



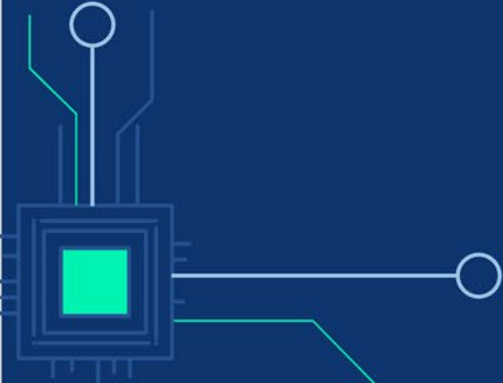
DIGITAL
TALENT
SCHOLARSHIP



VOCATIONAL SCHOOL GRADUATE ACADEMY

Mobile Programmer

Pertemuan #2- : Platform Operating System



KOMINFO



#JADIJAGOANDIGITAL

Badan Penelitian dan Pengembangan Sumber Daya Manusia

PROFIL PENGAJAR



Jabatan Akademik / Lektor / Kepala LAB Prodi Teknik Komputer

Latarbelakang Pendidikan Pengajar

- S1– STMIK Budi Darma (Skripsi : Aps Mobile Kompresi SMS)
- S2– Universitas Putra Indonesia YPTK Padang (Tesis : Aps Mobile Security SMS)

Riwayat Pekerjaan

- Dosen Tetap Politeknik Negeri Medan
- Trainer Pemrograman Java dan Mobile, Networking, Cyber Security, OS Server
- Konsultan Bidang Aplikasi, Networking, Cyber Security dan Server
- CEO PT. Nusa Tirta Teknologi

Sertifikat Kompetensi :

- Program : Senior Programmer (BNSP)
- Networking : Mikrotik, CISCO
- Server : Windows Server, Redhat
- Project : Comptia Project +

Contact Pengajar

Ponsel :-

Email : azanuddin@polmed.ac.id

Deskripsi Singkat

Deskripsi Singkat mengenai Topik

Materi ini berisi penjelasan mengenai cara menunjukkan *Platform Operating* Sistem serta Bahasa Pemrograman di dalam Perangkat Lunak (Dasar Java untuk Mobile Programming)

Tujuan Pelatihan

1. Peserta mampu menunjukkan arsitektur dasar sistem operasi berbasis mobile sesuai dengan perangkat keras yang digunakan.
2. Peserta mampu menunjukkan platform sistem operasi berbasis mobile sesuai dengan perangkat keras yang terkait.
3. Peserta mampu menunjukkan security pada platform sistem operasi berbasis mobile sesuai dengan perangkat keras yang digunakan
4. Peserta mampu merancang kebutuhan user berdasarkan spesifikasinya

Materi Yang akan disampaikan:

1. Sistem Operasi
2. Bahasa Pemrograman
3. Dasar pemrograman

Tugas :

Membuat algoritma untuk Pemrograman Mobile Android

Karakteristik Mobile Device

1. Portabel (Portability)
2. Berukuran Kecil (Small Size)
3. Wireless Communication
4. Memori Terbatas
5. Virtual Assistant

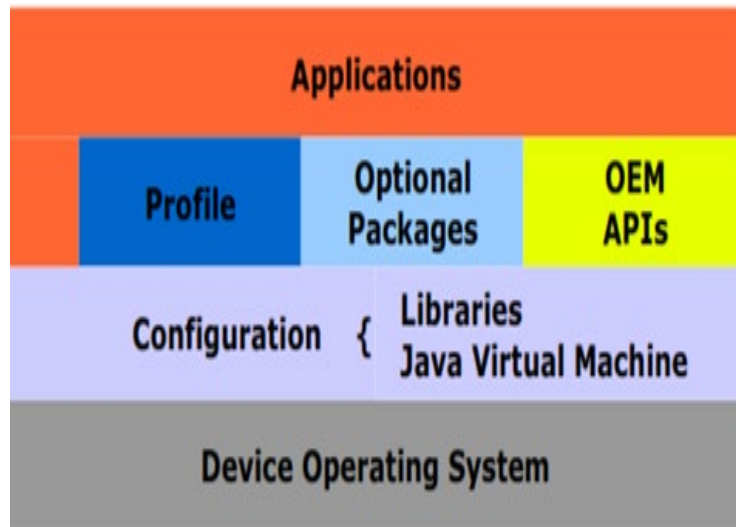
Pengertian Sistem Operasi Mobile



- ❖ **Software utama** yang melakukan manajemen dan kontrol terhadap *hardware* pada *mobile device* secara langsung.

Arsitektur Platform

1. Arsitektur Java Micro Edition



Configuration : minimal runtime dasar

- CDC(Connected Device Configuration)
- CLDC (Connected Limited Device Configuration)

Profile : library tambahan

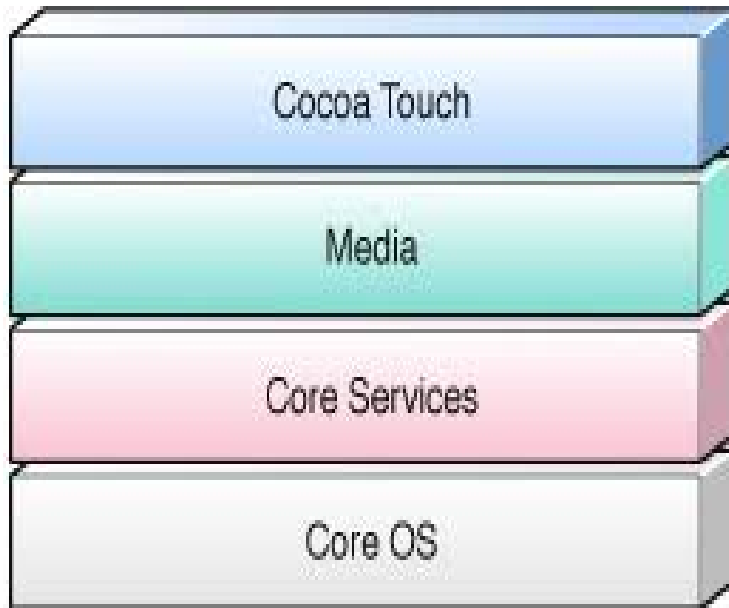
- MIDP 1.0 : Aplikasi berbasis text
- MIDP 2.0 : Aplikasi berbasis Multimedia

MIDlet : Aplikasi Java Micro Edition menggunakan teknologi MIDP

- ✓ `public void startApp() { }`
- ✓ `public void pauseApp() { }`
- ✓ `public void destroyApp(boolean unconditional) { }`

Arsitektur Platform

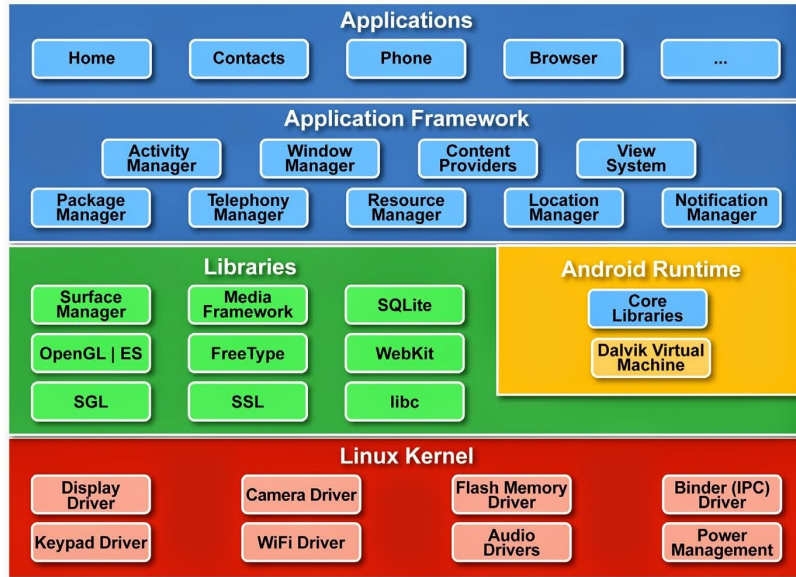
2. Arsitektur iOS



Cocoa Touch Layer

Lapisan Cocoa Touch berisi kunci kerangka kerja untuk membangun aplikasi IOS.

Arsitektur Android



- Android Terdiri dari tumpukan/*layer* perangkat lunak berbasis Linux

Layer Application and Widget



Layer Applications dan Widget

layer pertama pada OS Android : adalah layer Applications dan Widget.



Layer Applications Framework

layer kedua pada OS Android : adalah layer Applications Framework.

LAYER LIBRARIES



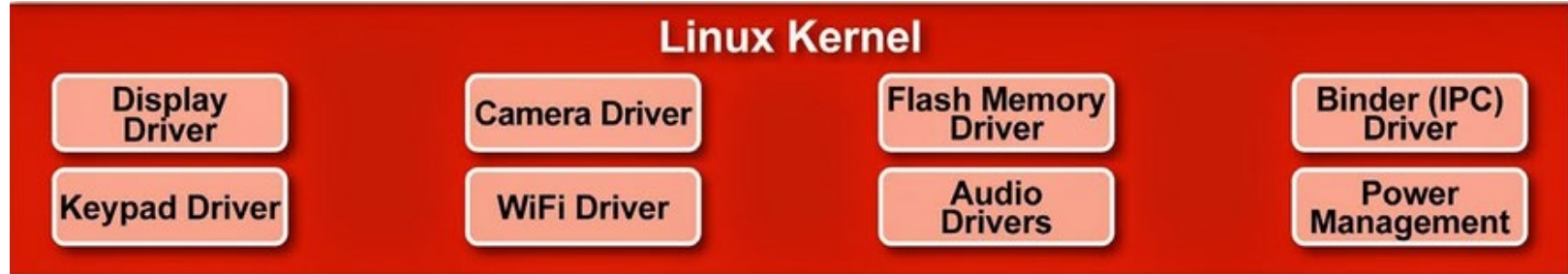
Libraries merupakan layer tempat fitur-fitur android berada.

LAYER ANDROID RUNTIME



- Android Runtime merupakan layer yang membuat aplikasi android bisa dijalankan.

Layer linux kernel



Linux Kernel merupakan layer tempat keberadaan inti dari operating system android. Layer ini berisi file-file system yang mengatur system processing, memory, resource, drivers, dan sistem android lainnya..

1. ANDROID



Fitur Perangkat Keras Android

1. Touch Screen/Layar HSentuh
2. GPS
3. Accelerometer
4. SD card

PERANGKAT KERAS IOS



- **IOS** (*Iphone Operating System*) adalah sistem operasi perangkat keras dari Apple.

SECURITY (ENKRIPSI)



Versi 3.0 dan selanjutnya memiliki fitur enkripsi *file system* (enkripsi menggunakan algoritma AES128, CBC, SHA256)



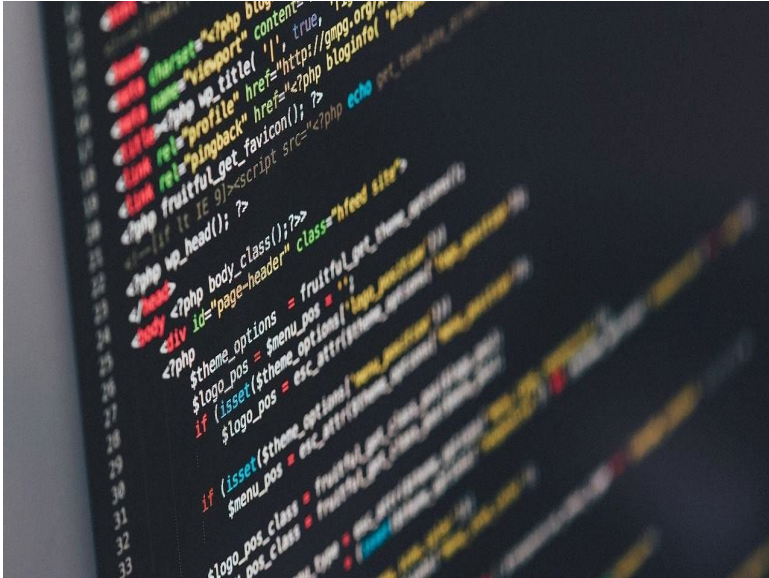
iOS4 dan selanjutnya memiliki fitur enkripsi dengan algoritma AES256 untuk mengenkripsi semua data yang tersimpan di memori



Memiliki beberapa algoritma enkripsi seperti AES, SHA1, SHA256, HMACSHA1, HMACSHA256



Enkripsi menggunakan kunci enkripsi simetrik saat transfer data antara *Blackberry devicedan Blackber ry enterprise serve*



- pemrograman yang ditujukan untuk pembuatan aplikasi diperangkat mobile.
- Diterapkan di mobile/Handphone atau Laptop.

BAHASA PEMROGRAMAN BERBASIS MOBILE



Beberapa faktor utama yang mempengaruhi pemilihan bahasa pemrograman tertentu adalah:

- Keseluruhan strategi pengembangan aplikasi seluler *native*, *cross platform*, *hybrid* dan sebagainya
- Skill set dari tim pengembangan
- Sasaran pengguna aplikasi dan dukungan platform yang dipilih
- Kebutuhan khusus aplikasi.
- Kategori aplikasi : Utility, games, B2B atau B2C dan lainnya.



Bahasa Pemrograman mana yang sesuai dengan Strategi Pengembangan Aplikasi Mobile?



Jenis Bahasa Pemrograman Mobile

Sistem Operasi Mobile dan bahasa pemrograman

1. Java Micro Edition (sebelumnya J2ME)
2. NET Compact Framework : C++, C#, VB.NET untuk Windows Mobile
3. Qualcomm's BREW : C atau C++
4. Symbian : C++
5. BlackBerry : Java
6. Android : Java
7. iPhone : Objective-C, Swift

Komponen Alat Bantu Pengembangan

Hardware:

Komputer development, mobile device

Software /Tool Pengembangan:

- Windows Phone : Windows Phone SDK, Visual Studio Express Expression Blend (hanya di windows)
- iOS : xCode , iOS SDK
- Android : Java JDK, Android SDK, Eclipse IDE, Android ADT, Android Studio IDE
- Java Micro Edition : Java JDK, Java Micro Edition SDK, Netbeans IDE, Eclipse
- Blackberry : Eclipse, Blackberry Plugin, Blackberry Simulator

Pemilihan Platform :

Android, iOS, Blackberry, Windows Phone, Symbian

Contoh Program Java Micro Edition

```
import javax.microedition.lcdui.*;
import javax.microedition.midlet.*;

public class HelloWorld extends MIDlet{
    private Form form;
    private Display display;

    public HelloWorld(){
        super();
    }

    public void startApp(){
        form = new Form("Hello World");
        String msg = "Hello World!!!!!!";
        form.append(msg);
        display = Display.getDisplay(this);
        display.setCurrent(form);
    }

    public void pauseApp(){}

    public void destroyApp(boolean unconditional){
        notifyDestroyed();
    }
}
```



Output pada Emulator

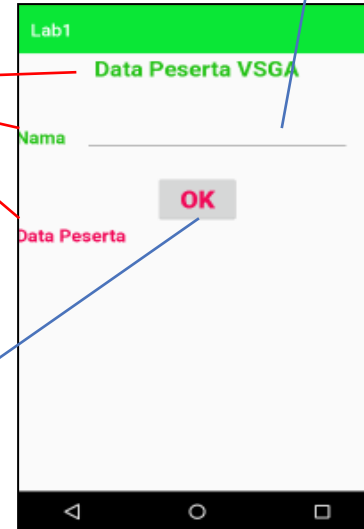
Contoh Program Android

```
public class MainActivity extends  
    AppCompatActivity {  
    Button btnOk, btnNext;  
    EditText editNama;  
    TextView txtHasil;  
    @Override  
    protected void onCreate(Bundle  
        savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        btnOk = (Button) findViewById(R.id.btnOk);  
        editNama = (EditText)  
            findViewById(R.id.editNama);  
        txtHasil = (TextView)  
            findViewById(R.id.txtHasil);  
        btnOk.setOnClickListener(new Button.  
            OnClickListener() {  
                @Override  
                public void onClick(View v) {  
                    txtHasil.setText("Hello " +  
                        editNama.getText().  
                            toString() + "! \nPeserta VSGA" + "");  
                }  
            });  
    }  
}
```

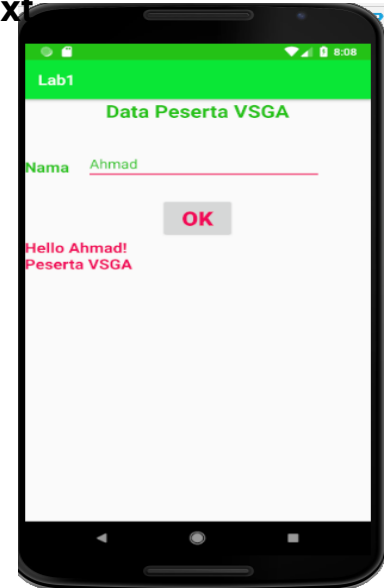
TextView

Button

EditText



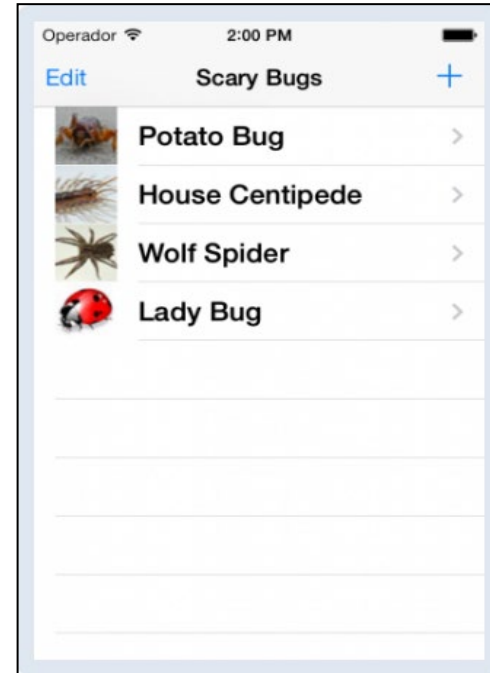
Desain User
Interface



Output Emulator/
/ AVD

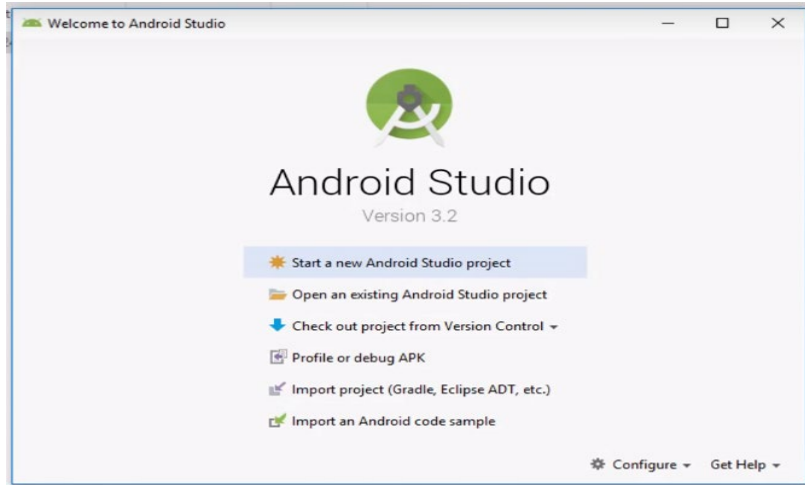
Contoh Program i Phone untuk i OS

```
#import "RWTScaaryBugDoc.h" // At beginning of
application:didFinishLaunchingWithOptions
RWTScaaryBugDoc *bug1 = [[RWTScaaryBugDoc alloc]
initWithTitle:@"Potato Bug" rating:4
thumbImage:[UIImage imageNamed:@"potatoBugThumb.jpg"]
fullImage:[UIImage imageNamed:@"wolfSpiderThumb.jpg"]
fullImage:[UIImage imageNamed:@"wolfSpider.jpg"]];
RWTScaaryBugDoc *bug4 = [[RWTScaaryBugDoc alloc]
initWithTitle:@"Lady Bug" rating:1
thumbImage:[UIImage imageNamed:@"ladybugThumb.jpg"]
fullImage:[UIImage imageNamed:@"ladybug.jpg"]];
NSMutableArray *bugs = [NSMutableArray
arrayWithObjects:bug1, bug2, bug3, bug4, nil];
UINavigationController *navController =
(UINavigationController *)
self.window.rootViewController;
RWTMasterViewController *masterController =
[navController.viewControllers objectAtIndex:0];
masterController.bugs = bugs;
```



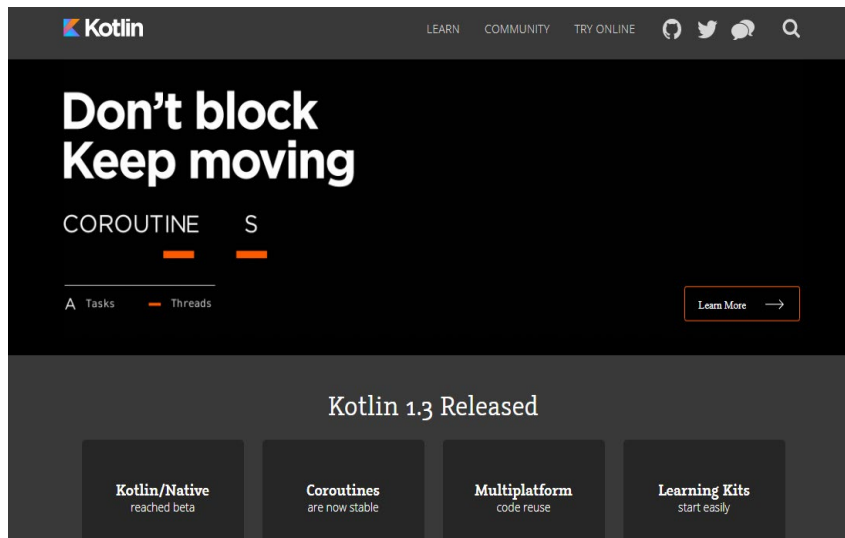
Output Emulator

Bahasa Pemrograman C++



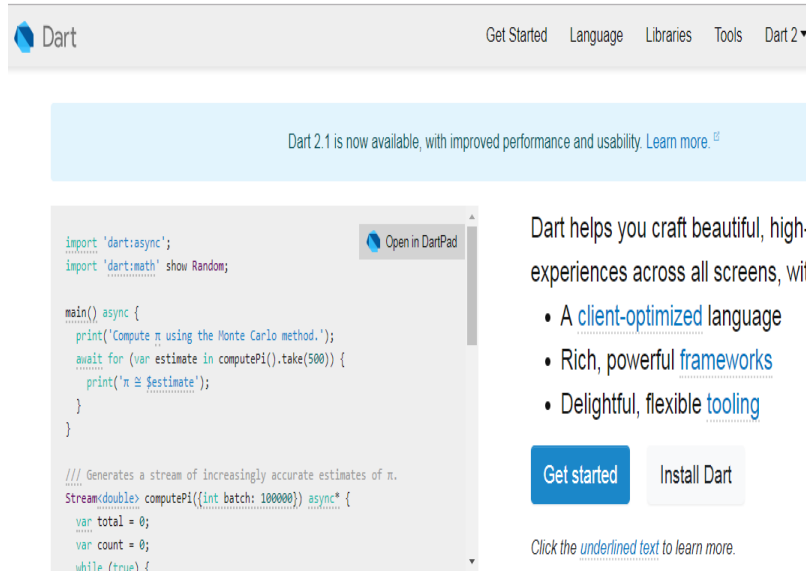
- IDE Android Studio dapat digunakan untuk mengembangkan aplikasi android dengan bahasa pemrograman C++

Bahasa Pemrograman Kotlin



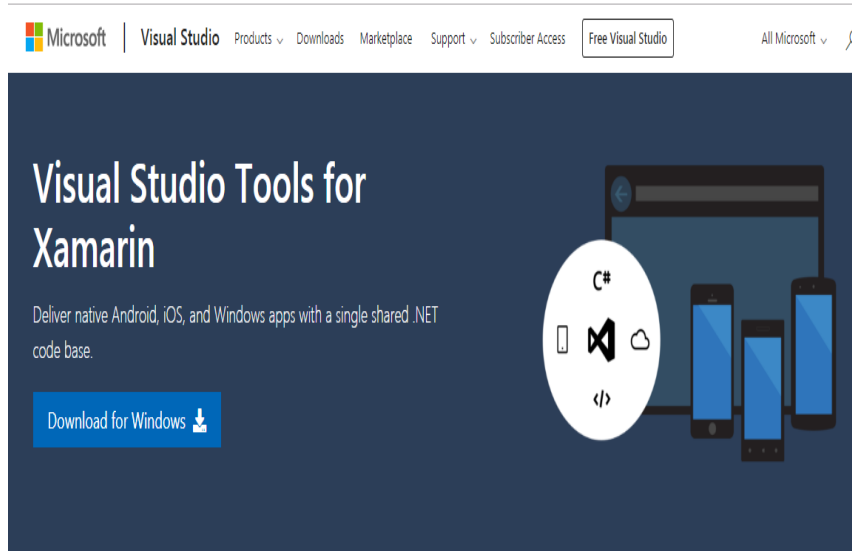
- **Kotlin** baru mendapatkan dukungan dari google sebagai salah satu bahasa pemrograman resmi untuk membuat aplikasi android.

Bahasa Pemrograman Dart



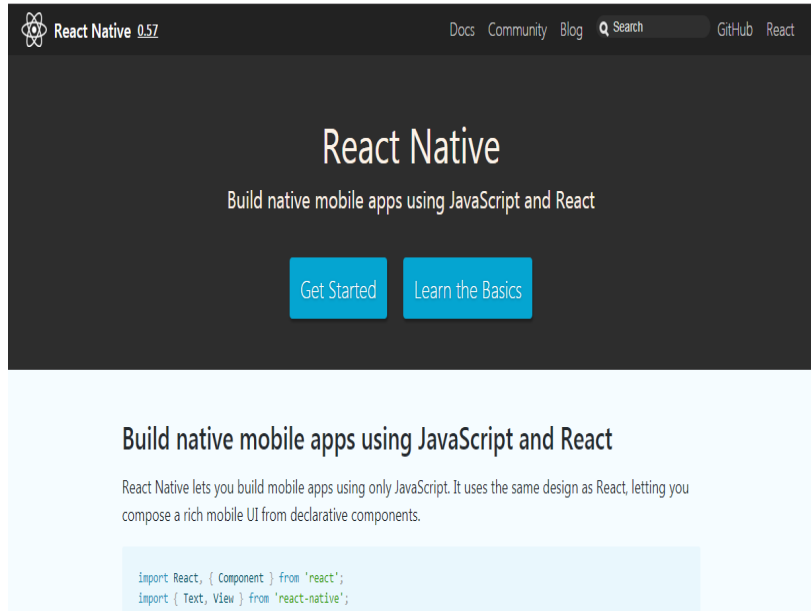
- Dart merupakan bahasa pemrograman yang dapat digunakan untuk membuat aplikasi berbagai platform.

Bahasa Pemrograman C#



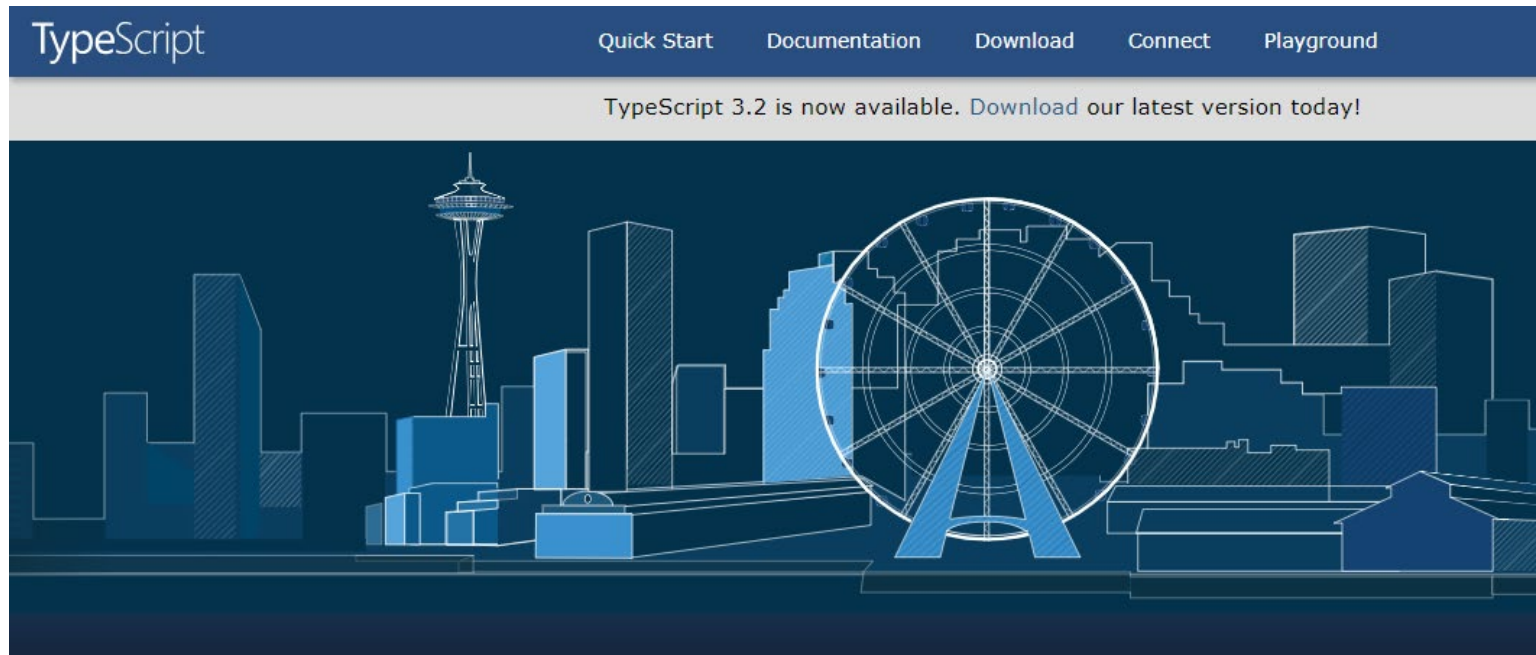
- Xamarin merupakan tools yang diperlukan untuk mengembangkan aplikasi android

Bahasa Pemrograman JavaScript



- React Native, Ionic, dan PhoneGap merupakan framework yang dapat digunakan untuk mengembangkan aplikasi android

Bahasa Pemrograman TypeScript



Bahasa Pemrograman SWIFT (IOS)



COMPILER DAN IDE

Swift sebuah bahasa pemrograman yang dikembangkan oleh **Apple Inc**

Objective-C



Sebelum adanya Swift, Objective-C merupakan bahasa yang dipakai untuk iOS.

CONTOH SISTEM OPERASI MOBILE



- iOS adalah sistem operasi mobile Apple.

ANDROID



- Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak.

WINDOWS MOBILE



- Windows Mobile adalah salah satu sistem operasi (OS) mobile yang dikembangkan oleh Microsoft dan di desain untuk digunakan pada smartphone dan perangkat nirkabel lainnya.

RIM BlackBerry



- BlackBerry OS adalah sistem operasi mobile yang dikembangkan RIM yang mendukung layanan multitasking untuk produk BlackBerry.

SYMBIAN



- Symbian OS adalah sistem operasi tak bebas yang dikembangkan oleh Symbian Ltd.

Versi Sistem Operasi Android Pelatihan

1. Android Versi 1.1 (2008)
2. Android Versi 1.5 (2009: Cupcake)
3. Android Versi 1.6 (2009: Donut)
4. Android Versi 2.0/2.1 (2009:Eclair)
5. Android versi 2.2 (2010 : Froyo : Frozen Yoghurt)
6. Android Versi 2.3 (2010: Gingerbread)
7. Android versi 3.0/3.1 (2011 : Honeycomb)
8. Android versi 4.0 (2011 :ICS : Ice Cream Sandwich)
9. Android versi 5.0 (2012 : Jelly Bean)
10. Android 4.4 (2013 : Kitkat)
11. Android 5.0 (2014 : Lollipop)
12. Android 6.0 (2015 : Marshmallow)
13. Android 7.0 (2016 : Nougat)
14. Android 8.0 (2017 : Oreo)
15. Android 9.0 (2018 : Pie)

Software Bahasa Pemrograman Mobile

Pada pelatihan ini perangkat lunak yang digunakan dalam membangun aplikasi berbasis mobile yaitu “Android Studio”

Apa itu Android ?

- Sistem operasi seluler berbasis linux kernel
- *User interface* layar sentuh
- Digunakan oleh lebih dari 80% smartphone
- Menjalankan perangkat seperti jam tangan, TV, dan mobil
- Lebih dari 2 Juta aplikasi Android di Google Play store
- Sangat mudah disesuaikan pada berbagai perangkat
- *Open source*

Interaksi User

- Isyarat sentuh: menggesek, mengetuk, mencubit
- Keyboard virtual untuk karakter, angka, dan emoji
- Mendukung Bluetooth, USB controller dan periferal

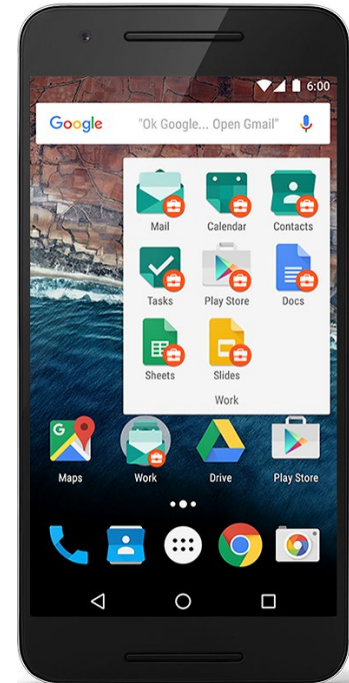
Android dan Sensor

Sensor bisa mengetahui aksi pengguna dan meresponsnya

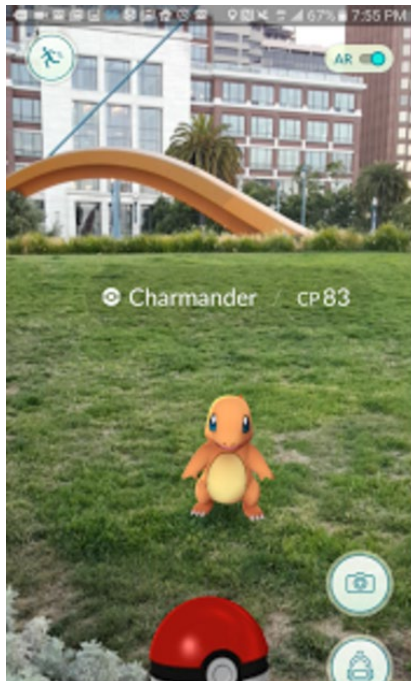
- Konten perangkat akan berotasi sesuai kebutuhan
- Berjalan akan menyesuaikan posisi di peta
- Memiringkan kemudi mobil virtual atau mengontrol mainan fisik
- Bergerak terlalu cepat menonaktifkan interaksi game

Layar Android

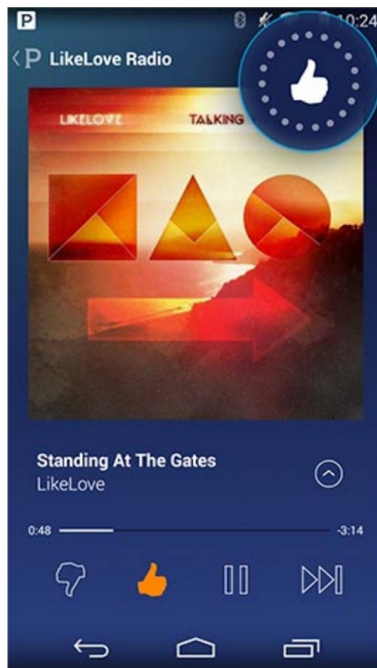
- Ikon peluncur aplikasi
- Widget pembaruan otomatis untuk konten langsung
- Bisa berisi beberapa halaman
- Folder untuk mengatur aplikasi
- "OK Google"



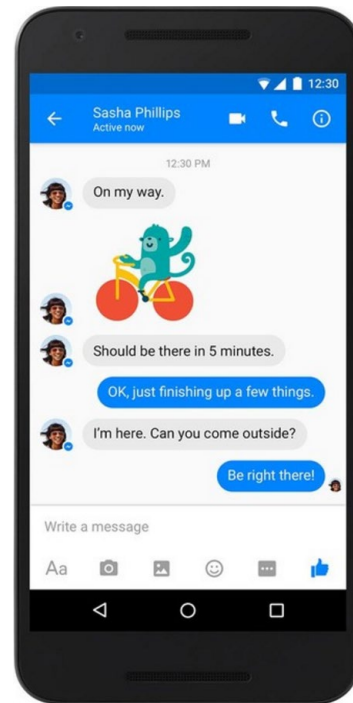
Contoh Aplikasi Android



Pokemon GO



Pandora

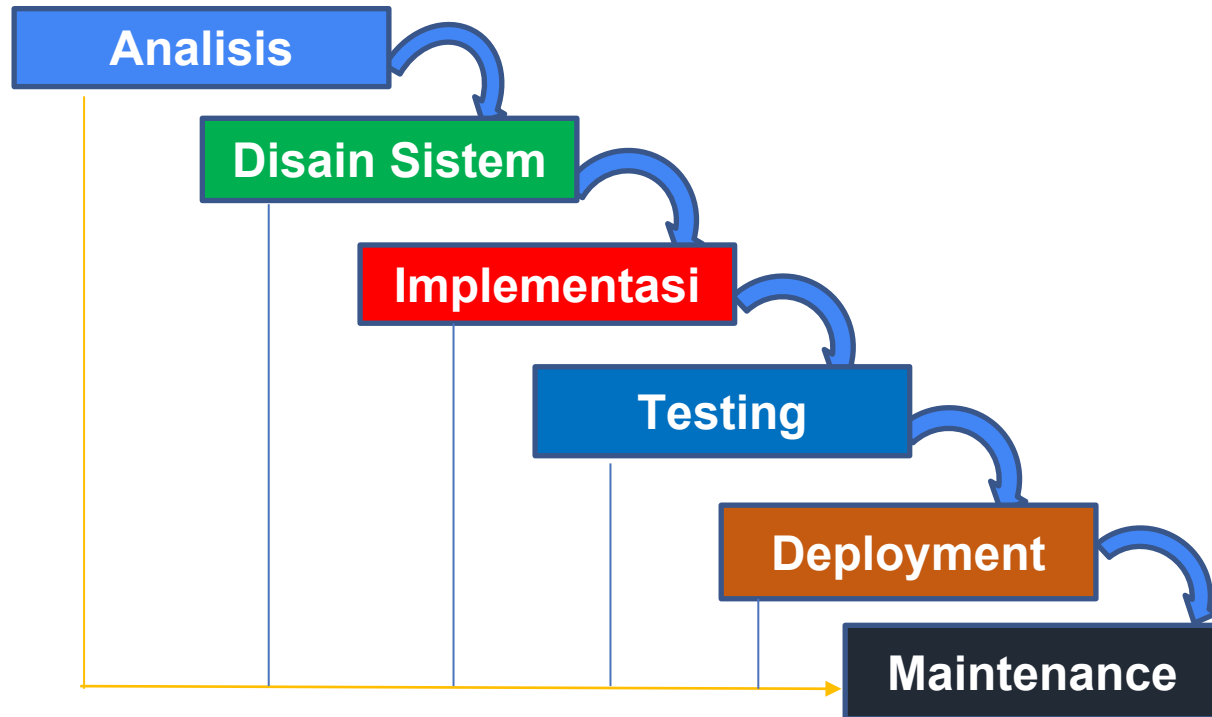


Facebook
Messenger

Android Software Developer Kit (SDK)

- *Development Tools* (debugger, monitor, editor)
- Pustaka (peta, perangkat yang dapat dikenakan)
- Perangkat virtual (emulator)
- Dokumentasi (developers.android.com)
- Kode contoh

Alur Membangun Aplikasi Mobile



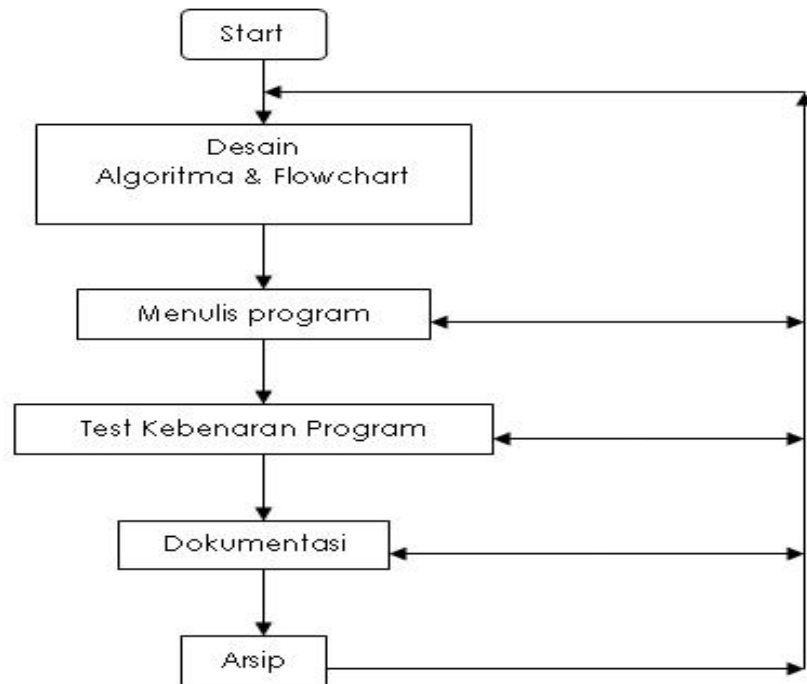
Algoritma Pemrograman

Pemrograman

1. Defenisi program
2. Tahapan pembuatan program

Algoritma Pemrograman

Tahapan Pembuatan Program



Algoritma Pemrograman

Algoritma

- Inti dari ilmu komputer.
- Defenisi Algoritma
- Algoritma blueprint dari program.
- Kriteria suatu algoritma:
 - Ada input dan output
 - Efektivitas dan efisien
 - Terstruktur

Algoritma Pemrograman

Contoh Algoritma 1

Mencari akar bulat positif dari bilangan bulat (integer) positif a:

- Masukkan bilangan bulat positif a
- Berikan harga awal x sama dengan 1
- Hitung y sebesar $x * x$
- Jika y sama dengan a maka cetak x sebagai akar dari a. Selesai
- Tambah nilai x dengan 1
- Pergi ke langkah 3.

Algoritma Pemrograman

Notasi Algoritma Flowchart

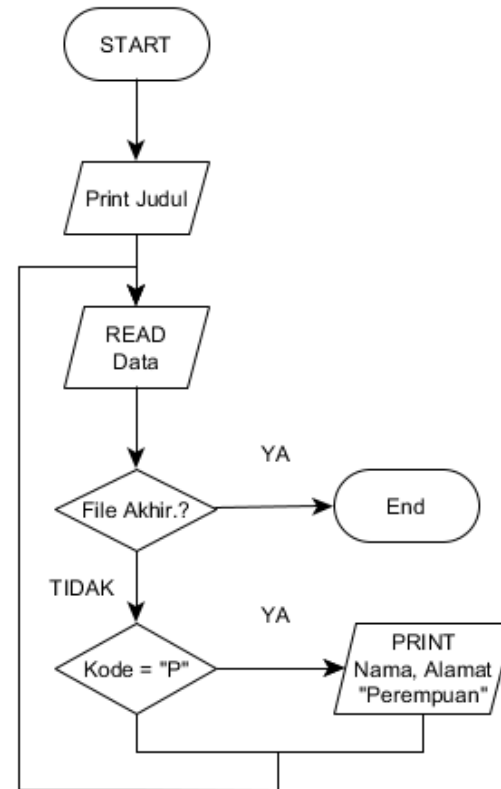
Penyelesaian masalah dalam bentuk bagan.

Ada 2 macam Flowchart :

- System Flowchart
- Program Flowchart

Algoritma Pemrograman

Contoh Program Flowchart








Algoritma Pemrograman

Simbol-simbol Flowchart

- *Flow Direction Symbols* (Simbol penghubung alur)
- *Processing Symbols* (Simbol proses).
- *Input-output Symbols* (Simbol input-output)
- Decision Symbol (Simbol Keputusan)
- Connector Symbol





Algoritma Pemrograman

Simbol-simbol Flowchart

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/ pemberian harga awal
	PROSES	Proses perhitungan/ proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi

Algoritma Pemrograman

Simbol-simbol Flowchart

SIMBOL	NAMA	FUNGSI
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/ proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

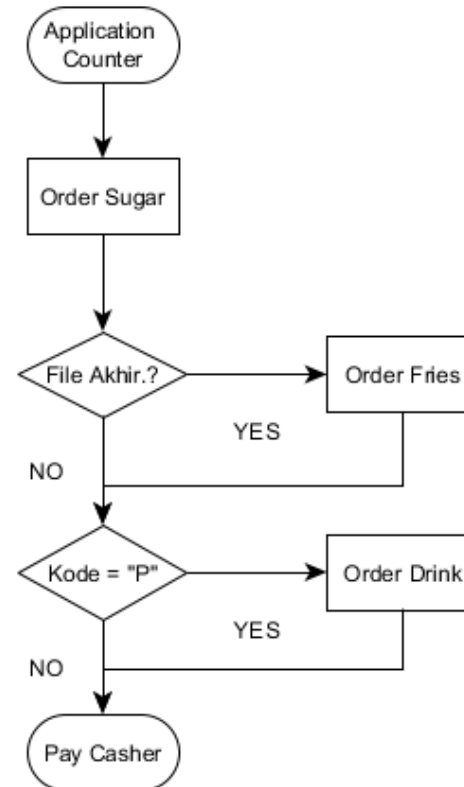
Algoritma Pemrograman

Syarat Pembuatan Flowchart

- Tidak ada kaidah yang baku.
- Flowchart = gambaran hasil analisa suatu masalah
- Flowchart dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.
- Secara garis besar ada 3 bagian utama:
 - Input
 - Proses
 - Output

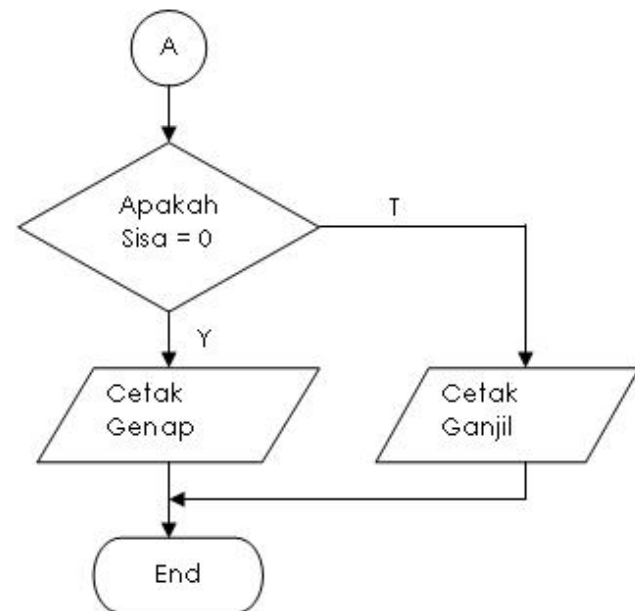
Algoritma Pemrograman

Contoh Flowchart 1



Algoritma Pemrograman

Contoh Flowchart 2



Tugas

1. Buatlah algoritma untuk menghitung luas dan keliling lingkaran. Dengan masukan berupa jari-jari.
2. Buatlah flowchart dari algoritma pada soal no 1 diatas.
3. Buatlah algoritma untuk mengecek bilangan di antara 2 bilangan masukan, apakah sama ataupun lebih besar salah satunya, dan tampilkan hasilnya.
4. Buatlah flowchart dari algoritma pada soal no 3 diatas.

Aplikasi Android

- Satu atau beberapa layar interaktif
- Ditulis menggunakan Bahasa Pemrograman Java dan XML
- Menggunakan Android Software Development Kit (SDK)
- Menggunakan pustaka Android dan Android Application Framework
- Dijalankan oleh mesin Android Runtime Virtual (ART)

Tantangan Development Android

- Beragam ukuran layar dan resolusi
- Kinerja: membuat aplikasi Anda responsif dan mulus
- Keamanan: menjaga kode sumber dan data pengguna agar tetap aman
- Kompatibilitas: berjalan dengan baik pada versi platform yang lebih lama
- Pemasaran: memahami pasar dan pengguna Anda

Dasar-Dasar Pemrograman Android Studio

1. Java Identifier

Java Identifier adalah suatu tanda yang mewakili nama-nama variabel, method, class, dan lain-lain.

Contoh Identifier : Hello, main, System, out.

Deklarasi identifier di Java adalah case-sensitive yaitu pendefenisian huruf besar dan huruf kecil adalah tidak sama

Dasar-Dasar Pemrograman Android Studio

1. Java Identifier

Syarat identifier :

- Dimulai dengan salah satu huruf, under- score “_”, atau tanda dollar “\$”.
- Huruf dapat berupa huruf besar maupun huruf kecil, dapat menggunakan nomor 0 sampai 9.
- Identifier tidak dapat menggunakan kata kunci dalam Java seperti class, public, void, dsb.

Dasar-Dasar Pemrograman Android Studio

Contoh Java Keyword

<code>abstract</code>	<code>boolean</code>	<code>break</code>	<code>byte</code>	<code>case</code>	<code>catch</code>
<code>char</code>	<code>class</code>	<code>const</code>	<code>continue</code>	<code>default</code>	<code>do</code>
<code>double</code>	<code>else</code>	<code>extends</code>	<code>final</code>	<code>finally</code>	<code>float</code>
<code>for</code>	<code>goto</code>	<code>if</code>	<code>implements</code>	<code>import</code>	<code>instanceof</code>
<code>int</code>	<code>interface</code>	<code>long</code>	<code>native</code>	<code>new</code>	<code>package</code>
<code>private</code>	<code>protected</code>	<code>public</code>	<code>return</code>	<code>short</code>	<code>static</code>
<code>strictfp</code>	<code>super</code>	<code>switch</code>	<code>synchronized</code>	<code>this</code>	<code>throw</code>
<code>throws</code>	<code>transient</code>	<code>try</code>	<code>void</code>	<code>volatile</code>	<code>while</code>
<code>assert</code>	<code>enum</code>				

Dasar-Dasar Pemrograman Android Studio

2. Pernyataan dan Block

Pernyataan merupakan satu atau lebih baris program yang diakhiri dengan titik koma (semi colon)

Contoh :

```
setContentView(R.layout.activity_main);
```

Block adalah satu atau lebih baris program yang terletak antara kurung kurawal buka dan kurung kurawal tutup.

Contoh :

```
{  
getMenuInflater().inflate(R.menu.activity_main, menu);  
return true;  
}
```

Dasar-Dasar Pemrograman Android Studio

Tipe data primitif

- ✓ Logical - boolean
- ✓ Textual - char
- ✓ Integral - byte, short, int, dan long
- ✓ Floating - double dan float

Dasar-Dasar Pemrograman Android Studio

2. Tipe data primitif

Primitive Data Types and Their Effective Sizes

Type	Effective Representation Size (bits)	Type	Effective Representation Size (bits)
boolean	1	char	16
byte	8	short	16
int	32	long	64
float	32	double	64

Dasar-Dasar Pemrograman Android Studio

3. Deklarasi Variabel

Sintak umum untuk mendeklarasikan dan menginisialisasi variabel

```
<modifier> <Tipe data> <Nama variabel> = <nilai awal>
```

Contoh : mendeklarasikan private variabel id dengan tipe data int dan variabel id diberikan nilai awal 20

```
private int id = 20;
```

Referensi

1. " ", Java TM Programming Language, Oracle America
2. Android Cook Book, McGraw-Hill/Osborne, 2013
3. Herbert Schildt, *Java2 : A beginner's Guide*, Second Edition, McGraw-Hill/Osborne
4. Matthew Mathias, Swift Programming, 2nd edition, Big Nerd Ranch
5. <https://developer.apple.com/library/archive/referencelibrary/GettingStarted/DevelopiOSAppsSwift/index.html/>
6. <https://developer.android.com/topic/libraries/architecture>
7. <https://www.oracle.com/technetwork/java/javase/overview/index.html>

Tim Pengajar

- Alif Akbar Fitrawan, S.Pd, M. Kom (Politeknik Negeri Banyuwangi);
- Anwar, S.Si, MCs. (Politeknik Negeri Lhokseumawe);
- Eddo Fajar Nugroho (BPPTIK Cikarang);
- Eddy Tungadi, S.T., M.T. (Politeknik Negeri Ujung Pandang);
- Fitri Wibowo (Politeknik Negeri Pontianak);
- Ghifari Munawar (Politeknik Negeri Bandung);
- Hetty Meileni, S.Kom., M.T. (Politeknik Negeri Sriwijaya) ;
- I Wayan Candra Winetra, S.Kom., M.Kom (Politeknik Negeri Bali) ;
- Irkham Huda (Vokasi UGM) ;
- Josseano Amakora Koli Parera, S.Kom., M.T. (Politeknik Negeri Ambon) ;
- I Komang Sugiarta, S.Kom., MMSI (Universitas Gunadarma) ;
- Lucia Sri Istiyowati, M.Kom (Institut Perbanas) ;
- Maksy Sendiang, ST, MIT (Politeknik Negeri Manado) ;
- Medi Noviana (Universitas Gunadarma) ;
- Muhammad Nashrullah (Politeknik Negeri Batam) ;
- Nat. I Made Wiryana, S.Si., S.Kom., M.Sc. (Universitas Gunadarma) ;
- Rika Idmayanti, ST, M.Kom (Politeknik Negeri Padang) ;
- Rizky Yuniar Hakkun (Politeknik Elektronik Negeri Surabaya) ;
- Robinson A.Wadu, ST., MT (Politeknik Negeri Kupang) ;
- Roslina. M.IT (Politeknik Negeri Medan) ;
- Sukamto, SKom., MT. (Politeknik Negeri Semarang) ;
- Syamsi Dwi Cahya, M.Kom. (Politeknik Negeri Jakarta) ;
- Syamsul Arifin, S.Kom, M.Cs (Politeknik Negeri Jember) ;
- Usmanudin (Universitas Gunadarma) ;
- Wandy Alifha Saputra (Politeknik Negeri Banjarmasin) ;

#JADIJAGOANDIGITAL TERIMA KASIH



digitalent.kominfo



DTS_kominfo



digitalent.kominfo



digital talent scholarship