

BAB II

TINJAUAN PUSTAKA

2.1 Android

Android merupakan sistem operasi yang berbasis linux dan dimodifikasi untuk perangkat bergerak (*mobile device*) yang terdiri dari sistem operasi, *middleware*, dan aplikasi-aplikasi utama [1]. Android dikembangkan oleh google yang mana android didesain untuk perangkat *mobile* dengan layar sentuh, utamanya pada telepon pintar (*smartphone*) ataupun tablet. Tidak hanya pada perangkat *mobile* kini android juga telah ada pada beberapa perangkat lain seperti televisi (*Android TV*), pada kendaraan (*Android Auto*), jam tangan (*Android Wear*), dan juga perangkat elektronik lainnya seperti *notebook*, konsol *game*, kamera digital, dan lainnya. Jadi android adalah sistem operasi yang digunakan pada perangkat elektronik berbasis layar sentuh (*touchscreen*), seperti *smartphone* ataupun *tablet*.

Kode program android dirilis oleh google dibawah lisensi *open source* sehingga berbagai perusahaan yang memproduksi *smartphone* dapat merilis sistem operasi android mereka sendiri tanpa menghilangkan inti dari android yang dikembangkan oleh google. Tidak hanya itu karena sifatnya yang *open source* membuat banyak pengembang dapat mengembangkan android mereka dengan berbagai fitur-fitur baru dan kemudian dibagikan kepada orang lain agar dapat digunakan.

Begitupun dengan para pembuat aplikasi, mereka bebas membuat aplikasi dengan kode-kode sumber yang dikeluarkan oleh google. Dengan seperti itu android memiliki jutaan *support* aplikasi gratis maupun berbayar yang dapat diunduh melalui *google play*. Aplikasi-aplikasi android tersebut juga telah banyak membantu pengguna android dalam kegunaannya masing-masing. Dan android telah menjadi kebutuhan masyarakat untuk berkomunikasi, menghibur diri, dan membantu mempermudah suatu pekerjaan.

2.1.1 Perkembangan Android

Semenjak dirilis pertama kali secara komersial pada tahun 2008, android telah mengalami banyak perubahan sampai saat sekarang ini. Pembaharuan terus dilakukan dimana pembaharuan tersebut bertujuan untuk menambah berbagai fitur-fitur baru serta memperbaiki kesalahan-kesalahan (*bugs*) yang ada pada versi terdahulunya. Setiap versi dari pembaharuan android diberikan kode nama masing-masing yang beberapa diantaranya diambil dari nama makanan penutup yang berurut berdasarkan alfabet. Berikut adalah versi-versi dari pembaharuan android:

Tabel 2.1 Perkembangan Android [2]

Nama Versi	No. Versi	Tanggal Rilis	Pembaharuan
<i>Alpha</i>	1.0	23 September 2008	Aplikasi <i>gmail</i> , <i>google talk</i> , <i>youtube</i> , <i>app market</i> , dan lainnya
<i>Beta</i>	1.1	9 Februari 2009	Memperbaiki <i>bug</i> pada versi <i>alpha</i>
<i>Cupcake</i>	1.5	27 April 2009	Perubahan pada tampilan UI, mendukung <i>keyboard</i> pihak ketiga, dan lainnya
<i>Donut</i>	1.6	15 September 2009	Opsi pencarian dengan teks dan suara ditingkatkan, terdapat fitur cuplikan aplikasi di <i>Android Market</i> , dan dukungan mesin sintetis multi bahasa.
<i>Eclair</i>	2.0-2.1	26 Oktober 2009	Mengoptimalkan kecepatan perangkat lunak & perubahan UI
<i>Frozen Yoghurt (Froyo)</i>	2.2	20 Mei 2010	Meningkatkan fungsi USB <i>Tethering</i> dan WiFi Hotspot

<i>Gingerbread</i>	2.3	6 Desember 2010	Memperbaiki desain antarmuka pengguna dengan meningkatkan performa dan kesederhanaan tampilan
<i>Honeycomb</i>	3.0 – 3.2	22 Februari 2011	Menambahkan <i>System Bar</i> yang memberi akses cepat ke notifikasi, status, dan tombol navigasi di bawah layar
<i>Ice Cream Sandwich</i>	4.0	18 Oktober 2011	Screenshot layar hp dengan tekan tombol daya dan <i>volume down</i> bersamaan, mendukung fitur pengenalan wajah, dan bisa membatasi pemakaian data
<i>Jelly Bean</i>	4.1-4.3	27 Juni 2012	Meningkatkan fungsi dan kinerja antarmuka pengguna yang cukup penting danustomisasi tata letak antarmuka dan mengatur ukuran <i>shortcut</i> dan <i>widget</i>
<i>KitKat</i>	4.4	3 September 2013	Performa sistem lebih tinggi dan layar sentuh bisa merespon lebih cepat
<i>Lollipop</i>	5.0	25 Juni 2014	Dukungan <i>Project Volta</i> yang berguna meningkatkan daya tahan baterai dan fitur <i>Factory Reset Protection</i>
<i>Marshmallow</i>	6.0	5 Oktober 2015	Tampilan yang didesain ulang dan punya skema manajemen daya bernama <i>doze</i>

<i>Nougat</i>	7.0-7.1	9 Maret 2016	Memungkinkan perangkat yang mendukung bisa meningkat versi Android secara <i>Over The Air</i> (OTA)
<i>Oreo</i>	8.0-8.1	21 Agustus 2017	Arsitektur modular yang membuatnya lebih mudah dan lebih cepat untuk pembuat hardware untuk menghadirkan pembaruan Android
<i>Pie</i>	9	6 Agustus 2018	Menghadirkan fitur <i>gesture</i> untuk berpindah dari satu aplikasi ke aplikasi lain dan mempunyai fitur <i>adaptive battery</i> dan <i>adaptive brightness</i>
Android Q	10	3 September 2019	Fitur Teks Otomatis yang bisa memberi teks pada video atau <i>podcast</i> yang direkam, tanpa harus terhubung ke WiFi atau data seluler
Android 11	11	8 September 2020	Mengelompokkan aplikasi dalam status notifikasi, fitur Balon yang bisa membuat Anda melanjutkan percakapan setelah mengakses aplikasi lain
<i>Snow Cone</i>	12	4 Oktober 2021	<i>Update</i> besar pada Desain Material yang kemudian disebut sebagai " <i>Material You</i> "

2.1.2 Fitur Android

Android memiliki berbagai fitur yang dapat digunakan oleh penggunanya. Berikut ini adalah fitur umum yang ada pada android:

1. Antarmuka

Antarmuka pada android yang paling utama adalah manipulasi secara langsung dimana pengguna menggunakan layar sentuh dan memberikan perintah dengan cara mengetuk, menggeser, dan memberikan arahan lainnya lewat layar sentuh tersebut dan juga pengguna dapat mengetik lewat *virtual keyboard* yang dapat diketik lewat layar sentuh tersebut.

Pada perangkat android juga biasanya dilengkapi dengan berbagai macam sensor untuk menambah interaksi antarmuka pengguna dengan android seperti contohnya yaitu layar dapat berubah posisi dari *potrait* menjadi *landscape* tergantung dengan pemosisian perangkat tersebut.

2. Aplikasi

Pada android terdapat berbagai macam aplikasi yang dapat digunakan pengguna. Sebagian aplikasi tersebut dikembangkan oleh google dan dijadikan sebagai aplikasi bawaan pada android. Selain itu ada juga aplikasi lainnya yang dapat diunduh oleh pengguna lewat layanan *Google Play Store* milik google. Pada *Google Play Store* tersebut terdapat berbagai macam aplikasi yang dikembangkan oleh berbagai pengembang aplikasi dari seluruh dunia dimana aplikasi mereka nanti juga dapat diunduh oleh pengguna android.

Hal ini dapat dilakukan karena google memberikan *software development kit* secara gratis yang dapat diunduh oleh para pengembang aplikasi agar mereka dapat mengembangkan aplikasi android mereka. Tidak hanya itu google juga merilis *software intergrated development environment (IDE)* yaitu perangkat lunak yang digunakan untuk mengembangkan aplikasi android yaitu *Android Studio*. Pada *Android Studio* juga telah termasuk

dengan *software development kit* tersebut dan berbagai *tools* lainnya untuk membantu para *developer* untuk mengembangka aplikasi android.

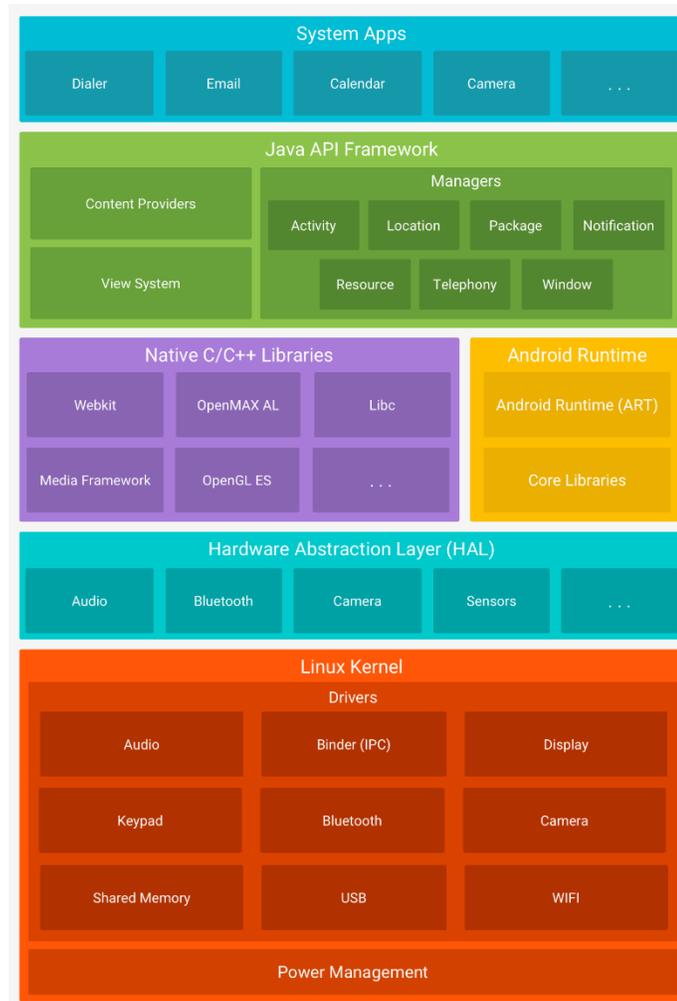
3. Manajemen *Memory*

Android didesain untuk mengatur berbagai macam proses untuk menjaga konsumsi daya tetap pada tingkat yang paling sedikit. Saat suatu aplikasi tidak digunakan, sistem akan mematikan sementara proses aplikasi tersebut tetapi tidak menghentikan prosesnya untuk mematikan penggunaan baterai dan penggunaan CPU (*Central Processing Unit*) agar nantinya jika pengguna ingin memakai aplikasi tersebut maka android akan menampilkan hal terakhir yang dibuka oleh pengguna.

Android mengatur penggunaan *memory* aplikasi secara otomatis. Saat kondisi *memory* rendah maka sistem dilatar belakang akan secara otomatis menutup proses yang tidak aktif dimulai dengan proses yang paling lama tidak aktif.

2.1.3 Arsitektur Android

Android merupakan sistem operasi yang *open source* dikembangkan berdasarkan kernel linux. Android dikembangkan untuk bekerja dengan membuat *stack software* (tumpukan perangkat lunak) agar dapat berjalan dengan baik pada berbagai perangkat. Berikut adalah diagram *software stack* yang menggambarkan komponen utama dari *platform* android:



Gambar 2.1 *Software Stack* Android [1]

1. Kernel Linux

Kernel Linux merupakan fondasi dari *platform* Android. Sebagai contoh, *Android Runtime* bergantung pada kernel Linux untuk menyediakan fungsionalitas seperti *threading* dan *low-level memory management*. Dengan menggunakan kernel Linux, Android akan mendapatkan keuntungan dari fitur kunci keamanan dan memperbolehkan manufaktur perangkat untuk mengembangkan *driver hardware* untuk *Hardware Abstraction Layer* (HAL).

2. *Hardware Abstraction Layer (HAL)*

Hardware Abstraction Layer (HAL) menyediakan standar antarmuka yang membuka kapabilitas perangkat keras dari suatu perangkat Android ke tingkat *Java API Framework* yang lebih tinggi. HAL terdiri dari berbagai macam library module yang di implementasikan kepada antarmuka untuk setiap komponen perangkat keras yang spesifik seperti kamera atau *Bluetooth module*.

3. *Android Runtime*

ART dibuat untuk menjalankan berbagai *virtual machine* pada perangkat dengan memory rendah dengan mengeksekusi DEX file yaitu *kode byte* yang didesain secara khusus untuk Android yang telah dioptimisasi untuk kapasitas *memory* yang minim.

4. *Native C/C++ Libraries*

Berbagai macam komponen dan layanan pada Android seperti ART dan HAL dibangun dengan *native code* yang membutuhkan *native libraries* yang ditulis dengan bahasa pemrograman C dan C++. Platform Android menyediakan *Java framework API* untuk membuka fungsi dari beberapa *native libraries* tersebut kepada aplikasi. Sebagai contoh, pengembang dapat mengakses layanan OpenGL ES lewat *framework* Android yaitu *Java OpenGL API* untuk menambahkan dukungan untuk menggambar dan memanipulasi grafik 2 dimensi dan 3 dimensi pada aplikasi yang mereka kembangkan.

5. *Java API Framework*

Seluruh fitur dari sistem operasi Android tersedia dengan menggunakan API yang ditulis menggunakan bahasa pemrograman Java. API tersebut menyediakan berbagai fungsi yang dibutuhkan pengembang untuk membuat aplikasi Android dengan menyediakan komponen dan layanan sistem yang *modular*.

6. Aplikasi Sistem

Android menyediakan berbagai macam aplikasi bawaan seperti aplikasi *email*, kontak, kalender, *internet browser* dan lain sebagainya. Aplikasi bawaan ini tidak diatur sebagai aplikasi utama dari perangkat Android tersebut sehingga sebagai contoh pengguna dapat mengunduh aplikasi *internet browser* yang lain lewat *Google Play Store* dan menjadi aplikasi *internet browser* tersebut sebagai aplikasi utama.

2.2 Android Studio

Android Studio merupakan *software integrated development environment (IDE)* yang resmi untuk membangun aplikasi Android. *Android Studio* dibangun berdasarkan IntelliJ IDEA yang merupakan *software integrated development enviroment* untuk membangun aplikasi atau *software* dengan bahasa pemrograman *Java* [3]. Seluruh fitur yang ada pada IntelliJ IDEA juga terdapat pada *Android Studio* yang kemudian ditambahkan lagi fitur-fitur lainnya agar dapat meningkatkan produktivitas para pengembang aplikasi Android yaitu seperti:

1. *System build gradle* yang fleksibel.
2. *Emulator* android yang kaya dengan fitur.
3. Dukungan untuk membangun aplikasi android untuk perangkat apapun baik itu pada *smartphone*, *Android TV*, *Android Wear*, dan perangkat android lainnya.
4. *Instant Run* yang dapat melakukan perubahan pada aplikasi tanpa harus melakukan instalasi ulang aplikasi.
5. *Code templates* dan integrasi GitHub untuk membantu para pengembang membangun aplikasi umum dan melakukan *import* kode *sample*.
6. Berbagai macam *testing tools* dan *frameworks*.
7. Dukungan untuk bahasa pemrograman C dan C++.

Android Studio pertama kali diumumkan oleh google pada 16 Mei 2013 pada *Google I/O Conference* dan menggantikan Eclipse sebagai *software IDE* resmi untuk *platform android*. Saat ini *Android Studio* telah mencapai versi 4.2.2 yang tersedia untuk diunduh secara gratis untuk sistem operasi *Windows*, *macOS*, dan *Linux*. Berikut adalah rekomendasi spesifikasi perangkat untuk menjalankan *Android Studio*:

Tabel 2.2 Spesifikasi perangkat untuk *Android Studio* [3]

Kriteria	Spesifikasi
Sistem Operasi	<i>Windows 7</i> atau lebih tinggi <i>Mac OS X 10.9.5</i> atau lebih tinggi <i>Linux</i>
RAM	Minimal RAM 3GB Rekomendasi RAM 8GB ditambah dengan RAM 1GB untuk <i>Android Emulator</i>
Kapasitas penyimpanan	Minimal kapasitas penyimpanan 500MB untuk <i>Android Studio</i> dan minimal 1,5GB untuk <i>Android SDK, Emulator, System Images, dan Caches</i>
Versi <i>Java</i>	<i>Java Development Kit (JDK) 8</i>
Resolusi Layar	Minimal resolusi layar 1280x800

2.3 *Java*

Java merupakan bahasa pemrograman komputer yang bersifat *concurrent, class-based, dan object-oriented* serta didesain agar mempunyai kebergantungan implementasi serendah mungkin. *Java* dikembangkan agar para pengembang aplikasi dapat *write once, run anywhere* (WORA) yang maksudnya setelah kode pemrograman *software* tersebut ditulis dan di *compile* maka *software* tersebut dapat dijalankan di berbagai *platform* yang mendukung bahasa pemrograman *Java* [4]. *Software* dan aplikasi yang dibangun dengan bahasa pemrograman *Java* akan di

compile ke dalam kode *byte* yang akan dapat berjalan di setiap *Java Virtual Machine* manapun tanpa bergantung pada arsitektur komputernya. Pada tahun 2016 tercatat bahwa *Java* merupakan salah satu bahasa pemrograman yang paling populer digunakan untuk aplikasi *web client-server* yang dilaporkan digunakan oleh 9 juta pengembang.

Awal dari perkembangan bahasa pemrograman *Java* dimulai dari bulan Juni 1991 dimana James Gosling, Mike Sheridan dan Patrick Naughton menginisiasi untuk mengembangkan *Java*. Tujuan awal dari perkembangan bahasa pemrograman *Java* yaitu untuk televisi interaktif tapi pada masa itu masih tidak memungkinkan. Penamaan awal dari bahasa pemrograman *Java* adalah Oak yang diambil namanya dari pohon Oak yang ada diluar kantornya Gosling. Kemudian nama tersebut diganti menjadi *Green* dan pada akhirnya dinamakan *Java* yang diambil dari kopi Jawa (Jawa yang dalam bahasa inggris adalah *Java*). Gosling mendesain *Java* agar syntaxnya mirip dengan bahasa pemrograman C/C++ sehingga para programmer akan familiar dengan *Java*. Ada 5 tujuan prinsip saat dikembangkannya bahasa pemrograman *Java* yaitu:

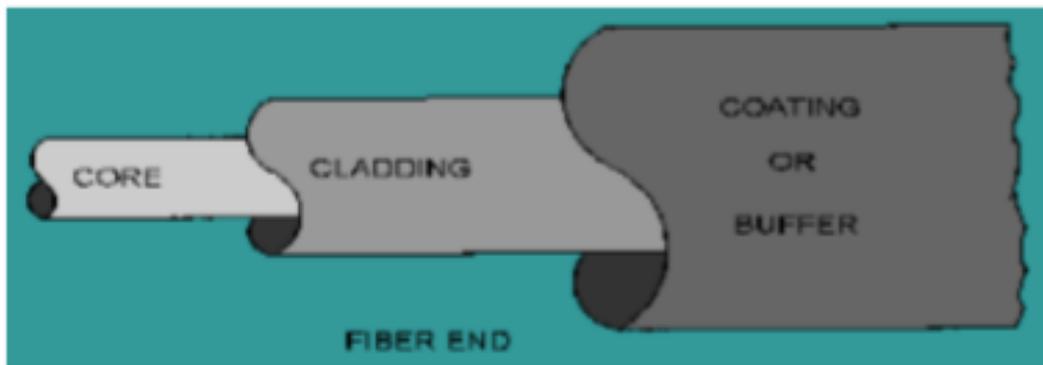
1. *Java* harus *simple*, *object-oriented* dan *familiar*.
2. *Java* harus *robust* dan *secure*.
3. *Java* harus *architecture neutral* dan *portable*.
4. *Java* harus dieksekusi dengan *high performance*.
5. *Java* harus *interpreted*, *threaded*, dan *dynamic*.

Sun Microsystems tempat dimana Gosling bekerja akhirnya merilis versi *Java* pertama yaitu 1.0 pada tahun 1995. Kemudian perkembangan *Java* terus dilakukan sampai dengan tahun 2009 saat *Sun Microsystems* diakuisisi oleh *Oracle Corporation*. Perkembangan *Java* tidak berhenti disitu dan terus berkembang sampai dengan saat ini dimana versi terakhir dari *Java* saat ini yaitu *Java 19*.

2.4 Fiber Optik

Fiber optik adalah saluran transmisi/sejenis kabel yang terbuat dari kaca atau plastik yang sangat halus dan lebih kecil dari sehelai rambut dan dapat digunakan untuk mentransmisikan sinyal cahaya dari suatu tempat ke tempat yang lain [5]. Sumber cahaya yang digunakan adalah LASER atau LED. Serat optik memiliki diameter lebih kurang 120 mikrometer, dan memiliki kecepatan transmisi yang sangat tinggi sehingga sangat bagus digunakan sebagai saluran komunikasi.

Pada dasarnya fungsi dari kabel fiber optik sama seperti jenis kabel yang lain yakni menghubungkan antar komputer atau pengguna satu sama lain dalam lingkup jaringan tertentu. Yang menjadi pembeda adalah kecepatan akses yang tinggi serta kemampuan transfer data lebih cepat [5]. Untuk kecepatan pengiriman data bisa sampai kisaran Gigabit per detiknya. Selain itu karena tidak membawa listrik kabel jenis ini juga tidak terpengaruhi gangguan elektromagnetik sehingga stabil dalam penggunaannya.



Gambar 2.2 *Fiber optic* [5]

2.5 *Fiber to The Home (FTTH)*

Fiber to The Home (FTTH) merupakan arsitektur jaringan kabel fiber optik yang dibuat hingga sampai ke rumah-rumah atau ruangan dimana terminal berada. Perkembangan teknologi ini tidak terlepas dari kemajuan perkembangan teknologi serat optik yang dapat menggantikan penggunaan kabel konvensional. Teknologi FTTH merupakan sepenuhnya jaringan optik dari pusat penyedia ke pemakai, dan biasanya digunakan *splitter* 1:16, yaitu sinyal *multiplex* dibagi ke 16 rumah yang berbeda [6].

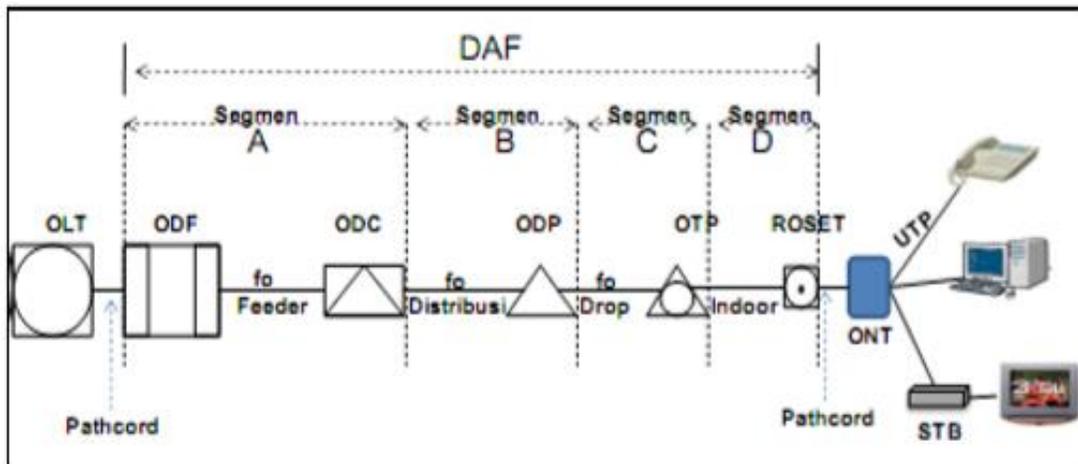
Fiber to The Home (FTTH) merupakan suatu transmisi sinyal optik dari pusat penyedia (*provider*) ke Kawasan pengguna dengan menggunakan fiber optik sebagai media penghantar. Biasanya jarak antara pusat layanan dengan pelanggan mencapai jarak maksimal 20 kilometer, dimana pada bagian *service provider* yang berada di kantor utama atau *Central Office (CO)* terdapat perangkat yang bernama OLT. OLT kemudian dihubungkan ke ONU yang terletak di rumah-rumah pelanggan melalui jaringan distribusi serat optik yang bernama *Fiber Access Terminal (FAT)*. Adapun beberapa keunggulan dari FTTH antara lain:

- a. FTTH menyediakan *range* yang lebar bagi pelanggan untuk melakukan komunikasi maupun aktivisasi terhadap layanan yang baru yang lebih cepat.
- b. Pendistribusian kabel optik langsung tertuju kepada pengguna layanan sehingga dapat menyediakan jumlah *bandwidth* maksimum untuk permintaan layanan di kemudian hari.
- c. FTTH menawarkan banyak layanan pelanggan berupa data, suara, dan video.
- d. FTTH memiliki desain arsitektur jaringan yang fleksibel yang dapat digunakan untuk mengakomodasi inovasi di masa mendatang.
- e. Mendukung pengembangan dan peningkatan jaringan masa depan.
- f. Dapat meminimalisasikan gangguan, sehingga dapat meningkatkan pemasukan dari pemilik jaringan dan bermanfaat bagi pelanggan FTTH.

- g. Merupakan bisnis yang sukses karena factor pengeluaran modal (CAPEX) dan biaya operasi (OPEX) yang relative seimbang.

2.5.1 Arsitektur Jaringan *Fiber to The Home* (FTTH)

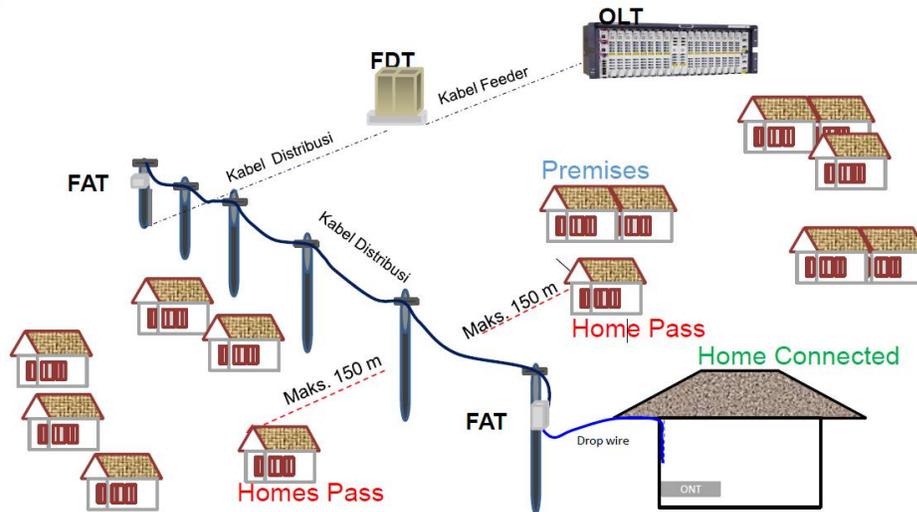
Secara umum jaringan FTTH dapat dibagi menjadi 4 segmen catuan kabel selain perangkat aktif seperti OLT dan ONT/ONU, yaitu segmen A berupa catuan kabel *feeder*, segmen B berupa catuan kabel distribusi, segmen C berupa catuan kabel penanggal/*drop*, dan segmen D berupa catuan kabel rumah. Arsitektur jaringan FTTH dapat dilihat pada gambar dibawah ini.



Gambar 2.3 Arsitektur jaringan FTTH [6]

2.5.2 Skema Jaringan FTTH

Pada skema jaringan FTTH terdapat 3 bagian yaitu:



Gambar 2.4 Skema jaringan FTTH [7]

a. Premises

Premises adalah rumah tinggal atau tempat usaha, baik hunian satu unit maupun hunian multi unit seperti apartemen dihitung sebagai satu tempat.

b. Homes Pass

Home Pass adalah jumlah potensi rumah atau bangunan dimana operator telekomunikasi memiliki kemampuan untuk menghubungkan alat produksi di daerah layanan tersebut. Didalam definisi ini tidak termasuk tempat hunian dimana lokasinya tersebut tidak terhubung dan atau pada jarak tertentu tanpa instalasi lebih lanjut dari kabel tertanam secara substansial seperti *feeder* dan kabel distribusi (fiber) untuk mencapai daerah dimana pelanggan baru yang memiliki potensi itu berada.

c. *Home Connected*

Home Connected adalah rumah atau bangunan yang terhubung ke jaringan FTTH dari menggunakan setidaknya satu layanan koneksi ini dan di dukung dengan kontrak komersial.

2.6 Perangkat-perangkat FTTH (*Fiber to The Home*)

Berdasarkan pada gambar 2.4 Skema jaringan FTTH, dapat dijelaskan beberapa elemen dan perangkat yang ada pada arsitektur jaringan FTTH, yaitu:

2.6.1 *Optical Line Terminal (OLT)*

Optical Line Terminal (OLT) adalah suatu perangkat aktif (*Opto-Elektrik*) yang berfungsi untuk mengubah sinyal elektrik menjadi sinyal optik, serta sebagai alat multiplexs. OLT merupakan perangkat yang berfungsi sebagai titik akhir dari pusat penyedia layanan *Passive Optical Network (PON)*. OLT juga berfungsi untuk mengumpulkan dan men-*switch* fungsi antara jaringan kabel dengan *interface PON* serta untuk fungsi manajemen.

Namun demikian, OLT memiliki 2 (dua) fungsi utama, yaitu untuk mengkonversi antara sinyal listrik yang digunakan oleh perangkat *provider* dengan sinyal fiber optik yang digunakan oleh perangkat *provider* dengan sinyal fiber optik yang digunakan oleh perangkat *provider* dengan sinyal fiber optik yang digunakan oleh jaringan PON, serta untuk proses *multiplexing* dengan perangkat pada ujung jaringan. Adapun persyaratan umum untuk OLT, yaitu:

- a. *Backplane OLT* menyediakan sistem *backup* (redundansi) dan koneksi *independent 10 Gigabit Ethernet full duplex* untuk masing-masing servis slot.
- b. Kemampuan *switching fabric OLT* yang mempunyai arsitektur *non-blocking 150 Gbps full duplex per shelf*.

c. OLT memiliki *universal service slot* untuk PON card.



Gambar 2.5 *Optical Line Terminal (OLT)* [7]

2.6.2 Kabel Feeder

Kabel *feeder* berfungsi untuk menyalurkan informasi berupa sinyal optik hasil konversi perangkat OLT, dan biasanya menggunakan kabel *single mode*. Ditinjau dari jenis instalasinya maka kabel *feeder* terdiri dari beberapa macam, yaitu kabel fiber optik tanam langsung (jarang diimplementasikan), kabel fiber optik *duct*, dan kabel fiber optik udara (*aerial*).

2.6.3 Fiber Distribution Terminal (FDT)

Fiber Distribution Terminal (FDT) merupakan suatu ruang yang berfungsi sebagai tempat untuk melakukan proses instalasi sambungan jaringan optik *single mode*. Ruang tersebut berbentuk kotak/kubah (*dome*) yang terbuat dari bahan material khusus. Didalam FDT terdapat beberapa perangkat seperti *connector*, *splicing*, maupun *splitter*. *Connector* digunakan sebagai penghubung kabel optik, *splice* digunakan untuk menyambung kabel optik satu sama lain, sedangkan *splitter*

merupakan perangkat pasif yang dapat memisahkan day optik dari satu *input fiber* ke beberapa *output fiber*.

FDT berfungsi sebagai titik terminasi ujung kabel *feeder* dan pangkal dari kabel distribusi, sebagai titik distribusi kabel dari kabel berkapasitas besar (*feeder*) menjadi beberapa kabel yang memiliki kapasitas yang lebih kecil (distribusi), sebagai tempat penyambungan, dan juga tempat pemasangan *splitter*.



Gambar 2.6 *Fiber Distribution Terminal (FDT)* [7]

2.6.4 Kabel *Distribution*

Kabel fiber optik distribusi dan seperti kabel fiber optik *feeder* yang berfungsi untuk meneruskan sinyal optik dari FDT ke FAT. Kabel distribusi yang digunakan biasanya adalah jenis *single mode*. Kabel distribusi memiliki kapasitas antara 6 *core* sampai dengan 48 *core*, tergantung dari keperluan kabel yang digunakan.

2.6.5 *Fiber Access Terminal (FAT)*

Fiber Access Terminal (FAT) merupakan tahap lanjut dari keluaran kabel distribusi dari arah FDT yang kemudian terhubung ke masing-masing ONU menggunakan kabel *drop*, atau dengan kata lain FAT digunakan untuk menghubungkan jaringan distribusi ke pelanggan. FAT berfungsi sebagai tempat pendistribusian satu kabel fiber optik (*Drop wire*) yang didalamnya terdapat sebuah *splitter* sebagai pembagi sinyal kabel fiber optik, menjadi beberapa bagian kabel fiber optik lainnya.

Tipe FAT digunakan adalah FAT tipe *wall / on pole*. FAT tipe *wall / on pole* adalah tipe FAT yang biasa dipasang pada dinding atau diatas tiang. FAT jenis ini digunakan untuk instalasi kabel *drop* atas tanah.



Gambar 2.7 *Fiber Access Terminal (FAT)* [7]



Gambar 2.8 *Splitter 1:16* [7]

2.6.6 Drop Wire

Kabel *drop* berfungsi untuk meneruskan sinyal optik dari FAT ke rumah-rumah pelanggan, dimana tipe kabel *drop* yang digunakan adalah tipe G 657 untuk menanggulangi lokasi instalasi yang belokan-belokan sehingga harus menggunakan optik dengan *bending insensitive*.



Gambar 2.9 *Drop Wire* [7]

2.7 Perhitungan Biaya dan Estimasi Waktu Pembangunan FTTH

Dalam memperhitungkan biaya dan estimasi waktu pembangunan jaringan fiber optik seperti *fiber to the home* (FTTH), biasanya banyak hal dilakukan perancang jaringan tersebut sebelum melakukan perhitungan biaya dan estimasi waktunya. Yang pertama yang dilakukan oleh perancang jaringan fiber optik adalah melakukan survei dimulai dari survei potensi pemasaran untuk penjualan layanan sampai dengan survei geologis lokasi untuk menentukan perangkat-perangkat fiber optik yang akan dipakai.

Dalam laporan ini penulis melakukan survei dan wawancara kepada salah satu *provider* (penyedia layanan internet) yang menggunakan fiber optik, yaitu PT. Indonesia Comnets Plus (ICON+) SBU SUMBAGSEL. Proses perancangan jaringan fiber optik FTTH yang dilakukan oleh PT. ICON+ dimulai dari survei yang dilakukan tim penjualan dan pembangunan, lalu hasil survei didata. Hasil data yang telah

didapat maka diberikan kepada tim BOQ (*Bill Order Quantity*) untuk dihitung berapa biaya yang akan dikeluarkan. Dalam perhitungan biaya alat yang akan digunakan, tim BOQ (*Bill Order Quantity*) melakukan penjumlahan, sebagai contoh:

Tabel 2.3 Contoh Perhitungan Biaya FTTH

Nama Perangkat	Jumlah	Harga
Splitter Distribusi 1:4 (inc. box FDT)	2	Rp. 6.000.000
Splitter Distribusi 1:16 (inc. box FAT)	8	Rp. 16.204.176
Splitter Distribusi 1:8 (inc. box FAT)	15	Rp. 23.040.720
<i>Jointbox 24c</i>	1	Rp. 1.145.000
ADSS	4 KM	Rp. 8.000.000
Total		Rp. 54.389.896

Untuk estimasi waktu pada pembangunan jaringan fiber optik FTTH, semua perangkat rata-rata instalasinya memakan waktu 7 hari. Dalam waktu 7 hari tersebut telah termasuk proses survey tim pembangunan pada lokasi yang akan dipasang perangkat pendukung fiber optik, seperti halnya penentuan jalur kabel fiber optik. Dan untuk penarikan kabel memakan waktu 1 hari untuk 1KM kabel.

Dari contoh diatas dapat diambil rumus sederhana dari penjumlahan perangkat-perangkat FTTH yang akan digunakan sebagai berikut:

$$TB/TW = ax + bx + cx \dots$$

Dengan,

- TB/TW: Total Biaya/Total Waktu
- a, b, c...: Estimasi biaya dan estimasi waktu
- x: Total jumlah dari masing-masing perangkat