

Sistem *Smart Healthcare* dalam Diagnosa Tingkat Stres Menggunakan Metode *Certainty Factor*

Ni Putu Vina Amandari^{a1}, Ngurah Agus Sanjaya ER^{a2}, I Gusti Ngurah Anom Cahyadi Putra^{a3}, Ida Bagus Gede Dwidasmara^{a4}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana Badung, Bali, Indonesia

¹vinamandari1@gmail.com

²agus_sanjaya@unud.ac.id

³anom.cp@unud.ac.id

⁴dwidasmara@unud.ac.id

Abstrak

Stres atau tegang merupakan suatu respon tubuh manusia yang terjadi ketika menghadapi suatu kondisi tertentu, seperti kondisi tertekan. Penyebab terjadinya stres pada seseorang bisa disebabkan oleh banyak faktor misalnya pekerjaan, lingkungan, masalah keluarga, atau kesehatan. Stres yang berlebihan dapat mempengaruhi kesehatan mental dan fisik seseorang. Maka dari itu diagnosis dini dan pengelolaan stres dapat membantu mencegah atau mengurangi risiko kesehatan ini. Pada penelitian ini, penulis merancang sistem diagnosis tingkat stres berbasis IoT menggunakan sensor suhu DS18B20 untuk membaca data suhu, pulse sensor untuk membaca detak jantung, dan GSR sensor untuk membaca tingkat konduktivitas kulit seseorang. Ketiga parameter tersebut digunakan sebagai inputan untuk melakukan diagnosa tingkat stres menggunakan metode certainty factor dengan output tingkatan stres terdiri dari rileks, tenang, cemas, dan tegang. Pengujian sistem dilakukan dengan cara melakukan diagnosa pada 20 subjek dengan setiap subjek melakukan 5 kali pengambilan data disetiap parameternya dimana kelima data disetiap parameternya kemudian dihitung nilai rata-ratanya untuk mendapatkan data yang lebih valid. Berdasarkan hasil perbandingan diagnosa sistem dengan kuisioner DASS 42 didapatkan tingkat akurasi sistem sebesar 85%.

Kata Kunci: *Internet of Things, DS18B20, Pulse Sensor, GSR, Stres, Certainty Factor*

1. Pendahuluan

Stres atau tegang merupakan suatu respon tubuh manusia yang terjadi ketika menghadapi suatu kondisi tertentu, seperti kondisi tertekan. Hal ini umumnya timbul karena pikiran dan tubuh bersama-sama membuat kesalahan yang sama sehingga memicu aksi dan reaksi fisiologis tubuh seseorang. Stres dapat terjadi kepada siapa saja tidak terkecuali pada mahasiswa. Mahasiswa berada pada masa transisi sebelum menjalani kehidupan yang sebenarnya di masyarakat dimana mereka seringkali dihadapkan pada berbagai tekanan baik dari segi akademik, sosial, dan pribadi dimana keadaan dapat memicu timbulnya stres.[1] Presentase mahasiswa yang mengalami stres akademik secara global sebesar 38-71%, di Asia Tenggara sebesar 39,6-61,3%, sedangkan di Indonesia presentase mahasiswa yang mengalami stres akademik sebesar 36,7-71,6%.[1]

Diagnosis tingkat stres sangat penting untuk kesehatan mental, karena membantu mengidentifikasi tingkat keparahan dan dampak stres terhadap kesejahteraan seseorang. Stres yang tidak diobati dapat menyebabkan berbagai masalah kesehatan mental, seperti kecemasan dan depresi, serta penyakit fisik seperti penyakit kardiovaskular. Menggunakan pendekatan tradisional dalam mendiagnosa tingkat stres seperti observasi dan wawancara memiliki beberapa keterbatasan dimana pendekatan ini mungkin membutuhkan waktu yang lebih lama untuk mengumpulkan data yang cukup dan akurat. Maka dari itu dibutuhkan alternatif pendekatan yang lebih efektif untuk mendiagnosa tingkat stres seseorang.

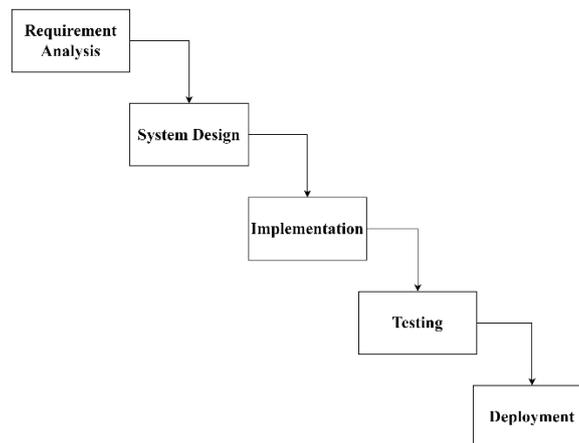
Stres dapat mempengaruhi respon fisiologi tubuh sehingga tingkat stres seseorang dapat diukur dan dianalisa berdasarkan detak jantung, suhu tubuh, dan konduktivitas kulit. *Wireless Sensor Networks* (WSN) merupakan salah satu teknologi yang menjadi trend seiring dengan perkembangan teknologi transmisi data dan kebutuhan manusia akan *Internet of Things* (IoT). IoT merupakan teknologi yang mampu menghubungkan benda-benda fisik ke dalam jaringan dan membuat benda-benda tersebut terhubung satu sama lain dan memfasilitasi adanya komunikasi. *Wireless Sensor Network*

merupakan salah satu unsur teknologi di dalam IoT yang membantu terciptanya konektivitas. IoT sudah banyak diterapkan di berbagai aplikasi untuk membantu pekerjaan manusia salah satunya yaitu pada sistem *Smart Healthcare*, dimana IoT akan membantu pasien atau pengguna untuk memperhatikan kesehatannya. *Smart Healthcare* merupakan layanan kesehatan yang ditunjang dengan kemajuan teknologi digital. Sistem ini menggunakan node sensor untuk menyimpan data pasien seperti profil pasien, detail demografis, hasil diagnosis, dan lain sebagainya.[2]

Berdasarkan permasalahan yang ada serta seiring dengan berkembangnya IoT maka dalam penelitian ini penulis membuat sebuah sistem yang mampu mendiagnosa tingkat stres seseorang dengan memanfaatkan beberapa sensor diantaranya sensor suhu, sensor konduktivitas kulit, dan sensor detak jantung. Hasil pembacaan dari ketiga sensor tersebut akan diproses dengan menggunakan metode *certainty factor* untuk mendapatkan hasil diagnosa tingkat stres seseorang. Data yang didapatkan dari ketiga sensor dan hasil diagnosa akan disimpan dalam bentuk rekam medis digital dimana rekam medis digital ini dapat diakses melalui website.

2. Metode Penelitian

Dalam penelitian ini, penulis mengembangkan sistem dengan menggunakan metode *System Development Life Cycle* (SDLC) model *Waterfall*. Tahapan pada model *waterfall* dapat dilihat melalui grafik pada Gambar 1 berikut.



Gambar 1. Metode Waterfall

2.1 Analisis Kebutuhan

Untuk mempermudah menganalisis sistem yang dibuat maka dibutuhkan analisis kebutuhan sistem yang terdiri dari kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional mencakup proses apa saja yang nantinya akan dilakukan oleh sistem sedangkan kebutuhan non fungsional menitikberatkan pada perilaku yang dimiliki oleh sistem.

a. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berisi proses apa saja yang nantinya dilakukan oleh sistem. Berikut ini kebutuhan fungsional dari sistem yang penulis buat :

- Sistem mampu membaca suhu tubuh, *Galvanic Skin Response* (GSR), dan detak jantung yang akan diambil menggunakan sensor
- Sistem mampu melakukan diagnosa tingkat stres seseorang menggunakan metode *certainty factor* berdasarkan data suhu tubuh, konduktivitas kulit, dan detak jantung
- Sistem mampu menampilkan data serta hasil diagnosa pengguna melalui halaman website

b. Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan.

1. Analisis Kebutuhan Perangkat Keras

Perangkat keras yang akan digunakan pada pembuatan sistem ini diantaranya :

- | | |
|---------------------|-----------------------|
| - Arduino Mega 2560 | - Sensor Suhu DS18B20 |
| - Modul WiFi ESP-01 | - GSR Sensor |

- Pulse Sensor
- Kabel Jumper
- Breadboard
- LCD
- Button
- Buzzer

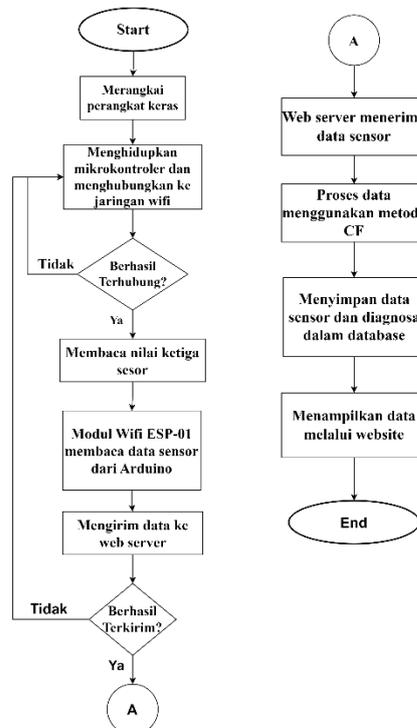
2. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam proses pembuatan sistem ini yaitu :

- Arduino IDE
- MySQL
- Code Editor
- Server side programming

2.2 Design System

a. Rancangan Alur Kerja *Client* dan *Server*



Gambar 2. Rancangan Alur Kerja *Client* dan *Server*

Melalui grafik yang dapat dilihat pada Gambar 2, alur kerja sistem dimulai dengan menghidupkan *hardware* kemudian arduino akan mencoba menghubungkan modul WiFi ESP-01 ke jaringan yang telah dikonfigurasi berdasarkan SSID dan *password* WiFi yang digunakan. Apabila berhasil terhubung ke jaringan, arduino akan membaca data dari ketiga sensor kemudian data ketiga sensor akan diproses terlebih dahulu sebelum dikirim ke modul WiFi ESP-01. Data yang telah diproses akan dikirim ke server melalui modul WiFi ESP-01. Server akan menerima data ketiga sensor kemudian memproses ketiga data tersebut menggunakan metode *certainty factor* untuk menghasilkan dignosa. Ketiga data sensor dan hasil diagnosa kemudian akan disimpan pada database sebelum ditampilkan melalui antarmuka website.

b. Rancangan Metode *Certainty Factor*

Certainty factor merupakan metode yang digunakan untuk mendefinisikan keyakinan suatu fakta berdasarkan tingkat keyakinan seorang pakar. Metode *Certainty Factor* memiliki kelebihan yaitu metode ini sangat cocok digunakan untuk sistem pakar karena keakuratan data yang diolah dapat terjaga.[3] Sebelum menentukan rule yang dibutuhkan metode *certainty factor* dalam mendiagnosa tingkat stres, diperlukan range parameter disetiap kondisinya. Parameter tingkat stress untuk usia dewasa muda di setiap kondisi dapat dilihat melalui Tabel 1 berikut. [4]

Tabel 1. Parameter Tingkat Stres Dewasa Muda

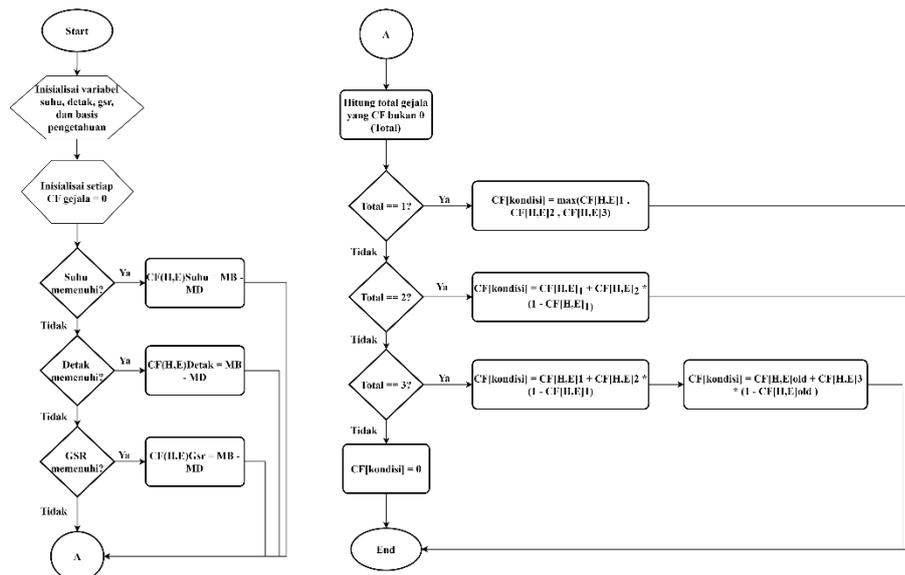
Kondisi	Parameter
---------	-----------

	Suhu Tubuh (°C)	Konduktivitas Kulit (Mv)	Detak Jantung (BPM)
Rileks / Relaxed	36-37	<2	60-70
Tenang / Calm	35-36	2-4	70-90
Cemas / Tense	33-35	4-6	90-100
Tegang / Stressed	<33	>6	>100

Setelah diketahui parameter tingkatan stres seperti pada Tabel 1, maka dibuatkan sebuah basis pengetahuan yang berisi nama kondisi, kode gejala, nama gejala serta nilai MB dan nilai MD di masing-masing gejala. Informasi terkait nama kondisi, nama gejala, hingga nilai MB dan MD disetiap gejala dapat dilihat melalui Tabel 2 berikut.

Tabel 2. Nilai MB dan MD Setiap Kondisi

Kode Kondisi	Nama Kondisi	Kode Gejala	Nama Gejala	MB	MD
K1	Rileks	G01	Suhu ≥ 36 dan Suhu ≤ 37	0.6	0.4
		G02	GSR < 2	0.6	0.4
		G03	Detak jantung ≥ 60 dan detak jantung ≤ 70	0.8	0.2
K2	Tenang	G04	Suhu ≥ 35 dan Suhu ≤ 36	0.6	0.4
		G05	GSR ≥ 2 dan GSR ≤ 4	0.8	0.2
		G06	Detak jantung ≥ 70 dan detak jantung ≤ 90	0.6	0.4
K3	Cemas	G07	Suhu ≥ 33 dan Suhu ≤ 35	0.6	0.4
		G08	GSR ≥ 4 dan GSR ≤ 6	0.6	0.4
		G09	Detak jantung ≥ 90 dan detak jantung ≤ 100	0.8	0.2
K4	Tegang	G10	Suhu < 33	0.6	0.4
		G11	GSR > 6	0.8	0.2
		G12	Detak jantung > 100	0.8	0.2



Gambar 3. Alur Kerja Metode *Certainty Factor*

Alur kerja metode *certainty factor* dalam sistem ini mulai dari inisialisasi basis pengetahuan hingga menghasilkan diagnosa tingkat stress dapat dilihat melalui *flowchart* pada Gambar 3.

1. Inisialisasi basis pengetahuan dan beberapa variable
2. Melakukan pengecekan di setiap gejala apakah memenuhi syarat kondisi atau tidak secara bertahap
3. Jika gejala memenuhi syarat suatu kondisi maka hitung nilai CF gejala tersebut kemudian variable jumlahCFBukanNol bertambah satu. Berikut rumus perhitungan nilai CF gejala [5][5]:

$$CF[H, E] = MB[H, E] - MD[H, E] \tag{1}$$

4. Melakukan pengecekan terhadap nilai variabel jumlahCFBukanNol yang menampung informasi mengenai total gejala yang memenuhi persyaratan suatu kondisi

- Apabila terdapat satu gejala yang memenuhi maka nilai CF kondisi akan sama dengan nilai CF gejala yang memenuhi
- Apabila terdapat dua gejala yang memenuhi maka, nilai CF kondisi dihitung dengan rumus berikut [5]:

$$CF\ Combine [H, E]1, 2 = CF[H, E]1 + CF[H, E]2 * (1 - CF[H, E]1) \tag{2}$$

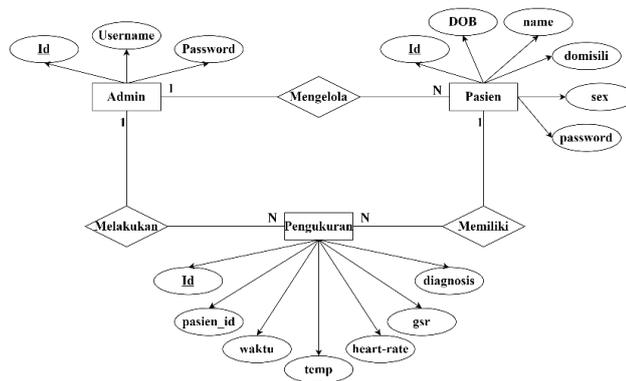
- Apabila terdapat tiga gejala yang memenuhi maka, nilai CF kondisi dihitung dengan rumus berikut[5]:

$$CF\ Combine [H, E]old, 3 = CF[H, E]old + CF[H, E]3 * (1 - CF[H, E]old) \tag{3}$$

- Apabila tidak terdapat gejala yang memenuhi, maka nilai CF kondisi = 0.

c. *Entity Relationship Diagram (ERD)*

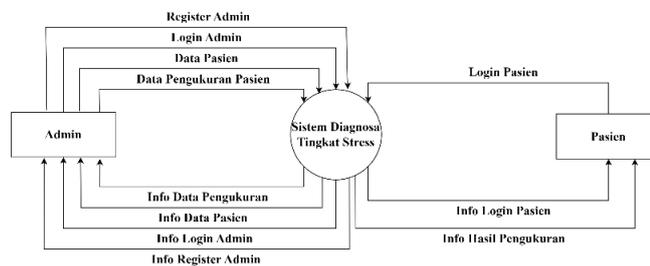
Entity Relationship Diagram (ERD) digunakan untuk memodelkan dan merancang desain basis data untuk memudahkan pemahaman tentang bagaimana data akan disimpan, diorganisir, dan diakses di dalam sistem. Basis data pada sistem ini terdiri dari tiga entitas yaitu Admin, Pasien, dan Pengukuran. Informasi lebih rinci terkait atribut di setiap entitas serta relasi yang dimiliki oleh entitas satu dengan lainnya dapat dilihat melalui Gambar 4 berikut.



Gambar 4. *Entity Relationship Diagram*

d. *Data Flow Diagram (DFD)*

1. DFD Level 0

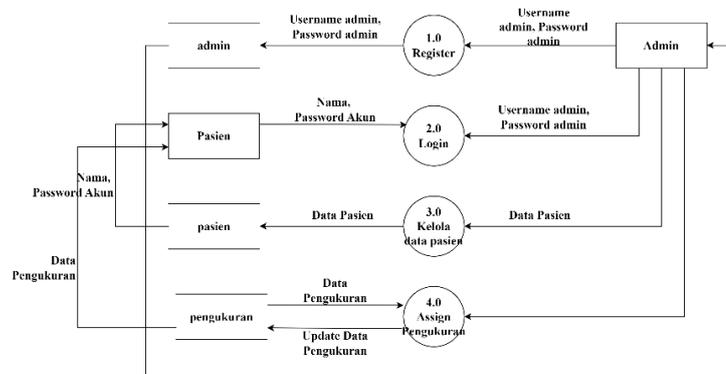


Gambar 5. DFD Level 0

DFD Level 0 digunakan untuk menunjukkan hubungan antara sistem dengan entitas eksternal tanpa penjelasan rinci mengenai proses internal. Seperti yang terlihat pada Gambar 5, terdapat dua entitas yang berinteraksi dengan sistem yaitu admin dan pasien. Alur data dari entitas admin yaitu admin dapat menginputkan data register, data login untuk admin, mengelola data pasien termasuk data pengukuran kemudian sistem akan memberikan respon berupa informasi bahwa register berhasil, respon ketika login berhasil, serta informasi mengenai hasil pengelolaan data pasien. Dari sisi pasien, pasien dapat menginputkan data untuk login kemudian sistem akan memberikan respon bahwa login berhasil dan menampilkan informasi data pengukuran.

2. DFD Level 1

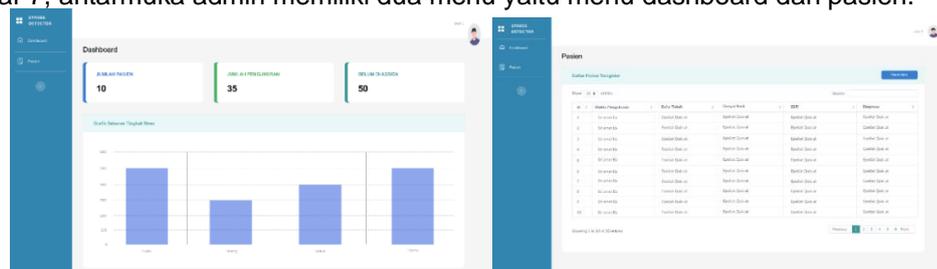
DFD Level 1 digunakan untuk merinci proses internal dan penjelasan lebih mendalam mengenai bagaimana data mengalir dalam sistem. Dapat dilihat melalui diagram pada Gambar 6, terdapat empat proses yaitu proses register, login, kelola data pasien, dan assign pengukuran. Pada proses register, admin akan memasukkan *username* dan *password* dimana kemudian kedua data tersebut akan disimpan pada database di tabel admin. Data yang sudah tersimpan akan digunakan admin untuk login ke sistem. Pada proses kelola data pasien, admin dapat mengelola data pasien kemudian pasien sebagai user dapat melakukan login berdasarkan nama dan password yang tersimpan pada tabel pasien. Pada proses *assign* pengukuran, data pengukuran pada tabel pengukuran akan di *assign* oleh admin menjadi data pengukuran milik id pasien dari tabel pasien yang dipilih admin untuk diassign. Data pengukuran yang di *assign* akan diupdate kolom *id_pasien* nya menjadi id milik pasien yang dipilih.



Gambar 6. DFD Level 1

e. Desain Antarmuka

Desain antarmuka website diimplementasikan ke dalam kode dengan bahasa pemrograman *PHP*, *HTML*, *CSS*, dan *framework bootstrap*. Website ini nantinya dapat diakses oleh dua pengguna yaitu sebagai admin dan sebagai user atau pasien. Seperti yang terlihat pada Gambar 7, antarmuka admin memiliki dua menu yaitu menu dashboard dan pasien.



Gambar 7. Desain Antarmuka Admin



Gambar 8. Desain Antarmuka User

Pada Gambar 8 menunjukkan desain antarmuka website user atau pasien dimana antarmuka user memiliki dua menu yaitu menu dashboard dan history.

2.3 Implementasi

Tahap implementasi ini meliputi implementasi rancangan perangkat keras, pembuatan database, pemrograman mikrokontroler agar dapat membaca data dari sensor serta mengirim data sensor ke database, pemrograman metode *certainty factor*, serta implementasi perancangan antarmuka website.

2.4 Pengujian

Pengujian tingkat akurasi hasil diagnosa metode *certainty factor* yang diterapkan pada sistem dilakukan dengan membandingkan hasil diagnosa sistem dengan hasil diagnosa kuisisioner *DASS 42*. *DASS 42* (*Depression Anxiety Stress Scale 42*) merupakan instrumen yang digunakan untuk mengukur tingkat depresi, kecemasan, dan stress pada individu. Instrumen ini memiliki 42 item pertanyaan yang mencakup tiga subvariabel yaitu fisik, emosi/psikologis, dan perilaku individu.[6]

Tabel 3. Conversion of DASS 42 test tool

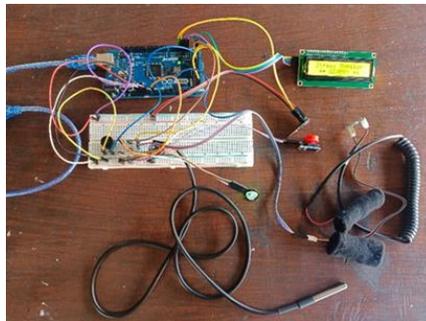
DASS 42	Tool
Depresi, Kecemasan, Stress; Normal dan Ringan	Rileks / Relaxed
Depresi, Kecemasan, Stress; Sedang	Tenang / Calm
Depresi, Kecemasan, Stress; Berat	Cemas / Anxiety
Depresi, Kecemasan, Stress; Sangat Berat	Tegang / Stress

Dapat dilihat melalui Tabel 3 bahwa kuisisioner DASS 42 memiliki 3 skala yaitu depresi, kecemasan dan stres dimana masing-masing skala terdiri dari 4 kondisi yaitu normal/ringan, sedang, berat, dan sangat berat. Dari ketiga skala tersebut apabila terdapat minimal 2 kondisi yang sama maka hal tersebut dapat digunakan sebagai patokan untuk mengukur tingkat kesesuaian yang paling mendominasi. Jika tidak terdapat kondisi yang sama maka kesesuaian tidak dapat disimpulkan.[6]

3. Hasil dan Pembahasan

3.1 Hasil Implementasi Perangkat Keras

Hasil implementasi perangkat keras dari sistem diagnosa tingkat stress menggunakan metode *certainty factor* dapat dilihat melalui Gambar 9 berikut. Perangkat keras dari sistem ini terdiri dari Arduino Mega, modul WiFi ESP-01, sensor suhu DS18B20, pulse sensor, GSR sensor, buzzer, LCD, button, dan breadboard.



Gambar 9. Hasil Implementasi Perangkat Keras

3.2 Hasil Implementasi Database

Perancangan database untuk sistem diagnosa tingkat stress ini dirancang menggunakan MySQL. Database sistem ini terdiri dari tiga tabel, yaitu tabel pengukuran, tabel pasien, dan tabel admin. Tabel pengukuran yang dapat dilihat pada Gambar 10 digunakan untuk menyimpan data yang dikirim dari sensor dan hasil diagnosa tingkat stress menggunakan metode *certainty factor*. Informasi mengenai column serta tipe data dari tabel ini dapat dilihat pada gambar 10 berikut.

Select data Show structure Alter table New item

Column	Type	Comment
id	int unsigned Auto Increment	
pasien_id	int [0]	
waktu	datetime [CURRENT_TIMESTAMP]	
temp	float	
heart-rate	int unsigned	
gsr	float	
diagnosis	tinyint unsigned	

Indexes

PRIMARY id

Gambar 10. Tabel Pengukuran

Tabel pasien yang dapat dilihat melalui Gambar 11 berikut digunakan untuk menyimpan data diri pasien atau subjek penelitian yang diambil datanya. Tabel ini terdiri dari enam column diantaranya column id sebagai *primary key*, column DOB untuk menyimpan data tanggal kelahiran pasien, column name untuk menampung nama pasien, column domisili untuk menyimpan Alamat pasien, column sex untuk menyimpan gender pasien, dan column password yang digunakan untuk menampung password setiap pasien.

Select data **Show structure** Alter table New item

Column	Type	Comment
id	int unsigned <i>Auto Increment</i>	
DOB	date <i>NULL</i>	
name	varchar(80) <i>NULL</i>	
domisili	varchar(80) <i>NULL</i>	
sex	enum('P','W') <i>NULL</i>	
password	varchar(255) <i>NULL</i>	

Indexes

PRIMARY id

Gambar 11. Tabel Pasien

Tabel admin digunakan untuk menyimpan akun admin, dimana admin nanti dapat melakukan login sesuai dengan akun yang dimilikinya. Dapat dilihat melalui Gambar 12, tabel ini terdiri dari tiga column diantaranya column id sebagai *primary key*, column username untuk menampung username admin yang dilakukan saat registrasi, dan column password yang digunakan untuk menampung password admin.

Select data **Show structure** Alter table New item

Column	Type	Comment
id	int <i>Auto Increment</i>	
username	varchar(20)	
password	varchar(255) <i>NULL</i>	

Indexes

PRIMARY id

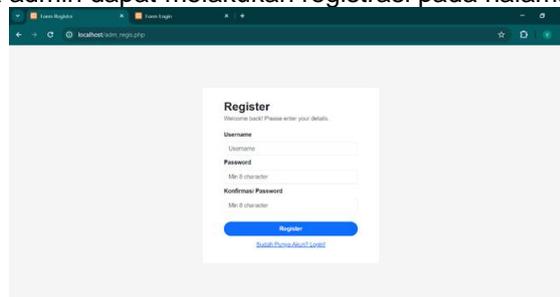
Gambar 12. Tabel Admin

3.3 Hasil Implementasi Antarmuka Website

a. Antarmuka Admin

- Register

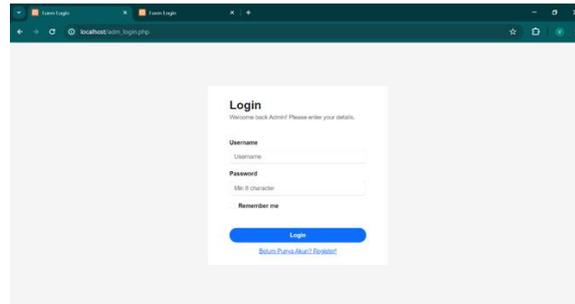
Tampilan halaman register admin dapat dilihat melalui Gambar 13 berikut. Untuk dapat masuk ke dalam sistem, admin memerlukan username dan password. Jika admin belum memiliki akun, maka admin dapat melakukan registrasi pada halaman registrasi.



Gambar 13. Halaman Register Admin

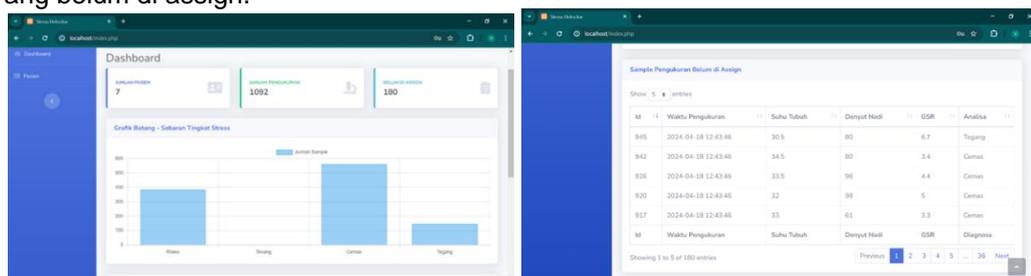
- Login

Sebelum masuk ke halaman utama website, admin harus melakukan login terlebih dahulu menggunakan username dan password. Setelah menginputkan username dan password akan dilakukan proses verifikasi sesuai dengan data admin yang ada di database. Jika admin salah menginputkan username atau password maka akan muncul pesan kesalahan bahwa username atau password salah. Tampilan halaman login admin dapat dilihat melalui Gambar 14 berikut.



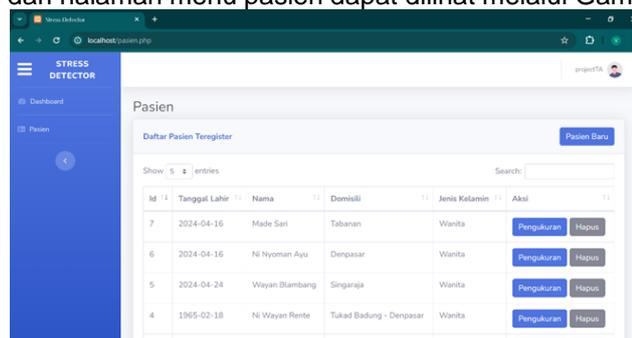
Gambar 14. Halaman Login Admin

- Menu dashboard
Seperti yang terlihat pada Gambar 15, halaman dashboard admin menampilkan beberapa informasi singkat mengenai jumlah pasien, jumlah pengukuran, jumlah data yang belum di assign, grafik sebaran tingkat stress, serta data-data pengukuran yang belum di assign.



Gambar 15. Menu Dashboard Admin

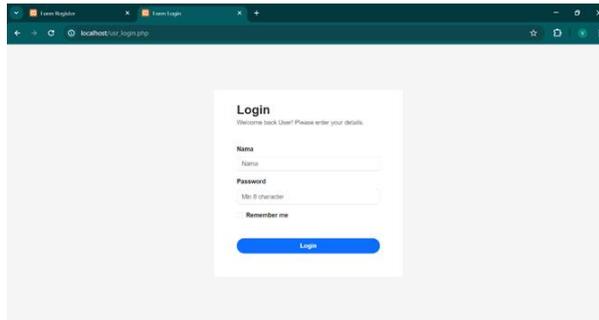
- Menu Pasien
Menu pasien menampilkan informasi data diri dari subjek penelitian yang diambil datanya. Pada halaman ini, admin dapat menambahkan data pasien baru, melakukan search data, melihat hasil pengukuran sesuai dengan data pasien, dan menghapus data pasien. Tampilan dari halaman menu pasien dapat dilihat melalui Gambar 16 berikut.



Gambar 16. Menu Pasien

b. Antarmuka User

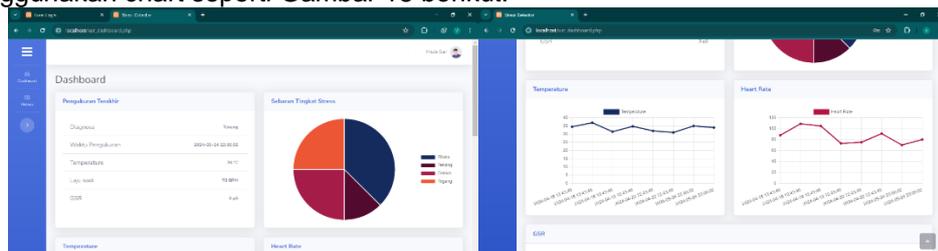
- Login
Seperti yang terlihat pada Gambar 17, untuk masuk ke halaman dashboard user akan diminta untuk login terlebih dahulu menggunakan nama serta password yang telah dibuatkan admin.



Gambar 17. Halaman Login User

- **Menu Dashboard**

Halaman dashboard user akan menampilkan informasi mengenai hasil diagnosa, perubahan nilai suhu tubuh, denyut nadi, dan GSR dari pasien dari waktu ke waktu menggunakan chart seperti Gambar 18 berikut.



Gambar 18. Menu Dashboard User

- **Menu History**

Menu history akan menampilkan sebuah tabel yang berisi informasi waktu pengukuran, nilai suhu tubuh, nilai denyut nadi, nilai GSR, dan hasil diagnosa pasien. Tampilan dari menu history dapat dilihat melalui Gambar 19 berikut.

Gambar 19. Menu History

3.3 Hasil Pengujian Diagnosa Sistem

Data yang digunakan untuk melakukan pengujian merupakan data primer dari 20 subjek penelitian dimana setiap subjek memiliki 5 data di setiap parameter. Kelima data di setiap parameter dihitung rata-ratanya untuk mendapatkan data masukan yang lebih stabil. Hasil diagnosa sistem dari 20 subjek penelitian dengan menggunakan nilai rata-rata di setiap parameter masing-masing subjek dapat dilihat melalui Tabel 4 berikut.

Tabel 4. Hasil Diagnosa Sistem

Mahasiswa	Suhu (°C)	Detak Jantung (BPM)	GSR (μS)	Hasil Diagnosa
1	35.1	71.6	2.146	Tenang
2	35.1	76.6	2.252	Tenang
3	31	102	0.242	Tegang
4	35.8	85	2.232	Tenang
5	35.95	71	1.84	Rileks
6	34.1	78	1.528	Rileks

Mahasiswa	Suhu (°C)	Detak Jantung (BPM)	GSR (μS)	Hasil Diagnosa
7	35.1	69.8	1.34	Rileks
8	33.9	68.4	2.328	Rileks
9	35.1	67.6	1.398	Rileks
10	33.1	106	0.776	Tegang
11	34.7	68.6	0.296	Rileks
12	35.4	69.2	1.834	Rileks
13	35	67.4	1.582	Rileks
14	34.5	76.2	2.266	Tenang
15	36.1	76.6	1.252	Rileks
16	34.8	67.6	1.774	Rileks
17	35.4	80.8	2.28	Tenang
18	33	93.2	2.228	Cemas
19	34.3	70.2	2.188	Tenang
20	32.4	95.2	2.49	Tenang

Hasil diagnosa sistem dibandingkan dengan hasil kuisiner DASS 42 untuk mengetahui akurasi dari diagnosa yang dihasilkan sistem. Hasil perbandingan diagnosa sistem dengan kuisiner DASS 42 dapat dilihat melalui Tabel 5 berikut.

Tabel 5. Perbandingan Diagnosa Sistem dengan Kuisiner DASS 42

Mahasiswa	Diagnosa Sistem	DASS 42	Kesimpulan
1	Tenang	Depresi sedang, cemas ringan, stres sedang	Valid
2	Tenang	Depresi sedang, cemas sedang, stres sedang	Valid
3	Tegang	Depresi sangat berat, cemas sangat berat, stres berat	Valid
4	Tenang	Depresi sedang, cemas berat, stres sedang	Valid
5	Rileks	Depresi normal, cemas sedang, stres normal	Valid
6	Rileks	Depresi sangat berat, cemas berat, stres berat	Tidak valid
7	Rileks	Depresi ringan, cemas sedang, stres ringan	Valid
8	Rileks	Depresi sedang, cemas ringan, stres normal	Valid
9	Rileks	Depresi normal, cemas normal, stres sedang	Valid
10	Tegang	Depresi sangat berat, cemas sangat berat, stres sedang	Valid
11	Rileks	Depresi normal, cemas sedang, stres ringan	Valid
12	Rileks	Depresi berat, cemas normal, stres ringan	Valid
13	Rileks	Depresi normal, cemas sedang, stres normal	Valid
14	Tenang	Depresi sedang, cemas sangat berat, stres sedang	Valid
15	Rileks	Depresi normal, cemas normal, stres normal	Valid
16	Rileks	Depresi normal, cemas sedang, stres normal	Valid
17	Tenang	Depresi sedang, cemas normal, stres sedang	Valid
18	Cemas	Depresi sangat berat, cemas sedang,	Tidak Valid

Mahasiswa	Diagnosa Sistem	DASS 42	Kesimpulan
		stres sedang	
19	Tenang	Depresi sedang, cemas ringan, stres sedang	Valid
20	Tenang	Depresi normal, cemas sangat berat, stres ringan	Tidak Valid
Total Data Valid			17
Total Data Keseluruhan			20

Berdasarkan Tabel 5, perhitungan dari akurasi metode *certainty factor* yang diterapkan dalam sistem menggunakan rumus sebagai berikut :

$$Akurasi = \frac{\text{Jumlah data valid}}{\text{Total data keseluruhan}} \times 100\%$$

$$Akurasi = \frac{17}{20} \times 100\% = 85\%$$

Berdasarkan pengujian yang dilakukan terkait dengan akurasi metode *certainty factor* yang diterapkan dalam sistem memberikan hasil akurasi sebesar 85% dimana pengujian dilakukan dengan cara membandingkan hasil diagnosa sistem dari 20 subjek penelitian dengan hasil kuisioner DASS 42 dari 20 subjek tersebut. Diantara 20 subjek penelitian terdapat 3 data yang dinyatakan tidak valid karena ketidaksesuaian antara hasil diagnosa sistem dengan hasil kuisioner DASS 42 dan terdapat 17 data yang dinyatakan valid.

4. Kesimpulan

Berdasarkan pada penelitian yang telah dilaksanakan mengenai sistem *smart healthcare* diagnosa tingkat stres menggunakan metode *certainty factor* dapat diambil kesimpulan mengenai tingkat akurasi metode *certainty factor* yang diterapkan dalam sistem dengan cara membandingkan hasil diagnosa sistem dengan hasil kuisioner DASS 42. Pengujian dilakukan terhadap 20 subjek dengan masing-masing subjek melakukan 5 kali pengambilan data di setiap parameternya. Kelima data di setiap parameter kemudian dihitung nilai rata-ratanya untuk mendapatkan nilai sensor yang lebih valid. Berdasarkan perbandingan hasil diagnosa sistem dengan hasil kuisioner DASS 42 untuk setiap subjeknya didapatkan kesimpulan bahwa tingkat akurasi yang didapatkan sebesar 85%.

References

- [1] M. Saifudin *et al.*, 'FAKTOR-FAKTOR YANG MEMPENGARUHI TINGKAT STRES AKADEMIK PADA MAHASISWA PROGRAM STUDI S1 KEPERAWATAN NON REGULER', *ARTIKEL PENELITIAN Jurnal Kesehatan*, vol. 12, no. 2, 2023, doi: 10.37048/kesehatan.v12i2.267.
- [2] A. Rachmayanti and W. Wirawan, 'Implementasi Algoritma Advanced Encryption Standard (AES) pada Jaringan Internet of Things (IoT) untuk Mendukung Smart Healthcare', *Jurnal Teknik ITS*, vol. 11, no. 3, 2022, doi: 10.12962/j23373539.v11i3.97042.
- [3] R. S. Putra and Y. Yuhandri, 'Sistem Pakar dalam Menganalisis Gangguan Jiwa Menggunakan Metode Certainty Factor', *Jurnal Sistim Informasi dan Teknologi*, vol. 3, pp. 227–232, 2021, doi: 10.37034/jsisfotek.v3i4.70.
- [4] P. Madona and F. Deza, 'Alat Pendeteksi Tingkat Stress Manusia Berdasarkan Suhu Tubuh, Kelembaban Kulit, Tekanan Darah dan Detak Jantung.', *Jurnal Elektro dan Mesin Terapan*, vol. 3, no. 2, pp. 31–42, Nov. 2017, doi: 10.35143/elementer.v3i2.194.
- [5] S. Pigi, Y. Dwi Prasetyo, and A. B. Arifa, 'IMPLEMENTASI METODE CERTAINTY FACTOR PADA SISTEM PAKAR DIAGNOSIS GANGGUAN GIZI ANAK BALITA BERBASIS MOBILE IMPLEMENTATION OF CERTAINTY FACTOR IN MOBILE-BASED EXPERT SYSTEM FOR DIAGNOSING TODDLER'S NUTRITIONAL DISORDERS', 2022.
- [6] H. R. Fajrin, Sasmeri, L. R. Prilia, B. Untara, and M. A. F. Nurkholid, 'Fuzzy logic method-based stress detector with blood pressure and body temperature parameters', *International Journal of Electrical and Computer Engineering*, vol. 14, no. 2, pp. 2156–2166, Apr. 2024, doi: 10.11591/ijece.v14i2.pp2156-2166.