

## PENERAPAN ALGORITMA PERCEPTRON UNTUK APLIKASI DIAGNOSA DEMAM BERDARAH

Lukman Hakim<sup>1</sup>, Henny Liani Senjaya<sup>2</sup>

Program Studi Teknik Informatika, Teknologi dan Desain, Universitas Bundamulia<sup>1,2</sup>

Jl. Lodan Raya No.2 Ancol, Jakarta Utara 14430

<sup>1</sup>[lhakim@bundamulia.ac.id](mailto:lhakim@bundamulia.ac.id), <sup>2</sup>[hennysenjaya@gmail.com](mailto:hennysenjaya@gmail.com)

### Abstrak

Penyakit demam berdarah merupakan virus yang disebabkan oleh nyamuk Aedes betina yang mengisap darah dan menularkan virus dari air ludah nyamuk tersebut yang menyebabkan nyeri ulu hati dan pendarahan saluran pencernaan. Tujuan penelitian menerapkan algoritma perceptron pada aplikasi diagnosa demam berdarah berbasis situs. Perancangan aplikasi diagnosa demam berdarah dengan menggunakan algoritma perceptron yang membantu dalam proses perhitungan nilai input dan bobot apabila nilai 0 tidak terjangkau penyakit dan 1 menunjukkan terjangkau demam berdarah, penentuan gejala diambil berdasarkan dari dokter spesialis dan dokter umum yang menangani pasien DBD. Hasil implementasi perhitungan dan bobot pada aplikasi diagnosa demam berdarah dengan algoritma perceptron tingkat keakuratan sistem diagnosa adalah 88% dari perbandingan pendapat dokter dan perhitungan algoritma perceptron.

Kata kunci : Algoritma Perceptron, Demam Berdarah, Nyamuk Aedes Aegypti.

### 1. Pendahuluan

Pada tahun 2014, sampai pertengahan bulan Desember tercatat penderita DBD di 34 provinsi di Indonesia sebanyak 71.668 orang, dan 641 diantaranya meninggal dunia. Angka tersebut lebih rendah dibandingkan tahun sebelumnya, yakni tahun 2013 dengan jumlah penderita sebanyak 112.511 orang dan jumlah kasus meninggal sebanyak 871 penderita. Demikian disampaikan Kepala Badan Penelitian dan Pengembangan Kesehatan (Balitbangkes) Kemenkes RI, Prof. dr. Tjandra yoga Aditama, Sp.P(K), MARS, DTM&H, DTCE, melalui surat elektronik yang diterima Pusat Komunikasi Publik Kemenkes RI.

Meskipun secara umum terjadi penurunan kasus tahun ini dibandingkan tahun sebelumnya namun pada beberapa provinsi mengalami peningkatan jumlah kasus DBD, diantaranya Sumatra Utara, Riau, Kepri, DKI Jakarta, Kalimantan Barat, Sulawesi Utara, Bali dan Kalimantan Utara. Tercatat ada lebih kurang 7 kabupaten/kota yang melaporkan terjadinya kejadian luar biasa (KLB) DBD pada tahun 2014 ini yaitu Kabupaten Morowali (Sulteng),

Kabupaten Sintang (Kalbar), Kabupaten Belitung Timur (Babel), Kabupaten Bangka Barat (Babel), Kabupaten Ketapang (Kalbar), Kabupaten Karimun (Riau) dan Kota Dumai (Riau).

Dari hal diatas dapat diketahui masih banyaknya penderita deman berdarah cukup tinggi, dari beberapa faktor yang mempengaruhi munculnya DBD antara lain rendahnya status kekebalan kelompok masyarakat dan kepadatan populasi nyamuk penular karena banyaknya tempat perindukan nyamuk yang biasanya terjadi pada musim penghujan. Kesadaran masyarakat yang belum mengetahui atau memahami tentang pentingnya memahami gejala deman berdarah.

Tujuan Penelitian adalah membuat aplikasi diagnosa demam berdarah dengan menggunakan algoritma perceptron. Menghasilkan aplikasi berbasis situs dengan jaringan syaraf tiruan berdasarkan pengetahuan dari para dokter spesialis dan dokter umum.

Manfaat adalah membantu masyarakat mendeteksi penyakit deman berdarah secara lebih dini dan dapat dilakukan dimana saja.

### 2. Landasan Teori

**Demam berdarah** dengue (DBD)

merupakan penyakit yang disebabkan oleh virus Dengue yang ditularkan dari orang ke orang melalui gigitan nyamuk Aedes (Ae). Ae aegypti merupakan vektor yang paling utama, namun spesies lain seperti

Ae.albopictus juga dapat menjadi vektor penular. Nyamuk penular dengue ini terdapat hampir di seluruh pelosok Indonesia, kecuali di tempat yang memiliki ketinggian lebih dari 1000 meter di atas permukaan laut[11].

### Sistem pakar

merupakan salah satu bagian dari *Artificial Intelligence*. Yang terdiri dari sistem pakar, pengolahan bahasa alami, pengenalan suara, robotika dan sistem sensor, *computer vision*, *intelligent computer-aided instruction* dan *game playing*. Sistem pakar adalah sistem perangkat lunak computer yang menggunakan ilmu, fakta, dan teknik berpikir dalam pengambilan keputusan untuk menyelesaikan masalah-masalah yang biasanya hanya dapat diselesaikan oleh para ahli dalam bidang tertentu. Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarik kesimpulan (*interface rules*) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam komputer yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu [11].

### Mesin Inferensi

adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan (Sutojo dkk, 2011).

### Jaringan Saraf Tiruan

Menurut Hermawan (2006) jaringan saraf tiruan didefinisikan sebagai suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan syaraf manusia. Jaringan saraf tiruan tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan atas asumsi sebagai berikut [3]:

1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut *neuron*.
2. Isyarat mengalir diantara sel saraf atau neuron melalui suatu sambungan penghubung.
3. Setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini akan digunakan untuk menggandakan atau mengalikan isyarat yang dikirim melaluinya.
4. Setiap sel saraf akan menerapkan fungsi aktivasi terhadap isyarat hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan isyarat keluarannya.

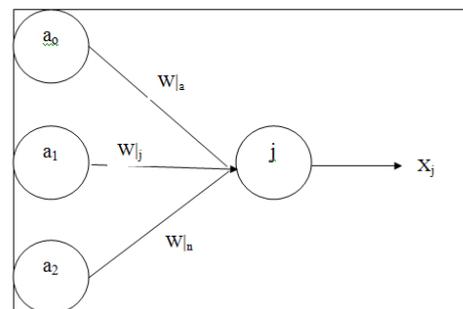
### Perceptron

Menurut Hermawan (2006) salah satu organisasi yang dikenal dan sering digunakan dalam paradigma jaringan saraf tiruan buatan adalah Perambatan Galat Mundur. Sebelum dikenal adanya jaringan saraf Perambatan Galat Mundur pada tahun 1950-1960an, dikenal dua paradigma penting yang nantinya

menjadi dasar dari jaringan saraf Perambatan Galat Mundur, yakni *Perceptron* [3].

Aturan belajar (*learning*) *perceptron* lebih handal jika dibandingkan dengan aturan *Hebb* (*Hebb rule*). Dengan asumsi yang sesuai, prosedur belajar iteratifnya dapat dibuktikan dengan konvergenya ke bobot yang benar, yakni bobot yang memungkinkan jaringan menghasilkan nilai output yang benar untuk setiap pola input pelatihan.

Arsitektur *Perceptron* belajar mengenali pola dengan metode belajar terbimbing. Pola yang diklasifikasikan biasanya berupa bilangan biner (kombinasi 1 dan 0) dan kategori pengklasifikasiannya juga diwujudkan dalam bilangan biner. *Perceptron* dibatasi untuk dua lapisan pengolah dengan satu lapisan bobot (diantaranya) yang dapat beradaptasi [6][15].



Gambar 1. unit pengolahan Perceptron

Elemen pada Gambar 2.2 adalah unit pengolah dasar dari *perceptron*. Unit pengolahan ini mendapat masukan dari unit pengolahan lain yang masing-masing dihubungkan melalui suatu bobot interkoneksi  $W_{ji}$ . Unit pengolah melakukan penjumlahan berbobot untuk seluruh masukannya (1).

$$S_j = \sum_{i=0}^n u_i w_{ji} \dots \quad (1)$$

Dengan:

$W_{ji}$  = bobot sambungan dari unit i ke unit j

$u_i$  = masukan yang berasal dari unit i

Sebuah nilai prasikap diberikan sebagai tambahan masukan kepada unit pengolah. Nilai prasikap masukan ini bernilai tetap yaitu +1, dan dihubungkan dengan unit pengolah j melalui pembobotan  $w_{j0}$  yang nilainya selalu beradaptasi selama jaringan mengalami pelatihan.

*Perceptron* menguji apakah hasil penjumlahan berbobot berada diatas atau dibawah nilai ambang yang telah ditentukan, dengan aturan:

$$\begin{aligned} \text{Jika } S_j > 0 \text{ maka } x_j &= 1 \\ \text{Jika } S_j < 0 \text{ maka } x_j &= 0 \end{aligned} \quad (2)$$

Dengan:

$x_j$  = nilai keluaran unit j

*Perceptron* dilatih dengan menggunakan sekumpulan pola yang diberikan kepadanya secara berulang-ulang selama latihan. Setiap pola yang diberikan merupakan pasangan pola masukan dan pola yang diinginkan. *Perceptron* melakukan penjumlahan berbobot terhadap tiap-tiap masukannya dan menggunakan fungsi ambang untuk menghitung keluarannya (Persamaan 1 dan Persamaan 2). Keluaran ini kemudian dibandingkan dengan hasil yang diinginkan, perbedaan yang dihasilkan dari perbandingan ini digunakan untuk merubah bobot-bobot yang ada dalam jaringan. Demikian dilakukan berulang-ulang sehingga dihasilkan keluaran yang sesuai dengan hasil yang diinginkan. Dengan menggunakan aturan *perceptron* sederhana, maka perubahan bobot dapat dirumuskan:

$$W_{j\text{baru}} = W_{j\text{lama}} + C (t_{jp} - x_{jp}) a_i \quad (3)$$

Dengan:

C = kecepatan belajar

$t_{jp}$  = nilai keluaran yang diinginkan unit j setelah diberikan pola p lapisan masukan.

$x_{jp}$  = nilai keluaran yang dihasilkan unit j setelah diberikan pola p lapisan masukan.

$a_i$  = masukan yang berasal dari unit i

Kesesuaian antara keluaran dengan sasaran yang diinginkan menunjukkan pencapaian kerja jaringan yang sedang dilatihkan dan dinyatakan dengan besarnya galat yang diperbolehkan agar keluaran hasil pengolahan jaringan dapat dianggap sama dengan pola yang diinginkan [6].

### 3. Hasil dan Pembahasan

Aplikasi ini dilengkapi registrasi untuk memudahkan data pengguna dan adanya login untuk masuk halaman diagnosa penyakit demam berdarah. Dapat dilihat pada gambar 2 halaman login dibawah berikut ini:



Gambar 2. halaman login

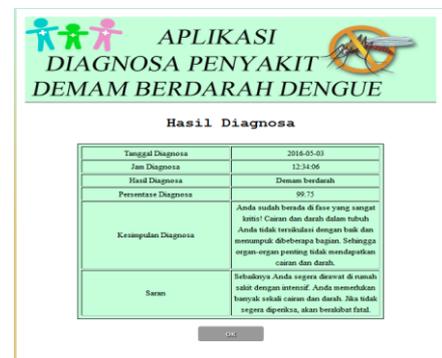
Halaman ini merupakan halaman diagnosa gejala yang sudah disesuaikan berdasarkan wawancara dengan para dokter mengenai gejala

pada penyakit demam berdarah. Gambar 3 dapat dilihat pada halaman berikut ini :



Gambar 3. halaman masukan gejala demam berdarah

Halaman ini merupakan halaman hasil diagnosa yang sudah diproses berdasarkan perhitungan nilai dan bobot dari pembelajaran algoritma perceptron. Dapat dilihat pada Gambar 4 halaman berikut ini :



Gambar 4. Halaman hasil diagnosa demam berdarah

### Pembahasan

#### Analisis Kebutuhan

Dari hasil wawancara dengan 5 dokter umum yang sudah bekerja selama ±5 tahun di rumah sakit, maka dari 15 gejala tersebut dijadikan sebagai pertanyaan untuk pengguna pada tabel 1 berikut ini:

Tabel 1. Pertanyaan Gejala demam berdarah

No.	Pertanyaan	Pilihan Jawaban	Nilai Jawaban
1.	Apakah Anda mengalami demam tubuh?	Tidak	0
		Ya	1
2.	Apakah suhu demam tubuh Anda naik turun atau tidak stabil?	Tidak	0
		Ya	1
3.	Berapa suhu tertinggi demam pada tubuh Anda?	< 33 derajat	0
		34-35 derajat	0.25
		36-37 derajat	0.5
		38-39 derajat	0.75
		>39 derajat	1
4.	Berapa lama Anda mengalami demam tubuh?	1 hari	0
		2-3 hari	0.25
		4-5 hari	0.5
		6-7 hari	0.75
		>7 hari	1
5.	Apakah Anda merasakan sakit pada kepala?	Tidak sakit	0
		Sedikit sakit	0.3
		Sakit	0.6
		Sangat sakit	1
6.	Apakah muncul bitnik merah di tubuh Anda?	Tidak ada	0
		Sedikit ada di	0.3

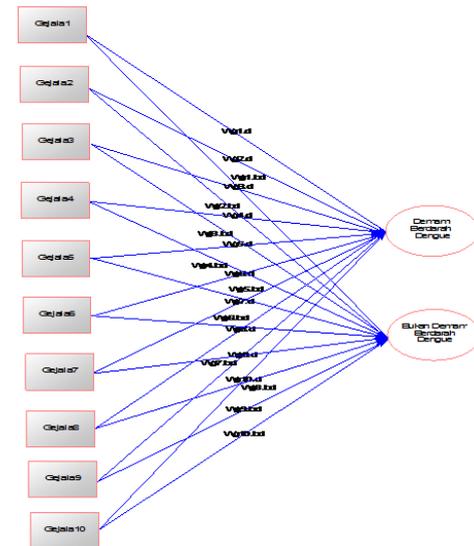
Pada tabel 2 merupakan hasil diagnosa yang diberikan oleh dokter, dan ini digunakan sebagai keluaran dari aplikasi diagnosa demam berdarah.

Tabel 2. hasil diagnosa demam berdarah

Persentase	Kesimpulan	Saran
0 %	Anda tidak mengalami demam berdarah. Tetap jaga kesehatan Anda.	Lakukan pencegahan demam berdarah dengan mengurangi populasi nyamuk di lingkungan Anda. Pelihara lingkungan yang bersih dan sehat. Dan lakukan 3M secara berkala.
0.001-19.9 %	Anda masih dalam fase sangat awal terinfeksi virus demam berdarah. Pada fase ini, Anda akan lebih cenderung merasakan tubuh lemas dan kehilangan tenaga.	Perbanyak asupan air putih. Tubuh Anda akan cepat mengalami dehidrasi. Konsumsi juga jus jambu biji.
20-39.9 %	Anda berada dalam fase demam. Tubuh Anda akan mengalami demam yang tinggi pada waktu tertentu, dan akan turun normal suhunya pada waktu tertentu. Jangan sampai terkecoh dengan demam tubuh yang seolah-olah sembuh.	Minumlah obat penurun demam yang mengandung paracetamol secara rutin. Bila 2 hari tidak ada perubahan, segera konsultasikan kepada dokter.
40-59.9 %	Anda berada dalam fase	Anda hanya dapat mengurangi

Dari hasil analisis data (fakta) yang sudah didapat maka dapat dirancang arsitektur

jaringan syaraf tiruan dengan algoritma perceptron pada gambar 5 berikut dibawah ini:



Gambar 5. Arsitektur Algoritma perceptron

Setelah itu, nilai pada lapisan gejala akan diperhitungkan dalam algoritma *perceptron*. Jika hasil perhitungan (lapisan output) adalah 0, maka dapat disimpulkan bahwa *user* tidak terjangkit demam berdarah. Tetapi sebaliknya, jika lapisan output adalah 1, maka *user* disimpulkan menderita demam berdarah.

Pencarian bobot dari masing-masing gejala adalah dengan melakukan perhitungan dari 100 data training yang ada. Langkah kerjanya adalah sebagai berikut:

1. Pencarian hasil hitung( $S_j$ ) dan learn dari bobot mula-mula( $W_{ji}$ ) yaitu 0 dengan rumus:

$$S_j = \sum_{i=0}^n u_i w_{ji}$$

Dan  $U_i$  adalah nilai dari masing-masing gejala yang ada dalam 100 data training.

2. Setelah di dapatkan nilai hitung( $S_j$ ) dan learn, jika *learn* tidak cocok dengan output\_dbd yang diinpukan pada lampiran 100 data *training*, maka akan dihitung bobot baru untuk masing-masing gejala dengan rumus:

$$W_{j\text{baru}} = W_{j\text{lama}} + C (t_{jp} - x_{jp}) a_j$$

Apabila bobot baru sudah ditemukan, maka akan dihitung kembali nilai  $S_j$  pada langkah pertama. Jika output sudah sama dengan *learn* dan cocok, maka akan berlanjut ke data *training* berikutnya sampai semua data *training* habis dihitung. Bobot baru yang didapat dari hasil algoritma tersebut disajikan dalam tabel 3. Berikut ini :

Tabel 3. Bobot untuk gejala yang sudah di hitung perceptron

No.	Nama gejala	Bobot yang didapat
1	Demam tubuh	0
2	Tinggi demam	0.00375
3	Suhu demam	0.004
4	Lama demam	0.0025
5	Sakit kepala	-0.002
6	Bintik merah	0.0002
7	Nyeri tulang	-0.0008
8	Nyeri sendi	0.0007
9	Mimisan	0.0005
10	Lemas	-0.0015
11	Mual perut	0.0016
12	Nyeri mata	0.0007
13	Nyeri perut	-0.0007
14	Haus	0.001
15	Warna bab	0.0022

Proses pencarian bobot baru seperti Gambar 6 coding pencarian bobot sampai seluruh data memiliki *output* dan *learn* yang cocok dan kemudian mendapatkan bobot baru untuk masing-masing gejala yang akan digunakan untuk proses perhitungan pada diagnosa penyakit demam berdarah. Bobot baru tersebut disimpan di dalam *database* gejala pada *field* bobot\_gejala.

```

if($baris['output_dbd']<$learn)
{
    $wdemam= $wdemam+ $c *($baris['output_dbd']-$learn) * $baris['na_demam'];
    $wsuhu= $wsuhu+ $c *($baris['output_dbd']-$learn) * $baris['na_suhu'];
    $wlama= $wlama+ $c *($baris['output_dbd']-$learn) * $baris['na_lama'];
    $wkepala= $wkepala+ $c *($baris['output_dbd']-$learn) * $baris['na_kepala'];
    $wbintik= $wbintik+ $c *($baris['output_dbd']-$learn) * $baris['na_bintik'];
    $wtulang= $wtulang+ $c *($baris['output_dbd']-$learn) * $baris['na_tulang'];
    $wsendi= $wsendi+ $c *($baris['output_dbd']-$learn) * $baris['na_sendi'];
    $wmimisan= $wmimisan+ $c *($baris['output_dbd']-$learn) * $baris['na_mimisan'];
    $wlemas= $wlemas+ $c *($baris['output_dbd']-$learn) * $baris['na_lemas'];
    $wmual= $wmual+ $c *($baris['output_dbd']-$learn) * $baris['na_mual'];
    $wtinggi= $wtinggi+ $c *($baris['output_dbd']-$learn) * $baris['na_tinggi'];
    $wmata= $wmata+ $c *($baris['output_dbd']-$learn) * $baris['na_mata'];
    $wperut= $wperut+ $c *($baris['output_dbd']-$learn) * $baris['na_perut'];
    $whaus= $whaus+ $c *($baris['output_dbd']-$learn) * $baris['na_haus'];
    $wbab= $wbab+ $c *($baris['output_dbd']-$learn) * $baris['na_bab'];
}
else{
    $cocok=true;
    echo "cocok = $cocok ";
}
    
```

Gambar 6. coding pencarian bobot

Perhitungan hasil diagnosa, pertama-tama program akan memanggil *database* gejala dimana bobot gejala dan kode gejala diambil dari *database* untuk dicocokkan dengan masukkan jawaban dari *user*. Kemudian nilai masukkan dari *user* akan dikalikan dengan bobot masing-masing gejala yang ada sesuai dengan kode gejalanya agar tidak tertukar. Gambar 7 merupakan coding implementasi algoritma perceptron dan data pembelajaran (gejala).

```

}
else if( $baris["kd_gejala"] == "g12")
{
    $hasil_perhitungan = $hasil_perhitungan + ($bobot * $mata);
}
else if( $baris["kd_gejala"] == "g13")
{
    $hasil_perhitungan = $hasil_perhitungan + ($bobot * $perut);
}
else if( $baris["kd_gejala"] == "g14")
{
    $hasil_perhitungan = $hasil_perhitungan + ($bobot * $haus);
}
else if( $baris["kd_gejala"] == "g15")
{
    $hasil_perhitungan = $hasil_perhitungan + ($bobot * $bab);
}
}

$hasil_perhitungan;

if($hasil_perhitungan > 0)
{
    $hasil_diagnosa = "Demam berdarah";
}
else
{
    $hasil_diagnosa = "Tidak Demam berdarah";
}
    
```

Gambar 7. Implementasi algoritma perceptron

Hasil pengujian diagnosa sistem yang dibuat dibandingkan dengan kecocokan diagnosa dokter, maka didapat kecocokan sebanyak 88%, data 100 data yang diuji. Ditunjukkan pada tabel 4 kecocokan diagnosa pengujian algoritma perceptron.

Tabel 4. Pengujian kecocokan diagnosa

no.	diagnosa pakar	diagnosa sistem	kesimpulan
1	demam berdarah	demam berdarah	cocok
2	demam berdarah	demam berdarah	cocok
3	tidak demam berdarah	demam berdarah	tidak cocok
4	demam berdarah	demam berdarah	cocok
5	tidak demam berdarah	demam berdarah	tidak cocok

#### 4. Simpulan

1. Hasil pengujian aplikasi diagnosa penyakit deman berdarah dengan metode jaringan syaraf tiruan menggunakan algoritma perceptron tingkat keakuratan atau kecocokan antara hasil diagnosa dokter dan aplikasi menunjukkan 88% akurat.
2. Aplikasi ini dapat memberikan kemudahan kepada masyarakat untuk lebih dini mengetahui gejala deman berdarah, sehingga mengurangi angka kematian pada penderita.

#### 5. Daftar Pustaka

[1] Almira Syawli, dkk. 2012. "Diagnosa Penyakit Diabetes Mellitus dengan Metode NaiveBayes Berbasis Desktop Application". Jurnal. Program Teknologi Informatika dan Komputer. Universitas Brawijaya. Malang

- [2] Hasan, I. 2009. *Pokok-Pokok Materi Statistik 1 (Statistik Deskriptif)*. Jakarta: PT. Bumi Aksara
- [3] Hermawan Arief. 2006. *Jaringan Saraf Tiruan*, Yogyakarta: Andi
- [4] Indriani, Gita. 2013. "Populasi Sampel dan Teknik Sampling". Laporan Tugas Akhir. Semarang: Universitas Negeri Semarang
- [5] Kroenke, David M., David J. Auer. 2010. *Database Processing*. USA: Pearson Education, Inc.
- [6] Lukman Hakim, Julius Giovanni, 2015. Pengolahan Aset Berbasis Mobile dengan Algoritma Perceptron. Proceeding Sentika Universitas Atmajaya Yogyakarta. Yogyakarta. Program Studi Teknik Informatika.
- [7] L., Jeffrey, Witthen, Lonnie D. Bentley, Kevin C. Dittman. 2014. *Metode Desain dan Analisis Sistem*. Yogyakarta: ANDI
- [8] Larsen, Rob. 2013. *Beginning HTML & CSS*. USA: John Willey & Sons, Inc.
- [9] Lungan, R. 2006. *Aplikasi Statistika dan Hitung Peluang*. Yogyakarta: Penerbit Graha Ilmu
- [10] Maria Kresentia Yukiko Wada. 2015. "Sistem Pakar Diagnosa Gejala Penyakit Dalam dengan Metode Forward Chaining". Skripsi. Fakultas Teknologi dan Desain. Universitas Bunda Mulia. Jakarta
- [11] Nur Anjas Sari. 2013. "Sistem Pakar Mendiagnosa Penyakit Demam Berdarah Menggunakan Metode *Certainty Factor*". Jurnal Pelita Informatika Budi Darma. Vol. 04. No. 03
- [12] Setiawanti, Nur. 2011. "Pengembangan Sistem Informasi Penjualan Perumahan Pesona Intan". Laporan Tugas Akhir. Garut: Sekolah Tinggi Teknologi Garut
- [13] Sisik, Bertha. 2012. *Pemrograman Web dengan PHP*. Bandung: Informatika Bandung
- [14] Sinartha, Wulan. 2012. "Pengertian Sistem Menurut Para Ahli". <https://www.scribd.com/doc/90824156/Pengertian-Sistem-Menurut-Para-Ahli>. Diakses: 22 September 2015
- [15] Sonia Grania. 2014. "Penerapan Aturan *Perceptron* pada Jaringan Saraf Tiruan Dalam Pembagian Jurusan". Skripsi. Fakultas Teknologi dan Desain. Universitas Bunda Mulia. Jakarta
- [16] Subsada Ngoen, Thompson. 2009. *Algoritma dan Struktur Data*. Jakarta: Mitra Wacana Media
- [17] Sutojo, T., Edy Mulyanto. 2011. *Kecerdasan Buatan*. Yogyakarta: ANDI
- [18] Yanni Immarul Za'iim. 2012. "Sistem Pakar untuk Mendiagnosa Penyakit Jantung Koroner Menggunakan Metode *Perceptron*". Jurnal. Program Studi Teknik Informatika. Universitas Dian Nuswantoro. Semarang