemukan kandungan BTA.
- c. Data yang digunakan adalah citra mikroskopik dari sampel.
- d. Tingkat keparahan ditentukan melalui jumlah BTA yang terkandung dalam sampel.

### 1.4 Tujuan Penelitian

Tujuan yang diinginkan terhadap terselesaikannya penelitian ini adalah untuk membangun sebuah sistem yang dapat memberikan bantuan dalam menyelesaikan sebuah tugas yang dibebankan kepada laboran dalam melakukan deteksi bakteri tuberkulosis. Sehingga diharapkan beban kerja dapat teringankan serta waktu penyelesaian tugas dapat dipersingkat.

### 1.5 Manfaat Penelitian

Melalui sistem ini, diharapkan timbul manfaat berupa :

- a. Peringan beban kerja laboran dalam melakukan deteksi bakteri pada sampel *sputum* pasien.
- b. Mempersingkat waktu pengerjaan deteksi bakteri.

### 1.6 Metodologi Penelitian

- a. Studi Literatur

Tahapan yang dilkakukan untuk mempelajari sumber masalah, langkah penyelesaian serta panduan penyelesaian melalui kajian literatur dari berbagai sumber, baik itu dari buku, website, jurnal ataupun bentuk tulisan penelitian lainnya yang berhubungan serta dapat membantu penyelesaian kasus pada sistem.

- b. Studi Kasus dan Pengumpulan Data

Tahapan untuk memahami secara langsung mengenai proses yang dilakukan oleh laboran dalam menemukan bakteri tuberkulosis, proses berjalannya pemeriksaan pasien hingga diagnosis serta mengetahui kasus-kasus yang mungkin didapatkan selama pengambilan citra sampel. Melalui tahapan ini pula data yang diperlukan selama proses penelitian akan dikumpulkan melalui pengambilan citra sampel menggunakan kamera pada mikroskop.

c. Perancangan Sistem

Tahapan ini dilakukan untuk memberikan gambaran secara umum mengenai sistem yang akan dibangun demi mempermudah proses pembuatan sistem kedepannya. Perancangan didasarkan atas analisis dari kebutuhan sistem yang sebelumnya dilakukan.

d. Implementasi Sistem

Setelah proses perancangan sistem terselesaikan, tahapan selanjutnya adalah dengan melakukan implementasi sistem. Pada tahap ini, sistem akan dibangun berdasarkan perancangan pada tahapan sebelumnya. Kebutuhan akan masukan, proses, keluaran dan antarmuka akan direalisasikan sehingga sistem mampu berjalan secara utuh.

e. Pengujian Akurasi

Demi menjaga kualitas serta reliabilitas dari sistem, diperlukan pengujian terhadap akurasi yang didapatkan oleh sistem dalam melakukan pendeteksian serta penentuan kriteria tingkat keparahan. Tahapan ini dilakukan dengan cara membandingkan pendeteksian yang dilakukan oleh sistem dan pendeteksian yang dilakukan oleh ahli dan diolah dalam bentuk *confussion matrix*. Selanjutnya, nilai *precision*, *recall* dan F1 dapat ditemukan melalui informasi yang terhimpun.

## 1.7 Sistemaiika Penelitian

Selama penyelesaian, tugas akhir akan terbagi menjadi beberapa bab yang akan mencakup proses penyelesaian tugas akhir. Susunan yang terdapat dalam tugas akhir yaitu :

- a. Bab I Pendahuluan
- b. Bab II Landasan Teori
- c. Bab III Analisis dan Perancangan Sistem
- d. Bab IV Hasil dan Pembahasan
- e. Bab V Kesimpulan dan Saran

## BAB II

### LANDASAN TEORI

#### 2.1 Penelitian Sejenis

Pemanfaatan teknologi pada bidang kesehatan khususnya komputer merupakan suatu hal yang mampu meningkatkan kualitas pelayanan kesehatan. Dengan memanfaatkan teknologi, kecepatan, ketepatan dan kepastian dalam menentukan diagnosis dan pemberian pelayanan dapat ditingkatkan. Banyak contoh yang dapat disebutkan mengenai pemanfaatan teknologi di dunia kesehatan, mulai dari sebuah sistem pembantu keputusan dokter atau ahli hingga pengarsipan data melalui database yang terintegrasi.

Salah satu contoh lainnya adalah dengan adanya pemanfaatan komputer untuk melakukan deteksi dan klasifikasi objek melalui citra yang ditangkap oleh alat khusus, misalnya mikroskop. Pada tulisan yang berjudul *Methods for Segmentation and Classification of Digital Microscopy Tissue Images* (Vu, et al., 2019), telah dilakukan sebuah penelitian untuk membantu para ahli dalam menentukan sebuah kelainan pada jaringan manusia dengan membedakan karakteristik dari citra yang ditangkap oleh mikroskop. Dalam tulisan tersebut para peneliti memanfaatkan pengolahan citra oleh komputer untuk mendapatkan hasil yang diharapkan.

Pada tulisan lainnya yang berjudul *Stain Deconvolution Using Statistical Analysis of Multi-Resolution Stain Colour Representation*, (Alsubaie, Trahearn, Raza, Snead, & Rajpoot, 2017) pengolahan citra memanfaatkan algoritma bernama *colour deconvolution*. Algoritma ini dipakai untuk mendapatkan kadar pewarnaan dari sebuah sampel yang nantinya dapat digunakan untuk menentukan objek dengan karakteristik tertentu. Masih dalam tulisan yang sama, disebutkan bahwa algoritma memerlukan sebuah matriks untuk membantu komputer dalam mendapatkan kadar pewarnaan yang diinginkan. Matriks ini dapat dicari melalui sebuah metode yang pernah dikeluarkan oleh (Macenko, et al., 2009) melalui tulisannya yang berjudul *A method for normalizing histology slides for quantitative analysis*.

Melalui algoritma *stain deconvolution* (Alsubaie, Trahearn, Raza, Snead, & Rajpoot, 2017), seperti yang disebutkan pada penjelasan sebelumnya, dan penggunaan metode pada tulisan berjudul *A method for normalizing histology slides for quantitative analysis* (Macenko, et al., 2009), penggunaannya ke dalam sistem patut dipertimbangkan. Meskipun kasus yang diteliti berbeda, di mana pada penelitian tersebut menggunakan citra kanker dan pada kasus

yang kini sedang diteliti menggunakan citra *sputum*, keduanya memiliki potensi untuk dimanfaatkan dalam melakukan proses pendeteksian bakteri. Hal ini merujuk pada fakta bahwa dalam melakukan pemeriksaan mikroskopik dari sampel *sputum*, sampel terlebih dahulu dilakukan pewarnaan dengan menggunakan dua jenis warna berbeda (Todar, 2006), sementara pada kasus kanker juga menggunakan dua pewarnaan yang berbeda.

## 2.2 Penyakit Tuberkulosis

Penyakit Tuberkulosis diakibatkan dari infeksi oleh *Mycobacterium Tuberculosis*, yang merupakan Bakteri Tahan Asam (BTA). *Mycobacterium Tuberculosis*, yang menjadi penyebab penyakit Tuberkulosis ini, memiliki ukuran yang cukup besar untuk sebuah bakteri, dengan sifat *nonmotile*. Bentuknya menyerupai batang dengan panjang sekitar 2-4 mikrometer dan lebar 0,2-0,5 mikrometer dan memiliki waktu regenerasi yang lambat. Sifat regenerasinya yang lambat ini lah yang menjadi salah satu pengaruh besar terhadap keganasan dari penyakit ini (Todar, 2006).

Sebagaimana yang telah disebutkan di atas, bahwa bakteri penyebab Tuberkulosis merupakan bakteri yang memiliki karakteristik BTA. Ini mengakibatkan bakteri akan mempertahankan warna, yang sebelumnya diberikan pada tahap pewarnaan awal, ketika sampel diluruhkan dengan asam alkohol. Pewarnaan awal sendiri dilakukan dengan menggunakan pewarna *fuchsin* yang berwarna merah muda. Kemudian sampel akan diberikan pewarnaan untuk kedua kalinya yang bertujuan untuk memberikan perbedaan yang kontras antara bakteri yang masih mengikat warna sebelumnya, dengan bagian lain disekitarnya. Pada umumnya, pewarnaan kedua dilakukan dengan menggunakan metilen biru. Metode pewarnaan ini biasa juga disebut dengan metode pewarnaan Ziehl-Neelsen (Todar, 2006).

Dalam melakukan diagnosis melalui mikroskop, sampel dilihat dengan menggunakan lensa objektif dengan perbesaran 100 kali dan dibantu dengan minyak imersi. Nantinya pemeriksa akan mencari bakteri yang sesuai dengan kriteria yang sebelumnya telah dibahas dan dilakukan sebanyak 100 lapang pandang per sampel. Selanjutnya, untuk menentukan tingkat keparahan dari Tuberkulosis yang digambarkan oleh sampel, pemeriksa perlu menghitung seberapa banyak bakteri yang ditemukan dalam satu lapang pandang dan atau pada satu sampel. Penentuan tingkat keparahan ini mengikuti panduan yang dikeluarkan oleh Kemenkes / *International Union Against Tuberculosis and Lung Disease (IUATLD)* (Fakultas Kedokteran Universitas Hasanuddin, 2017). Berikut ini merupakan tabel yang akan menggambarkan panduan tersebut.

Tabel 2.1 Panduan Diagnosa IUATLD

Diagnosis	Bakteri ditemukan
Negatif	0/100 Lapang Pandang
Meragukan ( <i>scanti</i> )	1-9/100 Lapang Pandang
1+	10-99/100 Lapang Pandang
2+	1-10/Lapang Pandang (min. 50 Lapang Pandang)
3+	>10/Lapang Pandang (min. 20 Lapang Pandang)

### 2.3 Pengolahan Citra

Pengolahan citra merupakan sebuah proses perhitungan ataupun rekayasa pada sebuah citra digital pada komputer dengan berbagai tujuan berbeda, seperti pengenalan objek oleh komputer, penjernihan citra, perbaikan kualitas citra atau rekayasa objek pada citra digital. Penggunaan pengolahan citra sendiri sangatlah banyak dan beragam, dimulai dari kegunaannya di industri hiburan, keamanan, lalu lintas, pembelajaran dan bahkan kesehatan.

Terdapat berbagai proses, metode dan tujuan dalam pengolahan citra. Pemanfaatan pengolahan citra bisa bertujuan untuk melakukan perbaikan kualitas sebuah citra, pengurangan noise, mempertajam gambar atau objek, mengenali objek dan atau penyusunan dari gambar beruntut (Suetens, 2009). Dengan memanfaatkan satu atau beberapa metode serta proses tertentu, citra dapat diolah untuk mendapatkan hasil yang diinginkan atau mendapatkan informasi yang ingin didapat.

Selain itu, terdapat beberapa tahapan yang perlu dilakukan untuk memaksimalkan hasil yang akan diperoleh dari suatu sistem pengolahan citra. Tahapan ini biasa disebut *pre-processing* yang mana tahapan ini bertujuan untuk melakukan pengolahan terlebih dahulu pada citra agar dapat mengekstraksi informasi dari citra lebih baik lagi (University of Tartu, 2014). *Pre-processing* sendiri juga terbagi menjadi beberapa metode yang dapat dimanfaatkan sesuai kebutuhan akhir dari sebuah sistem pengolahan citra,

Dalam bidang kesehatan, pengolahan citra sendiri dimanfaatkan di dalam banyak alat kesehatan. Sebagai contoh sederhana, penggunaan USG dalam pengecekan kesehatan janin dan foto MRI, keduanya memanfaatkan komputasi komputer dalam mengolah dan menampilkan citra kepada pengguna. Sehingga pada dasarnya, pemanfaatan pengolahan citra pada bidang kesehatan bukanlah hal yang baru atau aneh untuk diterapkan.

## 2.4 Colour Deconvolution

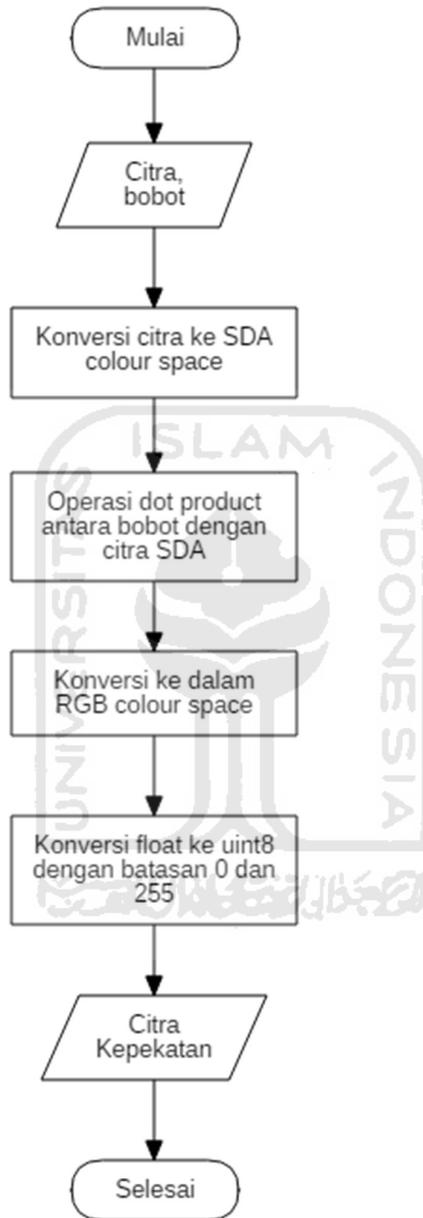
Merujuk pada penjelasan tulisan yang dikeluarkan Alsubaie dan kolega, *Colour Deconvolution* merupakan salah satu metode yang digunakan dalam pengolahan citra untuk mendapatkan *stain concentration values*. *Stain concentration values* sendiri merupakan sebuah nilai yang akan memberikan informasi mengenai seberapa pekat pewarnaan yang diterima pada sebuah citra, dengan sebuah matriks yang sebelumnya telah diberikan ke dalam algoritma (Alsubaie, Trahearn, Raza, Snead, & Rajpoot, 2017). Nantinya, tergantung dari seberapa banyak pewarnaan yang diberikan pada sebuah citra, metode *colour deconvolution* akan memberikan keluaran yang sesuai dengan jumlah pewarnaan tersebut.

Sebagai contoh, pada proses pewarnaan slide sampel TB, sampel akan diberikan pewarnaan sebanyak dua kali, biasanya dengan pewarna *carbol-fuchsin* (merah muda) dan *methylene-blue* (biru), supaya bakteri dapat ditemukan oleh laboran. Sehingga, penggunaan matriks yang tepat adalah dengan menggunakan sebuah matriks yang mendukung pewarnaan dengan dua warna. Nantinya, dari matriks yang telah dibentuk sebelumnya ini, metode *colour deconvolution* akan menghitung seberapa pekat warna yang ada dalam citra dari dua pewarnaan tersebut (merah muda dan biru).

Sementara itu, untuk menemukan matriks yang akan digunakan dalam algoritma *colour deconvolution*, Macenko dan koleganya telah memaparkan salah satu jalannya pada karya tulisnya berjudul *A method for normalizing histology slides for quantitative analysis*. Seperti halnya yang telah dijelaskan sebelumnya, nilai matriks akan terpengaruh oleh jumlah pewarnaan yang digunakan pada sampel. Nantinya, pada kasus dengan dua pewarnaan, matriks akan berbentuk matriks 3x3 dengan kolom pertama berisikan nilai untuk pewarnaan satu, kolom kedua untuk nilai pewarnaan dua dan kolom terakhir berisikan nilai *cross product* dari kedua kolom sebelumnya. Barulah setelah itu matriks yang telah ditemukan tersebut digunakan pada algoritma *colour deconvolution*.

Secara umum, algoritma *colour deconvolution* dilakukan dengan tahapan berikut. Pertama, diperlukan citra serta matriks bobot yang akan digunakan untuk memisahkan pewarnaan. Tahap selanjutnya, citra akan dikonversi ke dalam bentuk SDA *colour space* yang kemudian akan dilakukan perkalian *dot product* dengan bobot yang sebelumnya disebutkan. Setelah hasil *dot product* ditemukan, matriks tersebut kembali dikonversi ke dalam bentuk RGB. Pada posisi ini, matriks yang terbentuk masih berupa nilai *float* dengan nilai yang memiliki kemungkinan lebih besar dari 255 atau lebih rendah dari 0, yang tentunya berada di luar jangkauan RGB. Maka dari itu, matriks kembali diolah dengan mengubahnya menjadi tipe

data yang sesuai (uint8) dan membatasinya dalam rentang 0 hingga 255. Barulah matriks ini yang nantinya dapat dilihat dan digunakan sebagai nilai kepekatan dari pewarnaan tertentu.



Gambar 2.1 Alur *Colour Deconvolution*

## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

#### 3.1 Analisis Sistem

Analisis Sistem dilakukan guna mengetahui kebutuhan apa saja yang diperlukan sistem ketika nanti dibangun. Analisis kebutuhan terhadap sistem dilakukan dengan melakukan dan menyelesaikan beberapa tahapan sehingga menghasilkan Kebutuhan Masukan, Kebutuhan Proses, Kebutuhan Keluaran dan Kebutuhan Antarmuka. Secara umum, sistem akan terlihat seperti pada Gambar 3.1 yang membahas mengenai *Use Case Diagram* dari sistem. Untuk penjelasan lebih jelasnya, berikut ini merupakan penjabaran terhadap poin-poin yang baru saja disebutkan di atas.

##### 3.1.1 Kebutuhan Masukan

Dalam rangka untuk memulai prosesnya, sistem memerlukan masukan yang diberikan baik itu oleh pengguna atau kemungkinan sistem terkait lain. Sebagai contohnya, masukan yang diberikan oleh pengguna dapat berupa formulir pendaftaran atau file yang diunggah pengguna. Sementara contoh lain yang merupakan masukan dari sistem lain dapat berupa data mengenai sebuah rekam puskesmas yang telah tersimpan di dalam basis data ataupun hasil dari olahan proses dari sistem lainnya. Pada kasus ini, sistem memerlukan masukan yang berasal dari pegawai laboratorium.

Laboran merupakan pegawai yang langsung melakukan pengecekan terhadap sampel yang diterima mereka setiap harinya. Mereka akan mengambil sejumlah citra sampel tersebut dengan perbesaran 1000 kali melalui mikroskop. Nantinya laboran mampu melakukan masukan berupa citra sampel dan data sampel sebagai pelengkap informasi bagi sistem.

##### 3.1.2 Kebutuhan Proses

Dalam rangka mencapai tujuan dari sistem, diperlukan serangkaian proses yang akan mengolah data yang diberikan hingga mampu menghasilkan informasi yang diinginkan. Bagian ini akan menjabarkan mengenai proses yang akan dilakukan dalam sistem. Berikut ini merupakan proses dalam sistem ∴

a. Proses Manajemen Pengguna

Proses yang bertujuan untuk melakukan pengelolaan terhadap data pengguna (laboran) yang dapat mengakses fitur yang terdapat dalam sistem.

b. Proses Manajemen Puskesmas

Proses yang bertujuan untuk melakukan pengelolaan terhadap data Puskesmas baik itu untuk mendaftarkan, mengubah atau menghapus data terkait.

c. Proses Manajemen Sampel

Proses yang bertujuan untuk melakukan pengelolaan pada sampel yang masuk ke dalam sistem, proses ini sedikit banyak terkait pada pengelolaan Puskesmas.

d. Proses Diagnosis

Proses yang dilakukan dalam rangka mengetahui hasil perhitungan terhadap masukan yang telah diberikan. Hasil dari proses ini akan memberikan informasi pada pengguna terkait dengan pendeteksian bakteri pada tiap sampel.

### 3.1.3 Kebutuhan Keluaran

Melalui keluaran inilah nantinya pengguna akan dihadapkan oleh hasil olahan. Pada akhir proses yang telah diselesaikan, sistem akan memberikan hasil berupa informasi terkait sampel yang diberikan sebagai masukan, seperti jumlah bakteri terdeteksi, kelas keparahan penyakit dan penjelasan lain. Informasi tersebut nantinya akan ditampilkan melalui antarmuka kepada pengguna sehingga pengguna mampu memahami informasi yang diberikan.

### 3.1.4 Kebutuhan Antarmuka

Antarmuka yang diperlukan pada sistem adalah sebagai berikut :

- a. Antarmuka Login, merupakan halaman yang digunakan oleh pengguna agar dapat mengakses fitur lain dalam sistem. Di sini pengguna akan diberikan sebuah formulir yang akan diisi berdasarkan *username* serta *password* yang sebelumnya sudah didaftarkan. Melalui antarmuka ini Sistem mampu memberikan akses tertentu atau khusus kepada pengguna terdaftar melalui sesi.
- b. Antarmuka halaman utama, merupakan halaman hub bagi pengguna untuk mengakses ke fitur yang disediakan sistem. Sebagaimana yang telah dituliskan, pada halaman ini terdapat tautan-tautan yang akan menuntun pengguna untuk mengakses fitur-fitur yang disediakan. Dengan memanfaatkan sesi pengguna, nantinya halaman ini juga akan menyediakan tampilan yang menyesuaikan dengan pengguna yang menggunakan.

- c. Antarmuka manajemen pengguna, merupakan halaman yang hanya dapat diakses oleh admin terdaftar, yang digunakan untuk melakukan pengelolaan pengguna sistem. Halaman ini akan menampilkan daftar pengguna yang tersimpan di dalam basis data sistem. Melalui halaman ini, admin dapat melakukan pengelolaan dasar terhadap data pengguna, seperti menambah pengguna, mengubah data pengguna yang telah tersimpan atau menghapus pengguna dari basis data sistem.
- d. Antarmuka manajemen Puskesmas, merupakan halaman untuk melakukan pengelolaan terhadap data Puskesmas. Halaman akan menampilkan daftar Puskesmas terdaftar berikut aksi yang bisa dilakukan. Pada halaman ini juga, pengguna dapat mengetahui detail Puskesmas yang telah terdaftar dalam sistem.
- e. Antarmuka manajemen sampel, merupakan halaman yang digunakan untuk melakukan pengelolaan terhadap sampel, baik itu menambah, melihat detail atau menghapus. Tidak jauh berbeda dengan halaman manajemen Puskesmas, halaman akan menampilkan daftar sampel dan aksi yang bisa dilakukan. Selain itu, pada tiap sampel nantinya akan diberikan tautan yang akan mengantarkan pengguna pada halaman hasil sampel terkait.
- f. Antarmuka diagnosis sampel, merupakan halaman yang digunakan sebagai pintu masuk citra sampel yang nantinya akan didiagnosis. Pengguna akan memberikan masukkan lain berupa informasi yang relevan terhadap sampel yang diberikan. Halaman ini menjadi tahapan awal selama proses diagnosis citra oleh komputer, di mana citra yang dimaksudkan akan diunggah oleh pengguna untuk nantinya dilakukan diagnosis oleh sistem.
- g. Antarmuka hasil, merupakan halaman yang akan menampilkan hasil diagnosis dari citra yang sebelumnya dimasukkan. Pada halaman ini, sistem akan menyampaikan informasi terkait hasil diagnosis berupa jumlah bakteri yang terdeteksi, tingkat keparahan dan informasi Puskesmas sebagai informasi tambahan. Halaman ini yang nantinya akan digunakan pengguna untuk memahami hasil yang diselesaikan oleh sistem.

### **3.2 Pengumpulan Data**

Data yang diperlukan pada penelitian ini berupa data citra yang diambil menggunakan kamera mikroskop dengan lensa objektif perbesaran 100 kali. Karena menggunakan lensa objektif 100 kali ini, pengamatan melalui mikroskop akan dibantu dengan menggunakan minyak imersi yang akan membantu memfokuskan cahaya yang lewat. Mulanya sampel akan ditaruh dan dilihat menggunakan lensa dengan perbesaran yang lebih rendah, misal lensa dengan perbesaran 10 atau 40 kali, dan diatur fokusnya. Barulah setelah itu lensa ‘dipingirkan’

dan minyak imersi diteteskan pada sampel. Kemudian, dengan menggunakan lensa imersi (perbesaran 100), lensa kembali diatur untuk menemukan fokus yang sesuai. Selanjutnya hanya pengambilan citra dengan menggunakan kamera yang telah terpasang di mikroskop dan terhubung pada komputer. Pada penelitian kali ini, setiap sampelnya akan diambil sebanyak 10 citra lapang pandang untuk digunakan sebagai data.

Pengumpulan data dimulai dengan melakukan pengambilan sampel *sputum* pasien yang disediakan oleh Dinas Kesehatan Sleman. Pengambilan sampel *sputum* ini dilakukan pada tanggal 3 September 2019 dan diambil langsung di Gedung Dinas Kesehatan Sleman. Kumpulan sampel lain juga didapatkan dari Dinas Kesehatan Temanggung. Untuk sampel ini, pengambilan dilakukan pada tanggal 12 September 2019 di Temanggung.

Sementara itu, pengambilan citra melalui mikroskop dilakukan pada bulan September, November dan Desember. Pengambilan dilakukan di ruang laboratorium informatika lantai 4 menggunakan mikroskop dan kamera kepemilikan laboratorium yang telah tersedia. Dari sejumlah sampel yang tersedia, sekitar 12 sampel *sputum* dipilih untuk selanjutnya dilihat di bawah mikroskop. Pemilihan ini dilakukan untuk mendukung contoh kasus yang didapat (1+, 2+ dan 3+) serta kualitas usapan sampel *sputum* itu sendiri. Sekitar 100 citra didapatkan melalui pengumpulan ini.

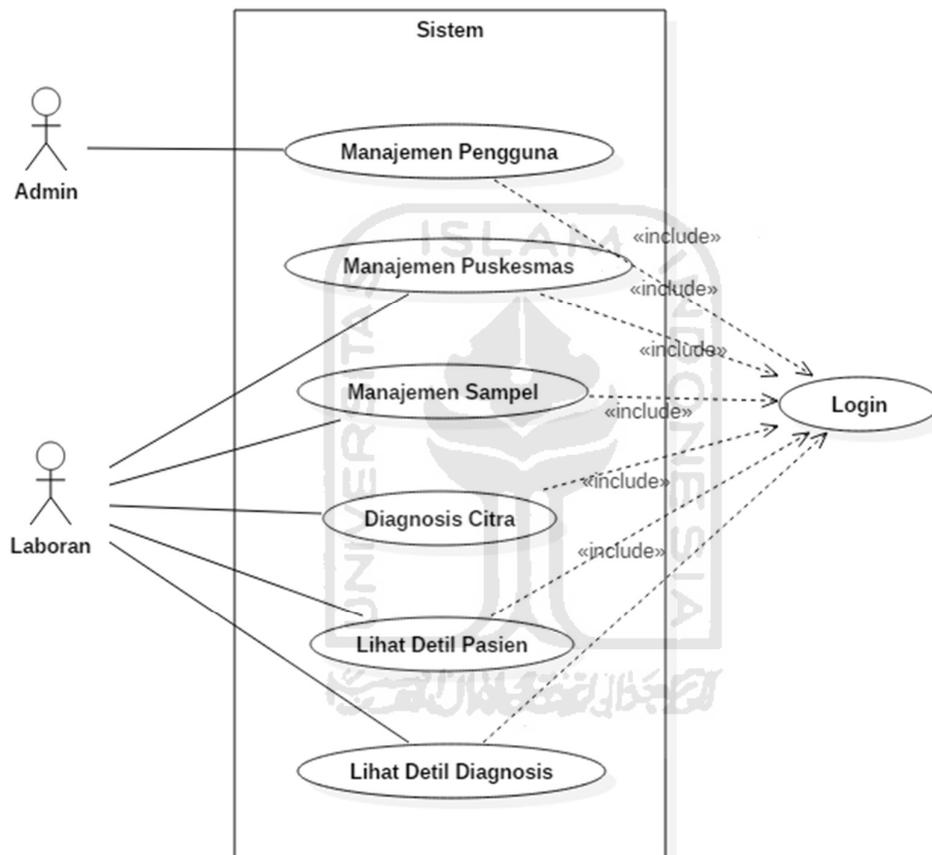
Selain itu, terdapat pula data citra yang telah didapatkan melalui penelitian sebelumnya. Pada akhirnya, dikarenakan satu dan dua hal, data ini yang nantinya akan digunakan pada hasil akhir dari sistem yang dibangun. Data ini didapat melalui sampel yang berbeda serta citra diambil di tempat serta alat yang berbeda juga. Data citra ini juga telah dilakukan pengecekan langsung ke ahli yang menandai objek-objek mana saja yang termasuk ke dalam bakteri penyebab tuberculosis. Data ini berjumlah 50 citra mikroskopik dan 42 citra acuan yang telah ditandai oleh ahli.

### **3.3 Perancangan Sistem**

Perancangan Sistem dimaksudkan untuk memberikan panduan selama proses pembuatan sistem nantinya. Pada perancangan ini akan dibahas mengenai beberapa hal terkait pengguna, proses yang akan terjadi di dalam sistem, antarmuka serta basisdata yang akan digunakan pada sistem. Berikut ini merupakan hasil dari perancangan yang telah dilakukan berdasarkan poin-poin pada bagian sebelumnya.

### 3.3.1 Use Case Diagram

Penggunaan use case diagram bertujuan untuk memberikan gambaran secara umum mengenai peranan serta kemampuan pengguna terhadap sistem. Pada sistem ini, pengguna, yaitu laboran, dapat melakukan akses terhadap hampir keseluruhan fitur yang diberikan oleh sistem minus manajemen pengguna yang akan diberikan kepada pengguna Admin dengan terlebih dahulu melakukan login pada sistem. Gambar 3.1 berikut akan menjelaskan mengenai use case pada sistem.



Gambar 3.1 Use Case Diagram

Melalui gambar yang telah disajikan, terdapat dua aktor yang dapat mengakses fitur pada sistem. Laboran, selanjutnya akan disebut sebagai pengguna, merupakan aktor yang berkerja dalam melakukan pengambilan citra sampel melalui mikroskop di laboratorium. Pengguna dapat mengakses berbagai fitur dalam sistem seperti pengelolaan puskesmas, pengelolaan sampel, melakukan diagnosis terhadap citra, serta melihat detil hasil diagnosis. Selain itu, sebelum pengguna mampu meangakses fitur-fitur tersebut, diperlukan bagi pengguna untuk

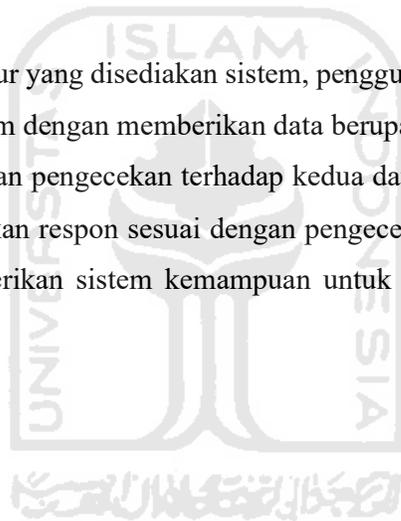
melakukan login ke dalam sistem terlebih dahulu. Aktor selanjutnya adalah admin yang akan dan hanya mengelola pengguna dari sistem. Fitur pengelolaan pengguna sendiri hanya dibebankan pada Admin, di mana Admin diharuskan untuk melakukan login terlebih dahulu sebelum dapat mengakses fitur tersebut.

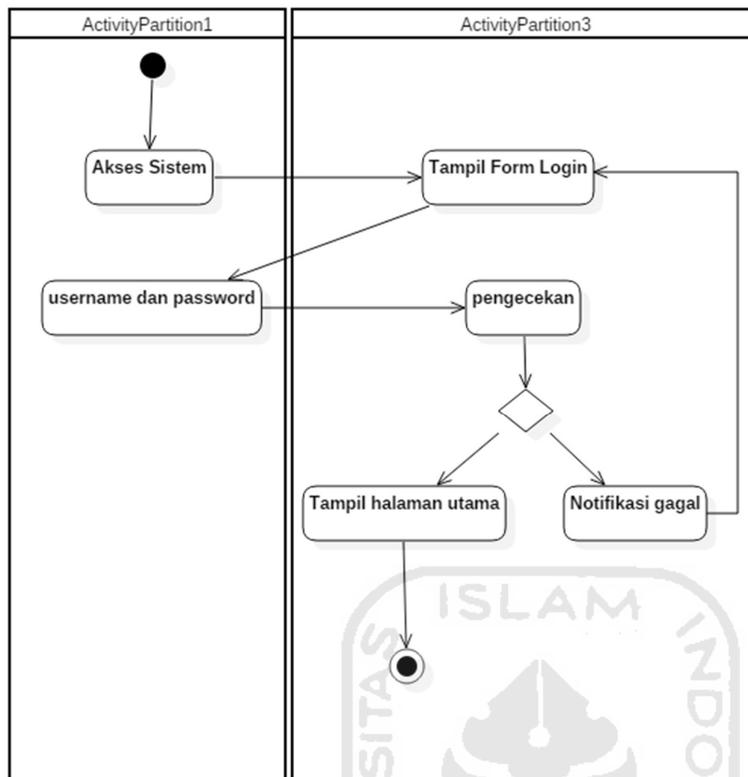
### 3.3.2 Activity Diagram

Activity diagram digunakan untuk memberikan gambaran mengenai kegiatan yang akan dikerjakan oleh sistem pada fitur tertentu tahap per tahap. Pada tiap fitur tersebut, nantinya akan diberikan alur kerja baik itu pada sisi sistem maupun pada sisi pengguna. Selanjutnya merupakan activity diagram pada berbagai proses yang dilakukan oleh sistem.

#### A. Login

Supaya dapat mengakses fitur yang disediakan sistem, pengguna terlebih dahulu diminta untuk melakukan login pada sistem dengan memberikan data berupa *username* dan *password*. Selanjutnya sistem akan melakukan pengecekan terhadap kedua data tersebut dan melakukan pencocokan. Lalu akan memberikan respon sesuai dengan pengecekan dan pencocokan yang telah dilakukan. Ini akan memberikan sistem kemampuan untuk memanfaatkan sesi untuk pengguna.

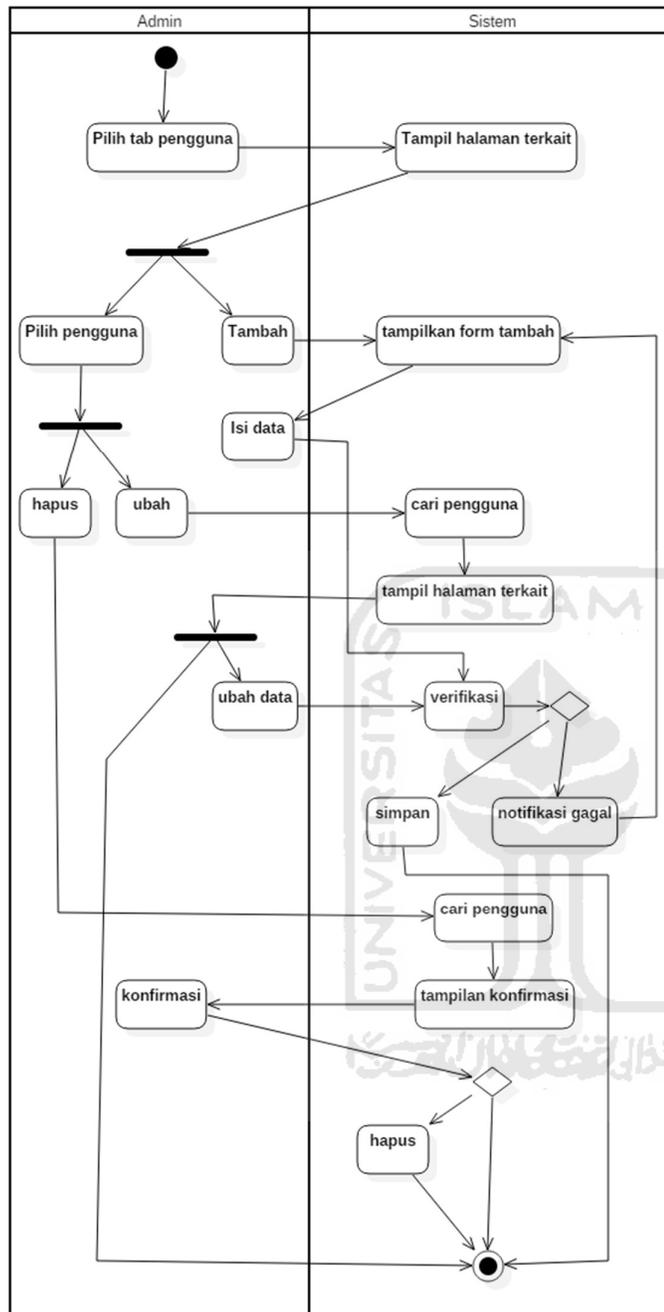




Gambar 3.2 Activity Diagram Login

## B. Manajemen Pengguna

Manajemen pengguna merupakan fitur yang disediakan sistem guna melakukan pengelolaan pengguna yang terdaftar dalam sistem. Fitur ini hanya tersedia untuk Admin, sehingga pengguna (laboran) tidak dapat mengakses fitur. Sehingga Admin perlu terlebih dahulu melakukan login ke dalam sistem supaya sistem mampu mengenalinya.

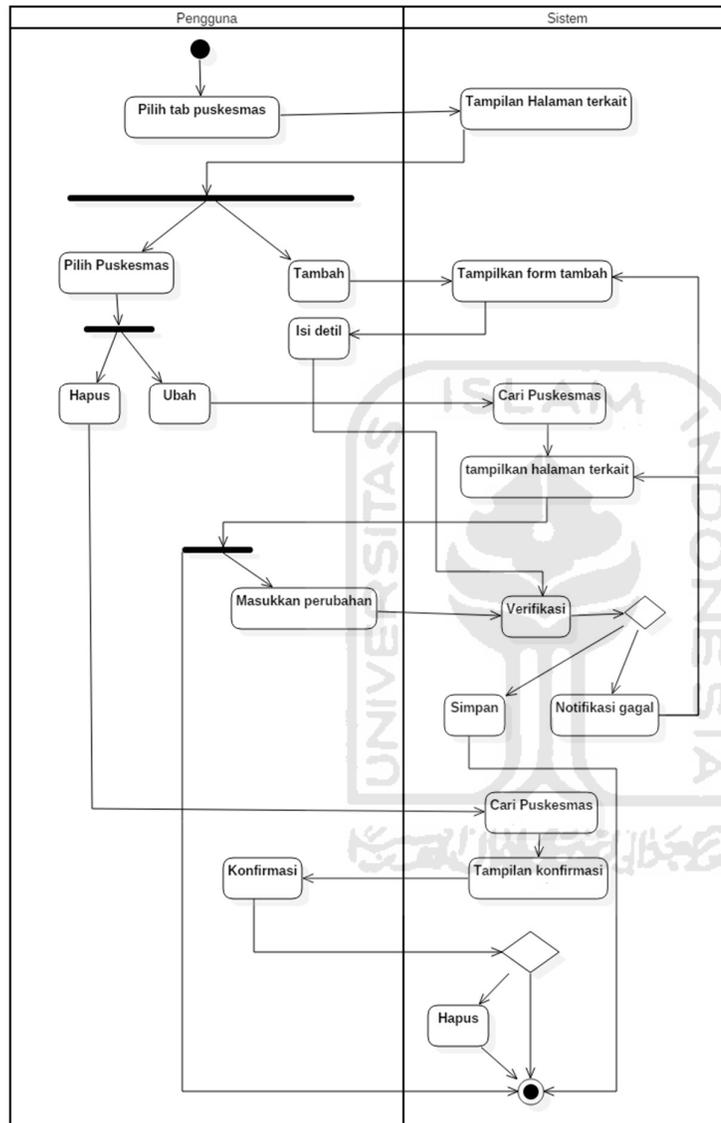


Gambar 3.3 Activity Diagram Manajemen Pengguna

### C. Manajemen Puskesmas

Merupakan aktivitas yang dilakukan untuk melakukan pengelolaan terhadap Puskesmas pada sistem. Pengguna dapat melakukan pemilihan terhadap Puskesmas yang diinginkan atau

menambahkan daftar Puskesmas baru dengan memasukkan beberapa informasi Puskesmas. Sebagaimana yang telah dijelaskan sebelumnya, sebelum mengakses fitur ini, pengguna diperlukan untuk melakukan login terlebih dahulu.

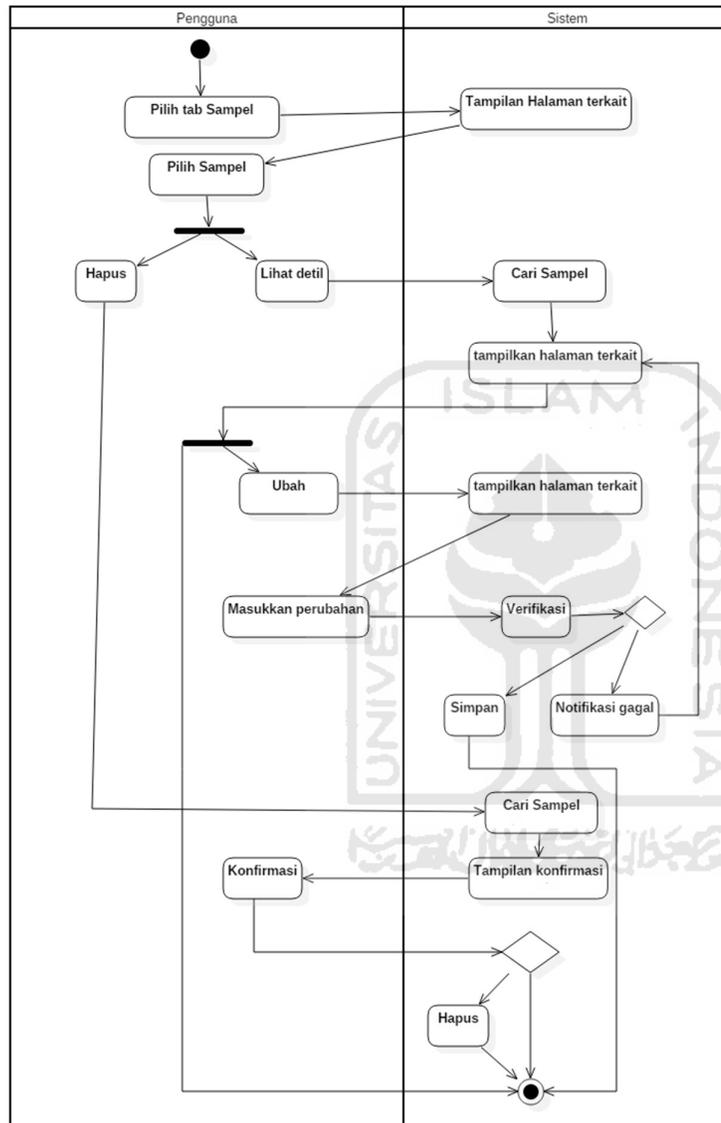


Gambar 3.4 Activity Diagram Manajemen Puskesmas

#### D. Manajemen Sampel

Tidak jauh berbeda dengan aktivitas manajemen puskesmas, manajemen sampel juga digunakan untuk mengelola sampel dengan fitur yang hampir serupa dengan manajemen puskesmas. Perbedaannya terletak pada cara pengguna mengakses fitur yaitu dengan cara

melakukan pencarian terlebih dahulu, baik itu dengan waktu masuk atau asal Puskesmas untuk dapat sampai pada halaman terkait aktivitas ini. Seperti halnya manajemen Puskesmas, pada fitur ini pengguna diharuskan untuk melakukan login terlebih dahulu.

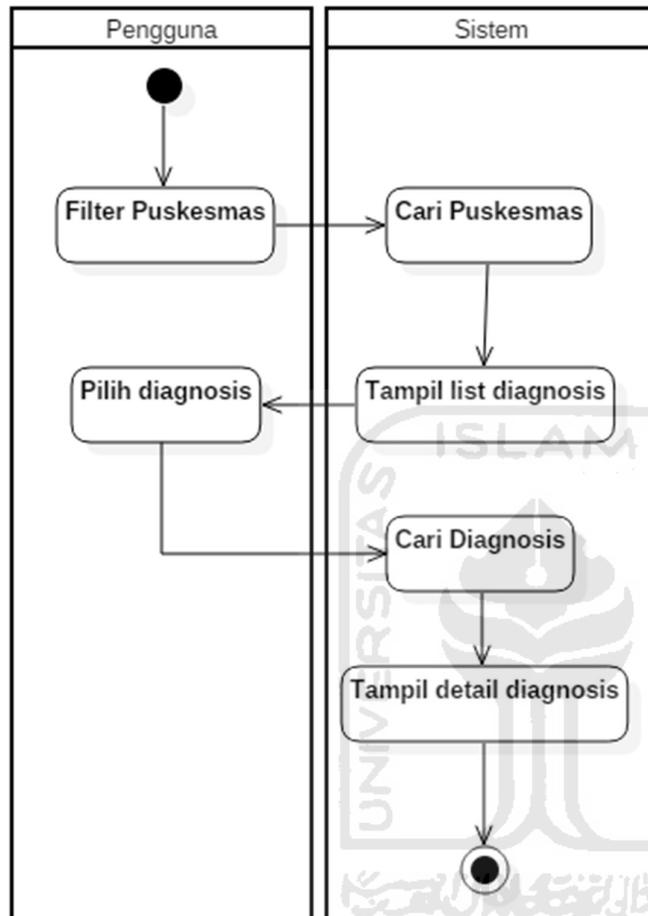


Gambar 3.5 Activity Diagram Manajemen Sampel

### E. Melihat Detil Diagnosis

Pengguna dapat melihat detil diagnosis terhadap citra yang telah dilakukan diagnosis oleh sistem. Pengguna akan terlebih dahulu memilih sampel mana yang ingin dilihat detilnya. Selanjutnya sistem akan memberikan respon sesuai dengan apa yang diminta berupa hasil

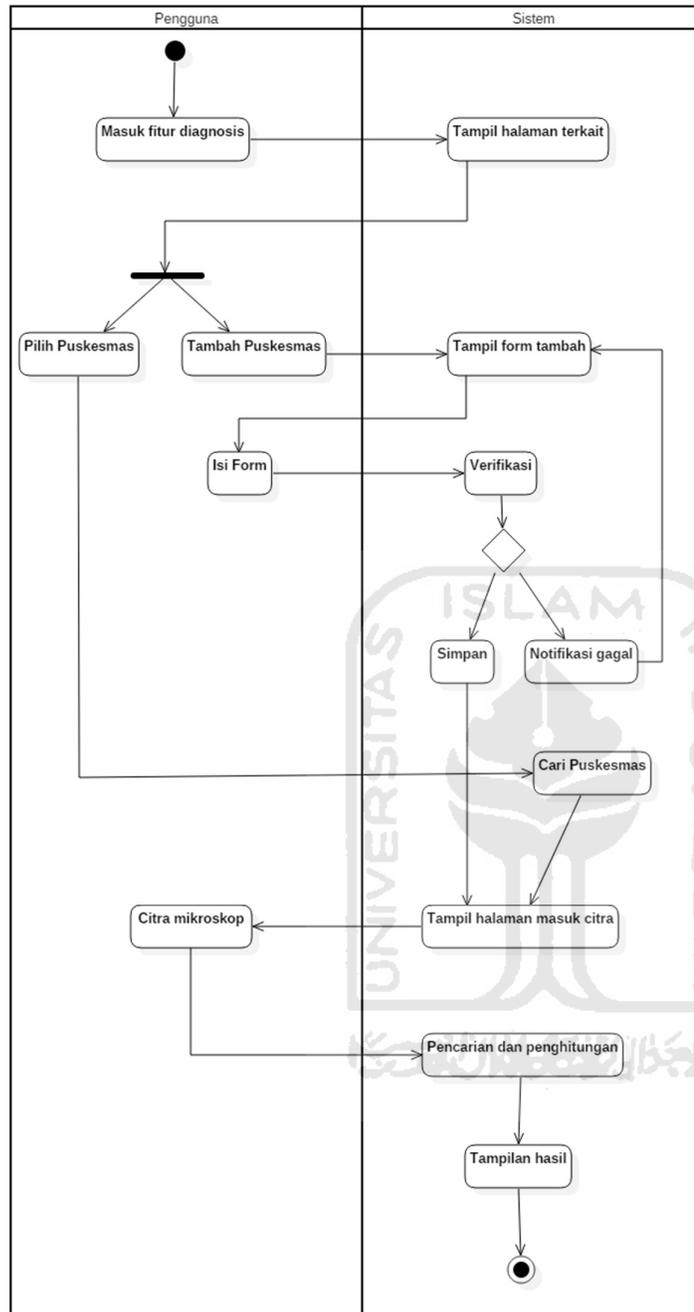
diagnosis, citra yang disimpan serta beberapa informasi terkait. Fitur ini dapat diakses melalui fitur manajemen sampel.



Gambar 3.6 Activity Diagram Lihat Detail

#### F. Diagnosis Citra

Penggana akan diminta terlebih dahulu untuk memilih Puskemas atau memasukkan data Puskesmas baru sebelum melanjutkan proses. Selanjutnya pengguna akan mengunggah citra yang ingin dilakukan diagnosis. Sistem akan memproses masukkan tersebut dan memberikan hasil diagnosis setelah menyelesaikannya.



Gambar 3.7 Activity Diagram Diagnosis Citra

### 3.3.3 Perancangan Basisdata

Perancangan basisdata sebuah sistem didasari atas keperluan dan kebutuhan dari sistem yang sebelumnya telah dijabarkan. Sementara basisdata sendiri adalah sebuah bagian pada sistem yang dibentuk dalam rangka menyimpan data yang dibutuhkan oleh sistem secara

terstruktur. Berikut ini merupakan hasil dari perancangan tersebut yang berupa sebuah relasi tabel dan beberapa struktur tabel dari sistem.

### A. Relasi Tabel

Setidaknya terdapat empat tabel yang akan menyimpan informasi yang sesuai dengan peruntukan masing-masing tabel. Pada gambar relasi ini akan menjabarkan secara singkat bagaimana keseluruhan tabel akan saling berhubungan nantinya. Berikut ini merupakan gambar relasi tabel tersebut.



Gambar 3.8 Tabel Relasi

### B. Struktur Tabel

#### Tabel Pengguna

Tabel Pengguna dimanfaatkan untuk menyimpan informasi terkait pengguna yang diperbolehkan untuk melakukan akses terhadap fitur dari sistem. Selain itu, data pengguna, khususnya id\_pengguna akan digunakan pada tabel sampel sebagai keperluan referensi. Terdiri dari empat kolom dengan kolom id\_pengguna sebagai primary key, nama, username dan password.

Tabel 3.1 Tabel Pengguna

Nama Kolom	Tipe Data	Keterangan
id_pengguna	Varchar (255)	Primary Key
nama	Varchar (255)	

username	Varchar (255)	
password	Varchar (255)	

### Tabel Puskesmas

Tabel Puskesmas digunakan untuk menyimpan informasi yang berkaitan dengan Puskesmas yang menyampaikan sampelnya. Sama seperti tabel pengguna, id\_puskesmas nantinya akan digunakan pada tabel sampel sebagai referensi. Dalam tabel ini, terdapat empat kolom yang berisikan id\_puskesmas, nama, alamat dan telp.

Tabel 3.2 Tabel Puskesmas

Nama Kolom	Tipe Data	Keterangan
id_puskesmas	Varchar	Primary Key
nama	Varchar	
alamat	Varchar	
telp	Varchar	

### Tabel Sampel

Tabel sampel berguna untuk menyimpan informasi mengenai sampel yang masuk dan diagnosis yang didapat. Pada tabel ini, terdapat enam kolom, yaitu id\_sampel, id\_puskesmas, id\_pengguna, waktu, total\_deteksi dan kesimpulan. Terdapat dua kolom referensi, yaitu id\_pengguna dan id\_puskesmas. Sementara kolom id\_sampel akan digunakan sebagai referensi pada tabel citra.

Tabel 3.3 Tabel Sampel

Nama Kolom	Tipe Data	Keterangan
id_sampel	Varchar	Primary Key
id_puskesmas	Varchar	Foreign Key
id_pengguna	Varchar	Foreign Key
waktu	date	
total_deteksi	Int	
kesimpulan	Int	

### Tabel Citra

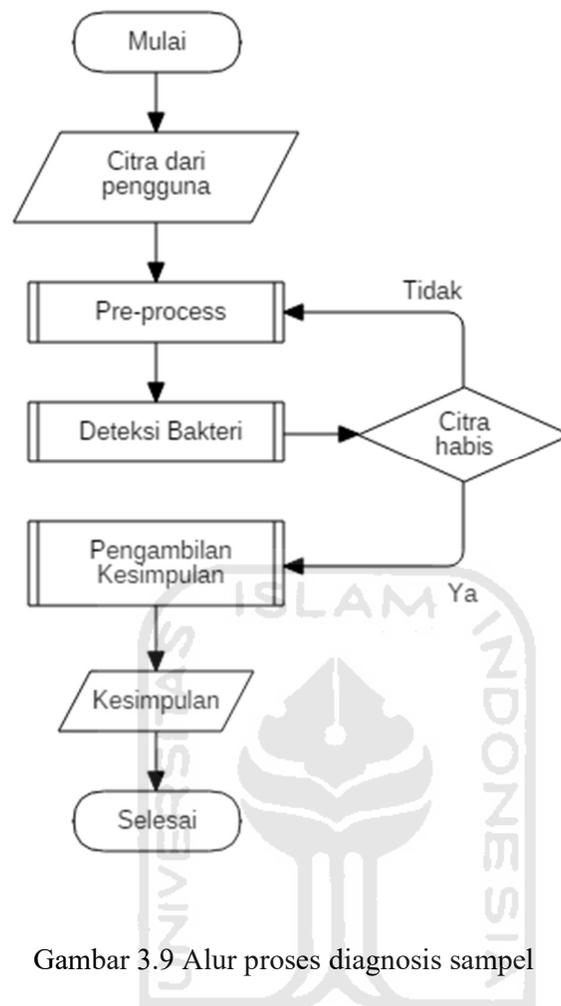
Tabel citra bertujuan untuk menyimpan tiap citra pada tiap sampel yang dimasukkan pada sistem. Tabel ini memiliki empat kolom, yaitu kolom `id_citra`, `id_sampel`, `citra` dan `jml_deteksi`. Terdapat satu kolom referensi yaitu kolom `id_sampel`.

Tabel 3.4 Tabel Citra

Nama Kolom	Tipe Data	Keterangan
<code>id_citra</code>	Varchar	Primary Key
<code>id_sampel</code>	Varchar	Foreign Key
<code>citra</code>	Varchar	Tautan ke citra terkait
<code>jml_deteksi</code>	Int	

### 3.3.4 Perancangan Algoritma Diagnosis

Dalam rangka menyelesaikan tugas yang diberikan kepada sistem, terdapat beberapa tahapan yang perlu dilalui oleh citra yang dimasukkan sebelum akhirnya objek dapat terdeteksi dan kesimpulan dapat diambil. Sebagaimana yang telah dijelaskan pada bab-bab sebelumnya, citra yang masuk akan melalui tahap *pre-processing* terlebih dahulu. Tahap ini berguna untuk menonjolkan beberapa karakteristik yang nantinya akan digunakan dan mengurangi *noise* yang mungkin ada. Selanjutnya, sistem akan mencari dan mengenali objek yang ada di dalam citra terkait dan melakukan pemilihan objek yang dianggap bakteri. Barulah setelah itu sistem akan mengambil kesimpulan melalui jumlah bakteri keseluruhan dari semua citra yang dimasukkan. Berikut ini merupakan gambaran dari proses tersebut yang digambarkan dalam bentuk diagram alir.

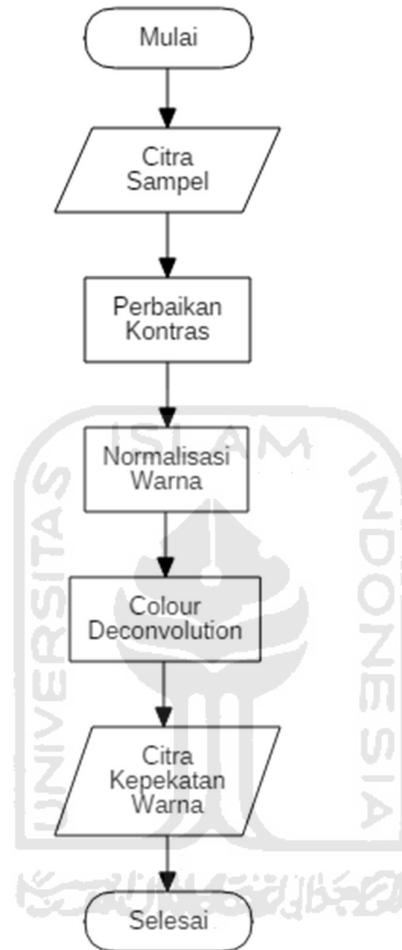


Gambar 3.9 Alur proses diagnosis sampel

### A. Pre-Process

Pada tahapan ini, citra yang masuk akan diolah sedemikian rupa sehingga menghasilkan citra yang cukup representatif ketika nantinya dilakukan pengenalan dan pemilihan objek pada proses berikutnya. Hal pertama yang dilakukan terhadap citra adalah dengan melakukan perbaikan kontras warna dari citra. Proses ini bertujuan untuk menonjolkan karakteristik dari objek serta latar belakang dari citra demi memudahkan sistem dalam menentukan objek kedepannya. Selanjutnya, warna dari citra akan dinormalisasi pada tiap *channel* warna pada citra tersebut. Hal ini dilakukan dengan maksud menyelaraskan pewarnaan dari citra, karena dari beberapa citra yang didapatkan, tidak semua citra memiliki pewarnaan yang sama atau mirip. Setelah dilakukan normalisasi, citra akan dimasukkan ke dalam algoritma *colour deconvolution* yang bertujuan untuk menemukan kepekatan dari dua pewarnaan yang diberikan pada sampel (merah muda dan biru). Proses ini akan menghasilkan dua buah citra kepekatan yang masing-masing menggambarkan kepekatan dari warna yang dimaksudkan. Mengingat

bakteri *mycobacterium tuberculosis* akan berwarna merah muda pada citra, pada tahapan selanjutnya hanya citra kepekatan dari warna merah muda saja yang dikerjakan oleh sistem.

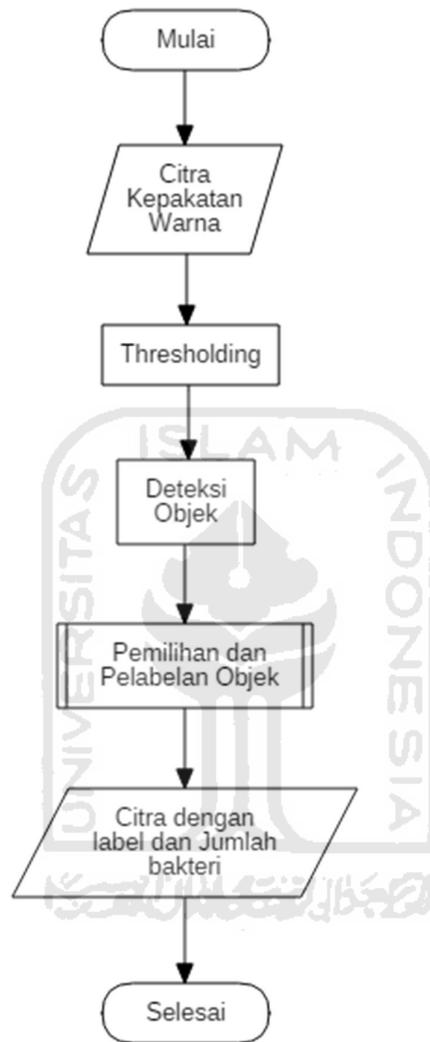


Gambar 3.10 Alur *pre-process*

## B. Deteksi Bakteri

Menggunakan citra kepekatan warna yang didapatkan dari proses sebelumnya, pendeteksian bakteri dimulai dengan melakukan *thresholding* untuk mendapatkan citra binari dari citra tersebut. Setelah citra binari didapat, sistem akan mengambil objek-objek yang terdapat pada citra binari tersebut. Objek-objek tadi nantinya akan dipilih dan diberikan label pada citra masukkan bila objek tersebut memenuhi kriteria yang diinginkan. Dari sini, sistem akan memberikan citra yang sudah diberikan label serta nilai dari seberapa banyak objek yang

dianggap sebagai bakteri oleh sistem, untuk digunakan pada tahapan pengambilan kesimpulan yang akan dikerjakan selanjutnya.

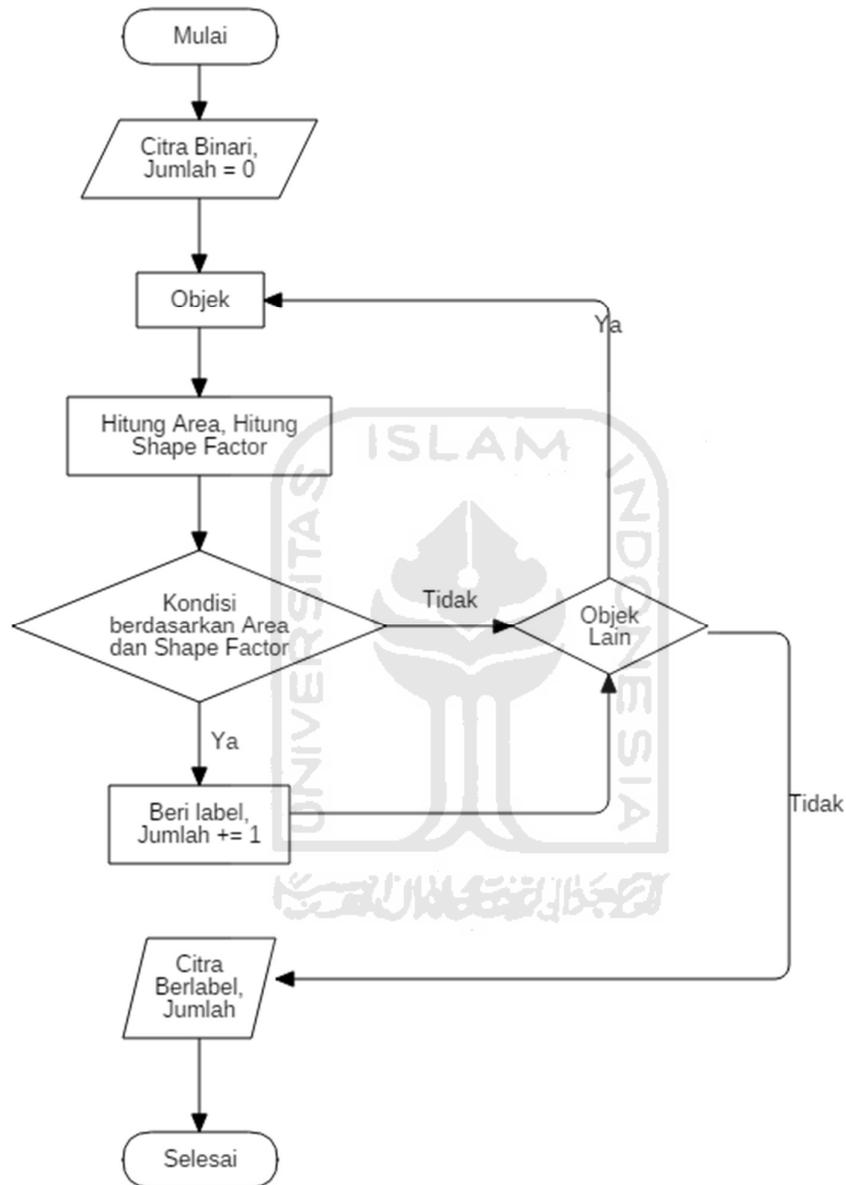


Gambar 3.11 Alur deteksi bakteri

### C. Pemilihan dan Pelabelan Objek

Tahapan ini berfungsi untuk menentukan objek mana saja yang sebelumnya telah ditemukan, yang termasuk atau dianggap sebagai objek bakteri oleh sistem. Mulanya, nilai Area dan *Shape Factor* dari objek akan dihitung. Selanjutnya, objek dengan nilai Area dan *Shape Factor* tertentu akan diberikan label dan nilai dari jumlah bakteri terdeteksi akan ditambah satu. Proses ini dilakukan pada setiap objek yang ditemukan pada citra binari, hingga

akhirnya proses akan mengeluarkan citra yang telah diberikan label serta jumlah bakteri yang ditemukan pada citra tersebut.



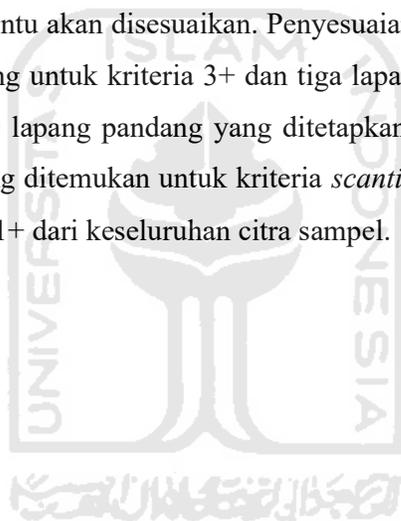
Gambar 3.12 Alur pemilihan dan pelabelan objek

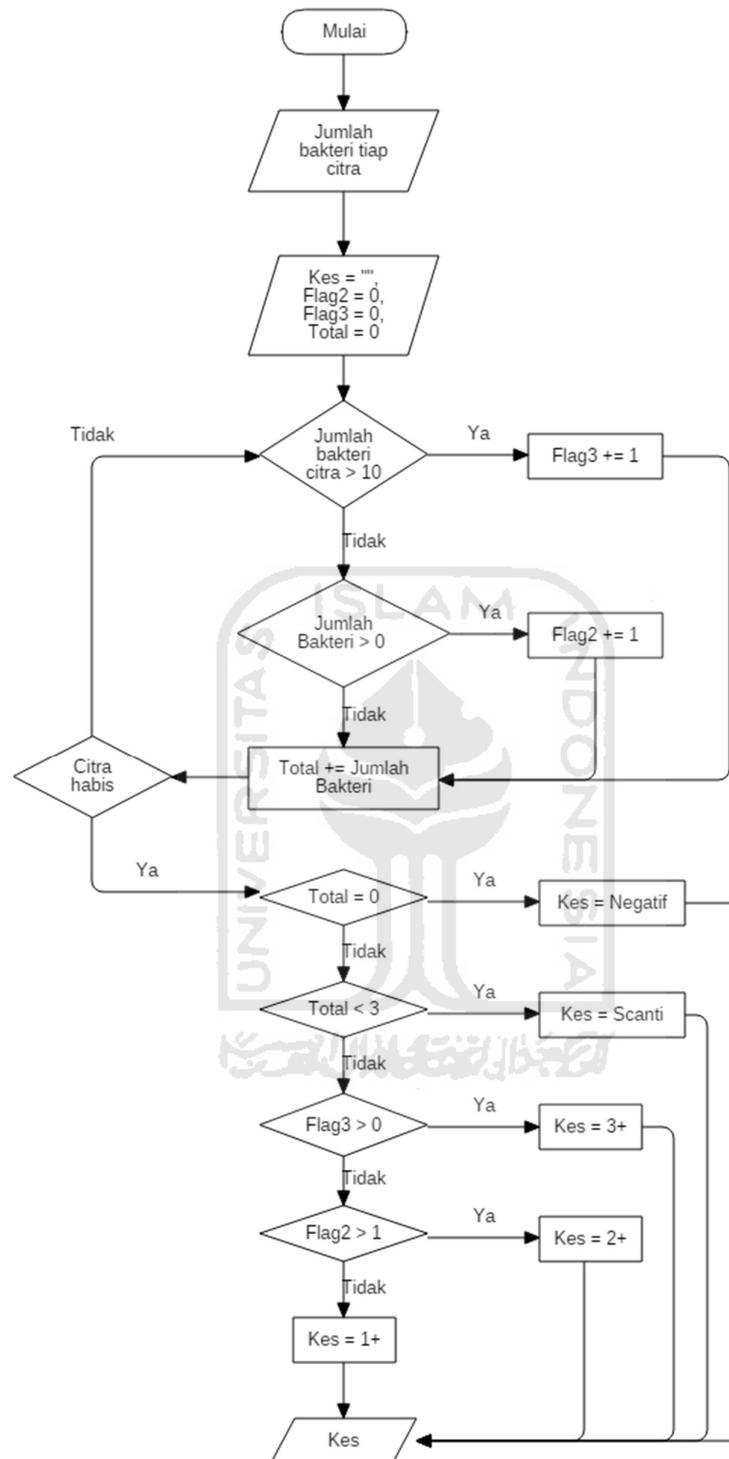
#### D. Pengambilan Kesimpulan

Setelah semua citra yang dimasukkan oleh pengguna diolah, sistem akan melakukan penghitungan terhadap hasil yang didapat sebelumnya. Merujuk pada panduan yang

dikeluarkan IUATLD seperti pada Tabel 2.1 sebelumnya, pada tahap ini terdapat beberapa hal yang perlu diperhatikan untuk mengambil kesimpulan. Pertama adalah jumlah total dari bakteri yang ditemukan pada keseluruhan citra. Kedua adalah jumlah lapang pandang yang terindikasi 3+. Sementara yang terakhir adalah jumlah lapang pandang yang terindikasi 2+. Melalui tiga poin ini nantinya, sistem akan menentukan tingkat keparahan dari sampel yang diberikan.

Sebelumnya, dikarenakan keterbatasan sumber daya komputasi yang ada, maka dalam melakukan diagnosis ini, sistem akan menggunakan pemodelan yang sekiranya mampu menggambarkan kondisi aktual yang diperlukan. Pemodelan ini berupa pembatasan jumlah citra yang diperlukan untuk melakukan diagnosis yang sebelumnya diperlukan 100 lapang pandang per sampel, pada sistem saat ini akan dibatasi dengan hanya menerima lima lapang pandang per sampel. Tentunya dengan pembatasan ini, kondisi yang diperlukan untuk memenuhi tingkat keparahan tertentu akan disesuaikan. Penyesuaian dilakukan dengan hanya membutuhkan satu lapang pandang untuk kriteria 3+ dan tiga lapang pandang untuk kriteria 2+ berbanding dengan 20 dan 50 lapang pandang yang ditetapkan pada panduan IUATLD. Sementara itu, jumlah bakteri yang ditemukan untuk kriteria *scanti* juga disesuaikan menjadi hanya satu dan dua bakteri untuk 1+ dari keseluruhan citra sampel.





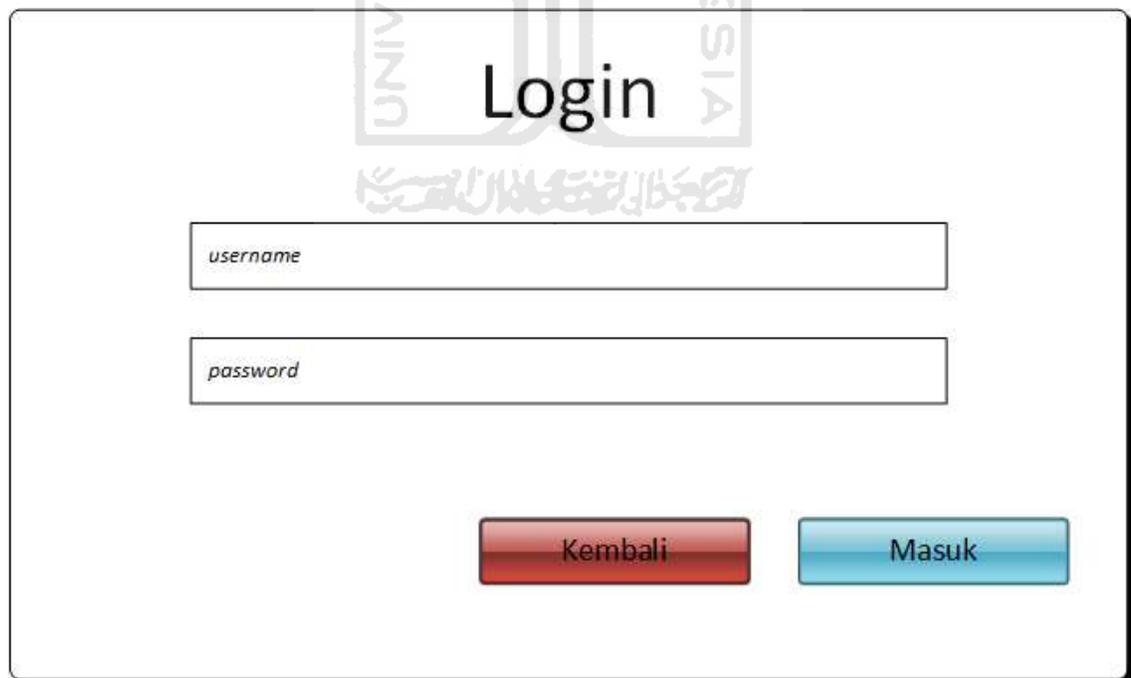
Gambar 3.13 Alur pengambilan kesimpulan

### 3.3.5 Perancangan Antarmuka

Melalui pembahasan sebelumnya yang menjelaskan mengenai Analisis sistem, telah didapatkan beberapa antarmuka yang nantinya diperlukan bagi sistem untuk menyampaikan informasi pada pengguna. Pada bagian perancangan ini akan diberikan gambaran secara umum bagaimana sistem akan menampilkan informasinya. Terdapat beberapa bagian antarmuka yang diperlukan, yang telah disebutkan pada bagian analisis sebelumnya. Bagian-bagian tersebut adalah antarmuka *login*, antarmuka halaman utama, antarmuka manajemen Puskesmas, antarmuka manajemen sampel, antarmuka diagnosis dan antarmuka hasil. Berikut ini merupakan rancangan yang telah dibentuk untuk tiap-tiap bagian antarmuka.

#### A. Halaman Login

Antarmuka Login, merupakan halaman yang digunakan oleh pengguna agar dapat mengakses fitur lain dalam sistem. Di sini pengguna akan diberikan sebuah formulir yang akan diisi berdasarkan *username* serta *password* yang sebelumnya sudah didaftarkan. Melalui antarmuka ini Sistem mampu memberikan akses tertentu atau khusus kepada pengguna terdaftar melalui sesi.



The image shows a login interface design. At the top center, the word "Login" is displayed in a large, bold, black font. Below the title, there are two input fields: the first is labeled "username" and the second is labeled "password". At the bottom of the form, there are two buttons: a red button labeled "Kembali" (Back) and a blue button labeled "Masuk" (Login).

Gambar 3.14 Rancangan antarmuka Login

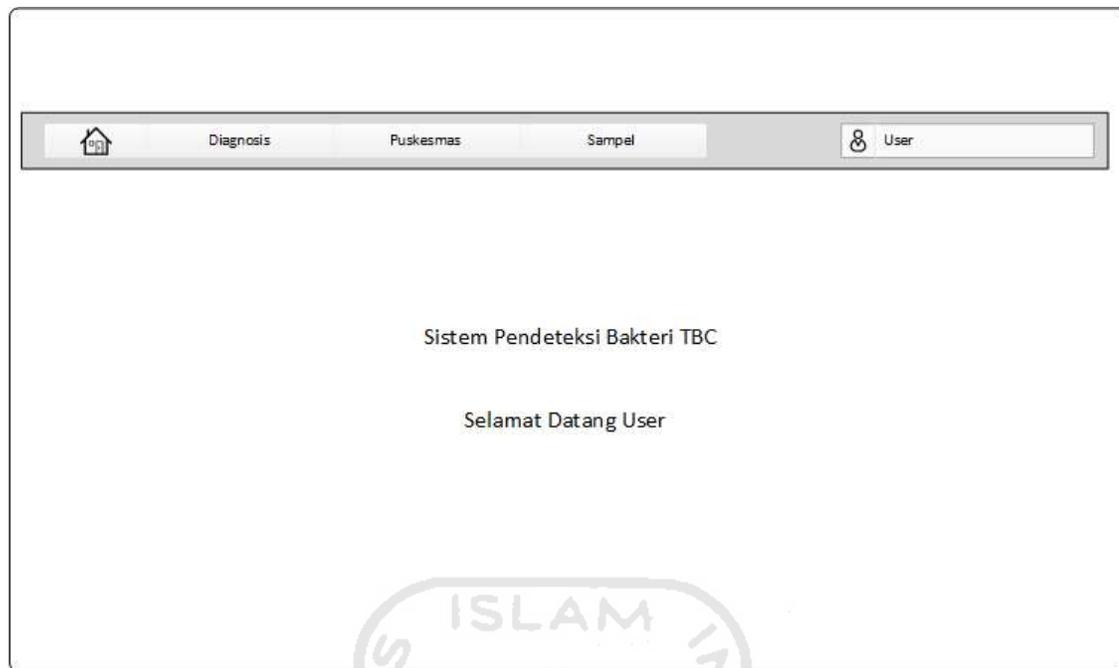
## B. Halaman Utama

Antarmuka halaman utama, merupakan halaman hub bagi pengguna untuk mengakses ke fitur yang disediakan sistem. Sebagaimana yang telah dituliskan, pada halaman ini terdapat tautan-tautan yang akan menuntun pengguna untuk mengakses fitur-fitur yang disediakan. Dengan memanfaatkan sesi pengguna, nantinya halaman ini juga akan menyediakan tampilan yang menyesuaikan dengan pengguna yang menggunakan.

Halaman utama yang terbagi menjadi dua jenis seperti yang ditunjukkan oleh gambar 3.15 dan gambar 3.16 bertujuan untuk menyesuaikan tampilan navigasi sesuai sesi yang sedang berlangsung. Pada gambar 3.15, halaman utama difungsikan saat belum terjadi login pada sistem, sehingga akses pada fitur terbatas. Sementara untuk gambar 3.16, ketika user telah melakukan login, sistem akan menampilkan navigasi yang sesuai dengan user yang mengaksesnya. Hal ini bertujuan supaya hanya user yang telah terdaftar yang mampu mengakses fitur pada sistem.



Gambar 3.15 Rancangan antarmuka Halaman Utama 1



Gambar 3.16 Rancangan antarmuka Halaman Utama 2

### C. Halaman Manajemen Puskesmas

Antarmuka manajemen Puskesmas, merupakan halaman untuk melakukan pengelolaan terhadap data Puskesmas. Halaman akan menampilkan daftar Puskesmas terdaftar berikut aksi yang bisa dilakukan. Pada halaman ini juga, pengguna dapat mengetahui detail Puskesmas yang telah terdaftar dalam sistem. Terdapat dua halaman yang berkaitan dengan manajemen Puskesmas ini. Pertama adalah halaman yang digunakan oleh pengguna untuk melihat daftar Puskesmas yang telah tersimpan. Sedangkan halaman selanjutnya adalah halaman yang digunakan untuk melakukan pendaftaran atau menambah puskesmas pada sistem.

Navigation: Home | **Diagnosis** | Puskesmas | Sampel | User

### Puskesmas

Nama	Alamat	Telpon
Puskesmas 1	Jl raya 1	564721
Puskesmas 2	Jl raya 2	564722
Puskesmas 3	Jl raya 3	564723
Puskesmas 4	Jl raya 4	564724

Buttons: **Tambah**, **Hapus**, **Ubah**

Gambar 3.17 Rancangan antarmuka Puskesmas 1

Navigation: Home | **Diagnosis** | Puskesmas | Sampel | User

### Puskesmas

Nama :

Alamat :

Telpon :

Buttons: **Kembali**, **Simpan**

Gambar 3.18 Rancangan antarmuka Puskesmas 2

Pada gambar 3.17, Puskemas yang telah terdaftar akan ditampilkan dalam bentuk tabel dengan menampilkan alamat serta nomor telepon untuk melengkapi informasi yang diberikan

kepada pengguna. Pengguna mampu melakukan penambahan Puskesmas terdaftar dengan menekan tombol 'Tambah' yang telah disediakan. Selain itu, pengguna dapat menghapus atau mengubah informasi Puskesmas dengan memilih tombol di samping baris Puskesmas terkait.

Jika pengguna menekan tombol 'Ubah' pada baris Puskesmas tertentu, maka halaman seperti yang digambarkan oleh gambar 3.18 akan ditampilkan. Halaman ini akan berisikan informasi Puskesmas yang telah tersimpan di dalam basisdata di mana pengguna berkemampuan untuk melakukan perubahan pada data Puskesmas seperti nama, alamat dan nomor telepon. Tombol 'Kembali' dan 'Simpan' disediakan untuk menggugurkan atau menyimpan perubahan yang sebelumnya dilakukan pengguna.

Tampilan pada gambar 3.18 juga akan digunakan untuk mendaftarkan Puskesmas baru pada sistem. Hanya saja, kolom nama, alamat dan nomor telepon akan kosong, tidak seperti kasus sebelumnya di mana kolom tersebut berisikan data pada Puskesmas terkait. Pengguna nantinya dapat mengisi data tersebut dan menyimpannya dengan menekan tombol 'Simpan' atau menggugurkannya dengan tombol 'Kembali'.

#### **D. Halaman Manajemen Sampel**

Antarmuka manajemen sampel, merupakan halaman yang digunakan untuk melakukan pengelolaan terhadap sampel, baik itu menambah, melihat detil atau menghapus. Tidak jauh berbeda dengan halaman manajemen Puskesmas, halaman akan menampilkan daftar sampel dan aksi yang bisa dilakukan. Selain itu, pada tiap sampel nantinya akan diberikan tautan yang akan mengantarkan pengguna pada halaman hasil sampel terkait. Pada antarmuka manajemen sampel, terdapat tiga halaman yang berkaitan dengan manajemen sampel. Halaman pertama adalah halaman yang akan menampilkan daftar sampel yang telah tersimpan. Halaman kedua adalah halaman yang berisikan informasi detil untuk sampel terkait. Sementara untuk halaman ketiga adalah halaman yang dapat digunakan pengguna untuk mengubah sampel yang telah tersimpan.

The screenshot shows a web interface with a navigation bar containing 'Diagnosis', 'Puskesmas', and 'Sampel' tabs, and a 'User' profile icon. Below the navigation bar, the title 'Sampel' is displayed. A table lists four samples with columns for 'No', 'Asal', and 'Hasil'. To the right of the table, there are four pairs of buttons: a red 'Hapus' button and a blue 'Detil' button for each row.

No	Asal	Hasil
No 1	Puskesmas 1	Hasil 1
No 2	Puskesmas 4	Hasil 1
No 3	Puskesmas 2	Hasil 1
No 4	Puskesmas 4	Hasil 4

Gambar 3.19 Rancangan antarmuka Sampel 1

The screenshot shows a web interface with a navigation bar containing 'Diagnosis', 'Puskesmas', and 'Sampel' tabs, and a 'User' profile icon. Below the navigation bar, the title 'Detil Hasil' is displayed. There are three input fields: 'Asal' with the placeholder 'asal', 'No' with the placeholder 'nomor sampel', and 'Hasil'. Below the 'Asal' field is a large empty box labeled 'Citra'. At the bottom right, there are two buttons: a red 'Kembali' button and a blue 'Ubah' button.

Gambar 3.20 Rancangan antarmuka Sampel 2

Gambar 3.21 Rancangan antarmuka Sampel 3

Sama halnya dengan halaman manajemen Puskesmas, data sampel akan ditampilkan dalam bentuk tabel. Selain itu, pada halaman ini tidak disediakan tombol 'Tambah' untuk menambah langsung sampel baru dan tombol 'Ubah' yang disesuaikan menjadi tombol 'Detil'. Tombol 'Detil' inilah yang akan mengantarkan pengguna pada halaman seperti gambar 3.20.

Gambar 3.20 akan menampilkan detil hasil dari sampel tertentu pada pengguna. Data yang ditampilkan bersifat statis, sehingga pengguna tidak mampu melakukan perubahan pada kolom-kolom yang ditampilkan. Untuk melakukan perubahan, pengguna diminta untuk menekan tombol 'Ubah' yang telah disediakan, di mana Sistem akan menampilkan halaman yang tidak jauh berbeda, seperti yang ditunjukkan oleh gambar 3.21. Bedanya, data yang ditampilkan dapat diganti oleh pengguna dan disimpan ke dalam basis data.

### E. Halaman Diagnosis

Antarmuka diagnosis sampel, merupakan halaman yang digunakan sebagai pintu masuk citra sampel yang nantinya akan didiagnosis. Pengguna akan memberikan masukan lain berupa informasi yang relevan terhadap sampel yang diberikan. Halaman ini menjadi tahapan awal selama proses diagnosis citra oleh komputer, di mana citra yang dimaksudkan akan diunggah oleh pengguna untuk nantinya dilakukan diagnosis oleh sistem.

The image shows a web application interface for a diagnosis system. At the top, there is a navigation bar with three tabs: 'Diagnosis', 'Puskesmas', and 'Sampel'. To the right of the tabs is a user profile icon labeled 'User'. Below the navigation bar, the main content area is titled 'Diagnosis'. It contains a form with two input fields: 'Asal' (a dropdown menu with 'asal' selected) and 'No' (a text input with 'nomor sampel'). There is a plus sign between these two fields. Below the 'Asal' field is a 'Citra' label and a large empty rectangular area for image upload. At the bottom right of this area is a button labeled 'Unggah Gambar'. Further right are two buttons: a red 'Ulang' button and a blue 'Diagnosis' button. A watermark for 'UNIVERSITAS ISLAM' is visible in the background.

Gambar 3.22 Rancangan antarmuka Diagnosis

Pada gambar 3.22, ditunjukkan halaman yang akan digunakan pengguna untuk melakukan diagnosis sampel. Pengguna diminta untuk terlebih dahulu mengisi beberapa kolom yang berkaitan dengan data sampel. Kolom asal berbentuk dropdown menu dengan opsi lain yaitu menambahkan Puskesmas yang sekiranya belum terdaftar. Selanjutnya pengguna dapat mengunggah citra sampel pada area yang telah disediakan oleh sistem. Ketika pengguna menekan tombol 'Diagnosis' sistem akan melakukan proses diagnosis yang selanjutnya akan mengarahkan pengguna pada halaman hasil.

#### **F. Halaman Hasil**

Antarmuka hasil, merupakan halaman yang akan menampilkan hasil diagnosis dari citra yang sebelumnya dimasukkan. Pada halaman ini, sistem akan menyampaikan informasi terkait hasil diagnosis berupa jumlah bakteri yang terdeteksi, tingkat keparahan dan informasi Puskesmas sebagai informasi tambahan. Halaman ini yang nantinya akan digunakan pengguna untuk memahami hasil yang diselesaikan oleh sistem.

The image shows a web application interface for displaying diagnosis results. At the top, there is a navigation bar with three tabs: 'Diagnosis', 'Puskesmas', and 'Sampel'. To the right of the tabs is a user profile icon labeled 'User'. Below the navigation bar, the main content area is titled 'Diagnosis'. It contains several input fields: 'Asal' with the value 'asal', 'No' with the value 'nomor sampel', and 'Hasil'. There are also two large empty rectangular boxes, one labeled 'Citra' and one labeled 'Hasil'. At the bottom right of the main content area, there are two buttons: a red button labeled 'Ulang' and a blue button labeled 'Simpan'. A watermark for 'UNIVERSITAS ISLAM' is visible in the background.

Gambar 3.23 Rancangan antarmuka Hasil

Gambar 3.23 menyediakan informasi mengenai hasil dari diagnosis yang baru saja dilakukan oleh sistem. Data yang ditampilkan bersifat statis, sehingga pengguna tidak dapat melakukan perubahan pada data yang ditampilkan pada halaman. Pengguna diberikan opsi untuk mengulang proses diagnosis dengan menekan tombol 'Ulang' atau menyimpan hasil diagnosis ke dalam basis data dengan menekan tombol 'Simpan'.

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Implementasi Sistem**

Pada bagian implementasi sistem, akan dijabarkan mengenai realisasi dari keseluruhan perancangan yang telah dijelaskan pada bab-bab sebelumnya, yang kemudian akan dilakukan pengujian untuk mengetahui keakuratan dalam menentukan bakteri dari sistem yang telah dibangun pada bagian berikutnya. Penjabaran akan dilakukan secara bertahap mulai dari bagian diagnosis hingga pengelolaan data yang tersimpan pada sistem. Selanjutnya akan dimulai penjelasan mengenai implementasi dari Sistem Pendeteksi Bakteri Mycobacterium Tuberculosis dan Penentu Tingkat Penyakit TBC, sesuai dengan perancangan yang telah dilakukan sebelumnya.

##### **4.1.1 Halaman Login**

Sebagaimana yang telah dijelaskan pada bab perancangan terkait, halaman ini akan digunakan sebagai pintu masuk untuk mengakses berbagai fasilitas yang disediakan oleh sistem. Sehingga tahap ini menjadi hal krusial bagi pengguna sebelum pengguna mampu memanfaatkan fitur pada sistem. Berikut ini merupakan gambaran aktual pada sistem yang telah dibangun untuk halaman login seperti pada Gambar 4.1 berikut.

Username

Password

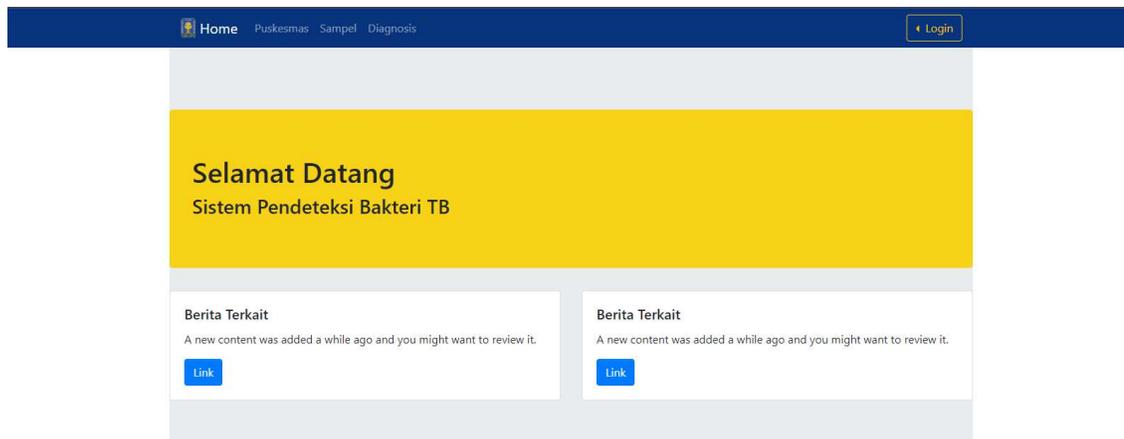
Remember me

[Sign in](#)

Gambar 4.1 Antarmuka Login

#### 4.1.2 Halaman Utama

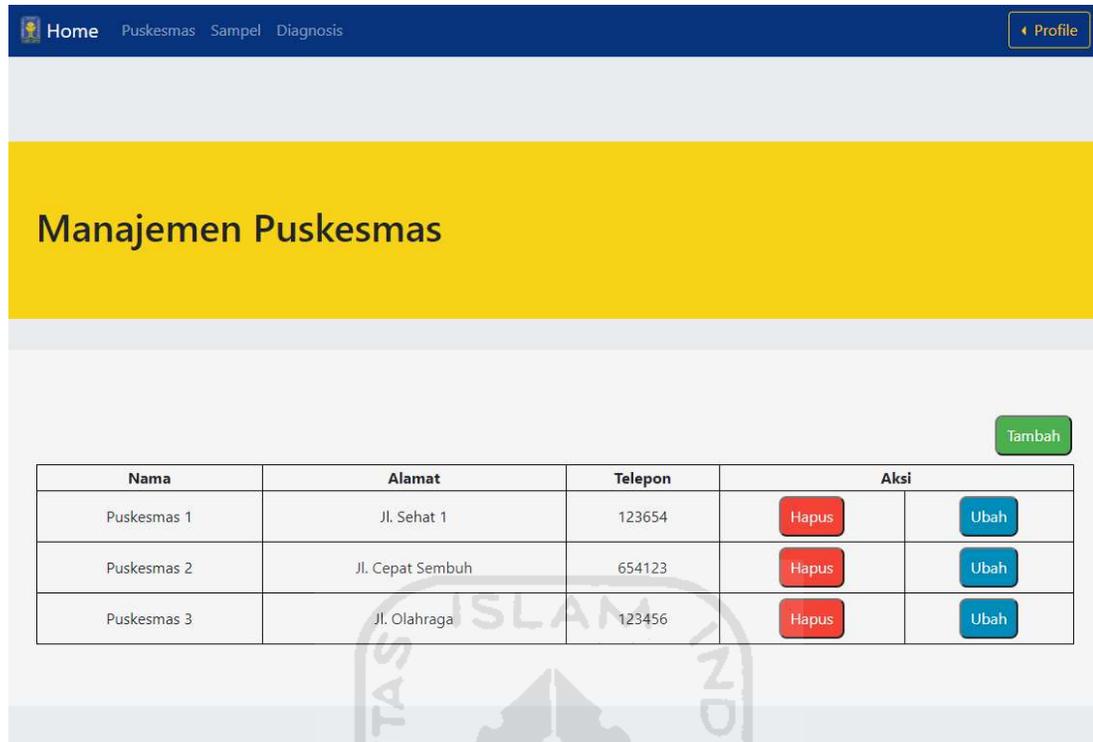
Halaman utama akan ditampilkan setelah pengguna, dalam hal ini laboran, melakukan login ke dalam sistem. Pada halaman ini, pengguna mampu berpindah dari satu fitur ke fitur lainnya melalui tautan yang ditampilkan. Halaman ini juga akan menampilkan beberapa informasi mengenai kegiatan yang terkini yang telah dilakukan laboran terkait. Berikut merupakan gambaran mengenai halaman yang dimaksud seperti pada Gambar 4.2 di bawah.



Gambar 4.2 Antarmuka Halaman Utama

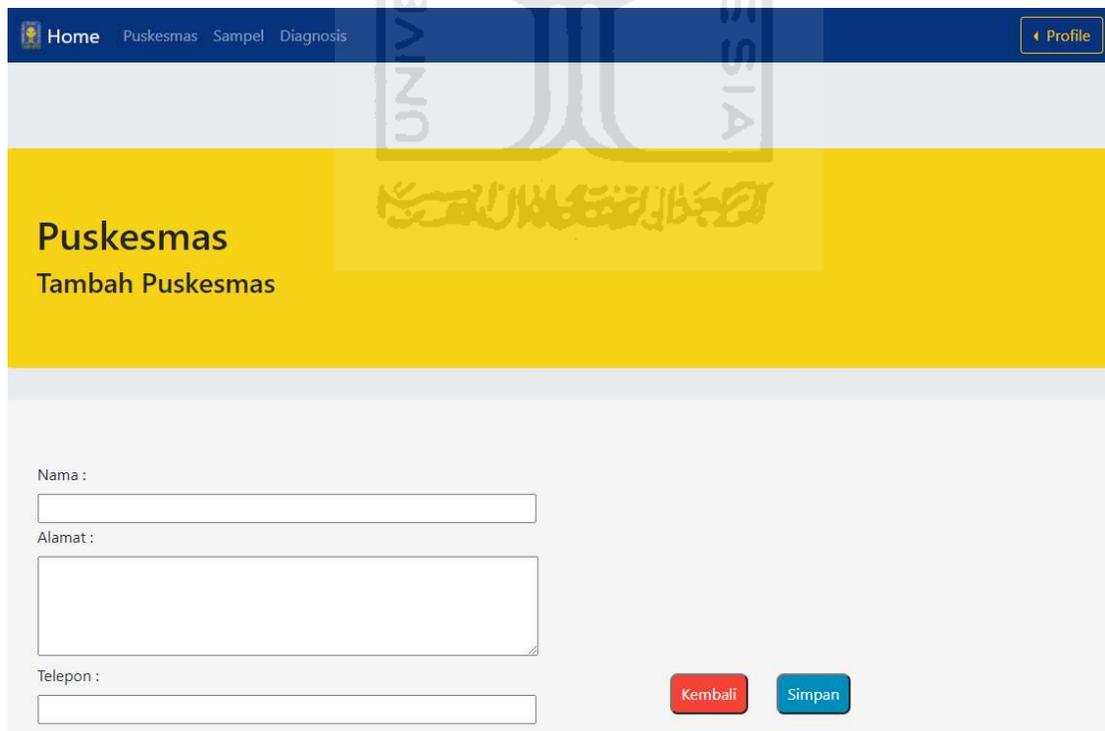
### 4.1.3 Halaman Manajemen Puskesmas

Halaman antarmuka Manajemen Puskesmas merupakan halaman yang berisi mengenai informasi Puskesmas yang telah terdaftar dalam sistem. Pada halaman ini akan terdapat daftar Puskesmas beserta beberapa informasi penting lainnya. Halaman juga menyediakan sarana bagi pengguna untuk melakukan penambahan, perubahan ataupun penghapusan puskesmas pada sistem.



Nama	Alamat	Telepon	Aksi	
Puskesmas 1	Jl. Sehat 1	123654	Hapus	Ubah
Puskesmas 2	Jl. Cepat Sembuh	654123	Hapus	Ubah
Puskesmas 3	Jl. Olahraga	123456	Hapus	Ubah

Gambar 4.3 Antarmuka Manajemen Puskesmas



Nama :

Alamat :

Telepon :

Kembali Simpan

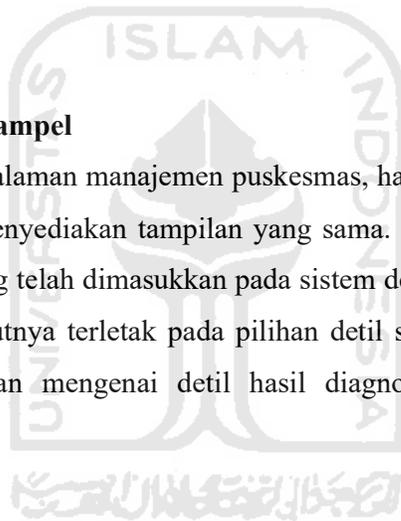
Gambar 4.4 Antarmuka Tambah Puskesmas

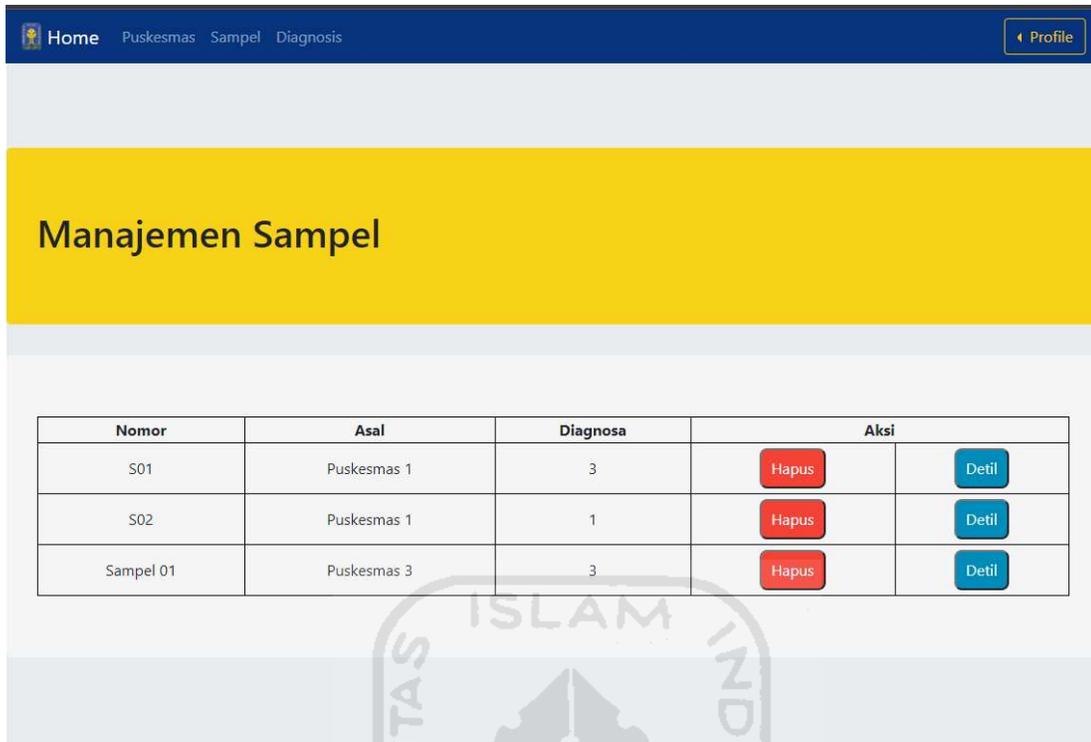
Pada Gambar 4.3, halaman menyediakan informasi kepada pengguna mengenai puskesmas yang telah terdaftar pada sistem dalam bentuk tabel sederhana. Pada halaman ini pengguna dapat menemukan informasi mengenai nama puskesmas, alamat dan nomor telepon yang bisa dihubungi. Selain itu, sistem juga menampilkan tombol yang dapat digunakan oleh pengguna untuk melakukan penambahan, pengubahan atau penghapusan data puskesmas dengan menekan tombol terkait.

Sementara itu, Gambar 4.4 menunjukkan mengenai halaman yang akan digunakan oleh pengguna selama melakukan penambahan atau pengubahan data puskesmas. Halaman memberikan informasi lengkap kepada pengguna di mana nanti bergantung pada kondisinya, pengguna mampu mengubah atau menambahkan isian pada kolom yang ditampilkan. Selanjutnya pengguna hanya perlu menyimpan masukan yang telah diberikan pada tombol terkait di halaman.

#### **4.1.4 Halaman Manajemen Sampel**

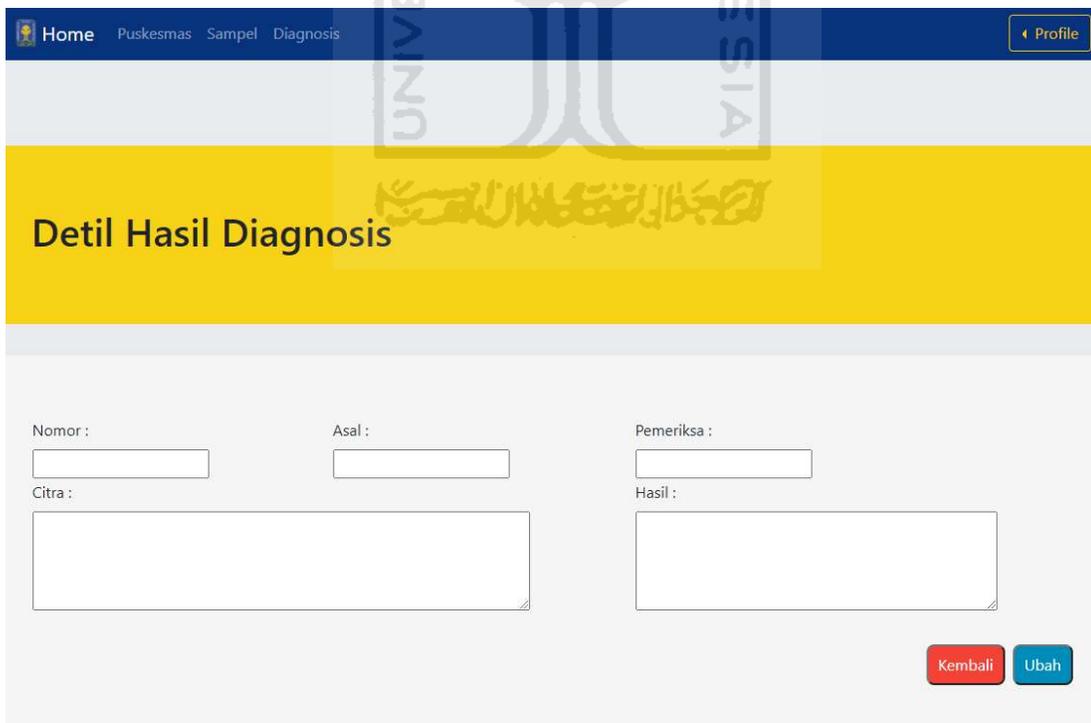
Tidak jauh berbeda dengan halaman manajemen puskesmas, halaman manajemen sampel seperti pada Gambar 4.5 juga menyediakan tampilan yang sama. Hanya saja halaman akan menampilkan sampel-sampel yang telah dimasukkan pada sistem dengan informasi mengenai sampel terkait. Perbedaan selanjutnya terletak pada pilihan detil sampel di mana pengguna nantinya akan diberikan halaman mengenai detil hasil diagnosis sampel seperti yang ditunjukkan oleh Gambar 4.6.





Nomor	Asal	Diagnosa	Aksi	
S01	Puskesmas 1	3	Hapus	Detil
S02	Puskesmas 1	1	Hapus	Detil
Sampel 01	Puskesmas 3	3	Hapus	Detil

Gambar 4.5 Antarmuka Manajemen Sampel



Nomor :

Asal :

Pemeriksa :

Citra :

Hasil :

Gambar 4.6 Antarmuka Detil Sampel

Home Puskesmas Sampel Diagnosis Profile

## Detil Hasil Diagnosis

Ubah Informasi

Nomor :

Asal :

Pemeriksa :

Citra :

Hasil :

Kembali Simpan

Gambar 4.7 Antarmuka Ubah Detil

Sementara itu, pengguna dapat melakukan perubahan beberapa informasi sampel melalui halaman seperti pada Gambar 4.7. Halaman ini dapat dibuka melalui halaman detil hasil (Gambar 4.6). Namun pengguna hanya dapat merubah beberapa informasi saja, seperti nomor sampel, asal sampel dan laboran pemeriksa.

#### 4.1.5 Halaman Diagnosis

Halaman diagnosis digunakan pengguna untuk melakukan diagnosis pada sampel yang telah diambil citranya. Pengguna perlu mengunggah citra sampel pada kolom yang telah disediakan. Selain itu, informasi mengenai nomor sampel, asal, serta laboran pemeriksa perlu dimasukkan ke dalam kolom terkait oleh pengguna. Sementara itu, untuk mengurangi kemungkinan kesalahan yang terjadi ketika melakukan pengunggahan citra mikroskopik *sputum*, pengguna akan diminta untuk memasukkan kumpulan citra tersebut dalam bentuk satu berkas file per sampel *sputum*. Halaman ini akan tampak seperti pada Gambar 4.8.

Gambar 4.8 Antarmuka Diagnosis

Selain terjadi proses *input* data dari pengguna, pada halaman ini nantinya sistem akan mengolah kumpulan citra yang dimasukkan pengguna dan menarik kesimpulan dari pengolahan tersebut. Seperti yang telah dibahas pada bagian sebelumnya, proses diagnosis dimulai dengan melakukan *pre-processing* terhadap citra yang dimasukkan ke dalam sistem. Pengolahan selanjutnya dilakukan dengan memisahkan pewarnaan pada citra melalui algoritma *colour deconvolution*. Setelah tahapan ini, salah satu citra kepekatan akan diolah kembali untuk membentuk citra baru berbentuk citra binari yang kemudian akan dilakukan pengenalan objek. Ketika objek telah dikenali, sistem akan memilah objek-objek mana saja yang akan ditandai sebagai bakteri dan menghitung jumlahnya. Berikut ini merupakan gambaran dari proses yang terjadi melalui potongan kode yang berasal dari sistem.

```
B,G,R = cv2.split(img)

histB,binsB = np.histogram(B.flatten(),256,[0,256])
cdfB = histB.cumsum()
histG,binsG = np.histogram(G.flatten(),256,[0,256])
cdfG = histG.cumsum()
histR,binsR = np.histogram(R.flatten(),256,[0,256])
cdfR = histR.cumsum()

cdf_mB = np.ma.masked_equal(cdfB,0)
cdf_mB = (cdf_mB - cdf_mB.min())*105/(cdf_mB.max()-cdf_mB.min()) + 150
```

```

cdfb = np.ma.filled(cdf_mB,0).astype('uint8')

cdf_mG = np.ma.masked_equal(cdfG,0)
cdf_mG = (cdf_mG - cdf_mG.min())*150/(cdf_mG.max()-cdf_mG.min())
cdfG = np.ma.filled(cdf_mG,0).astype('uint8')

cdf_mR = np.ma.masked_equal(cdfR,0)
cdf_mR = (cdf_mR - cdf_mR.min())*50/(cdf_mR.max()-cdf_mR.min())
cdfR = np.ma.filled(cdf_mR,0).astype('uint8')

newB = cdfB[B]
newG = cdfG[G]
newR = cdfR[R]

imgNoB = np.dstack((newB, newG, newR)).astype(np.uint8)

```

Gambar 4.9 Potongan kode *pre-processing*

```

w = htk.preprocessing.color_deconvolution.rgb_separate_stains_macenko_pca(dst,
255)
detected = htk.preprocessing.color_deconvolution.color_deconvolution(dst, w)

```

Gambar 4.10 Potongan kode algoritma *colour deconvolution*

Pada potongan kode mengenai penggunaan *colour deconvolution* di atas, perlu dijabarkan terlebih dahulu bahwa nilai  $w$  merupakan bobot yang akan digunakan pada algoritma tersebut. Seperti yang telah dijelaskan pada pembahasan sebelumnya, nilai  $w$  adalah variabel berupa matriks yang akan merepresentasikan dua jenis pewarnaan yang digunakan serta nilai *cross product* dari kedua matriks pewarnaan sebelumnya. Sementara itu, variabel  $dst$  merupakan citra yang sebelumnya telah diolah dan dilakukan denoising berwarna. Selanjutnya, proses akan berlanjut dengan melakukan konversi citra ke dalam bentuk binari dan akan dilanjutkan oleh potongan kode di bawah.

```

marker_count, markers = cv2.connectedComponents(segmented)
output2 = img.copy()
results = np.zeros((markers.shape), np.uint8)
cnt = np.zeros((markers.shape), np.uint8)
cnt [markers != 0] = 1
contours,hierarchy = cv2.findContours(cnt, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_
SIMPLE)
jml = 0
for contour in contours:
    area = cv2.contourArea(contour)
    perimeter = cv2.arcLength(contour, True)
    mask = np.zeros(segmented.shape,np.uint8)
    cv2.drawContours(mask, [contour], 0, 255, -1)
    pixelpoints = np.transpose(np.nonzero(mask))
    maxArea = 0
    if perimeter != 0:
        shape_factor = (4*math.pi*area)/(perimeter**2)
    else :
        shape_factor = 0

    rect = cv2.minAreaRect(contour)
    box = cv2.boxPoints(rect)

```

```

a, b, c, d = np.int0(box)
e = math.sqrt(
    (a[0] - b[0])**2
    + (a[1] - b[1])**2
)
f = math.sqrt(
    (c[0] - b[0])**2
    + (c[1] - b[1])**2
)
if e < f:
    g = float(e/f)
elif e > f:
    pass
    g = float(f/e)
else:
    g = 0
if g < 0.5 and g > 0 and shape_factor < 0.4 and shape_factor > 0 and area
> 50 and area <500:
    maxArea = area

x,y,w,h = cv2.boundingRect(contour)
cv2.rectangle(output2, (x,y), (x+w,y+h), (0,0,0), 3)
jml += 1

```

Gambar 4.11 Potongan kode pemilihan objek

Pada Gambar 4.11 di atas, sistem akan melakukan pengenalan serta pemilihan objek yang disangkakan bakteri. Pada tahap ini, digunakan citra binari yang didapat setelah melalui proses-proses sebelumnya. Sistem akan menghitung Area serta nilai *shape factor* dari objek yang ditemukan. Kedua variabel ini yang nantinya akan menentukan objek yang ditemukan tersebut termasuk ke dalam bakteri atau bukan. Nilai kondisional kedua variabel ini yang nantinya akan banyak mempengaruhi hasil dari pengenalan bakteri oleh sistem. Pada satu kondisi tertentu, sistem akan mengenali objek yang bukan bakteri sebagai bakteri, sehingga menyebabkan tingginya nilai *false positive* yang didapat. Sementara pada kondisi lain, objek selain bakteri yang dikenali sebagai bakteri akan jauh berkurang. Tetapi di sisi lain, banyak bakteri yang gagal terdeteksi oleh sistem dan menyebabkan naiknya nilai *false negative*. Sehingga diperlukan nilai yang tepat untuk membentuk rentang yang akan digunakan pada kedua variabel tersebut.

Setelah tiap citra yang masuk diolah dan dihitung, sistem akan merekap seluruh hasil yang didapat selama proses pengolahan tersebut. Nilai jumlah bakteri pada tiap-tiap citra akan dihitung dan ditentukan kondisinya. Nantinya, terdapat beberapa kondisi yang perlu dipenuhi untuk menarik kesimpulan tertentu. Sebagaimana yang telah dijelaskan pada bab perancangan, sistem akan melakukan pemodelan kasus untuk mengurangi beban kerja komputasi selama proses pengerjaan diagnosis. Melalui jumlah total bakteri dan kondisi-kondisi inilah nantinya sistem akan menentukan diagnosis terhadap sampel yang dimasukkan. Berikut ini merupakan potongan kode dari proses pengambilan kesimpulan ini.

```

for f in files:
    .
    .
    .
    p, jml = Kerjakan(url)
    if jml > 9:
        flag_kes3 += 1
    elif jml > 0:
        flag_kes2 += 1
    .
    .
    .
endfor
if flag_kes3 > 0:
    kesimpulan = "TBC Tingkat 3, pada salah satu gambar (Gambar " + str(i+1) +
") ditemukan lebih dari 10 bakteri"
elif flag_kes2 > 2:
    kesimpulan = "TBC Tingkat 2, karena ditemukan sejumlah bakteri sebanyak "
+str(total) + " bakteri."
elif total > 1:
    kesimpulan = "TBC Tingkat 1, karena ditemukan sedikit bakteri sebanyak "+st
r(total)+" bakteri."
elif total > 0:
    kesimpulan = "TBC scanti, karena ditemukan sangat sedikit bakteri sebanyak
"+str(total)+" bakteri."
elif total == 0:
    kesimpulan = "TBC Negatif, karena tidak ditemukan bakteri"

```

Gambar 4.12 Potongan kode pengambilan kesimpulan

#### 4.1.6 Halaman Hasil

Setelah sistem menyelesaikan diagnosis yang dilakukan terhadap citra sampel yang telah dimasukkan oleh pengguna sebelumnya, sistem akan menampilkan halaman hasil kepada pengguna. Halaman ini akan memberikan informasi kepada pengguna mengenai hasil diagnosis, sehingga pengguna mampu mengamati terlebih dahulu hasil yang didapatkan. Berikut ini merupakan gambaran dari halaman hasil, seperti yang ditampilkan pada Gambar 4.9.

Gambar 4.13 Antarmuka Hasil

Gambar 4.9 menyediakan informasi mengenai hasil dari diagnosis yang baru saja dilakukan oleh sistem. Data yang ditampilkan bersifat statis, sehingga pengguna tidak dapat melakukan perubahan pada data yang ditampilkan pada halaman. Pengguna diberikan opsi untuk mengulang proses diagnosis dengan menekan tombol 'Ulang' atau menyimpan hasil diagnosis ke dalam basisdata dengan menekan tombol 'Simpan'. Selain itu, pengguna juga mampu melihat citra hasil diagnosis dengan menekan pada citra yang diinginkan yang ditampilkan pada halaman hasil. Nantinya sistem akan memberikan pop-up berupa gambar terkait yang telah dideteksi sebelumnya beserta objek mana saja yang telah ditandai sebagai MTB.

#### 4.2 Pengujian Sistem

Pengujian sistem dilakukan dengan menguji seberapa akurat sistem mampu mendeteksi bakteri *mycobacterium tuberculosis* pada citra yang dimasukkan. Pengujian akurasi dilakukan dengan membandingkan hasil dari sistem dengan ahli yang sebelumnya telah menandai objek bakteri pada citra terkait. Perhitungan akurasi dilakukan dengan metode *confussion matrix*

untuk menemukan nilai akurasi. Berikut ini merupakan tabel *confussion matrix* dari perbandingan sistem dengan ahli yang didapatkan.

Tabel 4.1 Tabel *Confussion Matrix*

No	TP	TN	FP	FN
1	16	0	0	13
2	3	1	0	1
3	4	0	5	2
4	9	3	0	6
5	9	0	0	1
6	1	0	0	1
7	3	1	0	2
8	2	0	0	2
9	2	0	1	1
10	1	0	0	1
11	5	0	0	1
12	5	4	0	1
13	4	0	2	2
14	18	0	1	9
15	2	0	0	9
16	20	0	0	9
17	16	0	2	10
18	3	0	0	2
19	7	1	1	2
20	6	3	0	1
21	17	0	3	12
22	9	0	0	3
23	3	0	0	1
24	5	0	0	5
25	24	1	5	11
26	11	22	12	4
27	16	0	0	4
28	0	2	1	1
29	2	8	4	6
30	13	2	3	8
31	7	0	2	6
32	1	0	0	0
33	1	0	0	0
34	2	3	1	1
35	6	5	0	4
36	25	7	13	9
37	23	2	0	12

38	37	11	1	19
39	6	6	0	8
40	1	0	0	1
41	6	0	0	2
42	5	0	0	2
Total	333	81	52	179

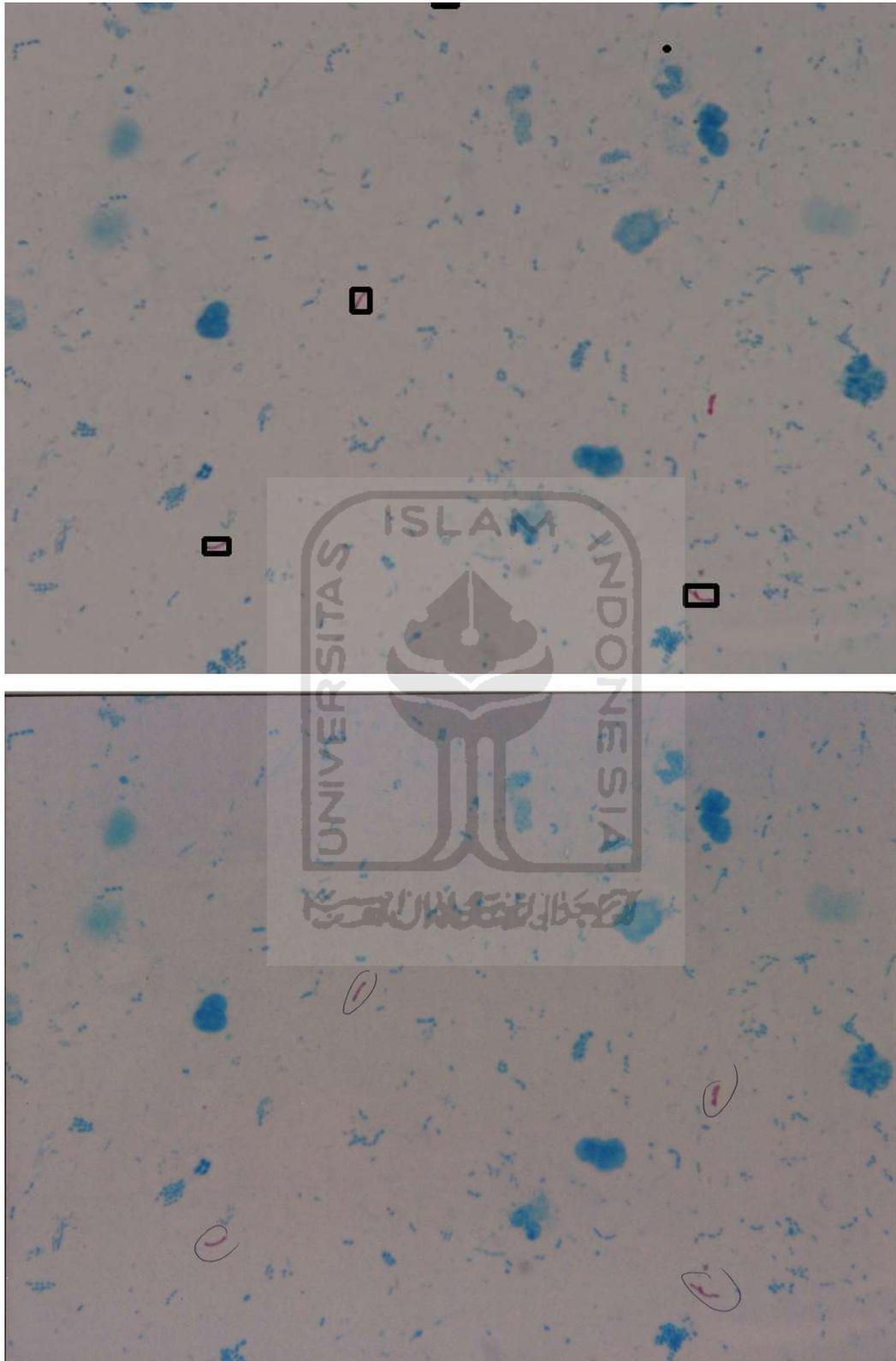
Melalui tabel 4.1 di atas, terlihat jika sistem masih memberikan kesalahan dalam mendeteksi objek bakteri pada citra yang diberikan. Terdapat 179 objek yang tidak terdeteksi oleh sistem, tetapi pada keadaan aktual objek tersebut merupakan bakteri tuberculosis yang dicari. Selain itu ada 52 objek yang seharusnya bukan merupakan objek bakteri, yang salah terdeteksi oleh sistem. Ada pula objek yang berhasil terdeteksi benar oleh sistem sebanyak 333 objek *True Positive* dan 81 objek *True Negative*. Nilai ini didapatkan dengan menggunakan beberapa citra yang dimasukkan ke dalam sistem.

Dari tabel di atas juga didapatkan nilai *Precision* sebesar 86.49% dan nilai *Recall* sebesar 65.04%. Nilai ini didapat dengan menghitung presentase benar dari semua objek yang disangkakan benar oleh sistem sebagai nilai *Precision*. Sementara untuk nilai *Recall* diperoleh dengan menghitung presentase seberapa banyak objek yang berhasil ditemukan oleh sistem dari keseluruhan objek aktual. Dari kedua nilai tadi, ditemukan nilai F1 dengan menggunakan rumus 4.1 sebesar 74.25%.

$$F1 \text{ score} = 2 \cdot \frac{(\text{Recall} \cdot \text{Precision})}{(\text{Recall} + \text{Precision})} \quad (4.1)$$

Sementara itu, penentuan tingkat keparahan juga akan terpengaruh oleh beberapa kegagalan yang dilakukan ketika melakukan deteksi objek. Terdapat beberapa citra yang terdeteksi tidak terbaca dengan cukup baik, sehingga mengakibatkan objek yang ditemukan tidak sesuai dan menyebabkan kesalahan penentuan tingkat keparahan. Tentunya hal ini akan berpengaruh terhadap diagnosis yang dilakukan oleh sistem.

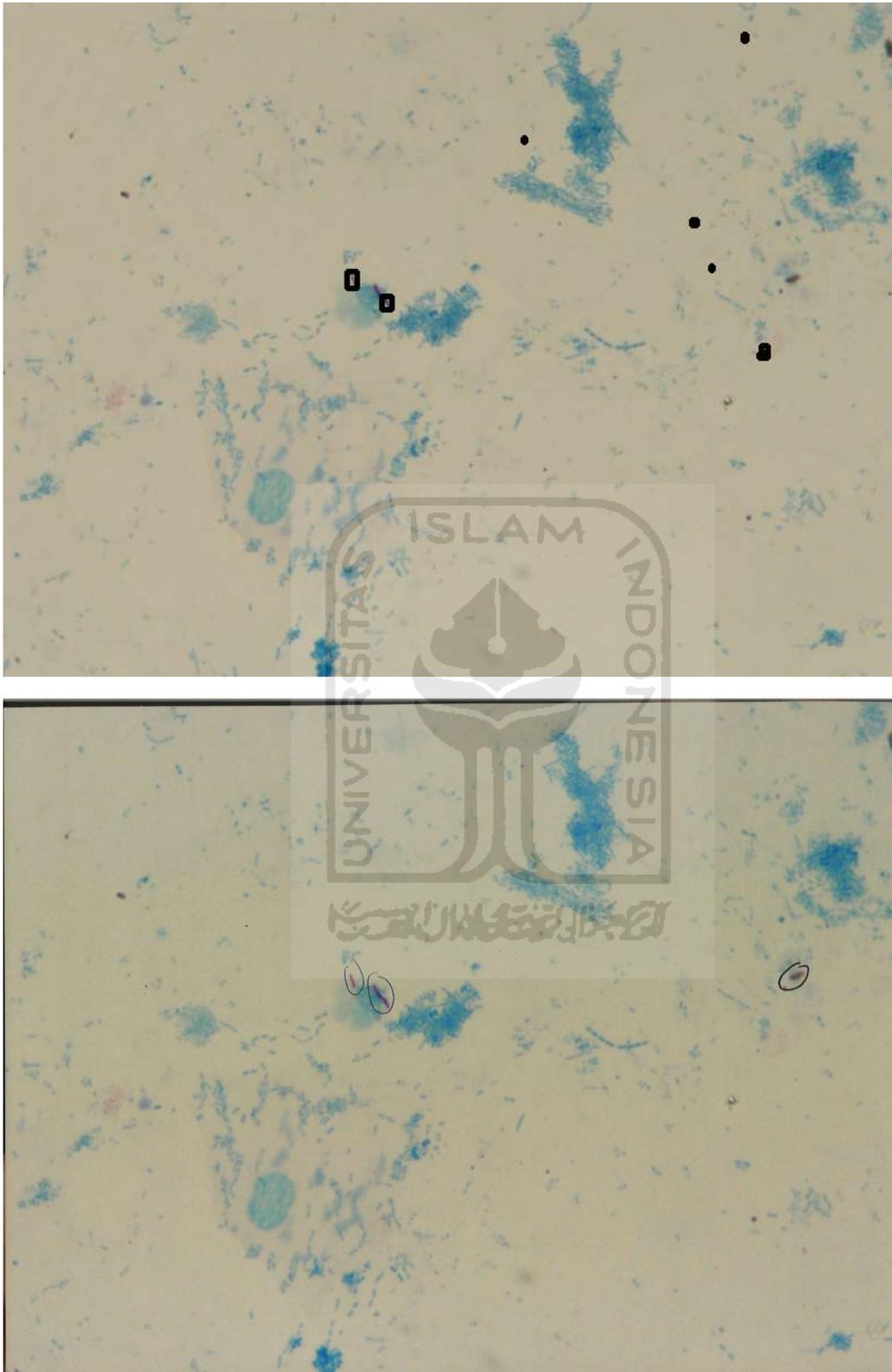
Berikut ini merupakan beberapa contoh gambar deteksi yang dilakukan oleh sistem, yang dibandingkan dengan gambar yang dideteksi oleh ahli :



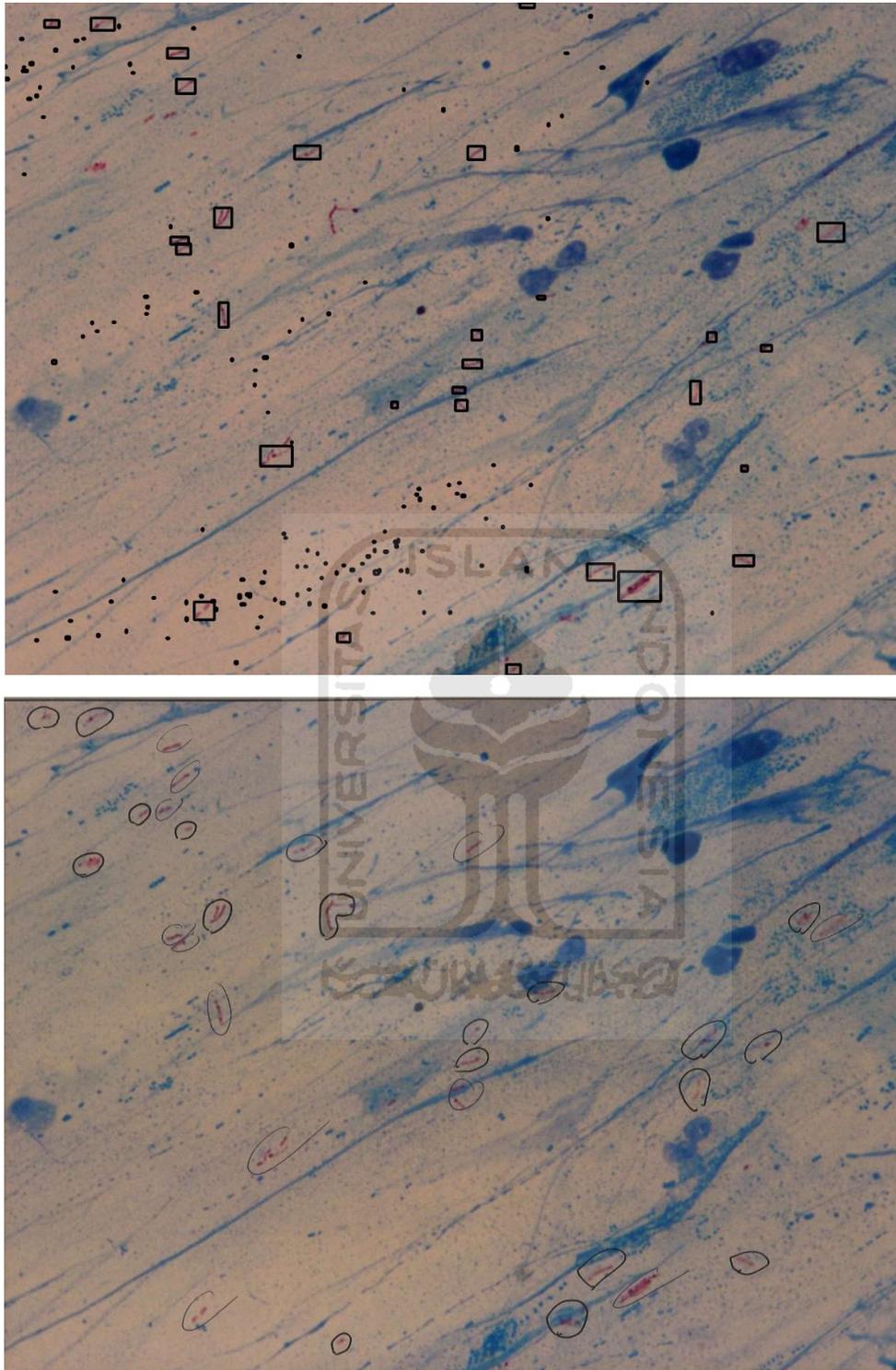
Gambar 4.14 Perbandingan gambar sistem (atas) dengan ahli (bawah) 1



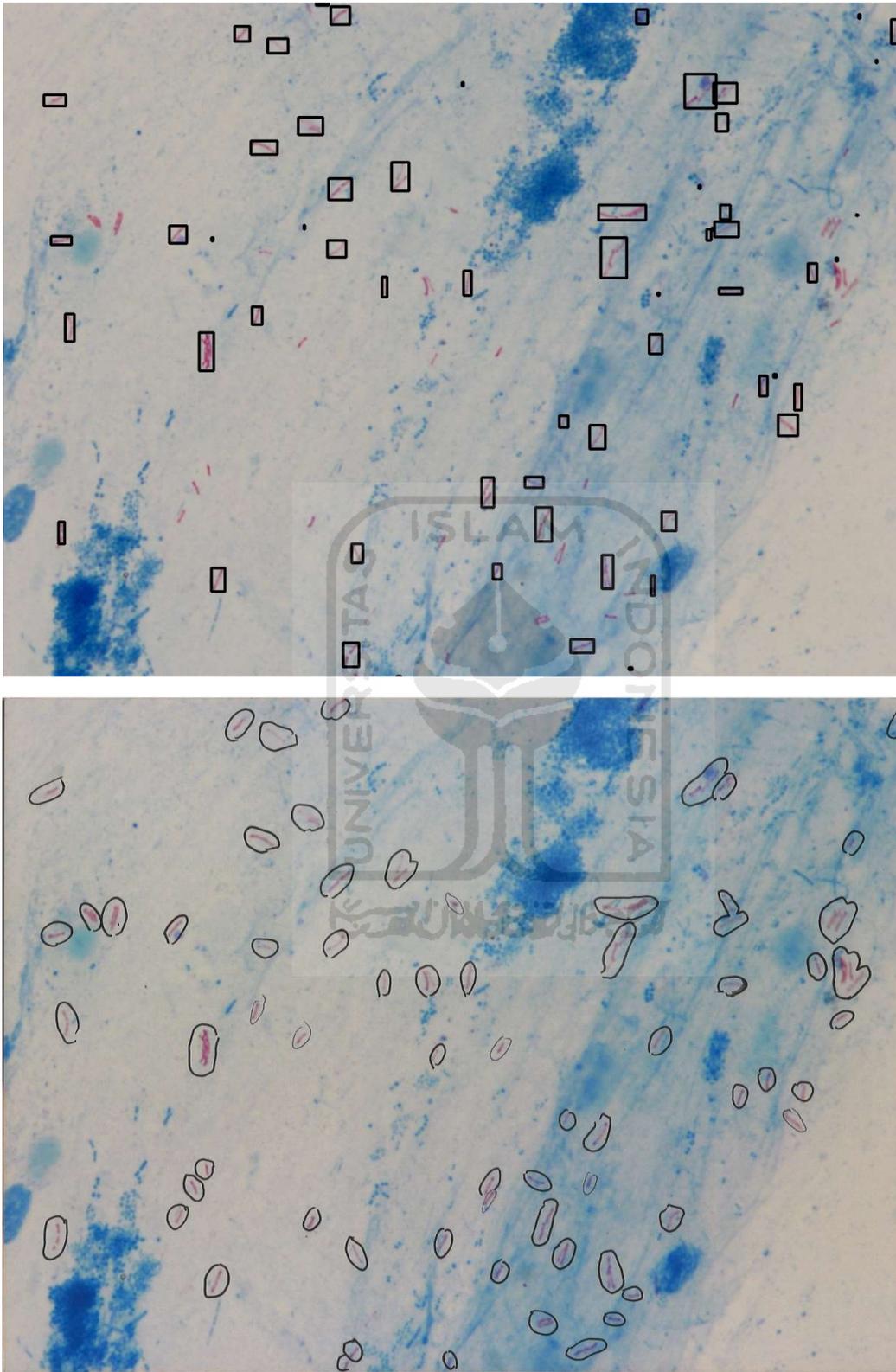
Gambar 4.15 Perbandingan gambar sistem (atas) dengan ahli (bawah) 2



Gambar 4.16 Perbandingan gambar sistem (atas) dengan ahli (bawah) 3



Gambar 4.17 Perbandingan gambar sistem (atas) dengan ahli (bawah) 4



Gambar 4.18 Perbandingan gambar sistem (atas) dengan ahli (bawah) 5

Selain itu, sistem memerlukan waktu kerja yang cukup tinggi, bergantung dari seberapa banyak objek yang ditemukan pada masing-masing kasus. Dari sekitar 50 citra yang dilakukan pendeteksian, sistem memerlukan waktu berkisar antara 2300 detik hingga 3100 detik dalam menyelesaikan keseluruhan tugasnya. Terdapat beberapa faktor yang mungkin mengakibatkan hal ini. Faktor pertama adalah karena keterbatasan sumber daya komputasi yang digunakan untuk mengolah citra sputum dan yang kedua adalah karena resolusi yang digunakan cukup besar dan pada pengolahan ini citra tidak dilakukan *resizing*. Tentunya ini akan berakibat pada seberapa banyak citra yang mampu didiagnosis per satuan waktu tertentu.

Biarpun begitu, salah satu keunggulan dari penggunaan komputer dalam melakukan suatu pekerjaan adalah kemampuan komputer untuk bekerja secara terus-menerus yang tentunya akan meningkatkan jumlah diagnosis yang bisa dikerjakan dalam satu hari. Selain itu, pada kondisi tertentu bisa saja sistem digunakan tanpa keberadaan ahli secara langsung di lokasi dengan tingkat ketepatan yang kurang lebih sama antara satu tempat dengan tempat lainnya. Tentunya, kedua hal ini akan berpengaruh terhadap kecepatan dan ketepatan diagnosis yang dilakukan sehingga mampu menentukan sikap yang sesuai dengan lebih cepat.

Selama proses penelitian, terdapat beberapa hambatan yang ditemui yang sedikit banyak mengganggu berjalannya penelitian. Beberapa di antaranya adalah keterbatasan kamera mikroskop yang menghasilkan kualitas gambar yang kurang baik, sehingga diperlukan kamera lainnya untuk digunakan. Keterbatasan pengetahuan dan pengalaman dalam mengoperasikan mikroskop. Sempat terjadi insiden di mana laptop yang digunakan untuk melakukan penelitian terjatuh dan menyebabkan sebagian besar data terkait penelitian hilang dikarenakan rusaknya HDD pada laptop. Selain itu, keterbatasan sumber daya komputasi pada laptop yang digunakan juga menyebabkan waktu pengolahan data yang cukup tinggi.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Melalui hasil pengujian serta pembahasan yang telah dijabarkan sebelumnya, maka diambil kesimpulan sebagai berikut:

- a. Sistem pendeteksi bakteri tuberkulosis berlaku terlalu agresif dalam menentukan sebuah objek yang disangka merupakan objek bakteri pada citra yang diberikan. Hal ini menyebabkan beberapa objek yang seharusnya bukan bakteri terdeteksi sebagai bakteri oleh sistem dan meningkatkan angka *false positive*. Sementara pada kesempatan lainnya, dengan mengubah variabel yang digunakan dalam menentukan objek sebagai bakteri. Sistem akan melakukan pendeteksian yang menyebabkan beberapa objek yang seharusnya disebut sebagai bakteri akan diabaikan dan mengakibatkan nilai *false negative* yang tinggi.
- b. Dikarenakan terjadinya beberapa kesalahan dalam melakukan pendeteksian bakteri, sistem juga pada akhirnya melakukan kesalahan dalam menentukan tingkat keparahan dari penyakit yang diderita. Beberapa campuran kasus mengakibatkan sistem mendiagnosis keparahan satu tingkat lebih tinggi dari seharusnya atau sebaliknya ketika menggunakan set variabel yang bertindak lebih pasif.
- c. Waktu pengerjaan diagnosis yang cenderung lama menyebabkan sistem memiliki waktu tunggu yang lama pula sebelum menentukan hasil diagnosis. Walaupun begitu, hal ini menjadi keterbatasan dari kinerja komputasi yang dimiliki oleh sistem itu sendiri.

#### **5.2 Saran**

Selama proses pembangunan sistem, terdapat beberapa kendala dan kekurangan yang dihadapi. Demi pengembangan ke depannya penelitian yang akan datang diharapkan untuk meningkatkan efisiensi sumber daya komputasi dalam melakukan diagnosis untuk mengurangi beban kerja selama proses diagnosis. Selain itu, kemampuan untuk menerima kualitas citra mikroskop yang berbeda juga diperlukan untuk mendukung kasus-kasus lainnya.

## DAFTAR PUSTAKA

- Alsubaie, N., Trahearn, N., Raza, S. A., Snead, D., & Rajpoot, N. (2017). Stain Deconvolution Using Statistical Analysis of Multi-Resolution Stain Colour Representation. *PLoS ONE* 12(1): e0169875. <https://doi.org/10.1371/journal.pone.0169875>.
- Fakultas Kedokteran Universitas Hasanuddin. (2017). *Buku Panduan Pemeriksaan Sputum BTA*. Makassar.
- Kementrian Kesehatan Republik Indonesia. (2018, 8 20). *Infodatin Tuberkulosis 2018 - Dicari Para Pemimpin untuk Dunia Bebas TBC*. Diambil kembali dari ementrian Kesehatan Republik Indonesia: <https://www.kemkes.go.id/article/view/18082000004/infodatin-tuberkulosis-2018---dicari-para-pemimpin-untuk-dunia-bebas-tbc.html>
- Macenko, M., Niethammer, M., Marron, J., Borland, D., Woosley, J., Guan, X., . . . Thomas, N. (2009). A method for normalizing histology slides for quantitative analysis. *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 1107-1110.
- Reinhard, E., Adhikhmin, M., Gooch, B., & Shirley, P. (2001). Color transfer between images. *IEEE Computer Graphics and Applications*, 34-41.
- Suetens, P. (2009). Introduction to digital image processing. Dalam P. Suetens, *Fundamentals of Medical Imaging* (hal. 1-13). Cambridge: Cambridge University Press.
- Thomas, D. (2012, March 4). *How do histologists diagnose tuberculosis?* Diambil kembali dari Histology Blog: <https://histologyblog.com/2012/03/04/how-do-histologists-diagnose-tb/>
- Todar, K. (2006). *Mycobacterium tuberculosis and Tuberculosis*. Diambil kembali dari Todar's Online Textbook of Bacteriology: <http://textbookofbacteriology.net/tuberculosis.html>
- University of Tartu. (2014). *1. Introduction to image processing*. Diambil kembali dari University of Tartu: <https://sisu.ut.ee/imageprocessing/book/1>
- Vu, Q. D., Graham, S., Kurc, T., To, M. N., Shaban, M., Qaiser, T., . . . Farahani, K. (2019). Methods for Segmentation and Classification of Digital Microscopy Tissue Images. *Enabling Computational Pathology: Overcoming Translational Barriers*.
- WHO. (2018). *Top 10 Causes of Death*. Diambil kembali dari WHO: <https://www.who.int/data/gho/data/themes/topics/causes-of-death>

**LAMPIRAN**

