

**APLIKASI KEHADIRAN DOSEN  
MENGUNAKAN OOP PADA PHP**

## **ABSTRAK**

**Kata Kunci:**

## ***ABSTRACT***

***Keywords:***

## **KATA PENGANTAR**

Assalamualaikum warahmatullahi wabarakatuh. Segala puji bagi Allah SWT yang telah memberikan kemudahan sehingga dapat menyelesaikan laporan Membuat Aplikasi Proyek II ini, tanpa pertolongan-Nya mungkin penulis tidak akan sanggup menyelesaikannya dengan baik. Shalawat dan salam semoga terlimpah curahkan kepada Nabi Muhammad SAW beserta sahabat dan keluarga Beliau.

Laporan ini disusun untuk memenuhi kelulusan matakuliah Proyek II pada Program Studi DIV Teknik Informatika. Proses Proyek II ini juga tidak terlepas dari bantuan pihak Pembimbing. Oleh karena itu, pada kata pengantar ini penulis menyampaikan terimakasih kepada :

1. Roni Habibi, S.Kom., M.T. selaku Pembimbing Internal dan Penguji Utama dalam penyusunan laporan Proyek II ini;
2. selaku Penguji Pendamping dalam Sidang Proyek II ini;
3. Rolly Maulana Awangga, S.T., M.T. selaku Koordinator Proyek II Tahun Akademik 2019/2020;
4. M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Program Studi DIV Teknik Informatika Tahun Akademik 2019/2020;
5. Dr. Ir. Agus Purnomo, M.T. selaku Direktur Politeknik Pos Indonesia Tahun Akademik 2019/2020.

Penulis telah membuat laporan ini dengan sebaik-baiknya, diharapkan memberikan kritik dan saran dari semua pihak yang bersifat membangun, terimakasih.

Bandung, Desember 2019

Penulis

## DAFTAR ISI

ABSTRAK .....	
<i>ABSTRACT</i> .....	
KATA PENGANTAR .....	
DAFTAR ISI .....	
DAFTAR GAMBAR .....	
DAFTAR TABEL .....	
DAFTAR SIMBOL .....	
BAB I .....	
PENDAHULUAN .....	<b>Error! Bookmark not def</b>
1.1 Latar Belakang .....	<b>Error! Bookmark not def</b>
1.2 Identifikasi Masalah .....	<b>Error! Bookmark not def</b>
1.3 Tujuan .....	<b>Error! Bookmark not def</b>
1.4 Ruang Lingkup .....	<b>Error! Bookmark not def</b>
1.5 Sistematika Penulisan .....	<b>Error! Bookmark not def</b>
BAB II .....	<b>Error! Bookmark not def</b>
LANDASAN TEORI .....	<b>Error! Bookmark not def</b>
2.1 Kajian Pustaka .....	<b>Error! Bookmark not def</b>
2.2 Analisis .....	<b>Error! Bookmark not def</b>
2.3 Basis Data .....	<b>Error! Bookmark not def</b>
2.4 Use Case Diagram .....	<b>Error! Bookmark not def</b>
2.5 Class Diagram .....	<b>Error! Bookmark not def</b>
2.6 Metode Penelitian .....	<b>Error! Bookmark not def</b>
2.7 Hasil Penelitian .....	<b>Error! Bookmark not def</b>
BAB III .....	<b>Error! Bookmark not def</b>
ANALISIS DAN PERANCANGAN .....	<b>Error! Bookmark not def</b>
3.1 Analisis .....	<b>Error! Bookmark not def</b>
3.1.1 Analisis Sistem yang sedang Berjalan pada Registrasi PMB Poltekpos .....	<b>Error! Bookmark not def</b>
3.1.2 Analisis Sistem yang akan Dibangun .....	<b>Error! Bookmark not def</b>

3.2	Perancangan .....	<b>Error! Bookmark not def</b>
3.2.1	Use Case Diagram.....	<b>Error! Bookmark not def</b>
3.2.2	Class Diagram .....	<b>Error! Bookmark not def</b>
BAB IV	.....	<b>Error! Bookmark not def</b>
IMPLEMENTASI	.....	<b>Error! Bookmark not def</b>
4.1	Lingkungan Implementasi.....	<b>Error! Bookmark not def</b>
4.1.1	Lingkungan Perangkat Keras .....	<b>Error! Bookmark not def</b>
4.1.2	Lingkungan Perangkat Lunak .....	<b>Error! Bookmark not def</b>
4.2	Pembahasan Hasil Implementasi.....	<b>Error! Bookmark not def</b>
BAB V	.....	<b>Error! Bookmark not def</b>
KESIMPULAN DAN SARAN	.....	<b>Error! Bookmark not def</b>
5.1	Kesimpulan.....	<b>Error! Bookmark not def</b>
5.2	Saran .....	<b>Error! Bookmark not def</b>
DAFTAR PUSTAKA	.....	<b>Error! Bookmark not def</b>

## DAFTAR GAMBAR

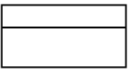
Gambar 3.1 Flowmap sistem yang sedang berjalan.....	<b>Error! Bookmark not def</b>
Gambar 3.2 Flowmap sistem yang akan dibangun .....	<b>Error! Bookmark not def</b>
Gambar 3.3 Perancangan use case diagram .....	<b>Error! Bookmark not def</b>
Gambar 3.4 Perancangan class diagram .....	<b>Error! Bookmark not def</b>
Gambar 4.1 Tampilan awal website.....	<b>Error! Bookmark not def</b>
Gambar 4.2 Tampilan menu untuk pendaftaran.....	<b>Error! Bookmark not def</b>
Gambar 4.3 Tampilan jalur registrasi .....	<b>Error! Bookmark not def</b>
Gambar 4.4 Tampilan jalur registrasi .....	<b>Error! Bookmark not def</b>
Gambar 4. 5 Tampilan registrasi data calon mahasiswa (2)	<b>Error! Bookmark not d</b>
Gambar 4.6 Tampilan registrasi data orang tua .....	<b>Error! Bookmark not def</b>
Gambar 4.7 Tampilan registrasi data sekolah .....	<b>Error! Bookmark not def</b>



## DAFTAR TABEL

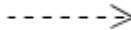
Tabel 3.1 Dokumen penginputan data calon mahasiswa .	<b>Error! Bookmark not def</b>
Tabel 3.2 Dokumen penginputan data orang tua .....	<b>Error! Bookmark not def</b>
Tabel 3.3 Dokumen pembuatan laporan calon mahasiswa	<b>Error! Bookmark not de</b>
Tabel 3.4 Dokumen pembuatan laporan orang tua .....	<b>Error! Bookmark not def</b>
Tabel 3.5 Spesifikasi minimum perangkat keras server yang digunakan untuk menganalisis .....	<b>Error! Bookmark not def</b>
Tabel 3.6 Spesifikasi minimum perangkat keras client ...	<b>Error! Bookmark not def</b>
Tabel 3.7 Spesifikasi minimum perangkat lunak server yang digunakan untuk menganalisis .....	<b>Error! Bookmark not def</b>
Tabel 3.8 Spesifikasi minimum perangkat lunak client...	<b>Error! Bookmark not def</b>

## DAFTAR SIMBOL




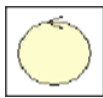
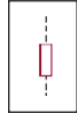

No.	Simbol	Nama	Keterangan
1		<i>Class</i>	Himpunan dari objek - objek yang berbagi atribut serta operasi yang sama.

1. **S  
i  
m  
b  
o  
l**






***C  
l  
a  
ss Diagram***

2. Simbol		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
-----------	---	-------------------	--

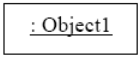

### *quence Diagram*



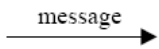
No.	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menggambarkan orang yang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.
3		<i>Boundary Class</i>	Menggambarkan sebuah penggambaran dari form.
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel.
5		<i>Lifeline</i>	Menggambarkan tempat mulai dan berakhirnya sebuah pesan.
6		<i>Line Message</i>	Menggambarkan pengiriman pesan.

### 3. Simbol Activity Diagram



No.	Simbol	Nama	Keterangan
1		<i>Actifity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Actifity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran





#### 4. Simbol *Collaboration Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Object</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Actor</i>	Actor juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom. Simbol Actor sama dengan simbol pada Actor Use Case Diagram.

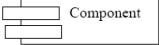
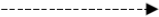
3		<i>Lifeline</i>	Lifeline mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk Lifeline adalah garis putus-putus vertikal yang ditarik dari sebuah obyek.
4		<i>Activation</i>	Activation dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah lifeline. Activation mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.
5		<i>Message</i>	Activation dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah lifeline. Activation mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.

## 5. Simbol Statechart Diagram

No .	Simbol	Nama	Keterangan
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali

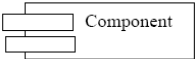


3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

## 6. Simbol *Component Diagram*

No.	Simbol	Nama	Keterangan
		<i>Component</i>	Sebuah komponen melambangkan sebuah entitas software dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.
		<i>Depedency</i>	Sebuah Dependency digunakan untuk menotasikan relasi

			antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.
--	--	--	---

## 7. Simbol *Deployment Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Component</i>	Pada deployment diagram, komponen-komponen yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka
2		<i>Node</i>	Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.
3		<i>Association</i>	Sebuah association digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara element-elemen hardware.

## 8. Simbol *Object Diagram*



# BAB I

## APLIKASI

### 1.1 Sejarah Aplikasi

Sebelum aplikasi pertama di dunia ditemukan, ada sejarah yang sangat panjang dalam proses ditemukannya aplikasi. Jauh sebelum aplikasi yang ada pada komputer dapat berjalan secanggih saat ini, semua perangkat lunak (*software*) selalu mengandalkan aljabar Boolean. Aljabar ini menggunakan kode binary digit (bit) yang terdiri dari dua angka yaitu 1 (benar/*on*) dan 0 (salah/*off*). Rumitnya (terlalu panjang) penggunaan kode binary digit ini membuat orang-orang pada saat itu mulai membuat kelompok-kelompok bit yang terdiri dari *nibble* (4 bit), *byte* (8 bit), *word* (2 byte), dan *double word* (32 bit).

Hadirnya kelompok-kelompok bit ini terbilang cukup membantu dalam berbagai kegiatan yang dilakukan di perangkat lunak komputer. Dengan berbagai kelompok bit yang ada pada saat itu, orang-orang mulai merakit kode-kode tersebut menjadi berbagai struktur instruksi seperti transfer, operasi logika, dan penyimpanan, hingga pada akhirnya terbentuklah kode-kode baru yang kita kenal dengan nama *assembler*. Kode-kode *assembler* ini lah yang nantinya menjadi cikal-bakal dibuatnya berbagai macam aplikasi yang saat ini dapat mempermudah berbagai kegiatan manusia.

## **1.2 Pengertian Aplikasi**

Aplikasi adalah suatu perangkat lunak atau dalam Bahasa Inggris adalah *Software* yang beroperasi pada computer yang berfungsi untuk membantu berbagai kegiatan yang dilakukan oleh seseorang atau *user*. Aplikasi dikelompokkan menjadi 3 kelompok, yaitu *Dekstop Application*, *Website Application* , dan *Mobile Application*.

Berikut ini adalah pengertian aplikasi menurut para ahli:

### **1. Ali Zaki dan Smitdev Community**

Menurut Ali Zaki dan *Smitdev Community*, Aplikasi merupakan komponen yang bermanfaat sebagai media untuk menjalankan pengolahan data ataupun berbagai kegiatan lainnya seperti pembuatan ataupun pengolahan dokumen dan file.

### **2. Sri Widiанти**

Menurut Sri Widiанти, Aplikasi merupakan sebuah *software* (perangkat lunak) yang bertugas sebagai *front end* pada sebuah sistem yang dipakai untuk mengelolah berbagai macam data sehingga menjadi sebuah informasi yang bermanfaat untuk penggunaanya dan juga sistem yang berkaitan.

### **3. Harip Santoso**

Menurut Harip Santoso, Aplikasi merupakan sebuah kelompok file (*class, form, report*) yang ditujukan sebagai pengeksesuksi aktivitas tertentu yang saling berkaitan seperti contohnya aplikasi *payroll* dan aplikasi *fixed asset*.

#### 4. **Yuhefizar**

Menurut Yuhefizar, Aplikasi adalah program yang sengaja dibuat dan dikembangkan sebagai pemenuh kebutuhan penggunanya dalam menjalankan suatu pekerjaan tertentu.

#### 5. **Hengky W. Pramana**

Menurut Hengky W. Pramana, **pengertian aplikasi** adalah satu unit perangkat lunak yang sengaja dibuat untuk memenuhi kebutuhan akan berbagai aktivitas ataupun pekerjaan, seperti aktivitas perniagaan, periklanan, pelayanan masyarakat, game, dan berbagai aktivitas lainnya yang dilakukan oleh manusia.

### 1.3 **Jenis-jenis Aplikasi**

Berikut ini adalah beberapa jenis dari aplikasi yang biasa digunakan beberapa orang.

#### 1.3.1 ***Multimedia Software***

*Multimedia software* adalah aplikasi yang dibuat untuk mengelola suatu objek yang dapat berupa gambar, teks, suara, dan grafik. Banyak sekali aplikasi multimedia yang telah dibuat saat ini, contohnya adalah *Windows Media Player*, *Adobe Photoshop*, dan lain-lain.

#### 1.3.2 ***Database Software***

*Database software* adalah perangkat lunak yang berfungsi untuk mengelola kumpulan data dalam jumlah besar. Macam-macam Aplikasi *Database Management System (DBMS)*, yaitu *MySQL*, *Oracle*, *Microsoft SQL Server*, *IBM DB2*, dan masih banyak lagi beberapa Aplikasi *DBMS*.

### **1.3.3 Presentation Software**

*Presentation Software* adalah suatu aplikasi yang menampilkan informasi dalam tayangan *slide show*, yang dapat dihias, dan dapat memasukkan gambar grafis. Contoh dari Aplikasi *Presentation Software* selain *Microsoft Power Point* adalah *Slidedog*, *Visme*, *Google Slide*, dan lain-lain.

### **1.3.4 Word Processing Software**

*Word Processing Software* adalah perangkat lunak atau *software* yang berfungsi untuk mengelolah kata atau *word* yang berhubungan dengan dokumen, seperti laporan, karya tulis, surat menyurat, artikel, dan brosur. Contoh aplikasinya adalah *Microsoft Word*, *Notepad*, dan *Google Docs*.

### **1.3.5 Spreadsheet Softwar**

*Spreadsheet Software* adalah aplikasi yang menampilkan data dalam bentuk baris dan kolom. Biasanya aplikasi ini digunakan untuk membuat data keuangan, data seseorang, dll. Contoh dari *Spreadsheet Software* adalah *Microsoft Excel*, *StarCalc*, *Siag*, dan sebagainya.

## **1.4 Fungsi Aplikasi**

Aplikasi memiliki fungsi yang digunakan. Pada bidang Pendidikan aplikasi memiliki fungsi sebagai bahan untuk mengajar. Contohnya pada saat siswa presentasi dapat menggunakan Aplikasi *Microsoft Power Point*. Tujuannya adalah untuk membuat presentasi tersebut lebih menarik dengan menggunakan Aplikasi *Microsoft Power Point*.

Tidak hanya di bidang pendidikan saja, aplikasi juga berfungsi pada bidang kedokteran, bisnis, dan militer. Pada bidang tersebut aplikasi digunakan untuk menginput data manusia, keuangan, dan masih banyak lagi.

## **BAB II**

### **BAHASA PEMROGRAMAN**

#### **2.1 Latar Belakang**

Sejak dulu hingga sekarang, tentu kita mengetahui bahwa di dunia komputer terdapat beraneka ragam bahasa pemrograman. Karena begitu banyaknya jenis-jenis bahasa pemrograman, bahasabahasa tersebut juga dikelompokkan berdasarkan kriteria tertentu. Ada yang mengelompokkannya menjadi 3 level bahasa yaitu: high level (Seperti Pascal dan Basic), middle level (Seperti Bahasa C), dan low level (Seperti Bahasa Assembly). Ada juga yang mengelompokkannya menjadi procedural/functional programming, object oriented programming, dan sebagainya.

Namun pada dasarnya, bahasa-bahasa pemrograman tersebut memiliki bagian-bagian yang serupa. Yang membedakan hanyalah tata bahasa yang digunakan. Seperti halnya belajar bahasa Indonesia, bahasa Jawa, ataupun bahasa asing. Masing-masingnya memiliki bagian-bagian yang sama, yang membedakan hanyalah istilah/symbol yang digunakan dan dialek bahasa tersebut.

.Bahasa ini memungkinkan seorang *programmer* dapat menentukan mana yang data yang akan diproses oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan langkah-langkah apa yang persis jenis yang akan diambil dalam berbagai situasi.

Menurut tingkat kedekatannya dengan mesin komputer, bahasa pemrograman terdiri dari :

1. Bahasa mesin, yang memberikan perintah ke komputer dengan menggunakan kode bahasa biner, misalnya 01100101100110.
2. Bahasa tingkat rendah, atau dikenal sebagai bahasa *assembly* yang memberikan perintah ke komputer dengan menggunakan kode pendek (*kode mnemonic*), misalnya [kode\_mesin | MOV], SUB, CMP, JMP, JGE, JL , LOOP, dll. Inggris *Intermediate*, yang merupakan bahasa komputer yang menggunakan campuran instruksi dalam kata-kata bahasa manusia lihat contoh di bawah Bahasa Tingkat Tinggi dan instruksi yang bersifat simbolik, misalnya, {,},?, <<, >> , &&, ||, dll.
3. Bahasa tingkat tinggi, yang merupakan bahasa komputer yang menggunakan instruksi berasal dari unsur kata-kata bahasa manusia, misalnya, mulai, akhir, jika, sementara, dan, atau, dll Komputer dapat memahami *compiler* bahasa manusia atau penerjemah program yang dibutuhkan.

## 2.2 Sejarah Bahasa Pemrograman

Penemuan bahasa pemrograman di mulai bersamaan dengan kemunculan komputer itu sendiri. Bahasa pemrograman dan komputer merupakan dwitunggal yang tidak terpisahkan karena sebuah komputer hanya bisa berfungsi jika ada program.

Sementara itu, sebuah program yang dibuat dengan bahasa pemrograman hanya bisa hidup dengan menggunakan komputer sebagai media. Bahasa pemrograman telah mengalami berbagai proses perkembangan. Kondisi bahasa pemrograman saat ini telah

berbeda jauh dengan bahasa pemrograman di tahun 40-an. Dahulu *progammer* memerintah komputer secara fisik menggunakan metode pengkabelan / *wiring* secara manual.

Metode *wiring* kemudian berkembang hingga menjadi bahasa pemrograman sederhana. Kualitas dan fitur berkembang dan bertambah dengan pesat. Bahasa pemrograman di awal kemunculannya hanya dapat digunakan untuk satu keperluan saja/ *one purpose only*, tetapi kini dapat digunakan untuk bermacam-macam keperluan.

Tokoh pertama yang dianggap memperkenalkan bahasa pemrograman pada komputer adalah Charles Babbage. Charles Babbage adalah seorang ilmuwan yang hidup di Inggris pada abad pertengahan. Beliau adalah orang yang pertama kali mendesain mesin menyerupai komputer. Mesin tersebut diisi beberapa program yang dapat berjalan di dalamnya (identik seperti bahasa *assembly* sekarang ini). Pembuat program tersebut adalah Byron Countess of Lovelace, putri dari Lord Byron.

Komputer modern yang menggunakan tenaga listrik pertama kali diciptakan pada tahun 1940-an. Ketika itu, komputer masih menjadi barang mewah dan hanya dimiliki oleh pihak-pihak yang memerlukan perhitungan data yang sangat banyak, seperti pihak militer dan universitas. Komputer pada waktu itu berukuran sangat besar sehingga memakan banyak tempat. Meskipun besar, komputer dianggap lambat menjalankan proses komputasi. Di awal



penciptaannya hanya sedikit tokoh yang menyarankan untuk membuat program pada komputer.

Dunia pemrograman kemudian mengalami perkembangan yang sangat pesat setelah ditemukan komponen-komponen elektronika. Penggunaan komponen elektromika menyebabkan komputer menjadi komputer berukuran kecil karena ditemukannya teknologi *Integrated Circuit (IC)* dan teknologi pembuatan *chip* lainnya. Selain ukuran yang mengecil, kelebihan komputer setelah revolusi elektronika terdapat pada kemampuan komputasinya yang meningkat tajam. Oleh karena itu kemampuannya meningkat, secara otomatis komputer mampu menangani pemrosesan algoritma dengan lebih cepat. Hal tersebut mendorong di buatnya berbagai macam bahasa pemrograman tingkat tinggi.

### **2.3 Pengertian Bahasa Pemrograman**

**Bahasa** adalah suatu sistem untuk berkomunikasi. Bahasa tertulis menggunakan simbol (yaitu huruf) untuk membentuk kata. Dalam ilmu komputer, bahasa manusia disebut bahasa alamiah, dimana komputer tidak bisa memahaminya, sehingga diperlukan suatu bahasa komputer.

**Program** merupakan sekumpulan instruksi yang merupakan penyelesaian masalah. Program ‘dimasukkan’ ke dalam komputer, komputer mengerjakan instruksi-instruksi di dalam program tersebut, lalu memberikan hasil atau keluaran yang diinginkan. Agar program dapat dilaksanakan oleh komputer, program tersebut harus ditulis dalam suatu bahasa yang dapat dimengerti komputer. Karena

komputer adalah mesin, maka program harus ditulis dalam bahasa yang khusus dibuat untuk berkomunikasi dengan komputer. Bahasa komputer yang digunakan dalam menulis program dinamakan bahasa pemrograman.

**Bahasa pemrograman** adalah instruksi standar untuk memerintah komputer yang memiliki fungsi tertentu. Bahasa pemrograman ini adalah satu set aturan sintaks dan semantik yang digunakan untuk mendefinisikan program komputer.

Bahasa ini memungkinkan seorang programmer dapat menentukan mana yang data yang akan diproses oleh komputer, bagaimana data ini akan disimpan / diteruskan, dan langkah-langkah apa yang persis jenis yang akan diambil dalam berbagai situasi.

## **2.4 Fungsi Bahasa Pemrograman**

Fungsi dari bahasa pemrograman yang memerintahkan komputer untuk mengolah data sesuai dengan logika yang kita inginkan. *Output* dari bahasa pemrograman dalam bentuk program / aplikasi. Contohnya adalah program yang digunakan oleh kasir di mall atau supermarket, penggunaan lampu lalu lintas di jalan raya, dll

Bahasa pemrograman yang kita tahu ada banyak orang di dunia, sekitar ilmu komputer dan teknologi saat ini. Pengembangannya mengikuti inovasi tinggi dalam dunia teknologi. Contoh bahasa pemrograman yang kita ketahui, antara lain, adalah untuk membuat aplikasi permainan, anti-virus, web, dan teknologi lainnya.

Bahasa pemrograman komputer yang kita tahu termasuk *Java*, *Visual Basic*, *C++*, *C*, *Cobol*, *PHP*, *Net*, dan ratusan bahasa lain. Tapi tentu saja, bahasa harus disesuaikan dengan fungsi dan perangkat yang menggunakannya.

## **2.5 Tingkat Bahasa Pemrograman**

Bahasa pemrograman juga memiliki beberapa tingkatan, antara lain sebagai berikut.

### **2.5.1 Bahasa Tingkat Tinggi**

Bahasa pemrograman ini masuk ke level sebagai bahasa dekat dengan bahasa manusia. Sebagai contoh, bahasa Basic, Visual Basic, Pascal, Java dan lainnya.dsb dan tidak lagi pada pemula sulit dimengerti.

### **2.5.2 Bahasa Tingkat Menengah**

Disebut tingkat menengah karena dapat masuk ke dalam bahasa tingkat rendah tinggi. Misalnya dalam bahasa program C.

### **2.5.3 Bahasa Tingkat Rendah**

Bahasa pemrograman masuk ke level ini karena bahasa masih jauh dari bahasa manusia. Contoh bahasa *Assembly*.

## **2.6 Macam Bahasa Pemrograman**

Bahasa pemrograman memiliki banyak macam. Pada buku ini hanya akan menjelaskan beberapa macam saja dari Bahasa pemrograman.

### 2.6.1 *HyperText Markup Language (HTML)*



*HyperText Markup Language (HTML)* adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser Internet. *HTML* saat ini merupakan standar internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium (W3C)*.

*HTML* berupa kode-kode tag yang menginstruksikan browser untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file *HTML* dapat dibuka dengan menggunakan browser web seperti *Mozilla Firefox* atau *Microsoft Internet Explorer*.

### 2.6.2 *PHP: Hypertext Preprocessor*



*PHP* adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini. *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama *FI* (*Form Interpreted*), yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari web.

*PHP* banyak dipakai untuk membuat situs web yang dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. *PHP* biasanya berjalan pada sistem operasi *linux* (*PHP* juga bisa dijalankan dengan hosting windows).

### 2.6.3 Javascript



*Javascript* adalah bahasa *scripting* yang handal yang berjalan pada sisi *client*. *JavaScript* merupakan sebuah bahasa *scripting* yang dikembangkan oleh *Netscape*. Untuk menjalankan *script* yang ditulis dengan *JavaScript* kita membutuhkan *JavaScript-enabled browser* yaitu browser yang mampu menjalankan *JavaScript*.

#### 2.6.4 CSS



*Cascading Style Sheets (CSS)* adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan *HTML*. Spesifikasi CSS diatur oleh *World Wide Web Consortium (W3C)*.

#### 2.6.5 Java

*Java* adalah sebuah teknologi yang diperkenalkan oleh Sun Microsystems pada pertengahan tahun 1990. Menurut definisi dari Sun, *Java* adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer standalone ataupun pada lingkungan jaringan. Kita lebih menyukai menyebut *Java* sebagai sebuah teknologi dibanding hanya sebuah bahasa pemrograman, karena *Java* lebih lengkap dibanding sebuah bahasa pemrograman konvensional.



Bahasa *Java* dapat dikategorikan sebagai sebuah bahasa pemrograman berorientasi objek, pemrograman terdistribusi dan bahasa pemrograman multithreaded. Objek *Java* dispesifikasi dengan membentuk kelas. Untuk masing-masing kelas *Java*, kompiler *Java* memproduksi sebuah file keluaran arsitektur netral yang akan jalan pada berbagai implementasi dari *Java Virtual Machine (JVM)*. Awalnya *Java* sangat digemari oleh komunitas pemrograman internet, karena *Java* mendukung untuk applets, dimana program dengan akses sumber daya terbatas yang jalan dalam sebuah web browser.

*Java* juga menyediakan dukungan level tinggi untuk *networking* dan objek terdistribusi. *Java* juga dianggap sebagai sebuah bahasa yang aman. Tampilan ini pada khususnya penting menganggap bahwa sebuah program *Java* boleh mengeksekusi silang sebuah jaringan terdistribusi. Sehingga bahasa *Java* saat ini termasuk bahasa pemrograman yang relatif mudah untuk dipelajari karena banyaknya contoh dan konsep yang beredar luas, baik berupa buku maupun di internet. Tetapi program yang dibuat dalam bahasa *Java* juga relatif lebih banyak membutuhkan waktu saat di eksekusi

(lebih lama) dikarenakan untuk menjalankan programnya dibutuhkan *JVM* (perantara antar program dan sistem operasi).

### 2.6.6 Bahasa C



Akar dari bahasa *C* adalah bahasa *BCPL* yang dikembangkan oleh Martin Richard pada tahun 1967. Bahasa ini memberikan ide kepada Ken Thompson yang kemudian mengembangkan bahasa yang disebut dengan *B* pada tahun 1970. Perkembangan selanjutnya dari bahasa *B* adalah bahasa *C* oleh Dennis Ritchie sekitar tahun 1972-an di *Bell Telephone Laboratories Inc.* (sekarang adalah *AT&T Bell Laboratories*).

Kelebihan Bahasa *C* adalah Bahasa *C* tersedia hampir di semua jenis komputer, kode Bahasa *C* sifatnya adalah *portable*, aplikasi yang ditulis dengan Bahasa *C* untuk suatu komputer tertentu dapat digunakan di komputer lain hanya dengan sedikit modifikasi, dan Bahasa *C* hanya menyediakan sedikit kata-kata kunci.

Bahasa *C* mempunyai struktur yang baik sehingga mudah untuk dipahami. Bahasa *C* mempunyai fungsi-fungsi sebagai program bagiannya. Selain bahasa tingkat tinggi, *C* juga dianggap



sebagai bahasa tingkat menengah. Bahasa *C* mampu menggabungkan kemampuan bahasa tingkat tinggi dengan bahasa tingkat rendah. Karena *C* sifatnya adalah kompiler, maka akan menghasilkan *executable program* yang banyak dibutuhkan oleh program-program komersial.

Kekurangan Bahasa *C* adalah bahasa pemrograman yang memiliki portabilitas tinggi. Program *C* yang kita tulis untuk satu jenis *platform*, bisa kita kompilasi dan jalankan di *platform* lain dengan tanpa ataupun hanya sedikit perubahan. Ini bisa diwujudkan dengan adanya standarisasi ANSI untuk *C*.

*C* adalah bahasa pemrograman dengan kata kunci (*keyword*) sedikit. Kata kunci disini adalah merupakan fungsi ataupun kata dasar yang disediakan oleh kompiler suatu bahasa pemrograman. Hal ini membawa pengaruh semakin mudahnya kita menulis program dengan bahasa tingkat rendah. Pengaruh lain dari sedikitnya kata kunci ini adalah proses eksekusi program *C* yang sangat cepat.

### 2.6.7 Python



*Python* merupakan bahasa pemrograman tingkat tinggi yang diracik oleh Guido van Rossum. *Python* banyak digunakan untuk membuat berbagai macam program, seperti: program *CLI*, *Program GUI (desktop)*, *Aplikasi Mobile*, *Web*, *IoT*, *Game*, Program untuk *Hacking*, dsb.

*Python* juga dikenal dengan bahasa pemrograman yang mudah dipelajari, karena struktur sintaknya rapi dan mudah dipahami. *Python* memang sangat sederhana dibandingkan bahasa yang lainnya. Tidak perlu ini dan itu untuk membuat program *Hello World!*. Bahkan *tagline* di *website*-nya menjelaskan, kalau *python* akan membuatmu bekerja lebih cepat dan efektif.

## **BAB III**

### **SOFTWARE UNTUK MEMBUAT APLIKASI**

#### **3.1     *Software untuk membuat Aplikasi Webstite***

Untuk membuat sebuah aplikasi dibutuhkan *software* pendukung agar aplikasi yang dibuat mendapatkan hasil yang maksimal. Membuat sebuah Aplikasi *Website* dibutuhkan *text editor software*. Berikut ini adalah beberapa contoh *text editor software*.

##### **3.1.1   *Notepad++***

*Notepad++* adalah sebuah penyunting teks dan penyunting kode sumber yang berjalan di sistem operasi *Windows*. *Notepad++* menggunakan komponen *Scintilla* untuk dapat menampilkan dan menyuntingan teks dan berkas kode sumber berbagai bahasa pemrograman.



*Notepad++* didistribusikan sebagai perangkat lunak bebas. Proyek ini dilayani oleh *Sourceforge.net* dengan telah diunduh lebih dari 27 juta kali dan dua kali memenangkan

penghargaan *SourceForge Community Choice Award for Best Developer Tool*.

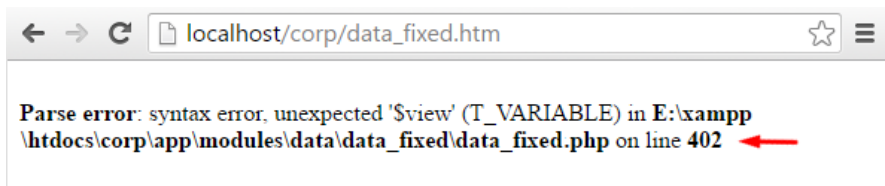
Didukung dalam hal ini adalah dimengerti dan diterjemahkan menjadi teks oleh *Notepad++*. Misalnya pada *C++*, fungsi-fungsinya akan di masukan kedalam daftar fungsi dan kata-katanya akan berubah warna sesuai dengan makna kata tersebut *di C++*. Berikut ini adalah beberapa Bahasa pemrograman yang didukung oleh *Notepad++* sejak versi 5.3.9.

1. *ActionScript, Ada, ASP, Assembler, autoIt*
2. *C, C++, C#, Caml, Cmake, COBOL, CSS*
3. *Flash ActionScript, Fortran*
4. *Haskell, HTML*
5. *Java, Javascript, JSP*
6. *KiXtart*
7. *LISP, Lua*
8. *Makefile, Matlab, MS-DOS*
9. *NSIS*
10. *Objective-C*
11. *Pascal, Perl, PHP, Postscript, PowerShell, Properties file, Python*
12. *R, Resource file, Ruby*
13. *Shell, Scheme, Smalltalk, SQL*
14. *TCL, TeX*
15. *Pascal, Perl, PHP, Postscript, PowerShell, Properties file, Python*
16. *XML*

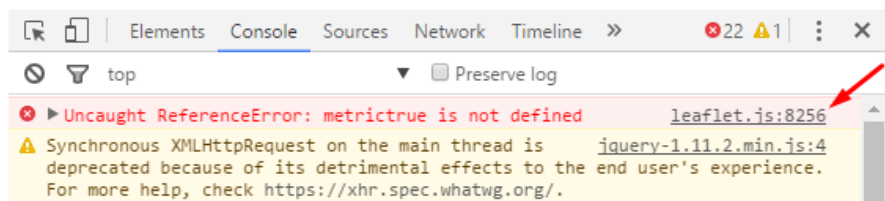
## 17. YAML

### 1. Go To Line

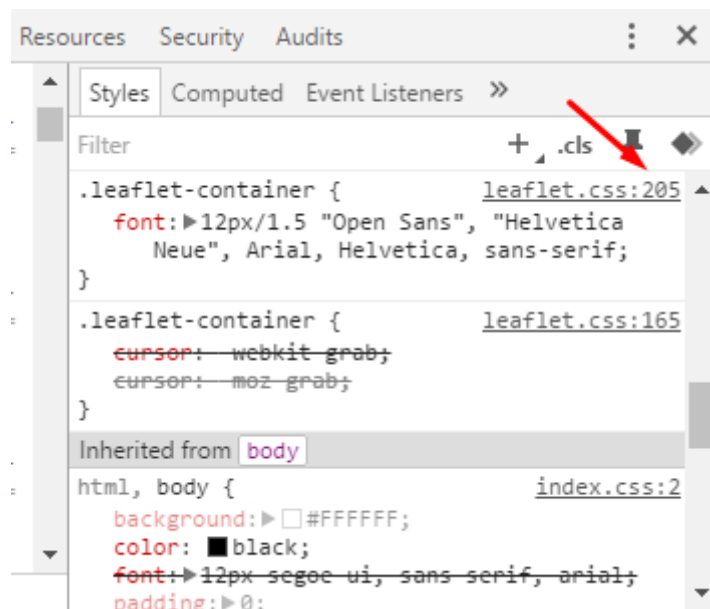
Bagi *programer*, pada kondisi tertentu nomor baris suatu kode akan menjadi begitu penting, terutama ketika melakukan *debugging* program seperti itu *PHP* maupun *JavaScript*, karena, jika terjadi *error*, biasanya program tersebut akan memberi tahu baris berapa yang bermasalah. seperti ini:



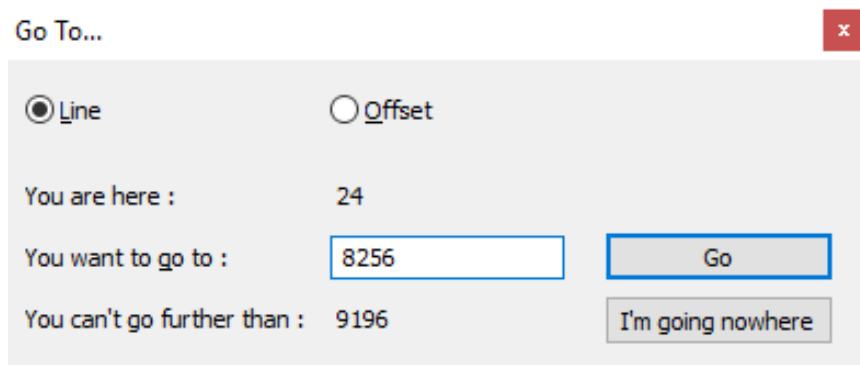
atau ini:



Tidak hanya itu saja, ketika kita meng-*inspect* suatu *element html* dan ingin merubah *style element* tersebut, kita dapat menuju ke nomor baris yang diinginkan.



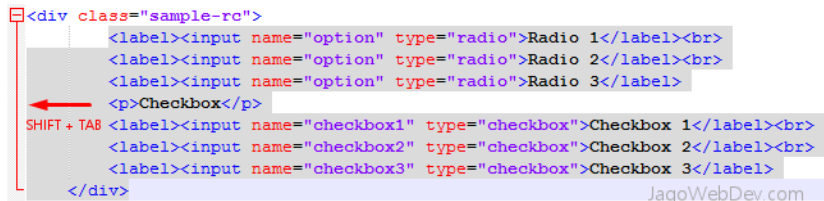
Penting bukan? Untuk itu memanfaatkan fitur ini, caranya dengan menekan tombol *Ctrl + G* dan isikan nomor baris yang ingin dituju kemudian klik *OK*.



## 2. Reverse Indent / Back Tab

Agar kode tampak rapi tentu kita akan sering membuat *indent* -menggeser kode ke kanan, yang biasanya dilakukan menggunakan tombol *Tab*,

bagaimana jika ingin mengembalikannya -menggeser indent ke arah kiri-?, caranya mudah, gunakan *shortcut Shift+Tab*



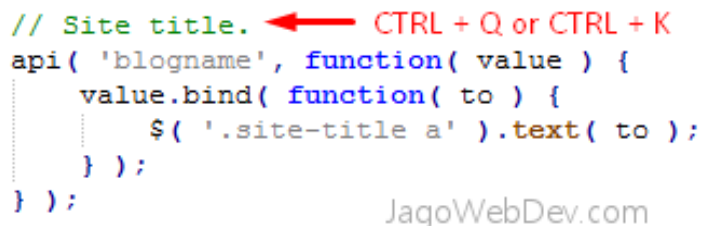
```
<div class="sample-rc">
  <label><input name="option" type="radio">Radio 1</label><br>
  <label><input name="option" type="radio">Radio 2</label><br>
  <label><input name="option" type="radio">Radio 3</label>
  <p>Checkbox</p>
  <label><input name="checkbox1" type="checkbox">Checkbox 1</label><br>
  <label><input name="checkbox2" type="checkbox">Checkbox 2</label><br>
  <label><input name="checkbox3" type="checkbox">Checkbox 3</label>
</div>
```

JagoWebDev.com

### 3. Single Line Comment / Uncomment

Untuk menjadikan suatu baris menjadi komentar, biasanya kita tambahkan diawal baris tersebut tanda *doubel slash* (*//*), secara umum hal tersebut dilakukan dengan mengarahkan cursor ke awal baris kemudian menekan tombol *slash* dua kali, namun hal ini dapat dipercepat dengan menekan tombol *Ctrl+K* atau *Ctrl+Q*.

Saya sendiri lebih nyaman menggunakan *Ctrl+Q* karena dapat dikerjakan dengan satu tangan kiri, sehingga lebih mudah. Untuk mengembalikannya, tekan kembali tombol *Ctrl+K* atau *Ctrl+Q*.



```
// Site title.
api( 'blogname', function( value ) {
  value.bind( function( to ) {
    $( '.site-title a' ).text( to );
  } );
} );
```

JagoWebDev.com

### 4. Block Comment / Uncomment

Block comment biasanya dilakukan dengan menambahkan tanda */\** di awal block dan ditutup dengan *\*/*, pada Dokumen HTML, tanda tersebut berupa *<!--* dan *-->*, dengan notepad++ pekerjaan tersebut dapat dipersingkat, caranya seleksi baris yang akan di comment kemudian tekan tombol **Ctrl+Shift+Q**.

Notepad++ akan menyesuaikan tanda yang digunakan sesuai dengan tipe file yang ada.

```
/* api( 'blogname', function( value ) {  
    value.bind( function( to ) {  
        $( '.site-title a' ).text( to );  
    } );  
} );  
  
api( 'blogdescription', function( value ) {  
    value.bind( function( to ) {  
        $( '.site-description' ).text( to );  
    } );  
} ); */
```

JagoWebDev.com

Untuk mengembalikannya kita tidak dapat menekan kembali shortcut tersebut seperti yang kita lakukan pada single line comment, melainkan kita harus mendefinisikan nya sendiri, caranya buka menu **Run > Modify Shortcut/Delete Command...**

Selanjutnya pada tab Main Menu, cari baris nomor 46. Block Uncomment, selanjutnya isikan shortcut yang diinginkan, saya sendiri memilik **Ctrl+Shift+W** agar mudah dioperasikan dengan satu tangan.

Shortcut mapper ✕

	Main menu	Macros	Run commands	Plugin commands	Scintilla commands
	Name	Shortcut			
42	Toggle Single Line Comment	Ctrl+Q			
43	Single Line Comment	Ctrl+K			
44	Single Line Uncomment	Ctrl+Shift+K			
45	Block Comment	Ctrl+Shift+Q			
46	Block Uncomment	Ctrl+Shift+W			
47	Function Completion	Ctrl+Spacebar			
48	Path Completion	Ctrl+Alt+Spacebar			

## 5. Seleksi Berdasarkan Kolom

Selain menyeleksi dan mengedit baris tertentu, dengan notepad++ kita dapat melakukannya untuk beberapa baris sekaligus, caranya **tekan dan tahan tombol alt** kemudian seleksi baris yang



ingin diedit. Fitur ini bermanfaat salah satunya ketika kita mengedit file csv atau mengubah tabel menjadi file csv.

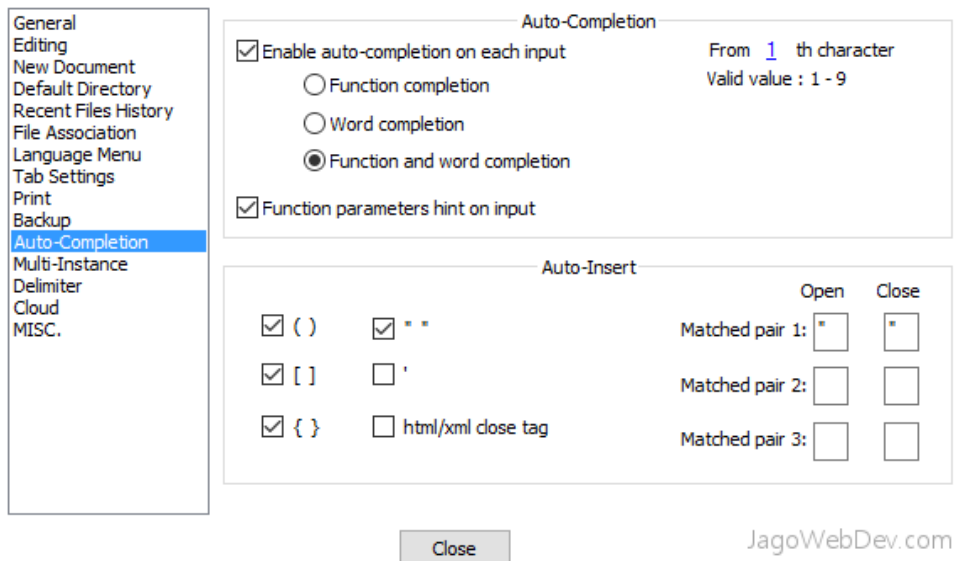
1	01/01/2016;11;10
2	01/01/2016;12;5
3	11/01/2016;11;2
4	10/02/2016;11;22
5	10/02/2016;12;12
6	01/01/2016;21;10
7	01/01/2016;22;5
8	11/01/2016;22;2
9	10/02/2016;21;22
10	10/02/2016;22;12
11	01/01/2016;31;10
12	01/01/2016;33;5
13	11/01/2016;33;2
14	10/02/2016;31;22
15	10/02/2016;33;12

## TIPS

Untuk menseleksi teks yang panjang, (1) tekan dan tahan tombol ALT kemudian klik pada teks yang ingin dijadikan awal seleksi, (2) gunakan mouse untuk menscroll jendela ke bawah, (3) tekan tombol SHIFT (tombol ALT tetap ditekan) kemudian klik pada teks yang akan dijadikan akhir seleksi.

## 6. Auto Completion

Fitur ini bermanfaat untuk memberikan tanda penutup untuk karakter tertentu, secara default Notepad++ akan memberikan penutup untuk tanda kurung (), kurung siku [], dan kurung kurawal {}, disamping itu sobat juga dapat menambahkan tanda lain misal dua tanda kutip “” atau satu tanda kutip ”, caranya buka menu **Setting > Preferences**, kemudian pilih **Tab Auto Completion**



## 7. Menambahkan dan Menghapus Nomor Urut

Misalkan kita memiliki kode atau daftar list dan ingin menambahkan nomor urut di depannya, untuk keperluan tersebut, buka jendela “Column / Multi-Selection Editor” melalui menu **Edit > Column Editor...** atau dengan shortcut **Alt + C**, kemudian isikan parameter yang ada.

## Column / Multi-Selection Editor



☐ Text to Insert

OK

Cancel

☒ Number to Insert

Initial number :

Increase by :

Repeat :  ☒ Leading zeros

Format

☒ Dec

☐ Hex

☐ Oct

☐ Bin

## Menghapus nomor urut

Ketika mengcopy – paste dari media lain baik berupa kode maupun artikel, terkadang terdapat nomor urut di depannya, Notepad++ telah menyediakan fitur untuk menghilangkannya melalui **TextFX > TextFX Tools > Delete Line Numbers or First Word**

```

1      1 // Site title.
2      2 api( 'blogname', function( value ) {
3      3         value.bind( function( to ) {
4      4             $( '.site-title a' ).text( to );
5      5         } );
6      6     } );
7
8      8 // Site tagline.
9      9 api( 'blogdescription', function( value ) {
10     10 value.bind( function( to ) {
11     11     $( '.site-description' ).text( to );
12     12     } );
13     13 } );

```



```

1      // Site title.
2      api( 'blogname', function( value ) {
3      value.bind( function( to ) {
4      $( '.site-title a' ).text( to );
5      } );
6      } );
7
8      // Site tagline.
9      api( 'blogdescription', function( value ) {
10     value.bind( function( to ) {
11     $( '.site-description' ).text( to );
12     } );
13     } );

```

## 8. Menghapus Baris Kosong

Seperti sebelumnya, ketika mengcopy -paste teks dari media lain, terkadang terdapat baris kosong diantara teks tersebut, disamping itu juga terkadang terdapat tambahan spasi di akhir tiap tiap baris, untuk menghilangkannya gunakan menu **TextFX > TextFX Edit > Trim Trailing Spaces** dan **TextFX > TextFX Edit > Delete Blank Lines**

```

1 Auto-completion for a subset of the API of some programming language
2
3 Bookmarks
4
5 Syntax highlighting and syntax folding
6
7 Indent guidelines for tab-indented files
8
9 Matching brace highlighting (for curly braces and square brackets)
10 Matching tag highlighting (for HTML and XML tags)

```

TextFX > TextFX Edit > Trim Trailing Spaces  TextFX > TextFX Edit > Delete Blank Lines

```

1 Auto-completion for a subset of the API of some programming language
2 Bookmarks
3 Syntax highlighting and syntax folding
4 Indent guidelines for tab-indented files
5 Matching brace highlighting (for curly braces and square brackets)
6 Matching tag highlighting (for HTML and XML tags)

```

JagoWebDev.com

## 9. Mengubah posisi baris

Untuk menggeser baris ke atas, gunakan kombinasi **Ctrl + Shift + Up** sedangkan untuk menggeser ke bawah gunakan kombinasi **Ctrl + Shift + Down**

```

1 1. Auto-completion for a subset of the API of some programming language
2 2. Bookmarks
3 3. Syntax highlighting and syntax folding
4 7. Regular expression find and replace (in perl compatible extent)
5 4. Indent guidelines for tab-indented files
6 5. Matching brace highlighting (for curly braces and square brackets)
7 6. Matching tag highlighting (for HTML and XML tags)
8 CTRL + SHIFT + UP / CTRL + SHIFT + DOWN
9

```

JagoWebDev.com

## 10. Line up lines

Ketika menulis atau meng copy-paste banyak variabel dengan panjang nama variabel yang berbeda beda, kemungkinan besar posisi tanda sama dengan tersebut tidak rata / tidak rapi, untuk merapikannya gunakan menu **TextFX > TextFX Edit > Line up multiple lines by (=)**

```

53 // Sidebar scrolling.
54 function resize() {
55     windowWidth = $window.width();
56     windowHeight = $window.height();
57     bodyWidth = $body.width();
58     bodyHeight = $body.height();
59     sidebarWidth = $sidebar.width();
60     sidebarHeight = $sidebar.height();
61
62     if ( 955 > windowWidth ) {
63         top = bottom = false;
64         $sidebar.removeAttr( 'style' );
65     }
66 }

```



```

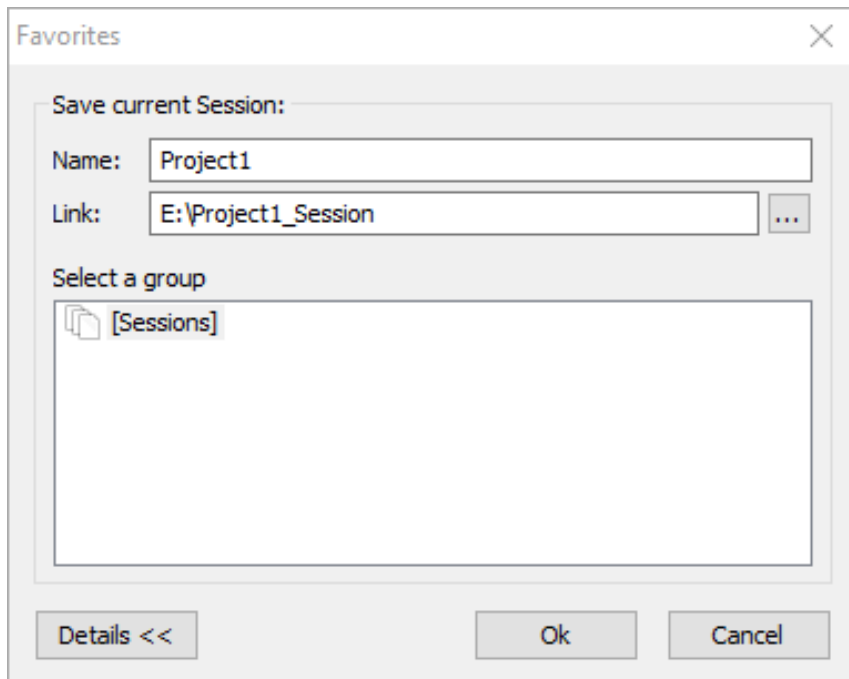
53 // Sidebar scrolling.
54 function resize() {
55     windowWidth  = $window.width();
56     windowHeight = $window.height();
57     bodyWidth    = $body.width();
58     bodyHeight   = $body.height();
59     sidebarWidth  = $sidebar.width();
60     sidebarHeight = $sidebar.height();
61
62     if ( 955 > windowWidth ) {
63         top = bottom = false;
64         $sidebar.removeAttr( 'style' );
65     }
66 }

```

Jika ingin pilihan opsi yang lebih lengkap, dapat menggunakan Plugin Code Alignment

## 11. Membuat Session

Ketika bekerja dengan beberapa file sekaligus misal file html + JavaScript + css, maka akan merepotkan jika kita harus membuka dan menutupnya secara manual, Notepad++ menyediakan fitur bernama Save Session dan Load Session, yang memungkinkan kita untuk menyimpan dan membuka beberapa file sekaligus sesuai yang telah kita tentukan. Fitur tersebut dapat diakses melalui menu File > Load Session atau File > Save Session

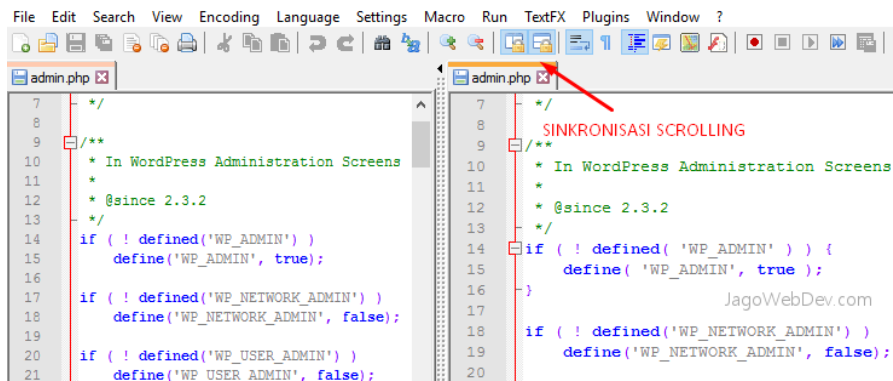


## 12. Membandingkan File – Sinkronisasi Scrolling

Seperti telah kita bahas sebelumnya, notepad++ memiliki plugin compare yang akan membandingkan dua file dan memberikan warna tertentu pada baris yang berbeda, sebenarnya Notepad++ memiliki fitur bawaan yang fungsinya sama persis dengan plugin tersebut.

Bagaimana caranya?

Caranya, klik salah tab tertentu dan pilih Move to Other View, kemudian pada menu View, pilih Synchronize Vertical Scrolling dan Synchronize Horizontal Scrolling.



Untuk mengembalikan ke posisi semula klik dan drag tab kembali ke posisi semula.

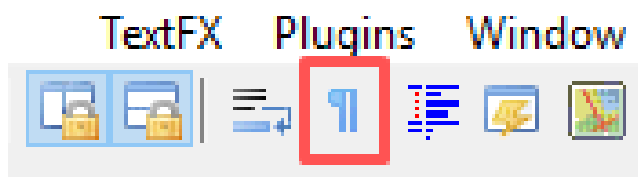
### 13. Menampilkan Tab, Spasi, dan Enter Pada Notepad++

Pernahkah sobat bingung membedakan antara spasi dan tab? atau mungkin dengan enter juga? jika ya maka “Show All Characters” adalah solusinya, fitur ini akan menampilkan karakter simbol, diantaranya **TAB** (Tanda Panah), **SPASI** (Tanda Titik), dan **ENTER** (LF-Line Feed) dan (CR-Carriage Return).

**CRLF** dan **LF** berlaku untuk sistem operasi windows, sedangkan **CR** untuk sistem operasi Linux.



Untuk mengaktifkan fitur ini, buka menu **View > Show Symbol > Show All Character** atau melalui icon pada toolbar.





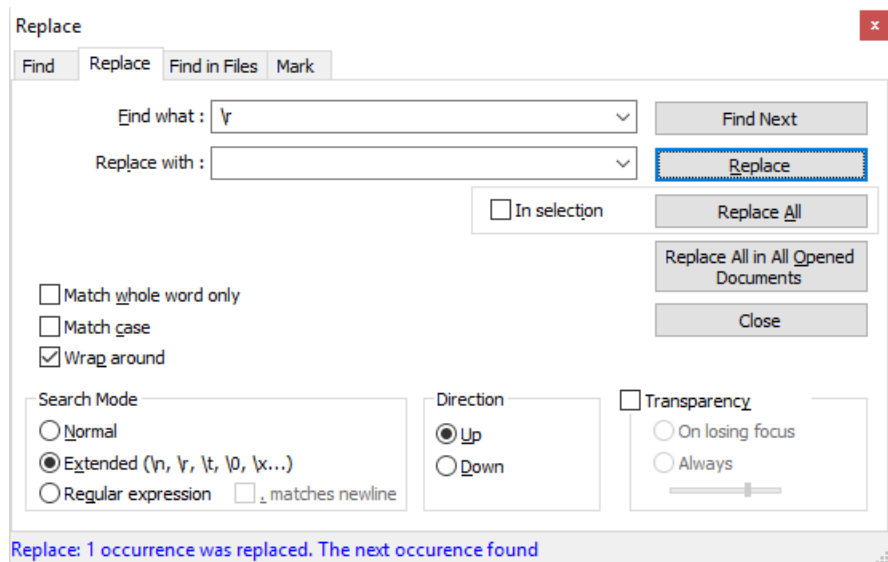
#### 14. Menghapus Tab dan Enter Melalui Extended Search

Selain dapat melakukan Find & Replace karakter umum, notepad++ juga dapat melakukan Find & Replace karakter khusus seperti tab (\t) dan enter (\n), dengan fitur ini kita dapat merapikan berbagai file seperti csv, php, js, html, dll atau mengecilkan nya dengan menghilangkan spasi, tab dan enter.

Untuk menggunakan fitur ini, buka menu “Find & Replace” atau **CTRL+H**, kemudian pada bagian Search Mode, pilih Extended (\n,\r,\t,\0,\x,...)

#### **PENTING**

Penting untuk diperhatikan bahwa jika ingin menghilangkan spasi, tab, atau enter (CR dan LF), silakan aktifkan fitur **Show All Character** yang telah kita bahas sebelumnya sehingga terlihat mana **SPASI**, **TAB**, **CR** dan **LF** nya, untuk menghilangkan enter, gunakan **\r** untuk **CR** dan **\n** untuk **LF**.

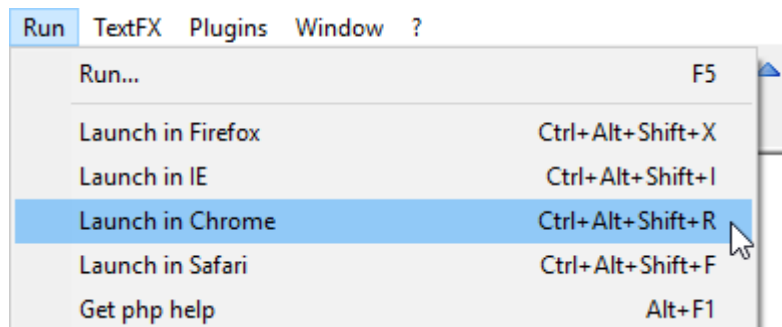


↓

3	1	→	A	→	2014	→	Jakarta	CRIP
4	2	→	B	→	2014	→	Medan	CRIP
5	3	→	C	→	2014	→	Semarang	CRIP
6	4	→	D	→	2014	→	Solo	IP
7	5	→	E	→	2014	→	Madiun	IP
8	6	→	F	→	2014	→	Jakarta	IP
9	7	→	G	→	2013	→	Solo	CRIP

## 15. Launch In Browser

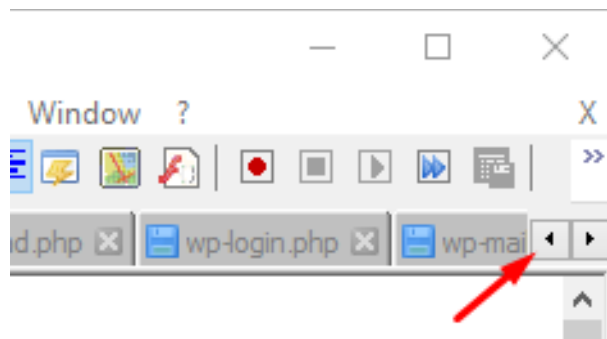
Fitur lain yang bermanfaat yaitu Launch in Browser, dimana secara otomatis kita dapat menjalankan browser untuk menampilkan kode yang sedang kita buat, caranya klik menu Run, selanjutnya pilih browser yang diinginkan,



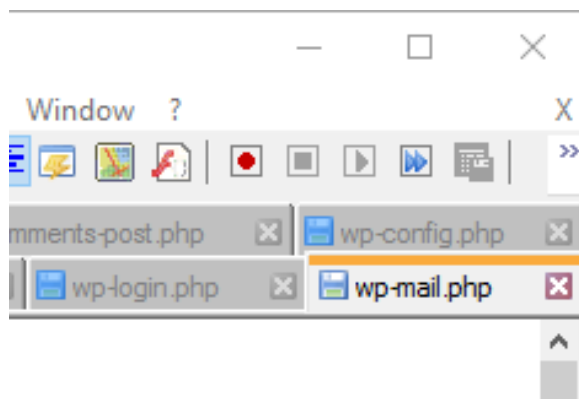
Jika browser tersebut tidak ada dalam daftar sobat dapat mendefinisikan nya dengan memilih sub menu Run..., kemudian pilih file exe browser tersebut, selanjutnya pilihan tersebut dapat disimpan pada shortcut dengan mengklik tombol Save..

## 16. Manajemen Tab

Pada Notepad++, dokumen yang terbuka disusun menjadi tab yang secara default diletakkan di sebelah atas dan di bagian bawah toolbar, jika file yang dibuka banyak hingga melebihi area yang ada, maka akan menyulitkan karena kita harus mencarinya dengan mengklik tanda panah ke kanan dan ke kiri.



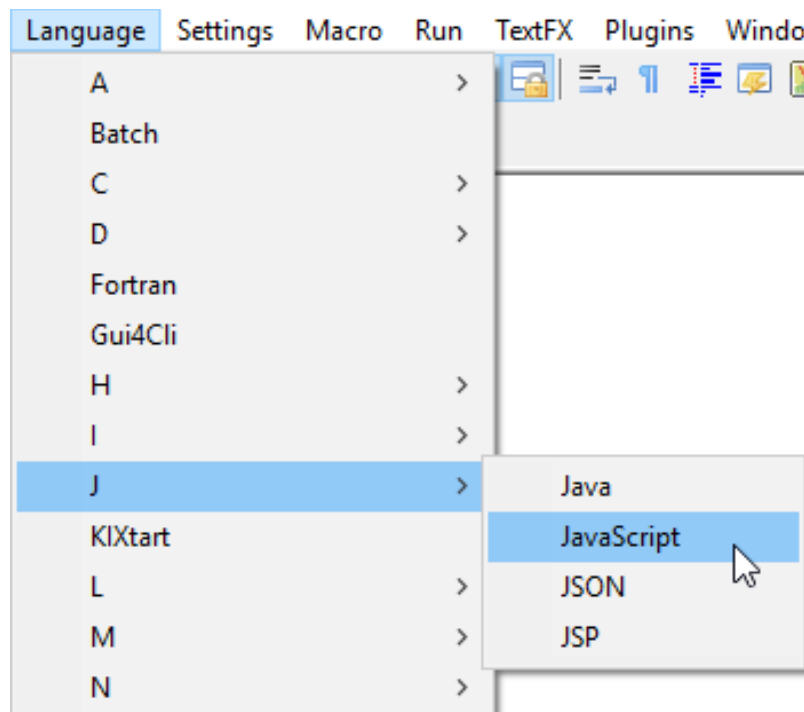
Hal tersebut dapat diatasi dengan membuat Tab tersebut menjadi multi line, caranya, buka menu **Settings > Preferences**, kemudian pilih tab **General**, pada bagian Tab Bar, centang pilihan **Multi-line**. Selain itu, pada bagian ini terdapat berbagai pilihan menarik lainnya seperti memunculkan tombol close pada tab atau menutup tab dengan double klik.



Terdapat alternatif lain yaitu menampilkan daftar file yang terbuka pada window tersendiri, hal ini dapat dilakukan dengan bantuan plugin Window Manager.

#### 17. Select Language

Notepad++ akan memberikan warna khusus untuk bagian tertentu dari kode seperti warna khusus untuk class, fungsi, string, variabel, numeric, dll, yang biasanya disebut dengan Syntax Highlight. Notepad++ akan otomatis menyesuaikan nya dengan jenis file yang kita buka (php, js, css, html, dll), namun demikian sobat dapat mendefinisikan nya sendiri melalui menu Language.



#### 18. Plugins

Banyak sekali plugin yang dapat kita gunakan untuk mempermudah dan mempercepat pekerjaan kita, seperti Code Alignment, NppFTP, Finger Text, NppExport, dan JSTools,

### 3.1.2 Microsoft Visual Studio 2010



*Visual Studio 2010* pada dasarnya adalah sebuah Bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu.

*Visual Studio 2010* (yang sering juga disebut dengan *VB .Net 2010*) selain disebut dengan bahasa pemrograman, juga sering disebut sebagai sarana (*tool*) untuk menghasilkan program-program aplikasi berbasis *windows*. Beberapa kemampuan atau manfaat dari *Visual Studio 2010* diantaranya seperti :

1. Untuk membuat program aplikasi berbasis *windows*.
2. Untuk membuat objek-objek pembantu program seperti, misalnya : kontrol *ActiveX*, *file Help*, aplikasi Internet dan sebagainya.

3. Menguji program (*debugging*) dan menghasilkan program berakhiran *EXE* yang bersifat *executable* atau dapat langsung dijalankan.

Visual Studio 2010 adalah bahasa yang cukup mudah untuk dipelajari. Bagi programmer pemula yang baru ingin belajar program, lingkungan *Visual Studio* dapat membantu membuat program dalam sekejap mata. Sedang bagi *programmer* tingkat lanjut, kemampuan yang besar dapat digunakan untuk membuat program-program yang kompleks, misalnya lingkungan *networking* atau *client server*.

Bahasa *Visual Studio* cukup sederhana dan menggunakan kata-kata bahasa Inggris yang umum digunakan. Kita tidak perlu lagi menghafalkan sintaks-sintaks maupun format-format bahasa yang bermacam-macam, di dalam *Visual Basic* semuanya sudah disediakan dalam pilihan-pilihan yang tinggal diambil sesuai dengan kebutuhan. Selain itu, sarana pengembangannya yang bersifat visual memudahkan kita untuk mengembangkan aplikasi berbasis *windows*, bersifat *mouse-driven* (digerakkan dengan *mouse*) dan berdaya guna tinggi.

### **Sejarah Singkat Visual Studio 2010**

Berikut ini beberapa point penting dalam sejarah perkembangannya :

1. Pertama kali di *release* dengan nama *Visual Basic* yang dikeluarkan pada tahun 1991, yaitu *Visual Basic* yang masih berbasis *DOS* dan untuk *Windows*

2. *Visual Basic 3.0* dirilis pada tahun 1993
3. *Visual Basic 4.0* dirilis pada tahun 1994 dengan dukungan untuk aplikasi 32 bit
4. *Visual basic 6.0* dirilis pada akhir tahun 1998
5. *Visual basic* untuk selanjutnya yaitu versi .Net yang dirilis awal 2002
6. Pada tahun 2003 *Visual Studio 2003* dirilis untuk memperbaiki kinerja dari *Visual Studio 2002* dengan meluncurkan *.NET Framework* versi 1.1.
7. Pada tahun 2005 *Microsoft* mengeluarkan *Visual Basic* Versi 8.0 atau biasa di sebut *Microsoft Visual Studio 2005*.
8. Pada tahun 2008 *Microsoft* juga mengeluarkan versi 9.0 atau biasa di sebut *Microsoft Visual Studio 2008*.
9. Hingga pada tahun 2010 *Visual Studio* sudah pada versi 10.0 atau biasa di sebut *Microsoft Visual Studio 2010*, dimana didalamnya sudah ditambahkan bahasa pemrograman yang baru yaitu *F#* yang merupakan penyempurnaan dari versi-versi sebelumnya.

### **Keistimewaan *Visual Studio 2010***

Beberapa keistimewaan *Visual Studio 2010* ini diantaranya seperti :

1. Menggunakan *platform* pembuatan program yang dinamakan *developer studio*, yang memiliki tampilan dan

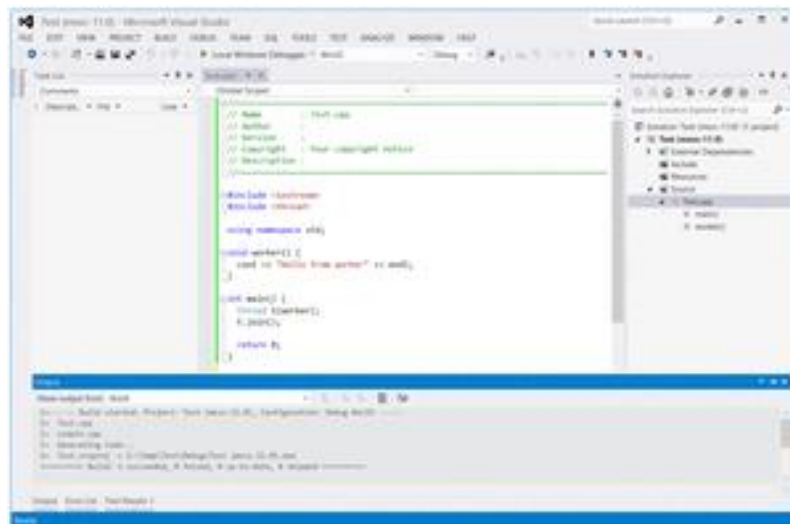
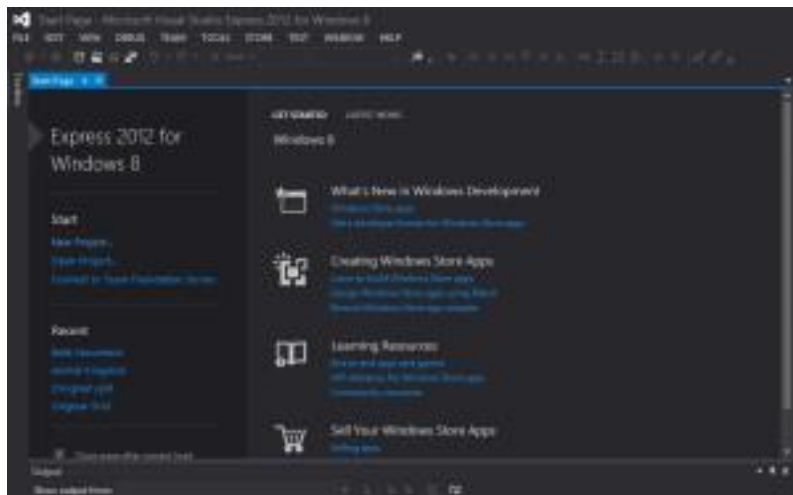
sarana yang sama dengan *Visual C++* dan *Visual J++*. Dengan begitu Anda dapat bermigrasi atau belajar bahasa pemrograman lainnya dengan mudah dan cepat.

2. Memiliki *compiler* handal yang dapat menghasilkan *file executable* yang lebih cepat dan lebih efisien dari yang sebelumnya.
3. Memiliki beberapa tambahan *wizard* yang baru. *Wizard* adalah sarana yang mempermudah di dalam pembuatan aplikasi dengan mengotomisasi tugas-tugas tertentu.
4. *Visual Studio 2010* mempunyai beberapa fitur untuk pengembangan berbagai macam aplikasi yang diantaranya; *Windows Development*, *Web Development*, *Office Development*, *Sharepoint Development*, *Cloud Development (Windows Azure)*, *Silverlight Tooling*, *Multi-Core Development*, *Customizable IDE*.

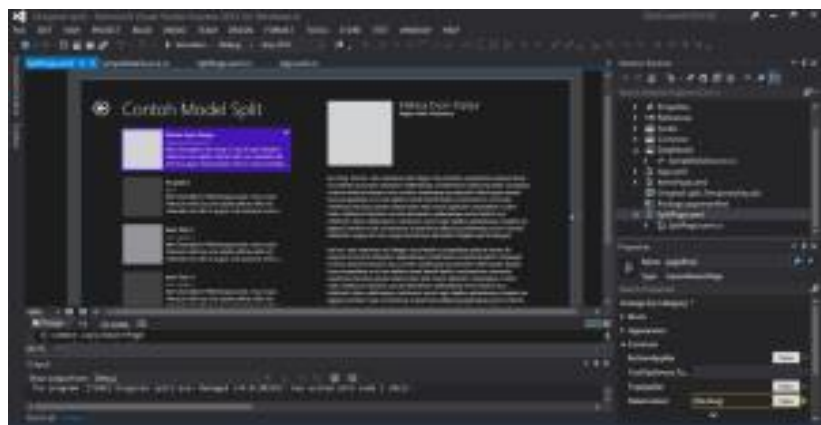
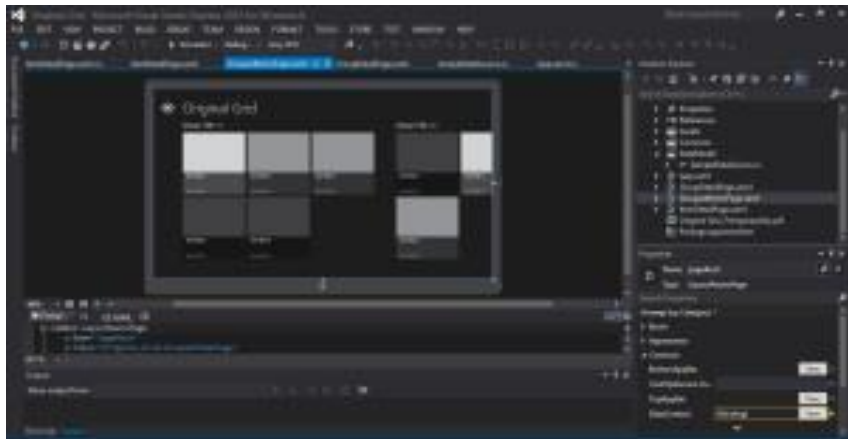
Nah Fitur-Fitur baru yang saya rasakan diantaranya :

1. Untuk User interface yang jelas sudah baik sekali , tampilanya simple tapi mengandung banyak fitur bagus, ada 2 tema yang dapat kita pilih yaitu Dark dan Light.



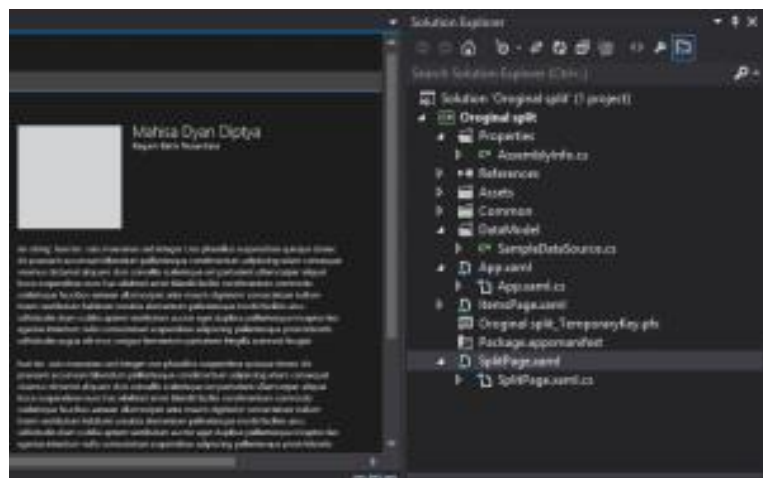


2. Kemampuannya membangun aplikasi Metro Style , dengan menggunakan c++ , c#, visual basic, HTML, JavaScript, XAML Kita sudah dapat membuat aplikasi Metro-Style untuk windows 8, tetapi kalau gak mau ribet Visual studio juga menyediakan 2 template Metro Style nya, Diantaranya model Grid dan Split



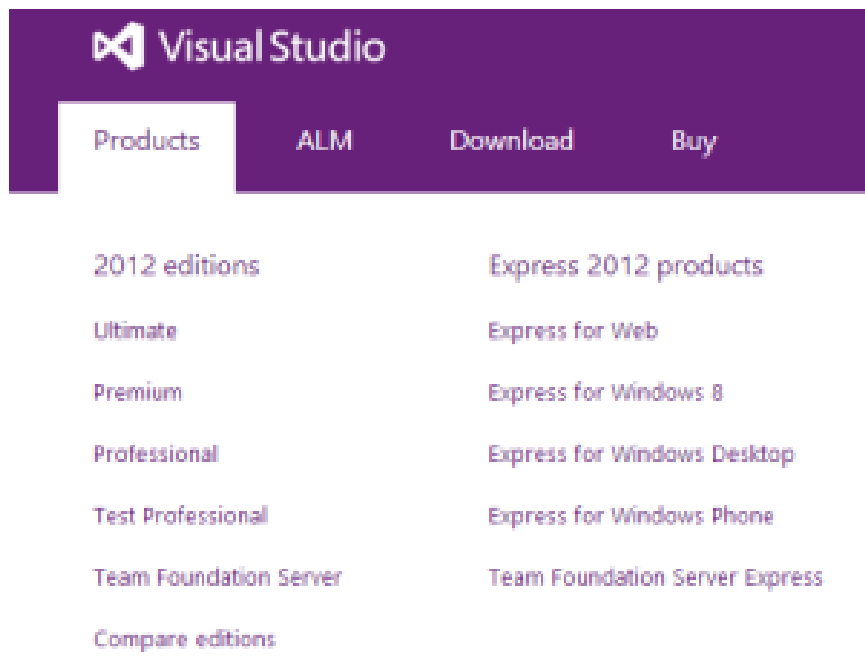
3. LightSwitch, dengan LightSwitch kita akan dapat mempublish aplikasi yang sudah kita buat ke Azure dengan cara yang lebih gampang dan simple, selain itu membuat performa menjadi lebih baik misalnya peningkatan proses loading pada saat membuka *project*, membuat screens dan tables, juga ketika kita Building Projects. oh iya sekedar info aja, pada versi sebelumnya LightSwitch ini terpisah dengan Visual Studio tetapi sekarang sudah termasuk/include di dalam visual Studio 2012.

4. Solution Explorer yang berbeda, Solution Explorer itu merupakan semacam jendela yang mengandung semua file aplikasi Visual Studio dan dimana setiap proyek dapat mengandung lebih dari satu file. Jadi, kita dapat melakukan penelusuran ke methods dan properties pada proyek kita dan juga memungkinkan kita untuk melakukan pencarian file dan yang paling saya suka karena dapat melakukan preview pada objek-objek dan item-item eksternal.



5. Oh iya yang terakhir yang terbaru juga, dukungan dari .NET Framework 4.5

Nah untuk versi dari Visual Studio 2012, terdapat pada gambar di bawah ini :



### 3.1.3 *Sublime Text*



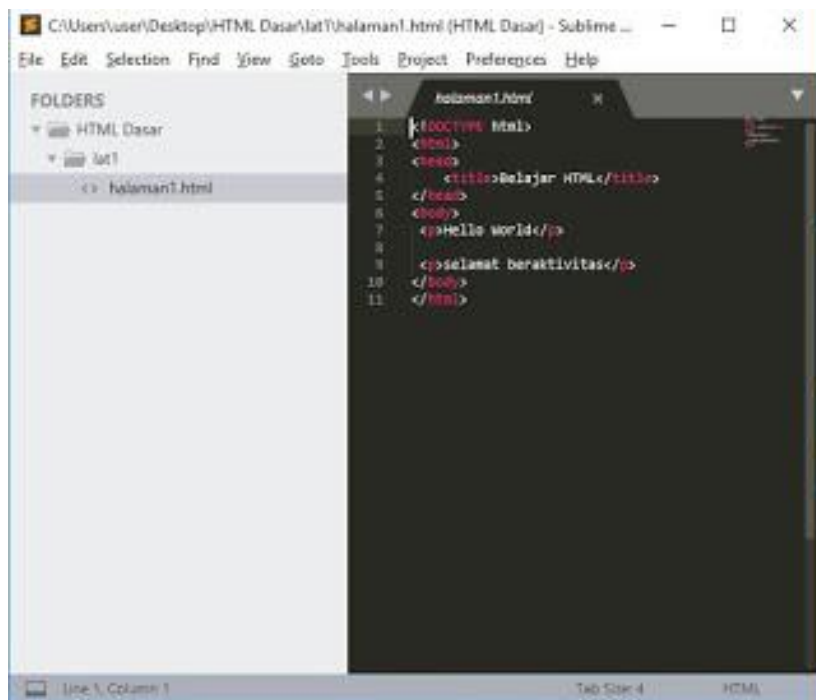
*Sublime text* adalah salah satu *text editor* yang kini cukup banyak peminatnya, dan penggunaan *software* ini bisa digunakan

juga oleh berbagai macam *platform OS (Operating System)*. *Sublime text* juga banyak sekali mendukung banyak bahasa pemrograman dan bahasa markup.

*Sublime text* diresmikan pada tahun 2008 dan sudah memiliki beberapa versi dan kini sudah sampai ke versi *Sublime text 3* juga. Penggunaan *sublime text* banyak sekali membantu pekerjaan seorang *web developer* atau *programer* lainnya seperti memudahkan *programer* dalam membedakan *syntak* dengan warna yang dimilikinya dan banyak juga kelebihan lainnya yang dapat dilakukan oleh *text editor* ini.

### **Cara menggunakan Sublime Text**

Cara dasar yang dapat kalian lakukan pertama sekali dalam menggunakan aplikasi ini yaitu silahkan buka Aplikasi *sublime text* dengan cara klik 2x atau juga bisa klik kanan dan pilih *open*. Selain itu kalian juga bisa memasukan folder *project* kalian kedalam *sublime text* agar nantinya dengan mudah untuk menemukan *file* lainnya yang terdapat pada folder *project* tanpa harus menggunakan *open file*, cara sangat mudah yaitu saat *sublime text* kalian sudah terbuka silahkan tarik folder *project* kalian kedalam *sublime text* maka akan otomatis *sublime text* akan membaca folder tersebut dan meletakkannya pada *side* kiri, atau juga bisa kalian menggunakan klik menu *file* pada bar dan pilih *open folder*.



## Kelebihan *Sublime Text*

Banyak kelebihan yang dimiliki oleh text editor ini, dan berikut saya akan memaparkan beberapa kelebihan dalam penggunaan sublime text diantara:

### 4. *Software* yang ringan

Kelebihan pertama yang dimiliki oleh *text editor* satu ini diantaranya yaitu *software* yang ringan dan tidak banyak memakan *RAM* pada komputer kalian.

### 5. *Support Platform*

Yang kedua yaitu *support* dibanyak *platform OS* seperti *Windows* dan lainnya.

## 6. Mini Map

Selain itu sublime text juga memiliki *mini map* disisi kanan atasnya untuk mempermudah kita dalam menemukan kode.

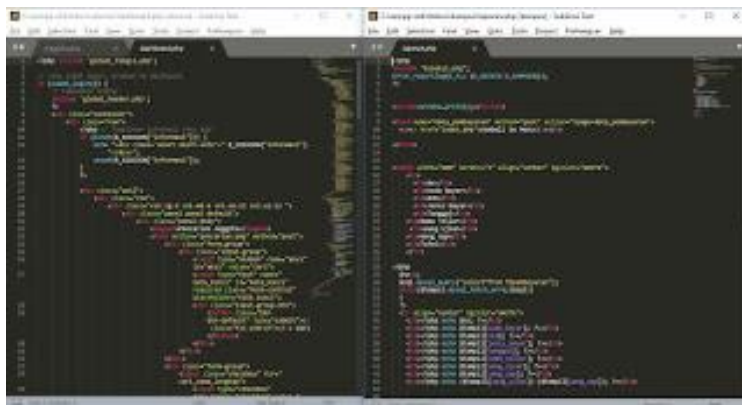
## 7. Mengganti *Background*.

Kelebihan lainnya yaitu kalian juga bisa mengganti *background* atau *scheme text editor* satu ini jika kalian bosan dengan warna bawaannya yaitu hitam dengan cara pilih menu *preferences* pada menu bar dan pilih *color scheme*.

### Fitur dimiliki *Sublime Text*

Selain kelebihan diatas sebenar fitur-fitur yang dimiliki *sublime text* ini juga merupakan kelebihan dari *text editor* ini, dan *sublime text* memiliki banyak sekali fitur-fitur yang menarik dan sangat mempermudah bagi seorang yang menuliskan *script* pada *text editor* ini. diantaranya yaitu:

### 1. Split Editing



Penggunaan *split mode* bisa dilakukan bagi kalian ingin membuka beberapa *file* atau beberapa *project* sekaligus tanpa harus menutup *project* lainnya.

## 2. Multi Selection

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title></title>
5  </head>
6  <body>
7
8  </body>
9  </html>
```

*Multi Selection* yaitu fitur yang digunakan untuk dapat menyeleksi beberapa baris *text* atau kode pada *script*, cara menggunakannya hanya dengan (**CTRL+Klik**) dengan fungsi ini sangat memudahkan *programer* dalam mengedit dalam banyak baris.

### 3.2 Software untuk Membuat Aplikasi Mobile

Tidak hanya dalam pembuatan aplikasi *website* yang membutuhkan *software* pendukung, untuk membuat Aplikasi *Mobile* juga membutuhkan *software* pendukung. Berikut ini adalah beberapa contoh *software* untuk membuat Aplikasi *Mobile*.

#### 3.2.1 Android Studio





Android Studio merupakan Integrated Development Environment (IDE) resmi untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Android berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembangan utama dibelakangnya, yaitu Google. Googlelah yang mengakuisisi Android dan kemudian membuatkan sebuah platform.

Platform android terdiri dari Sistem Operasi berbasis Linux, sebuah GUI (Graphic User Interface), sebuah web browser dan Aplikasi Studio End-User yang dapat di download dan juga para pengembang bisa dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat.

## **Sejarah Android**

Pada awalnya mulanya, Android Inc merupakan sebuah perusahaan software kecil yang didirikan pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior di beberapa perusahaan yang berbasis IT & Communication, Andy Rubin, Rich Miner, Nick Sears dan Chris White.

Menurut Rubin, Android Inc Didirikan untuk mewujudkan mobile device yang lebih peka terhadap lokasi dan preferensi pemilik. Dengan kata lain, Android Inc ingin mewujudkan mobile device yang lebih mengerti pemiliknya.

Konsep yang dimiliki Android Inc ternyata menggugah minat Google untuk memilikinya. Pada bulan Agustus 2005, Akhirnya Android Inc diakuisisi oleh Google Inc. seluruh sahamnya dibeli oleh Google. Banyak yang memperkirakan nilai pembelian Android Inc Oleh Google adalah sebesar USD 50 juta. saat itu Banyak yang berspekulasi bahwa akuisisi ini adalah langkah awal yang dilakukan Google untuk masuk kepasar mobile phone.



Andy Rubin, Rich Miner, Nick Sears dan Chris White tetap di Android Inc yang dibeli Google, sehingga akhirnya mereka semua menjadi bagian dari raksasa Google dan sejarah Android. Saat itulah mereka mulai menggunakan platform Linux untuk membuat sistem operasi bagi mobile phone.

Di atas alat editor dan pengembang hebat IntelliJ, Android Studio menawarkan lebih banyak fitur yang meningkatkan produktivitas Anda saat membangun aplikasi Android, seperti:

***Sistem build berbasis Gradle yang 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 contoh impor*
- 5). *Alat pengujian ekstensif dan kerangka kerja*
- 6). *Alat lint untuk menangkap kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya*
- 7). *Dukungan C++ dan NDK*
- 8). *Dukungan terintegrasi untuk Google Cloud Platform, sehingga mudah mengintegrasikan Google Cloud Messaging dan App Engine*
- 9). *Memberikan pengenalan fitur dasar Android Studio. Untuk ringkasan perubahan terbaru, lihat Android Studio Release*
- 10). *Struktur Proyek*

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

- a). Modul aplikasi Android
- b). Modul perpustakaan
- c). Modul Google App Engine Secara 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 berisi folder berikut:

**Bermanifestasi:** Berisi file AndroidManifest.xml.

**Java:** Berisi file kode sumber Java, termasuk kode uji JUnit.

**Res:** Berisi semua 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.

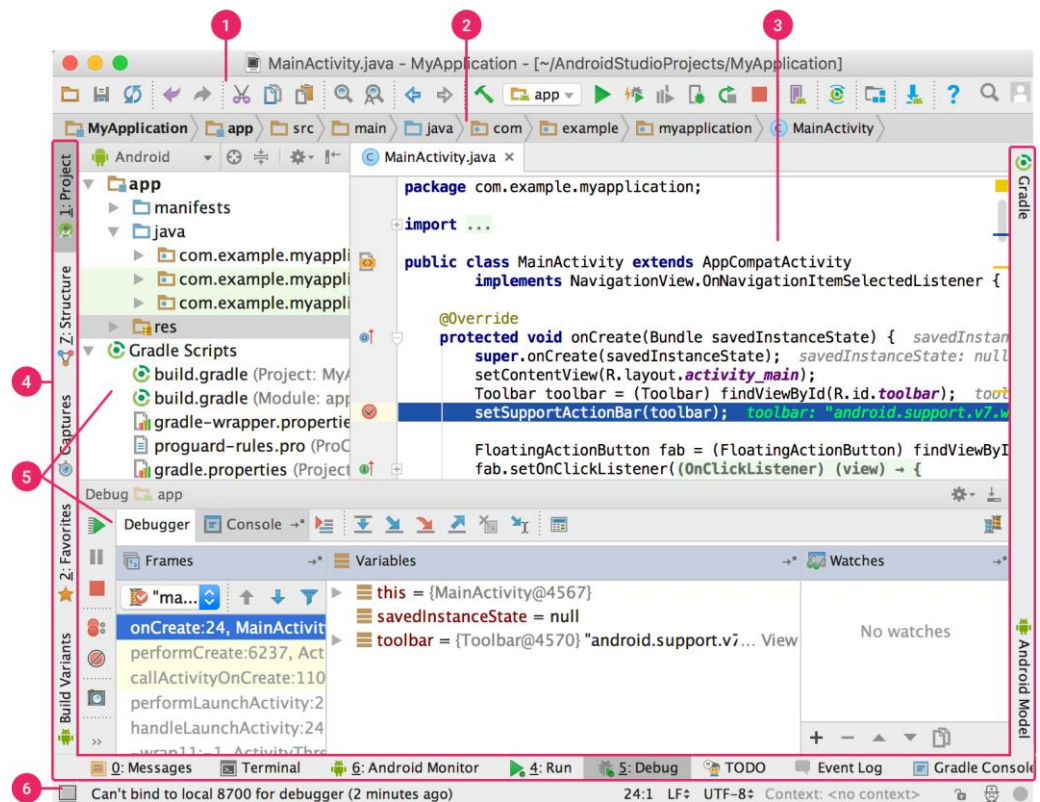
Anda juga dapat menyesuaikan tampilan file proyek agar fokus pada aspek spesifik pengembangan aplikasi Anda. Misalnya, memilih tampilan Masalah dari proyek Anda yang menampilkan tautan ke file

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.

## Antarmuka Pengguna

Jendela utama Android Studio terdiri dari beberapa area logis yang diidentifikasi dalam gambar 3.



**Gambar 3.** Jendela utama Android Studio.

1. **Toolbar** memungkinkan Anda melakukan berbagai tindakan, termasuk menjalankan aplikasi dan meluncurkan fitur Android.
2. **Menu navigasi** membantu Anda menjelajah project dan membuka file untuk diedit. Menu ini memberikan tampilan struktur yang lebih ringkas yang terlihat di jendela **Project**.
3. **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.
4. **Panel jendela fitur** berada di sisi luar jendela IDE dan berisi tombol-tombol yang memungkinkan Anda memperluas atau menciutkan setiap jendela fitur.
5. **Jendela fitur** memberi Anda akses ke tugas tertentu seperti pengelolaan project, penelusuran, kontrol versi, dan banyak lagi. Anda dapat memperluas dan menciutkan jendela ini.
6. **Status bar** menampilkan status project Anda dan IDE itu sendiri, serta semua peringatan atau pesan.


Anda dapat mengatur jendela utama untuk memperluas ruang layar dengan menyembunyikan atau memindahkan toolbar dan jendela fitur. Anda juga dapat menggunakan pintasan keyboard untuk mengakses sebagian besar fitur IDE.

Anda dapat menelusuri kode sumber, database, tindakan, elemen antarmuka pengguna, dan sebagainya kapan saja, dengan menekan

tombol Shift dua kali, atau mengklik kaca pembesar di sudut kanan atas jendela Android Studio. Tips ini sangat berguna jika, misalnya, Anda mencoba menemukan tindakan IDE tertentu yang Anda lupa cara memicunya.

## **Jendela fitur**

Sebagai ganti menggunakan perspektif preset, Android Studio mengikuti konteks Anda dan otomatis menampilkan jendela fitur yang relevan saat Anda bekerja. Secara default, jendela fitur yang paling umum digunakan disematkan ke panel jendela fitur di tepi jendela aplikasi.

- Untuk memperluas atau menciutkan jendela fitur, klik nama fitur di panel jendela fitur. Anda juga dapat menarik, menyematkan, melepaskan sematan, memasang, dan melepas jendela fitur.
- Untuk kembali ke tata letak jendela fitur default saat ini, klik **Window > Restore Default Layout** atau sesuaikan tata letak default dengan mengklik **Window > Store Current Layout as Default**.
- Untuk menampilkan atau menyembunyikan seluruh panel jendela alat, klik ikon jendela  di pojok kiri bawah jendela Android Studio.
- Untuk menemukan jendela alat tertentu, arahkan kursor ke atas ikon jendela dan pilih jendela alat tersebut dari menu.

Anda juga bisa menggunakan pintasan keyboard untuk membuka jendela alat. Tabel 1 mencantumkan pintasan jendela paling umum.

**Tabel 1.** Pintasan keyboard ke beberapa jendela alat yang berguna.

Jendela alat	Windows dan Linux
Project	Alt+1
Kontrol Versi	Alt+9
Run	Shift+F10
Debug	Shift+F9
Logcat	Alt+6
Kembali ke Editor	Esc
Menyembunyikan Semua Jendela Alat	Control+Shift+F12

Jika Anda ingin menyembunyikan semua toolbar, jendela alat, dan tab editor, klik **View > Enter Distraction Free Mode**. Langkah ini akan mengaktifkan *Distraction Free Mode*. Untuk keluar dari Distraction Free Mode, klik **View > Exit Distraction Free Mode**.

Anda dapat menggunakan *Speed Search* untuk menelusuri dan memfilter di dalam sebagian besar jendela fitur pada Android



Studio. Untuk menggunakan Speed Search, pilih jendela alat, lalu ketik kueri penelusuran Anda.

Untuk tips lainnya, lihat [Pintasan keyboard](#).

## Pelengkapan kode

Android Studio memiliki tiga jenis pelengkapan kode, yang dapat Anda akses menggunakan pintasan keyboard.

**Tabel 2.** Pintasan keyboard untuk pelengkapan kode.

Jenis	Deskripsi	Windows
Basic Completion	Menampilkan saran dasar untuk variabel, jenis, metode, ekspresi, dan sebagainya. Jika memanggil Basic Completion dua kali berturut-turut, Anda akan melihat lebih banyak hasil, termasuk anggota pribadi dan anggota statis yang tidak diimpor.	Control+
Smart Completion	Menampilkan opsi yang relevan berdasarkan konteks. Smart Completion mengetahui jenis dan alur data yang diharapkan. Jika Anda memanggil Smart Completion dua kali berturut-turut, Anda akan melihat lebih banyak hasil, termasuk chain.	Control+
Statement	Melengkapi pernyataan saat ini secara otomatis, seperti menambahkan tanda kurung, tanda kurung siku, tanda	Control+

Completion kurung kurawal, pemformatan, dll. yang tidak lengkap.

Anda juga dapat melakukan perbaikan cepat dan menampilkan tindakan maksud dengan menekan **Alt+Enter**.

### Menemukan kode contoh

Code Sample Browser di Android Studio membantu Anda menemukan contoh kode Android berkualitas tinggi yang disediakan Google berdasarkan simbol yang saat ini ditandai dalam project Anda. Untuk informasi selengkapnya, lihat Menemukan kode contoh.

### Navigasi

Berikut ini beberapa tips untuk membantu Anda menjelajah di dalam Android Studio.

- Beralih antar file yang baru saja diakses menggunakan tindakan *Recent Files*. Tekan **Control+E** (**Command+E** pada Mac) untuk memunculkan tindakan Recent Files. Secara default, file yang terakhir diakses akan dipilih. Anda juga dapat mengakses jendela fitur mana saja melalui kolom kiri dalam tindakan ini.
- Lihat struktur file saat ini menggunakan tindakan *File Structure*. Munculkan tindakan File Structure dengan menekan **Control+F12** (**Command+F12** pada Mac). Dengan

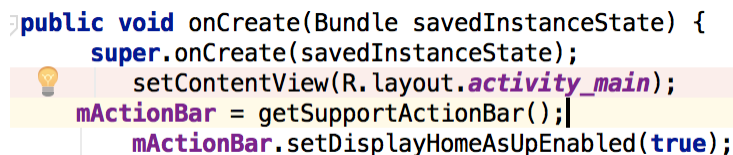
tindakan ini, Anda dapat membuka bagian mana pun dari file saat ini dengan cepat.

- Telusuri dan buka class tertentu dalam project menggunakan tindakan *Navigate to Class*. Munculkan tindakan ini dengan menekan **Control+N** (**Command+O** pada Mac). *Navigate to Class* mendukung ekspresi canggih, termasuk camel humps, jalur, baris navigasi ke, pencocokan nama tengah, dan banyak lagi. Jika Anda memanggilnya dua kali berturut-turut, hasil dari class project akan ditampilkan.
- Buka file atau folder menggunakan tindakan *Navigate to File*. Munculkan tindakan *Navigate to File* dengan menekan **Control+Shift+N** (**Command+Shift+O** pada Mac). Untuk menelusuri folder dan bukan file, tambahkan / (garis miring) di akhir ekspresi Anda.
- Buka metode atau kolom menurut nama menggunakan tindakan *Navigate to Symbol*. Munculkan tindakan *Navigate to Symbol* dengan menekan **Control+Shift+Alt+N** (**Command+Option+O** pada Mac).
- Temukan semua bagian kode yang merujuk ke class, metode, kolom, parameter, atau pernyataan di posisi kursor saat ini dengan menekan **Alt+F7** (**Option+F7** pada Mac).

## Gaya dan pemformatan

Saat Anda mengedit, Android Studio otomatis menerapkan pemformatan dan gaya seperti yang ditentukan dalam setelan gaya kode Anda. Anda dapat menyesuaikan setelan gaya kode menurut bahasa pemrograman, termasuk menentukan konvensi untuk tab dan indentasi, spasi, penggabungan, tanda kurung kurawal, dan baris kosong. Untuk menyesuaikan setelan gaya kode Anda, klik **File > Settings > Editor > Code Style (Preferences > Editor > Code Style** pada Mac.)

Meskipun IDE otomatis menerapkan pemformatan selagi Anda bekerja, Anda juga bisa memanggil tindakan *Reformat Code* secara eksplisit dengan menekan **Control+Alt+L (Opt+Command+L** pada Mac), atau otomatis mengindentasi semua baris dengan menekan **Control+Alt+I (Control+Option+I** pada Mac).



```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    mActionBar = getSupportActionBar();  
    mActionBar.setDisplayHomeAsUpEnabled(true);  
}
```

**Gambar 4.** Kode sebelum pemformatan.

```

} public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    mActionBar = getSupportActionBar();
    mActionBar.setDisplayHomeAsUpEnabled(true);
    // Get reference to the drawer layout and set event listener

```

Formatted 7 lines  
Show reformat dialog: ⌘⇧⌘L

**Gambar 5.** Kode setelah pemformatan.

## Dasar-dasar kontrol versi

Android Studio mendukung berbagai sistem kontrol versi (VCS), termasuk Git, GitHub, CVS, Mercurial, Subversion, dan Google Cloud Source Repositories.

Setelah mengimpor aplikasi Anda ke Android Studio, gunakan opsi menu VCS pada Android Studio untuk mengaktifkan dukungan VCS bagi sistem kontrol versi yang diinginkan, membuat repositori, mengimpor file baru ke kontrol versi, dan menjalankan operasi kontrol versi lainnya:

1. Dari menu **VCS** Android Studio, klik **Enable Version Control Integration**.
2. Dari menu drop-down, pilih sistem kontrol versi yang ingin dikaitkan dengan root project, lalu klik **OK**.

Menu VCS sekarang menampilkan sejumlah opsi kontrol versi berdasarkan sistem yang Anda pilih.

## Sistem build Gradle

---

Android Studio menggunakan Gradle sebagai dasar dari sistem build, dengan lebih banyak kemampuan khusus Android yang disediakan oleh plugin Android untuk Gradle. Sistem build ini berjalan sebagai fitur terintegrasi dari menu Android Studio, dan terpisah dari command line. Anda dapat menggunakan fitur-fitur sistem build untuk:

- Menyesuaikan, mengonfigurasi, dan memperluas proses pembuatan build.
- Membuat banyak APK untuk aplikasi Anda, dengan berbagai fitur yang menggunakan project dan modul yang sama.
- Menggunakan kembali kode dan resource di seluruh set sumber.

Berkat fleksibilitas Gradle, Anda dapat mencapai semua ini tanpa mengubah file sumber inti aplikasi Anda. File build Android Studio diberi nama `build.gradle`. File tersebut adalah file teks biasa yang menggunakan sintaks Groovy untuk mengonfigurasi build dengan elemen yang disediakan oleh plugin Android untuk Gradle. Setiap project memiliki satu file build tingkat atas untuk seluruh project dan file build tingkat modul terpisah untuk setiap modul. Saat Anda mengimpor project yang ada, Android Studio akan otomatis menghasilkan file build yang diperlukan.

Untuk mempelajari sistem build dan cara mengonfigurasinya lebih lanjut, lihat Mengonfigurasi build.

## **Varian build**

Sistem build dapat membantu Anda membuat beberapa versi berbeda untuk aplikasi yang sama dari satu project. Hal ini berguna saat Anda menyediakan aplikasi dalam versi gratis dan berbayar, atau jika Anda ingin mendistribusikan beberapa APK untuk berbagai konfigurasi perangkat di Google Play.

Untuk informasi selengkapnya tentang cara mengonfigurasi varian build, lihat [Mengonfigurasi varian build](#).

## **Dukungan multi-APK**

Dukungan multi-APK memungkinkan Anda membuat beberapa APK sekaligus secara efisien berdasarkan kepadatan layar atau ABI. Misalnya, Anda dapat membuat APK aplikasi terpisah untuk kepadatan layar hdpi dan mdpi, dengan tetap menganggapnya sebagai varian tunggal serta mengizinkannya berbagi setelan APK pengujian, javac, dx, dan ProGuard.

Untuk informasi selengkapnya tentang dukungan multi-APK, baca [Membuat beberapa APK](#).

## **Penyusutan resource**

Penyusutan resource di Android Studio otomatis menghapus resource yang tidak digunakan dari aplikasi terpaket dan dependensi library Anda. Misalnya, jika aplikasi Anda menggunakan [layanan Google Play](#) untuk mengakses fungsi Google Drive, dan saat ini

Anda tidak menggunakan Login dengan Google, penyusutan resource dapat menghapus beragam aset drawable untuk tombol `SignInButton`.

## **Mengelola dependensi**

Dependensi untuk proyek Anda ditetapkan menurut nama dalam file `build.gradle`. Gradle menangani penemuan dependensi dan menyediakannya di build Anda. Anda dapat mendeklarasikan dependensi modul, dependensi biner jarak jauh, dan dependensi biner lokal dalam file `build.gradle` Anda. Android Studio mengonfigurasi project untuk menggunakan Maven Central Repository secara default. (Konfigurasi ini disertakan dalam file build tingkat atas untuk project.) Untuk informasi selengkapnya tentang mengonfigurasi dependensi, baca [Menambahkan dependensi build](#).

## **Fitur profil dan debug**

---

Android Studio membantu Anda menjalankan proses debug dan meningkatkan performa kode, termasuk proses debug inline dan fitur analisis performa.

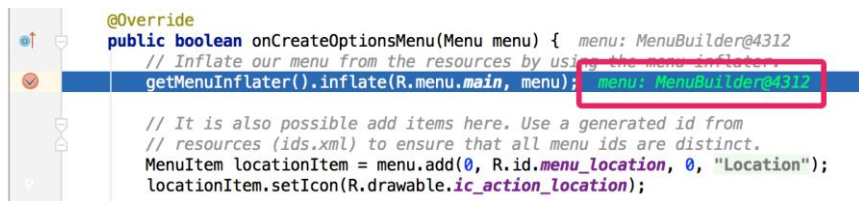
### **Proses debug inline**

Gunakan proses debug inline untuk menyempurnakan panduan kode Anda dalam tampilan debugger dengan verifikasi inline untuk




referensi, ekspresi, dan nilai variabel. Informasi debug inline meliputi:

- Nilai variabel inline
- Objek perujuk yang merujuk ke objek terpilih
- Nilai yang dihasilkan metode
- Ekspresi operator dan lambda
- Nilai tooltip



**Gambar 6.** Nilai variabel inline.

Untuk mengaktifkan proses debug inline, di jendela **Debug**, klik Settings  dan centang kotak Show Values Inline.

## Profiler performa

Android Studio menyediakan profiler performa agar Anda dapat melacak penggunaan memori dan CPU aplikasi, menemukan objek yang batal dialokasikan, menemukan kebocoran memori, mengoptimalkan performa grafis, dan menganalisis permintaan jaringan dengan mudah. Saat aplikasi Anda berjalan di perangkat atau emulator, buka tab **Android Profiler**.

Untuk informasi lebih lanjut tentang profiler performa, lihat [Alat profiling performa](#).

## **Heap dump**

Saat memprofilkan penggunaan memori di Android Studio, Anda dapat sekaligus memulai pembersihan sampah memori dan membuang heap Java ke cuplikan heap dalam file format biner HPROF khusus Android. Penampil HPROF akan menampilkan class, instance setiap class, dan struktur referensi untuk membantu Anda melacak penggunaan memori serta menemukan kebocoran memori.

Untuk informasi selengkapnya tentang menangani heap dump, lihat [Memeriksa heap dan alokasi](#).

## **Memory Profiler**

Anda dapat menggunakan Memory Profiler untuk melacak alokasi memori dan melihat di mana objek dialokasikan saat Anda melakukan tindakan tertentu. Dengan mengetahui alokasi ini, Anda dapat mengoptimalkan performa dan penggunaan memori aplikasi dengan menyesuaikan panggilan metode yang terkait dengan tindakan tersebut.

Untuk informasi tentang pelacakan dan analisis alokasi, lihat [Memeriksa heap dan alokasi](#).

## Akses file data

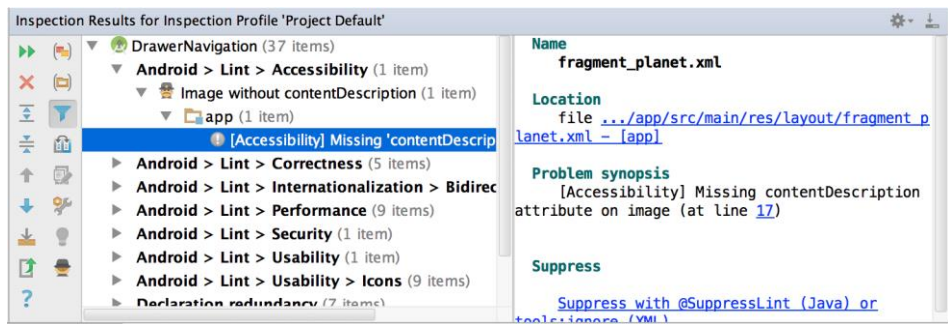
Android SDK Tools, seperti Systrace dan logcat, menghasilkan data performa dan proses debug untuk analisis aplikasi secara mendetail.

Untuk menampilkan file data yang dihasilkan, buka jendela alat Captures. Pada daftar file yang dihasilkan, klik dua kali file untuk melihat data. Klik kanan sembarang file `.hprof` untuk mengonversinya ke dalam format file standar Menyelidiki penggunaan RAM.

## Pemeriksaan kode

Setiap kali Anda mengompilasi program, Android Studio akan otomatis menjalankan Lint yang telah dikonfigurasi dan pemeriksaan IDE lainnya untuk memudahkan Anda mengidentifikasi serta memperbaiki masalah kualitas struktur kode Anda.

Fitur Lint memeriksa file sumber project Android Anda untuk menemukan potensi bug dan peluang pengoptimalan guna mencapai ketepatan, keamanan, performa, kegunaan, aksesibilitas, serta internasionalisasi.



**Gambar 7.** Hasil pemeriksaan Lint di Android Studio.

Selain pemeriksaan Lint, Android Studio juga menjalankan pemeriksaan kode IntelliJ dan memvalidasi anotasi untuk menyederhanakan alur kerja coding Anda.

Untuk informasi selengkapnya, lihat [Menyempurnakan kode dengan pemeriksaan lint](#).

## **Anotasi di Android Studio**

Android Studio mendukung anotasi variabel, parameter, dan nilai kembalian untuk membantu Anda merekam bug, seperti pengecualian pointer null dan konflik jenis resource. Android SDK Manager mengemas library Support-Annotations di Android Support Repository untuk digunakan dengan Android Studio. Android Studio memvalidasi anotasi yang sudah dikonfigurasi selama pemeriksaan kode.

Untuk penjelasan selengkapnya tentang anotasi Android, lihat [Menyempurnakan pemeriksaan kode dengan anotasi](#).

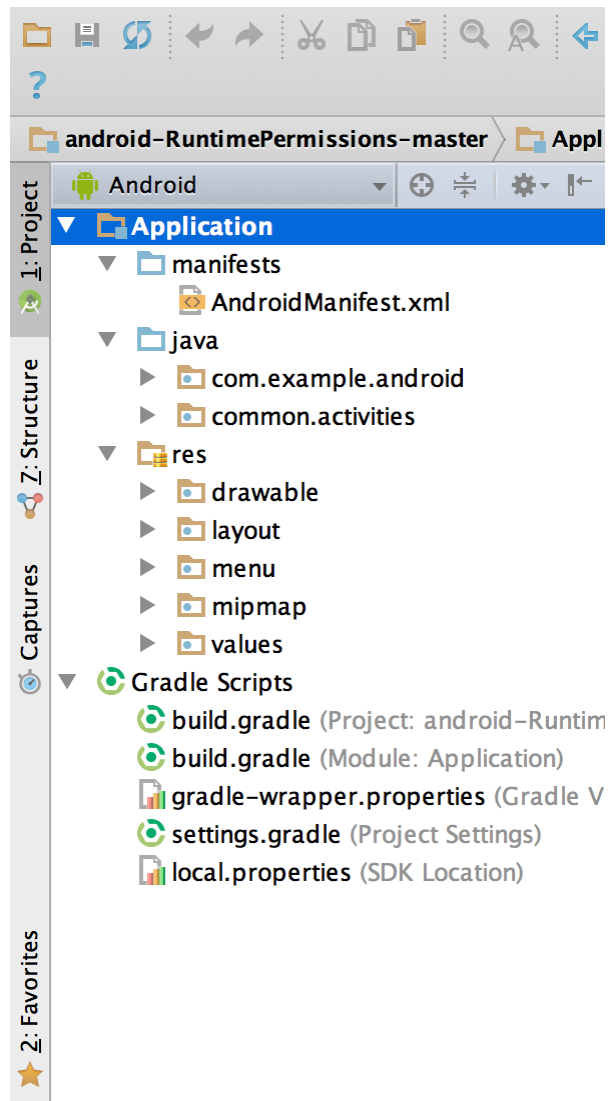
## **Pesan log**

Pada saat membangun dan menjalankan aplikasi dengan Android Studio, Anda dapat melihat entri log dan log perangkat iklan (logcat) dengan mengeklik Android Monitor di bagian bawah jendela.

Apabila Anda ingin men-debug aplikasi Anda dengan Android Device Monitor, Anda dapat meluncurkan Monitor Perangkat dengan mengklik **Tools> Android > Android Device Monitor**.

**Monitor Perangkat** adalah tempat Anda dapat menemukan perangkat DDMS lengkap untuk membuat profil aplikasi Anda, mengendalikan perilaku perangkat, dan banyak lagi. Ini juga mencakup alat Hierarchy Viewer untuk membantu mengoptimalkan tata letak Anda.

## Struktur project



**Gambar 1.** File project dalam tampilan Android.

Setiap project di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file resource. Jenis modul meliputi:

- Modul aplikasi Android
- Modul library
- Modul Google App Engine

Secara default, Android Studio menampilkan file project Anda dalam tampilan project Android, seperti yang ditunjukkan pada gambar 1. Tampilan ini disusun menurut modul untuk memberikan akses cepat ke file sumber utama project Anda.

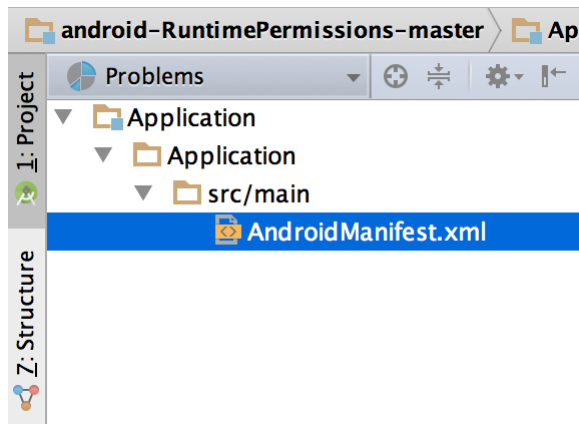
Semua file build terlihat di tingkat teratas di bagian **Gradle Script** dan setiap modul aplikasi berisi folder berikut:

- **manifes**: Berisi file `AndroidManifest.xml`.
- **java**: Berisi file kode sumber Java, termasuk kode pengujian JUnit.
- **res**: Berisi semua resource non-kode, seperti tata letak XML, string UI, dan gambar bitmap.

Struktur project Android pada disk berbeda dengan representasi tersatukan ini. Untuk melihat struktur file project sebenarnya, pilih **Project** dari menu drop-down **Project** (pada gambar 1, ditampilkan sebagai **Android**).

Anda juga dapat menyesuaikan tampilan file project untuk berfokus pada aspek spesifik dari pengembangan aplikasi Anda. Misalnya, memilih tampilan **Problems** pada project Anda akan menampilkan link ke file sumber yang berisi error coding dan sintaks yang

dikenali, seperti tag penutup elemen XML yang tidak ada dalam file tata letak.



**Gambar 2.** File project dalam tampilan Problems, menunjukkan file tata letak yang memiliki masalah.

### 3.2.2 Eclipse

#### SEJARAH ECLIPSE





Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platformindependent). Berikut ini adalah sifat dari Eclipse: 1. Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X. 2. Multi-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lain seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya. 3. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi. Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak seperti dokumentasi, pengujian perangkat lunak, pengembangan web, dan lain sebagainya. Pada saat ini, Eclipse merupakan salah satu IDE favorit karena gratis dan open source. Open source berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan membuat komponen yang disebut plugin.

1. Sejarah Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak pengembangan IBM Visual Age for Java 4.0. Produk Eclipse ini diluncurkan oleh IBM pada tanggal 5 November 2001. IBM menginvestasikan US\$ 40 juta untuk pengembangannya. Sejak 5 November 2001, konsorsium Eclipse Foundation mengambil alih pengembangan Eclipse lebih lanjut.

## 1.6 Eclipse

Eclipse adalah sebuah IDE integrated Development Environment untuk mengembangkan perangkat lunak dan dapat dijalankan disemua platform platform independent. Berikut ini adalah sifat dari Eclipse: 1. Multi-platform: target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X. 2. Multi-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti CC++, Cobol, Phthon, Perl, PHP dan lain sebagainya. 3. Multi-role: selain sebagai IDE untuk pengembangan aplikasi, eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya. Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan open source, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan plugin

### *1.6.1 Sejarah Eclipse*

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java 4.0. produk ini diluncurkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US 40 Juta untuk pengembangannya. Semenjak itu konsumen Eclipse foundation mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya.

### *1.6.2 Arsitektur Eclipse*

Sejak versi 3.0 eclipse pada dasarnya merupakan sebuah kernel, yang mengangkat plug-in. Apa yang dapat digunakan didalam Eclipse sebenarnya adalah fungsi dari plug-in yang sudah diinstal. Ini merupakan basis dari Eclipse yang dinamakan Rich Client Platform RCP. Berikut ini adalah komponen yang membentuk RCP: 1. Core platform 2. OSGi 3. SWT standar Widget Toolkit 4. Jface 5. Eclipse Workbench

Sejarah standar Eclipse selalu dilengkapi dengan JDT java Development Tools, Plug-in yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE Plug-in Development Environment untuk mengembangkan plug-in baru. Eclipse beserta plug-in nya diimplementasikan dalam bahasa pemrograman Java. Konsep Eclipse Adalah IDE yang terbuka open, mudah diperluas extensible untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, eclipse tidak saja untuk mengembangkna program java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal plug-in yang dibutuhkan. Apabila ingin menembangkan program CC++ terdapat Plug-in CDT CC++ Development tools. Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-in UML tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat Plug-in sesuai dengan keinginannya.

### *1.6.3 Perkembangan Eclipse*

Sejak tahun 2006, Eclipse foundation mengkoordinasikan peluncuran eclipse secara rutin dan simultan yang dikenal dengan nama Simultaneous release. Setiap versi peluncuran terdiri dari

eclipse platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan dari sistem ini adalah untuk menyediakan distribusi eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan untuk mempermudah deployment dan maintenance untuk sistem enterprise. Adapun versi eclipse yang telah diluncurkan yaitu: 1. Eclipse 3.0 2. Eclipse 3.1 3. Callisto 4. Europa 5. Ganymede 6. Galileo 7. Helios 8. Juno

### 1.7 ADT

Arsitektur Sejak versi 3.0,

Eclipse pada dasarnya merupakan sebuah kernel. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari plug-in yang sudah dipasang (diinstal). Ini merupakan basis dari Eclipse yang dinamakan Rich Client Platform (RCP). Berikut ini adalah komponen yang membentuk RCP: ♣ Core platform ♣ OSGi ♣ SWT (Standard Widget Toolkit) ♣ JFace ♣ Eclipse Workbench Secara standar Eclipse selalu dilengkapi dengan JDT (Java Development Tools), plug-in yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (Plug-in Development Environment) untuk mengembangkan plug-in baru. Eclipse beserta plug-in-nya diimplementasikan dalam bahasa pemrograman Java. Konsep Eclipse adalah IDE adalah 1. terbuka (open), 2. mudah diperluas (extensible) untuk apa saja, dan 3. tidak untuk sesuatu yang spesifik. Eclipse tidak saja untuk mengembangkan program Java, tetapi juga untuk

berbagai macam keperluan. Perluasan apapun cukup dengan menginstal plug-in yang dibutuhkan. Apabila ingin mengembangkan program C/C++ maka telah terdapat plug-in CDT (C/C++ Development Tools) yang dapat dipasang di Eclipse untuk Eclipse menjadi perangkat untuk pengembangan C/C++.

Pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-in UML2 tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat plug-in sesuai dengan keinginannya. Salah satu situs yang menawarkan plug-in yang gratis adalah Eclipse downloads by project. 3. Histori (versi-versi) Eclipse Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama Simultaneous Release. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan sistem ini adalah untuk menyediakan distribusi Eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan untuk mempermudah deployment dan maintenance untuk sistem enterprise, serta untuk kenyamanan. Peluncuran simultan dijadwalkan pada bulan Juni setiap tahunnya.

Instalasi Langkah-langkah instalasi Eclipse adalah sebagai berikut: 1. Memastikan JDK telah terpasang (terinstall) di komputer atau laptop. Homepage Eclipse adalah <http://www.eclipse.org>, dan pemrogram bisa mendownload Eclipse IDE (Integrated Development Environment).

2. File Eclipse yang telah diunduh (download) berbentuk (.zip). Pemrogram perlu mengekstrak menjadi folder. File hasil ekstrak disimpan di direktori C:/Program Files/ atau pada direktori lain. Eclipse Galileo bisa langsung digunakan tanpa memasang melalui proses instalasi terlebih dahulu. Eclipse Galileo tinggal menaruhnya. Kemudian Klik icon eclipse application.

3. Maka akan muncul tampilan awal eclipse seperti dibawah ini.

4. Kemudian muncul Workspace Launcher, yaitu direktori tempat menyimpan project yang dibuat. Maka untuk menyimpan di project di direktori tertentu, pilih Browse ◊ Direktory tertentu (Misal, E:\PROJECT\Project). Jika ingin direktorinya default maka, checklist Use this as default and not ask again

5. Selanjutnya akan tampil jendela seperti berikut ini.

Membuat Project Baru : Klik menu File ◊ New ◊ Project.  
Gambar

6 Buat Project Baru

7. Maka akan tampil seperti ini, kemudian Pilih Java Project, klik Next.

8. Isi Project name dengan nama Z2S, Klik Next.

9. Muncul tampilan seperti ini, kemudian Klik Finish. Klik Yes ketika eclipse menanyakan apakah anda mau mengganti view ke Java Perspective.

10. Tampilan jendela kerja Eclipse

**11.** Selanjutnya pada Project Z2S, terlebih dahulu buat package, yaitu Klik kanan Z2S ◇ Klik New ◇ Klik Package.

**12.** Pada Java Package, Name diisi dengan z2s.a000.u001.template ◇ Klik Finish

**13.** Selanjutnya buat Kelas baru di package z2s.a000.u001.template, yaitu Klik kanan z2s.a000.u001.template ◇ New Class

**14.** Pada Java Class, Name diisi dengan Template ◇ Klik Finish

**15.**

### **3.2.3 Ionik Framework**



#### **Pengenalan Ionic Framework**

Apa itu Ionic Framework ?

pertanyaan ini mungkin yang terbesit dalam hati dan pikiran anda ketika mendengar Tentang Ionic Framework. ionic framework merupakan platform yang digunakan untuk mengembangkan aplikasi mobile hybrid. maksud dari mobile hybrid disini adalah aplikasi mobile yang sekali dikembangkan bisa digunakan untuk

beberapa platform contohnya ios, dan android. ionic ini dibangun menggunakan bahasa pemrograman basis web (HTML5, JavaScript). ionic sendiri di targetkan untuk programmer web agar bisa mengembangkan aplikasi mobile menggunakan teknologi web.

ionic menggunakan node.js, Angular, HTML5, SASS, sebagai engine untuk membangun aplikasi mobile. berikut saya akan membahas beberapa kelebihan dari ionic framework. kelebihan ionic framework

1. opensource. ionic menggunakan lisensi opensource sehingga anda lebih mudah mengembangkan aplikasi mobile yang free atau komersial dengan menggunakan ionic ini
2. ionic framework ini dikenal lebih cepat dibandingkan dengan beberapa framework lain.
3. target ionic untuk android 4+ dan ios 7+. artinya bagi yang suka mengupdate perkembangan teknologi, anda tidak perlu khawatir akan kompatibilitas dengan sistem operasi yang lama. karena itu ionic sangat cocok untuk anda yang mengembangkan aplikasi mobile versi terbaru.

### **Kelebihan Ionic Framework**

Jadi jika anda punya website atau WebApp dan ingin dibuatkan aplikasi mobilya, Ionic Framework adalah pilihan yang tepat. Terus apa bedanya Ionic dengan Framework lain? cek disini

### **Ionic Menggunakan lisensi Opensource**



Ionic platform menggunakan lisensi opensource, anda boleh membuat aplikasi free ataupun comercial dengan Ionic.

### **Menggunakan teknologi web terbaru**

Ionic memanfaatkan AngularJS untuk implementasi logicnya. Mengapa tidak memakai jQuery? karena jQuery terkenal lambat di mobile sedangkan Angular menawarkan performa dan respon cepat serasa aplikasi native.

### **Target Hanya untuk Android 4 dan ios 7 Keatas**

Anda bisa melihat ini sebagai kelebihan ataupun kekurangan. Bagi yang suka dengan hal baru dan ingin kode yang bersih tanpa memikirkan kompatibilitas dengan versi mobile OS lama, maka ionic adalah platform yang paling pas.

### **Berbasis Apache Cordova /Phonegap**

Perhatikan point ini. Ionic hanya menyediakan frameworknya, untuk membungkusnya menjadi aplikasi Android atau iOS anda tetap pakai phonegap. Artinya bagi para programmer phonegap dengan platform lain, keahliannya tetap bisa dipakai.

Tool Resmi Ionic

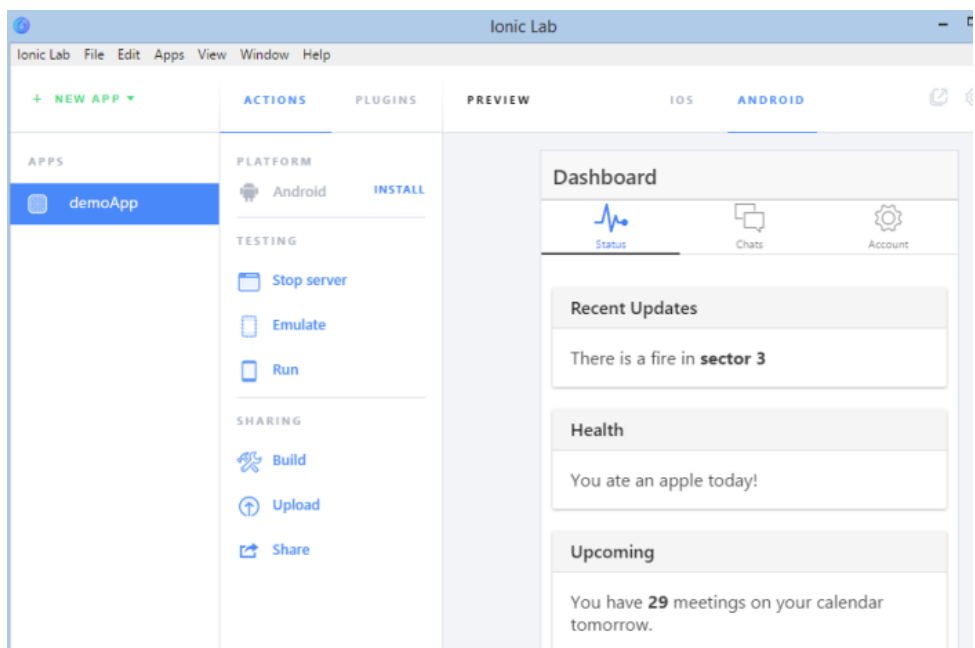
### **Ionic CLI**

Ionic CLI adalah tool command line yang digunakan untuk membuat project dan menjalankan project. Masih banyak fitur dari

ionic CLI ini, silahkan mengacu ke <http://ionicframework.com/docs/cli/start.html>.

## ionic Lab

IonicLab Adalah solusi bagi anda yang ingin kenyamanan. IonicLab adalah Versi GUI dari Ionic CLI, anda tinggal klik untuk melakukan operasi operasi yang semula harus mengetikan perintah di Console.



## Ionic Lab

## Ionic Creator

Berbeda dengan jQuery mobile atau platform lain yang memaksa kita coding manual untuk mendesain tampilan UInya. Ionic menyediakan Ionic Creator, produk free yang akan memudahkan anda mendesain tampilan UI dari aplikasi. Ionic Creator sendiri

adalah WebApp artinya anda harus mengaksesnya lewat Browser. Untuk Akun Free kita hanya bisa mendesain satu aplikasi saja, Jika ingin lebih belilah paket Premium. Anda bisa mencobanya di <https://creator.ionic.io/app/login>.

#### Ionic ViewApp

Aplikasi ini adalah Aplikasi Android. Fungsi utamanya adalah mempreview program Ionic langsung dari Android. Aplikasi yang bisa di preview oleh ViewApp bisa aplikasi yang kita buat atau aplikasi orang lain yang dishare di Ionic Market.

#### Apa yang harus dipelajari terlebih dahulu

Setelah kita tahu teknologi kelebihan ionic diatas, ada beberapa teknologi yang harus ada kuasai, minimal anda tahu sebelum terjun ke Ionic. Ilmu ini harus anda pahami kalau ingin membuat aplikasi bagus. Kalau anda tidak pernah mencoba atau mendengar teknologi dibawah ini, jangan buru buru coba ionic, kalau dipaksakan, paling paling anda berhenti ditengah jalan. Teknologi itu adalah

#### HTML5/CSS/Javascript

#### Nama komponen visual di mobile App (android/iOS)

#### AngularJS

#### Dasar dasar NodeJS/NPM

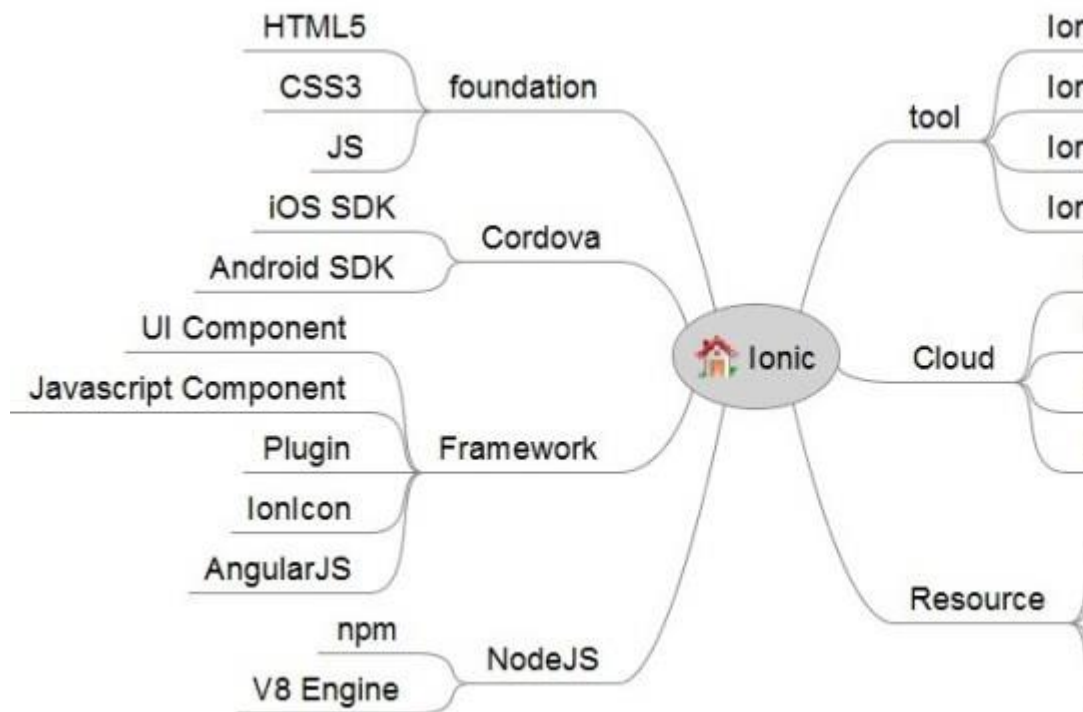
#### Phonegap/Apache Cordova

#### Command line (CMD/Bash/Terminal Linux)

#### Cara membuatApk atau membuild /compile iosApp.

Jadi 7 point itu harus anda pahami dulu, mengapa saya tekankan disini. Untuk point 7, jika anda hanya ingin membuat versi androidnya, maka anda tidak perlu belajar XCode. karena banyak programmer yang berusaha memakai teknologi baru tanpa memperhatikan ilmu dasarnya, makanya jangan heran banyak programmer stress dan banyak mahasiswa mentok di tengah jalan karena tergiur berbagai fitur framework yang ditawarkan.

Selain itu, alangkah baiknya anda memahami Ekosistem Ionic secara menyeluruh, minimal tahu teknologi dibalik ionic agar mudah dalam mempelajari dan medebug Program. Anda bisa melihat Ekosistem Ionic dan Hubungannya dengan teknologi lain dari diagram dibawah ini.



Jika anda tidak ingin memahami semuanya, saya sarankan fokuslah ke teknologi AngularJS karena jika diibaratkan HTML5,CSS3,JS adalah badannya maka angularJS adalah rohnya. Dengan bahasa lain, Framework Ionic (UI component) adalah Viewnya dan AngularJS adalah Controllernya.

### Ionic Platform

Selain Ionic Framework yang free dan opensource, Ionic juga memperkenalkan ionic Platform. Ionic Platform memberikan

layanan tambahan bagi para developer ionic. Ionic Platform menambahkan fitur Create, deploy, update, Analytics, Marketing (Ionic Market) dan Push notification kedalam aplikasi yang dibangun dengan ionic. Saat ini, ionic Platform masih dalam tahap Alpha testing dan anda gratis mencobanya.

## Ionic makes building, deploying, and scaling yo

Focus on what matters. Ionic takes care of the infrastructure. Get up and running instantly with prototyping, packaging, analytics, push notifications, and live updating tools.



CREATE



PACKAGE



ANALYZE



UPDATE

Untuk Platform ini memang masih alfa dan belum ready, namun jika sambutan dari Developer positif, maka kedepannya ionic adalah platform paling menjanjikan untuk membuat aplikasi mobile yang terasa Native untuk Platform Android dan iOS.

### 3.2.4 Reactive

React Native adalah sebuah framework javascript yang di kembangkan oleh facebook dan memungkinkan kita membuat aplikasi mobile android atau ios menggunakan teknologi web. Seperti yang kita ketahui sekarang ini juga sudah banyak framework javascript

yang bisa kita gunakan untuk membuat aplikasi android atau ios, akan tetapi untuk React Native ini berbeda dengan framework - framework javascript yang lainnya. Dengan React Native ini kita tidak membuat aplikasi Hybrid yang dimana aplikasi ini berjalan di javascript runtime akan tetapi kita akan membuat Real Aplikasi dimana untuk android akan di compile di java dan untuk IOS akan di compile di Objective-C.

### **Fitur React Native**

Didalam React Native ini terdapat beberapa komponen yang membuat nya bisa kita gunakan untuk membuat aplikasi mobile cross-platform. Berikut ini adalah beberapa fitur dari React Native ini :

#### ***ReactJS***

Sebelum react native di keluarkan oleh facebook yang pertama di kenalkan adalah ReactJS yaitu framework javascript yang bisa digunakan untuk membuat aplikasi web dengan java script. Jika kamu sudah pernah menggunakan ReactJS maka kamu akan mudah untuk memahami React Native dan bagi kamu yang belum pernah mencoba ReactJS juga tidak masalah karna saya sendiri juga belum pernah coba ReactJS sebelumnya.

## *Native*

Di Android dan IOS memiliki arsitektur yang berbeda, nah disini Native ini yang berfungsi untuk handle komponen dari masing - masing platform. Di native ini nanti kita akan mempelajari beberapa komponen untuk keperluan kita belajar membuat aplikasi dengan React Native.

## *Platform*

Untuk saat ini platform yang sudah terintegrasi adalah Android dan IOS. Mungkin kedepannya bisa lebih dari dua platform ini yang bisa terintegrasi dengan react native ini kita tunggu saja.

## **Kelebihan React Native**

Sebelum kita lanjut lebih dalam untuk coding kita perlu tau dulu sedikit apa sih kelebihan React Native dan kenapa kita harus mencoba membuat aplikasi dengan React Native ini. Nah berikut ini adalah kelebihan kita untuk menggunakan React Native.

## *JavaScript*

Seperti yang sudah kita ketahui beberapa waktu yang lalu di survey stackoverflow JavaScript adalah salah satu bahasa pemrograman yang paling banyak digunakan oleh developer di dunia. Dengan begitu artinya kita sebagai developer akan sangat mudah untuk mempelajari JavaScript ini karna komunitas yang sudah besar dan juga kita mempunyai banyak tempat untuk bertanya kalau kita

mengalami kebingungan. Dan bagi kamu yang sudah berpengalaman dengan web development tentunya sudah akrab dengan javascript ini dan kamu sudah punya dasar untuk mulai membuat aplikasi dengan react native ini.

### ***Code Sharing***

Sebagian codebase yang kamu tulis itu bisa kamu gunakan untuk berbeda platform dimana disini bisa digunakan untuk android dan ios. Dengan begitu untuk hal - hal yang mendasar jika kita memiliki fungsi - fungsi umum yang tidak bersentuhan langsung dengan komponen yang berbeda platform ini kita bisa kita gunakan codebase tersebut. Sebagai contoh semisal kita punya sebuah fungsi untuk mengkonversi currency, maka dengan begitu kita bisa gunakan untuk android ataupun di ios.

### ***Community***

React Native ini cukup diminati oleh para developer di dunia berikut ini adalah screenshot dari repository react native.



# CODEPOLITAN

## **Kelemahan React Native**

React Native sendiri juga memiliki kelemahan berikut ini adalah beberapa kelemahan dari react native

### ***Components***

Untuk mengembangkan aplikasi mobile dengan react native kita harus banyak memahami Components dari masing - masing platform. Dikarenakan android dan ios ini memiliki arsitektur yang berbeda dengan begitu Components Native dari masing - masing platform masih terbatas. Untuk itu jika kita ingin menggunakan

komponen yang belum ada maka kita harus membuatnya sendiri. Dimana dengan begini masih belum bisa dilakukan oleh pemula. Tetapi jangan khawatir sampai saat ini kontributor untuk react native ini sudah mencapai 1,231 orang, artinya dengan begitu perkembangan dari react native hari demi hari akan lebih baik. Atau kalau kamu punya pengalaman yang cukup untuk javascript dan mobile development maka silahkan berkontribusi untuk react native ini.

### ***JavaScript***

Nah java script ini juga menjadi salah satu kelemahan dari react native, dikarenakan java script ini berkembang begitu cepatnya yang mengakibatkan beberapa hal itu tidak bisa dilakukan dengan perubahan yang begitu cepat. Untuk itu kita harus pintar - pintar beradaptasi dengan JavaScript ini.

### ***Framework***

Framework React Native juga menjadi kelemahan karena perubahan yang begitu cepatnya yang mengakibatkan ketika kamu mengupdate react native ke versi yang baru untuk versi yang lama beberapa tidak bisa kamu gunakan. So dengan begitu kamu harus pandai - pandai melakukan manajemen codebase kamu agar tidak berpengaruh banyak ketika ada update dari framework react native ini.

### ***Platform***

Nah buat kamu yang ingin mengembangkan react native untuk ios maka kamu perlu punya macbook so memang karena ekosistem dari ios ini semua harus menggunakan produk dari Apple. Untuk android juga kamu butuh java dan gradle dimana dua hal ini yang menjadi issue dikalangan developer karena kebutuhan devices yang minimal dengan memory 4gb untuk bisa bekerja maksimal. Jadi memang salah satu tantangan jika kamu memutuskan untuk terjun di pengembangan aplikasi mobile kamu harus memiliki devices yang memadai.

### ***Performance***

Untuk performance juga masih menjadi perdebatan dikalangan developer ada yang mengatakan bagus dan ada yang mengatakan kurang bagus juga. Akan tetapi bagi saya untuk react native ini sendiri performancenya cukup lebih baik dibandingkan dengan kita menggunakan framework javascript yang lain karena pada dasarnya dengan menggunakan react native kita membuat aplikasi mobile dengan performance native, akan tetapi di beberapa kondisi react native belum sebaik kita pure menggunakan native.

## **BAB IV      Persiapan**

Nah sekarang kita lupakan tentang kelebihan dan kekurangan dari react native ini dan mari kita mulai untuk belajar bagaimana kita membuat aplikasi mobile dengan react native. Pada tutorial ini kita

akan belajar membuat aplikasi ios dan berikut ini hal - hal yang perlu kita siapkan untuk memulai membuat aplikasi IOS dengan menggunakan react native.

### ***MacBook dan OSX***

Pastikan kamu sudah pakai macbook dan osx karena disini kita akan latihan membuat aplikasi untuk IOS, untuk saat ini saya belum pernah coba untuk menggunakan hackintos. Akan tetapi buat kamu yang belum punya macbook mungkin bisa coba untuk menginstal laptop atau pc kamu sistem operasi hackintos. dan jangan lupa untuk menginstall Xcode karna nantinya kita perlu ios emulator untuk run aplikasi yang kita buat.

### ***NPM dan NodeJS***

Dikarenakan kita menggunakan bahasa pemrograman javascript maka kita perlu nodejs dan npm untuk kebutuhan instalasi dari react native ini.

## **BAB V          Persiapan Instalasi**

Sekarang mari kita lanjutkan dengan melakukan proses instalasi untuk beberapa kebutuhan mengembangkan aplikais dengan reac native ini.

### ***Instalasi Homebrew***

```
/usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

### ***Instalasi Watchman***

```
brew install watchman
```

### ***Instalasi React Native***

```
npm install -g react-native-cli
```

Jangan lupa pastikan kamu juga sudah menginstall **XCode**.

## **BAB VI      Buat Aplikasi Pertamamu**

Sekarang mari kita buat aplikasi IOS pertama kamu dengan mengikuti langkah - langkah berikut ini.

### ***Buat Project Aplikasi***

```
react-native init IOSReactTutorial
```

### ***Jalankan React Native Compiler***

Yang pertama kita masuk ke direktori dari project yang telah kita buat.

```
cd IOSReactTutorial
```

Kemudian jalankan react native packager, dalam pengembangan jangan matikan perintah ini tetap biarkan jalan selama kita dalam proses development.

```
react-native start
```

### ***Jalan Aplikasi Di Emulator***

Sekarang kita jalankan aplikasi yang kita buat di emulator.

```
react-native run-ios
```

Jika kamu berhasil maka akan muncul tampilan seperti berikut ini.



### **Mari Buat Hello Word**

Secara default berikut ini adalah beberapa boilerplate code yang di generate secara default oleh react native.

# CODEPOLITAN

Bisa kita lihat pada gambar di atas secara default react native akan mengenerate folder android dan ios dimana disana lah hasil dari compiler untuk aplikasi yang di jadikan native oleh React Native ini. Untuk di ios kita akan menggunakan file index.ios.js sebagai tempat kita menaruh code - code yang kita buat dengan react native ini. Sekarang mari kita buat aplikasi hello world dengan react native dengan membuat code sepeti berikut ini.

## 3.3 i

<b>BAB VII</b>	<b>F</b>
<b>BAB VIII</b>	<b>GB</b>
<b>BAB IX</b>	<b>H</b>
<b>BAB X</b>	<b>J</b>
<b>BAB XI</b>	<b>G</b>
<b>BAB XII</b>	<b>GF</b>
<b>BAB XIII</b>	<b>G</b>

1. Android studio  
<https://medium.com/@tulisannndeso/pengertian-android-studio-dan-berbagai-macam-bagiannya-a21db3160a6e>  
<https://developer.android.com/studio/intro?hl=id>
2. Eclipse  
<https://wi01.files.wordpress.com/2011/02/pengenalan-eclipse.pdf>  
<https://text-id.123dok.com/document/nq767wdny-sejarah-eclipse-arsitektur-eclipse-perkembangan-eclipse.html>
3. Framework ionic <https://edusoftcenter.com/mengenal-apa-itu-ionic-framework/>  
<http://www.candra.web.id/mengenal-ionic-framework/>