

MEMBANGUN APLIKASI DENGAN MEMANFAATKAN QR CODE BERBASIS ANDROID

MEMBANGUN APLIKASI DENGAN MEMANFAATKAN QR CODE BERBASIS ANDROID

**Roni Habibi, S.Kom., M.T.
Dinda Anik Masruro
Nuha Hanifatul Khonsa'**
Informatics Research Center



Kreatif Industri Nusantara

Penulis:

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

Editor:

M. Yusril Helmi Setyawan

Penyunting:

Syafrial Fachrie Pane

Khaera Tunnisia

Diana Asri Wijayanti

Desain sampul dan Tata letak:

Deza Martha Akbar

Penerbit:

Kreatif Industri Nusantara

Redaksi:

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

Distributor:

Informatics Research Center

Jl. Sariasisih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara
apapun tanpa ijin tertulis dari penerbit

*'Jika Kamu tidak dapat
menahan lelahnya
belajar, Maka kamu harus
sanggup menahan
perihnya Kebodohan.'*

Imam Syafi'i

CONTRIBUTORS

RONI HABIBI, S.KOM., M.T., DINDA ANIK MASRURO, NUHA HANIFATUL KHONSA',
Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia

CONTENTS IN BRIEF

1 Pengenalan QR Code, Android, dan Android Studio	1
2 Landasan Teori	23
3 Analisis dan Perancangan Sistem	51
4 Membangun Aplikasi	81
5 Implementasi	97
6 MFCC dan Neural Network	131
7 CNN	145

DAFTAR ISI

Daftar Gambar	xiii
Daftar Tabel	xix
Foreword	xxiii
Kata Pengantar	xxv
Acknowledgments	xxvii
Acronyms	xxix
Glossary	xxxii
List of Symbols	xxxiii
Introduction	xxxv
<i>Roni Habibi, S.Kom., M.T., Dinda Anik Masruro Nuha Hanifatul Khonsa'.</i>	
1 Pengenalan QR Code, Android, dan Android Studio	1
1.1 QR CODE	1
1.1.1 Sejarah QR Code	1
1.1.2 Langkah-langkah dalam pembuatan QR Code	4
1.1.3 Penerapan QR Code	5

1.1.4	QR Code dalam Berbagai Aspek Sehari Hari	6
1.1.5	QR Code dalam Industry	6
1.2	Android	6
1.2.1	Definisi Android	6
1.2.2	Sejarah Android	7
1.2.3	Versi Android	7
1.3	Android Studio	10
1.3.1	Definisi Android Studio	10
1.3.2	Sejarah Android Studio	10
1.3.3	Instalasi Android Studio	12
2	Landasan Teori	23
2.1	Tinjauan Pustaka	23
2.2	Teori Umum	24
2.2.1	Database	24
2.2.2	SQLITE	26
2.2.3	Penggunaan Bahasa Pemrograman JAVA	28
2.2.4	Penggunaan Bahasa Pemrograman PHP	30
2.2.5	MySQL	32
2.2.6	Algoritma	37
2.2.7	Analisi Sistem Berjalan	38
2.2.8	Analisis Sistem Yang akan Dibangun	41
3	Analisis dan Perancangan Sistem	51
3.1	Metode Waterfall	51
3.2	Analisis Sistem	53
3.2.1	Sistem yang sedang Berjalan	54
3.2.2	Sistem Yang Akan Dibangun	56
3.3	Perancangan	60
3.3.1	Use Case	60
3.3.2	Perancangan Antarmuka	75
4	Membangun Aplikasi	81
4.1	Memulai Project Baru	81
4.2	Error Yang Sering Terjadi Dalam Membuat Project Pada Android Studio	85
4.3	Issues #71	95
5	Implementasi	97

5.1	Pembuatan Aplikasi <i>QR CODE</i> Berbasis Android	97
5.1.1	Lingkungan Implementasi	97
5.1.2	Lingkungan Perangkat Keras	97
5.1.3	Lingkungan Perangkat Lunak	98
5.1.4	Aplikasi Scanner Inventory Barang	98
5.1.5	Implementasi Tampilan Utama	127
6	MFCC dan Neural Network	131
6.1	Teori	131
6.2	Praktek Program	134
6.2.1	GTZAN Genre Collection dan data dari freesound	134
6.2.2	Fungsi Display MFCC	136
6.2.3	Fungsi Extract Features Song	136
6.2.4	Fungsi Generate Features And Labels	137
6.2.5	Penggunaan Fungsi Generate Features And Labels Sangat Lama Ketika Meload Dataset Genre	138
6.2.6	Pemisahan Data Training Dan Data Set Sebesar 80%	139
6.2.7	Fungsi Sequential	140
6.2.8	Fungsi Compile	141
6.2.9	Fungsi Fit	142
6.2.10	Fungsi Evaluate	142
6.2.11	Fungsi Predict	143
6.3	Penanganan Error	144
6.3.1	Module Eror	144
7	CNN	145
7.1	Teori	145
7.1.1	Teks Tokenizer	145
7.1.2	kONSEP dasar K Fold Cross Validation pada dataset komentar Youtube	145
7.1.3	kode program for train, test in splits	146
7.1.4	Jelaskan apa maksudnya kode program <i>train_content = d['CONTENT'].iloc[train_idx]</i> dan <i>test_content = d['CONTENT'].iloc[test_idx]</i> . dilengkapi dengan ilustrasi atau gambar	146

7.1.5	Soal No. 5 Jelaskan apa maksud dari fungsi <i>tokenizer = Tokenizer(num_words=2000)</i> dan <i>tokenizer.fit_on_texts(train_content)</i> , dilengkapi dengan ilustrasi atau gambar	146
7.1.6	Jelaskan apa maksud dari fungsi <i>d_train_inputs =</i> <i>tokenizer.texts_to_matrix(train_content, mode='tfidf')</i> dan <i>d_test_inputs = tokenizer.texts_to_matrix(test_content,</i> <i>mode='tfidf')</i> , dilengkapi dengan ilustrasi kode dan atau gambar	147
7.1.7	Praktek	150
7.1.8	Penanganan Error	165
	Daftar Pustaka	167

DAFTAR GAMBAR

1.1	Contoh QR Code	3
1.2	Download Instalasi Android Studio	12
1.3	Tampilan Awal Saat Instalasi	13
1.4	Memilih komponen yang akan diinstal	13
1.5	Memilih lokasi file	14
1.6	Memilih lokasi file	14
1.7	Memilih lokasi file	15
1.8	Tampilan Awal pada Android Studio	16
1.9	Tampilan Awal pada Android Studio	16
1.10	Tampilan Awal pada Android Studio	17
1.11	Tampilan Awal pada Android Studio	17
1.12	Tampilan Awal pada Android Studio	18
1.13	Tampilan Awal pada Android Studio	18

1.14	Tampilan Awal pada Android Studio	19
1.15	Tampilan Awal pada Android Studio	19
1.16	Tampilan File project dalam tampilan Android Studio	21
1.17	Tampilan Jendela utama Android Studio	22
2.1	Contoh kode PHP	31
2.2	Contoh kode PHP untuk perulangan	31
2.3	Contoh hasil dari kode PHP untuk perulangan	32
2.4	Logo MySql	32
2.5	Penemu MySql	33
2.6	Simbol Flowmap	39
2.7	Tambilan Awal Visio	40
2.8	Simbol Usecase	43
2.9	Simbol Class Diagram	44
2.10	Simbol State Diagram	44
2.11	Simbol Sequence Diagram	45
2.12	Simbol Collaboration Diagram	46
2.13	Simbol Activity Diagram	46
2.14	Simbol Component Diagram	47
2.15	Simbol Deployment Diagram	47
2.16	Tampilan dari Power Desainer	48
2.17	Tampilan dari Enterprise Architecture (EA)	49
3.1	Tahap Metode Waterfall	52
3.2	Flowmap Sistem yang Sedang Berjalan	55
3.3	<i>Flowmap Sistem Yang Akan Dibangun</i>	57
3.4	<i>Use Case Diagram</i>	60
3.5	<i>Sequence Diagram: Login</i>	61
3.6	<i>Sequence Diagram:Scan</i>	62
3.7	<i>Sequence Diagram:Melihat Data</i>	63

3.8	<i>Sequence Diagram: Mengedit</i>	64
3.9	<i>Activity Diagram: Login</i>	65
3.10	<i>Activity Diagram: Scan</i>	66
3.11	<i>Activity Diagram: Melihat Data</i>	67
3.12	<i>Activity Diagram: Mengedit</i>	68
3.13	<i>State Chart Diagram: Login</i>	69
3.14	<i>State Chart Diagram: Scan</i>	70
3.15	<i>State Chart Diagram: Melihat Data Barang</i>	70
3.16	<i>State Chart Diagram: Mengedit Data</i>	71
3.17	<i>Class Diagram</i>	72
3.18	<i>Component Diagram</i>	73
3.19	<i>Deployment Diagram</i>	74
3.20	<i>Antarmuka Pengguna : Login</i>	76
3.21	<i>Antarmuka Pengguna : Registrasi</i>	77
3.22	<i>Antarmuka Pengguna : Halaman Utama Operator</i>	78
3.23	<i>Antarmuka Pengguna : Halaman Utama Admin</i>	79
4.1	<i>Tampilan Awal Android Studio</i>	82
4.2	<i>Memilih Project</i>	82
4.3	<i>Konfigurasikan project baru Anda dengan beberapa setelan.</i>	83
5.1	<i>Membuat Project Baru</i>	98
5.2	<i>Codingan Pada Build Gradle</i>	99
5.3	<i>Codingan Pada Build Gradle</i>	99
5.4	<i>Codingan Pada MainActiviyt.java</i>	101
5.5	<i>Codingan Pada MainActiviyt.java</i>	101
5.6	<i>Codingan Pada MainActiviyt.java</i>	101
5.7	<i>Codingan Pada MainActiviyt.java</i>	102
5.8	<i>Codingan Pada MainActiviyt.java</i>	102
5.9	<i>Codingan Pada MainActiviyt.java</i>	102

5.10	<i>Codingan Pada ActivityMain</i>	108
5.11	<i>Codingan Pada ActivityMain</i>	108
5.12	<i>Codingan Pada LoginActivity.java</i>	110
5.13	<i>Codingan Pada LoginActivity.java</i>	110
5.14	<i>Codingan Pada LoginActivity.java</i>	111
5.15	<i>Codingan Pada RegisterActivity.java</i>	113
5.16	<i>Codingan Pada RegisterActivity.java</i>	113
5.17	<i>Codingan Pada RegisterActivity.java</i>	113
5.18	<i>Codingan Pada ActivityLogin.xml</i>	116
5.19	<i>Codingan Pada ActivityLogin.xml</i>	116
5.20	<i>Codingan Pada ActivityLogin.xml</i>	116
5.21	<i>Codingan Pada ActivityLogin.xml</i>	117
5.22	<i>Codingan Pada ActivityRegister.xml</i>	119
5.23	<i>Codingan Pada ActivityRegister.xml</i>	119
5.24	<i>Codingan Pada ActivityRegister.xml</i>	120
5.25	<i>Codingan Pada ActivityRegister.xml</i>	120
5.26	<i>Codingan Pada Database</i>	123
5.27	<i>Codingan Pada Database</i>	123
5.28	<i>Codingan Pada Database</i>	124
5.29	<i>Tampilan Menu Login</i>	127
5.30	<i>Tampilan Menu Register</i>	128
5.31	<i>Tampilan Halaman Utama</i>	129
5.32	<i>Tampilan Halaman Scan QR</i>	130
6.1	Contoh Pembobotan Neural Network Tasya	132
6.2	Cara Membaca Hasil Plot MFCC Tasya	133
6.3	One Hot Encoding Tasya	133
6.4	Numpy Unique Tasya	134
6.5	To Categorical Tasya	134

6.6	Sequential Tasya	134
6.7	Meload Data Genre Collection Tasya	136
6.8	Display MFCC Tasya	136
6.9	Hasil Display MFCC Tasya	137
6.10	Extract Features Tasya	137
6.11	Fungsi Generate Features And Labels Tasya	138
6.12	Hasil Fungsi Generate Features And Labels Tasya	139
6.13	Pemisahan Data Training dan Data Set Tasya	140
6.14	Pemisahan Data Training dan Data Set Tasya	140
6.15	Pemisahan Data Training dan Data Set Tasya	141
6.16	Fungsi Compile Tasya	141
6.17	Fungsi Fit Tasya	142
6.18	Fungsi Evaluate Tasya	143
6.19	Fungsi Evaluate Tasya	143
6.20	Fungsi Predict Tasya	143
6.21	Module Error Tasya	144
6.22	Penyelesaian Module Error Tasya	144
7.1	Ilustrasi KFold Cross Tasya	146
7.2	Ilustrasi Text To Matrix Tasya	147
7.3	Ilustrasi np Absolute Tasya	147
7.4	Ilustrasi One Hot Encoding Tasya	148
7.5	Ilustrasi Neural Network Pemodelan Tasya	148
7.6	Algoritma Konvolusi Tasya	149
7.7	Algoritma Konvolusi Tasya	150
7.8	Algoritma Konvolusi Tasya	150
7.9	Algoritma Konvolusi Tasya	151
7.10	Algoritma Konvolusi Tasya	151
7.11	Algoritma Konvolusi Tasya	151

7.12	Algoritma Konvolusi Tasya	152
7.13	Kode Program Blok In 1 Tasya	152
7.14	Kode Program Blok In 2 Tasya	153
7.15	Kode Program Blok In 3 Tasya	154
7.16	Kode Program Blok In 4 Tasya	154
7.17	Kode Program Blok In 5 Tasya	155
7.18	Kode Program Blok In 6 Tasya	155
7.19	Kode Program Blok In 7 Tasya	155
7.20	Kode Program Blok In 8 Tasya	156
7.21	Kode Program Blok In 9 Tasya	157
7.22	Kode Program Blok In 10 Tasya	158
7.23	Kode Program Blok In 11 Tasya	158
7.24	Kode Program Blok In 12 Tasya	159
7.25	Kode Program Blok In 13 Tasya	161
7.26	Kode Program Blok In 14 Tasya	162
7.27	Kode Program Blok In 15 Tasya	162
7.28	Kode Program Blok In 16 Tasya	163
7.29	Kode Program Blok In 17 Tasya	163
7.30	Kode Program Blok In 18 Tasya	163
7.31	Kode Program Blok In 19 Tasya	164
7.32	Kode Program Blok In 20 Tasya	165
7.33	Error Tasya	165
7.34	Penanganan Error Kernel Tasya	166

DAFTAR TABEL

3.1	Deskripsi perangkat <i>User</i>	58
3.2	Deskripsi perangkat Lunak <i>User</i>	59

Listings

6.1	Kode Load Data Untuk MFCC	135
6.2	Code Fungsi Display MFCC	136
6.3	Panggil Genenrate Labels	138
6.4	Code Pemisahan Data Training Dan Testing	139
6.5	Code Fungsi Sequential	140
6.6	Code Fungsi Compile	141
6.7	Code Fungsi Fit	142
6.8	Code Fungsi Evaluate	142
6.9	Code Fungsi Predict	143
7.1	K Fold Cross Validation	145
7.2	Membuat model Neural Network	147
7.3	Compile model	149
src/Chapter7/1164086/in1.py		150
src/Chapter7/1164086/in2.py		152
src/Chapter7/1164086/in3.py		153
src/Chapter7/1164086/in4.py		154
src/Chapter7/1164086/in5.py		154
src/Chapter7/1164086/in6.py		155

src/Chapter7/1164086/in7.py	155
src/Chapter7/1164086/in8.py	156
src/Chapter7/1164086/in9.py	156
src/Chapter7/1164086/in10.py	157
src/Chapter7/1164086/in11.py	158
src/Chapter7/1164086/in12.py	158
src/Chapter7/1164086/in13.py	159
src/Chapter7/1164086/in14.py	161
src/Chapter7/1164086/in15.py	162
src/Chapter7/1164086/in16.py	162
src/Chapter7/1164086/in17.py	163
src/Chapter7/1164086/in18.py	163
src/Chapter7/1164086/in19.py	164
src/Chapter7/1164086/in19.py	164

FOREWORD

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa

KATA PENGANTAR

Alhamdulillah segala puji dan syukur kehadirat Allah SWT karena dengan rahmat dan karunia-Nya penulis dapat membuat buku sampai saat ini, Dan shalawat serta salam semoga tercurah kepada junjungan Nabi Akhir Zaman, Muhammad SAW.

Buku ini diciptakan untuk menjadi bahan rujuk bagi para pembaca untuk membuat aplikasi prediksi, dan semoga dapat bermanfaat bagi para pembacanya.

Assalamu'alaikum Warahmatullahi Wabarakatuh

RONI HABIBI , DINDA ANIK MASRURO, NUHA HANIFATUL KHONSA'

*Bandung, Jawa Barat
Januari, 2020*

ACKNOWLEDGMENTS

Terima kasih atas semua masukan dari dosen pembimbing,keluarga dan teman-teman agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Proyek II.

Dinda Anik Masruro Nuha Hanifatul Khonsa'

ACRONYMS

AI	Artificial Intelligence
ETL	Extract Transform Load
NLP	Natural Language Processing

GLOSSARY

cybernetics	Adalah sistem yang berinteraksi langsung dengan diri sendiri yang memahami dan menentukan proses tujuan.
Heuristik	Adalah sebuah metode yang mengembangkan efisiensi dalam proses pencarian.
Supervised	Adalah sebuah tugas pengumpulan data untuk menyimpulkan fungsi dari data pelatihan berlabel.
Unsupervised	Adalah Tidak adanya memiliki data latih, sehingga dari data yang ada kita mengelompokan data tersebut menjadi 2 ataupun 3 bagian.

SYMBOLS

A Amplitude

$\&$ Propositional logic symbol

a Filter Coefficient

B Number of Beats

INTRODUCTION

RONI HABIBI, S.KOM., M.T., DINDA ANIK MASRURO NUHA HANIFATUL KHONSA'.

Informatics Research Center
Bandung, Jawa Barat, Indonesia

Indonesia saat ini telah memasuki era industri 4.0 atau revolusi 4.0 dimana otomatisasi sistem produksi dengan memanfaatkan teknologi dan big data. sehingga harus ada pengolahan data yang dapat menunjang keberhasilan perusahaan di era 4.0 , dengan melakukan Prediksi untuk jangka pendek, menegah, maupun jangka panjang sangat berpengaruh dalam pengambilan keputusan kedepannya , dengan begitu di dalam buku ini menjelaskan prediksi pada penggunaan kebutuhan .

BAB 1

PENGENALAN QR CODE, ANDROID, DAN ANDROID STUDIO

1.1 QR CODE

QR merupakan kepanjangan dari *Quick Respons* yaitu kode batang dua dimensi yang diciptakan pada tahun 1994 oleh Denso, salah satu perusahaan besar di grup Toyota, dan disetujui sebagai standar internasional ISO (ISO / IEC18004) pada bulan Juni 2000. Kode batang dua dimensi ini awalnya dimaksudkan untuk digunakan dalam pengendalian produksi komponen otomotif, tetapi telah menyebar luas di bidang lain (Soon, 2008:60). Penyimpanan data yang dapat ditampung pada QR Code lebih banyak dibandingkan pada barcode yang terbatas satu dimensi saja.

1.1.1 Sejarah QR Code

Teknologi informasi memberikan pengaruh yang sangat besar pada kehidupan manusia pada setiap bidang kehidupan. Contohnya adalah pada bidang perindustrian, Pendidikan, komersial dan lain lainnya yang dengan diterapkannya teknologi informasi terbukti meningkatkan efektivitas dan efisiensi perusahaan. Salah satu teknologi yang cukup dikenal adalah barcode.

Tetapi yang terjadi pada masa ini, penggunaan barcode sudah sangat lazim di industri di seluruh dunia. Hal ini memiliki tujuan untuk memudahkan pelaku industri dalam mengelola inventori yang mereka miliki, karena barcode ini menyimpan data spesifik seperti kode produksi, nomor identitas, dan lain-lain sehingga sistem komputer dapat memahami informasi yang telah dikodekan dalam barcode dengan mudah yang tentunya berbanding lurus dengan perkembangan teknologi yang begitu pesatnya, sehingga memberikan dampak pada penggunaan barcode yang kini mulai digantikan dengan QR Code.

Munir dan Pasca (2011) telah melakukan sebuah penelitian untuk mengembangkan QR Code dari data berupa image. Hasil penelitian menunjukkan jika QR Code dapat menyimpan data yang berupa image/gambar tetapi dengan kapasitas kecil sehingga tidak efisien di dunia nyata. Ariadi (2011) menjelaskan tentang bagaimana analisis dan perancangan sebuah QR Codemenggunakan PHP (PHP: Hypertext Preprocessing). QR Code juga memerlukan mekanisme untuk mengecek kesalahan sehingga diperlukan algoritma error correction. Beberapa macam algoritma error correction diantaranya yaitu Hamming code, Bose-Choundhuri-Hocquenghem code dan Reed-Solomon Code. Algoritma error correction yang sering digunakan pada QR Code adalah algoritma Reed-Solomon Code. QR Code yang dilengkapi errorcorrection akan tahan terhadap kerusakan data hingga batas tertentu sehingga akan lebih aman.

Error correction bisa dikembangkan dengan lapangan berhingga, seperti penerapan GF(256) sebagai dasar perhitungan pada algoritma Reed-Solomon Code. Lapangan berhingga yaitu suatu lapangan yang memuat elemen sebanyak berhingga (Herstein, 1996:221). GF(256) adalah lapangan berhingga dengan jumlah elemen sebanyak 256 yaitu 0-255, tetapi karena elemen 0 tidak digunakan sehingga hanya terdapat 255 karakter pada GF(256).

Perkembangan jaman digital telah membuat QR Code semakin banyak dimanfaatkan di banyak industri dan kehidupan. Cara membuat kode batang QR Code yang mudah dan dapat diperoleh dengan gratis membuat penggunaannya semakin diminati oleh berbagai kalangan. Setiap ponsel pintar atau komputer yang memiliki fitur kamera sudah dapat digunakan sebagai mesin pembaca QR Code. Berdasarkan uraian diatas maka perlu dilakukan penelitian untuk mengembangkan QR Code yang dilengkapi dengan algoritma Reed-Solomon Code sebagai error correction. Proses encode error correction dengan algoritma Reed-Solomon Code akan menggunakan penerapan lapangan berhingga.



Gambar 1.1 Contoh QR Code

Fungsi utama dari QR Code sendiri yaitu kode yang dapat dengan mudah dibaca oleh pemindai QR Code dan memiliki respons cepat, yang sesuai dengan tujuannya adalah untuk menyampaikan informasi dengan cepat dan mendapatkan respons yang cepat pula. Berbeda dengan kode batang, yang hanya dapat menyimpan informasi secara horizontal atau secara searah sedangkan kode QR mampu menyimpan informasi secara horizontal dan vertical yaitu dari 2 macam arah, oleh karena itu secara otomatis Kode QR dapat menampung informasi yang lebih banyak daripada kode batang dengan kemampuan penyimpanan yang berbeda.

Perusahaan yang paling pertama memperkenalkan QR Code yaitu Denso Wave sebuah perusahaan komponen otomotif di Jepang yang berkantor pusat di Kariya, Prefektur Aichi, Jepang. QR code memiliki kecanggihan yang tinggi seperti bisa membuat data secara vertikal dan horizontal, mampu menampung data lebih banyak dibanding Barcode, untuk cara kerjanya pun sangat mudah hanya diperlukan alat pemindai (QR Code Reader) untuk membaca data yang ada di QR Code.

Sementara di Indonesia, QR Code pertama kali diterapkan pada surat kabar Kompas, yang diterbitkan oleh Kelompok Kompas Gramedia . Hal ini dapat dilihat pada surat kabar Kompas dimana terdapat QR Code yang merepresentasikan artikel dari surat kabar tersebut.

1.1.2 Langkah-langkah dalam pembuatan QR Code

Langkah-langkah pembuatan QR Code meliputi penentuan kapasitas data yang akan dikodekan dan encoded data. Langkah-langkah tersebut akan dijelaskan sebagai berikut.

1. Menentukan Kapasitas

Kapasitas dari QR Code ditentukan oleh versi, tingkat error correction dan tipe data yang akan dikodekan (misalnya numerik, alfanumerik, dan lain-lain). Langkah pertama yang perlu diperhatikan yaitu mempertimbangkan tingkat koreksi kesalahan, kemudian tentukan versi yang akan digunakan (Swetake, 2011:1).

2. Encode (Pengkodean) Data

Pada bagian ini, pengkodean data menjadi sebuah QR Code dilakukan melalui serangkaian perhitungan sebagai berikut.

(a) Menentukan Tipe Data

Data akan dibaca tipe datanya terlebih dahulu. Masing-masing tipe data akan disimpan ke dalam representasi bilangan biner 4 bit dan mempunyai panjang karakter penyimpanan tertentu.

(b) Mengubah Data ke Dalam Bentuk Biner

Data yang telah diketahui tipe datanya akan diubah ke dalam biner. Pada data ‘buku saya’ mempunyai tipe data alfanumerik sehingga akan dikonversi ke dalam 9 bit biner.

(c) Mengubah Biner ke Dalam Bentuk Desimal

Data yang sudah diubah ke dalam bentuk biner, akan diubah ke dalam bentuk desimal berdasarkan kapasitas dari masing-masing versi QR Code yang

telah ditentukan. Data biner ‘buku saya’ ke dalam desimal yaitu 32 74 13 118 89 192 242 20 236 17 236 17 236. Data tersebut adalah data hasil representasi biner sebelum dilakukan perhitungan error correction.

(d) Error Correction

Coding Message polynomial yang telah diperoleh pada proses konversi data ke dalam bentuk biner kemudian akan dilakukan perhitungan error correction.

(e) Alokasi Data

Data hasil encode yang berupa final message akan dialokasikan ke dalam bentuk gambar QR Code. Data yang akan dialokasikan adalah data hasil representasi biner dan data hasil perhitungan koreksi kesalahan.

(f) Penentuan Pola Data

Pola data akan menentukan banyaknya pinalti pada QR Code. Pinalti adalah banyaknya modul error yang dapat dikembalikan. Pinalti yang sedikit akan membuat pembacaan QR Code oleh mesin pemindai menjadi lebih mudah. Penentuan pola data dilakukan dengan mencoba semua pola data yang ada kemudian dipilih pola data yang menghasilkan pinalti minimum.

(g) Penentuan Format Informasi

Data Format informasi terdiri dari tingkat error correction dan indikator pembentuk pola sebanyak 15 bit. Format informasi terdiri dari 2 bit untuk error correction, 3 bit untuk pembentuk pola dan 10 bit untuk format informasi data.

1.1.3 Penerapan QR Code

QR Code dapat diakses pada ponsel yang memiliki aplikasi pembaca QR Code dan memiliki akses internet baik WiFi, 4G maupun 3G yang berfungsi sebagai penghubung antara ponsel dengan situs yang dituju melalui QR Code tersebut. Pengguna ponsel dapat mengaktifkan aplikasi pembaca QR Code lalu mengarahkan kamera ke QR Code yang ingin dibaca, dan tak lupa pastikan bahwa jaringan internet ada pada ponsel, selanjutnya program pembaca kode QR akan secara otomatis memindai data yang tertera pada QR Code. Jika QR Code tersebut berisikan alamat dari situs web dan sebagainya, maka pengguna dapat langsung mengakses situs tersebut tanpa harus mengetikkan alamat dari situs yang akan dituju.

Lain halnya jika ingin mengakses Barcode dengan ponsel tanpa kamera, maka hal pertama yang harus dilakukan oleh pengguna ialah dengan menjalankan aplikasi penelusuran yang ada pada ponsel, lalu masukkan URL halaman yang bersangkutan, selanjutnya masukkan “ID” atau 7 digit nomor yang tertera di bawah kode dan klik tombol Yes, maka pengguna akan memperoleh konten digital yang diinginkan. Hal ini tentunya mempermudah pelanggan dalam mendapatkan sebuah informasi.

Ada beberapa jenis aplikasi yang dapat membaca QR Code antara lain QR Code Reader by Scan. Inc, QR Droid Code Scanner by DroidLa, QR Code Scan Barcode

Scanner by pickwick santa, QR Code Reader by TWMobile, QR Barcode Scanner by Gamma Play.com semua aplikasi ini dapat di download melalui Google Play Store.

1.1.4 QR Code dalam Berbagai Aspek Sehari Hari

QR Code dalam Pendidikan, Dalam Pendidikan QR Code dapat dimanfaatkan dalam menyimpan informasi tertentu. Hal ini dapat dilihat dengan adanya QR Code dalam buku belajar mengajar disekolah, dimana QR Code dapat diakses dan menampilkan materi yang terkait dengan pelajaran yang sedang dibahas. Selain itu dapat dicontohkan yang lain ialah QR Code Presensi Seperti pada pengerjaan Projek 2 ini terdapat QR Code Presensi Bimbingan yang dapat di generate tiap minggu sesuai kebutuhan bimbingan masing masing Mahasiswa. Selain itu adapula QR Code Presensi dimana QR code ini dapat di akses tiap mahasiswa dan Dosen sehingga dapat diakses mengenai kehadiran Dosen yang akan mengampu mata kuliah yang bersangkutan pada hari tersebut.

QR Code dalam Keseharian juga dapat ditemukan contoh pada aplikasi Line, dimana kita dapat membagi akun line dengan hanya mengirimkan QR Code pada teman kita. QR Code tersebut akan dibaca informasi apa yang terkandung dalam pengkodean pada QR Code seperti adanya link dari profil line yang dapat diakses dan di add untuk menambahkan menjadi teman di line. Selain di line masih banyak lagi permodelan mengakses sebuah situs dari QR Code.

1.1.5 QR Code dalam Industry

Dalam [perindustrian QR Code dapat digunakan dalam berbagai hal contoh dalam cek stok persediaan barang pada Gudang, Melihat infirmasi yang tertera dalam Kemasan barang tersebut dan masih banyak lagi.

1.2 Android

1.2.1 Definisi Android

Android merupakan sistem operasi yang berbasiskan kernel Linux dan merupakan perangkat lunak Open Source yang dirancang oleh Google, biasanya android digunakan untuk perangkat dengan layar sentuh seperti smartphone dan tablet. Google sebagai pemilik dari android tidak pernah memungut biaya bagi siapapun yang menggunakan Sistem Operasi Android hal ini dikarenakan Sistem Operasi Android merupakan perangkat lunak open source. Berbeda halnya dengan sistem operasi Windows 10 Mobile milik Microsoft, pada Sistem Operasi Windows 10 Mobile perusahaan mengharuskan pengguna membayarkan royalty jika ingin menggunakan sistem operasi tersebut. Sedangkan pada iOS milik Apple, sistem operasi ini hanya dapat diakses dan digunakan di perangkat iPhone dan iPad saja dan Apple tidak merilisnya untuk perangkat lain. Walau Google menggratiskan Sistem Operasi Android ini

tetapi Google tetap mendapat keuntungan dari aplikasi yang di upload di play store dan iklan yang ada pada aplikai tersebut.

1.2.2 Sejarah Android

Android sebelum berada di bawah kepemilikan google android berada dibawah naungan Android, Inc. Perusahaan ini didirikan pada tahun 2003 oleh Andy Rubin, Rich Miner, Nick Sears dan Chris White yang berlokasi di Palo Alto, California. Pada mulanya android dibuat kamera digital bukan ponsel seperti yang sekarang ada. Hingga mereka menyadari adanya peluang untuk mengembangkan perangkat mobile. Pada tahun 2005, 17 Agustus Google mengakuisisi Android, Inc dimana para pendiri Android berada dalam naungan google hingga pada 22 Oktober 2008 terciptalah ponsel seluler pertama berbasis android yaitu HTC Dream.

1.2.3 Versi Android

Seperti halnya dengan system operasi lain Android pun memiliki beberapa kali pembaruan setelah pertama kali rilis pada tahun 2008. Pembaruan tersebut sangatlah penting karena dengan adanya pembaruan system berarti android telah dikembangkan menjadi system operasi yang lebih baik dalam artian lebih efisien dan efektif dantentunya lebih memudahkan penggunanya. Berikut Versi versi dari android:

1. Android 1.0 (Apple Pie)

Dirilis pada 23 September 2008 dengan fitur : Play Store, kamera, Web Browser, Sinkronisasi antara G-mail, Contacts dan Google Agenda, Google Maps dan dukungan streaming Youtube.

2. Android 1.1 (Banana Bread)

Dirilis pada bulan Februari 2009. Fiturnya yaitu tidak jauh berbeda dengan versi sebelumnya. HTC Dream merupakan smartphone Android pertama yang menggunakan versi ini.

3. Android 1.5 (Cupcake)

Dirilis pada awal bulan April 2009. Fitur tambahan : sudah Support Bluetooth A2DP, AVRCP, Soft-keyboard dengan prediksi text dan record atau watch videos.

4. Android 1.6 (Donut)

Dirilis pada 15 September 2009. Fitur tambahan : Gesture Framework hingga Turn-by-turn navigation, minimnya bug, dan ditambah lebih lengkapnya berbagai fitur yang disediakan oleh Google.

5. Android 2.0 (Eclair)

Dirilis pada 26 Oktober 2009. Fitur tambahan : multi-touch, Live Wallpaper dan juga flash kamera, HTML, Digital zoom, Support Microsoft Exchange, dan Updated UI.

6. Android 2.2.9 (Froyo)

Dirilis pada bulan Mei 2010 Versi ini merupakan salah satu sistem operasi Android yang juga telah disempurnakan, dengan tujuan untuk meningkatkan kecepatan kinerja dari sistem Android.

Fitur dan perbaikan :Peningkatan Speed, Implementasi JIT, USB Tethering, Aplikasi instalasi untuk perluasan memori atau storage, Support file upload pada aplikasi browser, Animated GIFs.

7. Android 2.3 (Gingerbread)

Dirilis pada bulan Desember 2010. Android 2.3 ini yang pertama kali diadopsi oleh salah satu perusahaan Smartphone paling populer, yaitu Samsung dengan menanamkan sistem operasi ini pada smartphone seri Nexus-nya.

8. Android 3.0 – 3.2.6 (Honeycomb)

Dirilis pada bulan Februari 2011. Versi ini lebih ditujukan untuk perangkat Tablet yang mana pada tahun itu sangat laris atau laku dipasaran.

Fitur dan perbaikan : Support Multi core, Support Tablet lebih baik, Updated 3D UI, Layar Utama (homescreens) yang dapat diatur, Melihat aplikasi yang barusan dibuka, Menyempurnakan layout keyboard, Transport protocol untuk Media atau Picture, video chat Google Talk, Google eBooks, “Private browsing”, System-wide Clipboard, HTTP Live streaming.

(a) Update 3.1 :

Peningkatan UI, Open Accessory API, USB host API, Support mouse, joysticks dan gamepad, Widget Home screen yang bisa di atur size atau ukurannya, Notifikasi MTP, RTP API untuk audio

(b) Update 3.2 :

Optimise pada berbagai tablets, Mode kompatibilitas display (zoom for fixed sized apps), Sinkronisasi Media dari SD card

(c) Update 3.2.1 :

Update Android Market merupakan automatic updates yang lebih mudah, Update Google Books, Peningkatan kinerja Wi-Fi, Perbaikan prediksi tulisan tangan dengan huruf Chinese

(d) Update 3.2.2 :

Perbaikan kecil

(e) Update 3.2.4 :

Update tambahan ‘Pay as you go’ bagi tablet

(f) Update 3.2.6 :

Perbaikan kecil

9. Android 4.0 (Ice Cream Sandwich)

Dirilis pada bulan Oktober 2011. Sistem operasi ini dapat bekerja dengan baik pada smartphone jenis apapun. dan merupakan versi yang paling banyak disukai pada saat itu.

Fitur tambahan : ekstra multitasking serta notifikasi yang lebih banyak.

10. Android 4.1.2 (Jelly Bean)

Dirilis pada 9 Juli 2012 melalui konferensi I/O Google. Fitur tambahan : memperbaiki rotasi layar, seperti Support resolusi video 4K, Support penulisan huruf Hebrew dan Arabic dari kanan ke kiri, peningkatan kinerja, dan sistem keamanan.

11. Android 4.4 (Kitkat)

Dirilis pada tahun 2013. Fitur tambahan : terdapat fitur Screen recording, untuk merekam kegiatan yang terjadi pada layar smartphone, Peningkatan akses notifikasi, New Translucent system UI, System wide settings untuk closed captioning, dan Peningkatan kinerja.

12. Android 5.0 (Lollipop)

Dirilis pada tahun 2014. Menyempurnakan berbagai fitur yang sudah ada. Nexus 6 sebagai salah satu ponsel yang pertama menggunakan Android Lollipop ini.

13. Android 6.0 (Marshmallow)

Dirilis pada tahun 2015. Pembaharuan : support USB Type-C, fasilitas autentikasi sidik jari dan daya baterai yang lebih baik.

14. Android 7.0 (Nougat)

Android Nougat versi 7.0 dirilis pada bulan Agustus 2016. Fitur terbaru : Support Multi window, Dapat langsung membalsas pesan dari menu notifikasi atau jendela., Tampilan panel notifikasi serta quick settings yang baru., Mode Doze yang lebih baik, (Doze Mode 2.0), Menu di antara system settings.

15. Android 8.0 (Oreo)

Dirilis pada bulan Agustus 2017. Fitur-fitur terbaru : Android O lebih berfokus pada kecepatan dan efisiensi,, Kecepatan Boot up 2X lebih cepat, Mode Picture in picture lebih flexible, Aplikasi yang berjalan di latarbelakang atau background lebih diperketat untuk lebih menghemat battery, Battery lebih tahan lama, Emoji yang diperbaharui dan diperbanyak

1.3 Android Studio

1.3.1 Definisi Android Studio

Android Studio merupakan Integrated Development Environment (IDE) resmi ntk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Android berubah mn platform ng gt cepat m mkkn inovasi. H n tk lepas pengembangan utama di belakangnya, t Google. Googleelah ng mengakuisisi Android n kmn membuatkan platform.

Platform android terdiri Sistem Operasi berbasis Linux, sebuah GUI (Graphic User Interface), sebuah web browser n Aplikasi Studio End-User ng dapat download n g para pengembang ngn leluasa berkarya serta menciptakan aplikasi ng tk n terbuka ntk digunakan g macam perangkat.

1.3.2 Sejarah Android Studio

Pada awalnya mulanya, Android In merupakan perusahaan software kecil ng didirikan pada bulan Oktober 2003 Palo Alto, California, USA. Didirikan senior beberapa perusahaan ng berbasis IT Communication, Andy Rubin, Rich Miner, Nick Sears n Chris White.

Mnt Rubin, Android In Didirikan ntk mewujudkan mobile device ng peka t lokasi n preferensi pemilik. Dngn kt lain, Android In ngn mewujudkan mobile device ng mengerti pemiliknya.

Konsep ng dimiliki Android In ternyata menggugah minat Google ntk memilikinya. Pada bulan Agustus 2005, Akhirnya Android In diakuisisi Google In. sahamnya dibeli Google. Bnk ng memperkirakan n pembelian Android Inch O Google sebesar USD 50 juta. saat t nk ng berspekulasi w akuisisi n langkah awal ng dilakukan Google ntk masuk kepasar mobile phone.

Andy Rubin, Rich Miner, Nick Sears n Chris White tetap Android In ng dibeli Google, sehingga akhirnya mk m mn bagian raksasa Google n sejarah Android. Saat itulah mk m mnggnkn platform Linux ntk mmr sistem operasi bagi mobile phone.

D t alat editor n pengembang hebat IntelliJ, Android Studio menawarkan nk fitur ng meningkatkan produktivitas An saat membangun aplikasi Android, seperti:

Sistem build berbasis Gradle ng fleksibel

1. Emulator cepat dan kaya fitur
2. Lingkungan terpadu yang dapat Anda kembangkan untuk semua perangkat Android
3. Instan Berjalan untuk mendorong perubahan pada aplikasi yang sedang berjalan tanpa membangun APK baru
4. Template kode dan integrasi GitHub untuk membantu Anda membuat fitur aplikasi umum dan kode nt impor
5. Alat pengujian ekstensif dan kerangka kerja

6. Alat lint untuk menangkap kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya
7. Dukungan C ++ n NDK
8. Dukungan terintegrasi untuk Google Cloud Platform, sehingga mudah mengintegrasikan Google Cloud Messaging dan App Engine Halaman ini
9. Memberikan pengenalan pada fitur dasar Android Studio. Untuk ringkasan perubahan terbaru, lihat juga Android Studio Relase
10. Struktur Proyek

Setiap proyek di Android Studio berisi satu atau lebih modul dengan file kode sumber dan file sumber daya. Jenis modul meliputi:

1. Modul aplikasi Android
2. Modul perpustakaan
3. Modul Google App EngineSecara default

Android Studio menampilkan file proyek Anda dalam tampilan proyek Android. Tampilan ini disusun oleh modul untuk menyediakan akses cepat ke file sumber utama proyek Anda.Semua file build terlihat di tingkat atas di bawah Gradle Scripts dan setiap modul aplikasi folder berikut:

1. Bermanifestasi: Berisi file Android Manifest.xml.
2. Java: Berisi file kode sumber Java, termasuk kode uji JUnit.
3. Res: **erisi** m sumber daya non-kode, seperti tata letak XML, string UI, dan gambar bitmap.

Struktur proyek Android pada disk berbeda dari representasi yang rata ini. Untuk melihat struktur file proyek yang sebenarnya, pilih Project from the Project dropdown.

Anda juga dapat menyesuaikan tampilan file proyek agar fokus pada aspek spesifik pengembangan aplikasi Anda. Misalnya, memilih tampilan Masalah proyek Anda yang menampilkan tautan ke file sumber yang berisi kesalahan pengkodean dan sintaks yang dikenali, seperti tag penutup elemen XML yang hilang dalam file tata letak.

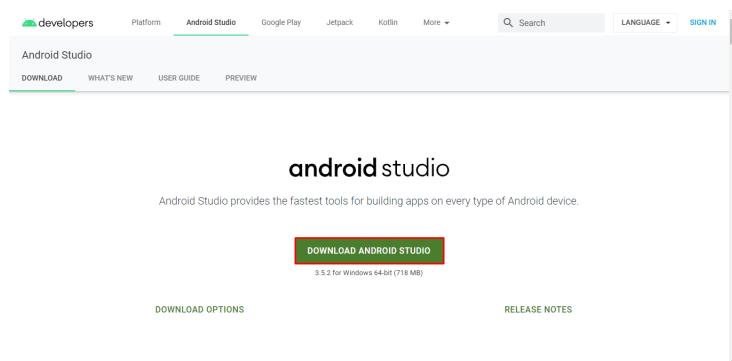
File proyek dalam tampilan Masalah, menampilkan file tata letak dengan masalah.

1.3.3 Instalasi Android Studio

Android studio adalah salah satu software dalam membuat aplikasi android yang paling populer dan mudah digunakan.Untuk dapat menginstal Android Studio kita membutuhkan:RAM minimal 3GB, direkomendasikan 8GB RAM dengan minimal 2 GB penyimpanan yang tersedia direkomendasikan 4 GB (500 MB untuk IDE dan 1,5 GB untuk Android SDK dan sistem emulator), dan dengan Resolusi layar minimum 1280 x 800. Instalasi Android Studio debagi menjadi dua yaitu Setup Android Studio dan Instalasi SDK.

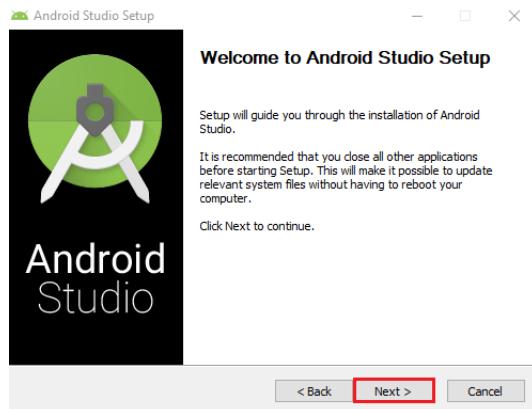
1.3.3.1 Setup Android Studio

1. Sebelum melakukan instalasi, hal pertama yang harus dilakukan adalah memastikan kita memiliki instalasi dari Android Studio, jika belum memiliki kita dapat mengunduh file Android Studio. Pada gambar 1.2 berikut gambar dari laman mendownload Instalasi Android Studio:



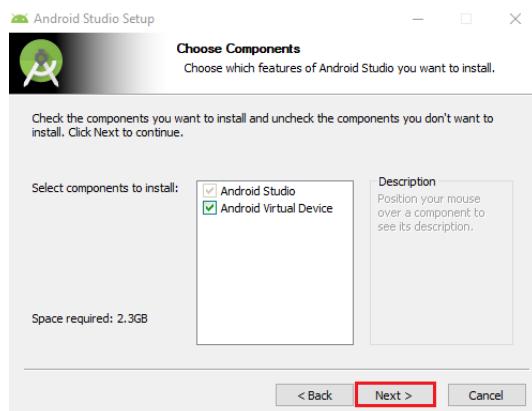
Gambar 1.2 Download Instalasi Android Studio

2. Setelah selesai download file Android Studio, buka file tersebut dan ikuti instruksi instalas, lalu akan muncul halaman seperti pada gambar 1.3 di bawah ini. Klik Next untuk melanjutkan ke proses instalasi.



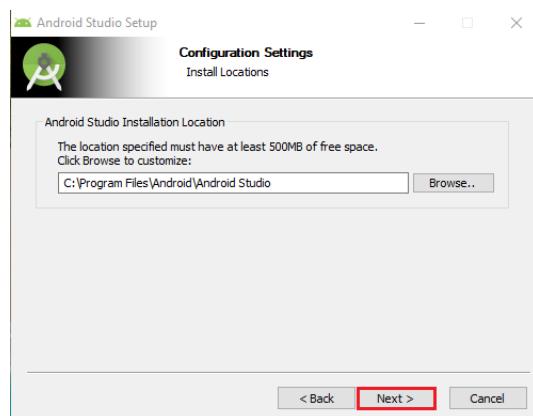
Gambar 1.3 Tampilan Awal Saat Instalasi

3. Lalu kita dapat memilih komponen tambahan untuk install Android Studio. AVD (Android Virtual Device) yang fungsinya adalah untuk mengkonfigurasi perangkat yang dijalankan dengan emulator Android. Sesuaikan komponen tambahan yang dipilih seperti pada gambar 1.4di bawah ini. Jika sudah klik Next untuk melanjutkan instalasi.



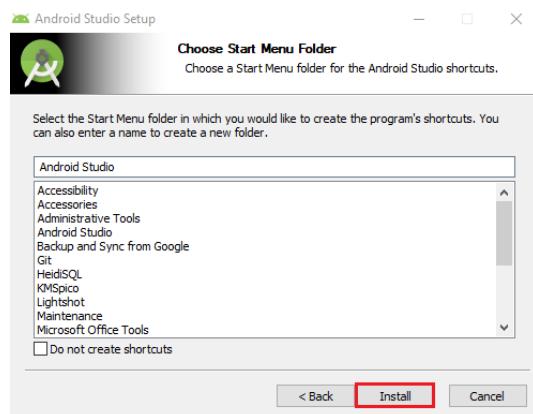
Gambar 1.4 Memilih komponen yang akan diinstal

4. Selanjutnya, kita dapat pilih lokasi untuk install Android Studio pada komputer. sebagai contoh di lokasi drive C:. Setelah menentukan lokasi instalasi Android Studio, klik Next untuk melanjutkan. Seperti pada gambar 1.5di bawah ini.



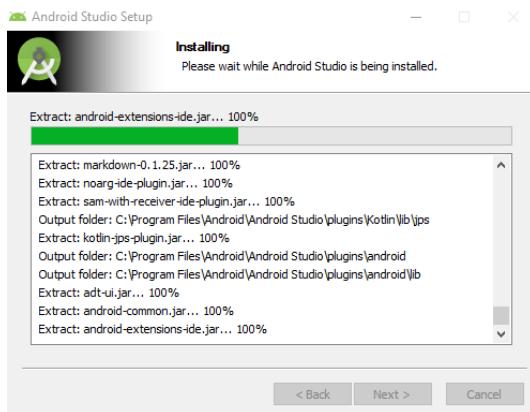
Gambar 1.5 Memilih lokasi file

5. Lalu kita dapat mengganti nama aplikasi Android Studio yang akan ditampilkan pada Start Menu. Akan tetapi, demi kemudahan saat mencari aplikasi ini, sebaiknya gunakan nama Android Studio saja. Klik Install untuk melanjutkan proses.Seperti pada gambar 1.6di bawah ini.



Gambar 1.6 Memilih lokasi file

6. Setelah menentukan nama aplikasi Android Studio, Anda bisa memulai proses instalasi Android Studio. Gambar 1.7 di bawah ini adalah proses instalasi Android Studio, tunggu hingga proses Selesai.

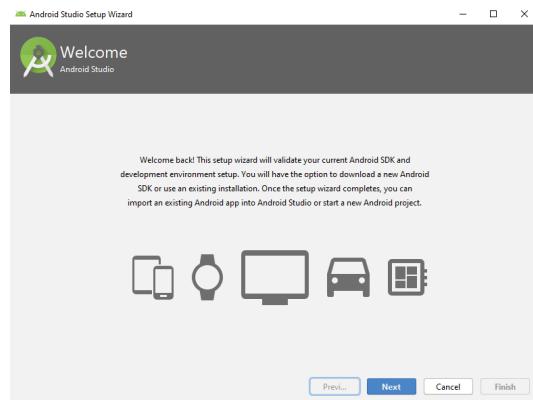


Gambar 1.7 Memilih lokasi file

1.3.3.2 Install SDK Android Studio SDK adalah seperangkat alat dan program perangkat lunak yang digunakan oleh pengembang untuk membuat aplikasi untuk platform tertentu.

Sebelum menginstall SDK Android Studio, pastikan perangkat Anda terhubung ke internet. Hal ini dikarenakan proses download untuk komponen-komponen SDK Android Studio membutuhkan koneksi internet.

1. Sebelumnya, kita telah berhasil menginstall Android Studio. Lalu kita dapat membuka aplikasi tersebut dan ikuti instruksi instalasi SDK di bawah ini: Hal yang akan muncul halaman seperti pada gambar 1.8 di bawah ini. Klik Next untuk melanjutkan ke proses instalasi.

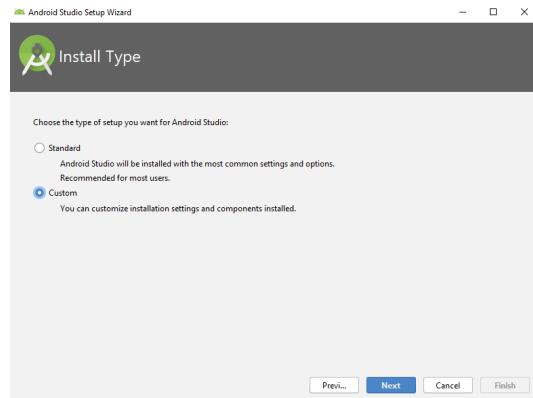


Gambar 1.8 Tampilan Awal pada Android Studio

2. Pilih tipe instalasi yang terdiri dari 2 yaitu:

- Standard: Untuk pilihan standard, Anda akan mendapatkan default pengaturan dan instalasi tambahan dari Android Studio.
- Custom: Untuk pilihan custom, Anda bisa memilih pengaturan dan komponen tambahan yang Anda perlukan saja.

Kali ini sebagai contoh kita dapat memilih Custom agar aplikasi tambahan yang tidak diinginkan tidak terinstall. Klik Next untuk melanjutkan instalasi. Seperti pada gambar?? di bawah ini:

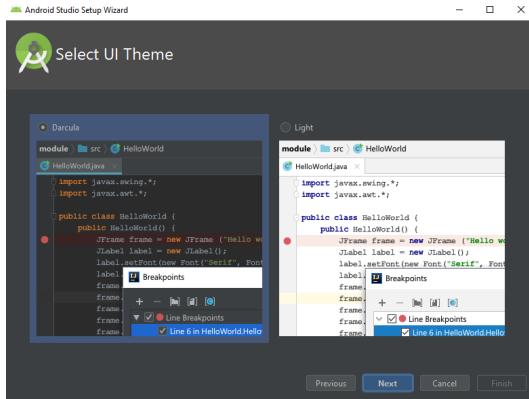


Gambar 1.9 Tampilan Awal pada Android Studio

3. Kemudian, pilih tema untuk tampilan di dashboard Android Studio. Ada dua pilihan pada tema Android Studio, yaitu:

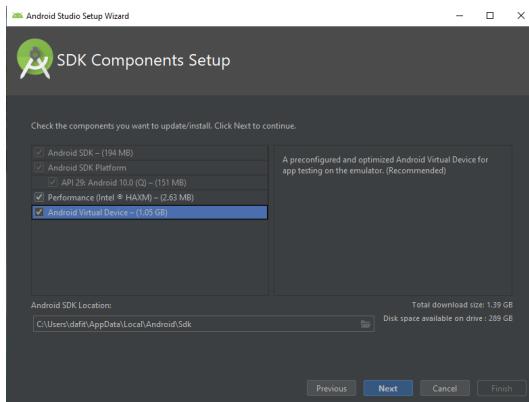
- Darcula : Bertema warna Dark (Hitam)
- Light : Bertema warna Light (Putih)

Di tutorial ini kita dapat memilih tema Darcula. Seperti pada gambar 1.10 Lalu klik Next untuk melanjutkan.



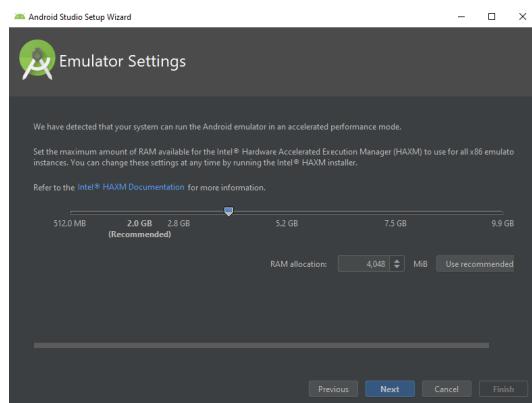
Gambar 1.10 Tampilan Awal pada Android Studio

4. Pilih Komponen SDK Tambahan, Langkah ini muncul jika kita memilih instalasi Custom. Lalu Sesuaikan komponen tambahan yang dipilih seperti pada gambar 1.11 di bawah ini. Kemudian klik Next untuk melanjutkan instalasi.



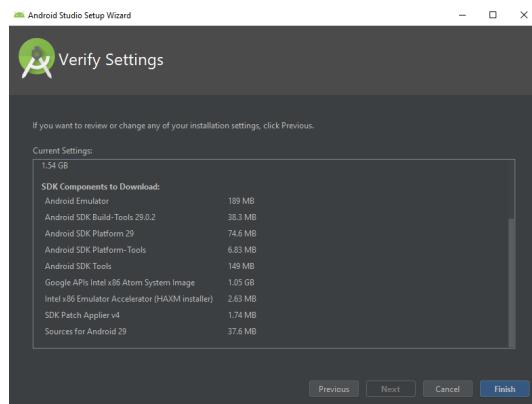
Gambar 1.11 Tampilan Awal pada Android Studio

5. Selanjutnya, tentukan RAM. Sebaiknya gunakan RAM minimal 4GB agar tidak memperlambat proses running Emulator ketika menjalankan aplikasi yang telah Anda buat. Klik Next untuk melanjutkan instalasi. seperti pada gambar 1.12 dibawah ini:



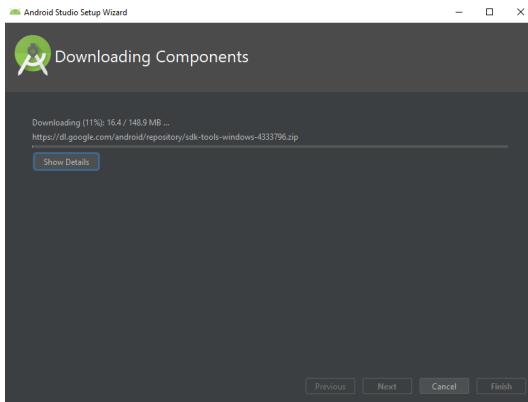
Gambar 1.12 Tampilan Awal pada Android Studio

6. Lalu kita akan diberikan informasi mengenai komponen tambahan SDK beserta ukuran filenya. Klik Finish untuk memulai proses download komponen-komponen tersebut. seperti pada gambar 1.13 berikut :



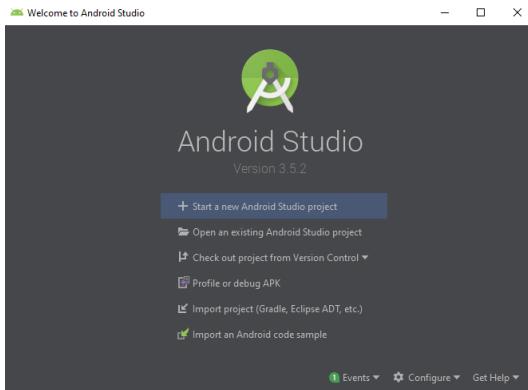
Gambar 1.13 Tampilan Awal pada Android Studio

7. Selanjutnya adalah proses download dan instalasi file komponen SDK cukup lama karena ukuran filenya yang cukup besar. Oleh karena itu, pastikan koneksi internet Anda stabil agar tidak terjadi error saat proses instalasi berlangsung.



Gambar 1.14 Tampilan Awal pada Android Studio

8. Setelah proses download dan install selesai, akan muncul tampilan seperti gambar 1.15 di bawah ini:



Gambar 1.15 Tampilan Awal pada Android Studio

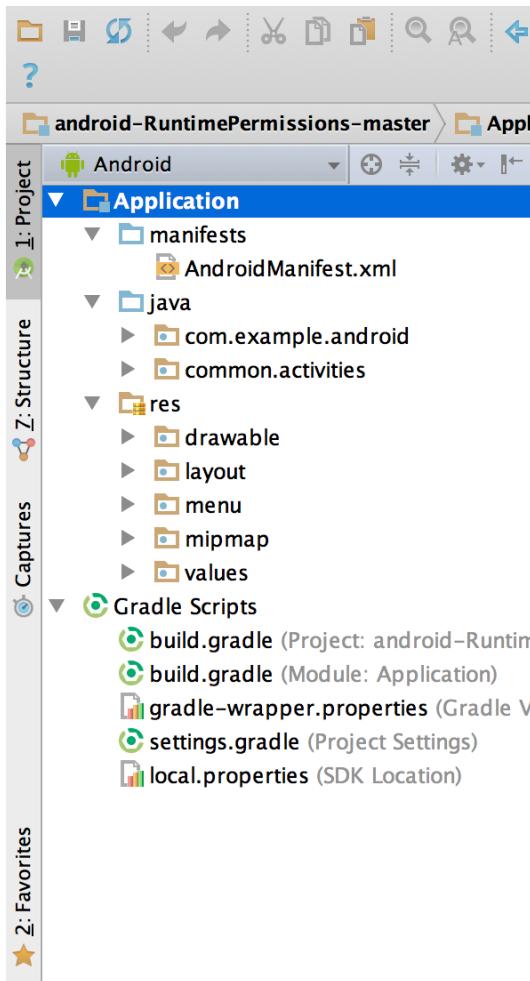
1.3.3.3 Mengenal Komponen Android Studio

- Struktur project

Jenis modul pada struktur proyek android studio meliputi Modul aplikasi, Android Modul library, Modul Google App Engine. Semua file build terlihat di tingkat teratas di bagian Gradle Script dan setiap modul aplikasi berisi folder berikut:

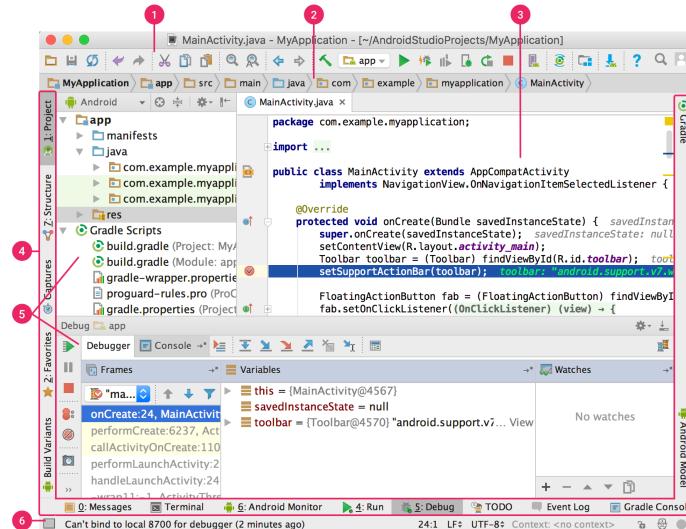
- (a) manifes: Berisi file Android Manifest.xml.
- (b) java: Berisi file kode sumber Java, termasuk kode pengujian JUnit.
- (c) res: Berisi semua resource non-kode, seperti tata letak XML, string UI, dan gambar bitmap.

Berikut File Project dalam Tampilan Android Studio:



Gambar 1.16 Tampilan File project dalam tampilan Android Studio

- Antarmuka Pengguna dalam Android Studio utama Android Studio terdiri dari beberapa area logis yang diidentifikasi dalam gambar berikut



Gambar 1.17 Tampilan Jendela utama Android Studio

- Toolbar memungkinkan Anda melakukan berbagai tindakan, termasuk menjalankan aplikasi dan meluncurkan fitur Android.
- Menu navigasi membantu Anda menjelajah project dan membuka file untuk diedit. Menu ini memberikan tampilan struktur yang lebih ringkas yang terlihat di jendela Project.
- Jendela editor adalah tempat Anda membuat dan memodifikasi kode. Tergantung jenis file yang ada, editor ini dapat berubah. Misalnya, saat menampilkan file tata letak, editor akan menampilkan Layout Editor.
- Panel jendela fitur berada di sisi luar jendela IDE dan berisi tombol-tombol yang memungkinkan Anda memperluas atau mencuitkan setiap jendela fitur.
- Jendela fitur memberi Anda akses ke tugas tertentu seperti pengelolaan project, penelusuran, kontrol versi, dan banyak lagi. Anda dapat memperluas dan mencuitkan jendela ini.
- Status bar menampilkan status project Anda dan IDE itu sendiri, serta semua peringatan atau pesan.

BAB 2

LANDASAN TEORI

2.1 Tinjauan Pustaka

Pembuatan aplikasi ini berbasis android yang didalamnya terdapat beberapa fitur akan dibangun. Sebelum membuat sebuah aplikasi ini, terdapat perancangan sistem terlebih dahulu.

Perancangan sistem yang dimaksud ialah perancangan UML (*Unified Modelling Language*) yaitu suatu metode permodelan secara visual sebagai sarana perancangan sistem berorientasi objek terdapat beberapa diagram yaitu Use case diagram, sequence diagram, collaboration diagram, activity diagram, statechart diagram, component diagram, deployment diagram, objek diagram, struktur menu, dan perancangan antarmuka terlebih dahulu.

Aplikasi ini juga memakai database yang berkaitan dengan fitur yang akan dibangun. Bahasa pemrograman yang digunakan adalah Java.

2.2 Teori Umum

2.2.1 Database

Database merupakan suatu kumpulan data yang sebelumnya sudah dihimpun dikumpulkan. Database atau basis data merupakan kumpulan berbagai data dan informasi yang ada dan disimpan di suatu media tertentu, umumnya pada komputer.

Untuk mempermudah akses komputerisasi merupakan tujuan dari pengolahan database dalam komputer Pengolahan sistem database disebut dengan istilah RDBMS atau Relational Database Management System yang berarti sebagai kumpulan tabel yang menyimpan data dan informasi yang saling berhubungan atau berelasai.

Database sendiri memiliki fungsi salah satunya untuk mengelompokkan data dengan tujuan untuk mempermudah identifikasi data. Pengelompokan bisa dilakukan dengan membuat beberapa tabel atau field yang berbeda. Dengan database kita mampu menghindari perulangan data atau redundansi. Pada dasarnya semua kata perintah telah direkam pada komputer sehingga dapat diketahui jika ada nama file ada data yang terinput ulang atau ganda. Database dapat mempermudah penggunaan dalam memasukkan data baru ataupun mengupdate data dan menghapusnya. Karena dengan database dapat melakukan riset dari jarak jauh dengan adanya akses untuk menelpon dan berkomunikasi melalui nomor yang tertera. Fungsi selanjutnya ialah menghemat daya penyimpanan pada produk tertentu. contoh kita dapat menyimpan data secara langsung melalui server dan jaringan internet sehingga tidak perlu lagi memberdayakan sumber daya manusia untuk menghimpun data dengan jumlah yang banyak serta dengan database kita dapat mengurangi penggunaan kertas sebagai langkah dalam menjaga bumi.

2.2.1.1 Manfaat Database Dengan menggunakan database kita mendapatkan manfaat dari penggunaan database tersebut seperti berikut:

1. Tidak Terjadi Redundansi Basis Data Database bisa membantu meminimalkan redundansi data. Redundansi sendiri merupakan terjadinya perulangan data atau data-data ganda dalam berkas-berkas yang berbeda.
2. Integritas Data Terjaga Database memiliki integritas data yang tinggi, database dapat memastikan keakuratan, aksesibilitas, konsistensi dan juga kualitas tinggi pada data.
3. Independensi Data Terjaga Database menjaga independensi data yang berarti bahwa orang lain tidak dapat merubah data meskipun data bisa diakses orang tersebut.
4. Kemudahan Berbagi Data Dengan database kita dapat berbagi data atau informasi dengan sesama pengguna.
5. Menjaga Keamanan Data Database dapat menjamin keamanan suatu informasi dan data, dimana kita dapat menyisipkan kode akses untuk data-data tertentu yang tidak bisa diakses bersama.

6. Kemudahan Akses Data Dengan database bisa memudahkan kita untuk dapat mengakses dan mendapatkan data karena semua data telah terorganisir dengan baik.

2.2.1.2 *Tipe-Tipe Database* Dibutuhkan software khusus untuk menyimpan dan mengambil data dan informasi dari database. Software ini sering disebut dengan System Managemen Basis Data (DBMS). Berikut ini adalah tipe database:

1. Analytical database; yaitu database untuk menyimpan informasi dan data yang diambil dari operasional dan eksternal database
2. Operational database; yaitu database yang menyimpan data mendetail yang dibutuhkan untuk mendukung operasi suatu organisasi secara keseluruhan
3. Distributed database; yaitu kelompok kerja lokal database dan departemen di berbagai kantor dan lokasi kerja yang lainnya.
4. Data warehouse; yaitu sebuah gudang data yang menyimpan berbagai data dari tahun-tahun sebelumnya hingga saat ini.
5. End-user database; yaitu basis data pengguna akhir yang terdiri dari berbagai file data yang dikembangkan dari end-user dalam workstation mereka.
6. End-user database; yaitu basis data pengguna akhir yang terdiri dari berbagai file data yang dikembangkan dari end-user dalam workstation mereka.
7. Real time database; yaitu sistem pengolahan yang dirancang dalam menangani beban kerja suatu negara yang bisa berubah-ubah, mengandung data terus menerus dan sebagian tidak berpengaruh terhadap waktu.
8. Document oriented database; yaitu salah satu perangkat lunak komputer yang dibuat untuk sebuah aplikasi dan berorientasi pada dokumen.
9. In memory database; yaitu database yang tergantung pada memori untuk menyimpan informasi/ data pada komputer
10. Navigational database; pada navigasi database, queries menemukan benda bagi yang mengikuti referensi dari objek tertentu
11. Hypermedia database on the web; sekumpulan halaman multimedia yang saling berhubungan dalam sebuah website, yang terdiri dari homepage dan hyperlink dari multimedia (gambar, teks, grafik audio, dan lain-lain)
12. External database; database yang menyediakan akses ke luar, dan data pribadi online
13. Relational database; merupakan standar komputasi bisnis, dan basis data yang paling umum dipakai saat ini.

2.2.1.3 Software dalam Membuat Database Jika kita menggunakan perangkat lunak database, maka berikut ini jenis-jenis software database terbaik yang bisa Anda gunakan:

1. Microsoft Access Salah satu software database ini adalah yang paling sering digunakan. Microsoft access sangat cocok digunakan untuk sebagian besar komputer yang relasional.
2. Oracle Salah satu software database ini sangat mampu untuk menyimpan data dengan ukuran yang maksimum hingga tera byte. Oracle paling banyak digunakan pada perusahaan-perusahaan terutama yang sedang berkembang karena memang untuk mengaksesnya tersedia secara gratis.
3. MySQL Salah satu software database yang open access untuk umum dan kompatibel pada sistem operasi Windows maupun Linux. Keunggulan yang bisa Anda gunakan dengan menggunakan program MySQL adalah bisa digunakan untuk multi user. Kelebihan lainnya dari MySQL yaitu tersedia gratis, query data yang cepat dan berlisensi resmi.
4. Firebird Bisa dibilang software database ini memiliki fitur sistem yang standar dan ringan yaitu ANSI SQL-99 dan SQL – 2003. Kompatibel untuk digunakan pada sistem operasi Windows, Linux maupun Unix.
5. PostgreSQL Menawarkan sistem database opensource dengan lisensi GPL/ General Public License. Software ini menggunakan bahasa pemrograman C++, C, SQL, PHP dan lainnya. Jika digunakan untuk pekerjaan pribadi, maka software ini sangat recommended digunakan.

2.2.2 SQLITE

SQLite ialah sebuah perpustakaan perangkat lunak yang menerapkan engine database SQL secara mandiri, tanpa memerlukan server, tanpa perlu melakukan konfigurasi, dan bersifat transaksional. SQLite adalah engine database SQL yang paling banyak digunakan di dunia. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp.

SQLite adalah sebuah engine database SQL yang langsung tertanam atau pada aplikasi. Tidak seperti database SQL lainnya, SQLite memiliki server yang bersatu dengan aplikasi. SQLite membaca dan menulis langsung ke file disk biasa. Walau server tertanam pada aplikasi database SQLite memiliki fitur lengkap dengan banyak tabel, indexs, trigger, dan tampilan, serta tersimpan pada satu file tunggal dalam hard-disk. Format file databasenya bersifat cross-platform. Sehingga kita dapat dengan bebas menyalin database antara sistem 32-bit dan 64-bit atau antara arsitektur yang berbeda platform. Fitur-fitur tersebut membuat SQLite menjadi pilihan populer sebagai Application File Format. SQLite telah didukung oleh banyak bahasa pemrograman seperti C, C++, BASIC, C, Python, Java dan Delphi.

2.2.2.1 Keunggulan SQLite

- SQLite tidak memerlukan proses atau sistem server yang terpisah untuk beroperasi (Serverless).
- SQLite hadir dengan zero-configuration, yang berarti tidak ada setup atau administrasi yang dibutuhkan.
- Database SQLite yang lengkap disimpan dalam file tunggal yang tersimpan dalam disk serta bersifat cross-platform.
- SQLite sangat kecil dan ringan, kurang dari 400KiB untuk konfigurasi lengkap atau kurang dari 250KiB dengan fitur opsional dihilangkan.
- SQLite bersifat mandiri, yang berarti tidak ada dependensi eksternal.
- Transaksi SQLite sepenuhnya sesuai dengan ACID, memungkinkan akses yang aman dari banyak proses.
- SQLite mendukung sebagian besar fitur bahasa query yang ditemukan dalam standar SQL92 (SQL2).
- SQLite ditulis dalam ANSI-C dan menyediakan API yang sederhana dan mudah digunakan.
- SQLite tersedia di semua sistem operasi baik ini UNIX (Linux, Mac OS-X, Android, iOS) dan Windows (Win32, WinCE, WinRT).

2.2.2.2 Cara Menggunakan SQLite

Untuk memulai database SQLite di Android. API yang diperlukan untuk menggunakan database di Android tersedia dalam paket android.database.sqlite.

1. Menentukan skema dan kontrak, Salah satu prinsip utama database SQL adalah skemanya, Class kontrak berfungsi untuk menampung konstanta yang menentukan nama URI, tabel, dan kolom. Class kontrak memungkinkan Anda menggunakan konstanta yang sama pada semua class lain dalam paket yang sama. Hal ini memungkinkan Anda mengubah nama kolom di satu tempat, kemudian mengatur agar perubahan tersebut disebarluaskan ke seluruh kode.

Cara yang tepat untuk mengatur class kontrak adalah dengan memberikan definisi yang bersifat global pada seluruh database Anda di tingkat root class tersebut. Kemudian, buat class dalam untuk setiap tabel. Setiap class dalam akan menghitung kolom tabel yang terkait.

2. Membuat database menggunakan SQL helper, Setelah menentukan tampilan database, Anda harus menerapkan metode yang akan membuat serta mengelola database dan tabel. Class SQLiteOpenHelper berisi kumpulan API yang berguna untuk mengelola database. Saat class ini digunakan untuk memperoleh referensi ke database, sistem hanya akan melakukan operasi pembuatan dan update database, yang mungkin memerlukan banyak waktu, hanya ketika diperlukan; bukan pada saat aplikasi dimulai. Anda hanya perlu memanggil getWritableDatabase() atau getReadableDatabase().

Masukkan data ke database dengan meneruskan objek ContentValues ke metode insert():

3. Membaca informasi dari database, menggunakan metode query() dengan meneruskan kriteria pemilihan dan kolom yang Anda inginkan. Metode ini menggabungkan elemen insert() dan update(), tetapi daftar kolomnya menentukan data yang ingin diambil ("proyeksi"), bukan data yang akan dimasukkan. Hasil kueri dikembalikan kepada Anda dalam objek Cursor.
4. Menghapus informasi dari database, untuk menghapus menggunakan perintah delete(). dengan mekanismenya bekerja dalam cara yang sama seperti argumen pemilihan untuk metode query().
5. Mengupdate database Saat Anda perlu mengubah subkumpulan nilai database, gunakan metode update().
6. Mempertahankan koneksi database, Karena getWritableDatabase() dan getReadableDatabase() sulit dipanggil jika database ditutup, koneksi database harus tetap terbuka selama Anda perlu mengaksesnya. Biasanya, akan lebih optimal untuk menutup database dalam onDestroy() Aktivitas pemanggilan.

2.2.3 Penggunaan Bahasa Pemrograman JAVA

Ada perbagai Bahasa pemrograman salah satunya yaitu Bahasa pemrograman Java. Bahasa pemrograman Java inilah yang biasanya digunakan untuk smartphone menggunakan perangkat lunak dengan bahasa pemrograman JAVA. Tidak hanya di komputer , Java juga dapat dijalankan di smartphone.

Java merupakan bahasa bahasa pemrograman tingkat tinggi yang berorientasi pada object atau biasa disebut dengan OOP dan program java tersusun dari bagian yang disebut dengan Class. Class sendiri terdiri dari metode- metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para programmer Java banyak mengambil keuntungan dari kumpulan class di pustaka class Java yang disebut dengan Java Application Programming Interface (API). Class- class ini diorganisasikan menjadi sekelompok yang disebut dengan paket (package). Java API telah menciptakan applet dan aplikasi canggih dengan menyediakan fungsionalitas yang memadai .

Jadi ada dua hal penting yang harus difahami dalam Java, yaitu bagaimana mempergunakan class pada Java API dan mempelajari bahasa Java. Tidak ada cara lain

selain class yang merupakan satu-satunya cara menyatakan bagian eksekusi program. Pada Java program javac untuk mengkompilasi file kode sumber Java menjadi class-class bytecode. File kode sumber mempunyai ekstensi *.java. Kompilator javac menghasilkan file bytecode class dengan ekstensi *.class. Interpreter adalah modul utama pada sistem Java yang digunakan aplikasi Java dan menjalankan program bytecode Java.

Jadi Java adalah bahasa pemrograman yang dapat membuat seluruh bentuk aplikasi tidak hanya desktop dan web namun juga bisa membuat aplikasi mobile dan sebagainya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain. Java merupakan bahasa pemrograman yang bersifat umum atau non-spesifik (general purpose). Bahasa Pemrograman Java berorientasi object (OOP-Object Oriented Programming), dan dapat dijalankan pada berbagai platform sistem operasi. Pada OOP, program komputer sebagai kelompok object yang saling berinteraksi.

2.2.3.1 OOP (Object Oriented Programming) Bahasa Pemrograman JAVA tidak lepas dan identik dari OOP atau Object Oriented Programming Pengertian OOP secara singkat adalah mengorganisasikan program sebagai kumpulan komponen yang disebut object. Object-object ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan object lain dan untuk memerintahkan object lain gunanya untuk meminta informasi tertentu atau meminta object lain mengerjakan sesuatu.

Class bertindak sebagai modul sekaligus tipe. Sebagai tipe maka pada saat dijalankan, program menciptakan object-object yang merupakan instan- instan Class. Class dapat mewarisi Class lain. Java tidak mengijinkan pewarisan jamak akan tetapi menyelesaikan kebutuhan pewarisan jamak dengan fasilitas antarmuka yang lebih elegan.

Perkembangan Java tidak hanya terfokus pada satu sistem operasi, namun dikembangkan untuk berbagai sistem operasi dan bersifat open source. Dengan slogannya “Write once, run anywhere”. Bahasa pemrograman java banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model object yang lebih sederhana. Aplikasi-aplikasi berbasis java pada umumnya dikompilasi ke dalam bentuk p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM).

2.2.3.2 Kelebihan Dari Bahasa Pemrograman JAVA

1. Mudah Untuk Dikembangkan
2. Sifatnya multiplatform
3. Memiliki kemudahan dalam menyusun suatu script
4. Apabila programmer berorientasi pada usability, maka Java
5. sangat mendukung dimana aplikasi yang dibuat dengan Bahasa pemrograman Java dapat bekerja di platform manapun.
6. Bahasa pemrograman yang berorientasi terhadap objek
7. Dinamis, karena Bahasa pemrograman Java sangat mudah untuk dikembangkan.

2.2.3.3 Kekurangan Dari Bahasa Pemrograman JAVA

1. Penggunaan memory yang cukup tinggi, sesuai dengan kelebihan fitur – fitur yang ditawarkan maka kebutuhan memory yang besar.
2. Mudah didekompilasi, dekompilasi ini adalah pengambilan source code. Jadi, Java merupakan salah satu bahasa pemrograman yang mudah mengalami dekompileasi.

2.2.4 Penggunaan Bahasa Pemrograman PHP

PHP adalah bahasa pemrograman script server-side yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum (wikipedia). PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor.

PHP dapat digunakan dengan gratis (free) dan bersifat Open Source. PHP dirilis dalam lisensi PHP License, sedikit berbeda dengan lisensi GNU General Public License (GPL) yang biasa digunakan untuk proyek Open Source.

Kemudahan dan kepopuleran PHP sudah menjadi standar bagi programmer web di seluruh dunia. Menurut wikipedia pada februari 2014, sekitar 82 persen dari web server di dunia menggunakan PHP. PHP juga menjadi dasar dari aplikasi CMS (Content Management System) populer seperti Joomla, Drupal, dan WordPress.

2.2.4.1 Fungsi PHP Dalam Pemrograman Web

Untuk membuat halaman web, sebenarnya PHP bukanlah bahasa pemrograman yang wajib digunakan. Kita bisa saja membuat website hanya menggunakan HTML saja. Web yang dihasilkan dengan HTML (dan CSS) ini dikenal dengan website statis, dimana konten dan halaman web bersifat tetap.

Sebagai perbandingan, website dinamis yang bisa dibuat menggunakan PHP adalah situs web yang bisa menyesuaikan tampilan konten tergantung situasi. Website dinamis juga bisa menyimpan data ke dalam database, membuat halaman yang berubah-ubah sesuai input dari user, memproses form, dll.

Untuk pembuatan web, kode PHP biasanya di sisipkan ke dalam dokumen HTML. Karena fitur inilah PHP disebut juga sebagai Scripting Language atau bahasa pemrograman script.

Sebagai contoh penggunaan PHP, misalkan kita ingin membuat list dari nomor 1 sampai nomor 10. Dengan menggunakan HTML murni, kita bisa membuatnya secara manual seperti kode berikut ini:

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Contoh list dengan HTML</title>
5      </head>
6  <body>
7      <h2>Daftar Absensi Mahasiswa</h2>
8          <ol>
9              <li>Nama Mahasiswa ke-1</li>
10             <li>Nama Mahasiswa ke-2</li>
11             <li>Nama Mahasiswa ke-3</li>
12             <li>Nama Mahasiswa ke-4</li>
13             <li>Nama Mahasiswa ke-5</li>
14             <li>Nama Mahasiswa ke-6</li>
15             <li>Nama Mahasiswa ke-7</li>
16             <li>Nama Mahasiswa ke-8</li>
17             <li>Nama Mahasiswa ke-9</li>
18             <li>Nama Mahasiswa ke-10</li>
19         </ol>
20     </body>
21 </html>
```

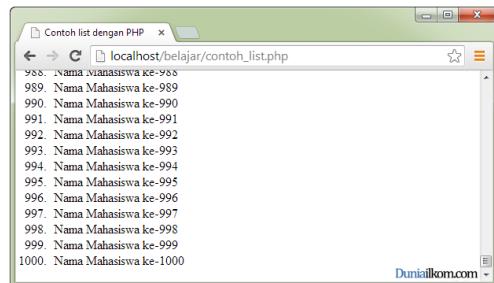
Gambar 2.1 Contoh kode PHP

Halaman HTML tersebut dapat dibuat dengan mudah dengan cara men-copy-paste tag `` sebanyak 10 kali dan mengubah sedikit angka-angka no urut di belakangnya. Namun jika yang kita inginkan adalah menambahkan list tersebut menjadi 100 atau 1000 list, cara copy-paste tersebut menjadi tidak efektif.

Jika menggunakan PHP, kita tinggal membuat perulangan for sebanyak 1000 kali dengan perintah yang lebih singkat seperti berikut ini:

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Contoh list dengan PHP</title>
5      </head>
6  <body>
7      <h2>Daftar Absensi Mahasiswa</h2>
8          <ol>
9              <?php
10                 for ($i = 1; $i <= 1000; $i++)
11                 {
12                     echo "<li>Nama Mahasiswa ke-$i</li>";
13                 }
14             >
15         </ol>
16     </body>
17 </html>
```

Gambar 2.2 Contoh kode PHP untuk perulangan



Gambar 2.3 Contoh hasil dari kode PHP untuk perulangan

Dengan menggunakan kode baris yang bahkan lebih sedikit, kita dapat membuat list tersebut menjadi 1000 kali, bahkan 100.000 kali dengan hanya mengubah sebuah variabel.

PHP tidak hanya dapat melakukan pengulangan tersebut, masih banyak hal lain yang bisa kita lakukan dengan PHP, seperti menginput data ke database, menghasilkan gambar, menkonversi halaman text menjadi PDF, management cookie dan session, dan hal lainnya.

2.2.5 MySql

MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (Structured Query Language) yang cukup terkenal. Database management system (DBMS) MySQL multi pengguna dan multi alur ini sudah dipakai lebih dari 6 juta pengguna di seluruh dunia.



Source: www.worldvectorlogo.com

Gambar 2.4 Logo MySql

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada.

Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis RDBMS (Relational Database Management System). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel.

SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server.

2.2.5.1 Sejarah MySQL MySQL adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia).



*Michael Monty Widenius
(source: www.wikipedia.org)*

Gambar 2.5 Penemu MySql

Sayangnya, UNIREG belum terlalu kompatibel dengan database dinamis yang dipakai di website. TcX kemudian mencari alternatif lain dan menemukan perangkat

lunak yang dikembangkan oleh David Hughes, yaitu miniSQL atau mSQL. Namun, ditemukan masalah lagi karena mSQL tidak mendukung indexing sehingga belum sesuai dengan kebutuhan TcX.

Pada akhirnya muncul kerjasama antara pengembang UNIREG (Michael Monty Widenius), mSQL (David Hughes), dan TcX. Kerjasama ini bertujuan untuk mengembangkan sistem database yang baru, dan pada 1995 dirilislah MySQL seperti yang dikenal saat ini. Saat ini pengembangan MySQL berada di bawah Oracle.

2.2.5.2 Kelebihan MySQL

- Mendukung Integrasi Dengan Bahasa Pemrograman Lain.

Website atau perangkat lunak terkadang dikembangkan dengan menggunakan berbagai macam bahasa pemrograman, jadi Anda tidak perlu khawatir jika menggunakan MySQL. Maka dari itu, MySQL bisa membantu Anda untuk mengembangkan perangkat lunak yang lebih efektif dan tentu saja lebih mudah dengan integrasi antara bahasa pemrograman.

- Tidak Membutuhkan RAM Besar.

MySQL dapat dipasang pada server dengan spesifikasi kecil. Jadi tidak perlu khawatir jika Anda hanya mempunyai server dengan kapasitas 1 GB karena Anda masih bisa menggunakan MySQL sebagai database Anda.

- Mendukung Multi User.

MySQL dapat dipakai oleh beberapa user dalam waktu bersamaan tanpa membuatnya crash atau berhenti bekerja. Ini dapat Anda manfaatkan ketika mengerjakan proyek yang sifatnya tim sehingga seluruh tim dapat bekerja dalam waktu bersamaan tanpa harus menunggu user lain selesai.

- Bersifat Open Source

MySQL adalah sistem manajemen database gratis. Meskipun gratis, bukan berarti database ini mempunyai kinerja buruk. Apalagi lisensi gratis yang dipakai adalah GPL di bawah pengelolaan Oracle sehingga kualitasnya termasuk baik. Selain itu, Anda juga tidak perlu khawatir jika terjadi masalah karena banyak komunitas dan dokumentasi yang membahas soal MySQL.

- Struktur Tabel yang Fleksibel.

MySQL mempunyai struktur tabel yang mudah dipakai dan fleksibel. Contohnya saat MySQL memproses ALTER TABLE dan lain sebagainya. Jika dibandingkan dengan database lain seperti Oracle dan PostgreSQL, MySQL tergolong lebih mudah.

- Tipe Data yang Bervariasi.

Kelebihan lain dari MySQL adalah mendukung berbagai macam data yang bisa Anda gunakan di MySQL. Contohnya float, integer, date, char, text, timestamp,

double, dan lain sebagainya. Jadi manajemen database sistem ini sangat membantu Anda untuk mengembangkan perangkat lunak yang berguna untuk penelitian database di server.

- Keamanan yang Terjamin.

Open source bukan berarti MySQL menyediakan keamanan yang buruk. Malah sebaliknya, MySQL mempunyai fitur keamanan yang cukup apik. Ada beberapa lapisan keamanan yang diterapkan oleh MySQL, seperti level nama host, dan subnetmask. Selain itu MySQL juga dapat mengatur hak akses user dengan enkripsi password tingkat tinggi.

2.2.5.3 Kekurangan MySQL

- Kurang Cocok untuk Aplikasi Game dan Mobile

Anda yang ingin mengembangkan aplikasi game atau perangkat mobile ada baiknya jika mempertimbangkan lagi jika ingin menggunakan MySQL. Kebanyakan pengembang game maupun aplikasi mobile tidak menggunakan MySQL karena memang database manajemen sistem ini masih kurang bagus dipakai untuk sistem aplikasi tersebut.

- Sulit Mengelola Database yang Besar

Jika Anda ingin mengembangkan aplikasi atau sistem di perusahaan dengan database yang cukup besar, ada baiknya jika menggunakan database manajemen sistem selain MySQL. MySQL dikembangkan supaya ramah dengan perangkat yang mempunyai spesifikasi rendah, itulah mengapa MySQL tidak memiliki fitur yang lengkap seperti aplikasi lainnya

- Technical Support yang Kurang Bagus

Sifatnya yang open source terkadang membuat aplikasi tidak menyediakan technical support yang memadai. Technical support MySQL diklaim kurang bagus. Hal ini membuat pengguna kesulitan. Apalagi jika pengguna mengalami masalah yang berhubungan dengan pengoperasian perangkat lunak tersebut dan membutuhkan bantuan technical support.

2.2.5.4 Mengenal SQL

- SQL merupakan bahasa pemrograman yang perlu Anda pahami karena dapat merelasikan antara beberapa tabel dengan database maupun antar database. Ada tiga bentuk SQL yang perlu Anda ketahui, yaitu Data Definition Language (DDL), Data Manipulation Language(DML), dan Data Control Language (DCL).

1. Data Definition Language (DDL)

DDL berguna pada saat Anda ingin mendefinisikan data di dalam database. Terdapat beberapa query yang dikelompokkan ke dalam DDL, yaitu:

Argumen DDL di atas perlu Anda pahami karena merupakan dasar penggunaan SQL di bagian awal pembuatan database. Contohnya saja jika belum menjalankan perintah “CREATE”, Anda belum bisa melanjutkan penggunaan argumen yang lainnya.

2. Data Manipulation Language (DML)

DML dapat Anda pakai setelah menjalankan perintah DDL. DML berfungsi untuk memanipulasi, mengubah, atau mengganti isi dari database (tabel) yang sudah ada.

3. Data Control Language (DCL)

Jika Anda sudah mempunyai user dan ingin mengatur hak akses masing-masing user, Anda sebaiknya memahami berbagai macam jenis DCL dan cara penggunaannya. DCL berguna untuk memberikan hak akses database, mendefinisikan space, mengalokasikan space, dan melakukan audit penggunaan database.

2.2.5.5 Cara Menginstall MySQL di Windows dan Linux MySQL server mendukung berbagai macam sistem operasi (cross-platform), khususnya Linux dan Windows. Proses instalasinya pun sangat mudah. Jika menggunakan Windows, Anda bisa menginstall XAMPP untuk menjalankan MySQL server yang di dalamnya sudah terdapat juga modul untuk menjalankan Apache, PHP, FileZilla, dan Tomcat.

Sedangkan di Linux, Anda dapat menginstall MySQL secara terpisah atau menginstall LAMP (Linux, Apache, MySQL, PHP) yang sudah ada modul Apache dan PHP juga. Selain itu, XAMPP juga sudah tersedia di Linux. Jadi Anda bisa menyesuaikan proses instalasi sesuai dengan Anda nyaman dan suka.

2.2.5.6 Cara Install MySQL Server di Windows

– Langkah 1. Unduh file instalasi XAMPP

Anda cukup mengunduh file instalasi XAMPP melalui halaman website apachefriends.org.

– Langkah 2. Install XAMPP

Setelah selesai proses unduh, silakan eksekusi file instalasi XAMPP untuk menjalankan proses instalasi. Kemudian ikuti instruksi yang ada.

– Langkah 3. Aktifkan MySQL server menggunakan GUI XAMPP

Saat Anda selesai menjalankan proses instalasi akan muncul panel kontrol XAMPP. Klik “Start” pada modul Apache dan MySQL. Sampai di sini MySQL server sudah dapat berjalan dan sudah dapat dipakai untuk menjalankan perintah SQL.

2.2.5.7 Cara Install MySQL Server di Linux

- Langkah 1. Install Server Web Apache

Sebelum Anda menjalankan proses instalasi, ada baiknya untuk memperbarui repositorinya terlebih dahulu. Untuk memperbarui repositori pada Linux caranya cukup mudah, Anda tinggal membuka Terminal kemudian mengetikkan baris komentar berikut:

```
sudo apt-get update
```

Kemudian jalankan perintah berikut untuk menginstall Apache.

```
sudo apt-get install apache2
```

- Langkah 2. Mengecek Proses Instalasi Apache

Jika proses instalasi Apache sudah selesai dan berhasil dijalankan, Anda bisa mengeceknya dengan mengakses web browser kemudian menuliskan alamat IP server atau alamat domain.

```
http://alamat-ip-address-server/
```

- Langkah 3. Instalasi MySQL

Proses instalasi MySQL server di Linux cukup menggunakan satu perintah saja.

```
sudo apt-get install mysql-server
```

Proses instalasi ini akan meminta Anda untuk memasukkan username dan password untuk akses ke dalam MySQL server.

- Langkah 4. Pengecekan Instalasi MySQL

Anda dapat memastikan apakah proses instalasi MySQL sudah berjalan baik atau belum dengan mengetikkan perintah di bawah ini.

```
sudo systemctl status mysql
```

Perintah di atas untuk mengecek apakah MySQL sudah berjalan atau belum. Jika belum, Terminal akan menampilkan status ‘Error’ atau ‘Failed’.

2.2.6 Algoritma

2.2.7 Analisi Sistem Berjalan

Menganalisis suatu sistem yang sedang berjalan merupakan salah satu tahap untuk manganalisis suatu sistem akankah sesuai dengan tujuan utama sistem itu sendiri yaitu mempermudah user sistem.

Dalam menganalisis sistem dapat kita mulai dari menganalisis sistem yang sedang berjalan. Dimana saat kita telah menganalisis sistem yang berjalan maka kita dapat menemui kekurangan dari sistem tersebut sehingga kita dapat mensempurnakan sistem agar sistem lebih efektif dan efisien yang tentunya menjadi sebuah sistem yang user friendly atau ramah pengguna.

Menanalisis Sistem berjalan selain terjun langsung untuk dapat mengetahui dan membuat flowmap sebagai gambaran mengenai proses bisnis yang sedang berjalan.

2.2.7.1 Pengertian Proses Bisnis Proses Bisnis sendiri ialah suatu kumpulan dari aktivitas (task) atau pekerjaan terstruktur yang saling berhubungan untuk menyelesaikan suatu masalah tertentu atau yang menghasilkan suatu keluaran (produk/output) atau layanan (demi meraih tujuan tertentu) dan mendukung pencapaian tujuan serta sasaran strategis dari suatu organisasi.

Suatu proses bisnis yang baik harus mempunyai tujuan mengefektifkan, mengefisienkan dan meningkatkan produktifitas dari suatu organisasi dimana jika suatu organisasi secara umum tentu saja mempunyai tujuan agar dapat bertahan hidup selama mungkin dan selalu meningkat kinerja serta produktifitasnya. Untuk mencapai tujuan tersebut, maka suatu organisasi membutuhkan suatu proses bisnis yang baik untuk mendukung berjalannya organisasi tersebut.

Tidak terkecuali suatu perusahaan sebagai suatu organisasi tentu saja membutuhkan suatu proses bisnis yang baik agar dapat mendukung pencapaian tujuan dari perusahaan tersebut.

2.2.7.2 Pengertian Flowmap Flowmap adalah campuran peta dan flow chart,yang menunjukan pergerakan benda dari satu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. Flowmap menolong analisis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengopresian.

▪ PEDOMAN-PEDOMAN DALAM MEMBUAT FLOWMAP

Bila seorang analis dan programmer akan membuat flowmap, ada beberapa petunjuk yang harus diperhatikan, seperti :

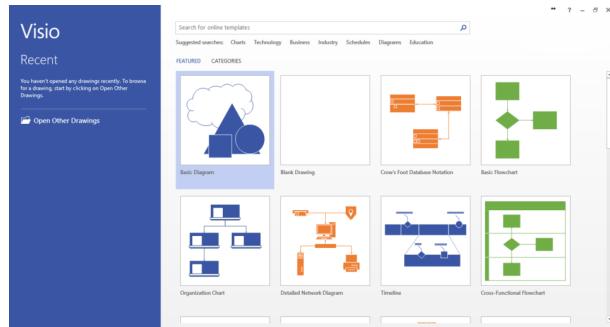
1. Flowmap digambarkan dari halaman atas ke bawah dan kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat di mengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus di tentukan secara jelas.
4. Setiap langkah dari aktivitas harus berada pada urutan yang benar.

5. Lingkup dan range dari aktifitas yang sedang di gambarkan harus di telusuri dengan hati-hati.
 6. Gunakan simbol-simbol flowchart yang standar agar dapat dimengerti.
- Simbol-Simbol Flowmap

Simbol	Keterangan
	Simbol dokumen Menunjukkan dokument <i>input</i> dan <i>output</i> .
	Simbol kegiatan manual Menunjukkan kegiatan atau pekerjaan manual
	Simbol proses Menunjukkan kegiatan proses dari operasi program komputer.
	Simbol keyboard Menunjukkan <i>input</i> yang menggunakan <i>on-line keyboard</i> .
	Simbol harddisk Menunjukkan <i>input</i> ataupun <i>output</i> menggunakan <i>harddisk</i> .
	Simbol garis alir Menunjukkan arus dari setiap proses.
	Simbol penghubung Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
	Simbol arsip Menunjukkan pengarsipan <i>file</i> tanpa menggunakan komputer.
	Simbol keputusan Digunakan untuk suatu penyeleksian kondisi di dalam program.

Gambar 2.6 Simbol Flowmap

- Tools yang Dapat Digunakan Dalam Membuat Flowmap salah satunya ialah Microsoft Visio



Gambar 2.7 Tambilan Awal Visio

Dari tampilan tersebut kita dapat memilih model Flowmao pada kanan bawah dengan warna hijau, Flowmap pada Microsoft Visio dapat kita atur warna background, jenis tulisan, ukuran tulisan dan vertikal maupun horizontal dari flowmap tersebut.

Jangan takut dengan bagaimana simbol flowmap pada Microsoft Visio karena sudah disediakan simbol simbol flowmap kita hanya tinggal memanfaatkan simbol tersebut bukan hanya simbol bahkan conector flowmap pada Microscos visio sangat mudah untuk diaplikasikan.

2.2.8 Analisis Sistem Yang akan Dibangun

Dalam menganalisi Sistem yang akan dibangun kita dapat menggunakan UML atau Unified Modeling Language. Design system tidak kalah penting untuk membuat sebuah program, tidak jarang perusahaan - perusahaan besar membutuhkan design system untuk project/program yang mereka buat.

UML (Unified Modeling Language) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOADD/object oriented analysis and design) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an. UML merupakan gabungan dari metode Booch,Rumbaugh (OMT) dan Jacobson. Tetapi UML mencakup lebih luas daripada OOAD. Pada pertengahan saat pengembangan UML, dilakukan standarisasi proses dengan OMG(Object Management Group) dengan harapan UML bakal menjadi bahasa standar pemodelan pada masa yang akan datang (yang sekarang sudah banyak dipakai oleh berbagai kalangan).

UML adalah bahasa untuk menspesifikasi,memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak)dari sistem perangkat lunak,seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object.UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corps. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai prespektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

UML sendiri memiliki kegunaan yaitu untuk untuk memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi object. Dan juga untuk menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin. **UML digunakan paling efektif pada domain seperti :**

- Sistem Informasi Perusahaan
- Sistem Perbankan dan Perekonomian
- Bidang Telekomunikasi
- Bidang Transportasi
- Bidang Penerbangan
- Bidang Perdagangan
- Bidang Pelayanan Elektronik
- Bidang Pengetahuan
- Bidang Pelayanan Berbasis Web Terdistribusi

Bagian-bagian UML

1. View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. Beberapa Jenis view dalam UML antara lain : use case view,logical view,component view,concurrency view, dan deployment view.
2. Use case View Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan external actors. Actor yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya.
3. Logical View Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (class, object, dan relationship) dan kolaborasi dinamis yang terjadi ketika object mengirim pesan ke object lain dalam suatu fungsi tertentu. View ini digambarkan dalam class diagrams untuk struktur statis dan dalam state, sequence, collaboration, dan activity diagram untuk model dinamisnya. View ini digunakan untuk perancang (designer) dan pengembang (developer).
4. Component View Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari code module diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administrative lainnya. View ini digambarkan dalam component view dan digunakan untuk pengembang (developer).
5. Concurrency View Membagi sistem ke dalam proses dan prosesor. View ini digambarkan dalam diagram dinamis (state, sequence, collaboration, dan activity diagrams) dan diagram implementasi (component dan deployment diagrams) serta digunakan untuk pengembang (developer), pengintegrasи (integrator), dan penguji (tester).
6. Deployment View Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (nodes) dan bagaimana hubungannya dengan lainnya. View ini digambarkan dalam deployment diagrams dan digunakan untuk pengembang (developer), pengintegrasи (integrator), dan penguji (tester).

Diagram

1. 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 digambaran hanya yang dilihat dari luar oleh actor (keadaan lingkungan sistem yang dilihat user) dan bukan bagaimana fungsi yang ada di dalam sistem.

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor 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> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

Gambar 2.8 Simbol Usecase

2. Class Diagram Menggambarkan struktur statis class di dalam sistem. Class merepresentasikan sesuatu yang ditangani oleh sistem. Class dapat berhubungan dengan yang lain melalui berbagai cara: associated (terhubung satu sama lain), dependent (satu class tergantung/menggunakan class yang lain), specialed (satu class merupakan spesialisasi dari class lainnya), atau package (grup bersama sebagai satu unit). Sebuah sistem biasanya mempunyai beberapa class diagram. Class memiliki tiga area pokok yaitu Nama (dan stereotype), Atribut, Metoda dan juga terdapat Atribut dan metoda dapat yang memiliki salah satu sifat yaitu Private , tidak dapat dipanggil dari luar class yang bersangkutan, Protected , hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya, Public , dapat dipanggil oleh siapa saja

Daftar Simbol

NO	GAMBAR	NAMA	KETERANGAN
1	—	Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2	◇	Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3	[]	Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4	()	Collaboration	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5	↔	Realization	Operasi yang benar-benar dilakukan oleh suatu objek.
6	----->	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7	—	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya

Gambar 2.9 Simbol Class Diagram

3. State Diagram Menggambarkan semua state (kondisi) yang dimiliki oleh suatu object dari suatu class dan keadaan yang menyebabkan state berubah. Kejadian dapat berupa object lain yang mengirim pesan. State class tidak digambarkan untuk semua class, hanya yang mempunyai sejumlah state yang terdefinisi dengan baik dan kondisi class berubah oleh state yang berbeda.

NO	GAMBAR	NAMA	KETERANGAN
1	□	Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2	○	Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3	●	Initial Node	Bagaimana objek dibentuk atau diawali.
4	◎	Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5	◇	Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6	↓ ↔ →	Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

Gambar 2.10 Simbol State Diagram

4. Sequence Diagram Menggambarkan kolaborasi dinamis antara sejumlah object. Kegunaanya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

a. An Actor



Menggambarkan orang yang sedang berinteraksi dengan sistem

b. Entity Class



Menggambarkan hubungan kegiatan yang akan dilakukan

c. Boundary Class



Menggambarkan sebuah penggambaran dari form

d. Control Class



Menggambarkan penghubung antara boundary dengan tabel

e. A focus Of Control & A life line



Menggambarkan tempat mulai dan berakhirnya sebuah message

f. A message



Menggambarkan Pengiriman Pesan

Gambar 2.11 Simbol Sequence Diagram

5. Collaboration Diagram Menggambarkan kolaborasi dinamis seperti sequence diagrams. Dalam menunjukkan pertukaran pesan, collaboration diagrams menggambarkan object dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan gunakan sequence diagrams, tapi jika penekanannya pada konteks gunakan collaboration diagram.

Simbol Collaboration Diagram

Simbol	Keterangan
	Actor
	Object instance : Objek yang dibuat, melakukan tindakan, dan / atau dimusnahkan selama lifeline
	Interaksi link : Merupakan indikasi bahwa obyek kejadian dan berkolaborasi aktor dan pertukaran pesan.

Gambar 2.12 Simbol Collaboration Diagram

6. Activity Diagram Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

Simbol	Keterangan
	Titik awal atau permulaan
	Titik akhir atau akhir dari aktivitas
	Aktivitas yang dilakukan oleh aktor
	Decision atau pilihan untuk mengambil keputusan
	Arah tanda panah alur Proses

Gambar 2.13 Simbol Activity Diagram

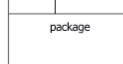
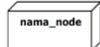
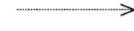
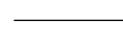
7. Component Diagram Menggambarkan struktur fisik kode dari komponen. Komponen dapat berupa source code, komponen biner, atau executable component.

Sebuah komponen berisi informasi tentang logic class atau class yang diimplementasikan sehingga membuat pemetaan dari logical view ke component view.

no	Simbol	Nama simbol	deskripsi
1		Package	Package merupakan simbol bungkus dari satu atau lebih komponen
2		Komponen	Komponen sistem
3		Dependency / kebergantungan	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
4		Interface / antar muka	Sama dengan konsep interface pada pemrograman berorientasi objek, yaitu sebagai antar muka komponen agar tidak mengakses komponen langsung
5		Link	Relasi antar komponen

Gambar 2.14 Simbol Component Diagram

8. Deployment Diagram Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat (nodes) satu sama lain dan jenis hubungannya. Di dalam nodes, executeable component dan object yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh node tertentu dan ketergantungan komponen.

Simbol	Deskripsi
Package  package	package merupakan sebuah bungkus dari satu atau lebih node
Node 	biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam node disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
Kebergantungan / dependency 	Kebergantungan antar node, arah panah mengarah pada node yang dipakai
Link 	relasi antar node

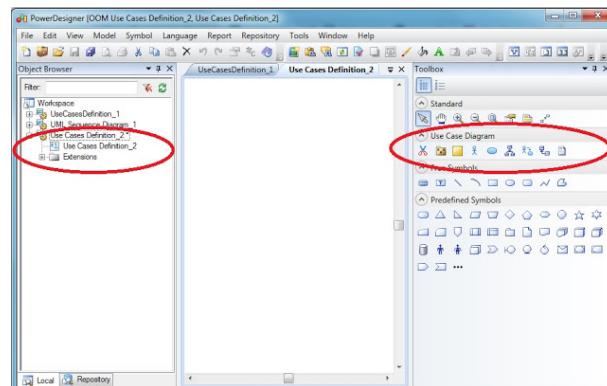
Gambar 2.15 Simbol Deployment Diagram

Tools yang Digunakan dalam Membuat Diagram Pada UML

Tools yang dapat kita gunakan ialah Power Desainer dan Enterprise Architecture (EA).

▪ Power Desainer

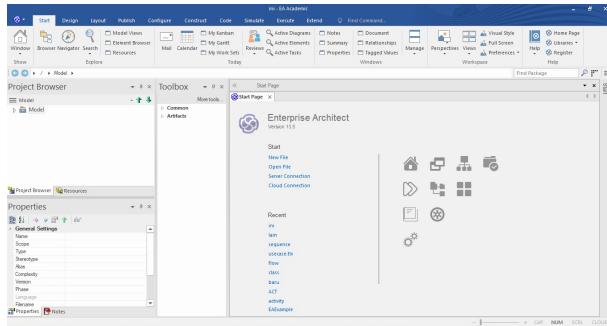
Power Designer adalah salah satu tools yang dapat dipergunakan untuk membangun atau merancang sebuah basisdata melalui ER-diagram, merancang sistem melalui Data Flow Diagram (DFD) serta mampu membuat program aplikasi. Dalam Power Designer, ada beberapa macam pemodelan data yang digunakan untuk perancangan basis data diantaranya pertama, Conceptual Data Model (CDM) adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (entity) serta hubungan (relationship) antara entitas-entitas itu. Sementara itu, kedua, Physical Data Model (PDM) yaitu model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Selain dapat digunakan dalam perancangan database Power Designer dapat digunakan menjadi tool mudah dalam membuat UML seperti berikut tampilannya



Gambar 2.16 Tampilan dari Power Desainer

- Enterprise Architecture (EA)

Enterprise Architecture (EA) adalah pembelajaran yang dilakukan untuk menentukan analisis, desain, perencanaan, dan pelaksanaan bagi suatu perusahaan, hal ini dilakukan untuk keberhasilan pengembangan dan pelaksanaan strategi. Seorang EA menerapkan prinsip arsitek dan belajar untuk memimpin organisasi melalui bisnis, informasi, proses dan perubahan teknologi yang berperan penting untuk melaksanakan strategi yang telah direncanakan dan menjadi salah satu tools yang sangat lengkap dalam membuat UML seberti berikut tampilannya



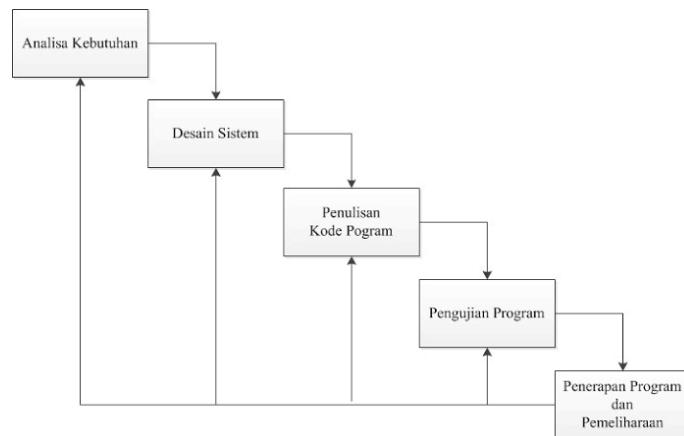
Gambar 2.17 Tampilan dari Enterprise Architecture (EA)

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metode Waterfall

Metode yang digunakan dalam pengembangan sistem Inventory menggunakan qr kode adalah metode waterfall. Alasan menggunakan metode ini adalah karena metode waterfall melakukan pendekatan secara sistematis dan berurutan dalam membangun suatu sistem. Proses metode waterfall yaitu pada pelaksanaannya secara bertahap sehingga tidak terfokus pada tahapan tertentu. Tahapan dari metode waterfall adalah:



Gambar 3.1 Tahap Metode Waterfall

1. Analisis Kebutuhan

Tahap analisis kebutuhan dilakukan dengan menganalisa kebutuhan user, analisa perangkat lunak dan perangkat keras yang dibutuhkan dalam pengembangan sistem serta kebutuhan lain dalam pembuatan basis data. Analisis kebutuhan perangkat keras pada sistem ini yaitu laptop dengan spesifikasi sebagai berikut Toshiba satellite L745, processor intel core i5, RAM DDR2 2 GB. Analisis kebutuhan perangkat lunak yang membantu pembuatan sistem ini yaitu Acer Z476-31TB, processor intel core i3, RAM DDR3L 4 GB. Analisis kebutuhan perangkat lunak yang membantu pembuatan sistem ini yaitu Sistem Operasi Windows 10, XAMPP versi 1.7.3, Android Studio, Bahasa Pemrograman Java, Mozilla firefox 9.0, dan MySQL Server 32 Bit.

2. Desain Sistem

Tahap selanjutnya yaitu mendesain sistem. Tahap ini dibuat sebelum tahap pengkodean. Tujuan dari tahap ini adalah memberikan gambaran tentang apa yang akan dikerjakan dan bagaimana tampilannya. Tahap ini memenuhi semua kebutuhan pengguna sesuai dengan hasil yang dianalisa seperti rancangan tampilan pengembangan sistem dan membantu mendefinisikan arsitektur sistem secara keseluruhan. Dokumentasi yang dihasilkan dari tahap desain sistem ini antara lain perancangan Use Case Diagram, Activity Diagram, Sequences Diagram, Class Diagram, dan perancangan interface.

3. Coding (Penulisan Kode Program)

Aktivitas pada tahap ini dilakukan pengkodean sistem. Penulisan kode program merupakan tahap penerjemahan desain sistem yang telah dibuat ke dalam bentuk perintah-perintah yang dimengerti komputer dengan mempergunakan bahasa pemrograman. Tahapan ini merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Sistem ini bahasa pemrograman yang dipakai adalah Java dan database MySQL.

4. Testing (Pengujian Program)

Pengujian dilakukan untuk memastikan bahwa software yang dibuat telah sesuai dengan desainnya dan semua fungsi dapat dipergunakan dengan baik tanpa ada kesalahan.

5. Operation and Maintenance (Pemeliharaan Program)

Tahap ini merupakan tahap terakhir dalam metode waterfall. Sistem dapat diimplementasikan. Pemeliharaan mencakup koreksi dari berbagai error yang tidak ditemukan pada tahap-tahap terdahulu, perbaikan atas implementasi dan pengembangan unit sistem, serta pemeliharaan program. Pemeliharaan sistem dapat dilakukan oleh seorang administrator untuk meningkatkan kualitas sistem agar jauh lebih baik.

3.2 Analysis Sistem

Analisis sistem adalah dilakukannya proses penilaian, identifikasi dan evaluasi komponen dan hubungan timbal-balik yang terkait dalam pengembangan sistem; definisi masalah, tujuan, kebutuhan, prioritas serta kendala-kendala sistem yang sedang berjalan.

Langkah pertama dalam mendesain sebuah sistem adalah menganalisis sistem yang sedang berjalan. Dalam hal ini kami menganalisis sistem inventory barang pada Sistem Inventory Barang Manual sebagai langkah dan tujuan utama untuk mengetahui bagaimana proses bisnis yg terjadi pada Sistem Inventory Pendataan Manual.

Mengetahui proses bisnis pada sebuah sistem dapat dilakukan dengan beberapa metode salah satunya seperti kuisioner maupun wawancara, hal ini dilakukan agar mendapatkan sebuah informasi yang valid. Mengetahui proeses bisnis adalah mengetahui bagai mana sistem yang ada berjalan, bagaimana sistem yang ada dilaksanakan, serta mengetahui apa kekurangan sistem yang ada ataupun apa masalah yang terjadi dengan sistem tersebut.

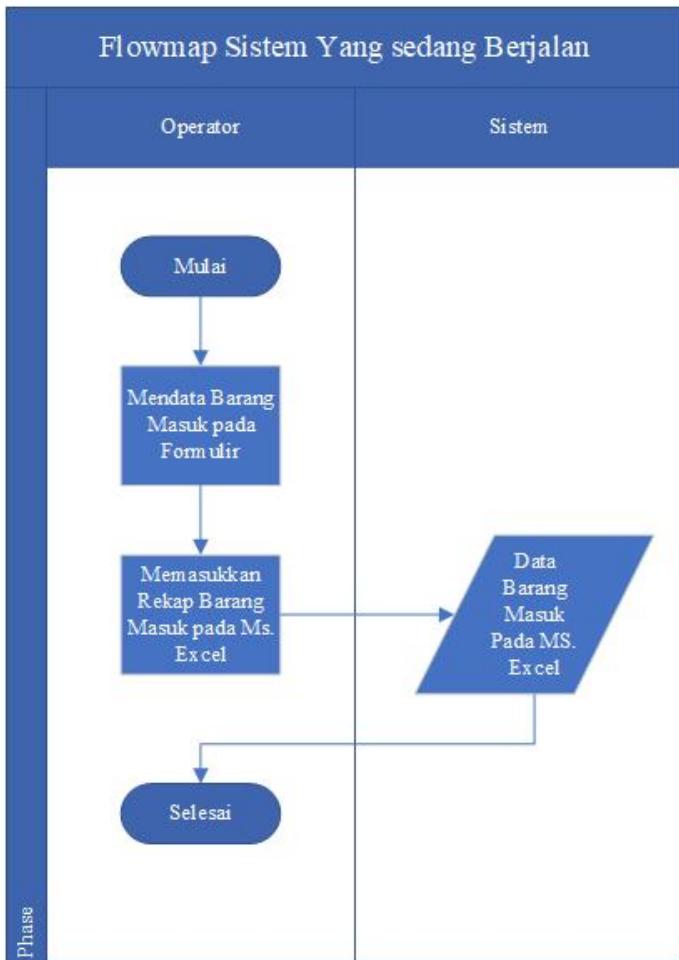
Analisis sistem dilakukan bukan hanya pada sistem yang sedang berjalan melainkan juga pada sebuah sistem yang akan dibangun. Setelah analisis dilakukan pada sistem yang berjalan maka dapat dilanjutkan dengan analisis sistem yang akan dibangun. Kita telah mendapatkan apa pada sistem yang sedang berjalan yang telah dirangkum pada flowmap sebagai langkah mudah dalam memahami proses bisnis pada sistem berjalan.

Analisis pada proses bisnis sistem ideal yang akan dibangun dapat dijabarkan menggunakan flowmap bagi mana kita mendesain sistem yang akan dibangung analisis kebutuhan dan dapat dilanjutkan dengan perancangan Sistem.

3.2.1 Sistem yang sedang Berjalan

Sistem yang sedang berjalan gambaran sistem yang berjalan adalah pengecekan barang secara manual dan dilanjutkannya dengan rekapitulasi data menggunakan *Microsoft Excel*. Untuk dapat lebih mudah dalam memahami sistem yang sedang berjalan kita dapat merangkum segala proses bisnis yang ada dengan menggunakan flowmap. Adapun flowmap dari sistem yang berjalan sebagai berikut:

1. Operator memulai mendaftar barang masuk dengan mengambil form data barang masuk yang telah disediakan.
2. Operator mendaftar barang masuk dengan keterangan sesuai dengan form data barang masuk.
3. Hasil data rekapitulasi dari form di inputkan pada *Microsoft Excel*.
4. Data barang dapat dilihat dari rekappitulasi pada *Microsoft Excel*.

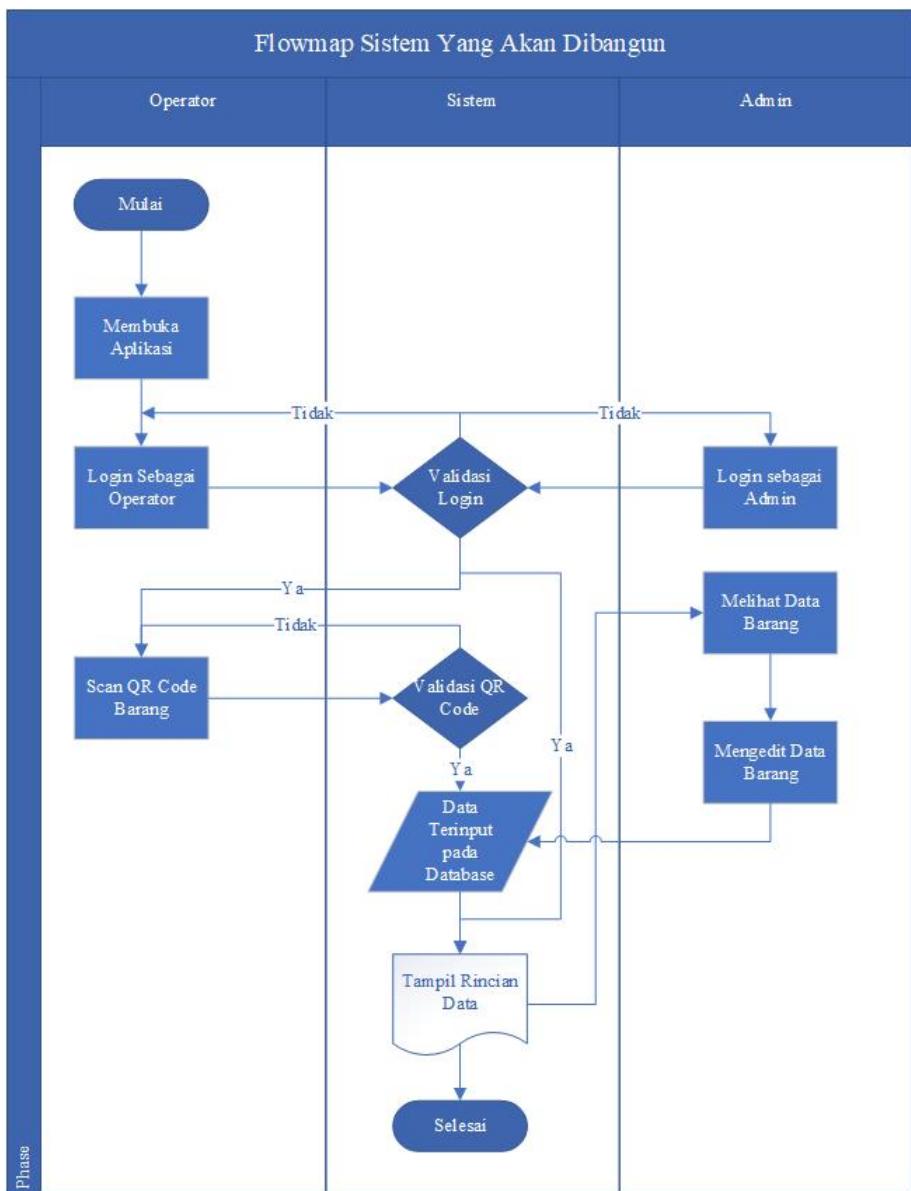


Gambar 3.2 Flowmap Sistem yang Sedang Berjalan

3.2.2 Sistem Yang Akan Dibangun

Dalam aplikasi Sistem Inventory Menggunakan QR Code ini terdapat 2 aktor yaitu Operator dan admin dimana, pegawai bagian pendataan barang masuk sebagai operator yang akan menscan QR Code pada barang secara otomatis dan data akan terinput pada database Inventory barang masuk sedangkan, admin dapat melihat maupun mengedit barang pada data barang yang ada. Adapun flowmap dari sistem yang akan kami bangun ialah:

1. Operator membuka aplikasi *scan QR Code*, login sebagai operator.
2. Untuk admin dapat login sebagai admin.
3. Operator menscan *QR Code* barang.
4. jika scan berhasil data barang akan terinput , jika belum maka scan ulang *QR Code* barang lagi.
5. Data barang masuk telah terinput dalam Database sistem.
6. Sistem akan menampilkan rincian data yang telah terinput.
7. Sedangkan Admin Setelah berhasil login sebagai admin akan masuk pada tampilan data barang dimana admin dapat melihat data barang yang ada maupun mengedit barang yang telah ada agar sesuai.



Gambar 3.3 Flowmap Sistem Yang Akan Dibangun

3.2.2.1 Kebutuhan Fungsional Kebutuhan fungsional adalah kebutuhan yang berhubungan dengan sistem yang berjalan. Dimana akan menjabarkan tentang fungsi-fungsi yang akan mendukung jalannya sistem, adapun kebutuhan fungsional yang akan dibuat yaitu sebagai berikut:

1. Scan QR Code oleh User
2. Penginputan data otomatis oleh sistem

3.2.2.2 Kebutuhan Non-Fungsional Analisis kebutuhan perangkat lunak dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras/hardware dan analisis perangkat lunak/software.

3.2.2.3 Kebutuhan Perangkat Keras dan Perangkat Lunak yang Digunakan

1. Berikut adalah spesifikasi Kebutuhan Perangkat Keras yang digunakan pada tabel 3.1.

Tabel 3.1 Deskripsi perangkat *User*

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Harddisk</i>	1 TB	Media untuk menyimpan data aplikasi yang dibuat
2	<i>Memory</i>	4 GB	<i>Memory System</i> yang digunakan
3	<i>Processor</i>	<i>Intel® core™ i3-6006U</i> <i>CPU @ 2,00 Ghz</i> <i>1.99 Ghz</i>	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada kecepatan prosesor komputer
4	Infrastruktur Jaringan		Bisa dianalogikan sebagai alur proses dari titik awal proses sampai pada akhir proses

2. Berikut adalah spesifikasi Kebutuhan Perangkat Lunak yang digunakan pada tabel 3.2.

Tabel 3.2 Deskripsi perangkat Lunak *User*

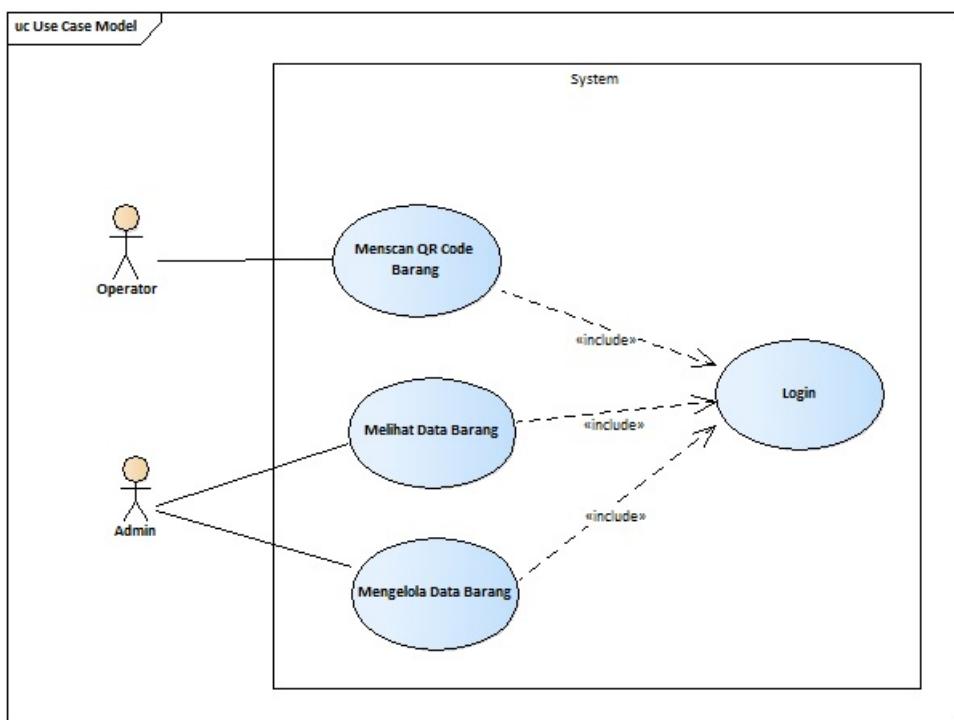
No	Tools	Fungsi	Keterangan
1	<i>Windows 10</i>	Sistem Operasi	-
3	<i>Android Studio</i>	Text editor	-
4	<i>Java PHP</i>	<i>Bahasa Program</i>	-
5	<i>MySql</i>	server basis data	-
6	<i>google chrome</i>	<i>browser</i>	-

3.3 Perancangan

Perancangan adalah merancang dan mendesain sebuah aplikasi ataupun sistem dengan baik yang isinya adalah langkah-langkah operasi dalam proses pengolahan data dan proses prosedur-prosedur untuk mendukung operasi sistem tersebut. Pada perancangan kali ini menggunakan UML atau Unified Modeling Language. Berikut diagram dari UML yang telah kita buat.

3.3.1 Use Case

Use Case Diagram merupakan teknik dalam pengembangan software maupun sistem informasi dengan tujuan untuk mengetahui kebutuhan fungsional dari sistem yang bersangkutan. Use case menjelaskan mengenai interaksi antara actor dan sistem dan dapat menjelaskan tipe interaksi antara actor dan sistem. Di bawah ini adalah pemodelan Use Case Diagram yang telah kami buat:



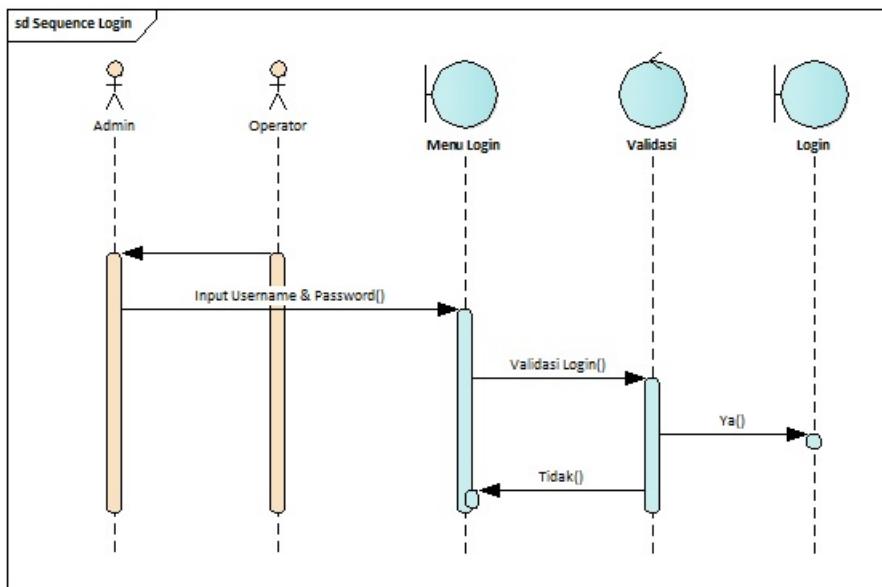
Gambar 3.4 Use Case Diagram

Use Case diagram tersebut terdapat dua aktor yaitu operator ialah pegawai yang menscan data dan admin sebagai pemantau dari sistem *inventory* yang dapat mengedit dan melihat data. Dari semua aktifitas tersebut hanya dapat diakses jika operator dan

admin berhasil login, hal ini dijelaskan dengan adanya include login pada gambar *use case* diatas.

3.3.1.1 Sequence Diagram Sequence Diagram merupakan interaksi antar objek pada urutan dalam sistem yang menjelaskan tahap demi tahap yang seharusnya terjadi. Berikut ini adalah beberapa sequence diagram dari aplikasi Sistem Inventory Barang Menggunakan QR Code.

1. Sequence Diagram: Login

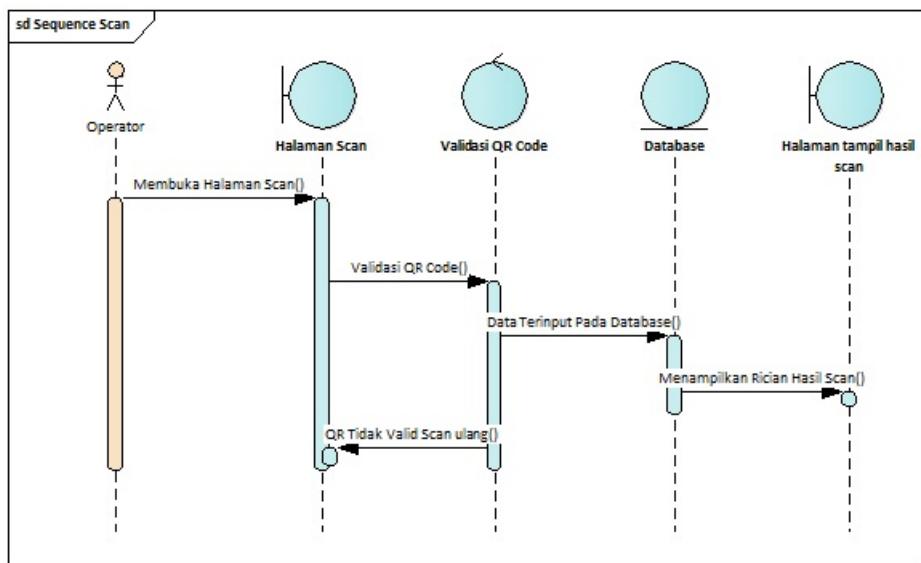


Gambar 3.5 Sequence Diagram: Login

Keterangan :

- User masuk ke form login.
- System meminta user menginputkan Username dan Password
- User menginputkan Username dan password / membuat akun terlebih dahulu jika belum memiliki Akun.
- Melakukan Validasi data dengan database.
- Login berhasil (Masuk ke halaman utama).

2. Sequence Diagram: Scan

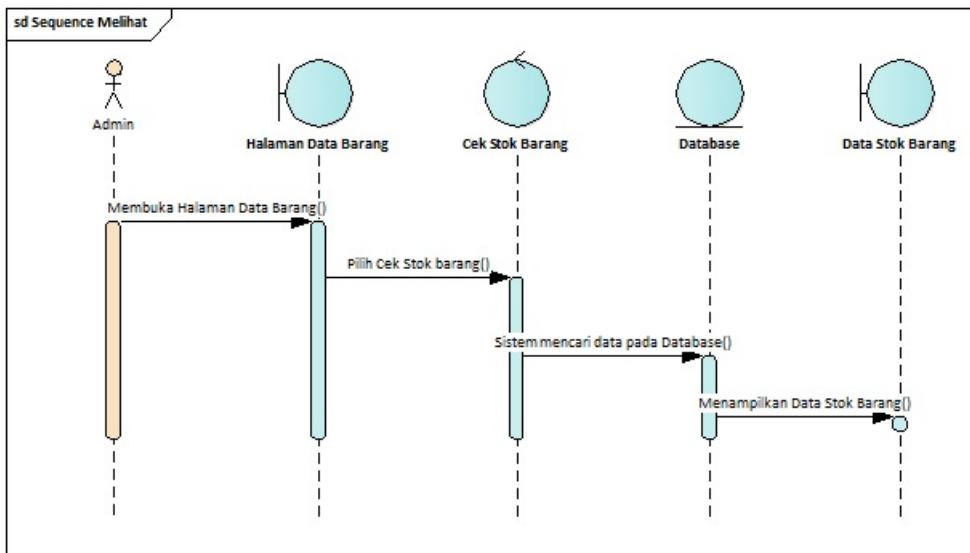


Gambar 3.6 Sequence Diagram:Scan

Keterangan :

- User masuk pada halaman utama.
- User memilih button Scan.
- Melakukan proses scan qr code.
- Data Barang masuk tersimpan pada database.
- Selesai.

3. Sequence Diagram: Melihat Data

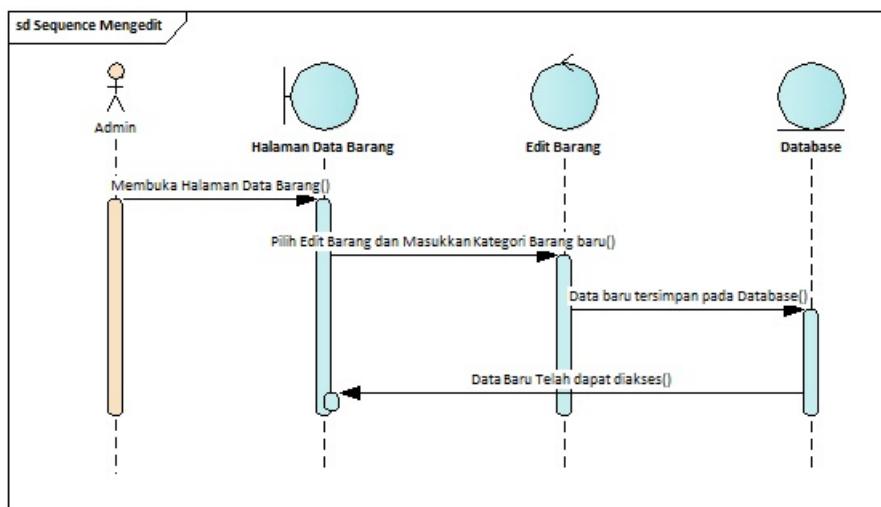


Gambar 3.7 Sequence Diagram: Melihat Data

Keterangan :

- (a) User Membuka Halaman Data Barang.
- (b) User Pilih Cek Stok Barang.
- (c) Sistem Mencari data pada database
- (d) Sistem Menampilkan Data Stok Barang.

4. Sequence Diagram: Mengedit Data



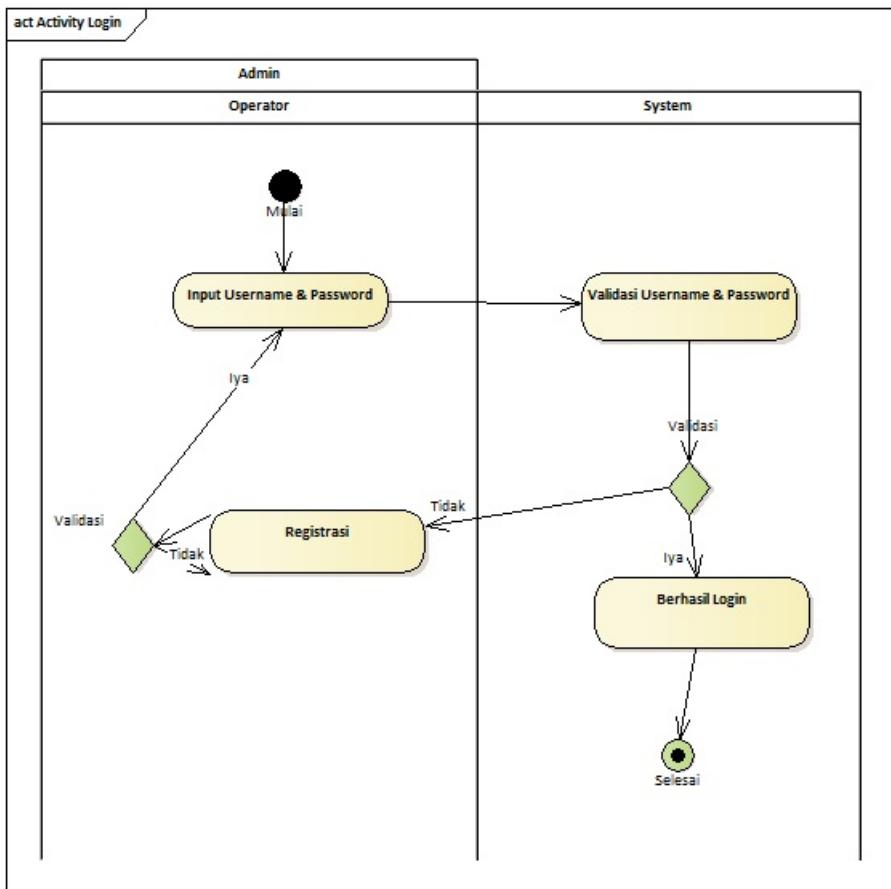
Gambar 3.8 Sequence Diagram:Mengedit

Keterangan :

1. User membuka Halaman Data Barang
2. User Memilih Edit Barang dan Masukkan Kategori Barang Baru.
3. Data baru tersimpan pada database.
4. Data baru dapat diakses.

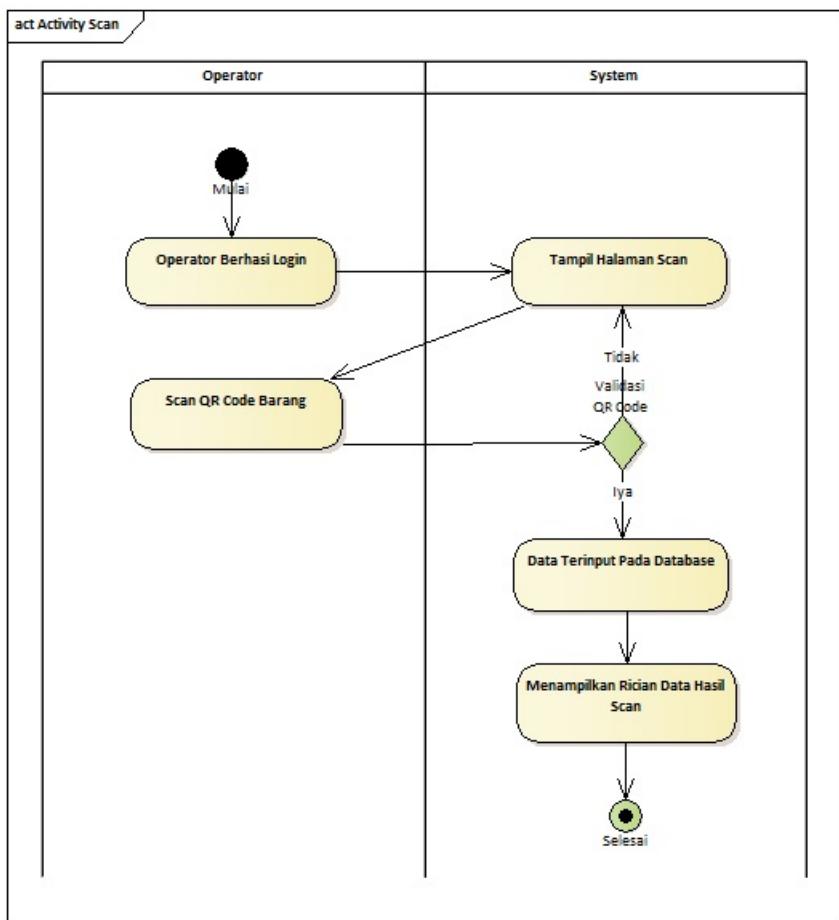
3.3.1.2 Activity Diagram Activity Diagram adalah Diagram penggambaran aliran aktivitas sebuah system dan sebagai deskripsi aktivitas yang dibentuk dalam suatu operasi. Berikut ini adalah beberapa gambar activity diagram dari aplikasi Sistem Inventory Barang Menggunakan QR Code.

1. Activity Diagram: Login



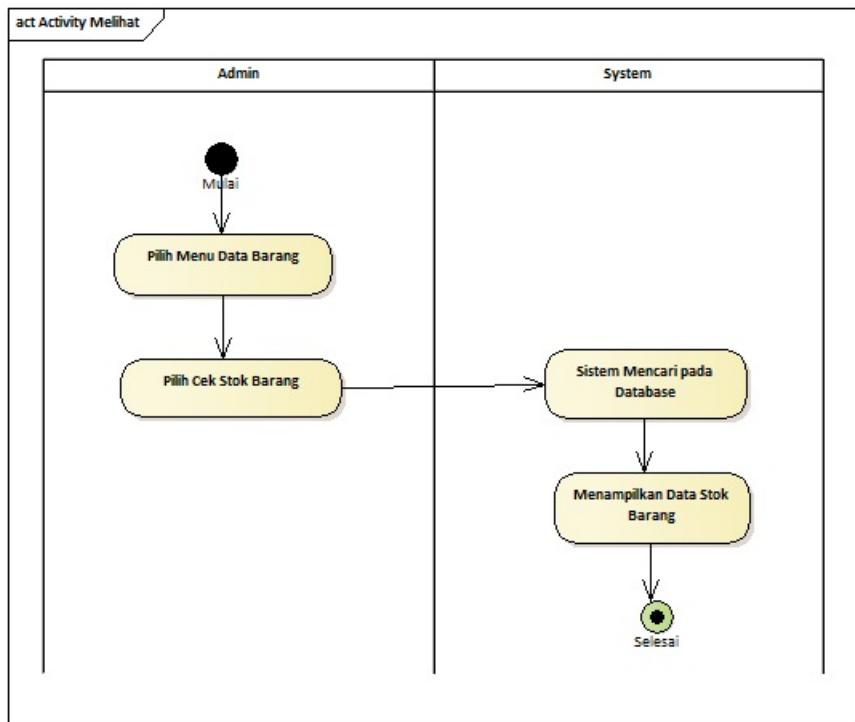
Gambar 3.9 Activity Diagram: Login

2. Activity Diagram: Scan



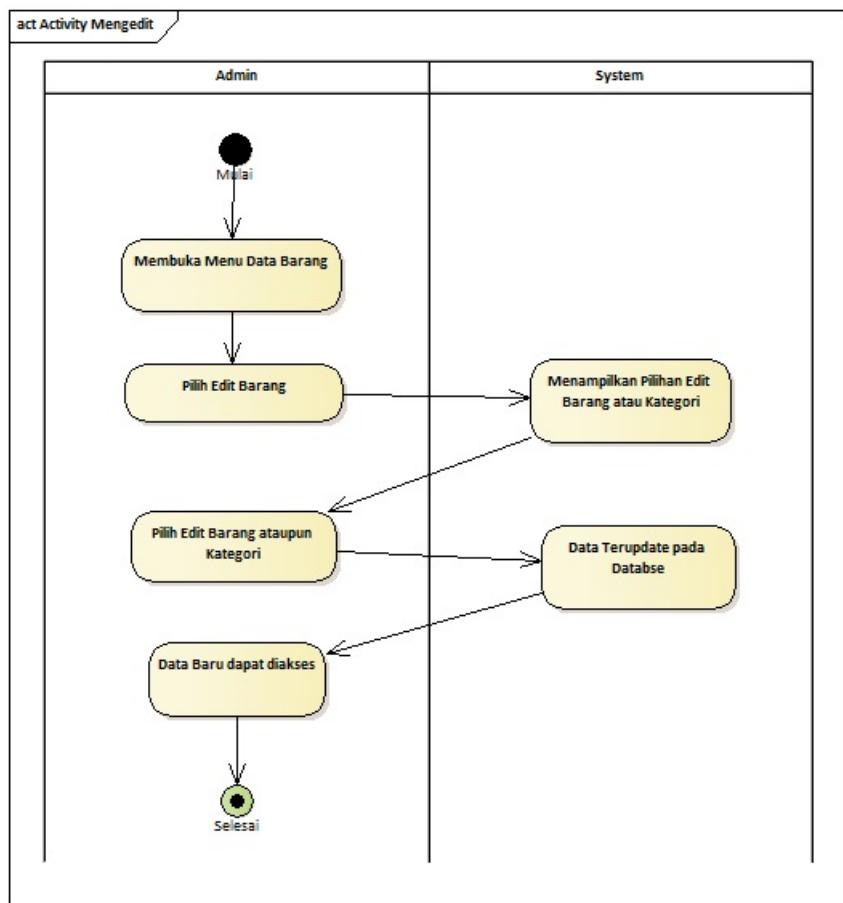
Gambar 3.10 Activity Diagram:Scan

3. Activity Diagram: Melihat Data



Gambar 3.11 Activity Diagram: Melihat Data

4. Activity Diagram: Mengedit

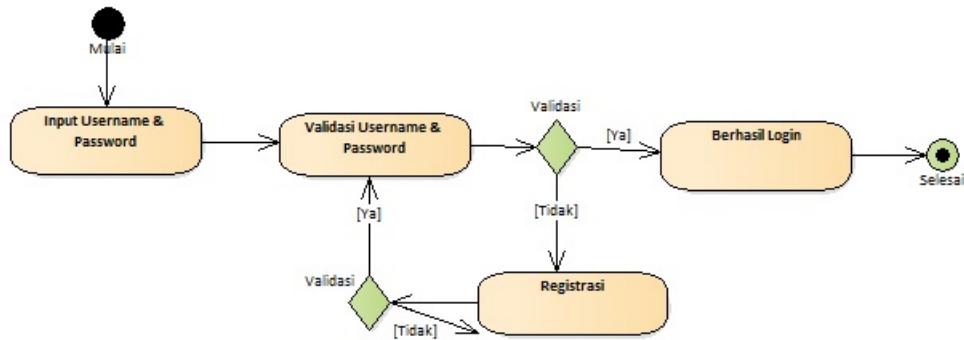


Gambar 3.12 Activity Diagram: Mengedit

3.3.1.3 State Chart Diagram Menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya) suatu objek pada sistem sebagai akibat dari stimulan yang diterima, State chart diagram digunakan untuk memodelkan behavior/method (lifecycle) sebuah kelas atau objek, Memperlihatkan urutan kejadian sesaat (state) yang dilalui sebuah objek, transisi dari sebuah state ke state lainnya.

1. State Chart Diagram: Login

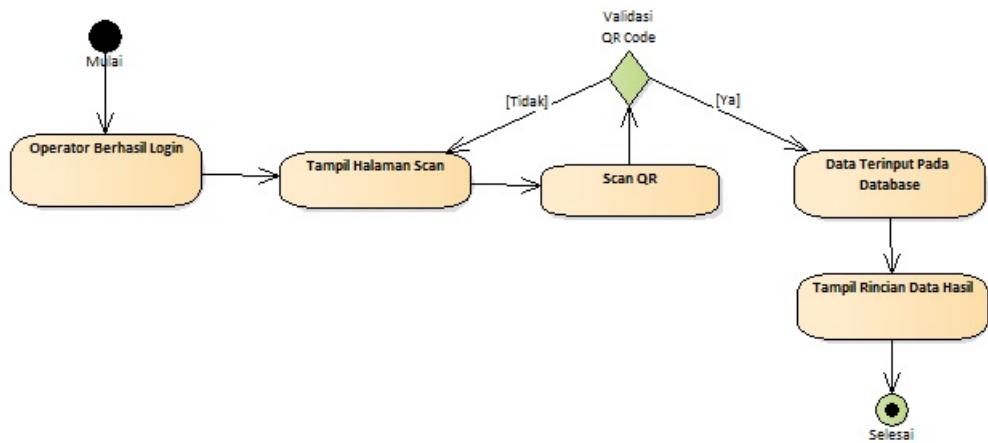
Berikut ini merupakan State chart diagram pada login yang menjelaskan proses login User.



Gambar 3.13 State Chart Diagram: Login

2. State Chart Diagram: Scan

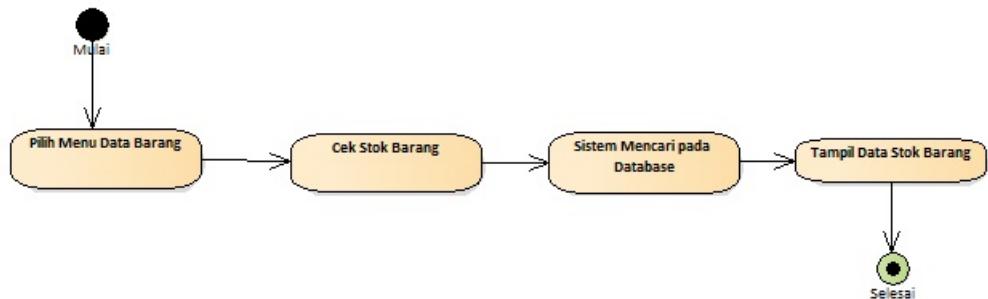
Berikut ini merupakan State chart diagram pada Scan yang menjelaskan proses Scan QR Code.



Gambar 3.14 State Chart Diagram: Scan

3. State Chart Diagram: Melihat Data

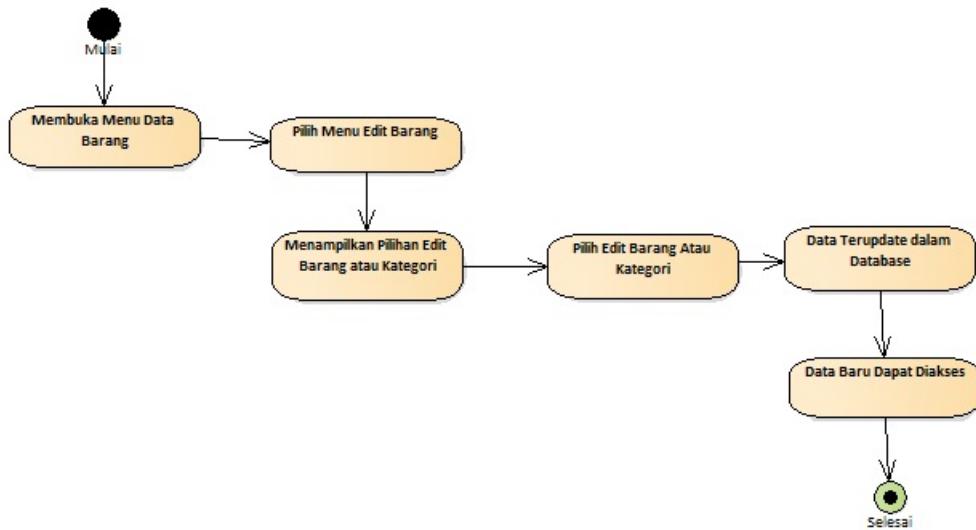
Berikut ini merupakan State chart diagram pada Melihat Data yang menjelaskan proses Melihat Data.



Gambar 3.15 State Chart Diagram: Melihat Data Barang

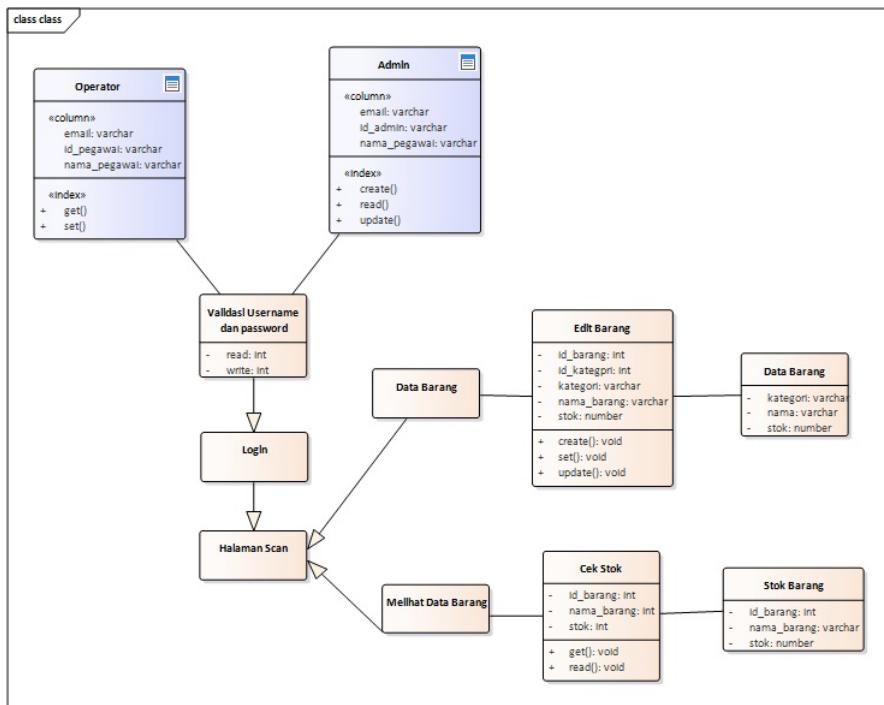
4. State Chart Diagram: Mengedit Data

Berikut ini merupakan State chart diagram pada Mengedit Data yang menjelaskan proses Mengedit Data Barang.



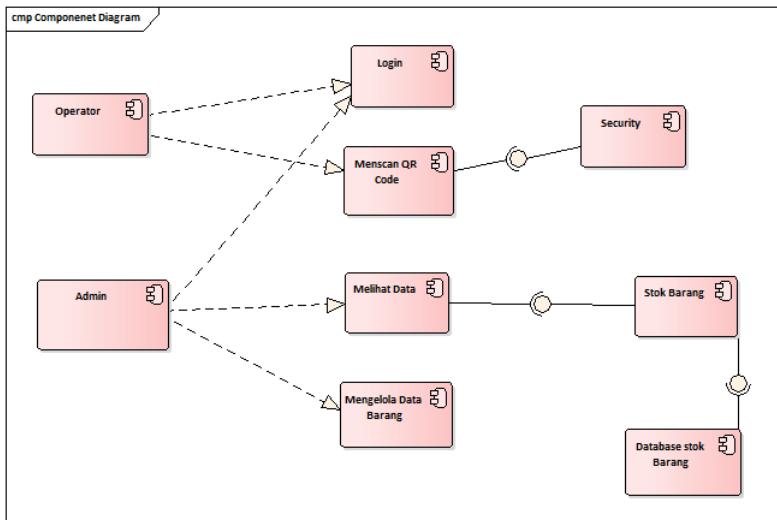
Gambar 3.16 State Chart Diagram: Mengedit Data

3.3.1.4 Class Diagram Class Diagram merupakan model statis yang menggambarkan struktur class, deskripsi class dan hubungan antar class. Adapun class diagram dari aplikasi Sistem Inventory Barang Menggunakan QR Code sebagai berikut:



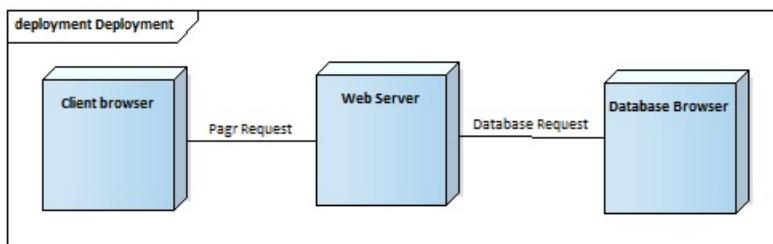
Gambar 3.17 Class Diagram

3.3.1.5 Component Diagram Component diagram yang terdapat pada aplikasi Sistem Inventory Barang Menggunakan QR Code. Dalam aplikasi tersebut terdapat menu login,scan,Melihat Data dan menu Mengedit Data. Aplikasi tersebut terkoneksi dengan database connection.



Gambar 3.18 Component Diagram

3.3.1.6 Deployment Diagram Deployment diagram merupakan alat untuk memvisualisasikan, menspesifikasi, dan mendokumentasikan komponen-komponen yang digunakan dalam sistem berupa hal-hal yang bersifat fisikal. Berikut ini Deployment Diagram dari Sistem Inventory Barang Menggunakan QR Code:



Gambar 3.19 Deployment Diagram

3.3.2 Perancangan Antarmuka

Merancang antarmuka merupakan bagian yang paling penting dari merancang sistem. Biasanya hal tersebut juga merupakan bagian yang paling sulit, karena dalam merancang antarmuka harus memenuhi tiga persyaratan yaitu sebuah antarmuka harus sederhana, sebuah antarmuka harus lengkap, dan sebuah antarmuka harus memiliki kinerja yang cepat.

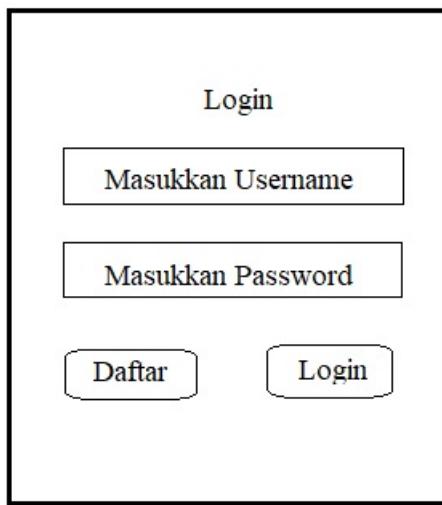
Tujuan dari Perancangan Antarmuka Pengguna adalah merancang interface yang efektif untuk sistem perangkat lunak. Efektif artinya siap digunakan, dan hasilnya sesuai dg kebutuhan. Kebutuhan disini merupakan kebutuhan penggunanya. Pengguna menilai sistem dari interface, bukan dari fungsinya melainkan dari user interfacenya. Jika desain user interfacenya buruk, maka itu sering jadi alasan untuk tidak digunakannya aplikasi tersebut. Selain itu interface yang buruk sebabkan pengguna membuat kesalahan fatal.

Prinsip-prinsip dalam merancang user interface, Berikut ini prinsip-prinsip Merancang Antarmuka pangguna:

1. User familiarity / Mudah dikenali : gunakan istilah, konsep dan kebiasaan user bukan computer (misal: sistem perkantoran gunakan istilah letters, documents, folders bukan directories, file, identifiers. – jenis document open office).
2. Consistency / “selalu begitu” : Konsisten dalam operasi dan istilah di seluruh sistem sehingga tidak membingungkan. – layout menu di open office mirip dgn layout menu di MS office.
3. Minimal surprise / Tidak buat kaget user : Operasi bisa diduga prosesnya berdasarkan perintah yang disediakan.
4. Recoverability/pemulihan : Recoverability ada dua macam: Confirmation of destructive action (konfirmasi terhadap aksi yang merusak) dan ketersediaan fasilitas pembatalan (undo).
5. User guidance / bantuan : Sistem manual online, menu help, caption pada icon khusus tersedia.
6. User diversity /keberagaman : Fasilitas interaksi untuk tipe user yang berbeda disediakan. Misalnya ukuran huruf bisa diperbesar.

Berikut ini rancangan User Interface atau Antar Muka dari Sistem Inventory Barang Menggunakan QR Code:

1. Antar Muka Pengguna Login



Gambar 3.20 Antarmuka Pengguna : Login

Keterangan :

- (a) Label : Login
- (b) Textbox 1 : Input Username
- (c) Textbox 2 : Input Password
- (d) Button 1 : Tombol Login
- (e) Button 2 : Tombol Daftar

2. Antarmuka Pengguna Registrasi

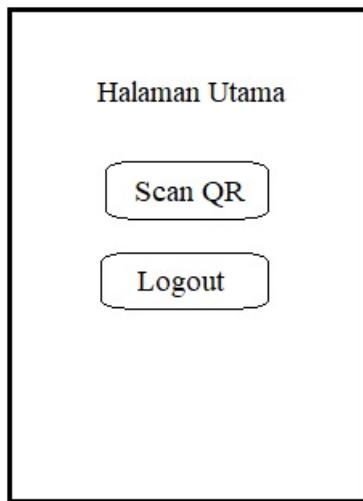
The diagram illustrates a user interface for a registration process. It features a central title 'Registrasi' at the top. Below the title are three rectangular input fields stacked vertically, each containing placeholder text: 'Masukkan Username', 'Masukkan Password', and 'Konfirmasi Password'. At the bottom of the interface are two rounded rectangular buttons: 'Daftar' on the left and 'Login' on the right.

Gambar 3.21 Antarmuka Pengguna : Registrasi

Keterangan :

- (a) Label : Registrasi
- (b) Textbox 1 : Input Username
- (c) Textbox 2 : Input Password
- (d) Textbox 3 : Input Konfirmasi Password
- (e) Button 1 : Tombol Login
- (f) Button 2 : Tombol Daftar

3. Antarmuka Pengguna Halaman Utama Operator

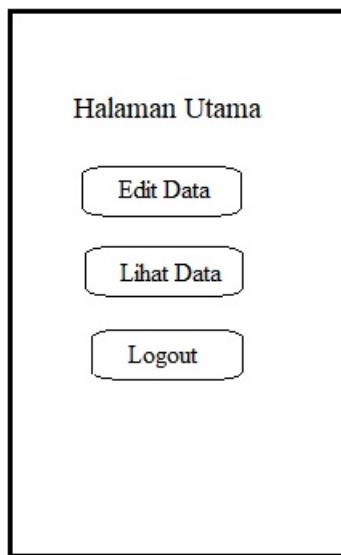


Gambar 3.22 *Antarmuka Pengguna : Halaman Utama Operator*

Keterangan :

- (a) Label : Halaman Utama
- (b) Label : Halaman Utama
- (c) Button 1 : Tombol Scan QR
- (d) Button 2 : Tombol Logout

4. Antarmuka Pengguna Halaman Utama Admin



Gambar 3.23 Antarmuka Pengguna : Halaman Utama Admin

Keterangan :

- (a) Button 1 : Tombol Edit Data
- (b) Button 2 : Tombol Lihat Data
- (c) Button 3 : Tombol Logout

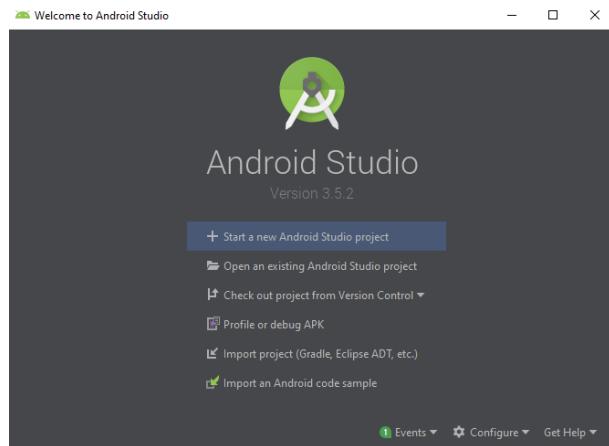
BAB 4

MEMBANGUN APLIKASI

4.1 Memulai Project Baru

1. Hal pertama yang dapat dilakukan jika belum membuka project, Android Studio akan menampilkan layar Selamat Datang, di mana Anda dapat membuat project baru dengan mengklik Start a new Android Studio project.

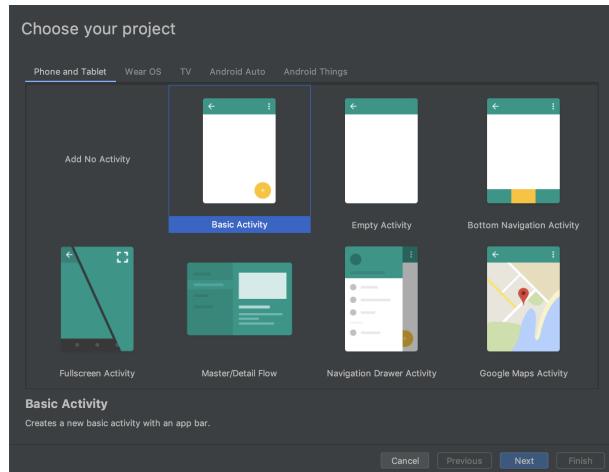
jika sudah ada project yang terbuka, dapat mulai dengan membuat project baru dengan memilih **File kemudian New lalu New Project dari menu utama.** Contoh tampilan utama pada gambar berikut:



Gambar 4.1 Tampilan Awal Android Studio

2. Memilih project Anda

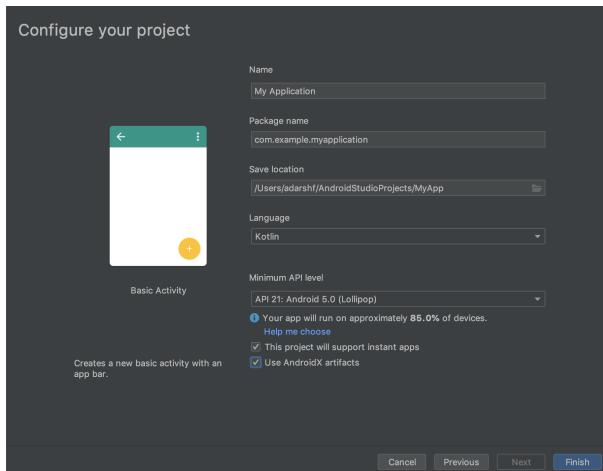
Di layar Choose your project yang muncul, Anda dapat memilih jenis project yang ingin dibuat dari kategori faktor bentuk perangkat, yang ditampilkan sebagai tab di dekat bagian atas wizard. Misalnya, gambar 4.2 menunjukkan project dengan Aktivitas Android dasar untuk ponsel dan tablet yang dipilih.



Gambar 4.2 Memilih Project

3. Mengonfigurasi project

Langkah selanjutnya adalah mengonfigurasi beberapa setelan dan membuat project baru seperti yang dijelaskan di bawah ini dan ditunjukkan pada gambar 4.3. Jika Anda membuat project Native C ++ , Anda dapat mempelajari opsi yang perlu dikonfigurasi lebih lanjut dengan membaca Membuat project baru dengan dukungan C/C ++.



Gambar 4.3 Konfigurasikan project baru Anda dengan beberapa setelan.

Keterangan dari gambar diatas:

- Tentukan Name project Anda.
- Tentukan Package name. Secara default, nama paket ini juga menjadi ID aplikasi Anda, yang dapat diubah nanti.
- Tentukan Save location tempat Anda ingin menyimpan project secara lokal di perangkat yang Anda gunakan.
- Pilih Language yang akan digunakan oleh Android Studio saat membuat kode contoh untuk project baru Anda. Ingat, Anda tidak dibatasi untuk hanya menggunakan bahasa yang dipakai dalam membuat project.
- Pilih Minimum API level yang ingin didukung aplikasi Anda. Ketika memilih level API yang lebih rendah, aplikasi Anda dapat menggunakan API Android modern dalam jumlah yang lebih sedikit. Namun, sebagian besar perangkat Android dapat menjalankan aplikasi Anda. Hal sebaliknya akan berlaku jika Anda memilih level API yang lebih tinggi. Jika ingin melihat lebih banyak data untuk membantu Anda memutuskan, klik Help me choose.
- Jika jenis project yang dipilih mendukung pengalaman instan melalui Google Play Instant, dan Anda ingin mengaktifkannya untuk aplikasi, centang kotak

di sebelah This project will support instant apps. Untuk mempelajari selengkapnya, baca Membuat aplikasi instan pertama Anda.

- (g) Jika ingin project Anda menggunakan library AndroidX secara default, yang merupakan pengganti yang lebih baik dari Android Support Library, centang kotak di sebelah Use AndroidX artifacts. Untuk mempelajari selengkapnya, baca Ringkasan AndroidX.
 - (h) Jika Anda siap membuat project, klik Finish.
4. Jika kita akan mengimpor project yang sudah ada, Kita dapat Klik File lalu New dan pilih Import Project.
- Di jendela yang muncul, masuk ke direktori utama project yang ingin Anda impor kemudian Klik OK.

4.2 Error Yang Sering Terjadi Dalam Membuat Project Pada Android Studio

1. Issues #1 Pada *issues #1 (ERROR: ; TextView app:layout_constraintRight_toRightOf = "parent" app : layout_constraintTop_toTopOf = "parent" app : layout_constraintVertical_bias = "0.177" >)*

merupakan error yang terjadi ketika syntax didalam salah satu program kurang. Solusi dari error ini adalah lebih memperhatikan penggunaan syntax pada program dan menambahkan garis miring atau slash sebelum kurung tutup.

```
<TextView  
.....  
app:layout_constraintRight_toRightOf="parent"  
app:layout_constraintTop_toTopOf="parent"  
app:layout_constraintVertical_bias="0.177"/>
```

2. Issues #2 Pada *issues #2 (ERROR: android:layout_width = "wrap_content")*

yaitu Error yang terjadi ketika tanda kutip pada match parent kurang 1 tanda kutip dua dibalikng. Fungsi dari tanda kutip dua digunakan untuk mendefinisikan tipe data string.

```
android:layout_width="wrap_content"
```

3. Issues #3 Pada *issues #3 (ERROR: android:hint: "username")* Error yang terjadi pada saat kesalahan pengetikan tanda titik dua (:) yang seharusnya seharusnya menggunakan tanda sama dengan (=) pada inputType.

```
android:hint="username"
```

4. Issues #4 Pada *issues #4 (Penjelasan ;Button ;)* Berikut merupakan kode dalam membuat button dari register:

```
<Button  
    android:id="@+id/btn_registerRegist"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="register"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintHorizontal_bias="0.325"  
    app:layout_constraintLeft_toLeftOf="parent"  
    app:layout_constraintRight_toRightOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.641" />
```

5. Issues #5 Pada *issues #5 (android:layoutwidth="wrap_content" android : layoutheight "wrap_content")*

Eror terjadi karena kurang underscore pada layout width dan layout height. sedangkan wrap content merupakan setting atau pengaturan lebar dan tinggi yang mengikuti layar

```
android:layout_width="wrap_content"  
android:layout_height="wrap_content"
```

6. Issues #6 Pada *issues #6 db = DatabaseHelper(this);* merupakan error yang terjadi karena kurangnya perintah new, dimana perintah new ini berfungsi sebagai create dari database

```
db = new DatabaseHelper(this);
```

7. Issues #7 Pada *issues #7 (public class RegisterActivity AppCompatActivity)* merupakan error yang terjadi ketika public claTss yang dibuat tidak mempunyai extends. fungsi dari extend adalah menurunkan contohnya seperti diatas publick class RegisterActivity menurunkan AppCompatActivity sehingga program dapat bekerja.

```
public class RegisterActivity extends AppCompatActivity
```

8. Issues #8 Pada *issues #8 (protected onCreate(Bundle savedInstanceState))* merupakan error yang terjadi ketika void pada perintah ini tidak dimasukkan.

```
protected void onCreate(Bundle savedInstanceState)
```

9. Issues #9 Pada *issues #9 (login.setOnClickListener)* error yang terjadi karena perintah set tidak tituliskan pada program.

```
login.setOnClickListener
```

10. Issues #10 Pada *issues #10 (@Override protected void onCreate(Bundle savedInstanceState) super.onCreate(savedInstanceState); setContentView(R.layout.activity_register);*

Error yang terjadi ketika terdapat perintah atau kode program yang tidak dimasukkan. Perintah tersebut adalah set content view yang tertuju kepada perintah yang jelas.

```
@Override  
protected void onCreate(Bundle savedInstanceState)  
super.onCreate(savedInstanceState);  
setContentView(R.layout.);  
@Override  
protected void onCreate(Bundle savedInstanceState) {
```

```
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_register);
```

11. Issues #11 Pada *issues #11 (username = (EditText) findViewById(R.id.editText_username); (EditText) findViewById(R.id.editText_password);)*

Merupakan error yang terjadi ketika kode program ada yang terlupa ketika dituliskan. Kode yang diatas terjadi krroran pkarena kurangnya regist pada usernameRgist dan passwordRegist.

```
username = (EditText) findViewById(R.id.editText_usernameReg);
password = (EditText) findViewById(R.id.editText_passwordReg);
```

12. Issues #12 Pada *issues #12 (xmlns:tools="tools")* merupakan eror yang terjadi ketika tools yang digunakan tidak di deskripsikan secara jelas.

```
xmlns:tools="http://schemas.android.com/tools"
```

13. Issues #13 Pada *issues #13 (strUsername = username.getText().toString();)* merupakan erroryang terjadi ketika String tidak dituliskan sehingga membuat kode program error. Karena string sebagai type data yang akan diinputkan.

```
String strUsername = username.getText().toString();
```

14. Issues #14 Pada *issues #14 (if(strPassword(strPasswordConf)))* error ini terjadi ketika equals tidak dituliskan padahal equals berarti sama, jika password yang dimasukkan sama maka akan.

```
if (strPassword.equals(strPasswordConf))
```

15. Issues #15 Pada *issues #15 (Error : android:layout_width = "match_parent" android : layout_height = "match_parent" tools : con".RegisterActivity")*

Merupakan error yang terjadi ketika kode program tersebut tidak ditentukan layout context yang akan dibuat.

```
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".RegisterActivity"
```

16. Issues #16 Pada *issues #16 (DatabaseHelper db Button login, register EditText username, password, passwordConf)* error yang terjadi karena kurangnya titik koma pada syntax terakhir.

```
DatabaseHelper db;
Button login, register;
EditText username, password, passwordConf;
```

17. Issues #17 Pada *issues #17 (android:text="LOGIN")* merupakan error yang terjadi ketika kurangnya tanda = untuk memilih type yang akan dideklarasikan.

```
    android:text="LOGIN"
```

18. Issues #18 Pada *issues #18 (import android.content.ContentValues; import android.content.Context; import android.database.Cursor;)* merupakan error yang terjadi ketika perintah import yang digunakan ada yang kurang. Kurang import sqlite sebagai database sehingga eror saat akan membuat database nya

```
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
```

19. Issues #19 Pada *issues #19 (db.execSQL("CREATE TABLE session(id integer PRIMARY KEY, login");)*) merupakan error yang terjadi ketika kurangnya text setelah login sebagai pendeklarasian type data yaitu text

```
db.execSQL("CREATE TABLE session(id integer PRIMARY KEY, log
```

20. Issues #20 Pada *issues #20*

```
Toast.makeText(getApplicationContext(), "Masuk Gagal", Toas
```

Error yang terjadi ketika ketika kurangnya tanda penutup pada show dan titik koma(;) di akhir.

```
Toast.makeText(getApplicationContext(), "Masuk Gagal", Toast.
```

21. Issues #21 Pada *issues #21 (public class extends AppCompatActivity)* Error yang terjadi ketika class yang ada tidak didefinisikan untuk kelas yang mana.

```
public class LoginActivity extends AppCompatActivity
```

22. Issues #22 Pada *issues #22 (URL url = new URL("https://script.google.com/macros/s/AKfycbwLH-76YpMOeuCXzaGlHkPPx1VHEX23GsKfzqp7DwydFH8PPrm");)*

Error yang terjadi kesalahan penulisan url

```
URL url = new URL("https://script.google.com/macros/s/
/AKfycbwLH-76YpMOeuCXzaGlHkPPx1VHEX23GsKfzqp7DwydFH8PPr
```

23. Issues #23 Pada *issues #23 (protected onPostExecute())* error yang terjadi ketika fungsi void tidak dapat dideklarasikan karena tidak dituliskan

```
protected void onPreExecute() {}
```

24. Issues #24 Pada *issues #24 (conn.setRequestMethod(GET);)* merupakan error yang terjadi ketika kurang penulisan petik dua diantara GET yang mengakibatkan pendeklarasian yang tidak jelas.

```
conn.setRequestMethod ("GET");
```

25. Issues #25 Pada *issues #25 (android:id="+id/btnLogout")*

Merupakan error yang terjadi ketika perintah yang digunakan tidak dapat dideklarisikan karena kurangnya @ sebelum tanda +.

```
android:id="+id/btn_logout"
```

26. Issues #26 Pada *issues #26 (findViewById(R.id.button);)* error yang terjadi ketika perintah yang dituliskan ada yang kurang yaitu garis miring di awal setelah kurung buka.

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

27. Issues #27 Pada *issues #27 (package com.;)* merupakan error yang terjadi ketika package tersebut tidak dideklarisasikan.

```
package com.scanner.scantobar;
```

28. Issues #28 Pada *issues #28 (import com.google.zxing.)* error yang terjadi ketika module yang dimasukkan tidak terbaca.

```
import com.google.zxing.integration.android.IntentIntegrator
```

29. Issues #29 Pada *issues #29 (import com.google.zxing.WriterE;)* merupakan error yang terjadi ketika ada write e seharusnya modul tersebut dibaca bukan ditulis.

```
import com.google.zxing.integration.android.IntentResult
```

30. Issues #30 Pada *issues #30 (org.json.JSONObject;)* merupakan error yang terjadi karena tidak ada fungsinya import.

```
import org.json.JSONObject;
```

31. Issues #31 Pada *issues #31 (import java.io.BufferedReader;)* merupakan error yang terjadi karena ketika kurangnya tanda ; untuk dapat dapat berjalan kode program tersebut.

- ```
import java.io.BufferedReader;
```
32. Issues #32 Pada *issues #32 (super.onActivityResult(requestCode, resultCode, data))* Merupakan error yang terjadi ketika kurang penulisan syntax yaitu titik koma(;) .
- ```
super.onActivityResult(requestCode, resultCode, data);
```
33. Issues #33 Pada *issues #33 (public class LoginActivity extends)* merupakan error yang terjadi ketika modul AppCompatActivity tidak dimasukkan.
- ```
public class LoginActivity extends AppCompatActivity
```
34. Issues #34 Pada *issues #34 (username = (EditText)findViewById(R.id.editText\_username); findViewById(R.id.editText\_password);)*
- Merupakan error yang terjadi ketika edit text tidak dapat dideklarasikan. Karena kurang penulisan perintah edit text.
- ```
username = (EditText) findViewById(R.id.editText_username);
password = (EditText) findViewById(R.id.editText_password);
```
35. Issues #35 Pada *issues #35 (android:textColor="@android:color")* error ini terjadi ketika warna yang akan digunakan belum dideklarasikan sehingga perintah tidak jelas.
- ```
android:textColor="@android:color/black"
```
36. Issues #36 Pada *issues #36 (*
- ```
login=findViewById(R.id.btn_login); register = findViewById(R.id.btn_register)
```
- Merupakan error yang terjadi ketika perintah tidak jelas.
- ```
login = (Button) findViewById(R.id.btn_login);
register = (Button) findViewById(R.id.btn_register);
```
37. Issues #37 Pada *issues #37 (db.execSQL("CREATE TABLE user(id integer PRIMARY KEY AUTOINCREMENT, username , password ))")* merupakan error terjadi pada kode ini dikarenakan type data pada username dan password tidak ditentukan=.
- ```
db.execSQL("CREATE TABLE user(
    id integer PRIMARY KEY AUTOINCREMENT, username text, pas
```
38. Issues #38 Pada *issues #38 (public checkSession(String sessionValues))*

Merupakan error yang terjadi ketika parameter boolean tidak dituliskan padahal boolean memiliki 2 nilai yaitu ya atau tidak, true or false.

```
public Boolean checkSession(String sessionValues)
```

39. Issues #39 Pada *issues #39 (SQLiteDatabase db : this.getReadableDatabase())*

Merupakan error yang disebabkan karena kesalahan penulisan syntax yaitu titik dua(:) yang seharusnya tanda sama dengan(=) dan kurangnya penulisan syntax titik koma(;) diakhir.

```
SQLiteDatabase db = this.getReadableDatabase();
```

40. Issues #40 Pada *issues #40 (app:layout_constraintHorizontal_bias =)*

Error pada kode tersebut disebabkan oleh ukuran layout yang belum diisikan sehingga tidak dapat di deklarisasikan.

```
app:layout_constraintHorizontal_bias="0.501"
```

41. Issues #41 Pada *issues #41 (DatabaseHelper db; login, register; username, password;)*

Error yang terjadi ketika tidak adanya pendeklarasian perintah login dan registrasi bagaimana, dan user password bagaimana.

```
DatabaseHelper db;  
Button login, register;  
EditText username, password;
```

42. Issues #42 Pada *issues #42 (import android.widget.EditText;)*

Error yang terjadi pada kode program ini adalah fungsi import untuk memasukkan edit text.

```
import android.widget.EditText;
```

43. Issues #43 Pada *issues #43 ((compile fileTree(dir: 'libs', include: ['*.jar']))* error pada kode program ini disebabkan oleh kode program yang tidak sesuai dengan versi androin seharunya compile digantikan dengan implementation.

```
implementation fileTree(dir: 'libs', include: ['*.jar'])
```

44. Issues #44 Pada *issues #44((URL url = new ();))* error ini terjadi karena tidak dimasukkan alamat url yang dituju.

```
URL url = new ("https://script.google.com/macros/s/AKfycbwLH-76YpMOeuCXzaGlHkPPx1VHEX23GsKfzqp7DwydFH8PPrnn
```

45. Issues #45 Pada *issues #45 (.setDoInput(true); .setDoOutput(true);)*

Merupakan error yang terjadi karena tidak adanya perintah conn. pada kode-program diatas sebagai connection

```
conn.setDoInput(true);
conn.setDoOutput(true);
```

46. Issues #46 Pada *issues #46 (*

```
protected void onPostExecute(String result) {
    Toast.makeText(getApplicationContext(), result,
        Toast.LENGTH_LONG).show();
}
```

Merupakan error yang terjadi karena tidak ada @Override diawal sebagai syntax yang dapat memperbolehkan sebuah kelas mempunyai 2 method atau lebih.

```
@Override
protected void onPostExecute(String result) {
    Toast.makeText(getApplicationContext(), result,
        Toast.LENGTH_LONG).show();
```

47. Issues #47 Pada *issues #47 result.toString();* error tersebut terjadi karena perintah return sebagai syntax pengembalian nilai tidak dituliskan.

```
return result.toString();
```

48. Issues #48 Pada *issues #48 ;androidx.constraintlayout.widget.ConstraintLayout*
xmlns:android="http" error ini terjadi disebabkan oleh url yang dituju tidak jelas atau belum ditentukan.

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
```

49. Issues #49 Pada *issues #49 android:id="@+id/btn_scan"*

Merupakan error yang terjadi karena adanya syntax yang kurang yaitu + dimana @+id perfungsi sebagai pemberi id pada elemen XML

```
android:id="@+id/btn_scan"
```

50. Issues #50 Pada *issues #50 Cursor cursor = db.rawQuery("SELECT FROM user WHERE username = ? AND password = ?", new String[] {username, password});* error pada program ini terjadi karena kurangnya tanda * untuk memanggil database.

```
Cursor cursor = db.rawQuery(  
    "SELECT * FROM user WHERE username = ? AND password = ?"  
, new String[]{username, password});
```

51. Issues #51 Pada *issues #51 ;?xml version="1.0" encoding="utf-8"?;*

Fungsi dari xml sendiri sebagai markup language yang mempunyai himpunan aturan untuk pengkodean dokumen dalam format yang dimengerti oleh manusia dan mesin dengan versi=1.0

52. Issues #52 Pada *issues #52 androidx.constraintlayout.widget.ConstraintLayout*

Constraint layout merupakan sebuah view group yang membolehkan kita untuk mengatur posisi dan ukuran widgets dengan suka-suka kita (fleksibel).

53. Issues #53 Pada *issues #53 ;TextView* fungsi dari TextView berfungsi sebagai sebagi penulisan text pada XML yang akan tampil sebagai user interface pada desain.

54. Issues #54 Pada *issues #54*

```
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"
```

merupakan syntax untuk mengatur layout lebar dan tinggi dari desain dengan wrap content berfungsi agar lebar dan tinggi dapat mengikuti object itu sendiri.

55. Issues #55 Pada *issues #55 android:layout_width = "match_parent"*

Fungsi dari Match Parent pada kode program tersebut agar lebar dan tinggi akan mengikuti layar pada perangkat Android.

56. Issues #56 Pada *issues #56 tools:context=".RegisterActivity"*

Syntax tersebut menjelaskan bahwa tools context file XML tersebut berada pada file RegisterActivity.java

57. Issues #57 Pada *issues #57 package com.scanner.scantobar;* Fungsi package adalah sebuah sarana untuk mengelompokkan atau mengorganisasikan kelas dan interface yang sama atau sekelompok menjadi satu unit tunggal dalam library.

58. Issues #58 Pada *issues #58 android:id="@+id/btnLoginRegist"*

Digunakan saat akan memberikan sebuah id pada elemen atau tag dari layout XML untuk membedakan kegunaan masing-masing elemen atau tag.

59. Issues #59 Pada *issues #59 android:text="register"* Syntax tersebut memiliki fungsi sebagai penulisan text pada button dengan text "register"

60. Issues #60 Pada *issues #60 (/androidx.constraintlayout.widget.ConstraintLayout;* Syntax penutup dari constraint layout sebelumnya yang berfungsi sebagai iew

group yang membolehkan kita untuk mengatur posisi dan ukuran widgets dengan suka-suka kita (leksibel).

61. Issues #61 Pada *issues #61 public class RegisterActivity extends AppCompatActivity* fungsi dari public class adalah klasifikasi dari objek yang dapat diakses oleh class lain atau dapat disebut bersifat umum.
62. Issues #62 Pada *issues #62 public class RegisterActivity extends AppCompatActivity* Fungsi extend di gunakan untuk mendapatkan sifat-sifat/atribut dari class induk ke dalam kelas dasar.
63. Issues #63 Pada *issues #63 @Override* Method Overloading adalah sebuah kemampuan yang membolehkan sebuah class mempunyai 2 atau lebih method dengan nama yang sama, yang membedakan adalah parameternya.
64. Issues #64 Pada *issues #64 protected void onCreate(Bundle savedInstanceState)* Void adalah method yang tidak memiliki nilai kembali/return,
65. Issues #65 Pada *issues #65 super.onCreate(savedInstanceState);* mengimplementasikan .onCreate untuk membuat SqliteDatabase dengan tabel sebagai penambahan data awal.
66. Issues #66 Pada *issues #66 login = (Button) findViewById(R.id.btnLoginRegist);* terdapat fungsi findViewById untuk mencari id yaitu (R.id.btn_loginRegist) yang sebelumnya telah didefinisikan pada XML dengan syntax @+id.
67. Issues #67 Pada *issues #67*

```
URL url = new URL  
("https://script.google.com/macros/s/  
AKfycbwLH-76YpMOeuCXzaGlHkPPx1VHEX23GsKfzqp7DwydFH8PPrm/e")
```

URL url = new url berfungsi sebagai pemanggil fungsi dari url tersebut.

68. Issues #68 Pada *issues #68 implementation fileTree(dir: 'libs', include: ['*.jar'])* Fungsi implementation adalah untuk mengimplementasikan suatu interface kedalam suatu class.
69. Issues #69 Pada *issues #69 return true;*
Terdapat fungsi return sebagai pengembalian nilai.
70. Issues #70 Pada *issues #70 public Boolean checkLogin(String username, String password)*
Terdapat perintah checkLogin dengan type data string pada username dan password.

4.3 Issues #71

Pada issues #71 ((*protected void onActivityResult*) *onActivityResult*) adalah sebuah fungsi untuk menampilkan hasil aktivitas.

71. Issues #72 Pada issues #72

```
if (scannedData != null) {  
    // Here we need to handle scanned data...  
    new SendRequest().execute();  
} else {  
}
```

terdapat fungsi dengan perintah if else, yaitu dimana ada If jika dengan kondisi benar dengan pernyataannya else kondisi lainnya dengan pernyataan lain.

72. Issues #73 Pada issues #73 *super.onActivityResult(requestCode, resultCode, data);* super berfungsi untuk merepresentasikan objek dari class induk.
73. Issues #74 Pada issues #74 *import android.database.sqlite.SQLiteDatabase;* Import Sqlite berfungsi mengakses SQLiteDatabase.
74. Issues #75 Pada issues #75 *public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion)* onUpgrade berfungsi untuk upgrade data pada database.
75. Issues #76 Pada issues #76 (*boolean first = true;*) Tipe Data Boolean adalah suatu tipe data yang hanya mengandung nilai true atau false.
76. Issues #77 Pada issues #77 (*Toast.makeText(getApplicationContext())*) Toast berfungsi untuk menampilkan pesan berupa Text, yang terlintas setelah pengguna melakukan aksi tertentu.
77. Issues #78 Pada issues #78 (*android:textSize="28dp"*) textSize berfungsi untuk mengatur ukuran tulisan.
78. Issues #79 Pada issues #79 (*android:textColor="@android:color/black"*) textColor memberikan warna pada text atau tulisan.
79. Issues #80 Pada issues #80 (*EditText*)EditText berfungsi untuk memasukkan teks diaplikasi Android. Jika pengguna diminta untuk memasukkan suatu teks, maka View ini lah yang menjadi sarana utama untuk melakukannya.

BAB 5

IMPLEMENTASI

5.1 Pembuatan Aplikasi *QR CODE* Berbasis Android

5.1.1 Lingkungan Implementasi

Sesudah menyelesaikan proses analisis, proses yang dilakukan selanjutnya adalah perancangan spesifikasi. Tahap ini merupakan tahap peletakan sistem agar dapat dioperasikan sehingga perlu penyesuaian terhadap rancangan yang meliputi lingkungan perangkat keras dan perangkat lunak.

5.1.2 Lingkungan Perangkat Keras

Dalam menganalisis aplikasi ini, spesifikasi perangkat keras yang digunakan adalah sebagai berikut :

1. Processor : IntelR coreTM i3-6006U CPU @2,00 Ghz 1.99 Ghz
2. Memory : 4 GB RAM
3. Harddisk : 1 TB HDD

5.1.3 Lingkungan Perangkat Lunak

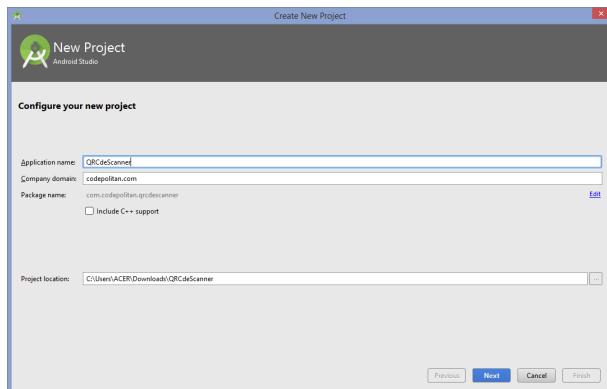
Dalam menganalisis aplikasi ini, spesifikasi perangkat lunak yang digunakan adalah sebagai berikut:

1. Sistem Operasi : Windows 10 Home
2. Bahasa Pemrograman : Java, PHP
3. Software Pendukung : Google Chrome

5.1.4 Aplikasi Scanner Inventory Barang

1. Buat Project Baru

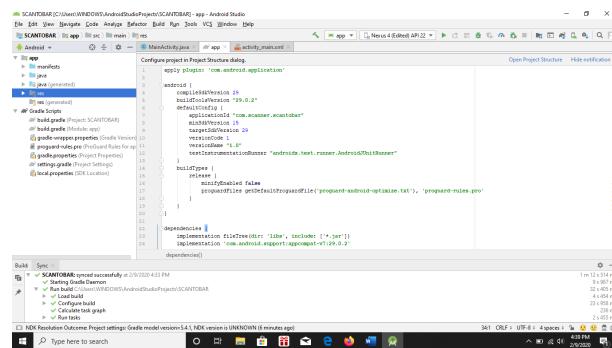
Untuk implementasi QR Code Scanner, kita buat sebuah project Android baru, buka file -New - New Project



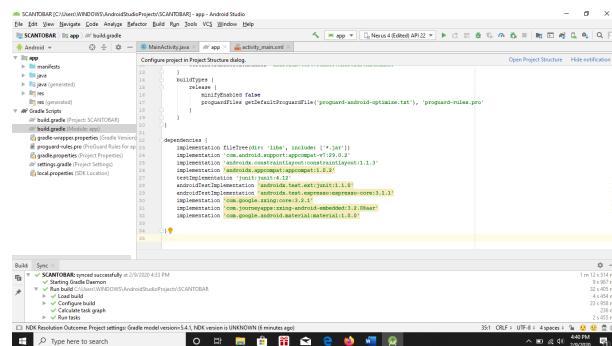
Gambar 5.1 Membuat Project Baru

2. Tambahkan Library Zxing

Library Zxing disini digunakan untuk membaca QR Code, tambahkan pada build.gradle pada android studio, kemudian Sync Now



Gambar 5.2 Codingan Pada Build Gradle



Gambar 5.3 Codingan Pada Build Gradle

* Codingan Pada build.gradle

apply plugin: 'com.android.application'

android

compileSdkVersion 29

buildToolsVersion "29.0.2"

defaultConfig

applicationId "com.scanner.scantobar"

minSdkVersion 15

targetSdkVersion 29

```
versionCode 1
versionName "1.0"
testInstrumentationRunner
"androidx.test.runner.AndroidJUnitRunner"
buildTypes
release
minifyEnabled false
proguardFiles
getDefaultProguardFile
('proguard-android-optimize.txt'),
'proguard-rules.pro'
dependencies
implementation fileTree(dir: 'libs', include: ['*.jar'])
implementation 'com.android.support:appcompat-v7:29.0.2'
implementation
'androidx.constraintlayout:constraintlayout:1.1.3'
implementation 'androidx.appcompat:appcompat:1.0.2'
testImplementation 'junit:junit:4.12'
androidTestImplementation 'androidx.test.ext:junit:1.1.0'
androidTestImplementation
'androidx.test.espresso:espresso-core:3.1.1'
implementation 'com.google.zxing:core:3.2.1'
implementation
'com.journeyapps:zxing-android-embedded:3.2.0@aar'
implementation 'com.google.android.material:material:1.0.0'
```

3. Mendefinisikan View Object

Dengan membuat class MainActivity.java yang nantinya akan diimplementasikan untuk membuat aplikasi.

* Codingan Pada MainActivity

```
package com.scanner.scantobar;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import android.app.Activity;
import android.content.Intent;
```



```
  package com.santosh.santoshtest;
  import android.os.Bundle;
  import android.app.Activity;
  import android.content.Intent;
  import android.os.Bundle;
  import android.view.View;
  import android.widget.TextView;
  import com.google.android.gms.integration.IntentIntegrator;
  import com.google.android.gms.integration.IntentResult;
  import java.util.List;
  import java.util.ArrayList;
  import java.io.BufferedReader;
  import java.io.InputStream;
  import java.io.InputStreamReader;
  import java.io.OutputStreamWriter;
  import java.net.URL;
  import java.net.HttpURLConnection;
  import java.util.Scanner;
  import java.util.concurrent.TimeUnit;
  import java.util.concurrent.ExecutorService;
  import java.util.concurrent.Executors;
  import java.util.concurrent.Future;
  import java.util.concurrent.TimeoutException;
  import java.util.concurrent.ExecutionException;
  import java.util.concurrent.CompletableFuture;
  import android.os.AsyncTask;
  import android.os.AsyncTask$FutureTask;
  import android.os.AsyncTask$OnPostExecute;
  import android.os.AsyncTask$OnPreExecute;
  import android.os.AsyncTask$OnProgressUpdate;
```

Gambar 5.4 Codingan Pada MainActiviyt.java

```
public class MainActivity extends AppCompatActivity {
    Session session;
    SessionManager sessionManager;
    DatabaseHelper databaseHelper;
    Log log;

    String sessionData;
    Button button;
    Boolean isSessionCreated = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        sessionManager = new SessionManager(getApplicationContext());
        session = sessionManager.createSession();
        sessionManager.setSession(session);
        databaseHelper = new DatabaseHelper(getApplicationContext());
        databaseHelper.createDatabase();
        Intent intent = new Intent(this, BackgroundService.class);
        startService(intent);
    }

    @Override
    public void onStartCommand(Intent intent, int flags, int startId) {
        sessionManager.setSession(session);
        sessionManager.getSession().setListener(new SessionListener() {
            @Override
            public void onSessionUpdate(Session session) {
                if (session.isSessionCreated()) {
                    log("Session created");
                } else {
                    log("Session updated");
                }
            }
        });
        sessionManager.getSession().start();
    }

    @Override
    public void onUpgradeSession(Session session, SessionManager sessionManager, int oldVersion, int newVersion) {
        sessionManager.setSession(session);
        sessionManager.getSession().setListener(new SessionListener() {
            @Override
            public void onSessionUpdate(Session session) {
                if (session.isSessionCreated()) {
                    log("Session created");
                } else {
                    log("Session updated");
                }
            }
        });
        sessionManager.getSession().start();
    }
}

private void log(String message) {
    log.i("MainActivity", message);
}
```

Gambar 5.5 Codingan Pada MainActiviyt.java



The screenshot shows the Android Studio interface with the code editor open to the `DatabaseHelper.java` file. The code implements a SQLiteOpenHelper class for managing a database. It includes methods for creating tables, inserting data, and querying data. The `onCreate` method creates a table for storing user login information. The `onUpgrade` method handles database schema changes by dropping the old table and creating a new one. The `getWritableDatabase` and `getReadableDatabase` methods return the database handle. The `insertUser` method inserts a new user record into the database. The `getUser` method queries the database for a user record based on their ID. The `close` method closes the database connection.

```
public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "UserDB";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "UserTable";
    private static final String COL_ID = "ID";
    private static final String COL_USERNAME = "username";
    private static final String COL_PASSWORD = "password";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_TABLE = "CREATE TABLE " + TABLE_NAME + "("
                + COL_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
                + COL_USERNAME + " TEXT NOT NULL, "
                + COL_PASSWORD + " TEXT NOT NULL);";
        db.execSQL(CREATE_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        String DROP_TABLE = "DROP TABLE IF EXISTS " + TABLE_NAME;
        db.execSQL(DROP_TABLE);
        onCreate(db);
    }

    public void insertUser(String username, String password) {
        SQLiteDatabase db = getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL_USERNAME, username);
        contentValues.put(COL_PASSWORD, password);
        db.insert(TABLE_NAME, null, contentValues);
        db.close();
    }

    public User getUser(int id) {
        SQLiteDatabase db = getReadableDatabase();
        Cursor cursor = db.query(TABLE_NAME, new String[]{COL_ID, COL_USERNAME, COL_PASSWORD}, COL_ID + "=?",
                new String[]{String.valueOf(id)}, null, null, null);
        if (cursor != null) {
            cursor.moveToFirst();
        }
        User user = new User();
        user.setId(cursor.getInt(0));
        user.setUsername(cursor.getString(1));
        user.setPassword(cursor.getString(2));
        cursor.close();
        db.close();
        return user;
    }

    public void close() {
        SQLiteDatabase db = getWritableDatabase();
        db.close();
    }
}
```

Gambar 5.6 Codingan Pada MainActiviyt.java

```

public class MainActiviyt extends AsyncTaskString {
    protected void onPreExecute() {
        super.onPreExecute();
    }

    protected void onProgressUpdate() {
        super.onProgressUpdate();
    }

    protected void onPostExecute() {
        super.onPostExecute();
    }

    @Override
    protected String doInBackground(String... params) {
        try {
            // Inserting data into DB
            String insertStatement = "INSERT INTO users (username, password) VALUES ('" + params[0] + "','" + params[1] + "')";
            DatabaseHelper db = new DatabaseHelper(getApplicationContext());
            db.insertData(insertStatement);
        } catch (Exception e) {
            Log.e("Error", "Error inserting data into DB");
        }
    }
}

```

Gambar 5.7 Codingan Pada MainActiviyt.java

```

public class MainActiviyt extends AsyncTaskString {
    protected void onPreExecute() {
        super.onPreExecute();
    }

    protected void onProgressUpdate() {
        super.onProgressUpdate();
    }

    protected void onPostExecute() {
        super.onPostExecute();
    }

    @Override
    protected String doInBackground(String... params) {
        try {
            String insertStatement = "INSERT INTO users (username, password) VALUES ('" + params[0] + "','" + params[1] + "')";
            DatabaseHelper db = new DatabaseHelper(getApplicationContext());
            db.insertData(insertStatement);
        } catch (Exception e) {
            Log.e("Error", "Error inserting data into DB");
        }
    }
}

```

Gambar 5.8 Codingan Pada MainActiviyt.java

```

public class MainActiviyt extends AsyncTaskString {
    protected void onPreExecute() {
        super.onPreExecute();
    }

    protected void onProgressUpdate() {
        super.onProgressUpdate();
    }

    protected void onPostExecute() {
        super.onPostExecute();
    }

    @Override
    protected String doInBackground(String... params) {
        try {
            String insertStatement = "INSERT INTO users (username, password) VALUES ('" + params[0] + "','" + params[1] + "')";
            DatabaseHelper db = new DatabaseHelper(getApplicationContext());
            db.insertData(insertStatement);
        } catch (Exception e) {
            Log.e("Error", "Error inserting data into DB");
        }
    }
}

```

Gambar 5.9 Codingan Pada MainActiviyt.java

```
import android.database.sqlite.SQLiteDatabase;
import android.os.AsyncTask;
import android.os.Build;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
import android.content.DialogInterface;
import android.database.Cursor;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.Toolbar;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.nio.charset.StandardCharsets;
import java.util.Iterator;
import javax.net.ssl.HttpsURLConnection;
public class MainActivity extends AppCompatActivity
DatabaseHelper db;
Button logout;
String scannedData;
Button scanBtn;
@Override
```

```
protected void onCreate (Bundle savedInstanceState)
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
db = new DatabaseHelper(this);
logout = findViewById(R.id.btn_logout);
Boolean checkSession = db.checkSession("ada");
if (checkSession == false)
Intent loginIntent = new Intent(MainActivity.this,
LoginActivity.class);
startActivity(loginIntent);
finish();
//logout
logout.setOnClickListener(new View.OnClickListener()
@Override
public void onClick(View v)
Boolean updateSession =
db.upgradeSession("kosong", 1);
Toast.makeText(getApplicationContext(),
"Berhasil Keluar",
Toast.LENGTH_SHORT).show();
Intent loginIntent = new
Intent(MainActivity.this, LoginActivity.class);
startActivity(loginIntent);
finish();
);
final Activity activity =this;
scanBtn = findViewById(R.id.btn_scan);
scanBtn.setOnClickListener(new View.OnClickListener()
@Override
public void onClick(View view)
IntentIntegrator integrator = new
IntentIntegrator(activity);
integrator.setDesiredBarcodeFormats(IntentIntegrator.QR_CODE_TYPES);
integrator.setPrompt("Scan");
```

```
integrator.setBeepEnabled(false);
integrator.setCameraId(0);
integrator.setBarcodeImageEnabled(false);
integrator.initiateScan();
);

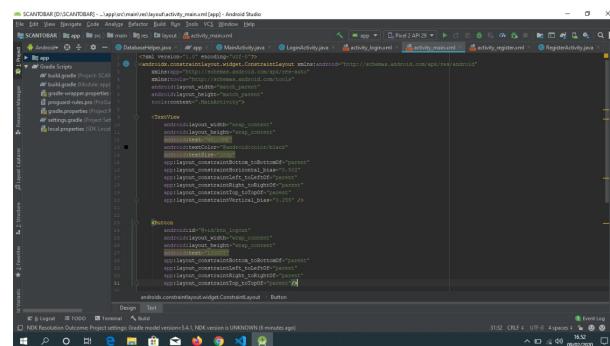
@Override
protected void onActivityResult(int requestCode, int
resultCode, Intent data)
IntentResult result =
IntentIntegrator.parseActivityResult(requestCode,resultCode,data);
if(result!=null)
scannedData = result.getContents();
if (scannedData != null)
// Here we need to handle scanned data...
new SendRequest().execute();
else
super.onActivityResult(requestCode, resultCode, data);
public class SendRequest extends AsyncTask<String, Void,
String>
protected void onPreExecute()
protected String doInBackground(String... arg0)
try
//Enter script URL Here
URL url = new
URL("Link Spreadsheet");
JSONObject postDataParams = new JSONObject();
//Passing scanned code as parameter
postDataParams.put("sdata",scannedData);
Log.e("params",postDataParams.toString());
HttpURLConnection conn = (HttpURLConnection)
url.openConnection();
conn.setReadTimeout(15000 /* milliseconds */);
conn.setConnectTimeout(15000 /* milliseconds */);
conn.setRequestMethod("GET");
```

```
conn.setDoInput(true);
conn.setDoOutput(true);
OutputStream os = conn.getOutputStream();
BufferedWriter writer = null;
if (android.os.Build.VERSION.SDK_INT >=
    android.os.Build.VERSION_CODES.KITKAT)
writer = new BufferedWriter(
new OutputStreamWriter(os,
StandardCharsets.UTF_8));
writer.write(getpostDataString(postDataParams));
writer.flush();
writer.close();
os.close();
int responseCode=conn.getResponseCode();
if (responseCode == HttpsURLConnection.HTTP_OK)
BufferedReader in=new BufferedReader(new
InputStreamReader(conn.getInputStream()));
StringBuffer sb = new StringBuffer();
String line="";
while((line = in.readLine()) != null)
sb.append(line);
break;
in.close();
return sb.toString();
else
return "false : " + responseCode;
catch(Exception e)
return "Exception: " + e.getMessage();
@Override
protected void onPostExecute(String result)
Toast.makeText(getApplicationContext(), result,
Toast.LENGTH_LONG).show();
public String getpostDataString(JSONObject params)
throws Exception
```

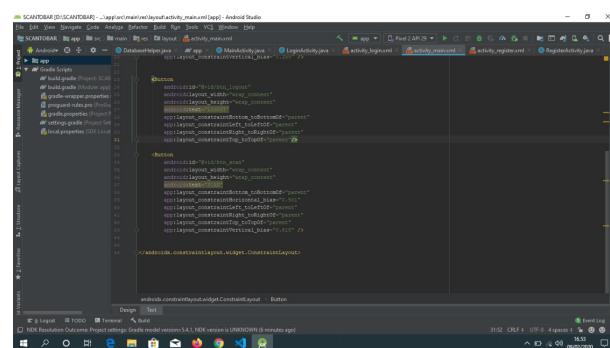
```
StringBuilder result = new StringBuilder();
boolean first = true;
Iterator<String> itr = params.keys();
while(itr.hasNext())
String key= itr.next();
Object value = params.get(key);
if (first)
first = false;
else
result.append("'");
result.append(URLEncoder.encode(key, "UTF-8"));
result.append("=");
result.append(URLEncoder.encode(value.toString(), "UTF-8"));
return result.toString();
```

4. Membuat User Interface

Untuk layoutnya kita design untuk membuat tampilan. dan juga menambahkan button, dimana nanti akan diarahkan ke scanner QR Codenya



Gambar 5.10 Codingan Pada ActivityMain



Gambar 5.11 Codingan Pada ActivityMain

* Codingan Pada ActivityMain

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:context=".MainActivity" >

    <ConstraintLayout
        android:id="@+id/constraintLayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#F0F0F0" >

        <Button
            android:id="@+id/button_scanner"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="10dp"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginTop="10dp"
            android:background="@color/colorPrimaryDark"
            android:text="Scan QR"
            android:textColor="#FFFFFF" />

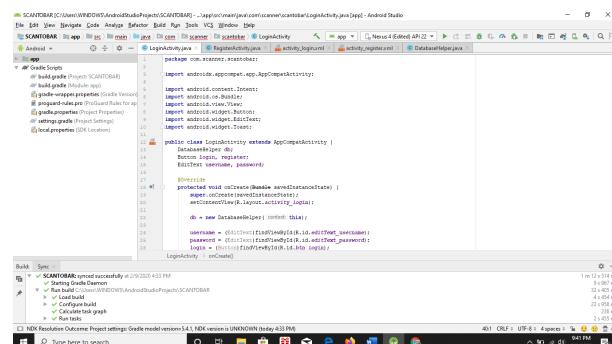
        <Button
            android:id="@+id/button_logout"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="10dp"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:layout_marginTop="10dp"
            android:background="@color/colorPrimaryDark"
            android:text="Logout"
            android:textColor="#FFFFFF" />
    </ConstraintLayout>
</CoordinatorLayout>

```

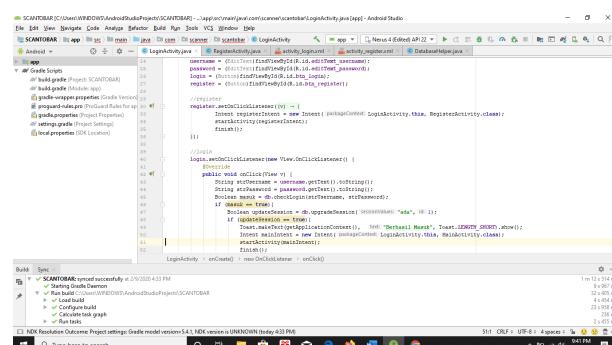
```
|TextView  
    android:layout_width = "wrap_content"  
    android:layout_height = "wrap_content"  
    android:text="WELCOME"  
    android:textColor="@android:color/black"  
    android:textSize="28dp"  
    app:layout_constraintBottom_toBottomOf = "parent"  
    app:layout_constraintHorizontal_bias = "0.502"  
    app:layout_constraintLeft_toLeftOf = "parent"  
    app:layout_constraintRight_toRightOf = "parent"  
    app:layout_constraintTop_toTopOf = "parent"  
    app:layout_constraintVertical_bias = "0.255" />  
  
|Button  
    android:id="@+id/btnLogout"  
    android:layout_width = "wrap_content"  
    android:layout_height = "wrap_content"  
    android:text="LOGOUT"  
    app:layout_constraintBottom_toBottomOf = "parent"  
    app:layout_constraintLeft_toLeftOf = "parent"  
    app:layout_constraintRight_toRightOf = "parent"  
    app:layout_constraintTop_toTopOf = "parent" />  
  
|Button  
    android:id="@+id/btnScan"  
    android:layout_width = "wrap_content"  
    android:layout_height = "wrap_content"  
    android:text="SCAN"  
    app:layout_constraintBottom_toBottomOf = "parent"  
    app:layout_constraintHorizontal_bias = "0.501"  
    app:layout_constraintLeft_toLeftOf = "parent"  
    app:layout_constraintRight_toRightOf = "parent"  
    app:layout_constraintTop_toTopOf = "parent"  
    app:layout_constraintVertical_bias = "0.618" />  
  
|android.support.design.widget.AppBarLayout  
    android:layout_width = "match_parent"
```

```
    android:layout_height = "wrap_content"
    android:theme="@style/AppTheme.AppBarOverlay"
    android.support.v7.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width = "match_parent"
    android:layout_height = "?attr actionBarSize"
    android:background="?attr/colorPrimary"
    app:popupTheme="@style/AppTheme.PopupOverlay" />
    android.support.design.widget.AppBarLayout,
```

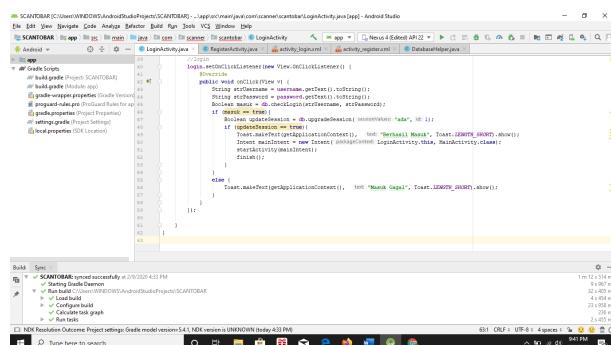
5. Menu Login



Gambar 5.12 Codingan Pada LoginActivity.java



Gambar 5.13 Codingan Pada LoginActivity.java



Gambar 5.14 Codingan Pada LoginActivity.java

* Codingan Pada Login

```

package com.scanner.scantobar;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class LoginActivity extends AppCompatActivity
DatabaseHelper db;
Button login, register;
EditText username, password;
@Override
protected void onCreate(Bundle savedInstanceState)
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);
db = new DatabaseHelper(this);
username = (EditText)findViewById(R.id.editText_username);
password = (EditText)findViewById(R.id.editText_password);
login = (Button)findViewById(R.id.btn_login);
register = (Button)findViewById(R.id.btn_register);
//register

```

```
register.setOnClickListener(new View.OnClickListener()
    @Override
    public void onClick(View v)
        Intent registerIntent = new
        Intent(LoginActivity.this, RegisterActivity.class);
        startActivity(registerIntent);
        finish();
    );
//login
login.setOnClickListener(new View.OnClickListener()
    @Override
    public void onClick(View v)
        String strUsername =
        username.getText().toString();
        String strPassword = password.getText().toString();
        Boolean masuk = db.checkLogin(strUsername, strPassword);
        if (masuk == true)
            Boolean updateSession = db.upgradeSession("ada",1);
            if (updateSession == true)
                Toast.makeText(getApplicationContext(), "Berhasil Masuk",
                Toast.LENGTH_SHORT).show();
            Intent mainIntent = new
            Intent(LoginActivity.this,
            MainActivity.class);
            startActivity(mainIntent);
            finish();
        else
            Toast.makeText(getApplicationContext(), "Masuk Gagal", Toast.LENGTH_SHORT).show();
    );
```

6. Menu Register

The screenshot shows the Android Studio interface with the KANTOBAR project open. The left sidebar displays the project structure with modules like KANTOBAR, KANTOBAR-Android, and KANTOBAR-UI. The main editor shows the `LogInActivity.java` file. The bottom-left corner shows the build log for the KANTOBAR module, which includes the command run-variant: debugAndroid, the path C:\Users\Ivan\AndroidStudioProjects\KANTOBAR\KANTOBAR\app\src\main\java\com\ivan\kantobar\activity\LogInActivity.java, and the output of the Java code. The bottom right shows the system tray with icons for battery, signal, and volume.

Gambar 5.15 Codingan Pada RegisterActivity.java

Gambar 5.16 Codingan Pada RegisterActivity.java



The screenshot shows the Android Studio interface with the code editor open to `LoginActivity.java`. The code implements a login logic using shared preferences. It checks if the user has already logged in, and if so, it retrieves the saved credentials and logs them in. If no credentials are found, it prompts the user to enter them. The code uses `EditText` for input fields and `Toast` for displaying messages.

```
public class LoginActivity extends AppCompatActivity {
    EditText etEmail, etPassword;
    Button btnLogin;
    TextView tvForgot;
    SharedPreferences sharedPreferences;
    String email, password;
    Boolean isEmailSaved, isPasswordSaved;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        etEmail = findViewById(R.id.et_email);
        etPassword = findViewById(R.id.et_password);
        btnLogin = findViewById(R.id.btn_login);
        tvForgot = findViewById(R.id.tv_forgot);

        sharedPreferences = getSharedPreferences("UserDetails", MODE_PRIVATE);
        isEmailSaved = sharedPreferences.contains("Email");
        isPasswordSaved = sharedPreferences.contains("Password");

        if (isEmailSaved & isPasswordSaved) {
            etEmail.setText(sharedPreferences.getString("Email", null));
            etPassword.setText(sharedPreferences.getString("Password", null));
        } else {
            etEmail.requestFocus();
            etEmail.setOnFocusChangeListener(new View.OnFocusChangeListener() {
                @Override
                public void onFocusChange(View v, boolean hasFocus) {
                    if (!hasFocus) {
                        etEmail.clearFocus();
                        etPassword.requestFocus();
                    }
                }
            });
        }
    }

    @Override
    public void onClick(View v) {
        if (v == btnLogin) {
            email = etEmail.getText().toString();
            password = etPassword.getText().toString();

            if (email.isEmpty() | password.isEmpty()) {
                Toast.makeText(getApplicationContext(), "Harus Mengisi Semua Form", Toast.LENGTH_SHORT).show();
            } else {
                Intent intent = new Intent(LoginActivity.this, MainActivity.class);
                startActivity(intent);
                finish();
            }
        }
    }

    @Override
    public void onBackPressed() {
        Intent intent = new Intent(LoginActivity.this, MainActivity.class);
        startActivity(intent);
        finish();
    }
}
```

Gambar 5.17 Codingan Pada RegisterActivity.java

* Codingan Pada Register

```
package com.scanner.scantobar;  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
public class RegisterActivity extends AppCompatActivity  
DatabaseHelper db;  
Button login, register;  
EditText username, password, passwordConf;  
@Override  
protected void onCreate(Bundle savedInstanceState)  
super.onCreate(savedInstanceState);  
setContentView(R.layout.activity_register);  
db = new DatabaseHelper(this);  
username = (EditText)  
findViewById(R.id.editText_usernameRegist);  
password = (EditText)  
findViewById(R.id.editText_passwordRegist);  
passwordConf = (EditText)  
findViewById(R.id.editText_passwordConfRegist);  
login = (Button) findViewById(R.id.btn_loginRegist);  
register = (Button)  
findViewById(R.id.btn_registerRegist);  
//login  
login.setOnClickListener(new View.OnClickListener())  
@Override  
public void onClick(View v)  
Intent LoginIntent = new  
Intent(RegisterActivity.this, LoginActivity.class);  
startActivity(LoginIntent);  
finish();
```

```
);

//register
register.setOnClickListener(new View.OnClickListener()
@Override
public void onClick(View v)
String strUsername =
username.getText().toString();
String strPassword =
password.getText().toString();
String strPasswordConf =
passwordConf.getText().toString();
if (strPassword.equals(strPasswordConf))
Boolean daftar =
db.insertUser(strUsername, strPassword);
if (daftar == true)
Toast.makeText(getApplicationContext(), "Daftar Berhasil",
Toast.LENGTH_SHORT).show();
Intent LoginIntent = new
Intent(RegisterActivity.this,
LoginActivity.class);
startActivity(LoginIntent);
finish();
else
Toast.makeText(getApplicationContext(), "Daftar Gagal",
Toast.LENGTH_SHORT).show();
else
Toast.makeText(getApplicationContext(), "Password Tidak Cocok",
Toast.LENGTH_SHORT).show();
);
```

7. Layout Activity Login

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Email"
        android:textColor="#000000"/>
    <EditText
        android:id="@+id/editText_email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textColor="#000000"/>
    <EditText
        android:id="@+id/editText_password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <Button
        android:id="@+id/button_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

```

Gambar 5.18 Codingan Pada ActivityLogin.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Email"
        android:textColor="#000000"/>
    <EditText
        android:id="@+id/editText_email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textColor="#000000"/>
    <EditText
        android:id="@+id/editText_password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <Button
        android:id="@+id/button_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

```

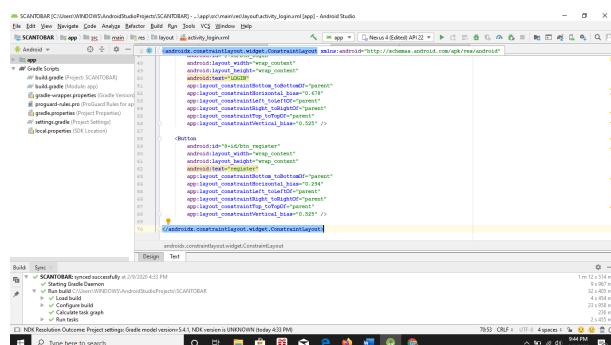
Gambar 5.19 Codingan Pada ActivityLogin.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:orientation="vertical">
    <Button
        android:id="@+id/button_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

```

Gambar 5.20 Codingan Pada ActivityLogin.xml



Gambar 5.21 Codingan Pada ActivityLogin.xml

* Codingan Pada Activity Login

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LoginActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="LOGIN"
        android:textColor="@android:color/black"
        android:textSize="28dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintHorizontal_bias="0.501"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.177" />

    <EditText
        android:id="@+id/editText_username"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#D9D9D9"
        android:fontFamily="sans-serif-normal"
        android:gravity="center"
        android:imeOptions="actionNext"
        android:inputType="text"
        android:padding="10dp"
        android:textColor="#000000"/>

```

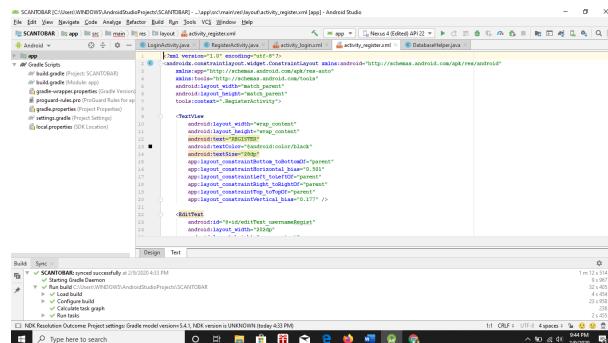
```
    android:layout_width = "202dp"
    android:layout_height = "wrap_content"
    android:hint="username"
    app:layout_constraintBottom_toBottomOf = "parent"
    app:layout_constraintLeft_toLeftOf = "parent"
    app:layout_constraintRight_toRightOf = "parent"
    app:layout_constraintTop_toTopOf = "parent"
    app:layout_constraintVertical_bias = "0.347" / >
    |EditText
        android:id="@+id/editText_password"
        android:layout_width = "202dp"
        android:layout_height = "wrap_content"
        android:hint="password"
        android:inputType="textPassword"
        app:layout_constraintBottom_toBottomOf = "parent"
        app:layout_constraintHorizontal_bias = "0.502"
        app:layout_constraintLeft_toLeftOf = "parent"
        app:layout_constraintRight_toRightOf = "parent"
        app:layout_constraintTop_toTopOf = "parent"
        app:layout_constraintVertical_bias = "0.435" / >
    |Button
        android:id="@+id/btn_login"
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:text="LOGIN"
        app:layout_constraintBottom_toBottomOf = "parent"
        app:layout_constraintHorizontal_bias = "0.678"
        app:layout_constraintLeft_toLeftOf = "parent"
        app:layout_constraintRight_toRightOf = "parent"
        app:layout_constraintTop_toTopOf = "parent"
        app:layout_constraintVertical_bias = "0.525" / >
    |Button
        android:id="@+id/btn_register"
        android:layout_width = "wrap_content"
```

```

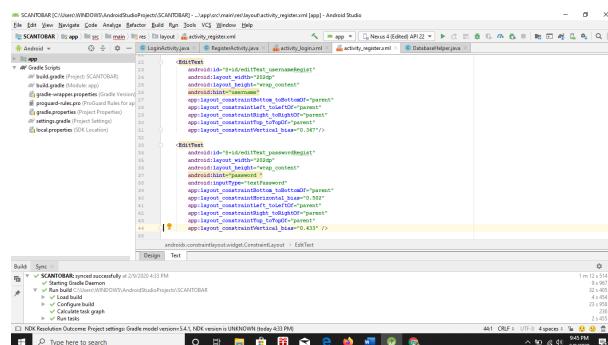
        android:layout_height = "wrap_content"
        android:text="register"
    app:layout_constraintBottom_toBottomOf = "parent"
    app:layout_constraintHorizontal_bias = "0.294"
    app:layout_constraintLeft_toLeftOf = "parent"
    app:layout_constraintRight_toRightOf = "parent"
    app:layout_constraintTop_toTopOf = "parent"
    app:layout_constraintVertical_bias = "0.525" />
    androidx.constraintlayout.widget.ConstraintLayout

```

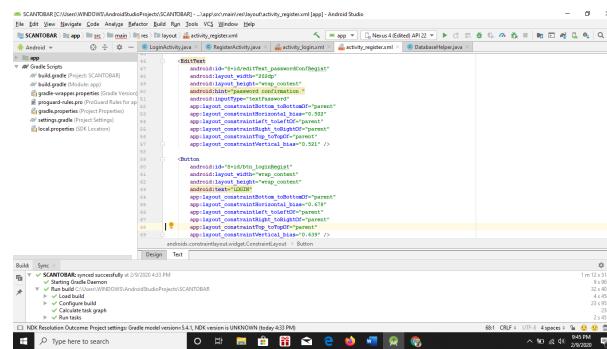
8. Layout Activity Register



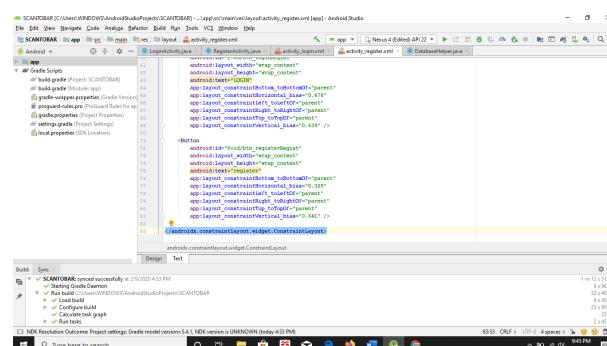
Gambar 5.22 Codingan Pada ActivityRegister.xml



Gambar 5.23 Codingan Pada ActivityRegister.xml



Gambar 5.24 Codingan Pada ActivityRegister.xml



Gambar 5.25 Codingan Pada ActivityRegister.xml

```
* Codingan Activity Register
?xml version="1.0" encoding="utf-8"?
androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width = "match_parent"
android:layout_height = "match_parent"
tools:context=".RegisterActivity";
    |TextView
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:text="REGISTER"
        android:textColor="@android:color/black"
        android:textSize="28dp"
        app:layout_constraintBottom_toBottomOf = "parent"
        app:layout_constraintHorizontal_bias = "0.501"
        app:layout_constraintLeft_toLeftOf = "parent"
        app:layout_constraintRight_toRightOf = "parent"
        app:layout_constraintTop_toTopOf = "parent"
        app:layout_constraintVertical_bias = "0.177" / >
    |EditText
        android:id="@+id/editText_usernameRegist"
        android:layout_width = "202dp"
        android:layout_height = "wrap_content"
        android:hint="username"
        app:layout_constraintBottom_toBottomOf = "parent"
        app:layout_constraintLeft_toLeftOf = "parent"
        app:layout_constraintRight_toRightOf = "parent"
        app:layout_constraintTop_toTopOf = "parent"
        app:layout_constraintVertical_bias = "0.347" / >
    |EditText
        android:id="@+id/editText_passwordRegist"
        android:layout_width = "202dp"
```

```
    android:layout_height = "wrap_content"
    android:hint="password "
    android:inputType="textPassword"
    app:layout_constraintBottom_toBottomOf = "parent"
    app:layout_constraintHorizontal_bias = "0.502"
    app:layout_constraintLeft_toLeftOf = "parent"
    app:layout_constraintRight_toRightOf = "parent"
    app:layout_constraintTop_toTopOf = "parent"
    app:layout_constraintVertical_bias = "0.433" / >
    |EditText
    android:id="@+id/editTextpasswordConfRegist"
    android:layout_width = "202dp"
    android:layout_height = "wrap_content"
    android:hint="password confirmation "
    android:inputType="textPassword"
    app:layout_constraintBottom_toBottomOf = "parent"
    app:layout_constraintHorizontal_bias = "0.502"
    app:layout_constraintLeft_toLeftOf = "parent"
    app:layout_constraintRight_toRightOf = "parent"
    app:layout_constraintTop_toTopOf = "parent"
    app:layout_constraintVertical_bias = "0.521" / >
    |Button
    android:id="@+id/btnloginRegist"
    android:layout_width = "wrap_content"
    android:layout_height = "wrap_content"
    android:text="LOGIN"
    app:layout_constraintBottom_toBottomOf = "parent"
    app:layout_constraintHorizontal_bias = "0.678"
    app:layout_constraintLeft_toLeftOf = "parent"
    app:layout_constraintRight_toRightOf = "parent"
    app:layout_constraintTop_toTopOf = "parent"
    app:layout_constraintVertical_bias = "0.639" / >
    |Button
    android:id="@+id/btnregisterRegist"
```

```
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:text="register"
        app:layout_constraintBottom_toBottomOf = "parent"
        app:layout_constraintHorizontal_bias = "0.325"
        app:layout_constraintLeft_toLeftOf = "parent"
        app:layout_constraintRight_toRightOf = "parent"
        app:layout_constraintTop_toTopOf = "parent"
        app:layout_constraintVertical_bias = "0.641" / >
    androidx.constraintlayout.widget.ConstraintLayout
```

9. Database Pada Aplikasi

The screenshot shows the Android Studio interface with the code editor open to the DatabaseHelper.java file. The code defines a SQLiteOpenHelper subclass named DatabaseHelper that extends SQLiteOpenHelper. It includes methods for onCreate, onUpgrade, and onConfigure. The onCreate method creates a table named 'User' with columns '_id', 'username', and 'password'. The onUpgrade method handles database version changes by dropping the old table and creating a new one. The onConfigure method sets the database name to 'SCANTOMAR.db'. The code uses the Context.MODE_PRIVATE flag for the database.

```
package com.smartron.scantomar;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import android.widget.Toast;

public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "SCANTOMAR.db";
    private static final int DATABASE_VERSION = 1;

    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE User (username TEXT, password TEXT)");
    }

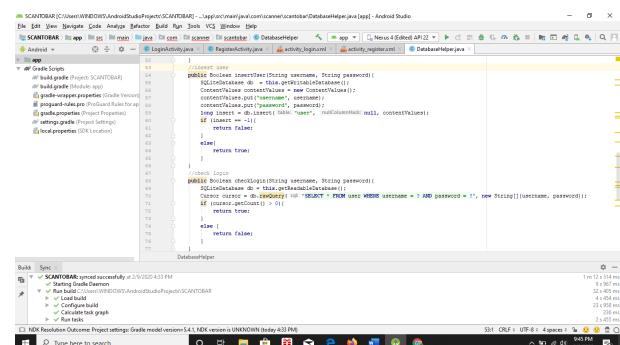
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS user");
        db.execSQL("CREATE TABLE IF NOT EXISTS user");
        onCreate(db);
    }

    public void onConfigure(SQLiteDatabase db) {
        db.setConfig(SQLiteDatabase.CONFIG_ENABLE懦, true);
        db.setConfig(SQLiteDatabase.CONFIG_ENABLE懦, true);
        db.setConfig(SQLiteDatabase.CONFIG_ENABLE懦, true);
    }
}
```

Gambar 5.26 Codingan Pada Database

```
public void upgradeSession(String sessionName, int id) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put("name", sessionName);
    contentValues.put("id", id);
    contentValues.put("username", null);
    db.update("sessions", contentValues, "name=?", new String[]{sessionName});
    if (upgrade == true)
        contentValues.put("username", null);
    else
        contentValues.put("username", sessionName);
    db.update("sessions", contentValues, "id=?", new String[]{id}, contentValues);
}
```

Gambar 5.27 Codingan Pada Database



Gambar 5.28 Codingan Pada Database

* Codingan Pada Database

```

package com.scanner.scantobar;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import androidx.annotation.Nullable;

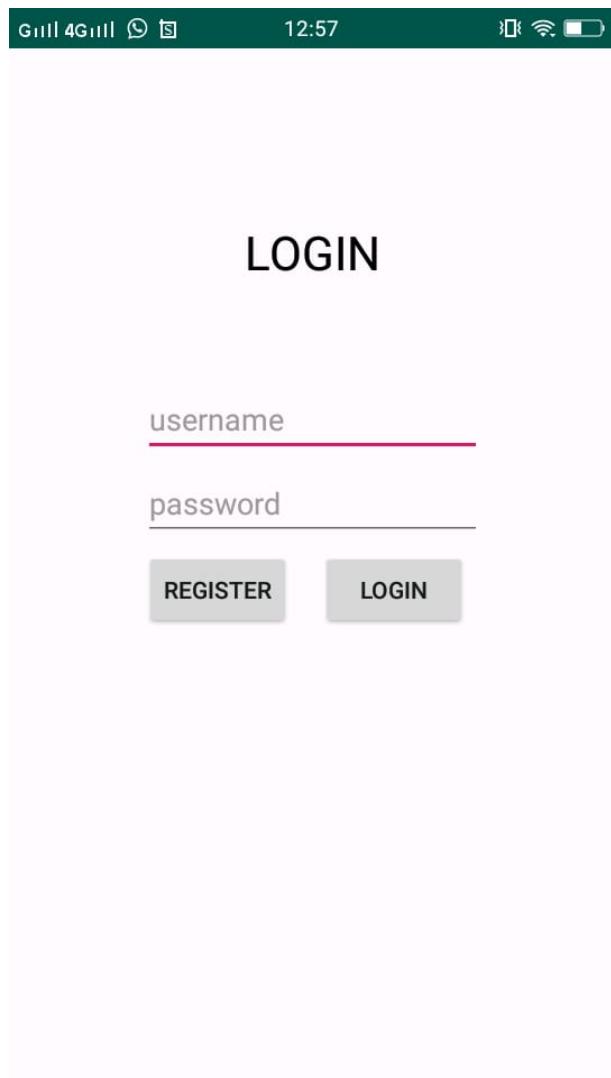
public class DatabaseHelper extends SQLiteOpenHelper
    public DatabaseHelper(Context context)
        super(context, "LoginSQLite.db", null, 1);
    @Override
    public void onCreate(SQLiteDatabase db)
        db.execSQL("CREATE TABLE session(id integer PRIMARY KEY, login text)");
        db.execSQL("CREATE TABLE user(id integer PRIMARY KEY AUTOINCREMENT, username text, password text)");
        db.execSQL("INSERT INTO session(id, login) VALUES(1, 'kosong')");
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
        int newVersion)
        db.execSQL("DROP TABLE IF EXISTS session");
        db.execSQL("DROP TABLE IF EXISTS user");
    onCreate(db);
}

```

```
//check session
public Boolean checkSession(String sessionValues)
SQLiteDatabase db = this.getReadableDatabase();
Cursor cursor = db.rawQuery( "SELECT * FROM session
WHERE login = ?", new String[]{sessionValues});
if (cursor.getCount() > 0)
return true;
else
return false;
//upgrade session
public Boolean upgradeSession(String sessionValues, int id)
SQLiteDatabase db = this.getWritableDatabase();
ContentValues contentValues = new ContentValues();
contentValues.put("login", sessionValues);
long update = db.update( "session", contentValues, "id="+id, null);
if (update == -1)
return false;
else
return true;
//insert user
public Boolean insertUser(String username, String
password)
SQLiteDatabase db = this.getWritableDatabase();
ContentValues contentValues = new ContentValues();
contentValues.put("username", username);
contentValues.put("password", password);
long insert = db.insert("user", null, contentValues);
if (insert == -1)
return false;
else
return true;
//check login
public Boolean checkLogin(String username, String password)
SQLiteDatabase db = this.getReadableDatabase();
```

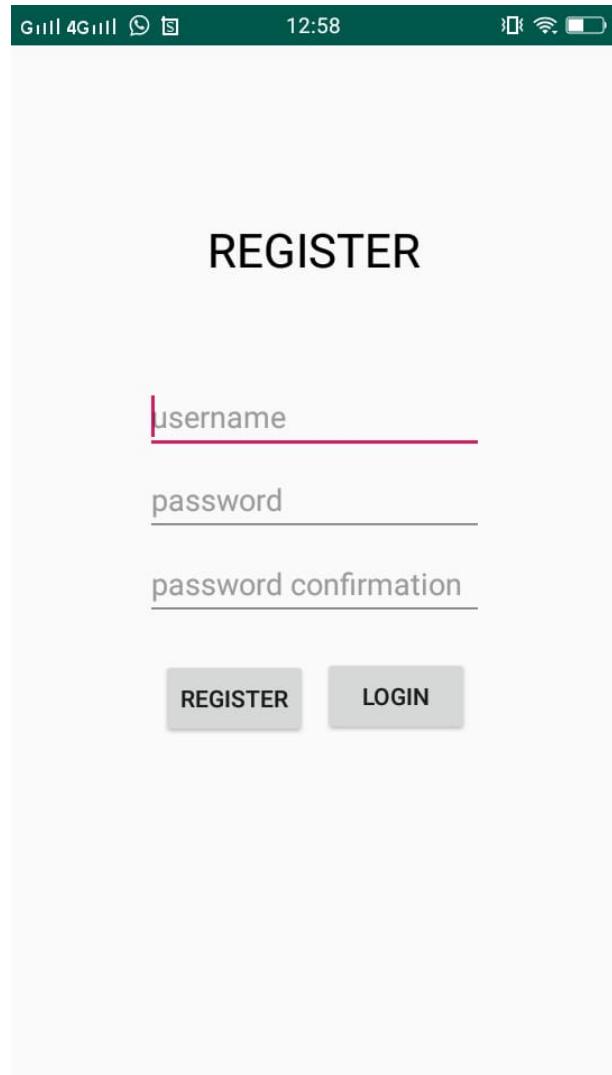
```
Cursor cursor = db.rawQuery("SELECT * FROM user WHERE
username = ? AND password = ?", new String[]{username, password});
if (cursor.getCount() > 0)
return true;
else
return false;
```

5.1.5 Implementasi Tampilan Utama



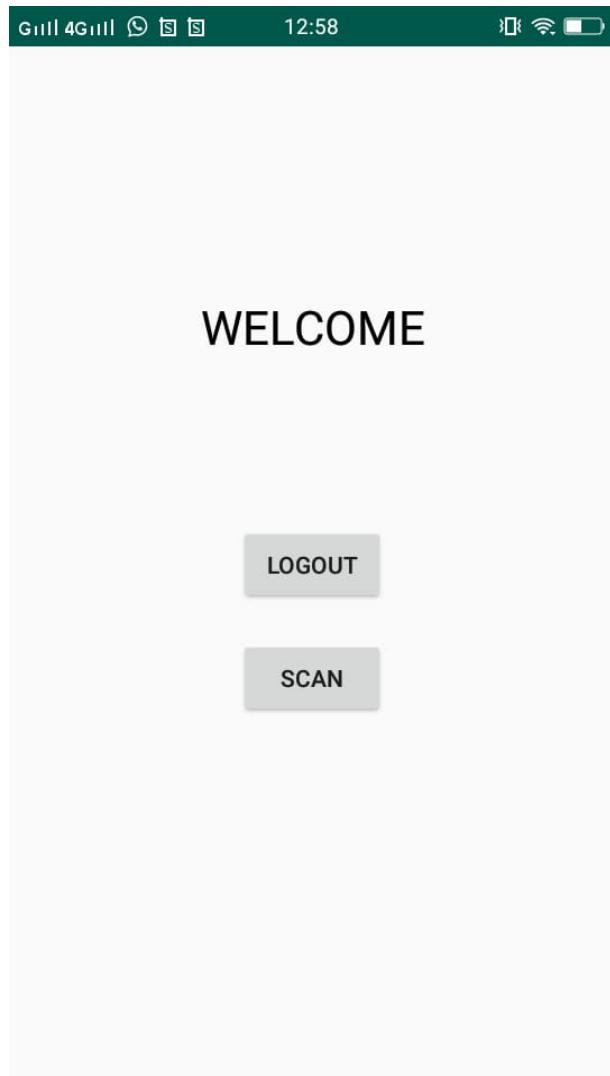
Gambar 5.29 Tampilan Menu Login

Membuka aplikasi maka akan tampil form login yang berisikan password dan user-name.



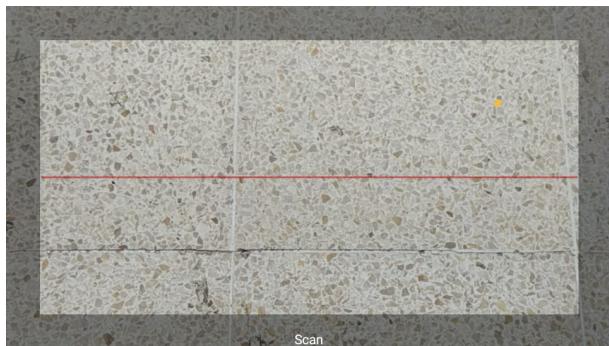
Gambar 5.30 Tampilan Menu Register

Jika belum memiliki akun maka dapat melakukan registrasi terlebih dahulu dengan mengisikan username dan password serta confirmasi password.



Gambar 5.31 *Tampilan Halaman Utama*

Setelah kita berhasil Login kita akan ditampilkan pada menu utama dimana terdapat pilihan scan atau logout.



Gambar 5.32 Tampilan Halaman Scan QR

Pada halaman scan ini kita dapat scan QR Code dengan hanya mengarahkan kamera hp pada QR Code yang dimaksudkan.

BAB 6

MFCC DAN NEURAL NETWORK

6.1 Teori

1. Kenapa fi

le suara harus di lakukan MFCC. dilengkapi dengan ilustrasi atau gambar.

Nilai-nilai MFCC meniru pendengaran manusia dan mereka biasanya digunakan dalam aplikasi pengenalan suara serta genre musik deteksi. Nilai-nilai MFCC ini akan dimasukkan langsung ke jaringan saraf. Agar dapat diubah menjadi bentuk vektor, dan dapat digunakan pada machine learning. Disebabkan machine learning hanya mengerti bilangan vektor saja. Ilustrasinya, Ketika ingin menggunakan file suara dalam machine learning, misalnya untuk melihat jam. Machine learning tidak memahami rekaman suara melainkan vektor. Maka rekaman tersebut akan diubah kedalam bentuk vektor kemudian vektor akan menyesuaikan dengan kata kata yang sudah disediakan. Jika cocok maka akan mengembalikan waktu yang diinginkan

2. Konsep dasar neural network dilengkapi dengan ilustrasi atau gambar

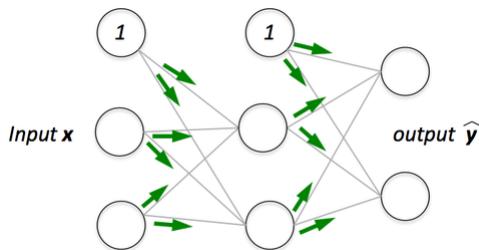
Neural Network ini terinspirasi dari jaringan saraf otak manusia. Dimana setiap neuron terhubung ke setiap neuron di lapisan berikutnya. Lapisan pertama

menerima input dan lapisan terakhir memberikan keluaran. Struktur jaringan, yang berarti jumlah neuron dan koneksinya, diputuskan sebelumnya dan tidak dapat berubah, setidaknya tidak selama training. Juga, setiap input harus memiliki jumlah nilai yang sama. Ini berarti bahwa gambar, misalnya, mungkin perlu diubah ukurannya agar sesuai dengan jumlah neuron input.

Ilustrasinya. misalkan kita ingin encode sebuah kalimat yaitu "what time is it" kemudian Anda menginisialisasi lapisan jaringan Anda dan hidden state. Bentuk dan dimensi hidden state akan tergantung pada bentuk dan dimensi jaringan saraf berulang Anda. Kemudian Anda mengulangi input Anda, meneruskan kata dan hidden state ke NN. NN mengembalikan output dan kondisi tersembunyi yang dimodifikasi. Anda terus mengulang sampai Anda kehabisan kata-kata. Terakhir Anda melewatkannya ke layer feedforward, dan itu mengembalikan prediksi. Bahwa kita ingin mengetahui pukul berapa sekarang.

- Konsep pembobotan dalam neural network. dilengkapi dengan ilustrasi atau gambar

Bobot mewakili kekuatan koneksi antar unit. Jika bobot dari node 1 ke node 2 memiliki besaran lebih besar, itu berarti bahwa neuron 1 memiliki pengaruh lebih besar terhadap neuron 2. Bobot penting untuk nilai input. Bobot mendekati nol berarti mengubah input ini tidak akan mengubah output. Bobot negatif berarti meningkatkan input ini akan mengurangi output. Bobot menentukan seberapa besar pengaruh input terhadap output. Seperti contoh berikut :



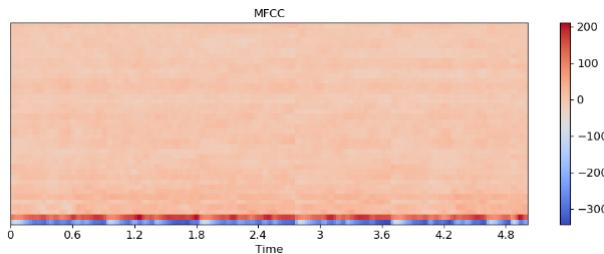
Gambar 6.1 Contoh Pembobotan Neural Network Tasya

- Konsep fungsi aktifasi dalam neural network. dilengkapi dengan ilustrasi atau gambar

Fungsi aktivasi digunakan untuk memperkenalkan non-linearitas ke jaringan saraf. Ini menekan nilai dalam rentang yang lebih kecil yaitu. fungsi aktivasi Sigmoid memeras nilai antara rentang 0 hingga 1. Ada banyak fungsi aktivasi yang digunakan dalam industri pembelajaran yang dalam dan ReLU, SeLU dan TanH lebih disukai daripada fungsi aktivasi sigmoid. Ilustrasinya, ketika fungsi aktivasi linier, jaringan saraf dua lapis mampu mendekati hampir semua fungsi. Namun, jika fungsi aktivasi identik dengan fungsi aktivasi $F(X) = X$, properti

ini tidak puas, dan jika MLP menggunakan fungsi aktivasi yang sama, seluruh jaringan setara dengan jaringan saraf lapis tunggal.

- Cara membaca hasil plot dari MFCC,dilengkapi dengan ilustrasi atau gambar Berikut merupakan hasil plot dari rekaman suara : Dari gambar tersebut dapat



Gambar 6.2 Cara Membaca Hasil Plot MFCC Tasya

diketahui :

- Terdapat 2 dimensi yaitu x sebagai waktu, dan y sebagai power atau desibel.
- Dapat dilihat bahwa jika berwarna biru maka power dari suara tersebut rendah, dan jika merah power dari suara tersebut tinggi
- Dibagian atas terdapat warna merah pudar yang menandakan bahwa tidak ada suara sama sekali dalam jangkauan tersebut.

- Jelaskan apa itu one-hot encoding,dilengkapi dengan ilustrasi kode dan atau gambar.

One-hot encoding adalah representasi variabel kategorikal sebagai vektor biner. Mengharuskan nilai kategorikal dipetakan ke nilai integer. Kemudian, setiap nilai integer direpresentasikan sebagai vektor biner yang semuanya bernilai nol kecuali indeks integer, yang ditandai dengan 1.

Label Encoding			One Hot Encoding				
Food Name	Categorical #	Calories					
Apple	1	95	→	Apple	Chicken	Broccoli	Calories
Chicken	2	231		1	0	0	95
Broccoli	3	50		0	1	0	231
				0	0	1	50

Gambar 6.3 One Hot Encoding Tasya

- fungsi dari np.unique dan to categorical dalam kode program,dilengkapi dengan ilustrasi atau gambar. Untuk np unique fungsinya yaitu menemukan elemen

unik array. Mengembalikan elemen unik array yang diurutkan. Ada tiga output opsional selain elemen unik:

- Indeks array input yang memberikan nilai unik
- Indeks array unik yang merekonstruksi array input
- Berapa kali setiap nilai unik muncul dalam array input.

```
>>> np.unique([1, 1, 2, 2, 3, 3])
array([1, 2, 3])
>>> a = np.array([[1, 1], [2, 3]])
>>> np.unique(a)
array([1, 2, 3])
```

Gambar 6.4 Numpy Unique Tasya

Untuk To Categorical fungsinya untuk mengubah vektor kelas (integer) ke matriks kelas biner.

```
# Consider an array of 5 labels out of a set of 3 classes {0, 1, 2}:
> labels
array([0, 2, 1, 2, 0])
# `to_categorical` converts this into a matrix with as many
# columns as there are classes. The number of rows
# stays the same.
> to_categorical(labels)
array([[ 1.,  0.,  0.],
       [ 0.,  0.,  1.],
       [ 0.,  1.,  0.],
       [ 0.,  0.,  1.],
       [ 1.,  0.,  0.]], dtype=float32)
```

Gambar 6.5 To Categorical Tasya

8. Fungsi dari Sequential dalam kode program,dilengkapi dengan ilustrasi atau gambar. Sequential berfungsi sebagai tumpukan linear lapisan. COntohnya sebagai berikut :

```
from keras.models import Sequential
from keras.layers import Dense

model = Sequential()
model.add(Dense(2, input_dim=1))
model.add(Dense(1))
```

Gambar 6.6 Sequential Tasya

6.2 Praktek Program

6.2.1 GTZAN Genre Collection dan data dari freesound

1. GTZAN Genre Collection berisikan klasifikasi genre musik. Terdapat 1000 audio dengan durasi maksimal 30 detik dan terdapat 10 genre musik didalamnya. Dalam setiap genre berisikan 100 tracks musik.

2. Data dari Freesound berisikan instrument alat musik tertentu dalam bentuk wav

Untuk Meload Data tersebut untuk digunakan pada MFCC caranya dapat dilihat seperti pada listing berikut.

```
1 \begin{verbatim}
2 import librosa
3 import librosa.feature
4 import librosa.display
5 import glob
6 import numpy as np
7 import matplotlib.pyplot as plt
8 from keras.models import Sequential
9 from keras.layers import Dense, Activation
10 from keras.utils import np_utils import to_categorical
11
12 # In []
13 audio_path = 'genres/classical/classical.00010.au'
14 x , sr = librosa.load(audio_path)
15 print(type(x), type(sr))
16 \end{verbatim}
```

Listing 6.1 Kode Load Data Untuk MFCC

- PEnjelasannya sebagain berikut :
- Codingan Diatas akan meload libray librosa yang akan digunakan untuk menggunakan mfcc
- Librosafeature akan meload feature dari librosa
- Librosadisplay akan mengambil fungsi display pada librosa
- glob merupakan modul pada python yang digunakan untuk meload segala jenis format file termasuk musik
- mengimport numpy sebagai np yang digunakan untuk data array dari musik
- import matplotlib untuk melakukan plotting dari audio
- Mengimport modul Sequential dari librari Keras untuk membuat suatu model
- Dense dan Activation sebagai Operasi linier di mana setiap input terhubung ke setiap output dengan bobot atau weight.
- Dari library keras akan meload modul to_categorical
- Kemudian untuk me load datanya, disini variabel audio_path berisikan direktori file tujuan yang digunakan.
- Variabel x dan sr berguna untuk meload variabel audio_path menggunakanlibrari Librosa
- Kemudia print atau tampilkan x dan s dalam bentuk array.Hasilnya seperti berikut :

```
In [28]: audio_path = 'genres/classical/classical.00010.au'
...: x , sr = librosa.load(audio_path)
...: print(type(x), type(sr))
<class 'numpy.ndarray'> <class 'int'>
```

Gambar 6.7 Meload Data Genre Collection Tasya

6.2.2 Fungsi Display MFCC

Berikut merupakan Code dari fungsi Display mfcc: Penjelasan dari code diatas yaitu

```
In [3]: def display_mfcc(song):
...:     y, sr = librosa.load(song)
...:     mfcc = librosa.feature.mfcc(y)
...:
...:     plt.figure(figsize=(10, 4))
...:     librosa.display.specshow(mfcc, x_axis='time', y_axis='mel')
...:     plt.colorbar()
...:     plt.title(song)
...:     plt.tight_layout()
...:     plt.show()
```

Gambar 6.8 Display MFCC Tasya

:

- def display_mfcc yaitu kita akan mendefinisikan fungsi yang diberinama display_mfcc dengan inputan song
- Variabel y akan meload variabel song
- Variabel MFCC akan menggunakan feauture mfcc pada Librosa untuk melakukan konversi audio menjadi bentuk vektor
- Kemudian hasil tadi akan diplotting.
- Berikut merupakan contoh dari plotting audio dari genre Classical. Ketikan kode berikut yang dimana akan memanggil fungsi dispkay mfcc utuk plotting dari audioyang dituju dapat dilihat dilisting berikut .

```
1 \begin{verbatim}
2 display_mfcc ('genres / classical / classical.00010.au ')
3 \end{verbatim}
```

Listing 6.2 Code Fungsi Display MFCC

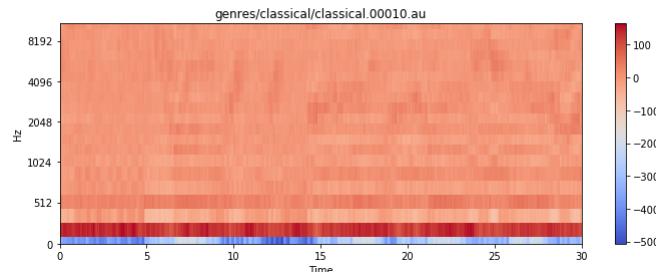
- Hasilnya sebagai berikut

6.2.3 Fungsi Extract Features Song

Berikut merupakan code dari fungsi extract features song : Penjelasan dari code diatas yaitu :

- Variabel y akan melaod variabel atau inputan f menggunakan librari Librosa
- Variabel mfcc akan melakukan mfcc dari variabel y

```
In [31]: display_mfcc('genres/classical/classical.00010.au')
```



Gambar 6.9 Hasil Display MFCC Tasya

```
In [7]: def extract_features_song(f):
    ...
    y, _ = librosa.load(f)
    ...
    # get Mel-frequency cepstral coefficients
    mfcc = librosa.feature.mfcc(y)
    ...
    # normalize values between -1,1 (divide by max)
    mfcc /= np.max(np.abs(mfcc))
    ...
    return np.ndarray.flatten(mfcc)[:25000]
```

Gambar 6.10 Extract Features Tasya

- Variabel mfcc kemudian akan dibagi oleh numpy amax dan dikembalikan lagi nilainya ke variabel mfcc.
- Hasil Tadi kemudian akan ditampilkan dalam bentuk array dengan mengambil 25000 row pertama
- Mengapa mengambil 25000 row pertama? dikarenakan Audio yang terdapat pada dataset ini tidak menentu durasinya. Dan kita harus mengambil data yang memiliki durasi yang sama untuk mempermudah dalam melakukan training.

6.2.4 Fungsi Generate Features And Labels

Berikut merupakan code dari fungsi Generate Features And Labels : Penjelasan dari code diatas yaitu :

- Variabel all features berisikan array kosong
- Variabel all labels berisikan array kosong
- Variabel genres disesuaikan dengan nama folder sebelumnya, dan berisikan folder folder dari genre yang ada
- Melakukan looping, untuk folder tadi
- Variabel sound_files akan mengambil file dari folder genres dan mengambil semua file dengan ekstensi au didalamnya.

```
In [8]: def generate_features_and_labels():
...:     all_features = []
...:     all_labels = []
...:
...:     genres = ['blues', 'classical', 'country', 'disco', 'hiphop', 'jazz',
...: 'metal', 'pop', 'reggae', 'rock']
...:     for genre in genres:
...:         sound_files = glob.glob('genres/' + genre + '/*.au')
...:         print('Processing %d songs in %s genre...' % (len(sound_files), genre))
...:         for f in sound_files:
...:             features = extract_features_song(f)
...:             all_features.append(features)
...:             all_labels.append(genre)
...:
...:     # convert labels to one-hot encoding cth blues : 10000000000 classic
...:     01000000000
...:     label_uniq_ids, label_row_ids = np.unique(all_labels,
...: return_inverse=True) #ke integer
...:     label_row_ids = label_row_ids.astype(np.int32, copy=False)
...:     onehot_labels = to_categorical(label_row_ids, len(label_uniq_ids)) #ke one
...: hot
...:     ...
...:     return np.stack(all_features), onehot_labels
```

Gambar 6.11 Fungsi Generate Features And Labels Tasya

- print akan menampilkan Tulisan Prossesing dengan jumlah file didalam folder dan nama folder tersebut
- Memanggil fungsi extract_features_song kedalam inputan f dalam sound_files dan melakukan vektorisasi dimasukan kedalam variabel features.
- Semua fitures akan dimasukan kedalam all_features.
- Semua genre diamsukan ke all_labels.
- Variabel label_uniq_ids dan label_row_ids mendefinisikan label unique dari all_labels kedalam bentuk integer.
- Kemudian variabel onehot_labels akan mengubahnya ke dalam bentuk one hot encoding dengan menggunakan to_categorical.Sehingga dimensinya menjadi 1000 x 10 dikarenakan terdapat 1000 lagu dan 10 binari untuk merepresentasikan one-hot encodingnya.
- Mengembalikan all_features dan onehot_labels kedalam satu matriks.

6.2.5 Penggunaan Fungsi Generate Features And Labels Sangat Lama Ketika Meload Dataset Genre

Dikarenakan TERdapat 10 folder dengan genre berbeda, dan didalamnya terdapat 100 audio. Dari setiap folder itu akan dilakukan features dan perubahan label. Karena banyaknya jumlah file maka proses loadnya pun lama. Berikut codingannya :

```
1 \begin{verbatim}
2 features , labels = generate_features_and_labels()
3 \end{verbatim}
```

Listing 6.3 Panggil Genenrate Labels

Akan didapatkan hasil seperti berikut yang dimana menunjukan proses bahwa sedang dilakukan ekstraksi audio ke features dan labels :

```
In [9]: features, labels = generate_features_and_labels()
Processing 100 songs in blues genre...
Processing 100 songs in classical genre...
Processing 100 songs in country genre...
Processing 100 songs in disco genre...
Processing 100 songs in hiphop genre...
Processing 100 songs in jazz genre...
Processing 100 songs in metal genre...
Processing 100 songs in pop genre...
Processing 100 songs in reggae genre...
Processing 100 songs in rock genre...
```

Gambar 6.12 Hasil Fungsi Generate Features And Labels Tasya

6.2.6 Pemisahan Data Training Dan Data Set Sebesar 80%

Pemisahan 80% digunakan untuk memudahkan dalam melakukan pengacakan atau pengocokan nantinya. Dimana 80% merupakan data training dan sisanya 20% merupakan datatestnya. data training perlu lebih banyak agar saat dilakukan pengocokan tidak teracak dalam urutan yang berbeda. Berikut code programnya pada listing ini :

```

1 \begin{verbatim}
2 training_split = 0.8
3
4 # last column has genre, turn it into unique ids
5 alldata = np.column_stack((features, labels))
6
7 np.random.shuffle(alldata)
8 splitidx = int(len(alldata) * training_split)
9 train, test = alldata[:splitidx, :], alldata[splitidx:,:]
10
11 print(np.shape(train))
12 print(np.shape(test))
13
14 train_input = train[:, :-10]
15 train_labels = train[:, -10:]
16
17 test_input = test[:, :-10]
18 test_labels = test[:, -10:]
19
20 print(np.shape(train_input))
21 print(np.shape(train_labels))
22 \end{verbatim}
```

Listing 6.4 Code Pemisahan Data Training Dan Testing

Penjelasan dari code diatas :

- Training split akan memisahkan training set sebanyak 80%
- Melakukan penumpukan features dan labels
- Melakukan Pengocokan untuk alldata dengan mengalikan isi dari alldata dengan training_split
- Memisahkan mana yang termasuk data train dan mana yang termasuk data test

- Menampilkan isi dari train dan test. Dapat dilihat bahwa untuk training terdapat 800 kolom dan untuk test 200 kolom dengan jumlah baris yang sama yaitu 25010

```
In [13]: np.random.shuffle(alldata)
...: splitidx = int(len(alldata) * training_split)
...: train, test = alldata[:splitidx,:], alldata[splitidx:,:]

In [14]: print(np.shape(train))
...: print(np.shape(test))
(800, 25010)
(200, 25010)
```

Gambar 6.13 Pemisahan Data Training dan Data Set Tasya

- Variabel train_input akan berisikan train dengan mengecualikan 10 baris terakhir
- Variabel train_labels berisikan train dengan mengambil sisa dari train_input atau hanya mengambil 10 baris terakhir saja
- Untuk variabel test_input dan test_label sama penjelasannya seperti diatas.
- Baris selanjutnya digunakan untuk menampilkan isi atau shape dari hasil training dan testing barusan, seperti berikut :

```
In [15]: train_input = train[:, :-10]
...: train_labels = train[:, -10:]

In [16]: test_input = test[:, :-10]
...: test_labels = test[:, -10:]

In [17]: print(np.shape(train_input))
...: print(np.shape(train_labels))
(800, 25000)
(800, 10)
```

Gambar 6.14 Pemisahan Data Training dan Data Set Tasya

6.2.7 Fungsi Sequential

Berikut code lengkapnya :

```
1 \begin{verbatim}
2 model = Sequential([
3     Dense(100, input_dim=np.shape(train_input)[1]),
4     Activation('relu'),
5     Dense(10),
6     Activation('softmax'),
7 ])
8 \end{verbatim}
```

Listing 6.5 Code Fungsi Sequential

Hasilnya seperti berikut : Dari hasil diatas dapat dijelaskan bahwa :

- Layer pertama dense dari 100 neuron untuk inputan

```
In [18]: model = Sequential([
    ...:     Dense(100, input_dim=np.shape(train_input)[1]),
    ...:     Activation('relu'),
    ...:     Dense(10),
    ...:     Activation('softmax'),
    ...: ])
WARNING:tensorflow:From E:\Anaconda3\lib\site-packages\tensorflow\python\framework\op_def_library.py:263: colocate_with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future version.
Instructions for updating:
Colocations handled automatically by placer.
```

Gambar 6.15 Pemisahan Data Training dan Data Set Tasya

- Activationnya menggunakan fungsi relu yaitu jika ada inputan dengan nilai maksimum maka inputan itu yang akan terpilih.
- Dense 10 mengkategorikan 10 neuron untuk jenis genrenya untuk output nya.
- Untuk dense diatas aktivasinya menggunakan fungsi Softmax

6.2.8 Fungsi Compile

Berikut code lengkapnya :

```
1 \begin{verbatim}
2 model.compile(optimizer='adam',
3                 loss='categorical_crossentropy',
4                 metrics=['accuracy'])
5 print(model.summary())
6 \end{verbatim}
```

Listing 6.6 Code Fungsi Compile

Hasilnya seperti berikut : Dari hasil diatas dapat dijelaskan bahwa :

```
In [19]: model.compile(optimizer='adam',
...:                     loss='categorical_crossentropy',
...:                     metrics=['accuracy'])
...: print(model.summary())
Layer (type)                  Output Shape           Param #
dense_1 (Dense)               (None, 100)          2500100
activation_1 (Activation)      (None, 100)          0
dense_2 (Dense)               (None, 10)           1010
activation_2 (Activation)      (None, 10)           0
=====
Total params: 2,501,110
Trainable params: 2,501,110
Non-trainable params: 0
None
```

Gambar 6.16 Fungsi Compile Tasya

- Menggunakan algortima adam sebagai optimizer. Adam yaitu algoritme pen-goptimalan yang dapat digunakan sebagai ganti dari prosedur penurunan gra-dien stokastik klasik untuk memperbarui bobot jaringan yang berulang berdasarkan data training.
- Loss nya menggunakan categorical_crossentropy untuk fungsi optimasi skor

6.2.9 Fungsi Fit

Berikut code lengkapnya :

```

1 \begin{verbatim}
2 model.fit(train_input , train_labels , epochs=10, batch_size=32,
3           validation_split=0.2)
4 \end{verbatim}
```

Listing 6.7 Code Fungsi Fit

Hasilnya seperti berikut : Dari hasil diatas dapat dijelaskan bahwa :

```

In [20]: model.fit(train_input, train_labels, epochs=10, batch_size=32,
...:           validation_split=0.2)
WARNING:tensorflow:From E:\Anaconda3\lib\site-packages\tensorflow\python\ops
\math_ops.py:3066: to_int32 (from tensorflow.python.ops.math_ops) is deprecated and w
be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Train on 640 samples, validate on 160 samples
Epoch 1/10
640/640 [=====] - 8s 13ms/step - loss: 2.1030 - acc: 0.2781
val_loss: 1.8095 - val_acc: 0.3500
Epoch 2/10
640/640 [=====] - 2s 4ms/step - loss: 1.3568 - acc: 0.5422 -
val_loss: 1.7641 - val_acc: 0.3937
Epoch 3/10
640/640 [=====] - 2s 3ms/step - loss: 1.0330 - acc: 0.6750 -
val_loss: 1.5691 - val_acc: 0.4125
Epoch 4/10
640/640 [=====] - 2s 3ms/step - loss: 0.7832 - acc: 0.7797 -
val_loss: 1.7463 - val_acc: 0.4188
Epoch 5/10
640/640 [=====] - 2s 3ms/step - loss: 0.5983 - acc: 0.8406 -
val_loss: 1.5068 - val_acc: 0.4688
Epoch 6/10
640/640 [=====] - 2s 3ms/step - loss: 0.4406 - acc: 0.9266 -
val_loss: 1.5222 - val_acc: 0.4875
Epoch 7/10
640/640 [=====] - 2s 3ms/step - loss: 0.3159 - acc: 0.9688 -
...: 100%|██████████| 640/640 - 8s 13ms/step - loss: 0.3159 - acc: 0.9688 -
```

Gambar 6.17 Fungsi Fit Tasya

- Melakukan pelatihan dengan epoch atau iterasi dengan rambatan balik sebanyak 10, kemudian dalam sekali epochs dilakukan 32 sampel yang diproses sebelum model diperbarui.
- Validation_split sebesar 20% untuk melakukan pengecekan pada cross score validation

6.2.10 Fungsi Evaluate

Berikut code lengkapnya :

```

1 \begin{verbatim}
2 loss , acc = model.evaluate(test_input , test_labels , batch_size=32)
3
4 print("Done!")
5 print("Loss: %.4f, accuracy: %.4f" %(loss , acc))
6 \end{verbatim}
```

Listing 6.8 Code Fungsi Evaluate

```
In [21]: loss, acc = model.evaluate(test_input, test_labels, batch_size=32)
200/200 [=====] - 0s 2ms/step
```

Gambar 6.18 Fungsi Evaluate Tasya

Hasilnya seperti berikut : Dari hasil diatas dapat dijelaskan bahwa :

- Melakukan evaluasi atau menemukan model terbaik yang mewakili data dan seberapa baik model yang dipilih akan bekerja di masa depan. Menggunakan test input dan test label.
- Kemudian Hasilnya dapat dilihat seperti berikut :

```
In [22]: print("Done!")
...: print("Loss: %.4f, accuracy: %.4f" % (loss, acc))
Done!
Loss: 1.3620, accuracy: 0.5350
```

Gambar 6.19 Fungsi Evaluate Tasya

- Dimana Loss yaitu hasil prediksi yang salah sebanyak 1,3620 dan keakurasiannya yaitu 53%

6.2.11 Fungsi Predict

Berikut code lengkapnya :

```
1 \begin{verbatim}
2 model.predict(test_input[:1])
3 \end{verbatim}
```

Listing 6.9 Code Fungsi Predict

Hasilnya seperti berikut : Dari hasil diatas dapat dijelaskan bahwa :

```
In [23]: model.predict(test_input[:1])
Out[23]:
array([[2.5113127e-01, 7.7192662e-03, 5.1440198e-02, 6.4860745e-03,
       6.4208927e-03, 6.2923378e-01, 3.7388336e-06, 2.6338635e-02,
       3.9711967e-03, 1.7254945e-02]], dtype=float32)
```

Gambar 6.20 Fungsi Predict Tasya

- Untuk melakukan prediksi diambil satu baris dari test_input
- Nilai yang tertinggi terdapat pada label kedua atau genre Classical
- Untuk baris yang dipilih prediksi yang tepat yaitu lagu tersebut termasuk kedalam genre Classical

6.3 Penanganan Error

6.3.1 Module Eror

Berikut merupakan eror yang dijumpai ketika menjalankan skrip diatas

```
...: import glob
...: import numpy as np
...: import matplotlib.pyplot as plt
...: from keras.models import Sequential
...: from keras.layers import Dense, Activation
...: from keras.utils import to_categorical
Traceback (most recent call last):
File "<ipython-input-1-3bc11cc81b24>", line 1, in <module>
    import librosa
ModuleNotFoundError: No module named 'librosa'
```

Gambar 6.21 Module Error Tasya

- Jenis error tersebut adalah ModuleNotFoundError : No Module named 'librosa' . Error ini terjadi dikarenakan target atau tujuannya tidak terinstall. Maka yang perlu dilakukan yaitu :
 - Buka Anconda atau conda prompt
 - Ketikan 'conda install -c conda-forge librosa dan enter
 - Tunggu sampai proses instalasi berhasil
 - Jika Sudah, jalankan kembali skrip tadi maka hasilnya seperti berikut

```
In [2]: import librosa
...: import librosa.feature
...: import librosa.display
...: import glob
...: import numpy as np
...: import matplotlib.pyplot as plt
...: from keras.models import Sequential
...: from keras.layers import Dense, Activation
...: from keras.utils import to_categorical
Using TensorFlow backend.
```

Gambar 6.22 Penyelesaian Module Error Tasya

- Tandanya eror sudah berhasil ditangani

BAB 7

CNN

7.1 Teori

7.1.1 Teks Tokenizer

Untuk memudahkan mesin memahami maksud dari apa yang kita inginkan dalam machine learning, kata pada teks disebut token, dan proses vektorisasi dari bentuk kata ke dalam token tersebut disebut tokenizer dan tokenizer akan merubah sebuah teks menjadi simbol, kata, ataupun biner dan bentuk lainnya kedalam token. Untuk lebih jelasnya perhatikan ilustrasi berikut. Disini saya mempunyai sebuah kalimat yaitu "Nama Saya Tasya Wiendhyra" maka ketika kita lakukan proses tokenizer maka akan berubah menjadi ['Nama', 'Saya', 'Tasya', 'Wiendhyra].

7.1.2 konsep dasar K Fold Cross Validation pada dataset komentar Youtube

```
1 kfolds = StratifiedKFold(n_splits=5)
2 splits = kfolds.split(d, d['CLASS'])
```

Listing 7.1 K Fold Cross Validation

StartifiedKFold berisikan presentasi sampel untuk setiap kelas. Dimana dalam ilustrasi ini sampel dibagi menjadi 5 dalam setiap class nya. Kemudian sampel tadi akan dimasukan kedalam class dari dataset youtube tadi.

Untuk ilustrasi lebih jelasnya, ada pada gambar berikut :

```
>>> from sklearn.model_selection import StratifiedKFold
>>> X = np.array([[1, 2], [3, 4], [1, 2], [3, 4]])
>>> y = np.array([0, 0, 1, 1])
>>> skf = StratifiedKFold(n_splits=2)
>>> skf.get_n_splits(X, y)
2
>>> print(skf)
StratifiedKFold(n_splits=2, random_state=None, shuffle=False)
>>> for train_index, test_index in skf.split(X, y):
...     print("TRAIN:", train_index, "TEST:", test_index)
...     X_train, X_test = X[train_index], X[test_index]
...     y_train, y_test = y[train_index], y[test_index]
TRAIN: [1 3] TEST: [0 2]
TRAIN: [0 2] TEST: [1 3]
```

Gambar 7.1 Ilustrasi KFold Cross Tasya

7.1.3 kode program for train, test in splits

Maksudnya yaitu untuk menguji apakah setiap data pada dataset sudah di split dan tidak terjadi penumpukan. Yang dimana maksudnya di setiap class tidak akan muncul id yang sama. Ilustrasinya misalkan kita memiliki 4 baju dengan model yang berbeda. Kemudian kita bagikan kedua anak, tentunya setiap anak yang menerima baju tidak memiliki baju yang sama modelnya.

7.1.4 Jelaskan apa maksudnya kode program *train_content = df['CONTENT']*. dan *test_content = df['CONTENT'].iloc[test_idx]*. dilengkapi dengan ilustrasi atau gambar

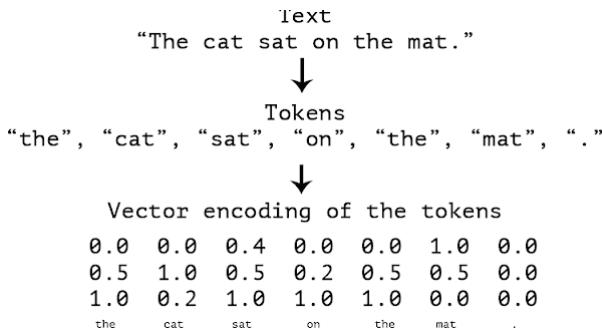
Maksudnya yaitu mengambil data pada kolom atau index CONTENT yang merupakan bagian dari train_idx dan test_idx. Ilustrasinya, ketika data telah diubah menjadi train dan test maka kita dapat memilihnya untuk ditampilkan pada kolom yang diinginkan.

7.1.5 Soal No. 5 Jelaskan apa maksud dari fungsi *tokenizer = Tokenizer(num_words=2000)* dan *tokenizer.fit_on_texts(train_content)*, dilengkapi dengan ilustrasi atau gambar

Dimana variabel tokenizer akan melakukan vektorisasi kata menggunakan fungsi Tokenizer yang dimana jumlah kata yang ingin diubah kedalam bentuk token adalah 2000 kata. Dan untuk *tokenizer.fit_on_texts(train_content)* maksudnya kita akan melakukan fit tokenizer hanya untuk dat trainnya saja tidak dengan data test nya untuk kolom CONTENT. Ilustrasinya, Jadi, jika Anda memberikannya sesuatu seperti, "Kucing itu duduk di atas tikar." Ini akan membuat kamus s.t. word_index ['the'] = 0; word_index ['cat'] = 1 itu adalah kata -*ç* kamus indeks sehingga setiap kata mendapat nilai integer yang unik.

7.1.6 Jelaskan apa maksud dari fungsi `d_train_inputs = tokenizer.texts_to_matrix(mode='tfidf')` dan `d_test_inputs = tokenizer.texts_to_matrix(test_content, mode='tfidf')`, dilengkapi dengan ilustrasi kode dan atau gambar

Maksudnya yaitu untuk variabel d_train_inputs akan melakukan tokenizer dari bentuk teks ke matrix dari data train_content dengan mode matriksnya yaitu tfidf begitu juga dengan variabel d_test_inputs untuk data test. Berikut gambar ilustrasinya



Gambar 7.2 Ilustrasi Text To Matrix Tasya

7.1.6.1 Jelaskan apa maksud dari fungsi `d_train_inputs = d_train_inputs/npamax(np.abs(d_train_inputs))` dan `d_test_inputs = d_test_inputs/np.absolute(d_test_inputs)`, dilengkapi dengan ilustrasi atau gambar Fungsi tersebut akan membagi matrix tfidf tadi dengan amax yaitu mengembalikan maksimum array atau maksimum sepanjang sumbu. Yang hasilnya akan dimasukan kedalam variabel d_train_inputs untuk data train dan d_test_inputs untuk data test dengan nominal absolut atau tanpa ada bilangan negatif dan koma.

```

>>> x = np.array([-1.2, 1.2])
>>> np.absolute(x)
array([ 1.2,  1.2])
  
```

Gambar 7.3 Ilustrasi np Absolute Tasya

7.1.6.2 Jelaskan apa maksud fungsi dari `d_train_outputs = np_utils.to_categorical(d['CLASS'])` dan `d_test_outputs = np_utils.to_categorical(d['CLASS'].iloc[test_idx])`, dilengkapi dengan ilustrasi atau gambar Dalam variabel d_train_output dan d_test_outputs akan dilakukan one hot encoding, dimana np_utilsakan mengubah vektor dengan bentuk integer ke matriks kelas biner untuk kolom CLASS dimana nantinya hanya akan ada dua pilihan yaitu 1 atau 0. 1 untuk spam 0 untuk non spam atau sebaliknya. Berikut gambar ilustrasinya :

```

model = Sequential()
  
```

```

> labels
array([0, 2, 1, 2, 0])
# `to_categorical` converts this into a matrix with as many
# columns as there are classes. The number of rows
# stays the same.
> to_categorical(labels)
array([[ 1.,  0.,  0.],
       [ 0.,  0.,  1.],
       [ 0.,  1.,  0.],
       [ 0.,  0.,  1.],
       [ 1.,  0.,  0.]], dtype=float32)

```

Gambar 7.4 Ilustrasi One Hot Encoding Tasya

```

2 model.add(Dense(512, input_shape=(2000,)))
3 model.add(Activation('relu'))
4 model.add(Dropout(0.5))
5 model.add(Dense(2))
6 model.add(Activation('softmax'))

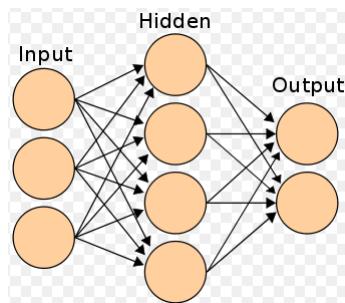
```

Listing 7.2 Membuat model Neural Network

Penjelasannya sebagai berikut :

- Melakukan pemodelan Sequential
- Layer pertama dense dari 512 neuron untuk inputan dengan inputan tadi yang sudah dijadikan matriks sebanyak 2000
- Activationnya menggunakan fungsi relu yaitu jika ada inputan dengan nilai maksimum maka inputan itu yang akan terpilih.
- Dropout ini untuk melakukan pembobotan, dimana pembobotan hanya dilakukan 50% saja agar tidak terjadi penumpukan data dari dense inputan tadi
- Dense 2 mengkategorikan 2 neuron untuk output nya yaitu 1 dan 0.
- Untuk dense diatas aktivasinya menggunakan fungsi Softmax.

Ilustrasinya seperti berikut :



Gambar 7.5 Ilustrasi Neural Network Pemodelan Tasya

```

1 model.compile(loss='categorical_crossentropy', optimizer='adamax',
2 metrics=['accuracy'])

```

Listing 7.3 Compile model

Melakukan peng compile-an dari model Sequential tadi dengan Loss yang dengang merupakan fungsi optimisasi skor menggunakan categorical_crossentropy , dan menggunakan algoritma adam sebagai optimizer. Adam yaitu algoritma pengoptimalan yang dapat digunakan sebagai ganti dari prosedur penurunan gradien stokastik klasik untuk memperbarui bobot jaringan yang berulang berdasarkan data training.Dengan metrik yaitu fungsi yang digunakan untuk menilai kinerja mode Anda disini menggunakan fungsi accuracy.

7.1.6.5 Jelaskan apa itu Deep Learning Deep Learning adalah subbidang machine learning yang berkaitan dengan algoritma yang terinspirasi oleh struktur dan fungsi otak yang disebut jaringan saraf tiruan atau Artificial Neural Networks. Jaringan saraf tiruan, algoritma yang terinspirasi oleh otak manusia, belajar dari sejumlah besar data. Demikian pula dengan bagaimana kita belajar dari pengalaman, algoritma pembelajaran yang mendalam akan melakukan tugas berulang kali, setiap kali sedikit mengubahnya untuk meningkatkan hasilnya.

7.1.6.6 Jelaskan apa itu Deep Neural Network, dan apa bedanya dengan Deep Learning Deep Neural Network adalah jaringan syaraf tiruan (JST) dengan beberapa lapisan antara lapisan input dan output. DNN menemukan manipulasi matematis yang benar untuk mengubah input menjadi output, apakah itu hubungan linear atau hubungan non-linear. Merupakan jaringan syaraf dengan tingkat kompleksitas tertentu, jaringan syaraf dengan lebih dari dua lapisan. Deep Neural Network menggunakan pemodelan matematika yang canggih untuk memproses data dengan cara yang kompleks.

DNN hanya terdiri dari dua laipsan yaitu input dan output, sedangkan dalam Deep learning kita dapat mendefinisikan layer sebanyak yang kita inginkan atau butuhkan.

7.1.6.7 Jelaskan dengan ilustrasi gambar buatan sendiri(langkah per langkah) bagaimana perhitungan algoritma konvolusi dengan ukuran stride (NPM mod3+1) x (NPM mod3+1) yang terdapat max pooling Stridennya 3

- terdapat data seperti berikut

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

Dengan $g(x,y)$

0	2	-1
-1	0	2
0	1	1

Gambar 7.6 Algoritma Konvolusi Tasya

- Kemudian hitung konvolusi untuk setiap matriksnya seperti berikut :

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12					

$$(0*1)+(2*2)+(-1*5)+(-1*3)+(0*2)+(2*5)+(0*3)+(1*2)+(1*4)= 12$$

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6				

$$(0*5)+(2*2)+(-1*1)+(-1*5)+(0*1)+(2*1)+(0*4)+(1*1)+(1*5)= 6$$

Gambar 7.7 Algoritma Konvolusi Tasya

– pertama

– Kedua

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15			

$$(0*2)+(2*2)+(-1*4)+(-1*1)+(0*1)+(2*5)+(0*1)+(1*5)+(1*3)= 15$$

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15			

$$(0*3)+(2*2)+(-1*4)+(-1*5)+(0*3)+(2*5)+(0*5)+(1*1)+(1*1)= 7$$

Gambar 7.8 Algoritma Konvolusi Tasya

– Ketiga

– Keempat

– Kelima

- Didapatkan hasil akhir nilai konvolusi dan juga max poolingnya seperti berikut

7.1.7 Praktek

Keterangannya sebagai berikut :

- Pertama kita akan mengimpor librari csv

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15
7	8	22

$$(0^*4)+(2^*1)\{(-1^*5)+(-1^*5)+(0^*1)+(2^*3)+(0^*1)+(1^*5)+(1^*5)=8$$

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15
7	8	22

$$(0^*1)+(2^*5)+(-1^*3)+(-1^*1)+(0^*3)+(2^*5)+(0^*5)+(1^*5)+(1^*1)=22$$

Gambar 7.9 Algoritma Konvolusi Tasya

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15
7	8	22

$$(0^*5)+(2^*3)+(-1^*5)+(-1^*5)+(0^*1)+(2^*1)+(0^*5)+(1^*5)+(1^*4)=7$$

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15
7	8	22

$$(0^*5)+(2^*1)+(-1^*3)+(-1^*1)+(0^*5)+(2^*5)+(0^*4)+(1^*3)+(1^*3)=14$$

Gambar 7.10 Algoritma Konvolusi Tasya

1	2	5	2	1	4
3	2	5	1	1	5
3	2	4	1	5	3
5	3	5	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

0	2	-1
-1	0	2
0	1	1

konvolusi→

12	6	15
7	8	22
7	14	1

$$(0^*1)+(2^*3)+(-1^*5)+(-1^*5)+(0^*5)+(2^*1)+(0^*3)+(1^*3)+(1^*1)=1$$

Gambar 7.11 Algoritma Konvolusi Tasya

- Dimana dari librai PIL atau Pillow atau Python Imaging Library akan diimpor modul Image yang di inisiasikan sebagai pil_image. Modul Image menyediakan kelas dengan nama yang sama yang digunakan untuk mewakili gambar PIL. Modul ini juga menyediakan sejumlah fungsi pabrik, termasuk fungsi untuk memuat image dari file, dan untuk membuat image baru.

Hasil konvolusi akhir

1	2	5	2	1	4
3	12	6	15	1	5
3	7	8	22	5	3
5	7	14	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

Kemudian untuk max pooling 2x2 dengan stride 3 karena $(1164086 \bmod 3 + 1 = 3)$ adalah

Menjadi

1	2	5	2	1	4
3	12	6	15	1	5
3	7	8	22	5	3
5	7	14	1	3	5
5	1	1	5	5	1
5	5	4	3	3	1

12	15	5
7	22	5
5	5	5

Gambar 7.12 Algoritma Konvolusi Tasya

- mengimpor librari image dari keras . Yang menghasilkan kumpulan data gambar tensor dengan augmentasi data waktu nyata. Data akan diulang (dalam batch).
- Berikut Hasilnya :

```
In [1]: import csv
... from PIL import Image as pil_image
... import keras.preprocessing.image
Using TensorFlow backend.
```

Gambar 7.13 Kode Program Blok In 1 Tasya

```
1 for row in csvreader:
2     if i > 0:
3         img = keras.preprocessing.image.img_to_array(pil_image.
open("HASYv2/" + row[0]))
4             # neuron activation functions behave best when input
values are between 0.0 and 1.0 (or -1.0 and 1.0),
5             # so we rescale each pixel value to be in the range 0.0
to 1.0 instead of 0-255
6             img /= 255.0
7             imgs.append((row[0], row[2], img))
8             classes.append(row[2])
9     i += 1
```

Keterangannya sebagai berikut :

- variabel imgs berisikan array kosong
- Variabel classes berisikan array kosong
- Membuka file csv dari Folder HSYv2 dengan nama file hasy-data-labels.csv sebagai csvfile
- Variabel csvreader akan menggunakan fungsi reader pada library csv untuk membaca file csv tadi yang disimpan di csvfile.

- Dimana variabel i dimulai dari nol.
- Untuk setiap baris pada csvreader
- Jika i lebih besar dari 0
- Jadi itu akan mengambil contoh Gambar PIL dan mengubahnya menjadi array numpy dengan mengambil data dari HSYv2 dan dimulai dari baris ke nol.
- Hasil dari variabel img akan dibagi dengan 255.0
- .append akan membuat list array baru untuk baris 0 baris 2 pada img.
- Menyimpan setiap class nya pada baris 2
- Penambahan i sebanyak 1.
- Hasilnya seperti berikut :

```
In [2]: imgs = []
...: classes = []
...: with open('HSYv2/hasy-data-labels.csv') as csvfile:
...:     csvreader = csv.reader(csvfile)
...:     i = 0
...:     for row in csvreader:
...:         if i > 0:
...:             img =
...:             keras.preprocessing.image.img_to_array(pil_image.open("HSYv2/" + row[0]))
...:             # neuron activation functions behave best when input values are
...:             # between 0.0 and 1.0 (or -1.0 and 1.0),
...:             # so we rescale each pixel value to be in the range 0.0 to 1.0
...:             instead of 0-255
...:             img /= 255.0
...:             imgs.append((row[0], row[2], img))
...:             classes.append(row[2])
...:         i += 1
```

Gambar 7.14 Kode Program Blok In 2 Tasya

Keterangannya sebagai berikut :

- Impor librari Random dari Python
- Melakukan pengacakan untuk imgs dengan Metode Shuffle untuk mengocok urutan di tempat. yaitu, mengubah posisi item dalam daftar.
- Membagi data dari imgs dengan cara mengalikan 80% dengan jumlah data dari imgs.
- Untuk data train mengambil hasil dari perhitungan sebelumnya.
- Untuk data test mengambil sisa dari jumlah yang telah dijadikan data train
- Hasilnya seperti berikut :

```
In [3]: import random
...: random.shuffle(imgs)
...: split_idx = int(0.8*len(imgs))
...: train = imgs[:split_idx]
...: test = imgs[split_idx:]
```

Gambar 7.15 Kode Program Blok In 3 Tasya

```
1 train_output = np.asarray(list(map(lambda row: row[1], train)))
2 test_output = np.asarray(list(map(lambda row: row[1], test)))
```

Keterangannya sebagai berikut :

- Impor librari Numpy yang di inisiasikan sebagai np
- Variabel train_input mengubah input menjadi sebuah array yang diambil dari baris 2, data train.
- Variabel test_input mengubah input menjadi sebuah array yang diambil dari baris 2, data test.
- Variabel train_output mengubah input menjadi sebuah array yang diambil dari baris 1, data train.
- Variabel train_output mengubah input menjadi sebuah array yang diambil dari baris 1, data test.
- Hasilnya seperti berikut

```
In [4]: import numpy as np
...:
...: train_input = np.asarray(list(map(lambda row: row[2], train)))
...: test_input = np.asarray(list(map(lambda row: row[2], test)))
...:
...: train_output = np.asarray(list(map(lambda row: row[1], train)))
...: test_output = np.asarray(list(map(lambda row: row[1], test)))
```

Gambar 7.16 Kode Program Blok In 4 Tasya

Keterangannya sebagai berikut :

- Impor Fungsi LabelEncoder
- Impor Fungsi OneHotEncoder
- Berikut hasilnya :

```
In [5]: from sklearn.preprocessing import LabelEncoder
...: from sklearn.preprocessing import OneHotEncoder
```

Gambar 7.17 Kode Program Blok In 5 Tasya

Keterangannya sebagai berikut :

- Variabel label_encoder akan memanggil fungsi LabelEncoder tadi.
- variabel integer_encoded akan menggunakan labelencoder untuk melakukan fit pada classes agar berubah datanya menjadi integer.
- Berikut hasilnya :

```
In [6]: label_encoder = LabelEncoder()
...: integer_encoded = label_encoder.fit_transform(classes)
```

Gambar 7.18 Kode Program Blok In 6 Tasya

Keterangannya sebagai berikut :

- Variabel onehot_encoder akan memanggil fungsi OneHotEncoder dimana tidak berisikan matriks sparse.
- Pada variabel integer_encoded akan diubah bentuknya dimana setiap nilai integer akan direpresentasikan sebagai vektor binari dengan nilai 0 kecuali index dari integer tersebut ditandai dengan 1.
- Melakukan fit untuk one hot encoder kedalam integer_encoder.
- Berikut hasilnya :

```
In [7]: onehot_encoder = OneHotEncoder(sparse=False)
...: integer_encoded = integer_encoded.reshape(len(integer_encoded), 1)
...: onehot_encoder.fit(integer_encoded)
C:/Users/DidinIrFandi/Anaconda3/lib/site-packages/sklearn/preprocessing/_encoders.py:
371: FutureWarning: The handling of integer data will change in version 0.22. Currently,
the categories are determined based on the range [0, max(values)], while in the future
they will be determined based on the unique values.
If you want the future behaviour and silence this warning, you can specify
"categories='auto'".
In case you used a LabelEncoder before this OneHotEncoder to convert the categories to
integers, then you can now use the OneHotEncoder directly.
warnings.warn(msg, FutureWarning)
Out[7]:
OneHotEncoder(categorical_features=None, categories=None,
               dtype=<class 'numpy.float64'>, handle_unknown='error',
               n_values=None, sparse=False)
```

Gambar 7.19 Kode Program Blok In 7 Tasya

```
| print("Number of classes: %d" % num_classes)
```

Keterangannya sebagai berikut :

- Variabel train_output_int akan mengubah data dari train_output menjadi LabelEncoder
- Dimana pada train_output setelah diubah labelnya menjadi integer dilakukan one hot encoding diambil dari train_output_int dan menggunakan .reshape untuk memberikan bentuk baru ke array tanpa mengubah datanya dengan keterangan jika index dari integer tersebut ditandai dengan 1 dan sisanya yang bukan nol.
- Variabel test_output_int akan mengubah data dari test_output menjadi LabelEncoder
- Dimana pada train_output setelah diubah labelnya menjadi integer dilakukan one hot encoding diambil dari test_output_int dan menggunakan .reshape untuk memberikan bentuk baru ke array tanpa mengubah datanya dengan keterangan jika index dari integer tersebut ditandai dengan 1 dan sisanya yang bukan nol.
- Variabel num_classes akan menampilkan jumlah data dari classes yang telah dilakukan label encoder
- Menampilkan tulisan "Number of classes : %d" dimana mengembalikan nilai integer dari num_classes.
- Hasilnya sebagai berikut :

```
In [8]: train_output_int = label_encoder.transform(train_output)
....: train_output =
onehot_encoder.transform(train_output_int.reshape(len(train_output_int), 1))
....: test_output_int = label_encoder.transform(test_output)
....: test_output =
onehot_encoder.transform(test_output_int.reshape(len(test_output_int), 1))
....:
....: num_classes = len(label_encoder.classes_)
....: print("Number of classes: %d" % num_classes)
Number of classes: 369
```

Gambar 7.20 Kode Program Blok In 8 Tasya

Keterangannya sebagai berikut :

- Impor Sequential dari model pada librari Keras.
- Impor Dense, Dropout, Flatten dari modul Layers pada librari Keras.
- Impor Conv2D, MaxPooling2D dari modul Layers pada librari Keras.
- Hasilnya seperti berikut :

```
In [9]: from keras.models import Sequential  
...: from keras.layers import Dense, Dropout, Flatten  
...: from keras.layers import Conv2D, MaxPooling2D
```

Gambar 7.21 Kode Program Blok In 9 Tasya

```
1 model.add(Flatten())  
2 model.add(Dense(1024, activation='tanh'))  
3 model.add(Dropout(0.5))  
4 model.add(Dense(num_classes, activation='softmax'))  
5  
6 model.compile(loss='categorical_crossentropy', optimizer='adam',  
7 metrics=['accuracy'])  
8  
9 print(model.summary())
```

Keterangannya sebagai berikut :

- Melakukan pemodelan Sequential.
- Menambahkan Konvolusi 2D dengan 32 filter konvolusi masing-masing berukuran 3x3 dengan algoritam activation relu dengan data dari train_input mulai dari baris nol.
- Menambahkan Max Pooling dengan matriks 2x2.
- Dilakukan lagi penambahan Konvolusi 2D dengan 32 filter konvolusi masing-masing berukuran 3x3 dengan algoritam activation relu.
- Menambahkan lagi Max Pooling dengan matriks 2x2.
- Mendefinisikan inputan dengan 1024 neuron dan menggunakan algoritma tanh untuk activationnya.
- Dropout terdiri dari pengaturan secara acak tingkat pecahan unit input ke 0 pada setiap pembaruan selama waktu pelatihan, yang membantu mencegah overfitting sebesar 50% .
- Untuk output layer menggunakan data dari variabel num_classes dengan furgsi activationnya softmax.
- Mengonfigurasi proses pembelajaran, yang dilakukan melalui metode compile,sebelum melatih suatu model.
- Menampilkan atau mencetak representasi ringkasan model yang telah dibuat.
- Hasilnya sebagai berikut :

Layer (type)	Output Shape	Param #
<hr/>		
conv2d_1 (Conv2D)	(None, 30, 30, 32)	896
max_pooling2d_1 (MaxPooling2D)	(None, 15, 15, 32)	0
conv2d_2 (Conv2D)	(None, 13, 13, 32)	9248
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 32)	0
flatten_1 (Flatten)	(None, 1152)	0
dense_1 (Dense)	(None, 1024)	1180672
dropout_1 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 369)	378225
<hr/>		
Total params: 1,569,041		
Trainable params: 1,569,041		
Non-trainable params: 0		
<hr/>		
None		

Gambar 7.22 Kode Program Blok In 10 Tasya

Keterangannya sebagai berikut :

- Impor Modul Callbacks dari Librari Keras.
- Variabel callback mendefinisikan Callback ini untuk menulis log untuk TensorBoard, yang memungkinkan Anda untuk memvisualisasikan grafik dinamis dari pelatihan dan metrik pengujian Anda, serta histogram aktivasi untuk berbagai lapisan dalam model Anda.
- Hasilnya sebagai berikut :

```
In [11]: import keras.callbacks
...: tensorboard = keras.callbacks.TensorBoard(log_dir='./logs/mnist-style')
```

Gambar 7.23 Kode Program Blok In 11 Tasya

```
1 score = model.evaluate(test_input , test_output , verbose=2)
2 print('Test loss:', score[0])
3 print('Test accuracy:', score[1])
```

Keterangannya sebagai berikut :

- Melakukan fit model dengan 32 ukuran subset dari sampel pelatihan Anda
- Epoch sebanyak 10 kali
- Vebrose=2 maksudnya menampilkan nomor dari epoch yang sedang berjalan atau yang sudah dijalankan.
- Validasi plit sebanyak 20% sebagai fraksi data pelatihan untuk digunakan sebagai data validasi.

- Menggunakan TensorBoard sebagai callback untuk diterapkan selama pelatihan dan validasi.
- Variabel score mengembalikan nilai evaluate untuk menampilkan data lost dan data accuracy dari test
- Menampilkan data loss dengan menghitung jumlah kemunculan nol .
- Menampilkan data accuracy dengan menghitung jumlah kemunculan 1.
- Berikut hasilnya :

```

Train on 107668 samples, validate on 26918 samples
Epoch 1/10
- 240s - loss: 1.5635 - acc: 0.6235 - val_loss: 1.0161 - val_acc: 0.7296
Epoch 2/10
- 232s - loss: 0.9918 - acc: 0.7267 - val_loss: 0.9090 - val_acc: 0.7448
Epoch 3/10
- 231s - loss: 0.8709 - acc: 0.7520 - val_loss: 0.8584 - val_acc: 0.7662
Epoch 4/10
- 244s - loss: 0.8009 - acc: 0.7660 - val_loss: 0.8609 - val_acc: 0.7572
Epoch 5/10
- 241s - loss: 0.7482 - acc: 0.7770 - val_loss: 0.8634 - val_acc: 0.7593
Epoch 6/10
- 242s - loss: 0.7076 - acc: 0.7847 - val_loss: 0.8703 - val_acc: 0.7624
Epoch 7/10
- 235s - loss: 0.6778 - acc: 0.7917 - val_loss: 0.8623 - val_acc: 0.7653
Epoch 8/10
- 232s - loss: 0.6474 - acc: 0.7988 - val_loss: 0.8708 - val_acc: 0.7636
Epoch 9/10
- 233s - loss: 0.6232 - acc: 0.8043 - val_loss: 0.8925 - val_acc: 0.7608
Epoch 10/10
- 234s - loss: 0.6033 - acc: 0.8077 - val_loss: 0.8951 - val_acc: 0.7570
Test loss: 0.8862248992399043
Test accuracy: 0.7599191607031478

```

Gambar 7.24 Kode Program Blok In 12 Tasya

```

1     for dropout in [0.0, 0.25, 0.50, 0.75]:
2         model = Sequential()
3         for i in range(conv2d_count):
4             if i == 0:
5                 model.add(Conv2D(32, kernel_size=(3, 3),
6 activation='relu', input_shape=np.shape(train_input[0])))
6             else:
7                 model.add(Conv2D(32, kernel_size=(3, 3),
8 activation='relu'))
9                 model.add(MaxPooling2D(pool_size=(2, 2)))
10                model.add(Flatten())
11                model.add(Dense(dense_size, activation='tanh'))
12                if dropout > 0.0:
13                    model.add(Dropout(dropout))
14                    model.add(Dense(num_classes, activation='softmax'))

```

Keterangannya sebagai berikut :

- impor modul time dari python anaconda
- Variabel result berisikan array kosong.
- Menggunakan convolution 2D yang dimana akan memiliki 1 atau 2 layer.

- Mendefinisikan dense_size dengan ukuran 128, 256, 512, 1024, 2048
- Mendefinsikan drop_out dengan 0, 25%, 50%, dan 75%
- Melakukan pemodelan Sequential
- Jika ini adalah layer pertama, kita perlu memasukkan bentuk input.
- Kalau tidak kita hanya akan menambahkan layer.
- Kemudian, setelah menambahkan layer konvolusi, kita akan melakukan hal yang sama dengan max pooling.
- Lalu, kita akan meratakan atau flatten dan menambahkan dense size ukuran apa pun yang berasal dari dense_size. Dimana akan selalu menggunakan algoritma tanh
- Jika dropout digunakan, kita akan menambahkan layer dropout. Menyebut dropout ini berarti, katakanlah 50%, bahwa setiap kali ia memperbarui bobot setelah setiap batch, ada peluang 50% untuk setiap bobot yang tidak akan diperbarui
- menempatkan ini di antara dua lapisan padat untuk dihindarkan dari melindunginya dari overfitting.
- Lapisan terakhir akan selalu menjadi jumlah kelas karena itu harus, dan menggunakan softmax. Itu dikompilasi dengan cara yang sama.
- Atur direktori log yang berbeda untuk TensorBoard sehingga dapat memberikan konfigurasi yang berbeda.
- Variabel start akan memanggil modul time atau waktu
- Melakukan fit atau compile
- Melakukan scoring dengan .evaluate yang akan menampilkan data loss dan accuracy dari model
- end merupakan variabel untuk melihat waktu akhir pada saat pemodelan berhasil dilakukan.
- Menampilkan hasil dari run skrip diatas
- Hasilnya sebagai berikut :

```

    ...
    2f, Accuracy: %.2f, Time: % sec" % (conv2d_count, dense_size, dropout, score[0],
score[1], elapsed))
    ...
    results.append((conv2d_count, dense_size, dropout, score[0],
score[1], elapsed))
Conv2D count: 1, Dense size: 128, Dropout: 0.00 - Loss: 1.13, Accuracy: 0.74, Time: 1572
sec
Conv2D count: 1, Dense size: 128, Dropout: 0.25 - Loss: 0.91, Accuracy: 0.76, Time: 1632
sec
Conv2D count: 1, Dense size: 128, Dropout: 0.50 - Loss: 0.80, Accuracy: 0.78, Time: 1662
sec
Conv2D count: 1, Dense size: 128, Dropout: 0.75 - Loss: 0.80, Accuracy: 0.78, Time: 1735
sec
Conv2D count: 1, Dense size: 256, Dropout: 0.00 - Loss: 1.28, Accuracy: 0.74, Time: 2153
sec
Conv2D count: 1, Dense size: 256, Dropout: 0.25 - Loss: 1.08, Accuracy: 0.76, Time: 2212
sec
Conv2D count: 1, Dense size: 256, Dropout: 0.50 - Loss: 0.91, Accuracy: 0.78, Time: 2184
sec
Conv2D count: 1, Dense size: 256, Dropout: 0.75 - Loss: 0.78, Accuracy: 0.78, Time: 2184
sec

```

Gambar 7.25 Kode Program Blok In 13 Tasya

```

1 model.add(Dense(128, activation='tanh'))
2 model.add(Dropout(0.5))
3 model.add(Dense(num_classes, activation='softmax'))
4 model.compile(loss='categorical_crossentropy', optimizer='adam',
     metrics=['accuracy'])
5 print(model.summary())

```

Keterangannya sebagai berikut :

- Melakukan pemodelan Sequential
- Untuk layer pertama, Menambahkan Convolutio 2D dengan dmensi 32, dan ukuran matriks 3x3 dengan function aktivasi yang digunakan yaitu relu dan menampilkan input_shape
- Dilakukan Max Pooling 2D dengan ukuran matriks 2x2
- Untuk layer kedua, melakukan Convolusi lagi dengan kriteria yang sama tanpa menambahkan input, ini dilakukan untuk mendapatkan data yang terbaik
- Flatten digunakan ntuk meratakan inputan
- Menambahkan dense input sebanyak 128 neuron dengan menggunakan function aktivasi tanh.
- Dropout sebanyak 50% untuk menghindari overfitting
- Menambahkan dense pada model untuk output dimana layer ini akan menjadi jumlah dari class yang ada.
- Mengcompile model yang didefinisikan diatas
- Menampilkan ringkasan dari pemodelan yang dilakukan
- Gambarnya seperti berikut :

```
.... print(model.summary())
=====
Layer (type)                 Output Shape              Param #
conv2d_12 (Conv2D)           (None, 30, 30, 32)      896
max_pooling2d_12 (MaxPooling) (None, 15, 15, 32)      0
conv2d_13 (Conv2D)           (None, 13, 13, 32)      9248
max_pooling2d_13 (MaxPooling) (None, 6, 6, 32)       0
flatten_11 (Flatten)         (None, 1152)             0
dense_21 (Dense)            (None, 128)              147584
dropout_8 (Dropout)          (None, 128)              0
dense_22 (Dense)            (None, 369)              47601
=====
Total params: 205,329
Trainable params: 205,329
Non-trainable params: 0
=====
None
```

Gambar 7.26 Kode Program Blok In 14 Tasya

Keterangannya sebagai berikut :

- Melakukan fit dengan join data train dan test agar dapat dilakukan pelatihan untuk jaringan pada semua data yang dimiliki.
- Hasilnya sebagai berikut :

```
In [15]: model.fit(np.concatenate((train_input, test_input)),
...                  np.concatenate((train_output, test_output)),
...                  batch_size=32, epochs=10, verbose=2)
Epoch 1/10
- 267s - loss: 1.7763 - acc: 0.5881
Epoch 2/10
- 249s - loss: 1.0767 - acc: 0.7061
Epoch 3/10
- 248s - loss: 0.9666 - acc: 0.7297
Epoch 4/10
- 249s - loss: 0.9096 - acc: 0.7411
Epoch 5/10
- 248s - loss: 0.8699 - acc: 0.7518
Epoch 6/10
- 253s - loss: 0.8428 - acc: 0.7551
Epoch 7/10
- 250s - loss: 0.8211 - acc: 0.7608
Epoch 8/10
- 247s - loss: 0.8050 - acc: 0.7639
Epoch 9/10
- 252s - loss: 0.7880 - acc: 0.7679
Epoch 10/10
- 251s - loss: 0.7751 - acc: 0.7697
Out[15]: <keras.callbacks.History at 0x22f13d79a20>
```

Gambar 7.27 Kode Program Blok In 15 Tasya

Keterangannya sebagai berikut :

- Menyimpan atau save model yang telah di latih dengan nama mathsymbols.model
- Hasilnya seperti berikut :

```
In [16]: model.save("mathsymbols.model")
└── mathsymbols.model 13/04/2019 03:45 MODEL File 2,443 KB
```

Gambar 7.28 Kode Program Blok In 16 Tasya

Keterangannya sebagai berikut :

- Simpan label enkoder (untuk membalikkan one-hot encoder) dengan nama classes.npy
- Hasilnya seperti berikut :

```
In [17]: np.save('classes.npy', label_encoder.classes_)
└── classes.npy 13/04/2019 03:45 NPY File 28 KB
```

Gambar 7.29 Kode Program Blok In 17 Tasya

Keterangannya sebagai berikut :

- Impor models dari librari Keras
- Variabel model2 akan memanggil model yang telah disave tadi
- Menampilkan ringkasan dari hasil pemodelan
- Hasilnya sebagai berikut :

```
.... model = load_model('mathsymbols.model')
.... print(model2.summary())
Layer (type)          Output Shape         Param #
conv2d_12 (Conv2D)    (None, 30, 30, 32)      896
max_pooling2d_12 (MaxPooling) (None, 15, 15, 32)   0
conv2d_13 (Conv2D)    (None, 13, 13, 32)      9248
max_pooling2d_13 (MaxPooling) (None, 6, 6, 32)   0
flatten_11 (Flatten)  (None, 1152)           0
dense_21 (Dense)     (None, 128)            147584
dropout_8 (Dropout)  (None, 128)            0
dense_22 (Dense)     (None, 369)            47601
Total params: 205,329
Trainable params: 205,329
Non-trainable params: 0
None
```

Gambar 7.30 Kode Program Blok In 18 Tasya

```

1 # do the prediction
2 prediction = model2.predict(newimg.reshape(1, 32, 32, 3))
3
4
5 # figure out which output neuron had the highest score, and
6 # reverse the one-hot encoding
7 inverted = label_encoder2.inverse_transform([np.argmax(prediction)])
# argmax finds highest-scoring output
8 print("Prediction: %s, confidence: %.2f" % (inverted[0], np.max(
9 prediction)))

```

Keterangannya sebagai berikut :

- Memanggil fungsi LabelEncoder
- Variabel label_encoder akan memanggil class yang disave sebelumnya.
- Function Predict akan mengubah gambar kedalam bentuk array
- Variabel prediction akan melakukan prediksi untuk model2 dengan reshape variabel newimg dengan bentukarray 4D.
- Variabel inverted akan mencari nilai tertinggi output dari hasil prediksi tadi
- Menampilkan hasil dari variabel prediction dan inverted
- Hasilnya sebagai berikut :

```

In [19]: label_encoder2 = LabelEncoder()
...: label_encoder2.classes_ = np.load('classes.npy')
...:
...: def predict(img_path):
...:     newimg = keras.preprocessing.image.img_to_array(pil_image.open(img_path))
...:     newimg /= 255.0
...:
...:     # do the prediction
...:     prediction = model2.predict(newimg.reshape(1, 32, 32, 3))
...:
...:     # figure out which output neuron had the highest score, and reverse the
one-hot encoding
...:     inverted = label_encoder2.inverse_transform([np.argmax(prediction)]) #
argmax finds highest-scoring output
...:     print("Prediction: %s, confidence: %.2f" % (inverted[0],
np.max(prediction)))

```

Gambar 7.31 Kode Program Blok In 19 Tasya

```

1 # do the prediction
2 prediction = model2.predict(newimg.reshape(1, 32, 32, 3))
3
4
5 # figure out which output neuron had the highest score, and
6 # reverse the one-hot encoding
7 inverted = label_encoder2.inverse_transform([np.argmax(prediction)])
# argmax finds highest-scoring output
8 print("Prediction: %s, confidence: %.2f" % (inverted[0], np.max(
9 prediction)))

```

Keterangannya sebagai berikut :

- Melakukan prediksi dari pelatihan dari gambar v2-00010.png
- Melakukan prediksi dari pelatihan dari gambar v2-00500.png
- Melakukan prediksi dari pelatihan dari gambar v2-00700.png
- Hasilnya sebagai berikut :

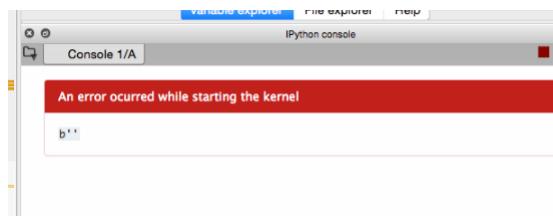
```
In [20]: predict("HASYv2/hasy-data/v2-00010.png")
...:
...: predict("HASYv2/hasy-data/v2-00500.png")
...:
...: predict("HASYv2/hasy-data/v2-00700.png")
Prediction: A, confidence: 0.85
Prediction: \pi, confidence: 0.81
Prediction: \alpha, confidence: 0.90
```

Gambar 7.32 Kode Program Blok In 20 Tasya

7.1.8 Penanganan Error

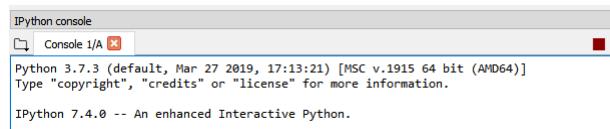
7.1.8.1 Error Starting Kernel

- Berikut merupakan screenshot error



Gambar 7.33 Error Tasya

- Eror tersebut merupakan eror yang terjadi dan membuat kita tidak dapat mengakses dan menggunakan kernel atau konsol pada spyder.
- Untuk penanganannya sebagai berikut :
 1. Tutup spyder yang sedang dijalankan
 2. Kemudian buka kembali spyder
 3. Atau jika tidak berhasil, buka anaconda promt dan ketikan "conda update spyder"
 4. Jika tidak berhasil juga bisa menginstall ulang anaconda
 5. Maka ketika diajalankan lagi hasilnya seperti berikut :



The screenshot shows a window titled "IPython console" with the tab "Console 1/A" selected. The window displays the following text:
Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.4.0 -- An enhanced Interactive Python.

Gambar 7.34 Penanganan Error Kernel Tasya

DAFTAR PUSTAKA
