

# Definisi dan Klasifikasi Struktur Sistem Operasi

# Danang Kurniawan 195120031

Fakultas Komputer danang.student@umitra.ac.id

#### Isi Resume:

Suatu sistem operasi modern merupakan suatu sistem yang besar dan kompleks sehingga strukturnya harus dirancang dengan hati-hati dan saksama supaya dapat berfungsi seperti yang diinginkan serta dapat dimodifikasi dengan mudah. Struktur sistem operasi merupakan komponen-komponen sistem operasi yang dihubungkan dan dibentuk di dalam kernel. Ada beberapa struktur sistem operasi dan pernah dicoba, diantaranya sebagai berikut



#### 1. Struktur Monolithic

Struktur sistem operasi di sistem ini tidak terstruktur. Sistem operasi sebagai kumpulan prosedur yang masing-masing dapat saling dipanggil jika dibutuhkan. Setiap prosedur yang ada di dalam sistem ini mempunyai interface yang sudah didefinisikan dengan baik. Dalam hal ini berupa parameter dan hasilnya, serta masing-masing prosedur bebas untuk saling memanggil jika dibutuhkan. Walaupun disebut tidak berstruktur, sebenarnya sistem monilisthic tetap mempunyai struktur walaupun kecil dan mendasar.

Contoh dari sistem operasi ini adalah MS-DOS dan UNIX. MS-DOS merupakan sistem operasi yang menyediakan fungsional dalam ruang yang sedikit sehingga tidak dibagi menjadi beberapa modul, sedangkan UNIX menggunakan struktur *monolitik* dimana prosedur dapat saling dipanggil oleh prosedur lain di sistem bila diperlukan dan kernel berisi semua layanan yang

disediakan sistem operasi untuk pengguna. Inisialisasi-nya terbatas pada fungsional perangkat keras yang terbagi menjadi dua bagian yaitu kernel dan sistem program. Kernel terbagi menjadi serangkaian interface dan device driver dan menyediakan sistem file, penjadwalan CPU, manajemen memori, dan fungsi-fungsi sistem operasi lainnya melalui system calls.

#### Evolusi:

Kebanyakan UNIX sampai saat ini berstruktur monolitik. Meskipun monolitik, yaitu seluruh komponen/subsistem sistem operasi terdapat di satu ruang alamat tetapi secara rancangan adalah berlapis. Rancangan adalah berlapis yaitu secara logik satu komponen/subsistem merupakan lapisan lebih bawah dibanding lainnya dan menyediakan layanan-layanan untuk lapisan-lapisan lebih atas. Komponen-komponen tersebut kemudia dikompilasi dan dikaitkan (di-link) menjadi satu ruang alamat. Untuk mempermudah dalam pengembangan terutama pengujian dan fleksibilitas, kebanyakan UNIX saat ini menggunakan konsep kernel loadable modules,yaitu:

- Bagian-bagian kernel terpenting berada di memori utama secara tetap.
- Bagian-bagian esensi lain berupa modul yang dapat ditambahkan ke kernel saat diperlukan dan dicabut begitu tidak digunakan lagi di waktu jalan (run time).

Contoh: UNIX berstruktur monolitik, MS-DOS

# 2. Sistem Berlapis (layered system)

Lapis-5	:	user program
Lapis-4	:	buffering untuk I/O device
Lapis-3	:	operator-console device driver
Lapis-2	1	menejemen memori
Lapis-1	:	penjadwalan CPU
Lapis-0	:	hardware

Lapis-6	:	user program
Lapis-5	:	device driver dan sceduler
Lapis-4	:	virtual memory
Lapis-3	:	I/O channel
Lapis-2	:	penjadwalan CPU
Lapis-1	:	instruksi interpreter
Lapis-0	:	hardware

Sistem operasi dibentuk secara hirarki berdasar lapisan-lapisan, dimana lapisan-lapisan bawa memberi layanan lapisan lebih atas. Lapisan yang paling bawah adalah perangkat keras, dan yang paling tinggi adalah userinterface. Sebuah lapisan adalah implementasi dari obyek abstrak yang merupakan enkapsulasi dari data dan operasi yang bisa memanipulasi data tersebut. Struktur berlapis dimaksudkan untuk mengurangi kompleksitas rancangan dan implementasi sistem operasi. Tiap lapisan mempunyai fungsional dan antarmuka masukan-keluaran antara dua lapisan bersebelahan yang terdefinisi bagus.

Sedangkan menurut Tanenbaum dan Woodhull, sistem terlapis terdiri dari enam lapisan, yaitu:

Lapis 5 – The operator

Berfungsi untuk pemakai operator.

Lapis 4 – User programs

Berfungsi untuk aplikasi program pemakai.

Lapis 3 – I/O management

Berfungsi untuk menyederhanakan akses I/O pada level atas.

Lapis 2 -Operator-operator communication

Berfungsi untuk mengatur komunikasi antar proses.

Lapis 1 -Memory and drum management

Berfungsi untuk mengatur alokasi ruang memori atau drum magnetic.

Lapis 0 -Processor allocation and multiprogramming

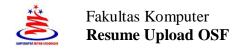
Berfungsi untuk mengatur alokasi pemroses dan switching, multi programming dan pengaturan prosessor.

Menurut Stallings, model tingkatan sistem operasi yang mengaplikasikan prinsip ini dapat dilihat pada tabel berikut, yang terdiri dari level-level dibawah ini:

## Level 1

Terdiri dari sirkuit elektronik dimana obyek yang ditangani adalah register memory cell, dan gerbang logika. Operasi pada obyek ini seperti membersihkan register atau membaca lokasi memori.

Level 2



Pada level ini adalah set instruksi pada prosesor. Operasinya adalah instruksi bahasa-mesin, seperti menambah, mengurangi, load dan store.

#### Level 3

Tambahan konsep prosedur atau subrutin ditambah operasi call atau return.

#### Level 4

Mengenalkan interupsi yang menyebabkan prosesor harus menyimpan perintah yang baru dijalankan dan memanggil rutin penanganan interupsi. Empat level pertama bukan bagian sistem operasi tetapi bagian perangkat keras. Meski pun demikian beberapa elemen sistem operasi mulai tampil pada level-level ini, seperti rutin penanganan interupsi. Pada level 5, kita mulai masuk kebagian sistem operasi dan konsepnya berhubungan dengan multi-programming.

## Level 5

Level ini mengenalkan ide proses dalam mengeksekusi program. Kebutuhan-kebutuhan dasar pada sistem operasi untuk mendukung proses ganda termasuk kemampuan men-suspend dan me-resume proses. Hal ini membutuhkan register perangkat keras untuk menyimpan agar eksekusi bisa ditukar antara satu proses ke proses lainnya.

## Level 6

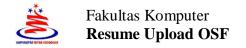
Mengatasi penyimpanan sekunder dari komputer. Level ini untuk menjadualkan operasi dan menanggapi permintaan proses dalam melengkapi suatu proses.

## Level 7

Membuat alamat logik untuk proses. Level ini mengatur alamat virtual ke dalam blok yang bisa dipindahkan antara memori utama dan memori tambahan. Cara-cara yang sering dipakai adalah menggunakan ukuran halaman yang tetap, menggunakan segmen sepanjang variabelnya, dan menggunakan cara keduanya. Ketika blok yang dibutuhkan tidak ada dimemori utama, alamat logis pada level ini meminta transfer dari level 6. Sampai point ini, sistem operasi mengatasi sumber daya dari prosesor tunggal. Mulai level 8, sistem operasi mengatasi obyek eksternal seperti peranti bagian luar, jaringan, dan sisipan komputer kepada jaringan.

## Level 8

Mengatasi komunikasi informasi dan pesan-pesan antar proses. Dimana pada level 5 disediakan mekanisme penanda yang kuno yang memungkinkan untuk sinkronisasi proses, pada level ini mengatasi pembagian informasi yang lebih banyak. Salah satu peranti yang paling sesuai adalah pipe (pipa) yang menerima output suatu proses dan memberi input ke proses lain.



#### Level 9

Mendukung penyimpanan jangka panjang yang disebut dengan berkas. Pada level ini, data dari penyimpanan sekunder ditampilkan pada tingkat abstrak, panjang variabel yang terpisah. Hal nini bertentangan tampilan yang berorientasikan perangkat keras dari penyimpanan sekunder.

#### Level 10

Menyediakan akses ke peranti eksternal menggunakan antarmuka standar.

#### Level 11

Bertanggung-jawab mempertahankan hubungan antara internal dan eksternal identifier dari sumber daya dan obyek sistem. Eksternal identifier adalah nama yang bisa dimanfaatkan oleh aplikasi atau pengguna. Internal identifier adalah alamat atau indikasi lain yang bisa digunakan oleh level yang lebih rendah untuk meletakkan dan mengontrol obyek.

## Level 12

Menyediakan suatu fasilitator yang penuh tampilan untuk mendukung proses. Hal ini merupakan lanjutan dari yang telah disediakan pada level 5. Pada level 12, semua info yang dibutuhkan untuk managemen proses dengan berurutan disediakan, termasuk alamat virtual di proses, daftar obyek dan proses yang berinteraksi dengan proses tersebut serta batasan interaksi tersebut, parameter yang harus dipenuhi proses saat pembentukan, dan karakteristik lain yang mungkin digunakan sistem operasi untuk mBerfungsi untuk mengatur alokasi ruang memori atau drum magnetic.

Lapis 0 -Processor allocation and multiprogramming

Berfungsi untuk mengatur alokasi pemroses dan switching, multi programming dan pengaturan prosessor.

Menurut Stallings, model tingkatan sistem operasi yang mengaplikasikan prinsip ini dapat dilihat pada tabel berikut, yang terdiri dari level-level dibawah ini:

#### Level 1

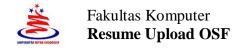
Terdiri dari sirkuit elektronik dimana obyek yang ditangani adalah register memory cell, dan gerbang logika. Operasi pada obyek ini seperti membersihkan register atau membaca lokasi memori.

## Level 2

Pada level ini adalah set instruksi pada prosesor. Operasinya adalah instruksi bahasa-mesin, seperti menambah, mengurangi, load dan store.

# Level 3

Tambahan konsep prosedur atau subrutin ditambah operasi call atau return.



## Level 4

Mengenalkan interupsi yang menyebabkan prosesor harus menyimpan perintah yang baru dijalankan dan memanggil rutin penanganan interupsi. Empat level pertama bukan bagian sistem operasi tetapi bagian perangkat keras. Meski pun demikian beberapa elemen sistem operasi mulai tampil pada level-level ini, seperti rutin penanganan interupsi. Pada level 5, kita mulai masuk kebagian sistem operasi dan konsepnya berhubungan dengan multi-programming.

#### Level 5

Level ini mengenalkan ide proses dalam mengeksekusi program. Kebutuhan-kebutuhan dasar pada sistem operasi untuk mendukung proses ganda termasuk kemampuan men-suspend dan me-resume proses. Hal ini membutuhkan register perangkat keras untuk menyimpan agar eksekusi bisa ditukar antara satu proses ke proses lainnya.

## Level 6

Mengatasi penyimpanan sekunder dari komputer. Level ini untuk menjadualkan operasi dan menanggapi permintaan proses dalam melengkapi suatu proses.

## Level 7

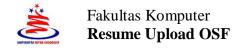
Membuat alamat logik untuk proses. Level ini mengatur alamat virtual ke dalam blok yang bisa dipindahkan antara memori utama dan memori tambahan. Cara-cara yang sering dipakai adalah menggunakan ukuran halaman yang tetap, menggunakan segmen sepanjang variabelnya, dan menggunakan cara keduanya. Ketika blok yang dibutuhkan tidak ada dimemori utama, alamat logis pada level ini meminta transfer dari level 6. Sampai point ini, sistem operasi mengatasi sumber daya dari prosesor tunggal. Mulai level 8, sistem operasi mengatasi obyek eksternal seperti peranti bagian luar, jaringan, dan sisipan komputer kepada jaringan.

#### Level 8

Mengatasi komunikasi informasi dan pesan-pesan antar proses. Dimana pada level 5 disediakan mekanisme penanda yang kuno yang memungkinkan untuk sinkronisasi proses, pada level ini mengatasi pembagian informasi yang lebih banyak. Salah satu peranti yang paling sesuai adalah pipe (pipa) yang menerima output suatu proses dan memberi input ke proses lain.

#### Level 9

Mendukung penyimpanan jangka panjang yang disebut dengan berkas. Pada level ini, data dari penyimpanan sekunder ditampilkan pada tingkat abstrak, panjang variabel yang terpisah. Hal nini bertentangan tampilan yang berorientasikan perangkat keras dari penyimpanan sekunder.



## Level 10

Menyediakan akses ke peranti eksternal menggunakan antarmuka standar.

#### Level 11

Bertanggung-jawab mempertahankan hubungan antara internal dan eksternal identifier dari sumber daya dan obyek sistem. Eksternal identifier adalah nama yang bisa dimanfaatkan oleh aplikasi atau pengguna. Internal identifier adalah alamat atau indikasi lain yang bisa digunakan oleh level yang lebih rendah untuk meletakkan dan mengontrol obyek. Leengontrol proses.

## Level 13

Menyediakan antarmuka dari sistem operasi dengan pengguna yang dianggap sebagai shell atau dinding karena memisahkan pengguna dengan sistem operasi dan menampilkan sistem operasi dengan sederhana sebagai kumpulan servis atau pelayanan.

Dari ketiga sumber diatas dapat kita simpulkan bahwa lapisan sistem operasi secara umum terdiri atas 4 bagian, yaitu:

# Perangkat keras

Lebih berhubungan kepada perancang sistem. Lapisan ini mencakup lapisan 0 dan 1 menurut Tanenbaum, dan level 1 sampai dengan level 4 menurut Stallings.

## 1. Sistem operasi

Lebih berhubungan kepada programer. Lapisan ini mencakup lapisan 2 menurut Tanenbaum, dan level 5 sampai dengan level 7 menurut Stallings.

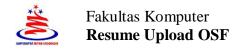
## 1. Kelengkapan

Lebih berhubungan kepada programer. Lapisan ini mencakup lapisan 3 menurut Tanenbaum, dan level 8 sampai dengan level 11 menurut Stallings.

## 1. Program aplikasi

Lebih berhubungan kepada pengguna aplikasi komputer. Lapisan ini mencakup lapisan 4 dan lapisan 5 menurut Tanebaum, dan level 12 dan level 13 menurut Stallings.

Lapisan n memberi layanan untuk lapisan n+1. Proses-proses di lapisan n dapat meminta layanan lapisan n-1 untuk membangunan layanan bagi lapisan n+1. Lapisan n dapat meminta layanan lapisan n-1. Kebalikan tidak dapat, lapisan n tidak dapat meminta layanan n+1. Masing-masing berjalan di ruang alamat-nya sendiri. Kelanjutan sistem berlapis adalah sistem berstruktur cincin seperti sistem MULTICS. Sistem MULTICS terdiri 64 lapisan cincin dimana satu lapisan berkewenangan berbeda. Lapisan n-1



mempunyai kewenangan lebih dibanding lapisan n. Untuk meminta layanan lapisan n-1, lapisan n melakukan trap. Kemudian, lapisan n-1 mengambil kendali sepenuhnya untuk melayani lapisan n.

# **Kelebihan Sistem Berlapis (layered system):**

- Memiliki rancangan modular, yaitu sistem dibagi menjadi beberapa modul & tiap modul dirancang secara independen.
- Pendekatan berlapis menyederhanakan rancangan, spesifikasi dan implementasi sistem operasi.

# **Kekurangan Sistem Berlapis (layered system):**

Fungsi-fungsi sistem operasi diberikan ke tiap lapisan secara hati-hati.
Contoh: Sistem operasi yang menggunakan pendekatan berlapis adalah
THE yang dibuat oleh Djikstra dan mahasiswa-mahasiswanya, serta sistem operasi MULTICS.

## 3. Kernel Mikro

Metode struktur ini adalah menghilangkan komponen-komponen yang tidak diperlukan dari kernel dan mengimplementasikannya sebagai sistem dan program-program level user. Hal ini akan menghasilkan kernel yang kecil. Fungsi utama dari jenis ini adalah menyediakan fasilitas komunikasi antara program client dan bermacam pelayanan yang berjalan pada ruang user.

# **Kelebihan Kernel Mikro:**

- kemudahan dalam memperluas sistem operasi
- mudah untuk diubah ke bentuk arsitektur baru
- kode vang kecil dan lebih aman

# **Kekurangan Kernel Mikro**:

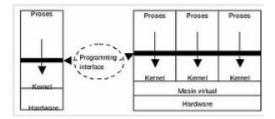
 kinerja akan berkurang selagi bertambahnya fungsi-fungsi yang digunakan.
 Contoh: sistem operasi yang menggunakan metode ini adalah TRU64 UNIX, MacOSX dan QNX.

# 4. Modular (Modules)

Kernel mempunyai kumpulan komponen-komponen inti dan secara dinamis terhubung pada penambahan layanan selama waktu boot atau waktu berjalan. Sehingga strateginya menggunakan pemanggilan modul secara dinamis (*Loadable Kernel Modules*). Umumnya sudah diimplementasikan oleh sistem operasi modern seperti Solaris, Linux dan MacOSX.

Sistem Operasi Apple Macintosh Mac OS X menggunakan **struktur** *hybrid*. Strukturnya menggunakan teknik berlapis dan satu lapisan diantaranya menggunakan Mach microkernel.

# 5. Mesin Maya ( Virtual Machine )



Mesin maya mempunyai sistem timesharing yang berfungsi untuk ,menyediakan kemampuan untuk multiprogramming dan perluasan mesin dengan antarmuka yang lebih mudah.

Struktur Mesin maya ( CP/CMS, VM/370 ) terdiri atas komponen dasar utama .

- Control Program, yaitu virtual machine monitor yang mengatur fungsi ari prosessor, memori dan piranti I/O. Komponen ini berhubungan langsung dengan perangkat keras.
- Conventional Monitor System, yaitu sistem operasi sederhanayang mengatur fungsi dari proses, pengelolaan informasi dan pengelolaan piranti.

# Kelebihan Mesin Maya (Virtual Machine):

- Konsep mesin virtual menyediakan proteksi yang lengkap untuk sumber daya system sehingga masing-masing mesin virtual dipisahkan mesin virtual yang lain. Isolasi ini tidak memperbolehkan pembagian sumber daya secara langsung.
- Sistem mesin virtual adalah mesin yang sempurna untuk riset dan pengembangan system operasi. Pengembangan system dikerjakan pada mesin virtual, termasuk di dalamnya mesin fisik dan tidak mengganggu operasi system yang normal.

# Kekurangan Mesin Maya ( Virtual Machine ):

 Konsep mesin virtual sangat sulit untuk mengimplementasikan kebutuhan dan duplikasi yang tepat pada mesin yang sebenarnya.

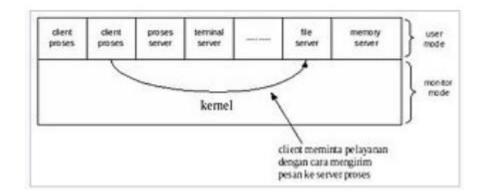
## Contoh:

- Sistem operasi MS-Windows NT dapat menjalankan aplikasi untuk MS-DOS, OS/2 mode teks dan aplikasi WIN16.
- IBM mengembangkan WABI untuk meng-emulasikan Win32 API sehingga sistem operasi yang menjalankan WABI dapat menjalankan aplikasiaplikasi untuk MS-Windows.

- Para pengembang Linux membuat DOSEMU untuk menjalankan aplikasi DOS pada sistem operasi Linux, WINE untuk menjalankan aplikasi-aplikasi MS-Windows.
- VMWare merupakan aplikasi komersial yang meng-abstraksikan perangkat keras intel 80x86 menjadi virtual mesin dan dapat menjalan beberapa sistem operasi lain (guest operating system) di dalam sistem operasi MS-Windos atau Linux (host operating system). VirtualBox merupakan salah satu aplikasi sejenis yang opensource.

# 6. Client-Server Model

Mengimplementasikan sebagian besar fungsi sistem operasi pada mode pengguna (user mode). Sistem operasi merupakan kumpulan proses dengan proses-proses dikategorikan sebagai server dan client, yaitu: Server, adalah proses yang menyediakan layanan.

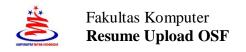


Client, adalah proses yang memerlukan/meminta layanan.

Proses client yang memerlukan layanan mengirim pesan ke server dan menanti pesan jawaban. Proses server setelah melakukan tugas yang diminta, mengirim hasil dalam bentuk pesan jawaban ke proses client. Server hanya menanggapi permintaan client dan tidak memulai dengan percakapan client. Kode dapat diangkat ke level tinggi, sehingga kernel dibuat sekecil mungkin dan semua tugas diangkat ke bagian proses pemaka. Kernel hanya mengatur komunikasi antara client dan server. Kernel yang ini popular dengan sebutan mikrokernel.

## **Kelebihan Client-Server Model:**

- Pengembangan dapat dilakukan secara modular.
- Kesalahan (bugs) di satu subsistem (diimplementasikan sebagai satu proses) tidak merusak subsistem-subsistem lain, sehingga tidak mengakibatkan satu sistem mati secara keseluruhan.
- Mudah diadaptasi untuk sistem tersebar.



# **Kekurangan Client-Server Model:**

- Layanan dilakukan lambat karena harus melalui pertukaran pesan.
- Pertukaran pesan dapat menjadi bottleneck.
- Tidak semua tugas dapat dijalankan di tingkat pemakai (sebagai proses pemakai).

# 7. Sistem Berorientasi Objek

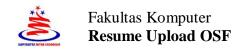
Sisten operasi merealisasikan layanan sebagai kumpulan proses disebut sistem operasi bermodel proses. Pendekatan lain implementasi layanan obiek-obiek. Sistem operasu adalah sebagai vand distrukturkan menggunakan objek disebut sistem operasi berorientasi objek. Pendekatan ini dimaksudkan untuk mengadopsi keunggulan teknologi berorientasi objek. Pada sistem yang berorientasi objek, layanan diimplementasikan sebagai kumpulan objek. Objek mengkapsulkan struktur data dan sekumpulan operasi pada struktur data itu. Tiap objek diberi tipe yang menandadi properti objek seperti proses, direktori, berkas, dan sebagainya. Dengan memanggil operasi yang didefinisikan di objek, data yang dikapsulkan dapat diakses dan dimodifikasi. Model ini sungguh terstruktur dan memisahkan antara layanan yang disediakan dan implementasinya. Sistem operasi MS Windows NT telah mengadopsi beberapa teknologi berorientasi objek tetapi belum keseluruhan.

# Kelebihan Sistem Berorientasi Objek:

 Terstruktur dan memisahkan antara layanan yang disediakan dan implementasinya.

# Kekurangan Sistem Berorientasi Objek:

 Sistem operasi MS Windows NT telah mengadopsi beberapa teknologi berorientasi objek tetapi belum keseluruhan.

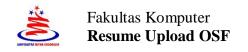


#### A. ID SECURITY

QWTD44112377-ASP-524414475

# **B. REFERENCE**

- [1] O. M. Febriani and A. S. Putra, "Sistem Informasi Monitoring Inventori Barang Pada Balai Riset Standardisasi Industri Bandar Lampung," *J. Inform.*, vol. 13, no. 1, pp. 90–98, 2014.
- [2] A. S. Putra, "Paperplain: Execution Fundamental Create Application With Borland Delphi 7.0 University Of Mitra Indonesia," 2018.
- [3] A. S. Putra, "2018 Artikel Struktur Data, Audit Dan Jaringan Komputer," 2018.
- [4] A. S. Putra, "ALIAS MANAGER USED IN DATABASE DESKTOP STUDI CASE DB DEMOS."
- [5] A. S. Putra, "COMPREHENSIVE SET OF PROFESSIONAL FOR DISTRIBUTE COMPUTING."
- [6] A. S. Putra, "DATA ORIENTED RECOGNITION IN BORLAND DELPHI 7.0."
- [7] A. S. Putra, "EMBARCADERO DELPHI XE 2 IN GPU-POWERED FIREMONKEY APPLICATION."
- [8] A. S. Putra, "HAK ATAS KEKAYAAN INTELEKTUAL DALAM DUNIA TEKNOLOGY BERBASIS REVOLUSI INDUSTRI 4.0."
- [9] A. S. Putra, "IMPLEMENTASI PERATURAN PERUNDANGAN UU. NO 31 TAHUN 2000 TENTANG DESAIN INDUSTRI BERBASIS INFORMATION TECHNOLOGY."
- [10] A. S. Putra, "IMPLEMENTATION OF PARADOX DBASE."
- [11] A. S. Putra, "IMPLEMENTATION OF TRADE SECRET CASE STUDY SAMSUNG MOBILE PHONE."
- [12] A. S. Putra, "IMPLEMENTATION PATENT FOR APPLICATION WEB BASED CASE STUDI WWW. PUBLIKLAMPUNG. COM."
- [13] A. S. Putra, "IMPLEMENTATION SYSTEM FIRST TO INVENT IN DIGITALLY INDUSTRY."
- [14] A. S. Putra, "MANUAL REPORT & INTEGRATED DEVELOPMENT ENVIRONMENT BORLAND DELPHI 7.0."
- [15] A. S. Putra, "PATENT AS RELEVAN SUPPORT RESEARCH."
- [16] A. S. Putra, "PATENT FOR RESEARCH STUDY CASE OF APPLE. Inc."
- [17] A. S. Putra, "PATENT PROTECTION FOR APPLICATION INVENT."
- [18] A. S. Putra, "QUICK REPORT IN PROPERTY PROGRAMMING."
- [19] A. S. Putra, "REVIEW CIRCUIT LAYOUT COMPONENT REQUIREMENT ON ASUS NOTEBOOK."
- [20] A. S. Putra, "REVIEW TRADEMARK PATENT FOR INDUSTRIAL TECHNOLOGY BASED 4.0."
- [21] A. S. Putra, "TOOLBAR COMPONENT PALLETTE IN OBJECT



## ORIENTED PROGRAMMING."

- [22] A. S. Putra, "WORKING DIRECTORY SET FOR PARADOX 7."
- [23] A. S. Putra, "ZQUERY CONNECTION IMPLEMENTED PROGRAMMING STUDI CASE PT. BANK BCA Tbk."
  - [24] A. S. Putra, D. R. Aryanti, and I. Hartati, "Metode SAW (Simple Additive Weighting) sebagai Sistem Pendukung Keputusan Guru Berprestasi (Studi Kasus: SMK Global Surya)," in *Prosiding Seminar Nasional Darmajaya*, 2018, vol. 1, no. 1, pp. 85–97.
- [25] A. S. Putra and O. M. Febriani, "Knowledge Management Online Application in PDAM Lampung Province," in *Prosiding International conference on Information Technology and Business (ICITB)*, 2018, pp. 181–187.
- [26] A. S. Putra, O. M. Febriani, and B. Bachry, "Implementasi Genetic Fuzzy System Untuk Mengidentifikasi Hasil Curian Kendaraan Bermotor Di Polda Lampung," *SIMADA (Jurnal Sist. Inf. dan Manaj. Basis Data)*, vol. 1, no. 1, pp. 21–30, 2018.
- [27] A. S. Putra, H. Sukri, and K. Zuhri, "Sistem Monitoring Realtime Jaringan Irigasi Desa (JIDES) Dengan Konsep Jaringan Sensor Nirkabel," *IJEIS* (*Indonesian J. Electron. Instrum. Syst.*, vol. 8, no. 2, pp. 221–232.
- [28] D. P. Sari, O. M. Febriani, and A. S. Putra, "Perancangan Sistem Informasi SDM Berprestasi pada SD Global Surya," in *Prosiding Seminar Nasional Darmajaya*, 2018, vol. 1, no. 1, pp. 289–294.