

SKRIPSI

**SISTEM ONLINE PEER REVIEW
PENILAIAN KARYA ILMIAH DOSEN**

**Disusun dan Diajukan oleh
ASTUTI MAYANGSARI
D42116003**



**DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR**

2021

**LEMBAR PENGESAHAN SKRIPSI
SISTEM ONLINE PEER REVIEW
PENILAIAN KARYA ILMIAH DOSEN**

Disusun dan diajukan oleh

ASTUTI MAYANGSARI


D42116003


Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin pada tanggal 24 November 2021 dan dinyatakan telah memenuhi syarat kelulusan.

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,


Dr. Amil Ahmad Ilham, S.T., M.IT.
Nip. 19731010 199802 1 002


Iqra Aswad, ST., MT.
Nip. 19901128 201904 3 001



Dr. Amil Ahmad Ilham, S.T., M.IT.
Nip. 19731010 199802 1 002

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini :

Name : Astuti Mayangsari
NIM : D42116003
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa hasil karya tulisan saya berjudul :

Sistem Online Peer Review Penilaian Karya Ilmiah Dosen

Adalah karya tulisan saya sendiri. Bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi ini yang saya tulis ini benar benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain. Maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 24 November 2021

Yang menyatakan,



ASTUTI MAYANGSARI

ABSTRAK

Proses penilaian karya ilmiah Dosen oleh peer review sebagai pengajuan kenaikan pangkat Dosen di Fakultas Teknik Universitas Hasanuddin selama ini dilakukan secara manual atau belum terkomputerisasi sehingga membutuhkan waktu yang lama dalam memproses dokumen-dokumen karya ilmiah Dosen. Pembuatan sistem online peer review untuk penilaian karya ilmiah Dosen ini diharapkan dapat mengatasi permasalahan tersebut. Pada sistem *online* ini diterapkan beberapa proses yang akan menunjang proses penilaian karya ilmiah, seperti adanya proses penginputan data kelengkapan karya ilmiah, proses pengajuan karya ilmiah, verifikasi kelayakan penilaian karya ilmiah, proses pembagian karya ilmiah kepada para tim penilai karya ilmiah dan proses penilaian karya ilmiah. Sistem *online* ini berbasis website dan diharapkan dapat mempermudah *peer review* dalam proses penilaian karya ilmiah Dosen, membantu admin dalam pembagian tugas untuk menilai tiap karya ilmiah dosen yang telah diajukan kepada tim penilai karya ilmiah sehingga dapat mempercepat proses penilaian.

Kata kunci: Dosen, Penilaian Karya Ilmiah, *Peer Review*, *Website*

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji dan syukur senantiasa penulis panjatkan atas kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini yang berjudul “**Sistem Online Peer Review Penilaian Karya Ilmiah Dosen**” sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Dalam proses pembuatan laporan akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai penyusunan Tugas Akhir. Oleh karena itu, dengan segala kerendahan hati, penulis mengucapkan rasa terima kasih sebesar-besarnya kepada :

1. Tuhan Yang Maha Esa yang melalui berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan tugas akhir ini.
2. Kedua orang tua penulis, Bapak Asmar dan Ibu Sri Puri Haryanti yang selalu memberikan doa, semangat serta sabar dalam mendidik penulis sejak kecil.
3. Bapak Dr. Amil Ahmad Ilham, S.T., M.IT, selaku pembimbing I dan bapak Iqra Aswad, S.T., M.T, selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian untuk mengarahkan penulis dalam penyusunan tugas akhir.
4. Segenap Dosen dan Staff Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin
5. Kak Muhammad Aryandi, yang merupakan *support system* sekaligus kakak

senior dari angkatan 2014 Teknik Informatika yang selalu ada mendukung, membantu dan juga ikut membimbing penulis selama perkuliahan sampe penyelesaian tugas akhir hingga saat ini.

6. Sahabat-sahabat penulis semasa SMP hingga saat ini masih selalu ada, Mar'Atul Wildani S., Sintia Tirtasari dan Tiara Shinta Istiqamah yang selalu ada memberi dukungan kepada penulis

7. Sahabat-sahabat penulis semasa kuliah, Asri Oktianawati dan Nishrina Nurul Amirah yang selalu ada untuk mendengarkan keluh kesah penulis, memberi dukungan, memberi solusi dalam setiap masalah selama perkuliahan dan mengarahkan penulis untuk terus menjadi baik selama perkuliahan hingga saat ini.

8. Teman-teman khusus penulis semasa kuliah, Dhinda Fitri W, Putri Angriani, Cici Purnamasari, Andi Marimar M, Sitti Nurfadillah, Tuti Amalia, Ayu Lestari R, Malyana Ariani, Rya Dita Purnama, Fabyola Larasati M, Tedi Setiady dan Muhammad Musyawir yang ikut serta selalu memberi dukungan, menemani, memberi solusi dan membantu semasa akhir-akhir perkuliahan.

9. Teman-teman Igniter 16 atas dukungan dan semangat yang diberikan.

10. Orang-orang berpengaruh lainnya yang tidak sempat penulis sebutkan satu-persatu yang telah memberi dukungan, semangat serta doanya selama penyusunan tugas akhir ini.

Makassar, November 2021

Penulis

DAFTAR ISI

ABSTRAK	i
DAFTAR ISI.....	iv
DAFTAR GAMBAR	
.....	vii
i	
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian	2
1.4. Manfaat Penelitian	2
1.5. Batasan Masalah.....	3
1.6. Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA.....	5
2.1. <i>Website</i>	5
2.1.1. Kategori <i>Website</i>	5
2.2. Bahasa Pemrograman.....	7
2.3. HTML	7
2.4. PHP	8
2.4.1 Kelebihan PHP	9
2.5. <i>Javascript</i>	10

2.6. Codeigniter.....	11
2.6.1 MVC (<i>Model View Controller</i>).....	11
2.7. Basis Data	13
2.7.1. Sistem Basis Data.....	14
2.7.2. MySQL.....	15
2.8. Xampp.....	16
2.9. <i>Visual Studio Code</i>	16
2.10. API.....	18
2.11. Penelitian Terkait	18
BAB III METODOLOGI PENELITIAN.....	20
3.1. Studi Literatur	20
3.2. Persiapan Kebutuhan Penelitian.....	21
3.2.1. Persiapan Kebutuhan Data	21
3.2.2. Persiapan Kebutuhan Perangkat	22
3.3. Perancangan Sistem	23
3.3.1. Perancangan <i>Use Case Diagram</i>	26
3.3.1.1. Identifikasi Aktor	26
3.3.1.2. Identifikasi Diagram <i>Use Case</i>	27
3.3.1.3. <i>Use Case Diagram</i>	28
3.3.1.4. <i>Use Case</i> Skenario	29
3.3.1.4.1. <i>Use Case</i> Skenario Input Karya Ilmiah	30
3.3.1.4.2. <i>Use Case</i> Mengajukan Karya Ilmiah.....	32
3.3.1.4.3. <i>Use Case</i> Lihat Hasil Penilaian	34

3.3.1.4.4. <i>Use Case</i> Lihat Karya Ilmiah	34
3.3.1.4.5. <i>Use Case</i> Verifikasi Karya Ilmiah.....	35
3.3.1.4.6. <i>Use Case</i> Bagi Tugas Penilaian.....	36
3.3.1.4.7. <i>Use Case</i> Lihat Hasil Penilaian	37
3.3.1.4.8. <i>Use Case</i> Lihat Tugas Penilaian.....	38
3.3.1.4.9. <i>Use Case</i> Penilaian Karya Ilmiah.....	39
3.3.1.4.10. <i>Use Case</i> Login	40
3.3.2. Perancangan <i>Activity Diagram</i>	41
3.3.2.1. <i>Activity Diagram</i> Login.....	41
3.3.2.2. <i>Activity Diagram</i> Input Karya Ilmiah.....	42
3.3.2.3. <i>Activity Diagram</i> Mengajukan Karya Ilmiah	45
3.3.2.4. <i>Activity Diagram</i> Verifikasi Karya Ilmiah	48
3.3.2.5. <i>Activity Diagram</i> Bagi Tugas Penilaian	50
3.3.2.6. <i>Activity Diagram</i> Penilaian Karya Ilmiah	52
3.3.2.7. <i>Activity Diagram</i> Lewat Batas Penilaian	54
3.3.3. Perancangan Database	55
3.3.3.1. Tabel Admin.....	55
3.3.3.2. Tabel Karya Ilmiah.....	56
3.3.3.3. Tabel Penilaian Karya Ilmiah.....	58
3.3.3.4. Tabel Dosen.....	59
3.3.3.5. Tabel Tim Penilai	59
3.3.3.6. Diagram ERD	59
3.4. Pembuatan Sistem	61

3.5. Uji Coba Sistem	61
BAB IV HASIL DAN PEMBAHASAN	63
4.1. Hasil Pembuatan Sistem.....	63
4.1.1. Halaman <i>Login</i>	63
4.1.2. Halaman Dosen	63
4.1.2.1. Halaman Utama Dosen.....	63
4.1.2.2. Halaman Menu Karya Ilmiah Import Data Dari Turnitin... 64	
4.1.2.3. Halaman Menu Karya Ilmiah Input Data Baru.....	68
4.1.3. Halaman Admin	71
4.1.3.1. Halaman Utama Admin.....	71
4.1.3.2. Halaman Menu Dosen	71
4.1.4. Halaman Tim Penilai.....	74
4.1.4.1. Halaman Utama Tim Penilai.....	74
4.1.4.2. Halaman Menu Penilaian Karya Ilmiah.....	76
4.3. Hasil Uji Coba Sistem.....	79
4.3.1. Hasil Uji Coba Halaman Login	79
4.3.2. Hasil Uji Coba Halaman Dosen	80
4.3.3. Hasil Uji Coba Halaman Admin	83
4.3.4. Hasil Uji Coba Halaman Tim Penilai.....	85
4.4. Hasil Analisis <i>Usability System</i>	87
BAB V PENUTUP.....	88
5.1. Kesimpulan	88
5.2. Saran.....	88

DAFTAR PUSTAKA	89
LAMPIRAN.....	92

DAFTAR GAMBAR

Gambar 2.1. Arsitektur MVC	13
Gambar 2.2. Konsep Basis Data.....	15
Gambar 3.1. Diagram Tahapan Penelitian.....	20
Gambar 3.2. Gambaran Umum Sistem.....	24
Gambar 3.3. Diagram Venn Tim Penilai.....	25
Gambar 3.4. Alur Sistem	29
Gambar 3.5. <i>Activity Diagram</i> Login	41
Gambar 3.6. <i>Activity Diagram</i> Input Karya Ilmiah Import Data Dari Turnitin ..	42
Gambar 3.7. <i>Activity Diagram</i> Inpur Karya Ilmiah Input Data Baru	44
Gambar 3.8. <i>Activity Diagram</i> Mengajukan Karya Ilmiah Import Data Dari Turnitin.....	46
Gambar 3.9. <i>Activity Diagram</i> Mengajukan Karya Ilmiah Input Data Baru.....	47
Gambar 3.10. <i>Activity Diagram</i> Verifikasi Karya Ilmiah	49

Gambar 3.11. <i>Activity Diagram</i> Bagi Tugas Penilaian	51
Gambar 3.12. <i>Activity Diagram</i> Penilaian Karya Ilmiah	52
Gambar 3.13. <i>Activity Diagram</i> Lewat Batas <i>Deadline</i>	54
Gambar 3.14. <i>Diagram</i> ERD Sistem	60
Gambar 4.1. Halaman Login	63
Gambar 4.2. Tampilan Menu Utama Dosen	64
Gambar 4.3. Tampilan Menu Karya Ilmiah Import Data Dari Turnitin	64
Gambar 4.4. Tampilan Halaman Aksi Detail Sebelum Data Lengkap	65
Gambar 4.5. Tampilan Halaman Aksi Detail Setelah Data Lengkap	66
Gambar 4.6. Tampilan Aksi <i>Input</i> Data	66
Gambar 4.7. Tampilan Form Aksi <i>Input</i> Data	67
Gambar 4.8. Tampilan Aksi <i>Edit</i>	67
Gambar 4.9. Tampilan Menu Karya Ilmiah <i>Input</i> Data Baru	68
Gambar 4.10. Tampilan Tambah Karya Ilmiah Sebelum Menentukan Jenis Karya Ilmiah	68

Gambar 4.11. Tampilan Tambah Karya Ilmiah Setelah Menentukan Jenis Karya Ilmiah	69
Gambar 4.12. Tampilan Setelah Tambah Karya Ilmiah.....	69
Gambar 4.13. Tampilan Halaman Utama Admin.....	71
Gambar 4.14. Tampilan Halaman Menu Dosen	71
Gambar 4.15. Tampilan Halaman Menu Detail Dosen	72
Gambar 4.16. Tampilan Fitur Verifikasi	72
Gambar 4.17. Tampilan Setelah Melakukan Verifikasi	73
Gambar 4.18. Tampilan Fitur Pilih Tim Penilai	73
Gambar 4.19. Halaman Utama Tim Penilai	75
Gambar 4.20. Tampilan Halaman Menu Penilaian Karya Ilmiah	76
Gambar 4.21. Tampilan Halaman Aksi Detail	76
Gambar 4.22. Tampilan Halaman Aksi Nilai	77
Gambar 4.23. Tampilan Halaman Penilaian Karya Ilmiah Setelah Proses Penilaian Selesai.....	78
Gambar 4.24. Tampilan Halaman Hasil	78

DAFTAR TABEL

Tabel 3.1. Identifikasi Aktor	26
Tabel 3.2. Identifikasi Diagram <i>Use Case</i>	27
Tabel 3.3. <i>Use Case</i> Skenario Input Import Data Dari Turnitin	30
Tabel 3.4. <i>Use Case</i> Skenario Input Input Data Baru	31
Tabel 3.5. <i>Use Case</i> Skenario Mengajukan Karya Ilmiah Import Data Dari Turnitin.....	32
Tabel 3.6. <i>Use Case</i> Skenario Mengajukan Karya Ilmiah Input Data Baru.....	33
Tabel 3.7. <i>Use Case</i> Skenario Lihat Hasil Penilaian.....	34
Tabel 3.8. <i>Use Case</i> Skenario Lihat Karya Ilmiah.....	34
Tabel 3.9. <i>Use Case</i> Skenario Verifikasi Karya Ilmiah	35
Tabel 3.10. <i>Use Case</i> Skenario Bagi Tugas Penilaian	36

Tabel 3.11. <i>Use Case</i> Skenario Lihat Hasil Penilaian.....	37
Tabel 3.12. <i>Use Case</i> Skenario Lihat Tugas Penilaian	38
Tabel 3.13. <i>Use Case</i> Skenario Penilaian Karya Ilmiah	39
Tabel 3.14. <i>Use Case</i> Skenario Login	40
Tabel 3.15. Admin.....	55
Tabel 3.16. Karya Ilmiah.....	56
Tabel 3.17. Penilaian Karya Ilmiah.....	58
Tabel 3.18. Dosen.....	59
Tabel 3.19. Tim Penilai	59
Tabel 4.1. Pengujian <i>Black Box</i> Pada Halaman Login.....	79
Tabel 4.2. Pengujian <i>Black Box</i> Pada Menu Karya Ilmiah Turnitin	80
Tabel 4.3. Pengujian <i>Black Box</i> Pada Menu Karya Ilmiah Lokal	81
Tabel 4.4. Pengujian <i>Black Box</i> Pada Halaman Menu Dosen.....	83
Tabel 4.5. Pengujian <i>Black Box</i> Pada Menu Detail Dosen.....	83
Tabel 4.6. Pengujian <i>Black Box</i> Pada Menu Penilaian.....	85

BAB I

PENDAHULUAN

1.1. Latar Belakang

Kenaikan jabatan akademik Dosen merupakan bentuk pemberian penghargaan pemerintah atas prestasi kerja yang dicapai Dosen. Peningkatan jabatan fungsional dapat diperoleh jika Dosen telah melakukan pengajuan angka kredit yang telah mencukupi serta syarat-syarat yang telah ditetapkan. Dalam rangka penjaminan mutu dan peningkatan kualitas Dosen dalam jabatan akademik Lektor Kepala dan Profesor, maka bagi pengusul pada jabatan akademik Lektor Kepala dan Profesor dengan masa kerja minimal sesuai dengan ketentuan Permenpan-RB dipersyaratkan memiliki Karya ilmiah (Ristekdikti, 2019)

Karya ilmiah yang telah dibuat oleh tiap dosen akan di nilai menggunakan format penilaian yang telah disediakan. Namun, saat ini pelaksanaan penilaian Karya ilmiah yang ada terkhusus studi kasus pada Fakultas Teknik Universitas Hasanuddin masih mengalami kesulitan dalam hal penilaian karya ilmiah. Metode pengumpulan dan pencatatan penilaian Karya ilmiah ini masih dilakukan secara manual dan belum terkomputerisasi, hal ini menyebabkan lambatnya proses penilaian karya ilmiah Dosen.

Pesatnya perkembangan teknologi dan informasi saat ini, dapat mengatasi permasalahan yang ada dengan menggunakan sistem *online* untuk melakukan penilaian karya ilmiah Dosen sehingga penilaian dapat dilakukan secara *online* dengan mudah. Sistem ini akan terintegrasi dengan *Apps Turnitin* Universitas Hasanuddin yaitu, sistem yang digunakan sebagai media penyimpanan dokumen-

dokumen karya ilmiah yang telah dicek plagiasinya. Hasil dari penelitian ini adalah sistem *online* yang dapat membantu rekan sejawat untuk melakukan proses penilaian karya ilmiah yang dilengkapi dengan fitur penginputan data karya ilmiah, verifikasi kelayakan penilaian karya ilmiah, pembagian karya ilmiah kepada tim penilai karya ilmiah, dan penginputan nilai karya ilmiah yang akan menghasilkan form penilaian yang dapat diunduh.

1.2. Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah dalam tugas akhir ini adalah :

1. Bagaimana membuat sistem manajemen pengelolaan *reviewer*/penilai karya ilmiah Dosen oleh rekan sejawat (*peer review*) ?
2. Bagaimana membuat sistem penilaian karya ilmiah Dosen secara *online* yang terintegrasi dengan *Apps Turnitin* ?

1.3. Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini adalah :

1. Untuk membuat sistem manajemen pengelolaan *reviewer*/penilai karya ilmiah Dosen oleh rekan sejawat (*peer review*).
2. Untuk membuat sistem penilaian karya ilmiah Dosen secara *online* yang terintegrasi dengan *Apps Turnitin*.

1.4. Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Hasil dari penelitian dapat membantu pihak kampus dalam melakukan proses penilaian karya ilmiah Dosen secara *online*.

2. Menambah pengetahuan dan keterampilan penulis dalam membuat sistem *online* yang terintegrasi dengan *Apps Turnitin* dengan *interface* sistem yang berbasis *website*.

1.5. Batasan Masalah

Batasan masalah diambil dari tugas akhir ini adalah sebagai berikut :

1. Sistem berbasis web ini hanya dapat digunakan dalam penilaian karya ilmiah Dosen serta penyimpanan dokumen atau arsip karya ilmiah yang telah ada.
2. Data karya ilmiah Dosen yang digunakan berasal dari *Apps Turnitin* dan juga ada yang berasal dari hasil inputan Dosen itu sendiri ke dalam sistem yang datanya akan masuk ke *database* sistem.
3. File karya ilmiah yang dapat terinput pada sistem hanya berupa file doc, docx atau pdf.
4. Lingkup penelitian adalah kumpulan Karya ilmiah dari Fakultas Teknik Universitas Hasanuddin.

1.6. Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan ini, maka akan diuraikan dalam beberapa tahapan dari penulisan secara sistematis, yaitu :

BAB I PENDAHULUAN

Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori yang mendasari dan berhubungan dengan penelitian terkait serta yang menunjang percobaan penelitian yang akan dilakukan.

BAB III METODOLOGI PENELITIAN

Dalam bab ini menjelaskan tentang tahapan penelitian mulai dari studi literatur, persiapan kebutuhan perancangan sistem, perancangan dan proses pembuatan sistem dan skenario pengujian sistem.

BAB IV HASIL DAN PEMBAHASAN

Dalam bab ini menjelaskan tentang penyajian data hasil perancangan, proses-proses yang dilakukan terhadap data-data yang telah diperoleh, dan pembahasan atau evaluasi hasil implementasi sistem secara keseluruhan.

BAB V PENUTUP

Bab ini berisi kesimpulan dan penelitian yang telah dibuat, serta saran-saran untuk dapat meningkatkan dan mengembangkan sistem pada masa yang akan datang.

BAB II

TINJAUAN PUSTAKA

2.1. Website

Website adalah sebuah sistem yang berisi beragam informasi baik berupa teks, gambar, audio maupun video dan dapat diakses melalui perangkat yang biasa disebut *web browser*. Web juga merupakan kumpulan informasi baik yang bersifat dinamis atau statis yang terdiri dari halaman yang dibuat (Oktasri & K. Denny, 2019).

Dalam beberapa dekade, *website* telah menjadi lebih besar dan kompleks. Pada awalnya *website* digunakan hanya untuk mempermudah tukar menukar dan melakukan pembaruan informasi kepada sesama peneliti yang dilakukan oleh Sir Tomothy Jon dalam perkembangan *website* yang dapat melakukan manajemen konten (Butkiewicz, 2011).

2.1.1. Kategori Website

Web dapat dibagi menjadi dua kategori, yaitu (Nugroho, 2014):

1. Web Statis

Web Statis adalah web yang *content* atau isinya tidak berubah-ubah. Maksudnya adalah isi dari dokumen yang ada di web tersebut tidak dapat diubah secara mudah. Ini dikarenakan karena *script* yang digunakan untuk membuat web statis tidak mendukung untuk mengubah isi dokumen. *Script* yang digunakan untuk membuat web statis ini seperti HTML dan CSS. Maka dari itu untuk perubahan isi dokumen pada web statis harus mengubah

isi file HTML atau CSS tersebut. Web statis ini biasanya digunakan oleh *website* yang menggunakan HTML, *Web Search Engine* atau *Web Company Profile*.

2. Web Dinamis

Web dinamis adalah web yang isinya dapat berubah-ubah setiap saat. Karena dalam teknologi pembuatan web dinamis sudah dirancang semudah mungkin bagi *user* yang menggunakan web dinamis tersebut. Untuk perubahan *content* atau isi dokumen dalam sebuah web dinamis lebih mudah daripada web statis. Ketika akan mengubah *content* atau isi dari sebuah web dinamis hanya perlu masuk ke bagian administrator web yang telah disediakan oleh *script* web dinamis. Untuk membuat web dinamis diperlukan beberapa komponen yaitu: a. *Client Side Scripting* merupakan jenis *script* yang pengolahannya (penerjemahannya) dilakukan di sisi *client*. Contoh *Client Side Scripting* : HTML (*Hypertext Markup Language*), Javascript, XML (*Extensible Markup Language*), CSS (*Cascading Style Sheet*). *Client Side Scripting* merupakan *script* yang digunakan untuk membuat halaman web statis. b. *Server Side Scripting* merupakan *script* yang pengolahannya di sisi server. Server yang dimaksud disini adalah sebuah komponen yang biasa disebut web server. Contoh *Server Side Scripting* : PHP (*Hypertext Preprocessor*), ASP (*Active Server Pages*), JSP (*Java Server Pages*). *Server Side Scripting* digunakan untuk membuat halaman

web dinamis. Hal ini disebabkan karena *Server Side Scripting* mempunyai kelebihan, diantaranya : mampu berinteraksi dengan banyak perangkat lunak basis data, mampu mengolah sumber daya yang terdapat di sistem operasi dan perangkat keras komputer, mampu dijalankan di semua sistem operasi (*multiplatform*), aman karena *script*-nya diproses di server yang tidak dapat dilihat dari sisi *client*.

2.2. Bahasa Pemrograman

Bahasa pemrograman, atau sering diistilahkan juga dengan bahasa komputer atau bahasa pemrograman komputer, adalah instruksi standar untuk memerintahkan komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi (Saragih R.R, 2018).

2.3. HTML

HTML atau *Hypertext Markup Language* adalah bahasa dasar untuk web scripting yang bersifat *client side* yang memungkinkan untuk menampilkan informasi dalam bentuk teks, grafik, serta multimedia dan juga menghubungkan antartampilan web page atau *hyperlink* (Sovia & Febio, 2011).

Dalam praktiknya, *Hypertext* berwujud sebuah link yang bisa mengantar Anda ke dunia internet yang luas. Untuk membantu si pengakses berpindah dari

satu tempat ke tempat lainnya, dibuatlah semacam dokumen yang nanti akan disebut dengan istilah website. Untuk membuat website, dibutuhkan *Markup*, yaitu Tag (semacam kode) yang mengatur bagaimana website tersebut akan ditampilkan di jendela browser, seperti *layout* dan tampilan-tampilan visual yang biasa kita lihat dalam website. Terakhir, *Language* yang merupakan penunjuk bahwa HTML adalah *script* pemrograman (Enterprise, 2016).

2.4. PHP

PHP atau *Hypertext Preprocessor* adalah bahasa server-side scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. PHP termasuk dalam *open source product*, sehingga *source code* dapat diubah dan didistribusikan secara bebas (Putu Tri Sabdojati, 2016).

PHP bahasa pemrograman yang berfungsi untuk membuat website dinamis. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan database, file dan folder, contohnya Blog, Toko Online, CMS, Forum, dan *Website Social Networking*. PHP adalah bahasa scripting, bukan bahasa tag-based seperti HTML. PHP termasuk bahasa *cross-platform*, ini artinya PHP bisa berjalan di sistem operasi yang berbeda-beda seperti, Windows, Linux, ataupun MAC (Istiono, Hijrah & Sutarya, 2016).

Untuk dapat berjalan, PHP membutuhkan web server, yang bertugas untuk memproses file php dan mengirimkan hasil pemrosesan yang akan ditampilkan di *browser client*. Oleh karena itu, PHP termasuk *server-side scripting* (*script* yang diproses di server). Untuk media penyimpanan datanya, PHP biasa menggunakan MySQL. Untuk menginstall dan mengkonfigurasi ketiga software tersebut

(Apache, PHP, MySQL) agar dapat berjalan dan selalu terhubung, memang cukup sulit. Maka dari itu dibuatlah paket software LAMP, XAMPP, MAMP, WAMP yang tinggal install dalam satu kali instalasi. Dalam satu kali instalasi, sudah mencakup ketiga *software* tersebut dan sudah dikonfigurasi untuk keperluan lingkungan pengembangan aplikasi web.

2.4.1 Kelebihan PHP

Adapun kelebihan bahasa pemrograman PHP dari bahasa pemrograman lain adalah sebagai berikut (Fahrozi, 2018) :

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.5. Javascript

Javascript adalah *script* program berbasis *client* yang dieksekusi oleh *browser* sehingga membuat halaman web bisa melakukan tugas-tugas tambahan yang tidak bisa dilakukan oleh *script* HTML biasa (Ipan ripai, 2017).

Kode pemrograman Javascript dapat disisipkan kedalam HTML pada awalnya, Javascript mulai diperkenalkan di *browser Netscape Navigator 2*. Namun waktu itu namanya bukan *Javascript*, yaitu *Livescript*. Mengingat pada waktu itu teknologi *Java* sedang panas-panasnya atau sedang tren, maka pihak Netscape memutuskan untuk mengganti menjadi *Javascript*, yang sepertinya nama tersebut lebih *marketible* dibandingkan *Livescript*. Selanjutnya pihak Microsoft mulai ikut memfasilitasi web *browser* buatannya, 'Internet Explorer' yang dapat mendukung *Javascript* (Permana, 2016).

Ada dua jenis bagaimana *Javascript* dibuat, pertama *Javascript* ditukis dalam file yang terpisah dengan HTML, kedua *Javascript* ditulis dalam file HTML. *Javascript* yang ditulis diluar HTML disebut eksternal *Javascript* dengan ekstensi *file.js*. Dalam HTML, penulisan *script* diawali dengan sebuah *tag script*. Selanjutnya *script* yang dijalankan harus diletakkan diantara `<script>` dan `</script>` *tag*. `<script>` memiliki beberapa atribut, namun yang terpenting adalah atribut *language* dan *type*. Karena *Javascript* bukan satu-satunya bahasa *scripting*, maka sangatlah perlu untuk memberitahukan kepada *browser* bahwa bahasa *script* yang digunakan adalah *Javascript* dan selanjutnya *browser* akan menjalankan modul *Javascript* untuk memprosesnya.

2.6. Codeigniter

Codeigniter adalah aplikasi *open-source* yang didasarkan pada kerangka kerja PHP dengan model MVC atau biasa disebut *Model View Controller* yang berfungsi membangun sebuah situs web dinamis menggunakan kode PHP. *Codeigniter* juga dapat memudahkan *developer* untuk membuat aplikasi web berbasis PHP, karena *framework* sudah ada jadi tidak perlu menulis semua kode program dari awal (Juanda, Fahri & Hardinata, 2021).

Codeigniter dibuat dengan tujuan memungkinkan pengembangan proyek lebih cepat daripada menulis kode dasar atau kode struktural, dengan menyediakan banyak pustaka yang biasanya digunakan di konstruksi. *Codeigniter* juga memiliki dokumentasi yang sangat lengkap beserta contoh-contoh implementasi kodenya. Sehingga dokumentasi yang lengkap ini menjadi salah satu alasan kuat kenapa banyak orang memilih *Codeigniter* sebagai kerangka kerja pilihan.

2.6.1. MVC (*Model View Controller*)

MVC atau *Model View Controller* adalah pola arsitektur yang digunakan sebagai kerangka kerja untuk mengembangkan aplikasi web. Seperti namanya, arsitektur ini memisahkan aplikasi yang sedang dirancang menjadi tiga komponen utama, yaitu *Model*, *View* dan *Controller* (Tang Elizabeth, 2020).

Konsep MVC sangatlah populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman *Smalltalk*, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti

manipulasi data, user interface dan bagian yang menjadi kontrol aplikasi. Berikut penjelasan dari komponen-komponen tersebut (Istiono, Hijrah & Sutarya, 2016) :

1. *Model*

Model, digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi. Hanya model yang mengandung data dan fungsi yang berhubungan dengan pemrosesan data. Sebuah model meringkas lebih dari sekedar data dan fungsi yang beroperasi di dalamnya. Pendekatan model yang digunakan untuk komputer model atau abstraksi dari beberapa proses dunia nyata.

2. *View*

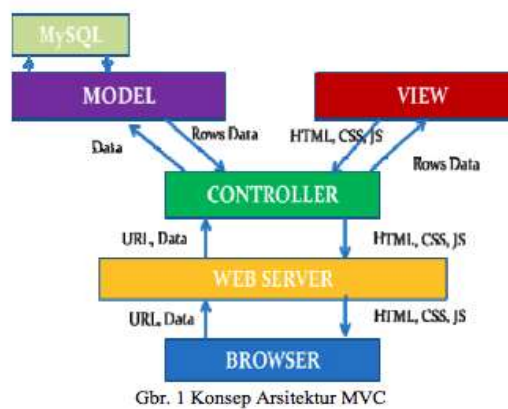
View, bertanggung jawab untuk pemetaan grafis ke sebuah perangkat. *View* biasanya memiliki hubungan 1-1 dengan sebuah permukaan layar dan tahu bagaimana untuk membuatnya. *View* melekat pada model dan *me-render* isinya ke permukaan layar. Selain itu, *View* secara otomatis menggambar ulang bagian layar yang terkena perubahan untuk menunjukkan perubahan tersebut.

3. *Controller*

Controller, menerima input dari pengguna dan menginstruksikan *Model* dan *View* untuk melakukan aksi berdasarkan masukan tersebut, sehingga *Controller* bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi.

Dengan menggunakan prinsip MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan *developer*, yaitu *programmer*

yang menangani bagian *model* dan *controller*, sedangkan *desainer* yang menangani bagian *view*, sehingga penggunaan arsitektur MVC dapat meningkatkan *maintainability* dan organisasi kode. Walaupun demikian dibutuhkan komunikasi yang baik antara *programmer* dan *designer* dalam menangani variabel-variabel yang akan ditampilkan. Gambar dibawah ini adalah bentuk arsitektur dari *framework Codeigniter*.



Gambar 2.1 Arsitektur MVC

2.7. Basis Data

Basis data secara umum dapat diartikan sebuah tempat penyimpanan data sebagai pengganti dari sistem konvensional yang berupa dokumen file. Basis data atau *database* ini didefinisikan sebagai kumpulan data yang dihubungkan secara bersama-sama dan gambaran dari data yang dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi (Sucipto, 2017).

Adapun perancangan basis data dalam tiga fase utama, yaitu:

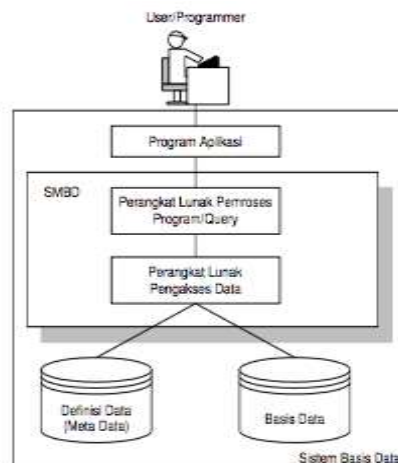
1. Perancangan *database* konseptual, merupakan proses membangun model dari data yang digunakan dalam sebuah organisasi dan tidak tergantung pada pertimbangan fisik.

2. Perancangan *database logical*, merupakan proses membangun model dari informasi yang digunakan dalam perusahaan berdasarkan model data spesifikasi, dan terbebas dari DBMS (*Database Management System*) tertentu dan pertimbangan fisik lainnya. Hasil akhir dari tahapan ini berupa sebuah kamus data yang berisi semua atribut beserta *key*-nya (*primary key*, *alternate key*, dan *foreign key*) dan *entity relational diagram* (ERD).
3. Perancangan *database* fisik, merupakan proses pembuatan deskripsi dari implementasi *database* pada penyimpanan sekunder yang menjelaskan relasi dasar, organisasi file dan indeks yang digunakan untuk akses yang efisien ke data dan setiap *integrity constraint* yang saling berhubungan dan juga pengukuran keamanan (*security*).

2.7.1. Sistem Basis Data

Gabungan antara basis data dan perangkat lunak termasuk didalamnya program aplikasi yang dibuat dan bekerja dalam satu sistem disebut dengan Sistem Basis Data (Dzacko Haidar, 2007).

C.J. Date menyatakan bahwa sistem basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data yang terkomputerisasi dengan tujuan untuk memelihara informasi tersebut saat dibutuhkan.



Gambar 2.2 Konsep Basis Data

2.7.2 MySQL

MySQL merupakan *database server open-source* yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat *software database* ini banyak digunakan oleh para praktisi untuk membangun suatu *project*. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh MySQL, memungkinkan bermacam-macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL (W Komputer, 2010).

MySQL berfungsi sebagai SQL (*Structured Query Language*) yang dimiliki sendiri dan sudah diperluas oleh MySQL umumnya digunakan bersamaan dengan PHP untuk membuat aplikasi server yang dinamis dan *powerfull*. Tidak sama dengan proyek-proyek seperti *Apache*, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL disponsori oleh sebuah perusahaan komersial Swedia MySQL AB (Istiono, Hijrah & Sutarya, 2016).

Keandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja yang pengoptimasi-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, MySQL mendukung operasi basis data transaksional maupun operasi basis data *non-transaksional*.

2.8. Xampp

Xampp merupakan perangkat lunak pendukung sistem operasi yang berfungsi sebagai web server dan berbasis *open-source*. Xampp juga merupakan perangkat lunak pendukung sistem operasi yang berfungsi untuk menjalankan website dengan bahasa pemrograman seperti, PHP dan menggunakan pengolah data MySQL sehingga dapat digunakan tanpa harus terkoneksi dengan jaringan internet. Xampp sangat mudah digunakan dan mendukung instalasi baik Linux maupun Windows (Oktasri & K. Denny, 2019).

Xampp sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, Perl dan juga program ini tersedia dalam GNU atau *General Public License* (Palit Randi V, 2015).

2.9. Visual Studio Code

Visual Studio Code adalah *software* yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari desktop. Muncul dengan *built-in* dukungan untuk *Javascript*, naskah dan Node.js dan memiliki array beragam ekstensi yang

tersedia untuk bahasa lain, termasuk C++, C#, Python dan PHP. Hal ini didasarkan sekitar Github ini elektron, yang merupakan versi *cross-platform* dari atom komponen kode *editing*, berdasarkan Javascript dan HTML5. Editor ini adalah fitur lengkap lingkungan pengembangan terpadu (IDE) dirancang untuk pengembangan yang bekerja dengan teknologi cloud yang terbuka Microsoft. Visual Studio Code menggunakan *open-source* NET perkakas untuk memberikan dukungan untuk memberikan dukungan untuk ASP.NET C # kode, membangun alat pengembang Omnisharp NET dan compiler roslyn. Antarmuka yang mudah untuk bekerja dengan, karena didasarkan pada gaya explorer umum, dengan panel di sebelah kiri, yang menunjukkan semua file dan folder anda memiliki akses ke panel editor di sebelah kanan, yang menunjukkan isi dari file yang telah dibuka. Dalam hal ini, editor telah dikembangkan dengan baik, dan menyenangkan pada mata. Ia juga memiliki fungsi yang baik, dengan *intellisense* dan *autocomplete* bekerja dengan baik untuk JSON, CSS, HTML dan Node.js (Minartiningtyas & Sumariata, 2019).

Visual Studio Code telah dirancang untuk bekerja dengan alat-alat yang ada, dan Microsoft menyediakan dokumentasi untuk membantu pengembang bersama, dengan bantuan untuk bekerja dengan ASP.NET 5, Node.js, dan Microsoft naskah, serta alat-alat yang dapat digunakan untuk membantu membangun dan mengelola aplikasi Node.js. Visual Studio Code benar-benar sedang ditargetkan pada pengembang JavaScript yang ingin alat pengembangannya lengkap untuk scripting server-side mereka dan yang mungkin ingin usaha dari Node.js untuk kerangka berbasis NET. Visual Studio Code,

adalah belum solid, lintas platform kode Editor ringan, yang dapat digunakan oleh siapa saja untuk membangun aplikasi untuk Web.

2.10. API

API atau *Application programming interface* merupakan suatu dokumentasi yang terdiri dari interface, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk “membongkar” suatu software, kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem function. Proses ini dikelola melalui sistem operasi. Keunggulan dari API ini adalah memungkinkan suatu aplikasi dengan aplikasi lainnya dapat saling berhubungan dan berinteraksi (Ramadhani M Fajar, 2016).

2.11. Penelitian Terkait

Berdasarkan hasil dan pembahasan pada penelitian berjudul “Sistem Informasi Penilaian Angka Kredit Dosen Berbasis Web (Studi kasus PAK Online Kopertais Wilayah I DKI Jakarta)”, dimana analisis dan perancangan sistem menggunakan metode *waterfall*. Sistem dapat diimplementasikan dengan baik, tanpa adanya permasalahan dan dibuktikan dengan pengujian melalui metode *User Acceptance Test* atau UAT (Sulhi Ahmad, 2020).

Kemudian pada penelitian berjudul “Perancangan Sistem Informasi Kenaikan Jabatan Fungsional Dosen Dengan *Framework* Laravel Berbasis Web”, maka peneliti dapat mengambil kesimpulan sistem dapat digunakan untuk

membantu pengajuan dalam memperoleh angka kredit dan menyimpan data diri dosen pengaju (Agasi A. & Hadi A., 2018).

Pada penelitian lain berjudul “Pengembangan Sistem Penilaian Angka Kredit Untuk Kenaikan Jabatan Akademik Dosen Pada Jurusan Sistem Informasi Fakultas Ilmu Komputer Universitas Sriwijaya”, maka peneliti dapat mengambil kesimpulan bahwa sistem mampu mengelola data penilaian yang diisi oleh dosen yang kemudian akan dihitung angka kreditnya, dan juga sistem ini mampu mengelola data pendidikan dan penunjang (Lestar E, Heroza & Zubiah I., 2017).