BAB II
DASAR TEORI

2.1 *IP MULTIMEDIA SUBSYSTEM* (IMS)

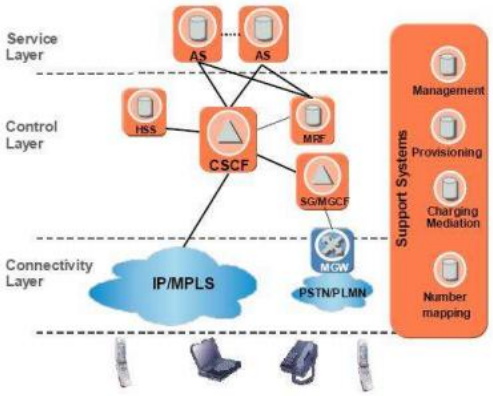
IP Multimedia Subsystem (IMS) merupakan sebuah teknologi komunikasi yang dapat menyatukan antara *wireless* dan *wired* dalam suatu jaringan yang *real time* dan mampu memberikan layanan multimedia secara interaktif. IMS mampu menyediakan layanan aplikasi berupa *streaming* (suara, video, dan gambar) yang lebih kompetitif dengan mobilitas yang lebih besar dan layanan yang lebih baik. Selain itu, IMS juga didesain untuk bekerja tanpa dibatasi area maupun domain yang ada.[4]

Standar *IP Multimedia Subsystem* (IMS) adalah mendefinisikan arsitektur umum yang menawarkan layanan VoIP dan multimedia. Standar IMS mendukung banyak teknologi akses jaringan termasuk GSM, WCDMA, CDMA2000, akses *bandwidth* jaringan tetap dan WLAN. Konsep IMS adalah memberikan layanan internet di manapun dan kapanpun dengan menggunakan teknologi seluler.[5]

2.1.1 Arsitektur IMS

Prinsip kerja IMS adalah dengan menggunakan *session* untuk menangani setiap layanan yang diminta oleh masing-masing pengguna. Jaringan IMS terbagi menjadi 3 lapisan, yaitu *Layer Application Server*, *Layer Session*

Control, dan *Layer Transport and Endpoint*. Arsitektur IMS dapat dilihat pada gambar 2.1 berikut.[4]



Gambar 2.1 Arsitektur *IP Multimedia Subsystem* (IMS)[5]

2.1.1.1 *Layer Application Server*

Layer application server menyediakan layanan *end user logic*. Pada arsitektur IMS dan pensinyalan SIP memiliki kemampuan yang cukup fleksibel untuk mendukung berbagai macam variasi dari *application server*. Layanan yang dilayani *Layer Application Server* adalah layanan *telephony* dan *non telephony*.[6]

2.1.1.2 *Layer Session Control*

Pada *layer session control* terdapat *Call Session Control Funtion* (CSCF). Fungsi CSCF adalah mengatur registrasi dan komunikasi data dari *endpoint* serta proses

routing dari pensinyalan SIP ke *application server* yang akan dituju. *Interworking* antara CSCF dengan lapisan *transport* dan *endpoint* bertujuan untuk menjamin QoS semua layanan yang melaluinya termasuk informasi registrasi *end user* yang sedang melakukan komunikasi, informasi *roaming*, layanan *telephony*, *voice mail*, dan informasi layanan *instant messaging*.

CSCF terbagi menjadi beberapa bagian, antara lain adalah *Proxy Call Session Control Function* (P-CSCF), *Serving Call Session Control Function* (S-CSCF), dan *Interrogating Call Session Control Function* (I-CSCF).[6]

2.1.1.3 *Layer Transport and Endpoint*

Layer Transport and Endpoint berfungsi untuk menginisiasi dan mengakhiri pensinyalan SIP. Terdapat *Realtime Transport Protocol* (RTP) yang digunakan untuk membangun *session* dan menyediakan layanan *bearer* seperti mengkonversi *voice* dari format analog atau digital menjadi paket IP. Untuk mengkonversi VoIP *bearer stream* menjadi format TDM PSTN disediakan media *gateway*-nya.[6]

2.1.2 Komponen IMS

Komponen utama dari IMS adalah *Proxy Call Session Control Function* (P-CSCF), *Serving Call Session Control Function* (S-CSCF), *Interrogating Call Session Control Function* (I-CSCF), dan *Home Subscriber Server* (HSS).

2.1.2.1 *Proxy Call Session Control Function* (P-CSCF)

Proxy Call Session Control Function (P-CSCF) merupakan titik pertama jalur pensinyalan antara terminal IMS dengan jaringan IMS. P-CSCF digunakan sebagai pintu keluar masuknya *server* SIP. P-CSCF berfungsi meneruskan permintaan SIP dan memberikan respon ke arah yang dituju. P-CSCF dapat menghasilkan nomor IPsec yang digunakan untuk keamanan terminal IMS. P-CSCF akan meminta identitas pengguna untuk digunakan pada titik lain di jaringan. Hal tersebut dilakukan P-CSCF pada saat melakukan pemeriksaan pengguna. Titik lain di jaringan tidak melakukan autentifikasi lagi karena sudah dilakukan oleh P-CSCF. P-CSCF juga melakukan pengecekan kebenaran permintaan dari SIP yang dikirim oleh terminal. Pengecekan ini dilakukan untuk menjaga agar terminal IMS terjaga dari permintaan SIP yang tidak sesuai.[6]

SIP merupakan protokol berbasis teks, ukurannya dapat sangat besar, oleh karena itu dalam P-CSCF terdapat compressor dan decompressor pesan SIP. Ketika pesan SIP dapat ditransmisikan melalui koneksi *bandwidth* dalam waktu singkat, maka transmisi pesan SIP yang besar dapat melalui kanal yang kecil dan akan dapat menggunakan waktu yang sebentar untuk melaluinya. Cara yang digunakan untuk mengurangi waktu pada saat mentransmisikan pesan SIP adalah dengan memampatkan pesan SIP di suatu sisi, kemudian dikirimkan, lalu dikembalikan pada satu sisi yang lain.[6]

P-CSCF dapat memasukkan *Policy Decision Function* (PDF) dimana PDF dapat diintegrasikan dengan P-CSCF atau dapat diimplementasikan sebagai unit tersendiri. PDF berfungsi memeriksa sumber jalur media, dan mengatur QoS melalui jalur media. P-CSCF akan melayani sejumlah terminal IMS tergantung pada kapasitas dari suatu titik jaringan.[6]

2.1.2.2 *Serving Call Session Control Function* (S-CSCF)

Serving Call Session Control Function (S-CSCF) merupakan titik sentral dari jalur pensinyalan. S-CSCF mempunyai fungsi sebagai *server* SIP, pengendali sesi, dan sebagai pendaftar SIP. S-CSCF dapat mengetahui

hubungan antara lokasi pengguna dan catatan alamat pengguna SIP.

Diameter digunakan untuk menghubungkan antara S-CSCF dengan HSS. Ada beberapa alasan mengapa S-CSCF berhubungan dengan HSS, yaitu :

1. Untuk mengunduh dari HSS vektor autentifikasi dari pengguna yang mencoba mengakses IMS. S-CSCF menggunakan vektor ini untuk mengautentifikasi pengguna.
2. Untuk mengunduh profil pelanggan dari HSS profil dengan memasukkan profil layanan yang berupa pemicu pesan SIP agar dirutekan melalui satu *server* aplikasi atau lebih.
3. Untuk memberikan informasi kepada HSS bahwa S-CSCF dialokasikan pengguna selama masa registrasi.

S-CSCF juga berfungsi untuk menjaga pengguna dari operasi yang tidak diijinkan.[6]

2.1.2.3 *Interrogating Call Session Control Function* (I-CSCF)

Interrogating Call Session Control Function (I-CSCF) merupakan SIP *proxy* yang terletak pada tepi domain administrasi. Alamat dari I-CSCF terdapat pada *Domain Name System* (DNS). Ketika SIP *server*

mengikuti prosedur SIP untuk mencari tempat SIP selanjutnya, maka SIP *server* mengambil alamat dari I-CSCF sebagai domain tujuan.[6]

I-CSCF dapat melakukan enkripsi dari pesan SIP yang berisi informasi penting seperti jumlah *server*, nama DNS dan kapasitasnya. I-CSCF berfungsi untuk menjaga kerahasiaan jaringan dan mencegah jaringan lain untuk mendapatkan informasi mengenai infrastruktur jaringan.

2.1.2.4 *Home Subscriber Server* (HSS)

Home Subscriber Server (HSS) merupakan tempat penyimpanan utama untuk informasi yang berhubungan dengan pengguna. HSS berisi semua data yang berkaitan dengan pengguna yang diperlukan untuk mengadakan sesi multimedia. Data tersebut berisi informasi lokasi, informasi keamanan, informasi profil pengguna, dan informasi S-CSCF yang telah dialokasikan untuk pengguna. Jika jumlah pendaftar terlalu besar untuk ditangani oleh satu buah HSS, maka dapat digunakan lebih dari satu HSS pada satu jaringan.[6]

2.1.3 *Session Initiation Protocol* (SIP)

Session Initiation Protocol (SIP) merupakan protokol yang digunakan untuk membangun dan

mengatur sesi multimedia jaringan IP yang dikeluarkan oleh *Internet Engineering Task Force* (IETF). SIP telah terpilih sebagai protokol pengontrol sesi untuk IMS. SIP meminjam prinsip desain dari *Simple Mail Transfer Protocol* (SMTP) dan *Hypertext Transfer Protocol* (HTTP) yang termasuk protokol paling sukses di internet. Karena SIP banyak mewarisi karakteristik dari kedua protokol tersebut, SIP terbukti membuat kemudahan dalam membangun layanan baru yang dibawa dengan bobot yang tidak besar.[6]

SIP memiliki kemampuan untuk melakukan *signaling*. SIP juga dapat digunakan untuk mengirimkan pesan dan meng*update* status *presence*. Sebagai protokol *signaling*, SIP digunakan untuk membangun, memodifikasi, dan menterminasi sesi multimedia pada jaringan IP. Sesi dapat diartikan sebagai suatu komunikasi di antara dua pihak dalam waktu tertentu. Contoh dari sesi tersebut adalah telepon, permainan *game* komputer, dan konferensi berbasis *video*.[7]

SIP memiliki beberapa komponen utama, yaitu :

a. *User Agent* (UA)

User Agent (UA) merupakan *end point* dari SIP yang mengirimkan dan menjadi tujuan dari pesan SIP.

UA dapat berupa terminal seperti *handphone*, atau dapat juga berupa *software* seperti yang ada pada laptop atau PC. UA terdiri dari dua jenis, yaitu *User Agent Client* (UAC) dan *User Agent Server* (UAS). *User Agent Client* (UAC) berfungsi untuk menginisiasi pesan dan mengirimkan *request* SIP, sedangkan *User Agent Server* (UAS) berfungsi untuk menerima dan memberikan respon terhadap *request* yang diterima. Respon tersebut seperti menerima, me-*redirect*, menolak, dan mengirimkan *request*. [7]

b. *Proxy Server*

Proxy server merupakan *server* yang bertugas menerima dan meneruskan pesan SIP. *Proxy server* dapat mengubah bagian tertentu dari pesan SIP tanpa mengganggu status dari *request* atau dialog pada sisi *end point*. Pesan SIP yang diterima dapat diteruskan ke lebih dari satu tujuan. *Server* ini disebut *Forking Proxy*.[7]

Ada beberapa jenis *proxy* lain selain *forking proxy*, yaitu *statefull proxy*, dan *stateless proxy*. *Statefull proxy* merupakan *proxy* yang menyimpan status dari *request* yang diterima sampai berakhirnya dialog. Dialog tersebut dapat berlangsung dalam kurun waktu tertentu. *Statefull proxy* dapat menjalankan

beberapa fungsi seperti melakukan *forking*, mengirim ulang *request* dan me-*redirect request*. *Stateless proxy* merupakan *proxy* yang tidak menyimpan *request* yang diterima. *Proxy* ini hanya bertugas meneruskan *request* dan respon yang diterima ke tujuan masing-masing. [7]

c. *Redirect Server*

Redirect server merupakan *server* yang bertugas menerima *request* dan mengirimkan respon berisi alamat sebenarnya dari user tertentu. Pada saat menerima *request*, *redirect server* akan mencari alamat dari tujuan yang terdapat dari *location server*. Alamat yang didapat kemudian akan dikirimkan ke pengirim *request*, dan pengirim akan mengirimkan *request* baru ke alamat yang diperoleh dari *redirect server* tersebut. [7]

d. *Registation Server*

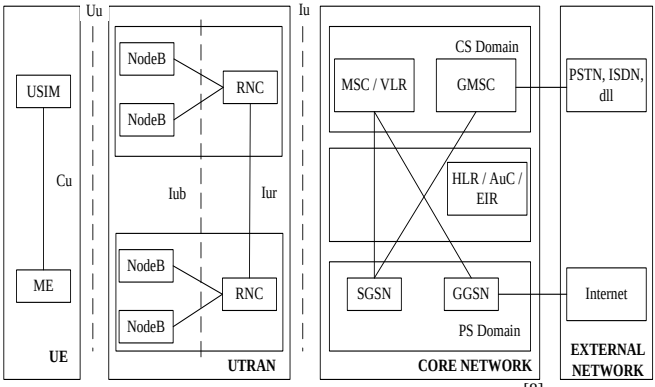
Registation server merupakan SIP *server* yang bertugas untuk menerima proses registrasi dari *user*. *Server* ini menyimpan informasi tentang hubungan antara identitas *user* dengan alamat IP yang dimiliki oleh *device* yang digunakan oleh *user*. Semua informasi tersebut akan tersimpan dalam *location server*.[7]

2.2 TEKNOLOGI JARINGAN SELULER 3G UMTS

Generasi ke tiga teknologi bergerak disebut dengan 3G atau *3rd Generation*. Teknologi 3G merupakan pengembangan dari teknologi sebelumnya yaitu 1G, 2G dan 2,5G yang dikenal dengan teknologi AMPS, CDMA, GSM/GPRS/EDGE. Salah satu teknologi 3G adalah *Universal Mobile Telecommunication Service* (UMTS) yang merupakan teknologi lanjutan dari GSM/GPRS/EDGE di mana salah satu tujuan utamanya adalah untuk memberikan kecepatan akses data yang lebih tinggi dibandingkan dengan GPRS dan EDGE. UMTS memiliki kecepatan akses data mencapai 384 kbps.

Perbedaan antara 3G dan 2G adalah pada *network*-nya, dimana pada 2G *node* yang bertanggung jawab untuk berhubungan langsung dengan *user* adalah *Base Transceiver Station* (BTS) dan *node* yang mengontrol BTS tersebut adalah *Base Station Controller* (BSC). Sedangkan pada 3G, *node* yang bertanggung jawab berhubungan langsung dengan *user* adalah *NodeB* dan yang mengkontrol *NodeB* tersebut adalah *Radio Network Controller* (RNC).[1]

Jaringan UMTS terdiri dari 3 bagian utama, yaitu *User Equipment* (UE), *UMTS Terrestrial Radio Access Network* (UTRAN), dan *Core Network*. Arsitektur jaringan UMTS dapat dilihat pada gambar 2.2 berikut.



Gambar 2.2 Arsitektur jaringan UMTS[8]

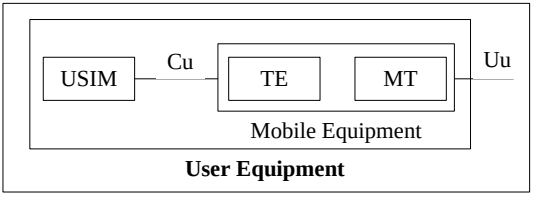
Terdapat beberapa *interface* dalam teknologi UMTS, yaitu :

a. *Interface* Cu, merupakan *interface* yang menghubungkan antara *Mobile Equipment* (ME) dengan *UMTS Subscriber Identity Module* (USIM).
b. *Interface* Uu, merupakan *interface* yang menghubungkan antara *User Equipment* (UE) dengan NodeB.
c. *Interface* Iu, merupakan *interface* yang menghubungkan antara *Radio Network Controller* (RNC) dengan *Core Network*. Iu terbagi menjadi 2, yaitu Iu-CS untuk mendukung data layanan *circuit switch* dan *interface* Iu-PS yang mendukung daya layanan *packet switch*. Untuk *interface* Iu-PS berfungsi sebagai penghubung antara RNC dengan SGSN.
d. *Interface* Iub, merupakan *interface* yang menghubungkan antara RNC dengan NodeB.

e. *Interface* Iur, merupakan *interface* yang menghubungkan antar RNC. [9]

2.2.1 *User Equipment* (UE)

User Equipment (UE) terdiri dari *Mobile Equipment* (ME) dan *UMTS Subscriber Identity Module* (USIM). Antara *Mobile Equipment* (ME) dengan USIM dapat saling terhubung dengan menggunakan *interface* Cu. Konfigurasi dari *User Equipment* (UE) dapat dilihat pada gambar 2.3 berikut.



Gambar 2.3 Konfigurasi *User Equipment* (UE)

Mobile Equipment (ME) merupakan perangkat yang digunakan *user* untuk melakukan komunikasi yang dapat mengirim dan menerima sinyal. ME terdiri dari 2 terminal, yaitu *Mobile Termination* (MT) yang berfungsi untuk mentransmisikan sinyal UE dan *Terminal Equipment* (TE) yang berfungsi sebagai terminal *end-to-end*, misalnya seperti komputer yang dihubungkan dengan ponsel.

UMTS Subscriber Identity Module (USIM) merupakan kartu atau SIM *card* yang berisi nomor dan identitas pelanggan. USIM memiliki kode yang fleksibel, *universal* dan unik sehingga dapat digunakan di mana saja walaupun dengan sistem seluler yang berbeda. [10]

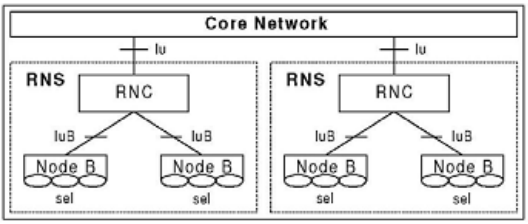
2.2.2 *UMTS Terrestrial Radio Access Network* (UTRAN)

UMTS Terrestrial Radio Access Network (UTRAN) merupakan jaringan akses radio terestrial pada UMTS. UTRAN berfungsi untuk menghubungkan antara UE dengan *Core Network*.

UTRAN terdiri dari *Radio Network Service* (RNS). RNS menghubungkan antara UE dengan UTRAN. RNS berfungsi mengatur pengiriman dan penerimaan sinyal dari setiap sel yang dilayani dan bertanggung jawab untuk mengontrol pergerakan pemakai, sumber radio, *handover*, serta pengiriman dan penerimaan paket data.[10]

Pada setiap RNS terdapat *Radio Network Controller* (RNC) dan NodeB. RNC dihubungkan ke suatu set elemen NodeB di mana masing-masing NodeB dapat melayani satu atau beberapa sel. Untuk menghubungkan RNC dengan NodeB digunakan *interface* Iub. Sedangkan untuk menghubungkan antar RNC digunakan *interface* Iur.[11]

Konfigurasi jaringan UTRAN dapat dilihat pada gambar 2.4 berikut.



Gambar 2.4 Konfigurasi UTRAN

RNC merupakan pengganti BSC pada jaringan GSM. Pada RNC terdapat *Controlling* RNC (CRNC) yang bertanggung jawab mengontrol NodeB. RNC yang menghubungkan UE dengan CN disebut dengan *Serving* RNC (SRNC). SRNC berfungsi untuk mengontrol sumber radio yang digunakan oleh UE saat sedang beroperasi dan berfungsi juga untuk mengakhiri *interface* Iu ke dan dari CN yang digunakan layanan UE. *Drift* RNC (DRNC) merupakan RNC yang digunakan untuk mengontrol pada keadaan dimana *soft handover* terjadi antara NodeB yang dikontrol oleh RNC yang berbeda dan selama atau setelah proses *soft handover* terjadi, sehingga akan ditemukan situasi bahwa UE berhubungan dengan NodeB. Terdapat juga *Serving* RNC (SRNC) *Relocation* yang termasuk RNC untuk mengontrol perpindahan UE dimana SRNC tidak dapat menangani sendiri perpindahan tersebut yang menyebabkan UTRAN

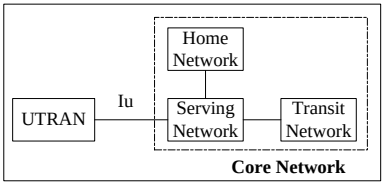
memutuskan untuk mengalihkan hubungan ke RNC lain sehingga dibutuhkan SRNC *Relocation*.[11]

NodeB merupakan unit fisik pada sel yang digunakan untuk mengirimkan atau menerima frekuensi. NodeB berfungsi seperti BTS pada jaringan GSM. *Interface* yang digunakan NodeB untuk berhubungan dengan UE adalah *interface* Uu, sedangkan yang menghubungkan NodeB dengan RCN adalah *interface* Iub. [11]

2.2.3 *Core Network*

Core Network merupakan *unit* kontrol pada jaringan UMTS. *Unit* ini mengontrol seluruh operasi jaringan. *Core Network* bertanggung jawab untuk mempertahankan dan memutuskan proses komunikasi yang sedang dilakukan pelanggan.[8]

Untuk melakukan layanan yang banyak, *core network* terbagi menjadi tiga divisi, yaitu *Serving Network*, *Home Network*, dan *Transit Network*. Ke-tiga divisi tersebut dapat dilihat pada gambar 2.5 berikut.



Gambar 2.5 Konfigurasi *core network*

Home network biasa disebut sebagai jaringan asal dari USIM yang berfungsi memberikan identitas dan lokasi pemakai. Identitas pemakai dapat dikenali walaupun berada pada daerah dengan sistem dan operator seluler yang berbeda. *Serving Network* berfungsi untuk melakukan akses jaringan untuk menyediakan koneksi dengan UE dan bertanggung jawab terhadap *call routing* serta pengiriman dan penerimaan data. Adanya hubungan antara *home network* dan *serving network* adalah untuk memenuhi spesifikasi data dan layanan dari UE yang sedang aktif sedangkan hubungan antara *serving network* dengan *transit network* adalah untuk menangani UE yang tidak aktif. *Transit network* berada diantara *serving network* dan *remote part*. Jika terdapat *call*, *remote network* berada di dalam jaringan yang menangani UE lalu *transit network* akan aktif jika UE berpindah sel (*handover*).

Pada *core network* atau jaringan inti berisi identifikasi pelanggan dan informasi lokasi pelanggan yang dapat mendukung layanan UMTS. *Core network* pada jaringan UMTS terdiri dari 2 tipe *switching*, yaitu *Circuit Switching* (CS) yang bersifat *real time* dan *Packet Switching* (PS) yang bersifat *non real time*.

2.2.3.1 *Circuit Switched Domain* (CS-*Domain*)

CS-*Domain* terdiri dari :

a. *Mobile Switching Centre* (MSC), berfungsi untuk mengontrol trafik pada jaringan inti.
b. *Visitor Location Register* (VLR), berfungsi untuk menyimpan informasi pelanggan secara sementara yang diperlukan MSC untuk melayani pelanggan dari area lain.
c. *Gateway* MSC (GMSC), merupakan gerbang penghubung antara jaringan UMTS dengan jaringan luar seperti PSTN atau telepon rumah yang bersifat *real time*. [10]

2.2.3.2 *Packet Switched Domain* (PS-*Domain*)

PS-*Domain* bertujuan untuk mendukung *routing* paket. PS-*Domain* terhubung dengan *packet switched network* seperti internet. Yang mendukung *packet switching* adalah *Serving GPRS Support Node* (SGSN) dan *Gateway GPRS Support Node* (GGSN).

a. *Serving GPRS Support Node* (SGSN)

SGSN memiliki tugas dalam proses *routing* paket yang datang dari atau menuju pelanggan melalui *radio access network*. SGSN juga bertanggung jawab untuk autentifikasi pelanggan, update lokasi pelanggan, enkripsi dan dekripsi, pembentukkan, pemeliharaan, dan pemutusan komunikasi. [10]

b. *Gateway GPRS Support Node* (GGSN)

GGSN merupakan komponen utama untuk mendukung *packet switched* yang bertugas sebagai gerbang *Packet Data Network* (PDN) dengan jaringan luar seperti internet dan bertugas juga untuk menentukan alamat IP. PDN terhubung dengan jaringan IP eksternal untuk dapat me*routing* paket. PDN menggunakan protokol *Packet Data Protocol* (PDP). Untuk dapat mengirimkan paket data ke jaringan IP eksternal, GGSN mengkonversi terlebih dahulu paket data ke format PDP. [10]

2.3 *VIDEO CONFERENCE*

Video conference termasuk dalam salah satu aplikasi multimedia yang memungkinkan komunikasi berupa data, suara, dan gambar yang bersifat *real time* dan dapat dilakukan dua arah (*duplex*). *Video conference* banyak digunakan untuk proses komunikasi jarak jauh sebagai pengganti tatap muka. *Video conference* dapat dilakukan dengan menggunakan komputer atau laptop yang sudah memiliki *webcam*, bahkan saat ini *video conference* sudah dapat dilakukan di mana saja dan kapan saja dengan menggunakan *smartphone*. Dengan menggunakan *single* IP *address* maka *video conference* dapat dilakukan antara dua buah komputer. [12]

Webcam yang digunakan untuk melakukan *video conference* memiliki resolusi gambar yang beraneka ragam. Semakin berkembangnya teknologi, kualitas resolusi dari *webcam* juga berkembang yang dulunya memiliki resolusi 160 x 120, sekarang sudah ada *webcam* dengan resolusi beberapa *megapixel*. Semakin besar resolusi maka semakin besar jumlah data yang dikirim sehingga *bandwidth* yang dibutuhkan juga semakin besar. Hal lain yang mempengaruhi ukuran data adalah *frame rate*. *Frame rate* merupakan jumlah gambar yang dikirimkan tiap detik. Semakin besar nilai *frame rate* maka video yang dihasilkan akan semakin baik. Dalam *video conference*, *microphone* dapat digunakan untuk *input*-an audio. [13]

Menurut Gough (2006), *video conference* dapat dibagi menjadi 3 jenis, yaitu :

1. *Personal Video Conferencing*, jenis ini melibatkan dua orang yang berinteraksi antara satu dengan yang lain. Pada jenis ini terdapat komunikasi berupa video dan audio yang memungkinkan juga ditambahkannya pengiriman berupa teks. Contoh dari pengaplikasian *video conference* jenis ini adalah pada *software instant messaging* (IM) seperti Yahoo! Messenger.
2. *Business Video Conferencing*, jenis ini tidak jauh berbeda dari jenis *Personal Video Conferencing* hanya saja terdapat tambahan fitur, yaitu kemampuan untuk berkomunikasi tidak

hanya oleh dua orang saja tetapi dapat lebih, dapat melakukan *file sharing*, dapat melakukan presentasi, dan terdapat juga fasilitas *whiteboard*, serta fitur-fitur lainnya. Biaya yang dibutuhkan untuk melakukan *video conference* jenis ini cukup besar.

3. *Web Video Conferencing*, merupakan jenis *video conference* yang dilakukan satu arah. *Web video conferencing* terdapat pada halaman *web* dan biasa digunakan untuk seminar dengan menggunakan *web.* Peserta seminar dapat melihat video yang dikirimkan oleh pembicara seminar tetapi peserta tidak dapat mengirimkan videonya kepada pengirim. [13]

2.4 *QUALITY OF SERVICE* (QOS)

Quality of Service (QoS) merupakan sebuah metode pengukuran seberapa baik jaringan. QoS digunakan untuk mengukur kinerja dari jaringan yang telah dispesifikasikan dengan suatu servis. QoS ditujukan untuk membantu *end user* mendapatkan kinerja yang lebih handal sehingga dapat menyediakan layanan yang lebih baik melalui teknologi yang berbeda-beda. [14]

2.4.1 Teknik *Quality of Service* (QoS)

Terdapat tiga jenis metode QoS yang sering digunakan, yaitu *best-effort service, integrated service*, dan *differentiated service*.

2.4.1.1 *Best-Effort Service*[2]

Best-Effort Service digunakan untuk melakukan semua usaha agar dapat mengirimkan sebuah paket ke suatu tujuan. Metode ini tidak menjamin bahwa paket akan sampai ke tujuan yang dimaksud. Metode ini tidak dapat digunakan untuk paket yang sensitif terhadap *network delay*, fluktuasi *bandwidth*, dan perubahan kondisi jaringan. Sebagai contoh adalah aplikasi telepon pada jaringan yang membutuhkan *bandwidth* yang tetap, agar dapat berfungsi dengan baik, dalam hal ini, penerapan *best-effort service* akan mengakibatkan panggilan telepon terputus atau gagal.

2.4.1.2 *Integrated Service* (*IntServ*) [2]

Integrated Service (*IntServ*) menyediakan aplikasi dengan tingkat jaminan layanan melalui negosiasi parameter jaringan secara *end-to-end*. Aplikasi akan meminta tingkat layanan yang dibutuhkan untuk dapat beroperasi dan bergantung pada mekanisme QoS untuk menyediakan sumber daya jaringan yang dimulai sejak awal transmisi. Aplikasi tidak akan mengirimkan trafik sebelum menerima tanda bahwa jaringan mampu menerima beban yang akan dikirimkan dan mampu menyediakan QoS yang diminta secara *end-to-end*. Suatu

jaringan akan melakukan proses *admission control* yang merupakan metode untuk mencegah jaringan mengalami *over loaded*.

Jika QoS yang diminta tidak disediakan, maka jaringan tidak akan mengirimkan tanda ke aplikasi agar dapat memulai pengiriman data. Jika QoS disediakan dan aplikasi memulai mengirimkan data, maka sumber daya pada jaringan yang sudah dipesan akan terus dikelola secara *end-to-end* sampai aplikasi tersebut selesai. Metode *IntServ* ditujukan untuk aplikasi yang peka terhadap *delay* dan keterbatasan *bandwidth*. Contoh aplikasi tersebut adalah *video conference* dan VoIP. Metode *IntServ* tidak tepat digunakan untuk aplikasi semacam web karena aliran trafik datanya banyak, tetapi datanya kecil.

Arsitektur pada medote *IntServ* berdasarkan pada sistem pencadangan sumber daya per aliran trafik. Setiap aplikasi harus mengajukan permintaan *bandwidth* untuk dapat melakukan transmisi data. Sistem pemesanan sumber daya memerlukan protokol sendiri. Salah satu protokolnya adalah RSVP.

2.4.1.3 *Differentiated Service* (*DiffServ*)

Differentiated Service (*DiffServ*) menyediakan suatu set perangkat klasifikasi dan mekanisme antrian

terhadap protokol atau aplikasi dengan prioritas tertentu di atas jaringan yang berbeda. *DiffServ* bergantung pada kemampuan *edge router* untuk memberikan klasifikasi dari paket-paket yang berbeda tipenya. [2]

Keuntungan dengan menggunakan metode *DiffServ* adalah scalability dimana metode *DiffServ* mampu mengumpulkan banyak *flow* sehingga dapat menangani jumlah *flow* yang besar serta dapat digunakan pada kecepatan yang tinggi. Selain itu metode *DiffServ* juga sederhana dan mudah untuk diterapkan karena tidak banyak menyimpang dari dasar IP. [2]

Arsitektur pada medote *DiffServ* terdiri dari tiga komponen, yaitu *policy* dan *resource manager*, *edge routers*, dan *core routers*. *Policy* dan *resource manager* bertugas untuk membuat kebijakan-kebijakan dan mendistribusikannya kepada *DiffServ* router. *Edge routers* bertugas memberikan tanda pada paket dengan sebuah kode *point* sesuai dengan kebijakan yang telah dispesifikasikan oleh administrator jaringan. Sedangkan *core routers* bertugas untuk memeriksa paket datang yang sebelumnya telah diberi tanda dengan *code point* oleh *edge router*. [2]

Diffserv menggunakan mekanisme QoS dengan membagi trafik atas kelas-kelas yang berbeda dan tiap

kelas tersebut juga memiliki perlakuan yang berbeda-beda. *Diffserv* melakukan identifikasi kelas dengan memasang kode yang disebut dengan *Differentiated Service Code Point* (DSCP). Perlakuan yang berbeda-beda ini disebut dengan *Per-Hop Behavior* (PHB). PHB merupakan suatu mekanisme *forwarding* paket yang dilakukan tiap *node diffserv*. PHB digunakan untuk mengidentifikasi perlakuan yang akan diberikan kepada sebuah *flow* khusus. Berikut merupakan standar PHB yang biasa digunakan :[15]

1. *Expedited Forwarding* (EF), digunakan untuk trafik layanan *real time* dengan prioritas tertinggi, bersifat *low delay*, *low loss*, dan *low latency*. Contoh yang menggunakan kode ini adalah VoIP.
2. *Assured Forwarding* (AF), digunakan untuk trafik layanan yang lebih rendah yang pengirimannya masih memberikan toleransi terhadap *delay* dan *jitter* tetapi masih memerlukan jaminan *bandwidth*.
3. *Class Selector* (CS), digunakan untuk mempertahankan kompatibilitas dengan IP *precedence field*.
4. *Default* PHB (DF), digunakan untuk trafik yang bersifat *best effort*.

DSCP merupakan pengembangan dari *Type of Service* (ToS). Pada awalnya ToS digunakan untuk menyediakan QoS pada jaringan IP, namun setalah munculnya *diffserv* model, ToS telah digantikan dengan *IP Precedence* atau nilai DSCP. Berikut merupakan nilai IP *Precedence* yang digunakan untuk menentukan prioritas suatu layanan. [15]

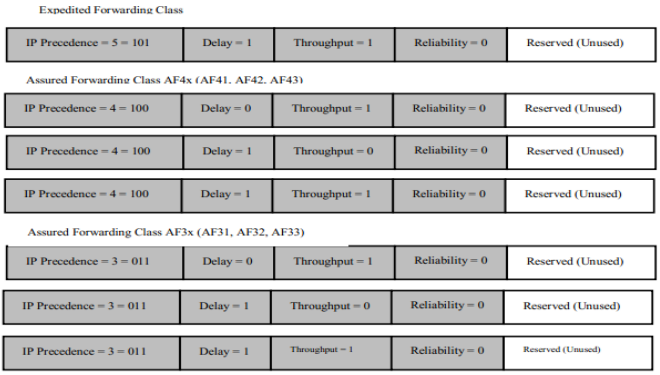
Tabel 2.1 Nilai-Nilai IP *Precedence*[15]

IP *Precedence*	**Binary**	**Priority**
0	000	*Routine*
1	001	*Priority*
2	010	*Immediate*
3	011	*Flash*
4	100	*Flash override*
5	101	*Critical*
6	110	*Internetwork Control*
7	111	*Network Control*

Dari nilai-nilai IP *Precedence* yang menjadi prioritas paling penting adalah *Critical*, *Flash override*, dan *Flash*. Pada umumnya *Critical* (5) digunakan untuk trafik VoIP atau trafik yang bersifat *realtime* / *time sensitive*. *Flash override* (4) digunakan untuk trafik *video*, sedangkan *Flash* (3) digunakan untuk multimedia

streaming. Trafik yang lain pada umumnya dikelompokkan ke *Routine* (0) atau trafik *best effort*. [15]

Berikut merupakan pembagian kelas-kelas dalam DSCP.



Gambar 2.6 Kelas-Kelas DSCP [15]

Nilai dari DSCP adalah IP *Precedence* ditambahkan dengan *delay*, *throughput*, dan *reliability*. Pada DSCP, variabel *delay* dan *throughput* disebut dengan *drop probability*. Nilai DSCP yang biasa digunakan adalah pada kelas *Expedited Forwarding* (EF) dan *Assured Forwarding* (AF). [15]

Berikut merupakan nilai DSCP dan kelas layanannya.

Tabel 2.2 Nilai DSCP dan Kelas Layanan[15]

Nama	**Biner**	**Desimal**	***IP Prec.***	***Drop Pecedence***	**Layanan**
CS0	000 000	0	0		Standar (DNS, DHCP)

Tabel 2.3 Nilai DSCP dan Kelas Layanan[15] (Lanjutan)

Nama	**Biner**	**Desi-mal**	***IP Prec.***	***Drop Prec.***	**Layanan**
CS0	000 000	0	0		Standar (DNS, DHCP)
CS1	001 000	8	1		*Low Priority Data* (semua trafik yang tidak mendapat jaminan *bandwidth*)
AF11	001 010	10	1	*Low*	*High Throughput Data* (*Transfer File, email, store and forward application*)
AF12	001 100	12	1	*Medium*	
AF13	001 110	14	1	*High*	
CS2	010 000	16	2		OAM (OAM&P)
AF21	010 010	18	2	*Low*	Data *latency* rendah (transaksi web, *transfer* keuangan)
AF22	010 100	20	2	*Medium*	
AF23	010 110	22	2	*High*	
CS3	011 000	24	3		*Broadcast Video* (*broadcast* TV&*live events, video surveillance, video on demand*)

Tabel 2.4 Nilai DSCP dan Kelas Layanan[15] (Lanjutan)

Nama	**Biner**	**Desi-mal**	***IP Prec.***	***Drop Prec.***	**Layanan**
AF31	011 010	26	3	*Low*	*Multimedia Streaming (Buffered streaming audio, webcast)*
AF32	011 100	28	3	*Medium*	
AF33	011 110	30	3	*High*	
CS4	100 000	32	4		*Real-time interactive* (*video conference*, permainan interaktif)
AF41	100 010	34	4	*Low*	*Multimedia Conferencing* (H323/v2 *video conferencing*)
AF42	100 100	36	4	*Medium*	
AF43	100 110	38	4	*High*	
CS5	101 000	40	5		*Signaling* (*Peer-to-peer IP*, *IP telephony signaling*)
EF	101 110	46	5		*Telephony* (*VoIP, Voice*)
CS6	110 000	48	6	*Routing*	Network Control (Network Routing)
CS7	111 000	56	7	*Network*	

2.4.2 Parameter *Quality of Service* (QoS)

2.4.2.1 *End-to-End Delay*

End-to-End Delay merupakan total waktu yang dibutuhkan suatu informasi atau data dari pengirim ke penerima pada suatu jaringan. Beberapa hal yang dapat mempengaruhi *delay* adalah jarak, media fisik, dan waktu. Semakin besar *delay* maka semakin menurun nilai QoS. [14]

Delay dapat dihitung dengan menggunakan persamaan 2.1 berikut.

$$Delay\ rata-rata = \left(\frac{total\ \ delay}{total\ \ paket\ \ yang\ \ diterima}\right) \quad \text{.....(2.1)}$$

Berikut merupakan standarisasi nilai *End-to-End Delay* menurut ITU-T G.114.

Tabel 2.5 Standarisasi *End-to-End Delay* [16]

***Delay* (ms)**	**Keterangan**	**Kategori**
0 – 150	Dapat diterima	Baik
150 – 400	Dapat diterima, namun administrator jaringan harus waspada terhadap segala sesuatu yang dapat mempengaruhi kualitas jaringan.	Cukup
> 400	Secara umum tidak dapat diterima, namun untuk kasus-kasus khusus nila batas ini dapat berubah	Buruk

2.4.2.2 *Jitter*

Jitter merupakan variasi *delay* yang terjadi antar paket pada jaringan. Nilai *jitter* dipengaruhi oleh besarnya tumbukan antar-paket dan variasi beban trafik dalam suatu jaringan. Beban trafik yang semakin besar akan menyebabkan tumbukan antar-paket atau yang biasa disebut *congestion*, sehingga nilai *jitter* juga akan semakin besar. Semakin kecil nilai *jitter* maka semakin baik kualitas dari layanannya. [14]

Jitter dapat dihitung dengan menggunakan persamaan 2.2 dan 2.3 berikut.

$$Jitter = \frac{total\ \ variasi\ \ delay}{total\ \ paket\ \ yang\ \ diterima} \quad \text{.....(2.2)}$$

$$Total\ variasi\ delay = Delay - (rata - rata\ delay) \quad \text{.....(2.3)}$$

Berikut merupakan standarisasi nilai *Jitter* menurut ITU-T G.114.

Tabel 2.6 Standarisasi *Jitter* [16]

***Jitter* (ms)**	**Keterangan**	**Kategori**
0 – 20	Dapat diterima	Baik
20 - 50	Dapat diterima	Cukup
> 50	Tidak dapat diterima	Buruk

2.4.2.3 *Packet Loss*

Packet Loss merupakan jumlah paket yang hilang saat proses pengiriman terjadi. *Packet loss* dapat disebabkan karena *collision* dan *congestion* yang dapat mempengaruhi semua aplikasi dan akan mengurangi

efisiensi jaringan meskipun jumlah *bandwidth* yang dimiliki aplikasi tersebut tercukupi. Semakin kecil nilai *packet loss* maka semakin baik kualitas layanannya.

Nilai *packet loss* biasa dinyatakan dalam persen (%) yang dapat dihitung dengan persamaan 2.4 berikut. [17]

$$Packet\ Loss = \frac{packet\ \ sent - packet\ \ received}{packet\ \ sent} \times 100\% \quad (2.4)$$

Berikut merupakan standarisasi nilai *Packet Loss* menurut ITU-T G.114.

Tabel 2.7 Standarisasi *Packet Loss* [16]

***Packet Loss* (%)**	**Keterangan**	**Kategori**
0 – 1	Dapat diterima	Baik
1 – 5	Dapat diterima	Cukup
> 10	Tidak dapat diterima	Buruk

2.4.2.4 *Throughput*

Throughput merupakan jumlah bit yang berhasil dikirim pada suatu jaringan. *Throughput* juga merupakan kecepatan transfer data yang diukur dalam satuan *bit per second* (bps). *Throughput* dapat diketahui dari total kedatangan paket yang sukses sampai tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut seperti pada persamaan 2.5 berikut.[2]

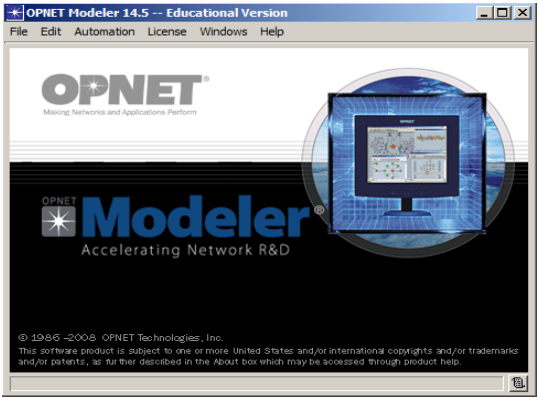
$$Throughput = \frac{Paket\ \ data\ \ yang\ \ diterima}{Lama\ \ pengamatan} \quad (2.5)$$

2.5 OPNET MODELER 14.5

Optimized Network Engineering Tools (OPNET) merupakan sebuah *software* yang digunakan untuk melakukan simulasi jaringan komunikasi yang memungkinkan digunakannya berbagai macam terknologi jaringan. OPNET digunakan untuk merancang jaringan komunikasi dan menguji performa dari jaringan tersebut. Dari hasil pengujian, OPNET dapat menghasilkan keluaran berupa grafik yang dapat dilihat lebih detail dengan mengkonversi grafik tersebut ke dalam *excel*.

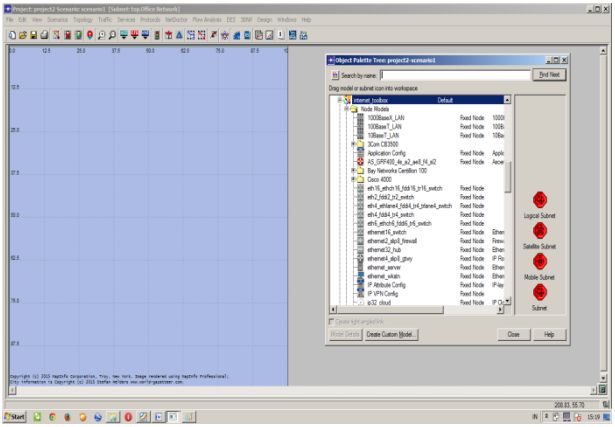
OPNET digunakan oleh banyak perusahaan perlengkapan jaringan di dunia untuk meningkatkan desain dari *network devices*, seperti teknologi VoIP, TCP, OSPFv3, MPLS, IPv6, dan lain sebagainya. [12]

Tampilan OPNET Modeler 14.5 dapat dilihat pada gambar 2.7 berikut.



Gambar 2.7 Tampilan OPNET Modeler 14.5

Simulasi jaringan dengan menggunakan OPNET Modeler 14.5 dapat dilakukan pada *Project Editor*. Terdapat berbagai macam model jaringan yang dapat digunakan, antara lain *World*, *Enterprise*, *Campus*, *Office*, *Logical*, dan *Choose from maps*. Gambar 2.8 berikut merupakan contoh gambar tampilan *Project Editor* dengan model *Office Network* beserta tampilan *Object Pallete* untuk memasukkan komponen-komponen yang digunakan dalam membuat suatu jaringan yang diinginkan.



Gambar 2.8 Tampilan *Project Editor* model *Office Network* dan Tampilan *Object Pallete*

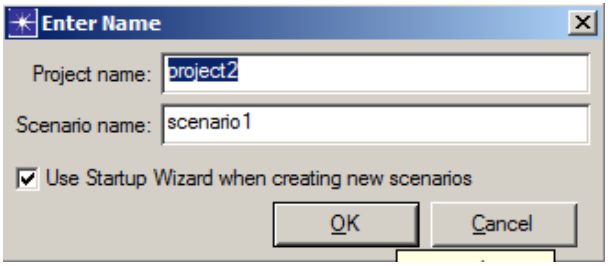
Terdapat berbagai macam menu pada *Project Editor*. Menu-menu tersebut digunakan untuk membuat dan menjalannya suatu model jaringan. Setiap menu memiliki fungsinya masing-masing. Pada menu bar terdapat menu *File*, *Edit*, *View*, *Scenarios*,

Topology, *Traffic*, *Services*, *Protocols*, *NetDoctor*, *Flow Analysis*, *DES*, *3DNV*, *Design*, *Windows*, dan *Help*. Tampilan dari menu bar dapat dilihat pada gambar 2.9 berikut.

Gambar 2.9 Tampilan menu bar

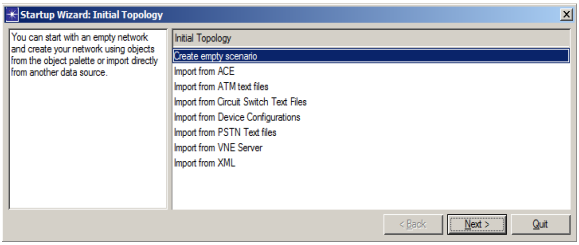
Untuk membuat suatu topologi jaringan pada OPNET Modeler 14.5, berikut langkah-langkah yang harus dilakukan :

1. Membuka OPNET Modeler 14.5 dilakukan dengan cara memilih *Start* – *Program* – OPNET Modeler 14.5.
2. Menu *File* – *New* – *Project* digunakan untuk membuat *file project* baru.
3. Memasukkan nama *project* pada *project name*, kemudian klik OK.



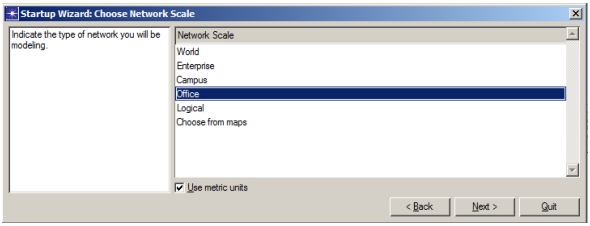
Gambar 2.10 Memasukkan nama *project* baru

4. Akan muncul *Initial Topology*, pilih *Create Empty Scenario* kemudian klik *Next*.



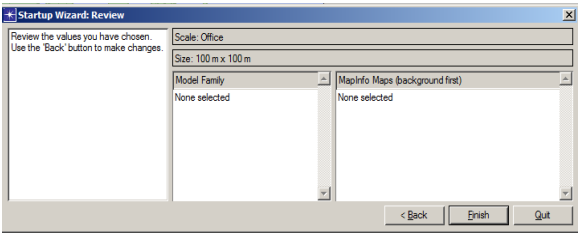
Gambar 2.11 *Initial Topology*

5. Berikutnya dilakukan pemilihan *network scale*, pilih *Office* kemudian klik *Next*.



Gambar 2.12 *Choose Network Scale*

6. Mengisikan *specify size*, kemudian klik *Next* terus sampai terdapat pilihan *Finish*.



Gambar 2.13 *Review – Finish*

7. *Object Palette* akan muncul, kemudian dapat dilakukan pemilihan komponen yang digunakan untuk membentuk suatu topologi jaringan.