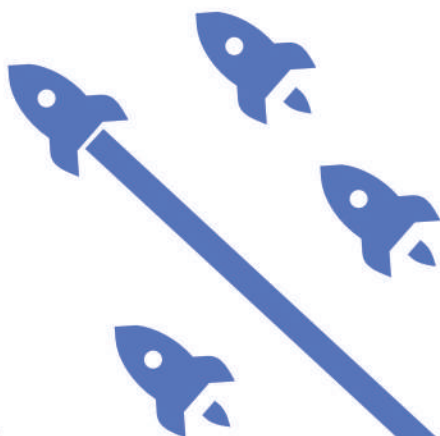


Novian Adi Prasetyo | Muhamad Azrino Gustalika  
Muhammad Lulu Latif Usman | Fahrudin Mukti Wibowo  
Muhammad Rizqan Fadhil



# Android Kotlin

## Untuk Pemula



# Android Kotlin Untuk Pemula

Buku ini berisi pengembangan perangkat lunak berbasis mobile Android dengan menggunakan Bahasa Pemrograman Kotlin dan Software Android Studio. Kotlin adalah bahasa pemrograman yang dikembangkan oleh JetBrains, perusahaan perangkat lunak yang terkenal dengan produk-produk seperti IntelliJ IDEA (sebuah lingkungan pengembangan terintegrasi, atau IDE) dan Kotlin sendiri. Kotlin dirancang untuk menjadi bahasa yang ekspresif, ringkas, dan mudah dipahami, sambil tetap kompatibel dengan Java, bahasa pemrograman yang telah lama digunakan dalam pengembangan aplikasi Android.

Buku ini mengawali pembaca dengan pengenalan terhadap Kotlin, bahasa pemrograman yang telah menjadi pilihan populer bagi pengembang Android. Selanjutnya, pembaca akan diajak untuk memahami dasar-dasar pengembangan Android, termasuk konsep seperti Activity, Layouts, dan Views. Buku ini tidak hanya memberikan teori, tetapi juga memberikan contoh-contoh kode yang mudah diikuti, serta latihan-latihan praktis untuk memastikan pemahaman konsep.

Pada Buku ini memiliki 12 Bab dalam pembuatan Aplikasi Android dengan Bahasa Pemrograman Kotlin. Pembahasan disajikan dimulai dari pengenalan dasar tentang kotlin hingga pembuatan CRUD dengan menggunakan REST API. Buku ini akan membahas secara lengkap dan secara jelas bagaimana memahami Bahasa Pemrograman Kotlin dalam pengembangan Aplikasi Android



0858 5343 1992  
eurekamediaaksara@gmail.com  
Jl. Banjaran RT.20 RW.10  
Bojongsari - Purbalingga 53362



# ANDROID KOTLIN UNTUK PEMULA

Novian Adi Prasetyo  
Muhamad Azrino Gustalika  
Muhammad Lulu Latif Usman  
Fahrudin Mukti Wibowo  
Muhammad Rizqan Fadhil



**PENERBIT CV.EUREKA MEDIA AKSARA**

## ANDROID KOTLIN UNTUK PEMULA

**Penulis** : Novian Adi Prasetyo  
Muhamad Azrino Gustalika  
Muhammad Lulu Latif Usman  
Fahrudin Mukti Wibowo  
Muhammad Rizqan Fadhil

**Desain Sampul** : Ardyan Arya Hayuwaskita

**Tata Letak** : Herlina Sukma

**ISBN** : 978-623-151-891-0

Diterbitkan oleh : **EUREKA MEDIA AKSARA, NOVEMBER 2023**  
**ANGGOTA IKAPI JAWA TENGAH**  
**NO. 225/JTE/2021**

**Redaksi:**

Jalan Banjaran, Desa Banjaran RT 20 RW 10 Kecamatan Bojongsari  
Kabupaten Purbalingga Telp. 0858-5343-1992

Surel : eurekamediaaksara@gmail.com

Cetakan Pertama : 2023

**All right reserved**

Hak Cipta dilindungi undang-undang  
Dilarang memperbanyak atau memindahkan sebagian atau seluruh  
isi buku ini dalam bentuk apapun dan dengan cara apapun,  
termasuk memfotokopi, merekam, atau dengan teknik perekaman  
lainnya tanpa seizin tertulis dari penerbit.

## KATA PENGANTAR

Alhamdulillah, puji dan syukur kehadiran Allah SWT karena buku ini telah selesai disusun. Buku ini berisi penjelasan pengenalan Bahasa pemrograman perangkat bergerak dengan menggunakan kotlin yang dimulai dari instalasi kotlin di Bahasa pemrograman android sampai fungsi-fungsi yang digunakan dalam pemrograman android dengan menggunakan kotlin. Ditambah lagi terdapat beberapa contoh-contoh implementasi kotlin dalam beberapa studi kasus. Dengan adanya buku ini diharapkan dapat membantu para mahasiswa dalam mempelajari bahasa pemrograman kotlin beserta contoh-contoh implementasi kotlin dalam pemrograman perangkat bergerak yang bermanfaat terutama bagi orang awam yang belum mengenal bahasa pemrograman kotlin itu sendiri.

Penulis juga menyadari jika didalam penyusunan buku ini masih terdapat kekurangan, namun penulis meyakini sepenuhnya bahwa sekecil apapun buku ini tetap akan memberikan sebuah manfaat bagi pembaca. Akhir kata untuk penyempurnaan buku ini, maka kritik dan saran dari pembaca sangatlah berguna untuk penulis kedepannya.

Purwokerto, September 2023

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>iv</b>
<b>DAFTAR GAMBAR.....</b>	<b>vi</b>
<b>BAB 1 KOTLIN FUNDAMENTAL.....</b>	<b>1</b>
A. Arti dan Sejarah Android .....	1
B. Android Studio .....	4
C. Bahasa Pemrograman Kotlin.....	5
D. Android Software Development Kit (Sdk).....	8
E. Java .....	8
<b>BAB 2 INSTALASI ANDROID STUDIO.....</b>	<b>10</b>
A. Instalasi Android Studio.....	10
B. Langkah Membuat Project Baru.....	15
C. Setting Android Studio di Smartphone.....	18
<b>BAB 3 ACTIVITY LIFECYCLE.....</b>	<b>19</b>
A. Teori Activity .....	19
B. Langkah membuat Activity.....	20
<b>BAB 4 INTENT DAN FRAGMENT .....</b>	<b>34</b>
A. Teori Intent dan Fragment.....	34
B. Langkah Membuat Intent Explicit Dan Fragment.....	36
C. Langkah Membuat Intent Implicit.....	44
D. Mengirim Data Antar Fragment .....	50
<b>BAB 5 RECYCLERVIEW.....</b>	<b>64</b>
A. Teori RecyclerView .....	64
B. Langkah-langkah Membuat List View .....	66
C. Langkah Membuat Grid View .....	78
D. Langkah Membuat Card View.....	83
E. Langkah Membuat Icon Launcher .....	90
<b>BAB 6 SCROLLING DAN GOOGLE MAP ACTIVITY .....</b>	<b>92</b>
A. Teori Scrolling Activity .....	92
B. Teori Maps Activity.....	92
C. Langkah Membuat Scrolling Activity.....	93
D. Langkah Membuat Google Map Activity.....	99
<b>BAB 7 BROADCAST RECEIVER.....</b>	<b>108</b>
A. Teori Pendukung.....	108
B. Langkah Membuat Broadcast Receiver.....	110

<b>BAB 8 SHARED PREFERENCES .....</b>	<b>123</b>
A. Teori Pendukung .....	123
B. Langkah Membuat Shared Preferences .....	126
<b>BAB 9 CRUD SQLITE.....</b>	<b>149</b>
A. Teori Pendukung .....	149
B. Langkah-langkah latihan CRUD SQLITE .....	151
<b>BAB 10 AUTH FIREBASE.....</b>	<b>189</b>
A. Teori Pendukung .....	189
B. Langkah-langkah latihan Auth Firebase.....	190
<b>BAB 11 CRUD FIRESTORE.....</b>	<b>229</b>
A. Teori Pendukung .....	229
B. Langkah-langkah latihan CRUD Firestore .....	230
C. Langkah-langkah latihan Auth Phone Number.....	269
<b>BAB 12 CRUD REST API.....</b>	<b>284</b>
A. Teori Pendukung .....	284
B. Langkah-langkah latihan CRUD Rest API .....	285
<b>DAFTAR PUSTAKA .....</b>	<b>351</b>
<b>TENTANG PENULIS .....</b>	<b>353</b>

## DAFTAR GAMBAR

Gambar 1. 1 Arsitektur Android .....	2
Gambar 1. 2 Clean Architecture menggunakan KMM.....	7
Gambar 2. 1 Tampilan Awal Android Studio .....	11
Gambar 2. 2 Configure SDK Manager .....	11
Gambar 2. 3 SDK Manager .....	12
Gambar 2. 4 Setting Android SDK.....	12
Gambar 2. 5 Download Android SDK.....	13
Gambar 2. 6 Install Android SDK .....	13
Gambar 2. 7 Setting SDK Manager pada Tools.....	14
Gambar 2. 8 Penambahan Project Baru .....	15
Gambar 2. 9 Setting Activity.....	15
Gambar 2. 10 Setting Activity Property.....	16
Gambar 2. 11 Setting Bahasa Pemrograman .....	16
Gambar 2. 12 Tampilan Project .....	17
Gambar 2. 13 Setting Device Debuging.....	18
Gambar 3. 1 Hasil running Aplikasi .....	23
Gambar 3. 2 Pembuatan Modul Baru .....	23
Gambar 3. 3 Setting Device Debuging.....	24
Gambar 3. 4 Setting View Activity.....	24
Gambar 3. 5 Hasil running Aplikasi .....	27
Gambar 3. 6 Hasil Debuging Aplikasi dengan Logcat .....	27
Gambar 3. 7 Membuat Activity Baru .....	28
Gambar 3. 8 Setting Activity Baru.....	28
Gambar 3. 9 Hasil running Aplikasi .....	32
Gambar 3. 10 Hasil Debugging Aplikasi Menggunakan Logcat....	33
Gambar 4. 1 Alur Intent .....	36
Gambar 4. 2 Hasil running Aplikasi Intent 1 .....	43
Gambar 4. 3 Hasil running Aplikasi Intent 2 .....	43
Gambar 4. 4 Hasil running Aplikasi .....	50
Gambar 4. 5 Menambahkan Fragment Baru .....	53
Gambar 4. 6 Setting Fragment Baru.....	53
Gambar 4. 7 Hasil running Aplikasi menggunakan Frament .....	56
Gambar 4. 8 Hasil running Aplikasi menggunakan Fragment .....	63
Gambar 5. 1 Membuat Package Baru.....	68



Gambar 5. 2 Pembuatan Layout Baru .....	69
Gambar 5. 3 Menambahkan File Kotlin Baru .....	71
Gambar 5. 4 Hasil Running Aplikasi.....	77
Gambar 5. 5 Hasil running Aplikasi Tampilan Menu .....	82
Gambar 5. 6 Hasil running Aplikasi.....	82
Gambar 5. 7 Hasil Aplikasi .....	89
Gambar 5. 8 Menambahkan Image Asset .....	90
Gambar 5. 9 Setting Image Asset.....	90
Gambar 5. 10 Penggunaan Icon .....	91
Gambar 6.1 Hasil Aplikasi .....	98
Gambar 6.2 Tambah Icon - New.....	104
Gambar 6.3 Tambah Icon - pilih Asset.....	104
Gambar 6.4 Tambah Icon - Select Icon .....	105
Gambar 6.5 Hasil Aplikasi .....	107
Gambar 7. 1 Tambah Icon - New .....	110
Gambar 7. 2 Langkah Pembuatan Broadcast Reciever - 1 .....	116
Gambar 7. 3 Langkah Pembuatan Broadcast Reciever -2.....	116
Gambar 7. 4 Hasil Running Aplikasi.....	122
Gambar 8. 1 Hasil Running Aplikasi.....	133
Gambar 8. 2 Hasil Generate Fungsi.....	142
Gambar 8. 3 Hasil Running Aplikasi.....	148
Gambar 8. 4 Hasil Running Aplikasi 2.....	148
Gambar 8. 5 Hasil Running Aplikasi 3.....	148
Gambar 9. 1 New Vector Asset.....	153
Gambar 9. 2 Configure Vector Asset Plus.....	154
Gambar 9. 3 Configure Vector Asset Delete .....	154
Gambar 9. 4 Menu Form.....	179
Gambar 9. 5 Member Interface.....	180
Gambar 9. 6 Hasil Running.....	188
Gambar 10. 1 Tambahkan Project.....	190
Gambar 10. 2 Pembuatan Project.....	191
Gambar 10. 3 Hasil Pembuatan Project.....	191
Gambar 10. 4 Authentication Firebase .....	191
Gambar 10. 5 Authentication Email .....	192
Gambar 10. 6 Authentication pada Android Studio .....	192
Gambar 10. 7 Connect Firebase.....	193

Gambar 10. 8 Connect Android Studio dan Firebase pada Browser.....	194
Gambar 10. 9 Hasil Connect Firebase .....	194
Gambar 10. 10 Add Firebase Authentication .....	194
Gambar 10. 11 Add Vector Asset .....	195
Gambar 10. 12 Add Email Icon.....	196
Gambar 10. 13 Add Phone Icon.....	196
Gambar 10. 14 UI Login .....	201
Gambar 10. 15 Proses Login .....	211
Gambar 10. 16 Hasil Running Setelah Login .....	213
Gambar 10. 17 Hasil Running Aplikasi Create Account .....	219
Gambar 10. 18 Sign-In Method Google .....	224
Gambar 10. 19 SHA-1 dan SHA-256 .....	224
Gambar 10. 20 Gradle Sign-In Report.....	225
Gambar 10. 21 Copy SHA-1 dan SHA-256.....	225
Gambar 10. 22 Setelan Project Firebase .....	225
Gambar 10. 23 Download Gradle google-service.json .....	226
Gambar 10. 24 Directory Project.....	227
Gambar 10. 25 Directory Project.....	227
Gambar 11. 1 Skema Cloud Firestore.....	230
Gambar 11. 2 Pembuatan Cloud Firestore .....	231
Gambar 11. 3 Pembuatan Koleksi .....	232
Gambar 11. 4 Penambahan Dokumen Baru .....	232
Gambar 11. 5 Pembuatan Icon Vector Asset .....	235
Gambar 11. 6 Running Aplikasi .....	256
Gambar 11. 7 Hasil running Kategori.....	259
Gambar 11. 8 Hasil Running Menggunakan Create Cloud Fire...	262
Gambar 11. 9 Hasil Running Read Cloud Fire.....	265
Gambar 11. 10 Hasil Running Update Cloud Fire.....	267
Gambar 11. 11 Hasil Running Delete Cloud Fire.....	269
Gambar 11. 12 Firebase Authentication.....	281
Gambar 11. 13 Firebase Cek Project.....	281
Gambar 11. 14 Firebase Dashboard.....	282
Gambar 11. 15 Android Device Verification .....	282
Gambar 11. 16 Running Aplikasi Login Menggunakan Nomor HP.....	283

Gambar 12. 1 Konsep Rest API.....	284
Gambar 12. 2 Uji Coba menggunakan Method POST .....	286
Gambar 12. 3 Register Menggunakan Postman .....	287
Gambar 12. 4 Login Menggunakan Postman .....	287
Gambar 12. 5 Insert Menggunakan Postman.....	288
Gambar 12. 6 Menggunakan Method GET Postman .....	288
Gambar 12. 7 Menggunakan Method PUT Postman .....	288
Gambar 12. 8 Menggunakan Method DELETE Postman .....	289
Gambar 12. 9 Menambahkan Icon.....	291
Gambar 12. 10 Daftar Icon yang di Tambahkan.....	292
Gambar 12. 11 Refactoring .....	293
Gambar 12. 12 Running Aplikasi Fragment .....	300
Gambar 12. 13 Running Aplikasi.....	339
Gambar 12. 14 Running Aplikasi.....	348
Gambar 12. 15 Running Aplikasi.....	349
Gambar 12. 16 Running Aplikasi.....	350



## **ANDROID KOTLIN UNTUK PEMULA**

Novian Adi Prasetyo  
Muhamad Azrino Gustalika  
Muhammad Lulu Latif Usman  
Fahrudin Mukti Wibowo  
Muhammad Rizqan Fadhil



# BAB

# 1

# KOTLIN FUNDAMENTAL

## A. Arti dan Sejarah Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, middleware dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk membuat aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat software untuk ponsel/smartphone. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan hardware, software, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Arsitektur Android terdiri dari Applications, Application Framework, Libraries, Android Runtime dan Kernel Arsitektur lengkap Linux dapat dilihat pada Gambar 1.1 [1].

# BAB 2

## INSTALASI ANDROID STUDIO

### A. Instalasi Android Studio

Pada pemrograman perangkat bergerak dengan menggunakan Bahasa pemrograman perangkat bergerak menggunakan bahasa pemrograman kotlin memerlukan software yang digunakan untuk membuat aplikasi android yaitu Android Studio[4], ada beberapa tahapan yang perlu dilakukan agar bisa menginstal android studio

1. Buka alamat untuk mengunduh software android studio pada link berikut ini <https://developer.android.com/studio> maka akan muncul tampilan seperti ini
2. Kemudian instal Android Studio ke perangkat laptop Anda dan Ikuti petunjuknya sampai selesai. Saat instalasi berlangsung Anda diminta untuk menginstal SDK (Software Development Kit) yang membutuhkan koneksi internet. Maka pastikan saat menginstal Anda terhubung dengan internet.
3. Setelah selesai melakukan instalasi Android Studio, akan muncul seperti gambar 2.1. Artinya Aplikasi Android Studio sudah bisa digunakan

# BAB

# 3

## ACTIVITY LIFECYCLE

### A. Teori Activity

Saat pengguna menelusuri, keluar, dan kembali ke aplikasi Anda, instance Activity dalam aplikasi Anda melakukan transisi ke berbagai status dalam siklus prosesnya. Class Activity menyediakan sejumlah callback yang memungkinkan aktivitas mengetahui bahwa status telah berubah: bahwa sistem membuat, menghentikan, atau melanjutkan suatu aktivitas, atau menutup proses tempat beradanya aktivitas.

Dalam metode callback siklus proses, Anda dapat mendeklarasikan cara aktivitas berperilaku saat pengguna meninggalkan dan memasuki kembali aktivitas itu. Misalnya, jika sedang mem-build pemutar video streaming, Anda dapat menghentikan sementara video itu dan mengakhiri koneksi jaringan saat pengguna beralih ke aplikasi lain. Saat pengguna kembali ke aplikasi, Anda dapat menghubungkan ulang ke jaringan dan memungkinkan pengguna melanjutkan video tersebut dari titik yang sama. Dengan kata lain, setiap callback memungkinkan Anda melakukan pekerjaan tertentu yang sesuai dengan perubahan status yang diberikan. Melakukan pekerjaan yang tepat dan pada waktu yang tepat, serta menangani transisi dengan benar membuat aplikasi Anda lebih andal dan efektif [5].

# BAB 4

## INTENT DAN FRAGMENT

### A. Teori Intent dan Fragment

Intent adalah sebuah kelas dalam programming Android yang berfungsi untuk perpindahan halaman. suatu objek yang terdapat dalam suatu activity dimana objek tersebut dapat komunikasi dengan activity yang lain, baik activity pada fungsi internal android misal seperti memanggil activity dalam satu package atau beda package yang masih berada dalam satu project. merupakan objek tipe `android.content.Intent`. Melalui metode `startActivity()` yang digunakan untuk memulai sebuah activity lain. sebuah mekanisme yang digunakan untuk melakukan sebuah aksi dari komponen aplikasi. Untuk dapat melakukan sebuah aksi pada sebuah intent [6], ada 3 cara yang dapat dilakukan:

1. Menjalankan sebuah activity lain baik dengan data ataupun tanpa data
2. Membuat sebuah service untuk menjalankan pekerjaan tertentu pada sebuah background/non main thread.
3. Mengirimkan sebuah broadcast. Pesan yang dikirimkan dalam keadaan tertentu, misalkan ketika booting atau sedang melakukan pengisian data baru mengirimkan data

Ada 2 macam model intent dalam pemrograman android

#### 1. Explicit Intent

Dikatakan intent explicit karena intent tersebut biasanya dibuat oleh seorang programmer berdasarkan kebutuhan yang ada. Untuk memanggil intent tersebut



# BAB

# 5

# RECYCLERVIEW

## A. Teori RecyclerView

Pada bab ini kita akan membahas tentang kegunaan dari macam-macam RecyclerView yang terdiri dari listview, gridview, dan cardview.

### 1. List View

List View merupakan sebuah widget yang terdapat pada komponen UI (user interface) dari sebuah aplikasi. Widget yang dipakai untuk menampilkan nilai-nilai dan data dalam bentuk list atau daftar. Item-item data yang ditampilkan dalam ListView biasanya dapat tersusun dalam urutan berjajar ke samping atau ke bawah. tampilan ListView selalu dapat dengan mudah di-scroll oleh user aplikasi. Data-data pada ListView didapat dari sebuah query database atau array yang telah ditetapkan sebagai sebuah data model. Untuk bisa bekerja dengan baik, ListView punya sebuah adapter. Adapter adalah sebuah class yang berfungsi untuk mengatur resource view, model dan susunan item di ListView. Resource view mengatur item yang terlihat dan tersembunyi di ListView dan adapter hampir selalu mempunyai jumlah resource view yang tetap, tidak berubah, sesuai dengan item-item yang muncul di layar aplikasi.

ListView masih cukup banyak digunakan oleh developer aplikasi, terutama jika aplikasi yang dibuat masih sederhana. Para app developer pemula juga seringkali menjadikan tugas membuat ListView Android Studio

# BAB

# 6

## SCROLLING DAN GOOGLE MAP ACTIVITY

### A. Teori Scrolling Activity

Di Android proses scroll biasanya dicapai menggunakan class `ScrollView`. Setiap tata letak standar yang mungkin melampaui batas penampungnya harus disarankan di `ScrollView` untuk memberikan tampilan scroll yang dikelola oleh framework. Penerapan scroller kustom hanya diperlukan untuk skenario khusus. Scroller digunakan untuk menganimasikan scrolling dari waktu ke waktu, yang menggunakan prinsip fisika scrolling standar platform. Scroller itu sendiri sebenarnya tidak menggambar apa pun. Scroller melacak offset scroll untuk Anda dari waktu ke waktu, tetapi tidak otomatis menerapkan posisi tersebut pada tampilan Anda. `ScrollView` digunakan agar konten pada layar aplikasi dapat di Scroll secara vertikal, karena jika tidak menggunakan `ScrollView`, jika konten kita melebihi ukuran layar maka konten tersebut akan terpotong [8].

### B. Teori Maps Activity

Maps API dari Google play service adalah salah satu API yang dapat dimanfaatkan dalam mengembangkan aplikasi android yang memberikan fitur berbasis lokasi. Seperti yang kita ketahui saat ini Google bukan lagi hanya sebuah Mesin Pencari tapi telah berubah menjadi perusahaan teknologi raksasa dengan berbagai macam produk inovatifnya. Salah satunya adalah Google Maps. Kita sebagai developer tidak hanya dapat

# BAB

# 7

## BROADCAST RECEIVER

### A. Teori Pendukung

Broadcast receivers adalah komponen-komponen aplikasi Android yang memungkinkan kita mengirim atau menerima pesan-pesan. Broadcast receivers mendaftarkan sistem broadcast dan/atau custom broadcast serta mendapat notifikasi melalui intent dengan cara-cara berikut:

1. Oleh sistem, saat terjadi system event pada suatu aplikasi yang terdaftar.
2. Oleh aplikasi lain, jika suatu aplikasi terdaftar untuk custom event.

Broadcast receivers digunakan untuk menerima pesan jika ada perubahan dari suatu sistem. Misalnya saat ponsel dihubungkan dengan charger, sistem akan mendeteksi dan mengirim pesan, kemudian ponsel/aplikasi menerima pesan tersebut yang ditandai dengan tampilan ponsel sedang charging. sebuah broadcast sesungguhnya adalah pesan yang dibungkus menggunakan objek intent [10]. Broadcast bisa dibuat secara implisit atau implisit. Berikut adalah perbedaannya:

1. Sebuah implicit broadcast adalah pesan yang tidak hanya ditujukan pada aplikasi kita saja, tetapi siapapun. Untuk menerima pesan implicit ini, kita perlu menggunakan sebuah IntentFilter dan mendeklarasikannya dalam manifest. Kita perlu melakukan langkah ini karena sistem operasi Android akan memberikan pesan broadcast pada setiap aplikasi yang sudah memiliki Intent Filter untuk event Karena cara

# BAB

# 8

## SHARED PREFERENCES

### A. Teori Pendukung

SharedPreferences adalah sebuah class yang memiliki kemampuan ataupun fungsi untuk menampung data dalam key/value yang berpasangan. Data dalam SharedPreferences nantinya dapat digunakan dalam aktivitas apa saja dalam android studio [11]. Tipe data dan Metode dalam SharedPreferences dibagi menjadi beberapa tipe data dan metode, yaitu [12]:

1. getBoolean()
2. getFloat()
3. getInt()
4. getLong()
5. getString()
6. getStringSet()

Pada penyimpanan menggunakan SharedPreferences, Android menggunakan preferences APIs untuk menyediakan jalan yang baik bagi pengguna dalam menggunakan ataupun mengubah pengaturan mereka. Sebagai contoh, user interface untuk pengaturan dapat menggunakan PreferenceActivity ataupun PreferenceFragment, lalu jika pengguna mengubah pengaturan maka data yang terdapat didalam SharedPreferences akan berubah ataupun terupdate.

Shared Preferences adalah cara mudah untuk menyimpan data yang tidak terlalu sensitif pada Android. Data yang disimpan dalam Shared Preferences dapat diakses oleh aplikasi

# BAB

# 9

## CRUD SQLITE

### A. Teori Pendukung

Android menyediakan salah satu mekanisme penyimpanan data berulang yang dapat tersimpan secara terstruktur dan berelasi dengan menggunakan SQLite. SQLite merupakan database yang bersifat open source yang mendukung operasi relasi standar yang umum terdapat pada engine database seperti sintaks SQL dan operasi transaksi. Meskipun berjalan seperti database, pada umumnya sqlite berukuran kecil dan mampu berjalan pada peranti dengan memori terbatas seperti smartphone [13].

SQLite ini dipakai oleh aplikasi-aplikasi native dengan penyimpanan data yang tidak bersifat kompleks seperti Google Chrome dan Firefox. Contohnya adalah untuk menyimpan data bookmark website yang dilakukan oleh pengguna dan juga aplikasi mobile seperti aplikasi contact bawaan OS Android yang menggunakannya sebagai penyimpanan data lokal yang mendukung proses sinkronisasi ke server.

SQLite hanya mendukung beberapa tipe data seperti text untuk penyimpanan data dalam bentuk string, int untuk menyimpan data dalam bentuk bilangan bulat, real untuk menyimpan data dalam bentuk bilangan pecahan/bilangan presisi. Jadi apabila ingin menyimpan data yang tidak didukung oleh SQLite maka diharuskan dilakukan proses konversi tipe data yang sesuai dengan tipe data yang didukung sebelum melakukan penyimpanan data.

# BAB

# 10

## AUTH FIREBASE

### A. Teori Pendukung

Firestore awalnya dibuat oleh perusahaan startup bernama Evolve buatan James Tamplin dan Andrew Lee pada tahun 2011. Awalnya Firestore dirancang sebagai API chat service yang memungkinkan untuk membuat fungsionalitas online chat pada sebuah website. Namun, seiring berjalannya waktu, banyak developer yang memanfaatkan Firestore untuk menyinkronkan data dalam aplikasi, seperti menyimpan status permainan yang dilakukan secara real time di seluruh penggunaannya (real time multiplayer games). Akhirnya di tahun 2012, Tamplin dan Lee memisahkan produk Firestore menjadi dua, yakni sebagai chat system dan real-time architecture. Setelah tiga tahun diluncurkan, semakin banyak developer yang menggunakan Firestore, kurang lebih 110.000 developer sudah terdaftar. Tim Firestore juga mengatakan bahwa banyaknya developer yang terdaftar tidak akan memengaruhi service Firestore dan akan terus berfungsi seperti biasanya. Hingga pada akhirnya pada tahun 2014, Google mengumumkan untuk mengakuisisi Firestore [14].

Tim Firestore mengatakan bahwa bergabungnya Firestore dengan Google akan meningkatkan peluang produk mereka terus berkembang. Apalagi dengan bakat engineer, resource, dan infrastruktur dari Google, mereka dapat melakukan lebih banyak bahkan jauh lebih cepat lagi dalam pengembangan Firestore. Hal tersebut terbukti sampai saat ini Google terus

# BAB

# 11

## CRUD FIRESTORE

### A. Teori Pendukung

Seperti yang sudah dipaparkan sebelumnya, Firebase adalah platform yang dikembangkan oleh Google untuk memudahkan proses membangun aplikasi. Tak hanya hanya itu, Firebase juga memungkinkan Anda untuk meningkatkan performa aplikasi yang sudah berjalan atau sudah ada di Play Store. Hingga saat ini, semakin banyak developer yang menyukai Firebase, baik dari startup hingga perusahaan global. Hal ini terbukti dengan berbagai feedback positif terkait penggunaan Firebase, salah satunya datang dari Fabulous, aplikasi daily planner dan habit tracking yang tersedia di App Store dan Google Play Store. Sami Ben Hassine, Co Founder dan CEO Fabulous, mengatakan bahwa mereka dapat meningkatkan retensi dua kali lipat dengan alur khusus untuk onboarding pengguna. Mereka juga mengatakan bahwa waktu yang dibutuhkan untuk mengimplementasikan alur onboarding baru hanya satu hari saja.

Cloud Firestore merupakan NoSQL document database yang memungkinkan Anda untuk membangun aplikasi tanpa server (serverless) dan aman dalam skala global. Anda juga dapat menyimpan data aplikasi di cloud, menyinkronkan data di seluruh perangkat secara online dan offline, serta mengambil data tersebut dengan query yang lebih ekspresif. Lalu, apa itu Database NoSQL? Dalam database SQL, kita harus mengikuti aturan yang didefinisikan dalam bentuk tabel dan bersifat strict

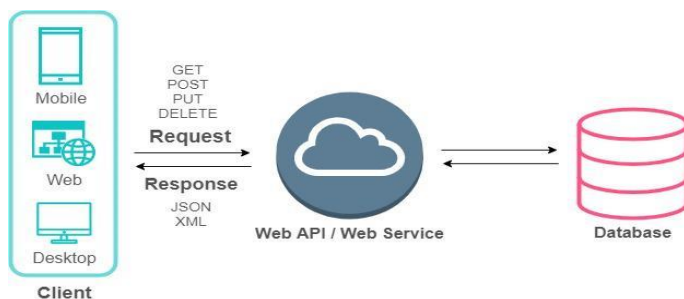
# BAB

# 12

## CRUD REST API

### A. Teori Pendukung

Aplikasi di internet pada umumnya dibuat dengan beberapa bahasa pemrograman dan harus dapat berjalan dengan baik pada sistem operasi, bahkan perangkat yang berbeda. REST API merupakan arsitektur yang dirancang pada web service yang telah diterima secara luas oleh publik untuk penggunaan sederhana dalam paradigma mobile cloud. RESTful web service bekerja dengan mengidentifikasi URI (Uniform Resource Identifier) kemudian memodifikasinya dengan perintah GET, POST, PUT atau DELETE yang dapat direpresentasikan dalam beberapa bahasa pemrograman [17]. REST ini hampir selalu menggunakan HTTP (Hypertext Transfer Protocol) untuk berkomunikasi data. HTTP adalah protokol yang memungkinkan Anda saling bertukar data antar server dan client.



Gambar 12. 1 Konsep Rest API



## DAFTAR PUSTAKA

- D. Krajci Iggy and Cummings, "History and Evolution of the Android OS," in *Android on x86: An Introduction to Optimizing for Intel® Architecture*, Berkeley, CA: Apress, 2013, pp. 1–8. doi: 10.1007/978-1-4302-6131-5\_1.
- Android, "Android untuk Developer." <https://developer.android.com/> (accessed Jun. 30, 2023).
- JetBrains, "FAQ: What Is Kotlin." <https://kotlinlang.org/docs/faq.html> (accessed Jun. 30, 2023).
- Google, "Mengenal Android Studio." <https://developer.android.com/studio/intro?hl=id> (accessed Jun. 30, 2023).
- Google, "Memahami Siklus Proses Aktivitas", Accessed: Jun. 30, 2023. [Online]. Available: <https://developer.android.com/guide/components/activities/activity-lifecycle?hl=id>
- Google, "Fragment", Accessed: Jun. 30, 2023. [Online]. Available: <https://developer.android.com/guide/components/fragments?hl=id>
- Google, "Create dynamic lists with RecyclerView ", Accessed: Jun. 30, 2023. [Online]. Available: <https://developer.android.com/develop/ui/views/layout/recyclerview>
- Google, "ScrollView", Accessed: Jun. 30, 2023. [Online]. Available: <https://developer.android.com/reference/android/widget/ScrollView>
- Google, "Menambahkan peta." <https://developer.android.com/training/maps?hl=id> (accessed Jun. 30, 2023).
- Google, "Broadcasts overview".

- Google, "Save simple data with SharedPreferences." <https://developer.android.com/training/data-storage/shared-preferences> (accessed Jun. 30, 2023).
- Google, "SharedPreferences", Accessed: Jun. 30, 2023. [Online]. Available: <https://developer.android.com/reference/android/content/SharedPreferences>
- Google, "Save data using SQLite." <https://developer.android.com/training/data-storage/sqlite> (accessed Jun. 30, 2023).
- Firebase, "Apa itu Firebase?", Accessed: Jun. 30, 2023. [Online]. Available: <https://firebase.google.com/?hl=id>
- Firebase, "Firebase Authentication", Accessed: Jun. 30, 2023. [Online]. Available: <https://firebase.google.com/docs/auth?hl=id>
- Firebase, "Cloud Firestore." <https://firebase.google.com/docs/firestore?hl=id> (accessed Jun. 30, 2023).
- I. O. Suzanti, N. Fitriani, A. Jauhari, and A. Khozaimi, "REST API Implementation on Android Based Monitoring Application," *J Phys Conf Ser*, vol. 1569, no. 2, p. 022088, 2020, doi: 10.1088/1742-6596/1569/2/022088.

## TENTANG PENULIS



Novian Adi Prasetyo, seorang pengajar di bidang Ilmu Informatika dari Institut Teknologi Telkom Purwokerto, Ia berhasil meraih gelar Sarjana (S1) dari STIKOM Yos Sudarso Purwokerto dan melanjutkan studi untuk meraih gelar Magister (S2) di Universitas Atma Jaya Yogyakarta. Sebelum bergabung sebagai dosen, Novian memiliki pengalaman sebagai seorang programmer di perusahaan manufaktur, di mana ia bertanggung jawab dalam mengelola sistem enterprise resource planning. Setelah itu, ia melanjutkan karirnya sebagai programmer di sebuah software house yang mengembangkan aplikasi sistem keuangan.

Sebagai dosen, Novian melaksanakan kegiatan Tri Dharma perguruan tinggi, termasuk mengajar beberapa mata kuliah seperti Pemrograman Perangkat Bergerak, Pemrograman Website, dan Interaksi Manusia Komputer. Ia juga aktif dalam penelitian, dan pada tahun 2023, terlibat dalam proyek pembuatan Sistem Informasi "Banyumas Belajar Pemilu" bekerja sama dengan KPU Kabupaten Banyumas. Selain itu, pada tahun yang sama, Novian juga berpartisipasi dalam proyek pengabdian kepada masyarakat dengan mengimplementasikan aplikasi point of sales untuk mendukung kegiatan batik ecoprint di Purbalingga.

Kiprah dan kontribusi Novian sebagai dosen tidak hanya memberikan manfaat bagi mahasiswa di kampusnya, tetapi juga memberikan dampak positif pada masyarakat lebih luas melalui penelitian dan pengabdian yang ia lakukan. Dengan pengetahuan dan pengalaman dalam bidang Ilmu Informatika, Novian terus berupaya untuk berkontribusi dalam pengembangan teknologi dan pendidikan di Indonesia.



Muhamad Azrino

Gustalika,S.Kom., M.Tr.T. atau akrab dipanggil Pak Azrino, ialah salah satu dosen Teknik Informatika di Institut Teknologi Telkom Purwokerto. Beliau lahir di Semarang dan mempunyai dua tanggal lahir yaitu secara administratif pada 14 Agustus 1993. Pak Azrino melakukan pendidikannya di bangku SD Dharma Wanita Universitas Brawijaya yang sekarang berubah nama menjadi Brawijaya Smart School serta SMP Negeri 13 Kota Malang, lalu meneruskannya di SMK Negeri 5 Malang. Setelah tamat SMK, tepatnya pada tahun 2011 beliau meneruskan kuliahnya di Universitas Muhammadiyah Malang. Pada saat kuliah S1 Pak Azrino telah aktif dalam mengikuti kegiatan asisten praktikum atau asisten lab di kampus untuk membimbing mahasiswa angkatan di bawah saya untuk mengajari pemrograman mulai dari C++, PHP, hingga JAVA. Sejak saat itu beliau belajar coding atau source code secara otodidak dengan membaca setiap buku mengenai pemrograman android dan melihat youtube. Setelah lulus S1 Pak Azrino bekerja sebagai Programmer di Salah satu software house di Malang. Beliau lalu melanjutkan studi S2 di Politeknik Negeri Malang dengan mengambil jurusan Teknik Terapan Elektro dengan penjurusan Computer Science dan lulus pada Mei 2019. Semenjak kuliah beliau telah aktif menjadi freelance. Pada tahun 2020 beliau mengabdikan dirinya di Institut Teknologi Telkom Purwokerto. Pada saat itu beliau mengajar Pemrograman Perangkat Bergerak. Saat ini Pak Azrino masih

mengajar di bidang pengkodean tepatnya mata kuliah Algoritma Struktur Data dan Pemrograman Perangkat Bergerak Android.



Muhammad Lulu Latif Usman, S.Pd., M.Han. atau biasa dipanggil Lulu merupakan seorang Dosen pada program Studi Teknik Informatika. Penulis menyelesaikan Pendidikan SLTA di SMKN 1 Bawang Banjarnegara Jurusan Teknik Komputer dan Jaringan, Pendidikan Strata 1 di Universitas Negeri Yogyakarta Jurusan Pendidikan Teknik Informatika, dan Pendidikan Strata 2 di Universitas Pertahanan Jurusan Teknologi Penginderaan. Selain menjadi Dosen di Program Studi Teknik Informatika Penulis juga merupakan seorang asesor dalam bidang Web Desain. Kompetensi keahlian penulis adalah dalam bidang Rekayasa Perangkat Lunak dan Multimedia dengan spesifikasi Sistem Informasi Geografis, Pemrograman Perangkat Bergerak, Pemrograman Web, Proyek Perangkat Lunak, dan Pemrograman Database.



Fahrudin Mukti Wibowo, lahir di kota Purwokerto. Selepas lulus dari SMA N 1 Purwokerto, kemudian melanjutkan pendidikan D3 di AKATEL Sandy Putra Purwokerto, S1 di Universitas Muhammadiyah Magelang, S2 di Universitas Gadjah Mada Yogyakarta, dan sedang menempuh S3 di Universiti Tun Husein Onn Malaysia. Bidang keilmuan yang ditekuni adalah Informatika dan bekerja sebagai pengajar di Institut Teknologi Telkom Purwokerto. Meskipun bidang utamanya adalah di jaringan computer dan sensor, penulis juga menekuni bidang Perangkat Lunak untuk Perangkat Bergerak. Penulis juga aktif di penelitian, kegiatan pengabdian kepada masyarakat dan menjadi narasumber.



Muhammad Rizqan Fadhil, S. Kom. dilahirkan di kota Banjarbaru, Kalimantan Selatan pada tanggal 19 September 2001. Penulis menyelesaikan Pendidikan SD, SMP di kota Martapura dan SMK di kota Banjarbaru. Pada tahun 2019 penulis menempuh pendidikan program S1 di Jurusan Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom

Purwokerto, Jawa Tengah. Bidang keahlian yang ditekuni oleh penulis adalah Mobile Application Development.