



**APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI (SISTEM
ISYARAT BAHASA INDONESIA) MENGGUNAKAN YOLOV8 BERBASIS
WEBSITE**

LAPORAN TUGAS AKHIR

HERU SETIAWAN SAPUTRA

20670087

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNIK DAN INFORMATIKA

UNIVERSITAS PGRI SEMARANG

2024



**APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI (SISTEM
ISYARAT BAHASA INDONESIA) MENGGUNAKAN YOLOV8 BERBASIS
WEBSITE**

LAPORAN TUGAS AKHIR

**Diajukan kepada Fakultas Teknik dan Informatika Universitas PGRI
Semarang untuk Penyusunan Skripsi**

HERU SETIAWAN SAPUTRA

20670087

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

2024

HALAMAN PERSETUJUAN

**APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI (SISTEM
ISYARAT BAHASA INDONESIA) MENGGUNAKAN YOLOV8
BERBASIS WEBSITE**

Disusun dan diajukan oleh

HERU SETIAWAN SAPUTRA

NPM 20670087

**Telah disetujui oleh pembimbing untuk dilanjutkan untuk menempuh sidang
laporan tugas akhir pada tanggal 3 Juli 2024**

Pembimbing Utama,



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

NIDN. 0626028201

Pembimbing Pendamping,



Febrian Murti Dewanto, SE., M.Kom.

NIDN. 0606027801

TUGAS AKHIR
APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI (SISTEM
ISYARAT BAHASA INDONESIA) MENGGUNAKAN YOLOV8
BERBASIS WEBSITE

Disusun dan diajukan oleh
HERU SETIAWAN SAPUTRA

20670087

Telah dipertahankan di depan Dewan Penguji pada tanggal 15 Juli 2024 dan
dinyatakan telah memenuhi syarat Dewan Penguji

Ketua,

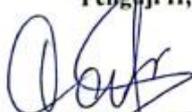
Ibnu Fata Husodo, S.T., M.T
NIDN. 0602126902

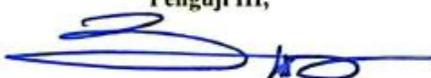
Sekretaris,

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

Penguji I,

Noora Qotrun Nada, S.T., M.Eng
NIDN. 0626028201

Penguji II,

Febrian Murti D., SE., M. Kom
NIDN. 0619048202

Penguji III,

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

MOTTO DAN PERSEMBAHAN

MOTTO

"Bukan jumlah yang menentukan, melainkan semangat dan keyakinan dalam memperjuangkan tujuan."

Jendral Soedirman

“Terimalah tantangan sebagai kesempatan untuk berkembang.”

Eren Yeager (Attack on Titan)

Persembahan:

Saya persembahkan tugas akhir ini untuk :

1. Kedua orang tua saya.
2. Kakak – kakak saya.
3. Almamaterku Universitas PGRI Semarang

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Heru Setiawan Saputra

NPM : 20670087

Program Studi : Informatika

Fakultas : Teknik dan Informatika

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

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

Semarang,

Yang membuat pernyataan

Heru Setiawan Saputra

20670087

ABSTRAK

Bahasa isyarat adalah sistem komunikasi yang menggunakan gerakan tubuh, termasuk gerakan tangan dan wajah, untuk menyampaikan makna kata-kata atau konsep. Setiap negara memiliki bahasa isyarat yang unik, yang berbeda dari satu negara ke negara lainnya. Sistem Isyarat Bahasa Indonesia (SIBI) mengambil konsep bahasa isyarat negara Amerika yakni American Sign Language (ASL). Bahasa isyarat SIBI ini memiliki ciri khas yakni menggunakan satu tangan untuk menyampaikan pesan yang membentuk pola atau simbol tertentu yang bermakna. Kemunculan teknologi Artificial Intelligence (AI) yang baru-baru ini mencuat telah menawarkan potensi untuk membantu masyarakat dalam memahami gerakan Bahasa Isyarat SIBI. Dengan menerapkan pengolahan citra menggunakan teknologi Machine Learning, yang merupakan salah satu cabang utama dari AI, dapat diproyeksikan untuk memprediksi jenis dan kategori Bahasa Isyarat. Dalam rangka mengatasi masalah ini, sebuah aplikasi deteksi kata dasar bahasa isyarat SIBI dikembangkan menggunakan YOLOv8 berbasis *website*. Metode pengembangan yang digunakan adalah model *Waterfall* dengan tingkat akurasi mencapai 98%. Aplikasi ini telah melalui tiga jenis pengujian, yaitu *black box*, UAT (*User Acceptance Testing*), dan pengujian lapangan. Hasil pengujian *black box* menunjukkan tingkat keberhasilan 100%, UAT mencapai 92%, sementara hasil pengujian lapangan menunjukkan bahwa aplikasi berhasil mendeteksi kata dasar bahasa isyarat dengan baik dan akurat secara *real-time*, memberikan manfaat bagi penyandang tunarungu dan tunawicara.

Kata Kunci : YOLOv8, Sistem Isyarat Bahasa Indonesia, Aplikasi Deteksi, Website, AI.

PRAKATA

Dengan rasa syukur yang tak terhingga kepada Allah SWT atas berkah dan rahmat-Nya, penulis berhasil menyelesaikan tugas akhir ini dengan lancar. Tugas akhir berjudul " APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI (SISTEM ISYARAT BAHASA INDONESIA) MENGGUNAKAN YOLOV8 BERBASIS WEBSITE” disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer.

Proses penyusunan tugas akhir ini tidak terlepas dari berbagai rintangan dan hambatan yang dihadapi. Namun, dengan bantuan, arahan, dan dukungan yang diberikan oleh berbagai pihak, terutama pembimbing, penulis berhasil mengatasi setiap kendala dengan baik. Oleh karena itu, penulis ingin mengucapkan rasa terima kasih yang tulus kepada:

1. Dr. Sri Suciati, M. Hum. selaku Rektor Universitas PGRI Semarang.
2. Bapak Ibnu Toto Husodo, ST., MT 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 Universitas PGRI Semarang.
4. Ibu Noora Qotrun Nada, S.T., M.Eng selaku Dosen Pembimbing I Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang yang telah membimbing penulis saat menyusun tugas akhir dengan penuh dedikasi yang tinggi.
5. Bapak Febrian Murti Dewanto, S.E., M.Kom. selaku Dosen Pembimbing II Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang yang telah membimbing penulis saat menyusun tugas akhir dengan penuh dedikasi yang tinggi.
6. Seluruh Dosen beserta Staf di Program Studi Informatika Fakultas Teknik dan Informatika Universitas PGRI Semarang atas segala ilmu dan bimbingan selama masa studi.

7. Kedua orang tua tercinta Bapak Sunarto dan Ibu Dariyah, serta kakak-kakak penulis yang senantiasa mendoakan penulis demi terselesaikannya tugas akhir ini.
8. Untuk NPM 20670133 yang selalu memberikan semangat dan dukungan.
9. Serta untuk semua teman-teman dekat penulis yang tidak bisa disebutkan satu persatu yang telah memberikan semangat dalam penyusunan tugas akhir ini.

Akhir kata penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dan memberikan dukungan. Penulis berharap tugas akhir ini dapat memberikan manfaat bagi banyak orang khususnya di bidang Informatika.

Semarang, 15 Juli 2024

Heru Setiawan Saputra

DAFTAR ISI

SAMPUL LUAR.....	i
SAMPUL DALAM.....	ii
HALAMAN PERSETUJUAN.....	Error! Bookmark not defined.
HALAMAN PENGESAHAN.....	Error! Bookmark not defined.
MOTTO DAN PERSEMBAHAN	v
PERNYATAAN KEASLIAN TULISAN.....	vi
ABSTRAK	vii
PRAKATA.....	viii
DAFTAR ISI	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	3
C. Rumusan Masalah	3
D. Batasan Masalah.....	3
E. Tujuan Penelitian	4
F. Manfaat Penelitian	4
BAB II KAJIAN PUSTAKA DAN TEORI	6
A. Tinjauan Pustaka	6
B. Landasan Teori.....	10
C. Kerangka Berpikir.....	23
BAB III METODOLOGI PENELITIAN.....	25

A. Pendekatan Penelitian	25
B. Fokus Penelitian.....	25
C. Desain Penelitian.....	25
D. Teknik Pengumpulan Data.....	26
E. Teknik Analisis Data.....	27
BAB IV HASIL DAN PEMBAHASAN.....	29
A. Hasil	29
1. Analisis Kebutuhan	29
2. Desain.....	32
3. Implementasi Model.....	45
4. Implementasi Sistem	63
5. Pengujian	66
B. Pembahasan.....	81
1. Pemodelan Data.....	81
2. Pengembangan Aplikasi	81
BAB V KESIMPULAN DAN SARAN.....	86
A. Kesimpulan	86
B. Saran.....	87
DAFTAR PUSTAKA	88
LAMPIRAN.....	92

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu.....	6
Tabel 2. 2 Simbol Class Diagram.....	17
Tabel 2. 3 Use Case Diagram	19
Tabel 2. 4 Squence Diagram	20
Tabel 2. 5 Activity Diagram	21
Tabel 4. 1 Jumlah Image Dataset	29
Tabel 4. 2 Kebutuhan Perangkat Keras Computer	31
Tabel 4. 3 Kebutuhan Perangkat Lunak Computer	31
Tabel 4. 4 Data Pelatihan atau Training	48
Tabel 4. 5 Data Validasi.....	49
Tabel 4. 6 Data Test.....	50
Tabel 4. 7 Pengujian Black Box	66
Tabel 4. 8 User Acceptence Testing (UAT).....	70
Tabel 4. 9 White Box Testing.....	72

DAFTAR GAMBAR

Gambar 2. 1 Model Waterfall.....	16
Gambar 2. 2 Kerangka Berpikir.....	23
Gambar 3. 1 Skema Desain Penelitian.....	26
Gambar 4. 1 Use Case Diagram.....	33
Gambar 4. 2 Activity Diagram Halaman.....	34
Gambar 4. 3 Activity Diagram Deteksi.....	35
Gambar 4. 4 Activity Diagram Kamus.....	36
Gambar 4. 5 Activity Diagram Tentang.....	36
Gambar 4. 6 Sequence Diagram Beranda.....	37
Gambar 4. 7 Sequence Diagram Deteksi.....	38
Gambar 4. 8 Sequence Diagram Kamus.....	39
Gambar 4. 9 Sequence Diagram Tentang.....	40
Gambar 4. 10 Class Diagram.....	41
Gambar 4. 11 Desain User Interface Halaman Beranda.....	42
Gambar 4. 12 Desain User Interface Halaman Deteksi.....	43
Gambar 4. 13 Desain User Interface Halaman Kamus.....	44
Gambar 4. 14 Desain User Interface Halaman Tentang.....	45
Gambar 4. 15 Dataset Class.....	46
Gambar 4. 16 Labelling Data.....	47
Gambar 4. 17 Tahap preprocessing data.....	51
Gambar 4. 18 Training Model.....	52
Gambar 4. 19 Confusion Matrix Normalized.....	53
Gambar 4. 20 Perhitungan Confusion Matrix.....	54
Gambar 4. 21 F1-Confidence Curve.....	55
Gambar 4. 22 Precision-Confidence Curve.....	56
Gambar 4. 23 Recall-Confidence Curve.....	58
Gambar 4. 24 Precision-Recall Curve.....	59
Gambar 4. 25 Bahasa isyarat “Adik”.....	60
Gambar 4. 26 Bahasa isyarat “Ayah”.....	61

Gambar 4. 27 Bahasa isyarat “Baik”	61
Gambar 4. 28 Bahasa isyarat “Berhenti”	62
Gambar 4. 29 Bahasa isyarat “Buruk”	62
Gambar 4. 30 Halaman Beranda	63
Gambar 4. 31 Halaman Deteksi	64
Gambar 4. 32 Halaman kamus	65
Gambar 4. 33 Halaman Tentang.....	65
Gambar 4. 34 Flowgraph Basis Path.....	79
Gambar 4. 35 Pengujian Aplikasi dengan Penyandang tunawicara.....	80
Gambar 4. 36 Pengujian Aplikasi dengan Penyandang tunawicara.....	80

BAB I

PENDAHULUAN

A. Latar Belakang

Komunikasi adalah sebuah kegiatan yang esensial bagi setiap individu dalam kehidupan manusia. Fungsinya adalah untuk memperoleh pemahaman terhadap maksud dan tujuan yang ingin disampaikan oleh seseorang terhadap diri kita atau individu lainnya. Hasil penelitian menunjukkan bahwa sekitar 75% dari waktu yang dimiliki manusia digunakan untuk melakukan kegiatan komunikasi. Hal ini meliputi berbagai bentuk komunikasi, baik itu dengan orang lain maupun dengan diri sendiri melalui proses *self-talk*. Studi lain mengungkapkan bahwa manusia, terutama perempuan, memiliki kecenderungan untuk menggunakan sekitar 20 ribu kata setiap harinya dalam proses komunikasi, yang dapat mencakup ekspresi pendapat, ungkapan perasaan, atau bahkan hanya sekadar mengekspresikan pemikiran dan permasalahan yang sedang dipertimbangkan[1].

Berdasarkan informasi yang diperoleh dari Sistem Informasi Penyandang Disabilitas Kementerian Kesehatan Republik Indonesia pada bulan Maret 2022, jumlah individu yang mengalami disabilitas di Indonesia mencapai 212.240. Terjadi peningkatan jumlah ini dalam periode dua tahun terakhir. Pada bulan Maret 2020, jumlah individu dengan disabilitas tercatat sebesar 197.582, sementara pada bulan yang sama tahun 2021, jumlahnya meningkat menjadi 207.604. Jumlah individu yang mengalami disabilitas tunawicara pada bulan Maret 2022 adalah sebanyak 19.392, yang setara dengan sekitar 9,14% dari total jumlah individu yang mengalami disabilitas di Indonesia[2].

Bahasa isyarat adalah sistem komunikasi yang menggunakan gerakan tubuh, termasuk gerakan tangan dan wajah, untuk menyampaikan makna kata-kata atau konsep. Setiap negara memiliki bahasa isyarat yang unik, yang

berbeda dari satu negara ke negara lainnya. Di Indonesia, terdapat dua jenis bahasa isyarat, yaitu Bahasa Isyarat Indonesia (BISINDO) dan Sistem Isyarat Bahasa Indonesia (SIBI). BISINDO berkembang secara alami di kalangan komunitas penyandang disabilitas, sementara SIBI merupakan bahasa isyarat Indonesia yang telah disepakati secara nasional. SIBI digunakan secara khusus dalam konteks formal, seperti di lingkungan pendidikan dan acara resmi[3], [4]. SIBI mengambil konsep bahasa isyarat negara Amerika yakni American Sign Language (ASL). Bahasa isyarat SIBI ini memiliki ciri khas yakni menggunakan satu tangan untuk menyampaikan pesan yang membentuk pola atau simbol tertentu yang bermakna[5]. Namun, kurangnya penerapan atau penyuluhan sejak usia dini mengenai metode komunikasi menggunakan Sistem Isyarat Bahasa Indonesia (SIBI) berdampak pada rendahnya pemahaman tentang topik tersebut[6].

Kemunculan teknologi Artificial Intelligence (AI) yang baru-baru ini mencuat telah menawarkan potensi untuk membantu masyarakat dalam memahami gerakan Bahasa Isyarat SIBI. Dengan menerapkan pengolahan citra menggunakan teknologi Machine Learning, yang merupakan salah satu cabang utama dari AI, dapat diproyeksikan untuk memprediksi jenis dan kategori Bahasa Isyarat. Hal ini dapat dilakukan melalui platform internet dan website. YOLOv8 dibangun berdasarkan kemajuan mutakhir dalam *deep learning* dan *computer vision*, menawarkan kinerja yang tak tertandingi dalam hal kecepatan dan ketepatan. Desainnya yang disederhanakan membuatnya cocok untuk berbagai aplikasi dan mudah disesuaikan dengan platform perangkat keras yang berbeda, mulai dari perangkat edge hingga cloud API[7]. Penerapan teknologi ini diharapkan dapat memberikan dampak positif terhadap lingkungan dan kepedulian masyarakat terhadap orang tunawicara.

Oleh sebab itu, peneliti saat ini merasa tertarik untuk mengeksplorasi efektivitas penggunaan metode terbaru, khususnya YOLOv8, dalam konteks penerapan Aplikasi Deteksi Kata Dasar Bahasa Isyarat. Dengan hadirnya Aplikasi Deteksi Kata Dasar Bahasa Isyarat dengan menggunakan metode

YOLOv8, akan memberikan kemudahan bagi siapa pun yang berkeinginan untuk berkomunikasi melalui bahasa isyarat secara efisien. Penulis menggunakan metode pengembangan Waterfall dengan diimplementasikan dengan Aplikasi Website. Dengan demikian, penulis mengajukan judul penelitian “APLIKASI DETEKSI KATA DASAR BAHASA ISYARAT SIBI MENGGUNAKAN METODE YOLOV8 BERBASIS WEBSITE”.

B. Identifikasi Masalah

Permasalahan penelitian dapat diidentifikasi sebagai berikut :

Terjadi hambatan dalam interaksi komunikatif antara individu dengan disabilitas dan masyarakat umum karena minimnya pemahaman mengenai makna dari gestur bahasa isyarat yang dipergunakan.

C. Rumusan Masalah

Rumusan masalah pada penelitian ini adalah :

Bagaimana merancang dan membangun Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI menggunakan YOLOv8 Berbasis Website?

D. Batasan Masalah

Mengingat terbatasnya tenaga dan waktu yang ada, serta menjaga dan menghindari pembahasan masalah yang terlalu luas, maka penulis membatasi permasalahan pada Aplikasi Deteksi Bahasa Isyarat yang meliputi :

1. Sistem Deteksi Bahasa Isyarat yang dibuat hanya mencakup sejumlah terbatas kata dalam kamus bahasa isyarat, yang meliputi 20 gerakan kata bahasa isyarat yang sering dipakai dalam interaksi sehari-hari di masyarakat.
2. Aplikasi Deteksi Bahasa Isyarat hanya bisa diakses melalui website.
3. Aplikasi dibangun menggunakan dataset berbentuk image dan digunakan secara realtime yang bersifat publik.

4. Aplikasi telah dirancang dengan menggunakan Platform Roboflow dan Google Colaboratory untuk keperluan pemodelan, sedangkan Flask digunakan untuk pembuatan situs webnya, yang didasarkan pada bahasa pemrograman Python dan PHP.
5. Aplikasi dirancang menggunakan model Waterfall dan pengujian sistem dilakukan menggunakan metode Blackbox dan UAT. Dan penulis menggunakan model Waterfall hanya sampai dalam tahap pengujian.

E. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dijabarkan sebelumnya, maka tujuan dari penelitian ini yaitu sebagai berikut :

1. Merancang dan membangun Aplikasi Deteksi Bahasa Isyarat SIBI menggunakan YOLOv8 Berbasis Website.
2. Memberikan solusi yang inovatif dan berguna bagi komunitas yang menggunakan bahasa isyarat sebagai sarana komunikasi.

F. Manfaat Penelitian

1. Secara teoritis, penelitian ini diharapkan bermanfaat sebagai :
 - a. Penelitian ini mendorong pengembangan teknologi deteksi *real-time* dan pemrosesan bahasa isyarat, yang dapat membuka pintu bagi inovasi baru dalam bidang kecerdasan buatan dan visi komputer.
 - b. Penelitian ini dapat memberikan wawasan baru tentang aspek-aspek komunikasi non-verbal, khususnya dalam konteks bahasa isyarat. Hal ini dapat membantu memperkaya teori-teori komunikasi yang ada.
 - c. Studi tentang bagaimana individu dengan disabilitas menggunakan aplikasi deteksi bahasa isyarat juga dapat memberikan wawasan tentang interaksi manusia dan komputer dalam konteks penggunaan teknologi oleh pengguna dengan kebutuhan aksesibilitas khusus.

2. Secara praktis kegunaan hasil penelitian ini adalah :

a. Bagi Pribadi

Manfaat penelitian ini bagi peneliti adalah memberikan kesempatan untuk terus belajar serta meningkatkan pemahaman tentang bahasa isyarat dan interaksi komunikatif dengan individu yang menggunakan bahasa isyarat. Hal ini menciptakan peluang bagi peneliti untuk memperluas pengetahuan dan wawasan mereka dalam domain bahasa isyarat serta memperdalam pemahaman tentang dinamika komunikasi dengan menggunakan bahasa tersebut bersama komunitas penggunanya.

b. Bagi Masyarakat Umum

Aplikasi ini membantu mengurangi hambatan komunikasi antara individu dengan gangguan pendengaran dan masyarakat umum. Hal ini memungkinkan interaksi sosial yang lebih lancar dan efisien tanpa kebutuhan akan penerjemah atau pengetahuan tentang bahasa isyarat.

BAB II
KAJIAN PUSTAKA / TEORI

A. Tinjauan Pustaka

Sebelumnya terdapat penelitian yang menggunakan metode YOLO versi yang terbaru maupun yang terlama untuk mendeteksi bahasa isyarat. Beberapa penelitian memiliki kriteria dan pola yang berbeda satu sama lain. Berikut merupakan perbandingan penelitian sebelumnya menggunakan metode YOLO.

Tabel 2. 1 Penelitian Terdahulu

No.	Nama dan Tahun Penelitian	Judul	Metode	Hasil
1.	Agung Ma'ruf, Mardi Hardjianto (2023)	Penerapan Algoritma You Only Look Once Version 8 untuk Identifikasi Abjad Bahasa Isyarat Indonesia	YOLOv8	Dari kumpulan 26 huruf abjad, 25 huruf berhasil terdeteksi secara sempurna, sedangkan satu huruf mencapai tingkat keberhasilan sebesar 96,5%. Evaluasi proses ini mengungkapkan variasi nilai, termasuk akurasi mencapai 99,8%, presisi 99,4%, dan recall 99,8%.
2.	Dimas Permana, Joko Sutopo (2023)	Aplikasi Pengenalan Abjad Sistem Isyarat	YOLOv5	Beberapa kelas, seperti 'A' dan 'E', menunjukkan akurasi deteksi yang rendah

		Bahasa Indonesia (SIBI) dengan Algoritma YOLOv5		karena gerakan yang hampir serupa dengan kelas 'S' atau 'A'. Kelas 'D' juga menunjukkan akurasi yang rendah karena memerlukan variasi posisi yang lebih kompleks. Kelas 'J' memiliki gestur yang cukup rumit karena adanya gerakan lanjutan, sehingga akurasinya juga rendah. Pengujian deteksi secara real-time menggunakan kamera ponsel menunjukkan akurasi sebesar 77%, yang secara keseluruhan menunjukkan hasil pengujian yang memuaskan.
3.	Dicky Luthfy, Casi Setianingsih, Marisa W Paryasto (2023)	<i>Indonesian Sign Language Classification Using You Only Look Once</i>	YOLOv5	Mampu melakukan deteksi secara real-time terhadap 26 alfabet dalam Bahasa Isyarat BISINDO tanpa terpengaruh oleh latar belakang atau tingkat pencahayaan, namun

				<p>dipengaruhi oleh jarak kamera dan objek.</p> <p>Konfigurasi optimal pada penelitian ini meliputi penggunaan dataset dengan pembagian 70% data untuk pelatihan, 20% untuk validasi, dan 10% untuk pengujian, dengan 300 epochs, ukuran batch 16, dan tingkat pembelajaran 0,01, menghasilkan nilai mAP@0.5IoU sebesar 99,27%.</p>
4.	<p>Muhammad Rifki Pratama, Heri Pratikno, Yosefine Triwidyastuti, dan Musayyanah (2023)</p>	<p>Pengenalan Gestur Jari Tangan Sebagai Media Pembelajaran Berhitung Bagi PAUD Berbasis Visi Komputer dan Deep Learning</p>	YOLO	<p>Hasil pengujian dalam penelitian ini menunjukkan bahwa akurasi deteksi kesepuluh jari tangan untuk proses berhitung menggunakan metode MediaPipe mencapai 89,9%, dengan frame per second (FPS) berkisar antara 20-25. Sementara itu, metode CNN menunjukkan persentase akurasi</p>

				deteksi sebesar 20%, dengan nilai FPS antara 10-12. Implikasi dari temuan ini adalah peningkatan pengalaman belajar yang menyenangkan, merangsang, dan mendukung perkembangan anak secara holistik.
5.	Shobhit Tyagi, Prashant Upadhyay, Hoor Fatima, Sachin Jain, Avinash Kumar Sharma (2023)	<i>American Sign Language Detection using YOLOv5 and YOLOv8</i>	YOLOv5 dan YOLOv8	Eksperimen menunjukkan bahwa YOLOv8 terbaru memberikan hasil yang lebih baik dibandingkan dengan versi YOLO lainnya dalam hal presisi dan mAP, sementara YOLOv7 memiliki nilai recall yang lebih tinggi selama pengujian daripada YOLOv8. Model yang diusulkan adalah ringan, cepat, dan menggunakan dataset huruf Bahasa Isyarat Amerika untuk pelatihan dan pengujian.

				Model kustom ini mencapai presisi 95%, recall 97%, dan mAP @0.5 96%, menunjukkan kemampuan model dalam pengenalan gerakan tangan secara real-time.
--	--	--	--	--

B. Landasan Teori

1. Aplikasi

Aplikasi adalah perangkat lunak yang dirancang untuk memenuhi kebutuhan dan memfasilitasi aktivitas pengguna, seperti tugas-tugas di instansi pemerintah, kegiatan komersial, atau keperluan individu. Istilah "aplikasi" sering kali dikaitkan dengan pemecahan masalah dalam komputasi, dan sering digunakan secara bersamaan dengan perangkat lunak. Asal-usul kata "aplikasi" berasal dari bahasa Inggris "application", yang berarti penerapan atau penggunaan. Dalam konteks teknis, aplikasi merujuk pada program yang telah dirancang untuk melakukan fungsi tertentu bagi pengguna atau kebutuhan aplikasi lainnya[8].

2. Sistem Isyarat Bahasa Indonesia (SIBI)

Bahasa isyarat merupakan sebuah sistem komunikasi yang tidak mengandalkan bunyi ucapan manusia atau tulisan dalam penggunaannya. Secara khusus, bahasa isyarat ditandai oleh keunikan karena variasi yang ada di setiap negara. Di Indonesia, terdapat dua kategori utama dalam perkembangan bahasa isyarat, yaitu Sistem Isyarat Bahasa Indonesia (SIBI) dan Bahasa Isyarat Indonesia (BISINDO)[9].

Sistem Isyarat Bahasa Indonesia (SIBI) adalah bahasa isyarat resmi di Indonesia yang telah disahkan melalui Undang-Undang No.2 Tahun 1989. SIBI didasarkan pada Bahasa Isyarat Amerika (ASL), yang menggunakan gerakan satu tangan, sehingga SIBI memiliki kesamaan dengan ASL. Bahasa Isyarat Bahasa Indonesia (SIBI) lebih umum digunakan dalam kegiatan resmi, seperti acara pemerintah dan kegiatan sekolah. Meskipun demikian, penggunaan SIBI dianggap sulit, sehingga banyak individu tuna rungu lebih memilih menggunakan Bahasa Isyarat Indonesia (BISINDO)[10].

3. *Machine Learning*

Machine learning, atau yang sering disebut sebagai pembelajaran mesin, merupakan sebuah subbidang dalam kecerdasan buatan (AI) yang terfokus pada pengembangan algoritma dan model statistik. Hal ini memungkinkan komputer untuk belajar dan meningkatkan kinerjanya dalam tugas-tugas tertentu tanpa perlu diprogram secara eksplisit. Sejumlah penelitian telah dilakukan untuk mengeksplorasi cara agar *machine learning* dapat belajar secara mandiri tanpa perlu diprogram secara eksplisit[11]. Tujuan utama dari *machine learning* adalah mengembangkan model yang mampu membuat prediksi atau keputusan berdasarkan data yang telah dijadikan bahan latihan. Proses pelatihan model dengan berbagai teknik akan menghasilkan pengalaman yang dapat mengenali pola-pola yang terdapat dalam data tersebut. Dalam *Machine Learning* terbagi menjadi 3 tipe berdasarkan cara pembelajarannya yaitu:

a) *Supervised Learning*

Supervised Learning, atau yang dikenal sebagai pembelajaran diawasi, merujuk pada algoritma pembelajaran di mana komputer dipersiapkan dengan masukan data yang telah diberi label untuk output tertentu. Dalam konteks pembelajaran terawasi, kelas atau kategori dari setiap objek dalam data sudah diketahui. Biasanya, *Supervised Learning* digunakan untuk tugas-tugas seperti klasifikasi

dan regresi. Contohnya adalah pengklasifikasian jenis-jenis bunga iris[12].

b) *Unsupervised Learning*

Unsupervised Learning merupakan salah satu jenis algoritma machine learning yang bertujuan untuk mengekstraksi informasi atau pola dari dataset[13]. Model yang digunakan akan menemukan pola-pola tersembunyi yang ada dalam data yang diberikan. Biasanya, *Unsupervised Learning* digunakan untuk tugas-tugas seperti pengelompokan data (*clustering*), contohnya adalah segmentasi pelanggan dalam konteks bisnis.

c) *Reinforcement Learning*

Reinforcement Learning adalah sebuah algoritma yang bertujuan untuk mempelajari perilaku optimal dalam suatu lingkungan dengan tujuan memperoleh manfaat maksimal. Algoritma *Reinforcement Learning* dirancang agar mampu merespons secara adaptif terhadap perubahan yang terjadi dalam lingkungan tersebut. Fokus utama dari Reinforcement Learning adalah pada pengidentifikasian tindakan-tindakan yang dapat menghasilkan nilai maksimal dari imbalan kumulatif yang diberikan oleh lingkungan[14].

4. *Deep Learning*

Deep learning merupakan salah satu cabang dari kecerdasan buatan dan machine learning yang melibatkan pengembangan *neural network* dengan *multiple layer*. Tujuan utamanya adalah untuk meningkatkan akurasi dalam berbagai tugas seperti deteksi objek, pengenalan suara, terjemahan bahasa, dan sebagainya. Pendekatan deep learning difokuskan pada pembelajaran representasi data yang semakin kompleks dan abstrak seiring dengan peningkatan kedalaman jaringan. Hal ini memungkinkan model untuk mengenali pola-pola yang sangat kompleks dan non-linear dalam data. *Deep learning* lebih sesuai untuk menangani masalah-masalah yang kompleks dan memiliki kemampuan untuk beradaptasi dengan

situasi yang baru. Dengan demikian, deep learning merupakan bentuk yang lebih kompleks dari *machine learning* karena dapat memproses data yang tidak terstruktur[15].

5. YOLO (You Only Look Once)

YOLO merupakan pengembangan dari metode Convolutional Neural Network(CNN). YOLO membagi gambar menjadi wilayah-wilayah, memprediksi kotak pembatas dan probabilitas agar dapat mengklasifikasikan sebagai objek manusia atau bukan manusia baik berada di areadengan suhu rendah maupun suhu tinggi.YOLO mampu mendeteksi secara real-time[16]. Metode You Only Look Once(YOLO) merupakan salah satu metode yang paling cepat dan akurat pada pendeteksian objek bahkan mampu melebihi hingga 2 kali kemampuan algoritma lain. YOLO mempunyai banyak versi yang sering diterapkan yaitu mulai versi YOLO, YOLOv2 hingga yang terbaru saat ini adalah YOLOv8[17].

YOLOv8 merupakan versi terkini dari algoritma YOLO yang diperkenalkan pada tahun 2020. Versi ini menunjukkan beberapa peningkatan pada algoritma ini terus dikembangkan untuk meningkatkan kecepatan dan akurasi deteksi objek pada berbagai aplikasi secara praktis[18].

6. Roboflow

Roboflow merupakan sebuah platform web yang berfungsi sebagai pengelola kumpulan dataset dan juga merupakan sebuah kerangka kerja pengembangan dalam bidang computer vision. Platform ini memungkinkan pengguna untuk melakukan pengumpulan data yang lebih baik, prapemrosesan dataset, serta melatih model. Dengan menggunakan Roboflow, pengguna dapat berbagi dataset dan melakukan proses anotasi atau penandaan objek yang akan dideteksi menggunakan bounding box. Selain itu, platform ini juga menyediakan fitur prapemrosesan data seperti konversi ke grayscale, serta augmentasi data[19].

7. Flask

Flask merupakan suatu kerangka kerja pengembangan aplikasi web yang dibuat menggunakan bahasa pemrograman Python, yang termasuk dalam kategori microframework. Fungsinya adalah untuk menyediakan kerangka kerja dan tampilan bagi aplikasi web. Dengan menggunakan Flask dan bahasa pemrograman Python, pengembang dapat membuat dan mengelola struktur serta perilaku aplikasi web dengan lebih efisien[20].

8. Bahasa Pemrograman Python

Python adalah sebuah bahasa pemrograman tingkat tinggi yang umumnya digunakan untuk berbagai keperluan pengembangan perangkat lunak, pemrograman web, analisis data, kecerdasan buatan, serta aplikasi-aplikasi lainnya. Bahasa ini dikembangkan oleh seorang programmer bernama Guido Van Rossum pada akhir 1980-an di Centrum Wiskunde & Informatica (CWI) di Belanda. Salah satu ciri khas dari Python adalah penggunaan indentasi (jorong) untuk membedakan blok kode, berbeda dengan kebanyakan bahasa pemrograman lain yang menggunakan simbol-simbol tertentu sebagai penanda blok kode[21].

9. Model Waterfall

Model waterfall, juga dikenal sebagai model sekuensial linier atau alur hidup klasik, adalah suatu pendekatan pengembangan perangkat lunak yang bergerak secara terurut dari tahap analisis, desain, pengembangan, pengujian, hingga tahap pemeliharaan[22].

1. Analisis Kebutuhan (*Requirements Analysis*)

Tahap pertama ini melibatkan proses identifikasi dan pemahaman terhadap permintaan kebutuhan pengguna dan pemangku kepentingan untuk sistem atau aplikasi yang hendak dikembangkan. Pengguna akan dihubungi oleh tim pengembangan untuk mengumpulkan dan memeriksa persyaratan fungsional dan non-fungsional. Dokumen persyaratan yang menguraikan fitur,

kemampuan, dan tujuan dari sistem yang akan dikembangkan adalah produk akhir dari langkah ini.

2. Desain (*Design*)

Pada tahapan ini yaitu melakukan perancangan sistem dilakukan berdasarkan dokumen kebutuhan. Perancangan sistem terdiri dari beberapa aspek, termasuk perancangan arsitektur, desain antarmuka pengguna, desain basis data, dan desain modul atau komponen sistem. Tujuan dari tahap ini adalah menghasilkan rancangan yang jelas dan komprehensif untuk sistem yang akan dikembangkan.

3. Pengembangan (*Implementation*)

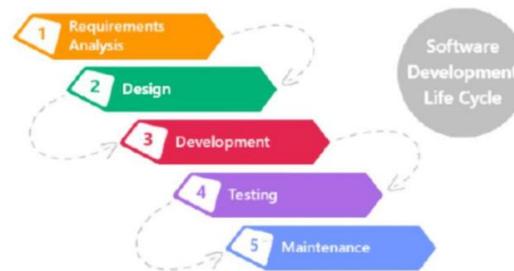
Setelah perancangan selesai, tim pengembang akan mulai mengimplementasikan rancangan yang telah dirancang. Desain yang disetujui berfungsi sebagai dasar untuk menulis kode program. Pengkodean, pengujian unit, dan integrasi komponen pada sistem adalah bagian dari langkah implementasi. Tahap pengembangan ini menghasilkan kemampuan aplikasi atau sistem yang dihasilkan untuk berfungsi sesuai dengan desain yang ditentukan oleh perancang sistem.

4. Pengujian (*Testing*)

Pada tahapan ini, sistem yang telah diimplementasikan akan dilakukan pengujian untuk memastikan bahwa sistem tersebut dapat berfungsi sesuai dengan persyaratan yang telah ditetapkan. Pengujian dilakukan untuk mengidentifikasi dan memperbaiki cacat atau kesalahan dalam sistem. Tes yang dapat dilakukan meliputi tes unit, tes integrasi, tes fungsional, dan tes kinerja. Tujuan dari tahap ini adalah memastikan kualitas dan kehandalan sistem sebelum diluncurkan.

5. Pemeliharaan (*Maintenance*)

Setelah sistem diuji dan diluncurkan, tahap pemeliharaan dimulai. Pemeliharaan melibatkan pemantauan kinerja sistem, pemecahan masalah, dan peningkatan sistem jika diperlukan. Pemeliharaan dapat mencakup perbaikan bug, peningkatan fungsionalitas, dan penyesuaian dengan perubahan kebutuhan pengguna. Tujuan dari tahap ini adalah memastikan sistem tetap berjalan dengan baik dan dapat diandalkan dalam jangka panjang. parafrasekan dalam bahasa ilmiah



Gambar 2. 1 Model Waterfall

10. Definisi Website

Website merupakan suatu fasilitas dalam jaringan internet yang menghubungkan dokumen secara lokal maupun dari jarak jauh. Dokumen yang terdapat dalam website disebut halaman web, dan pengguna dapat berpindah dari satu halaman ke halaman lainnya melalui tautan yang terdapat di web, yang memungkinkan pengguna untuk berinteraksi dengan konten yang tersedia. Halaman web ini dapat diakses dan ditampilkan menggunakan perangkat lunak peramban (browser) seperti Google, Mozilla Firefox, Safari, dan peramban web lainnya[23].

11. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah suatu bahasa visual yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan

sistem pengembangan perangkat lunak berbasis objek. UML tidak bersifat sebagai bahasa pemrograman, melainkan sebagai model yang dapat diimplementasikan dalam berbagai bahasa pemrograman. Model-model yang dibuat dalam UML dapat dihubungkan dengan berbagai bahasa pemrograman seperti Python dan PHP memungkinkan pemetaan yang konsisten antara model dan implementasi dalam kode program. Struktur diagram UML terdiri dari Use Case diagram, Activity diagram, Class diagram, dan Sequence diagram[24]. Berikut adalah diagram dalam UML :

a. *Class Diagram*

Diagram kelas bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi kolaborasi serta relasi. Pada Tabel 2.2. adalah simbol-simbol yang ada pada diagram kelas :

Tabel 2. 2 Simbol Class Diagram

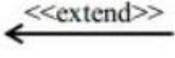
Simbol	Keterangan
	<i>Generalization</i> : Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i> : Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i> : Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

	<i>Collaboration</i> : Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i> : Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i> : Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Association</i> : Apa yang menghubungkan antara objek satu dengan objek lainnya.

b. *Use Case Diagram*

Use Case Diagram menggambarkan sejumlah external actors dan hubungannya ke use case yang diberikan oleh sistem. Use case adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari use case symbol namun dapat juga dilakukan dalam Activity diagrams. Use case digambarkan hanya yang dilihat dari luar oleh actor (keadaan lingkungan sistem yang dilihat user) dan bukan bagaimana fungsi yang ada di dalam sistem. Tabel 2.3 adalah symbol-simbol dalam Use Case Diagram:

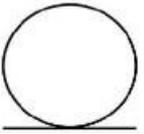
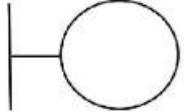
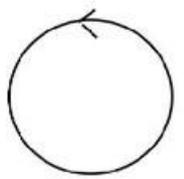
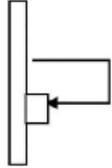
Tabel 2. 3 Use Case Diagram

Simbol	Keterangan
	Aktor: Mewakil peran orang, <i>system</i> yang lain, atau alat ketika berkomunikasi dengan <i>use case</i> .
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan actor.
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i> .
	Generalisasi: Menunjukkan spesialisasi <i>actor</i> untuk dapat berpartisipasi dengan <i>use case</i> .
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi.

c. *Sequence Diagram*

Diagram ini bersifat dinamis. Diagram sequence merupakan diagram interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu. Pada Tabel 2.4 adalah simbol-simbol sequence diagram :

Tabel 2. 4 Squence Diagram

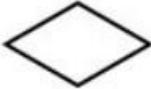
Simbol	Keterangan
	<i>Entity Class</i> : Gambaran sistem sebagai landasan dalam menyusun basis data.
	<i>Boundary Class</i> : Menangani komunikasi antar lingkungan sistem .
	<i>Control Class</i> : Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika.
	<i>Recursive</i> : Pesan untuk dirinya.
	<i>Activation</i> : Mewakili proses durasi aktivasi sebuah operasi.
	<i>Life Line</i> : Komponen yang digambarkan garis putus terhubung dengan objek.

d. *Activity Diagram*

Diagram ini bersifat dinamis. Diagram ini adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktivitas ke

aktivitas lainnya dari suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi – fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek. Tabel 2.5 adalah simbol-simbol pada activity diagram :

Tabel 2. 5 Activity Diagram

Simbol	Keterangan
	Status awal: Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas: Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / <i>Decision</i> : Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / <i>Join</i> ; Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir: Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	<i>Swimlane</i> : <i>Swimlane</i> Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

12. Blackbox Testing

Black Box Testing adalah sebuah teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari sistem, di mana pengujian dilakukan dengan memeriksa nilai input dan output tanpa memerlukan pengetahuan tentang struktur kode dari perangkat lunak tersebut. Kelebihan utama dari metode Black Box Testing adalah kemampuannya untuk melakukan pengujian tanpa memerlukan pengetahuan tentang bahasa pemrograman yang digunakan serta sudut pandang pengguna, sehingga memungkinkan untuk menemukan inkonsistensi dalam perangkat lunak. Tujuan dari pengujian Black Box adalah untuk memastikan bahwa proses aplikasi berfungsi sesuai dengan kebutuhan yang diharapkan[25].

13. User Acceptance Testing (UAT)

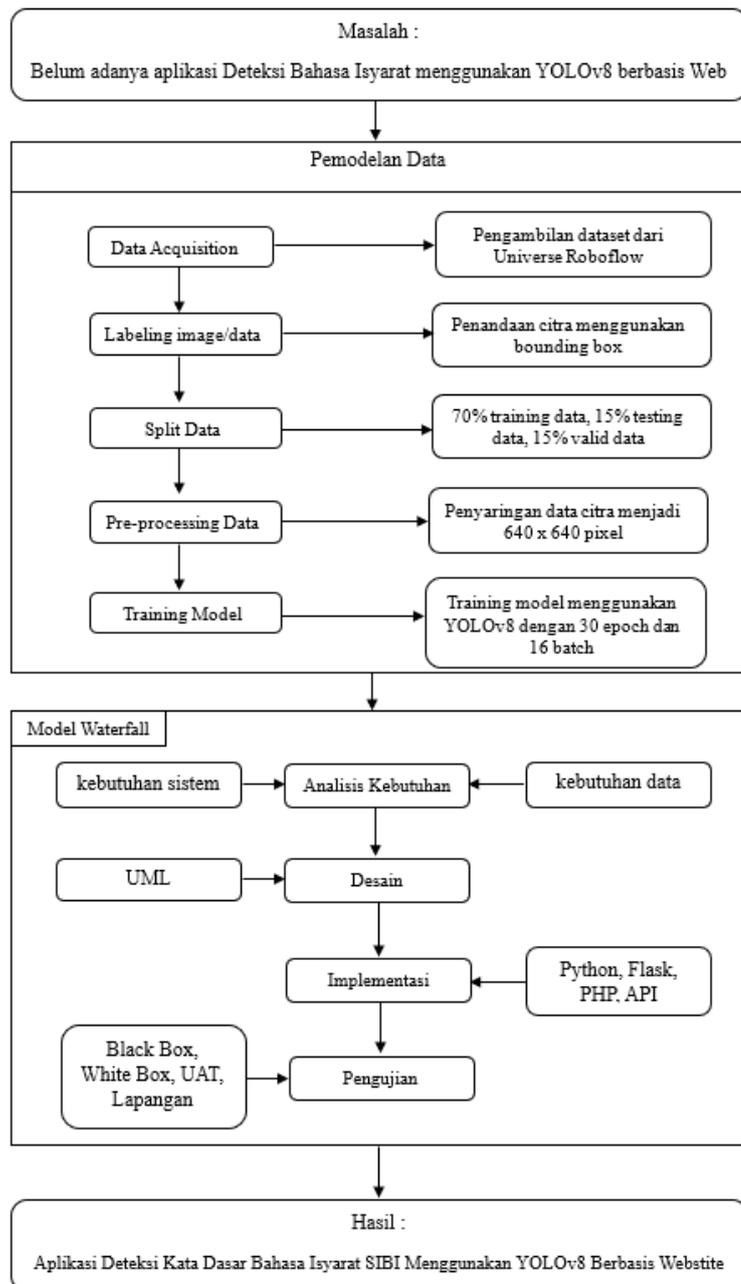
User Acceptance Testing (UAT) merupakan serangkaian pengujian yang dilakukan oleh pengguna akhir, yang umumnya merupakan staf atau karyawan perusahaan yang secara langsung berinteraksi dengan sistem. Tujuan dari UAT adalah untuk memverifikasi bahwa sistem telah berjalan sesuai dengan kebutuhan dan harapan pengguna. Proses ini melibatkan pengguna menggunakan teknik pengujian black box untuk memvalidasi sistem terhadap spesifikasi yang telah ditetapkan. Selama UAT, pengguna bertanggung jawab untuk memastikan bahwa semua fungsionalitas yang relevan telah diuji dan memastikan bahwa perangkat lunak memenuhi standar yang ditetapkan. UAT merupakan salah satu tahap terakhir dan krusial dalam prosedur pengembangan perangkat lunak sebelum produk dinyatakan siap untuk dirilis ke pasar[26].

14. White Box Testing

Pengujian White Box adalah metode pengujian perangkat lunak yang memeriksa struktur internal atau kode sumber aplikasi. Metode ini digunakan untuk mendeteksi kesalahan logis dalam kode program,

membantu debugging, menemukan kesalahan ketik, dan mengungkap asumsi pemrograman yang keliru. Pengujian ini berfokus pada verifikasi jalur, kondisi, dan aliran data dalam kode[27].

C. Kerangka Berpikir



Gambar 2. 2 Kerangka Berpikir

Kerangka berpikir adalah suatu representasi konseptual yang mengilustrasikan bagaimana teori berinteraksi dengan berbagai faktor yang dianggap signifikan. Dalam konteks ini, kerangka berpikir berfungsi sebagai landasan yang mendasari pemahaman yang lebih luas, menjadi dasar bagi pemikiran dan proses penelitian yang akan dilakukan[28].

BAB III

METODE PENELITIAN

A. Pendekatan Penelitian

Pendekatan penelitian merupakan suatu strategi dan prosedur yang dirancang untuk mengarahkan proses penelitian, yang meliputi langkah-langkah yang didasarkan pada asumsi umum guna menentukan metode pengumpulan dan analisis data. Pendekatan penelitian ini memainkan peran penting dalam memfasilitasi identifikasi dan penyelesaian permasalahan yang dihadapi. Dalam penelitian ini, penulis memilih pendekatan penelitian yang mengadopsi metode *waterfall*. Metode *waterfall* dipilih untuk pengembangan sistem dalam penelitian ini karena mencerminkan pendekatan yang sistematis dan terstruktur secara jelas. Prosesnya dimulai dari identifikasi kebutuhan sistem dan berlanjut melalui tahap analisis, desain, pengkodean, pengujian, dan pemeliharaan. Metode *waterfall* dianalogikan seperti air terjun karena ibarat air terjun yang jatuh dari satu tahap ke tahap berikutnya, menyelesaikannya secara berurutan.

B. Fokus Penelitian

Fokus penelitian difokuskan pada pembahasan sebagai berikut:

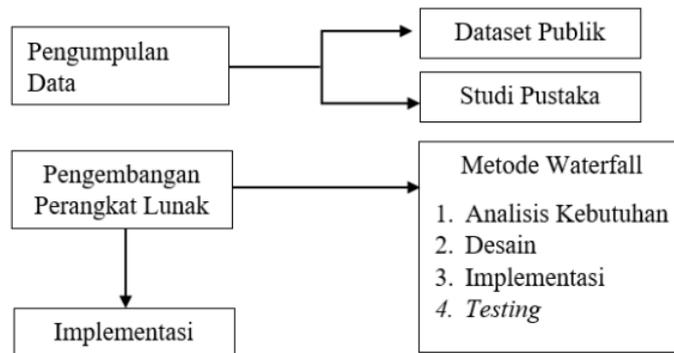
1. Aplikasi berjalan dengan baik dan berbasis website;
2. Aplikasi mendeteksi kata dasar bahasa isyarat SIBI menggunakan *webcam* secara *realtime*;
3. Target pengguna aplikasi adalah orang tunawicara dan masyarakat umum yang ingin belajar bahasa isyarat.

C. Desain Penelitian

Desain penelitian merupakan tahap awal yang sangat krusial dalam proses penelitian karena memberikan landasan yang jelas bagi peneliti dalam mengarahkan seluruh proses penelitian. Dalam konteks ini, skema desain

penelitian dijelaskan dalam Gambar 3.1. Proses penelitian untuk skripsi ini terdiri dari langkah-langkah sebagai berikut:

1. Analisis: Pada tahap ini, penulis mengumpulkan persyaratan sistem yang diperlukan untuk penelitian. Peneliti mengidentifikasi jenis data yang diperlukan dengan memperoleh data yang relevan dan tepat.
2. Perancangan: Pada tahap ini, penulis merancang arsitektur, desain, dan spesifikasi teknis perangkat lunak yang akan dikembangkan. Perancangan melibatkan pembuatan diagram alir dan desain antarmuka pengguna.
3. Pengembangan/Implementasi: Tahap ini melibatkan pembuatan kode program sistem sesuai dengan perancangan yang telah dilakukan sebelumnya. Sistem dikembangkan dalam bentuk unit-program kecil yang kemudian diintegrasikan menjadi kesatuan yang utuh.
4. Pengujian: Tahap ini dilakukan untuk memastikan bahwa perangkat lunak yang dibuat sesuai dengan desainnya dan berfungsi dengan baik. Pengujian dilakukan oleh tim penguji untuk memeriksa kesalahan atau ketidaksesuaian dalam program tersebut.



Gambar 3. 1 Skema Desain Penelitian

D. Teknik Pengumpulan Data

a. Dataset Publik

Dalam metode ini, peneliti melakukan akuisisi data melalui pencarian citra di internet. Dataset yang digunakan dalam penelitian ini

terdiri dari 2.162 citra yang berkaitan dengan Bahasa Isyarat, yang diperoleh dari *website* Roboflow. Dataset ini dibagi menjadi beberapa kategori atau kelas yang mencakup berbagai gerakan bahasa isyarat lainnya. Kelas-kelas tersebut meliputi : adik, aku, aku cinta kamu, ayah, bahagia, baik, bau, berhenti, buruk, cantik, datang, diam, dimana, ibu, kakek, kapan, lari, maaf, selamat tinggal, telpon.

Setelah diunduh, dataset citra dipindahkan ke dalam website Roboflow untuk di augmentasi. Kemudian, penulis melakukan pembagian dataset citra ke dalam data pelatihan sebesar 70%, validasi sebesar 15%, dan pengujian sebesar 15%, yang akan digunakan untuk melatih model deteksi objek YOLOv8 di platform Google Colab.

b. Studi Pustaka

Teknik ini melibatkan proses pencarian, pengumpulan, dan analisis literatur atau karya ilmiah yang berkaitan dengan topik penelitian yang sedang diselidiki. Pendekatan ini bertujuan untuk memberikan pemahaman yang mendalam kepada peneliti tentang topik penelitian serta memperkuat landasan teoritis dari penelitian sebelumnya. Kegiatan ini mencakup:

- a. Menggunakan jurnal sebagai bahan penelitian terkait.
- b. Menggunakan website/internet guna mencari informasi terkait.
- c. Menggunakan buku yang berkaitan dengan penelitian.

E. Teknik Analisis Data

Teknik analisis data yang dilakukan dalam penelitian ini mencakup langkah-langkah berikut:

1. Akuisisi Data: Pengumpulan keseluruhan data citra dilakukan dengan mengunduhnya dari sumber data yang tersedia, yaitu Roboflow.
2. Analisis Kebutuhan Pengguna: Proses ini melibatkan evaluasi terhadap kebutuhan pengguna yang diperlukan untuk pengembangan aplikasi.

3. Pengolahan dan Pemrosesan Data: Data yang terkumpul kemudian diolah dan diproses menggunakan metode atau algoritma yang telah ditentukan untuk menghasilkan aplikasi sesuai dengan rencana yang telah disusun.
4. Kesimpulan: Tahap terakhir adalah pembuatan simpulan akhir yang menggambarkan hasil analisis dan temuan yang didapat dari seluruh proses penelitian.

BAB IV

HASIL PENGEMBANGAN DAN PEMBAHASAN

A. Hasil

Penulis menggunakan metode Waterfall dalam pengembangan aplikasi ini. Tahapan pengembangannya mengikuti urutan yang ada dalam metode ini adalah sebagai berikut:

1. Analisis Kebutuhan

A. Analisis Kebutuhan Data

Data yang diperlukan untuk penelitian terdiri dari citra yang digunakan sebagai sampel pelatihan untuk mengembangkan model deteksi citra. Sampel citra tersebut diunduh dari internet melalui platform Roboflow, kemudian diproses di Roboflow untuk melatih model. Penulis mengambil dataset dari Roboflow dengan nama file “Projek Akhir Satu” yang sebelumnya sudah di modeling yang berjumlah 51 kelas dan berjumlah 11.319 *image*, tapi penulis hanya mengambil 20 kelas 2.162 *image* untuk dilatih ulang. Pada dataset sebelumnya pembuat dataset menggunakan pembagian data menjadi *training data* 82%, *valid data* 9% dan *test data* 9%. Akan tetapi penulis melakukan proses ini melibatkan pembagian data menjadi 75% dari total sampel untuk *training data*, 15% untuk *valid data*, dan 15% untuk *test data*. Untuk jumlah image bisa dilihat pada Tabel 4.1

Tabel 4. 1 Jumlah Image Dataset

No.	Kelas	Jumlah Image
1.	Adik	133
2.	Aku	90
3.	Aku cinta kamu	116
4.	Ayah	136

5.	Bahagia	166
6.	Baik	102
7.	Bau	103
8.	Berhenti	96
9.	Buruk	123
10.	Cantik	89
11.	Datang	175
12.	Diam	117
13.	Dimana	101
14.	Ibu	112
15.	Kakek	116
16.	Kapan	127
17.	Lari	89
18.	Maaf	123
19.	Selamat tinggal	105
20.	Telpon	112

Jumlah raw dataset (dataset mentah) dan trained dataset pada platform Roboflow melakukan augmentasi data secara otomatis untuk meningkatkan variasi data yang tersedia untuk pelatihan. Hal ini dapat menghasilkan dataset yang lebih besar atau jumlah gambar yang berbeda dari yang dimulai. Oleh karena itu, jumlah dataset yang sudah di latih biasanya akan bertambah atau berkurang.

B. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem adalah langkah penting dalam pengembangan perangkat lunak yang bertujuan untuk memahami, mendokumentasikan, dan mengatur persyaratan yang harus dipenuhi oleh sistem tersebut. Tujuan dari analisis kebutuhan sistem adalah untuk

mengidentifikasi permasalahan yang akan diatasi atau kebutuhan yang akan dipenuhi oleh sistem yang akan dikembangkan. Dalam konteks penelitian ini, berikut adalah kebutuhan yang akan digunakan oleh peneliti untuk merancang aplikasi:

1) Kebutuhan Perangkat Keras *Computer*

Berikut ini adalah spesifikasi perangkat keras minimum yang mendukung aplikasi kata dasar bahasa isyarat SIBI, yaitu :

Tabel 4. 2 Kebutuhan Perangkat Keras Computer

RAM	GPU	CPU	Penyimpanan	Resolusi
8GB (minimal)	Nvidia GeForce GTX 1050 / AMD Radeon RX 560 atau yang lebih tinggi	Intel Core i5 / AMD Ryzen 3atau yang lebih tinggi	20GB (Minimal Tersedia)	1280x720 (Minimal)

2) Kebutuhan Perangkat Lunak *Computer*

Tabel 4. 3 Kebutuhan Perangkat Lunak Computer

Sistem Operasi	Aplikasi
Windows 10 / 11	- Google Colaboratory - Roboflow - Visual Studio Code - Figma - Chrome Browser

	<ul style="list-style-type: none"> - Draw.io - Google Drive
--	---

C. Analisis Kebutuhan Fungsional

- 1) Halaman awal pada aplikasi yaitu Beranda yang berfungsi untuk memberikan informasi bahwa aplikasi siap digunakan.
- 2) Aplikasi dapat menampilkan halaman Kamus untuk memberikan pemahaman kepada pengguna tentang gerakan bahasa isyarat.
- 3) Aplikasi dapat menampilkan halaman Deteksi untuk deteksi bahasa isyarat secara *realtime*.
- 4) Aplikasi dapat menampilkan halaman Tentang untuk pengguna bisa mengetahui siapa yang membuat aplikasi.

2. Desain

Tujuan dari desain sistem ini adalah untuk menyediakan panduan yang jelas dan terperinci selama proses implementasi atau penulisan kode aplikasi[29]. Desain yang dibuat oleh penulis bertujuan untuk mengembangkan aplikasi deteksi kata dasar SIBI (Sistem Isyarat Bahasa Indonesia) antara lain :

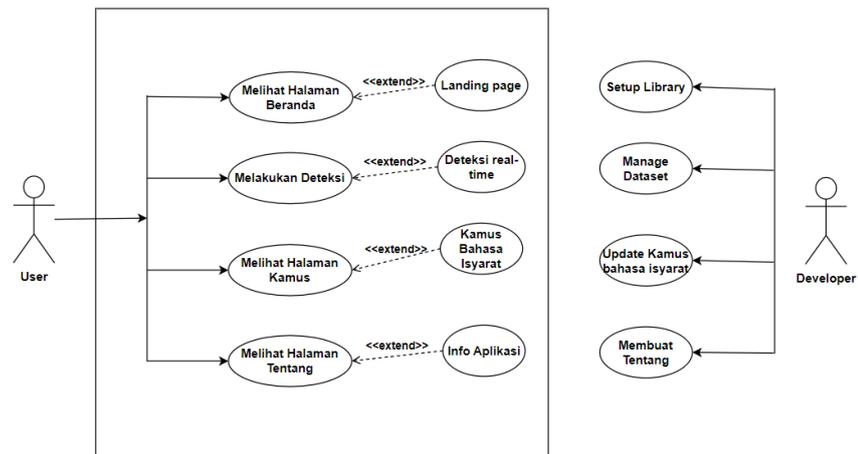
A. Desain *Unified Modeling Language* (UML)

Dalam desain *Unified Modeling Language* (UML), penulis menggunakan berbagai jenis diagram untuk menggambarkan berbagai aspek aplikasi atau sistem yang akan dikembangkan. Beberapa jenis diagram UML yang digunakan oleh penulis antara lain:

1) *Use Case Diagram*

Dalam desain *Unified Modeling Language* (UML), diagram *Use Case* digunakan untuk menjabarkan interaksi antara aktor-aktor eksternal dengan sistem yang sedang dikembangkan. Dengan

menggunakan diagram Use Case, penulis dapat memperoleh gambaran yang jelas tentang fungsionalitas yang diperlukan oleh sistem dan bagaimana aktor akan berinteraksi dengan sistem tersebut. Diagram Use Case ini tersedia dalam Gambar 4.1.



Gambar 4. 1 Use Case Diagram

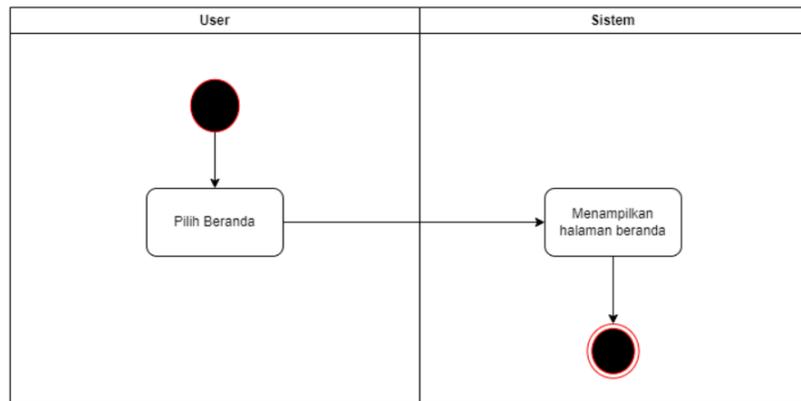
Berdasarkan Diagram pada gambar 4.1, dapat disimpulkan bahwa dalam sistem, terdapat dua peran utama, yakni Pengguna (*User*) dan Pengembang (*Developer*). Pengguna memiliki otoritas penuh atas penggunaan aplikasi, mulai dari mengaksesnya hingga menutupnya. Mereka juga dapat mengakses dan membaca kamus tentang bahasa isyarat serta melakukan deteksi langsung secara *real-time*. Di sisi lain, Pengembang memiliki wewenang untuk mengupdate *website* dan mengelola dataset model.

2) Activity Diagram

Diagram ini adalah gambaran visual dari alur kerja atau proses yang menggambarkan serangkaian aktivitas sebagai simpul dan hubungan antara aktivitas tersebut dalam bentuk panah. Diagram semacam ini umumnya digunakan dalam pengembangan perangkat lunak untuk merepresentasikan langkah-langkah yang

diperlukan untuk menyelesaikan tugas tertentu atau rangkaian aktivitas dalam sistem. Diagram tersebut membantu pengembang dan pemangku kepentingan dalam memahami urutan aktivitas, pengambilan keputusan, dan aliran kontrol dalam proses yang telah ditentukan.

a) *Activity Diagram Halaman*



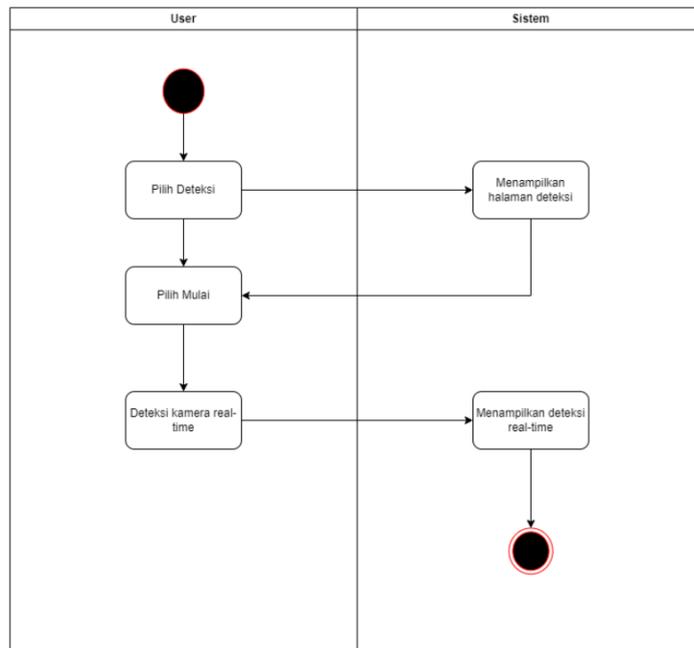
Gambar 4. 2 Activity Diagram Halaman

Gambar 4.2 menggambarkan situasi di mana ketika pengguna mengakses aplikasi, halaman utama akan ditampilkan dengan fitur Beranda yang bertujuan untuk memperkenalkan pengguna terhadap aplikasi. Pengguna diberi kesempatan untuk membaca informasi ringkas yang tersedia pada situs web tersebut.

b) *Activity Diagram Deteksi*

Diagram pada Gambar 4.3 berfungsi untuk melakukan deteksi bahasa isyarat, yang merupakan salah satu fitur utama dalam aplikasi yang sedang dalam tahap pengembangan oleh penulis. Halaman ini menyediakan opsi kepada pengguna untuk melakukan deteksi secara langsung (real-time). Ketika pengguna

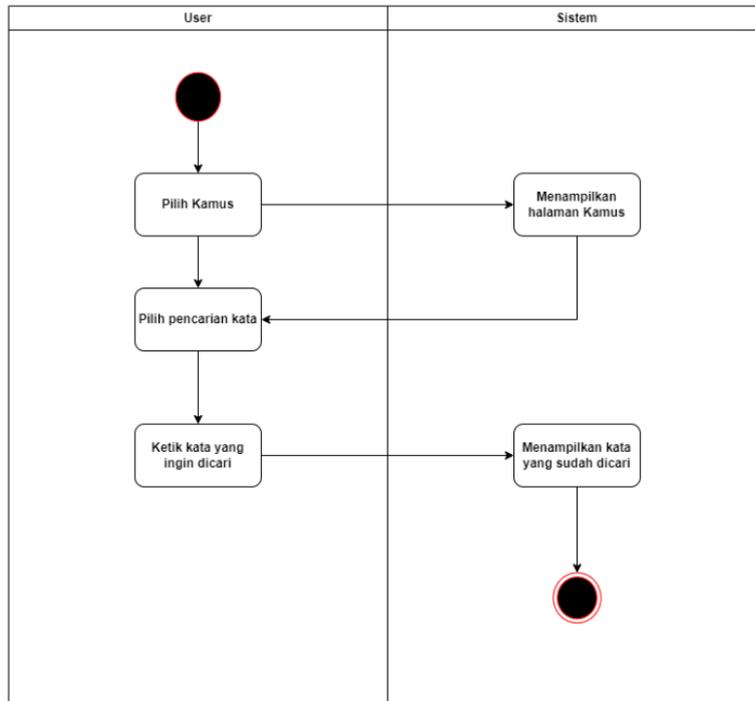
membuka fitur deteksi setelah mengakses web aplikasi, sistem akan secara otomatis mengaktifkan webcam untuk mendeteksi gerakan bahasa isyarat. Setelah webcam diaktifkan, diperlukan sedikit waktu untuk memulai deteksi gerakan bahasa isyarat hingga hasil deteksi ditampilkan dalam frame per detik (FPS).



Gambar 4. 3 Activity Diagram Deteksi

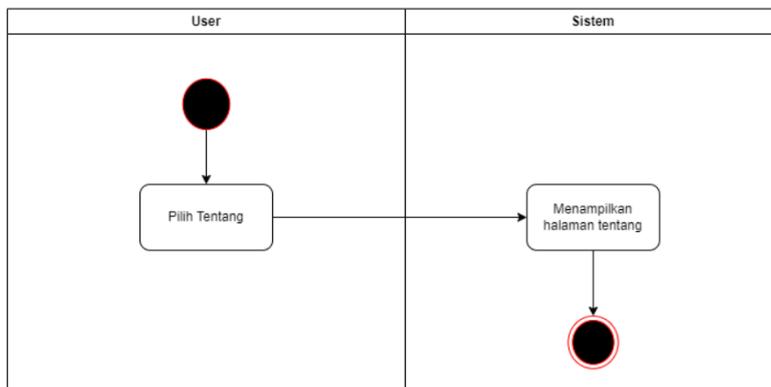
c) *Activity Diagram Kamus*

Berdasarkan gambar 4.4 ketika pengguna membuka aplikasi dan memilih halaman Kamus maka sistem akan menampilkan isi halaman Kamus, dimana di menu ini terdapat fitur pencarian (search) yang akan memudahkan pengguna untuk mencari kata bahasa isyarat.



Gambar 4. 4 Activity Diagram Kamus

d) *Activity Diagram* Tentang



Gambar 4. 5 Activity Diagram Tentang

Berdasarkan gambar 4.5 ketika pengguna membuka aplikasi dan memilih halaman Tentang maka sistem akan menampilkan isi halaman Tentang, dimana di menu ini terdapat

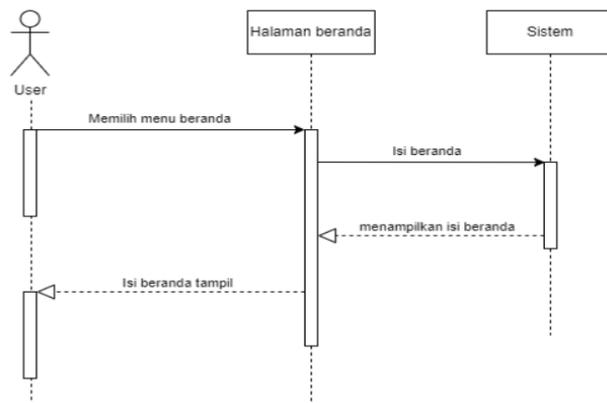
keterangan tentang aplikasi deteksi Bahasa Isyarat yang sedang dikembangkan oleh penulis.

3) *Sequence Diagram*

Diagram urutan atau sequence adalah alat visual yang digunakan untuk menggambarkan interaksi antara objek-objek dalam suatu sistem secara berurutan sepanjang waktu. Dalam diagram ini, objek-objek yang terlibat dalam interaksi diwakili oleh garis vertikal yang disebut "*lifeline*", yang menunjukkan durasi keberadaan objek tersebut selama proses berlangsung.

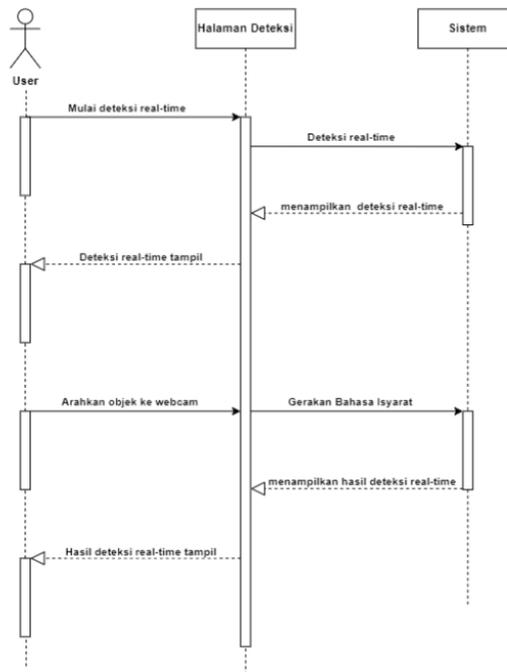
a) *Sequence Diagram Beranda*

Menurut diagram pada Gambar 4.6, langkah awalnya adalah *user* memilih jenis artikel yang ingin dibaca, kemudian sistem akan menampilkan konten artikel yang dipilih oleh *user* tersebut.



Gambar 4. 6 Sequence Diagram Beranda

b) *Sequence Diagram* Deteksi



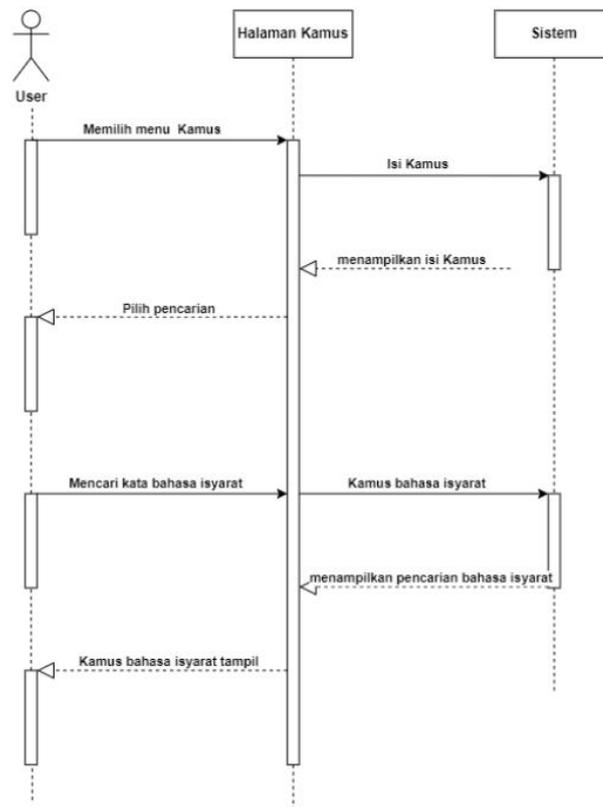
Gambar 4. 7 Sequence Diagram Deteksi

Berdasarkan diagram pada Gambar 4.7, proses dimulai dengan pengguna memulai deteksi real-time dengan membuka kamera web. Selanjutnya, pengguna hanya perlu mengarahkan gerakan tangan bahasa isyarat ke kamera web untuk mendapatkan hasil deteksi secara real-time. Deteksi akan langsung muncul dengan bounding box di gerakan bahasa isyarat yang tepat.

c) *Sequence Diagram* Kamus

Berdasarkan Gambar 4.8, proses pertama pengguna memilih menu Kamus, berikutnya sistem akan menampilkan isi menu Kamus. Pengguna bisa mencari kata bahasa isyarat melalui fitur pencarian yang ada pada halaman Kamus.

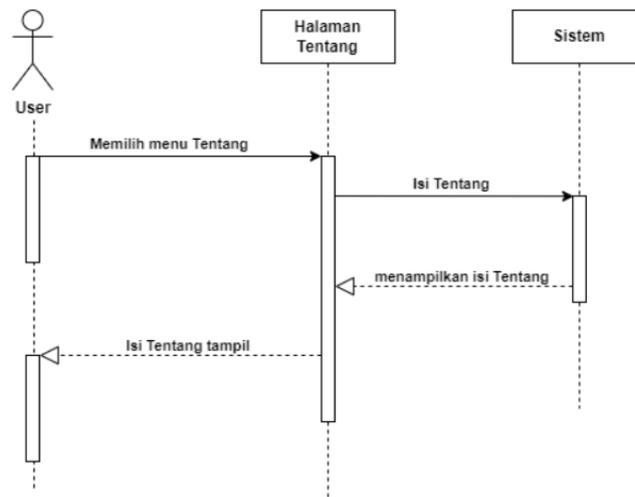
Pengguna hanya perlu ketik kata bahasa isyarat apa yang ingin dicari



Gambar 4. 8 Sequence Diagram Kamus

d) *Sequence Diagram* Tentang

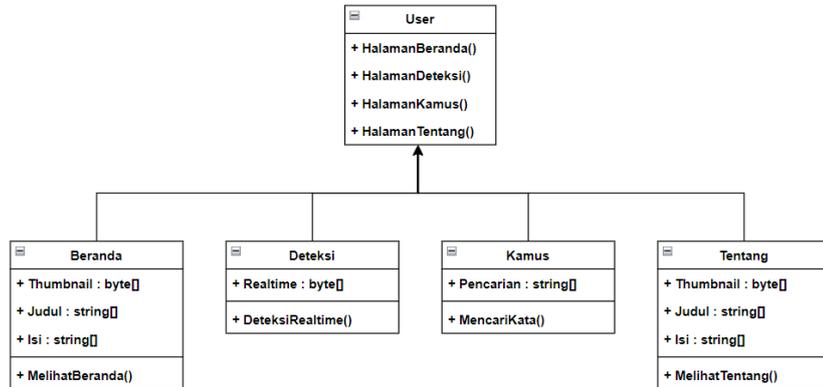
Berdasarkan Gambar 4.9, proses pertama pengguna memilih menu Tentang, berikutnya sistem akan menampilkan isi menu Tentang.



Gambar 4. 9 Sequence Diagram Tentang

4) *Class Diagram*

Class Diagram adalah jenis diagram dalam Unified Modeling Language (UML) yang bertujuan untuk menggambarkan struktur statis suatu sistem atau aplikasi. Diagram ini mengilustrasikan kelas-kelas yang ada dalam sistem, hubungan antara kelas-kelas tersebut, serta atribut dan metode yang dimiliki oleh setiap kelas. Elemen-elemen penting dalam Class Diagram mencakup kelas, atribut, metode, dan relasi antar kelas. Tujuan dari Class Diagram adalah menyajikan gambaran yang jelas tentang struktur kelas dalam sistem, sehingga memfasilitasi pemahaman tentang bagaimana sistem beroperasi dan bagaimana komponen-komponen dalam sistem berinteraksi satu sama lain. Berikut ini adalah Class Diagram yang menggambarkan deteksi kata dasar bahasa isyarat SIBI menggunakan YOLO v8 Berbasis *Website*:



Gambar 4. 10 Class Diagram

B. Desain User Interface

Dalam desain *User Interface* (UI) untuk aplikasi deteksi kata dasar SIBI (Sistem Isyarat Bahasa Indonesia), langkah pertama yang diambil oleh penulis adalah merancang desain *wireframe*. *Wireframe* adalah representasi visual sederhana dari antarmuka pengguna yang menunjukkan struktur dasar situs web atau aplikasi tanpa warna, gambar, atau desain yang rumit. *Wireframe* digunakan untuk menunjukkan tata letak, susunan komponen utama, dan fungsi antarmuka pengguna tanpa memperhatikan detail visual yang rumit. Beberapa desain antarmuka pengguna yang disertakan dalam wireframe ini meliputi desain beranda, desain halaman deteksi, desain halaman kamus, dan desain halaman tentang.

1) Desain Halaman Beranda

Pada Gambar 4.11, Halaman Beranda adalah halaman pertama yang muncul saat membuka aplikasi. Pada Beranda ini, terdapat beberapa komponen utama yang mencakup kalimat selamat datang dan tombol mulai untuk memulai penggunaan aplikasi deteksi bahasa isyarat. Di bagian atas halaman terdapat header berupa beranda, deteksi, kamus, dan tentang untuk bisa berpindah

ke halaman satu ke halaman lainnya. Di bawah *header*, terdapat *tagline* aplikasi yang singkat namun informatif, menjelaskan apa yang dapat dilakukan oleh aplikasi tersebut. Sedangkan di bagian bawah *tagline*, terdapat tombol mulai yang memungkinkan pengguna untuk memulai penggunaan aplikasi deteksi bahasa isyarat dengan sekali klik. Dibawah tombol mulai terdapat beberapa fitur kamus bahasa isyarat dan deteksi bahasa isyarat yang berupa tombol agar bisa langsung mengarahkan ke halaman kamus dan deteksi. Yang terakhir adalah footer, yang berisikan beberapa info aplikasi. Dengan demikian, Beranda menjadi pengantar yang sempurna untuk menyambut pengguna saat mereka memulai pengalaman mereka dengan aplikasi bahasa isyarat.

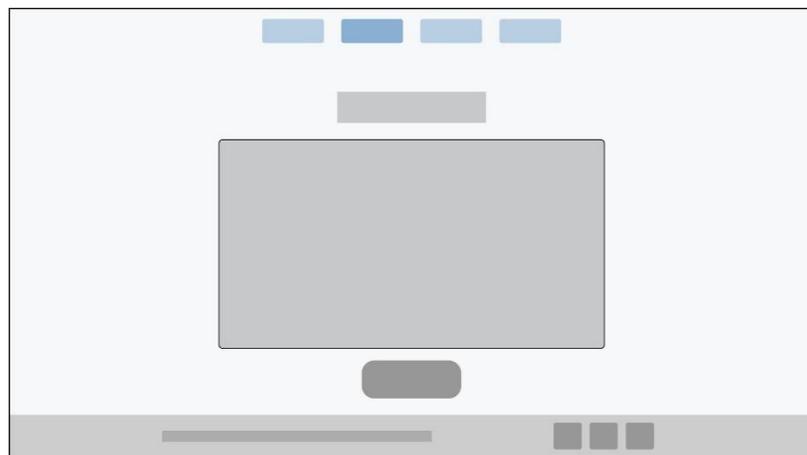


Gambar 4. 11 Desain User Interface Halaman Beranda

2) Desain Halaman Deteksi

Pada Gambar 4.12 menampilkan sebuah halaman yang memungkinkan pengguna untuk menggunakan fitur deteksi bahasa isyarat secara langsung. Halaman ini dilengkapi dengan header dan

footer yang serupa dengan yang terdapat pada halaman Beranda. Fokus utama dari halaman ini adalah pada fitur kamera *real-time* yang memungkinkan pengguna untuk melakukan deteksi secara langsung melalui kamera perangkat mereka. Hasil deteksi ditampilkan secara langsung pada layar kamera. Pengguna dapat mengaktifkan fitur deteksi hanya dengan menekan tombol mulai. Dengan adanya komponen-komponen ini, pengguna dapat dengan mudah mengakses dan menggunakan fitur deteksi dengan efektif.

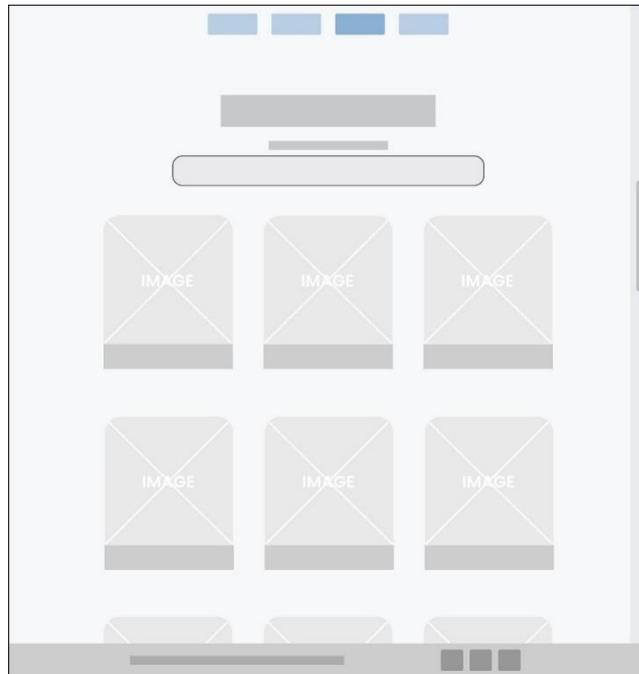


Gambar 4. 12 Desain User Interface Halaman Deteksi

3) Desain Halaman Kamus

Gambar 4.13 menampilkan sebuah halaman yang mempertahankan struktur header dan footer yang serupa dengan yang terdapat pada halaman Beranda. Fungsionalitas utama halaman ini adalah menampilkan serangkaian gerakan bahasa isyarat dalam kamus terbatas, yang mencakup 20 gerakan. Selain itu, terdapat juga sebuah fitur pencarian yang memungkinkan pengguna untuk mencari gerakan bahasa isyarat yang spesifik. Fitur ini dirancang untuk mempermudah pengguna dalam menemukan gerakan bahasa isyarat yang mereka cari. Dengan demikian, halaman ini bertujuan

untuk menyediakan akses yang mudah dan terorganisir terhadap kamus gerakan bahasa isyarat bagi pengguna aplikasi.

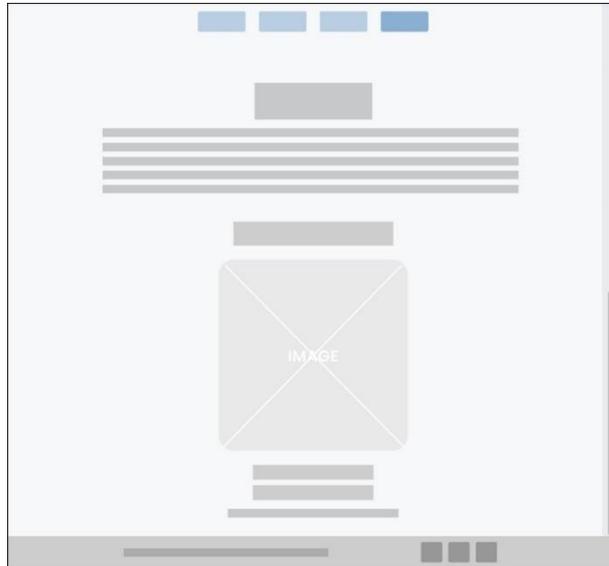


Gambar 4. 13 Desain User Interface Halaman Kamus

4) Desain Halaman Tentang

Gambar 4.14 menampilkan sebuah halaman yang menonjolkan komponen-komponen penting dalam menyampaikan informasi kepada pengguna mengenai aplikasi. Dengan konsistensi, halaman ini menyertakan header dan footer di bagian atas dan bawah, sesuai dengan tata letak yang diperkenalkan pada halaman Beranda. Konten utama halaman ini terletak di tengah, berfokus pada penyediaan informasi yang komprehensif mengenai aplikasi tersebut. Ini termasuk deskripsi singkat mengenai aplikasi, penjelasan tentang fitur-fitur utama yang disediakan, serta tujuan dari penggunaan aplikasi tersebut. Melalui penyajian ini, pengguna

diberikan pemahaman yang lebih mendalam mengenai apa yang dapat mereka harapkan saat menggunakan aplikasi tersebut.



Gambar 4. 14 Desain User Interface Halaman Tentang

Sebagai tambahan, halaman ini juga menampilkan foto developer yang bertanggung jawab atas pembuatan aplikasi. Kehadiran foto ini bertujuan untuk memberikan identitas dan transparansi kepada pengguna, memungkinkan mereka untuk mengenali siapa yang bertanggung jawab atas pengembangan aplikasi yang mereka gunakan. Dengan demikian, halaman ini tidak hanya berfungsi sebagai sumber informasi yang berharga, tetapi juga memperkuat koneksi antara pengguna dan pengembang aplikasi.

3. Implementasi Model

A. *Data Acquisition*

Dalam penelitian ini, pengumpulan data dilakukan melalui teknik scrapping data dari Google, dimana informasi terkait dikumpulkan dan dataset diunduh dari situs web Roboflow dengan judul

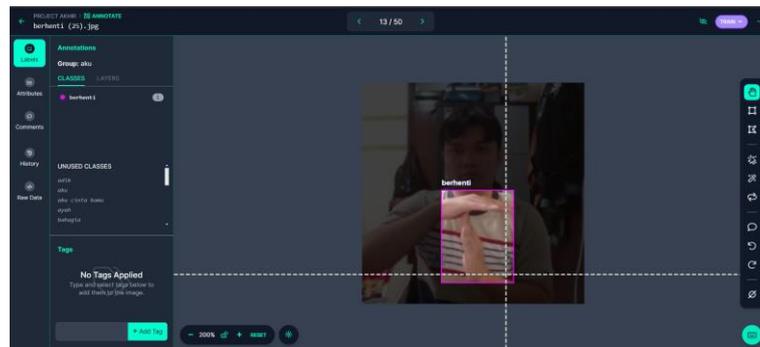
"Projek Akhir satu". Data yang terkumpul terdiri dari 20 kelas yang berbeda, yaitu adik, aku, aku cinta kamu, ayah, bahagia, baik, bau, berhenti, buruk, cantik, datang, diam, dimana, ibu, kakek, kapan, lari, maaf, selamat tinggal, dan telpon. Contoh dari beberapa kelas dataset ditampilkan dalam Gambar 4.15.



Gambar 4. 15 Dataset Class

B. *Labelling Data*

Proses *Labelling Data* dilaksanakan melalui penandaan citra menggunakan kotak pembatas (*bounding box*), sambil menetapkan kelas yang relevan pada setiap citra. Metode *Labelling Data* dilakukan dengan menggunakan website Roboflow, di mana setiap citra diberikan kotak pembatas untuk menandai area yang signifikan. Proses ini memungkinkan pengayaan dataset dengan informasi yang diperlukan untuk pelatihan dan penilaian model dalam analisis lanjutan citra. Berikut beberapa contoh dalam melakukan *Labelling Data* dapat dilihat pada Gambar 4.16 :



Gambar 4. 16 Labelling Data

C. *Split Data*

Pada tahap pengelompokan data, dataset disegmentasikan menjadi tiga bagian, yakni dataset pelatihan (*training*) dataset validasi (*validation*) dan dataset pengujian (*testing*). Dataset pelatihan merangkap sebanyak 70% dari total keseluruhan dataset, sementara dataset validasi terdiri dari 15% dari keseluruhan dataset dan dataset pengujian terdiri dari 15% dari keseluruhan dataset. Dataset pelatihan validasi dan pengujian ini akan dimanfaatkan dalam proses pelatihan dan pembuatan model. Pembagian ini menjadi penting dalam mengoptimalkan performa model, memastikan bahwa model telah melalui pembelajaran dari sejumlah besar data yang beragam secara memadai, serta mampu diuji dengan menggunakan data yang belum pernah dipertimbangkan sebelumnya. Berikut merupakan rincian *Split Data* untuk *training*, *validation*, dan *testing data folder*.

a. *Training Data Folder*

Training data folder digunakan untuk menyimpan data citra yang digunakan untuk melatih model. Untuk pembagian jumlah file citra di training data folder pada penelitian ini dapat dilihat pada Tabel 4.4.

Tabel 4. 4 Data Pelatihan atau Training

No.	Kelas	Jumlah Training
1.	Adik	97
2.	Aku	59
3.	Aku cinta kamu	84
4.	Ayah	95
5.	Bahagia	121
6.	Baik	69
7.	Bau	65
8.	Berhenti	73
9.	Buruk	89
10.	Cantik	60
11.	Datang	131
12.	Diam	89
13.	Dimana	66
14.	Ibu	81
15.	Kakek	77
16.	Kapan	87
17.	Lari	64
18.	Maaf	86
19.	Selamat tinggal	68
20.	Telpon	78

b. *Validation Data Folder*

Validation data folder digunakan untuk menyimpan data yang digunakan untuk mengevaluasi kinerja model saat proses pelatihan. Untuk pembagian jumlah file citra di validation data folder pada penelitian ini dapat dilihat pada Tabel 4.5.

Tabel 4. 5 Data Validasi

No.	Kelas	Jumlah Valid
1.	Adik	16
2.	Aku	16
3.	Aku cinta kamu	16
4.	Ayah	19
5.	Bahagia	27
6.	Baik	16
7.	Bau	21
8.	Berhenti	9
9.	Buruk	14
10.	Cantik	18
11.	Datang	18
12.	Diam	17
13.	Dimana	18
14.	Ibu	14
15.	Kakek	21
16.	Kapan	18
17.	Lari	15
18.	Maaf	18
19.	Selamat tinggal	21
20.	Telpon	15

c. *Testing Data Folder*

Test data folder digunakan untuk menyimpan data citra yang digunakan untuk menguji kinerja final model setelah proses pelatihan. Untuk 36 pembagian jumlah file citra di test data folder pada penelitian ini dapat dilihat pada Tabel 4.6.

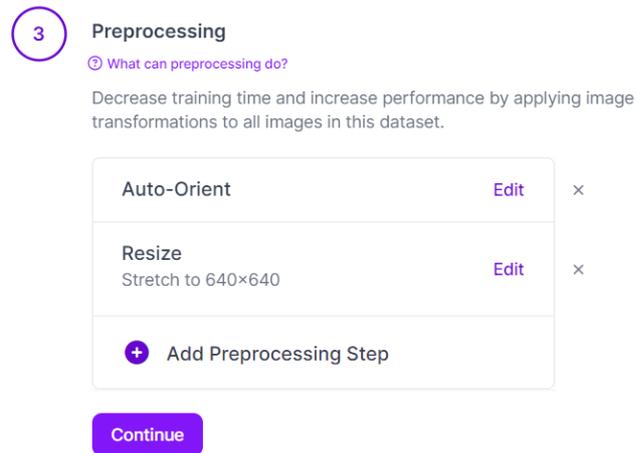
Tabel 4. 6 Data Test

No.	Kelas	Jumlah Test
1.	Adik	20
2.	Aku	15
3.	Aku cinta kamu	16
4.	Ayah	22
5.	Bahagia	18
6.	Baik	17
7.	Bau	17
8.	Berhenti	14
9.	Buruk	20
10.	Cantik	11
11.	Datang	26
12.	Diam	11
13.	Dimana	17
14.	Ibu	17
15.	Kakek	18
16.	Kapan	22
17.	Lari	10
18.	Maaf	19
19.	Selamat tinggal	16
20.	Telpon	19

D. *Pre-processing Data*

Tahap *preprocessing data* (pra-pemrosesan) adalah tahap penting dalam analisis data yang bertujuan untuk membersihkan, mengubah, dan mempersiapkan data mentah agar sesuai untuk analisis lebih lanjut. Tahap *preprocessing* dapat mencakup beberapa langkah, tergantung pada jenis data dan tujuan analisisnya. Pada tahap ini,

dilakukan penyaringan data yang tepat untuk disertakan dalam *dataset* dan dilakukan penyesuaian ukuran setiap citra menjadi 640 x 640 *pixel*. Tahap *preprocessing data* dapat dilihat pada gambar 4.17:



Gambar 4. 17 Tahap preprocessing data

E. *Training Model*

Proses pelatihan model dalam penelitian ini menggunakan YOLOv8. Dataset yang telah dipersiapkan untuk pelatihan diunggah ke platform Roboflow untuk selanjutnya dilakukan proses pelatihan dengan menghubungkan ke platform Google Colaboratory. Model dilatih menggunakan 2.162 gambar yang terbagi menjadi 20 kelas, antara lain sebagai berikut: adik, aku, aku cinta kamu, ayah, bahagia, baik, bau, berhenti, buruk, cantik, datang, diam, dimana, ibu, kakek, kapan, lari, maaf, selamat tinggal, telpon.. Proses pelatihan menggunakan 30 putaran (epochs) dengan batch size 16.

Dataset yang sudah selesai dilakukan proses *generate* data di Roboflow, maka roboflow akan membuat versi dari dataset tersebut. Versi yang sudah dibuat kemudian *diexport* untuk dilanjutkan ke tahap selanjutnya untuk dilakukan proses custom training dengan memilih format model YOLOv8. *Custom train model* dilakukan menggunakan

platform Google Colaboratory dengan menuliskan source code untuk export versi dataset yang telah siap dilatih dari Roboflow. Training model dapat dilihat pada gambar 4.18:

```

30 epochs completed in 0.341 hours.
Optimizer stripped from runs/detect/train2/weights/last.pt, 6.3MB
Optimizer stripped from runs/detect/train2/weights/best.pt, 6.3MB

Validating runs/detect/train2/weights/best.pt...
Ultralytics YOLOv8.0.196 Python-3.10.12 torch-2.3.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 3809548 parameters, 0 gradients, 8.1 GFLOPs

```

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95
all	325	347	0.959	0.943	0.972	0.68
adik	325	16	0.943	0.875	0.933	0.638
aku	325	16	1	0.917	0.966	0.605
aku cinta kamu	325	16	1	1	0.995	0.811
ayah	325	19	1	0.947	0.974	0.711
bahagia	325	27	0.95	0.926	0.973	0.634
baik	325	16	0.841	0.938	0.9	0.455
bau	325	21	0.885	0.857	0.96	0.639
berhenti	325	9	1	1	0.995	0.746
buruk	325	14	0.977	1	0.995	0.779
cantik	325	18	1	1	0.995	0.662
datang	325	18	1	0.854	0.967	0.756
diam	325	17	0.911	0.941	0.964	0.736
dimana	325	18	0.947	1	0.978	0.718
ibu	325	14	0.943	1	0.995	0.529
kakek	325	21	1	1	0.995	0.594
kapan	325	18	0.996	0.944	0.971	0.722
lari	325	15	0.932	0.916	0.946	0.633
maaf	325	18	0.973	0.944	0.992	0.773
selamat tinggal	325	21	0.957	0.952	0.993	0.82
telpon	325	15	0.928	0.855	0.951	0.632

```

Speed: 0.4ms preprocess, 2.8ms inference, 0.0ms loss, 4.8ms postprocess per image
Results saved to runs/detect/train2
Learn more at https://docs.ultralytics.com/modes/train

```

Gambar 4. 18 Training Model

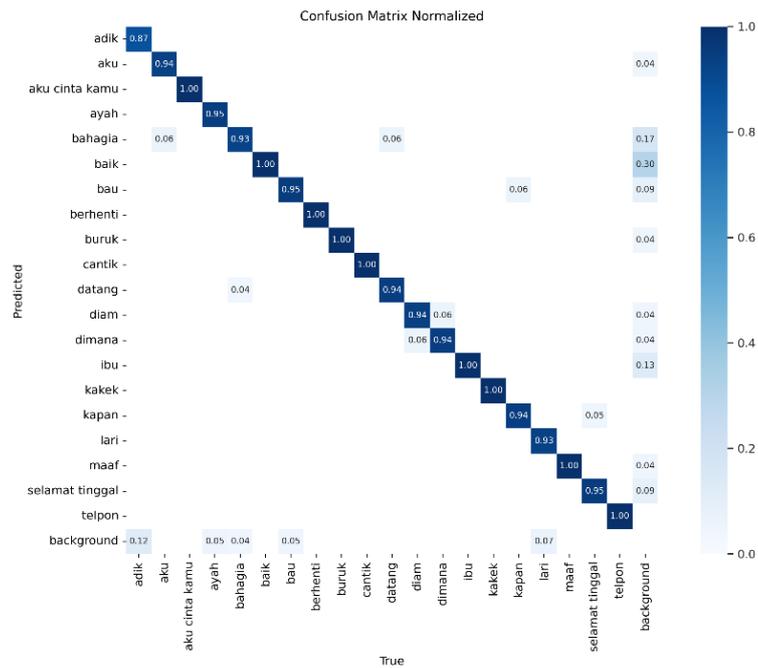
F. Evaluasi Model

Setelah menyelesaikan proses pelatihan model, langkah berikutnya adalah melakukan evaluasi model. Evaluasi model bertujuan untuk menilai seberapa baik kemampuan model dalam menggeneralisasi pola yang telah dipelajari dari data pelatihan terhadap data baru yang tidak pernah dilihat sebelumnya.

1) Confusion Matrix Normalized

Confusion Matrix Normalized adalah alat evaluasi yang digunakan untuk mengukur kinerja deteksi objek dengan membandingkan hasil prediksi model terhadap label yang sebenarnya pada dataset validasi. Matriks ini menampilkan jumlah prediksi yang benar (*true positives*), jumlah prediksi yang salah (*false positives*), serta objek yang gagal terdeteksi (*false negatives*) untuk setiap kelas objek yang diprediksi. Dengan menganalisis matriks kebingungan, pengembang dapat memperoleh wawasan

yang lebih mendalam tentang kelemahan dan kekuatan model dalam mengidentifikasi berbagai kelas objek. Hasil dari *Confusion Matrix Normalized* ini dapat dilihat pada Gambar 4.19.



Gambar 4. 19 Confusion Matrix Normalized

Untuk menghitung akurasi *Confusion Matrix* dilakukan dengan menerapkan perhitungan persamaan 1 pada rumus berikut ini:

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 4. 20 Perhitungan Confusion Matrix

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

$$Accuracy = \frac{87+94+100+95+93+100+95+100+100+100+94+94+94+100+100+94+93+100+95+100}{1852+12+5+4+5+7} \times 100\%$$

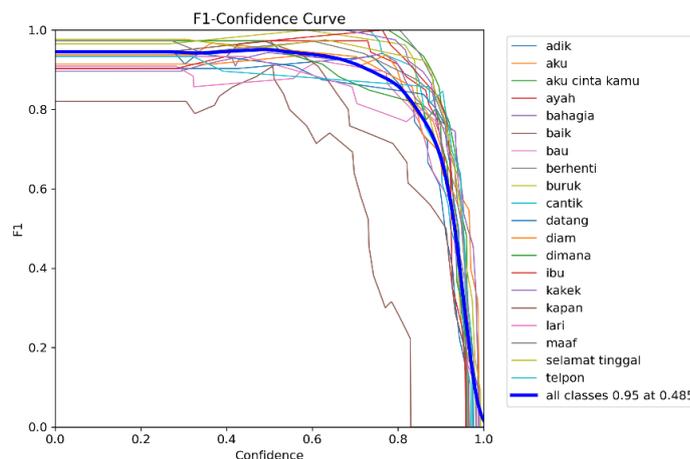
$$Accuracy = \frac{1852}{1852 + 33} \times 100\%$$

$$= 0,98 \times 100\% = 98\%$$

Hasil akurasi yang diperoleh untuk seluruh data dalam proses pelatihan model dengan 30 *epoch* adalah 98%. Akurasi dianggap baik jika mencapai 76%-100%, cukup baik jika beradadi kisaran 55%-75%, kurang baik jika berada di kisaran 40%- 56%, dan tidak baik jika di bawah 40%. Hal ini menunjukkan bahwa model sangat baik dan akurat dalam mendeteksi bahasa isyarat.

2) *F1-Confidence Curve*

F1-Confidence Curve adalah representasi grafis yang mengilustrasikan korelasi antara F1-score (sebuah metrik yang mengukur keseimbangan antara presisi dan recall) dengan tingkat kepercayaan (*confidence level*) dari model klasifikasi. Kurva ini memberikan gambaran tentang sejauh mana model mampu menjaga keseimbangan antara presisi dan recall pada berbagai tingkat kepercayaan. Dengan mengamati *F1-Confidence Curve*, pengguna dapat memilih ambang (*threshold*) yang tepat untuk menyesuaikan *trade-off* antara presisi dan recall sesuai dengan kebutuhan khusus dari aplikasi atau skenario penggunaan tertentu. Hasil dari *F1-Confidence Curve* ini tersedia dalam Gambar 4.21.



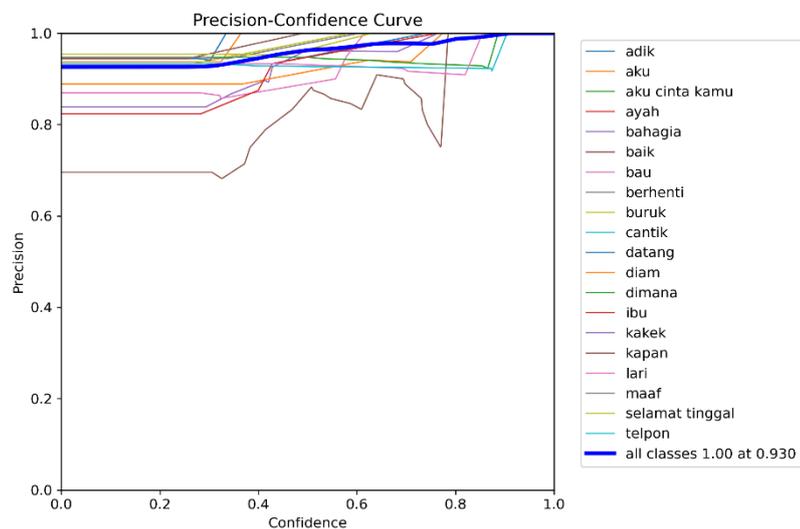
Gambar 4. 21 F1-Confidence Curve

Grafik F1 yang dihasilkan dari proses pelatihan model YOLOv8 dengan 30 *epoch* menunjukkan bahwa skor F1 mencapai nilai rata-rata tertinggi sebesar 0,95 pada nilai *confidence* 0,485. Grafik ini memperlihatkan hubungan antara nilai *confidence* dan skor F1, yang mencerminkan kemampuan model dalam menyeimbangkan presisi dan recall pada berbagai tingkat keyakinan

dalam prediksi. Nilai *confidence* 0,485 merupakan titik di mana model mencapai keseimbangan optimal antara presisi dan recall, yang menghasilkan skor F1 tertinggi.

3) *Precision-Confidence Curve*

Precision-Confidence Curve adalah alat visualisasi yang menggambarkan hubungan antara tingkat presisi dan tingkat kepercayaan (*confidence level*) dari model klasifikasi. Kurva ini memberikan pemahaman tentang bagaimana presisi model berubah seiring dengan perubahan tingkat kepercayaan. Dengan menganalisis *Precision-Confidence Curve*, pengguna dapat memahami trade-off antara tingkat presisi dan tingkat kepercayaan dari model, serta memilih ambang (*threshold*) yang sesuai untuk memenuhi kebutuhan spesifik dari aplikasi atau kasus penggunaan tertentu. Hasil dari *Precision-Confidence Curve* ini tersedia dalam Gambar 4.22.



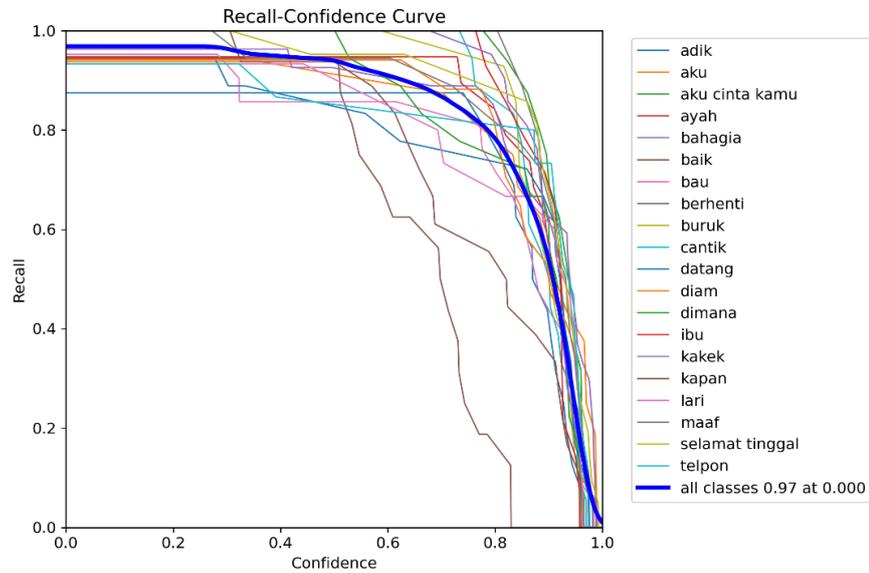
Gambar 4. 22 Precision-Confidence Curve

Grafik di atas memvisualisasikan nilai precision terhadap nilai confidence. Dari hasil pelatihan yang diperoleh, ditemukan bahwa nilai precision mencapai rata-rata maksimal sebesar 1,00 pada nilai confidence 0,930. Hal ini menandakan bahwa pada titik confidence tersebut, model mencapai tingkat presisi tertinggi, di mana semua prediksi yang dibuat oleh model adalah benar tanpa adanya kesalahan positif.

4) *Recall-Confidence Curve*

Recall-Confidence Curve adalah representasi grafis yang menunjukkan hubungan antara tingkat recall (seberapa banyak dari semua kelas positif yang berhasil diprediksi oleh model) dan tingkat kepercayaan (*confidence level*) dari model klasifikasi. Kurva ini membantu dalam memahami bagaimana tingkat recall model berubah seiring dengan perubahan tingkat kepercayaan. Dengan menganalisis *Recall-Confidence Curve*, pengguna dapat mengevaluasi seberapa baik model dapat mengidentifikasi kelas positif pada berbagai tingkat kepercayaan, dan memilih ambang (*threshold*) yang sesuai untuk mencapai tingkat recall yang diinginkan berdasarkan kebutuhan spesifik dari aplikasi atau kasus penggunaan tertentu.

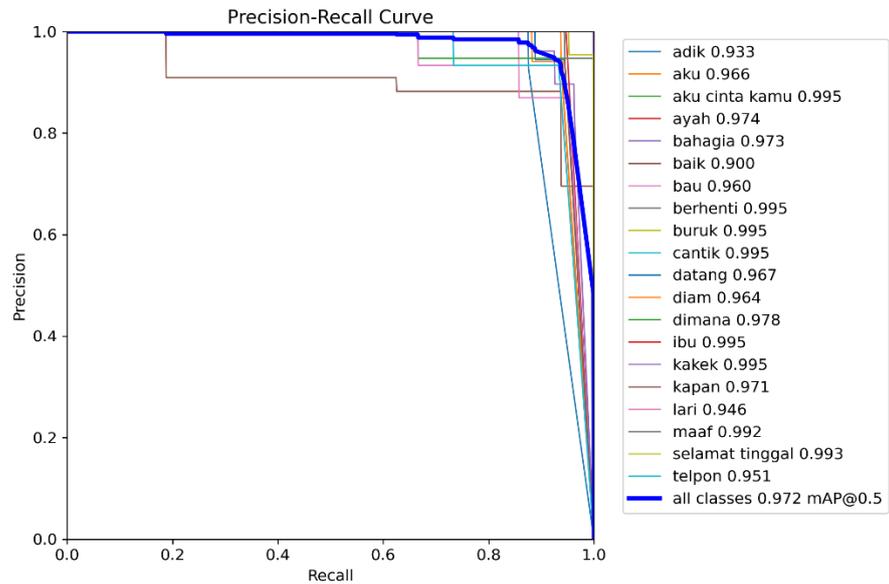
Grafik yang ditampilkan menggambarkan nilai *recall* terhadap nilai *confidence*. Dari hasil pelatihan yang diperoleh, didapati bahwa nilai recall mencapai rata-rata maksimal sebesar 0,97 pada nilai *confidence* 0,00. Hal ini menunjukkan bahwa pada tingkat keyakinan yang sangat rendah (0,00), model berhasil mengingat kembali sebagian besar *instance* positif yang sebenarnya. Hasil dari *Recall-Confidence Curve* ini tersedia dalam Gambar 4.23:



Gambar 4. 23 Recall-Confidence Curve

5) Precision-Recall Curve

Precision-Recall Curve adalah sebuah grafik yang mengilustrasikan hubungan antara *presisi* (tingkat proporsi prediksi positif yang benar) dan *recall* (tingkat proporsi kelas positif yang berhasil diprediksi) pada berbagai *threshold* pengambilan keputusan. Kurva ini memberikan gambaran tentang *trade-off* antara *presisi* dan *recall* yang dihasilkan oleh model klasifikasi. Dengan menganalisis *Precision-Recall Curve*, pengguna dapat memilih *threshold* yang sesuai untuk mencapai keseimbangan yang optimal antara presisi dan recall, sesuai dengan prioritas dan kebutuhan spesifik dari aplikasi atau kasus penggunaan tertentu. Hasil dari *Precision-Recall Curve* ini tersedia dalam Gambar 4.24:



Gambar 4. 24 Precision-Recall Curve

Grafik yang ditampilkan menggambarkan nilai *Precision-Recall* terhadap nilai *confidence*. Dari hasil pelatihan yang diperoleh, nilai *Precision-Recall* mencapai rata-rata maksimal sebesar 0,972 pada nilai $mAP@0,5$. Semakin tinggi skor mAP , semakin baik performa yang dihasilkan oleh model, menunjukkan bahwa model memiliki keseimbangan yang baik antara presisi dan recall pada tingkat kepercayaan yang ditunjukkan.

Skor mAP , atau "*mean Average Precision*", merupakan metrik evaluasi umum dalam deteksi objek di bidang *computer vision*. Metrik ini mengukur kemampuan suatu model dalam mengidentifikasi dan mengklasifikasikan objek dalam gambar dengan akurasi tinggi. Proses perhitungan mAP melibatkan perhitungan nilai *precision* dan *recall* untuk setiap kelas objek yang diidentifikasi, diikuti dengan perhitungan *average precision* (AP) untuk setiap kelas, dan akhirnya mengambil rata-rata dari semua nilai AP untuk mendapatkan skor mAP keseluruhan. Skor mAP memberikan gambaran tentang seberapa

baik model dapat menemukan objek yang relevan dalam gambar dan seberapa akurat model dalam mengklasifikasikan objek tersebut.

G. Test Model

Model yang sudah jadi kemudian di dicoba untuk mendeteksi menggunakan gambar dari luar *dataset* dari beberapa *class* untuk mengujinya. Menggunakan tingkat *confidence score* 0.25 model hanya akan menampilkan *output* dengan skor ≥ 0.25 Berikut adalah beberapa dari hasil-hasilnya:

1) Bahasa isyarat “Adik”



Gambar 4. 25 Bahasa isyarat “Adik”

2) Bahasa isyarat “Ayah”



Gambar 4. 26 Bahasa isyarat “Ayah”

3) Bahasa isyarat “Baik”



Gambar 4. 27 Bahasa isyarat “Baik”

4) Bahasa isyarat “Berhenti”



Gambar 4. 28 Bahasa isyarat “Berhenti”

5) Bahasa isyarat “Buruk”



Gambar 4. 29 Bahasa isyarat “Buruk”

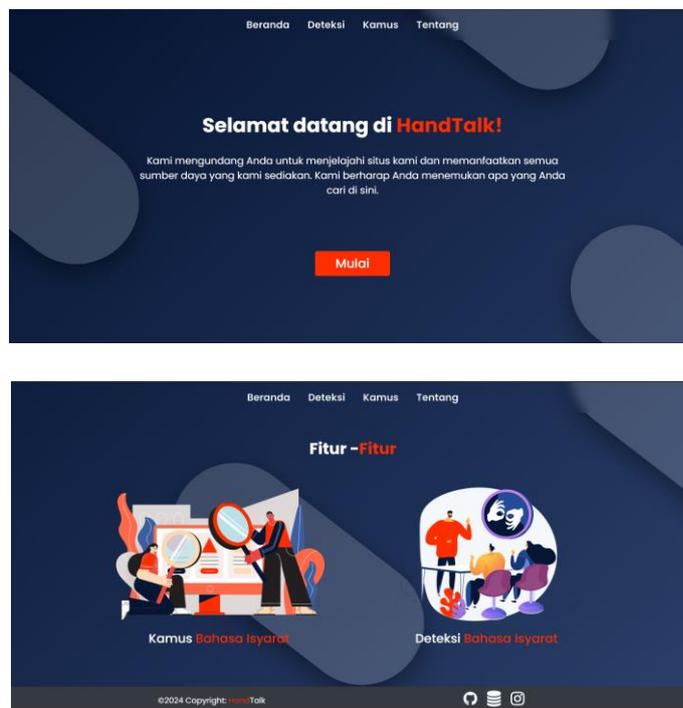
4. Implementasi Sistem

a) *Deployment*

Setelah selesai mengembangkan model, langkah berikutnya adalah melakukan deployment model. Proses deployment melibatkan integrasi panggilan API dari model deteksi objek yang telah ada di Roboflow ke dalam aplikasi atau situs web yang telah disiapkan dengan menggunakan framework Flask. Selanjutnya, penulis melakukan uji coba aplikasi untuk memverifikasi kinerja model dan melakukan optimalisasi jika diperlukan. Terakhir, setelah berhasil dikembangkan, aplikasi tersebut siap untuk digunakan dalam penelitian lebih lanjut oleh penulis.

b) Hasil Aplikasi

1) Halaman Beranda



Gambar 4. 30 Halaman Beranda

Beranda adalah halaman pertama yang muncul saat membuka aplikasi. Pada halaman ini, terdapat informasi mengenai aplikasi yang berupa *tagline*. Ini menunjukkan bahwa aplikasi telah berhasil diakses. Pada halaman beranda terdapat tombol mulai untuk bisa langsung mengarahkan ke halaman deteksi dan ada juga tombol untuk masuk ke halaman kamus. Halaman ini dapat dilihat pada Gambar 4.30.

2) Halaman Deteksi

Halaman tersebut merupakan fitur utama dalam aplikasi deteksi bahasa isyarat, yang dikenal sebagai fitur *real-time detection* atau deteksi secara langsung. Pengguna dapat mengaksesnya melalui menu navigasi yang tersedia dalam aplikasi. Cara kerjanya adalah pengguna memberikan gerakan bahasa isyarat ke webcam, memungkinkan aplikasi untuk langsung mendeteksi gerakan yang disampaikan oleh pengguna. Halaman tersebut bisa dilihat pada Gambar 4.31:



Gambar 4. 31 Halaman Deteksi

3) Halaman Kamus

Halaman kamus dalam aplikasi deteksi bahasa isyarat adalah fitur yang memungkinkan pengguna untuk mengakses daftar kata-kata atau frasa yang terkait dengan bahasa isyarat. Halaman ini

dirancang untuk membantu pengguna dalam memahami makna dan arti dari berbagai gerakan atau isyarat dalam bahasa isyarat. Ada 20 gerakan bahasa isyarat dalam kamus yang terdapat pada website aplikasi. Halaman kamus dapat dilihat pada gambar 4.32:



Gambar 4. 32 Halaman kamus

4) Halaman Tentang

Halaman Tentang dalam aplikasi deteksi bahasa isyarat adalah bagian yang dirancang untuk memberikan informasi tentang aplikasi itu sendiri. Ini memberikan kesempatan kepada pengguna untuk mempelajari lebih lanjut tentang tujuan, visi, misi, dan informasi lainnya yang terkait dengan aplikasi. Di halaman Tentang juga terdapat informasi developer yang sudah membangun aplikasi website ini. Halaman ini bisa dilihat pada gambar 4. 33:



Gambar 4. 33 Halaman Tentang

5. Pengujian

Pengujian dilakukan untuk memastikan kinerja aplikasi yang baik. Dalam penelitian ini jenis pengujian yang digunakan yaitu *Black Box*, *White box*, *User Acceptance Testing* (UAT) dan Pengujian Lapangan..

a) *Black Box Testing*

Dalam pengujian ini pengujian ini dilakukan oleh 3 dosen Informatika. Proses pengujian *black box* dibagi menjadi 3 tahap yaitu perencanaan pengujian, hasil pengujian, dan kesimpulan pengujian. Setelah merancang pengujian *black box* kemudian rancangan tersebut dicetak menjadi lembar kuesioner dan dibagikan kepada 3 dosen Informatika untuk melakukan pengujian.

1) Hasil Perhitungan Pengujian Black Box

Hasil pengujian *black box* dapat dilihat pada tabel 4.7.

Tabel 4. 7 Pengujian Black Box

Nama Pengujian	Test Case	Hasil yang Diharapkan	Hasil yang Didapatkan	Keterangan					
				Diterima			Ditolak		
				1	2	3	1	2	3
Header	User menekan beberapa tombol yang ada pada header	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada header	Aplikasi akan menuju ke halaman yang dituju	✓	✓	✓			

Halaman Beranda	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Beranda	Aplikasi akan menampilkan isi dari semua halaman Beranda	✓	✓	✓			
	User menekan tombol “Mulai” dan fitur “Deteksi Bahasa Isyarat”	User dapat masuk ke halaman Deteksi	Aplikasi akan menuju ke halaman Deteksi						
	User menekan tombol pada fitur “Kamus Bahasa Isyarat”	User dapat masuk ke halaman Kamus	Aplikasi akan menuju ke halaman Kamus	✓	✓	✓			
Halaman Deteksi	User menekan tombol “Mulai”	User dapat membuka Webcam dan mendeteksi gerakan bahasa isyarat	Aplikasi akan mendeteksi gerakan bahasa isyarat secara <i>real-time</i>	✓	✓	✓			

Halaman Kamus	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Kamus	Aplikasi akan menampilkan isi dari semua halaman Kamus	✓	✓	✓			
	User mencari kata di fitur “Pencarian kata”	User dapat mencari semua kata yang ada di halaman Kamus	Aplikasi akan menampilkan kata yang telah dicari	✓	✓	✓			
Halaman Tentang	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Tentang	Aplikasi akan menampilkan halaman Tentang	✓	✓	✓			
Footer	User menekan beberapa tombol yang ada pada Footer	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada footer	Aplikasi akan menuju ke halaman yang dituju	✓	✓	✓			

2) Kesimpulan Hasil Pengujian Black Box

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

- Pengujian pertama
Tercapai $\frac{9}{9} \times 100\% = 100\%$

Gagal $\frac{0}{9} \times 100\% = 0\%$
- Pengujian kedua
Tercapai $\frac{9}{9} \times 100\% = 100\%$

Gagal $\frac{0}{9} \times 100\% = 0\%$
- Pengujian ketiga
Tercapai $\frac{9}{9} \times 100\% = 100\%$

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

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

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

Berdasarkan analisis tersebut, dari 9 pengujian yang dilakukan oleh 3 responden, hasilnya menunjukkan bahwa tingkat keberhasilan pengujian *black box* mencapai 100%. Sementara kegagalan tidak terjadi sama sekali, sehingga persentasenya adalah 0%. Kesimpulannya, aplikasi berjalan sesuai dengan fungsinya yang diharapkan

b) *User Acceptance Testing (UAT)*

Dalam pengujian ini, lima responden, terdiri dari masyarakat umum dan orang tunawicara yang bisa menggunakan bahasa isyarat SIBI, dilibatkan untuk melakukan Uji Penerimaan Pengguna (*User Acceptance Testing/UAT*). Pengujian ini melibatkan pengguna yang akan menggunakan aplikasi dalam kehidupan sehari-hari mereka, memberikan umpan balik yang berharga mengenai kinerja dan kegunaannya. Melalui proses ini, pengembang dapat memastikan bahwa aplikasi tidak hanya berfungsi secara teknis tetapi juga memenuhi kebutuhan dan harapan pengguna, khususnya dalam membantu petani mendeteksi kata dasar bahasa isyarat secara efektif.

1) Hasil Perhitungan Pengujian *User Acceptance Testing (UAT)*

Berikut ini merupakan hasil kuesioner pengujian User Acceptance Testing (UAT) yang telah disebarakan kepada 5 responden. Hasil pengujian UAT dapat dilihat pada Tabel 4.9.

Tabel 4. 8 User Acceptence Testing (UAT)

Pertanyaan	Hasil Pengujian				
	Responden 1	Responden 2	Responden 3	Responden 4	Responden 5
1	4	5	4	5	5
2	5	5	5	5	5
3	5	4	4	4	3
4	5	3	4	4	4
5	5	5	5	5	4
6	4	4	5	4	5
7	5	5	5	5	5
8	5	4	4	4	4

9	5	5	5	5	5
10	5	5	5	5	5
Jumlah skor	48	45	46	46	45
Persentase	96%	90%	92%	92%	90%
Total	460%				

2) Kesimpulan Hasil Pengujian *User Acceptance Testing* (UAT)

Dari hasil evaluasi persentase untuk setiap pertanyaan yang mencakup aspek kegunaan, kemudahan penggunaan, dan *User Interface* (UI), yang telah diujikan oleh 5 responden. Nilai-nilai tersebut dijumlah dan kemudian dicari nilai rata-ratanya. Nilai rata-rata ini dapat dihitung dengan menggunakan persamaan sebagai berikut:

$$\text{Persentase rata-rata} = \frac{\text{jumlah total persentase}}{\text{jumlah responden}}$$

$$\text{Persentase rata-rata} = \frac{460\%}{5} = 92\%$$

Dengan daftar kategori sebagai berikut:

0% - 20% = Sangat Kurang

21% - 40% = Kurang

41% - 60% = Cukup Baik

61% - 80% = Baik

81% - 100% = Sangat Baik

Dari perhitungan tersebut, diperoleh *presentase* rata-rata dari 5 responden sebesar 92%. Dengan demikian, dapat disimpulkan bahwa pengujian UAT pada aplikasi ini memperoleh kategori yang sangat baik.

c) *White Box Testing*

White box testing adalah teknik pengujian perangkat lunak yang melibatkan pemeriksaan kode sumber dari program untuk mendeteksi adanya kesalahan. Tujuan dari pengujian white box adalah memastikan setiap bagian dari kode berfungsi sesuai dengan spesifikasi yang telah ditetapkan. Hasil pengujian white box dapat dilihat pada Tabel 4.9:

Tabel 4. 9 White Box Testing

No	Source Code	Pengertian
1.	<pre>@app.route('/deteksi') def deteksi(): return render_template('deteksi.html')</pre>	Menampilkan Halaman Deteksi
2.	<pre>var btnStart = document.getElementById("btn-start"); let nyala = true; let stopExecution = true; function startVidio() { var model; var cameraMode = "user"; // or "environment" const startVideoStreamPromise = navigator.mediaDevices.getUserMedia({ audio: false, video: { facingMode: { exact: cameraMode,</pre>	Open Camera

	<pre> }, }, }); </pre>	
3.	<pre> var publishable_key = "rf_sn5V2fW3tmavrUyZ374LgNqkDr53"; var toLoad = { model: "project-akhir-9tevr", version: 1, }; const loadModelPromise = new Promise(function (resolve, reject) { roboflow .auth({ publishable_key: publishable_key, }) .load(toLoad) .then(function (m) { model = m; resolve(); }); }); </pre>	Memuat model dari Roboflow
4.	<pre> else { startVideoStreamPromise.then(function (stream) { return new Promise(function (resolve) { video.srcObject = stream; video.onloadeddata = function () { video.pause(); resolve(); }; }); }); } </pre>	Matikan Camera

	<pre> }; }); }); </pre>	
5.	<pre> const renderPredictions = function (predictions) { var dimensions = videoDimensions(video); var scale = 1; ctx.clearRect(0, 0, ctx.canvas.width, ctx.canvas.height); predictions.forEach(function (prediction) { const x = prediction.bbox.x; const y = prediction.bbox.y; const width = prediction.bbox.width; const height = prediction.bbox.height; // Draw the bounding box. ctx.strokeStyle = prediction.color; ctx.lineWidth = 4; ctx.strokeRect((x - width / 2) / scale, (y - height / 2) / scale, width / scale, height / scale </pre>	Mendeteksi dengan bounding box

	<pre>); // Draw the label background. ctx.fillStyle = prediction.color; const textWidth = ctx.measureText(prediction.class).width; const textHeight = parseInt(font, 10); // base 10 ctx.fillRect((x - width / 2) / scale, (y - height / 2) / scale, textWidth + 8, textHeight + 4); }); </pre>	
6.	<pre> predictions.forEach(function (prediction) { const x = prediction.bbox.x; const y = prediction.bbox.y; const width = prediction.bbox.width; const height = prediction.bbox.height; // Draw the text last to ensure it's on top. ctx.font = font; ctx.textBaseline = "top"; ctx.fillStyle = "#000000"; ctx.fillText(prediction.class, </pre>	Mendeteksi secara realtime

	<pre> (x - width / 2) / scale + 4, (y - height / 2) / scale + 1); }); }; var prevTime; var pastFrameTimes = []; const detectFrame = function () { if (!model) return requestAnimationFrame(detectFrame); model .detect(video) .then(function (predictions) { requestAnimationFrame(detectFra me); renderPredictions(predictions); if (prevTime) { pastFrameTimes.push(Date.now () - prevTime); if (pastFrameTimes.length > 30) pastFrameTimes.shift(); var total = 0; _.each(pastFrameTimes, function (t) { total += t / 1000; }); </pre>	
--	--	--

	<pre> var fps = pastFrameTimes.length / total; \$("#fps").text(Math.round(fps)); } prevTime = Date.now(); }) .catch(function (e) { console.log("CAUGHT", e); requestAnimationFrame(detectFra me); }); }; } else { startVideoStreamPromise.then(function (stream) { return new Promise(function (resolve) { video.srcObject = stream; video.onloadeddata = function () { video.pause(); resolve(); }; }); }); }); </pre>	
--	--	--

1. Basis Path untuk fungsi Deteksi

Basis path (jalur dasar) untuk pengujian white box dari fungsi prediksi yang diimplementasikan di main.js. Basis path testing adalah teknik yang digunakan untuk memastikan bahwa semua jalur eksekusi yang mungkin dalam kode telah diuji

setidaknya sekali. Berikut adalah langkah langkah dalam basis path testing untuk fungsi deteksi.

a) Mengidentifikasi jalur dasar

Mengidentifikasi semua jalur eksekusi dalam kode yang relevan dengan fungsi deteksi. Setiap node yang telah ditetapkan sebelumnya akan menjadi bagian dari jalur eksekusi.

b) Menghitung Kompleksitas Siklomatik

Kompleksitas siklomatik adalah metrik yang digunakan untuk mengukur jumlah jalur linier independen dalam program. Rumus untuk menghitung kompleksitas siklomatik ($V(G)$) adalah:

$$V(G) = E - N + 2 \quad (4.14)$$

Dimana:

- E adalah jumlah edge (garis alur dalam flow graph)
- N adalah jumlah node (titik keputusan dalam flow graph)

2. *Flow Graph* untuk Fungsi Deteksi

Berikut adalah representasi flow graph dari kode fungsi deteksi pada gambar 4.34.

Nodes: 6

Edges: 6

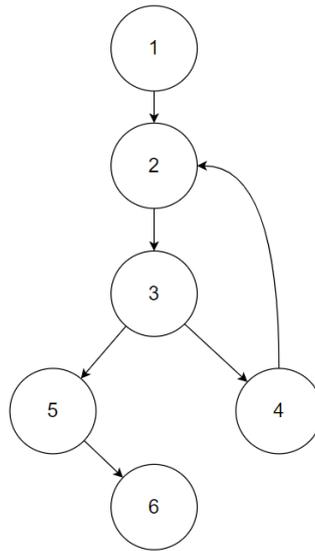
Menggunakan rumus kompleksitas siklomatik:

$$V(G) = E - N + 2 = 6 - 6 + 2 = 2$$

Ini menunjukkan bahwa ada 2 path jalur independen dalam fungsi prediksi. Jalur Independen:

Path 1: 1, 2, 3, 4, 2

Path 2: 1, 2, 3, 4, 5, 6



Gambar 4. 34 Flowgraph Basis Path

Hasil pengujian white box yang mendapatkan hasil 2 independent path dengan rentang nilai 1 sampai 10, yang berarti tidak kompleks dan mudah untuk perbaikan pada code program sehingga dikatakan baik. Pengujian white box memastikan bahwa setiap path diuji dengan baik untuk menjaga kualitas dan stabilitas aplikasi.

d) Pengujian Lapangan

Pengujian lapangan aplikasi deteksi kata dasar bahasa isyarat ini dilakukan melalui orang tunawicara yang sudah paham dan mengerti dalam gerakan bahasa isyarat. Pengujian ini bertujuan untuk mengevaluasi kemampuan aplikasi dalam mendeteksi kata dasar bahasa isyarat secara langsung dengan orang tunawicara.

Selama pengujian deteksi bahasa isyarat dilakukan, user telah memverifikasi bahwa hasil gerakan kata bahasa isyarat pada aplikasi sudah sesuai dan akurat. Dan pengujian sudah memverifikasi pada setiap

kata bahasa isyarat yang ditampilkan pada halaman kamus sudah sesuai dengan kata dasar bahasa isyarat SIBI.



Gambar 4. 35 Pengujian Aplikasi dengan Penyandang tunawicara

Hasil dari pengujian lapangan ini menunjukkan bahwa aplikasi deteksi kata dasar bahasa isyarat memiliki manfaat bagi penyandang tunawicara dan tunarungu. Dengan kemampuannya untuk melakukan deteksi kata dasar bahasa isyarat secara *real-time*, aplikasi ini dapat membantu bagi orang yang ingin belajar bahasa isyarat dan dapat berkomunikasi dengan orang tunawicara.



Gambar 4. 36 Pengujian Aplikasi dengan Penyandang tunawicara

B. Pembahasan

1. Pemodelan Data

Pemodelan data mencakup identifikasi jenis data yang diperlukan, seperti citra gerakan-gerakan kata dasar bahasa isyarat. Pengumpulan data yang akurat sangat penting untuk melatih model deteksi bahasa isyarat dengan tepat. Pada pemodelan data diawali dengan mengambil dataset publik yang diambil dari platform Universe Roboflow, yang dimana dataset akan dilatih (training) dan di generate ulang menggunakan fitur yang ada di Roboflow dan Google Colaboratory agar akurasi menjadi lebih baik dan akurat. Tahap ini sangat penting dikarenakan untuk pemodelan pada aplikasi deteksi kata dasar bahasa isyarat.

Penulis mengambil dataset dari Roboflow dengan nama file “Projek Akhir Satu” yang sebelumnya sudah di modeling yang berjumlah 51 kelas dan berjumlah 11.319 *image*, tapi penulis hanya mengambil 20 kelas 2.162 *image* untuk dilatih ulang. Pada dataset sebelumnya pembuat dataset menggunakan pembagian data menjadi *training data* 82%, *valid data* 9% dan *test data* 9%. Akan tetapi penulis melakukan proses ini melibatkan pembagian data menjadi 75% dari total sampel untuk *training data*, 15% untuk *valid data*, dan 15% untuk *test data*.

2. Pengembangan Aplikasi

Pengembangan aplikasi ini menggunakan model Waterfall yang dimana terdiri dari beberapa tahapan. Adapun pembahasan tiap tahapan yang telah dilakukan dari model tersebut yaitu sebagai berikut:

a. Analisis Kebutuhan

Pengembangan aplikasi melibatkan identifikasi infrastruktur teknis yang diperlukan untuk mendukung aplikasi. Ini mencakup perangkat keras untuk pengambilan gambar dan pemrosesan data, serta perangkat lunak yang digunakan untuk pengembangan aplikasi. Tahap analisis kebutuhan ini sangat penting karena menjadi dasar bagi desain

dan pengembangan aplikasi selanjutnya. Dengan memahami dan merinci kebutuhan secara detail, pengembang dapat memastikan bahwa aplikasi yang dibuat sesuai dengan kebutuhan pengguna dan mampu memberikan solusi yang efektif untuk deteksi kata dasar bahasa isyarat.

b. Desain

Setelah mengumpulkan kebutuhan aplikasi, langkah berikutnya adalah tahap desain. Dalam tahap ini, digunakan model perancangan Unified Modeling Language (UML), yang mencakup empat jenis diagram UML: Use Case, Activity, Sequence, dan Class Diagram. Selain itu, juga digunakan desain antarmuka atau User Interface (UI), dengan merancang Wireframe sederhana untuk mempermudah pengembangan aplikasi.

Use Case Diagram digunakan untuk mengidentifikasi dan menggambarkan berbagai fungsi yang akan dilakukan oleh aplikasi, serta hubungan antara pengguna (aktor) dengan fungsi-fungsi tersebut. Ini membantu dalam memahami interaksi pengguna dengan sistem dan memastikan semua kebutuhan fungsional telah tercakup.

Activity Diagram menggambarkan alur kerja atau aktivitas yang terjadi dalam aplikasi, menunjukkan langkah-langkah yang harus diambil dari awal hingga akhir dalam berbagai proses, seperti proses deteksi bahasa isyarat. Dengan Activity Diagram, setiap proses dijelaskan secara rinci dan efisien.

Setelah mengumpulkan kebutuhan aplikasi, langkah berikutnya adalah tahap desain. Dalam tahap ini, digunakan model perancangan Unified Modeling Language (UML), yang mencakup empat jenis diagram UML: Use Case, Activity, Sequence, dan Class Diagram. Selain itu, juga digunakan desain antarmuka atau User Interface (UI), dengan merancang Wireframe sederhana untuk mempermudah pengembangan aplikasi.

Use Case Diagram digunakan untuk mengidentifikasi dan menggambarkan berbagai fungsi yang akan dilakukan oleh aplikasi, serta hubungan antara pengguna (aktor) dengan fungsi-fungsi tersebut. Ini membantu dalam memahami interaksi pengguna dengan sistem dan memastikan semua kebutuhan fungsional telah tercakup.

Activity Diagram menggambarkan alur kerja atau aktivitas yang terjadi dalam aplikasi, menunjukkan langkah-langkah yang harus diambil dari awal hingga akhir dalam berbagai proses, seperti proses deteksi penyakit. Dengan Activity Diagram, setiap proses dijelaskan secara rinci dan efisien.

Sequence Diagram menggambarkan interaksi antara objek dalam urutan waktu tertentu, membantu memahami urutan kejadian dalam sistem. Dengan Sequence Diagram, interaksi antar komponen sistem dapat diatur dengan lancar dan sesuai dengan yang diharapkan.

Class Diagram menggambarkan struktur sistem dalam hal kelas-kelas dan hubungan antar kelas tersebut, membantu menentukan atribut dan metode yang dimiliki oleh setiap kelas. Dengan Class Diagram, struktur data dan logika aplikasi dapat terorganisir dengan baik sesuai dengan kebutuhan fungsional yang telah diidentifikasi.

Selain model UML, desain antarmuka atau User Interface (UI) juga menjadi fokus utama dalam tahap ini. Wireframe sederhana berfungsi sebagai blueprint visual untuk tata letak dasar elemen-elemen antarmuka pengguna, memastikan antarmuka pengguna intuitif dan mudah digunakan.

Dengan desain yang baik, aplikasi dapat dikembangkan dengan lebih efisien dan efektif, sesuai dengan kebutuhan pengguna yang telah diidentifikasi.

c. Implementasi

Implementasi yang pertama adalah implementasi model. Model dikembangkan dari mulai pengolahan dataset sampai dengan training model. Setelah model selesai dikembangkan, selanjutnya adalah implementasi kedua yaitu implementasi sistem. Pada implementasi sistem, model yang sudah selesai di training kemudian diambil API yang ada pada platform Roboflow untuk dimasukkan kedalam sistem pada file main.js langkah berikutnya adalah melakukan pengujian untuk memastikan bahwa model berfungsi dengan baik di lingkungan website. Pengujian ini meliputi apakah deteksi gerakan bahasa isyarat bisa menampilkan hasil secara real-time. Setelah pengujian selesai dan hasilnya memuaskan, aplikasi website yang mengandung model ini siap untuk didistribusikan atau dipublikasikan untuk digunakan oleh pengguna.

d. Pengujian

Langkah terakhir dalam proses pengembangan adalah pengujian untuk memverifikasi kualitas dari aplikasi yang dikembangkan. Penulis melakukan 3 jenis pengujian: *Black Box Testing*, *White Box Testing* *User Acceptance Testing* (UAT), dan Pengujian Lapangan. Dalam *Black Box Testing*, hasilnya adalah 100% keberhasilan, sedangkan pengujian yang gagal mendapat 0% dari 3 responden dan 9 pengujian dilengkapi dengan masukan dari responden yang tercantum di Lampiran. Pada pengujian *White Box* mendapatkan hasil 2 independent path dengan rentang nilai 1 sampai 10, yang berarti tidak kompleks dan mudah untuk perbaikan pada code program sehingga dikatakan baik. Pengujian white box memastikan bahwa setiap path diuji dengan baik untuk menjaga kualitas dan stabilitas aplikasi. Selanjutnya, pengujian *User Acceptance Testing* (UAT) menghasilkan tingkat keberhasilan yang sangat baik, dengan presentase sebesar 92% dari 5 responden yang menjawab 10 pertanyaan. Yang terakhir adalah pengujian lapangan

menghasilkan hasil bahwa aplikasi dapat mendeteksi bahasa isyarat dengan baik dan akurat dan dapat bermanfaat bagi penyandang tunarungu dan tunawicara bahkan bisa bermanfaat bagi yang ingin belajar tentang bahasa isyarat.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

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

1. Dari proses pelatihan *dataset* bahasa isyarat menggunakan YOLO v8 dengan 30 *epoch* dan jumlah *dataset* sebanyak 2.162, diperoleh akurasi sebesar 98%. *Dataset* tersebut dibagi menjadi 1.639 data (75%) untuk pelatihan, 354 data (15%) untuk validasi dan 345 data (15%) untuk pengujian. Hasil ini menunjukkan bahwa model yang dikembangkan memiliki kemampuan yang cukup baik dalam mendeteksi gerakan bahasa isyarat. Secara keseluruhan, penelitian ini menunjukkan potensi yang sangat baik dari penggunaan YOLO v8 dalam aplikasi deteksi bahasa isyarat berbasis *website*.
2. Pengujian black box dilakukan oleh 3 responden dengan total 9 pengujian. Beberapa halaman yang diuji meliputi *header*, halaman beranda, halaman deteksi, halaman kamus, halaman tentang, dan *footer*. Hasil dari pengujian ini menunjukkan tingkat keberhasilan sebesar 100%, dengan tingkat kegagalan 0%.
3. Pengujian *user acceptance testing* dilakukan oleh 5 responden menggunakan 10 pertanyaan. Pertanyaan mencakup aspek kegunaan, kemudahan penggunaan, dan *User Interface* (UI). Hasil pengujian ini menunjukkan persentase sebesar 92% dengan kategori sangat baik.
4. Pengujian lapangan aplikasi deteksi bahasa isyarat pada orang tunawicara menunjukkan bahwa aplikasi yang dikembangkan oleh penulis mampu menghasilkan deteksi bahasa isyarat dengan tepat dan akurat yang dimana telah diuji oleh penguji yang mengetahui gerakan kata dasar bahasa isyarat.
5. Penelitian ini memberikan banyak manfaat bagi penyandang tunawicara

khususnya bagi orang yang ingin mempelajari gerakan kata dasar bahasa isyarat.

B. Saran

Berdasarkan penelitian yang telah dilakukan, penulis memberikan beberapasarana sebagai pertimbangan untuk penelitian selanjutnya, yaitu:

1. Meng-*hosting* aplikasi deteksi bahasa isyarat untuk mempermudah pengguna dalam menggunakan aplikasi untuk mengenali kata dasar dalam bahasa isyarat.
2. Menyelidiki peluang untuk mengembangkan aplikasi deteksi kata dasar bahasa isyarat untuk platform lain seperti Android, agar dapat dijangkau oleh lebih banyak pengguna.
3. Menambahkan fitur deteksi berdasarkan *upload image*.
4. Memperluas daftar kata dasar dalam bahasa isyarat untuk menarik lebih banyak pengguna yang ingin mempelajari gerakan-gerakan bahasa isyarat lainnya.

DAFTAR PUSTAKA

- [1] R. A. Mursita, “Respon tunarungu terhadap penggunaan sistem bahasa isyarat indonesa (sibi) dan bahasa isyarat indonesia (bisindo) dalam komunikasi,” *Inklusi*, vol. 2, no. 2, pp. 221–232, 2015.
- [2] A. Ma’ruf and M. Hardjianto, “Penerapan Algoritme You Only Look Once Version 8 Untuk Identifikasi Abjad Bahasa Isyarat Indonesia,” in *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, 2023, pp. 567–576.
- [3] I. B. A. Peling, I. M. P. A. Ariawan, and G. B. Subiksa, “Deteksi Bahasa Isyarat Menggunakan Tensorflow Lite dan American Sign Language (ASL),” *Jurnal Krisnadana*, vol. 3, no. 2, pp. 90–100, 2024.
- [4] M. Sholawati, K. Auliasari, and F. X. Ariwibisono, “Pengembangan Aplikasi Pengenalan Bahasa Isyarat Abjad Sibi Menggunakan Metode Convolutional Neural Network (CNN),” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 6, no. 1, pp. 134–144, 2022.
- [5] M. Y. Trinurais, R. K. A. Wibowo, B. R. Prakosa, and M. B. Subkhi, “Deteksi Bahasa Isyarat Berdasarkan SIBI (Sistem Bahasa Isyarat) menggunakan Transfer Learning,” in *STAINS (SEMINAR NASIONAL TEKNOLOGI & SAINS)*, 2024, pp. 361–369.
- [6] W. Saputro and D. B. Sumantri, “Klasifikasi citra dalam gerak tangan bahasa isyarat sibi menggunakan algoritma k-nn,” *INTECOMS: Journal of Information Technology and Computer Science*, vol. 5, no. 2, pp. 180–188, 2022.
- [7] G. Jocher, “Ultralytics YOLOv8 Docs,” Ultralytics. Accessed: Feb. 25, 2024. [Online]. Available: <https://docs.ultralytics.com/>

- [8] T. Qurohman, M. Y. Romdoni, A. Fatoni, R. Fatullah, and A. Shavira, “Rancang Bangun Aplikasi Distribusi Dan Penanganan Keluhan Pada Universitas Banten Jaya,” *Jurnal Ilmiah Sains dan Teknologi*, vol. 8, no. 1, pp. 51–62, 2024.
- [9] A. S. Nugraheni, A. P. Husain, and H. Unayah, “Optimalisasi penggunaan bahasa isyarat dengan sibi dan bisindo pada mahasiswa difabel tunarungu di prodi pgmi uin sunan kalijaga,” *Holistika: Jurnal Ilmiah PGSD*, vol. 5, no. 1, pp. 28–33, 2023.
- [10] S. Apendi, C. Setianingsih, and M. W. Paryasto, “Deteksi Bahasa Isyarat Sistem Isyarat Bahasa Indonesia Menggunakan Metode Single Shot Multibox Detector,” *eProceedings of Engineering*, vol. 10, no. 1, 2023.
- [11] B. Mahesh, “Machine learning algorithms-a review,” *International Journal of Science and Research (IJSR).[Internet]*, vol. 9, no. 1, pp. 381–386, 2020.
- [12] E. H. A. Prastyo, I. G. L. P. E. Prisma, and R. Wiratsongko, “Implementasi Web Scraping Pada Situs Berita Menggunakan Metode Supervised learning,” *Inovate: Jurnal Ilmiah Inovasi Teknologi Informasi*, vol. 5, no. 1, pp. 58–66, 2020.
- [13] D. Yolanda, M. H. Hersyah, and E. Marozi, “Implementasi Metode Unsupervised Learning Pada Sistem Keamanan Dengan Optimalisasi Penyimpanan Kamera IP,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1099–1105, 2021.
- [14] A. El Gourari, M. Raoufi, M. Skouri, and F. Ouatik, “The implementation of deep reinforcement learning in e-learning and distance learning: Remote practical work,” *Mobile Information Systems*, vol. 2021, pp. 1–11, 2021.
- [15] A. Raup, W. Ridwan, Y. Khoeriyah, S. Supiana, and Q. Y. Zaqiah, “Deep Learning dan Penerapannya dalam Pembelajaran,” *JIIP-Jurnal Ilmiah Ilmu Pendidikan*, vol. 5, no. 9, pp. 3258–3267, 2022.

- [16] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," *Jurnal Nasional Ilmu Komputer*, vol. 2, no. 3, pp. 213–232, 2021.
- [17] I. Munadhif, D. H. Fathoni, and M. A. Jamiin, "Pengendalian CCTV Menggunakan You Only Look Once (YOLO)," in *Prosiding Seminar Nasional Terapan Riset Inovatif (SENTRINOV)*, 2020, pp. 958–965.
- [18] Y. Yanto, F. Aziz, and I. Irmawati, "YOLO-V8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 3, pp. 1437–1444, 2023.
- [19] N. J. Hayati, D. Singasatia, and M. R. Muttaqin, "Object Tracking Menggunakan Algoritma You Only Look Once (YOLO) v8 untuk Menghitung Kendaraan," *Komputa: Jurnal Ilmiah Komputer dan Informatika*, vol. 12, no. 2, pp. 91–99, 2023.
- [20] I. Irmayanti, "Perancangan Sistem Informasi Penyewaan Thermoking Pada PT. Moderen Prima Transportasi Menggunakan Python Dengan Framework Flask," *Jurnal Sistem dan Teknologi Informasi Cendekia (JuSTICe)*, vol. 1, no. 1, pp. 24–34, 2023.
- [21] M. S. Negara and A. Z. Mardiansyah, "Implementasi Machine Learning dengan Metode Collaborative Filtering dan Content-Based Filtering pada Aplikasi Mobile Travel (Bangkit Academy)," *Jurnal Begawe Teknologi Informasi (JBegaTI)*, vol. 5, no. 1, pp. 126–136, 2024.
- [22] H. Larasati and S. Masripah, "Analisa dan perancangan sistem informasi pembelian grc dengan metode waterfall," *Jurnal Pilar Nusa Mandiri*, vol. 13, no. 2, pp. 193–198, 2017.
- [23] R. N. Oktavianti, "Sistem Informasi E-Learning Berbasis Web Pada SMA Negeri 3 Cikarang Utara," *REMIK: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 8, no. 1, pp. 410–424, 2024.

- [24] G. A. H. Gigih, M. R. Maulani, and D. Hamidin, "Sistem Informasi Desa Berbasis Web Untuk Memajukan Produk UMKM dan Layanan Masyarakat di Desa Bapangsari Kabupaten Purworejo," *Jurnal Teknik Informatika*, vol. 15, no. 2, pp. 71–77, 2023.
- [25] R. P. Fajar, "Teknik Boundary Value Analysis pada Blackbox Testing untuk Aplikasi Buku Catatan Harian," *Jurnal Repositor*, vol. 6, no. 1, 2024.
- [26] D. Destiarini, A. Rahman, and K. Sumartayasa, "Analisa Kualitas Website BPJS Kesehatan Dengan Metode WebQual 4.0 Dan User Acceptance Testing Di Wilayah Kabupaten Ogan Komering Ulu.," *Jurnal Media Infotama*, vol. 19, no. 2, pp. 237–243, 2023.
- [27] M. E. Khan and F. Khan, "A comparative study of white box, black box and grey box testing techniques," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 6, 2012.
- [28] S. Riawan and R. W. Amelia, "Pengaruh Kompetensi Dan Efikasi Diri Terhadap Kinerja Karyawan Di Departemen Quality Control Pada Pt Dellifood Sentosa Corpindo Kota Tangerang," *Journal of Research and Publication Innovation*, vol. 2, no. 1, pp. 342–351, 2024.
- [29] R. F. Wijaya and R. B. Utomo, "Metode Waterfall Dalam Rancang Bangun Sistem Informasi Manajemen Kegiatan Masjid Berbasis Web," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 3, no. 5, pp. 563–571, 2023.

LAMPIRAN

Lampiran 1 Lembar Bimbingan Dosen Pembimbing



UNIVERSITAS PGRI SEMARANG

FAKULTAS TEKNIK DAN INFORMATIKA

Kampus : Jalan Sidodadi Timur Nomor 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 : Heru Setiawan Saputra
NPM : 20670087
Program Studi : Informatika
Judul Skripsi : Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI (Sisitem Isyarat Bahasa Indonesia) Menggunakan YOLOv8 Berbasis Website
Dosen Pembimbing I : Noora Qotrun Nada, ST., M.Eng.
Dosen Pembimbing II : Febrian Murti Dewanto, SE., M.Kom.

No.	Hari Tanggal	Uraian Bimbingan	Paraf
1.	kamis/15-02-2024	Konsultasi Judul skripsi	
2.	senin/19-02-2024	Bab 1	
3.	selasa/7-05-2024	Bab 2 dan revisian bab 1	
4.	selasa/28-05-2024	Bab 3 dan revisian bab 2	
5.	selasa/11-06-2024	Bab 4 dan revisian bab 3	
6.	selasa/12-07-2024	Uk cm ok, pastikan tak	
7.	Rabu/13-07-2024	ada plagiarisme. Acc. sidang	
8.			

Dosen Pembimbing I,

Noora Qotrun Nada, ST., M.Eng.

NIDN. 0626028201

Mahasiswa,

Heru Setiawan Saputra

NPM. 20670087



UNIVERSITAS PGRI SEMARANG

FAKULTAS TEKNIK DAN INFORMATIKA

Kampus : Jalan Sidodadi Timur Nomor 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 : Heru Setiawan Saputra
NPM : 20670087
Program Studi : Informatika
Judul Skripsi : Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) Menggunakan YOLOv8 Berbasis Website.
Dosen Pembimbing I : Noora Qotrun Nada, ST., M.Eng.
Dosen Pembimbing II : Febrian Murti Dewanto, SE., M.Kom.

No.	Hari Tanggal	Uraian Bimbingan	Paraf
1	Senin/2-2-2024	Konsultasi judul skripsi	
2	Selasa/20-2-2024	Bab 1	
3	Kamis/14-3-2024	Revisi Bab 1 dan Bab 2	
4	Rabu/20-3-2024	Konsultasi Bab 3 dan revisi Bab 2	
5	Selasa/8-6-2024	Bab 3 dan konsultasi Bab 4	
6	Selasa/18-6-2024	Bab 4 dan bimbingan proyek	
7	Rabu/3-7-2024	Revisi Bab 4	
8	reb. 13/7	ACE Skripsi	

Dosen Pembimbing II,

Febrian Murti Dewanto, SE., M.Kom..
NIDN. 0606027801

Mahasiswa,

Heru Setiawan Saputra
NPM. 20670087

Lampiran 2 Kuesioner Pengujian Black Box

Kuesioner *Black Box Testing* pada Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis Website

Nama Penguji : Bambang Agus Heriambana, S.kom, M.kom

Tanggal Pengujian : 25 Juni 2024

Nama pengujian	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Keterangan	
				Diterima	Ditolak
Header	User menekan beberapa tombol yang ada pada header	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada header	Aplikasi akan menuju ke halaman yang dituju	✓	
Halaman Beranda	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Beranda	Aplikasi akan menampilkan isi dari semua halaman Beranda	✓	
	User menekan tombol	User dapat masuk ke	Aplikasi akan menuju ke	✓	

	"Mulai" dan fitur "Deteksi Bahasa Isyarat"	halaman Deteksi	halaman Deteksi		
	User menekan tombol pada fitur "Kamus Bahasa Isyarat"	User dapat masuk ke halaman Kamus	Aplikasi akan menuju ke halaman Kamus	✓	
Halaman Deteksi	User menekan tombol "Mulai"	User dapat membuka Webcam dan mendeteksi gerakan bahasa isyarat	Aplikasi akan mendeteksi gerakan bahasa isyarat secara <i>real-time</i>	✓	
Halaman Kamus	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Kamus	Aplikasi akan menampilkan isi dari semua halaman Kamus	✓	
	User mencari kata di fitur	User dapat mencari semua kata yang ada di	Aplikasi akan menampilkan kata yang telah dicari	✓	

	"Pencarian kata"	halaman Kamus			
Halaman Tentang	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Tentang	Aplikasi akan menampilkan halaman Tentang	✓	
Footer	User menekan beberapa tombol yang ada pada Footer	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada footer	Aplikasi akan menuju ke halaman yang dituju	✓	

Saran dari penguji :

.....

.....

.....

.....

Semarang,



.....
NIDN. 0601088201

Kuesioner *Black Box Testing* pada Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis *Website*

Nama Penguji : Ramadhan Renaldy, s.kom, m.kom
Tanggal Pengujian : 25 Juni 2024

Nama pengujian	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Keterangan	
				Diterima	Ditolak
Header	User menekan beberapa tombol yang ada pada header	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada header	Aplikasi akan menuju ke halaman yang dituju	✓	
Halaman Beranda	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Beranda	Aplikasi akan menampilkan isi dari semua halaman Beranda	✓	
	User menekan tombol	User dapat masuk ke	Aplikasi akan menuju ke	✓	

	“Mulai” dan fitur “Deteksi Bahasa Isyarat”	halaman Deteksi	halaman Deteksi		
	User menekan tombol pada fitur “Kamus Bahasa Isyarat”	User dapat masuk ke halaman Kamus	Aplikasi akan menuju ke halaman Kamus	✓	
Halaman Deteksi	User menekan tombol “Mulai”	User dapat membuka Webcam dan mendeteksi gerakan bahasa isyarat	Aplikasi akan mendeteksi gerakan bahasa isyarat secara <i>real-time</i>	✓	
Halaman Kamus	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Kamus	Aplikasi akan menampilkan isi dari semua halaman Kamus	✓	
	User mencari kata di fitur	User dapat mencari semua kata yang ada di	Aplikasi akan menampilkan kata yang telah dicari	✓	

	"Pencarian kata"	halaman Kamus			
Halaman Tentang	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Tentang	Aplikasi akan menampilkan halaman Tentang	✓	
Footer	User menekan beberapa tombol yang ada pada Footer	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada footer	Aplikasi akan menuju ke halaman yang dituju	✓	

Saran dari penguji :

A Aplikasi terlalu lama saat dijalankan sehingga user harus menunggu lama juga.

Di halaman tentang beri penjelasan tentang Yolo

Semarang, 25 Juni 2024

@Mky

Ramadhan Renaldy

NIDN

Npp. 249901659

Kuesioner *Black Box Testing* pada Aplikasi Deteksi Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis *Website*

Nama Penguji : Nugroho Dwi Saputro, S.kom, M.kom

Tanggal Pengujian : 25 Juni 2024

Nama pengujian	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Keterangan	
				Diterima	Ditolak
Header	User menekan beberapa tombol yang ada pada header	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada header	Aplikasi akan menuju ke halaman yang dituju	✓	
Halaman Beranda	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Beranda	Aplikasi akan menampilkan isi dari semua halaman Beranda	✓	
	User menekan tombol	User dapat masuk ke	Aplikasi akan menuju ke	✓	

	“Mulai” dan fitur “Deteksi Bahasa Isyarat”	halaman Deteksi	halaman Deteksi		
	User menekan tombol pada fitur “Kamus Bahasa Isyarat”	User dapat masuk ke halaman Kamus	Aplikasi akan menuju ke halaman Kamus	✓	
Halaman Deteksi	User menekan tombol “Mulai”	User dapat membuka Webcam dan mendeteksi gerakan bahasa isyarat	Aplikasi akan mendeteksi gerakan bahasa isyarat secara <i>real-time</i>	✓	
Halaman Kamus	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Kamus	Aplikasi akan menampilkan isi dari semua halaman Kamus	✓	
	User mencari kata di fitur	User dapat mencari semua kata yang ada di	Aplikasi akan menampilkan kata yang telah dicari	✓	

	"Pencarian kata"	halaman Kamus			
Halaman Tentang	User melakukan gulir atau scroll ke bawah	User dapat melihat halaman Tentang	Aplikasi akan menampilkan halaman Tentang	✓	
Footer	User menekan beberapa tombol yang ada pada Footer	User dapat masuk ke halaman yang dituju setelah menekan tombol yang ada pada footer	Aplikasi akan menuju ke halaman yang dituju	✓	

Saran dari penguji :

.....

.....

.....

.....

Semarang,

Nugraha Dwi S.

NIDN. 0623058802.

Lampiran 3 Pengujian UAT

**Kuesioner Pengujian User Acceptance Testing (UAT) pada Aplikasi Deteksi
Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia)
menggunakan YOLOv8 Berbasis Web**

Nama Penguji : Ilham Lutfi Aditya
Tanggal Pengujian : 23 Juni 2024

No.	Pertanyaan	Skor				
		Tidak setuju	Kurang setuju	Cukup setuju	Setuju	Sangat setuju
Aspek kegunaan						
1.	Apakah aplikasi berguna bagi orang yang ingin belajar bahasa isyarat, khususnya penyandang tunarungu dan tunawicara?				✓	
2.	Apakah aplikasi deteksi bahasa isyarat memberikan informasi tentang gerakan bahasa isyarat?					✓
Aspek kemudahan pengguna (user)						
3.	Apakah aplikasi deteksi bahasa isyarat mudah di operasikan?					✓

4.	Apakah sistem berjalan dengan cepat dan responsif?					✓
5.	Seberapa responsif deteksi bahasa isyarat terhadap gerakan pengguna? Apakah deteksi berjalan dengan lancar dan tanpa jeda yang mencolok?					✓
6.	Seberapa cepat deteksi bahasa isyarat merespons gerakan pengguna? Apakah ada penundaan yang mengganggu antara gerakan yang dilakukan oleh pengguna dan respons dari aplikasi?				✓	
Aspek tampilan (user interface)						
7.	Apakah aplikasi deteksi memiliki tampilan yang mudah dipahami?					✓
8.	Apakah aplikasi deteksi bahasa isyarat memiliki tampilan yang menarik?					✓

**Kuesioner Pengujian User Acceptance Testing (UAT) pada Aplikasi Deteksi
Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia)
menggunakan YOLOv8 Berbasis Web**

Nama Penguji : M. Khotibul Kurniawan

Tanggal Pengujian : 25 Juni 2024

No.	Pertanyaan	Skor				
		Tidak setuju	Kurang setuju	Cukup setuju	Setuju	Sangat setuju
Aspek kegunaan						
1.	Apakah aplikasi berguna bagi orang yang ingin belajar bahasa isyarat, khususnya penyandang tunarungu dan tunawicara?					✓
2.	Apakah aplikasi deteksi bahasa isyarat memberikan informasi tentang gerakan bahasa isyarat?					✓
Aspek kemudahan pengguna (user)						
3.	Apakah aplikasi deteksi bahasa isyarat mudah di operasikan?				✓	

4.	Apakah sistem berjalan dengan cepat dan responsif?			✓		
5.	Seberapa responsif deteksi bahasa isyarat terhadap gerakan pengguna? Apakah deteksi berjalan dengan lancar dan tanpa jeda yang mencolok?					✓
6.	Seberapa cepat deteksi bahasa isyarat merespons gerakan pengguna? Apakah ada penundaan yang mengganggu antara gerakan yang dilakukan oleh pengguna dan respons dari aplikasi?				✓	
Aspek tampilan (user interface)						
7.	Apakah aplikasi deteksi memiliki tampilan yang mudah dipahami?					✓
8.	Apakah aplikasi deteksi bahasa isyarat memiliki tampilan yang menarik?				✓	

9.	Apakah aplikasi memiliki tampilan background dan tema yang enak dipandang?					✓
10.	Apakah aplikasi deteksi bahasa isyarat perlu dikembangkan lagi?					✓

Keterangan :

- 1 = Tidak setuju
- 2 = Kurang setuju
- 3 = Cukup setuju
- 4 = Setuju
- 5 = Sangat setuju

Semarang, 25 Juni 2024

M. Khoirul K.

**Kuesloner Pengujian User Acceptance Testing (UAT) pada Aplikasi Deteksi
Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia)
menggunakan YOLOv8 Berbasis Web**

Nama Penguji : Fatkhul Maulana Ramadhany

Tanggal Pengujian : 25 Juni 2024

No.	Pertanyaan	Skor				
		Tidak setuju	Kurang setuju	Cukup setuju	Setuju	Sangat setuju
Aspek kegunaan						
1.	Apakah aplikasi berguna bagi orang yang ingin belajar bahasa isyarat, khususnya penyandang tunarungu dan tunawicara?					✓
2.	Apakah aplikasi deteksi bahasa isyarat memberikan informasi tentang gerakan bahasa isyarat?					✓
Aspek kemudahan pengguna (user)						
3.	Apakah aplikasi deteksi bahasa isyarat mudah di operasikan?				✓	

4.	Apakah sistem berjalan dengan cepat dan responsif?				✓	
5.	Seberapa responsif deteksi bahasa isyarat terhadap gerakan pengguna? Apakah deteksi berjalan dengan lancar dan tanpa jeda yang mencolok?					✓
6.	Seberapa cepat deteksi bahasa isyarat merespons gerakan pengguna? Apakah ada penundaan yang mengganggu antara gerakan yang dilakukan oleh pengguna dan respons dari aplikasi?				✓	
Aspek tampilan (user interface)						
7.	Apakah aplikasi deteksi memiliki tampilan yang mudah dipahami?					✓
8.	Apakah aplikasi deteksi bahasa isyarat memiliki tampilan yang menarik?				✓	

9.	Apakah aplikasi memiliki tampilan background dan tema yang enak dipandang?					✓
10.	Apakah aplikasi deteksi bahasa isyarat perlu dikembangkan lagi?					✓

Keterangan :

1 = Tidak setuju

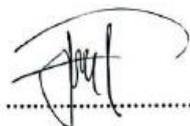
2 = Kurang setuju

3 = Cukup setuju

4 = Setuju

5 = Sangat setuju

Semarang, 25 Juni 2024



fatmud

**Kuesioner Pengujian User Acceptance Testing (UAT) pada Aplikasi Deteksi
Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia)
menggunakan YOLOv8 Berbasis Web**

Nama Penguji : Ari Wibowo

Tanggal Pengujian : 25 Juni 2024

No.	Pertanyaan	Skor				
		Tidak setuju	Kurang setuju	Cukup setuju	Setuju	Sangat setuju
Aspek kegunaan						
1.	Apakah aplikasi berguna bagi orang yang ingin belajar bahasa isyarat, khususnya penyandang tunarungu dan tunawicara?				✓	
2.	Apakah aplikasi deteksi bahasa isyarat memberikan informasi tentang gerakan bahasa isyarat?					✓
Aspek kemudahan pengguna (user)						
3.	Apakah aplikasi deteksi bahasa isyarat mudah di operasikan?				✓	

4.	Apakah sistem berjalan dengan cepat dan responsif?				✓	
5.	Seberapa responsif deteksi bahasa isyarat terhadap gerakan pengguna? Apakah deteksi berjalan dengan lancar dan tanpa jeda yang mencolok?					✓
6.	Seberapa cepat deteksi bahasa isyarat merespons gerakan pengguna? Apakah ada penundaan yang mengganggu antara gerakan yang dilakukan oleh pengguna dan respons dari aplikasi?					✓
Aspek tampilan (<i>user interface</i>)						
7.	Apakah aplikasi deteksi memiliki tampilan yang mudah dipahami?					✓
8.	Apakah aplikasi deteksi bahasa isyarat memiliki tampilan yang menarik?				✓	

9.	Apakah aplikasi memiliki tampilan background dan tema yang enak dipandang?					✓
10.	Apakah aplikasi deteksi bahasa isyarat perlu dikembangkan lagi?					✓

Keterangan :

1 = Tidak setuju

2 = Kurang setuju

3 = Cukup setuju

4 = Setuju

5 = Sangat setuju

Semarang, 25 Juni 2024

Arif W

**Kuesioner Pengujian User Acceptance Testing (UAT) pada Aplikasi Deteksi
Kata Dasar Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia)
menggunakan YOLOv8 Berbasis Web**

Nama Penguji : Ahmad Mubhar
 Tanggal Pengujian : 25 Juni 2024

No.	Pertanyaan	Skor				
		Tidak setuju	Kurang setuju	Cukup setuju	Setuju	Sangat setuju
Aspek kegunaan						
1.	Apakah aplikasi berguna bagi orang yang ingin belajar bahasa isyarat, khususnya penyandang tunarungu dan tunawicara?					✓
2.	Apakah aplikasi deteksi bahasa isyarat memberikan informasi tentang gerakan bahasa isyarat?					✓
Aspek kemudahan pengguna (user)						
3.	Apakah aplikasi deteksi bahasa isyarat mudah di operasikan?			✓		

4.	Apakah sistem berjalan dengan cepat dan responsif?				✓	
5.	Seberapa responsif deteksi bahasa isyarat terhadap gerakan pengguna? Apakah deteksi berjalan dengan lancar dan tanpa jeda yang mencolok?				✓	
6.	Seberapa cepat deteksi bahasa isyarat merespons gerakan pengguna? Apakah ada penundaan yang mengganggu antara gerakan yang dilakukan oleh pengguna dan respons dari aplikasi?					✓
Aspek tampilan (user interface)						
7.	Apakah aplikasi deteksi memiliki tampilan yang mudah dipahami?					✓
8.	Apakah aplikasi deteksi bahasa isyarat memiliki tampilan yang menarik?				✓	

9.	Apakah aplikasi memiliki tampilan background dan tema yang enak dipandang?					✓
10.	Apakah aplikasi deteksi bahasa isyarat perlu dikembangkan lagi?					✓

Keterangan :

- 1 = Tidak setuju
- 2 = Kurang setuju
- 3 = Cukup setuju
- 4 = Setuju
- 5 = Sangat setuju

Semarang, 25 Juni 2024

Triuly
 .Ahmad....Muzhar..

Lampiran 4 Revisi Sidang

LEMBAR REVISI UJIAN SKRIPSI

Nama Mahasiswa : Heru Setiawan Saputra
N P M : 20670087
Judul : Aplikasi Deteksi Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis Website

No	Uraian Revisi	Keterangan
①	Utr Brmbngun ditandatangani Yheru.	Revisi acc Noora 18/7/24.
②	Perbaiki typo:	

Pengesahan Penguji I


Noora Gofrun Nada, S.T., M.ENG
NIP/NPP. 158201485

*) Revisi Maksimal 7 Hari Setelah Pelaksanaan Ujian Skripsi

LEMBAR REVISI UJIAN SKRIPSI/TUGAS AKHIR

Nama Mahasiswa : Heru Setiawan Saputra
N P M : 20670087
Judul : Aplikasi Deteksi Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis Website

No	Uraian Revisi	Keterangan
	D. halaman dan yg d. lupa	→ 17/7
	YOLO ?	→ 17/7

Pengesahan Penguji II



Febrian Murti Dewanto, SE, M. Komi
NIP/NPP. 057801172

*) Revisi Maksimal 7 Hari Setelah Pelaksanaan Ujian Skripsi

LEMBAR REVISI UJIAN SKRIPSI/TUGAS AKHIR

Nama Mahasiswa : Heru Setiawan Saputra
N P M : 20670087
Judul : Aplikasi Deteksi Bahasa Isyarat SIBI (Sistem Isyarat Bahasa Indonesia) menggunakan YOLOv8 berbasis Website

No	Uraian Revisi	Keterangan
1	Use Case Diagram, Seolah? Aplikasi? Ada? ✓	19/7/2024 ✓
2	Kerangka Berpikir ✓	
3	pada pembahasan, tambahkan proses pemola data ✓	
4	celite box. ✓	

Pengesahan Penguji III

Bambang Agus H. S. Kom, M. Kom
NIP/NPP. 148201433

*) Revisi Maksimal 7 Hari Setelah Pelaksanaan Ujian Skripsi