

**APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN
BUATAN DENGAN VISUALISASI BERBASIS *WEB***

(Skripsi)

Oleh

TANIA ARGADINI

NPM 1915061015



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2023

**APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN
BUATAN DENGAN VISUALISASI BERBASIS *WEB***

Oleh

Tania Argadini

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2023

ABSTRAK

APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN BUATAN DENGAN VISUALISASI BERBASIS WEB

Oleh

Tania Argadini

Penyakit jantung adalah sebuah kondisi dimana jantung tidak dapat melaksanakan tugasnya dengan baik. Deteksi penyakit jantung diperlukan untuk pencegahan dan deteksi dini sehingga tidak semakin parah. Sehingga aplikasi segmentasi jantung berbasis web ini dibuat untuk mempermudah dalam melakukan segmentasi sebuah citra MRI (*Magnetic Resonance Imaging*). Metode pengembangan yang digunakan adalah *AI Project Cycle* untuk pengembangan model dengan tahapan *Problem Scoping, Data Acquisition, Data Exploration, Modelling, Evaluation* dan menggunakan metode *SDLC (Software Development Life Cycle)* untuk mengembangkan sistem rekayasa perangkat lunak dengan tahapan *User Interface Design, Front-End Development, Back-End Development, Deployment, dan Testing*. Pembuatan aplikasi web segmentasi jantung pada citra MRI ini telah berhasil dilakukan dan sudah mampu mendeteksi dan memberikan informasi hasil segmentasi pada citra MRI, dengan akurasi yang tinggi yaitu 0,99. Aplikasi web juga sudah dapat memenuhi fungsi utamanya, ditunjukkan melalui pengujian *System Usability Scale (SUS)* ke 12 responden. Hasil *SUS* menunjukkan bahwa Aplikasi Segmentasi Jantung Menggunakan Kecerdasan Buatan Dengan Visualisasi Berbasis Web telah terbukti sesuai dengan kebutuhan dan preferensi pengguna dimana menghasilkan skor rata-rata sebesar 71 dengan *grade scale "C"*, *adjective rating "Good"* dan masuk dalam kategori "*Acceptable*".

Kata kunci — Segmentasi, CNN, U-Net, *Deep Learning*

ABSTRACT

HEART SEGMENTATION APPLICATION USING ARTIFICIAL INTELLIGENCE WITH WEB-BASED VISUALIZATION

By

Tania Argadini

Heart disease is a condition in which the heart is unable to perform its functions properly. The detection of heart disease is necessary for prevention and early detection to prevent further complications. Therefore, this web-based heart segmentation application was developed to facilitate the segmentation of a Magnetic Resonance Imaging (MRI) image. The AI Project Cycle methodology was used for model development, including the stages of Problem Scoping, Data Acquisition, Data Exploration, Modeling, and Evaluation. Additionally, the Software Development Life Cycle (SDLC) method was employed to develop the software engineering system, involving User Interface Design, Front-End Development, Back-End Development, Deployment, and Testing phases. The development of this web-based heart segmentation application for MRI images has been successfully accomplished, capable of detecting and providing segmentation results with a high accuracy of 0.99. The web application also fulfills its primary functions, as demonstrated through System Usability Scale (SUS) testing with 12 respondents. The SUS results indicate that the Artificial Intelligence-based Heart Segmentation Application with Web-Based Visualization has proven to meet the needs and preferences of users, with an average score of 71 corresponding to a "Good" rating on the grade scale and falling into the "Acceptable" category.

Keywords : Segmentation, CNN, U-Net, Deep Learning.

Judul Skripsi : **APLIKASI SEGMENTASI JANTUNG
MENGUNAKAN KECERDASAN BUATAN
DENGAN VISUALISASI BERBASIS WEB**

Nama Mahasiswa : **Tania Argadini**

Nomor Pokok Mahasiswa : 1915061015

Jurusan : Teknik Elektro

Fakultas : Teknik

MENYETUJUI

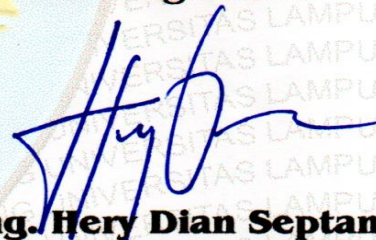
1. Komisi Pembimbing

Pembimbing Pendamping

Pembimbing Utama



Rio Ariestia Pradipta, S.Kom., M.T.I.
NIP 19860323 201903 1 013



Ing. Hery Dian Septama, S.T.
NIP 19850915 200812 1 001

2. Mengetahui

Ketua Jurusan
Teknik Elektro

Ketua Program Studi
Teknik Informatika



Herlinawati, S.T., M.T.
NIP 19710314 199903 2 001

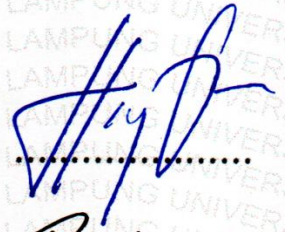


Mona Arif Muda, S.T., M.T.
NIP 19711112 200003 1 002

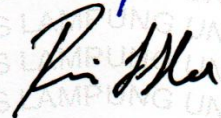
MENGESAHKAN

1. Tim Penguji

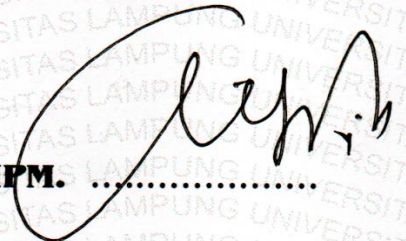
Ketua : Ing. Hery Dian Septama, S.T.



Sekretaris : Rio Ariestia Pradipta, S.Kom., M.T.I.



Penguji : Ir. Gigih Forda Nama, S.T., M.T.I., IPM.



2. Dekan Fakultas Teknik



Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.f.
NIP 19750928 200112 1 002



Tanggal Lulus Ujian Skripsi : 25 Mei 2023

SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini , menyatakan bahwa skripsi saya dengan judul “Aplikasi Segmentasi Jantung Menggunakan Kecerdasan Buatan Dengan Visualisasi Berbasis *Web*” dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 16 Juni 2023

Pembuat pernyataan,



Tania Argadini

NPM. 1915061015

RIWAYAT HIDUP



Penulis lahir di Tekad, 22 Juli 2001. Penulis merupakan anak pertama dari dua bersaudara dari pasangan Bapak Suparyono dan Ibu Irena Helga Suhastin. Penulis menyelesaikan pendidikannya di SD Negeri 1 Tekad pada tahun 2013, SMP Negeri 1 Pulaupanggung pada tahun 2016, dan SMA Negeri 1 Talangpadang pada tahun 2019. Pada tahun 2019 penulis terdaftar sebagai mahasiswa Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan, antara lain:

1. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Pengembangan Keteknikan, Divisi Penelitian dan Pengembangan pada tahun 2019.
2. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Pengembangan Keteknikan, Divisi Pengabdian Masyarakat pada tahun 2020.
3. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil AI Mastery Program di Orbit Future Academy pada tahun 2022.
4. Mengikuti program Magang di PT. Kazee Digital Indonesia sebagai Data Scientist pada tahun 2022.
5. Mengikuti program Magang Bersertifikat Kampus Merdeka dari Kementerian Pendidikan dan Budaya sebagai Creative Coder di PT. Sembilan Matahari pada tahun 2022.
6. Melaksanakan Kuliah Kerja Nyata di Desa Talang Sepuh, Kecamatan Talang Padang, Kabupaten Tanggamus, Provinsi Lampung pada bulan Juni sampai dengan Agustus 2022.

MOTTO

“why do cute people like me have to ngerjain skripsi????”

(Penulis)

“Semua manusia di muka bumi ini bingung. Nanti ga bingung kalau udah di surga.”

(Aldi Taher)

“Allah tidak membebani seseorang itu melainkan sesuai dengan kesanggupannya.”

(Q.S Al-Baqarah: 286)

“Did that child suffer in pain for such a long time just to become me?”

(My Sea - IU)

“No matter what I do, tick-tock, tick-tock

Life is still going on, it just flows.”

(Life Is Still Going On – NCT DREAM)

*Bismillaahirrohmaanirrahiim,
Dengan mengharapkan ridho dari Allah SWT,
Alhamdulillah, Atas Izin Allah yang Maha Kuasa
Kupersembahkan karyaku ini untuk*

Diri Sendiri

TANIA ARGADINI

Ayah dan Ibu Tercinta

SUPARYONO

dan

IRENA HELGA SUHASTIN

Adik Tersayang

FAIRUZ ABIMANYU

SANWACANA

Segala puji bagi Allah, atas limpahan nikmat-Nya yang diberikan kepada penulis sehingga dapat menyelesaikan Tugas Akhir ini. Shalawat dan salam senantiasa dicurahkan kepada Nabi Muhammad, suri teladan yang mampu membuka sesuatu yang terkunci, penutup dari semua yang terdahulu, penolong kebenaran dengan jalan yang benar, dan petunjuk kepada jalan-Mu yang lurus.

Tugas Akhir dengan judul “APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN BUATAN DENGAN VISUALISASI BERBASIS *WEB*” ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung.

Pada kesempatan ini, penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T. selaku Kepala Jurusan Teknik Elektro Universitas Lampung.
3. Bapak Mona Arif Muda, S.T.,M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung dan telah membantu proses kelancaran pengerjaan penelitian.
4. Bapak Ing. Hery Dian Septama, S.T. selaku dosen pembimbing utama yang telah memberikan nasihat, arahan, dan bimbingan rutin, motivasi, dan pandangan kehidupan kepada penulis di setiap kesempatan dengan baik dan ramah dalam mempersiapkan diri menjadi seorang Sarjana Teknik.
5. Bapak Rio Ariestia Pradipta, S.Kom, M.T.I. selaku dosen pembimbing pendamping yang telah memberikan bimbingan, arahan, dan nilai-nilai kehidupan kepada penulis dengan baik dan ramah.

6. Bapak Ir. Gigih Forda Nama, S.T.,M.T.I., IPM selaku dosen penguji yang telah memberikan kritik dan saran yang membangun kepada penulis dalam mengerjakan skripsi ini.
7. Segenap Dosen di Jurusan Teknik Elektro yang telah memberikan ilmu yang bermanfaat, wawasan, dan pengalaman bagi penulis.
8. Mbak Rika dan Segenap Staff di Jurusan Teknik Elektro dan Fakultas Teknik yang telah sangat membantu penulis baik dalam hal administrasi dan hal-hal lainnya.
9. Ibu yang sangat saya cintai, Ibu Irena Helga Suhastin, yang sangat penulis banggakan dan sayangi yang telah mendukung penuh sehingga penulis dapat menyelesaikan laporan tugas akhir ini.
10. Keluarga besar yang sangat saya cintai pula Papa, Bimbim, Mbah, Uwo, Ayah, Ibu, Bang Rio dan Bang Reza yang juga selalu mendukung dan mendoakan penulis selama ini.
11. Sahabat – sahabat grup laut nabila, presil, royyan, irfan, ais, rachel, naufal, iqbal, sanjaya, rama, agung dan surya yang sangat penulis sayangi. Terima kasih sudah menjadi sahabat terbaik selama awal kuliah sampai saat ini.
12. Angkatan ETERNITY 2019 yang telah menjadi teman seperjuangan sejak mahasiswa baru. Terimakasih telah mewarnai masa perkuliahan penulis dan menulis banyak cerita bersama.
13. Seluruh teman-teman yang terlibat langsung maupun tidak langsung dalam penyelesaian skripsi.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan skripsi ini. Penulis mengharapkan kritik dan saran konstruktif dari semua pihak demi kemajuan bersama. Penulis berharap skripsi ini dapat bermanfaat bagi kita semua.

Bandar Lampung, 16 Juni 2023

Tania Argadini

DAFTAR ISI

	Halaman
DAFTAR ISI	iv
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan Laporan	3
II. TINJAUAN PUSTAKA.....	5
2.1 <i>Orbit Future Academy</i>	5
2.2 <i>Jantung</i>	5
2.3 <i>Magnetic Resonance Imaging (MRI)</i>	7
2.4 <i>Web</i>	7
2.5 <i>User Interface (UI)</i>	8
2.6 <i>User Experience</i>	8
2.7 <i>Anaconda</i>	8
2.8 <i>Figma</i>	9
2.9 <i>Python</i>	9
2.10 <i>Artificial Intelligence</i>	10
2.11 <i>Machine Learning</i>	11
2.12 <i>Deep Learning</i>	12
2.13 <i>Convolutional Neural Network (CNN)</i>	13
2.14 <i>U-Net</i>	14

1.14.1	<i>Convolution Layer</i>	16
1.14.2	<i>Maxpooling Layer</i>	16
2.14.3	<i>Upsampling Layer</i>	16
2.14.4	<i>Concatenate</i>	17
2.15	Citra	17
2.16	Citra Digital.....	17
2.17	Jenis Citra.....	18
2.18	Segmentasi Citra	20
2.19	<i>TensorFlow</i>	21
2.20	<i>AI Project Cycle</i>	21
2.21	<i>SDLC (Software Development Life Cycle)</i>	23
2.22	<i>Google Cloud Platform</i>	23
2.23	<i>SQLite</i>	25
2.24	<i>Blackbox Testing</i>	25
2.25	<i>System Usability Scale</i>	26
2.26	Penelitian Terkait	28
III.	METODELOGI PENELITIAN	32
3.1	Waktu dan Tempat Penelitian	32
3.2	<i>Project Scope</i>	33
3.3	Alat Penelitian.....	33
3.4	Tahapan Penelitian	34
3.3.1	<i>Problem Scoping</i>	35
3.3.2	<i>Data Acquisition</i>	36
3.3.3	<i>Data Exploration</i>	37
3.3.4	<i>Modelling</i>	37
3.3.5	<i>Evaluation</i>	37
3.3.6	<i>User Interface Design</i>	37
3.3.7	<i>Front-End Development</i>	40
3.3.8	<i>Back-End Development</i>	40
3.3.9	<i>Deployment</i>	40
3.3.10	<i>Testing</i>	40
IV.	HASIL DAN PEMBAHASAN.....	42
4.1	<i>Problem Scoping</i>	42
4.2	<i>Data Acquisition</i>	42
4.3	Data Exploration	43
4.4	<i>Modelling</i>	44
4.5	<i>User Interface Design</i>	56
4.5.1	<i>Halaman Login</i>	57

4.5.2	<i>Halaman Home</i>	57
4.5.3	<i>Halaman Result</i>	58
4.5.4	<i>Halaman Sign In</i>	58
4.6	Front-End Development	59
4.6.1	<i>Halaman Sign Up</i>	59
4.6.2	<i>Halaman Sign In</i>	62
4.6.3	<i>Halaman Home</i>	64
4.6.4	<i>Halaman About</i>	65
4.6.5	<i>Halaman Drop File</i>	67
4.6.6	<i>Halaman Result</i>	69
4.7	Back-End Development.....	71
4.7.1	<i>Sign Up</i>	74
4.7.2	<i>Sign In</i>	76
4.7.3	<i>Upload</i>	77
4.7.4	<i>Predict</i>	79
4.8	Deployment.....	81
4.8.1	<i>Create Instance Compute Engine</i>	81
1.8.1.1	<i>Menginstall Anaconda</i>	82
1.8.1.2	<i>Running Code</i>	83
4.9	Testing	85
4.9.1	<i>Blackbox Testing</i>	85
4.9.2	<i>Kesimpulan Blackbox Testing</i>	90
4.9.3	<i>Image Testing</i>	90
4.9.4	<i>System Usability Scale</i>	92
4.9.5	<i>Kesimpulan System Usability Scale</i>	94
V.	KESIMPULAN.....	96
5.1	Kesimpulan	96
5.2	Saran	97
	DAFTAR PUSTAKA	98

DAFTAR TABEL

	Halaman
Tabel 2.1 Warna dan Nilai Penyusun Warna	19
Tabel 2.2 Penelitian Terkait	30
Tabel 3.1 Jadwal Penelitian.....	32
Tabel 3.2 Alat Penelitian.....	33
Tabel 3.3 Definisi <i>Use Case</i>	38
Tabel 4.1 Hasil Pengujian Blackbox Testing Halaman <i>Sign Up</i>	85
Tabel 4.2 Hasil Pengujian Blackbox Testing Halaman <i>Sign In</i>	87
Tabel 4.3 Hasil Pengujian Blackbox Testing Halaman <i>Logout</i>	89
Tabel 4.4 Hasil Pengujian Blackbox Testing Halaman <i>Predict</i>	90
Tabel 4.5 Data Asli Kuisisioner	92
Tabel 4.6 Perhitungan Skor SUS	93

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Bagian Pada Jantung.....	6
Gambar 2.2 Hasil MRI.....	7
Gambar 2.3 Skema <i>Artificial Intelligence</i>	10
Gambar 2.4 Skema Pembagian <i>Machine Learning</i>	11
Gambar 2.5 Ilustrasi <i>Deep Learning</i>	12
Gambar 2.6 Proses <i>Convolutional Neural Network</i>	13
Gambar 2.7 Arsitektur U-NET.....	15
Gambar 2.8 Koordinat Citra Digital	18
Gambar 2.9 Citra Berwarna.....	19
Gambar 2.10 Citra Keabuan.....	19
Gambar 2.11 Citra Biner	20
Gambar 2.12 Hasil Segmentasi Pada Citra MRI	21
Gambar 2.13 AI Project Cycle	21
Gambar 2.14 Pertanyaan SUS	26
Gambar 2.15 Chart Nilai Pertanyaan SUS	27
Gambar 2.16 Rumus Perhitungan SUS.....	28
Gambar 2.17 Grafik Kesimpulan SUS.....	28
Gambar 3.1 Diagram Alir Penelitian	34
Gambar 3.2 <i>Actiivty Diagram</i> Aplikasi.....	38
Gambar 3.3 <i>Use Case Diagram</i>	39
Gambar 4.1 File Dataset.....	43
Gambar 4.2 <i>Raw Image</i>	43
Gambar 4.3 <i>Source Code Import Library</i>	44

Gambar 4.4 <i>Source Code Download Dataset</i>	45
Gambar 4.5 <i>Source Code Generate Main Tensor</i>	45
Gambar 4.6 <i>Source Code Split, Validation and Test Main Tensor</i>	46
Gambar 4.7 <i>Source Code Visualization Main Tensor</i>	47
Gambar 4.8 Hasil <i>Visualization Main Tensor</i>	47
Gambar 4.9 <i>Source Code Visualization Train Tensor</i>	48
Gambar 4.10 Hasil <i>Visualization Train Tensor</i>	49
Gambar 4.11 <i>Source Code Evaluate Model Training</i>	49
Gambar 4.12 <i>Source Code Evaluate Validation Set</i>	50
Gambar 4.13 Hasil <i>Evaluate Validation Set</i>	51
Gambar 4.14 <i>Source Code Comparison Image</i>	52
Gambar 4.15 Hasil <i>Comparison Image</i>	52
Gambar 4.16 <i>Source Code Overlaped Color Plot</i>	53
Gambar 4.17 Hasil <i>Overlaped Color Plot</i>	54
Gambar 4.18 <i>Source Code Visualization Prediction Test</i>	54
Gambar 4.19 Hasil <i>Visualization Prediction Test</i>	55
Gambar 4.20 <i>Source Code Segmented Image</i>	55
Gambar 4.21 Hasil Segmented Citra	56
Gambar 4.22 Halaman <i>Login</i>	57
Gambar 4.23 Halaman <i>Home</i>	57
Gambar 4.24 Halaman <i>Result</i>	58
Gambar 4.25 Halaman <i>Sign In</i>	58
Gambar 4.26 Halaman <i>Sign Up</i>	60
Gambar 4.27 <i>Source Code Halaman Sign Up</i>	61
Gambar 4.28 Halaman <i>Sign In</i>	62
Gambar 4.29 <i>Source Code Halaman Sign In</i>	63
Gambar 4.30 Halaman <i>Home</i>	64
Gambar 4.31 <i>Source Code Halaman Home</i>	65
Gambar 4.32 Halaman <i>About</i>	66
Gambar 4.33 <i>Source Code Halaman About</i>	66
Gambar 4.34 Halaman <i>Drop File</i>	68
Gambar 4.35 <i>Source Code Drop File</i>	68

Gambar 4.36 Halaman <i>Result</i>	70
Gambar 4.37 <i>Source Code Result</i>	71
Gambar 4.38 <i>Source Code Model</i>	72
Gambar 4.39 <i>Source Code View</i>	73
Gambar 4.40 Halaman <i>Sign Up</i>	74
Gambar 4.41 <i>Contoller Sign Up</i>	75
Gambar 4.42 Halaman <i>Sign In</i>	76
Gambar 4.43 <i>Contoller Sign In</i>	76
Gambar 4.44 Halaman <i>Upload</i>	77
Gambar 4.45 <i>Contoller Upload</i>	78
Gambar 4.46 Halaman <i>Predict</i>	79
Gambar 4.47 <i>Contoller Predict</i>	80
Gambar 4.48 <i>Instance Compute Engine</i>	82
Gambar 4.49 Menginstal <i>Anaconda</i>	82
Gambar 4.50 <i>Running Code</i>	83
Gambar 4.51 Konfigurasi <i>PM2</i>	84
Gambar 4.52 Informasi Penggunaan Ruang <i>Disk</i>	84
Gambar 4.53 Rincian Penggunaan Ruang <i>Disk</i>	85
Gambar 4.54 Hasil Pengujian Halaman <i>Sign Up</i>	87
Gambar 4.55 Hasil Pengujian Halaman <i>Sign In</i>	89
Gambar 4.56 Citra <i>MRI</i>	91
Gambar 4.57 Hasil Pengujian Citra <i>MRI</i>	92
Gambar 4.58 <i>Acceptability, Grade Scale, Dan Adjective Rating</i>	94

I. PENDAHULUAN

1.1 Latar Belakang

Jantung adalah organ vital yang berfungsi sebagai pemompa darah untuk memenuhi kebutuhan oksigen dan nutrisi ke seluruh tubuh. Penyakit jantung adalah sebuah kondisi dimana jantung tidak dapat melaksanakan tugasnya dengan baik. Data WHO menyatakan bahwa sebanyak 7,3 juta penduduk dunia meninggal dikarenakan penyakit jantung. Meskipun penyakit jantung merupakan penyakit yang tidak menular, penyakit ini merupakan jenis penyakit yang mematikan nomor satu di dunia. Namun, kesadaran masyarakat akan pentingnya kesehatan jantung masih sangat rendah.[1]

Untuk itu diperlukan pencegahan dan deteksi sejak dini agar kondisi tersebut tidak semakin parah. Selama ini, dokter dan ahli radiologi masih melakukan deteksi penyakit pada jantung menggunakan MRI atau *X-ray*. Namun, pada data citra tersebut masih terdapat bagian-bagian lain seperti ginjal, empedu, dan hati. Bagian-bagian lain ini akan mengganggu proses deteksi penyakit pada jantung. Sehingga dibutuhkanlah suatu sistem yang bisa mendapatkan bagian jantung saja secara otomatis.

Mengatasi permasalahan tersebut diperlukan suatu teknologi yang dapat menampilkan dengan jelas informasi sesuai kebutuhan, salah satu caranya adalah dengan pengolahan citra digital berupa segmentasi gambar. Segmentasi merupakan proses partisi atau mengelompokkan gambar digital ke beberapa daerah. Segmentasi memegang peranan yang sangat penting dalam pengolahan citra digital karena segmentasi bertujuan untuk menyederhanakan ataupun merubah representasi gambar menjadi sesuatu yang lebih bermakna dan mudah

dianalisa. Segmentasi dilakukan berdasarkan pengelompokan nilai piksel yang sama dalam suatu bagian citra.

Salah satu metode segmentasi yang berkembang pesat saat ini adalah *Deep Learning*. *Deep Learning* merupakan salah satu sub bidang dari *Machine Learning*. *Deep Learning* dapat mendeteksi suatu pola secara otomatis dari data yang diberikan. Metode segmentasi berbasis *Deep Learning* yang sering digunakan untuk pengolahan data gambar adalah *Convolutional Neural Network* (CNN). Salah satu arsitektur dari CNN yang banyak digunakan untuk melakukan segmentasi ialah arsitektur U-Net.[2]

Arsitektur U-Net merupakan arsitektur *Convolutional Neural Network* yang *popular* pada analisis citra medis karena keakuratannya dalam melakukan segmentasi dan dapat meningkatkan akurasi terhadap diagnosis penyakit yang diperiksa. Arsitektur U-Net memiliki dua jalur yaitu jalur *encoder* dan jalur *decoder*. Proses jalur *encoder* digunakan untuk pengurangan ukuran matriks input sedangkan jalur *decoder* digunakan untuk mengembalikan ukuran matriks ke ukuran semula dengan meminimalkan jumlah *feature maps* sehingga gambar bisa tersegmentasi[3].

Untuk itu penelitian ini akan membantu dokter dan ahli radiologi dalam melakukan segmentasi pada citra MRI melalui aplikasi segmentasi jantung menggunakan kecerdasan buatan dengan visualisasi berbasis *web*.

1.2 Rumusan Masalah

Berdasarkan latar belakang, rumusan masalah untuk penelitian ini adalah bagaimana membuat sebuah aplikasi berbasis *web* untuk melihat hasil segmentasi jantung pada citra MRI (*Magnetic Resonance Imaging*).

1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah hasil prediksi segmentasi jantung pada MRI dataset yang didapat dari salah satu *user* github dengan *username* @erikycd[4].

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah membuat aplikasi segmentasi jantung menggunakan kecerdasan buatan dengan visualisasi berbasis *web* dengan fitur prediksi gambar.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat melakukan segmentasi dengan hasil evaluasi kinerja yang akurat serta dapat membantu dokter dan ahli radiologi dalam melakukan segmentasi pada citra MRI melalui aplikasi segmentasi jantung berbasis *web*. Aplikasi segmentasi jantung ini merupakan suatu sistem yang bisa mendapatkan bagian jantung saja secara otomatis.

1.6 Sistematika Penulisan Laporan

Sistematika penulisan yang digunakan pada skripsi ini adalah sebagai berikut:

BAB I : PENDAHULUAN

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

BAB II : TINJAUAN PUSTAKA

Pada bab ini membahas mengenai dasar teori yang digunakan sebagai sumber dalam memahami permasalahan dalam melakukan penelitian mengenai APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN BUATAN DENGAN VISUALISASI BERBASIS *WEB*.

BAB III : METODOLOGI PENELITIAN

Pada bab ini membahas mengenai metodologi penelitian yang digunakan dalam APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN BUATAN DENGAN VISUALISASI BERBASIS *WEB*.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini membahas mengenai hasil serta pembahasan yang diperoleh dalam penelitian APLIKASI SEGMENTASI JANTUNG MENGGUNAKAN KECERDASAN BUATAN DENGAN VISUALISASI BERBASIS *WEB*.

BAB V : KESIMPULAN

Pada bab ini berisi tentang kesimpulan dari hasil penelitian yang telah dilakukan serta saran-saran sebagai masukan untuk penelitian lebih lanjut di masa mendatang.

DAFTAR PUSTAKA**LAMPIRAN**

II. TINJAUAN PUSTAKA

2.1 *Orbit Future Academy*

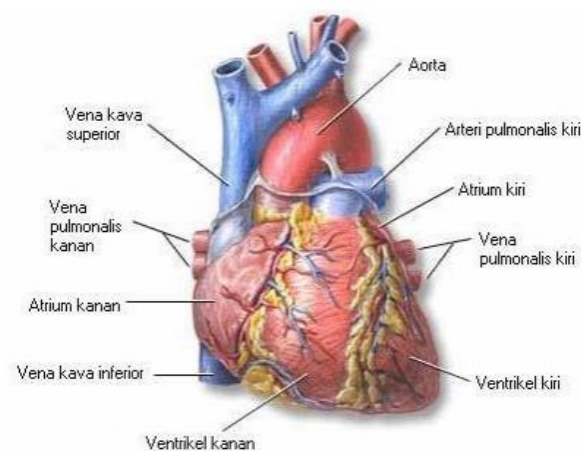
MBKM (Magang Bersertifikat Kampus merdeka) adalah program persiapan karir yang merupakan bagian dari kebijakan Merdeka Belajar oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia yang memberikan kesempatan bagi mahasiswa untuk mengasah kemampuan sesuai bakat dan minat dengan terjun langsung ke dunia kerja sebagai persiapan karir masa depan.

Orbit Future Academy (OFA) merupakan salah satu mitra terpilih yang didirikan pada tahun 2016 dengan tujuan untuk meningkatkan kualitas hidup melalui inovasi, edukasi, dan pelatihan keterampilan. *Label* atau *brand* Orbit merupakan kelanjutan dari warisan mendiang Prof. Dr. Ing. B. J. Habibie (presiden Republik Indonesia ke-3) dan istrinya, Dr. Hasri Ainun Habibie. Mereka berdua telah menjadi penggerak dalam mendukung perkembangan inovasi dan teknologi pendidikan di Indonesia. OFA mengkurasi dan melokalkan program/kursus internasional untuk *upskilling* atau *reskilling* pemuda dan tenaga kerja menuju pekerjaan masa depan. Hal ini sesuai dengan slogan OFA, yakni “*Skills-for-Future-Jobs*”.

2.2 **Jantung**

Jantung merupakan organ vital dalam tubuh manusia, di mana kelainan bahkan yang kecil sekalipun dapat berdampak besar pada kinerja tubuh kita. Jantung adalah organ berbentuk otot berongga yang terletak di tengah-tengah dada. Pada jantung, terdapat bagian kanan dan kiri yang masing-masing memiliki ruang atas yang disebut atrium untuk mengumpulkan darah, dan ruang bawah yang disebut

ventrikel untuk memompa darah. Untuk memastikan aliran darah hanya berlangsung dalam satu arah, setiap ventrikel dilengkapi dengan satu katup masuk dan satu katup keluar[5]. Fungsi utama jantung adalah menyediakan oksigen ke seluruh tubuh dan membersihkan tubuh dari hasil metabolisme (karbondioksida). Jantung melaksanakan fungsi tersebut dengan mengumpulkan darah yang kekurangan oksigen dari seluruh tubuh dan memompanya ke dalam paru-paru, dimana darah akan mengambil oksigen dan membuang karbondioksida. Jantung kemudian mengumpulkan darah yang kaya oksigen dari paru-paru dan memompanya ke jaringan di seluruh tubuh.

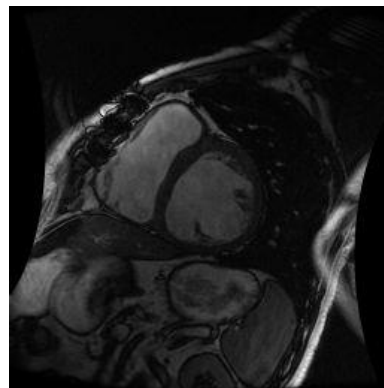


Gambar 2.1 Bagian Pada Jantung

Jantung memiliki empat ruang yang terdiri dari atrium kiri, atrium kanan, ventrikel kiri, dan ventrikel kanan. Komponen jantung bagian kiri terdiri dari atrium kiri dan ventrikel kiri, sementara bagian kanan terdiri dari atrium kanan dan ventrikel kanan. Atrium kiri dan kanan dipisahkan oleh septum atrium, sedangkan ventrikel kiri dan kanan dipisahkan oleh septum ventrikel. Ventrikel kiri berperan utama dalam memompa darah ke seluruh tubuh melalui arteri aorta, sementara ventrikel kanan memompa darah ke paru-paru melalui arteri pulmonalis. Karena tuntutan fungsinya yang berbeda, kedua ventrikel memiliki struktur yang berbeda pula.

2.3 *Magnetic Resonance Imaging (MRI)*

MRI ditemukan pada tahun 1969 oleh Raymond V. Damadian untuk melihat struktur tubuh manusia. MRI merupakan suatu alat kedokteran di bidang pemeriksaan diagnostik radiologi, yang menghasilkan rekaman gambar potongan penampang tubuh / organ manusia dengan menggunakan medan magnet berkekuatan antara 0,064 – 1,5 tesla (1 tesla = 1000 Gauss) dan resonansi getaran terhadap inti atom hidrogen[6]. Contoh hasil MRI dapat dilihat pada gambar 2.2 berikut.



Gambar 2.2 Hasil MRI

2.4 *Web*

World wide web, yang juga dikenal sebagai *web*, adalah layanan yang menyajikan informasi menggunakan konsep hyperlink atau tautan, yang memudahkan pengguna komputer untuk melakukan penelusuran atau *browsing* informasi melalui internet. Keunggulan ini menjadikan *web* sebagai layanan yang paling cepat berkembang. Melalui *web*, kita dapat memberikan sorotan atau garis bawahan pada kata-kata atau gambar dalam sebuah dokumen untuk menghubungkannya dengan media lain seperti dokumen, frase, klip film, atau *file* suara. *Web* memungkinkan penghubungan dari lokasi apa pun dalam sebuah dokumen atau gambar ke lokasi apa pun di dokumen lain. Dengan menggunakan *browser* yang memiliki *Graphical User Interface (GUI)*, tautan dapat dihubungkan ke tujuan dengan mengarahkan kursor ke tautan tersebut dan menekannya[7].

2.5 *User Interface (UI)*

User Interface adalah bentuk tampilan yang berhubungan dengan *user*. Selain itu *user interface* sendiri bisa diartikan sebagai penghubung antara *user* dengan sebuah sistem operasi. *User Interface* merupakan salah satu unsur yang paling penting dari sebuah sistem. Jika sebuah *user interface* dirancang sederhana, maka akan memperlambat *user* dalam mengoperasikan sistem operasi. Bahkan *user interface* yang lemah dapat menyebabkan kegagalan pada suatu sistem. Karena itu perancangan *user interface* harus dibuat sebaik mungkin. Proses perancangan *user interface* dibuat dengan pembuatan model terlebih dahulu, kemudian digambarkan. Hasilnya adalah sebuah *prototype design* yang kemudian akan dievaluasi untuk menguji kualitasnya. Dalam perancangan *user interface*, harus memperhatikan *user* dan konsep pengerjaannya, baik itu pemahaman tentang karakteristik dan perilaku dari *user* yang umum serta populasi dari *user*[8].

2.6 *User Experience*

User experience adalah persepsi seseorang dan responnya dari penggunaan sebuah produk, sistem, atau jasa. *User Experience (UX)* menilai seberapa kepuasan dan kenyamanan seseorang terhadap sebuah produk, sistem, dan jasa. Sebuah prinsip dalam membangun UX adalah khalayak mempunyai kekuasaan dalam menentukan tingkat kepuasan sendiri (*customer rule*). Seberapa pun bagus fitur sebuah produk, sistem, atau jasa, tanpa khalayak yang dituju dapat merasakan kepuasan, kaidah, dan kenyamanan dalam berinteraksi maka tingkat UX menjadi rendah. Perkembangan dunia digital dan *mobile* menjadikan UX menjadi lebih *complicated* dan multidimensi. Kini seseorang dapat mengakses sebuah situs dari berbagai piranti[9].

2.7 *Anaconda*

Anaconda adalah sebuah platform *open-source* untuk bahasa pemrograman *Python*. Tujuan utamanya adalah untuk menyederhanakan manajemen paket dan

penyebaran aplikasi. *Anaconda* menggunakan sistem manajemen paket yang dikelola oleh *conda*. Platform ini mendistribusikan *Python* dengan kinerja yang sangat baik. *Anaconda* memiliki antarmuka yang menarik dan menyediakan berbagai library yang dapat digunakan dalam pemrograman *Python*, seperti *Jupyter Notebook*, *Jupyter Lab*, dan *Spyder*. *Anaconda* juga dirancang untuk menjadi *user-friendly*, sehingga dapat diakses di berbagai platform seperti *Linux*, *MacOS*, dan *Windows*. Hal ini memberikan kenyamanan bagi pengguna. *Anaconda* juga dilengkapi dengan *installer* paket yang handal, menyediakan paket yang lengkap dan selalu terupdate. Oleh karena itu, *Anaconda* sangat direkomendasikan untuk digunakan dalam pengembangan *machine learning* dengan *Python*[10].

2.8 Figma

Figma adalah aplikasi desain berbasis *cloud* dan alat *Prototyping* untuk suatu *project* digital salah satu *design tool* yang biasanya digunakan untuk membuat tampilan aplikasi *mobile*, *desktop*, *website* dan lain-lain. Figma bisa digunakan di sistem operasi *Windows*, *Linux* maupun *MacOS* dengan terkoneksi ke internet. Umumnya Figma banyak digunakan oleh seseorang yang bekerja dibidang UI/UX, *web design* dan bidang lainnya yang sejenis. Selain mempunyai kelengkapan fitur layaknya *Adobe XD*, Figma memiliki keunggulan yaitu untuk pekerjaan yang sama dapat dikerjakan oleh lebih dari satu orang secara bersama-sama walaupun ditempat yang berbeda. Hal tersebut bisa dikatakan kerja kelompok dan karena kemampuan aplikasi figma tersebut lah yang membuat aplikasi ini menjadi pilihan banyak UI atau UX *designer* untuk membuat *Prototype website* atau aplikasi dengan waktu yang cepat dan efektif[11].

2.9 Python

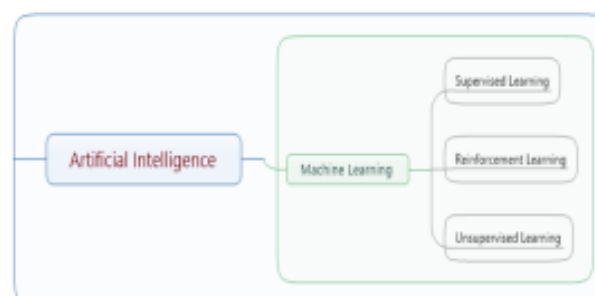
Bahasa pemrograman *Python* ini pertama kali dibuat oleh Guido van Rossum pada awal tahun 1990 di negeri Belanda sebagai pengganti bahasa pemrograman yang disebut ABC. Walaupun Guido adalah orang yang pertama kali menciptakan bahasa pemrograman ini, tetapi bahasa pemrograman *Python* yang digunakan

sekarang merupakan kontribusi dari berbagai sumber. Bahasa pemrograman *Python* merupakan bahasa pemrograman yang dapat dikembangkan oleh siapa saja karena bersifat *open source* atau dengan kata lain bahasa pemrograman ini gratis, dapat digunakan tanpa lisensi, dan dapat dikembangkan semampu yang dapat dilakukan. Sebenarnya bahasa pemrograman *Python* ini mudah dipelajari karena penulisan sintaks yang lebih fleksibel. Selain itu, bahasa pemrograman *Python* ini memiliki efisiensi tinggi untuk struktur data level tinggi, pemrograman berorientasi objek lebih sederhana tetapi efektif, dapat bekerja pada *multi platform*, dan dapat digabungkan dengan bahasa pemrograman lain untuk menghasilkan aplikasi yang diinginkan[12].

2.10 Artificial Intelligence

Kecerdasan Buatan atau *Artificial Intelligence* adalah kecerdasan yang ditambahkan pada sebuah sistem yang dapat diatur sesuai dengan kebutuhan ilmiah. Sistem ini dapat menafsirkan suatu data benar atau salah dengan melakukan pembelajaran terhadap data tersebut. Kecerdasan Buatan diciptakan untuk melakukan fungsi dan tugas tertentu dimana keluaran pekerjaan tersebut dapat menyerupai pekerjaan yang dilakukan oleh manusia[13].

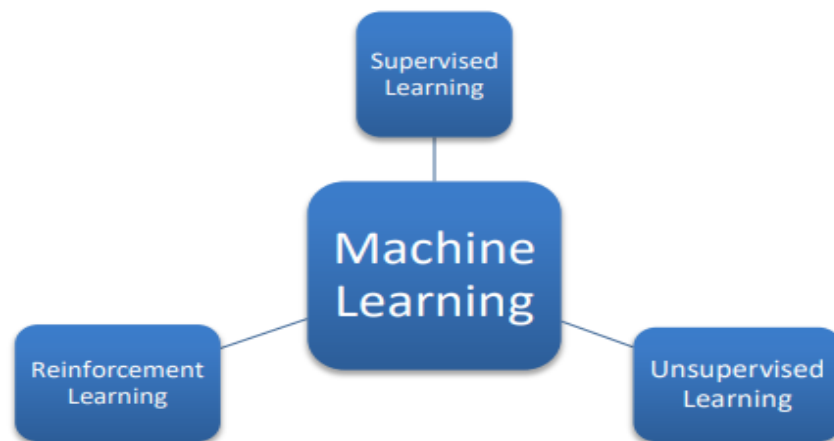
Artificial Intelligence merupakan salah satu bidang dalam ilmu komputer yang dibuat sebagai *software* atau pun *hardware* dengan berbagai fungsi sebagai sesuatu yang dapat berpikir seperti manusia. Kecerdasan buatan ini berfungsi untuk memecahkan masalah pada berbagai bidang seperti bisnis, bahasa alami, matematika, *games*, pertanian, pembuatan keputusan, medis, robotika, analisis *sains*, penalaran, dan lainnya.



Gambar 2.3 Skema *Artificial Intelligence*

2.11 Machine Learning

Machine learning adalah bagian dari kecerdasan buatan, yang mempelajari pengenalan pola untuk mendapatkan solusi yang optimal. *Machine learning* memungkinkan komputer menemukan solusi data yang optimal secara otomatis. *Machine Learning* dapat dikatakan sebagai sebuah aplikasi atau algoritma matematika yang melakukan pembelajaran melalui data dan menghasilkan prediksi masa depan berdasarkan data tersebut. Prediksi atau regresi digunakan oleh mesin untuk menerka keluaran dari suatu data masukan berdasarkan data yang sudah dipelajari dalam *training*. Adapun *Machine Learning* dibagi menjadi 3 jenis: *Supervised Learning*, *Unsupervised Learning*, dan *Reinforcement Learning*[14].



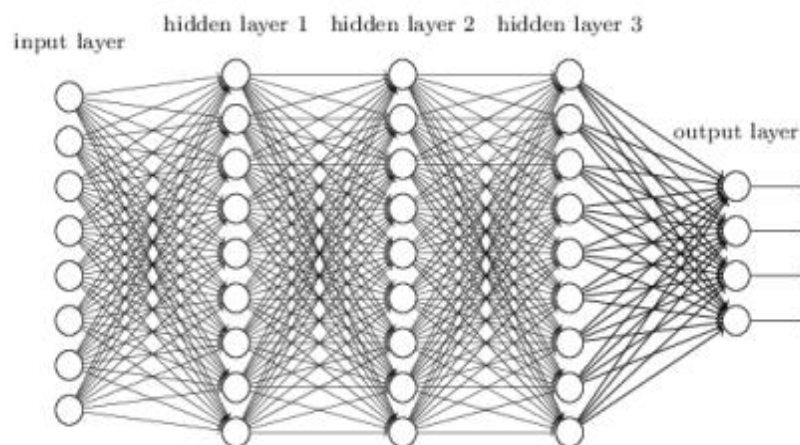
Gambar 2.4 Skema Pembagian *Machine Learning*

Berdasarkan gambar 2.4 skema pembagian *Machine Learning* diatas *Supervised Learning* menggunakan teknik dengan metode klasifikasi dimana kumpulan data sepenuhnya akan diberikan *label* untuk mengklasifikasikan kelas yang tidak dikenal. *Machine* akan mempelajari ciri atau *feature* khusus pada setiap *label* dan membuat model matematisnya. Sedangkan untuk teknik *Unsupervised Learning* menggunakan metode yang disebut *cluster* di mana pada metode ini objek tidak akan diberikan *label* khusus terlebih dahulu. *Machine* akan melakukan pembelajaran dari *feature-feature* khusus objek dan hasilnya *machine* yang akan memberikan *label* khusus untuk mengidentifikasi kelas yang ditentukan.

Kemudian untuk *Reinforcement Learning* biasanya berada antara *Supervised Learning* dan *Unsupervised Learning* di mana jenis ini mempelajari tentang sistem pengambilan keputusan oleh *machine* setelah melakukan pembelajaran pada data yang diberikan[15].

2.12 Deep Learning

Deep Learning adalah jaringan yang terdiri dari beberapa *layer*. *Layer-layer* tersebut merupakan kumpulan dari *node-node*. Sebuah *node* hanya sebagai tempat perhitungan terjadi. Sebuah *node* input digabung dengan *weight* (bobot), setelah *input* dan *weight* ini dijumlahkan dan jumlahnya melewati tahapan yang disebut fungsi aktivasi *node*, untuk menentukan seberapa sejauh sinyal yang berlangsung lebih lanjut melalui jaringan, hal tersebut mempengaruhi hasil akhir. Perbedaan *Deep Learning* dengan *neural network* yaitu lebih banyaknya *hidden layer* pada *Deep Learning*. Jika lebih dari tiga layer (termasuk input dan output) maka memenuhi syarat sebagai “deep” learning. Jadi *Deep Learning* bisa didefinisikan secara teknis yaitu *Machine Learning* yang mempunyai lebih dari satu *hidden layer*. Ilustrasi *Deep Learning* dapat dilihat pada Gambar 2.5 terdapat 4 *layer* dan tiap *layer* mempunyai jumlah *node* yang berbeda-beda[16].



Gambar 2.5 Ilustrasi *Deep Learning*

Deep learning adalah salah satu bidang *machine learning* yang memanfaatkan banyak *layer* pengolahan informasi *nonlinier* untuk melakukan ekstraksi fitur, pengenalan pola, dan klasifikasi. *Deep Learning* adalah sebuah pendekatan dalam

penyelesaian masalah pada sistem pembelajaran komputer yang menggunakan konsep hierarki. Konsep hierarki membuat komputer mampu mempelajari konsep yang kompleks dengan menggabungkan dari konsep-konsep yang lebih sederhana. Jika digambarkan sebuah *graf* bagaimana konsep tersebut dibangun di atas konsep yang lain, *graf* ini akan dalam dengan banyak *layer*, hal tersebut menjadi alasan disebut sebagai *deep learning* [17]. Metode *Deep Learning* yang sedang berkembang salah satunya adalah *Convolutional Neural Network*. Jaringan ini menggunakan masukan berupa gambar, kemudian akan melalui lapisan konvolusi dan diolah berdasarkan filter yang ditentukan, setiap lapisan ini menghasilkan pola dari beberapa bagian citra yang memudahkan proses klasifikasi[18].

2.13 Convolutional Neural Network (CNN)

Convolutional Neural Network merupakan jaringan syaraf yang dikhususkan untuk memproses data yang memiliki struktur *grid*. Sebagai contoh dasarnya adalah berupa citra dua dimensi. Nama konvolusi sendiri merupakan operasi aljabar linear yang mengkalikan matriks dari filter pada citra yang akan diproses. Proses ini disebut dengan lapisan konvolusi dan merupakan salah satu jenis dari banyak lapisan yang bisa dimiliki dalam satu jaringan. Meskipun begitu lapisan konvolusi merupakan lapisan utama yang paling penting untuk digunakan. Jenis lapisan lain yang biasa digunakan adalah *Pooling Layer*, yakni lapisan yang digunakan untuk mengambil nilai maksimal atau nilai rata-rata dari bagian-bagian piksel pada citra[19]. Berikut gambaran alur proses CNN :



Gambar 2.6 Proses *Convolutional Neural Network*

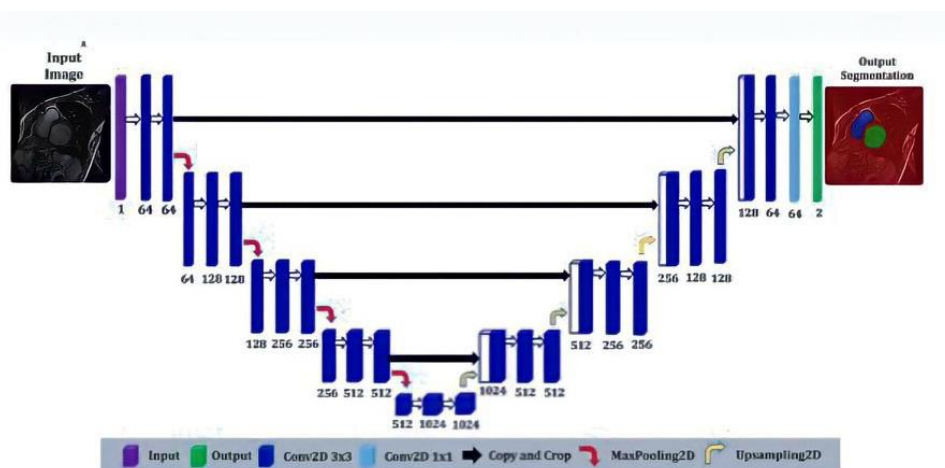
Metode dengan menggunakan *Convolutional Neural Network* (CNN) merupakan salah satu metode dari *Machine Learning* yang merupakan pengembangan dari *Multi Layer Perceptron* (MLP) yang mana dirancang untuk mengolah atau membuat data dari dua dimensi. *Convolutional Neural Network* juga merupakan salah satu jenis metode dari *Deep Neural Network* yang dikarenakan didalamnya memiliki tingkat jaringan dan mempunyai banyak penerapan yang dilakukan di dalam citra. Prinsip kerja dari metode ini mempunyai kesamaan pada metode *Multi Layer Perceptron*, tetapi di dalam metode *Convolutional Neural Network* setiap neuronnya disajikan dalam bentuk dua dimensi yang mana tidak sama seperti pada metode *Multi Layer Perceptron* yang setiap *neuron* hanya memiliki ukuran satu dimensi[20].

2.14 U-Net

Arsitektur U-Net merupakan arsitektur *Convolutional Neural Network* yang populer pada analisis citra medis karena keakuratannya dalam melakukan segmentasi dan dapat meningkatkan akurasi terhadap diagnosis penyakit yang diperiksa. Dalam konteks segmentasi medis, *Convolutional Neural Network* terdapat arsitektur lain seperti V-Net dan SegNet. V-Net adalah arsitektur CNN yang dirancang khusus untuk segmentasi medis dalam tiga dimensi (3D). Arsitektur ini menggunakan konvolusi tiga dimensi dan digunakan untuk memproses *input* citra medis dalam tiga dimensi. V-Net memiliki kekurangan yaitu kemampuan segmentasi mungkin terbatas pada citra medis yang memiliki banyak *detail* pada struktur internalnya.[21]. Selanjutnya SegNet adalah salah satu arsitektur *Convolutional Neural Network* (CNN) yang didesain untuk melakukan segmentasi pada gambar dengan waktu komputasi yang lebih efisien tetapi kekurangan SegNet tidak memiliki kemampuan untuk melakukan segmentasi pada objek yang saling tumpang tindih dalam satu gambar[22]. Keuntungan menggunakan arsitektur U-Net antara lain adalah kemampuannya dalam menangani tugas segmentasi gambar dengan resolusi tinggi. Arsitektur U-Net juga dapat menghasilkan segmentasi gambar yang akurat dan detail, karena informasi lokal dari gambar asli dapat dipertahankan dan dapat menghasilkan segmentasi yang akurat bahkan dengan jumlah data pelatihan yang relatif kecil. Penggunaan

arsitektur CNN yang tepat akan tergantung pada jenis citra medis yang digunakan dan kebutuhan dari aplikasi segmentasi medis.

Segmentasi citra merupakan suatu rangkaian pemrosesan citra yang dilakukan dengan memberi *label* pada setiap piksel citra, dimana citra tersebut dibagi menjadi beberapa bagian sehingga piksel-piksel yang terdapat dalam satu bagian memiliki karakteristik yang sama. Proses segmentasi citra jantung dilakukan dengan menerapkan arsitektur U-Net. Arsitektur U-Net memiliki dua jalur yaitu jalur *encoder* dan *decoder*. Jalur *encoder* digunakan untuk menangkap informasi fitur dari citra *input* dan melakukan pengurangan dimensi pada ukuran citra *input*, sedangkan jalur *decoder* digunakan untuk menangkap informasi fitur dari hasil *encoder* dan sebagai tempat keluarnya hasil segmentasi. Arsitektur U-Net yang digunakan untuk segmentasi citra jantung pada penelitian ini dapat dilihat pada gambar 2.7.



Gambar 2.7 Arsitektur U-NET

Gambar 2.7 menunjukkan arsitektur U-Net yang terdiri dari dua jalur yaitu jalur *encoder* disisi sebelah kiri dan jalur *decoder* disisi sebelah kanan. Jalur *encoder* dimulai dengan proses *double convolution layer* 3×3 dan fungsi aktivasi ReLU yaitu fungsi aktivasi yang sering digunakan dalam jaringan saraf tiruan, yang membuat jumlah *feature maps* menjadi 64, lalu diikuti dengan proses *maxpooling* yang berukuran 2×2 . Pada jalur *encoder*, penelitian ini menggunakan empat blok konvolusi, dimana di setiap blok konvolusi akan meningkat 2 kali lipat jumlah

feature maps nya. Kemudian dilanjutkan dengan blok kelima yang merupakan penghubung antara jalur *encoder* dan jalur *decoder*, dimana tahapannya sama dengan blok pertama hanya saja proses *maxpooling* nya tidak perlu diikuti. Selanjutnya jalur *decoder* dimulai dengan proses *upsampling* berukuran 2×2 , kemudian dilanjutkan dengan proses yang sama seperti blok pertama tanpa proses *maxpooling*. Pada jalur *decoder*, penelitian ini menggunakan empat blok konvolusi, dimana jumlah *feature maps* setiap blok akan berkurang 2 kali lipat sampai jumlah *feature maps* nya sama dengan jumlah *feature maps* semula, dimana tahap akhir pada jalur ini yaitu proses *convolution layer* 1×1 dan fungsi aktivasi sigmoid yang digunakan untuk menghasilkan gambar yang tersegmentasi.

1.14.1 Convolution Layer

Convolution layer merupakan komponen utama dari metode CNN yang digunakan untuk operasi konvolusi pada *output layer* sebelumnya, dimana hasil akhirnya berupa *feature maps*. Hasil dari *convolution layer* bisa dihitung menggunakan persamaan (1) [23] berikut.

$$c_{i,j} = \left(\sum_{u=0}^{n-1} \sum_{v=0}^{n-1} a_{u+i,v+j} \times k_{i+1,j+1} \right) + b_q \quad (1)$$

dimana i merupakan baris, j adalah kolom, n adalah ukuran tinggi kernel, $c_{i,j}$ merupakan matriks input, $k_{i,j}$ merupakan matriks kernel, dan b_q merupakan bias untuk kernel ke- q .

1.14.2 Maxpooling Layer

Maxpooling layer merupakan salah satu lapisan pada *pooling layer* yang digunakan untuk mengekstraksi *patch* dari *feature maps* hasil konvolusi dengan melakukan pergeseran berdasarkan nilai terbesar dari setiap bagian tersebut[24].

2.14.3 Upsampling Layer

Upsampling layer merupakan lapisan yang digunakan untuk membuat ukuran *feature maps* ke dimensi yang lebih tinggi, dimana cara kerjanya dilakukan dengan meningkatkan lebar dan tinggi dari matriks input[25].

2.14.4 Concatenate

Concatenate digunakan untuk menggabungkan *feature maps* hasil proses konvolusi setiap blok pada jalur *encoder* dengan *feature maps* hasil proses *upsampling* pada jalur *decoder* menjadi satu buah matriks input baru dengan ukuran matriks yang baru[26].

2.15 Citra

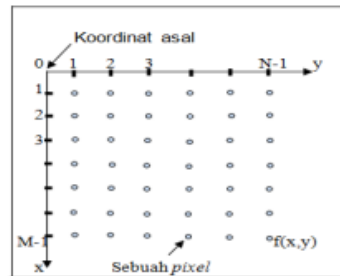
Citra atau dikenal juga sebagai gambar, merujuk pada informasi visual yang berbentuk grafis. Citra memiliki karakteristik yang berbeda dari data teks, karena kaya akan informasi visual. Dalam konteks matematika, citra dapat dianggap sebagai fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Sumber cahaya memancarkan cahaya ke objek, dan objek memantulkan sebagian dari cahaya tersebut. Cahaya pantulan ini kemudian ditangkap oleh perangkat optik seperti mata manusia, kamera, *scanner*, dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. Citra merupakan gambar dua dimensi yang melalui proses *sampling*. Proses *sampling* terdiri dari dua jenis, yaitu *downsampling* dan *upsampling*. *Downsampling* merupakan proses mengurangi jumlah piksel atau resolusi spasial citra sehingga menghasilkan citra dengan ukuran yang lebih kecil. Sementara itu, *upsampling* adalah proses meningkatkan jumlah piksel atau meningkatkan resolusi gambar. Citra sebagai keluaran dari suatu sistem perekaman data dapat bersifat :

1. Optik berupa foto.
2. Analog berupa sinyal video seperti gambar pada monitor televisi.
3. Digital yang dapat langsung di simpan pada media penyimpan magnetik[27].

2.16 Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data 2 dimensi. Citra digital adalah sebuah larik (*array*) yang berisi nilai-nilai *real* maupun kompleks yang direpresentasikan dengan deretan bit tertentu.

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x , y , dan nilai amplitudo f secara keseluruhan berhingga (finite) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut citra digital. Gambar 2.8 menunjukkan posisi koordinat citra digital.



Gambar 2.8 Koordinat Citra Digital

Citra digital merupakan suatu array dua dimensi atau suatu matriks yang elemennya menyatakan tingkat keabuan dari elemen gambar; jadi informasi yang terkandung bersifat diskrit. Citra digital tidak selalu merupakan hasil langsung data rekaman suatu sistem. Kadang-kadang hasil rekaman data bersifat kontinu seperti gambar pada monitor televisi, foto sinar-X, dan lain sebagainya. Dengan demikian untuk mendapatkan suatu citra digital diperlukan suatu proses konversi, sehingga citra tersebut selanjutnya dapat diproses dengan komputer[28].

2.17 Jenis Citra

Ada tiga jenis citra yang umum digunakan dalam pemrosesan citra. Ketiga jenis citra tersebut yaitu, citra berwarna, citra berskala keabuan, dan citra biner.

a. Citra Berwarna atau biasa dinamakan citra RGB, merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (merah), G (hijau), dan B (biru). Setiap komponen warna menggunakan delapan bit (nilainya berkisar antara 0 sampai dengan 255). Dengan demikian, kemungkinan warna yang dapat disajikan mencapai $255 \times 255 \times 255$ atau 16.581.375 warna. Tabel 2.1 menunjukkan contoh warna dan nilai R, G, dan B.

Tabel 2.1 Warna dan Nilai Penyusun Warna

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255

Penyimpanan pada citra *true color* di dalam memori berbeda dengan citra *grayscale*. Setiap piksel dari sebuah citra grayscale 256 gradasi warna diwakili oleh 1 byte. Sedangkan 1 piksel citra *true color* diwakii oleh 3 *byte*, dimana masing-masing *byte* menjelaskan warna merah (Red), hijau (Green), dan biru (Blue). Gambar 2.9 adalah contoh citra berwarna.



Gambar 2.9 Citra Berwarna

b. Citra Berskala Keabuan Sesuai dengan nama yang melekat, citra jenis ini menangani gradasi warna hitam dan putih, yang tentu saja menghasilkan efek warna abu-abu. Pada jenis gambar ini, warna dinyatakan dengan intensitas. Dalam hal ini, intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih. Contoh citra berskala keabuan seperti Gambar 2.10.



Gambar 2.10 Citra Keabuan

Istilah lain citra keabuan adalah citra *grayscale* yaitu citra yang nilai pixelnya merepresentasikan derajat keabuan atau instensitas warna putih. Nilai instensitas paling rendah merepresentasikan warna hitam dan nilai intensitas paling tinggi

merepresentasikan warna putih. Pada umumnya citra *grayscale* memiliki kedalaman pixel 8 bit (256 derajat keabuan), tetapi ada juga citra *grayscale* yang kedalaman pixelnya bukan 8 bit, misalnya 16 bit untuk penggunaan yang memerlukan ketelitian tinggi.

c. Citra Biner adalah citra dengan setiap piksel hanya dinyatakan dengan sebuah nilai dari dua kemungkinan (yaitu nilai 0 dan 1). Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Citra jenis ini banyak dipakai dalam pemrosesan citra, misalnya untuk kepentingan memperoleh tepi bentuk suatu objek[29]. Gambar 2.11 menyatakan citra biner.



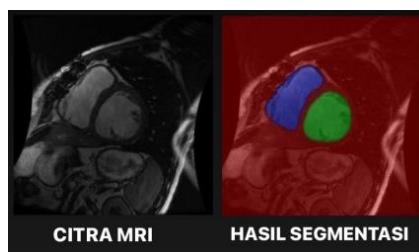
Gambar 2.11 Citra Biner

2.18 Segmentasi Citra

Segmentasi citra adalah proses membagi citra digital menjadi beberapa daerah atau kelompok, dimana masing-masing daerah terdiri dari sekumpulan piksel. Segmentasi citra digunakan untuk mencari obyek yang ingin di cari dan batas-batas bentuk objek seperti garis, kurva dalam citra. Segmentasi merupakan proses partisi gambar digital ke beberapa daerah dengan tujuan untuk menyederhanakan ataupun merubah representasi gambar menjadi sesuatu yang lebih bermakna dan mudah dianalisa[30].

Segmentasi citra medis adalah proses membagi citra medis menjadi beberapa bagian atau wilayah yang terpisah untuk memudahkan analisis dan interpretasi. Tujuan utama segmentasi citra medis adalah untuk memperoleh informasi yang spesifik tentang organ di dalam citra medis sehingga dokter atau tenaga medis dapat membuat keputusan diagnosis dan perencanaan perawatan yang lebih baik.

Dalam konteks citra medis, segmentasi dapat dilakukan pada berbagai jenis citra, seperti pada citra MRI.



Gambar 2.12 Hasil Segmentasi Pada Citra MRI

2.19 TensorFlow

TensorFlow adalah suatu *open-source software library* untuk *dataflow programming*. *TensorFlow* merupakan suatu *library* matematika simbolik yang juga digunakan untuk aplikasi *Machine Learning* seperti *Neural Network*. *TensorFlow* dikembangkan oleh Google Brain Team dan juga digunakan *research* oleh Google sendiri. *Tensorflow* dirilis pertama kali pada 11 Februari 2017. Keunggulan *TensorFlow* adalah dapat dijalankan di beberapa CPU dan GPU (*Graphic Processing Unit*). *Tensorflow* dapat digunakan pada 64-bit *Linux*, *macOS*, *Windows*, dan *Mobile Computing Platforms* termasuk Android dan iOS[31].

2.20 AI Project Cycle

AI Project Cycle merupakan sebuah metode yang dilakukan untuk membuat proyek berbasis AI secara utuh. Metode ini akan memandu secara terorganisir bagaimana pembuatan dan penyelesaian proyek AI[32]. *AI Project Cycle* dibagi menjadi 5 *stage* seperti yang ditunjukkan pada gambar 2.13 berikut.



Gambar 2.13 AI Project Cycle

1. *Problem Scoping*

Tahap ini dilakukan untuk mengidentifikasi masalah dan juga batasannya sehingga tujuan atau target akan semakin jelas dan terarah dan solusi AI yang dibuat akan lebih mudah dibuat. Cara terbaik untuk melakukan tahap ini adalah dengan mendefinisikan 4W yaitu *What*, *Who*, *Why*, dan *Where* dari masalah yang akan diidentifikasi.

2. *Data Acquisition*

Pada tahap ini dilakukan proses pengumpulan data dimana data yang akan dibutuhkan dan akan di proses pada proyek ini akan dikumpulkan pada tahap ini. Data dapat berupa fakta, statistik, referensi dan metode-metode pelaksanaan. Dalam hal *Machine Learning* atau *Deep Learning* maka data ini adalah data yang akan dilatih untuk membentuk suatu model pembelajaran. Data dapat diambil dari berbagai sumber seperti *website*, *sensor*, *survey*, *kamera*, *API*, dan lain-lain.

3. *Data Exploration*

Tahap ini bertujuan untuk memastikan data yang disiapkan memiliki parameter sesuai dengan kebutuhan proses pembentukan solusi AI. Data akan disusun dengan pola tertentu untuk membentuk data agar siap dilakukan pemodelan.

4. *Modelling*

Tahap ini merupakan tahapan inti dimana proses pembuatan algoritma dalam bahasa pemrograman tertentu sebagai metode pembelajaran mesin (*training phase*) dibuat. *Modelling* bertujuan untuk menemukan pola-pola khusus dari data-data yang diberikan sebelumnya untuk membuat suatu keputusan dan atau melakukan prediksi.

5. *Evaluation*

Tahap ini berisikan peninjauan terhadap performa model yang telah dilatih, apakah model sudah baik atau belum. Umumnya model akan diberikan *score* dengan parameter khusus. Jika belum didapatkan model yang baik, maka dapat dilakukan *hyperparameter tuning* dimana proses konfigurasi proses *modelling*

akan diubah. Proses ini dilakukan dengan metode *trial and error* hingga mendapat model terbaik.

2.21 SDLC (*Software Development Life Cycle*)

SDLC (*Software Development Life Cycle*) berarti sebuah siklus pengembangan perangkat lunak yang terdiri dari beberapa tahapan dan penting keberadaannya terutama dari segi pengembangannya[33]. SDLC berfungsi untuk menggambarkan tahapan-tahapan utama dan langkah-langkah dari setiap tahapan yang secara garis besar terbagi dalam lima kegiatan utama, yaitu: analisis, desain, implementasi, pengujian, dan pemeliharaan. SDLC sendiri tidak mengharuskan untuk menggunakan metode tertentu dalam setiap tahapan yang ada. Ada banyak metode pengembangan perangkat lunak yang dapat digunakan dalam SDLC, seperti *Waterfall*, *Agile*, *Rapid Application Development (RAD)*, *Scrum*, dan sebagainya. Pemilihan metode pengembangan perangkat lunak akan tergantung pada kebutuhan dan karakteristik dari proyek pengembangan perangkat lunak itu sendiri. Jadi, SDLC juga dapat disesuaikan dengan kebutuhan dan keadaan proyek tertentu[34]. Pada Penelitian ini menggunakan metode *Waterfall* dengan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modelling*), konstruksi (*contruction*), serta penyerahan sistem ke para pengguna (*deployment*) yang disesuaikan dengan kebutuhan karakteristik aplikasi[35].

2.22 Google Cloud Platform

Google Cloud Platform adalah layanan *public cloud computing* dari Google yang terdiri dari beragam layanan. Dengan menggunakan layanan yang ada pada *Google Cloud Platform*, kita dapat membuat aplikasi *virtual machine*, jaringan, hingga *machine learning* sesuai keinginan dan kebutuhan kita. Google Cloud Platform (GCP) menyediakan berbagai layanan dan alat yang dapat digunakan dalam pembuatan model *machine learning* yang andal, *scalable*, dan terintegrasi dengan teknologi Google lainnya. Dalam pembuatan model *machine learning*,

GCP menawarkan beberapa layanan dan alat seperti *Google Colab*, *Cloud AutoML*, *AI Platform*, *TensorFlow*, *Kubeflow*, *Dataflow*, *BigQuery ML*, dan *Cloud TPU*. Pengguna GCP dapat memilih layanan dan alat yang sesuai dengan kebutuhan dan kemampuan teknis untuk membangun aplikasi *machine learning* yang berkualitas tinggi dan dapat dijalankan pada skala besar. GCP juga menawarkan harga yang terjangkau untuk layanan dan alat yang disediakan sehingga dapat membantu pengguna menghemat biaya pembuatan dan operasional aplikasi *machine learning*[36].

Platform ini menyediakan berbagai macam layanan *hosting* mulai dari komputasi, penyimpanan dan pengembangan aplikasi yang berjalan di perangkat keras Google. Beberapa produk *cloud computing* di *Google Cloud* meliputi:

- *Google Compute Engine*, merupakan layanan yang memberikan dukungan instance VM untuk *hosting*.
- *Google App Engine*, merupakan platform yang berfungsi sebagai layanan yang memberikan *developer* perangkat lunak dan akses ke *hosting* Google yang dapat diskalakan.
- *Google Cloud Storage*, merupakan *platform* penyimpanan *cloud* yang dirancang untuk menyimpan kumpulan data besar dan tidak terstruktur. Google juga menawarkan opsi penyimpanan *database*, 20 termasuk penyimpanan non-relasional *Cloud Datastore for NoSQL*, penyimpanan *cloud SQL for MySQL*, dan database *Cloud Bigtable* asli Google.
- *Google Kubernetes Engine (GKE)*, merupakan sistem manajemen dan orkestrasi untuk kontainer Docker yang berjalan dalam layanan *cloud* publik Google. *Google Kubernetes Engine* didasarkan pada Kubernetes, sistem manajemen kontainer *open source* Google.
- *Google Cloud's operation suite*, merupakan seperangkat alat terintegrasi untuk memantau, mencatat, dan melaporkan layanan terkelola yang mendorong aplikasi dan sistem di *Google Cloud*.
- *Serverless Computing*, menyediakan alat dan layanan *Cloud Functions* dan *Cloud Run*.
- Layanan *Database*, mencakup *Cloud Bigtable* *CloudSpanner*, dan *CloudSQL*.

2.23 SQLite

SQLite adalah sebuah perpustakaan (*library*) yang mengimplementasikan mesin basis data *self-contained*, *serverless*, *zero-configuration*, dan *transactional*. Dalam konteks *self-contained*, *SQLite* membutuhkan sedikit sekali ketergantungan terhadap perpustakaan eksternal atau sistem operasi. *Serverless* berarti *SQLite* dapat mengakses basis data langsung dari *file* tanpa melalui proses server, dan tidak mendukung akses jarak jauh (*remote access*). Dalam hal ini, *database SQLite* tidak dapat dikendalikan dari jarak jauh melalui jaringan komputer, baik itu melalui jaringan lokal (*intranet*) maupun *internet*. Hal ini berbeda dengan sebagian besar mesin basis data SQL yang umumnya diimplementasikan sebagai proses server yang terpisah. *Zero-configuration* menandakan bahwa *SQLite* tidak memerlukan instalasi sebelum digunakan. Sifat *transactional SQLite* mengacu pada basis data yang dapat menjalankan transaksi, di mana proses *query* mengikuti prinsip *Atomic, Consistent, Isolated, and Durable* (ACID) dalam melakukan perubahan. *SQLite database* juga merupakan sebuah sistem manajemen basis data relasional yang bersifat *ACID-compliant* dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. Namun ada satu hal yang masih menjadi kekurangan di dalam *database SQLite* yaitu masalah keamanan data karena *database* tersebut masih bersifat *plain*, dalam arti tidak terlindungi oleh enkripsi. *Database SQLite* tidak memiliki mekanisme untuk memproteksi data seperti *password* atau enkripsi. Siapapun yang memiliki akses ke fisik file *database*, maka yang bersangkutan akan bisa membukanya dan melihat isi data di dalamnya[37].

2.24 Blackbox Testing

Blackbox testing adalah deskripsi dari interaksi sistem yang spesifik dimana tester harus menguji respon dari sistem. *Blackbox testing* mengutamakan interaksi dari sistem dan tidak berfokus kepada mekanisme cara kerja dari sistem tersebut. Ada beberapa cara untuk melakukan *blackbox testing* adalah sebagai berikut :

- a. *Boundary Value Analysis* (BVA)
- b. *Cause-Effect Graphing* (CEG)

- c. *Decision Tables* (DT)
- d. *Equivalence Partitioning* (EP)
- e. *Orthogonal Array Testing* (OAT)
- f. *Random Testing* (RT)
- g. *Specification-Based Mutation Testing* (SBMT)
- h. *State-Transition Diagram Testing* (STT)
- i. *Syntax Testing* (ST)
- j. *Worst Case Testing* (WCT)[38].

2.25 *System Usability Scale*

System Usability Scale (SUS) digunakan untuk mengukur seberapa mudah pengguna dapat menggunakan suatu produk atau sistem dengan tujuan untuk meningkatkan pengalaman pengguna dan memaksimalkan efisiensi serta efektivitas penggunaan produk atau sistem tersebut. SUS memiliki 10 pertanyaan dan 5 pilihan jawaban. Pilihan jawaban terdiri dari sangat tidak setuju sampai sangat setuju. SUS memiliki skor minimal 0 dan skor maksimal 100[39]. SUS dalam bahasa aslinya menggunakan bahasa Inggris. Berikut 10 pertanyaan dari *System Usability Scale* yang sudah diterjemahkan dalam bahasa Indonesia.

No.	Item in Indonesian
1	Saya berpikir akan menggunakan sistem ini lagi.
2	Saya merasa sistem ini rumit untuk digunakan.
3	Saya merasa sistem ini mudah untuk digunakan.
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada sistem ini.
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.
8	Saya merasa sistem ini membingungkan.
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.

Gambar 2.14 Pertanyaan SUS

Gambar 2.14 diatas menampilkan daftar 10 pertanyaan SUS. SUS memiliki 5 pilihan jawaban. Mulai dari sangat tidak setuju, tidak setuju, ragu-ragu, setuju, dan sangat setuju. Skor masing-masing jawaban mulai dari 1 sampai 5. Berikut pilihan jawaban beserta skornya.

Jawaban	Skor
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2
Ragu-ragu (RG)	3
Setuju (S)	4
Sangat Setuju (SS)	5

	STS	TS	RG	ST	SS
1. Saya berpikir akan menggunakan sistem ini lagi.	1	2	3	4	5

Gambar 2.15 Chart Nilai Pertanyaan SUS

Gambar 2.15 adalah sebuah *chart* penilaian setiap jawaban SUS. Setelah melakukan pengumpulan data dari responden, kemudian data tersebut dihitung. Dalam cara menggunakan *System Usability Scale* ada beberapa aturan dalam perhitungan skor SUS. Berikut ini aturan-aturan saat perhitungan skor pada kuesionernya:

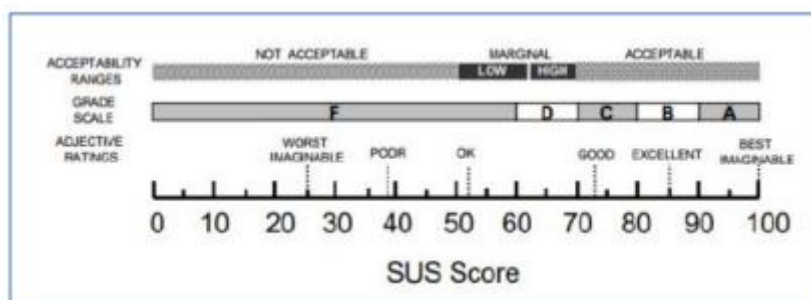
- Setiap pertanyaan bernomor ganjil, skor setiap pertanyaan yang didapat dari skor pengguna akan dikurangi 1.
- Setiap pertanyaan bernomor genap, skor akhir didapat dari nilai 5 dikurangi skor pertanyaan yang didapat dari pengguna.
- Skor SUS didapat dari hasil penjumlahan skor setiap pertanyaan yang kemudian dikali 2,5.

$$\bar{x} = \frac{\sum x}{n}$$

\bar{x}	=	skor rata-rata
$\sum x$	=	jumlah skor SUS
n	=	jumlah responden

Gambar 2.16 Rumus Perhitungan SUS

Gambar 2.16 menunjukkan rumus perhitungan SUS yaitu adalah rata rata dari seluruh jumlah skor sus dibagi dengan jumlah responden. Aturan perhitungan skor untuk berlaku pada 1 responden. Untuk perhitungan selanjutnya, skor SUS dari masing-masing responden dicari skor rata-ratanya dengan menjumlahkan semua skor dan dibagi dengan jumlah responden.



Gambar 2.17 Grafik Kesimpulan SUS

Gambar 2.17 adalah grafik kesimpulan nilai skor SUS ada 3 teknik penilaian yaitu berdasarkan *acceptability range*, *grade scale* dan *adjective ratings*. Kesimpulan dari cara menggunakan *System Usability Scale* adalah setelah dihitung didapatkan skor rata-rata SUS dari semua responden. Skor tersebut kemudian disesuaikan dengan penilaian SUS. Masuk kategori mana hasil pengujian dengan skor rata-rata yang sudah didapat. Skor rata-rata SUS dari banyaknya penelitian adalah 68, maka jika nilai SUS di atas 68 akan dianggap di atas rata-rata dan nilai di bawah 68 di bawah rata-rata. Jika skor yang didapat dibawah 68 berarti ada masalah pada usability dan butuh perbaikan. Namun kesimpulan akhir bisa juga ditentukan melalui penilaian seperti pada gambar diatas[40].

2.26 Penelitian Terkait

Penelitian terkait berisi artikel ilmiah ataupun jurnal yang berhubungan dengan penelitian ini dan dijadikan referensi. Beberapa hal yang dapat dijadikan referensi seperti kesamaan studi kasus ataupun penggunaan metode dalam pengembangan sistem. Adapun beberapa artikel atau jurnal ilmiah yang penulis jadikan referensi untuk penelitian ini adalah :

1. Penelitian dari Redho Islami yang berjudul “Implementasi *Deep Learning* Dalam Mendeteksi Penyakit Menggunakan *Convolutional Neural Network* Dan *Tensorflow*”[41] menerangkan bahwa Pemanfaatan *computer vision* sangat berguna bagi bidang kesehatan dengan menggunakan *convolutional neural network* dan *tensorflow* dalam mendeteksi sebuah objek penyakit. Penelitian ini menghasilkan sebuah sistem implementasi *artificial intelligence* untuk mengklasifikasikan dan mendeteksi penyakit yang ada di dalam saluran pencernaan (gastrointestinal) dengan metode *deep learning* menggunakan *convolutional neural network* dan *tensorflow*.
2. Penelitian dari Jenny Alanna Engka yang berjudul “Penerapan *Machine Learning* Dalam Sistem Klasifikasi Penyakit Manusia Dengan Model *Decision Tree* Dan *Neural Network*”[42] menjelaskan bahwa dengan menggunakan *machine learning*, sebuah sistem dapat mengklasifikasi penyakit manusia berdasarkan gejala yang dialami seseorang. Penelitian ini menghasilkan desain model *machine learning* terbaik untuk mengklasifikasi penyakit menggunakan *decision tree* dan *neural network* dan menggunakan model tersebut untuk dikonversi menjadi *tensorflow lite*.
3. Penelitian dari Jiyuan Shi yang berjudul “*Improvement of Damage Segmentation Based on Pixel-Level Data Balance Using VGG-Unet*”[25] menjelaskan bahwa untuk segmentasi kerusakan berukuran besar (seperti korosi), *Cropping Segmentation* memberikan hasil yang lebih baik, sedangkan untuk segmentasi kerusakan kecil (seperti retak), hasilnya lebih baik dengan *Squashing Segmentation*. Hal ini disebabkan oleh perbedaan konsentrasi data yang *valid* dalam *dataset*.
4. Penelitian dari S Niyas yang berjudul “*Medical Image Segmentation with 3D Convolutional Neural Networks: A Survey*”[43] menjelaskan bahwa *Convolutional Neural Networks* memiliki peran analisis gambar medis berbantuan komputer dalam membantu praktisi medis untuk diagnosis klinis dan menentukan rencana perawatan optimal.
5. Penelitian dari Olaf Ronneberger yang berjudul “*U-Net: Convolutional Networks for Biomedical Image Segmentation*”[44] menerangkan bahwa

Arsitektur U-net mampu mencapai performa yang sangat baik pada aplikasi segmentasi biomedis yang berbeda-beda.

Tabel 2.2 Penelitian Terkait

Judul	Peneliti	Tahun	Hasil Penelitian
<i>U-Net: Convolutional Networks for Biomedical Image Segmentation</i>	Olaf Ronneberger	2015	Arsitektur U-net mampu mencapai performa yang sangat baik pada aplikasi segmentasi biomedis yang berbeda-beda.
Implementasi <i>Deep Learning</i> Dalam Mendeteksi Penyakit Menggunakan <i>Convolutional Neural Network</i> Dan Tensorflow	Redho Islami	2020	Implementasi <i>artificial intelligence</i> untuk mengklasifikasikan dan mendeteksi penyakit yang ada di dalam saluran pencernaan (<i>gastrointestinal</i>) dengan metode <i>deep learning</i> menggunakan <i>convolutional neural network</i> dan <i>tensorflow</i> .
Penerapan <i>Machine Learning</i> Dalam Sistem Klasifikasi Penyakit Manusia Dengan Model <i>Decision Tree</i> Dan <i>Neural Network</i>	Jenny Alanna Engka	2021	Desain model <i>machine learning</i> terbaik untuk mengklasifikasi penyakit menggunakan <i>decision tree</i> dan <i>neural network</i> dan menggunakan model tersebut untuk dikonversi menjadi tensorflow lite.
<i>Improvement of Damage Segmentation Based on Pixel-Level Data Balance Using VGG-UNet</i>	Jiyuan Shi	2021	<i>Squashing Segmentation</i> lebih baik dalam mendeteksi posisi keseluruhan dari kerusakan. Namun, dibandingkan dengan <i>Cropping Segmentation</i> , metode ini tidak memiliki kinerja yang baik dalam mendeteksi lokasi yang tepat dari kerusakan, karena beberapa informasi fitur hilang selama proses kompresi sebelum pelatihan.

Judul	Peneliti	Tahun	Hasil Penelitian
<i>Medical Image Segmentation with 3D Convolutional Neural Networks: A Survey</i>	S Niyas	2022	Peran analisis gambar medis berbantuan komputer dalam membantu praktisi medis dalam diagnosis klinis dan menentukan rencana perawatan optimal.

III. METODELOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Waktu penelitian dilakukan pada bulan Januari 2023 sampai dengan Mei 2023 yang bertempat di Universitas Lampung. Berikut adalah tabel 3.1 yang menunjukkan jadwal kegiatan penelitian yang dilakukan.

Tabel 3.1 Jadwal Penelitian

No	Aktivitas	Januari 2023				Februari 2023				Maret 2023				April 2023				Mei 2023			
		Minggu ke -																			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	<i>Problem Scoping</i>	■																			
2	<i>Data Acquisition</i>		■																		
3	<i>Data Exploration</i>			■	■																
4	<i>Modelling</i>					■	■														
5	<i>Evaluation</i>							■	■												
6	<i>User Interface Design</i>									■	■										
7	<i>Front-End Development</i>											■	■								
8	<i>Back-End Development</i>													■	■	■					
9	<i>Deployment</i>																	■	■		
10	<i>Testing</i>																			■	■
11	<i>Reporting</i>					■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

3.2 Project Scope

Pengerjaan penelitian ini berkaitan dengan beberapa penelitian lain yang berkolaborasi untuk membangun aplikasi segmentasi jantung menggunakan kecerdasan buatan dengan visualisasi berbasis *web* dengan pembagian anggota dan deskripsi pengerjaan sebagai berikut :

1. *Modelling* atau pemodelan adalah proses memilih metode yang digunakan. Bagian ini berisi proses mengetahui kinerja model yang digunakan. Pada bagian ini dikerjakan oleh Muhammad Alana dari Universitas Sriwijaya dan Jatmiar Fikriaziz dari Universitas Jendral Soedirman.
2. *User Interface Design* adalah proses merancang antarmuka pengguna yang digunakan dalam aplikasi. Pada bagian ini dikerjakan oleh Nindy dari Universitas Teknologi Yogyakarta dan Tania Argadini dari Universitas Lampung.
3. *Front-End Development* adalah proses mengimplementasikan desain aplikasi ke dalam kode-kode yang dapat dijalankan oleh browser. Pada bagian ini dikerjakan oleh Tania Argadini dari Universitas Lampung.
4. *Back-End Development* melibatkan pembuatan dan pengelolaan *server*, *database*, dan logika aplikasi yang bekerja di balik layar untuk memproses permintaan dari pengguna dan menyediakan data yang diperlukan. Pada bagian ini dikerjakan oleh Muhammad Ahyaruddin dari Universitas Teknologi Digital Indonesia

3.3 Alat Penelitian

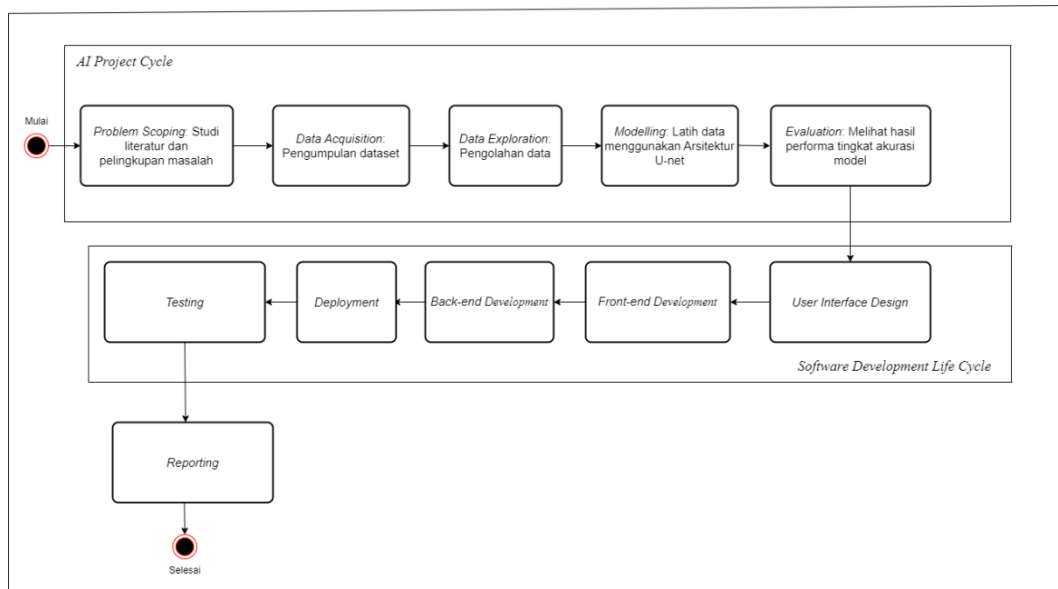
Adapun alat yang digunakan dalam penelitian ini adalah sebagai berikut :

Tabel 3.2 Alat Penelitian

No	Nama Alat	Spesifikasi	Deskripsi
1.	Laptop	Lenovo 81H4 AMD A9 dengan RAM 4 GB dan sistem operasi windows 10	Perangkat keras yang digunakan dalam proses pembuatan Aplikasi

2.	Aplikasi Visual Studio Code	Version 1.74.2	Perangkat lunak yang digunakan dalam proses pembuatan Aplikasi
3.	Figma	Web Version	Aplikasi yang digunakan untuk mendesain rancangan antarmuka Aplikasi.

3.4 Tahapan Penelitian



Gambar 3.1 Tahapan Penelitian

Gambar 3.1 adalah tahapan penelitian yang didasarkan pada *AI Project Cycle* kemudian dijabarkan lebih luas lagi sesuai kebutuhan dengan menggunakan metode SDLC (*Software Development Life Cycle*) untuk pembuatan dan perubahan sistem yang digunakan untuk mengembangkan sistem rekayasa perangkat lunak. Proses *AI project cycle* meliputi *problem scoping*, *data acquisition*, *data exploration*, *modelling* dan *evaluation*. Sedangkan untuk proses SDLC meliputi *user interface design*, *front-end development*, *back-end development*, *deployment*, dan *testing*.

3.3.1 *Problem Scoping*

3.3.1.1 **Analisa Masalah**

Setelah menentukan ide yang akan digunakan dalam penelitian ini maka langkah awal yang dilakukan adalah menganalisis masalah dan solusi yang ditawarkan. Pada *problem scoping* ini terdapat beberapa analisis masalah terkait ide proyek yang akan dibuat dengan rumusan 4W.

Kesimpulan dari rumusan ini ditemukan beberapa analisa masalah terkait adalah sebagai berikut. Permasalahan utama pada proyek ini adalah untuk membuat suatu aplikasi yang dapat melakukan segmentasi pada citra MRI dikarenakan masalah utamanya adalah pada citra MRI terdapat berbagai organ dalam tubuh seperti hati, jantung, empedu dan ginjal. Sehingga dibutuhkan aplikasi untuk melakukan segmentasi untuk dapat mendapatkan bagian jantung saja secara otomatis.

3.3.1.2 **Analisa 4W**

Berikut adalah penjabaran 4W dalam *problem scoping* penelitian ini :

- *Who*

1. Siapa saja yang mengalami masalah tersebut?

Dokter dan Ahli radiologi.

- *What*

1. Apa masalah utamanya?

Hasil yang didapatkan pada citra MRI masih terdapat bagian-bagian lain seperti ginjal, empedu, dan hati. Bagian-bagian lain ini akan mengganggu proses deteksi penyakit pada jantung.

- *Where*

1. Dimana saja si calon pengguna mengalami masalah ini?

Di rumah sakit dan klinik.

- *Why*

1. Apa nilai utama dari permasalahan ini?

Nilai utama dari permasalahan ini memisahkan bagian yang berbeda pada citra MRI untuk dapat mendapatkan bagian jantung.

2. Bagaimana kamu bisa memperbaiki (sebagian atau seluruh) dari masalah tersebut?

Diperlukan suatu teknologi yang dapat menampilkan dengan jelas informasi sesuai kebutuhan, salah satu caranya adalah dengan pengolahan citra digital berupa segmentasi gambar.

3.3.1.3 Analisa Kebutuhan

Analisa kebutuhan merupakan tahap melakukan identifikasi terhadap tujuan pengembangan sistem, mengidentifikasi kebutuhan pengguna meliputi kebutuhan fungsional dari aplikasi yang dibuat. Adapun setelah dilakukannya identifikasi terhadap tujuan pengembangan aplikasi dan kebutuhan pengguna didapat kebutuhan fungsional sebagai berikut :

Kebutuhan Fungsional:

1. Sistem yang dapat melakukan *sign up*.
2. Sistem yang dapat melakukan *sign in*.
3. Sistem yang dapat menampilkan hasil prediksi segmentasi pada citra MRI.
4. Sistem yang dapat melakukan *logout*.

3.3.2 Data Acquisition

Pada tahap ini dilakukan pengumpulan data gambar berupa *raw image* agar layak untuk diolah dan dianalisis lebih pelabelan data mentah ke dalam bentuk yang sesuai untuk algoritma *machine learning*. Pada tahap ini diambil *dataset* yang terdiri dari 122 *image*.

3.3.3 Data Exploration

Pada tahap ini *dataset* yang disiapkan akan diolah terlebih dahulu. Pengolahan data dilakukan agar format setiap *image* sesuai dengan prasyarat algoritma pendeteksian yang dibuat. Setelah mendapatkan *dataset* berupa *raw image* citra jantung, maka data tersebut akan diolah dengan merubah format *image*. *Raw image* yang diperoleh berupa gambar dengan format '.tiff'. *Image* akan diubah menjadi format '.jpg' untuk mengecilkan ukuran *file image* tanpa merubah kualitasnya.

3.3.4 Modelling

Modelling atau pemodelan adalah proses memilih metode yang digunakan. Pada tahap ini berisi proses mengetahui kinerja model yang digunakan, mengoptimalkan parameter pada model dan evaluasi model sehingga dapat diketahui seberapa baik kinerja model tersebut sebelum digunakan dalam produksi.

3.3.5 Evaluation

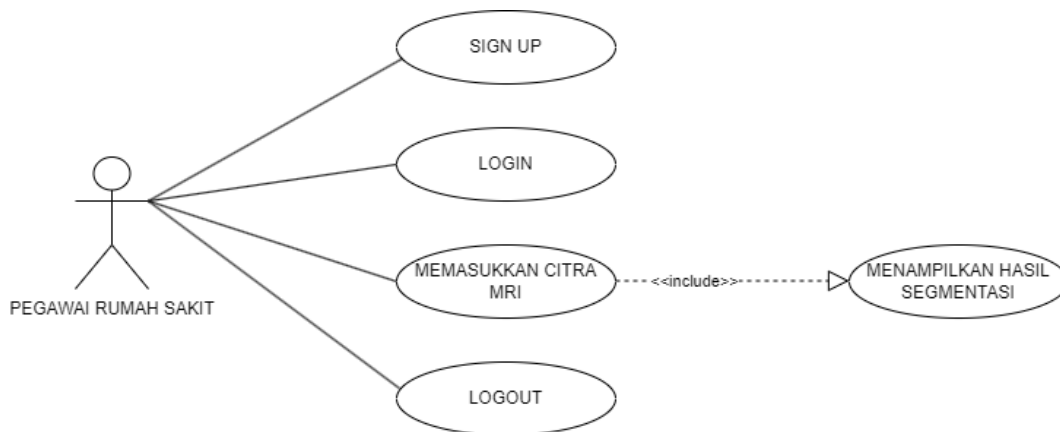
Tahap berikutnya adalah *evaluation* dimana terdiri dari dua tahap evaluasi. Evaluasi pertama dilakukan ketika tahap *modelling* dimana model akan dievaluasi untuk setiap visualisasi dari hasil pemodelan. Setelah dilakukan *training* untuk pembuatan model, kemudian dilakukan uji deteksi model terhadap *dataset* yang sudah disiapkan sebelumnya. Setelah proses itu dilakukan maka didapatkan hasil akurasi sebesar 99%. Berdasarkan proses pelatihan dengan *batch_size* 4 yaitu 4 data per *batch* dan dilakukan sebanyak 40 *epochs* atau 40 kali iterasi.

3.3.6 User Interface Design

Pada tahap *user interface design*, penulis membuat tahapan desain fungsional berupa *use case diagram* dan *activity diagram* penelitian pada sistem yang akan dibangun. *Use case diagram* digunakan dalam pengembangan perangkat lunak untuk menggambarkan interaksi antara aktor dengan sistem. *Activity diagram*

dijelaskan dalam bagian yang terpisah dari tiap aktivitas yang ada dalam program diproses berdasarkan hasil perencanaan persyaratan sebelum sistem dibangun. Berikut *use case diagram* dan *activity diagram* yang digunakan, yaitu:

3.3.6.1 Use Case Diagram



Gambar 3.2 Use Case Diagram Aplikasi

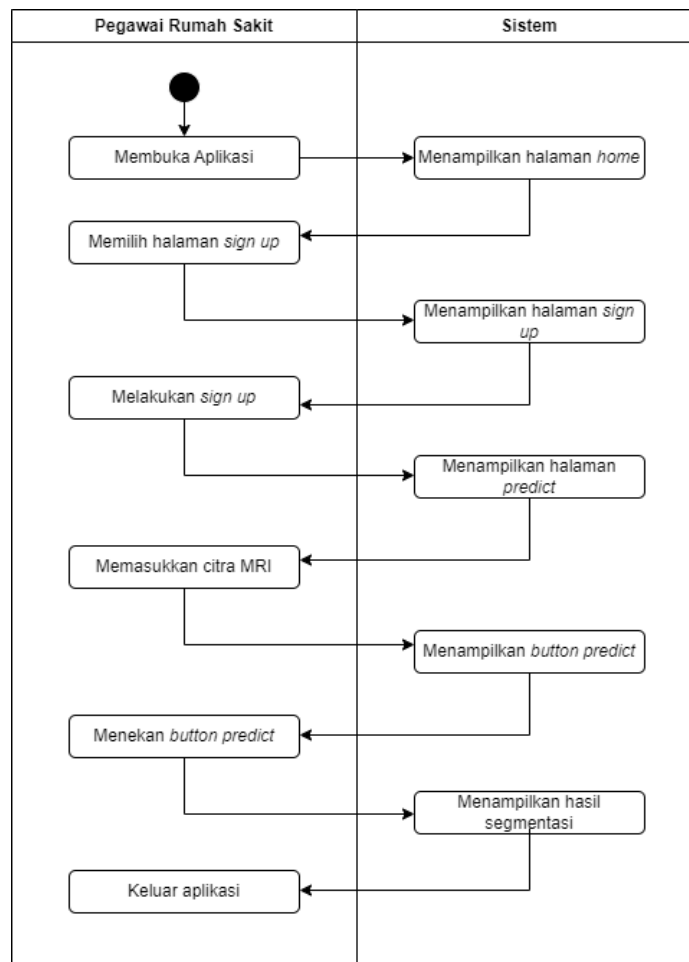
Berdasarkan pada gambar 3.2 *Use Case diagram* aplikasi merupakan hak yang dimiliki oleh sebuah pegawai rumah sakit pada saat menggunakan aplikasi. Pegawai rumah sakit dapat hak akses berupa *sign up* selanjutnya memasukkan citra berupa gambar dan kemudian mengetahui hasil segmentasi. Tahapan terakhir yaitu keluar pada tahap ini ketika pegawai rumah sakit ingin keluar dari aplikasi maka pegawai rumah sakit memiliki hak akses untuk keluar. Berikut merupakan tabel dari definisi *usecase* :

Tabel 3.3 Definisi Use Case

NO	Usecase	Deskripsi
1	<i>Sign Up</i>	Pada tahapan ini dimana pegawai rumah sakit melakukan pendaftaran dengan mengisi kolom <i>Name</i> , <i>NIP</i> , <i>Password</i> dan <i>Re-enter Password</i> agar dapat mengakses aplikasi.
2	<i>Login</i>	Pada tahapan ini dimana pegawai rumah sakit memasukan <i>NIP</i> dan <i>password</i> agar dapat mengakses

		aplikasi.
3	Memasukkan Citra MRI	Pada tahapan ini pegawai rumah sakit memasukkan citra MRI untuk mengetahui hasil segmentasi.
4	<i>Logout</i>	Pada tahapan ini pegawai rumah sakit dapat keluar jika gambar telah mengetahui hasil segmentasi.
5	Menampilkan Hasil Segementasi	Pada tahapan ini pegawai rumah sakit dapat melihat hasil dari pemrosesan sebuah segmentasi pada citra MRI.

3.3.6.2 Activity Diagram



Gambar 3.3 Activity Diagram

Berdasarkan pada gambar 3.3 menunjukkan alur penggunaan aplikasi yaitu dimulai dengan pegawai rumah sakit membuka aplikasi lalu sistem menampilkan halaman *home*. Selanjutnya pegawai rumah sakit melakukan *sign up* dan sistem

akan menampilkan halaman *predict*. Selanjutnya pegawai rumah sakit memasukkan citra MRI dan menekan *button predict* kemudian sistem akan menampilkan hasil segmentasi. Terakhir pegawai rumah sakit dapat keluar dari aplikasi.

3.3.7 Front-End Development

Pada tahap *front-end development* adalah proses membangun komponen yang berinteraksi dengan pengguna. Contohnya adalah antarmuka pengguna, tombol, data yang dimasukkan pengguna, situs *web*, dan fitur pengalaman pengguna (*User Experience*). Penulis membangun dan mengembangkan *user interface design* yang telah dibuat pada figma menjadi bagian dari *website* yang pengguna lihat pada tampilan sebuah *website*.

3.3.8 Back-End Development

Pada tahap *back-end development* merupakan proses yang memungkinkan situs *web*, aplikasi, atau program berfungsi. Selain itu, tahap ini termasuk menangani apa yang terjadi pada server dan *database* situs *web* atau aplikasi menggunakan model arsitektur MVC (*Model View Controller*).

3.3.9 Deployment

Deployment merupakan aktivitas membangun sebuah sistem perangkat lunak. Pada tahap *deployment* penulis melakukan pemindahan situs *web* dari *local environment* ke *live server*.

3.3.10 Testing

Pengujian atau *testing* yang dilakukan menggunakan metode *Blackbox Testing* dan *System Usability Scale*. Pengujian dilakukan untuk menemukan kesalahan-kesalahan yang mungkin terjadi sehingga dapat dilakukan perbaikan serta mengukur seberapa mudah pengguna dalam menggunakan aplikasi. Pengujian dilakukan dengan *blackbox testing* atau pengujian secara langsung pada *website*.

Hasil akhir dari pengujian adalah bahwa aplikasi telah bebas dari kesalahan atau *bug* sehingga diharapkan saat sampai ke tangan pengguna aplikasi sudah siap dan mudah dalam digunakan. Pengujian juga dilakukan pada citra yang di *rotate right* 90°, *rotate left* 90°, dan *rotate* 180°.

V. KESIMPULAN

5.1 Kesimpulan

Simpulan yang diperoleh berdasarkan penelitian yang telah dilakukan adalah sebagai berikut :

1. Pembuatan aplikasi *web* segmentasi jantung pada citra MRI dengan metode U-net telah berhasil dilakukan. Aplikasi sudah menggunakan *Convolutional Neural Network* sebagai algoritma dalam proses pemodelannya. Algoritma tersebut pada aplikasi juga sudah mampu menampilkan hasil prediksi dengan akurasi yang tinggi yaitu 0,99.
2. Hasil dari pengujian menggunakan *blackbox testing* dengan total 11 skenario pengujian didapatkan hasil bahwa semua kondisi pengujian yang dilakukan memiliki hasil pengujian yang sesuai dengan hasil yang diharapkan pada setiap pengujian. Semua skenario memiliki status sesuai harapan sehingga dapat disimpulkan aplikasi telah memenuhi kebutuhan pengguna.
3. Aplikasi *web* sudah dapat memenuhi fungsi utamanya berdasarkan pengalaman pengguna, ditunjukkan melalui pengujian menggunakan teknik *System Usability Scale* dengan 12 responden didapat nilai sebesar 71 yang masuk ke kategori *Acceptable* pada penilaian *Acceptability Range*, masuk ke kategori *Good* pada penilaian *Adjective Range*, dan masuk predikat C pada penilaian *Grade Scale*.

5.2 Saran

Saran yang diperoleh berdasarkan penelitian yang telah dilakukan adalah sebagai berikut :

1. Model yang dibuat untuk kasus segmentasi jantung sudah bagus, tetapi untuk hasil segmentasinya akan lebih bagus jika diberikan label untuk setiap objek segmentasi.
2. Untuk pengembangan tidak hanya gambaran potongan aksial, tetapi bisa dengan MRI potongan lain seperti potongan 3D dan dapat mengidentifikasi struktur jantung selain ventrikel kiri dan ventrikel kanan.

DAFTAR PUSTAKA

- [1] M. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk Mendeteksi Penyakit Jantung," *Fakt. Exacta*, vol. 7, no. 4, pp. 366–371, 2014.
- [2] Y. Yang, B. Lian, L. Li, C. Chen, and P. Li, "DBSCAN clustering algorithm applied to identify suspicious financial transactions," *Proc. - 2014 Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. CyberC 2014*, pp. 60–65, 2014, doi: 10.1109/CyberC.2014.89.
- [3] T. Naraloka, L. I. Kesuma, A. Sukmawati, and M. Cristianti, "Arsitektur U-Net pada Segmentasi Citra Hati sebagai Deteksi Dini Kanker Liver," *Techno.Com*, vol. 21, no. 4, pp. 753–764, 2022, doi: 10.33633/tc.v21i4.6669.
- [4] "erikycd/Heart_segmentation: Heart ventricle segmentation with U-Net." https://github.com/erikycd/Heart_segmentation (accessed May 13, 2023).
- [5] J. Dian, F. D. Silalahi, and N. D. Setiawan, "Sistem Monitoring Detak Jantung Untuk Mendeteksi Tingkat Kesehatan Jantung Berbasis Internet Of Things Menggunakan Android," *JUPITER (Jurnal Penelit. Ilmu dan Teknol. Komputer)*, vol. 13, no. 2, pp. 69–75, 2021, [Online]. Available: <https://jurnal.polsri.ac.id/index.php/jupiter/article/view/3669>
- [6] S. Notosiswoyo, Mulyono., Suswati, "Pemanfaatan Magnetic Resonance Imaging sebagai Sarana Diagnosa Pasien," *Stroke Pathophysiol. Diagnosis, Manag.*, vol. XIV, pp. 461–479, 2004.
- [7] H. Riyadli, A. Arliyana, and F. E. Saputra, "Rancang Bangun Sistem Informasi Keuangan Berbasis WEB," *J. Sains Komput. dan Teknol. Inf.*, vol. 3, no. 1, pp. 98–103, 2020, doi: 10.33084/jsakti.v3i1.1770.
- [8] Arsyad Achmadi, D. Junaedi, and E. Darwiyanto, "Rekomendasi User Interface Pada Website Dikti Menggunakan Metode Goal Directed

- Design,” *e-Proceeding Eng.*, vol. 4, no. 3, pp. 5063–5069, 2017.
- [9] M. B. Wiryawan, “User Experience (Ux) sebagai Bagian dari Pemikiran Desain dalam Pendidikan Tinggi Desain Komunikasi Visual,” *Humaniora*, vol. 2, no. 2, p. 1158, 2011, doi: 10.21512/humaniora.v2i2.3166.
- [10] D. Rolon-Mérette, M. Ross, T. Rolon-Mérette, and K. Church, “Introduction to Anaconda and Python: Installation and setup,” *Quant. Methods Psychol.*, vol. 16, no. 5, pp. S3–S11, 2020, doi: 10.20982/tqmp.16.5.s003.
- [11] Rully Pramudita, Rita Wahyuni Arifin, Ari Nurul Alfian, Nadya Safitri, and Shilka Dina Anwariya, “Penggunaan Aplikasi Figma Dalam Membangun Ui/Ux Yang Interaktif Pada Program Studi Teknik Informatika Stmik Tasikmalaya,” *J. Buana Pengabdian*, vol. 3, no. 1, pp. 149–154, 2021, doi: 10.36805/jurnalbuanapengabdian.v3i1.1542.
- [12] H. Kurniawan, B. Setiyono, and R. R. Isnanto, “Aplikasi Penjawab Pesan Singkat Otomatis Dengan Bahasa Python,” *Dr. Diss. Jur. Tek. Elektro Fak. Tek. Undip*, 2011.
- [13] M. Siahaan, C. H. Jasa, K. Anderson, and M. Valentino, “Penerapan Artificial Intelligence (AI) Terhadap Seorang Penyandang Disabilitas Tunanetra,” *Inf. Syst. Technol.*, vol. 01, no. 02, pp. 186–193, 2020.
- [14] B. Siswoyo, “MultiClass Decision Forest Machine Learning Artificial Intelligence,” *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 1–7, 2020, doi: 10.30871/jaic.v4i1.1155.
- [15] A. Roihan, P. A. Sunarya, and A. S. Rafika, “Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper,” *IJCIT (Indonesian J. Comput. Inf. Technol.)*, vol. 5, no. 1, pp. 75–82, 2020, doi: 10.31294/ijcit.v5i1.7951.
- [16] M. Rizki, S. Basuki, and Y. Azhar, “Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory(LSTM) Untuk Prediksi Curah Hujan Kota Malang,” *J. Repos.*, vol. 2, no. 3, pp. 331–338, 2020, doi: 10.22219/repositor.v2i3.470.
- [17] S. R. Dewi, “Deep Learning Object Detection Pada Video,” *Deep Learn. Object Detect. Pada Video Menggunakan Tensorflow Dan Convolutional*

Neural Netw., pp. 1–60, 2018.

- [18] R. D. Nurfita and G. Ariyanto, “Implementasi Deep Learning berbasis Tensorflow untuk Pengenalan Sidik Jari,” *Emit. J. Tek. Elektro*, vol. 18, no. 1, pp. 22–27, 2018, doi: 10.23917/emit.v18i01.6236.
- [19] K. P. Danukusumo, “Convolutional neural network untuk mendeteksi bangunan,” pp. 10–22, 2017, [Online]. Available: <http://e-journal.uajy.ac.id/12425/>
- [20] V. M. P. Salawazo, D. P. J. Gea, R. F. Gea, and F. Azmi, “Implementasi Metode Convolutional Neural Network (CNN) Pada Penegagalan Objek Video CCTV,” *J. Mantik Penusa*, vol. 3, no. 1, pp. 74–79, 2019.
- [21] F. Milletari, N. Navab, and S. A. Ahmadi, “V-Net: Fully convolutional neural networks for volumetric medical image segmentation,” *Proc. - 2016 4th Int. Conf. 3D Vision, 3DV 2016*, pp. 565–571, 2016, doi: 10.1109/3DV.2016.79.
- [22] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017, doi: 10.1109/TPAMI.2016.2644615.
- [23] W. Chen, B. Yang, J. Li, and J. Wang, “An approach to detecting diabetic retinopathy based on integrated shallow convolutional neural networks,” *IEEE Access*, vol. 8, pp. 178552–178562, 2020, doi: 10.1109/ACCESS.2020.3027794.
- [24] M. M. Islam, T. N. Poly, B. A. Walther, H. C. Yang, and Y. C. Li, “Artificial intelligence in ophthalmology: A meta-analysis of deep learning models for retinal vessels segmentation,” *J. Clin. Med.*, vol. 9, no. 4, 2020, doi: 10.3390/jcm9041018.
- [25] J. Shi *et al.*, “Improvement of damage segmentation based on pixel-level data balance using vgg-unet,” *Appl. Sci.*, vol. 11, no. 2, pp. 1–17, 2021, doi: 10.3390/app11020518.
- [26] M. Gan and C. Wang, “Dual-Stage U-Shape Convolutional Network for Esophageal Tissue Segmentation in OCT Images,” *IEEE Access*, vol. 8, no. 6, pp. 215020–215032, 2020, doi: 10.1109/ACCESS.2020.3041767.

- [27] Y. N. Nabusa, "Pengolahan Citra Digital Perbandingan Metode Histogram Equalization Dan Spesifikasi Pada Citra Abu-Abu," *J-Icon*, vol. 7, no. 1, pp. 87–95, 2019.
- [28] N. Nafi'iyah, "Algoritma Kohonen dalam Mengubah Citra Graylevel Menjadi Citra Biner," *J. Ilm. Teknol. Inf. Asia*, vol. 9, no. 2, pp. 49–55, 2015.
- [29] S. Sujito and M. Yunus, "Perbandingan Strategi Pelabelan Objek Pada Citra Digital Dengan Metode Flood Filling," *J. Teknol. Inf. Teor. Konsep, dan Implementasi*, vol. 7, no. 2, pp. 139–148, 2016.
- [30] I. P. E. Sutariawan, G. R. Dantes, and K. Y. Ernanda Aryanto, "Segmentasi Mata Katarak Pada Citra Medis Menggunakan Metode Operasi Morfologi," *J. Ilmu Komput. Indones.*, vol. 3, no. 1, pp. 23–31, 2019, doi: 10.23887/jik.v3i1.2750.
- [31] M. T. Stefanus Christian Adi Pradhana, Untari Novia Wisesty S.T.,M.T., Febryanthi Sthevanie S.T., "Pengenalan Aksara Jawa dengan Menggunakan Metode Convolutional Neural Network," *e-Proceeding Eng.*, vol. 7, no. 1, pp. 2558–2567, 2020.
- [32] F. Azimah and K. Rizky Nova Wardani, "Sistem Pendeteksi Gejala Awal Covid-19 dengan Penggunaan Metode Al Project Cycle," *J. Locus Penelit. dan Pengabd.*, vol. 1, no. 6, pp. 405–418, 2022, doi: 10.58344/locus.v1i6.135.
- [33] L. Siregar, "Review Pengujian Keamanan Perangkat Lunak dalam Software Development Life Cycle (SDLC)," *J. Appl. Sci. Electr. Eng. Comput. Technol.*, vol. 1, no. 3, pp. 1–11, 2020, doi: 10.30871/aseect.v1i3.2380.
- [34] I. G. S. Widharma, "Perancangan Simulasi Sistem Pendaftaran Kursus Berbasis Web Dengan Metode Sdlc," *Matrix J. Manaj. Teknol. dan Inform.*, vol. 7, no. 2, p. 38, 2017, doi: 10.31940/matrix.v7i2.527.
- [35] Aceng Abdul Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp. 1–5, 2020.
- [36] J. A. Falaq, R. Tulloh, and M. Iqbal, "Implementasi Jaringan Hotspot Berbayar Berbasis Voucher Menggunakan Platform Google Cloud

- Platform,” vol. 7, no. 4, pp. 861–876, 2021.
- [37] S. Supriyono, “Software Testing with the approach of Blackbox Testing on the Academic Information System,” *IJISTECH (International J. Inf. Syst. Technol.)*, vol. 3, no. 2, pp. 227–233, 2020.
- [38] U. Meminimalisir and P. Covid, “PENGEMBANGAN MONITORING DENGAN MENGGUNAKAN KAMERA AMG8833 BERBASIS IoT,” vol. 11, no. 2, 2023.
- [39] S. Di, S. Royal, and S. Royal, “PENERAPAN SYSTEM USABILITY SCALE (SUS) DALAM PENGUKURAN KEBERGUNAAN WEBSITE PROGRAM,” vol. 4307, no. 1, pp. 43–49, 2022.
- [40] Z. Sharfina and H. B. Santoso, “An Indonesian adaptation of the System Usability Scale (SUS),” *2016 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACISIS 2016*, pp. 145–148, 2017, doi: 10.1109/ICACISIS.2016.7872776.
- [41] Miftahul Reski Putra Nasjum, “Implementasi Deep Learning Dalam Mendeteksi Penyakit Menggunakan Convolutional Neural Network Dan Tensorflow,” *Kaos GL Derg.*, vol. 8, no. 75, pp. 147–154, 2020.
- [42] J. A. ENGKA, “Penerapan Machine Learning Dalam Sistem Klasifikasi Penyakit Manusia Dengan Model Decision Tree Dan Neural Network,” *Penerapan Mach. Learn. Dalam Sist. Klasifikasi Penyakit Mns. Dengan Model Decis. Tree Dan Neural Netw.*, vol. 1, no. 69, pp. 5–24, 1967.
- [43] S. Niyas, S. J. Pawan, M. Anand Kumar, and J. Rajan, “Medical image segmentation with 3D convolutional neural networks: A survey,” *Neurocomputing*, vol. 493, pp. 397–413, 2022, doi: 10.1016/j.neucom.2022.04.065.
- [44] W. Weng and X. Zhu, “INet: Convolutional Networks for Biomedical Image Segmentation,” *IEEE Access*, vol. 9, pp. 16591–16603, 2021, doi: 10.1109/ACCESS.2021.3053408.