



**EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

TUGAS AKHIR

ADAM VIRGIAWAN WIDIANTO

NPM 20670026

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK DAN INFORMATIKA
UNIVERSITAS PGRI SEMARANG**

2024



**EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

TUGAS AKHIR

**Diajukan kepada Fakultas Teknik dan Informatika
Universitas PGRI Semarang untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana**

ADAM VIRGIAWAN WIDIANTO

NPM 20670026

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK DAN INFORMATIKA
UNIVERSITAS PGRI SEMARANG**

2024

TUGAS AKHIR
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)

Disusun dan diajukan oleh

ADAM VIRGLAWAN WIDIANTO

NPM 20670026

Telah disetujui oleh pembimbing untuk dilanjutkan untuk disusun menjadi
TUGAS AKHIR pada tahun 2024

Semarang, 9 Agustus 2024

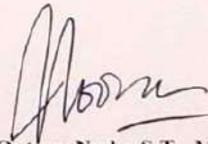
Pembimbing Utama,



Aris Tri Joko Harjanto, S.Kom., M.Kom.

NIDN. 0619048202

Pembimbing Pendamping,



Noora Qotrun Nada, S.T., M.Eng.

NPP. 158201485

TUGAS AKHIR

**EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

Disusun dan diajukan oleh

ADAM VIRGIAWAN WIDIANTO

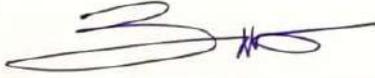
NPM 20670026

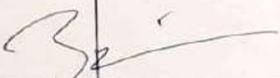
Telah dipertahankan di depan Dewan Penguji

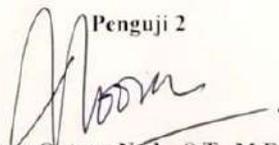
Pada Tanggal 16 Agustus 2024 dan dinyatakan telah Memenuhi syarat

Dewan Penguji


Ketua
Abdi Toto Husodo, S.T., M.T.
NPP. 136901387

Sekretaris

Bambang Agus H., S.Kom., M.Kom
NIDN. 0601088201

Penguji 1

Aris Tri Joko Harjanto, S.Kom., M.Kom
NIDN. 0619048202

Penguji 2

Noora Qotrun Nada, S.T., M.Eng.
NPP. 158201485

Penguji 3


Bambang Agus H., S.Kom., M.Kom
NIDN. 0601088201

MOTTO DAN PERSEMBAHAN

Moto :

“Yakinlah ada sesuatu yang menantimu selepas banyak kesabaran yang kau jalani, yang akan membuatmu Terpana hingga kau lupa betapa pedihnya rasa sakit.” –

Ali bin Abi Thalib

Persembahan:

Alhamdulillah segala puji bagi Allah SWT atas segala penyusunan skripsi ini.

Kuprsembahkan skripsi ini untuk :

1. Kepada orang tuaku tercinta yang selalu menemani, merawat, membimbing, serta melindungiku sejak aku lahir hingga saat ini, kasih sayang yang senantiasa diberikan serta doa yang selalu menyertainya memberikan dukungan sepenuh hati dengan ikhlas dan tanpa pamrih yaitu Bapak Noerhadi dan Ibu Sriwigati
2. Seluruh dosen Program Studi Informatika Universitas PGRI Semarang yang senantiasa membimbing dan mengarahkan saya untuk menyelesaikan tugas akhir ini.
3. Almamaterku tercinta Universitas PGRI Semarang.

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini :

Nama : Adam Virgiawan Widianto
NPM : 20670026
Program Studi : Informatika
Fakultas : Teknik dan Informatika

Menyatakan dengan sebenarnya bahwa tugas akhir yang saya buat ini benar-benar merupakan hasil karya saya sendiri, bukan plagiarisme.

Apabila pada kemudian hari tugas akhir ini terbukti hasil plagiarisme, saya bersedia menerima sanksi atas perbuatan tersebut.

Semarang, 16 Agustus 2024

Yang membuat pernyataan



Adam Virgiawan Widianto

NPM 20670026

ABSTRAK

Stunting merupakan masalah gizi yang serius, di mana pertumbuhan anak tidak sesuai dengan usianya akibat kekurangan gizi kronis. Berdasarkan data Survei Status Gizi Balita Indonesia (SSGBI), prevalensi stunting di Jawa Tengah mengalami penurunan dari 27,68% pada 2019 menjadi 20,9% pada 2021. Namun, masalah ini masih signifikan, terutama di daerah seperti Desa Tamanrejo, Kecamatan Limbangan, Kabupaten Kendal, yang memiliki kasus stunting yang tinggi. Posyandu rutin yang dilaksanakan di desa ini berfungsi untuk memantau pertumbuhan anak dengan pengukuran berat badan dan tinggi badan, namun pencatatan manual yang dilakukan berisiko terhadap ketidaktepatan data dan memakan waktu lama. Penelitian ini bertujuan untuk mengembangkan sistem Early Warning Stunting berbasis machine learning dengan menggunakan algoritma Support Vector Machine (SVM) untuk meningkatkan efisiensi dan akurasi dalam identifikasi dan penanganan kasus stunting. Metode waterfall diterapkan dalam pembangunan sistem ini, meliputi tahap requirement, design, implementation, verification, dan maintenance. Penelitian ini menggunakan teknik evaluasi seperti black box, white box, dan User Acceptance Testing (UAT) untuk menguji sistem. Hasil penelitian menunjukkan bahwa model SVM mencapai akurasi 96% dalam klasifikasi stunting, dengan nilai precision, recall, dan F1-score rata-rata masing-masing sebesar 96%. Pengujian black box menunjukkan tingkat keberhasilan 100%, sedangkan pengujian white box memperoleh hasil 100% pada path testing. UAT menunjukkan tingkat kepuasan pengguna sebesar 94,4% dengan kategori sangat baik. Temuan ini menegaskan potensi sistem ini sebagai alat efektif untuk deteksi dini kasus stunting dan memberikan dasar untuk intervensi yang lebih cepat dan tepat oleh tenaga medis.

Kata Kunci: Stunting, Survei Status Gizi Balita Indonesia (SSGBI), *waterfall*, *Support Vector Machines*, *Machine Learning*

PRAKATA

Puji syukur atas kehadiran Allah SWT karena atas limpahan rahmat-Nya, penulis dapat menyusun dan menyelesaikan Tugas Akhir ini dengan lancar. tugas akhir yang berjudul “Early Warning Stunting Dengan Algoritma Support Vector Machine (SVM)” ini disusun untuk memenuhi syarat memperoleh gelar Sarjana Komputer.

Penyusunan Tugas Akhir ini tidak lepas dari berbagai hambatan dan rintangan serta kesulitan-kesulitan. Namun, berkat bimbingan, bantuan, nasehat dan dorongan serta saran-saran dari berbagai pihak, khususnya Pembimbing, segala hambatan dan rintangan serta kesulitan tersebut dapat teratasi dengan baik. Oleh karena itu, dalam kesempatan ini dengan tulus hati penulis menyampaikan terima kasih kepada:

1. Ibu Dr Sri Suciati, M.Hum selaku Rektor Universitas PGRI Semarang yang telah memberi kesempatan kepada penulis untuk menimba ilmu di Universitas PGRI Semarang.
2. Bapak Ibnu Toto Husodo, S.T., M.T. selaku Dekan Fakultas Teknik dan Informatika Universitas PGRI Semarang yang telah memberikan ijin kepada penulis untuk melakukan penelitian.
3. Bapak Bambang Agus Herlambang, S. Kom., M. Kom. selaku Ketua Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang.
4. Bapak Aris Tri Jaka Harjanta, S.Kom, M.Kom selaku Pembimbing I Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang yang telah memberikan penuh kesabaran dalam Menyusun tugas akhir ini hingga selesai.
5. Ibu Noora Qotrun Nada, S.T., M.Eng. selaku Pembimbing II Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang yang telah memberikan penuh kesabaran dalam Menyusun tugas akhir ini hingga selesai.

6. Bapak dan Ibu Dosen Program Studi Informatika yang telah memberikan bekal ilmu kepada penulis, selama belajar di Universitas PGRI Semarang.
7. Kepala Desa Tamanrejo Bapak Mustofa, SE. yang telah memberikan fasilitas untuk melakukan penelitian di posyandu Desa Tamanrejo
8. Kedua orang tua tercinta Ayahhanda Noerhadi dan Ibunda Sriwigati, keduanya yang telah membuat segalanya menjadi mungkin sehingga saya bisa menyelesaikan tugas akhir ini hingga selesai. Terima kasih untuk segala pengorbanan yang telah dilakukan kepadaku selama ini. Aku sangatlah bersyukur dengan keberadaan dan disituasi apapun aku sangat bangga dengan kedua orang tuaku.
9. Kedua Adikku Anisa Devanti Winuristika dan juga Sultan Maulana AlFahri yang selalu menemani selama proses pengerjaan tugas akhir ini.
10. Calon Istri Sea Duta Alfa Harcinlani yang telah menemani, meluangkan waktu dan juga memberikan semangat terus untuk maju dan berproses tanpa kenal lelah dalam memberikan dukungan, semangat serta doa.
11. Teruntuk Rizal Ashari dan Naufal Agung Rizkina yang telah membantu penulis dalam bentuk support mental dan bertukar pikiran dalam penulisan Tugas Akhir ini.
12. Serta teman-teman dekatku yang memberikan semangat dan juga mengingatkan untuk terus optimis dalam mengerjakan Tugas Akhir ini.

Akhir kata, penulis berharap laporan tugas akhir ini dapat bermanfaat bagi pembaca dan kita semua yang membutuhkan.

Semarang, 16 Agustus 2024

Penulis

DAFTAR ISI

SAMPUL DALAM	ii
LEMBAR PENGESAHAN	Error! Bookmark not defined.
LEMBAR PERSETUJUAN.....	Error! Bookmark not defined.
MOTTO DAN PERSEMBAHAN	v
PERNYATAAN KEASLIAN TULISAN.....	Error! Bookmark not defined.
ABSTRAK	vii
PRAKATA.....	viii
DAFTAR ISI	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xiv
BAB I	1
PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	3
C. Pembatasan Masalah	3
D. Rumusan Masalah	3
E. Tujuan Penelitian.....	4
F. Manfaat Penelitian	4
G. Penegasan Istilah.....	4
BAB II.....	5
KAJIAN PUSTAKA	5
A. Tinjauan Pustaka	5
B. Landasan Teori	9
C. Kerangka Berfikir.....	32
BAB III	33
METODE PENELITIAN	33
A. Pendekatan Penelitian	33
B. Lokasi Penelitian.....	33
C. Variable Penelitian	35

D. Proses Experimen.....	36
E. Prosedur Penelitian.....	36
F. Teknik Pengumpulan Data	37
G. Teknik Analisis Data	37
BAB IV	41
HASIL DAN PEMBAHASAN.....	41
A. Hasil	41
B. Pembahasan.....	81
BAB V.....	84
KESIMPULAN DAN SARAN.....	84
A. Kesimpulan	84
B. Saran.....	85
DAFTAR PUSTAKA	86

DAFTAR TABEL

Tabel 2. 1 Tinjauan Pustaka	6
Tabel 2. 2 Simbol <i>Class Diagram</i>	21
Tabel 2. 3 Simbol <i>Use Case Diagram</i>	23
Tabel 2. 4 Notasi <i>Activity Diagram</i>	24
Tabel 2. 5 Simbol <i>Sequence Diagram</i>	25
Tabel 4. 1 <i>Requirements</i> Pembuatan Streamlit	62
Tabel 4. 2 Pengujian <i>Black Box</i>	67
Tabel 4. 3 Hasil Pengujian <i>Black Box</i>	69
Tabel 4. 4 Pengujian <i>White Box</i>	74
Tabel 4. 5 <i>Value Test</i>	77
Tabel 4. 6 Pengujian <i>User Acceptance Testing</i>	78
Tabel 4. 7 Bobot Penilaian Kuesioner	80
Tabel 4. 8 Hasil Pengujian UAT	80

DAFTAR GAMBAR

Gambar 2. 1 Grafik Pertumbuhan	11
Gambar 2. 2 Grafik z-score laki-laki.....	12
Gambar 2. 3 Grafik z-score perempuan	12
Gambar 2. 4 <i>hyperplane</i>	13
Gambar 2. 5 Flow Graph Nation.....	28
Gambar 2. 6 Kerangka Berfikir.....	32
Gambar 3. 1 Peta Desa Tamanrejo	34
Gambar 4. 1 <i>Flowchart</i> SVM.....	44
Gambar 4. 2 Hasil <i>Read Data</i>	45
Gambar 4. 3 Hasil Visualisasi	46
Gambar 4. 4 Kode <i>Encoding Categorical Variable</i>	46
Gambar 4. 5 Kode <i>Future Selection</i>	47
Gambar 4. 6 Kode <i>Train-Test Split</i>	48
Gambar 4. 7 Kode <i>Handle Imbalance</i>	49
Gambar 4. 8 Kode <i>Feature Scaling</i>	50
Gambar 4. 9 Hasil Perhitungan	52
Gambar 4. 10 <i>Confusion Matrix</i>	52
Gambar 4. 11 Hasil Perhitungan	54
Gambar 4. 12 Penerapan Linier Kernel.....	54
Gambar 4. 13 <i>Use Case Diagram</i>	56
Gambar 4. 14 <i>Activity Diagram</i>	57
Gambar 4. 15 <i>Sequence Diagram</i>	58
Gambar 4. 16 Rancangan Halaman Pengenalan Stunting.....	59
Gambar 4. 17 Rancangan Halaman Deteksi Stunting.....	60
Gambar 4. 18 Rancangan Halaman Keputusan Menteri.....	61
Gambar 4. 19 Kode Prediksi Data Stunting Pada Streamlit.....	63
Gambar 4. 20 Tampilan Halaman Tentang Stunting	64
Gambar 4. 21 Tampilan Halaman Deteksi Stunting.....	65
Gambar 4. 22 Tampilan Halaman Keputusan Menteri	66
Gambar 4. 23 Diagram alir.....	76

DAFTAR LAMPIRAN

Lampiran 1 Pengujian Black Box	90
Lampiran 2 Lembar Pengujian UAT	105
Lampiran 3 Lembar Pembimbingan skripsi	115
Lampiran 4 Lembar Permohonan Ijin Penelitian	118
Lampiran 5 Lembar Validasi Ahli	119
Lampiran 6 Dataset	122

BAB I

PENDAHULUAN

A. Latar Belakang

Permasalahan yang seringkali terjadi pada anak dengan pertumbuhan yang tidak sesuai usianya merupakan kekurangan gizi kronis atau seringkali disebut stunting. Berdasarkan data Survei Status Gizi Balita Indonesia (SSGBI) tahun 2019, prevalensi stunting Jawa Tengah masih di angka 27,68 persen. Pada 2021 turun menjadi 20,9 persen. Status gizi anak ditentukan berdasarkan standar antropometri indikator yang paling umum digunakan untuk menghitung angka stunting adalah indeks tinggi badan pada umur (TB/U), berat badan pada umur (BB/U), dan berat badan pada tinggi badan (BB/TB)[1]. Permasalahan anak stunting juga banyak terjadi di Desa Tamanrejo Kecamatan Limbangan yang ada di Kabupaten Kendal, dari tingginya kasus stunting ini membuat Bidan Desa dan juga Puskesmas menggelar kegiatan Posyandu rutin yang dilaksanakan 4 kali di setiap bulannya[2]. Posyandu yang diselenggarakan ini merupakan bentuk pemantauan terhadap pertumbuhan anak dengan cara melakukan pengukuran berat badan dan tinggi badan anak. Dari kegiatan Posyandu, Kader Posyandu akan memperoleh hasil laporan stunting dari pencatatan manual yang kemudian akan dilakukan penginputan data menggunakan Microsoft Excel. Pengidentifikasian dan penginputan data secara manual beresiko mengalami ketidaktepatan data serta status gizi yang menjadi indikator penentuan stunting yang beranekaragam dapat menghabiskan waktu lama. Dengan adanya kegiatan riset yang dilakukan menggunakan metode klasifikasi pada data stunting dengan penerapan data mining ini diharapkan dapat membantu proses pengolahan data secara tepat dan efisien.

Ketidaklengkapan atau ketidakakuratan informasi dalam evaluasi stunting dapat memperlambat identifikasi kasus-kasus stunting yang memerlukan perhatian khusus dan intervensi segera. Permasalahan ini

membutuhkan penanganan dalam mengklasifikasi untuk menangani kasus-kasus stunting secara lebih efisien. Diperlukan pendekatan yang lebih canggih dalam pengumpulan dan analisis data, serta perbaikan dalam infrastruktur pendataan dan penilaian stunting. Dalam menangani masalah evaluasi stunting pada anak-anak, diperlukan penerapan teknik evaluasi yang lebih inovatif dengan menggunakan *machine learning* (ML) untuk memprediksi kemungkinan anak teridentifikasi mengalami stunting, sehingga langkah-langkah preventif dapat dilakukan secara lebih tepat dan proaktif. Penerapan ML dalam klasifikasi melibatkan sejumlah algoritma yang dapat digunakan. Untuk mengembangkan model klasifikasi yang optimal, diperlukan eksperimen perbandingan beberapa algoritma menggunakan dataset yang spesifik agar mendapatkan gambaran yang komprehensif terkait performa algoritma klasifikasi dalam *early warning* kasus stunting dengan tingkat akurasi dan responsifitas yang lebih baik[3].

Dalam penelitian ini, *Support Vector Machine* (SVM) adalah Teknik yang digunakan di bidang *Machine Learning* (ML), algoritma klasifikasi termasuk SVM. Sebagai subbidang kecerdasan buatan, pembelajaran ML memungkinkan sistem untuk belajar secara otomatis tanpa pemrograman yang membosankan. Hasil klasifikasi dari SVM bisa digunakan sebagai *early warning* stunting, pengolahan data yang banyak menggunakan ML disebut juga data mining. *Data mining* adalah proses menganalisis sebuah kasus untuk mengetahui algoritma mana yang bekerja paling baik. Kumpulan data besar dapat ditambang menggunakan Teknik penambangan data untuk mengungkap pola dan informasi[4].

Machine learning atau dikenal dengan pembelajaran mesin adalah ilmu komputer yang bisa bekerja tanpa diprogram secara eksplisit. Banyak peneliti berpikir bagaimana cara untuk membuat kemajuan menuju AI terhadap tingkat manusia. *Machine learning* ini merupakan kecerdasan buatan yang mempelajari bagaimana membuat data. *Machine learning* ini biasa disingkat

dengan ML. Ini dibutuhkan untuk menerapkan teknik yang cepat dan kuat dalam menemukan masalah baru.

Secara definisi, *machine learning* merupakan ilmu atau studi yang mempelajari tentang algoritma dan model statistik yang digunakan oleh sistem komputer untuk melakukan task tertentu tanpa instruksi eksplisit. *Machine learning* bergantung pada pola dan kesimpulan. Untuk mendapatkan pola dan kesimpulan tersebut, algoritma *machine learning* menghasilkan model matematika yang didasari dari data sampel yang sering disebut dengan *training data*[5].

B. Identifikasi Masalah

Permasalahan yang dapat diidentifikasi dari latar belakang di atas adalah belum adanya sistem yang bisa melakukan *early warning* pada data stunting yang dapat membantu proses secara tepat dan efisien

C. Pembatasan Masalah

Mengingat terbatasnya tenaga dan waktu yang ada, serta menjaga dan menghindari pembahasan masalah yang bisa menjadi terlalu luas, maka penulis membatasi permasalahan pada Sistem *Early Warning* Stunting, yang meliputi:

1. Sistem ini dibuat untuk menghasilkan sebuah sistem yang menangani proses pendataan manual dan juga sebuah sistem yang bisa melakukan prediksi kasus stunting secara tepat dan efisien
2. Metode Klasifikasi pada Algoritma *Support Vector Machine* (SVM) hanya akan dilakukan sampai pada tahap pengujian.

D. Rumusan Masalah

Berdasarkan latar belakang penelitian yang berjudul “Early Warning Stunting Dengan Algoritma Support Vector Machine (SVM)” tersebut, maka dapat dirumuskan permasalahannya, yaitu Bagaimana membangun sebuah sistem yang dapat melakukan *early warning* kasus stunting secara tepat dan efisien?

E. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membangun sebuah sistem yang dapat melakukan melakukan *early warning* stunting secara tepat dan efektif berdasarkan dengan pertumbuhan dari balita

F. Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat bagi semua pihak yang terkait dalam proses yang dilaksanakan, sebagai berikut :

1. Bagi Mahasiswa

Untuk melatih serta mengukur seberapa daya tangkap peneliti dalam mempraktekan ilmu yang diperoleh di bangku kuliah dan menerapkannya dalam lingkungan kehidupan yang membutuhkan.

2. Bagi Akademik

Sebagai tolak ukur keberhasilan proses belajar mengajar yang dapat digunakan sebagai bahan evaluasi bagi pihak akademik dan sebagai referensi bagi mahasiswa dalam penelitian lebih lanjut yang berkaitan dengan studi yang dibahas.

3. Bagi Masyarakat

Diharapkan dengan adanya sistem ini penanganan stunting dapat dilakukan lebih cepat dan langsung dapat diketahui apakah balita terdeteksi mengalami gejala stunting secara tepat dan efisien.

G. Penegasan Istilah

1. *Black Box* memiliki tujuan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang[6].
2. *White Box* merupakan metode desain uji kasus yang digunakan untuk mengontrol struktur dari desain procedural produk.
3. UAT merupakan jenis pengujian yang akan diserahkan pada pengguna untuk mengetahui apakah perangkat lunak sudah memenuhi harapan dan bekerja seperti yang diharapkan.

BAB II KAJIAN PUSTAKA

A. Tinjauan Pustaka

Dalam penelitian ini, penulis menggali informasi dari penelitian-penelitian sebelumnya sebagai bahan perbandingan, baik mengenai kekurangan atau kelebihan yang sudah ada. Selain itu, penulis juga menggali informasi yang ada sebelumnya tentang teori yang berkaitan dengan judul yang digunakan untuk memperoleh landasan teori ilmiah. Pada penelitian sebelumnya oleh Putri Apriyani, Arif Rinaldi Dikananda, Irfan Ali (2023) yang berjudul “Penerapan Algoritma K-Means dalam Klasterisasi Kasus Stunting Balita Desa Tegalwangi”. Pada aplikasi ini Putri Apriyani, Arif Rinaldi Dikananda, Irfan Ali, merancang sebuah sistem dengan menggunakan *rapidminer* dengan menerapkan metode algoritma *K-Means* dalam klasterisasi. Persamaan dengan judul yang dibuat oleh penulis adalah sama-sama membahas mengenai sistem yang digunakan untuk melakukan penerapan penanggulangan kasus Stunting[7].

Pada penelitian selanjutnya oleh Nur Fitriyani Sahamony, Terttiaavini, Harsih Rianto (2024) yang berjudul “Analisis Perbandingan Algoritma Machine Learning untuk Prediksi Stunting pada Anak”. Pada aplikasi ini Nur Fitriyani Sahamony, Terttiaavini, Harsih Rianto menganalisa dan merancang sebuah perbandingan kinerja model *mechine learning* menggunakan beberapa model *mechine learning* yaitu *Random forest*, *Logistic Regression*, *Naïve Bayes*, *SVM*, dan *Neural Network*. Persamaan dengan judul yang dibuat penulis adalah, sama-sama melibatkan model SVM sebagai pemodelan algoritma yang dipakai[3].

Kemudian pada penelitian selanjutnya oleh Amanda Iksanul Putri, Yulia Syarif, Puguh Jayadi, Fadlan Arrazak, Febi Nur Salisah (2023) yang berjudul “Implementasi Algoritma Decision Tree dan Support Vector Machine (SVM)

untuk Prediksi Risiko Stunting pada Keluarga”. Pada perancangan aplikasi tersebut, Amanda Iksanul Putri, Yulia Syarif, Puguh Jayadi, Fadlan Arrazak, Febi Nur Salisah membuat sebuah sistem prediksi stunting menggunakan platform *ripedminer* dengan metode Klasifikasi *Support Vector Machine* (SVM) dan Klasifikasi *Decision Tree*. Persamaan dengan judul penulis yaitu alur sistem yang dibuat menggunakan metode Klasifikasi *Support Vector Machine* yang dikembangkan dan menjadi sebuah sistem yang dapat bermanfaat[4].

Pada penelitian berikutnya oleh Raden Harya Dananjaya, Sutrisno, dan Fany A Wellianto (2022) yang berjudul “Akurasi Penggunaan Metode Support Vector Machine Dalam Prediksi Penurunan Pondasi Tiang”. Pada penelitian tersebut Raden Harya Dananjaya, Sutrisno, dan Fany A Wellianto menganalisa prediksi dari penurunan pondasi tiang dengan aplikasi Jupyter notebook dengan Bahasa python menggunakan metode akurasi *Support Vector Machine*. Persamaan dengan judul penulis yaitu sama-sama menggunakan metode akurasi *Support Vector Machine* yang bertujuan untuk menghitung tingkat akurasi sebuah model yang diuji[8].

Melihat dari penelitian sebelumnya, maka penulis akan melakukan penelitian tentang Early Warning Stunting Dengan Algoritma Support Vector Machine (SVM) Study Kasus Desa Tamanrejo. Perbandingan dengan penelitian sebelumnya yaitu, studi kasus yang dilakukan berkaitan dengan klasifikasi kasus terjadinya stunting di Desa Tamanrejo Tabel 2.1

Tabel 2.1 Tinjauan Pustaka

No	Peneliti	Judul	Metode	Hasil
1	Putri Apriyani, Arif Rinaldi Dikananda, Irfan Ali	Penerapan Algoritma K- Means dalam Klasterisasi Kasus	Algoritma K- Means	Cluster model .Hasil DBI(Davies- bouldin

No	Peneliti	Judul	Metode	Hasil
		Stunting Balita Desa Tegalwangi		index) yang terbaik dengan nilai $K=2$ di dapat sebesar 0.007 mendekati 0 yang artinya semakin kecil nilai DBI yang didapat (non-negatif ≥ 0),Semakin baik clusternya
2	Nur Fitriyani Sahamony, Terttiaavini, Harsih Rianto	Analisis Perbandingan Algoritma Machine Learning untuk Prediksi Stunting pada Anak	Random forest, Logistic Regression, Naïve Bayes, SVM, dan Neural Network	Hasil uji menunjukkan bahwa Naïve Bayes memiliki performa tertinggi dengan akurasi = 98,57%, F1-Score = 0,99, serta recall dan precision yang sangat

No	Peneliti	Judul	Metode	Hasil
				tinggi. Random Forest juga memberikan hasil baik dengan akurasi = 98,29%, namun Naïve Bayes diidentifikasi sebagai model terbaik.
3	Amanda Iksanul Putri, Yulia Syarif, Puguh Jayadi, Fadlan Arrazak, Febi Nur Salisah	Implementasi Algoritma Decision Tree dan Support Vector Machine (SVM) untuk Prediksi Risiko Stunting pada Keluarga	Klasifikasi Support Vector Machine (SVM) dan Klasifikasi Decision Tree	Algoritma Decision Tree didapatkan akurasi sebesar 96.15% sedangkan algoritma SVM mendapatkan akurasi 62.48%

No	Peneliti	Judul	Metode	Hasil
4	Raden Harya Dananjaya, Sutrisno, dan Fany A Wellianto	Akurasi Penggunaan Metode Support Vector Machine Dalam Prediksi Penurunan Pondasi Tiang	Support Vector Machine (SVM)	Tingkat Akurasi Proporsi data training sebesar 90%

B. Landasan Teori

Landasan teori pada penelitian adalah mencari teori-teori, konsep-konsep dan dari hasil penelitian yang dapat dijadikan sebagai landasan teoritis untuk pelaksanaan penelitian. Landasan teori sangat perlu dikarenakan agar penelitian yang dilakukan mempunyai dasar yang kokoh, dan bukan sekedar perbuatan coba-coba.

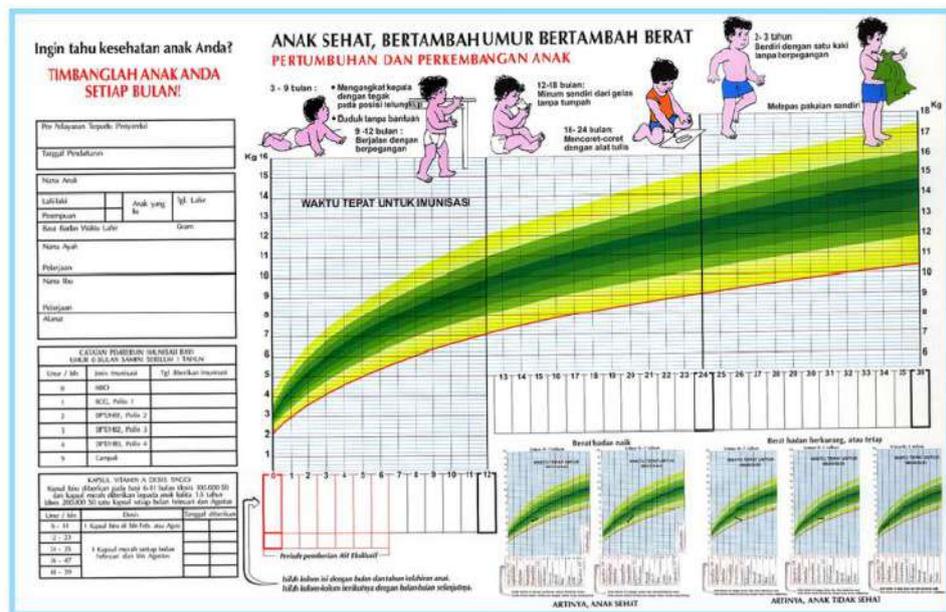
1. Stunting

Menurut WHO (2015), stunting adalah gangguan pertumbuhan dan perkembangan anak akibat kekurangan gizi kronis dan infeksi berulang, yang ditandai dengan panjang atau tinggi badannya berada di bawah standar. Selanjutnya menurut WHO (2020) stunting adalah pendek atau sangat pendek berdasarkan panjang / tinggi badan menurut usia yang kurang dari -2 standar deviasi (SD) pada kurva pertumbuhan WHO yang terjadi dikarenakan kondisi irreversibel akibat asupan nutrisi yang tidak adekuat dan/atau infeksi berulang / kronis yang terjadi dalam 1000 HPK[9].

Sesuai dengan keputusan menteri kesehatan republik indonesia nomor hk.01.07/menkes/1928/2022 tentang pedoman nasional pelayanan kedokteran tata laksana stunting, stunting selalu diawali oleh perlambatan penambahan berat badan (weight faltering) yang dapat

terjadi sejak in utero dan berlanjut setelah lahir. Penelitian di Malawi menunjukkan bahwa bayi-bayi yang dilahirkan lebih pendek akan terus mengalami length faltering selama masa bayi (infancy). Faktor prediktor paling kuat untuk terjadinya stunting di usia 12 bulan pada penelitian tersebut adalah perlambatan pertumbuhan yang terjadi dalam tiga bulan pertama kehidupan. Jika rerata BB/U pada penimbangan selama 3 bulan pertama sejak lahir berada kurang dari <-1 SD maka risiko untuk mengalami stunting di usia 12 bulan adalah 14 kali lipat. Anak stunting berisiko mengalami peningkatan morbiditas dan mortalitas, penurunan kekebalan sistem imun dan peningkatan risiko infeksi. Efek jangka panjang menyebabkan kegagalan seorang anak mencapai potensi kognitif dan kemampuan fisiknya, sehingga akan memengaruhi kapasitas kerja dan status sosial ekonomi di masa depan. Selain itu, pada anak stunting akan terjadi penurunan oksidasi lemak sehingga rentan mengalami akumulasi lemak sentral dan resistensi insulin. Hal ini menyebabkan risiko lebih tinggi untuk mengalami penyakit-penyakit degeneratif seperti diabetes, hipertensi, dislipidemia, serta fungsi reproduksi yang terganggu pada masa dewasa. Tingginya beban masalah stunting di Indonesia, karena prevalensi yang masih tinggi dan risiko dampak jangka panjang yang dapat memengaruhi kualitas sumber daya manusia Indonesia, menjadi latar belakang sangat diperlukannya suatu Pedoman Nasional Pelayanan Kesehatan (PNPK) untuk pencegahan, deteksi dini dan tata laksana segera bayi dan balita stunting di Indonesia. Stunting ditegakkan berdasarkan anamnesis, pemeriksaan fisik dan pengukuran antropometrik. Harus dicari adanya faktor-faktor risiko dan red flags pada stunting. Pemeriksaan laboratorium dan penunjang lain dilakukan sesuai indikasi menurut hasil pemeriksaan awal tersebut. Perlu untuk membedakan antara pendek yang merupakan varian normal dengan kondisi patologis. Pendek yang bersifat patologis dikategorikan menjadi proporsional akibat faktor pranatal atau pascanatal, dan disproporsional akibat kelainan genetik. Selain itu, bila ditemukan

pendek dengan status gizi baik atau gizi lebih diperlukan pemeriksaan penunjang untuk menyingkirkan diagnosis stunting, dan dilakukan evaluasi terhadap potensi tinggi genetik berdasarkan tinggi badan kedua orang tua, riwayat pranatal dan pascanatal, perlambatan pertumbuhan (weight faltering) dan malnutrisi sebelumnya.

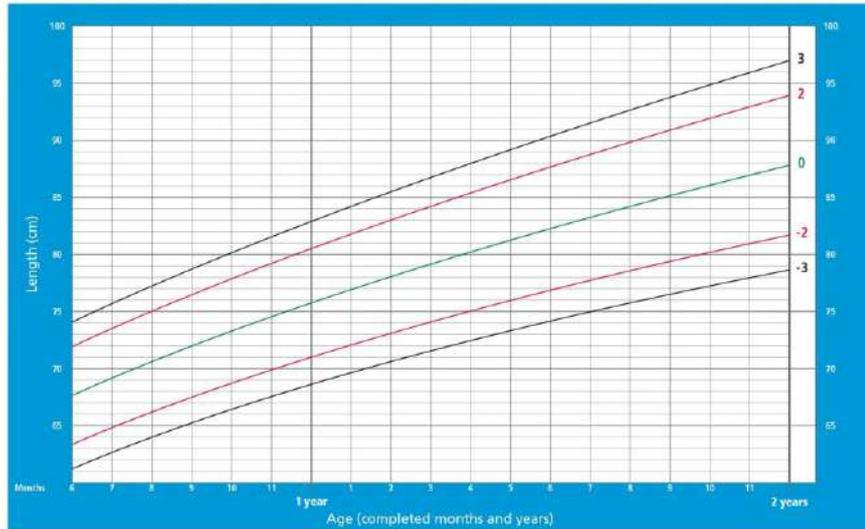


Gambar 2.1 Gambar grafik pertumbuhan

Pada Gambar 2.1 tersebut menunjukkan sebuah grafik dari Kartu Menuju Sehat atau KMS kartu tersebut dapat menjadi gambaran apakah batita mengalami stunting atau normal.

Length-for-age BOYS

6 months to 2 years (z-scores)

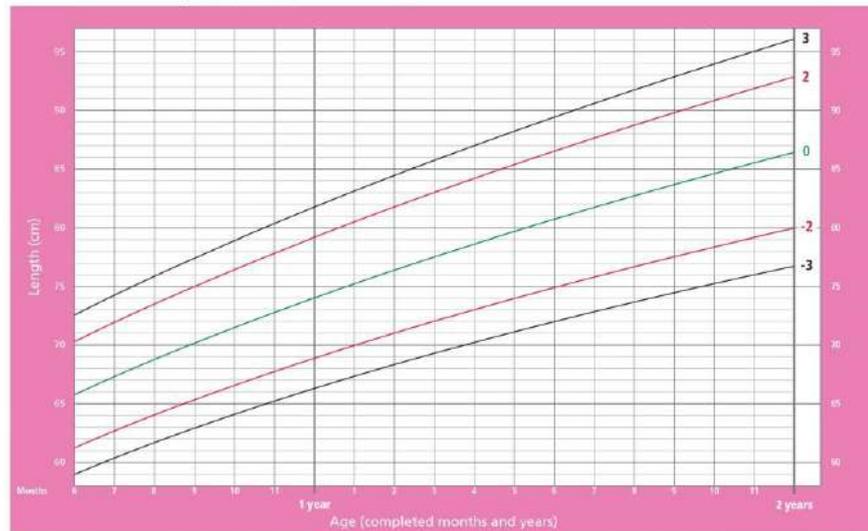


WHO Child Growth Standards

Gambar 2.2 Gambar Grafik z-score laki-laki

Length-for-age GIRLS

6 months to 2 years (z-scores)



WHO Child Growth Standards

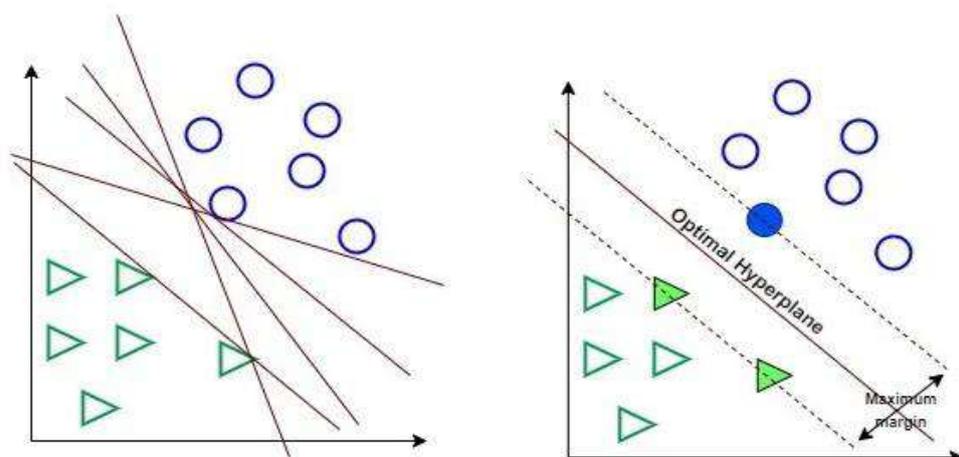
Gambar 2.3 Gambar Grafik z-score Perempuan

Pada Gambar 2.2 dan 2.3 tersebut menunjukkan sebuah grafik dari WHO atau World Health Organization yang berdasarkan dari z-score atau yang artinya ukuran statistik yang menunjukkan penyimpangan suatu titik data dari rata-rata dalam satuan standar deviasinya, pada grafik tersebut bisa disebut stunting kalau plotting di grafik tinggi badan < -2 standar deviasi.

2. Algoritma Support Vector Machine (SVM)

Support Vector Machine adalah model ML multifungsi yang dapat digunakan untuk menyelesaikan permasalahan klasifikasi, regresi, dan pendeteksian outlier. Termasuk ke dalam kategori *supervised learning*.

SVM adalah salah satu metode yang paling populer dalam *machine learning*. Siapa pun yang tertarik untuk masuk ke dalam dunia ML, perlu mengetahui SVM. Tujuan dari algoritma SVM adalah untuk menemukan *hyperplane* terbaik dalam ruang berdimensi-N (ruang dengan N-jumlah fitur) yang berfungsi sebagai pemisah yang jelas bagi titik-titik data input.



Gambar 2.4 *hyperplane*

Gambar 2.4, menunjukkan beberapa kemungkinan bidang (*hyperplane*) untuk memisahkan data lingkaran dan data kotak. Algoritma SVM kemudian mencari *hyperplane* terbaik yang dapat memisahkan kedua kelas secara optimal. Seperti tampak pada gambar di atas, sebuah *hyperplane* optimal berhasil dibuat dan mampu memisahkan kedua kelas sehingga memiliki *margin* yang maksimal.

Beberapa keunggulan *Support Vector Machine* antara lain:

1. SVM efektif pada data berdimensi tinggi (data dengan jumlah fitur atau atribut yang sangat banyak).
2. SVM efektif pada kasus di mana jumlah fitur pada data lebih besar dari jumlah sampel.
3. SVM menggunakan subset poin pelatihan dalam fungsi keputusan (disebut *support vector*) sehingga membuat penggunaan memori menjadi lebih efisien.

Konsep dasar SVM adalah mencari *hyperplane* dengan margin terbesar. *Hyperplane* ini adalah suatu d -dimensi (d adalah fitur dalam data) yang membagi ruang fitur menjadi dua bagian, mewakili dua kelas berbeda. Margin adalah jarak terdekat antara *hyperplane* dan *support vectors*. Usaha dalam mencari lokasi *hyperplane* ini merupakan proses dari pembelajaran pada SVM. Data dinotasikan sebagai $x_i \in R$ sedangkan label masing-masing dinotasikan $y_i \in \{1,0\}$ untuk $i = 1, 2, \dots, I$ yang mana I adalah banyak data. Asumsi 1 dan 0 dapat terpisah secara sempurna oleh *hyperplane* berdimensi d yang didefinisikan pada persamaan (1).

$$w^{\vec{}} \cdot x + b = 0 \quad (1)$$

Keterangan :

w : Vector bobot.

x ; Vektor Fitur.

b : Bias atau intercept

Pola \vec{x}_i yang termasuk kelas 1 dapat dirumuskan sebagai pola yang memenuhi pertidaksamaan (2).

$$\vec{w} \cdot \vec{x}_i + b \geq 1 \quad (2)$$

Keterangan :

w : Vektor bobot.

x_i : Vektor fitur dari contoh data.

b : Bias atau intercept

Sedangkan pola \vec{x}_i yang termasuk kelas 0 dirumuskan dengan pertidaksamaan (3).

$$\vec{w} \cdot \vec{x}_i + b \leq -1 \quad (3)$$

Keterangan :

w : Vektor bobot.

x_i : Vektor fitur dari contoh data.

b : Bias, nilai konstanta yang menyesuaikan hasil prediksi.

Margin terbesar ditemukan dengan memaksimalkan nilai jarak *hyperplane* dan titik terdekatnya dengan persamaan (4).

$$\frac{1}{\|\vec{w}\|} \quad (4)$$

Keterangan :

$\|\vec{w}\|$: panjang dari vektor bobot \vec{w} .

Hal ini dapat dirumuskan sebagai *quadratic programming problem* yaitu mencari titik minimal persamaan (5) dengan memperhatikan *constraint* persamaan (6).

$$\min_w t(w) = \frac{1}{2} \|\vec{w}\|^2 \quad (5)$$

$$y_i(\vec{w} \cdot \vec{x}_i) - 1 \geq 0, \forall i \quad (6)$$

Keterangan :

\min_w : Nilai minimum dari fungsi objektif terhadap vector bobot \vec{w} .

$t(w)$: Fungsi objektif yang ingin diminimalkan.

$\|\vec{w}\|$: panjang dari vektor bobot \vec{w} .

y_i : Label kelas untuk titik data i .

Problem dapat dipecahkan dengan Teknik komputasi *langrange Multiplier*, seperti persamaan berikut.

$$L(\vec{w}, b, \alpha) = \frac{1}{2} \|\vec{w}\|^2 - \sum_{i=1}^l \alpha_i (y_i (\vec{w} \cdot \vec{x}_i + b) - 1), i = 1, 2, \dots, l$$

(7)

Keterangan :

w : Vector bobot

b : Bias atau intercept dari hyperplane.

α_i : pengali langrange untuk setiap titik data i .

y_i : Label kelas untuk titik data i .

x_i : Vektor fitur dari contoh data i .

l : Jumlah total titik data.

α_i adalah *Langrange Multipliers*, yang bernilai 0 atau positif $\alpha_i \geq 0$. Nilai optimal dari persamaan dapat dihitung dengan meminimalakan L terhadap \vec{w} dan b dan memaksimalkan L terhadap α_i . Berdasarkan sifat bahwa pada titik optimal $L = 0$, persamaan dapat dimodifikasi sebagai maksimalisasi problem yang mengandung α_i .

$$\sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j (\vec{x}_i \cdot \vec{x}_j)$$

Keterangan :

α_i : pengali langrange untuk setiap titik data i .

y_i : Label kelas untuk titik data i .

x_i : Vektor fitur dari contoh data i .

l : Jumlah total titik data.

$\vec{x}_i \cdot \vec{x}_j$: Hasil perkalian titik (dot product) antara vektor fitur \vec{x}_i dan \vec{x}_j

Dari hasil perhitungan diperoleh α_i yang bernilai positif. Data yang berkorelasi dengan α_i yang positif inilah yang disebut *support vector*[24].

$$\alpha_i \geq 0 (i = 1, 2, \dots, l) \quad \sum_{i=1}^l \alpha_i y_i = 0$$

Keterangan :

α_i : pengali langrange untuk setiap titik data i .

l : Jumlah total titik data.

y_i : Label kelas untuk titik data i .

3. Klasifikasi

Klasifikasi adalah proses untuk menemukan model atau fungsi yang dapat menggambarkan dan membedakan kelas data atau konsep, dengan tujuan agar model tersebut dapat digunakan untuk memprediksi kelas yang belum diketahui dari suatu objek pengamatan. Metode klasifikasi yang umum digunakan pada disiplin ilmu statistika adalah Analisis Diskriminan dan Regresi Logistik. Namun, semakin populernya era data yang menunjukkan bahwa terjadinya pertumbuhan pesat dari volume data yang luar biasa banyak sehingga menghasilkan set data besar (big data), maka sangat dibutuhkan alat analisis yang kuat dan serbaguna untuk mengungkap informasi berharga dari set data besar dan untuk mengubah set data besar tersebut menjadi pengetahuan yang terorganisir[18].

Disisi lain, pesatnya perkembangan teknologi kecerdasan buatan maka berkembanglah metode *machine learning*, yaitu mesin yang dikembangkan untuk bisa belajar dengan sendirinya tanpa arahan dari penggunanya Dimana dalam pengembangannya berdasarkan disiplin ilmu lain seperti statistika, matematika dan data mining. Metode klasifikasi pada *machine learning* yang sering digunakan pada metode *machine learning* adalah *Classification and Regression Trees* (CART), *Random Forest*, *Naïve Bayes*, *Support Vector Machines* (SVM), dan lain-lain[19].

4. Machine Learning

Machine learning atau dikenal dengan pembelajaran mesin adalah ilmu komputer yang bisa bekerja tanpa diprogram secara eksplisit. Banyak peneliti berpikir bagaimana cara untuk membuat kemajuan

menuju AI terhadap tingkat manusia. Machine learning ini merupakan kecerdasan buatan yang mempelajari bagaimana membuat data. Machine learning ini biasa disingkat dengan ML. Ini dibutuhkan untuk menerapkan teknik yang cepat dan kuat dalam menemukan masalah baru.

Secara definisi, Machine learning merupakan ilmu atau studi yang mempelajari tentang algoritma dan model statistik yang digunakan oleh sistem komputer untuk melakukan task tertentu tanpa instruksi eksplisit. Machine learning bergantung pada pola dan kesimpulan. Untuk mendapatkan pola dan kesimpulan tersebut, algoritma Machine learning menghasilkan model matematika yang didasari dari data sampel yang sering disebut dengan training data.

Pemakaian teknik ini berkaitan dengan pembelajaran mesin dan AI. Mesin ini membuktikan kepada algoritma atau program yang berjalan di komputer. Oleh karena itu, jika kita ingin belajar Machine learning, pastikan terus berinteraksi dengan data. Semua pengetahuan Machine learning pasti akan melibatkan data[5].

5. Streamlit

Streamlit adalah sebuah framework berbasis Python dan bersifat open source yang dibuat untuk memudahkan dalam membangun aplikasi web di bidang sains data dan Machine learning yang interaktif[10].

Tersedia widget bawaan untuk masukan pengguna, seperti pengunggahan gambar, penggeser, masukan teks, dan elemen hypertext markup language (HTML) lain yang sudah dikenal, seperti checkboxes dan radio buttons. Setiap kali pengguna berinteraksi dengan aplikasi Streamlit, skrip Python dijalankan kembali dari atas ke bawah. Hal ini merupakan sebuah konsep penting yang perlu diingat saat mempertimbangkan berbagai status aplikasi yang akan dipilih. Streamlit merupakan aplikasi yang tidak berbayar dan pengguna tidak perlu memiliki pengetahuan pengembangan front-end yang mahir untuk mengoperasikannya.

Streamlit dapat dijalankan pada editor Anaconda serta bahasa Python seri 3.7 ke atas, tetapi tidak mendukung pada editor Jupyter Notebook, sehingga harus dikonversi ke editor Pycharm atau Visual Code. Tampilan beranda pada aplikasi Streamlit dapat dipisahkan menjadi dua bagian, yaitu buttons, untuk pemilihan menu, serta tampilan visual chart. Hal ini menyebabkan dibutuhkan library NumPy serta Pandas untuk menampilkan grafik. Keluaran grafik sejalan dengan hasil olah data metode pembelajaran mesin menggunakan kombinasi lapisan tersembunyi LSTM dan GRU. Buttons berfungsi untuk memilih dataset dari kategori negara, jenis hewan, arsitektur lapisan tersembunyi, optimizer, serta pilihan untuk epoch dan prediksi dalam beberapa tahun kedepan.

6. *Google Collaboratory*

Google Collaboratory atau *Google Colab* adalah platform berbasis *cloud* untuk menulis, menjalankan, dan berbagi kode *Python* melalui web browser. Platform ini dirancang bagi *analyst*, *developer*, peneliti, dan pendidik yang bekerja di bidang data *science* dan *machine learning* dengan menyediakan *environment* komputasi yang fleksibel dan mudah diakses tanpa biaya.

Google Colab juga menawarkan kemampuan untuk menjalankan Jupyter Notebook (*web app open-source* untuk kombinasi kode, teks terformat, dan visualisasi data) langsung dari web browser tanpa perlu konfigurasi apa pun. Salah satu fitur utama dari *Google Colab* adalah kemampuannya untuk berkolaborasi secara *real-time*. Mirip dengan berbagi dokumen di Google Docs, *user* dapat berbagi *notebook* mereka dengan orang lain. Selain itu, karena *notebook* disimpan di Google Drive, file ini diakses dari perangkat mana pun dan kapan pun, mirip dengan dokumen lain di ekosistem *Google*.

Dalam konteks analisis data, *Google Colab* memiliki akses ke *library Python* yang sudah terinstal seperti *TensorFlow*, *PyTorch*, dan

OpenCV, memungkinkan *analyst* mengembangkan dan menjalankan model *Machine learning* lebih cepat dan efisien[11].

7. Website

Website adalah kumpulan dari halaman – halaman situs, yang terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam World Wide Web (WWW) di dalam Internet. Sebuah halaman web biasanya berupa dokumen yang ditulis dalam format HTML (Hyper Text Markup Language), yang selalu bisa diakses melalui HTTP, yaitu sebuah protokol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser[12].

Berikut merupakan pengertian website menurut para ahli :

Suwanto Raharjo. Menurut Suwanto Raharjo, layanan web adalah salah satu Internet yang paling banyak dipergunakan dibandingkan dengan layanan lain seperti ftp, gopher, news atau bahkan email[20].

Website Menurut Gregorius adalah kumpulan halaman web yang saling terhubung dan file-filenya saling terkait. Web terdiri dari page atau halaman, dan kumpulan halaman yang dinamakan homepage. Homepage berada pada posisi teratas, dengan halaman-halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut child page, yang berisi hyperlink ke halaman lain dalam web[21].

Definisi Website menurut Hakim Lukmanul adalah fasilitas internet yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada website disebut dengan web page dan link dalam website memungkinkan pengguna bisa berpindah dari satu page ke page lain (hyper text), baik diantara page yang disimpan dalam server yang sama maupun server diseluruh dunia. Pages diakses dan dibaca

melalui browser seperti Netscape Navigator, Internet Explorer, Mozilla Firefox, Google Chrome dan aplikasi browser lainnya[22].

8. *Unified Modeling Language*

Pada rancangan desain sistem ini menggunakan Pemodelan *Unified Modeling Language (UML)*. Dengan *Use-Case Diagram Unified Modeling Language (UML)* merupakan notasi lengkap untuk membuat visualisasi model suatu sistem. *UML* ini digunakan dalam 27 pemodelan obyek guna menyajikan sistem yang berorientasi pada obyek pada orang lain, akan sangat sulit dilakukan jika pemodelan tersebut dilakukan dalam bentuk kode bahasa pemrograman. Kesulitan yang muncul adalah timbulnya ketidakjelasan dan salah terjemah (*interpretation*) dalam pembacaan kode pemrograman untuk pemodelan obyek tersebut[13].

a. *Class Diagram*

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi[14].

Tabel 2.2 Simbol Class Diagram

No	Simbol	Nama	Keterangan
1		Class	Kelas pada struktur system atau bisa diartikan sebagai template yang berguna dalam membuat suatu objek serta mengaplikasikan perilaku objek dalam suatu sistem. Dalam UML sendiri, kelas

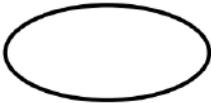
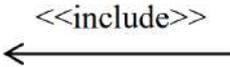
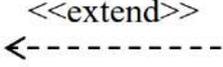
No	Simbol	Nama	Keterangan
			dapat mewakili sekumpulan objek ataupun hanya satu objek dengan catatan memiliki struktur serta perilaku yang sama.
2		Asosiasi	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
3		Asosiasi berarah	Relasi antarkelas dengan makna kelas satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
4		Extend	Relasi antarkelas dengan makna seluruhnya merupakan tambahan fungsional dari class lainnya jika suatu kondisi terpenuhi
5		Agregasi	Relas antarkelas dengan makna semuabagian (whole-part)

b. *Use Case Diagram*

Use case diagram system Menurut Pooley dalam Probowo Pudjo Widodo dan Herlawati model *use-case* dapat dijabarkan dalam

diagram *use case*, tetapi yang perlu diingat, diagram tidak identik dengan model karena model lebih luas dari diagram. Secara umum *use case* menggambarkan *external view* dari sistem yang akan kita buat modelnya. Disamping itu *use case* menggambarkan fungsi tertentu dalam suatu sistem berupa 28 komponen, kejadian atau kelas.

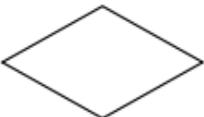
Tabel 2.3 Simbol Use Case Diagram

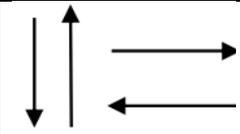
No	Simbol	Keterangan
1		Aktor : Mewakil peran orang, system yang lain, atau alat ketika berkomunikasi dengan Use Case
2		Use Case : Abstraksi dan interaksi antara sistem dan aktor
3		Association : Abstraksi dari penghubung antara aktor dengan Use Case Tabel Simbol Use Case Diagram
4		Generalisasi : Menunjukkan spesialisasi actor untuk dapat berpartisipasi dengan Use Case
5		Menunjukkan bahwa suatu Use Case seluruhnya merupakan fungsionalitas dari Use Case lainnya
6		Menunjukkan bahwa suatu Use Case seluruhnya merupakan tambahan fungsional dari Use Case lainnya jika suatu kondisi terpenuhi

c. *Activity Diagram*

Activity diagram adalah diagram yang menggambarkan aspek dinamis dari sistem. *Activity diagram* pada dasarnya adalah diagram alir untuk mewakili aliran bentuk satu aktivitas ke aktivitas kegiatan lain. Kegiatan ini dapat digambarkan sebagai operasi dari sistem. Jadi aliran kontrol diambil dari satu operasi ke yang lain. Aliran ini dapat berurutan, bercabang atau bersamaan

Tabel 2.4 Notasi Activity Diagram

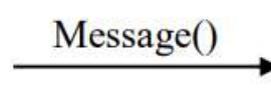
No	Gambar	Nama	Keterangan
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Action State	dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk dan diawali
4		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang

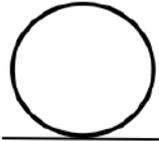
No	Gambar	Nama	Keterangan
			harus diambil pada kondisi tertentu
6		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

d. *Sequence Diagram*

Diagram ini bersifat dinamis. *Sequence diagram* merupakan diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu. Berikut adalah simbol-simbol *sequence diagram*

Tabel 2.5 Simbol Sequence Diagram

No	Gambar	Nama	Keterangan
1		LifeLine	Objek entity, antarmuka yang saling berinteraksi
2		Aktor	Digunakan untuk menggambar user / pengguna
3		Message()	Message Spesifikasi dari komunikasi antar objek yang memuat informasi

No	Gambar	Nama	Keterangan
			tentang aktivitas yang terjadi
4		Boundary	Digunakan untuk menggambarkan sebuah form
5		Control Class	Digunakan untuk menghubungkan boundary dengan table
6		Entity Class	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan

9. Pengujian *Black Box*

Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut[15]. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black box* memungkinkan perekaya sa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program[6].

Pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

- a. Fungsi – fungsi yang tidak benar atau hilang.
- b. Kesalahan *interface*.
- c. Kesalahan dalam struktur data atau akses database eksternal.
- d. Kesalahan kinerja.
- e. Inisialisasi dan kesalahan terminasi

Dengan mengumpulkan data pengujian menggunakan pendekatan secara wawancara dan observasi langsung terhadap fungsi sistem untuk mendapatkan hasil *presentase* rata-rata dari jumlah responden penguji yang dilakukan dengan perhitungan sebagai berikut:

$$\begin{aligned} & \text{Jumlah presentase rata – rata tercapai} \\ & = \frac{(\text{presentase hasil tercapai})}{(\text{Jumlah Responden})} \end{aligned}$$

10. Pengujian *White Box*

a. Pengertian

Pengertian uji coba *white box* merupakan metode desain uji kasus yang menggunakan struktur kontrol dari desain prosedural untuk menghasilkan kasus-kasus uji. Dengan menggunakan metode uji coba *white box*, para pengembang software dapat menghasilkan kasus-kasus uji yang :

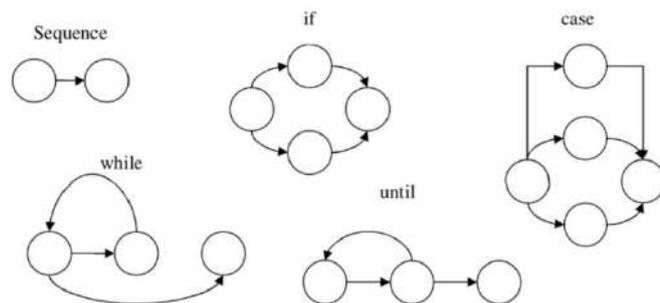
- 1) Menjamin bahwa seluruh *independent paths* dalam modul telah dilakukan sedikitnya satu kali
- 2) Melakukan seluruh keputusan logikal baik dari sisi benar maupun salah
- 3) Melakukan seluruh perulangan sesuai batasannya dan dalam batasan operasionalnya
- 4) Menguji struktur data internal untuk memastikan validitasnya.

b. Uji coba berbasis alur (Basis Path Testing)

Uji coba berbasis alur merupakan teknik uji coba white box pertama yang diusulkan oleh Tom McCabe. Metode berbasis alur memungkinkan perancang kasus uji untuk menghasilkan ukuran kompleksitas logikal dari desain prosedural dan menggunakan ukuran ini untuk mendefinisikan himpunan basis dari alur eksekusi. Kasus uji dihasilkan untuk melakukan sekumpulan basis yang dijamin untuk mengeksekusi setiap perintah dalam program, sedikitnya satu kali selama uji coba.

c. Notasi Graf Alur

Notasi sederhana untuk mereperentasikan alur kontrol disebut graf alur (*flow graph*).



Gambar 2.5 Flow Graph Nation

d. *Cyclomatic Complexity*

Cyclomatic complexity merupakan *software metric* yang menyediakan ukuran kuantitatif dari kompleksitas logikal suatu program. Ketika digunakan dalam konteks metode uji coba berbasis alur, nilai yang dikomputasi untuk kompleksitas *cyclomatic* mendefinisikan jumlah *independent path* dalam himpunan basis suatu program dan menyediakan batas atas untuk sejumlah uji coba yang harus dilakukan untuk memastikan bahwa seluruh perintah telah dieksekusi sedikitnya satu kali. Hasil kompleksitas *cyclomatic* menggambarkan banyaknya path dan batas atas sejumlah uji coba

yang harus dirancang dan dieksekusi untuk seluruh perintah dalam program dengan menggunakan rumus perhitungan sebagai berikut:

$$V(G) = (E - N + 2), \text{ atau}$$

$$V(G) = (P + 1), \text{ atau}$$

$$V(G) = R$$

Dengan keterangan sebagai berikut :

E : Jumlah *edge* pada *flow graph*

N : Jumlah *node* pada *flow graph*

P : Jumlah *node* simpul

R : *Region* atau area yang dibatasi oleh *node* dan *edge*

V(G) : *Cyclomatic Complexity*

11. *Software Testing dan Vertifikasi*

Pada tahap software testing dan verifikasi adalah tahapan untuk mendapatkan kualitas dari software dapat ternilai jelas fungsi dan kinerjanya, sehingga secara implisit menyatakan karakteristik yang diharapkan oleh semua pengembang software[15]. Testing/Pengujian merupakan proses eksekusi suatu program dengan maksud meneliti kesalahan program. Testing/pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya. Suatu pengujian harus mengacu pada suatu resiko-resiko pengembangan sistem. Maka dapat disimpulkan bahwa pengujian yang baik tidak hanya ditujukan untuk menemukan kesalahan pada perangkat lunak tetapi juga untuk dapat menemukan kesalahan secara lebih teliti.

UAT (*User Acceptance Test*) adalah suatu proses pengujian yang dilakukan oleh pengguna dengan hasil output sebuah dokumen hasil uji yang dapat dijadikan bukti bahwa software sudah diterima dan sudah memenuhi kebutuhan yang diminta. UAT tidak jauh beda dengan kusioner pada tahap awal pembuatan aplikasi.[16]

12. Akurasi SVM

Metode ini menerapkan prinsip-prinsip algoritma kecerdasan buatan yang disusun berdasarkan teori statistik dan optimasi yang memungkinkan komputer untuk mempelajari tata cara klasifikasi dan regresi, meningkatkan akurasi dalam memprediksi hasil, dan menghindari kekurangan *overfitting*. Cara kerja SVM adalah dengan melakukan regresi pada data pondasi kedalam dua kelas atau lebih, kemudian memisahkannya dengan sebuah garis bantu vektor yang disebut dengan *hyper-plane*[8].

Hasil pengukuran akurasi dilakukan untuk mengetahui kinerja dari sistem early warning stunting dengan Algoritma SVM studi kasus desa Tamanrejo berdasarkan data manual yang telah dibuat. Pengukuran tersebut dilihat dari perbandingan akurasi klasifikasi menggunakan metode SVM berdasarkan hasil dari klasifikasi data yang dilakukan. Untuk menghitung tingkat akurasi pengujian dapat menggunakan rumus yang ditunjukkan.

$$\text{Tingkat Akurasi} = \frac{\text{Data Berhasil}}{\text{Jumlah Data}} \times 100\%$$

Dimana:

Jumlah Data adalah banyaknya data masukkan yang diproses sistem
Data berhasil adalah kecocokan antara keluaran sistem dengan data manualnya.

13. Python

Python merupakan bahasa pemrograman tingkat tinggi yang sering digunakan untuk pengembangan perangkat lunak, pemrograman web, analisis data, kecerdasan buatan, dan berbagai

aplikasi lainnya. Python dikembangkan oleh seorang programmer bernama Guido Van Rossum pada akhir 1980-an di Centrum Wiskunde & Informatica (CWI) di Belanda. Python menggunakan indentasi untuk membedakan blok kode, tidak seperti bahasa pemrograman lain yang menggunakan symbol sebagai pembeda blok kode[23].

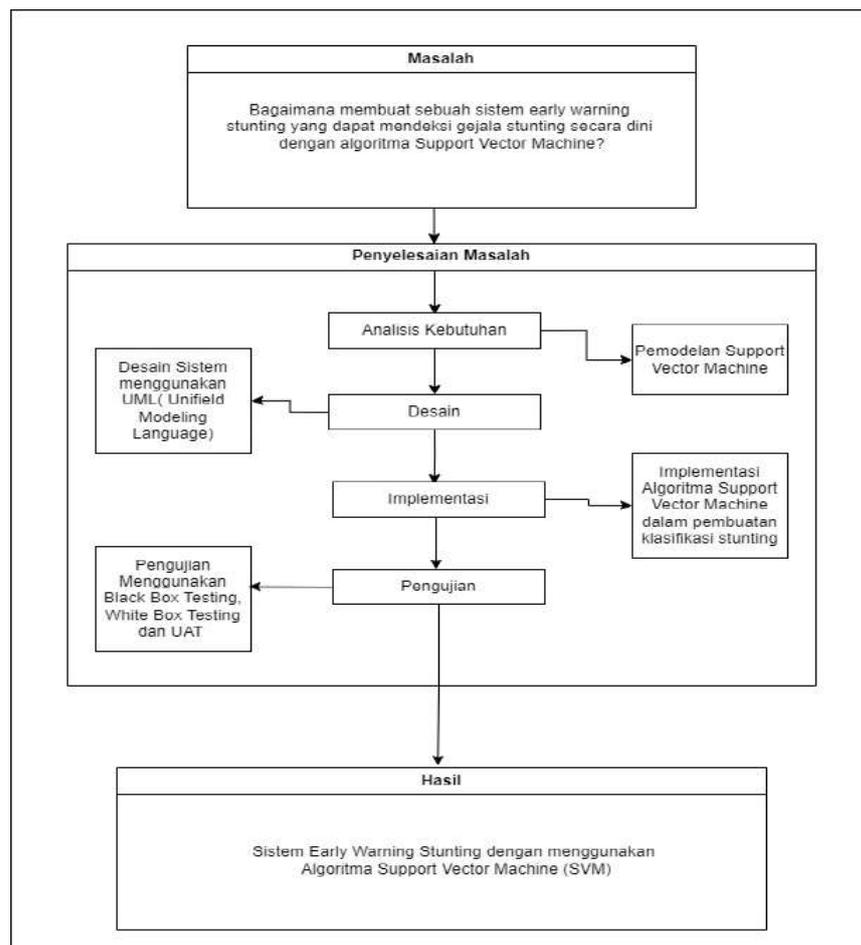
14. *Early Warning*

Sistem Peringatan Dini atau Early Warning System (EWS), EWS merupakan sistem peringatan dini yang menggunakan skoring dalam menilai perburukan kondisi pasien dan serta meningkatkan manajemen kegawatdaruratan secara keseluruhan. Skor peringatan dini EWS direkomendasikan sebagai bagian dari penilaian awal pasien dan respons terhadap kerusakan organ. Kondisi pasien yang berisiko lebih awal dapat diidentifikasi menggunakan EWS. Penilaian tanda perburukan kondisi pasien, dikombinasikan dengan pemantauan berbasis tanda vital, dapat mendukung deteksi dini kerusakan fisiologis pasien. Pentingnya deteksi dini telah mendorong dan mengaktifkan respon medis layanan kesehatan di Kanada, Australia, dan Inggris dalam penerapan sistem skoring peringatan dini. Sistem peringatan dini berguna untuk pemantauan atau deteksi dini sebelum kondisi pasien memburuk sehingga dapat mengambil tindakan atau jalur rujukan yang tepat. Berdasarkan penyakit yang mendasarinya, tanda-tanda klinis perburukan kondisi pasien seringkali serupa, dimana terlihat pada fungsi pernapasan, kardiovaskular, dan neurologis. Observasi pasien yang efektif adalah kunci pertama untuk menentukan kondisi pasien. Sangat penting untuk memiliki praktik keperawatan yang lebih baik sehingga dapat memberikan laporan secepat mungkin agar bisa menurunkan angka kesakitan dan kematian[25].

C. Kerangka Berfikir

Kerangka berfikir merupakan model konseptual tentang bagaimana teori berhubungan dengan berbagai faktor yang telah diidentifikasi sebagai masalah yang penting.

Sistem *early warning* stunting ini merupakan sistem yang dirancang untuk mempermudah pendektasian terjadinya penyakit stunting di Desa Tamanrejo. Sistem yang dirancang ini menghasilkan tampilan dari inputan data yang bisa langsung di koversi kedalam ekstensi csv yang berguna untuk sample data agar dapat meakukan klasifikasi penyakit stunting melalui google colab. Diharapkan dengan adanya sistem ini dapat mempermudah bidan ataupun kader posyandu Desa Tamanrejo dalam mendeketsi penyakit Stunting secara tepat dan efisien.



Gambar 2.5 Kerangkar Berfikir

BAB III

METODE PENELITIAN

A. Pendekatan Penelitian

Dalam pembangunan sistem diperlukan suatu pendekatan dan pengembangan sistem yang akan menentukan proses penyelesaian rekayasa perangkat lunak. Sistem Early Warning Stunting ini dilakukan peneliti di Desa Tamanrejo Kecamatan Limbangan, Kabupaten Kendal. Metode pendekatan sistem menggunakan metode *waterfall* yang dapat melakukan inputan data serta melakukan klasifikasi data Stunting dengan tepat dan efisien sehingga penanganan Stunting dapat diketahui secara tepat dan cepat agar penanganan Stunting langsung bisa dilakukan. Metode *waterfall* sendiri harus dilakukan secara berurutan yang terdiri dari

1. Requirement Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.
2. Design Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.
3. Implementation Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.
4. Verification Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit

testing(dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

- Maintenance Ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya [17].

B. Lokasi Penelitian

Lokasi penelitian dilakukan di sebuah Desa Tamanrejo, yang berada di Kecamatan Limbangan Kabupaten Kendal. Jarak lokasi dari Universitas PGRI Semarang jika ditempuh dengan kendaraan sekitar 28 km dan dapat ditempuh dengan kendaraan pribadi sekitar 1 jam 30 menit. Peta lokasi terdapat pada Gambar 3.1, dan fokus penelitian ini yaitu pada *early warning* stunting yang banyak terjadi di Desa Tamanrejo.



Gambar 3.1 Peta Desa Tamanrejo

C. Variable Penelitian

Variabel penelitian adalah suatu atribut atau objek dalam kegiatan yang mempunyai variasi tertentu yang ditetapkan oleh peneliti untuk ditarik kesimpulannya. Variabel ini mempermudah peneliti dalam menetapkan data yang dikumpulkan. Penulis menggunakan variabel bebas (*independent variable*) yang nantinya variabel tersebut dapat mempengaruhi variabel lainnya, dengan kaitannya dengan permasalahan yang diteliti, maka yang menjadi variabel bebas adalah Sistem Early Warning Stunting. Adapun indikator yang menentukan suatu kualitas sistem informasi adalah :

1. *Audience*, terdiri dari bidan, kader posyandu dan masyarakat
2. *Credibility*, kepercayaan pengguna dalam menggunakan website yang berguna untuk efektifitas dan efisiensi pengelolaan.
3. *Accuracy*, memiliki dua definisi yaitu yang pertama informasinya haruslah sesuai dengan kenyataan dan yang kedua adalah website secara grammatical (penulisannya) bebas dari eror (kesalahan penulisan).
4. *Objectivity*, dalam website haruslah bebas dari hal-hal yang berbau politik, rasis, ataupun perbedaan jenis kelamin.
5. *Coverage*, memenuhi semua kriteria kebutuhan pengguna seperti kredibilitas, akurasi dan objektif.
6. *Currency*, informasi selalu update sesuai dengan pengelolaan hasil pada lapangan.
7. *Aesthetic or Visual Appeal*, komposisi website haruslah sesuai dengan nilai-nilai estetika agar menarik bagi penggunanya.
8. *Navigation*, pengguna dapat menggunakan seluruh website di dalamnya dengan leluasa.
9. *Accessibility*, mudah diakses oleh browser apapun.

D. Proses Experimen

1. Tahap persiapan, meliputi :
 - a. Perancangan penelitian,
 - b. Studi pustaka,
 - c. Pembuatan sistem informasi,
 - d. Validasi sistem informasi.
2. Tahap pelaksanaan penelitian, meliputi :
 - a. Pengelompokan sampel pada satu kelas penelitian,
 - b. Melaksanakan testing *black box*, *white box* dan UAT untuk mengetahui kelayakan sistem informasi,
 - c. Penggunaan sistem *early warning stunting*.
3. Pengelolaan dan analisis data.
4. Menyimpulkan hasil penelitian dengan alur yang sistematis.

E. Prosedur Penelitian

Prosedur penelitian merupakan tahapan yang akan dilakukan untuk memudahkan peneliti dalam melakukan penelitian. Dalam proses ini, peneliti melakukan beberapa tahapan, diantaranya :

1. Seorang peneliti bisa mendapatkan data penelitian menggunakan wawancara dengan terjun langsung pada lokasi dan menyiapkan pertanyaan-pertanyaan yang berkaitan antara tema penelitian dan hasil yang diharapkan.
2. Observasi merupakan salah satu langkah dalam pengambilan data yang sering dipergunakan. Observasi ini berkaitan erat dengan tata cara penelitian yang dilakukan dengan terjun langsung ke lokasi tanpa menggunakan perantara.
3. Teknik kepustakaan yaitu dengan tata cara mencari informasi dan wawasan dengan membaca buku atau artikel terlebih dahulu, kemudian dijadikan sebagai landasan teori yang menentukan hasil penelitian.

F. Teknik Pengumpulan Data

Pada penelitian kali ini peneliti menggunakan 2 cara pengumpulan data yaitu mengumpulkan data di internet yang berupa dataset sebanyak 500 file data yang diperoleh dari *Kaggle.com* data ini terdiri dari umur, jenis kelamin, tinggi badan, berat badan, dan status gizi dan data yang kedua dengan cara pendekatan wawancara dan observasi. Wawancara dilakukan dengan melalui tanya jawab dengan Bapak Kepala Desa Tamanrejo, Ibu Kepala Desa Tamanrejo, Bidan Desa Tamanrejo serta Kader Posyandu Desa Tamanrejo yang menangani dan mendata informasi dari balita-balita yang terkena stunting atau tidak untuk memperoleh data tentang masalah-masalah yang timbul pada pendeksian penyakit stunting yang terjadi. Observasi dilakukan untuk mengamati proses pendataan balita oleh Bidan/Kader Posyandu proses yang masih dilakukan secara manual.

G. Teknik Analisis Data

Pada penelitian kali ini peneliti menganalisis data menggunakan Teknik deskriptif kuantitatif yang menggambarkan tentang sistem informasi. Data yang diperoleh didapatkan dari hasil wawancara dengan Bapak Kepala Desa Tamanrejo, Ibu Kepala Desa Tamanrejo, Bidan Desa Tamanrejo dan Kader Posyandu Desa Tamanrejo. Analisis ini digunakan untuk menggambarkan karakteristik data disetiap variable cara ini digunakan agar mempermudah dalam memahami data pada setiap proses hasil analisis digunakan sebagai acuan untuk membuat sistem informasi. Analisis data lebih detail dengan cara menggunakan perhitungan manual dari SVM dengan menggunakan sample data

$$x_1 = [12,1,70,10]$$

$$x_2 = [20,1,50,20]$$

$$y_1 = -1 = \text{Stunting}$$

$$y_2 = 1 = \text{Normal}$$

Menghitung Kernel Linier

$$K(x_i, x_i) = x_i \cdot x_i = 12 \cdot 20 + 1 \cdot 1 + 70 \cdot 50 + 10 \cdot 20 = 3941$$

Lalu Menghitung Maximize

$$\alpha_1 + \alpha_2 - \frac{1}{2} (\alpha_1^2 K(x_1, x_1) + (\alpha_2^2 K(x_2, x_2) + 2\alpha_1 \alpha_2 y_1 y_2 K(x_1, x_1)))$$

Dengan Kendala

$$\alpha_1 y_1 + \alpha_2 y_2 = 0$$

Lalu menghitung kernel untuk titik yang sama

Kernel Linier

$$K(x_i, x_i) = (x_i, x_i) :$$

$$K(x_1, x_1) = 12^2 + 1^2 + 70^2 + 10^2 = 144 + 1 + 4900 + 100 = 5.145$$

$$K(x_2, x_2) = 20^2 + 1^2 + 50^2 + 20^2 = 400 + 1 + 2500 + 400 = 3.301$$

Dual objective kernel linier

$$\text{Maximize } \alpha_2 - \frac{1}{2} (\alpha^2 (5.145 + 3.301 - 2 \cdot 3.941))$$

$$= 2\alpha - \frac{1}{2} (\alpha^2 (5.145 + 3.301 - 7.882))$$

$$= 2\alpha - \frac{1}{2} (\alpha^2 (573))$$

$$= 2\alpha_2 - 286,5\alpha^2$$

Mencari nilai maksimum

$$\frac{d}{d\alpha} (2\alpha - 286,5\alpha^2) = 2 - 573\alpha = 0$$

$$286,5\alpha = 2$$

$$\alpha = \frac{2}{573} = \frac{1}{286,5}$$

Menghitung vector bobot w dengan $\alpha_i - \alpha_i = \frac{1}{250}$

$$w = \frac{1}{286,5} \cdot (-1) \cdot [12, 1, 70, 10] + \frac{1}{286,5} \cdot 1 \cdot [20, 1, 50, 20]$$

$$w = \left[-\frac{12}{286,5}, -\frac{1}{286,5}, -\frac{70}{286,5}, -\frac{10}{286,5} \right] + \left[\frac{20}{286,5}, \frac{1}{286,5}, \frac{50}{286,5}, \frac{20}{286,5} \right]$$

$$w = [-0.04, -0.003, -0.24, -0.03] + [0.06, 0.003, 0.17, 0.06]$$

$$w = [0.02, 0, -0.07, 0.03]$$

Lalu menghitung bias b

$$b = y_1 - w \cdot x_1$$

$$b = -1 - ([0.02, 0, -0.07, 0.03] \cdot [12, 1, 70, 10])$$

$$b = -1 - (0.02 \cdot 12 + 0 \cdot 1 + (-0.07) \cdot 70 + 0.03 \cdot 10)$$

$$b = -1 - (0.24 - 4.9 + 0.3)$$

$$b = -1 - (-4.36)$$

$$b = -1 + 4.36$$

$$b = 3.36$$

Vector bobot = [0.02, 0, -0.07, 0.03]

Bias = 3.36

Prediksi Data

[12, 1, 70, 10]

$$f(x) = w \cdot x + b$$

$$f([160, 90, 1]) = [0.02, 0, -0.07, 0.03]$$

$$\cdot [12, 1, 70, 10] + 3.36$$

$$= (0.02 \cdot 12) + (0 \cdot 1) + ((-0.07) \cdot 70) + (0.03 \cdot 10) + 3.36$$

$$= 0.24 + 0 + 4.9 + 0.3 + 3.36$$

$$= -1$$

Prediksi untuk data [12,1,70,10] adalah -1, yang berarti data tersebut masuk dalam kategori stunting.

BAB IV

HASIL PENGEMBANGAN DAN PEMBAHASAN

Pada bab ini akan membahas mengenai tentang hasil dan pembahasan dari penelitian yang telah dilakukan oleh penulis. Hal ini meliputi pengembangan dan perancangan aplikasi klasifikasi stunting yang terkena stunting dan normal

A. Hasil

Metode yang digunakan penulis untuk pembuatan aplikasi ini yaitu metode *Waterfall*, berikut merupakan tahap-tahap pembuatan aplikasi berdasarkan urutan pada metode *Waterfall* :

1. Analisis Kebutuhan

A. Analisis Kebutuhan Data

Data yang diperlukan untuk penelitian ini merupakan hasil gabungan dari berbagai sumber yang digunakan untuk mengklasifikasi berbagai jenis data. Dataset ini diperoleh dari internet, khususnya dari website Kaggle, serta dari Desa Tamanrejo. Proses awal dimulai dengan mengunduh dataset dari sumber-sumber ini, yang kemudian diunggah ke Google Colab untuk memudahkan penyimpanan dan aksesibilitasnya selama proses analisis.

Dataset tersebut kemudian diproses melalui serangkaian langkah untuk melakukan klasifikasi antara individu yang terdeteksi mengalami stunting dan yang memiliki status gizi normal. Proses klasifikasi ini didasarkan pada berbagai fitur yang relevan seperti tinggi badan, berat badan, umur, dan status gizi masing-masing individu. Langkah-langkah ini sangat penting untuk mengidentifikasi pola-pola yang ada dalam data, yang nantinya dapat digunakan untuk membantu dalam pengembangan strategi penanganan dan pencegahan stunting yang lebih efektif di masa depan. Setelah dataset dikumpulkan dan diunggah ke Google Colab, langkah selanjutnya adalah melakukan pra-pemrosesan data. Hal ini

mencakup pembersihan data untuk mengatasi nilai-nilai yang hilang atau tidak valid, normalisasi data untuk memastikan konsistensi dalam skala dan rentang nilai, serta ekstraksi fitur yang relevan untuk analisis lebih lanjut.

Selanjutnya, dataset dipisahkan menjadi dua kelompok utama berdasarkan status gizi: stunting dan normal. Ini dilakukan dengan menggunakan teknik klasifikasi SVM. Proses ini membantu dalam mengidentifikasi karakteristik atau pola yang membedakan kedua kelompok tersebut, seperti distribusi tinggi badan dan berat badan berdasarkan umur. Analisis lanjutan kemudian dilakukan untuk mengeksplorasi hubungan antara variabel-variabel tersebut dan untuk mengidentifikasi faktor-faktor risiko yang berkaitan dengan stunting. Hasil dari analisis ini dapat memberikan wawasan yang berharga bagi para peneliti dan praktisi kesehatan dalam merancang intervensi yang lebih efektif untuk pencegahan dan penanggulangan stunting.

B. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan proses untuk memahami, mendokumentasikan, dan mengatur persyaratan yang harus dipenuhi oleh suatu sistem perangkat lunak. Tujuan dari analisis kebutuhan sistem yaitu untuk mengetahui permasalahan yang akan dihadapi atau kebutuhan yang akan diterapkan pada sistem. Berikut kebutuhan yang akan digunakan peneliti untuk membuat aplikasi:

1) Kebutuhan Perangkat Keras

Kebutuhan perangkat keras atau hardware merupakan komponen fisik yang memiliki spesifikasi atau kriteria tertentu agar dapat menjalankan sistem dengan baik. Kebutuhan hardware yang dibutuhkan untuk menjalankan sistem ini yaitu :

- a. AMD Ryzen 5 5600H 12CPUs, 3.3GHz

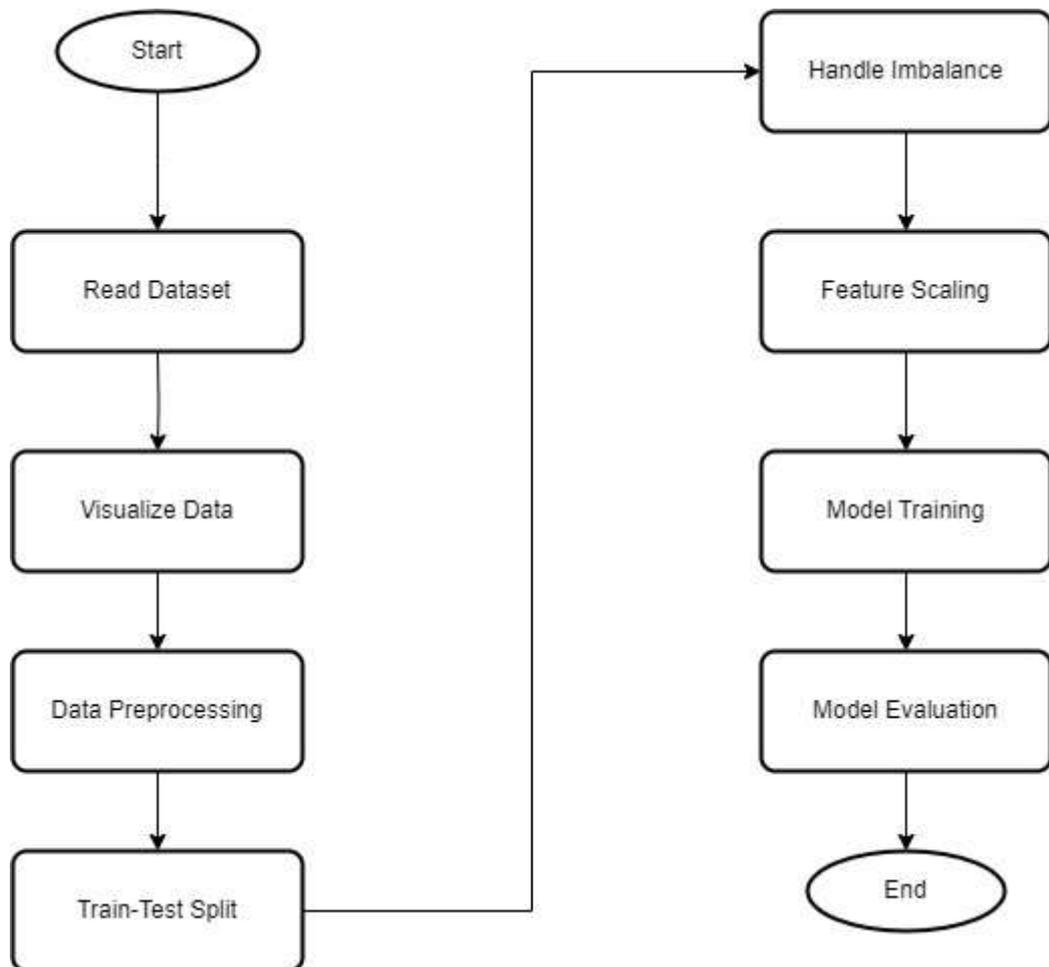
- b. Memory 8000MB RAM atau yang lebih tinggi
- c. Keyboard dan Mouse
- d. Kapasitas SSD 500GB atau yang lebih tinggi
- e. GPU AMD Radeon Graphics 4000MB atau byang lebih tinggi.

2) Kebutuhan Perangkat Lunak

Berikut ini kebutuhan perangkat lunak yang digunakan untuk membuat aplikasi klasifikasi stunting, yaitu :

- a. Google Colaboratory
- b. Visual Studio Code
- c. Anaconda
- d. Streamlit
- e. Github
- f. Draw.io
- g. Kaggle
- h. Google Drive
- i. Git

Selain itu Model SVM (Support Vector Machine) adalah salah satu algoritma yang kuat untuk masalah klasifikasi dan regresi. Berikut adalah alur dari model Support Vector Machine (SVM) dapat dilihat dari Gambar 4.1.



Gambar 4.1 *Flowchart SVM*

Berikut ini adalah penjelasan dari alur dari model svm :

- a) *Read Data* : Langkah pertama dalam proses machine learning adalah memuat dataset ke dalam *Google Colab*. Dataset ini berisi data mentah yang akan digunakan untuk melatih dan menguji model machine learning. Membaca dataset berarti mengimpor data dari sumbernya (misalnya file CSV, Excel, atau database) disini penulis menggunakan sumber file yang berbentuk CSV ke dalam struktur data yang akan diproses oleh algoritma SVM pada

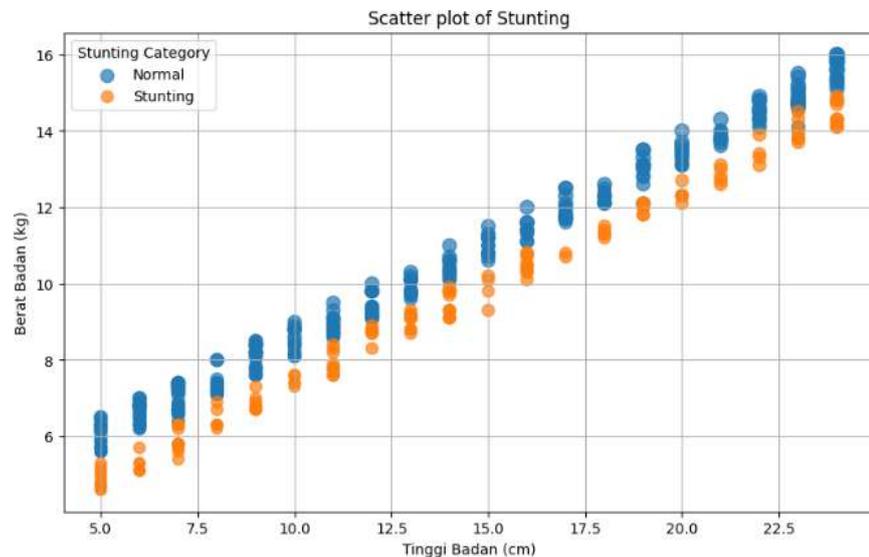
machine learning. Ini adalah langkah kritis karena kualitas dataset akan mempengaruhi performa akhir dari model yang dibangun. Jika dataset telah dimasukkan akan menampilkan seperti Gambar 4.2.

	Umur (bulan)	Jenis Kelamin	Tinggi Badan (cm)	Status Gizi	Berat Badan (kg)
0	5	laki-laki	66.9	0	5.7
1	5	laki-laki	64.1	0	5.6
2	5	laki-laki	57.1	1	4.7
3	5	laki-laki	71.2	0	5.9
4	5	laki-laki	64.6	0	5.6
...
505	24	perempuan	92.1	0	15.4
506	24	perempuan	92.1	0	15.2
507	24	perempuan	96.3	0	16.0
508	24	perempuan	87.1	0	15.3
509	24	perempuan	78.6	1	14.9

510 rows x 5 columns

Gambar 4.2 Hasil *Read Data*

- b) *Visualize Data* : Setelah dataset dimuat, langkah selanjutnya adalah memvisualisasikan data untuk melihat karakteristiknya. Dengan cara membuat grafik atau plot untuk mengeksplorasi distribusi data, hubungan antar variabel, dan pola-pola yang mungkin ada dalam data. Visualisasi data membantu dalam mengidentifikasi tren, anomali, atau korelasi yang mungkin ada dalam dataset. Ini juga membantu dalam menentukan strategi preprocessing dan pemilihan fitur yang relevan untuk model machine learning didalam program ini penulis menggunakan *Scatter Plot* dengan memvisualisasikan hubungan antara dua variabel dengan menempatkan titik data dalam ruang dua dimensi. Berikut ini adalah visualisasi yang dilakukan oleh penulis seperti Gambar 4.3.



Gambar 4.3 Hasil Visualisasi

- c) *Data Preprocessing* : Pra-pemrosesan data adalah serangkaian langkah untuk membersihkan, memodifikasi, dan menyiapkan data mentah sebelum diberikan ke model machine learning. Langkah ini bertujuan untuk meningkatkan kualitas data dan memastikan bahwa data siap untuk proses pembelajaran. Pra-pemrosesan data mencakup mengatasi nilai yang hilang, menangani pencilan, mengubah format data, dan melakukan normalisasi atau penskalaan fitur. Tujuannya adalah untuk memaksimalkan akurasi dan performa model machine learning dengan memastikan data yang digunakan sesuai dengan kebutuhan dan kualitas yang baik. Disini penulis menggunakan dua tahap preprocessing yaitu

- *Encoding Categorical Variables*

```
df['Jenis Kelamin'] = df['Jenis Kelamin'].map({'laki-laki': 1, 'perempuan': 0})
```

Gambar 4.4 Kode *Encoding Categorical Variables*

Pada Gambar 4.4, variabel kategorikal 'Jenis Kelamin' yang awalnya memiliki nilai 'laki-laki' dan 'perempuan' diubah menjadi representasi numerik. Dalam hal ini, 'laki-laki' dikodekan sebagai 1 dan 'perempuan' dikodekan sebagai 0. Proses ini disebut label encoding, di mana nilai-

nilai kategorikal diubah menjadi nilai numerik agar dapat digunakan oleh algoritma machine learning. Ini penting karena model machine learning memerlukan data numerik untuk melakukan perhitungan.

- *Feature Selection*

```
x = df[['Umur (bulan)', 'Berat Badan (kg)', 'Tinggi Badan (cm)', 'Jenis Kelamin']]  
y = df['Status Gizi']
```

Gambar 4.5 Kode *Feature Selection*

Pada Gambar 4.5 melibatkan pemilihan subset fitur atau variabel dari dataset untuk digunakan dalam pembuatan model machine learning. Di sini, kita memilih fitur-fitur 'Umur (bulan)', 'Berat Badan (kg)', 'Tinggi Badan (cm)', dan 'Jenis Kelamin' untuk dimasukkan ke dalam variabel X (fitur) dan 'Status Gizi' sebagai variabel y (target). Pemilihan fitur dilakukan untuk memilih subset fitur yang dianggap paling relevan atau informatif untuk memprediksi variabel target (dalam hal ini, 'Status Gizi'). Dengan memilih fitur-fitur yang paling penting, kita dapat meningkatkan kualitas model, mengurangi overfitting, dan mempercepat waktu pelatihan.

d) *Train-Test Split*: Pemisahan data latih dan uji adalah langkah untuk membagi dataset menjadi dua bagian: data latih yang digunakan untuk melatih model dan data uji yang digunakan untuk menguji performa model yang sudah dilatih. Memisahkan dataset menjadi data latih dan uji penting untuk menghindari overfitting (di mana model terlalu cocok dengan data latih) dan untuk mengukur kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya. Disini penulis menggunakan *Train-Test Split* seperti Gambar 4.6.

```
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

Gambar 4.6 Kode *Train-Test Split*

Penjelasan dari Gambar yang paling awal adalah memastikan bahwa telah mengimpor fungsi dari `from sklearn.model_selection import train_test_split` kemudian untuk parameter

- `X` : Variabel independen (fitur) dari dataset.
- `y`: Variabel dependen (target atau label yang ingin diprediksi).
- `test_size`: Ukuran dari data uji sebagai proporsi dari dataset keseluruhan. Dalam contoh ini penulis menggunakan, `test_size=0.3` yang berarti 30% dari data akan dialokasikan sebagai data uji, sementara 70% akan menjadi data latih.
- `random_state`: Parameter opsional yang digunakan untuk mengontrol pengacakan data saat pemisahan. Nilai ini memastikan bahwa hasil dari pemisahan data akan konsisten jika kode dieksekusi ulang dengan nilai `seed` yang sama.

Kemudian untuk Output dari Variable nya adalah

- `X_train`: Subset dari fitur yang akan digunakan untuk melatih model. Ini adalah bagian dari `X` yang akan digunakan untuk menyesuaikan (fit) model.
- `X_test`: Subset dari fitur yang akan digunakan untuk menguji model. Ini adalah bagian dari `X` yang akan digunakan untuk mengevaluasi performa model.
- `y_train`: Subset dari target yang sesuai dengan `X_train`. Ini adalah bagian dari `y` yang sesuai dengan data latih.
- `y_test`: Subset dari target yang sesuai dengan `X_test`. Ini adalah bagian dari `y` yang sesuai dengan data uji.

Dalam implementasi *Train-Test Split* ini setelah memanggil `train_test_split`, dataset (X dan y) akan dibagi secara acak menjadi dua subset: data latih dan data uji. Kemudian proporsi pembagian antara data latih dan data uji ditentukan oleh `test_size`. Dalam contoh ini, 30% dari dataset akan digunakan sebagai data uji, sedangkan 70% sisanya akan menjadi data latih. Lalu `random_state=42` memastikan bahwa pembagian data dilakukan secara deterministik dan hasilnya akan konsisten setiap kali kode dieksekusi ulang dengan nilai `seed` yang sama.

- e) *Handle Imbalance*: Penanganan ketidakseimbangan karena ada distribusi yang tidak merata dari kelas atau label dalam dataset. Misalnya, dalam masalah klasifikasi, satu kelas mungkin lebih banyak atau lebih sedikit daripada kelas lainnya. Teknik-teknik seperti *oversampling* (menambah sampel dari kelas minoritas), *undersampling* (mengurangi sampel dari kelas mayoritas), atau menggunakan algoritma khusus seperti SMOTE digunakan untuk memperlakukan masalah ketidakseimbangan ini agar model dapat belajar dengan baik dari kedua kelas. Disini penulis menggunakan Teknik *SMOTE (Synthetic Minority Over-sampling Technique)* seperti pada Gambar 4.7.

```
smote = SMOTE(random_state=42)
X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)
```

Gambar 4.7 Kode *Handle Imbalance*

SMOTE bekerja dengan cara mengidentifikasi sampel-sampel dari kelas minoritas yang berdekatan dalam ruang fitur, kemudian membuat sampel-sampel baru yang serupa tetapi tidak identik dengan menggunakan interpolasi linier antara pasangan sampel tersebut. Kemudian untuk implementasi penulis menuliskan penjelasan dari Gambar seperti :

`smote = SMOTE(random_state=42)`: Membuat objek SMOTE dengan menetapkan `random_state` untuk memastikan reproduktibilitas.

`X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)`: Memanggil metode `fit_resample` dari objek SMOTE untuk melakukan oversampling pada kelas minoritas dalam dataset latih (`X_train` dan `y_train`).

- `X_train`: Variabel independen (fitur) dari dataset latih.
- `y_train`: Variabel dependen (target) dari dataset latih.

Hasil dari operasi ini adalah `X_train_smote` dan `y_train_smote`, yang merupakan dataset latih yang telah diperluas (oversampled) untuk menangani ketidakseimbangan kelas.

f) *Feature Scaling*: Beberapa algoritma machine learning memerlukan fitur yang memiliki skala yang seragam untuk berfungsi secara optimal. Penskalaan fitur dilakukan untuk menyesuaikan rentang nilai dari setiap fitur dalam dataset. Normalisasi (scaling ke rentang [0, 1]) atau standardisasi (scaling ke mean 0 dan deviasi standar 1) adalah contoh teknik penskalaan fitur yang digunakan untuk memastikan bahwa perbedaan skala antar fitur tidak mengganggu kinerja model. seperti pada Gambar penulis menggunakan Teknik StandardScaler dari library scikit-learn di Python StandardScaler mengubah fitur-fitur sehingga memiliki mean 0 dan deviasi standar 1. Prosedur ini dilakukan dengan cara mengurangi rata-rata dari setiap fitur dan kemudian membaginya dengan deviasi standar dari fitur tersebut.

```
scaler = StandardScaler()
X_train_smote = scaler.fit_transform(X_train_smote)
X_test = scaler.transform(X_test)
```

Gambar 4.8 Kode *Feature Scaling*

Penjelasan dari Gambar 4.8 StandardScaler() membuat objek scaler yang akan digunakan untuk mengubah skala fitur. Lalu `fit_transform(X_train_smote)`: Metode ini menghitung rata-rata dan deviasi standar dari setiap fitur dalam `X_train_smote`, lalu mentransformasikan (menyesuaikan dan mengubah skala) data latih berdasarkan nilai-nilai tersebut. Dan yang terakhir `transform(X_test)`: Metode ini menggunakan rata-rata dan deviasi standar yang sudah dihitung dari `X_train_smote` untuk menyesuaikan dan mengubah skala data uji (`X_test`). Penting untuk menggunakan rata-rata dan deviasi standar yang sama dari data latih untuk mentransformasikan data uji, agar tidak mempengaruhi konsistensi dan interpretasi hasil model.

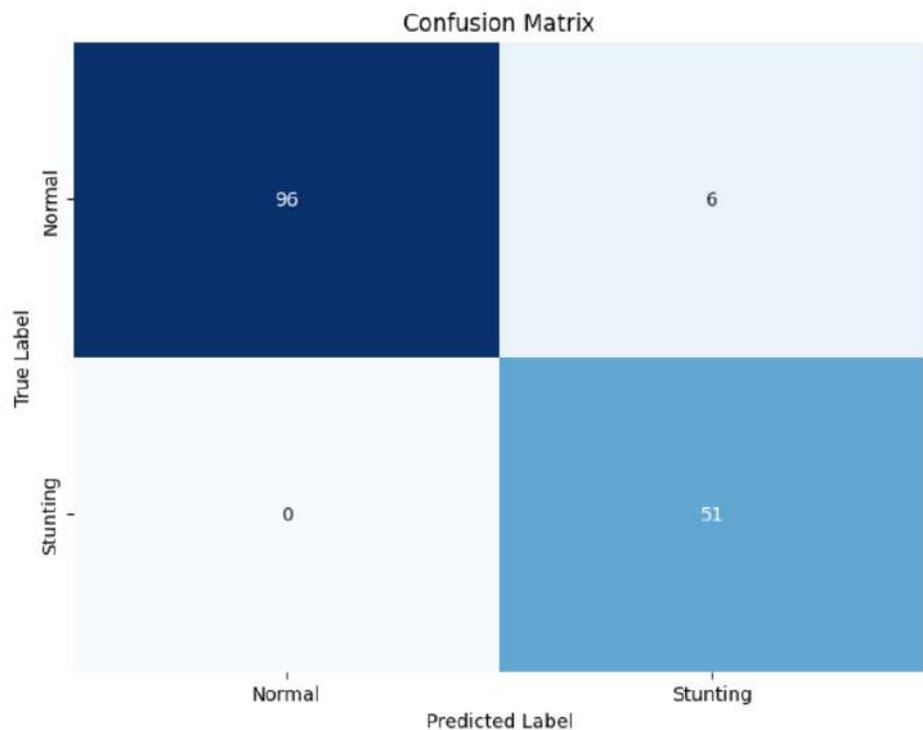
- g) Model *Training*: Evaluasi Model Akurasi dalam konteks klasifikasi, akurasi memberikan Gambaran tentang seberapa baik model dapat memprediksi dengan benar. Namun, akurasi tidak selalu menjadi ukuran yang baik jika dataset memiliki ketidakseimbangan kelas (dalam kasus ini penulis mengatasinya dengan cara SMOTE). karena menggunakan SMOTE untuk menangani ketidakseimbangan, akurasi memberikan perkiraan umum tentang performa model secara keseluruhan. Seperti pada model ini akurasi yang penulis peroleh sebesar 0,96 berarti model berhasil memprediksi 96% kasus dengan benar pada data uji seperti pada Gambar 4.9.

Akurasi: 0.9607843137254902

Laporan Klasifikasi:

	precision	recall	f1-score	support
0	1.00	0.94	0.97	102
1	0.89	1.00	0.94	51
accuracy			0.96	153
macro avg	0.95	0.97	0.96	153
weighted avg	0.96	0.96	0.96	153

Gambar 4.9 Hasil Perhitungan



Gambar 4.10 *Confusion Matrix*

Laporan klasifikasi memberikan informasi yang lebih detail tentang performa model untuk setiap kelas yang diprediksi. Ini mencakup:

- **Presisi (*Precision*)**: Perbandingan antara *True Positive* (TP) dengan banyaknya data yang diprediksi positif dalam presisi

menggunakan FP atau *False Positive*. Presisi yang tinggi menunjukkan sedikitnya jumlah false positive.

$$Precision = \frac{TP}{TP + FP} \quad (4.1)$$

- **Recall:** Perbandingan antara *True Positive* (TP) dengan banyaknya data yang sebenarnya positif dalam recall menggunakan FN atau *False Negative*. Dan dapat dituliskan secara sistematis dibawah ini. Recall yang tinggi menunjukkan sedikitnya jumlah false negative.

$$Recall = \frac{TP}{TP + FN} \quad (4.2)$$

- **F1-score:** *Harmonic mean* atau rata-rata dari presisi dan *recall*. Dapat dituliskan secara sistematis dibawah ini. *F1-score* yang tinggi menunjukkan baiknya keseimbangan antara presisi dan *recall*.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4.3)$$

- **Dukungan (Support):** Jumlah instance yang benar-benar milik kelas tersebut dalam data uji.

Disini penulis memilih 4 laporan klasifikasi untuk setiap model dan menghasilkan pada kelas stunting yang ditulis dengan angka 1 dengan presisi 0,89 berarti 89% prediksi stunting itu benar, sedangkan untuk recall menghasilkan 1,00 berarti model mengidentifikasi 100% dari semua kasus stunting dan f1-score menghasilkan 0,94 atau 94% yang berarti rata rata dari presisi dan recall untuk kelas stunting.

Sedangkan untuk kelas normal penulis juga masih menggunakan 4 laporan klasifikasi yang ditulis dengan angka 0 dengan presisi 1,00 berarti 100% prediksi normal itu benar, sedangkan untuk recall menghasilkan 0,94 berarti model mengidentifikasi 94% dari semua

kasus normal dan f1-score menghasilkan 0.97 atau 97% yang berarti rata rata dari presisi dan recall untuk kelas normal. Berikut hasil perhitungan pada Gambar 4.11.

```
Akurasi: 0.9607843137254902
Laporan Klasifikasi:
              precision    recall  f1-score   support

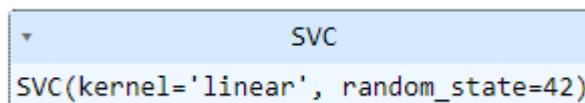
     0         1.00        0.94        0.97        102
     1         0.89        1.00        0.94         51

 accuracy                0.96        153
 macro avg              0.95        0.97        0.96        153
 weighted avg          0.96        0.96        0.96        153
```

Gambar 4.11 Hasil Perhitungan

Pemilihan kernel dalam model Support Vector Machine (SVM) merupakan keputusan yang mempengaruhi performa dan hasil dari model tersebut. Linear Kernel menjadi pilihan penulis karena data yang dimiliki dapat dipisahkan dengan garis lurus atau bidang linear dalam ruang fitur asli. Karakteristik ini memungkinkan SVM dengan Linear Kernel untuk menemukan hyperplane yang optimal untuk memisahkan kelas dengan margin maksimal di antara support vectors.

Keunggulan dari Linear Kernel sendiri adalah kesederhanaan dalam implementasi dan komputasi. Operasi *dot product* yang digunakan dalam Linear Kernel relatif sederhana dan efisien secara komputasional, sehingga cocok untuk data dengan dimensi tinggi atau jumlah fitur yang besar. Kecepatan ini membuat SVM dengan Linear Kernel dapat bekerja lebih cepat dalam fase pelatihan dan prediksi, yang penting dalam aplikasi yang membutuhkan respons cepat atau pengolahan data dalam skala besar. Berikut penerapan Linier Kernel pada Gambar 4.12.



```
SVC
SVC(kernel='linear', random_state=42)
```

Gambar 4.12 Penerapan Linier Kernel

C. Analisis Kebutuhan Fungsional

- 1) Aplikasi dapat menampilkan halaman utama yaitu berisi Prediksi Stunting pada balita.
- 2) Aplikasi dapat menampilkan halaman prediksi yang memuat umur, berat badan(kg), tinggi badan (cm), jenis kelamin dan juga tombol prediksi.
- 3) Aplikasi dapat memprediksi hasil data yang kita inputkan dan mengeluarkan *output* prediksi stunting atau normal.

2. Desain

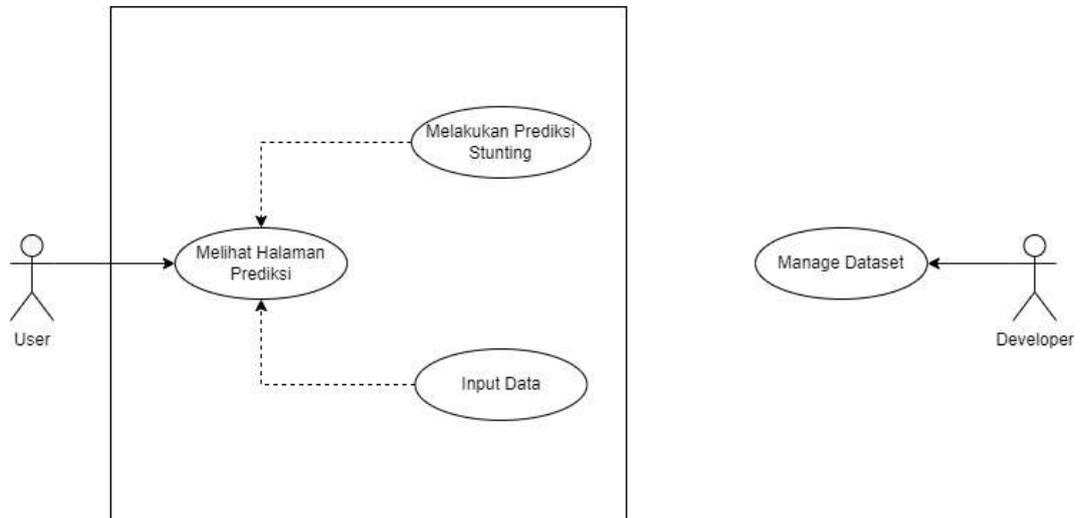
Fungsi dari desain sistem ini adalah agar pada saat proses implementasi atau penulisan kode lebih mudah, karena yang akan dibuat sudah ada gambaran jelas. Desain yang dilakukan oleh penulis untuk membuat aplikasi klasifikasi stunting, antara lain:

a. Desain *Unified Modelling Language* (UML)

Di dalam desain *Unified Modelling Language* (UML) melibatkan penggunaan berbagai jenis diagram untuk menggambarkan berbagai aspek aplikasi atau sistem yang akan dikembangkan. Berikut ini beberapa jenis diagram UML yang digunakan penulis, yakni:

1) *Use Case Diagram*

Use Case Diagram digunakan untuk menggambarkan interaksi antara aktor-aktor eksternal dengan sistem yang sedang dikembangkan. Dengan menggunakan *Use case diagram*, penulis dapat memiliki gambaran yang jelas tentang fungsionalitas yang diperlukan sistem dan bagaimana aktor akan berinteraksi dengan sistem tersebut



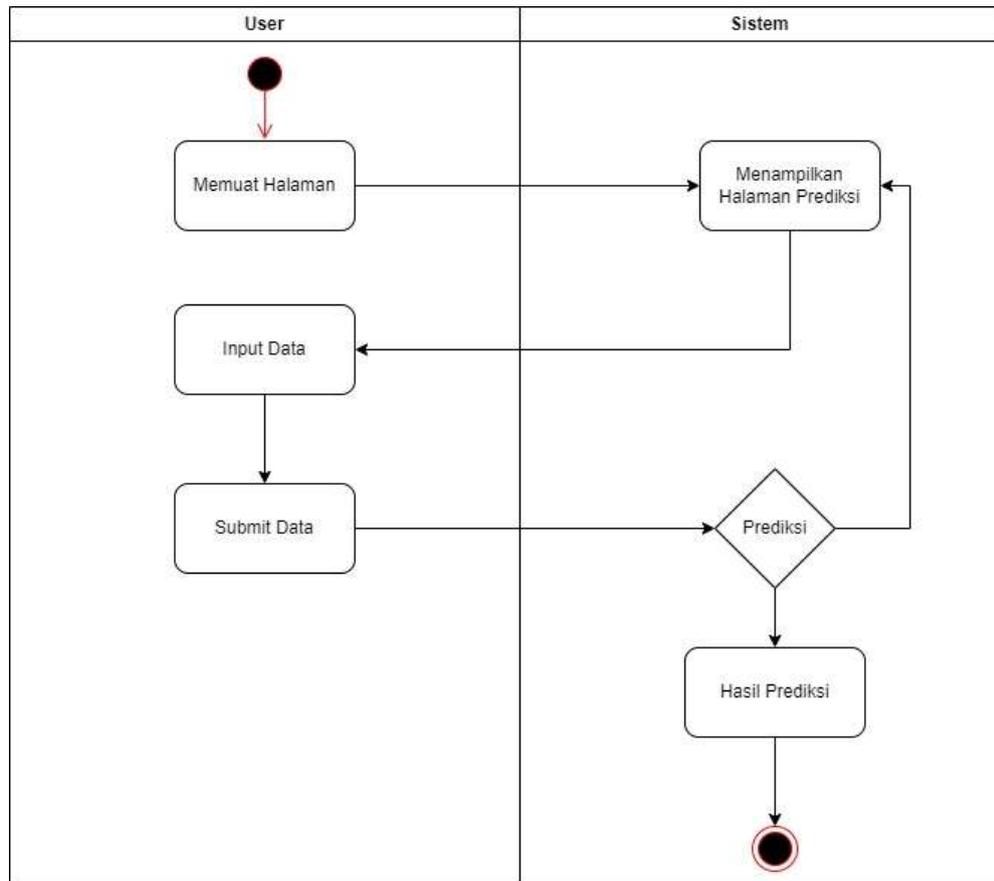
Gambar 4.13 Use Case Diagram

Berdasarkan Gambar 4.13, dapat dijelaskan bahwa, dalam sistem terdapat dua aktor yaitu *user* dan *developer*. *User* mendapatkan akses penuh pada aplikasi yang dimulai mengakses halaman prediksi, menginput data serta melakukan prediksi pada stunting. Sedangkan untuk *developer* mendapatkan hak akses untuk melakukan manajemen dataset.

2) Activity Diagram

Activity Diagram digunakan untuk memodelkan proses dari perspektif alur kerja terstruktur dan menggambarkan bagaimana aktivitas berinteraksi satu sama lain.

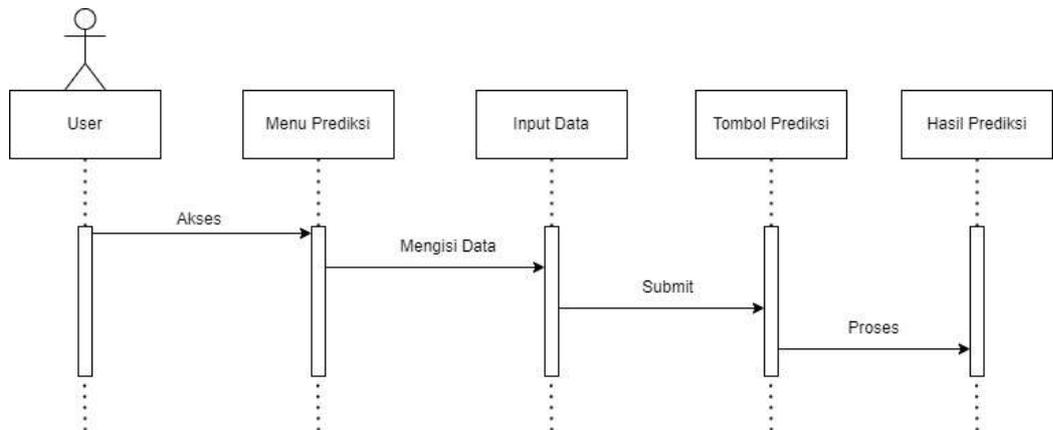
Berdasarkan Gambar 4.14, dijelaskan bahwa jika *user* berhasil mengakses sistem maka akan memuat sebuah halaman yang menampilkan halaman prediksi dan *user* juga mengisi data pada menu untuk melakukan proses submit data lalu data akan diprediksi jika benar maka sistem akan menampilkan hasil prediksinya jika salah sistem akan kembali untuk meminta mengisi data ulang pada halaman prediksi tersebut.



Gambar 4.14 Activity Diagram

3) Sequence Diagram

Sequence diagram atau diagram urutan digunakan untuk menggambarkan interaksi antar objek-objek dalam sebuah sistem secara urutan waktu. Dalam diagram ini, objek-objek yang terlibat dalam interaksi direpresentasikan sebagai kotak vertical yang disebut “*lifeline*”, yang menunjukkan waktu objek tersebut selama proses berlangsung.



Gambar 4.15 *Sequence Diagram*

Berdasarkan Gambar 4.15, langkah-langkah dari penggunaan sistem prediksi dimulai dengan pengguna yang mengakses menu prediksi. Pengguna kemudian mengisi data yang diperlukan untuk melakukan prediksi, seperti tinggi badan, berat badan, umur, dan status gizi lainnya. Setelah data diisi, pengguna kemudian menekan tombol 'Submit' untuk memulai proses prediksi. Sistem kemudian melakukan analisis terhadap data yang dimasukkan menggunakan algoritma SVM untuk mengklasifikasikan apakah anak yang bersangkutan memiliki status normal atau stunting. Setelah proses prediksi selesai, hasilnya ditampilkan kepada *user*, yang menunjukkan apakah anak tersebut tergolong stunting atau normal berdasarkan data yang telah diinputkan.

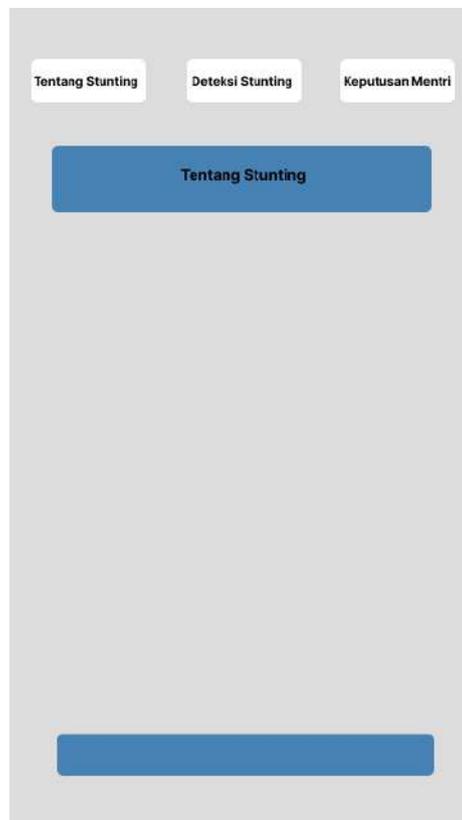
b. Desain *User Interface*

Pada bagian ini dijelaskan mengenai rancangan antarmuka utama perangkat lunak yang memfasilitasi pengguna untuk berinteraksi dengan perangkat lunak. Berikut ini penjabaran rancangan antarmuka tersebut.

1) Halaman Pengenalan Stunting

Halaman pengenalan stunting ini merupakan halaman yang digunakan untuk menampilkan informasi tentang stunting tersebut halaman ini sekaligus menjadi halaman utama saat mengakses aplikasi

ini sehingga aplikasi akan menampilkan informasi diawal saat mengakses aplikasi ini.



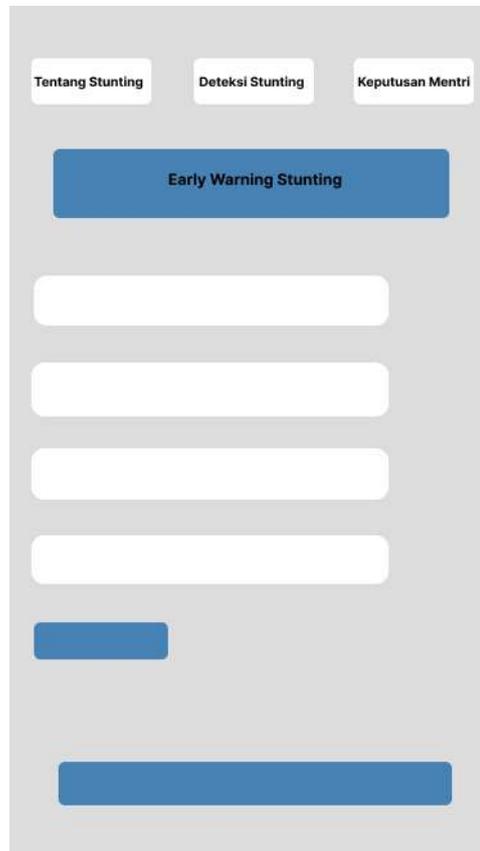
Gambar 4.16 Rancangan Halaman Pengenalan Stunting

2) Halaman Deteksi Stunting

Pada halaman deteksi stunting ini, Dilengkapi dengan beberapa form, termasuk umur, berat badan, tinggi badan, dan jenis kelamin. Pengguna dapat mengisi form-form tersebut untuk melakukan prediksi stunting dengan akurat. Setelah pengisian form selesai, terdapat satu tombol yang memungkinkan pengguna untuk melakukan prediksi berdasarkan data yang telah diinput sebelumnya.

Hasil prediksi akan ditampilkan dalam bentuk list yang memuat dua warna yang berbeda: hijau dan merah. Warna hijau menunjukkan bahwa baduta yang diprediksi dalam kondisi normal, sedangkan warna merah menandakan bahwa baduta tersebut terdeteksi mengalami stunting.

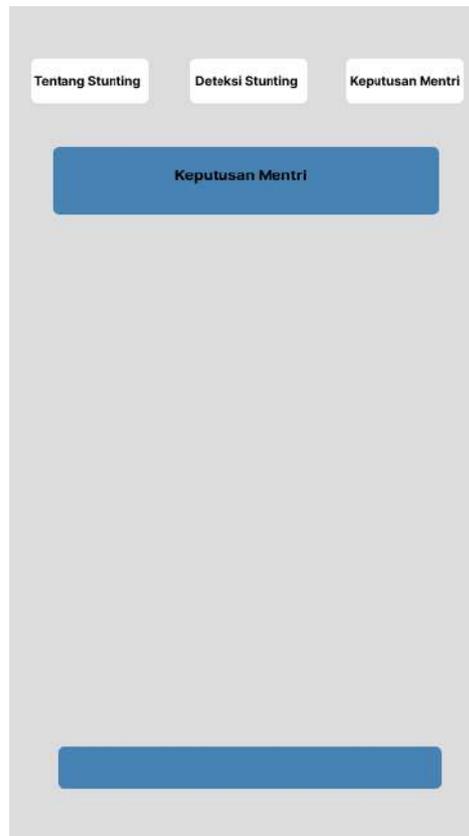
Kondisi stunting ini memerlukan penanganan segera untuk mencegah dampak yang lebih lanjut terhadap pertumbuhan dan perkembangan anak.



Gambar 4.17 Rancangan Halaman Deteksi Stunting

3) Halaman Keputusan Menteri

Halaman Keputusan Menteri ini berisikan hasil dari keputusan Kementerian Kesehatan Nomor HK.01.07/MENKES/1928/2022 menampilkan ringkasan – ringkasan isi dari keputusan tersebut, tidak lupa juga menyertakan link yang langsung direct ke website dari KEMENKES



Gambar 4.18 Rancangan Halaman Keputusan Menteri

3. Implementasi sistem

Setelah melatih model dengan menggunakan SVM, tahap berikutnya adalah mendeploy model yang telah diunduh ke server. Adapun tahapan-tahapan implementasi sistem sebagai berikut :

a. Pembuatan Streamlit

Pada penelitian ini menggunakan streamlit sebagai *framework* pembuatan sistem. Streamlit sendiri adalah sebuah *framework open-source* yang digunakan untuk membangun antarmuka pengguna (UI) interaktif untuk aplikasi *data science*. *Framework* ini dirancang khusus untuk mempermudah pengembangan aplikasi web dengan menggunakan *Python*. Sebelum pembuatan streamlit penulis mempersiapkan *requirements* untuk mendukung pembuatan streamlit. Adapun *requirements* yang digunakan dapat dilihat pada Tabel 4.1.

Tabel 4.1 *Requirements* Pembuatan Streamlit

No	Jenis <i>Requirements</i>	Versi
1.	Streamlit	1.36.0
2.	Pickle	0.21.3
3.	NumPy	1.26.3
4.	Pandas	2.2.2
5.	Scikit-learn	1.5.0
6.	Matplotlib	3.9.1

Berdasarkan tabel 4.1 bahwa terdapat berbagai *requirements* yang dipakai untuk mengelola dependensi yang dibutuhkan dalam proyek. Dari beberapa *requirement* tersebut terdapat fungsinya, yakni pertama streamlit digunakan untuk membangun aplikasi, pickle digunakan untuk menyimpan objek Python ke dalam file dan kemudian memuat kembali objek dari file tersebut. Numpy digunakan untuk komputasi numerik yang kuat dan efisien, pandas digunakan untuk membuat analyst dapat membaca, menulis, dan memodifikasi data dalam csv, scikit-learn digunakan untuk preprocessing data serta model selection, dan *requirements* yang terakhir adalah matplotlib yang digunakan untuk membuat visualisasi grafik.

Untuk tahap berikutnya yaitu penulisan kode untuk pembuatan streamlit. Dalam pembuatan kode diperlukan model yang berformat .pkl yang digunakan untuk pemrosesan prediksi label. Pada pembuatan kode juga mengubah label yang berawal dari *encoding* yang tipenya *integer* saat pelatihan model dengan SVM diubah menjadi *string*, misal untuk label 0 akan diubah menjadi 'Normal'. Langkah selanjutnya yaitu penulisan kode untuk prediksi data dapat dilihat pada Gambar 4.19.

```

%%writefile app.py
import streamlit as st

# Load the model and scaler
with open('model.pkl', 'rb') as model_file:
    model = pickle.load(model_file)

with open('scaler.pkl', 'rb') as scaler_file:
    scaler = pickle.load(scaler_file)

# Define the application
def main():
    st.title('Stunting Prediction App')

    # Get input from the user
    umur_bulan = st.number_input('Umur (bulan)')
    berat_badan = st.number_input('Berat Badan (kg)')
    tinggi_badan = st.number_input('Tinggi Badan (cm)')
    jenis_kelamin = st.selectbox('Jenis Kelamin', ['laki-laki', 'perempuan'])

    # Preprocess the input data
    data = [[umur_bulan, berat_badan, tinggi_badan, jenis_kelamin]]
    data_scaled = scaler.transform(data)

    # Make a prediction
    prediction = model.predict(data_scaled)[0]

    # Display the prediction
    if prediction == 0:
        st.write('Normal')
    else:
        st.write('Stunting')

if __name__ == '__main__':
    main()

```

Gambar 4.19 Kode Prediksi Data Stunting Pada Streamlit

Setelah penulisan kode berhasil, langkah terakhir yaitu pengujian streamlit dengan menuliskan perintah di terminal “streamlit run app.py” app.py merupakan file utama streamlit. Dari penulisan perintah tersebut terdapat *output* berupa *link* yang berupa *localhost* untuk menjalankan proyek streamlit. Untuk penelitian ini *link* nya berupa <http://localhost:8501/>.

b. Hasil Aplikasi

a. Halaman Tentang Stunting

Pada halaman ini yaitu halaman tentang stunting memuat informasi tentang stunting yang meliputi pengertian stunting kemudian penyebab stunting, dampak stunting dan juga pencegahan stunting

itu sendiri. Untuk implementasi halaman tentang stunting dapat dilihat pada Gambar 4.20.



Gambar 4.20. Tampilan Halaman Tentang Stunting

b. Halaman Deteksi Stunting

Pada halaman ini yang ditunjukkan kepada user y terdiri dari beberapa form yaitu ada umur, berat badan, tinggi badan dan jenis kelamin serta ada tombol submit prediksi yang digunakan untuk melakukan prediksi dengan pengelolaan data klasifikasi stunting

dengan metode SVM (Support Vector Machine), sehingga hasil prediksi bisa muncul sesuai dengan data yang sudah dimasukan jika tombol prediksi ditekan maka akan muncul hasil di atasnya apakah baduta tersebut berada pada kategori stunting atau normal. Seperti dapat dilihat pada Gambar 4.21.

Tentang Stunting Deteksi Stunting Keputusan Menteri

Early Warning Stunting

Silakan masukkan rincian di bawah ini untuk memperkirakan apakah anak tersebut mengalami stunting.

Umur (bulan) 24

Berat Badan (kg) 10.00

Tinggi Badan (cm) 85

Jenis Kelamin laki-laki

Prediksi

© 2024 Prediksi Stunting, Adam Virgilawan (20670026)

Gambar 4.21. Tampilan Halaman Deteksi Stunting

c. Halaman Tentang Stunting

Pada halaman Keputusan Menteri ini memuat informasi tentang keputusan Kementrian Kesehatan Nomor HK.01.07/MENKES/1928/2022 pada halaman ini berisikan tentang ringkasan keputusan kemudian pedoman pemeriksaan fisik lalu juga

ada diagnosis nya kemudian tidak lupa disertakan sumber yaitu website dari Kemenkes itu sendiri agar dapat melihat lebih detail. Untuk implementasi halaman tentang stunting dapat dilihat pada Gambar 4.22.



Gambar 4.22. Tampilan Halaman Keputusan Menteri

4. Pengujian

Pengujian dilakukan untuk memastikan kualitas aplikasi berfungsi dengan baik. Pada penelitian ini menggunakan 3 jenis pengujian yaitu *black box testing*, *white box testing*, *user acceptance testing*. Untuk pengujian pada penalties ini dapat dirinci sebagai berikut:

a. Pengujian *Black Box*

Pengujian *black box* digunakan untuk memastikan aplikasi berfungsi sesuai dengan harapan pengguna. Pengujian *black box* dilakukan oleh tiga dosen informatika. Pengujian *black box* dibagi menjadi tiga bagian, yaitu rencana pengujian, hasil pengujian, dan kesimpulan pengujian.

1) *Form Pengujian Black Box*

Form pengujian *black box* aplikasi dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian *Black Box*

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
Input valid	Masukan umur 24 bulan kemudian masukan berat badan 35 kg masukan tinggi badan 70 cm pilih gender 'Perempuan' lalu klik tombol pada prediksi	Hasil Prediksi Stunting akan menampilkan prediksi normal		
Input invalid	Masukan umur -20 sampai -100 bulan kemudian berat 20 kg – 30 kg lalu tinggi badan 50-70 cm pilih gender	Aplikasi menangani input invalid dan menampilkan pesan error umur harus lebih dari 0 bulan dan aplikasi tidak melakukan prediksi.		

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
Input invalid	Masukan Umur 40 bulan kemudian masukan berat 0 kg sampai -30kg lalu tinggi badan 100 cm dan pilih gender 'perempuan'	Aplikasi menangani input invalid dan menampilkan pesan error berat badan harus lebih dari 0 kg dan aplikasi tidak melakukan prediksi.		
Input invalid	Masukan Umur 25-30 bulan kemudian berat 25-36 kg Masukan tinggi badan 0 cm sampai -90 cm dan pilih gender 'Laki-laki' Klik tombol pada prediksi	Aplikasi menangani input invalid dan menampilkan pesan error tinggi badan harus lebih dari 0 cm dan aplikasi tidak melakukan prediksi.		
Input maksimal	Masukkan Umur 60 kemudian masukan berat 50 kg lalu tinggi 150 cm dan pilih gender 'laki- laki'	Aplikasi memberikan prediksi yang sesuai.		
Input kombinasi	Masukkan Umur 30 lalu masukan berat	Aplikasi memberikan		

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
	20,50 kg kemudian masukan tinggi 20 cm dan pilih gender 'perempuan'	prediksi yang sesuai.		
Respons Aplikasi	Masukkan data lakukan prediksi secara berulang-ulang secara singkat	Aplikasi akan menampilkan hasil prediksi yang sama secara cepat		
Konsistensi	Masukkan data yang diisi dengan nilai yang sama	Aplikasi akan menampilkan hasil prediksi sesuai dengan data yang diisi		
Tampilan	Masukkan umur bebas kemudian masukan berat badan bebas masukan tinggi badan bebas dan pilih jenis kelamin 'laki-laki'	Aplikasi akan menampilkan prediksi normal dengan warna hijau dan jika prediksi stunting akan berwarna merah		

2) Hasil perhitungan Pengujian *Black Box*

Hasil pengujian *black box* dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Pengujian *Black Box*

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan					
			Diterima			Ditolak		
			1	2	3	1	2	3
Input valid	Masukan umur 24 bulan kemudian masukan berat badan 35 kg masukan tinggi badan 70 cm pilih gender 'Perempuan' lalu klik tombol pada prediksi	Hasil Prediksi Stunting akan menampilkan prediksi normal	√	√	√			
Input invalid	Masukan umur -20 sampai -100 bulan kemudian berat 20 kg – 30 kg lalu tinggi badan 50-70 cm pilih gender	Aplikasi menangani input invalid dan menampilkan pesan error umur harus lebih dari 0 bulan dan aplikasi tidak melakukan prediksi.	√	√	√			
Input invalid	Masukan Umur 40 bulan kemudian masukan berat 0 kg sampai -30kg lalu tinggi badan 100 cm dan pilih gender 'perempuan'	Aplikasi menangani input invalid dan menampilkan pesan error berat badan harus lebih dari 0 kg dan aplikasi tidak	√	√	√			

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan							
			Diterima			Ditolak				
			1	2	3	1	2	3		
		melakukan prediksi.								
Input invalid	Masukan Umur 25-30 bulan kemudian berat 25-36 kg Masukan tinggi badan 0 cm sampai -90 cm dan pilih gender 'Laki-laki' Klik tombol pada prediksi	Aplikasi menangani input invalid dan menampilkan pesan error tinggi badan harus lebih dari 0 cm dan aplikasi tidak melakukan prediksi.	√	√	√					
Input maksimal	Masukkan Umur 60 kemudian masukan berat 50 kg lalu tinggi 150 cm dan pilih gender 'laki- laki'	Aplikasi memberikan prediksi yang sesuai.	√	√	√					
Input kombinasi	Masukkan Umur 30 lalu masukan berat 20,50 kg kemudian masukan tinggi 20 cm dan pilih gender 'perempuan'	Aplikasi memberikan prediksi yang sesuai.	√	√	√					
Respons Aplikasi	Masukkan data lakukan prediksi secara berulang-	Aplikasi akan menampilkan hasil prediksi yang	√	√	√					

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan							
			Diterima			Ditolak				
			1	2	3	1	2	3		
	ulang secara singkat	sama secara cepat								
Konsistensi	Masukkan data yang diisi dengan nilai yang sama	Aplikasi akan menampilkan hasil prediksi sesuai dengan data yang diisi	√	√	√					
Tampilan	Masukkan umur bebas kemudian masukan berat badan bebas masukan tinggi badan bebas dan pilih jenis kelamin 'laki-laki'	Aplikasi akan menampilkan prediksi normal dengan warna hijau dan jika prediksi stunting akan berwarna merah	√	√	√					

3) Kesimpulan Hasil Pengujian *Black Box*

Berdasarkan pengujian *black box* dari 9 pengujian aplikasi yang didapat dari 3 responden, berikut hasil pengujian *black box* :

- Pengujian pertama

$$\text{Tercapai} : \frac{9}{9} \times 100\% = 100\%$$

$$\text{Gagal} : \frac{0}{9} \times 100\% = 0\%$$

- Pengujian kedua

$$\text{Tercapai} : \frac{9}{9} \times 100\% = 100\%$$

$$\text{Gagal} : \frac{0}{9} \times 100\% = 0\%$$

- Pengujian ketiga

$$\text{Tercapai} : \frac{9}{9} \times 100\% = 100\%$$

$$\text{Gagal} : \frac{0}{9} \times 100\% = 0\%$$

$$\text{Jumlah Presentase rata-rata tercapai} = \frac{300}{3} = 100\%$$

$$\text{Jumlah Presebtase rata-rata gagal} = \frac{0}{3} = 0\%$$

Berdasarkan penghitungan di atas, bahwa dari 9 pengujian oleh 3 responden menunjukkan tingkat keberhasilan pengujian *black box* memiliki presentase 100%, sedangkan yang gagal memiliki presentase 0%. Dapat disimpulkan aplikasi berjalan sesuai fungsionalnya.

b. Pengujian *White Box*

Pengujian *white box* merupakan salah satu teknik pengujian perangkat lunak yang melibatkan pemeriksaan *source code* dari program yang dibuat untuk diteliti apakah kode yang dibuat terdapat kesalahan atau tidak. Tujuan diadakan pengujian *white box* untuk memastikan bahwa setiap bagian dari kode berfungsi sebagaimana mestinya dan sesuai dengan spesifikasi yang ditetapkan sebelumnya. Berikut hasil pengujian *white box* dapat dilihat pada Tabel 4.4.

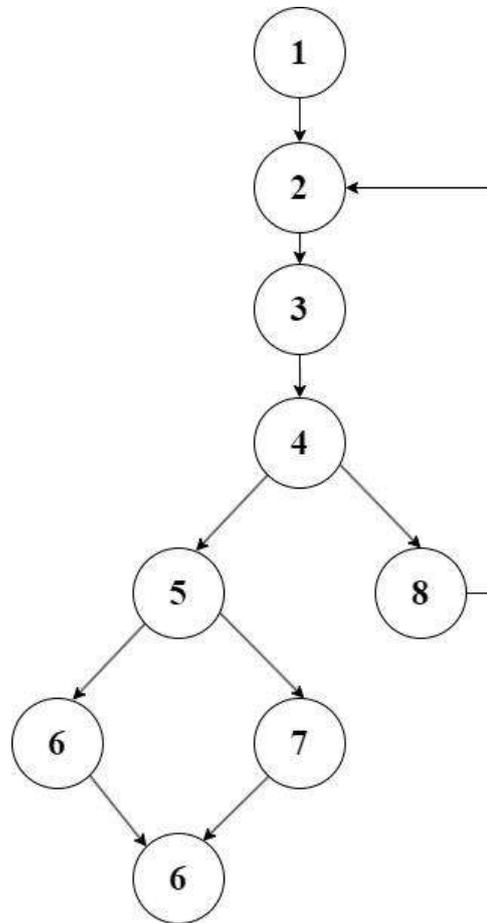
Tabel 4.4 pengujian *White Box*

<i>Node</i>	<i>Script</i>	<i>Penjelasan Code</i>
1	<pre>with open('model.pkl', 'rb') as model_file: model = pickle.load(model_file) with open('scaler.pkl', 'rb') as scaler_file: scaler = pickle.load(scaler_file)</pre>	<i>Load model</i>
2	<pre>with st.form(key='prediction_form'): umur_bulan = st.number_input('Umur (bulan)', min_value=-20, max_value=60, value=24, help="Masukkan usia anak dalam bulan.") berat_badan = st.number_input('Berat Badan (kg)', min_value=0.0, max_value=50.0, value=10.0, step=0.1, help="Masukkan berat badan anak dalam kilogram.") tinggi_badan = st.number_input('Tinggi Badan (cm)', min_value=-100, max_value=150, value=85, help="Masukkan tinggi badan anak dalam centimeter.") jenis_kelamin = st.selectbox('Jenis Kelamin', ['laki-laki', 'perempuan'], help="Pilih jenis kelamin anak.")</pre>	<i>Input data</i>
3	<pre>submit_button = st.form_submit_button(label='Prediksi')</pre>	<i>Button prediksi</i>
4	<pre>if submit_button: # Input validation if berat_badan <= 0: st.error("Berat badan harus lebih dari 0 kg.") elif tinggi_badan <= 0: st.error("Tinggi badan harus lebih dari 0 cm.") elif umur_bulan <= 0: st.error("Umur harus lebih dari 0 bulan.") else: jenis_kelamin_numeric = 1 if jenis_kelamin == 'laki-laki' else 0</pre>	<i>Sistem prediksi</i>

<i>Node</i>	<i>Script</i>	<i>Penjelasan Code</i>
	<pre> # Preprocess the input data data = [[umur_bulan, berat_badan, tinggi_badan, jenis_kelamin_numeric]] data_scaled = scaler.transform(data) # Make a prediction prediction = model.predict(data_scaled)[0] # Display the prediction if prediction == 0: st.success('Anak terdeteksi Normal') else: st.error('Anak tersebut terdeteksi Stunting SEGERA LAKUKAN PENANGANAN') </pre>	
5	prediction = model.predict(data_scaled)[0]	Proses prediksi
6	if prediction == 0: st.success('Anak terdeteksi Normal')	Hasil Prediksi Normal
7	else: st.error('Anak tersebut terdeteksi Stunting SEGERA LAKUKAN PENANGANAN')	Hasil Prediksi Stunting
8	if berat_badan <= 0: st.error("Berat badan harus lebih dari 0 kg.") elif tinggi_badan <= 0: st.error("Tinggi badan harus lebih dari 0 cm.") elif umur_bulan <= 0: st.error("Umur harus lebih dari 0 bulan.")	Prediksi error
9	End	Akhir Program

Selanjutnya dilakukan proses penggambaran dari alur script uji lift

pada sistem, berikut ini adalah flow diagram dari script penentuan nilai uji lift pada sistem.



Gambar 4.23 diagram alir

Berdasarkan diagram alir pada gambar 4.23 pengujian *white box* pada Gambar langkah yang akan dihitung *cyclomatic complexity* yaitu :

- a) Rumus pertama menghitung *cyclomatic complexity* dari *edge* (panah) dan *node* (lingkaran)

$$V(G) = E - N + 2$$

$$V(G) = 10 - 9 + 2 = 3$$

Keterangan :

E : Jumlah busur pada *flowgraph*

N : Jumlah simpul pada *flowgraph*

V(G) : *Cyclomatic complexity*

Dari hasil perhitungan perhitungan diatas, kompleksitas siklomatis yang dihasilkan adalah 3 *independent path*

Berdasarkan hasil dari *cyclomatic complexity*, dapat ditentukan *independent path* sebagai berikut :

Path 1 = 1-2-3-4-5-6-9

Path 2 = 1-2-3-4-5-7-9

Path 3 = 1-2-3-4-8

b) *Value Test*

Selanjutnya dilakukan pengujian *value test* yaitu sebagai pengujian pada setiap *independent path*.

Tabel 4.5 *Value Test*

NO	<i>Path</i>	<i>Input</i>	<i>Output</i>	Keterangan
1	1-2-3-4-5-6-9	Memasukan data normal	Menghasilkan prediksi normal	Berhasil
2	1-2-3-4-5-7-9	Memasukan data stunting	Menghasilkan prediksi stunting	Berhasil
3	1-2-3-4-8	Gagal melakukan prediksi pada sistem	Pesan kesalahan	Berhasil

Hasil pengujian pada Tabel 4,5 diperoleh hasil sebagai berikut :

$$\text{Tercapai} = 3/3 \times 100\% = 100\%$$

$$\text{Gagal} = 0/3 \times 100\% = 0\%$$

Berdasarkan pengujian *white box* yang telah dilakukan dengan menggunakan beberapa tahapan seperti *cyclomatic complexity*, *independent path*, dan *value test* bahwa pengujian ini memiliki keberhasilan 100% dan kegagalan mencapai 0%. Sehingga dapat ditarik kesimpulan bahwa pengujian *white box* ini berjalan dengan baik

c. Pengujian *User Acceptance Testing* (UAT)

User acceptance testing dilakukan untuk memastikan bahwa aplikasi klasifikasi sampah terurai dan tidak terurai memenuhi kebutuhan dan harapan *user*. Pengujian ini merupakan kesempatan bagi *user* untuk memberikan umpan balik dan memastikan bahwa aplikasi siap digunakan dalam lingkungan produksi atau operasional. *User acceptance testing* diujikan pada 5 responden dengan 15 pertanyaan. Kelayakan pada aplikasi dapat dilihat pada Tabel 4.6.

Tabel 4.6 Pengujian *User Acceptance Testing* (UAT)

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?					
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif					
	Apakah aplikasi ini					

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
4	bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?					
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?					
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?					
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?					
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?					
Angket Aspek <i>User Interface</i>						
9	Apakah tampilan pada aplikasi menarik?					
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?					
11	Apakah tata letak pada menu aplikasi sudah sesuai?					
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?					
Angket Aspek Kegunaan						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					
14	Apakah fitur pada aplikasi ini berguna dan relevan?					
15	Apakah aplikasi ini melakukan prediksi dengan tepat?					

Tabel 4.7 Bobot Penilaian Kuesioner

Jawaban	Skor	Presentase
Sangat Setuju	5	81%-100%
Setuju	4	61%-80%
Cukup Setuju	3	41%-60%
Kurang Setuju	2	21%-40%
Tidak Setuju	1	0%-20%

Berikut ini merupakan hasil kuesioner pengujian *user acceptance testing* yang telah disebarakan kepada 5 responden. Hasil pengujian *user acceptance testing* dapat dilihat pada Tabel 4.8.

Tabel 4.8 Hasil Pengujian *User Acceptance Testing* (UAT)

Pertanyaan	Hasil Pengujian				
	Responden 1	Responden 2	Responden 3	Responden 4	Responden 5
1	5	5	5	5	5
2	5	5	5	4	4
3	5	5	5	4	5
4	5	4	4	5	4
5	5	5	5	4	5
6	5	5	5	4	5
7	5	5	5	5	4
8	4	4	4	5	4
9	4	4	4	5	4
10	5	4	5	5	5
11	4	4	5	5	4
12	4	5	4	4	5
13	5	5	5	5	5

Pertanyaan	Hasil Pengujian				
	Responden 1	Responden 2	Responden 3	Responden 4	Responden 5
14	5	5	5	5	4
15	5	5	5	4	4
Jumlah Skor	71	70	71	69	67
Persentase	94%	93%	94%	92%	89%
Total	462%				

Dari hasil presentase setiap pertanyaan mulai dari aspek kemanfaatan, aspek kemudahan, aspek *user interface* (UI), dan aspek kegunaan yang diuji oleh 5 orang responden, kemudian dicari nilai rata-ratanya untuk mencapai tingkat penerimaan terhadap aplikasi yang dibuat oleh responden. Nilai rata-ratanya dapat dihitung dengan menggunakan persamaan sebagai berikut:

$$\text{Presentase Rata – rata} = \frac{\text{Jumlah Total Presentase}}{\text{Total Responden}}$$

$$\text{Presentase Rata – rata} = \frac{462\%}{5} = 92,4\%$$

Dengan perhitungan diatas didapatkan rata-rata dari keempat aspek diatas diperoleh sebesar 94,4%. Dapat disimpulkan bahwa pengujian UAT pada aplikasi ini dapatkan dikategorikan sangat baik.

B. Pembahasan

Dari hasil penelitian menggunakan metode *waterfall* di atas maka aplikasi *Early Warning Stunting* dengan 4 tahapan yang meliputi analisis kebutuhan, desain, implementasi, dan pengujian.

1. Analisis Kebutuhan

Pada tahapan pertama yaitu tahapan analisis kebutuhan. Pada tahapan ini diperlukan analisis kebutuhan yang dipakai untuk pengembangan aplikasi. Beberapa analisis kebutuhan yaitu analisis kebutuhan data, analisis kebutuhan sistem atau aplikasi, dan analisis kebutuhan fungsional. Dalam tahapan ini, kebutuhan untuk aplikasi early warning stunting yang akan dikembangkan, dikumpulkan, dipahami serta diraikan secara rinci.

2. Desain

Setelah pengumpulan kebutuhan aplikasi selesai, tahapan berikutnya yaitu desain. Dalam tahapan ini, penulis menggunakan model perancangan *Unified Modelling Language* (UML). Penulis menggunakan 3 jenis diagram UML. Pertama *use case diagram* pada diagram ini *user* sebagai aktor yang mempunyai akses penuh dalam mengakses halaman prediksi, penginputan data dan melakukan prediksi stunting. Kedua penulis menggunakan *activity diagram* terdapat hanya satu *activity diagram* yang berfungsi untuk mengakses halaman prediksi hingga melakukan prediksi pada aplikasi. Kemudian yang terakhir penulis menggunakan *sequence diagram* yang digunakan sebagai penggambaran interaksi antar objek-objek dalam sebuah sistem secara berurutan waktu, terdapat satu *sequence diagram* yang menggambarkan pengurutan interaksi *user* dengan objek-objek sehingga mendapatkan hasil prediksi yang sesuai.

Selain menggunakan model perancangan *Unified Modelling Language* (UML) penulis juga menggunakan desain antarmuka atau *user interface* (UI). Pada desain antarmuka penulis merancang sebuah desain jenis wireframe yang dimana desain masih berwarna hitam putih dan tidak kompleks. Pada desain antarmuka penulis merancang sebuah desain halaman menu prediksi

3. Implementasi

Tahapan selanjutnya yaitu tahapan implementasi. Pada tahapan

ini penulis menggunakan model implementasi sistem. Yang dimana *output* pada pembuatan model pada penelitian ini yaitu ekstensi file .pkl yang terdapat didalam implementasi sistem, dimana pengembangan sistem atau aplikasinya berawal dari pembuatan streamlit dan diakhiri dengan hasil aplikasi yang telah dikembangkan dan menghasilkan sebuah akurasi sebesar 0,96 yang berarti model berhasil memprediksi 96% dengan dua data yaitu berasal dari desa Tamanrejo dan juga Kaggle karena saat menggunakan data dari desa Tamanrejo saja hanya menghasilkan akurasi sebesar 40%.

4. Pengujian

Tahapan terakhir yaitu pengujian yang dilakukan untuk memastikan aplikasi berfungsi dengan baik. Penulis melakukan 3 pengujian yaitu pengujian *Black Box*, pengujian *White Box*, dan pengujian *User Acceptance Testing* (UAT). Pada pengujian *Black Box* mendapatkan hasil presentasi 100% untuk pengujian berhasil, sedangkan pengujian gagal mendapatkan presentase 0% dari 3 responden dan 9 pengujian yang diakhiri saran yang terdapat dapat dalam lampiran. Kemudian terdapat pengujian *White Box* mendapatkan hasil yang memuaskan dimana mendapatkan presentase 100% dengan pengujian yang tercapai, sedangkan untuk pengujian gagal mendapatkan presentase 0% dari *value test* yang terdiri dari tiga *path*. Yang terakhir penulis melakukan pengujian *User Acceptance Testing* (UAT) dengan mendapatkan hasil sangat baik dengan presentase sebesar 92,4% dari 5 responden dan 15 pertanyaan.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan penelitian yang dilakukan oleh penulis, maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Dari hasil *testing* saat menggunakan model SVM menghasilkan presentase akurasi sebanyak 96% dengan menggunakan sample data *testing* yang berjumlah 510 data, dengan nilai rata – rata *precision*, *recal*, dan *F1-score* sebesar 96%.
2. Pengujian *black box* yang dilakukan oleh 3 responden dengan 9 pengujian dengan mengujikan halaman utama pada menu prediksi menghasilkan tingkat keberhasilan sebesar 100% dan tingkat kegagalan sebesar 0%.
3. Pengujian *white box* menggunakan teknik *path testing* yang dilakukan oleh peneliti. Menu yang diuji merupakan menu prediksi. Hasil dari pengujian ini memperoleh presentase 100% tercapai dengan melakukan pengujian pada 3 *path* atau jalur
4. Pengujian *User Acceptance Testing* (UAT) yang dilakukan oleh 5 responden dengan 15 pertanyaan. Aspek yang ditanyakan adalah aspek kemanfaatan, aspek kemudahan, aspek *user interface* (UI), dan aspek kegunaan. Hasil dari pengujian ini memperoleh presentase sebesar 94,4% dengan kategori sangat baik.
5. Pada penelitian ini memiliki potensi sebagai alat untuk mendeteksi secara dini kasus stunting pada baduta sehingga dapat dilakukan penanganan yang lebih lanjut oleh dokter

B. Saran

Berdasarkan penelitian yang telah dilakukan, terdapat saran yang penulis berikan sebagai pertimbangan untuk penelitian selanjutnya. Penulis berharap untuk mengimplementasikan di *mobile platform* seperti Android ataupun iOS, sehingga dapat diakses dengan lebih mudah melalui *smartphone*. Hal ini diharapkan dapat meningkatkan kemudahan dan aksesibilitas bagi pengguna serta memperluas jangkauan dan dampak dari penelitian ini.

DAFTAR PUSTAKA

- [1] Jatengprov, “Komitmen Jateng Turunkan Stunting Tuai Apresiasi BKKBN”, 13 Desember 2022, [Online] Tersedia: <https://jatengprov.go.id/publik/komitmen-jateng-turunkan-stunting-tuai-apresiasi-bkkbn/> [Diakses 16 Juli 2024]
- [2]P. D. Tamanrejo, “Profil Desa Tamanrejo.”, 7 September 2022, [Online] Tersedia : <https://tamanrejo.kendalkab.go.id/public/menu/profil> [Diakses 16 Juli 2024]
- [3]I. P. Putri, T. Terttiaavini, and N. Arminarahmah, “Analisis Perbandingan Algoritma Machine Learning untuk Prediksi Stunting pada Anak,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 1, pp. 257–265, 2024, doi: 10.57152/malcom.v4i1.1078.
- [4]A. I. Putri and Y. Syarif, “Implementation of Decision Tree and Support Vector Machine (SVM) Algorithm for Stunting Risk Prediction Implementasi Algoritma Decision Tree dan Support Vector Machine (SVM) untuk Prediksi Risiko Stunting pada Keluarga,” vol. 3, no. October, pp. 349–357, 2023.
- [5]R. K. Dinata, U. Malikussaleh, N. Hasdyna, U. Islam, and K. Indonesia, “Machine Learning,” no. August, 2020.
- [6]S. Nidhra, “Black Box and White Box Testing Techniques - A Literature Review,” *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012, doi: 10.5121/ijesa.2012.2204.
- [7]P. Apriyani, A. R. Dikananda, and I. Ali, “Penerapan Algoritma K-Means dalam Klasterisasi Kasus Stunting Balita Desa Tegalwangi,” *Hello World J. Ilmu Komput.*, vol. 2, no. 1, pp. 20–33, 2023, doi: 10.56211/helloworld.v2i1.230.
- [8]R. H. Dananjaya, S. Sutrisno, and F. A. Wellianto, “Akurasi Penggunaan Metode Support Vector Machine Dalam Prediksi Penurunan Pondasi Tiang,” *Matriks Tek. Sipil*, vol. 10, no. 3, p. 298, 2022, doi: 10.20961/mateksi.v10i3.64519.
- [9]F. Fitriani and D. Darmawi, “Hubungan Pengetahuan Dan Sikap Ibu

- Dengan Kejadian Stunting Pada Balita Di Desa Arongan Kecamatan Kuala Pesisir Kabupaten Nagan Raya,” *J. Biol. Educ.*, vol. 10, no. 1, pp. 23–32, 2022, doi: 10.32672/jbe.v10i1.4114.
- [10]U. I. Indonesia, “Membuat Aplikasi Web Sains Data Dengan Mudah Menggunakan Streamlit.” <https://informatics.uui.ac.id/2021/03/15/streamlit-membuat-aplikasi-web-sains-data/>
- [11]Revou.co, “Google Colab.”, [Online] Tersedia : <https://revou.co/kosakata/google-colab> [Diakses 16 Juli 2024]
- [12]A. S. Abdullah, H. Setiawan, N. Ummi, J. Industri, U. Sultan, and A. Tirtayasa, “Perancangan Sistem Informasi Berbasis Website dengan Metode Framework For The Applications of System Thinking,” vol. 1, no. 4, pp. 358–367, 2013.
- [13]O. K. K. Juman and U. Modeling, “Konsepsi dasar UML (Unified Modelling Language),” pp. 1–13.
- [14]A. R. L. Francisco, “Konsep Dasar Sistem,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013, [Online]. Available: <https://repository.bsi.ac.id/index.php/unduh/item/1726/9.-BAB-II-LANDASAN-TEORI.pdf>.
- [15]D. Febiharsa, “Uji Fungsionalitas (Blackbox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (Silsp) Batik Dengan Appperfect Web Test Dan Uji Pengguna Bahwa Black-Box Testing Merupakan Pengujian Perangkat Lunak Yang Merupakan Eksternal Sedangkan Blackbox Testin,” Vol. 1, Pp. 117–126, 2019.
- [16]A. R. Yusmita, H. Anra, and H. Novriando, “Sistem Informasi Pelatihan pada Kantor Unit Pelaksana Teknis Latihan Kerja Industri (UPT LKI) Provinsi Kalimantan Barat,” *J. Sist. dan Teknol. Inf.*, vol. 8, no. 2, p. 160, 2020, doi: 10.26418/justin.v8i2.36797.
- [17]A. Abdul Wahid, “Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi,” *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp. 1–5, 2020.

- [18] A. P. Wibawa, M. Guntur, A. Purnama, M. Fathony Akbar, and F. A. Dwiyanto, "Metode-metode Klasifikasi," *Prosiding Seminar Ilmu Komputer dan Teknologi Informasi*, vol. 3, no. 1, 2018.
- [19] P. R. Sihombing and I. F. Yuliati, "Penerapan Metode Machine Learning dalam Klasifikasi Risiko Kejadian Berat Badan Lahir Rendah di Indonesia," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 20, no. 2, pp. 417–426, May 2021, doi: 10.30812/matrik.v20i2.1174
- [20] Suwanto Raharjo. 2014. *Kolaborasi PHP 5 Dan MySql Untuk Pengembangan Website*. Jakarta : Andi Publisher.
- [21] Agung, Gregorius. 2000. *Microsoft Frontpage 2000 Webbot*. Jakarta: PT. Elex Media Komputindo.
- [22] Hakim, Lukmanul dan Uus Musalini. 2004. *Cara Cerdas Menguasai Layout, Desain dan Aplikasi Web*. Jakarta: PT Elex Media Komputindo.
- [23] Rahmadhika, M. K., & Thantawi, A. M. (2021). Rancang Bangun Aplikasi Face Recognition Pada Pendekatan CRM Menggunakan Opencv Dan Algoritma Haarcascade. *IKRA-ITH Informatika : Jurnal Komputer Dan Informatika*, 5(1), Article 1.
- [24] O. Bangun, H. Mawengkang, and S. Efendi, "Metode Algoritma Support Vector Machine (SVM) Linier Dalam Memprediksi Kelulusan Mahasiswa," *Jurnal Media Informatika Budidarma*, vol. 6, no. 4, p. 2006, Oct. 2022, doi: 10.30865/mib.v6i4.4572.
- [25] Y. Pramana, I. Fradianto, M. A. Maulana, and S. Fauzan, "Pengembangan Aplikasi Early Warning System Di Rumah Sakit Universitas Tanjungpura Kalimantan Barat," *J. Penelit. Keperawatan*, vol. 8, no. 1, pp. 55–62, 2022, doi: 10.32660/jpk.v8i1.581.

LAMPIRAN

Lampiran 1 Pengujian *Black Box*

LEMBAR PENGUJIAN BLACK BOX
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)

A. Identitas Penguji

Nama : Febrina MURTI D. SE, M. Com.
 Jabatan :
 Tanggal Pengujian : 31 Juli 2024.

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
 NPM : 20670026

C. Tabel Pengujian

1. Pengujian Black Box User

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
Input valid	Masukan umur 24 bulan kemudian masukan berat badan 35 kg masukan tinggi badan 70 cm pilih gender 'Perempuan' lalu klik tombol pada prediksi	Hasil Prediksi Stunting akan menampilkan prediksi normal	✓	
Input invalid	Masukan umur -20 sampai -100 bulan kemudian berat 20 kg - 30 kg lalu tinggi badan 50-70 cm pilih gender	Aplikasi menangani input invalid dan menampilkan pesan error umur harus	✓	

	'Laki-laki' lalu klik tombol pada prediksi	lebih dari 0 bulan dan aplikasi tidak melakukan prediksi.		
Input invalid	Masukan Umur 40 bulan kemudian masukan berat 0 kg sampai -30kg lalu tinggi badan 100 cm dan pilih gender 'perempuan'	Aplikasi menangani input invalid dan menampilkan pesan error berat badan harus lebih dari 0 kg dan aplikasi tidak melakukan prediksi.	✓	
Input invalid	Masukan Umur 25-30 bulan kemudian berat 25-36 kg Masukan tinggi badan 0 cm sampai -90 cm dan pilih gender 'Male' Klik tombol pada prediksi	Aplikasi menangani input invalid dan menampilkan pesan error tinggi badan harus lebih dari 0 cm dan aplikasi tidak melakukan prediksi.	✓	

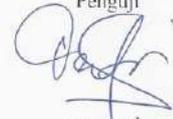
Input maksimal	Masukkan Umur 60 kemudian masukan berat 50 kg lalu tinggi 150 cm dan pilih gender 'laki-laki'	Aplikasi memberikan prediksi yang sesuai.	✓	
Input kombinasi	Masukkan Umur 30 lalu masukan berat 20,50 kg kemudian masukan tinggi 20 cm dan pilih gender 'perempuan'	Aplikasi memberikan prediksi yang sesuai.	✓	

Respons Aplikasi	Masukkan data lakukan prediksi secara berulang-ulang secara singkat	Aplikasi akan menampilkan hasil prediksi yang sama secara cepat	✓	
Konsistensi	Masukkan data yang diisi dengan nilai yang sama	Aplikasi akan menampilkan hasil prediksi sesuai dengan data yang diisi	✓	
Tampilan	Masukkan umur bebas kemudian masukan berat badan bebas masukan tinggi badan bebas dan pilih jenis kelamin 'laki-laki'	Aplikasi akan menampilkan prediksi normal dengan warna hijau dan jika prediksi stunting akan berwarna merah	✓	

D. Saran dan Masukan

-

Penguji



Febrian.

(.....)

LEMBAR PENGUJIAN BLACK BOX
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)

A. Identitas Penguji

Nama : Nur Latifah Dwi MS, MKom
 Jabatan : Dosen
 Tanggal Pengujian : 30 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
 NPM : 20670026

C. Tabel Pengujian

1. Pengujian Black Box User

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
Input valid	Masukan umur 24 bulan kemudian masukan berat badan 35 kg masukan tinggi badan 70 cm pilih gender 'Perempuan' lalu klik tombol pada prediksi	Hasil Prediksi Stunting akan menampilkan prediksi normal	✓	
Input invalid	Masukan umur -20 sampai -100 bulan kemudian berat 20 kg - 30 kg lalu tinggi badan 50-70 cm pilih gender	Aplikasi menangani input invalid dan menampilkan pesan error umur harus	✓	

	'Laki-laki' lalu klik tombol pada prediksi	lebih dari 0 bulan dan aplikasi tidak melakukan prediksi.		
Input invalid	Masukan Umur 40 bulan kemudian masukan berat 0 kg sampai -30kg lalu tinggi badan 100 cm dan pilih gender 'perempuan'	Aplikasi menangani input invalid dan menampilkan pesan error berat badan harus lebih dari 0 kg dan aplikasi tidak melakukan prediksi.	✓	
Input invalid	Masukan Umur 25-30 bulan kemudian berat 25-36 kg Masukan tinggi badan 0 cm sampai -90 cm dan pilih gender 'Male' Klik tombol pada prediksi	Aplikasi menangani input invalid dan menampilkan pesan error tinggi badan harus lebih dari 0 cm dan aplikasi tidak melakukan prediksi.	✓	

Input maksimal	Masukkan Umur 60 kemudian masukan berat 50 kg lalu tinggi 150 cm dan pilih gender 'laki-laki'	Aplikasi memberikan prediksi yang sesuai.	✓	
Input kombinasi	Masukkan Umur 30 lalu masukan berat 20,50 kg kemudian masukan tinggi 20 cm dan pilih gender 'perempuan'	Aplikasi memberikan prediksi yang sesuai.	✓	

Respons Aplikasi	Masukkan data lakukan prediksi secara berulang-ulang secara singkat	Aplikasi akan menampilkan hasil prediksi yang sama secara cepat	✓	
Konsistensi	Masukkan data yang diisi dengan nilai yang sama	Aplikasi akan menampilkan hasil prediksi sesuai dengan data yang diisi	✓	
Tampilan	Masukkan umur bebas kemudian masukan berat badan bebas masukan tinggi badan bebas dan pilih jenis kelamin 'laki-laki'	Aplikasi akan menampilkan prediksi normal dengan warna hijau dan jika prediksi stunting akan berwarna merah	✓	

D. Saran dan Masukan

Penguji


(Nur Lailiyah Bani MS, Mkom...)

LEMBAR PENGUJIAN BLACK BOX
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)

A. Identitas Penguji

Nama : Setyaningsih W
 Jabatan : Dosen
 Tanggal Pengujian : 31 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widianto
 NPM : 20670026

C. Tabel Pengujian

1. Pengujian Black Box User

Uji Coba	Skenario Uji Coba yang diharapkan	Hasil Pengamatan	Keterangan	
			Diterima	Ditolak
Input valid	Masukan umur 24 bulan kemudian masukan berat badan 35 kg masukan tinggi badan 70 cm pilih gender 'Perempuan' lalu klik tombol pada prediksi	Hasil Prediksi Stunting akan menampilkan prediksi normal	✓	
Input invalid	Masukan umur -20 sampai -100 bulan kemudian berat 20 kg - 30 kg lalu tinggi badan 50-70 cm pilih gender	Aplikasi menangani input invalid dan menampilkan pesan error umur harus	✓	

	'Laki-laki' lalu klik tombol pada prediksi	lebih dari 0 bulan dan aplikasi tidak melakukan prediksi.		
Input invalid	Masukan Umur 40 bulan kemudian masukan berat 0 kg sampai -30kg lalu tinggi badan 100 cm dan pilih gender 'perempuan'	Aplikasi menangani input invalid dan menampilkan pesan error berat badan harus lebih dari 0 kg dan aplikasi tidak melakukan prediksi.	✓	
Input invalid	Masukan Umur 25-30 bulan kemudian berat 25-36 kg Masukan tinggi badan 0 cm sampai -90 cm dan pilih gender 'Male' Klik tombol pada prediksi	Aplikasi menangani input invalid dan menampilkan pesan error tinggi badan harus lebih dari 0 cm dan aplikasi tidak melakukan prediksi.	✓	

Input maksimal	Masukkan Umur 60 kemudian masukan berat 50 kg lalu tinggi 150 cm dan pilih gender 'laki-laki'	Aplikasi memberikan prediksi yang sesuai.	✓	
Input kombinasi	Masukkan Umur 30 lalu masukan berat 20,50 kg kemudian masukan tinggi 20 cm dan pilih gender 'perempuan'	Aplikasi memberikan prediksi yang sesuai.	✓	

Respons Aplikasi	Masukkan data lakukan prediksi secara berulang-ulang secara singkat	Aplikasi akan menampilkan hasil prediksi yang sama secara cepat	✓	
Konsistensi	Masukkan data yang diisi dengan nilai yang sama	Aplikasi akan menampilkan hasil prediksi sesuai dengan data yang diisi	✓	
Tampilan	Masukkan umur bebas kemudian masukan berat badan bebas masukan tinggi badan bebas dan pilih jenis kelamin 'laki-laki'	Aplikasi akan menampilkan prediksi normal dengan warna hijau dan jika prediksi stunting akan berwarna merah	✓ /	

D. Saran dan Masukan

tampilan dibuat lebih responsif

Penguji


(...Setyaningsih Wibowo...)

Lampiran 2 lembar Pengujian UAT

**LEMBAR PENGUJIAN UAT (USER ACCEPTANCE TESTING)
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

A. Identitas Penguji

Nama : Muhammad Afrizal Prero
Tanggal Pengujian : Selasa, 30 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
NPM : 20670026

C. Tabel Pengujian

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					✓
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?					✓
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif					✓
4	Apakah aplikasi ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?				✓	
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?					✓
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?					✓
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?					✓

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?				✓	
Angket Aspek <i>User Interface</i>						
9	Apakah tampilan pada aplikasi menarik?				✓	
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?					✓
11	Apakah tata letak pada menu aplikasi sudah sesuai?					✓
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?				✓	
Angket Aspek Kegunaan						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					✓
14	Apakah fitur pada aplikasi ini berguna dan relevan?					✓
15	Apakah aplikasi ini melakukan prediksi dengan tepat?					✓

Penguji


 (Muhamad Afrizal Prero)

**LEMBAR PENGUJIAN UAT (USER ACCEPTANCE TESTING)
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

A. Identitas Penguji

Nama : Fatih Darrowul Jannah
Tanggal Pengujian : 30 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
NPM : 20670026

C. Tabel Pengujian

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					✓
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?					✓
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif					✓
4	Apakah aplikasi ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?					✓
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?					✓
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?					✓
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?					✓

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?				✓	
<i>Angket Aspek User Interface</i>						
9	Apakah tampilan pada aplikasi menarik?				✓	
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?					✓
11	Apakah tata letak pada menu aplikasi sudah sesuai?				✓	
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?				✓	
<i>Angket Aspek Kegunaan</i>						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					✓
14	Apakah fitur pada aplikasi ini berguna dan relevan?					✓
15	Apakah aplikasi ini melakukan prediksi dengan tepat?					✓

Penguji


 (..... Fauzi D.J)

**LEMBAR PENGUJIAN UAT (USER ACCEPTANCE TESTING)
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

A. Identitas Penguji

Nama : Ri2al Ashari
Tanggal Pengujian : 30 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
NPM : 20670026

C. Tabel Pengujian

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					✓
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?					✓
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif					✓
4	Apakah aplikasi ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?				✓	
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?					✓
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?					✓
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?					✓

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?				✓	
<i>Angket Aspek User Interface</i>						
9	Apakah tampilan pada aplikasi menarik?				✓	
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?				✓	
11	Apakah tata letak pada menu aplikasi sudah sesuai?				✓	
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?					✓
<i>Angket Aspek Kegunaan</i>						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					✓
14	Apakah fitur pada aplikasi ini berguna dan relevan?					✓
15	Apakah aplikasi ini melakukan prediksi dengan tepat?					✓

Penguji


 (.....Rizka Fikriani.....)

**LEMBAR PENGUJIAN UAT (USER ACCEPTANCE TESTING)
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

A. Identitas Penguji

Nama : Diva Dewana P
Tanggal Pengujian : 30 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto
NPM : 20670026

C. Tabel Pengujian

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					✓
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?				✓	
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif					✓
4	Apakah aplikasi ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?				✓	
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?					✓
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?					✓
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?				✓	

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?				✓	
Angket Aspek User Interface						
9	Apakah tampilan pada aplikasi menarik?				✓	
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?					✓
11	Apakah tata letak pada menu aplikasi sudah sesuai?				✓	
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?					✓
Angket Aspek Kegunaan						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					✓
14	Apakah fitur pada aplikasi ini berguna dan relevan?				✓	
15	Apakah aplikasi ini melakukan prediksi dengan tepat?				✓	

Penguji

(.....
Riva Dewana P.....)

**LEMBAR PENGUJIAN UAT (USER ACCEPTANCE TESTING)
EARLY WARNING STUNTING DENGAN ALGORITMA SUPPORT
VECTOR MACHINE (SVM)**

A. Identitas Penguji

Nama : Nanfal Agung Rizkha

Tanggal Pengujian : 31 Juli 2024

B. Identitas Peneliti

Nama : Adam Virgiawan Widiyanto

NPM : 20670026

C. Tabel Pengujian

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
Angket Aspek Kemanfaatan						
1	Apakah aplikasi ini ini bermanfaat bagi user dari segi Produktivitas?					✓
2	Apakah aplikasi ini ini bermanfaat bagi user baik dari segi Kinerja?				✓	
3	Apakah aplikasi ini bermanfaat bagi user baik dari segi Efektif				✓	
4	Apakah aplikasi ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses?					✓
Angket Aspek Kemudahan						
5	Sejauh mana anda merasa aplikasi ini mudah untuk dipahami dan digunakan tanpa bantuan tambahan?				✓	
6	Apakah aplikasi Early Warning Stunting ini mudah digunakan?				✓	
7	Apakah aplikasi ini cukup mudah dipahami dan dijalankan?					✓

No	Pertanyaan	Skor				
		Tidak Setuju	Kurang Setuju	Cukup Setuju	Setuju	Sangat Setuju
8	Apakah aplikasi Early Warning Stunting ini mudah dipahami?					✓
Angket Aspek User Interface						
9	Apakah tampilan pada aplikasi menarik?					✓
10	Apakah font tulisan pada aplikasi dapat terbaca dengan baik?					✓
11	Apakah tata letak pada menu aplikasi sudah sesuai?					✓
12	Apakah penggunaan warna pada aplikasi sudah terlihat nyaman dan sesuai dengan pengguna?				✓	
Angket Aspek Kegunaan						
13	Seberapa efektif menurut Anda fitur dalam membantu untuk memprediksi stunting?					✓
14	Apakah fitur pada aplikasi ini berguna dan relevan?					✓
15	Apakah aplikasi ini melakukan prediksi dengan tepat?				✓	✓

Penguji


 (Nuzul Anung R.)

Lampiran 3 Lembar Pembimbingan skripsi



UNIVERSITAS PGRI SEMARANG
FAKULTAS TEKNIK DAN INFORMATIKA

Kampus : Jalan Sidodadi Timur No 24 Dr. Cipto, Semarang - Indonesia 50125

Telp: (024) 8316377, Faks: (024) 8448217 E-mail: upgrismg@gmail.com Homepage: www.upgrismg.ac.id

LEMBAR PEMBIMBINGAN SKRIPSI

Nama Mahasiswa : Adam Virgiawan Widiyanto
 N P M : 20670026
 Program Studi : Informatika
 Judul Skripsi : EARLY WARNING STUNTING DENGAN ALGORITMA
 SUPPORT VECTOR MACHINE (SVM)
 Dosen Pembimbing I : Aris Tri Joko Harjanto, S.Kom., M.Kom
 Dosen Pembimbing II : Neeta Gobun Naha, S.T., M.Eng

No	Hari, tanggal	Uraian Bimbingan	Paraf
1	2 Mei 2024	awal BAB I AOC BAB I	
2	9 Juni 2024	BAB II BAB III	
3	10 Juli 2024	BAB IV penyusunan BAB V	
4	8 Agustus 2024	AOC Ujian T.A.	

Dosen Pembimbing I,

Aris Tri Joko Harjanto, S.Kom., M.Kom.
 NIDN. 0619048202

Mahasiswa,

Adam Virgiawan W
 NPM. 20670026



UNIVERSITAS PGRI SEMARANG
FAKULTAS TEKNIK DAN INFORMATIKA

Kampus : Jalan Sidodadi Timur No 24 Dr. Cipto, Semarang - Indonesia 50125

Telp: (024) 8316377, Faks: (024) 8448217 E-mail: upgrismg@gmail.com Homepage: www.upgrismg.ac.id

LEMBAR PEMBIMBINGAN SKRIPSI

Nama Mahasiswa : Adam Virgiawan Widianto
NPM : 20670026
Program Studi : Informatika
Judul Skripsi : EARLY WARNING STUNTING DENGAN ALGORITMA
SUPPORT VECTOR MACHINE (SVM)

Dosen Pembimbing I : Aris Tri Joko Harjanto, S.Kom., M.Kom

Dosen Pembimbing II : Noora Qotrun Nada, S.T., M.Eng

No	Hari, tanggal	Uraian Bimbingan	Paraf
6	8 Agustus 2024	acc. u/ dilanjutkan ke sibang skripsi!	 

Dosen Pembimbing II,

Noora Qotrun Nada, S.T., M.Eng.
NPP. 158201485

Mahasiswa,

Adam Virgiawan W
NPM. 19670042

Lampiran 4 Lembar Permohonan Ijin Penelitian

 **UNIVERSITAS PGRI SEMARANG**
FAKULTAS TEKNIK DAN INFORMATIKA
Kampus : Jl. Sidodadi Timur No 24, Dr. Cipto, Semarang – Indonesia 50125
Telp.(024)8452230, Faks.(024)844217, E-mail:fti@upgris.ac.id. Website:\lfti.upgris.ac.id

Nomor : 1177 /AM/FTI/XII/2023
Lamp. : --
Hal : Permohonan ijin penelitian

7 Desember 2023

Kepada Yth.
Kepala Desa Tamanrejo Kecamatan Limbangan
Desa Tamanrejo Kecamatan Limbangan
Kab. Kendal

Kami beritahukan dengan hormat, bahwa mahasiswa kami:

NO.	N P M	NAMA MAHASISWA	PROGRAM STUDI
1.	20670026	Adam Virgiawan Widianto	Informatika
2.			
3.			
4.			
5.			

Akan mengadakan penelitian dengan judul:

PREDIKSI KASUS STUNTING PADA BALITA

Sehubungan dengan hal tersebut kami mohon perkenan Bapak/Ibu memberikan ijin bagi mahasiswa tersebut untuk melakukan penelitian.

Atas perkenan dan kerjasama yang baik, kami ucapkan terima kasih.

Dekan,



IBNU TOTO HUSODO, S.T., M.T.
NPP. 136901387

Lampiran 5 Lembar Validasi Ahli

LEMBAR VALIDASI AHLI

Nama Validator : dr. Dian Ayu Z. M. Glzi
 Instansi : Universitas PGRI Semarang

Nama Sistem/ Aplikasi : *Early Warning Stunting*
 Penyusun : Adam Virgiawan W.
 Aris Tri Jaka Harjanta, S.Kom., M.Kom.
 Noora Qotrun Nada, S.T., M.Eng.
 Instansi : Universitas PGRI Semarang

Dengan hormat,
 Sehubungan dengan dibuatnya Aplikasi Early Warning Stunting, kami memohon kesediaan Bapak/ Ibu untuk memberikan penilaian terhadap Aplikasi yang dibuat tersebut. Angket penilaian ini dimaksudkan untuk mengetahui pendapat Bapak/ Ibu tentang Aplikasi Early Warning Stunting yang kami buat, sehingga dapat diketahui layak atau tidaknya Aplikasi Early Warning Stunting ini. Penilaian, komentar dan saran yang Bapak/ Ibu berikan akan digunakan sebagai indikator kualitas dan pertimbangan untuk memperbaiki aplikasi Early Warning Stunting yang kami buat. Atas perhatian dan kesediaan untuk mengisi angket penilaian aplikasi Early Warning Stunting, kami ucapkan terima kasih.

Petunjuk Pengisian:

Penilaian dilakukan dengan memberikan tanda () pada kolom yang sesuai dengan penilaian Bapak/ Ibu untuk setiap butir dala lembar penilaian dibawah ini.

Aspek Penilaian

a. Kemanfaatan (Usefulness)

No	Pertanyaan	Sangat Setuju (5)	Setuju (4)	Tidak Pasti (3)	Tidak Setuju (2)	Sangat Tidak Setuju (1)	Skor
1	Apakah Aplikasi Early Warning Stunting ini bermanfaat bagi user dari segi Produktivitas kesehatan (Productivity)?		✓				
2	Apakah Aplikasi Early Warning Stunting ini bermanfaat bagi user baik dari segi Kinerja untuk kesehatan (Performance)?		✓				
3	Apakah Aplikasi Early Warning Stunting ini bermanfaat bagi user baik dari segi Efektif untuk kesehatan (Effective)?		✓				
4	Apakah Aplikasi Early Warning Stunting ini bermanfaat bagi user baik dari segi kegunaan untuk pengelolaan proses kesehatan (Usefulness)?		✓				

b. Kemudahan Penggunaan (Ease of Use)

No	Pertanyaan	Sangat Setuju (5)	Setuju (4)	Tidak Pasti (3)	Tidak Setuju (2)	Sangat Tidak Setuju (1)	Skor
1	Apakah Aplikasi Early Warning Stunting ini mudah dipelajari?		✓				
2	Apakah Aplikasi Early Warning Stunting ini mudah diperintahkan?		✓				
3	Apakah Aplikasi Early Warning Stunting ini mudah digunakan untuk kesehatan?		✓				

c. Isi (Content)

No	Pertanyaan	Memuaskan (5)	Bagus (4)	Cukup (3)	Buruk (2)	Tidak Pernah (1)	Skor
1	Apakah isi Aplikasi Early Warning Stunting yang di hasilkan sudah dipenuhi?		✓				
2	Apakah informasi Aplikasi Early Warning Stunting sudah sesuai dengan kebutuhan?		✓				
3	Apakah informasi Aplikasi Early Warning Stunting sudah cukup?		✓				
4	Apakah informasi Aplikasi Early Warning Stunting sudah tepat ?		✓				

d. Akurasi (Accuracy)

No	Pertanyaan	Memuaskan (5)	Bagus (4)	Cukup (3)	Buruk (2)	Tidak Pernah (1)	Skor
1	Apakah Aplikasi Early Warning Stunting sudah akurat menghasilkan informasi dalam kesehatan?		✓				
2	Apakah sudah puas dengan akurasi Aplikasi Early Warning Stunting?		✓				

e. Bentuk (Format)

No	Pertanyaan	Memuaskan (5)	Bagus (4)	Cukup (3)	Buruk (2)	Tidak Pernah (1)	Skor
1	Apakah penyajian informasi sudah sesuai ?		✓				
2	Apakah informasi yang dihasilkan system/aplikasi sudah jelas?		✓				

f. Kemudahan (Ease)

No	Pertanyaan	Memuaskan (5)	Bagus (4)	Cukup (3)	Buruk (2)	Tidak Pernah (1)	Skor
1	Apakah Aplikasi Early Warning Stunting User friendly ?		✓				
2	Apakah Aplikasi Early Warning Stunting mudah diakses?		✓				

g. Ketepatan Waktu (Timeliness)

No	Pertanyaan	Memuaskan (5)	Bagus (4)	Cukup (3)	Buruk (2)	Tidak Pernah (1)	Skor
1	Apakah konten informasi Aplikasi Early Warning Stunting tampil dengan tepat waktu ?		✓				
2	Apakah informasi Aplikasi Early Warning Stunting up-to-date?		✓				

Komentar secara umum

Silahkan cek aplikasi sejenis dari kementerian - App "Primaku".

Kesimpulan:

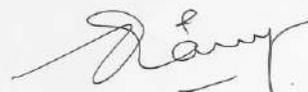
Aplikasi Early Warning Stunting ini dinyatakan:

- ① Layak diujicobakan di lapangan tanpa ada revisi
2. Layak diujicobakan di lapangan dengan revisi
3. Tidak layak diujicobakan di lapangan

Lingkari salah satu angka diatas.

Demikian angket ini saya isi dengan sebenarnya, tanpa ada pengaruh dari pihak manapun.

Semarang, 29 Juli 2024
Validator,



Dr. Dian Ayu Z, M. Gizi

Lampiran 6 Dataset

Umur (bulan)	Jenis Kelamin	Tinggi Badan (cm)	Status Gizi	Berat Badan (kg)
5	laki-laki	66.9	0	5.7
5	laki-laki	64.1	0	5.6
5	laki-laki	57.1	1	4.7
5	laki-laki	71.2	0	5.9
5	laki-laki	64.6	0	5.6
5	laki-laki	74.4	0	6.2
5	laki-laki	76.2	0	6.3
5	laki-laki	56	1	4.8
5	laki-laki	77.8	0	6.5
5	laki-laki	72.3	0	6.4
5	laki-laki	57.4	1	4.7
5	laki-laki	57	1	4.6
5	laki-laki	68.7	0	5.7
5	laki-laki	58.3	1	4.8
5	laki-laki	54	1	4.7
5	laki-laki	61	1	5.1
5	laki-laki	63.6	0	5.6
5	laki-laki	58.1	1	5.3
5	laki-laki	60.7	1	5.2
5	laki-laki	57.7	1	5
5	laki-laki	69	0	5.9
5	laki-laki	77.3	0	6.2
5	perempuan	63.7	0	5.6
5	perempuan	68.9	0	5.8
5	perempuan	68.2	0	5.7
5	perempuan	60.7	0	5.6
5	perempuan	76.7	0	6.1
5	perempuan	64.9	0	5.6
5	perempuan	56.1	1	4.7
5	perempuan	54.6	1	4.6
5	perempuan	60.7	0	5.6
5	perempuan	72.2	0	6.3
5	perempuan	52.3	1	4.8
5	perempuan	52.8	1	4.9
5	perempuan	54.6	1	5
5	perempuan	69	0	5.7
5	perempuan	71.1	0	6.3
5	perempuan	78.5	0	6.5
6	laki-laki	63.7	0	6.3

6	laki-laki	56.1	1	5.1
6	laki-laki	75.6	0	6.7
6	laki-laki	66	0	6.3
6	laki-laki	79.5	0	6.8
6	laki-laki	64.5	0	6.4
6	laki-laki	57.2	1	5.1
6	laki-laki	68.6	0	6.3
6	laki-laki	64.1	0	6.3
6	laki-laki	71.4	0	6.5
6	laki-laki	71.6	0	6.3
6	laki-laki	68.9	0	6.3
6	laki-laki	64.6	0	6.2
6	laki-laki	69	0	6.5
6	laki-laki	59.6	1	5.3
6	laki-laki	56.2	1	5.1
6	laki-laki	69.4	0	6.3
6	laki-laki	55.1	1	5.3
6	laki-laki	61.3	1	5.7
6	perempuan	72.7	0	6.8
6	perempuan	79.5	0	6.8
6	perempuan	81	0	6.9
6	perempuan	70	0	6.3
6	perempuan	69.2	0	6.2
6	perempuan	80.1	0	6.8
6	perempuan	67.1	0	6.3
6	perempuan	71.4	0	6.5
6	perempuan	80.5	0	7
6	perempuan	74.3	0	6.8
6	perempuan	61.7	0	6.4
6	perempuan	62.4	0	6.5
6	perempuan	80.9	0	7
6	perempuan	75.2	0	6.7
6	perempuan	66.1	0	6.4
6	perempuan	66.8	0	6.5
6	perempuan	77.4	0	6.8
6	perempuan	67.1	0	6.3
6	perempuan	76.3	0	6.8
6	perempuan	79.7	0	6.8
6	perempuan	78	0	6.6
7	laki-laki	78.3	0	7.3
7	laki-laki	70.6	0	6.7
7	laki-laki	72.2	0	6.7

7	laki-laki	69.5	0	6.6
7	laki-laki	55.6	1	5.8
7	laki-laki	69.3	0	6.8
7	laki-laki	75.8	0	7.3
7	laki-laki	59.3	1	5.6
7	laki-laki	70.7	0	6.8
7	laki-laki	57.7	1	6.3
7	laki-laki	69.8	0	6.8
7	laki-laki	74.3	0	6.6
7	laki-laki	73	0	6.9
7	laki-laki	59.9	1	5.4
7	laki-laki	70.4	0	6.7
7	laki-laki	64	1	6.2
7	laki-laki	73.2	0	6.7
7	laki-laki	61.4	1	5.8
7	laki-laki	71.7	0	6.7
7	laki-laki	70.8	0	6.8
7	laki-laki	79.8	0	7.1
7	laki-laki	61.4	1	5.8
7	laki-laki	82	0	7.4
7	laki-laki	78.2	0	7.2
7	laki-laki	62	1	5.8
7	laki-laki	67.3	0	6.8
7	perempuan	71.7	0	6.7
7	perempuan	58.2	1	5.7
7	perempuan	61.5	1	6.3
7	perempuan	71.7	0	6.6
7	perempuan	72.4	0	6.8
7	perempuan	77.7	0	7.4
7	perempuan	70.4	0	6.8
7	perempuan	74.7	0	7.4
7	perempuan	74.7	0	7.3
7	perempuan	54.3	1	5.8
7	perempuan	65.8	0	6.8
7	perempuan	70	0	6.7
7	perempuan	62.5	1	6.3
7	perempuan	60.1	1	5.8
7	perempuan	71.1	0	6.7
7	perempuan	65	0	6.6
7	perempuan	75.3	0	7.3
7	perempuan	72.2	0	6.7
7	perempuan	54.4	1	5.8

7	perempuan	79.6	0	7.3
7	perempuan	79.4	0	7.2
7	perempuan	69.9	0	6.4
7	perempuan	79.2	0	7.4
7	perempuan	54.9	1	5.7
7	perempuan	69.2	0	6.7
8	laki-laki	68.8	0	7.3
8	laki-laki	63.7	1	6.3
8	laki-laki	69.4	0	7.3
8	laki-laki	71.5	0	7.3
8	laki-laki	67.1	0	7.4
8	laki-laki	72.3	0	7.2
8	laki-laki	66	1	6.7
8	laki-laki	71.9	0	7.3
8	laki-laki	76.6	0	7.2
8	laki-laki	59.9	1	6.3
8	laki-laki	71.8	0	7.2
8	laki-laki	75.3	0	7.4
8	laki-laki	72.2	0	7.3
8	laki-laki	81.3	0	8
8	laki-laki	67	0	7.4
8	perempuan	68.3	0	7.2
8	perempuan	72.1	0	7.4
8	perempuan	63.6	1	6.9
8	perempuan	70.5	0	7.1
8	perempuan	72.9	0	7.5
8	perempuan	74.3	0	7.3
8	perempuan	64.4	0	7.2
8	perempuan	70.4	0	7.1
8	perempuan	78.4	0	8
8	perempuan	55.8	1	6.2
8	perempuan	73.4	0	7.3
8	perempuan	67.2	0	7.4
8	perempuan	60.9	1	6.3
8	perempuan	74.3	0	7.1
9	laki-laki	77.6	0	8.5
9	laki-laki	71	0	7.6
9	laki-laki	75.2	0	7.8
9	laki-laki	61.2	1	6.7
9	laki-laki	61.8	1	6.9
9	laki-laki	72.5	0	7.6
9	laki-laki	84.9	0	8.4

9	laki-laki	78.5	0	8.5
9	laki-laki	82.8	0	8.2
9	laki-laki	59	1	6.7
9	laki-laki	66.8	0	7.6
9	laki-laki	74.1	0	7.8
9	perempuan	82.8	0	8.1
9	perempuan	77.9	0	8.2
9	perempuan	80.4	0	8.4
9	perempuan	67.7	0	8
9	perempuan	59.8	1	7
9	perempuan	79.3	0	8.2
9	perempuan	75.8	0	7.6
9	perempuan	74	0	7.8
9	perempuan	79.7	0	8.2
9	perempuan	60.7	1	6.7
9	perempuan	76.5	0	7.7
9	perempuan	59.5	1	6.8
9	perempuan	80	0	8.4
9	perempuan	60	1	6.7
9	perempuan	64.7	1	7.3
9	perempuan	61.4	1	6.8
9	perempuan	81.5	0	8.4
10	laki-laki	77.6	0	8.4
10	laki-laki	78.5	0	8.2
10	laki-laki	77.8	0	8.4
10	laki-laki	83.2	0	8.8
10	laki-laki	84.1	0	8.9
10	laki-laki	60.5	1	7.4
10	laki-laki	73.1	0	8.4
10	laki-laki	85.5	0	8.6
10	perempuan	76.4	0	8.1
10	perempuan	83.9	0	8.8
10	perempuan	58.3	1	7.3
10	perempuan	69.1	0	8.4
10	perempuan	80.8	0	8.8
10	perempuan	64.5	1	7.6
10	perempuan	69	0	8.1
10	perempuan	69.6	0	8.4
10	perempuan	68.2	0	8.3
10	perempuan	82	0	8.8
10	perempuan	74.5	0	8.1
10	perempuan	78.6	0	8.5

10	perempuan	64.9	1	7.6
10	perempuan	62.1	1	7.4
10	perempuan	81.5	0	9
11	laki-laki	72	0	8.7
11	laki-laki	81.6	0	9.1
11	laki-laki	66.5	1	7.6
11	laki-laki	75.9	0	8.6
11	laki-laki	82.7	0	9.1
11	laki-laki	78.2	0	9
11	laki-laki	68.4	1	8.2
11	laki-laki	62.7	1	7.7
11	laki-laki	67.5	1	7.8
11	laki-laki	75	0	8.8
11	laki-laki	79	0	8.9
11	laki-laki	75	0	8.7
11	laki-laki	67.4	1	7.8
11	laki-laki	66.2	1	7.6
11	laki-laki	77.2	0	8.7
11	laki-laki	72	0	8.6
11	perempuan	62.7	1	7.8
11	perempuan	72.8	0	8.7
11	perempuan	66.5	1	8.3
11	perempuan	61.8	1	7.9
11	perempuan	82.7	0	9.5
11	perempuan	80.2	0	8.9
11	perempuan	67.4	1	8.4
11	perempuan	73.3	0	9
11	perempuan	78.6	0	8.8
11	perempuan	81.3	0	9.3
11	perempuan	72.4	0	8.9
11	perempuan	71.8	0	8.7
12	laki-laki	80.8	0	9.1
12	laki-laki	79.4	0	9.4
12	laki-laki	74.3	0	9.2
12	laki-laki	77	0	9.1
12	laki-laki	85.5	0	10
12	laki-laki	76.3	0	9.3
12	laki-laki	76.3	0	9.4
12	laki-laki	64.5	1	8.3
12	laki-laki	82.1	0	9.1
12	laki-laki	71.3	0	9.3
12	laki-laki	86.7	0	9.8

12	laki-laki	69.8	1	8.7
12	laki-laki	68.8	1	8.8
12	laki-laki	68.9	1	8.7
12	perempuan	69	0	9.4
12	perempuan	75.3	0	9.4
12	perempuan	73.3	0	9.3
12	perempuan	78	0	9.2
12	perempuan	80.1	0	9.3
12	perempuan	83.3	0	9.8
12	perempuan	80.6	0	9.2
12	perempuan	67.6	1	8.8
12	perempuan	66.3	1	8.9
12	perempuan	84.8	0	9.8
12	perempuan	76.9	0	9.2
12	perempuan	82.1	0	9.8
13	laki-laki	77.7	0	9.6
13	laki-laki	75.5	0	9.8
13	laki-laki	71.1	1	9.1
13	laki-laki	65.6	1	9.2
13	laki-laki	74.5	0	9.8
13	laki-laki	69.9	1	9.2
13	laki-laki	76	0	9.7
13	laki-laki	71.2	1	9.1
13	laki-laki	85.2	0	10.1
13	laki-laki	68.7	1	8.7
13	laki-laki	75	0	9.7
13	laki-laki	70.5	1	9.2
13	laki-laki	87.4	0	10.1
13	perempuan	73	0	9.8
13	perempuan	64.2	1	8.8
13	perempuan	76.4	0	9.8
13	perempuan	71.7	0	9.9
13	perempuan	75.2	0	9.7
13	perempuan	69.1	1	9.1
13	perempuan	74.1	0	9.8
13	perempuan	88	0	10.3
13	perempuan	68.1	1	9.3
13	perempuan	84.7	0	10.2
13	perempuan	73	0	9.7
13	perempuan	84.7	0	10.1
13	perempuan	66.9	1	8.8
14	laki-laki	76.5	0	10.3

14	laki-laki	66.9	1	9.3
14	laki-laki	78.6	0	10.1
14	laki-laki	76.9	0	10.3
14	laki-laki	72.5	1	9.8
14	laki-laki	71	1	9.7
14	laki-laki	72.6	1	9.9
14	laki-laki	83.3	0	10.3
14	laki-laki	69.8	1	9.3
14	laki-laki	89.4	0	10.6
14	laki-laki	69.3	1	9.8
14	laki-laki	67.3	1	9.1
14	laki-laki	82.7	0	10.3
14	laki-laki	74	0	10.2
14	laki-laki	76.2	0	10.3
14	laki-laki	68.8	1	9.1
14	perempuan	66.5	1	9.3
14	perempuan	72	0	10.3
14	perempuan	71.2	0	10.5
14	perempuan	70.3	1	9.8
14	perempuan	73.5	0	10.3
14	perempuan	72.9	0	10.1
14	perempuan	64	1	9.1
14	perempuan	70.4	1	9.8
14	perempuan	76.8	0	10.4
14	perempuan	75.4	0	10.2
14	perempuan	70.3	1	9.8
14	perempuan	86	0	11
14	perempuan	79.3	0	10.3
14	perempuan	87.4	0	10.6
14	perempuan	72.6	0	10.3
14	perempuan	80.6	0	10.4
14	perempuan	87.5	0	10.7
15	laki-laki	69.7	1	9.3
15	laki-laki	84.3	0	10.8
15	laki-laki	81.5	0	10.6
15	laki-laki	76.7	0	10.8
15	laki-laki	88.3	0	11.5
15	laki-laki	79.5	0	10.8
15	laki-laki	84.7	0	11
15	laki-laki	73.2	1	10.2
15	laki-laki	89.8	0	11.3
15	perempuan	83.8	0	10.7

15	perempuan	75.9	0	10.8
15	perempuan	76.7	0	11
15	perempuan	64.8	1	9.8
15	perempuan	87.8	0	11.2
15	perempuan	72.8	0	10.8
15	perempuan	69.7	1	10.1
15	perempuan	87.6	0	11.2
15	perempuan	72.5	0	10.8
16	laki-laki	76.9	0	11.1
16	laki-laki	88.9	0	11.6
16	laki-laki	71	1	10.5
16	laki-laki	90.8	0	11.6
16	laki-laki	86.2	0	11.1
16	laki-laki	81.5	0	11.3
16	laki-laki	71	1	10.3
16	laki-laki	74.5	1	10.8
16	laki-laki	86.4	0	11.4
16	laki-laki	67.5	1	10.3
16	laki-laki	71.6	1	10.5
16	perempuan	69.9	1	10.1
16	perempuan	89.4	0	12
16	perempuan	72.9	1	10.7
16	perempuan	65.9	1	10.4
16	perempuan	70.4	1	10.8
16	perempuan	68.2	1	10.4
16	perempuan	82.4	0	11.4
17	laki-laki	87.5	0	11.7
17	laki-laki	87.4	0	11.8
17	laki-laki	80.9	0	12
17	laki-laki	93.1	0	12.1
17	laki-laki	92.9	0	12.3
17	laki-laki	77	0	11.6
17	laki-laki	88.8	0	11.7
17	laki-laki	89.8	0	12.5
17	laki-laki	85.7	0	11.7
17	perempuan	68	1	10.7
17	perempuan	81.5	0	11.8
17	perempuan	76	0	11.9
17	perempuan	78.5	0	12
17	perempuan	91.7	0	12.5
17	perempuan	66	1	10.8
17	perempuan	79.6	0	11.8

17	perempuan	78.7	0	11.9
18	laki-laki	71.6	1	11.5
18	laki-laki	85.2	0	12.1
18	laki-laki	73.7	1	11.3
18	laki-laki	91.4	0	12.6
18	laki-laki	72.7	1	11.4
18	laki-laki	82.8	0	12.3
18	laki-laki	80.1	0	12.1
18	perempuan	76.1	0	12.3
18	perempuan	75.6	0	12.5
18	perempuan	68.1	1	11.3
18	perempuan	68.2	1	11.2
18	perempuan	84.8	0	12.1
18	perempuan	88.1	0	12.3
18	perempuan	83.2	0	12.2
18	perempuan	78.3	0	12.4
19	laki-laki	87.6	0	12.1
19	laki-laki	95	0	13.1
19	laki-laki	95.5	0	13.3
19	laki-laki	73.9	1	11.8
19	laki-laki	91.8	0	13.1
19	laki-laki	81.9	0	12.8
19	laki-laki	93.3	0	13.5
19	laki-laki	77.3	1	12.1
19	laki-laki	92.6	0	13.1
19	laki-laki	79.3	0	13
19	perempuan	80.6	0	12.6
19	perempuan	73.7	1	12.1
19	perempuan	70.1	1	12
19	perempuan	83.5	0	12.8
19	perempuan	67.9	1	11.8
19	perempuan	93.4	0	13.5
19	perempuan	87.2	0	13
19	perempuan	75.3	1	12.1
19	perempuan	70	1	12
20	laki-laki	95.6	0	13.6
20	laki-laki	79.7	0	13.1
20	laki-laki	79.5	0	13.3
20	laki-laki	77.6	1	12.7
20	laki-laki	85	0	13.1
20	laki-laki	87.6	0	13.4
20	laki-laki	75.9	1	12.3

20	laki-laki	83.6	0	13.3
20	laki-laki	92	0	13.5
20	perempuan	84.1	0	13.2
20	perempuan	73.3	1	12.3
20	perempuan	73	1	12.3
20	perempuan	71.8	1	12.1
20	perempuan	73.4	1	12.3
20	perempuan	93.7	0	14
20	perempuan	94.6	0	13.7
20	perempuan	78.2	0	13.1
20	perempuan	82.8	0	13.4
21	laki-laki	79.9	0	13.6
21	laki-laki	77	1	13.1
21	laki-laki	74.6	1	12.6
21	laki-laki	93.8	0	13.8
21	laki-laki	86.8	0	14
21	laki-laki	87.6	0	13.8
21	laki-laki	74.7	1	12.8
21	laki-laki	80	0	13.8
21	laki-laki	73.3	1	12.7
21	perempuan	84.1	0	13.9
21	perempuan	83.6	0	14
21	perempuan	83	0	13.8
21	perempuan	91.1	0	13.7
21	perempuan	95.9	0	14.3
21	perempuan	73.1	1	13
21	perempuan	89.1	0	13.8
22	laki-laki	97.4	0	14.6
22	laki-laki	80.6	0	14.2
22	laki-laki	89	0	14.1
22	laki-laki	93.6	0	14.5
22	laki-laki	83.1	0	14.3
22	laki-laki	75.3	1	13.1
22	laki-laki	87.6	0	14.3
22	perempuan	96.2	0	14.8
22	perempuan	83.6	0	14.4
22	perempuan	97.4	0	14.8
22	perempuan	96.5	0	14.9
22	perempuan	71.8	1	13.4
22	perempuan	76.9	1	13.9
22	perempuan	74.3	1	13.3
22	perempuan	88.4	0	14.5

22	perempuan	84.2	0	14.3
23	laki-laki	81.3	0	14.1
23	laki-laki	97.5	0	15.5
23	laki-laki	94.8	0	14.6
23	laki-laki	83.4	0	14.9
23	laki-laki	87	0	14.7
23	laki-laki	86.2	0	14.8
23	laki-laki	89.8	0	15
23	laki-laki	96.2	0	15.4
23	laki-laki	76.7	1	13.8
23	laki-laki	83.3	0	14.7
23	laki-laki	75.7	1	13.8
23	laki-laki	77.9	1	14
23	laki-laki	97.4	0	15.2
23	perempuan	75.2	1	13.7
23	perempuan	92.9	0	14.7
23	perempuan	97.4	0	15.1
23	perempuan	80.9	0	14.8
23	perempuan	76.5	1	14.5
23	perempuan	85	0	14.7
23	perempuan	93.6	0	14.9
23	perempuan	76.2	1	14.3
23	perempuan	85.1	0	14.8
23	perempuan	72.2	1	13.9
23	perempuan	93.9	0	14.8
23	perempuan	90.4	0	14.7
24	laki-laki	97.7	0	16
24	laki-laki	82.8	0	15.3
24	laki-laki	75.8	1	14.3
24	laki-laki	89.4	0	15.3
24	laki-laki	78.6	1	14.8
24	laki-laki	83.9	0	15.4
24	laki-laki	77.9	1	14.3
24	laki-laki	99.3	0	15.9
24	laki-laki	77.4	1	14.2
24	laki-laki	100.8	0	15.8
24	laki-laki	78.5	1	14.8
24	laki-laki	96.7	0	15.9
24	perempuan	99.5	0	15.6
24	perempuan	73.7	1	14.1
24	perempuan	88.9	0	15.1
24	perempuan	99.9	0	15.6

24	perempuan	77.7	1	14.7
24	perempuan	99.4	0	15.8
24	perempuan	75.9	1	14.1
24	perempuan	77.8	1	14.8
24	perempuan	92.1	0	15.4
24	perempuan	92.1	0	15.2
24	perempuan	96.3	0	16
24	perempuan	87.1	0	15.3
24	perempuan	78.6	1	14.9