

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Penyakit Tidak Menular (PTM) adalah penyebab kematian terbanyak di Indonesia. Kematian akibat PTM tidak hanya terjadi di perkotaan melainkan juga pedesaan. Diabetes *Gestasional* dalam kehamilan merupakan penyebab tidak langsung dalam kematian yang disebabkan oleh penyakit yang telah diderita ibu hamil. Penyebab langsung adalah kematian yang disebabkan oleh komplikasi obstetri dalam periode kehamilan, persalinan maupun nifas, akibat penanganan, kelalaian atau pengobatan yang tidak tepat. Terdapat beberapa tipe *diabetes mellitus* yaitu tipe 1 (bergantung insulin), tipe 2 (tidak bergantung insulin) dan *Diabetes Mellitus Gestasional*[1].

World Health Organization(WHO) (2019) menyatakan tipe diabetes yang paling sering terjadi adalah *Diabetes Mellitus* tipe 2 dan kejadiannya meningkat secara drastis di negara dengan pendapatan rendah. Badan Kesehatan Dunia (WHO) memprediksi adanya peningkatan jumlah penyandang *Diabetes Mellitus* tipe 2 yang menjadi salah satu ancaman kesehatan global. Meningkatnya prevalensi *diabetes mellitus* di wilayah Indonesia tentu saja harus dicegah. Salah satu cara mencegahnya adalah dengan mengetahui faktor-faktor yang mempengaruhi terjadinya penyakit *diabetes mellitus* di masyarakat[2].

Angka kejadian *Diabetes Mellitus (DM)* di dunia dari tahun ke tahun terus meningkat, data terakhir dari *Who Health Organization (WHO)* menunjukkan pada tahun 2000 sebanyak 150 juta penduduk dunia menderita *DM* dan angka ini akan

menjadi dua kali lipat pada tahun 2025. Peningkatan angka penderita penyakit ini akan menjadi di negara berkembang karena pertumbuhan populasi, punaan, diet yang tidak sehat, obesitas dan kurang aktivitas fisik. Di Indonesia, menurut data Riset Kesehatan Dasar 2013 didapatkan proporsi kejadian DM sebesar 6,9% pada penduduk usia ≥ 15 tahun. *Diabetes Mellitus Gestasional (DMG)* adalah salah satu jenis *diabetes mellitus* yang terjadi pada ibu hamil. WHO (2013) mendefinisikan *DMG* sebagai derajat apapun intoleransi glukosa dengan omset atau pengakuan pertama selama kehamilan. Kehamilan sendiri merupakan stres bagi metabolisme karbohidrat ibu. Pada kehamilan terjadi peningkatan produksi hormon-hormon antagonis insulin, antara lain: progesteron, estrogen, human placenta lactogen dan kortisol. Peningkatan hormon-hormon tersebut menyebabkan terjadinya resistensi insulin dan peningkatan kadar glukosa darah.[3]

DMG menyebabkan komplikasi yang signifikan dan berpotensi bagi ibu dan janin termasuk preeklamsia, eklamsi, polihidramnion, makrosomia janin, trauma kelahiran, kelahiran operatif, komplikasi metabolik dan kematian perinatal. *DMG* meningkatkan morbiditas neonatus, yaitu hipoglikemia, ikterus, polisitemia dan makrosomia. Hal ini terjadi karena bayi dari ibu *DMG* mensekresi insulin lebih besar sehingga merangsang pertumbuhan bayi makrosomia. *DMG* juga dapat meningkatkan risiko bagi ibu 3-5% untuk menjadi diabetes mellitus di masa mendatang. Wanita dengan *DMG* hampir tidak pernah memberikan keluhan, sehingga perlu dilakukan skrining. Deteksi dini sangat diperlukan untuk menjaring *diabetes mellitus gestasional* agar dapat dikelola sebaik-baiknya

terutama dilakukan pada ibu dengan faktor resiko. Dengan adanya deteksi dini pada ibu hamil juga dapat membantu untuk meningkatkan kesejahteraan ibu baik selama kehamilan ataupun sesudah kehamilan[3].

Penyebab *DMG* juga belum diketahui secara pasti, namun faktor yang sering memicu adalah perubahan hormon. Saat hamil, plasenta akan memproduksi hormon tambahan seperti hormon estrogen, human *placentalactogen* (HPL), dan hormon yang meningkatkan resistensi insulin. Seiring berjalannya waktu, hormon-hormon tersebut akan meningkat dan mempengaruhi kinerja insulin. Semakin tinggi pengaruh hormon terhadap insulin, kadar gula dalam darah pun akan meningkat dan hal ini meningkatkan risiko terkena *diabetes gestasional*. Selain itu, seorang wanita juga berisiko terkena penyakit jika sudah memasuki usia 25 tahun ke atas saat hamil, memiliki tekanan darah tinggi (hipertensi), memiliki keluarga dengan sejarah diabetes, BMI di atas 25, pernah melahirkan bayi diatas 4.5 kg, pernah keguguran, pernah mengalami *diabetes gestasional* sebelumnya[1].

PUSTU Desa Pasir Intan adalah salah satu Puskesmas Pembantu yang ada di Desa Pasir Intan atau DK2 SKPC Kecamatan Bangun Purba, Kabupaten Rokan Hulu. Hasil dari observasi banyak ibu hamil yang belum mengetahui penyakit yang mudah menyerang saat masa kehamilan, sehingga ibu hamil di wilayah tersebut kurang menjaga kebugaran tubuhnya seperti dengan berolahraga dan mengatur pola makan. Selain menjaga kesehatannya ibu hamil juga harus menjaga kesehatan pada calon bayinya. Penyakit yang sering dialami oleh ibu hamil adalah penyakit *Diabetes Gestasional*. Wilayah tersebut ada beberapa kriteria ibu hamil

yang mengalami penyakit diabetes (gula darah), seperti Memiliki berat badan berlebih, memiliki riwayat tekanan darah tinggi, pernah mengalami *diabetes gestasional* pada kehamilan sebelumnya, pernah mengalami kematian bayi dalam kandungan. Berkaitan dengan hal tersebut, diperlukan suatu upaya bersama untuk memprediksi secara dini penyakit gula darah pada ibu hamil yang menginfeksi. Mengobati secara tepat untuk menanggulangi penyakit ini.

Dalam menanggapi kejadian penyakit *diabetes gestasional* pada ibu hamil tersebut, mendorong penulis untuk mencoba menerapkan metode *Learning Vector Quantization (LVQ)* untuk mendeteksi penyakit *diabetes gestasional* berdasarkan gejala yang muncul pada ibu hamil. Metode *Learning Vector Quantization (LVQ)* adalah metode klasifikasi pada pola untuk setiap unit yang ada, dan *output*-nya akan mempresentasikan kategori tertentu atau kelas yang telah ditentukan sebelumnya. Kelas-kelas yang diperoleh akan bergantung pada jarak antar vektor *input LVQ* juga merupakan metode pelatihan yang dilakukan dalam *supervised learning* (pembelajaran yang terawasi) dan termasuk dalam kategori *single layer*, artinya hanya memiliki *layer input* dan *layer output* yang saling terhubung dengan adanya suatu bobot[4].

Pada penelitian sebelumnya, Risti Vika Arvianti yang berjudul “Klasifikasi Penyakit Diabetes *Mellitus* Tipe 2 Menggunakan *Learning Vector Quantization (LVQ)*“. Pada penelitian ini proses klasifikasi penyakit *diabetes mellitus* dengan menggunakan 115 data diabetes mellitus dapat dihitung secara manual dan dapat dihitung secara matlab. Berdasarkan hasil pengujian parameter diperoleh nilai akurasi sebesar 91.30%. Proses normalisasi data menggunakan

metode *Min-Max* langkah awal yang dilakukan adalah menentukan nilai *Minimal* dan *Maksimal* setiap variabel. Proses klasifikasi pada penyakit *diabetes mellitus* tipe 2 dilakukan dengan menginputkan 5 data dan diperoleh 10 *layer*. Selanjutnya menghitung bobot tiap *layer* tersebut dengan rumus jarak *Euclidean*, nilai jarak *Euclidean* terkecil akan dipilih menjadi *output*. Hasil klasifikasi menggunakan metode *Learning Vector Quantization* dengan data testing sebanyak 40% dilakukan percobaan menggunakan epoch 1000 hingga 10000 dengan selang 1000. Model terbaik nilai minimum terdapat pada epoch 1000 dengan *learning rate* 0.008, Model terbaik dengan nilai *minimum* pada epoch 3000 dengan *learning rate* 0.004, Model terbaik dengan nilai *minimum* terdapat pada epoch 9000 pada *learning rate* 0.008. Pada data Training 70% dan 80% dengan epoch 1000 sampai dengan 10.000 diperoleh model dengan nilai *minimum* yang terdapat pada *learning rate* 0.001 sampai dengan 0.009[5].

Selanjutnya pada penelitian Rianti Nurpalah, dkk yang berjudul “Deteksi Dini *Diabetes Melitus Gestasional (DMG)*” Melalui pemeriksaan Glukosa Darah Sebagai Upaya Pencegahan Komplikasi Pada Ibu Hamil. Berdasarkan hasil pengujian yang telah dilakukan yaitu pemeriksaan glukosa darah pada ibu hamil sebagai upaya deteksi dini *DMG* di Wilayah Kelurahan Karikil yang diikuti oleh 25 orang ibu hamil sebagai responden, didapatkan hasil bahwa seluruh responden 100% memiliki kadar glukosa darah normal. Ibu hamil yang memiliki kadar glukosa darah normal diharapkan tetap menjalankan menjaga pola makan dan pola hidup yang sehat supaya kesehatannya bisa tetap terjaga serta bayi yang dilahirkan memiliki kesehatan yang optimal[6].

Berdasarkan permasalahan di atas, maka penulis merancang sebuah aplikasi yang digunakan untuk membantu mendeteksi penyakit gula darah (*Gestasional*) pada ibu hamil yang akan direlasikan dalam tugas skripsi dengan judul **“Deteksi Penyakit Gula Darah pada Ibu Hamil menggunakan metode *Learning Vector Quantization (LVQ)*”**.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, yang menjadi pokok permasalahan dalam hal ini adalah bagaimana menerapkan metode *Learning Vector Quantization (LVQ)* untuk mendeteksi penyakit *diabetes gestasional* pada ibu hamil?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk menerapkan metode *Learning Vector Quantization (LVQ)* dalam mengenali pola penyakit *diabetes gestasional* pada ibu hamil.

1.4 Batasan Masalah

Dalam penelitian ini, Batasan masalah dari penelitian adalah sebagai berikut :

1. Metode yang digunakan untuk mengenali penyakit Diabetes Gestasional adalah *Learning Vector Quantization (LVQ)*.
2. Variabel yang digunakan sebanyak 13 gejala
3. Stusi kasus PUSTU Desa Pasir Intan, Kecamatan Bangun Purba, Kabupaten Rokan Hulu.

4. Aplikasi dirancang dengan menggunakan bahasa pemrograman *PHP* dan *MySQL*.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sistem dapat memberikan hasil yang baik dalam pengenalan penyakit *diabetes gestasional* pada ibu hamil dan dapat dijadikan sebagai media pembelajaran, Serta dapat menjadi referensi untuk penelitian selanjutnya.

1.6 Metodologi Penelitian

Tahapan-tahapan yang akan dilakukan pada pelaksanaan penelitian adalah sebagai berikut :

1. Studi Literatur

Pada tahap ini penulis mengumpulkan bahan referensi berkaitan dengan gejala penyakit terkena penyakit *diabetes gestasional*. *Learning Vector Quantization (LVQ)*, dari berbagai jurnal, skripsi, buku, artikel dan berbagai sumber referensi lainnya.

2. Analisis Masalah

Pada tahap ini dilakukan analisis untuk setiap informasi yang telah di peroleh dari tahap sebelumnya agar mendapatkan pemahaman akan masalah dan metode yang akan digunakan untuk menyelesaikan permasalahan.

3. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem sesuai dengan hasil dari tahap sebelumnya.

4. Implementasi

Pada tahap ini hasil dari analisis dan perancangan sistem akan di implementasikan ke dalam kode program.

5. Pengujian

Pada tahap ini dilakukan pengujian terhadap pengenalan penyakit *diabetes gestasional* pada ibu hamil untuk memastikan bahwa proses akurasi dapat memberikan hasil yang baik.

6. Dokumentasi dan Penyusunan Laporan

Pada tahap terakhir membuat dokumentasi dan menyusun laporan hasil dari analisi dan implementasi dari penelitian tersebut.

1.7 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari lima bagian utama sebagai berikut :

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini berisi teori-teori yang digunakan pada penelitian ini. Teori-teori yang berhubungan dengan pengenalan gejala *diabetes gestasional* pada ibu hamil, menggunakan metode *Learning Vector Quantization (LVQ)*.

BAB 3 METODOLOGI PENELITIAN

Bab ini berisi tahapan-tahapan dalam pengumpulan data, perancangan sistem, perumusan masalah dan analisa.

BAB 4 ANALISA DAN PERANCANGAN

Bab ini berisi analisa dan perancangan aplikasi penerapan metode *Learning Vector Quantization (LVQ)* dalam pengenalan penyakit *diabetes gestasional* pada ibu hamil serta pemodelan sistem.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi implementasi dari analisis, perancangan dan pengujian pada aplikasi yang berhasil dibangun.

BAB 6 PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian yang telah dilakukan dan saran-saran untuk pengembangan aplikasi atau penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 Kecerdasan Buatan

Kecerdasan buatan berasal dari bahasa Inggris “*Artificial Intelligence*” atau singkatan *AI*, yaitu *intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. Kecerdasan Buatan *AI (Artificial Intelligence)* merupakan salah satu bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik daripada yang dilakukan manusia[7].

Salah satu bagian dari sistem kecerdasan buatan adalah sistem pakar yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut. Sistem pakar dapat pula diartikan sebagai sebuah program komputer yang didesain untuk menggantikan seorang pakar di bidang tertentu. Kecerdasan buatan dapat membantu manusia dalam membuat keputusan, mencari informasi secara lebih akurat, atau membuat komputer lebih mudah digunakan dengan tampilan yang menggunakan bahasa natural sehingga mudah dipahami[8].

2.2 Jaringan Syaraf Tiruan (JST)

Jaringan syaraf tiruan (JST) atau umumnya hanya disebut *neural network* (NN), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan sistem syaraf manusia. JST merupakan sistem adaptif yang dapat mengubah strukturnya untuk memecahkan masalah berdasarkan informasi *eksternal* maupun *internal* yang mengalir melalui jaringan tersebut. Oleh karena sifatnya yang adaptif, JST juga sering disebut dengan jaringan adaptif. Secara sederhana, JST adalah sebuah alat pemodelan data statistik *non-linier*. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara *input* dan *output* untuk menemukan pola-pola pada data.

Saat ini bidang kecerdasan buatan dalam usahanya menirukan intelegensi manusia, belum mengadakan pendekatan dalam bentuk fisiknya melainkan dari sisi yang lain. Pertama-tama diadakan studi mengenai teori dasar mekanisme proses terjadinya intelegensi. Bidang ini disebut *Cognitive Science*. Dari teori dasar ini dibuatlah suatu model untuk disimulasikan pada komputer, dan dalam perkembangannya yang lebih lanjut dikenal berbagai sistem kecerdasan buatan yang salah satunya adalah jaringan syaraf tiruan. Dibandingkan dengan bidang ilmu yang lain, jaringan syaraf tiruan relatif masih baru[9].

Secara umum lapisan penyusun yang terdapat pada jaringan syaraf tiruan dapat dibedakan menjadi tiga, yaitu[10] :

1. Lapisan masukan (*input layer*)

Lapisan ini terletak di bagian terluar yang merupakan tempat unit-unit *input* menerima masukan dari luar. *Input-an* merupakan parameter-parameter

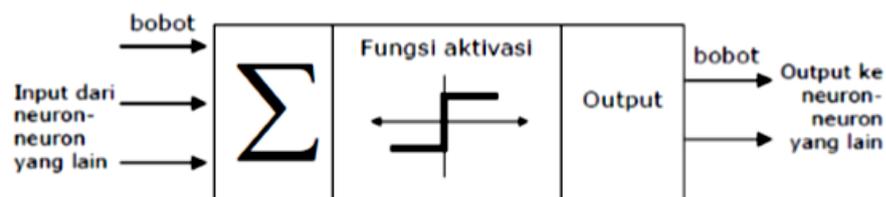
yang akan diolah untuk diperoleh *output*. Bagian ini terdiri dari unit-unit *input* yang langsung berhubungan dengan permasalahan yang ingin diselesaikan.

2. Lapisan tersembunyi (*hidden layer*)

Lapisan ini terletak di bagian tengah yang merupakan tempat pemrosesan *input* yang akan diselesaikan dengan jaringan saraf tiruan. *Neuron* pada lapisan ini disebut juga dengan *hidden unit*. Pada lapisan ini digunakan fungsi aktivasi yang nantinya akan mengolah *input*-an yang diterima. Terdapat beberapa fungsi aktivasi seperti; fungsi undak biner, fungsi bipolar, fungsi Gaussian dan lain-lain.

3. Lapisan keluaran (*output layer*)

Lapisan ini merupakan lapisan akhir dari jaringan saraf tiruan yang merupakan *output* dari permasalahan yang telah di-*input*-kan. *Neuron* pada lapisan ini disebut dengan *output unit* yang selanjutnya dikonversi terhadap penyelesaian masalah.



Gambar 1. 1 Struktur Neuron Jaringan Syaraf Tiruan

2.3 *Learning Vector Quantization (LVQ)*

Learning Vector Quantization (LVQ) adalah suatu metode yang digunakan untuk melakukan pelatihan terhadap lapisan-lapisan kompetitif yang terawasi. Lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vektor *input* yang diberikan. Vektor-vektor input akan dikelompokkan

dalam kelas yang sama apabila beberapa vektor *input* memiliki jarak yang sangat berdekatan. Diasumsikan bahwa serangkaian pola pelatihan dengan klasifikasi yang tersedia bersama dengan distribusi awal vektor referensi. Setelah pelatihan, kelas yang sama akan ditugaskan untuk melakukan klasifikasi vektor masukan sebagai unit keluaran, sedangkan yang mempunyai vektor referensi diklasifikasikan sebagai vektor masukan.

Learning Vector Quantization (LVQ) merupakan jaringan lapisan tunggal (*single-layer net*) di mana lapisan masukan terkoneksi secara langsung dengan setiap neuron pada keluaran. Koneksi antar neuron tersebut dihubungkan dengan bobot/*weight*. Bobot merupakan nilai matematis dari koneksi yang mentransfer data dari satu lapisan ke lapisan lainnya, yang berfungsi untuk mengatur jaringan sehingga dapat menghasilkan *output* yang diinginkan. Bobot pada *LVQ* sangat penting, karena dengan bobot ini *input* dapat melakukan pembelajaran dalam mengenali suatu pola. Vektor bobot berfungsi untuk menghubungkan setiap neuron pada lapisan *input* dengan masing-masing neuron pada lapisan *output*[11].

Langkah-langkah algoritma pelatihan *LVQ* terdiri atas[12] :

1. Inisialisasi bobot awal (W) dan parameter *LVQ*, yaitu *max Epoch*, α , *dec α* dan *min α* .
2. Masukkan data *input* (X) dan kelas target (T).
3. Tetapkan kondisi awal: *epoch* = 0
4. Kerjakan jika: (*epoch* < *max Epoch*) dan ($\alpha \geq \text{min } \alpha$).
 - a. *Epoch* = *epoch* + 1

- b. Tentukan J sedemikian hingga $\|X_i W_j\|$ minimal menggunakan perhitungan rumus jarak *euclidian*.

$$D(j) = \sum (W_{ij} - X_i)^2$$

- c. Perbaiki W_j dengan ketentuan: Jika $T = C_j$ maka

$$W_j(t+1) = W_j(t) + \alpha(t)[X(t) - W_j]$$

Jika $T \neq C_j$ maka

$$W_j(t+1) = W_j(t) + \alpha(t)[X(t) - W_j(t)]$$

- d. Kurangi nilai α dengan:

$$\alpha = \alpha - \alpha * Dec \alpha$$

5. Tes kondisi berhenti dengan *output* berupa bobot optimal

2.3.1 Konsep dari LVQ

Konsep dasar dari LVQ adalah *competitive mlearning neural networks*, dimana semua sel akan dilatih untuk membentuk lapisan *input*, dan semua sel tersebut juga akan mendapatkan *output* yang sama, dan semua sel tersebut juga akan mendapatkan *output* yang sama. Lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor *input*. Berdasarkan pada konsep “*winner take-all*”, kelas-kelas yang didapat sebagai hasil dari lapisan-lapisan kompetitif ini hanya bergantung pada jarak antara vektor-vektor *input*. Vektor *input* yang mendekati sama maka lapisan kompetitif akan mengklasifikasikan kedua vektor *input* tersebut ke dalam kelas yang sama[13].

2.3.2 Normalisasi

Normalisasi atau yang biasa disebut dengan proses penskalaan nilai *atribut* dari data sehingga bisa terletak pada rentang tertentu. Normalisasi data bertujuan

untuk menempatkan nilai dalam dalam *range* 0-1 dengan standarisasi. Untuk menskalakan dalam jangkauan [0,1] menggunakan persamaan sebagai berikut[5]

$$\bar{x}_{ik} = \frac{\bar{x}_{ik} - \min(x_k)}{\max(x_k) - \min(x_k)} \dots\dots\dots(1)$$

2.4 Gula Darah (*Diabetes Gestasional*)

Gula darah atau disebut *Diabetes Gestasional (DMG)* adalah suatu keadaan intoleransi glukosa yang terjadi atau ditemukan pada masa kehamilan pertama kali, sekitar 3-5% dari seluruh kehamilan mengalami *DMG*, yang berakibat lebih dari 20.000 kasus setiap tahunnya. Prevelensi berkisar antara 1-14% dari seluruh kehamilan, tergantung dari populasi subjek dan kriteria diagnosis yang digunakan, berbagai kriteria diagnosis, didapatkan prevalensi *diabetes gestasional* pada kehamilan berkisar 1-3%. didapatkan pada 0,1-12% kasus, dengan rata-rata 2,5% kasus.

DMG menyebabkan komplikasi yang signifikan dan berpotensi bagi ibu dan janin termasuk preeklamsia, eklamsi, polihidramnion, makrosomia janin, trauma kelahiran, kelahiran operatif, komplikasi metabolik dan kematian perinatal. *DMG* meningkatkan morbiditas neonatus, yaitu hipoglikemia, ikterus, polisitemia dan makrosomia. Hal ini terjadi karena bayi dari ibu *DMG* mensekresi insulin lebih besar sehingga merangsang pertumbuhan bayi makrosomia. *DMG* juga dapat meningkatkan risiko bagi ibu 3-5% untuk menjadi *diabetes mellitus* di masa mendatang.

Wanita dengan *DMG* hampir tidak pernah memberikan keluhan, sehingga perlu dilakukan skrining. Deteksi dini sangat diperlukan untuk menjarang *diabetes mellitus gestasional* agar dapat dikelola sebaik-baiknya terutama dilakukan pada

ibu dengan faktor resiko. Dengan adanya deteksi dini pada ibu hamil juga dapat membantu untuk meningkatkan kesejahteraan ibu baik selama kehamilan ataupun sesudah kehamilan[1].

2.4.1 Gejala Gula Darah (*Diabetes Gestasional*)

Pada tahap awal *DMG* tipe 2 biasanya tidak menunjukkan gejala *diabetes*.

Gejala umum penderita *diabetes* adalah sebagai berikut[14] :

1. Memiliki berat badan berlebih.
2. Memiliki riwayat tekanan darah tinggi.
3. Pernah mengalami *diabetes gestasional* pda kehamilan sebelumnya.
4. Pernah mengalami keguguran.
5. Pernah melahirkan anak dengan berat bdan 4,5 kg atau lebih.
6. Memiliki riwayat *diabetes* dalam keluarga.
7. Mengalami PCOS.
8. Berusia lebih dari 35 tahun.
9. Pernah mengalami kematian bayi dalam kandungan.
10. Pernah melahirkan bayi dengan keadaan cacat bawaan.
11. Indek masa tubuh lebih dari 30.
12. Hormon insulin
13. Meningkatnya resistensi insulin.

2.4.2 Faktor Risiko Gula Darah (*Diabetes Gestasional*)

Kasus kejadian *DMG* selama ini dipicu ole beberapa faktor risiko yang dapat meningkatkan risiko terjadinya *DMG* pada ibu hamil. Faktor risiko ini terdiri dari beberapa yang dimana terdapat faktor risiko yang dapat dimodifikasi

dan tidak dapat dimodifikasi. Sehingga dengan diketahuinya jumlah faktor risiko yang dimiliki oleh ibu yang sedang hamil, dapat meningkatkan kewaspadaan terutama pada faktor risiko yang dapat dimodifikasi. Faktor risiko lain adalah indeks massa tubuh (IMT) sangat mempengaruhi angka kejadian *DMG*. Sebagaimana telah diketahui, bahwa berat badan ibu sebelum dan selama kehamilan sangat berpengaruh terhadap tingkat kejadian *DMG*, hal ini berhubungan dengan resistensi insulin. Seorang ibu dengan kategori *overweight* dan obesitas pada sebelum atau selama kehamilan akan lebih berisiko terkena *DMG* dibandingkan dengan ibu yang memiliki IMT normal dan kurang/*underweight*, bahkan pada ibu yang *underweight* dapat lebih mencegah terjadinya *DMG* karena dihubungkan dengan penurunan jumlah sel-B. Sedangkan pada ibu dengan *overweight* dan obesitas selain dapat mengakibatkan *DMG*, berdampak juga pada pertumbuhan janin yang berlebih sehingga bayinya makrosomia[15].

2.5 Website

Web adalah sebuah aplikasi yang dijalankan menggunakan *Web* browser dimana pada pembuatan aplikasi *Web* terdiri dari beberapa penggunaan bahasa pemrograman berupa *Hypertext Markup Language (HTML)*, *Cascading Style Sheet (CSS)*, *Hypertext Preprocessor (PHP)*, *Javascript*, dan lain-lain. Aplikasi berbasis *Web* adalah sistem perangkat lunak yang berdasarkan pada teknologi dan standar *World Wide Web Consortium (W3C)*. Yang menyediakan sumber daya *Web* spesifik seperti konten dan layanan melalui sebuah antar muka pengguna dan browser *Web*. Berbeda dengan *Website* statis, aplikasi *Web* atau *Website* dinamis

merupakan sebuah *Web* yang memiliki proses yang kompleks dimana pengguna dapat menjalankan perintah yang diinginkan dan dikirim ke *server* untuk diproses lebih lanjut kemudian diolah dan dikembalikan kepada pengguna dengan tampilan isi yang berbeda beda sesuai dengan permintaan pengguna[16].

Website adalah keseluruhan halaman-halaman *Web* yang terdapat dari sebuah domain yang mengandung informasi. Pengertian *Website* dalam penelitian ini adalah sebagai media/tempat pemrosesan aplikasi dan *interface*. Jadi *Website* aplikasi adalah sebuah aplikasi berbentuk *Web* yang dapat diakses melalui media *browser*. *Browser* akan membaca dan menampilkan halaman *website* yang tersimpan di *webserver* dengan protocol yang disebut HTTP (*Hypertext Transfer Protocol*). Terdapat banyak browser yang tersedia saat ini, contohnya: *Internet Explorer*, *Mozilla Firefox*, *Opera*, dan lainnya[17].

2.6 Database MySQL (Structured Query Language)

Database (basis data) adalah kumpulan suatu informasi yang disimpan didalam komputer secara sistematis dimana suatu informasi tersebut dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi. Ada beberapa perangkat lunak atau *software open source* yang disediakan untuk membuat suatu basis data. Perangkat lunak tersebut merupakan suatu pemrograman yang dikategorikan sebagai bahasa pemrograman tingkat tinggi (*high level language*) salah satunya adalah *MySQL*. *Database MySQL* dapat dibuat menggunakan tampilan *PhpMyAdmin* atau menggunakan sebuah *script* dalam *PHP*. Istilah *SQL* dapat diartikan sebagai suatu bahasa yang digunakan untuk mengakses suatu data dalam *database relational* dan terstruktur sedangkan

MySQL adalah *software* atau *tools* untuk mengelola atau memanajemen *SQL* dengan menggunakan *Query* atau bahasa khusus. *MySQL* juga merupakan salah satu *software* yang bersifat *open source*. Sebagai suatu pengelola *database* terbesar dan paling banyak digunakan tentunya *MySQL* ini memiliki fitur atau kapabilitas tertentu. Salah satu yang paling dicari oleh para pengguna[18].

MySQL adalah *Relational Database Management System (RDBMS)* yang didistribusikan secara gratis dibawah lisensi *General Public Lisensi (GPL)* setiap orang bebas menggunakannya, tetapi tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* adalah basis data yang mengandung satu atau jumlah tabel. Tabel dari atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel[19].

2.7 Hyper Text Markup Language (HTML)

Hyper Text Markup Language (HTML) adalah *script* pemrograman yang mengatur bagaimana kita menyajikan informasi di dunia internet dan bagaimana informasi itu membawa kita melompat dari satu tempat ke tempat lainnya. *HTML* dibuat oleh Tim Berners-Lee ketika masih bekerja dengan CERN dan dipopulerkan pertama kali oleh *Browser Mosaic*. Awal tahun 1990 *HTML* mengalami perkembangan yang sangat maju. Setiap pengembangan *HTML* pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya[20].

Tugas atau fungsi dari sebuah *HTML* yaitu[18]:

1. Membuat teks dasar, seperti mengatur paragraf, font, struktur.

2. Menentukan *layout Website*.
3. Membuat *list*.
4. Membuat *table*.
5. Menyisipkan sebuah video, audio, gambar ataupun gif.
6. Membuat *link*

Saat ini versi terbaru dari *HTML* adalah *HTML 5*. *HTML 5* berisi beberapa fitur baru, tetapi tetap membawa fitur terdahulu dari versi *HTML* sebelumnya.

2.8 Visual Studio Code

Visual Studio Code adalah kode editor sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux* dan *macOS*. Ini termasuk dukungan untuk *debugging*, kontrol git yang tertanam dan GitHub, penyorotan sintaksis, penyelesaian kode cerdas, snippet dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi dan menginstal ekstensi yang menambah fungsionalitas tambahan[21].

2.9 Xampp (X-platform, Apache, MySQL, PHP, Perl)

Xampp adalah singkatan dari (*X-platform, Apache, MySQL, PHP, Perl*). Perangkat lunak berbasis *web server* yang bersifat *open source* (bebas), serta mendukung di berbagai sistem operasi, baik *Windows*, *Linux* atau *MacOs*. *XAMPP* digunakan sebagai *standalone server* (berdiri sendiri) atau biasa disebut dengan *localhost*. Hal tersebut memudahkan dalam proses pengeditan desain dan pengembangan aplikasi. Penggunaan dari *XAMPP* sangat dibutuhkan untuk dapat mengembangkan *software* ataupun tampilan *website* dengan lebih mudah, cepat

dan terstruktur. Terdapat tiga komponen penyusun utama dari *tools* ini yaitu *htdocs*, *Control Panel*, dan *PHPMYAdmin*[20].

2.10 *PhpMyAdmin*

PhpMyAdmin merupakan aplikasi berbasis *web* yang dikembangkan menggunakan bahasa pemrograman *PHP*. Melalui *PhpMyAdmin*, *user* dapat melakukan perintah *Query* tanpa harus mengetikkan seperti pada MS DOS. Perintah tersebut misalnya administrasi *User* dan *privileges*, *export* dan *import database*, manajemen *database*, manajemen label dan struktur tabel, dan sebagainya. *PhpMyAdmin* sangat *user friendly* dan sangat mudah untuk digunakan[22].

PHP adalah sebuah bahasa *script server side* yang dapat digunakan dengan bahasa *HTML* atau dokumen secara bersamaan untuk membangun sebuah aplikasi *web*. Bahasa *PHP* mirip dengan bahasa *C*, *Perl* dan *Java* dengan keunikan tersendiri[19].

2.11 *UML (Unified Modeling Language)*

Unified Modeling Language (UML) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. *Unified Modeling Language (UML)* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. *UML* hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan *UML* tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya *UML* paling banyak digunakan pada metodologi berorientasi objek[23]

Beberapa diagram yang terdapat pada *UML* adalah sebagai berikut[24]:

1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat.

2. *Class Diagram*

Class Diagram adalah sebuah diagram yang menggambarkan tipe-tipe objek yang berada di dalam sistem, lengkap dengan relasi-relasi antara mereka. *Class diagram* juga menunjukkan operasi-operasi dan *property-property* yang dimiliki oleh masing-masing *class*.

3. *Sequence Diagram*

Sequence diagram adalah salah satu diagram yang terdapat pada *UML*. *Sequence Diagram* merupakan diagram yang menggambarkan bagaimana urutan suatu objek-objek dalam suatu grup dapat bekerja secara bersama-sama. *Sequence diagram* bisa juga disebut sebagai *event diagram* atau *event scenario*.

4. *Activity Diagram*

Activity diagram adalah salah satu diagram yang juga terdapat pada *UML*. *Activity diagram* berguna untuk menunjukkan aliran dari suatu aktivitas ke aktivitas lainnya. *Activity diagram* juga dapat membantu orang-orang yang terlibat dalam pembuatan sistem, baik dari sisi pengembang maupun dari sisi pengguna supaya mampu memiliki pemahaman yang sama.

2.12 Penelitian Terkait

Tabel 2. 1 Penelitian Terkait

No	Nama Penulis / Tahun	Judul	Metode	Hasil
1.	Rianti Nurpalah, dkk (2023)	Deteksi Dini <i>Diabetes Mellitus Gestasional (Dmg)</i> Melalui Pemeriksaan Glukosa Darah Sebagai Upaya Pencegahan Komplikasi Pada Ibu Hamil	-	Berdasarkan kegiatan yang telah dilakukan yaitu pemeriksaan glukosa darah pada ibu hamil sebagai upaya deteksi dini <i>DMG</i> di Wilayah Kelurahan Karikil yang di ikuti oleh 25 orang ibu hamil sebagai responden, didapatkan hasil bahwa seluruh responden 100% memiliki kadar glukosa darah normal. Ibu hamil yang memiliki kadar glukosa darah normal diharapkan tetap menjalankan menjaga pola makan dan pola hidup yang sehat supaya kesehatannya bisa tetap terjaga serta bayi yang dilahirkan memiliki kesehatan yang optimal.
2.	Risti Vika Arvianti (2019)	Klasifikasi Penyakit <i>Diabetes Mellitus</i> Tipe 2 Menggunakan <i>Learning Vector Quantization (LVQ)</i>	<i>Learning Vector Quantization (LVQ)</i>	Proses normalisasi data menggunakan metode <i>Min-Max</i> langkah awal yang harus dilakukan adalah menentukan

				<p>nilai <i>Minimal</i> dan <i>Maksimal</i> setiap variabel. Pada penelitian ini didapatkan nilai minimal pada variabel usia sebesar 20 dan nilai maksimal usia yaitu 88, nilai minimal pada variabel obesitas yaitu 35 dan nilai maksimal obesitas yaitu 92, nilai minimal pada variabel kadar gula darah adalah 30 dan nilai maksimal kadar gula darah adalah 510. Selanjutnya dilakukan proses normalisasi dan hasilnya dilakukan pembulatan sehingga data akhir berbentuk biner.</p>
3.	Nurdiana Djamiludin S.Kep, Ns, M.Kep, Vera Mila Oktavia Mursalin	Gambaran <i>Diabetes Melitus Gestasional</i> Pada Ibu Hamil di RSUD	-	<p>Kejadian <i>diabetes melitus gestasional</i> pada ibu hamil di RSUD Prof. Dr. H. Aloei Saboe Kota Gorontalo adalah sebanyak 17 orang (56,7%), dan yang normal berjumlah 13 orang (43,3%).Dimana 16 orang ibu yang mengalami <i>diabetes melitus gestasional</i> tidak memiliki riwayat DM, sedangkan 1 orang ibu yang mengalami <i>diabetes melitus gestasional</i> memiliki</p>

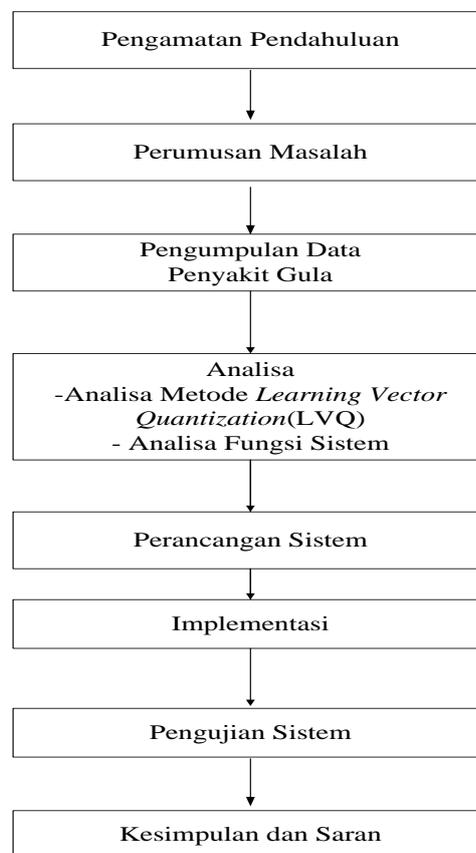
				riwayat <i>diabetes mellitus</i> .
4.	Esi Putri Silmina, dkk (2020)	Perancangan Sistem Pakar Diagnosis Penyakit <i>Diabetes Melitus Gestasional</i> Pada Ibu Hamil Menggunakan Fuzzy Mamdani	Fuzzy Mamdani	Kesimpulan dari penelitian ini adalah <i>fuzzy mamdani</i> dapat menyelesaikan perhitungan sistem pakar untuk diagnosis penyakit <i>Diabetes Melitus Gestasional</i> . Terdapat 6 variabel input, 64 rule pada proses defuzifikasi dan 2 status kehamilan yaitu kehamilan normal dan <i>gestational diabetes</i> .
5.	Elvia Budianita, dkk (2018)	Penerapan Metode <i>Learning Vector Quantization 2 (LVQ 2)</i> Untuk Menentukan Gangguan Kehamilan Trimester I	<i>Learning Vector Quantization (LVQ)</i>	Berdasarkan hasil dari pengujian parameter algoritma <i>LVQ 2</i> untuk sistem penentuan gangguan kehamilan trimester I pada penelitian ini, dapat disimpulkan bahwa jumlah data latih, nilai learning rate, serta nilai alfa mempengaruhi hasil akurasi pengujian pada <i>LVQ2</i> . Pada saat window = 0 data uji ke 12 selalu tidak sesuai dengan target hal ini disebabkan karena pada data uji ke 12 dengan target kelas = 4 terdapat gejala yang mirip dengan kelas = 3. Akan tetapi dengan menggunakan window, data uji ke

				<p>12 dapat dikenali sesuai dengan target kelas sebenarnya. Hal ini dapat disimpulkan bahwa window pada pembelajaran <i>Learning Vector Quantization 2 (LVQ2)</i> mempengaruhi dalam mengenali data yang memiliki kesamaan gejala (mirip).</p>
--	--	--	--	--

BAB 3

METODOLOGI PENELITIAN

Penelitian ini dilakukan dengan melaksanakan tahapan demi tahapan yang berhubungan. Tahapan- tahapan tersebut dijabarkan dalam metode penelitian. Metode penelitian diuraikan kedalam bentuk skema yang jelas, teratur, dan sistematis. Berikut tahapan-tahapan penelitian dapat dilihat pada gambar 3.1 :



Gambar 3. 1 Metodologi Penelitian

Penjelasan dari tahapan-tahapan penelitian pada gambar 3.1 dapat dilihat pada penjelasan dibawah ini:

3.1 Pengamatan Pendahuluan

Pengamatan pendahuluan merupakan tahapan awal yang dilakukan dalam penelitian ini, yang menggunakan metode *Learning Vector Quantization (LVQ)* yang dijadikan sebagai penelitian studi pustaka dalam penelitian Skripsi ini. Pada penelitian metode ini yaitu menggunakan pengenalan gejala penyakit gula darah pada ibu hamil.

3.2 Perumusan Masalah

Berdasarkan hasil dari tahapan pengamatan pendahuluan sebelumnya, maka tahapan selanjutnya adalah tahapan perumusan masalah. Pada tahapan perumusan masalah akan dirumuskan masalah yang dianggap sebagai penelitian dalam Skripsi ini. Permasalahan-permasalahan yang menjadi rumusan masalah dalam penelitian ini didapatkan dari penelitian terkait data pengamatan pendahuluan sebelumnya. Solusi yang didapatkan pada tahapan perumusan masalah ini yang akan menjadi judul penelitian Skripsi ini “Deteksi Penyakit Gula Darah pada Ibu Hamil Menggunakan Metode *Learning Vector Quantization (LVQ)*”..

3.3 Pengumpulan Data

Pengumpulan data adalah tahapan-tahapan yang bertujuan dalam memperoleh data-data informasi yang berhubungan dengan penelitian Skripsi ini. Pada tahapan pengumpulan data ini juga berguna untuk mengumpulkan semua kebutuhan data yang akan diproses nantinya menggunakan metode *Learning Vector Quantization (LVQ)*. Dalam pengumpulan data ini ada dua data yang dikutip adalah sebagai berikut:

1. Data penyakit gula darah pada ibu hamil dan data gejala untuk diproses atau di inputkan.
2. Data dalam metode *Learning Vector Quantization (LVQ)*. Data yang diterjemahkan dalam program untuk dijabarkan dalam pengenalan penyakit gula darah pada ibu hamil.

3.4 Analisa

Tahapan selanjutnya adalah melakukan analisa metode sistem dari penelitian Skripsi ini. Adapun tahapan analisa dalam penelitian Skripsi ini adalah sebagai berikut:

3.4.1 Analisa Metode *Learning Vector Quantization (LVQ)*

Learning Vector Quantization (LVQ) merupakan sebuah teknik yang terinspirasi dari cara manusia, menghasilkan persepsi menghasilkan akurasi yang lebih kuat dalam penentuan gejala penyakit gula darah. Metode *Learning Vector Quantization (LVQ)* untuk melakukan pembelajaran pada lapisan-lapisan kompetitif yang terawasi. Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vector-vektor masukan yang diberikan. Hal ini memberikan metode *Learning Vector Quantization (LVQ)* kemampuan dalam mengenali sebuah penyakit berdasarkan gejala.

3.4.2 Analisa Fungsional Sistem

Setelah melakukan tahapan analisa terhadap metode *Learning Vector Quantization (LVQ)* maka selanjutnya adalah analisis *fungsional* yang akan

dibangun. Adapun tahapan-tahapan analisis *funksional* yaitu dalam pembuatan *flowchart*, *Unified Modeling Language (UML)*, dan perancangan *user interface*.

3.5 Perancangan Sistem

Setelah melakukan tahapan dilakukan, maka tahapan selanjutnya adalah perancangan sistem. Tahapan perancangan sistem terdiri dari:

1. Perancangan struktur menu yang akan digunakan pada sistem yang akan dibangun.
2. Tahapan rancangan *database* beserta atribut yang dibutuhkan.
3. Tahapan perancangan *user interface* atau antar muka pengguna terhadap sistem yang akan digunakan.

3.6 Implementasi Sistem

Beberapa komponen pendukung yang memiliki peran yang sangat penting dalam implementasi sistem diantaranya adalah perangkat keras (*hardware*). Adapun spesifikasi dari perangkat keras (*hardware*) yang digunakan sebagai berikut :

1. Perangkat keras (*hardware*), antara lain :

Processor : Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz 1.19GHz

Memory : 4.00 GB (3.75 GB usable)

System type : 64-bit operating system, x64-based processor

Hardisk : 131 GB

2. Perangkat lunak (*software*), antara lain :

Sistem operasi : *Windows 10*

Tool : *AMPPS, Visual Studio Code*

3.7 Pengujian

1. *Black Box*

Pengujian *Black Box* adalah pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji kode atau sisi internal programnya. Artinya, hanya sisi fungsi, antarmuka dan alurnya saja tanpa menyentuh kode atau *script* dari perangkat lunak. *Black Box* berfokus pada pengujian dari masing-masing spesifikasi fungsional perangkat lunak. Seorang tester dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada fungsional perangkat lunak. Dengan demikian, setiap *input* pada perangkat lunak dapat diperhatikan kinerjanya berdasarkan fungsi yang ingin dicapai. Apabila terjadi *error* atau tidak sesuai dengan tujuan yang diharapkan maka dilakukan penganalisaan sistem kembali hingga tidak ditemukan *error*, sehingga ditemukan informasi tentang penyakit beserta solusinya berdasarkan macam gejala yang timbul.

Langkah-langkah untuk melakukan pengujian *Black Box*:

1. Analisa kebutuhan dan spesifikasi
2. Penguji memilih *input* yang valid untuk memeriksa apakah SUT memprosesnya dengan benar. Selain itu, beberapa *input* yang tidak valid dipilih untuk memverifikasi bahwa SUT dapat mendeteksinya.
3. Penguji menentukan *output* yang diharapkan untuk semua *input* tersebut.
4. Penguji perangkat lunak membuat kasus uji dengan *input* yang dipilih.
5. Kasus uji dijalankan.
6. Review hasil, pengujian yang telah dilaksanakan dievaluasi pelaksanaannya, sehingga menghasilkan berbagai catatan dan dokumentasi

mengenai bug, cacat, kesalahan dan berbagai fungsi yang telah berjalan sebagaimana mestinya.

3.8 Kesimpulan dan Saran

Dengan adanya Deteksi penyakit gula darah ini dapat membantu mendiagnosa penyakit diabetes pada ibu hamil berdasarkan kriteria yang muncul. sehingga pengguna aplikasi ini khusus pihak tenaga medis dalam menganalisa penyakit gula darah pada ibu hamil secara efisien. Aplikasi ini juga bisa mempermudah dalam penanganan dan solusi yang terjadi pada ibu hamil dengan berdasarkan macam gejala yang timbul.