

## 2.1 Sejarah dan Pengertian Android

### 2.1.1 Penjelasan Sejarah Android

Pada awal mulanya, Android Inc merupakan sebuah perusahaan perangkat lunak pendatang baru untuk ponsel atau smartphone yang berdiri pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior di berbagai perusahaan berbasis *Informatics Technology and Communication*, Andy Rubin, Rich Miner, Nick Sears, dan Chris White.

Menurut Andy Rubin, Android Inc didirikan untuk mewujudkan *mobile device* yang lebih memiliki kepekaan terhadap lokasi dan preferensi pemilik. Dengan kata lain, Android Inc memiliki tujuan untuk mewujudkan *mobile device* yang lebih mengerti pemilik/penggunanya.

Untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti/perangkat keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [1].

Android dirilis perdana pada tanggal 5 November 2007, Android bersama *Open Handset Alliance* menyuarakan dukungan untuk pengembangan *open source* pada perangkat *mobile*. Di lain sisi, Google juga merilis kode-kode Android yang berada di bawah lisensi Apache, yaitu sebuah lisensi perangkat lunak dan open platform perangkat seluler.

Pada masa kini, sebagian besar vendor-vendor *smartphone* sudah memproduksi berbagai *smartphone* berbasis Android, seperti HTC, Motorola, Samsung, LG, Oppo, Vivo, Xiaomi, dan beraneka ragam vendor *smartphone* lainnya. Hal ini disebabkan karena Android merupakan suatu sistem operasi yang *open source* sehingga bebas didistribusikan dan digunakan oleh vendor manapun.

### 2.1.2 Pengertian Android



Android merupakan suatu platform perangkat lunak untuk perangkat mobile atau *smartphone*. 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 dapat dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat seluler.

Fitur-fitur pada Android antara lain adalah :

- a) Framework aplikasi, memungkinkan daur ulang dan penggantian komponen.
- b) Browser terintegrasi berbasis engine *Open Source WebKit* yang juga digunakan di browser iPhone dan Nokia S60v3.
- c) Rancangan handset. Platform disesuaikan dengan kebutuhan VGA (*Video Graphics Adapter*) yang lebih besar, library grafik 2D dan 3D yang berdasarkan pada spesifikasi OpenGL ES 1.0 serta layout smartphone yang tradisional.
- d) *Multi-touch*. Android memiliki dukungan bawaan untuk *multi-touch* yang tersedia pada handset terbaru seperti HTC Hero
- e) Dukungan hardware tambahan. Android mendukung penggunaan kamera, layar sentuh, GPS (*Global Positioning System*), pengukur kecepatan, magnetometer, akselerasi 2D bit blits (dengan orientasi hardware, scaling, konversi format piksel) dan akselerasi grafis 3D.

### **2.1.3 Android X**

**AndroidX** merupakan *Android Extension Library* terbaru yang sudah dikenalkan pada tahun 2018 lalu. Untuk saat ini, versi AndroidX yang sudah dapat digunakan publik dan untuk production adalah versi 1.0.0. AndroidX hadir untuk menggantikan library lama android (android.support) pada aplikasi generasi mendatang.

Versi stabil dari AndroidX baru direlease pada September 21, 2018. Kehadiran versi yang sudah siap dipakai publik ini menyebabkan

banyak developer yang mencoba untuk melakukan migrasi terhadap dependency dan plugin yang tersedia agar compatible dengan AndroidX. Contohnya adalah plugin `google_maps_flutter` yang melakukan migrasi pada versi 0.2.0 [18].

AndroidX adalah package yang dirancang ulang untuk membuat nama paket lebih jelas. Jadi mulai sekarang hirarki android hanya untuk kelas default android, yang dilengkapi dengan sistem operasi android dan perpustakaan / dependensi lainnya akan menjadi bagian dari androidx. Jadi mulai sekarang semua perkembangan baru akan diperbarui di androidx [19].

## **2.2 Pengertian Android Studio**



# 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. Google lah yang mengakuisisi Android dan kemudian membuatkan sebuah platform.

Android studio merupakan IDE resmi untuk Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) 3 Edisi. 1 Volume. 1 Agustus 2015 ISSN : 2089-9033 pengembangan aplikasi Android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android [2].

Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools).

Android studio memiliki fitur :

- a. Projek berbasis pada Gradle Build
- b. Refactory dan pembenahan bug yang cepat
- c. Tools baru yang bernama “Lint” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung Proguard And App-signing untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah

f. Didukung oleh Google Cloud Platform untuk setiap aplikasi yang dikembangkan.

### **2.3 Pengertian Java Development Kit (JDK)**

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan JRE adalah sebuah implementasi dari Java Virtual Machine yang benar-benar digunakan untuk menjalankan program java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber compiler java, bundling, debuggers, development libraries dan lain sebagainya [3].

### **2.4 Pengertian JavaScript Object Notation**

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data [JSON] [4].

JSON terbuat dari dua struktur:

- a) Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.
- b) Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence).

## **2.5 Pengertian Google Authenticator**

Google Authenticator adalah autentikator perangkat lunak berbasis autentikator yang mengimplementasikan layanan verifikasi dua langkah menggunakan waktu berbasis [One-time Password Algorithm](#) (TOTP; dispesifikasikan dalam RFC 6238) dan HMAC berbasis [One-time Password algorithm](#) (HOTP; ditentukan dalam RFC 4226) , untuk mengautentikasi pengguna aplikasi seluler oleh Google [5].

Saat masuk ke situs yang mendukung Authenticator (termasuk layanan Google) atau menggunakan autentikator yang mendukung aplikasi pihak ketiga seperti pengelola kata sandi atau layanan hosting file, Authenticator menghasilkan kata sandi satu kali enam hingga enam digit yang harus dimasukkan oleh pengguna selain rincian login mereka yang biasa.

Versi perangkat lunak sebelumnya adalah open-source tetapi rilis berikutnya adalah milik. [6]

## **2.7 Pengertian Firebase Authentication**

Sebagian besar aplikasi perlu mengetahui identitas pengguna. Dengan mengetahui identitas pengguna, aplikasi dapat menyimpan data pengguna secara aman di cloud dan memberikan pengalaman personal yang sama di setiap perangkat pengguna.

Firebase Authentication menyediakan layanan backend, SDK yang mudah digunakan, dan library UI yang siap pakai untuk mengautentikasi pengguna ke aplikasi Anda. Firebase Authentication mendukung autentikasi menggunakan sandi, nomor telepon, penyedia identitas gabungan yang populer, seperti Google, Facebook, dan Twitter, dan lain-lain.

Firebase Authentication terintegrasi erat dengan layanan Firebase lainnya dan sistem ini memanfaatkan berbagai standar industri, seperti OAuth 2.0 dan OpenID Connect, sehingga dapat dengan mudah diintegrasikan dengan backend kustom Anda [7].

Arsitektur Android terdiri dari: Applications, Application Framework, Libraries, Android Runtime dan Kernel Linux. Arsitektur lengkap platform Android dapat dilihat pada Gambar 1.



## **2.8 Pengertian Bahasa Pemrograman Java**

Berikut ini adalah pengertian java menurut beberapa para ahli:

- 1) Java adalah bahasa pemrograman untuk menciptakan isi yang aktif dalam halaman web, juga dapat dijalankan dalam semua komputer [8].
- 2) Java adalah bahasa pemrograman objek murni karena semua kode programnya dibungkus dalam kelas [9].

Java adalah bahasa pemrograman yang berkembang yang berorientas objek. Teknologi object oriented memandang software sebagai sebuah interaksi antarbagian dalam sebuah sistem dan menggambarkan bagian tersebut ke dalam satu objek yang memiliki sifat/property/data dan kemampuan untuk melakukan suatu tugas tertentu.

Java menurut definisi dari Sun adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer standalone ataupun pada lingkungan jaringan [10].

Java dikembangkan pada bulan Agustus 1991, dengan nama semula Oak. Pada Januari 1995, karena nama Oak dianggap kurang komersial, maka diganti menjadi Java. Pada Desember 1998, Sun memperkenalkan nama “Java 2” (J2) sebagai generasi kedua dari java platform. Konvensi nama baru ini diterapkan untuk semua edisi Java yaitu Standard Edition (J2SE), Enterprise Edition (J2EE), dan Micro Edition (J2ME). Ada tiga platform Java yang telah

didefinisikan (Gambar 2.1), yang masing-masing diarahkan untuk tujuan tertentu dan untuk lingkungan komputasi yang berbeda-beda:

- Standard Edition (J2SE): J2SE merupakan inti dari bahasa pemrograman Java. J2SE Didesain untuk jalan pada komputer desktop dan komputer workstations.
  - Enterprise Edition (J2EE): Dengan built-in mendukung untuk servlets, JSP, dan XML, edisi ini ditujukan untuk aplikasi berbasis server.
  - Micro Edition (J2ME): Didesain untuk piranti dengan memori terbatas, layar display terbatas dan power pemrosesan yang juga terbatas.
- Gambar 2.1 Tiga platform Java 2

### **2.8,1 Kelebihan Java**

1. Sederhana dan Ampuh Java dirancang untuk mudah dipelajari, terutama bagi programmer yang telah mengenal C/C++ akan mudah sekali untuk berpindah ke Java.
2. Pemakai dapat belajar membuat program dengan Java secara cepat jika telah memahami konsep dasar pemrograman berorientasi objek.
3. Java tidak memiliki hal-hal yang mengejutkan dan aneh. Java memberi anda kemampuan untuk menuangkan semua ide, karena bahasa pemrograman 5 ini bukan merupakan scripting language (bahasa naskah) yang menghilangkan kemampuan kita untuk berinovasi, tetapi dengan cara berorientasi objek yang mudah dan jelas.
4. Aman Java dirancang sebagai bahasa pemrograman yang handal dan aman. Aplikasi-aplikasi yang dibangun dengan bahasa Java

sangat handal dengan manajemen memori yang bagus. Aplikasi Java juga dikenal sangat secure, yaitu kasus-kasus seperti buffer overflow yang umumnya menjadi lubang keamanan aplikasi-aplikasi berbasis C/C++ tidak terjadi di Java, karena pengaturan securitynya yang bagus. Seperti yang kita ketahui ancaman virus dan penyusup ada dimana-mana, bahkan dokumen word processor dapat mengandung virus. Salah satu prinsip kunci perancangan Java adalah keselamatan dan keamanan. Java tidak pernah memiliki fasilitas dan kemampuan yang tidak aman sampai perlu ditangani secara khusus untuk pengamanannya. Maka karena program Java tidak dapat memanggil fungsifungsi global dan memperoleh akses ke berbagai sumber dalam sistem, terdapat sejumlah pengawasan yang dapat dilakukan oleh program Java yang tidak dapat dilakukan oleh sistem lain.

5. Berorientasi-Objek Paradigma pemrograman berorientasi objek merupakan paradigma pemrograman masa depan. Java merupakan bahasa pemrograman berorientasi objek. Java bukan turunan langsung dari bahasa pemrograman manapun, juga sama sekali tidak kompetibel dengan semuanya. Java memiliki keseimbangan, menyediakan mekanisme peng-class-an sederhana, dengan model antar muka dinamik yang intuitif hanya jika diperlukan.

6. Kokoh Java membatasi anda dari beberapa hal kunci supaya anda dapat menemukan kesalahan lebih cepat saat mengembangkan program. Java langsung memeriksa program saat anda menuliskannya, dan sekali lagi ketika program di jalankan. Karena Java adalah bahasa yang sangat ketat dalam hal tipe data dan deklarasi, banyak kesalahan umum terjadi saat kompilasi. Hal ini akan lebih menghemat waktu jika dibandingkan dengan keharusan

menjalankan program terlebih dahulu dan memeriksa semua bagian program untuk melihat ketidakcocokan dinamis selama program berjalan. Ini adalah contoh di mana Java lebih luwes dan kokoh dari beberapa bahasa lain, tetapi dengan imbalan yang layak untuk kelebihan itu.

7. Interaktif Java memiliki beberapa kemampuan yang memungkinkan program melakukan beberapa hal pada saat bersamaan, tanpa harus kesulitan menangani proses yang akan terjadi selanjutnya. Jalinan program-program Java yang mudah digunakan memungkinkan kita untuk memikirkan pembuatan perilaku khusus, tanpa harus mengintegrasikan perilaku tersebut dengan model pemrograman global yang mengatur perulangan kejadian.

8. Netral Terhadap Berbagai Arsitektur Java telah mengambil beberapa keputusan yang sulit dalam pembuatan bahasa Java dan bagaimana program 6 dijalankan, jadi anda dapat sepenuhnya percaya “tulis sekali, jalan di mana saja, kapan saja, selamanya”.

9. Terinterpretasi dan Berkinerja Tinggi Java dilengkapi keajaiban lintasplatform yang luar biasa dengan kompilasi ke dalam representasi langsung yang disebut kode-byte Java (Java byte-code), yang dapat diterjemahkan oleh sistem manapun yang memiliki program Java didalamnya. Java, bagaimanapun dirancang untuk tetap berkinerja baik pada CPU yang tidak terlalu kuat. Walaupun Java merupakan bahasa terinterpretasi, kodekode Java telah dirancang dengan hati-hati sehingga mudah diterjemahkan ke dalam bahasa asli suatu mesin untuk menghasilkan kinerja yang tinggi. Sistem program Java yang melakukan optimasi tepat waktu tersebut

tidak kehilangan keuntungan dari program yang netral terhadap platform. “lintas platform berkinerja tinggi” bukan sekedar omong-kosong. Dalam aplikasi Java (\*.class) merupakan Java bytecode yang berjalan di atas jvm (Java Virtual Machine), yang kemudian jvm-lah yang akan menginterpretasikan kode-kode tersebut ke kode native atau kode mesin dari arsitektur yang bersangkutan. Hal sangat menarik karena urusan arsitektur mesin bukan jadi masalah bagi programmer tapi menjadi urusan kompiler pada bahasa pemrograman Java.

## 2.9 E-Learning

### 2.9.2 Pengertian E-Learning



Istilah e-Learning mengandung pengertian yang sangat luas, sehingga banyak pakar yang menguraikan tentang definisi e-Learning dari berbagai sudut pandang. Salah satu definisi yang cukup dapat diterima banyak pihak misalnya dari Darin E. Hartley [Hartley, 2001] yang menyatakan: e-Learning merupakan suatu jenis belajar mengajar yang memungkinkan tersampainya bahan ajar ke siswa dengan menggunakan media Internet, Intranet atau media jaringan komputer lain.

LearnFrame.Com dalam Glossary of e-Learning Terms [Glossary, 2001] menyatakan suatu definisi yang lebih luas bahwa: e-Learning adalah sistem pendidikan yang menggunakan aplikasi elektronik untuk mendukung belajar mengajar dengan media Internet, jaringan komputer, maupun komputer standalone.

Definisi lain e-Learning dengan berbagai sudut pandang dapat dipelajari secara lengkap dari:

<http://www.google.com/search?num=30&hl=en&lr=&ie=UTF-8&oe=UTF-8&q=define%3A%20e-learning>

E-Learning atau sistem pembelajaran elektronik merupakan pembelajaran jarak jauh (distance learning) yang memanfaatkan teknologi komputer dan jaringan komputer (internet dan intranet)[11].

Dari puluhan atau bahkan ratusan definisi yang muncul dapat kita simpulkan bahwa sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam proses belajar mengajar dapat disebut sebagai suatu e-Learning.

Aplikasi pada mobile learning mengacu kepada penggunaan perangkat bergerak salah satunya pada telepon gengam yang dapat mengakses materi pelajaran, arahan dan aplikasi yang berkaitan dengan pelajaran kapan dan dimana saja. Pembelajaran dengan mobile learning dapat dijadikan salah satu alternatif untuk memecahkan permasalahan dalam bidang pendidikan, terutama masalah pemerataan akses informasi pendidikan, kualitas konten yang berupa materi pelajaran dengan bentuk teks maupun gambar. Proses yang dilakukan untuk membuat aplikasi berbasis web mobile dapat berperan sebagai aplikasi native dengan perantara PhoneGap yang mengakses sistem secara langsung pada platform Android dapat ditunjukkan pada Gambar 1. Tahap Aplikasi Mobile Learning Pada aplikasi mobile learning terdapat beberapa proses yang dilakukan yaitu download terlebih dahulu aplikasi learning management system (LMS) yang digunakan pada aplikasi ini adalah moodle sebuah program aplikasi yang dapat merubah sebuah media pembelajaran kedalam bentuk web[12].

Selanjutnya menginstal moodle di localhost kemudian membuat user dan menentukan fitur-fitur yang dibutuhkan pada aplikasi e-Learning versi mobile. Setelah selesai melakukan proses instalasi moodle kemudian dapat mengaktifkan web service pada moodle, dimana web service adalah layanan yang memungkinkan dua buah sistem atau lebih yang saling tidak terkait dapat saling berkomunikasi seperti halnya client dan server[13], dalam perancangan ini moodle menyediakan service yang memungkinkan sistem lain dapat memanfaatkan service moodle tersebut sehingga saling berkomunikasi. Web service yang digunakan adalah berbasis

REST untuk kebutuhan sistem yang terdapat pada moodle. Kemudian perancangan web mobile untuk mengakses web service tersebut dari aplikasi berbasis web untuk perangkat mobile. Agar aplikasi moodle berbasis web dapat mengakses fungsi-fungsi internal pada perangkat mobile pada platform Android dan untuk dapat mempermudah penginstalan maka aplikasi web mobile tersebut dicompile dengan menggunakan kerangka kerja/framework PhoneGap yang bertugas untuk menjadikan aplikasi e-Learning berbasis web sebagai aplikasi native sehingga akan dihasilkan aplikasi native untuk Android, seperti yang ditunjukkan pada Gambar 2. Gambar 2. Proses Mengubah Web Application Menjadi Native Application Dengan menggunakan antar muka program (API) yang disediakan oleh PhoneGap sehingga aplikasi berbasis web mobile dapat berperan sebagai aplikasi native dengan perantara PhoneGap yang mengakses sistem secara langsung.

### **2.9.2 E-Learning dari masa ke masa**

Uraian singkat tentang perkembangan e-Learning dari masa ke masa adalah seperti di bawah [Cross, 2002]:

1990: CBT (Computer Based Training) Era dimana mulai bermunculan aplikasi e-Learning yang berjalan dalam PC standalone ataupun berbentuk kemasan CD-ROM. Isi berupa materi dalam bentuk tulisan maupun multimedia (video dan audio) dalam format MOV, MPEG-1 atau AVI. Perusahaan perangkat lunak Macromedia mengeluarkan tool pengembangan bernama Authorware, sedangkan Asymetrix (sekarang bernama Click2learn) juga mengembangkan perangkat lunak bernama Toolbook.



1994: Paket-Paket CBT Seiring dengan mulai diterimanya CBT oleh masyarakat, sejak tahun 1994 muncul CBT dalam Kuliah Umum IlmuKomputer.Com Copyright © 2003-2005 IlmuKomputer.Com 4 bentuk paket-paket yang lebih menarik dan diproduksi secara massal.

1997: LMS (Learning Management System) Seiring dengan perkembangan teknologi internet di dunia, masyarakat dunia mulai terkoneksi dengan Internet. Kebutuhan akan informasi yang cepat diperoleh menjadi mutlak, dan jarak serta lokasi bukanlah halangan lagi. Disinilah muncul sebutan Learning Management System atau biasa disingkat dengan LMS. Perkembangan LMS yang semakin pesat membuat pemikiran baru untuk mengatasi masalah interoperability antar LMS yang ada dengan suatu standard. Standard yang muncul misalnya adalah standard yang dikeluarkan oleh AICC (Airline Industry CBT Committee), IMS, IEEE LOM, ARIADNE, dsb.

1999: Aplikasi e-Learning Berbasis Web Perkembangan LMS menuju ke aplikasi e-Learning berbasis Web secara total, baik untuk pembelajar (learner) maupun administrasi belajar mengajarnya. LMS mulai digabungkan dengan situs-situs portal yang pada saat ini boleh dikata menjadi barometer situs-situs informasi, majalah, dan surat kabar dunia. Isi juga semakin kaya dengan berpaduan multimedia, video streaming, serta penampilan interaktif dalam berbagai pilihan format data yang lebih standard, berukuran kecil dan stabil.

## **2.10 Pengertian Sistem Operasi**

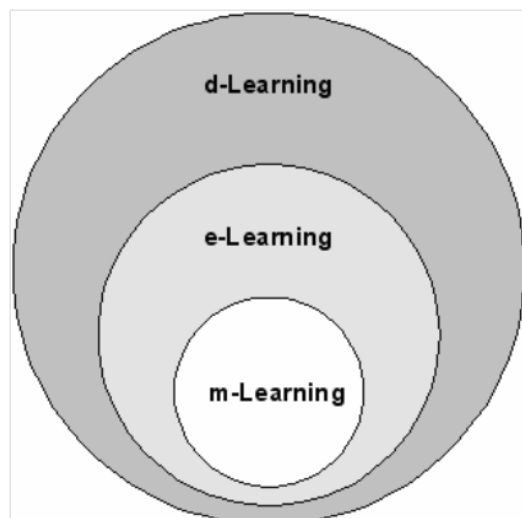
Sistem operasi (*Operating System*) adalah seperangkat program yang mengelola sumber daya perangkat keras komputer, dan menyediakan layanan umum untuk aplikasi perangkat lunak. Sistem operasi adalah jenis yang paling penting dari perangkat lunak sistem dalam sistem komputer. Tanpa sistem operasi, pengguna tidak dapat menjalankan program aplikasi pada komputer mereka, kecuali program aplikasi booting. Sistem operasi mempunyai penjadwalan yang sistematis mencakup perhitungan penggunaan memori, pemrosesan data, penyimpanan data, dan sumber daya lainnya. Untuk fungsi-fungsi perangkat keras seperti sebagai masukan dan keluaran dan alokasi memori, sistem operasi bertindak sebagai perantara antara program aplikasi dan perangkat keras komputer [14], meskipun kode aplikasi biasanya dieksekusi langsung oleh perangkat keras dan seringkali akan menghubungi OS atau terputus oleh itu.

Sistem operasi yang ditemukan pada hampir semua perangkat yang berisi komputer-dari ponsel dan konsol permainan video untuk superkomputer dan server web. Contoh sistem operasi modern adalah Linux, Android, iOS, Mac OS X, dan Microsoft Windows.

## **2.11 Pengertian *Mobile Learning***



Mobile Learning didefinisikan oleh Clark Quinn (2000) sebagai *The intersection of mobile computing and e-learning: accessible resources wherever you are, strong search capabilities, rich interaction, powerful support for effective learning, and performance-based assessment. E-Learning independent of location in time or space.* Merujuk dari definisi tersebut maka, Mobile Learning adalah model pembelajaran yang memanfaatkan teknologi informasi dan komunikasi.



Pada konsep pembelajaran tersebut Mobile Learning membawa manfaat ketersediaan materi ajar yang dapat diakses setiap saat dan visualisasi materi yang menarik. Hal penting yang perlu diperhatikan bahwa tidak setiap materi pengajaran cocok memanfaatkan Mobile Learning[16].

Stevanus Wisnu Wijaya menjelaskan bahwa materi ajar yang tidak cocok mengadopsi konsep Mobile Learning antara lain: materi yang bersifat hands on, keterampilan sebagaimana dokter gigi, seni musik khususnya mencipta lagu, interview skills, team work seperti marketing maupun materi yang membutuhkan pengungkapan ekspresi seperti tarian [15]. Mempertimbangkan hal-hal tersebut di atas maka penerapan Mobile Learning lebih baik pada jenjang pendidikan tinggi.

Mobile learning atau m-learning sering didefinisikan sebagai e-learning melalui perangkat komputasi mobile. Mobile Learning juga merupakan penyampaian bahan pembelajaran elektronik pada alat komputasi mobile agar dapat diakses darimana saja dan kapan saja [17]. Pada umumnya, perangkat mobile berupa telepon seluler digital dan PDA. Namun, secara lebih umum dapat didefinisikan sebagai perangkat apapun yang berukuran cukup kecil, dapat bekerja sendiri, dapat dibawa setiap waktu dalam kehidupan sehari-hari, dan yang dapat digunakan untuk beberapa bentuk pembelajaran. Perangkat kecil ini dapat dilihat sebagai alat untuk mengakses konten, baik disimpan secara local pada device maupun dapat dijangkau melalui interkoneksi. Perangkat ini juga dapat menjadi alat untuk berinteraksi dengan orang lain, baik melalui suara, maupun saling bertukar pesan tertulis, gambar diam dan gambar bergerak.

Mobile learning berhubungan dengan mobilitas pelajar, dalam arti pelajar semestinya mampu terlibat dalam kegiatan pendidikan tanpa harus melakukannya di sebuah lokasi fisik tertentu [20]. Mobile learning dimungkinkan dengan semakin banyaknya perangkat bergerak yang ada seperti smart phone, PC tablet, dan handphone. Dengan demikian, proses belajar dapat bersifat personal dan dapat dilakukan dimana saja dan kapan saja.

Fleksibilitas mobile learning dapat dilakukan dengan teknologi wireless. Mobile learning dapat dijalankan dalam lingkungan pendidikan, di dalam maupun di luar kelas [21]. Yang harus diperhatikan adalah bagaimana pelajar dapat dengan mudah mengakses bahan-bahan kuliah, menghubungi sesama pelajar dan instruktur kapanpun, dimanapun.

Teknologi wireless yang berkembang, baik dari sisi teknologi maupun cakupan luasannya, memungkinkan akses perangkat bergerak terhadap internet semakin mudah. Sebagai contoh jaringan bergerak generasi ketiga (3G) memberikan bandwidth yang lebih lebar yang memungkinkan adanya layanan multimedia dalam lingkungan GSM [22]. Namun kemudahan ini bukan tanpa masalah. Ada kemungkinan perangkat bergerak berada pada blank spot, dimana perangkat bergerak tersebut sama sekali tidak mendapatkan sinyal wireless.

Terdapat tipe aplikasi perangkat bergerak sebagai berikut [23] :

1. Wireless Internet Pada tipe ini, aplikasi yang dibentuk adalah aplikasi internet pada umumnya. Seluruh logika bisnis dan data disimpan pada server. Klien hanya membutuhkan browser untuk mengakses internet.

2. Smart Client Tipe yang kedua adalah tipe smart client, di mana sebagai ganti dari browser, aplikasi dipasang pada perangkat bergerak. Aplikasi memiliki logika bisnis maupun media penyimpanan. Jadi aplikasi dapat dijalankan tanpa jaringan wireless. Jaringan wireless dibutuhkan jika pengguna membutuhkan data yang ada di server.

3. Messaging Tipe aplikasi messaging memiliki banyak macam, dari surat elektronik, pesan peringatan, pesan notifikasi tentang keadaan tertentu sampai dengan pengiriman pesan antar aplikasi.

Aplikasi mobile learning termasuk aplikasi perangkat bergerak dan dapat dibangun berdasar dari tiga tipe aplikasi di atas. Bagian selanjutnya dari artikel ini membahas bagaimana E-Learning diperluas kemampuannya ke dalam M-Learning.

## **2.12 Pengertian Diskusi**

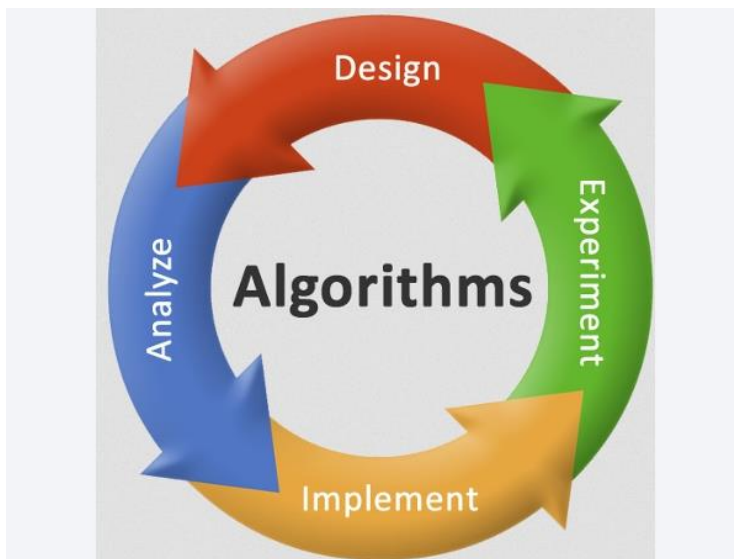
Diskusi merupakan pertemuan ilmiah untuk bertukar pikiran mengenai suatu masalah yang dilakukan oleh sekelompok orang [24].

## **2.13 Pengertian Forum Diskusi**

Forum diskusi online adalah sebuah situs bertukar pikiran secara online dimana orang dapat memasukkan suatu topik dan memulai percakapan dalam bentuk pesan yang diposting. Untuk bisa bergabung dalam pembahasan dan percakapan pada suatu forum pengguna internet harus mendaftar terlebih dahulu [25]. Keanggotaan dan Anonimitas Forum anonim menawarkan anonimitas secara penuh atau anonimitas semu, memperbolehkan pengiriman secara anonim. CAPTCHA, otentikasi id, dan tripcode merupakan cara-cara yang diterapkan untuk mencegah serangan spam pada

forum-forum anonim. Anggota yang terdaftar di forum memiliki hak yang lebih baik, misalnya kemampuan untuk menyunting kirimannya sendiri, memulai topik, mengontrol pengaturan dan akses ke profil pengguna lain. Anggota diidentifikasi dengan nama pengguna (username) yang unik. Profil umumnya memiliki gambar berupa avatar dan blok tanda tangan yang ditambahkan pada akhir setiap kiriman. Anggota juga memiliki hak untuk mengirim pesan pribadi ke pengguna lainnya. Terkadang anggota terdaftar juga memiliki hak untuk menghapus kiriman sebelumnya dan menutup topik yang dibuatnya [26].

#### 2.14 Pengertian Algoritma



Algoritma adalah metode efektif yang diekspresikan sebagai rangkaian terbatas. Algoritma juga merupakan kumpulan perintah untuk menyelesaikan suatu masalah dimana masalah di selesaikan di tuntut secara sistematis, terstruktur dan logis. Dalam penyelesaian

lagoritma itu sendiri, tak jarang sering terdapat berbagai masalah yang timbul contohnya struktur yang tidak tepat, algoritma yang tidak logis, serta kebingungan dalam menyelesaikan algortma itu sendiri. Metode yang penulis buat di bernama “EL-Goritma” atau singkatan dari e-learning algoritma. EL-Goritma itu sendiri penulis memasukkan beberapa fasilitas yang nantinya akan menunjang pembelajaran seperti Ive chat, video call, room diskusi, dan uang belajar.Target dari metode ini adalah mampu mempermudah mahasiswa maupun pelajar dalam menyelesaikan tugas-tugas algoritma dengan mendapatkan referensi dan sumber-sumber dari para dosen maupun pengajar lain[27].

Algoritma adalah jantung ilmu komputer atau informatika. Banyak cabang ilmu komputer yang diacu dalam terminologi algoritma. Namun, jangan beranggapan algoritma selalu identik dengan ilmu komputer saja. Dalam kehidupan sehari-haripun banyak terdapat proses yang dinyatakan dalam suatu algoritma. Cara-cara membuat kue atau masakan yang dinyatakan dalam suatu resep juga dapat disebut sebagai algoritma. Pada setiap resep selalu ada urutan langkah-lankah membuat masakan. Bila langkah-langkahnya tidak logis, tidak dapat dihasilkan masakan yang diinginkan. Ibu-ibu yang mencoba suatu resep masakan akan membaca satu per satu langkah-langkah pembuatannya lalu ia mengerjakan proses sesuai yang ia baca. Secara umum, pihak (benda) yang mengerjakan proses disebut pemroses (processor). Pemroses tersebut dapat berupa manusia, komputer, robot atau alatalat elektronik lainnya. Pemroses melakukan suatu proses dengan melaksanakan atau “mengeksekusi”

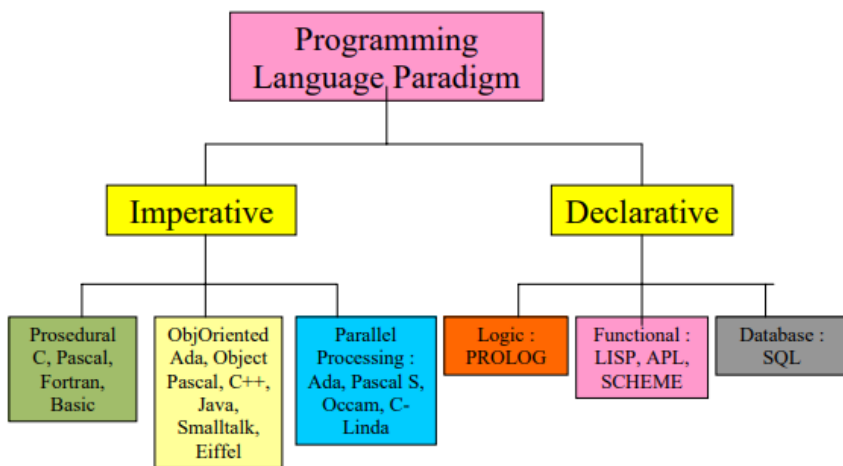


algoritma yang menjabarkan proses tersebut. Melaksanakan Algoritma berarti mengerjakan langkah-langkah di dalam Algoritma tersebut. Pemroses mengerjakan proses sesuai dengan algoritma yang diberikan kepadanya. Juru masak membuat kue berdasarkan resep yang diberikan kepadanya, pianis memainkan lagu berdasarkan papan not balok. Karena itu suatu Algoritma harus dinyatakan dalam bentuk yang dapat dimengerti oleh pemroses. Jadi suatu pemroses harus :

1. Mengerti setiap langkah dalam Algoritma
2. Mengerjakan operasi yang bersesuaian dengan langkah tersebut.

Komputer hanyalah salah satu pemroses. Agar dapat dilaksanakan oleh komputer, algoritma harus ditulis dalam notasi bahasa pemrograman sehingga dinamakan program. Jadi program adalah perwujudan atau implementasi teknis Algoritma yang ditulis dalam bahasa pemrograman tertentu sehingga dapat dilaksanakan oleh komputer.[28]

Sampai saat ini terdapat puluhan bahasa pemrogram. Yang dapat dibedakan berdasarkan tujuan dan fungsinya. Diantaranya adalah :



## 2.15 Git



Git merupakan sistem pengontrol versi yang dikembangkan oleh Linus Torvalds (yang juga menciptakan dan mengembangkan Linux). Sistem pengontrol versi membantu para developer dalam melacak perubahan yang mereka lakukan terhadap basis kode. Tak hanya itu, sistem ini juga mencatat siapa saja yang membuat perubahan serta *me-restore* kode yang telah dihapus atau dimodifikasi.

## 2.15 FireBase

Firestore Realtime Database adalah database yang di-host di cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika Anda membuat aplikasi lintas-platform dengan SDK Android, iOS, dan JavaScript, semua klien akan berbagi sebuah instance Realtime Database dan menerima update data terbaru secara otomatis.

Memiliki kemampuan utama sebagai berikut

**Realtime :** Sebagai ganti permintaan HTTP biasa, Firestore Realtime Database menggunakan sinkronisasi data—setiap kali data berubah, semua perangkat yang terhubung akan menerima update dalam waktu milidetik. Memberikan pengalaman yang kolaboratif dan imersif tanpa perlu memikirkan kode jaringan.

**Offline :** Aplikasi Firestore tetap responsif bahkan saat offline karena SDK Firestore Realtime Database menyimpan data ke disk. Setelah konektivitas pulih, perangkat klien akan menerima setiap perubahan yang terlewat dan melakukan sinkronisasi dengan status server saat ini.

**Dapat Diakses dari Perangkat Klien :** Firestore Realtime Database dapat diakses secara langsung dari perangkat seluler atau browser web; server aplikasi tidak diperlukan. Keamanan dan validasi data dapat diakses melalui Aturan Keamanan Firestore Realtime Database yang merupakan kumpulan aturan berbasis ekspresi dan dijalankan ketika data dibaca atau ditulis.

Menskalakan di beberapa database : Dengan Firebase Realtime Database pada paket harga Blaze, Anda dapat mendukung kebutuhan data aplikasi Anda pada skala tertentu dengan membagi data Anda di beberapa instance database di project Firebase yang sama. Menyederhanakan autentikasi dengan Firebase Authentication pada project Anda dan mengautentikasi pengguna di instance database Anda. Mengontrol akses ke data di tiap database dengan Aturan Firebase Realtime Database khusus untuk tiap instance database.

### Cara Kerja FireBase

Firebase Realtime Database memungkinkan Anda untuk membuat aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke database, langsung dari kode sisi klien. Data disimpan di drive lokal. Bahkan saat offline sekalipun, peristiwa realtime terus berlangsung, sehingga pengguna akhir akan merasakan pengalaman yang responsif. Ketika koneksi perangkat pulih kembali, Realtime Database akan menyinkronkan perubahan data lokal dengan update jarak jauh yang terjadi selama klien offline, sehingga setiap perbedaan akan otomatis digabungkan.

Realtime Database menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga Aturan Keamanan Firebase Realtime Database, untuk menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan Firebase Authentication, developer dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya.

Realtime Database adalah database NoSQL, sehingga memiliki pengoptimalan dan fungsionalitas yang berbeda dengan database terkait. API Realtime Database dirancang agar hanya mengizinkan operasi yang dapat dijalankan dengan cepat. Hal ini memungkinkan Anda untuk membangun pengalaman realtime yang luar biasa dan dapat melayani jutaan pengguna tanpa mengorbankan kemampuan respons. Oleh karena itu, perlu dipikirkan bagaimana pengguna mengakses data, kemudian buat struktur data sesuai dengan kebutuhan tersebut

.

#### Alur Implementasi

1. Mengintegrasikan Firebase Realtime Database SDK :  
Sertakan klien dengan cepat melalui Gradle, CocoaPods, atau skrip.
2. Membuat Referensi Realtime Database
3. Menetapkan Data dan Mendeteksi Perubahan
4. Mengaktifkan Persistensi Offline
5. Melindungi data

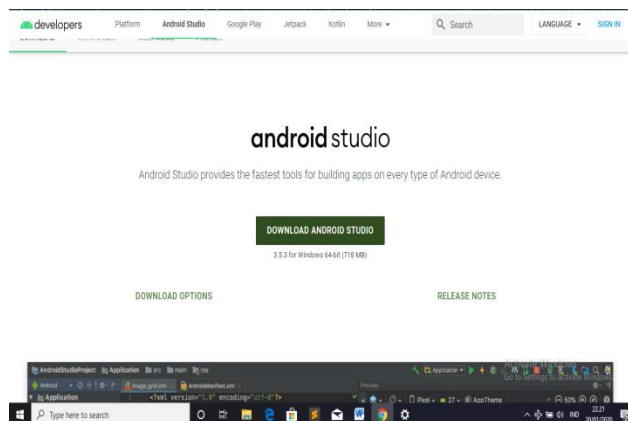
## BAB 2

### Penginstalan

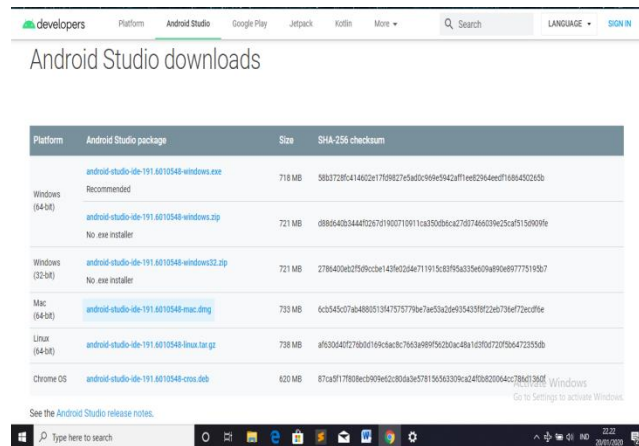
#### Cara penginstallan aplikasi android studio

Untuk menginstall aplikasi android dimana menggunakan bahasa java, kita harus mendownload ide terlebih dahulu, kita menggunakan android studio.

##### 1. Pertama, download aplikasi android studio



2. Lalu pilih download options untuk memilih berdasarkan os yang kalian punya



3. Karna saya menggunakan windows x64. Maka saya akan mendownload yang tersebut.

#### Download Android Studio

Before downloading, you must agree to the following terms and conditions.

8.1.1 If you use any API to retrieve data from Google, you acknowledge that the data may be protected by intellectual property rights which are owned by Google or those parties that provide the data (or by other persons or companies on their behalf). Your use of any such API may be subject to additional Terms of Service. You may not modify, rent, lease, loan, sell, distribute or create derivative works based on this data (either in whole or in part) unless allowed by the relevant Terms of Service.

8.1.2 If you use any API to retrieve a user's data from Google, you acknowledge and agree that you shall retrieve data only with the user's explicit consent and only when, and for the limited purposes for which, the user has given you permission to do so. If you use the Android Recognition Service API, documented at the following URL: <https://developer.android.com/reference/android/speech/RecognitionService>, as updated from time to time, you acknowledge that the use of the API is subject to the Data Processing Addendum for Products where Google is a Data Processor, which is located at the following URL: <https://privacy.google.com/businesses/gdprprocessorterms/>, as updated from time to time. By clicking to accept, you hereby agree to the terms of the Data Processing Addendum for Products where Google is a Data Processor.

9. Terminating this License Agreement

☒ I have read and agree with the above terms and conditions

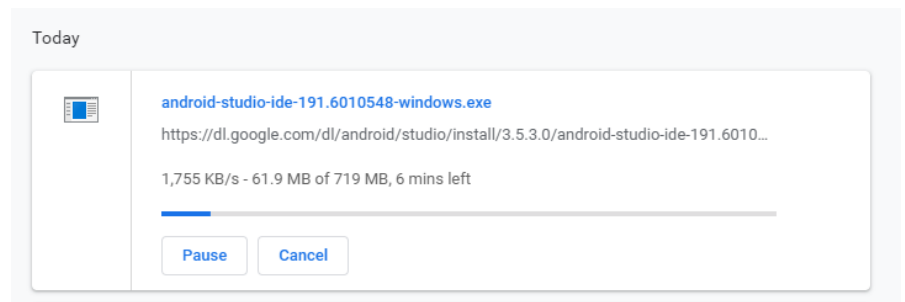
[DOWNLOAD ANDROID STUDIO FOR WINDOWS](#)

android-studio-ide-191.6010548-windows.exe

Activate Windows  
Go to Settings to activate Windows

4. Baca semua terms and condition dan beri tanda ceklis lalu tunggu hingga download yang berjalan hingga

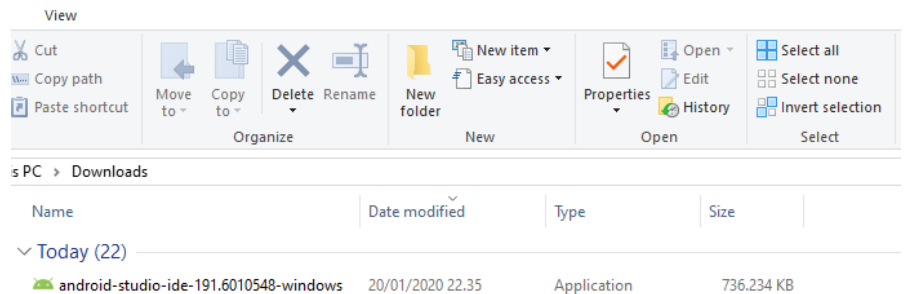
selesai



## 5. Download sudah selesai



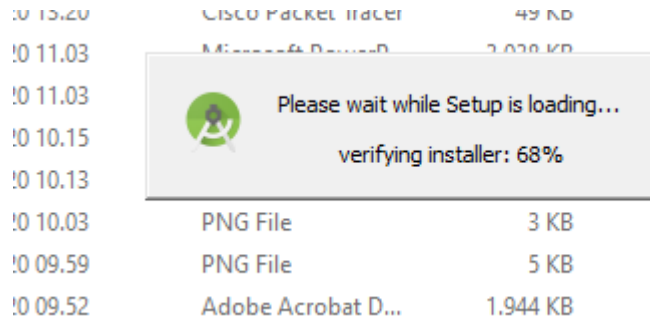
## 6. Masuk ke folder tempat downloadan ada berada, saya menaruh di folder download



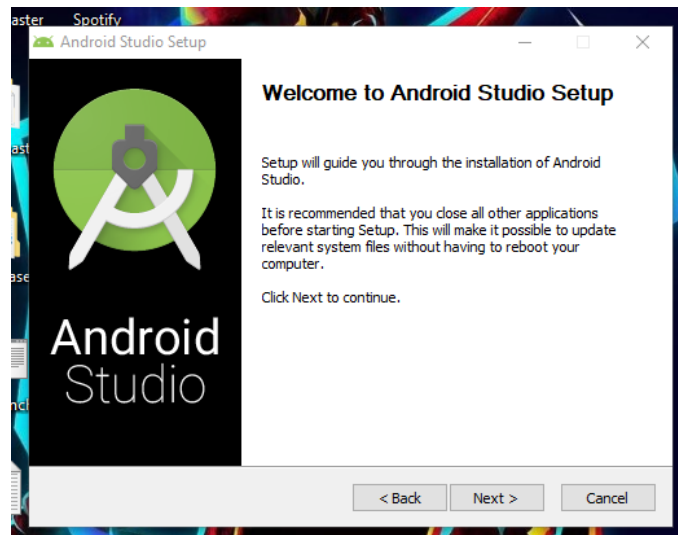
## 7. Lalu buka aplikasi dengan cara klik 2x



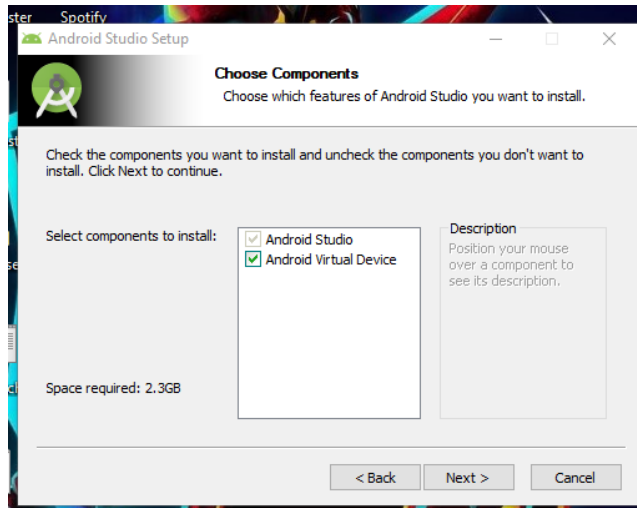
8. Maka akan muncul bahwa sedang melakukan setup



9. Setelah itu akan masuk ke halaman instalasi

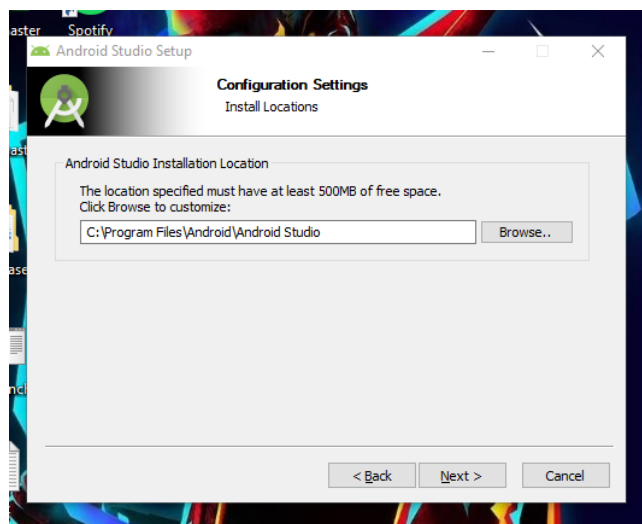


10. Klik next untuk melanjutkan
11. Centang keduanya, untuk menambahkan android virtual device jika menginginkannya.



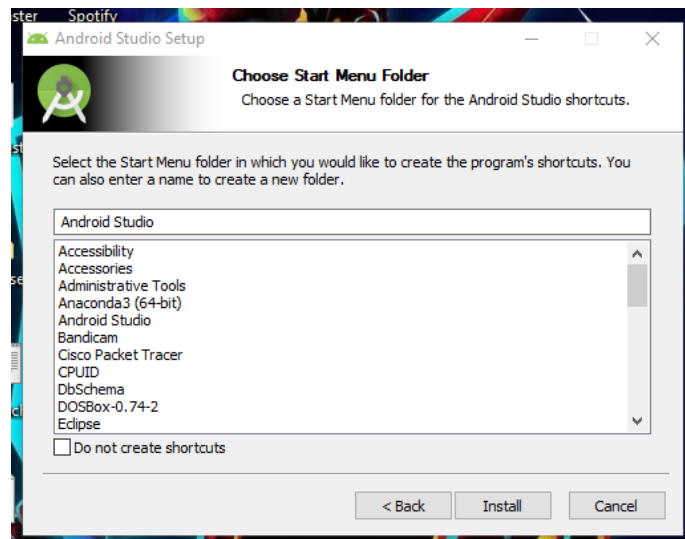
12. Lalu pilih next

13. Lalu pilih lokasi dimana kalian ingin menginstallnya  
pastikan tersedia setidaknya ruang penyimpanan 500 mb.



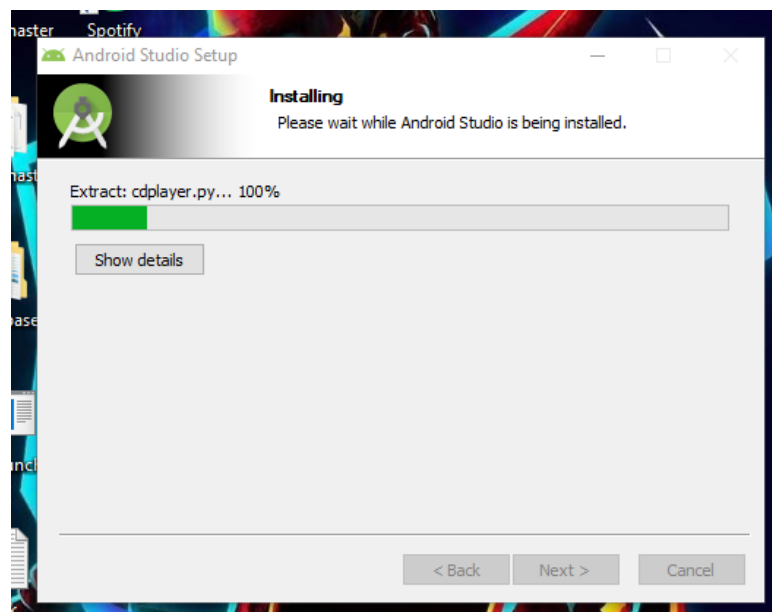
14. Setelah itu pilih next

15. Karna kita akan berfokus maka pilih android

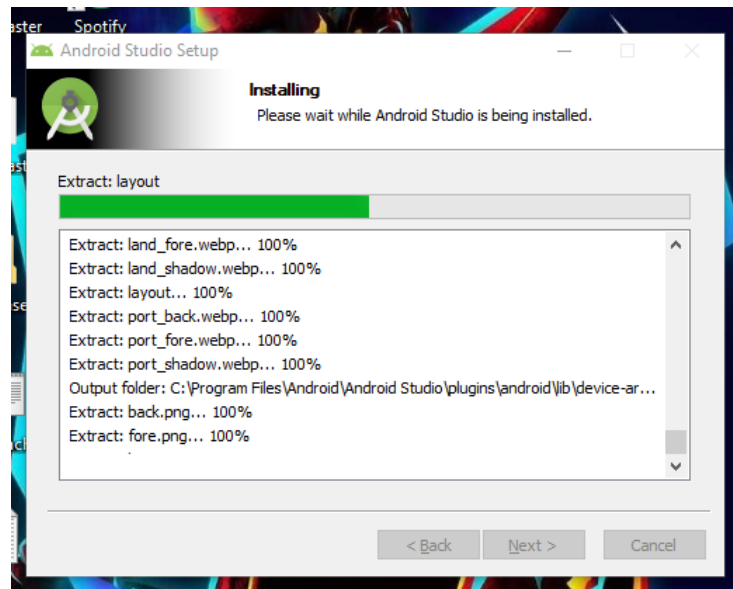


16. Lalu pilih install

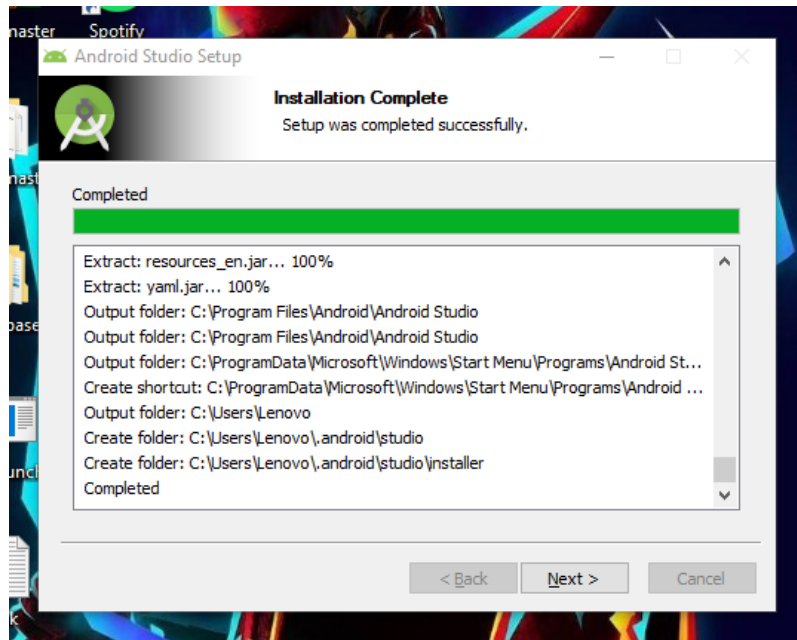
17. Tunggu sampai instalasi selesai



18. Kalian bisa pilih show details untuk melihat lebih detail apa saja yang diinstall



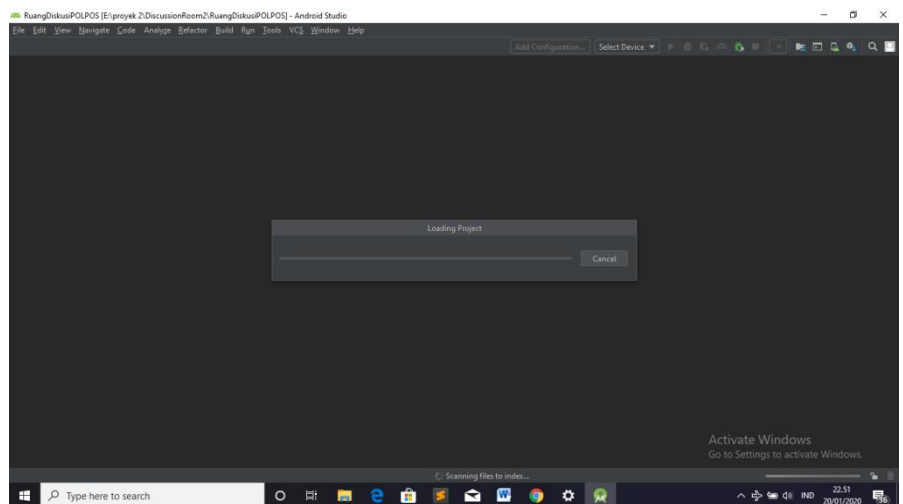
19. Setelah selesai pilih next



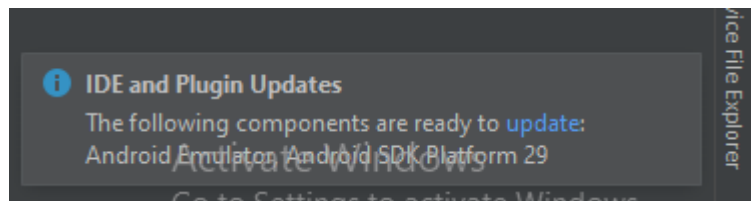
20. Kalian bisa langsung membuka android studio dengan memilih ceklis lalu finish



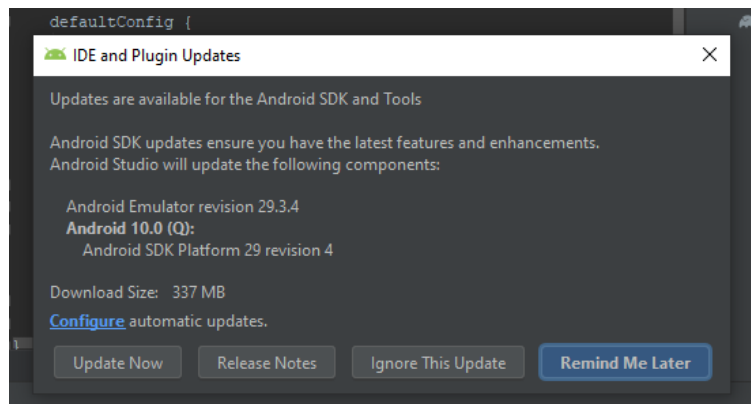
21. Tunggu load hingga selesai



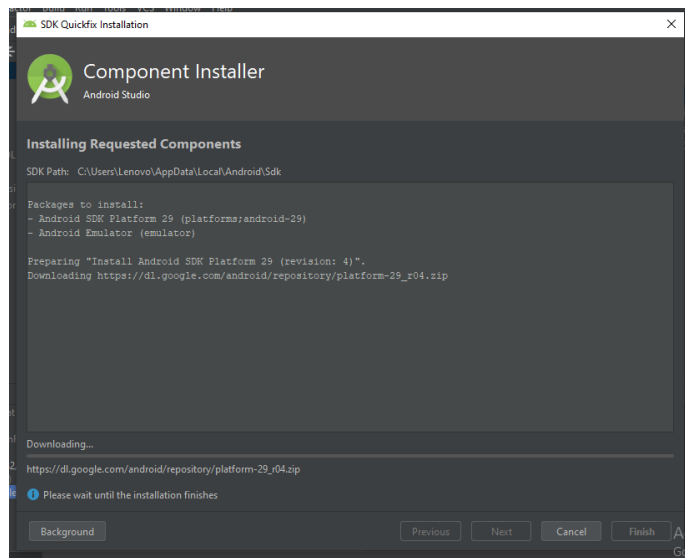
22. Setelah masuk kalian bisa mengupdate gradle dan plug in untuk mendapatkan versi yang terbaru



23. Setelah itu pilih update now



24. Tunggu hingga proses update selesai



25. Setelah selesai maka android studio sudah dapat kita gunakan untuk debugging

## Cara penginstalan java jdk



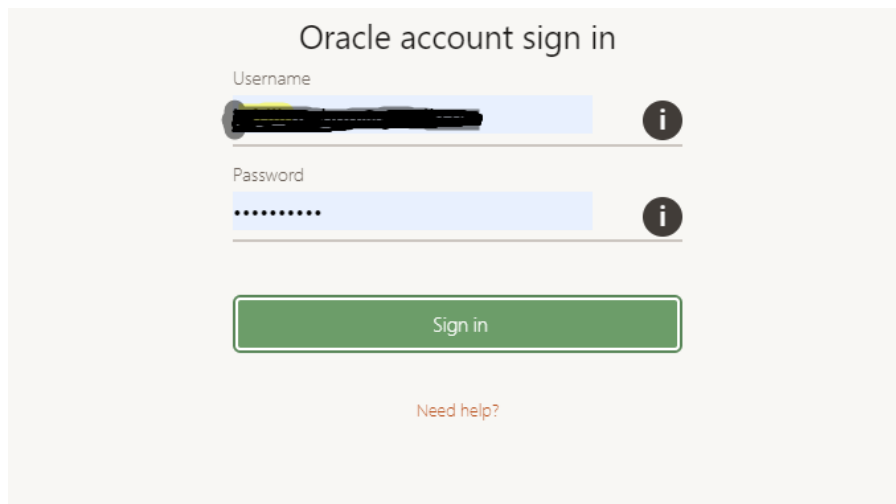
### 1. Download java development kit melalui oracle



### 2. Pilih berdasarkan os yang kalian miliki, karna saya windows maka saya akan memilih windows

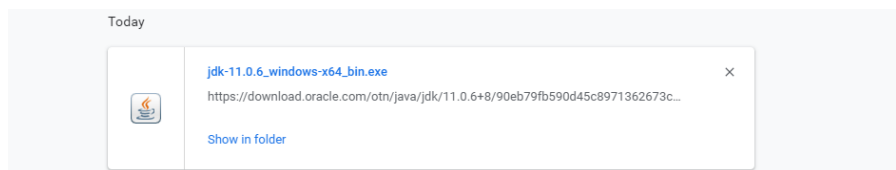


- Setelah memilih maka lakukan login akun oracle anda



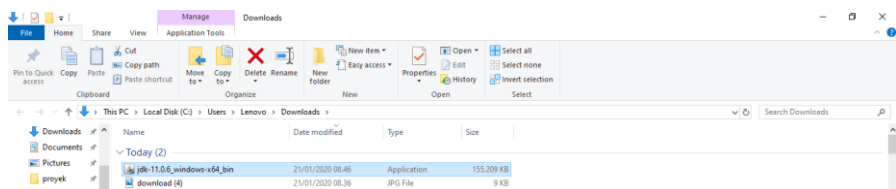
- Setelah login maka akan mulai mendownload

- Tunggu hingga download selesai



- Setelah itu pilih show in folder untuk mengetahui dimana file downloadan ini berada

- Setelah itu kita akan berada di folder tempat file ini berada

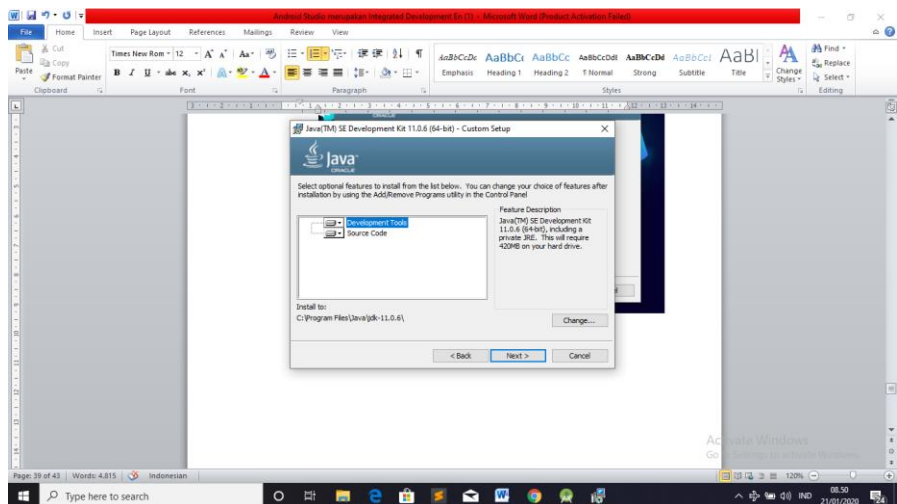


- Setelah itu buka aplikasi untuk menginstall

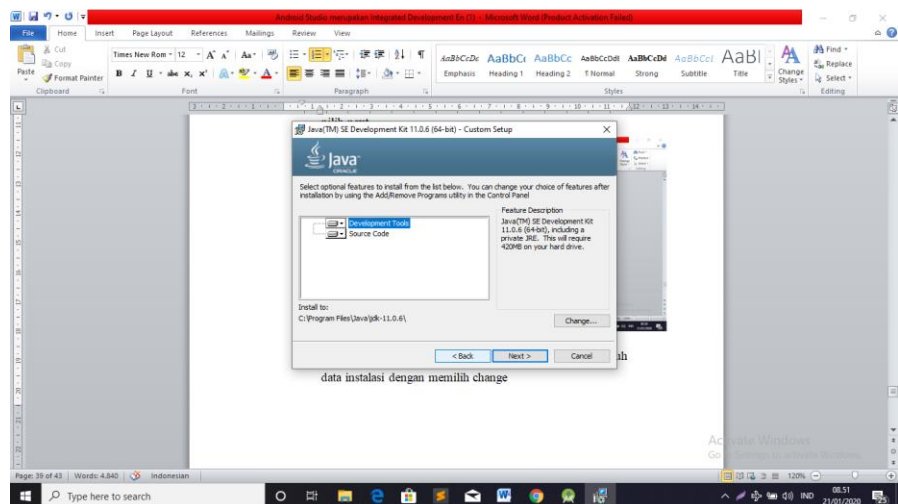
9. Saat masuk ke tampilan awal pilih next



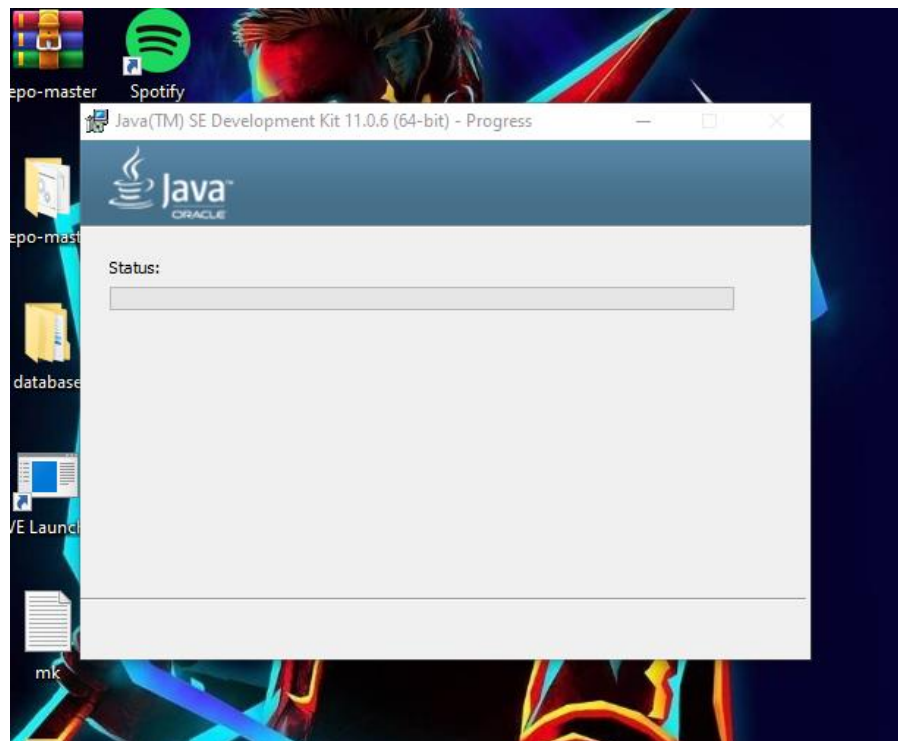
10. Karna kita memerlukan development tools maka pilih itu dan pilih next



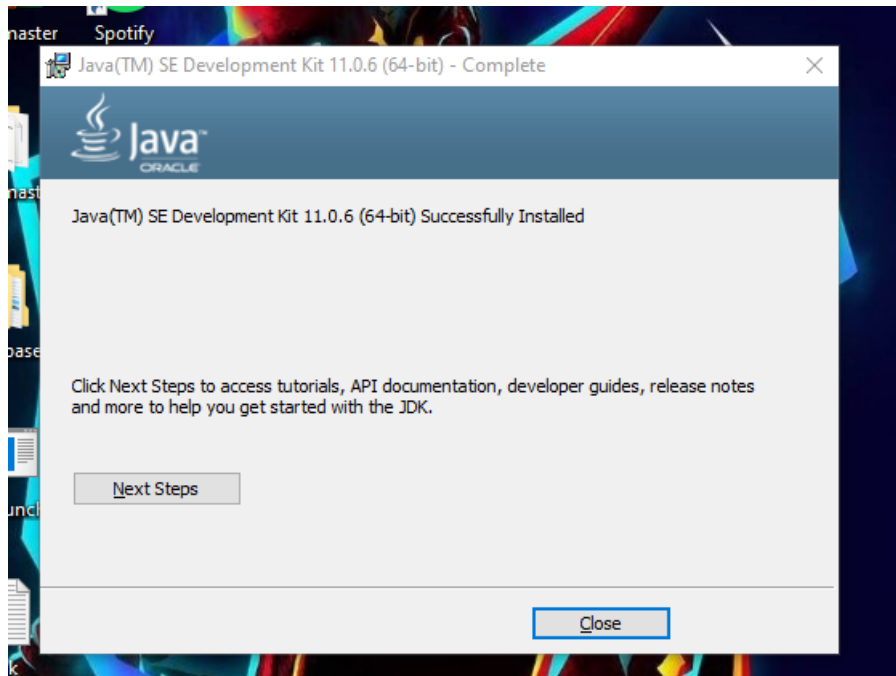
11. Kalian juga bisa menentukan direktori tempat untuk menaruh data instalasi dengan memilih change



12. Tunggu hingga proses selesai

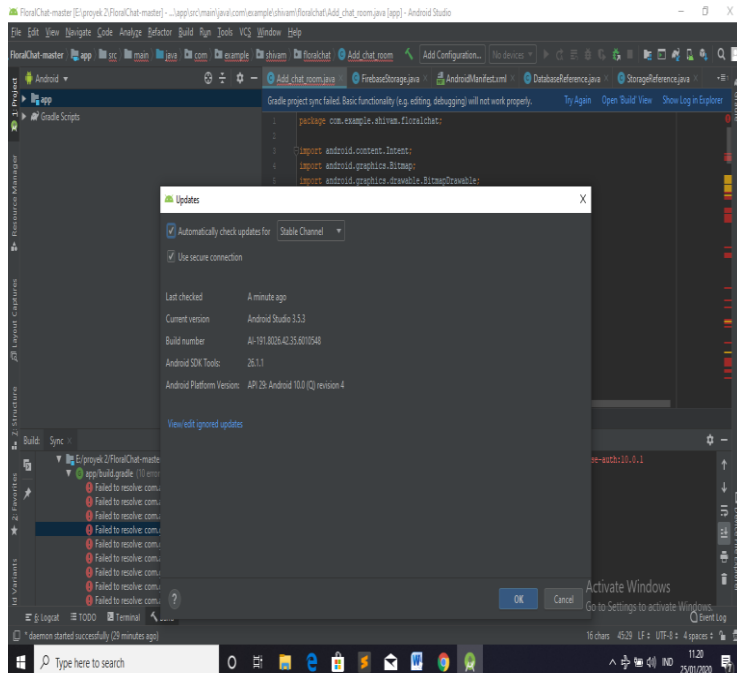


### 13. Java sudah terinstall

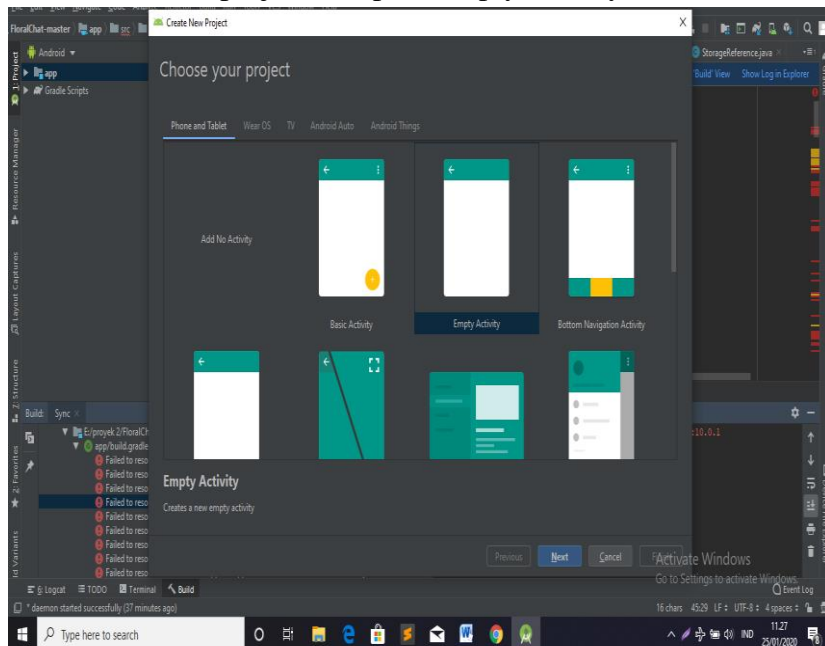


# CARA MEMBUAT LOGIN MENGGUNAKAN GOOGLE AUTHENTICATOR

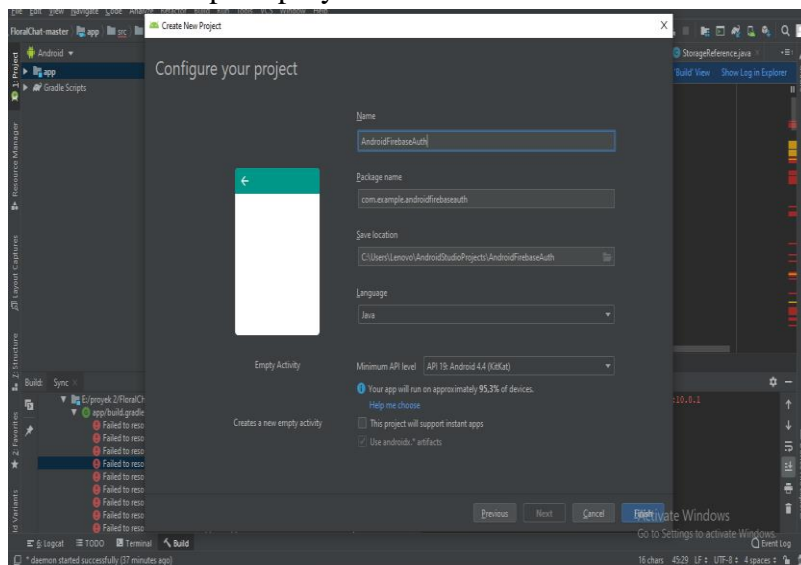
1. Pertama, pastikan terlebih dahulu bahwa Android Studio telah menggunakan versi terbaru 3.3 keatas dan gradle versi terbaru 4.10.1 keatas dan semua library ke versi yang terbaru.



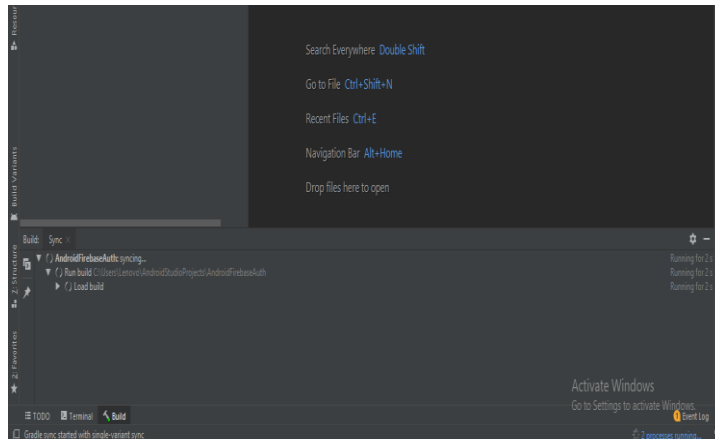
## 2. Lalu create new project lalu pilih empty activity



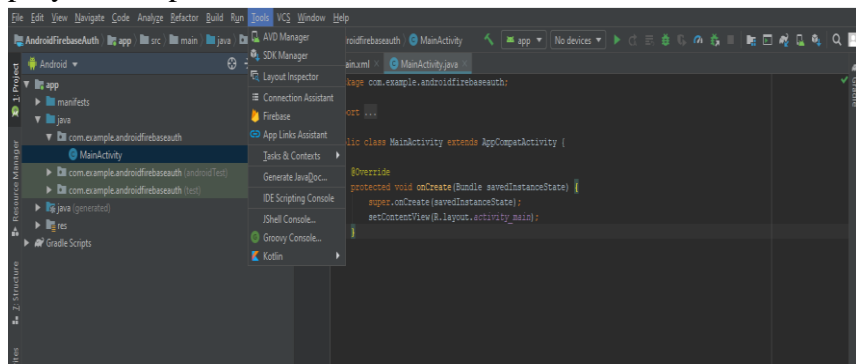
## 3. Lalu beri nama pada proyek kita



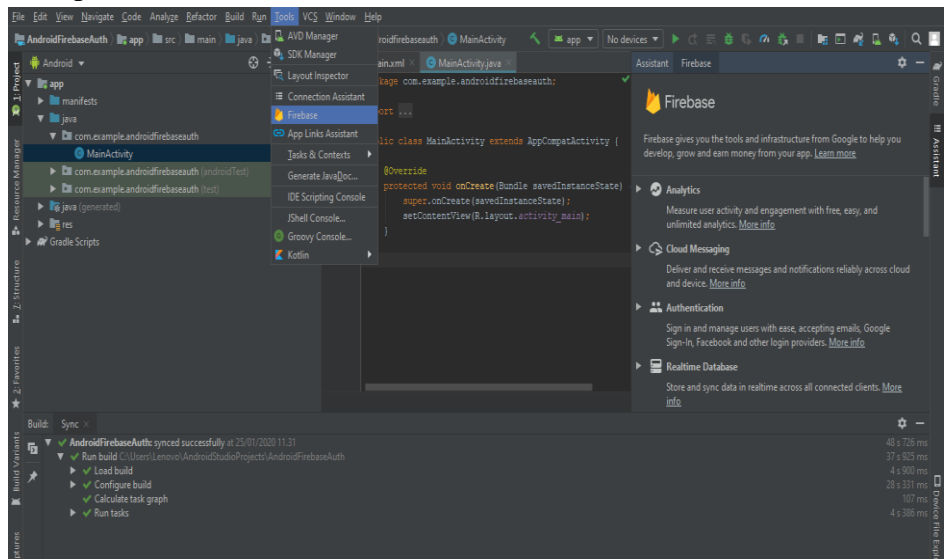
4. Lalu pilih finish, dan tunggu hingga proses syncing selesai



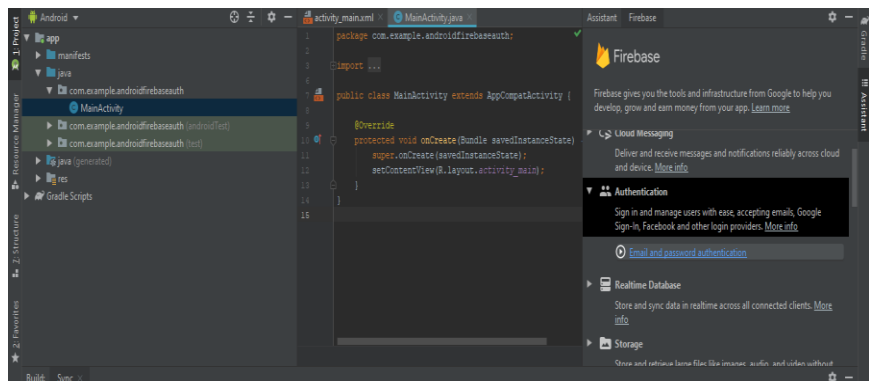
5. Sekarang kita akan menambahkan firebase auth kedalam proyek kita, pilih tools



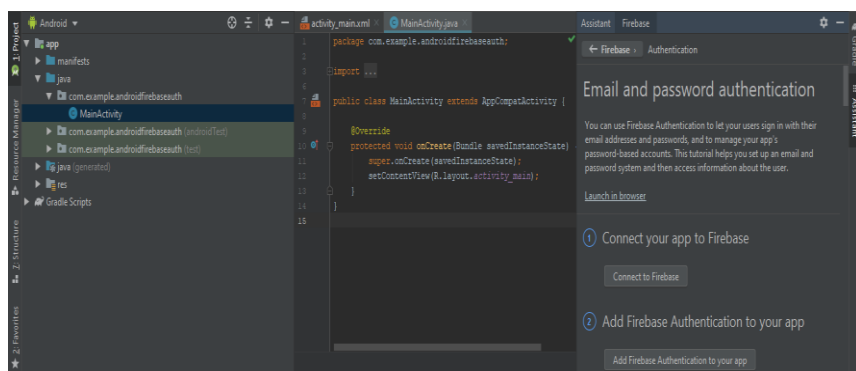
## 6. Lalu pilih Firebase



## 7. Pilih authentication

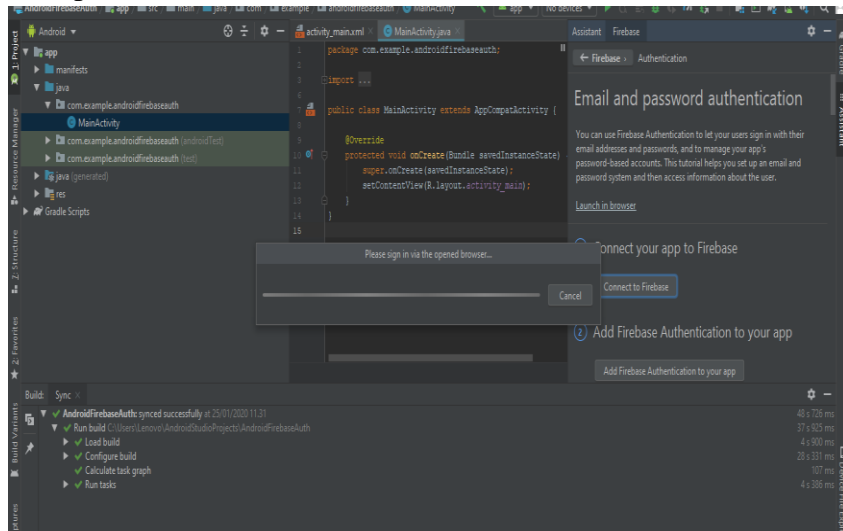


## 8. Pilih Email and Password Authentication

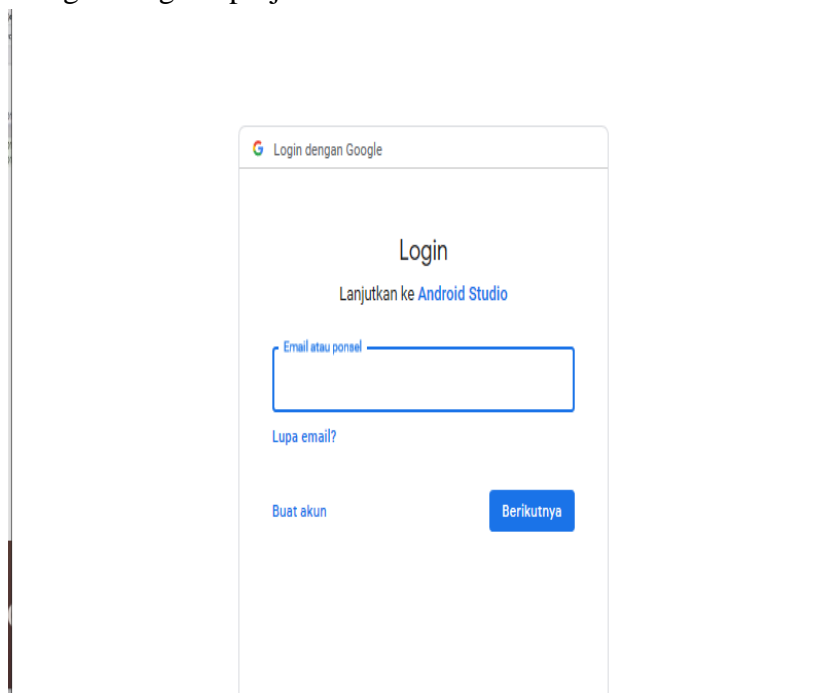




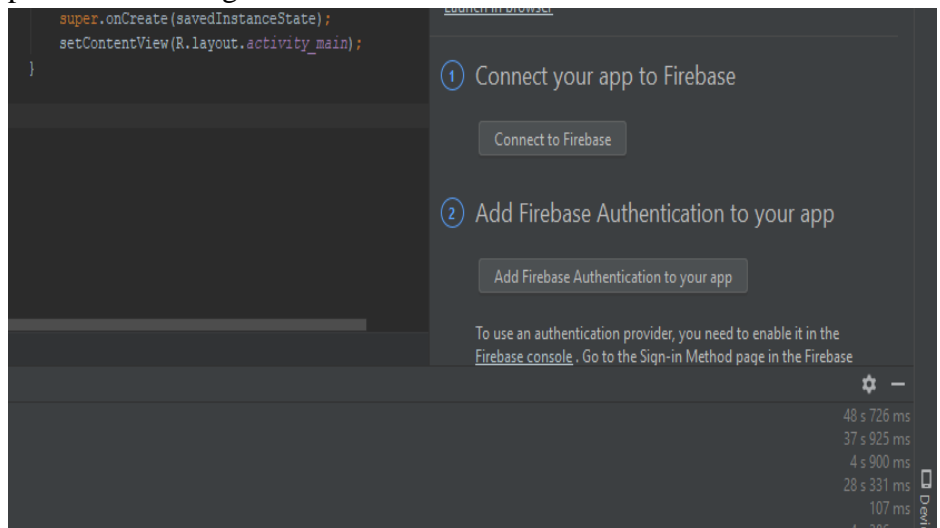
## 9. Lalu pilih connect to Firebase



## 10. Login terlebih dahulu menggunakan gmail untuk menghubungkan proyek ke Firebase

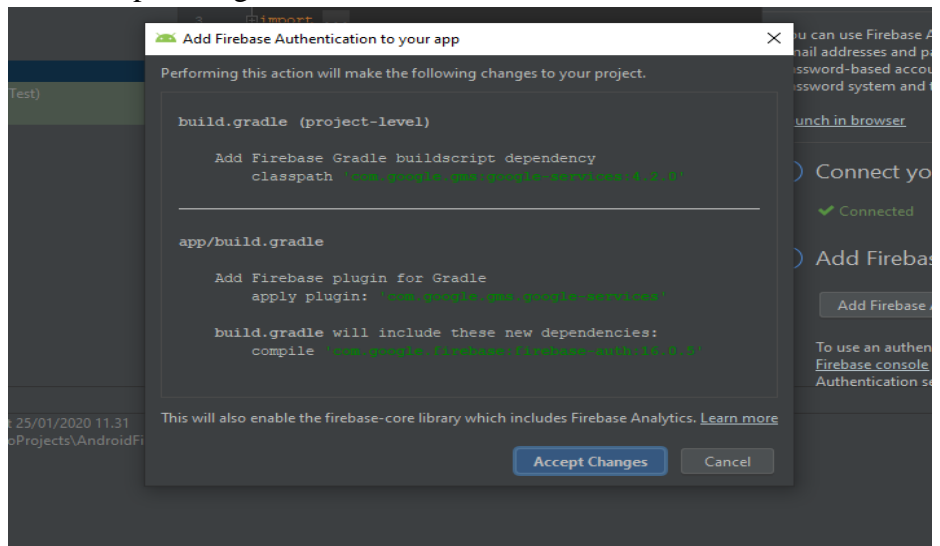


11. Setelah itu pilih connect to Firebase dan tunggu hingga proses connecting selesai

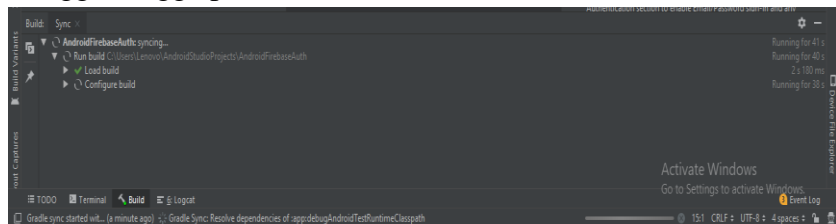


12. Selanjutnya pilih add Firebase Authentication

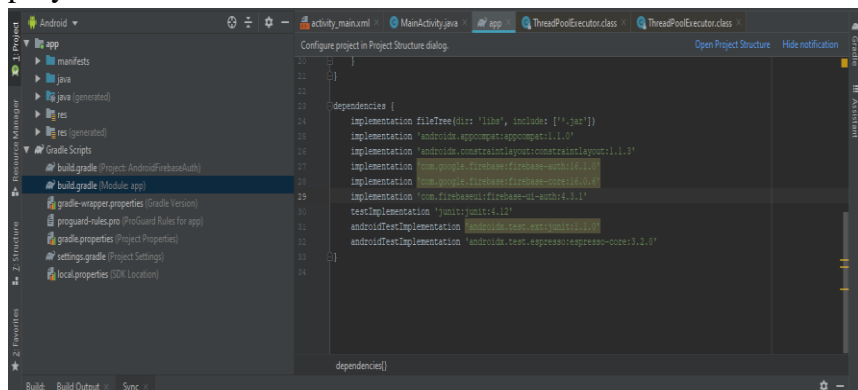
13. Pilih accept Changes



#### 14. Tunggu hingga proses build selesai

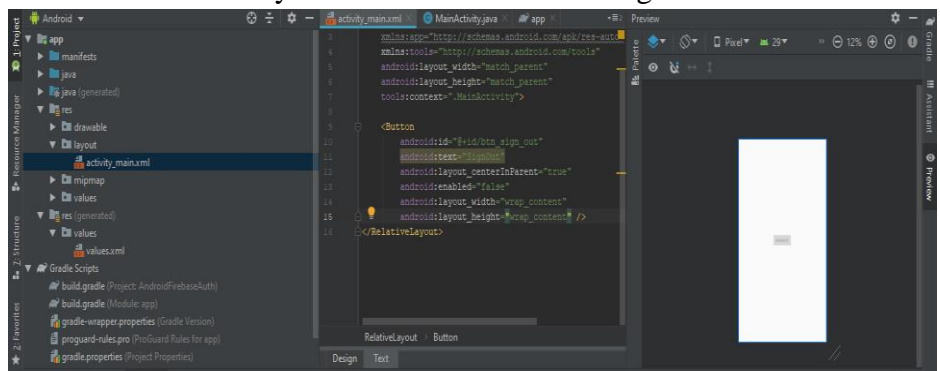


#### 15. Tambahkan Firebase UI untuk mempermudah pembuatan proyek ini



#### 16. Lalu kita akan membuat button sign out

#### 17. Masuk ke activity xml dan buat sebagai berikut



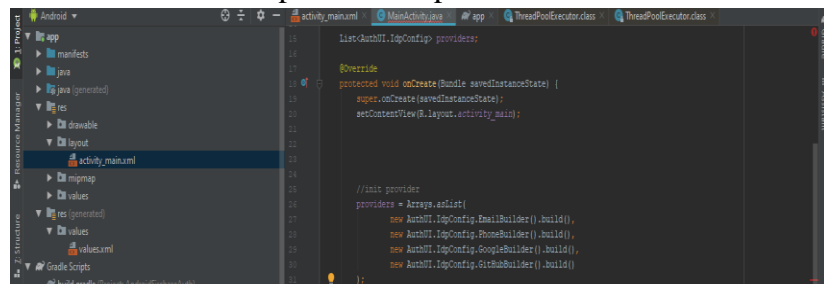
<Button

android:id="@+id/btn\_sign\_out"  
android:text="SignOut"  
android:layout\_centerInParent="true"

```
android:enabled="false"
android:layout_width="wrap_content"
android:layout_height="wrap_content" />
```

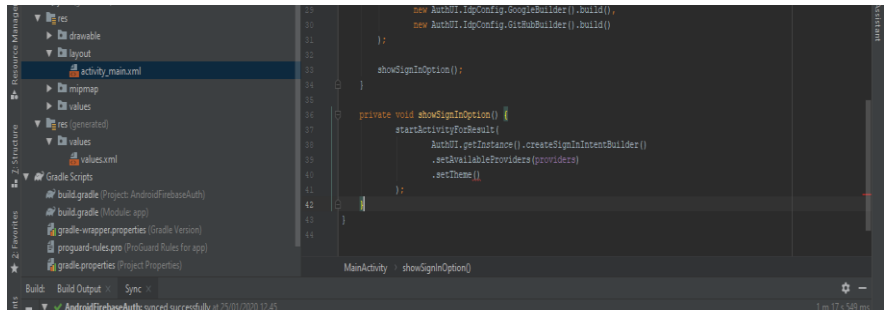
18. Setelah itu panggil sign in activity dari firebaseUI

19. Setelah itu build init providers seperti berikut



```
providers = Arrays.asList(
    new
    AuthUI.IdpConfig.EmailBuilder().build(),
    new
    AuthUI.IdpConfig.PhoneBuilder().build(),
    new
    AuthUI.IdpConfig.GoogleBuilder().build(),
    new
    AuthUI.IdpConfig.GitHubBuilder().build()
);
```

20. Buat method sign in

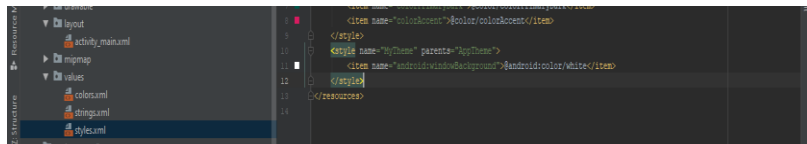


```
showSignInOption();
}
```

```
private void showSignInOption() {
    startActivityForResult(
```

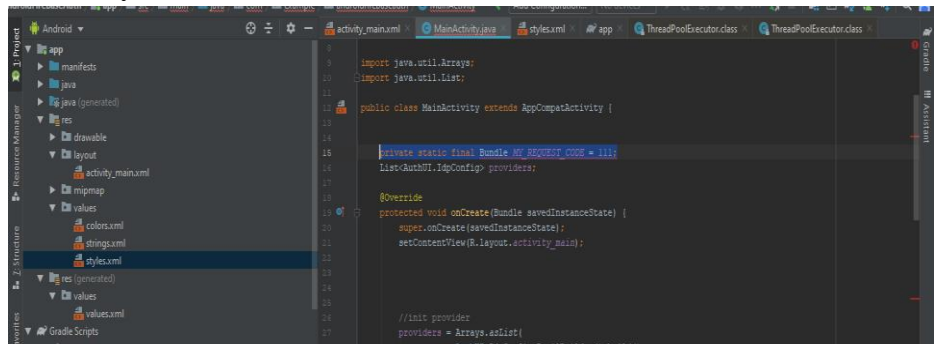
```
AuthUI.getInstance().createSignInIntentBuilder()
    .setAvailableProviders(providers)
    .setTheme()
);
```

## 21. Tambahkan style pada style xml



```
<style name="MyTheme" parents="AppTheme">
    <item
name="android:windowBackground">@android:color/white
</item>
</style>
```

22. Setelah itu kita buat constarin field request code beserta methodnya

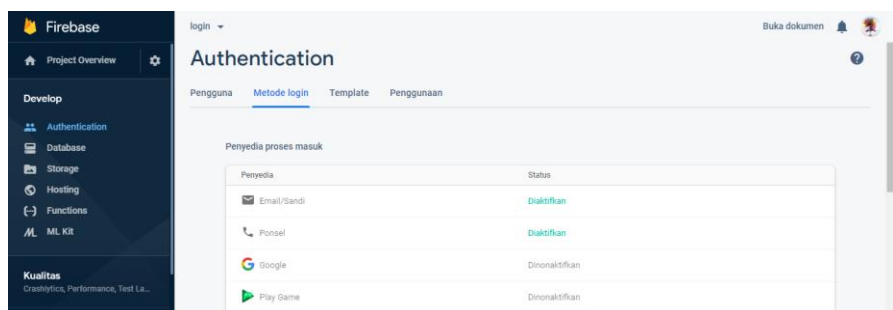


```
private void showSignInOption() {  
    startActivityForResult(  
  
        AuthUI.getInstance().createSignInIntentBuilder()  
            .setAvailableProviders(providers)  
            .setTheme(R.style.MyTheme)  
            .build(MY_REQUEST_CODE  
  
    );  
}
```

23. Pada private static final masukan request code angka bebas sebanyak 3 digit

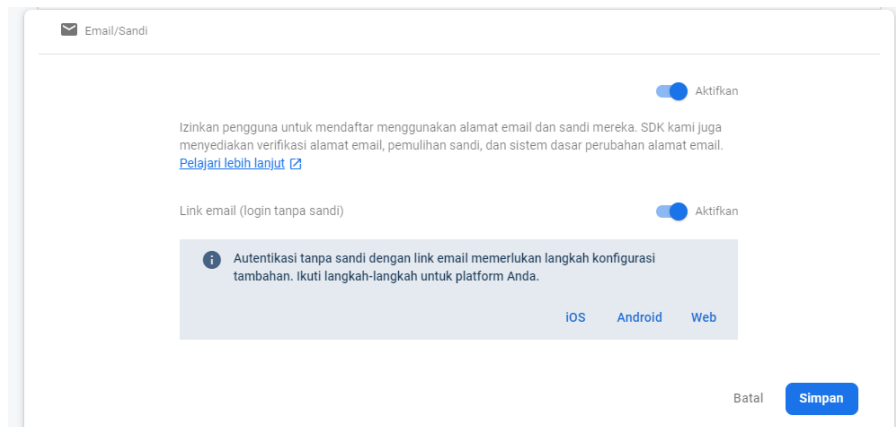
```
private static final Bundle MY_REQUEST_CODE = 111;
```

24. Setelah itu masuk ke firebase console



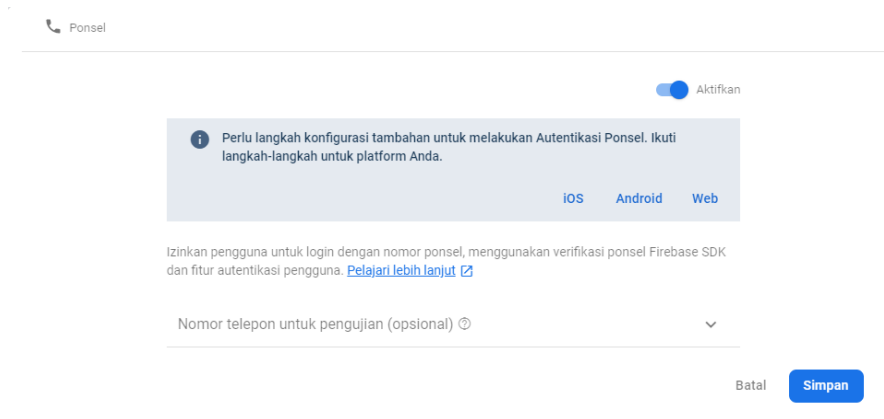
25. Karna kita memilih 3 providers ma aktifkan keempat providers tersebut

26. Aktifkan email provider



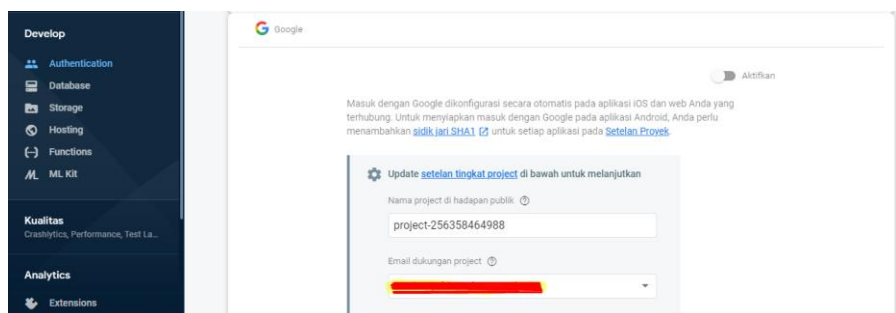
The screenshot shows the 'Email/Sandi' provider configuration page in the Firebase console. At the top, there's a toggle switch labeled 'Aktifkan' which is turned on. Below it, a text block explains that users can register with email and password, and the SDK provides verification, password reset, and address change features. A link 'Pelajari lebih lanjut' is provided. Further down, another toggle switch for 'Link email (login tanpa sandi)' is also turned on. A light blue information box contains a message: 'Autentikasi tanpa sandi dengan link email memerlukan langkah konfigurasi tambahan. Ikuti langkah-langkah untuk platform Anda.' Below this box are links for 'iOS', 'Android', and 'Web'. At the bottom right, there are 'Batal' and 'Simpan' buttons.

27. Aktifkan phone number provider



The screenshot shows the 'Ponsel' provider configuration page. A toggle switch labeled 'Aktifkan' is turned on. A light blue information box states: 'Perlu langkah konfigurasi tambahan untuk melakukan Autentikasi Ponsel. Ikuti langkah-langkah untuk platform Anda.' with links for 'iOS', 'Android', and 'Web'. Below the box, text explains that users can login with phone numbers using Firebase SDK verification and authentication features, with a 'Pelajari lebih lanjut' link. There is a field for 'Nomor telepon untuk pengujian (opsional)' with a dropdown arrow. At the bottom right, 'Batal' and 'Simpan' buttons are visible.

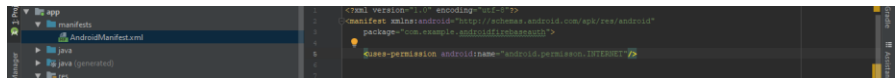
28. Aktifkan google provider



The screenshot shows the 'Google' provider configuration page. A toggle switch labeled 'Aktifkan' is turned on. Text explains that Google authentication is configured automatically on iOS and web, but requires adding a 'sidik jari SHA1' for Android. A link 'Setelan Proyek' is provided. Below, a section titled 'Update setelah tautkan project di bawah untuk melanjutkan' contains a form. The 'Nama project di hadapan publik' field has the value 'project-256358464988'. The 'Email dukungan project' field has a dropdown menu. On the left side of the image, a partial view of the Firebase console sidebar is visible, showing categories like 'Develop', 'Kualitas', and 'Analytics'.

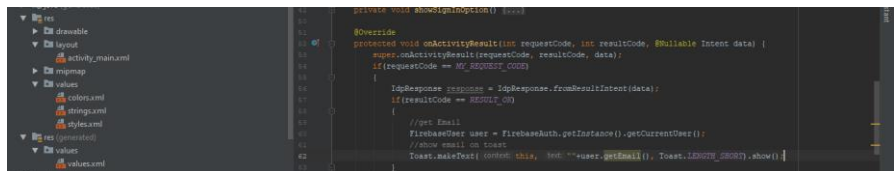
29. Di google provider kita harus menginputkan gmail aktif

30. Setelah itu kita menambahkan izin menggunakan internet



```
<uses-permission  
    android:name="android.permission.INTERNET"/>
```

31. Sekarang kita harus override onActivityResult



@Override

```
protected void onActivityResult(int requestCode, int  
resultCode, @Nullable Intent data) {
```

```
    super.onActivityResult(requestCode, resultCode, data);
```

```
    if(requestCode == MY_REQUEST_CODE)
```

```
    {
```

```
        IdpResponse response =
```

```
        IdpResponse.fromResultIntent(data);
```

```
        if(resultCode == RESULT_OK)
```

```
        {
```



```
FirebaseUser user =  
FirebaseAuth.getInstance().getCurrentUser();  
  
    Toast.makeText(this, ""+user.getEmail(),  
    Toast.LENGTH_SHORT).show();
```

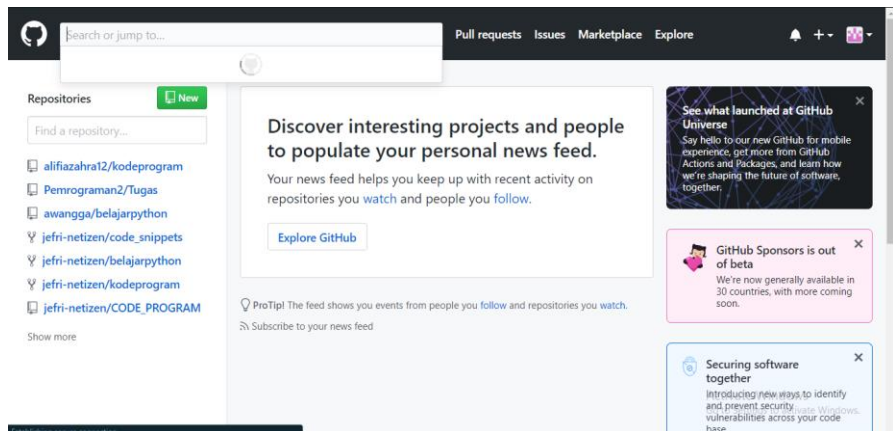
32. Sekarang tambahkan button sign out

```
btn_sign_out.setEnabled(true);  
  
else  
  
    {  
  
    Toast.makeText(this, ""+response.getError().getMessage(),  
    Toast.LENGTH_SHORT).show();  
  
    }
```

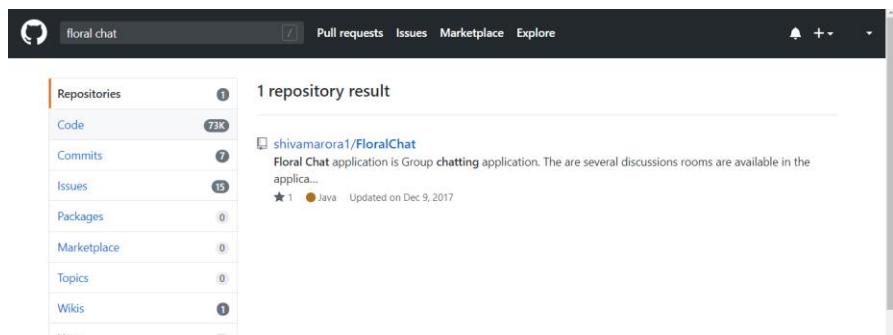
33. Tambahkan fungsi button sign out

# Cara mengupload projek dari github ke Android Studio

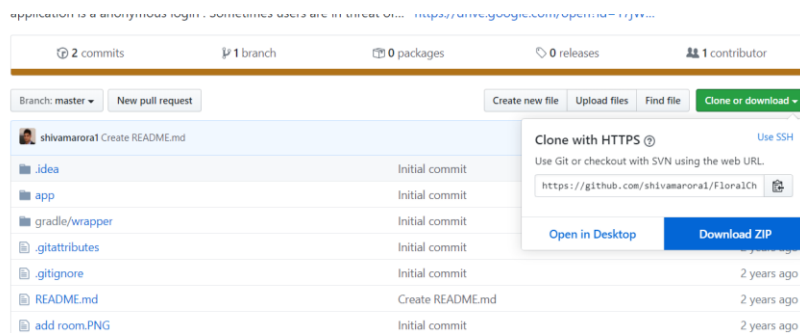
1. Login ke akun github
2. Lalu cari contoh aplikasi



3. Lalu pilih repository yang ingin kalian buka di android studio.



#### 4. Download zip



#### 5. Buka android studio

6.





































































































































































































## **REFERENSI**

- [1] "The Google Geocoding API," Google Developer, [Online]. Available: <https://developers.google.com/maps/documentation/geocoding/>. [Accessed 01 06 2015].
- [2] , Android Developer;, "Android Studio," Android Developer, [Online]. Available: <http://developer.android.com/sdk/>. [Accessed 01 04 2015].
- [3] R. A. Sukanto dan M. Shalahuddin, Rekayasa Perangkat Luna. Bandung: Informatika, 2013.
- [4] "Pengenalan JSON," JSON, [Online]. Available: [www.json.org/json-id.html](http://www.json.org/json-id.html). [Accessed 05 04 2015].

[5] [GitHub - google/google-authenticator: Open source version of Google Authenticator \(except the Android app\)](#)". *GitHub*. These implementations support the HMAC-Based One-time Password (HOTP) algorithm specified in RFC 4226 and the Time-based One-time Password (TOTP) algorithm specified in RFC 6238.

[6] Willis, Nathan (22 January 2014). "*FreeOTP multi-factor authentication*". *LWN.net*. Retrieved 10 August 2015.

[7] [https://firebase.google.com/docs/auth/?gclid=Cj0KCQiAmZDxBRDIARIsABnkbYTn4\\_Z9a1qu-mTw0vAhrwB5iEeIy221VjbuvFY\\_xqg7KbfXf-AEqwAaAhj8EALw\\_wcB](https://firebase.google.com/docs/auth/?gclid=Cj0KCQiAmZDxBRDIARIsABnkbYTn4_Z9a1qu-mTw0vAhrwB5iEeIy221VjbuvFY_xqg7KbfXf-AEqwAaAhj8EALw_wcB)

[8] Asropudin, Pipin. Kamus Teknologi Informasi. Bandung: Titian Ilmu, 2013.

[9] Bambang Hariyanto, Esensi-esensi Bahasa Pemrograman Java, Informatika, Bandung, 2003.

[11] Efferndi, Empty. (2005) E-Learning Konsep dan Aplikasi. Yogyakarta: Andi Yogyakarta

[12] Melfachrozi M, "Penggunaan Aplikasi ELearning" <http://repository.unand.ac.id/6/1/roziemoodle.pdf> (diakses pada tanggal 23 Januari 2013)

[13] Yadi Utama, "Pengenalan Web Service" [http://elib.unikom.ac.id/download.php?id=1\\_32941](http://elib.unikom.ac.id/download.php?id=1_32941) (diakses pada tanggal 23 Januari 2013)

[14] Stallings (2005). *Operating Systems, Internals and Design Principles*. Pearson: Prentice Hall. hlm. 6. Quinn, C. (2000). *Mlearning, Mobile Wireless in Your Pocket Learning*

[15] Wijaya, Stevanus Wisnu. *Mobile Learning Sebagai Model Pembelajaran Alternatif Bagi Pemulihan Pendidikan Di Daerah Bencana Alam Gempa Bumi Yogyakarta*. Universitas Sanata Dharma

[16] Sri Nugraha, Aditya. (2010). *Pengembangan dan Implementasi Mobile Learning Berbasis J2ME untuk Mata Pelajaran Keterampilan Komputer dan Pengelolaan Informasi*. Pendidikan Ilmu Komputer UPI. Bandung

[17] Ally, Mohamed. (2009). *Mobile Learning Transforming the Delivery of Education and Training*. Atabasca University: AU Press.

[18] [medium.com/ppl-a-4-yuk-recycle/yuksharing-flutter-androidx-a-app-compatibility-4644854b2e4c](https://medium.com/ppl-a-4-yuk-recycle/yuksharing-flutter-androidx-a-app-compatibility-4644854b2e4c)

[19]

<https://stackoverflow.com/questions/51280090/what-is-androidx>

[20] Kukulska-Hulme, A., dan Traxler, J eds, 2007. *Mobile Learning A Handbook for Educators and Trainers*. Oxfordshire: Routledge.

[21] Yuen, S.C. & Yuen, P.K.. *Mobile Learning: Learning On The Go*. In: D. Taniar, ed. 2009. *Mobile Computing*. London: Information Science Reference. Ch.1.10.

[22] Chao,H ,et.al. Environment for Mobile Learning. In:D. Taniar, ed. 2009. Mobile Computing. London:Information Science Reference. Ch.1.11.

[23] Mallick, M., 2003. Mobile and Wireless Design Essentials. Indiana:Wiley Publishing Inc.

[24] <http://kbbi.web.id/diskusi> diakses pada 2 Oktober 2015

[25] Saputra, Agus, Forum Diskusi PHP dan MySQL. Cetakan Pertama. Cv.ASFA Solution, Cirebon, 2014.

[26] Sidik, B., dan Pohan, H., Pemrograman Web dengan HTML, Informatika, Bandung, 2012.

tanggal 23 januari 2013)

[27] Maulana, G.G., 2017. Pembelajaran Dasar Algoritma dan Pemrograman Menggunakan El-goritma Berbasis Web. *Jurnal Teknik Mesin Mercuri Buana*, 6(2), pp.69-73.

[28] Budiyo, Alex. "Pengantar Algoritma dan Pemrograman." *Ilmukomputer. com* (2003).