

BAB II

LANDASAN TEORI

Landasan teori adalah kerangka fikir atau teori dasar dalam sebuah penelitian. Landasan teori dalam penyusunan skripsi digunakan sebagai dasar yang kuat dalam sebuah penelitian yang dapat dipertanggungjawabkan.

2.1 Prestasi

(Baiti, 2010) Prestasi adalah hasil yang telah dicapai, dilakukan dan dikerjakan oleh seseorang. Prestasi setiap orang tidak selalu sama dalam berbagai bidang. Misalnya prestasi dalam bidang kesenian, olahraga, sastra, kepemimpinan, ilmu pengetahuan, teknologi, dan sebagainya.

2.2 Mahasiswa

(Hartaji, 2012) Mahasiswa adalah seseorang yang sedang dalam proses menimba ilmu ataupun belajar dan terdaftar sedang menjalani pendidikan pada salah satu bentuk perguruan tinggi yang terdiri dari akademik, politeknik, sekolah tinggi, institut dan universitas. Dalam Kamus Bahasa Indonesia (KBI), mahasiswa didefinisikan sebagai orang yang belajar di Perguruan Tinggi (Kamus Bahasa Indonesia Online, kbbi.web.id).

2.3 Alumni

(Almanfaluthi, 2009) Alumni adalah orang-orang yang telah mengikuti atau tamat dari suatu sekolah atau perguruan tinggi. Alumni merupakan bagian yang tidak dapat dipisahkan dalam sebuah siklus pendidikan.

2.4 Pengertian Sistem Informasi

(Abdul Kadir, 2014) Dalam praktik, istilah sistem informasi lebih sering dipakai tanpa embel-embel berbasis komputer walaupun dalam kenyataannya komputer merupakan bagian yang penting.

2.5 Pengertian Basis Data

(Abdul Kadir, 2014) Basis Data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi yang dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas.

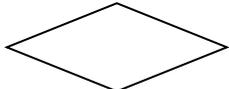
2.6 Entity Relationship Diagram(ERD)

(A.S, Rosa, M.Shalahudin, 2015) ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. Simbol *Entity Relationship Diagram* ditunjukkan pada Tabel 2.1

Tabel 2.1 Simbol *Entity Relationship Diagram*

Simbol	Keterangan
<p style="text-align: center;"><i>Entity</i></p> 	<p><i>Entitas/Entity</i> merupakan data inti yang akan disimpan, bakal tabel pada basis data. Penamaan entitas biasanya lebih ke data benda dan belum merupakan nama tabel.</p>
<p style="text-align: center;">Atribut</p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.</p>

Tabel 2.1 Simbol *Entity Relationship Diagram* (lanjutan)

<p style="text-align: center;">Relasi</p> 	<p>Relasi yang mengkhubungkan antar entitas.</p>
<p>Asosiasi/ <i>Association</i></p> 	<p>Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian.</p>

ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *tenary* (satu relasi menghubungkan tiga buah relasi atau *N-ary* (satu relasi menghubungkan banyak entitas).

2.7 Data Flow Diagram (DFD)

(A.S, Rosa, M.Shalahudin, 2015) *Data Flow Diagram (DFD)* atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*Output*). DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan program-program terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur. Simbol *Data Flow Diagram* ditunjukkan pada Tabel 2.2

Tabel 2.2 Simbol *Data Flow Diagram*

Simbol	Keterangan
<p><i>External Entity</i></p> 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem.
<p>Arus data (<i>data flow</i>)</p> 	Menggambarkan aliran data .
<p>Proses (<i>process</i>)</p> 	Proses atau fungsi yang mentransformasikan data masukan menjadi keluaran.
<p>Simpanan data (<i>data store</i>)</p> 	Komponen yang berfungsi untuk menyimpan data atau <i>file</i>

Berikut ini tahapan-tahapan perancangan dengan menggunakan DFD:

- a. Membuat DFD Level 0 atau *Context Diagram*.

Menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain.

- b. Membuat DFD Level 1.

Menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan.

- c. Membuat DFD Level 2.

Menggambarkan modul-modul yang di *breakdown* lebih detail dari DFD level 1.

d. Membuat DFD Level 3 dan seterusnya.

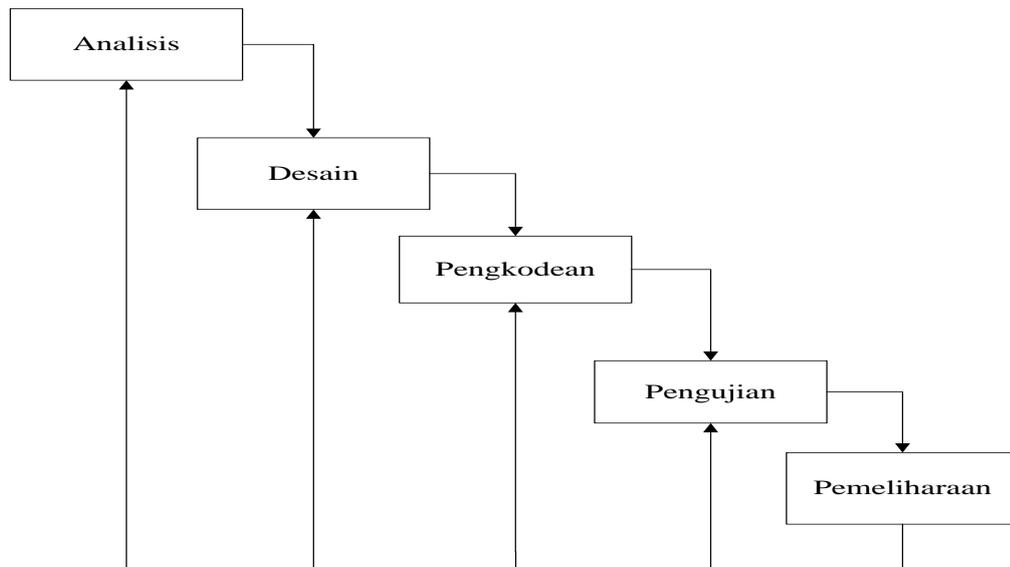
Merupakan *breakdown* dari modul pada DFD level di atasnya.

2.8 Metode Pengembangan Sistem

Menurut A.S, Rosa, M.Shalahudin, 2015 proses pengembangan mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan dalam mengembangkan sistem sebelumnya. Adapun tahapan-tahapan yang terdapat dalam metode *waterfall* yaitu:

- a. Analisis (*Analysis*) Kebutuhan Perangkat Lunak, adalah proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan yang dibutuhkan oleh *user* dan perangkat lunak yang digunakan untuk pemenuhan kebutuhan tersebut.
- b. Perancangan (*Design*), tahapan ini lebih menekankan pada tahap desain sistem secara menyeluruh, desain sistem dilakukan untuk menindak lanjuti tahap sebelumnya dan sebagai acuan pembuatan program.
- c. Pembuatan kode program, desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.
- d. Pengujian, pada tahap pengujian ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.
- e. Pendukung (*support*) atau pemeliharaan (*maintenance*), adalah proses pengembangan mulai dari analisis spesifikasi perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru hanya melakukan perawatan kepada perangkat lunak yang ada.

Metode Pengembangan Sistem *Waterfall* ditunjukkan pada Gambar 2.1



Gambar 2.1 Metode Pengembangan Sistem *Waterfall*

2.9 Kamus Data

(A.S, Rosa, M.Shalahudin, 2015) Kamus Data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum. Kamus data biasanya berisi:

- Nama, nama dari data.
- Digunakan pada, merupakan proses-proses yang terkait data.
- Deskripsi, merupakan deskripsi data.
- Informasi tambahan, seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.
- kamus data memiliki beberapa symbol untuk menjelaskan informasi tambahan.

Simbol – simbol Kamus Data ditunjukkan pada Tabel 2.3

Table 2.3 Simbol – simbol Kamus Data

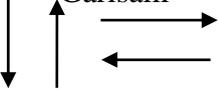
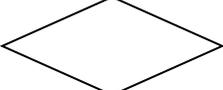
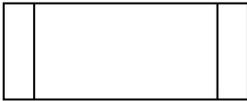
Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau.....
{ }n	N kali diulang atau bernilai banyak
()	Data optional
....	Batas komentar

Kamus data pada DFD nanti harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai, sehingga harus ada yang diperbaiki baik perancangan basis datanya, perancangan DFD nya atau keduanya.

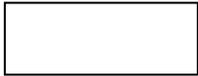
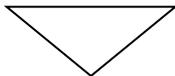
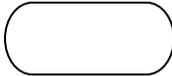
2.10 Flowchart

(Cybernur, 2016) *Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya. Simbol *Flowchart* ditunjukkan pada Tabel 2.4

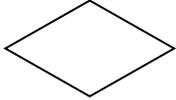
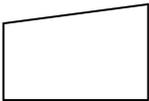
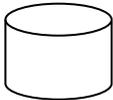
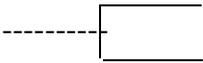
Tabel 2.4 Simbol *Flowchart*

Simbol	Keterangan
<p data-bbox="373 539 564 573"><i>Input / Output</i></p> 	<p data-bbox="651 533 1353 622">Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i>.</p>
<p data-bbox="421 680 517 714">Proses</p> 	<p data-bbox="651 680 1353 770">Simbol proses (<i>processing symbol</i>) atau simbol pengolah yang digunakan suatu proses.</p>
<p data-bbox="405 831 533 864">Garisalir</p> 	<p data-bbox="651 831 1353 920">Simbol garis alir (<i>flow lines symbol</i>), digunakan untuk menunjukkan arus dari proses.</p>
<p data-bbox="384 987 553 1021">Penghubung</p> 	<p data-bbox="651 987 1353 1122">Simbol penghubung (<i>connector symbol</i>), digunakan untuk menunjukkan sambungan dari bagan alir yang terputus dihalaman yang sama / dihalaman yang lain.</p>
<p data-bbox="395 1184 542 1218">Keputusan</p> 	<p data-bbox="651 1189 1353 1279">Simbol keputusan (<i>decision symbol</i>), digunakan untuk suatu penyelesaian kondisi didalam program.</p>
<p data-bbox="347 1364 590 1397">Proses Terdefinisi</p> 	<p data-bbox="651 1352 1353 1487">Simbol proses terdefinisi digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain.</p>
<p data-bbox="405 1561 533 1594">Terminal</p> 	<p data-bbox="651 1588 1353 1677">Simbol terminal digunakan untuk menunjukkan awal dan akhir dari suatu program.</p>

Tabel 2.4 Simbol *Flowchart* (lanjutan)

<p>Dokumen</p> 	<p>Menunjukkan dokumen yang digunakan untuk input dan output baik secara manual maupun komputerisasi.</p>
<p>Proses manual</p> 	<p>Menunjukkan pekerjaan yang dilakukan secara manual.</p>
<p>Proses komputerisasi</p> 	<p>Menunjukkan proses dari operasi program komputer.</p>
<p>Simpanan</p> 	<p>Menunjukkan arsip.</p>
<p>Terminator</p> 	<p>Digunakan untuk memberikan awal dan akhir suatu proses.</p>
<p>Garis alir</p> 	<p>Digunakan untuk menunjukkan arus dari proses.</p>

Tabel 2.4 Simbol *Flowchart* (lanjutan)

<p>Decision</p> 	<p>Digunakan untuk suatu penyeleksian kondisi di dalam program.</p>
<p>Keyboard</p> 	<p>Menunjukkan input yang menggunakan keyboard.</p>
<p>Hard disk</p> 	<p>Media penyimpanan, menggunakan perangkat hard disk.</p>
<p>Keterangan</p> 	<p>Digunakan untuk memberikan keterangan yang lainnya.</p>
<p>Penghubung</p> 	<p>Simbol yang digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang sama maupun di halaman yang lain.</p>

2.11 Relasi

(Wawa Wikusna, 2016) Menunjukkan adanya hubungan/relasi diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Jenis-jenis hubungan:

- a. *One to one*, setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, dan begitu sebaliknya setiap

- entitas pada himpunan entitas B berhubungan dengan paling banyak satu entitas pada himpunan entitas A.
- b. *One to many*, setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan tidak sebaliknya dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak satu entitas pada himpunan entitas A.
 - c. *Many to one*, setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, dan tidak sebaliknya dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.
 - d. *Many to many*, setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan sebaliknya dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

2.12 Pengkodean

(Wawa Wikusna, 2016) alasan untuk membuat suatu pengkodean adalah untuk efisiensi ruang penyimpanan dari pemakaiannya, ada dua bentuk pengkodean:

- a. Eksternal (*user-defined coding*), mewakili pengkodean yang telah digunakan secara terbuka dan dikenal dengan baik oleh pemakai awam. Contoh: Nomor mahasiswa dan Kode mata kuliah => sudah dikenal baik oleh pemakai awam.
- b. Internal (*system coding*), menggambarkan bagaimana data disimpan dalam kondisi sebenarnya, sehingga lebih berorientasi pada mesin. Ada tiga bentuk pengkodean:
 - a. Sekuensial, pengkodean dilakukan dengan mengasosiasikan data dengan kode yangurut.

Contoh: predikat kelulusan “Sangat Memuaskan”, ”Cukup Memuaskan”, “Memuaskan” => dikodekan dengan huruf “A”, “B”, “C”

- b. *Mnemonic*, pengkodean dilakukan dengan membentuk suatu singkatan dari data yang hendak dikodekan.

Contoh: “Laki-laki” => dikodekan ‘L’ ; “Perempuan” => dikodekan “P”

- c. Blok, pengkodean dinyatakan dalam format tertentu.

Contoh : Nomor mahasiswa dengan format XX.YY.ZZZZ => terdiri atas XX = 2 digit tahun masuk, YY = 2 digit kode jurusan, ZZZZ = 4 digit nomor urut.

2.13 Bahasa Pemrograman dan Perangkat Lunak Pendukung

Bahasa pemrograman yang digunakan dalam penulisan skripsi ini adalah PHP, HTML5, dan CSS sedangkan perangkat lunak pendukung yang digunakan adalah *MySql*, XAMPP, dan *Sublime Text*.

(A.S, Rosa, M.Shalahudin, 2015) PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun *website* adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke *database*, *diedit*, dihapus, dan ditampilkan pada *website* yang diatur oleh PHP.

(Loka Dwiartara) Pengembangan demi pengembangan terus berlanjut, ratusan fungsi ditambahkan sebagai fitur dari bahasa PHP, dan di awal tahun 1999, netcraft mencatat, ditemukan 1.000.000 situs di dunia telah menggunakan PHP. Ini membuktikan bahwa PHP merupakan bahasa yang paling populer digunakan oleh dunia *web development*. Hal ini mengagetkan para *developer*nya termasuk Rasmus sendiri, dan tentunya sangat diluar dugaan sang pembuatnya. Kemudian Zeev Suraski dan Andi Gutsman selaku *core developer* (programmer inti) mencoba untuk menulis ulang *PHP Parser*, dan diintegrasikan dengan menggunakan *Zend scripting engine*, dan mengubah jalan alur operasi PHP. Dan semua fitur baru tersebut di rilis dalam PHP 4.13 Juli 2004, evolusi PHP, PHP

telah mengalami banyak sekali perbaikan disegala sisi, dan wajar jika *netcraft* mengumumkan PHP sebagai bahasa *web* populer didunia, karena tercatat 19 juta domain telah menggunakan PHP sebagai server side scriptingnya. PHP saat ini telah mendukung *XML* dan *Web Services*, mendukung *SQLite*. Tercatat lebih dari 19 juta domain telah menggunakan PHP sebagai *server scriptingnya*. Benar-benar PHP sangat mengejutkan. Keunggulan PHP adalah sebagai berikut:

- a. Gratis, apa yang membuat PHP begitu berkembang sangat pesat hingga jutaan domain menggunakan PHP, begitu populernya PHP? Jawabannya adalah karena PHP itu gratis. Saya sendiri menyukai bahasa yang satu ini selain mudah juga karena gratis.
- b. *Cross platform*, artinya dapat di gunakan di berbagai sistem operasi, mulai dari *linux*, *windows*, *mac os* dan *os* yang lain.
- c. Mendukung banyak *database*, PHP telah mendukung banyak *database*, ini mengapa banyak *developer web* menggunakan PHP Adabas D Adabas D, *dBase dBase*, *Empress Empress*, *FilePro (read-only) FilePro (read-only)* *Hyperwave*, *IBM DB2*, *Informix*, *Ingres*, *InterBase*, *FrontBase mSQL*, *Direct MS-SQL*, *MySQL*, *ODBC*, *Oracle (OCI7 and OCI8)*, *Ovrimos*, *PostgreSQL SQLite*, *Solid*, *Sybase*, *Velocis*, *Unix dbm*.
- d. *On The Fly*, PHP sudah mendukung *on the fly*, artinya dengan PHP anda dapat membuat *document text*, *Word*, *Excel*, *PDF*, menciptakan *image* dan *flash*, juga menciptakan file-file seperti *zip*, *XML*, dan banyak lagi.

(A.S, Rosa, M.Shalahudin, 2015) HTML singkatan dari *Hyper Text Markup Language*, yaitu skrip yang mengatur berupa *tag-tag* untuk membuat dan mengatur struktur *website*. Beberapa tugas utama HTML dalam membangun *website* diantaranya sebagai berikut:

- a. Menentukan layout *website*.
- b. Memformat *text* dasar seperti pengaturan paragraf, dan *format font*.

- c. Membuat *list*.
- d. Membuat *table*.
- e. Menyisipkan gambar, video, dan audio.
- f. Membuat *link*.
- g. Membuat formulir

(Rohi Abdulloh, 2015) CSS singkatan dari *Cascading Style Sheets*, yaitu skrip yang digunakan untuk mengatur desain *website*. Walaupun HTML mempunyai kemampuan untuk mengatur tampilan *website*, namun kemampuannya sangat terbatas. Fungsi CSS adalah memberikan pengaturan yang lebih lengkap agar struktur *website* yang dibuat dengan HTML terlihat lebih rapi dan indah.

Menurut Wahana Komputer (2009) Microsoft Visio 2007 adalah salah satu program yang dapat digunakan untuk membuat diagram. Aplikasi ini menyediakan banyak fasilitas pembuatan diagram untuk menggambarkan informasi dan sistem. Visio dapat menghasilkan suatu diagram mulai dari yang sederhana hingga diagram yang lebih kompleks. Anda hanya perlu shap dengan menarik shap ke halaman kerja. Oleh karena Visio merupakan salah satu produk dari Microsoft Office lainnya, maka visio dapat berkolaborasi dengan produk Microsoft word, Exel, Powerpoint dan project, bentuk kolaborasi dapat dilakukan dengan fasilitas ekspor dan impor data

(Loka Dwiartara) *MySQL* bersifat *RDBMS (Relational Database Management System)* yang memungkinkan seorang admin dapat menyimpan banyak informasi ke *table-table*, dimana *table-table* tersebut saling berkaitan satu sama lain. Keuntungan *RDBMS* sendiri adalah kita dapat memecah *database* kedalam *table-table* yang berbeda, setiap *table* memiliki informasi yang berkaitan dengan *table* yang lainnya.

(Andi, 2009) XAMPP adalah salah satu paket *software web server* yang terdiri dari *Apache, MySQL, PHP* dan *phpMyAdmin*. Mengapa menggunakan XAMPP?

Karena XAMPP sangat mudah penggunaannya, terutama jika Anda seorang pemula. Proses instalasi XAMPP sangat muda, karena tidak perlu melakukan konfigurasi Apache, PHP dan *MySQL* secara manual, XAMPP melakukan instalasi dan konfigurasi secara otomatis.

(Pemula belajar.com) *Sublime Text Editor* adalah *editor* teks untuk berbagai bahasa pemrograman termasuk pemrograman PHP. *Sublime Text Editor* merupakan *editor text* lintas-*platform* dengan *Python application programming interface* (API). *Sublime Text Editor* juga mendukung banyak bahasa pemrograman dan bahasa *markup*, dan fungsinya dapat ditambah dengan *plugin*, dan *Sublime Text Editor* tanpa lisensi perangkat lunak.

2.14 Penelitian Sebelumnya

- Nama jurnal : Perancangan Sistem Informasi Poin Pelanggaran Dan Prestasi Siswa Berbasis Sms Gateway Pada Sekolah Menengah Kejuruan Negeri 3 Pacitan
- Penulis : Ellen Ershinta Putri dan Azis Ahmadi
- Tahun : 2015
- Perangkat lunak : SMS Gateway, PHP, HTML, *MySQL* yang digunakan
- Kelebihan : Sistem ini memberikan kemudahan kepada admin dan pihak orang tua siswa untuk mengetahui jumlah pelanggaran yang dilakukan ataupun bimbingan terhadap permasalahan yang dihadapi oleh siswa. Sehingga nantinya berpengaruh bagi perkembangan siswa selanjutnya dalam penyesuaian dirinya baik dilingkungan sekolah maupun masyarakat.