

## BAB II

### LANDASAN TEORI

#### 2.1 Sekolah Tinggi Keguruan dan Ilmu Pendidikan Aisyiyah Riau

Aisyiyah merupakan organisasi otonom dalam wadah gerakan Muhammadiyah. Aisyiyah sebagai gerakan wanita Islam, merasa terpanggil untuk berperan mencerdaskan kehidupan berbangsa. Naluri 'keibuan' yang tumbuh dalam organisasi ini kemudian terefleksi dalam sebuah gerakan tarbiyah (Pendidikan). Hal ini selaras dengan gerakan Muhammadiyah dan cita-cita dasar dari bangsa Indonesia. Dalam rangka mengimplementasikan semangat tersebut, maka atas prakarsa pimpinan wilayah Aisyiyah Riau, pada tahun 1998 didirikan program pendidikan Diploma I Pendidikan Guru Taman Kanak-Kanak. Program ini berdiri dengan izin operasional DEPDIKBUD Riau. Latar belakang berdirinya diawali oleh keresahan ibu-ibu Aisyiyah terhadap perkembangan anak usia dini di wilayah Provinsi Riau. Tahun 2002 STKIP Aisyiyah Riau mengupayakan peningkatan status prodi dari diploma menjadi S-1 (Strata 1). Selain itu juga diupayakan penambahan prodi pada waktu yang sama. Upaya ini berhasil dengan diterbitkannya SK Mendiknas No:205/O/D/2002 tanggal 5 September tahun 2002 tentang pemberian izin operasional program Studi Diploma-II PGTK dan Strata I Pendidikan Ekonomi.

Seiring dengan perkembangan dunia pendidikan anak usia dini dan kepercayaan masyarakat, kemudian diupayakan peningkatan status program Diploma II PGTK menjadi program Studi S-1 PG-Paud. Langkah ini juga berhasil dengan diterbitkannya SK Mendiknas No : 2094/D/T/2008 tanggal 07 juli 2008 tentang izin penyelenggaraan program studi Strata I Pendidikan Guru Pendidikan Anak Usia Dini (PG-PAUD). Tahun 2011 adalah tahun dimana STKIP Aisyiyah Riau mulai melakukan upaya peningkatan kualitas secara berkesinambungan. Upaya tersebut diupayakan secara sinergis diantara segenap pemangku kepentingan pendidikan tinggi. Diantara upaya tersebut antara lain:

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

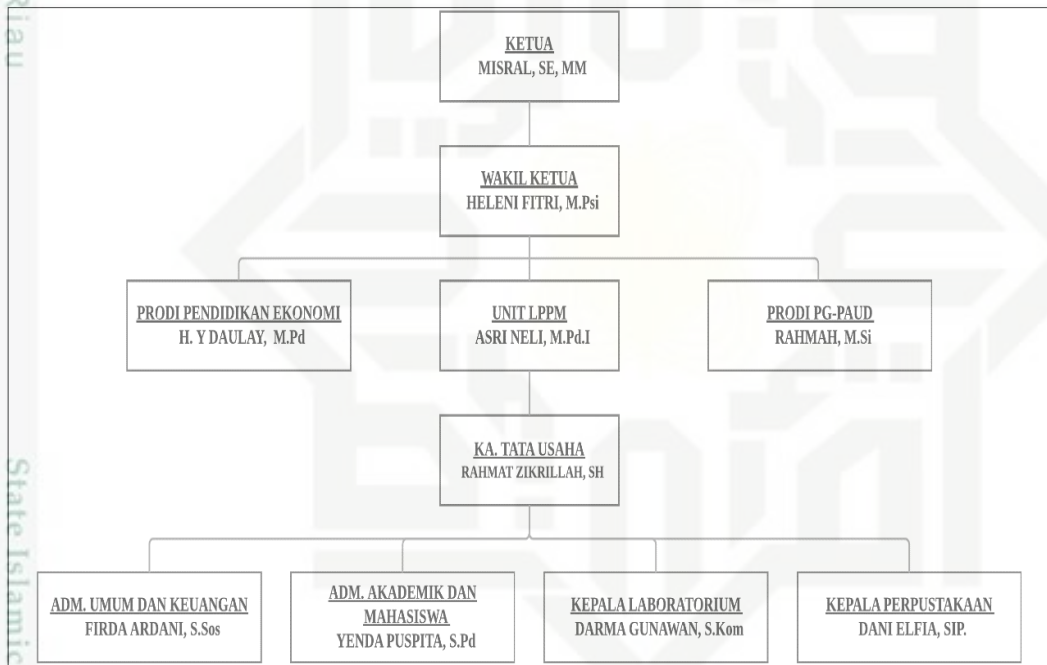
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1. Evaluasi kurikulum S-1 PG-PAUD dan S-1 Pendidikan Ekonomi yang melibatkan kaum akademisi, praktisi, dan pengguna alumni.
2. Perbaiki sistem administrasi akademik dan perkuliahan dengan pendekatan SCL (*student centre learning*).
3. Peningkatan program penelitian dan pengabdian masyarakat oleh dosen dan mahasiswa.

### 2.1.1 Struktur Pamong STKIP Aisyiah Riau

Secara garis besar struktur pamog STKIP Aisyiah Riau dapat dilihat pada Gambar 2.1



Gambar 2.1 Struktur Pamog STKIP Aisyiah Riau

### 2.1.2 Tugas Pokok dan Fungsi Jabatan

Tugas pokok dan fungsi jabatan pada STKIP Aisyiah adalah:

1. Ketua  
 Ketua adalah pimpinan Sekolah Tinggi Keguruan dan Ilmu Pendidikan Aisyiah Riau yang berada dibawah dan bertanggung jawab langsung kepada Majelis Dikti Pimpinan Pusat Aisyiah. Ketua mempunyai tugas:

**Hak Cipta Dilindungi Undang-Undang**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

- a. Memimpin, menyelenggarakan pendidikan, penelitian dan pengabdian kepada masyarakat, membina tenaga kependidikan, mahasiswa, tenaga administrasi serta hubungan dengan lingkungan.
- b. Membina dan melaksanakan kerjasama dengan instansi, badan swasta dan masyarakat untuk memecahkan persoalan yang timbul, terutama yang menyangkut bidang tanggung jawab.

2. Wakil Ketua

Dalam melaksanakan tugas sehari-hari, ketua dibantu 1 (satu) orang wakil ketua yang berada di bawah dan bertanggung jawab langsung kepada ketua. Wakil ketua mempunyai tugas membantu ketua dalam memimpin pelaksanaan pendidikan, penelitian dan pengabdian kepada masyarakat, serta memimpin pelaksanaan kegiatan dibidang pembinaan serta layanan kesejahteraan mahasiswa, ke Aisiyahan dan perlengkapan. Apabila ketua berhalangan, ketua menunjuk wakil ketua sebagai pelaksana harian.

3. Administrasi Akademik Dan Kemahasiswaan

Bagian administrasi akademik dan kemahasiswaan mempunyai tugas memberikan layanan administrasi dibidang akademik di lingkungan Sekolah Tinggi Keguruan dan Ilmu Pendidikan (STKIP) 'Aisiyiah Riau. Untuk menyelenggarakan tugas tersebut pada bagian administrasi akademik dan kemahasiswaan mempunyai fungsi:

- a. Melaksanakan administrasi pendidikan.
- b. Melaksanakan administrasi kerjasama.
- c. Melaksanakan administrasi minat, penalaran dan informasi mahasiswa.
- d. Melaksanakan administrasi kesejahteraan mahasiswa.

4. Kepala Perpustakaan

Kepala perpustakaan dalam menjalankan tugasnya dibantu oleh seorang sekretaris perpustakaan yang diangkat oleh Ketua atas usul kepala perpustakaan. Kepala perpustakaan mempunyai tugas memberikan pelayanan bahan perpustakaan untuk keperluan pendidikan, penelitian dan pengabdian kepada masyarakat. Untuk menyelenggarakan tugas, kepala perpustakaan mempunyai fungsi:

**Hak Cipta Dilindungi Undang-Undang**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

- a. Menyediakan dan mengolah bahan pustaka.
- b. Memberikan layanan dan pendayagunaan bahan pustaka.
- c. Memelihara bagan pustaka.
- d. Melakukan pelayanan refrensi.
- e. Melakukan urusan administrasi perpustakaan.

5. Kepala Laboratorium

Pusat Komputer dipimpin oleh seorang kepala laboratorium yang ditunjuk diantara tenaga akademik/tenaga teknis komputer senior di lingkungan pusat komputer. Kepala laboratorium mempunyai tugas mengumpulkan mengolah, menyajikan dan menyimpan data dan informasi serta memberikan layanan untuk program-program pendidikan, penelitian dan pengabdian kepada masyarakat. Pusat Komputer adalah unit pelaksana teknis dibidang pengolahan data yang berada dibawah dan bertanggung jawab langsung kepada Ketua.

6. Lembaga Penelitian dan Pengabdian kepada Masyarakat

Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM) adalah pelaksana pembinaan dan pengembangan ilmu pendidikan, teknologi dan pengabdian pada masyarakat dengan berpedoman pada ketentuan yang ditetapkan oleh pimpinan ketua. Dalam melaksanakan tugas tersebut Lembaga Penelitian dan Pengabdian kepada masyarakat menyelenggarakan fungsi:

- a. Perumusan kebijakan penelitian dan pengabdian masyarakat.
- b. Pelaksanaan penelitian lingkup ilmu pengetahuan, teknologi, dan pendidikan yang dimaksudkan untuk menunjang pengembangan konsepsi pembangunan nasional, wilayah, dan/atau daerah.
- c. Pelaksanaan tugas administrasi lembaga penelitian.
- d. Pelaksanaan pengembangan pola dan konsepsi pembangunan nasional, wilayah dan/atau daerah berdasar nilai-nilai lokal, ilmu pengetahuan, teknologi, dan/atau seni.
- e. Pengamalan hasil-hasil penelitian lingkup ilmu pengetahuan, teknologi dan pendidikan.



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

- f. Peningkatan relevansi program pengabdian sesuai kebutuhan masyarakat
- g. Pelaksanaan pemberian bantuan keahlian kepada masyarakat dalam melaksanakan pembangunan.

## 2.2 Sistem Repository

Institusi akademis dunia memiliki kekayaan berupa arsip, cetak dan bentuk non cetak serta berbagai media penyimpanan. Harta karun ini mengandung aset ilmiah, teknologi, budaya dan sejarah yang pada dasarnya tidak tersedia bagi peneliti dan masyarakat umum. Sistem *repository* dibuat untuk mengelola, melestarikan dan memelihara output intelektual digital dari institusi. Pustakawan dan profesional informasi mengambil inisiatif dalam merencanakan, membuat dan mengelola *repository* untuk konservasi dan pelestarian keluaran intelektual dan memenuhi tujuan organisasi mereka. Kekayaan intelektual yang dihasilkan oleh perguruan tinggi sebagai hasil proses pembelajaran dan penelitian tidak hanya tersimpan di perpustakaan, tetapi juga telah dilakukan upaya desiminasi secara terbuka (Ariyanto, 2010).

Inti dari *repository* adalah membuat publikasi penelitian dan pengembangan yang tersedia di Internet. *Repository* digunakan organisasi pendidikan dan lembaga untuk menyebarkan penelitian dan hasil publikasi lainnya. Dokumen yang diterbitkan seperti jurnal, makalah, artikel, buku, paten, dan laporan teknis. *Repository* sekarang menjadi *platform* penting untuk berbagi pengetahuan yang dihasilkan oleh organisasi. Perpustakaan perguruan tinggi harus pandai merespon perubahan perilaku civitas akademik dalam mengakses informasi. Mereka menjadikan informasi sebagai kebutuhan pokok mereka, terbukti mereka harus rela terus mengikuti informasi dan mengupdate informasi lewat sarana teknologi informasi berupa komputer, gadget dan sarana lain. *Local content* adalah sumber-sumber informasi yang khas dan unik yang dihasilkan dan dimiliki oleh suatu institusi yang tersedia dalam format digital dan salah satunya adalah jurnal-jurnal yang diterbitkan pada lingkup universitas maupun fakultas atau jurusan (Ariyanto, 2010).

#### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

*Repository* juga merupakan sebuah basis data yang digunakan bersama yang memuat informasi mengenai objek yang dihasilkan atau digunakan oleh suatu organisasi, misalnya perangkat lunak, dokumen, peta, sistem informasi, maupun komponen diskrit dari sistem produksi. Objek tersebut umumnya disimpan dalam berbagai sistem penyimpanan (*storage*), dan deskripsi dari objek tersebut disimpan dalam *repository*. Dalam *repository* umumnya juga terdapat informasi lokasi, sejarah revisi, proses pembuatan, pengaturan hak akses, serta beberapa informasi penting lain yang berhubungan dengan objek yang disimpan. Selain itu, sebuah sistem *repository* juga mengatur sistem *checkout* dan *checkin* berupa fasilitas unggah dan unduh informasi, dan aliran proses. Tanpa adanya sistem *repository*, maka informasi penting akan sulit diakses, baik oleh komunitas yang lebih luas maupun oleh si pembuat sendiri. Dengan bantuan sistem *repository*, maka tantangan ini akan dapat teratasi, tentunya dengan tetap memperhatikan privasi dan hak cipta pembuat informasi. Untuk membangun sebuah sistem *repository*, maka diperlukan proses analisis kebutuhan sistem. (Tjandra, 2016).

### 2.2.1 Identifikasi *Institutional Repository* Perguruan Tinggi

Dalam *repository* perguruan tinggi terdapat bermacam jenis koleksi, salah satunya adalah *local content* berupa karya ilmiah hasil karya civitas akademik. Dalam pengelolaannya dapat berupa *hardcopy*, koleksi CD, maupun *softfile*. Menurut Ariyanto (2010) sumber informasi berupa literatur wajib disimpan di perpustakaan dengan keputusan pimpinan lembaga induknya seperti: skripsi, tesis, disertasi, makalah-makalah seminar, dan laporan penelitian atau laporan kegiatan, serta publikasi internal (jurnal, buku, majalah, bulletin). Dalam tulisan ini, pengertian *institutional repository* mencakup dua hal di atas yaitu koleksi lokal maupun literatur kelabu yang dibutuhkan maupun dihasilkan oleh perguruan tinggi. Koleksi tersebut dikelola untuk perkembangan pengetahuan serta referensi penelitian berikutnya (Ariyanto, 2010).

### 2.2.2 Manfaat *Institutional Repository*

Produktifitas karya intelektual menjadikan penulis akan semakin dikenal, hal ini sangat menguntungkan sebagai media promosi bagi penulis agar semakin

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

dikenal, disamping juga sebagai media promosi bagi intitusi itu sendiri. Karya-karya yang kredibel, aktual dan member solusi akan menjadi rujukan dan sebagai sarana promosi kelembagaan yang efektif dan berdampak positif (Ariyanto, 2010).

### 2.3 *Object Oriented Analysis and Design (OOAD)*

OOAD adalah metode analisis yang memeriksa *requirements* dari sudut pandang kelas kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. Ada beberapa ciri khas dari pendekatan ini menurut (Fatta, 2007):

1. *Object* adalah struktur yang mengenkapsulasi atribut dan metode yang beroperasi berdasarkan atribut-atribut. *Object* adalah abstraksi dari benda nyata dimana data dan proses diletakkan bersama untuk memodelkan struktur dan prilaku dari objek dunia nyata.
2. *Object class* adalah sekumpulan objek yang berbagi struktur yang sama dan prilaku yang sama.
3. *Inheritance*, merupakan properti yang muncul ketika tipe entitas atau *object class* disusun secara hirarki dan setiap tipe entitas atau *object class* menerima atau mewarisi atribut dan metode dari pendahulunya.

#### 2.3.1 *Dasar Object Oriented Analysis and Design (OOAD)*

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu Analisis Berorientasi Objek (OOA) dan Desain Berorientasi Objek (OOD). Analisis Berorientasi Objek adalah tahapan untuk menganalisis spesifikasi atau kebutuhan akan sistem yang akan dibangun dengan konsepberorientasi objek. Sedangkan Desain Berorientasi Objek adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem (Salahuddin, 2014).



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

### 2.3.2 Kelebihan dan Kekurangan Perancangan Berorientasi Objek

Adapun kelebihan dan kekurangan perancangan berorientasi objek sebagai berikut:

1. Kelebihan Perancangan Berorientasi Objek:

- a. Dibandingkan dengan metode SSAD, OOAD lebih mudah digunakan dalam pembangunan sistem.
- b. Dibandingkan dengan SSAD, waktu pengembangan, level organisasi, ketangguhan, dan penggunaan kembali (*reuse*) kode program lebih tinggi dibandingkan dengan metode OOAD (Sommerville, 2003).
- c. Tidak ada pemisahan antara fase desain dan analisis, sehingga meningkatkan komunikasi antara user dan developer dari awal hingga akhir pembangunan sistem.
- d. Analis dan programmer tidak dibatasi dengan batasan implementasi sistem, jadi desain dapat diformulasikan yang dapat dikonfirmasi dengan berbagai lingkungan eksekusi.
- e. Relasi obyek dengan entitas (*thing*) umumnya dapat di mapping dengan baik seperti kondisi pada dunia nyata dan keterkaitan dalam sistem. Hal ini memudahkan dalam memahami desain (Sommerville, 2003).
- f. Memungkinkan adanya perubahan dan kepercayaan diri yang tinggi terhadap kebenaran software yang membantu untuk mengurangi resiko pada pembangunan sistem yang kompleks.
- g. *Encapsulation* data dan method, memungkinkan penggunaan kembali pada proyek lain, hal ini akan memperjelas proses desain, pemrograman dan reduksi harga.
- h. OOAD memungkinkan adanya standarisasi obyek yang akan memudahkan memahami desain dan mengurangi resiko pelaksanaan proyek.
- i. Dekomposisi obyek, memungkinkan seorang analis untuk memecah masalah menjadi pecahan-pecahan masalah dan bagian-bagian yang dimanage secara terpisah. Kode program dapat dikerjakan bersama-sama. Metode ini memungkinkan pembangunan software dengan cepat,



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

sehingga dapat segera masuk ke pasaran dan kompetitif. Sistem yang dihasilkan sangat fleksibel dan mudah dalam memelihara.

## 2.4 Unified Modeling Language

*Unified Modeling Language* (UML) adalah hasil kerja dari konsorium berbagai organisasi yang berhasil digunakan sebagai standar baku dalam OOAD. UML bisa berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem (Munawar, 2005).

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasabahasa berorientasi objek. Tujuan utama dalam desain UML adalah:

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

## 2.5 Diagram-Diagram UML







UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. UML mempunyai sejumlah aturan untuk menggabungkan atau

mengkombinasikan elemen elemen tersebut. UML sudah cukup banyak menyediakan diagram yang bisa membantu mendefinisikan sebuah aplikasi. Sangat penting untuk bisa mengeluarkan setiap diagram tersebut, karena semua diagram bisa mewakili stakeholder yang berbeda di sistem tersebut.



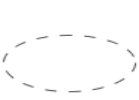

### 2.5.1 Diagram Use Case

*Use Case* adalah deskripsi fungsi sebuah system dari prespektif pengguna. *Use case* bekerja dengan cara mendiskripsikan tipikal interaksi antara pengguna sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut skenario. Setiap skenario mendeskripsikan urutan kejadian. Ide dasarnya bagaimana melibatkan pengguna sistem di fase-fase awal analisis dan perancangan sistem. Dengan demikian diharapkan akan bisa dibangun suatu sistem yang bisa membantu pengguna untuk bisa membuat pandangan di luar sistem (Munawar, 2005). Simbol dan keterangan *use case* dapat dilihat pada Tabel 2.1

Tabel 2.1 Simbol Diagram *Use Case*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		Generalization	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		Include	Menspesifikasikan bahwa use case sumber secara eksplisit.
5		Extend	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
6		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.

Tabel 2.2 Tabel 2.1 Simbol Diagram *Use Case* (Lanjutan)





No	Gambar	Nama	Keterangan
7		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

(Sumber: Edgar dan Johan, 2013).

### 2.5.2 Diagram Activity


Diagram *activity* menggambarkan aliran fungsionalitas sistem. Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. Aktivitas dalam digram dipresentasikan dengan bentuk bujur sangkar bersudut tidak lancip, yang didalam nya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja. Ada sebuah keadaan mulai (*start state*) yang menunjukkan dimulainya aliran kerja, dan sebuah keadaan selesai (*end state*) yang menunjukkan akhir diagram, titik keputusan dipresentasikan dengan *diamond*. Diagram *activity* tidak perlu dibuat untuk setiap aliran kerja, tetapi diagram ini akan sangat berguna untuk aliran kerja yang kompleks dan melebar. Simbol dan keterangan diagram *activity* dapat dilihat pada Tabel 2.2.

Tabel 2.2. Simbol Diagram *Activity*

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan



Tabel 2.2. Simbol Diagram *Activity* (Lanjutan)

No	Gambar	Nama	Keterangan
5		Fork Node	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran




(Sumber: Edgar dan Johan, 2013)

### 2.5.3 Diagram *Sequence*

Diagram *sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

Simbol dan keterangan diagram *sequence* dapat dilihat pada Tabel 2.3.

Tabel 2.3 Simbol Diagram *Sequence*

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi informasi tentang aktifitas yang terjadi.








(Sumber: Edgar dan Johan, 2013)

### 2.5.4 Diagram *Class*

Diagram *class* merupakan diagram paling umum yang dijumpai dalam pemodelan berbasis UML. Didalam *Class diagram* terdapat *class* dan *interface* beserta atribut-atribut dan operasinya, relasi yang terjadi antar objek, *constraint* terhadap objek-objek yang saling berhubungan dan *inheritance* untuk organisasi *Class* yang lebih baik. *Class diagram* juga terdapat *static view* dari elemen pembangun sistem. Pada intinya *Class diagram* mampu membantu proses pembuatan sistem dengan memanfaatkan konsep *forward* ataupun *reverse engineering* (Edgar dan Johan, 2013). Simbol dan keterangan diagram *activity* dapat dilihat pada Tabel 2.4.



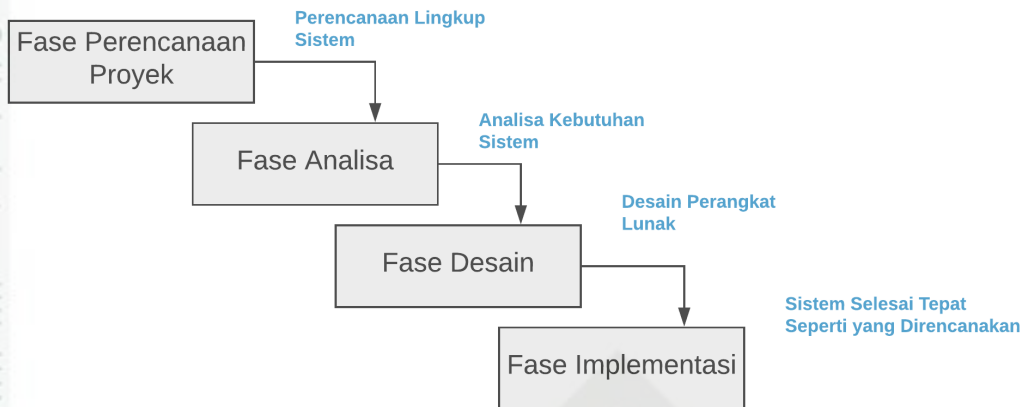
Tabel 2.4 Simbol Diagram *Class*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

(Sumber: Edgar dan Johan, 2013).

## 2.6 SDLC Model Air Terjun (*Waterfall*)

Salah satu model SDLC yang paling awal dan paling banyak digunakan adalah SDLC model air terjun (*Waterfall*). Model SDLC yang dikembangkan oleh Satzinger, Jackson, & Burd (2010, p40) menggambarkan pendekatan sekuensial beberapa tahap yang biasanya disebut juga dengan model air terjun. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atauurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap *support*.



Gambar 2.2. Model SDLC *Waterfall*

(Sumber: Satzinger dkk, 2010)

Model SDLC menurut Satzinger ini mengatakan bahwa hal pertama yang dilakukan adalah dengan mendefinisikan perumusan permasalahan untuk kelak dilakukan suatu pemecahan dari perumusan tersebut (*plan*). Selanjutnya tim proyek menganalisis, mendefinisikan dan memahami secara menyeluruh masalahnya beserta kebutuhan untuk selanjutnya dicari solusi (*analysis*). Setelah masalah dipahami, solusi ditinjau secara mendalam (*design*). Sistem ini kemudian dibangun, dikembangkan dan diinstal (*implementation*).

Berdasarkan Gambar 2.2 beberapa tahapan dalam SDLC model air terjun ini dijelaskan sebagai berikut:

1. Fase Perencanaan (*Project Planning Phase*)

Kegiatan awal dari SDLC yang bertujuan untuk mengidentifikasi lingkup sistem yang baru dan rencana dari suatu proyek. Aktivitas yang tercantum dalam perencanaan ini adalah:

- a. Mendefinisikan masalah

Kegiatan yang paling penting dari perencanaan proyek adalah untuk mendefinisikan dengan tepat masalah bisnis dan ruang lingkup solusi yang dibutuhkan.

- b. Menghasilkan jadwal proyek dan menentukan staf.

Sebuah jadwal proyek yang menghasilkan rincian daftar tugas, kegiatan, dan staf yang diperlukan dikembangkan.

- c. Konfirmasi kelayakan proyek.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Analisis kelayakan menyelidiki kelayakan anggaran, organisasi, pelaksana teknis, sumber daya, dan pelaksanaan jadwal proyek.

d. Peluncuran proyek

Total dari keseluruhan rencana proyek yang telah ditinjau kepada pimpinan, proyek mulai dilaksanakan

2. Fase Analisis (*Analysis Phase*)

Analisis kebutuhan sistem dilakukan untuk identifikasi dan evaluasi permasalahan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan. Menurut Rosa dan Shalahuddin (2011), pada tahap ini dilakukan dalam analisis sistem antara lain:

a. Analisa Identifikasi Masalah

Identifikasi masalah merupakan langkah awal dari analisis sistem. Dalam tahap ini didefinisikan masalah yang harus dipecahkan.

b. Analisa Kebutuhan

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan pengguna.

c. Analisa Kelayakan Sistem

Studi kelayakan digunakan untuk menentukan kemungkinan keberhasilan solusi yang diusulkan. Tahapan ini berguna untuk memastikan bahwa solusi yang diusulkan tersebut benar-benar dapat dicapai.

3. Fase Design (*Design Phase*)

Desain perangkat lunak adalah proses multi langkah yang berfokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Menurut Rosa dan Shalahuddin (2011), tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

#### 4. Fase Implementasi

Segala bentuk masukan (*input*) yang dilakukan dari desain sistem, sistem yang telah dikembangkan menjadi sebuah program yang disebut dengan unit, yang mana berhubungan dengan tahapan selanjutnya. Dari tiap-tiap unit tersebut dikembangkan dan di uji coba untuk mengetahui fungsi yang berkaitan dengan unit tersebut.

#### 2.6.1 Kelemahan Menggunakan Metode *Waterfall*

1. Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
2. Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan yang berakibat pada tahapan selanjutnya.
3. Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidak pastian pada saat awal pengembangan.
4. Pelanggan harus sabar, karena pembuatan perangkat lunak akan dimulai ketika tahap desain sudah selesai. Sedangkan pada tahap sebelum desain bisa memakan waktu yang lama.
5. Pada kenyataannya, jarang mengikuti urutan sekuensial seperti pada teori. Iterasi sering terjadi menyebabkan masalah baru.

#### 2.6.2 Kelebihan Menggunakan Metode *Waterfall*

1. Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
2. Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.
3. Metode ini masih lebih baik digunakan walaupun sudah tergolong kuno, daripada menggunakan pendekatan asal-asalan. Selain itu, metode ini juga masih masuk akal jika kebutuhan sudah diketahui dengan baik.



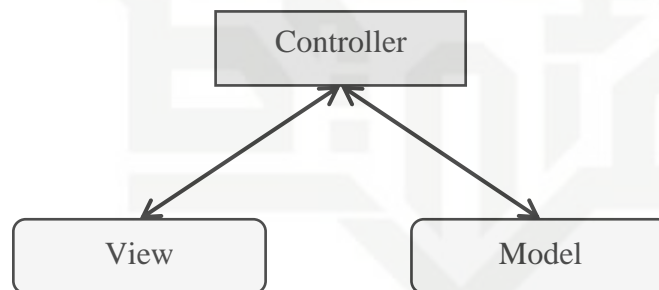
## 2.7 Framework CodeIgniter

Hidayatullah (2014) menjelaskan *CodeIgniter* adalah salah satu *framework* PHP yang paling *powerful* saat ini karena di dalamnya terdapat fitur lengkap aplikasi *web* dimana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu *CodeIgniter* juga saat ini banyak digunakan khususnya bagi *developer web* untuk mengembangkan aplikasi berbasis *web* tersebut.

Menurut Hidayatullah (2014) ada empat keuntungan yang didapat dari penggunaan *framework* adalah:

1. Menghemat waktu pengembangan.
2. Penggunaan ulang program/kode.
3. Bantuan komunitas.
4. Kumpulan program terbaik.

*Model View Controller* atau yang disebut dengan MVC adalah suatu metode yang memisahkan *data logic* (Model) dari *presentation logic* (View) dan *process logic* (Controller) atau secara sederhana adalah memisahkan antara desain *interface*, data dan proses (Hidayatullah, 2014). Tiga komponen pada *framework codeigniter* dapat dilihat pada Gambar 2.3.



Gambar 2.3 Metode MVC  
 (Sumber: Hidayatullah, 2014).

Terdapat tiga komponen dalam metode MVC menurut Hidayatullah (2014) yaitu:

1. Model  
 Model mengelola basis data (RDBMS) seperti MySQL ataupun RDMS. Model berhubungan dengan *database* sehingga biasanya dalam model

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

akan berisi *class* ataupun fungsi untuk membuat (*create*), melakukan pembaruan (*update*), menghapus data (*delete*), mencari data (*search*), dan mengambil data (*select*) pada *database*.

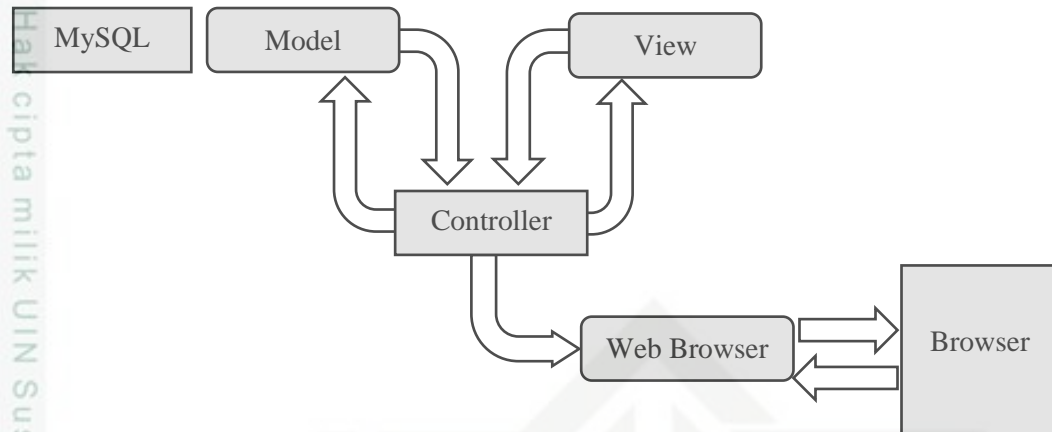
2. *View*

*View* adalah bagian *user interface* atau bagian yang nantinya merupakan tampilan untuk *end-user*. *View* bisa berupa halaman html, css, rss, *javascript*, *jquery*, *ajax*, dan lain-lain.

3. *Controller*

Merupakan penghubung antara *view* dan model, maksudnya ialah karena model tidak dapat berhubungan langsung dengan *view* ataupun sebaliknya. Sehingga tugas *controller* ialah sebagai jembatan pemrosesan data atau alur *logic* program, menyediakan variabel yang akan ditampilkan di *view*, pemanggilan model sehingga model dapat mengakses *database*.

Untuk mengetahui bagaimana alur MVC pada *CodeIgniter*, digambarkan *flow diagram* dalam hal ini *user* mengakses *browser* untuk menjalankan *web* yang berbasis *codeIgniter*, dapat dilihat pada Gambar 2.4.



Gambar 2.4 *Flow Diagram CodeIgniter*  
 (Sumber : Hidayatullah, 2014)

## 2.8 MySql

MySQL adalah sistem manajemen basis data Open Source yang paling populer, dikembangkan, didistribusikan, dan didukung oleh Oracle Corporation. Database MySQL bersifat relasional dan menyimpan data dalam tabel terpisah daripada meletakkan semua data dalam satu gudang besar. Struktur basis data diatur ke dalam file fisik yang dioptimalkan untuk kecepatan. Model logis, dengan objek seperti database, tabel, pandangan, baris, dan kolom, menawarkan lingkungan pemrograman yang fleksibel. Pengguna membuat aturan yang mengatur hubungan antara berbagai bidang data, seperti satu-ke-satu, satu-ke-banyak, unik, wajib atau opsional, dan pointer di antara tabel yang berbeda (Mysql.com). MySQL sebuah perangkat lunak sistem manajemen basis data SQL *database management system* yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia (Ramadhani dkk, 2013). Beberapa kelebihan yang dimiliki MySQL, diantaranya:

1. MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah (*Multi user*).
2. MySQL memiliki kecepatan yang bagus dalam menangani *query* sederhana, yaitu dapat memproses lebih banyak SQL per saana waktu.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

3. MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, text, date, timestamp*, dan masih banyak lagi.
4. MySQL memiliki operator dan fungsi secara penuh dan mendukung perintah *select* dan *where* dalam perintah *query*.
5. MySQL memiliki keamanan yang bagus karena beberapa lapisan keamanan seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang lengkap serta sandi terenkripsi.
6. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris.
7. MySQL dapat melakukan koneksi dengan klien menggunakan protocol TCP, Unix soket (UNIX), atau Named Pipes (NT).
8. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa.
9. MySQL memiliki *interface* terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

## 2.9 Blackbox Testing

Pengujian fungsional atau pengujian *blackbox testing* merupakan pendekatan pengujian yang ujinya diturunkan dari spesifikasi program atau komponen. Sistem merupakan kotak hitam yang perilakunya hanya dapat ditentukan dengan mempelajari input dan output yang berkaitan. Nama lain untuk cara ini adalah pengujian fungsional karena pengujian hanya berkepentingan dengan fungsionalitas dan berfokus pada impelementasi perangkat lunak.

Pendekatan ini dapat diterapkan pula pada sistem yang disusun sebagai fungsi atau sebagai objek. Pengujian memberikan input kepada komponen atau sistem dan meliputi output yang dihasilkan. Jika output bukan merupakan yang diramalkan, berarti uji telah dengan hasil mendeteksi masalah dengan perangkat lunak tersebut (Sommerville, 2006). *Blackbox testing* berusaha menemukan kesalahan dalam kategori sebagai berikut:





Selanjutnya penelitian oleh Irwansyah (2015) dengan judul “Sistem Informasi Repository Beban Kerja Dosen” menyimpulkan bahwa setiap perguruan tinggi memiliki kegiatan tri dharma perguruan tinggi yang dilakukan oleh dosen meliputi 4 bidang, yaitu: Bidang pendidikan dan pengajaran, penelitian, pengabdian kepada masyarakat, penunjang yang ditetapkan oleh pimpinan universitas. Oleh karena itu beban kerja dosen harus terdistribusi secara proporsional dan terukur. Sebagai satuan ukuran beban kerja dosen dinyatakan dalam satuan kredit semester disingkat SKS. Diharapkan semua aktivitas dosen yang di ukur sebagai beban dosen dalam menjalankan tri dharma perguruan tinggi harus dilakukan secara melembaga. Sistem Informasi *Repository Digital* Beban Kerja Dosen ini dapat membantu dosen yang ingin mengetahui informasi beban kerja yang di tanggung, serta membantu pimpinan dalam mengetahui informasi beban kerja yang di tanggung setiap dosen. Berdasarkan hasil pengujian dan penilaian responen mnunjukkan bahwa aplikasi ini dinilai positif dan berhasil (Irwansyah, 2015).

**Hak Cipta Dilindungi Undang-Undang**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.