

BAB II

KAJIAN PUSTAKA

Pada bab ini dijelaskan tentang penelitian yang berhubungan dengan landasan teori untuk digunakan sebagai pedoman dalam melakukan penelitian.

2.1 Penelitian Terkait

Pada bab ini dijelaskan penelitian yang digunakan, untuk itu akan dibahas penelitian sebelumnya guna membandingkan dalam tugas akhir ini.

Penelitian pertama yang mempunyai judul yaitu perangkian obat tradisional berdasarkan gejala penyakit menggunakan metode cosine similarity. Dalam judul ini didapatkan bahwa metode cosine similarity yang digunakan untuk memberikan peringkat pada jenis obat-obatan yang sangat mirip. Pada saat pengguna memasukkan nama penyakit yang diderita, kemudian sistem akan mengukur nilai kemiripan pada penyakit tersebut. Dalam proses ini akan menganalisa pemobotan dengan metode TF-IDF untuk mencari frekuensi kata dan menghitung bobot kata setiap dokumen (Perdana, 2014).

Pada penelitian berikutnya mempunyai judul yaitu rancang bangun aplikasi pengklasifikasian halaman web berdasarkan konten menggunakan metode cosine similarity. Dapat dibuktikan bahwa metode cosine similarity dapat membedakan dokumen secara keseluruhan. Dalam tugas akhir ini didapatkan bahwa menggunakan aplikasi ini dapat membedakan dokumen sekitar 30%, dan ini membuktikan bahwa tingkat akurasi dengan menggunakan aplikasi jauh lebih rendah dari pada menggunakan halaman web itu sendiri yang telah ditentukan sebelumnya (Prima and Rhosady, 2013).

Penelitian berikut yang berjudul Perhitungan Kemiripan Dokumen Bahasa Indonesia Menggunakan Metode Cosine Similarity (Studi Kasus : Abstrak Tugas Akhir Fakultas Informatika Universitas Telkom) menggunakan metode cosine similarity untuk menghitung nilai kemiripan antar dokumen teks. Untuk menghitung banyaknya term yang muncul pada sebuah dokumen atau yang dikenal sebagai bobot term, Irfan Pahlevi dkk menggunakan pembobotan Term Frequency-Inverse Document Frequency (TF-IDF). Selain itu, penelitian ini juga menggunakan Algoritma Nazief & Andriani untuk mengubah kata menjadi bentuk

kata dasar. Dalam penelitian ini didapatkan beberapa tahap pengumpulan data yang diambil diantaranya dari 341 dokument. Didapatkan bahwa hasil penelitian yang diambil secara manual dan dapat dihitung oleh sistem kemudian akan dihasilkan grafik perbandingan. Perhitungan nilai ini antara korelasi nilai -1 sampai 1. Dan dapat dijelaskan ketika mendekati nilai 1 maka nilai tersebut bersifat linier positif dan apabila ketika mendekati nilai -1 maka nilai itu bersifat negatif. Dari perhitungan tersebut didapatkan dengan hasil korelasi ialah 0.5729. Untuk itu perhitungan TF-IDF dan cosine similarity memiliki nilai positif jika mendekati nilai 1. Hal ini membuktikan kalau dengan aplikasi yang dibangun oleh peneliti ini adalah sudah baik (Pahlevi, Bijaksana, and Tech n.d.).

Penelitian yang terakhir dengan judul Perancangan dan Pembuatan Aplikasi Pencarian Informasi Beasiswa dengan menggunakan cosine similarity. Dengan menggunakan metode cosine similarity guna mendapatkan data beasiswa yang tepat ketika data dibutuhkan oleh user. Didapatkan metode IR (Informasi Retrieval) dalam penelitian ini guna mendapatkan dokument pada database. Penggunaan metode IR (Informasi Retrieval) diharapkan pada saat pencarian dokument, tingkat kemiripan terhadap dokument tersebut sangat akurat oleh karena itu pencarian informasi beasiswa dapat dihitung dengan menggunakan metode cosine similarity guna menghitung nilai kemiripan suatu dokument. Tingkat kemiripan suatu dokument memungkinkan perbandingan dokument yang masukkan berdasarkan query (Kurniawan, Solihin, and Hastarita, 2014).

Pada tabel 2.1 dijelaskan tentang review tugas akhir, pada judul perhitungan kemiripan dokumen bahasa Indonesia dengan menggunakan metode cosine similarity yaitu membandingkan perhitungan kemiripan secara manual dan selanjutnya akan dicari nilai dari korelasi antar dua nilai secara manual maupun dengan sistem. Dinyatakan korelasi linier positif apabila berada pada *range* mendekati 1 dan korelasi linier negatif apabila mendekati -1. Dan menggunakan inputan 341 abstrak tugas akhir mahasiswa fakultas informatika universitas Telkom dengan outputnya nilai korelasi person.

Untuk judul Implementasi metode Cosine Similarity dan Algoritma Smith-Waterman untuk mendeteksi kemiripan teks dengan menggunakan cosine similarity. Proses awal dimulai dengan memasukkan teks asli kemudian dilakukan tahap preprocessing dimana hasilnya adalah teks tunggal yang telah menjadi kata dasar. Teks hasil

preprocessing akan diproses menggunakan Cosine Similarity dan Algoritma Smith-Waterman dengan inputan 4 teks yang berbeda dan mempunyai output Nilai kemiripan teks berdasarkan struktur teks dan nilai kemiripan teks berdasarkan urutan kata.

Untuk judul Mengukur tingkat kesamaan paragraf menggunakan *Vector Space Model* untuk mendeteksi Plagiarisme dengan metode *Vector Space Model*. untuk hasil yang di dapat yaitu Terdapat dua tahap dalam membangun prototipe. Tahap pertama yaitu mengubah format dokumen dan membangun indeks. Tahap kedua yaitu menghitung nilai similaritas antar paragraf. Mempunyai input 15 query dengan output adalah Nilai tingkat kemiripan antar *quer*.

Untuk judul Uncovering highly obfuscated plagiarism cases using fuzzy semantic-based similarity model dengan menggunakan metode *Fuzzy semantic-based similarity model*. Mengadopsi pendekatan deteksi berdasarkan kalimat yang paling banyak muncul. Dalam pendekatan, kalimat dibagi menjadi 4 karakter yang unik dan frekuensi masing-masing karakter dihitung. Dengan inputan 2 teks yang mempunyai output nilai kemiripan antar dokumen.

Untuk judul An efficient and scalable plagiarism checking system using Bloom Filters dengan menggunakan metode *Bloom Filters*. Terdapat dua tahap, yaitu tahap menyimpan dan tahap estimasi. Tahap menyimpan dilakukan dengan memarsing setiap dokumen terpisah untuk menghasilkan substring dengan cara *preprocessing (tokenizing, stopwords removal, dan stemming)*. Tahap estimasi yaitu tahap dimana dokumen akan melewati satuan CU dan menghasilkan set data yang akan dimasukkan dengan inputan dokumen yang ada pada database dan mempunyai output adalah Nilai kemiripan dan waktu yang dibutuhkan dalam menentukan kemiripan.

Tabel 2.1 Review Judul Tugas Akhir/ Skripsi

Judul	Metode	Review		
		Desain dan Analisis	Input	Output
Perhitungan kemiripan dokumen bahasa Indonesia	<i>Cosine Similarity</i>	Membandingkan perhitungan kemiripan secara manual dan selanjutnya dicari nilai korelasi antar dua nilai baik secara manual maupun sistem.	341 abstrak tugas akhir mahasiswa Fakultas Informatika Universitas Telkom	Nilai korelasi Pearson

		Dinyatakan korelasi linier positif apabila berada pada <i>range</i> mendekati 1 dan korelasi linier negatif apabila mendekati -1		
Implementasi metode <i>Cosine Similarity</i> dan Algoritma <i>Smith-Waterman</i> untuk mendeteksi kemiripan teks	<i>Cosine Similarity</i>	Proses awal dimulai dengan memasukkan teks asli kemudian dilakukan tahap <i>preprocessing</i> dimana hasilnya adalah teks tunggal yang telah menjadi kata dasar. Teks hasil <i>preprocessing</i> akan diproses menggunakan <i>Cosine Similarity</i> dan Algoritma <i>Smith-Waterman</i> .	4 teks dari 4 dokumen berbeda	Nilai kemiripan teks berdasarkan struktur teks dan nilai kemiripan teks berdasarkan urutan kata
Mengukur tingkat kesamaan paragraf menggunakan <i>Vector Space Model</i> untuk mendeteksi Plagiarisme	<i>Vector Space Model</i>	Terdapat dua tahap dalam membangun prototipe. Tahap pertama yaitu mengubah format dokumen dan membangun indeks. Tahap kedua yaitu menghitung nilai similaritas antar paragraf	15 <i>query</i>	Nilai tingkat kemiripan antar <i>query</i>
Uncovering highly obfuscated plagiarism cases using fuzzy semantic-based similarity model	<i>Fuzzy semantic-based similarity model</i>	Mengadopsi pendekatan deteksi berdasarkan kalimat yang paling banyak muncul. Dalam pendekatan, kalimat dibagi menjadi 4 karakter yang unik dan frekuensi masing-masing karakter dihitung.	2 teks	Nilai kemiripan kedua kalimat
An efficient and scalable plagiarism checking system using Bloom Filters	<i>Bloom Filters</i>	Terdapat dua tahap, yaitu tahap menyimpan dan tahap estimasi. Tahap menyimpan dilakukan dengan memarsing setiap dokumen terpisah untuk	Dokumen	Nilai kemiripan dan waktu yang dibutuhkan dalam menentukan kemiripan

		menghasilkan substring dengan cara <i>preprocessing</i> (<i>tokenizing</i> , <i>stopwords removal</i> , dan <i>stemming</i>). Tahap estimasi yaitu tahap dimana dokumen akan melewati satuan CU dan menghasilkan set data yang akan dimasukkan kedalam proses Bloom Filter		
--	--	--	--	--

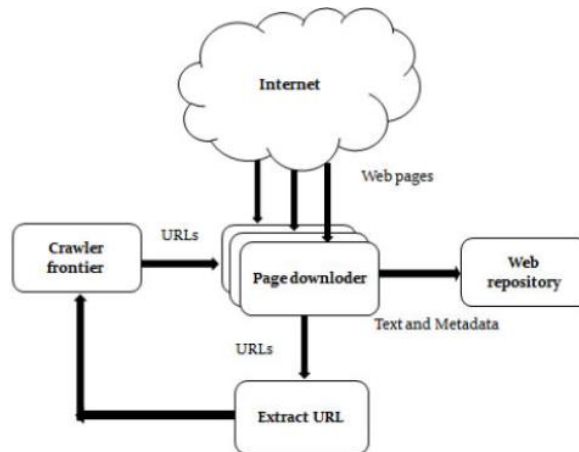
2.2 Landasan Teori

Pada bab ini dijelaskan tentang teori dasar yang berkaitan dengan tugas akhir untuk dapat memecahkan suatu masalah meliputi antara lain web *crawler*, *Term Frequency-Invers Document Frequency* (TF-IDF), dan *cosine similarity*.

2.2.1 Web Crawler

Web Crawler adalah suatu program aplikasi yang digunakan untuk menelusuri halaman web dengan mengikuti hyperlink yang telah disediakan sebelumnya. Penelitian yang dilakukan oleh Kumar dkk mengatakan bahwa web crawler lebih efisien dari pada crawler BF (Kumar et al, 2018). Untuk menemukan pada website cukup dengan memasukkan kata kunci yang akan diketahui judul, deskripsi, isi dan lain-lain. Web crawler hanya bisa menjalankan link hypertext guna menemukan rute yang terbaik untuk menjelajahi suatu halaman website (Rungsawang and Angkawattanawit, 2005).

Untuk saat ini *Web Crawler* yang paling banyak digunakan pada mesin pencari guna mengumpulkan data pada website. Terdapat tiga komponen utama *web crawler* adalah *frontier*, *page downloader*, dan *web repository* (Haiyan, 2017). Berikut ini adalah rancangan dari *web crawler*.



gambar 2. 1 Komponen *Web Crawler*

Berikut ini adalah proses dasar dari ketiga komponen utama *web crawler* (Haiyan 2017):

1. *Crawler Frontier* adalah ketika mendaftar pada *URL* yang belum dikunjungi kemudian diidentifikasi pada *hyperlink* dari halaman web tersebut.
2. *Page Downloader* adalah mendownload halaman website sesuai dengan *URL* yang dapat diterima dari *crawler frontier*.
3. *Web Repository* dadalah Penyimpanan data pada halaman web dengan standar halaman HTML.

2.2.2 *Term Frequency-Invers Document Frequency (TF-IDF)*

Proses pembobotan kata pada dasarnya dilakukan dengan menghitung frekuensi kemunculan *kata* dalam dokumen. Frekuensi *kata* merupakan petunjuk yang mewakili isi dokumen. Semakin banyak kemunculan maka nilai kesesuaian juga semakin besar. Selain frekuensi *kata*, munculan kata dalam dokumen (*term scarcity*) juga harus diperhatikan karena dipandang sebagai kata yang lebih penting dari pada kata yang muncul di banyak dokumen. Pembobotan akan memperhitungkan frekuensi istilah *Invers Document Frequency* pada dokumen yang mengandung suatu kata (Karmayasa and Mahendra 2010). Untuk memperoleh perhitungan bobot (w) dengan menggunakan persamaan *tf-idf* dapat di peroleh rumus :

$$w = tf \times idf$$

Keterangan :

W = bobot dari suatu kata dalam satu dokumen

Tf = frekuensi kemunculan *kata* dalam dokumen

Idf = *Invers Document Frequency*, dimana rumusnya

$$Idf = \log\left(\frac{N}{n}\right)$$

Keterangan :

N = jumlah semua dokumen

n = jumlah dokumen yang mengandung *term*.

Dalam hal ini IDF adalah kata yang terdapat pada keseluruhan dokumen, yang memperlihatkan perbedaan antara suatu kata pada setiap dokumen. TF adalah suatu kata dalam sebuah dokumen. TF-IDF adalah metode yang sangat cocok digunakan untuk perhitungan bobot suatu dokument yang mempunyai nilai kesamaan.

2.2.3 *Cosine Similarity*

Untuk menemukan website seperti k-similarity digunakan teknik dasar yang memiliki jumlah data yang sangat banyak dan disimpan kedalam database yang besar. Oleh sebab itu algoritma yang tepat guna melakukan proses kemiripan terhadap suatu dokument dalam waktu yang singkat dibutuhkan algoritma seperti cosine similarity. Alewiwi dkk melakukan penelitian dimana fitur utama yang akan digunakan untuk menemukan suatu dokument dibedakan menjadi 3 fase yaitu fase deteksi duplikat, fase istilah, dan fase gabungan (Alewiwi, Orencik, and Savaş 2016).

Cosine Similarity adalah metode untuk menghitung kemiripan suatu dokument. Perhitungan dengan metode cosine similarity didasarkan pada vector space similarity measure. Cosine similarity dapat menghitung kemiripan antar objek dengan menggunakan kata kunci dari sebuah dokument tersebut (Nurdiana, Jumadi, and Nursantika, 2016).

Metode cosine similarity dapat digunakan untuk menghitung suatu sudut vektor dan mengukur kemiripan antar dokument dengan kesamaan antara vektor query dan vektor dokument yang menghasilkan nilai sudut cosinus x antar dua vektor. Nilai suatu sudut cosinus yang menentukan kesamaan dua vektor mengandung nilai terkecil 0 dan nila terbsar 1. Nilai 0 menandakan bahwa dokument yang dibandingkan tidak adanya kemiripan sama sekali dan apabila

mendekati nilai 1 maka dokument itu mempunyai nilai kemiripan yang cukup besar (Pahlevi, Bijaksana, and Tech n.d.).

metode *cosine similarity* dalam setiap dokumen yang dibandingkan akan membentuk sebuah segitiga, sehingga dapat diterapkan hukum *cosinus* untuk menyatakan bahwa (Imbar et al, 2014):

$$\cos (C) = a^2 + b^2 + c^2 / 2ab$$

dimana

$$a^2 = a_x^2 + a_y^2, b^2 = b_x^2 + b_y^2$$

dan

$$c^2 = (b_x - a_x)^2 + (a_y - b_y)^2$$

Pengantian untuk nilai a , b , dan c sehingga didapatkan :

$$\cos C = \frac{a_x b_x + a_y b_y}{\sqrt{a_x^2 + a_y^2} \times \sqrt{b_x^2 + b_y^2}}$$

Dokumen dikatakan identik apabila sudutnya adalah nol derajat (0°) dan kesamaannya satu (1), dan apabila dokumen itu dikatakan tidak identik mempunyai sudut 90° dan kesamaan adalah nol (0).

2.3 PHP (*Hypertext Preprocessor*)

PHP diciptakan oleh Rasmus Lerdorf, seorang software engineer asal Greenlan pada tahun 1995. Pada awalnya PHP digunakan Rasmus sebagai pencatat jumlah pengunjung pada website pribadinya. Tetapi karena perkembangan PHP disukai oleh komunitas nya, maka beliau pun merilis bahasa PHP kepublik dengan lisensibopen-source. Pada saat ini, PHP adalah server-side scripting yang paling banyak digunakan pada website di seluruh dunia, saat ini sudah mencapai versi 5 dan statistiknya terus bertambah (Kadir, 2009).

PHP (*Hypertext Preprocessor*), adalah bahasa pemrograman yang bersifat serverside, artinya bahasanya berbentuk script yang disimpan dan dijalankan di komputer server (WebServer) sedangkan hasilnya dikirimkan ke komputer client (WebBrowser) dalam bentuk script HTML (*Hypertext Mark up Language*) (Hakim, 2008).

Sedangkan HTML hanya dapat menampilkan konten statis dan untuk PHP sendiri bisa berinteraksi dengan database, file dan folder sehingga dapat membuat PHP bisa menampilkan konten yang dinamis dari sebuah website. Website Social Networking adalah contoh aplikasi web yang dapat dibuat menggunakan PHP dan termasuk dalam bahasa cross-platform, yang artinya PHP dapat dijalankan pada sistem operasi yang berbeda-beda seperti Windows, Linux, dan Mac. Program PHP ditulis dalam bentuk file plain text (teks biasa) yang mempunyai akhiran “.php”.

2.3.1 Karakteristik Script PHP

- File PHP dapat disimpan dengan ekstensi ialah : *.php3, *.php4, *.php
- Script PHP bisa ditulis atau diapit diantara tag (<?php dan ?> , <script language='php'> dan </script> , <? dan ?> , <% dan %>) , tetapi tanda yang resmi dan paling banyak digunakan adalah yang pertama, yaitu <?php dan ?>
- File PHP dapat menginduk atau disisipkan pada bahasa script lainnya atau dapat berdiri sendiri.

2.3.2 Kemampuan PHP

PHP sendiri mempunyai kemampuan untuk membuat website menjadi dinamis dengan kata lain dapat berinteraksi dengan pengunjung website. Contohnya adalah shopping cart yang dimiliki oleh kebanyakan website yang menjual barang atau biasa disebut dengan e-commerce. Untuk yang pernah membeli barang di internet pastilah pernah menggunakan shopping cart (Peranginangin, 2006).

Proses yang dilakukan ketika user berbelanja menggunakan shopping cart ialah sebagai berikut :

- ketika user memilih suatu barang maka secara otomatis shopping cart akan memasukkan data barang yang dipilih.
- Pada proses berikutnya akan menambahkan harga barang yang dipilih oleh user dan untuk proses pengurangan harga barang akan dilakukan jika user melakukan penghapusan data barang yang telah dipilih sebelumnya.
- Proses inibakan terus menerus dilakukan oleh user ketika memilih barang yang akan dibeli dan untuk mengakhiri proses ini dapat dilakukan dengan mengklik tombol checkout.

- Ketika user mengklik tombol checkout pada shopping cart maka proses selanjutnya yaitu menghitung total biaya yang harus dibayarkan oleh user.

Keuntungan lainnya ketika menggunakan bahasa pemrograman PHP adalah program yang telah dibuat dapat berjalan pada banyak sistem operasi, contoh yaitu sistem operasi Microsoft Windows, Linux, Mac OS, dan UNIX.

2.4 MySQL

MySQL adalah perangkat lunak atau software pada sistem manajemen basisdata SQL atau DBMS Multithread dan multi user. MySQL sendiri merupakan turunan dari salah satu konsep utama pada database. MySQL ditemukan oleh Michael "Monty" Widenius pada tahun 1979, seorang programmer komputer yang berasal dari Swedia dan beliau mengembangkan sebuah sistem database sederhana yang disebut UNIREG dan menggunakan koneksi low-level ISAM database engine dengan indexing (Suja, 2005).

2.5 System Development Life Cycle (SDLC)

2.5.1 Definisi SDLC

Dimulai fase awal perencanaan dan berakhir dengan menghasilkan produk yang diinginkan. Sistem analis ditugaskan untuk memecahkan masalah bisnis. Berkaitan dengan kegiatan pemecahan masalah, perlu diatur dan difokuskan untuk menghasilkan tujuan. Seorang analis untuk mencapai hasil ini dengan cara mengorganisir suatu proyek, sehingga akhirnya akan menghasilkan suatu sistem informasi yang dikembangkan melalui fase-fase pengembangan. Menurut Satzinger, Jackson, & Burd (2010, p38), Systems Development Life Cycle (SDLC) adalah seluruh proses ruang lingkup sistem yang dimulai pada tahap membangun (building), menyebarkan (deploying), menggunakan (using), dan memperbarui (updating) sistem informasi.

System Development Life Cycle (SDLC) merupakan salah satu metodologi yang digunakan untuk mengembangkan sistem informasi. Menurut Azhar Susanto (2004, p341), System Development Life Cycle adalah salah satu metode pengembangan sistem informasi yang populer pada saat sistem informasi pertama kali dikembangkan.

Menurut Kendall & Kendall (2006), System Development Life Cycle (SDLC) adalah pendekatan yang dilakukan secara bertahap dalam hal melakukan analisa dan membangun rancangan sistem dengan menggunakan siklus-siklus secara spesifik terhadap kegiatan penggunaannya.

Sehingga berdasarkan definisi yang telah diuraikan diatas, dapat dijelaskan bahwa System Development Life Cycle merupakan metode yang dilakukan oleh analis dan programmer dalam membangun sistem informasi melalui beberapa fase bertahap mulai dari perencanaan sampai dengan implementasi. SDLC merupakan salah satu kunci konsep dasar dalam sistem informasi.

2.5.2 Pendekatan dan Fase-Fase dalam SDLC

Dalam lingkungan pengembangan saat ini, banyak pendekatan yang digunakan dalam pengembangan sistem informasi melalui SDLC. Menurut Satzinger, Jackson, & Burd (2010, p39), dalam dunia dimana teknologi informasi berjalan seiring dengan perubahan perkembangannya, ada dua pendekatan SDLC dengan menggunakan perspektif yang berbeda, yaitu :

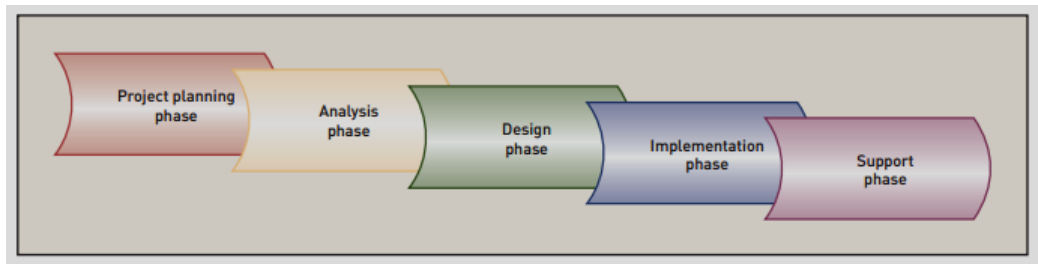
2.5.2.1 Pendekatan Prediktif (*Predictive Approach*)

Pendekatan Prediktif adalah pendekatan SDLC yang mengatakan bahwa pembangunan suatu proyek dikatakan terencana apabila diatur terlebih dahulu dengan sistem informasi yang dapat dikembangkan sesuai rencana. Metode ini digunakan untuk mengembangkan sistem yang baik dan mudah untuk dipahami.

2.5.2.2 Pendekatan Adaptif (*Adaptive Approach*)

Pendekatan Adaptif digunakan ketika persyaratan sistem atau kebutuhan pengguna tidak dipahami dengan baik. Dalam situasi ini, proyek tidak dapat direncanakan sepenuhnya pada awal. Beberapa persyaratan sistem mungkin belum perlu ditentukan setelah beberapa pekerjaan mulai dilakukan.

SDLC memiliki beberapa fase yang dibutuhkan dengan serangkaian aktivitas mulai dari fase awal hingga fase akhir. Pada gambar 2.2 fase-fase dalam SDLC memiliki 5 (lima) fase utama yaitu fase perencanaan (*Project Planning*), fase analisis (*Analysis*), fase design (*Design*), fase implementasi (*Implementation*) dan fase dukungan (*Support*).



Gambar 2.2 Fase-Fase dalam SDLC
 Sumber : Satzinger, Jackson, & Burd, (2010, p40)

Pada tabel 2.2 Fase dan tugas dalam SDLC dapat dijelaskan pada fase *Project Planning* dan mempunyai tugas Mengidentifikasi ruang lingkup pada sistem baru, memastikan proyek ini layak dan dapat mengembangkan rencana sumber daya dan anggaran. Kemudian pada fase *Analys* mempunyai tugas Memahami dan merinci kebutuhan serta persyaratan untuk mengolah sistem baru. Untuk fase *Design* mempunyai tugas Merancang sistem yang menghasilkan solusi berdasarkan yang ditetapkan sebagai keputusan yang dibuat selama analisis. Merancang sistem yang menghasilkan solusi berdasarkan yang ditetapkan sebagai keputusan yang dibuat selama analisis. Selanjutnya fase *Implementation* mempunyai tugas Membangun, menguji sistem informasi yang handal dengan pengguna yang siap untuk mendapatkan keuntungan seperti yang diharapkan dari penggunaa sistem ini. Yang terakhir fase *Support* mempunyai tugas Menjaga sistem agar mampun berjalan secara baik pada awal dan selama bertahun-tahun.

Tabel 2.2 Fase dan Tugas dalam SDLC
 Sumber : Satzinger, Jackson, & Burd, (2010, p40)

Fase	Tugas
<i>Project Planning</i>	Mengidentifikasi ruang lingkup pada sistem baru, memastikan proyek ini layak dan dapat mengembangkan rencana sumber daya dan anggaran.
<i>Analys</i>	Memahami dan merinci kebutuhan serta persyaratan untuk mengolah sistem baru.
<i>Design</i>	Merancang sistem yang menghasilkan solusi berdasarkan yang ditetapkan sebagai keputusan yang dibuat selama analisis.

<i>Implementation</i>	Membangun, menguji sistem informasi yang handal dengan pengguna yang siap untuk mendapatkan keuntungan seperti yang diharapkan dari penggunaa sistem ini.
<i>Support</i>	Menjaga sistem agar mampun berjalan secara baik pada awal dan selama bertahun-tahun.

