

**PENGEMBANGAN DAN ANALISIS KUALITAS SISTEM INFORMASI
BIMBINGAN TUGAS AKHIR SKRIPSI *ONLINE* UNTUK MAHASISWA
TINGKAT AKHIR PENDIDIKAN TEKNIK ELEKTRONIKA FT UNY**

TUGAS AKHIR SKRIPSI



Diajukan Kepada Fakultas Teknik Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan
Guna Memperoleh Gelar Sarjana Pendidikan

Disusun Oleh:
Zumrotul Ahkamiyati
NIM. 10520241027

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2016**

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

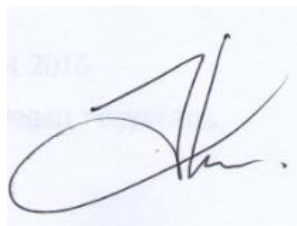
**PENGEMBANGAN DAN ANALISIS KUALITAS SISTEM INFORMASI
BIMBINGAN TUGAS AKHIR SKRIPSI *ONLINE* UNTUK MAHASISWA
TINGKAT AKHIR PENDIDIKAN TEKNIK ELEKTRONIKA FT UNY**

Disusun oleh:

Zumrotul Ahkamiyati
NIM 10520241027

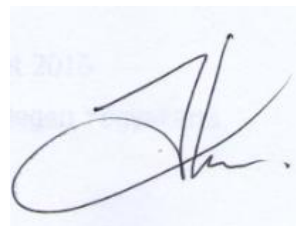
telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan
Ujian Akhir Tugas Akhir Skripsi bagi yang bersangkutan.

Mengetahui,
Ketua Program Studi
Pendidikan Teknik Informatika,



Handaru Jati, Ph.D.
NIP. 19740511 199903 1 002

Yogyakarta, Februari 2016
Disetujui,
Dosen Pembimbing,



Handaru Jati, Ph.D.
NIP. 19740511 199903 1 002

HALAMAN PENGESAHAN



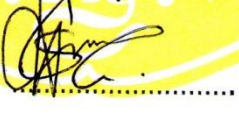
Tugas Akhir Skripsi

PENGEMBANGAN DAN ANALISIS KUALITAS SISTEM INFORMASI BIMBINGAN TUGAS AKHIR SKRIPSI *ONLINE* UNTUK MAHASISWA TINGKAT AKHIR PENDIDIKAN TEKNIK ELEKTRONIKA FT UNY

Disusun Oleh :
Zumrotul Ahkamiyati
NIM 10520241027

Telah dipertahankan di depan tim penguji Tugas Akhir Skripsi Program Studi Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta pada tanggal 24 Maret 2016

TIM PENGUJI

Nama/Jabatan	Tanda Tangan	Tanggal
Handaru Jati, Ph.D. Ketua Penguji/Pembimbing		2/2/2016
Ponco Wali Pranoto, M.Pd. Sekretaris		31/3/2016
Nurkhamid, Ph.D. Penguji		31/3/2016

Yogyakarta, Maret 2016

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



Dr. Moch. Bruri Triyono

NIP. 19560216 198603 1 003

HALAMAN PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Zumrotul Ahkamiyati

NIM : 10520241027

Program Studi : Pendidikan Teknik Informatika

Judul TAS : Pengembangan dan Analisis Kualitas Sistem Informasi Bimbingan Tugas Akhir Skripsi *Online* Untuk Mahasiswa Tingkat Akhir Pendidikan Teknik Elektronika FT UNY

menyatakan bahwa skripsi ini benar-benar karya saya sendiri *). Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Yogyakarta, Februari 2016

Yang menyatakan,



Zumrotul Ahkamiyati

NIM. 10520241027

HALAMAN MOTTO

"Done is better than perfect"

"Hope is a good thing, the best thing. No good thing ever dies"

The Shawshank Redemption

"The journey of a thousand miles begins with one step"

Lao Tzu

"Never leave that till tomorrow wich you can do today"

Benjamin Franklin

"Dimana bumi dipijak, disitu langit dijunjung"

"Petualangan mengajarkan kita bahwa kemanapun kita pergi, tujuan utamanya

adalah pulang dengan selamat"

HALAMAN PERSEMBAHAN

Segala puji bagi Tuhan yang Maha Esa atas segala karunia dan kuasa-Nya sehingga saya mampu menyelesaikan skripsi ini. Skripsi ini saya persembahkan untuk :

1. Kedua orang tua saya, Bapak Iswanto dan Ibu Siti Martiah yang senantiasa memberikan dukungan beserta doanya.
2. Roni Royani, yang telah memberikan motivasi dan banyak membantu dalam penyelesaian skripsi ini.
3. Bapak dan Ibu Dosen yang telah memberikan bimbingan serta wawasan selama ini.
4. Keluarga besar PT. Qwords Company yang memberikan semangat agar saya segera menyelesaikan skripsi ini
5. Kawan-kawan cemiwel dan Teman-teman PTI E yang telah memberikan bantuan dan inspirasi dalam pengerjaan skripsi ini.
6. Serta semua pihak yang telah memberikan dukungan, bantuan, dan doa dalam penyusunan hingga penyelesaian skripsi ini.

PENGEMBANGAN DAN ANALISIS KUALITAS SISTEM INFORMASI BIMBINGAN TUGAS AKHIR SKRIPSI *ONLINE* UNTUK MAHASISWA TINGKAT AKHIR PENDIDIKAN TEKNIK ELEKTRONIKA FT UNY

Oleh :
Zumrotul Ahkamiyati
10520241027

ABSTRAK

Penelitian ini bertujuan untuk (1) memantau progress pengerjaan skripsi, mengefektifkan waktu bimbingan, dan bimbingan terjadwal baik dengan mengembangkan sistem informasi bimbingan skripsi, (2) mengatasi kegagalan dalam pengembangan perangkat lunak sistem informasi bimbingan skripsi, dengan menguji kualitas perangkat lunak berdasarkan standard pengujian kualitas perangkat lunak ISO 25010 dari aspek *functional suitability*, *performance efficiency*, *usability*, *security*, *reliability*, *compatibility*, *maintainability*, dan *portability*.

Penelitian ini menggunakan metode pengembangan *Research and Development (R&D)* dengan model pengembangan *waterfall* yang terdiri dari 4 tahap, yaitu tahap analisis kebutuhan, tahap desain, tahap implementasi, dan tahap pengujian.

Dari penelitian menunjukkan bahwa (1) perangkat lunak sistem informasi bimbingan skripsi dikembangkan menggunakan *PHP* dan *Bootstrap*. (2) Hasil pengujian pada aspek (1) *functional suitability* mendapatkan nilai sebesar 100% dan masing-masing subkarakteristik memperoleh nilai $X=1$, (2) *performance efficiency* memperoleh nilai rata-rata waktu memuat halaman 2,6 detik, (3) *usability* memperoleh kategori sangat layak dengan persentase sebesar 85,30%, (4) *security* memperoleh tingkat keamanan level 2 atau medium, (5) *reliability* memperoleh nilai 1 atau 100%, (6) *compatibility*, pengujian tidak dilakukan, karena resource hanya digunakan untuk aplikasi ini saja, (7) *maintainability* memperoleh nilai 67,3% dengan kategori cukup, (8) *portability* telah dipenuhi, karena dapat dijalankan di berbagai *browser*.

Kata kunci : Sistem Informasi Bimbingan Skripsi, Kualitas Perangkat Lunak, ISO 25010

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT atas berkat rahmat dan karunia-Nya, Tugas Akhir Skripsi dalam rangka untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan dengan judul "Analisis Pengembangan Sistem Informasi Bimbingan Tugas Akhir Skripsi Online Untuk Mahasiswa Tingkat Akhir Pendidikan Teknik Elektronika FT UNY" dapat disusun sesuai dengan harapan. Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan dan kerjasama dengan pihak lain. Berkenaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada yang terhormat:

1. Handaru Jati, Ph.D selaku Dosen Pembimbing TAS yang telah banyak memberikan semangat, dorongan, dan bimbingan selama penyusunan Tugas Akhir Skripsi ini.
2. Muhammad Munir, M.Pd, Dessy Irmawati, M.T, dan Prof. Herman Dwi Surjono, Drs., M.Sc., MT., Ph.D. selaku Validator instrumen TAS yang memberikan saran/masukan perbaikan sehingga penelitian TAS dapat terlaksana sesuai dengan tujuan.
3. Handaru Jati, Ph.D., Ponco Wali Pranoto, M.Pd., dan Nurkhamid, Ph.D. selaku Ketua Penguji, Sekretaris, dan Penguji yang memberikan koreksi perbaikan secara komprehensif terhadap TAS ini.
4. Dr. Fatchul Arifin, S.T., M.T. selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Handaru Jati, Ph.D selaku Ketua Program Studi Pendidikan Teknik Informatika beserta dosen dan staf yang telah memberikan bantuan

dan fasilitas selama proses penyusunan pra proposal sampai dengan selesainya Tugas Akhir Skripsi ini

5. Dr. Moch. Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi.
6. Dr. Eko Marpanaji, selaku Dosen Penasihat Akademik yang selalu memberikan arahan dan bimbingan dalam menempuh studi ini.
7. Dessy Irmawati, M.T selaku Dosen di Jurusan Pendidikan Teknik Elektronika yang telah memberikan bantuan dalam pelaksanaan penelitian Tugas Akhir Skripsi ini.
8. Semua pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan satu per satu atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.
9. Akhirnya, semoga segala bantuan yang telah berikan semua pihak di atas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah SWT dan Tugas Akhir Skripsi ini menjadi informasi bermanfaat bagi pembaca atau pihak lain yang membutuhkannya.

Yogyakarta, 1 Maret 2016

Penulis,

Zumrotul Ahkamiyati

NIM 10520241027

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
HALAMAN MOTTO	v
HALAMAN PERSEMBAHAN	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xviii
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	5
C. Batasan Masalah	6
D. Rumusan Masalah	7
E. Tujuan Penelitian	7
F. Manfaat Penelitian	8
BAB II KAJIAN PUSTAKA	9
A. Deskripsi Teori	9
1. Perangkat Lunak	9
2. <i>Unified Modeling Language</i> (UML)	10
3. Sistem Informasi	13

	4. Kualitas Perangkat Lunak	14
	5. Penggunaan Sistem Informasi pada Perguruan Tinggi	23
	6. Analisis dan Perancangan Sistem Informasi	23
	B. Hasil Penelitian Yang Relevan	26
	C. Kerangka Pikir	27
BAB III	METODE PENELITIAN	29
	A. Desain Penelitian	29
	1. Analisis Kebutuhan Perangkat Lunak	30
	2. Perancangan (<i>Design</i>)	31
	3. Pengkodean (<i>Coding</i>)	31
	4. Pengujian (<i>Testing</i>)	32
	5. Pemeliharaan (<i>Maintenance</i>)	34
	B. Subyek Penelitian	34
	C. Metode Penelitian	34
	D. Instrumen Penelitian	35
	1. Instrumen <i>Functional Suitability</i>	36
	2. Instrumen <i>Performance Efficiency</i>	39
	3. Instrumen <i>Compatibility</i>	39
	4. Instrumen <i>Usability</i>	39
	5. Instrumen <i>Reliability</i>	41
	6. Instrumen <i>Security</i>	41
	7. Instrumen <i>Maintainability</i>	42
	8. Instrumen <i>Portability</i>	42
	E. Teknik Analisis Data	42

	1. Analisis Faktor Kualitas <i>Functional Suitability</i>	42
	2. Analisis Faktor Kualitas <i>Performance Efficiency</i>	43
	3. Analisis Faktor Kualitas <i>Compatibility</i>	45
	4. Analisis Faktor Kualitas <i>Usability</i>	45
	5. Analisis Faktor Kualitas <i>Reliability</i>	46
	6. Analisis Faktor Kualitas <i>Security</i>	47
	7. Analisis Faktor Kualitas <i>Maintainability</i>	47
	8. Analisis Faktor Kualitas <i>Portability</i>	48
BAB IV	HASIL DAN PEMBAHASAN	49
	A. Tahapan Analisis Kebutuhan	49
	1. Analisis Kebutuhan Fungsional	49
	2. Analisis Kebutuhan Perangkat Lunak	50
	B. Tahapan Desain	50
	1. Desain UML (<i>Unified Modeling Language</i>)	51
	a. Desain <i>Use Case Diagram</i>	51
	b. Desain <i>Activity Diagram</i>	56
	c. Desain <i>Sequence Diagram</i>	59
	2. Perancangan Desain <i>Interfaces</i>	62
	3. Perancangan Database	64
	C. Tahapan Implementasi	64
	1. Implementasi Fungsi	64
	2. Implementasi Database	67
	D. Tahap Pengujian	70
	1. Hasil Pengujian <i>Functional Suitability</i>	70

	2. Hasil Pengujian <i>Performance Efficiency</i>	73
	3. Hasil Pengujian <i>Compatibility</i>	79
	4. Hasil Pengujian <i>Usability</i>	79
	5. Hasil Pengujian <i>Reliability</i>	82
	6. Hasil Pengujian <i>Security</i>	83
	7. Hasil Pengujian <i>Maintainability</i>	84
	8. Hasil Pengujian <i>Portability</i>	86
	E. Pembahasan Hasil Penelitian	87
BAB V	KESIMPULAN DAN SARAN	89
	A. Kesimpulan	89
	B. Saran	90
	DAFTAR PUSTAKA	91
	LAMPIRAN	94

DAFTAR GAMBAR

Gambar 1.	Kerangka Pikir	28
Gambar 2.	Metode <i>Waterfall</i> dari Sommerfille	30
Gambar 3.	<i>Use Case</i> Diagram Sistem	52
Gambar 4.	Diagram <i>Use Case</i> Admin	53
Gambar 5.	Diagram <i>Use Case</i> Mahasiswa	54
Gambar 6.	Diagram <i>Use Case</i> Dosen	55
Gambar 7.	<i>Activity Diagram</i> Login	56
Gambar 8.	<i>Activity Diagram</i> Tambah Data	56
Gambar 9.	<i>Activity Diagram</i> Ubah Data	57
Gambar 10.	<i>Activity Diagram</i> Delete Data	57
Gambar 11.	<i>Activity Diagram</i> Upload Data	58
Gambar 12.	<i>Activity Diagram</i> Download Data	58
Gambar 13.	<i>Sequence Diagram</i> Login	59
Gambar 14.	<i>Sequence Diagram</i> Tambah data	59
Gambar 15.	<i>Sequence Diagram</i> Edit Data	60
Gambar 16.	<i>Sequence Diagram</i> Hapus Data	60
Gambar 17.	<i>Sequence Diagram</i> Upload.....	61
Gambar 18.	<i>Sequence Diagram</i> Download.....	61
Gambar 19.	Desain <i>User Interface</i> Login	62
Gambar 20.	Desain <i>User Interface</i> Halaman Utama Admin	62
Gambar 21.	Desain <i>User Interface</i> Halaman Utama Dosen	63
Gambar 22.	Desain <i>User Interfaces</i> Menu utama Mahasiswa	63
Gambar 23.	Perancangan <i>Database</i>	64

Gambar 24. Halaman <i>Login</i>	65
Gambar 25. Halaman Dosen	65
Gambar 26. Halaman Mahasiswa	66
Gambar 27. Halaman Admin	66
Gambar 28. Database Sistem Informasi	67
Gambar 29. Tabel Admin	67
Gambar 30. Tabel Mahasiswa	68
Gambar 31. Tabel Dosen	68
Gambar 32. Tabel Jurusan	68
Gambar 33. Tabel Prodi	68
Gambar 34. Tabel pembimbing	69
Gambar 35. Tabel Bimbingan	69
Gambar 36. Tabel konsultasi	69
Gambar 37. Tabel Pesan	69
Gambar 38. Tabel Fakultas	70
Gambar 39. Tabel settingan	70
Gambar 40. Tabel Kategori	70
Gambar 41. Grade Pengujian <i>Efficiency</i> Halaman Login.....	73
Gambar 42. Grade Pengujian <i>Efficiency</i> Halaman Profil Mahasiswa	73
Gambar 43. Grade Pengujian <i>Efficiency</i> Halaman Ubah Profil Mahasiswa	74
Gambar 44. Grade Pengujian <i>Efficiency</i> Halaman Catatan Bimbingan	74
Gambar 45. Grade Pengujian <i>Efficiency</i> Halaman Konsultasi Online.....	75
Gambar 46. Grade Pengujian <i>Efficiency</i> Halaman Profil Dosen	75
Gambar 47. Grade Pengujian <i>Efficiency</i> Halaman Ubah Profil Dosen.....	76
Gambar 48. Grade Pengujian <i>Efficiency</i> Halaman Daftar Mahasiswa	76

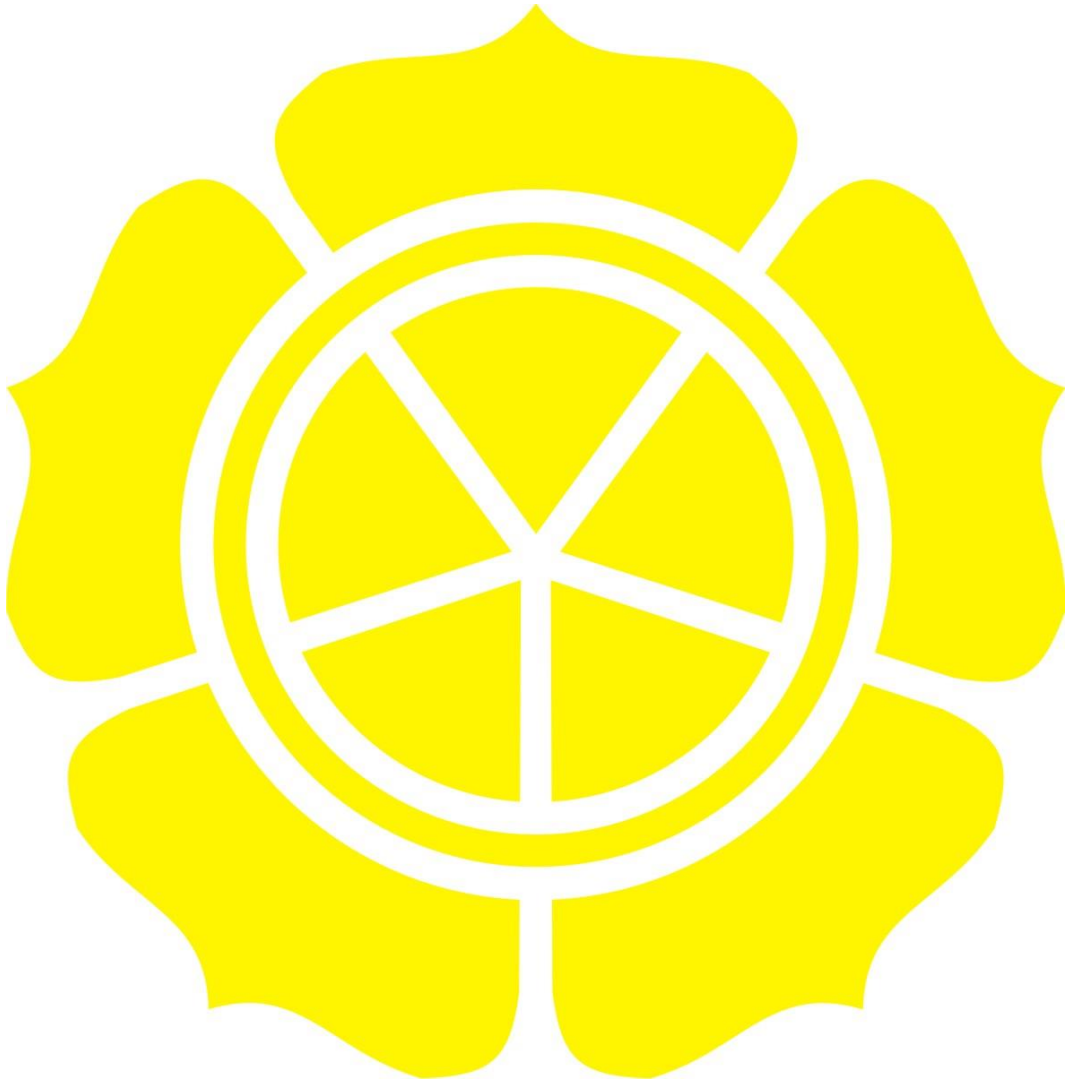
Gambar 49. Grade Pengujian <i>Efficiency</i> Halaman Daftar Dosen.....	77
Gambar 50. Grade Pengujian <i>Efficiency</i> Halaman Tambah Data Jurusan.....	77
Gambar 51. Grade Pengujian <i>Efficiency</i> Halaman Tambah Data Prodi.....	78
Gambar 52. Grade Pengujian <i>Efficiency</i> Halaman Ganti Password	78
Gambar 53. Hasil Perhitungan Reliabilitas dengan Alpha Cronbach.....	81
Gambar 54. Hasil pengujian <i>Reliability</i> menggunakan Loadimpact.....	82
Gambar 55. Hasil pengujian <i>Reliability</i> menggunakan Loadimpact	82
Gambar 56. Laporan Pengujian <i>Security</i>	83
Gambar 57. Hasil Scan dengan Acunetix Web Vulnerability Scanner	83
Gambar 58. Hasil screenshot menggunakan PHP Copy/Paste Detector	84

DAFTAR TABEL

Tabel 1. Data Jumlah Mahasiswa Pada Setiap Dosen Pembimbing	3
Tabel 2. Karakteristik model ISO-25010	15
Tabel 3. Kriteria Skor Pengujian Usability	19
Tabel 4. TQI Score untuk Standard Aspek Maintainability	21
Tabel 5. Instrumen <i>functional completeness</i> dan <i>functional appropriateness</i> .	36
Tabel 6. Instrumen <i>functional correctness</i>	38
Tabel 7. Instrumen Penelitian <i>Usability</i>	40
Tabel 8. Instrumen Penelitian <i>Security</i>	41
Tabel 9. Aturan Yslow	43
Tabel 10. Skor <i>Performance Efficiency</i> pada aturan YSlow.....	44
Tabel 11. Aturan PageSpeed Insights untuk menguji <i>performance efficiency</i>	44
Tabel 12. Kriteria skor pengujian <i>usability</i>	46
Tabel 13. Nilai Alpha Croncbach	46
Tabel 14. Kategori Penilaian <i>Maintainability</i>	47
Tabel 15. Pengujian Faktor Kualitas Portability	48
Tabel 16. Definisi Aktor	51
Tabel 17. Hasil Pengujian <i>Functional Suitability</i>	71
Tabel 18. Hasil Pengujian Subkarakteristik <i>Functional Suitability</i>	72
Tabel 19. Hasil Pengujian <i>Performance Efficiency</i>	79
Tabel 20. Hasil Pengujian <i>Usability</i>	80
Tabel 21. Tabel Hasil Pengujian Portability di <i>Browser</i>	85

DAFTAR LAMPIRAN

Lampiran 1. Surat Pernyataan Validasi Instrumen	95
Lampiran 2. Form Testing Aplikasi	96
Lampiran 3. Pengujian Aspek <i>Functional Suitability</i>	97
Lampiran 4. Pengujian Aspek <i>Usability</i>	98



BAB I PENDAHULUAN

A. Latar Belakang

Undang-undang baru tentang Standar Nasional Perguruan Tinggi telah resmi disahkan pada 11 Juni 2014. Undang-undang baru tersebut termuat dalam Peraturan Menteri Pendidikan dan Kebudayaan (Permendikbud) No 49/2014. Peraturan baru tersebut mengatur sejumlah standard untuk Perguruan Tinggi, mulai dari kurikulum, kompetensi, hingga lama waktu studi. Pada pasal 17 menjelaskan bahwa lama masa studi bagi mahasiswa program sarjana (S1) dan diploma 4 (D4) tidak boleh lebih dari 5 tahun. Dengan adanya aturan baru tersebut, seluruh Universitas yang ada di Indonesia wajib memberlakukannya pada mahasiswa program sarjana ataupun diploma 4 untuk menyelesaikan masa studinya maksimal 5 tahun.

Dengan adanya peraturan baru tersebut, mahasiswa tingkat akhir dihimbau agar dapat menyelesaikan pengerjaan tugas akhir skripsi dengan baik dan efisien. Tugas akhir skripsi seringkali dianggap sebagai suatu beban berat yang harus ditanggung oleh mahasiswa tingkat akhir yang hendak menyelesaikan kuliahnya. Hal tersebut disebabkan oleh beberapa hal, di antaranya :

- Mahasiswa belum memiliki pengalaman dalam penulisan karya ilmiah
- Mahasiswa belum memiliki pandangan atau pemikiran terkait topik yang akan diteliti
- Mahasiswa memiliki kegiatan di luar kampus dan tidak bisa mengatur waktu dengan baik
- Mahasiswa mengalami kesulitan mencari buku referensi yang relevan
- Kurangnya pengetahuan mahasiswa dalam hal metodologi penelitian

Beberapa permasalahan yang dirasakan mahasiswa di atas sebenarnya dapat dibantu dengan adanya bimbingan skripsi. Bimbingan skripsi diharapkan dapat menjadi bantuan yang diberikan oleh dosen pembimbing kepada mahasiswa untuk menyelesaikan skripsi sesuai dengan topik dan tujuannya. Bimbingan tersebut bertujuan membantu memberi informasi kepada mahasiswa untuk memahami dan menggunakan kesempatan secara efektif untuk menyelesaikan laporan skripsinya.

Sebagai bagian dari proses penulisan tugas akhir ataupun skripsi, proses bimbingan antara dosen dan mahasiswa bimbingan menjadi sangat penting, agar mahasiswa mampu mengefektifkan waktu penyusunan skripsi dan mampu menyusun skripsi dengan baik hingga skripsi siap diujikan dan berkualitas. Oleh karena hal itu, diharapkan bimbingan yang efektif antara dosen dan mahasiswa tersebut dapat membantu mahasiswa untuk mengembangkan diri dan menyelesaikan laporan skripsi dengan waktu yang cepat (Winkel, 1991).

Namun, permasalahan timbul ketika antara kedua belah pihak sulit melakukan komunikasi (bimbingan) dikarenakan berbagai hal, misalnya saja jadwal yang tidak sinkron, perbedaan lokasi diantara keduanya, ataupun permasalahan-permasalahan yang lain.

Adapun kendala yang terjadi dalam proses bimbingan tugas akhir yang terjadi selama ini sebagai berikut:

1. Motivasi mahasiswa yang masih kurang untuk mengerjakan tugas akhir, karena mahasiswa juga memiliki kegiatan yang banyak di luar kampus, contohnya seperti bekerja yang membuat mahasiswa mengulur-ulur waktu untuk mengerjakan skripsi dan melakukan bimbingan.

2. Mahasiswa yang menggunakan metode *research and development*, seringkali merasa kesulitan di pengembangan proyeknya.
 3. Jadwal dosen yang padat sehingga mahasiswa kesulitan melakukan bimbingan.
 4. Kurang adanya pantauan penyelesaian skripsi dari dosen pembimbing.
 5. Beban mahasiswa bimbingan pada dosen pembimbing yang tidak seimbang.
- Seperti tabel 1 berikut yang menampilkan data bimbingan dosen.

Tabel 1. Data Jumlah Mahasiswa Pada Setiap Dosen Pembimbing.

Nama Dosen	Elektronika	Informatika	Jumlah
Abdul Halim	13	0	13
Adi Dewanto	0	31	31
Aris Nasuha	7	9	16
Djoko Santoso	21	13	34
Eko Marpanaji	11	24	35
Handaru Jati	4	31	35
Herman Dwi Surjono	12	21	33
Kadarisman	1	9	10
Masduki	15	17	32
Munir	22	20	42
Mashoedah	0	4	4
Pramudi	3	14	17
Putu	18	7	25
Rahmatul	2	6	8
Ratna	0	20	20
Suparman	25	9	34
Suprpto	7	8	15
Slamet	28	7	35
Sri Waluyanti	7	0	7
Totok Sukardiyono	13	27	40
Umi Rochayati	12	23	35
Soenarto	0	0	0
Priyanto	0	0	0
Nurkhamid	0	0	0
Fatchul	0	0	0
Desi	6	4	10
Indri	0	0	0
	227	304	531
Achmad Fatchi	8	8	16
Indri	1	1	2
	450	586	1036

(Sumber : Koordinator Skripsi Jurusan PTE, 2016)

6. Waktu bimbingan yang kurang, karena dosen pembimbing sering ada tugas atau penelitian di luar kampus.
7. Belum adanya kesepahaman antara dosen dan mahasiswa ketika bimbingan.
8. Mahasiswa takut bertemu dosen pembimbing karena merasa sulit menyesuaikan diri.

Saat ini, hampir semua perguruan tinggi memanfaatkan teknologi informasi, baik di luar atau di dalam negeri. Adanya teknologi internet memiliki pengaruh yang besar pada aktivitas pendidikan di perguruan tinggi. Dengan berkembangnya internet, saat ini telah banyak perguruan tinggi yang menerapkan sistem informasi *online* untuk proses administrasinya. Dalam beberapa kasus, kegiatan yang dulunya dilakukan secara konvensional, sekarang ini mulai banyak yang dilakukan secara *online*. Seperti materi kuliah yang dulu disampaikan secara langsung, dengan sistem *e-learning* mahasiswa bisa mengunduh materi tersebut melalui internet. Contohnya lagi, jika dulu mahasiswa harus pergi ke perpustakaan untuk mencari literatur, saat ini literatur bisa dicari secara *online*, sudah banyak *e-book* dan *e-journal* yang ada di internet.

Melihat paradigma teknologi informasi yang sudah begitu pesat, untuk mengatasi permasalahan dalam bimbingan tugas akhir, peneliti akan membuat sebuah aplikasi yang berguna untuk memudahkan komunikasi antara mahasiswa dan dosen dalam hal efektivitas melakukan bimbingan. Sistem informasi yang berbasis *online* ini diharapkan dapat memajemen waktu mahasiswa untuk melakukan bimbingan dengan dosen dan dosen dapat memantau perkembangan mahasiswa bimbingannya dalam membuat laporan tugas akhir dan atau skripsi. Dengan adanya aplikasi ini diharapkan mahasiswa dan dosen dapat tetap

melakukan bimbingan walaupun tidak secara tatap muka dan dapat menyelesaikan proyek sesuai dengan rencana pengerjaan.

Proses pengembangan perangkat lunak sistem informasi bimbingan tugas akhir skripsi ini merupakan proses yang rumit dan memiliki tingkat kegagalan yang tinggi. Pengembangan perangkat lunak ini membutuhkan waktu, tenaga, dan biaya yang tidak sedikit. Oleh karenanya, dibutuhkan perencanaan yang baik dan pengujian yang teliti. Pengujian perangkat lunak merupakan tahapan kritis dalam tahap pengembangan. Hasil dari pengujian tersebut akan mempresentasikan apakah pengembangan perangkat lunak sistem informasi ini menghasilkan sistem informasi yang baik dan telah sesuai dengan kebutuhan, baik fungsional maupun non fungsional.

B. Identifikasi Masalah

Berdasarkan uraian pada latar belakang, penulis dapat mengidentifikasi beberapa permasalahan sebagai berikut:

1. Motivasi mahasiswa yang masih kurang untuk mengerjakan tugas akhir, karena mahasiswa juga memiliki kegiatan yang banyak di luar kampus, contohnya seperti bekerja yang mengakibatkan mahasiswa mengulur-ulur waktu dalam mengerjakan tugas akhir skripsinya.
2. Mahasiswa yang menggunakan metode *research and development*, seringkali merasa kesulitan di pengembangan proyeknya
3. Jadwal dosen yang padat sehingga mahasiswa kesulitan melakukan bimbingan
4. Kurang adanya pantauan progress penyelesaian skripsi dari dosen pembimbing
5. Beban mahasiswa bimbingan pada dosen pembimbing yang tidak seimbang

6. Waktu bimbingan yang kurang, karena dosen pembimbing sering ada tugas atau penelitian di luar kampus
7. Belum adanya kesepahaman antara dosen dan mahasiswa ketika bimbingan
8. Mahasiswa takut bertemu dosen pembimbing karena merasa sulit menyesuaikan diri
9. Pengembangan perangkat lunak sistem informasi bimbingan tugas akhir skripsi merupakan proses yang rumit dan memiliki tingkat kegagalan yang tinggi

C. Batasan Masalah

Berdasarkan identifikasi masalah di atas, penulis akan memfokuskan permasalahan yang akan diteliti. Sehingga penulis akan membatasi beberapa permasalahan dari banyaknya permasalahan yang timbul, di antaranya:

1. Kurang adanya pantauan progress penyelesaian skripsi dari dosen pembimbing
2. Waktu bimbingan yang kurang, karena dosen pembimbing sering ada tugas atau penelitian di luar kampus
3. Jadwal dosen yang padat sehingga mahasiswa kesulitan melakukan bimbingan
4. Pengembangan perangkat lunak sistem informasi bimbingan tugas akhir skripsi merupakan proses yang rumit dan memiliki tingkat kegagalan yang tinggi.

D. Rumusan Masalah

Setelah mengidentifikasi masalah dan batasan masalah yang ada, penulis merumuskan pada permasalahan yang akan diteliti, antara lain:

1. Bagaimana memantau progress pengerjaan skripsi dengan baik, waktu bimbingan skripsi efektif meskipun dosen sering ada tugas penelitian, dan bimbingan terjadwal dengan baik?
2. Bagaimana mengatasi kegagalan dalam pengembangan perangkat lunak sistem informasi bimbingan tugas akhir skripsi di Jurusan Pendidikan Teknik Elektronika FT UNY?

E. Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan dari penelitian ini adalah:

1. Untuk memantau progress pengerjaan skripsi, mengefektifkan waktu bimbingan, dan bimbingan terjadwal dengan baik, dilakukan pengembangan aplikasi sistem informasi bimbingan tugas akhir skripsi *online*.
2. Untuk mengatasi kegagalan dalam pengembangan perangkat lunak, dilakukan pengujian kualitas aplikasi sistem informasi bimbingan tugas akhir skripsi *online* di Jurusan Pendidikan Teknik Elektronika Fakultas Teknik UNY berdasarkan standard ISO/IEC 25010 pada aspek *functional suitability, performance efficiency, usability, security, reliability, compatibility, maintainability, dan portability*.

F. Manfaat Penelitian

1. Bagi mahasiswa

Diharapkan penelitian ini bisa membantu mahasiswa tingkat akhir agar lebih mudah dalam melakukan bimbingan tugas akhir skripsi dengan dosen pembimbing dan dapat menyelesaikan tugas akhir seefisien mungkin.

2. Bagi dosen

Diharapkan penelitian ini bisa membantu dosen untuk memantau proses pengerjaan tugas akhir skripsi yang dilakukan oleh mahasiswa bimbingannya tanpa mengganggu kegiatan atau kesibukan lainnya.

3. Bagi Koordinator Skripsi

Diharapkan penelitian ini dapat membantu koordinator skripsi di tingkat Jurusan untuk memantau proses bimbingan skripsi antara mahasiswa dan dosen.

BAB II KAJIAN PUSTAKA

A. Deskripsi Teori

1. Perangkat Lunak

Software atau perangkat lunak adalah program komputer yang berisi instruksi untuk menjalankan tugas-tugas tertentu. *Software* berperan sebagai jembatan yang menerjemahkan instruksi dari pengguna komputer kepada komputer untuk dieksekusi.

Dalam bukunya, Pressman mendeskripsikan bahwa *software* adalah instruksi (program komputer) yang bila dieksekusi memberikan fungsi yang diinginkan dan kinerja, struktur data yang memungkinkan program memanipulasi informasi secara memadai, dan dokumen yang menggambarkan operasi dan penggunaan program.

Adapun karakteristik *software* adalah sebagai berikut:

1. Perangkat lunak dibangun atau direkayasa, tidak dihasilkan dengan menggunakan mesin seperti perangkat keras.
2. Perangkat lunak tidak akan usang atau lusuh oleh waktu, namun mutunya dapat menurun, oleh karena itu harus selalu diperbarui
3. Sebagian perangkat lunak akan dibuat menurut pesanan atau kebutuhan
4. Perangkat lunak dibangun dengan fleksibel, sehingga mudah dimodifikasi.

Terdapat tujuh kategori terkait jenis dari perangkat lunak, diantaranya :

1. *System Software*, yaitu kumpulan dari program yang dibuat untuk melayani program lain. Seperti *Operating system* dan *compilers*.
2. *Application Software*, yaitu suatu perangkat lunak yang berdiri sendiri dan digunakan untuk kebutuhan yang spesifik.

3. *Engineering and scientific Software*, yaitu suatu perangkat lunak yang biasanya didominasi oleh penggunaan algoritma. Seperti program untuk kebutuhan astronomi, pabrik, dan biologi.
4. *Embedded Software*, yaitu perangkat lunak yang ditanam pada sistem tertentu, digunakan untuk mengatur fungsi pada pengguna maupun dirinya sendiri.
5. *Web – based software*, yaitu perangkat lunak berbasis *website*, atau yang biasa disebut *webapps*.
6. *Product-line Software*, yaitu perangkat lunak yang dibuat untuk membantu kebutuhan pengguna yang bersifat spesifik, dan dapat digunakan oleh pengguna yang berbeda. Contohnya seperti *database management*, *computer graphic*, dan *multimedia*.
7. *Artificial intelligence software*, yaitu perangkat lunak yang dibuat untuk dapat menyelesaikan suatu masalah yang kompleks, yang tidak dapat diselesaikan dengan perhitungan ataupun analisis langsung. Contohnya seperti pada jaringan syaraf tiruan dan *robotic*.

2. Unified Modeling Language (UML)

Menurut Adi Nugroho (2010), "*Unified Modeling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek". Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Dari pengertian yang dikemukakan diatas, dapat dikatakan bahwa *Unified Modeling Language* (UML) adalah sebuah bahasa pemodelan yang digunakan untuk pembuatan sistem yang berorientasi objek.

Menurut Widodo (2011), "Beberapa literature menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi". Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

a. Diagram kelas (*Class Diagram*)

Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi.

b. Diagram paket (*Package Diagram*)

Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.

c. Diagram use-case (*Usecase Diagram*)

Diagram ini memperlihatkan himpunan use-case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini digunakan untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

d. Diagram interaksi dan sequence (*Sequence Diagram*)

Diagram ini menekankan pada pengiriman pesan dalam suatu waktu tertentu.

e. Diagram komunikasi (*Communication Diagram*)

Diagram ini sebagai pengganti diagram kolaborasi UML yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.

f. Diagram statechart (*Statechart Diagram*)

Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (state), transisi, kejadian serta aktivitas.

g. Diagram aktivitas (*Activity Diagram*)

Diagram ini memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.

h. Diagram komponen (*Component Diagram*)

Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

i. Diagram deployment (*deployment diagram*)

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*).

Kesembilan diagram ini tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semuanya dibuat sesuai kebutuhan. Pada UML dimungkinkan kita menggunakan diagram-diagram lainnya misalnya data *flow* diagram, *entity relationship diagram*, dan sebagainya.

Menurut Nugroho (2010), sesungguhnya tidak ada batasan yang tegas diantara berbagai konsep dan konstruksi dalam UML, tetapi untuk menyederhanakannya, kita membagi sejumlah besar konsep dan dalam UML menjadi beberapa *view*. Suatu *view* pada dasarnya merupakan sejumlah konstruksi pemodelan UML yang merepresentasikan suatu aspek tertentu dari sistem atau perangkat lunak yang sedang kita kembangkan. Pada peringkat paling atas, *view-view* sesungguhnya dapat dibagi menjadi tiga area utama, yaitu: klasifikasi struktural (*structural classification*), perilaku dinamis (*dynamic behaviour*), serta pengolahan atau manajemen model (*model management*).

3. Sistem Informasi

Sistem adalah kumpulan dari beberapa elemen yang bekerjasama untuk mencapai suatu tujuan tertentu. Menurut Scott (1996) suatu hal dapat dikatakan sebagai sistem apabila terdiri dari unsur masukan (*input*), pengolahan (*processing*), serta keluaran (*output*). Di buku yang lain, Gapsperz (2002) menyebutkan ada empat ciri yang menandai suatu sistem, yaitu lingkungan, unsur-unsur, keterkaitan, dan memiliki fungsi dan tujuan yang sama. Banyak ahli yang memiliki deskripsi berbeda terkait konsep dari sebuah sistem. Secara ringkas, sistem dapat dijelaskan sebagai komponen-komponen yang saling berkaitan untuk mencapai tujuan yang sama, dan untuk mencapai tujuan tersebut sistem memiliki aturan tertentu, *input*, *output*, dan proses yang dapat mengubah *input* menjadi *output*.

Dalam dunia informatika, sistem informasi didefinisikan sebagai alat untuk menyajikan informasi dengan cara sedemikian rupa sehingga bermanfaat bagi penerimanya (Kertahadi, 1995).

Menurut Burch dan Grudnistki (1989), sistem informasi terdiri dari komponen-komponen berikut :

1. Blok Masukan, merupakan data yang masuk kedalam sistem informasi.
2. Blok Model, merupakan prosedur yang memanipulasi data input untuk menghasilkan data output.
3. Blok Keluaran, produk dari sistem informasi yang merupakan hasil dari pengolahan data input.
4. Blok Teknologi, merupakan tool box untuk menerima input, menjalankan model, menyimpan dan mengakses data.

5. Blok Database, kumpulan data yang saling berhubungan satu sama lain.
6. Blok Kendali, rancangan pengendalian untuk mencegah hal-hal yang dapat merusak system atau untuk mengatasi permasalahan yang terjadi pada system dengan cepat.

4. Kualitas Perangkat Lunak

Kualitas perangkat lunak ditentukan dari sudut pandang pemakai (*user*) apakah perangkat lunak yang dirancang telah sesuai dengan kebutuhannya. Crosby (1979) mendefinisikan kualitas atau mutu sebagai "*conformance to requirements*". Selama seseorang dapat berdebat tentang perbedaan antara kebutuhan, keinginan dan kemauannya, definisi kualitas harus mempertimbangkan perspektif pemakai tersebut. Suatu produk harus menyediakan fungsi yang dibutuhkan oleh pengguna. Jika memiliki kecacatan, maka produk tersebut perlu dipertimbangkan. Jika tingkat cacat minimumnya belum dicapai maka tidak ada yang perlu dipertimbangkan. Sesuatu yang berhubungan dengan pertimbangan dan penilaian cacat suatu produk perangkat lunak seperti halnya kegunaan, kecocokan, kemampuan, dan lainnya tergantung pada pemakai tersebut memandang dan menilainya termasuk didalamnya aplikasinya dan lingkungan software yang menyertainya (Humphrey, 1994).

Menurut IEEE, definisi kualitas perangkat lunak dibagi menjadi dua, yaitu:

1. Kemampuan sistem dapat memenuhi semua kebutuhan.
2. Kemampuan sistem dapat memenuhi kebutuhan pengguna

Perangkat lunak dikatakan berkualitas ketika banyak pengguna yang menggunakannya, dan dapat dikatakan gagal apabila pengguna meninggalkannya karena beberapa kesalahan atau kecacatan.

Menurut International Standards Organization (ISO), kualitas adalah “totalitas fitur dan karakteristik dari produk atau jasa yang menanggung pada kemampuannya untuk memuaskan kebutuhan yang ditentukan atau tersirat”.

Untuk menentukan kualitas suatu perangkat lunak, diperlukan standard yang bisa mengukur sejauh mana kualitas perangkat lunak tersebut. Terdapat beberapa model kualitas perangkat lunak dan pada setiap model memiliki faktor-faktor yang menjadi poin utama tersendiri.

Model kualitas perangkat lunak yang digunakan dalam penelitian ini adalah model ISO-25010. Model ini merupakan pengembangan dari model ISO-9126. Pada model ini terdapat karakteristik tambahan dan subkarakteristik yang dipindahkan ke karakteristik lain. Standard pengujian ini dianggap sebagai standard pengujian perangkat lunak yang sesuai dengan perubahan teknologi dan informasi saat ini. Karakteristik model ISO-25010 dapat dilihat pada tabel 2 berikut:

Tabel 2. Karakteristik model ISO-25010

Karakteristik	Sub karakteristik
<i>Functional Suitability</i>	<i>Functional completeness</i> <i>Functional correctness</i> <i>Functional appropriateness</i>
<i>Performance Efficiency</i>	<i>Time behavior</i> <i>Resource Utilization</i> <i>Capacity</i>
<i>Compatibility</i>	<i>Co-existence</i> <i>Interoperability</i>
<i>Usability</i>	<i>Appropriateness</i> <i>Recognisability</i> <i>Learnability</i> <i>Operability</i> <i>User Error Protection</i> <i>User Interface Aesthetics</i> <i>Accessibility</i>
<i>Reliability</i>	<i>Maturity</i> <i>Availability</i>

Sambungan tabel 2

	<i>Fault Tolerance</i> <i>Recoverability</i>
<i>Security</i>	<i>Confidentiality</i> <i>Integrity</i> <i>Non-repudiation</i> <i>Authenticity</i> <i>Accountability</i>
<i>Maintainability</i>	<i>Modularity</i> <i>Reusability</i> <i>Analyzability</i> <i>Changeability</i> <i>Modifiability</i> <i>Testability</i>
<i>Portability</i>	<i>Adaptability</i> <i>Installability</i> <i>Replaceability</i>

Berikut ini masing-masing karakteristik yang ada pada model ISO-25010 yang digunakan untuk mengukur kualitas perangkat lunak yang diteliti, di antaranya:

a. *Functional suitability*

Karakteristik ini mempresentasikan tentang sejauh mana suatu sistem menyediakan fungsi-fungsi yang telah ditetapkan dan memenuhi kebutuhan. Karakteristik ini terdiri dari sub karakteristik berikut:

1. *Functional completeness*, sejauh mana fungsi dapat mencakup semua tugas dan tujuan pengguna.
2. *Functional correctness*, sejauh mana sistem menyediakan hasil yang benar sesuai dengan kebutuhan.
3. *Functional appropriateness*, sejauh mana fungsi dapat memfasilitasi untuk menyelesaikan tugas tertentu.

Menurut Fatkhurrohman yang dikutip dari Zyrmak (2001), *functionality* adalah kemampuan perangkat lunak berfokus pada kesesuaian satu set fungsi untuk dapat melakukan tugas tertentu.

Pengukuran karakteristik ini dilakukan oleh ahli pemrograman dengan rumus dari matriks Feature Completeness (Acharya dan Sinha, 2013). Berikut rumusnya:

$$X = \frac{I}{P}$$

P = jumlah fungsi yang dirancang

I = jumlah fungsi yang berhasil diimplementasikan

Dalam matriks Feature Completeness, nilai yang mendekati 1 mengindikasikan banyaknya fitur yang berhasil diimplementasikan. Hasil diukur dalam skala $0 \leq X \leq 1$. Perangkat lunak dikatakan baik dalam *functional suitability* jika X mendekati nilai 1.

b. *Performance Efficiency*

Karakteristik ini mempresentasikan kinerja relatif terhadap jumlah sumber daya yang digunakan dibawah ketentuan yang ditetapkan. Menurut McCall yang dikutip Roger S. Pressman (2010), *efficiency* adalah jumlah sumberdaya komputasi dan kode yang diperlukan program untuk mampu melaksanakan fungsinya secara baik dan benar.

Pengukuran aspek ini dilakukan menggunakan web tool GTmetrix. GTmetrix merupakan tool yang dikembangkan oleh Gossamer Threads, sebuah perusahaan Vancouver yang memiliki pengalaman lebih dari 16 tahun dalam bidang teknologi website. GTmetrix digunakan untuk mengukur performa dari website serta

memberikan rekomendasi untuk perbaikan performa website. Menurut Jean Galea (2012), GTmetrix dapat membantu untuk mengembangkan lebih cepat dan lebih efisien.

Menurut Jakob Nielsen (2010), waktu maksimal seorang pengguna untuk berfokus pada website yang sedang diaksesnya, maksimal 10 detik. Sehingga perangkat lunak dikatakan baik jika memiliki *response time* kurang dari 10 detik.

c. *Compatibility*

Menurut Anne Mette Hass (2014), *compatibility testing* dilakukan dengan tujuan untuk mengukur sejauh mana produk kompatibel dengan produk lain dalam suatu lingkungan. Karakteristik ini memiliki 2 subkarakteristik, yaitu *coexistence* dan *interoperability*. *Coexistence* menginterpretasikan sejauh mana produk *software* bisa berdampingan dengan produk lain dengan menggunakan *resource* yang sama. *Interoperability* menginterpretasikan sejauh mana produk *software* bisa berinteraksi atau terintegrasi dengan *software* lainnya.

Pengujian *compatibility* dilakukan untuk mengevaluasi kompatibilitas *software* dengan lingkungan. Lingkungan yang dimaksud berisi beberapa unsur di bawah ini:

- Sistem Operasi lain
- *Database* lain
- Jaringan *Hardware* dan *Platform* lain
- *Software* sistem lain
- *Browser* lain

Pengujian sistem ini hanya bisa dilakukan dengan mengujinya menggunakan sistem operasi lain, *database* lain, *hardware* dan *platform* lain, *software* lain, dan

browser lain. Dikarenakan tidak ada *software* pembandingnya untuk mengukur kompatibilitas, maka pengujian aspek ini tidak bisa dilakukan.

d. *Usability*

Karakteristik ini mempresentasikan sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektif, efisien dan kepuasan dalam konteks tertentu. Menurut Jacob Nielsen (1993), *usability* adalah atribut kualitas yang menilai betapa mudahnya *user interface* dari perangkat yang digunakan.

Pengukuran ini dilakukan dengan mengukur komponen *usefulness*, *ease of use*, *easy of learning*, dan *satisfaction* yang dikenal dengan *USE Questionnaire*. *USE Questionnaire* ini dikembangkan oleh Arnold M. Lund yang terdiri dari 30 pernyataan. Untuk skala kuesionernya menggunakan Skala Likert. Menurut Rensis Likert (1932), sikap dapat diukur dan intensitas suatu pengalaman adalah linear yaitu duduk di sebuah kontinum dari sangat setuju sampai sangat tidak setuju. Menurut skala Likert, sistem informasi dikatakan layak jika persentasenya lebih dari 61%. Hal tersebut bisa dilihat pada tabel kriteria skor di bawah ini:

Tabel 3. Kriteria skor pengujian *usability*

Presentase Pencapaian (%)	Interpretasi
0% - 20%	Sangat Tidak Layak
21% - 40%	Kurang Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

e. *Reliability*

Karakteristik ini mempresentasikan sejauh mana sistem, produk atau komponen fungsi dalam kondisi tertentu mempunyai kinerja yang ditetapkan

untuk jangka waktu tertentu. Menurut Pao (1999), *reliability* adalah peluang kegagalan perangkat lunak untuk jangka waktu tertentu di lingkungan tertentu. Pengukuran ini dilakukan menggunakan web tool LoadImpact. LoadImpact menguji beban server atau website (*load testing*) yang nanti hasilnya bisa dijadikan tolak ukur ketahanan website. LoadImpact akan memeriksa seberapa baik perangkat lunak atau website berjalan di bawah beban kerja. Pengukuran karakteristik ini diukur menggunakan Model Nelson dengan rumus sebagai berikut:

$$R1 = 1 - \frac{ne}{n}$$

Keterangan:

R1 = nilai reliability

ne = jumlah input yang gagal

n = jumlah input

Menurut Asthana & Oliverly (2009) dalam standard Telcordia, hasil pengujian dikatakan memenuhi aspek *reliability* jika presentase bernilai minimal 95%.

f. *Security*

Karakteristik ini mempresentasikan sejauh mana suatu produk atau sistem melindungi informasi dan data sehingga orang atau produk atau sistem lainnya memiliki tingkat akses data yang sesuai dengan jenis dan tingkat otorisasi mereka. *Software* dikatakan memiliki sekuritas yang baik jika bisa mencegah akses yang tidak sah, baik sengaja maupun tidak sengaja (Zyrmiak, 2010).

Menurut Web Application Security Consortium yang dikutip Fatkhurrohman, celah keamanan yang paling sering dieksploitasi adalah Cross Site Scripting (XSS) dan SQL Injection. Dalam pengujian aspek *security* dan mendeteksi

tingkat sekuritas perangkat lunak, dapat menggunakan perangkat lunak Acunetix Web Vulnerability Scanner (Vieira, 2009).

g. *Maintainability*

Karakteristik ini mempresentasikan tingkat efektivitas dan efisiensi produk atau sistem dapat dimodifikasi untuk memperbaikinya, memperbaikinya atau beradaptasi dengan perubahan lingkungan. Menurut IEEE, *maintainability* adalah kemudahan perangkat lunak untuk dimodifikasi, diperbaiki, ataupun dikembangkan di lingkungan berbeda.

Pengujian ini dilakukan dengan menghitung duplikasi kode yang ada di dalam program. Menurut S. Ducasse, M. Rieger, dan S. Demeyer (1999), duplikasi kode adalah salah satu faktor yang sangat mempersulit pemeliharaan dan perubahan sistem perangkat lunak. Artinya, semakin sedikit duplikasi kode, maka semakin baik *maintainability* dari perangkat lunak. Indeks *maintainability* dihitung dengan rumus tertentu dari *lines-of-code*, McCabe dan Halstead. Menurut TIOBE Quality Indicator, hasil dari pengujian duplikasi kode dihitung menggunakan rumus:

$$Skor_d = \min(-30 \times \log_{10}(code_duplication) + 60, 100)$$

Dari perhitungan, semakin sedikit nilai duplikasi kodenya, maka akan semakin tinggi TQI Score, dan akan semakin baik aspek *maintainability* dari perangkat lunak. TQI Score dapat dilihat pada tabel 4 berikut ini.

Tabel 4. TQI Score Untuk Standard Aspek *Maintainability*

Code Duplication	TQI Score (<i>Skor_d</i>)	Grade	Interpretasi
<= 0.10 %	>= 90 %	A	Sangat Baik
<= 0.22 %	>= 80 %	B	Baik
<= 0.46 %	>= 70 %	C	Cukup Baik
<= 2.15 %	>= 50 %	D	Cukup
<= 4.64 %	>= 40 %	E	Lemah
> 4.64 %	< 40 %	F	Sangat Lemah

h. *Portability*

Karakteristik ini mempresentasikan tingkat efektivitas dan efisiensi sistem, produk atau komponen dapat ditransfer dari satu perangkat keras, perangkat lunak atau lingkungan operasional atau penggunaan lain ke yang lain.

Peter J. Brown dalam buku Ville Salonen (2012) mendefinisikan bahwa software dikatakan *portable* jika bisa dijalankan pada komputer lain, sehingga perangkat lunak yang sama bisa dijalankan pada banyak komputer yang berbeda. *Software* idealnya bisa dipindahkan antar lingkungan tanpa modifikasi kode sumber. Software dikatakan *portable* jika bisa diakses menggunakan *browser* yang berbeda, dari *browser* desktop maupun *browser mobile* (Ville Salonen, 2012).

5. Penggunaan Sistem Informasi pada Perguruan Tinggi

Penggunaan teknologi informasi dan komunikasi membuat kinerja organisasi lebih efektif, efisien dan kompetitif (Akadun, 2009). Oleh karena itu sistem informasi diharapkan dapat menjadi salah satu hal yang harusnya dimiliki oleh perguruan tinggi. Dengan adanya sistem informasi, diharapkan perguruan tinggi dapat menyediakan layanan informasi dengan baik dan efisien kepada publik atau civitas akademinya.

Untuk memberikan pelayanan yang baik kepada mahasiswa dan dosen, tidak sedikit perguruan tinggi membuat sistem informasi terintegrasi atau sistem informasi akademik. Hal tersebut bertujuan untuk meningkatkan kinerja, kualitas layanan, daya saing, dan membantu dalam pengelolaan data mahasiswa, dosen dan karyawan di perguruan tinggi yang bersangkutan.

6. Analisis dan Perancangan Sistem Informasi

Kesuksesan suatu sistem informasi ditentukan dari analisis dan perancangan yang baik. Tahap analisis menentukan masalah apa yang dihadapi dan harus diselesaikan, serta membatasi seberapa besar lingkup yang hendak diselesaikan. Tahap perancangan akan menentukan seperti apa sistem akan berfungsi untuk menyelesaikan permasalahan yang telah dianalisa. Oleh karena itu, baik analisis maupun perancangan akan sangat menentukan bagaimana sistem akan bekerja.

Sistem informasi yang baik dihasilkan dari pengembangan sistem dengan metode yang terstandarisasi. Salah satunya adalah *System Development Life Cycle* (SDLC). Menurut Wahyono (2004), *system development life cycle* adalah model konseptual yang digunakan dalam manajemen proyek yang menggambarkan tahap-tahap yang terlibat dalam proyek pengembangan sistem informasi, dari studi kelayakan awal melalui pemeliharaan aplikasi selesai. Langkah-langkah pada SDLC meliputi:

1. Mendefinisikan Masalah

Mengidentifikasi proyek-proyek yang potensial, seberapa besar keuntungan yang diperoleh, durasi waktu yang tersedia, dan apakah sumber daya yang anda mampu menyelesaikan proyek. Tahapan ini menghasilkan laporan kelayakan

yang berisi definisi masalah dan rangkuman tujuan yang ingin dicapai dari proyek yang dipilih.

2. Inisiasi dan Perencanaan

Pada tahapan ini, analis menyusun rencana kerja yang matang untuk menjalankan tahapan-tahapan selanjutnya. Penentuan rencana kerja secara detail, durasi yang diperlukan untuk masing-masing tahapan, maupun sumber daya manusianya. Hasil dari tahap ini adalah jadwal pelaksanaan proyek sistem informasi yang dikembangkan.

3. Analisis

Pada tahapan analisis terdapat langka-langkah yang harus diselesaikan oleh seorang pengembang, di antaranya:

- a) Langkah awal yang dilakukan pada tahap analisis adalah mengumpulkan informasi tentang permasalahan yang terjadi. Kemudian menentukan batasan masalah, yaitu permasalahan apa saja yang dapat diselesaikan dengan sistem informasi. Apabila sebelumnya telah ada sistem untuk permasalahan tersebut, mengidentifikasi kelemahan pada sistem sebelumnya dan memperbaiki dengan sistem baru.
- b) Setelah menemukan kelemahan pada sistem sebelumnya, langkah selanjutnya adalah *system requirement*, mendefinisikan apa saja sebenarnya yang dibutuhkan oleh sistem lama ataupun sistem baru untuk mengatasi masalahnya.
- c) Dari tahap pengumpulan informasi akan didapatkan banyak sekali informasi terkait kebutuhan yang diperlukan. Namun dalam beberapa kasus, ketersediaan waktu dan sumber daya untuk menyelesaikan semua

requirement tidak mencukupi. Oleh karena itu maka seorang pengembang harus memprioritaskan kebutuhan yang dianggap kritis dan penting.

- d) Setelah menyusun dan mengevaluasi kebutuhan, seorang pengembang harus menyiapkan rencana kedua atau alternatif apabila nantinya susunan kebutuhannya tidak diterima klien.
- e) Mengulas kebutuhan yang telah dianalisa dengan pihak klien agar nantinya tidak terjadi kesalahan ketika masuk pada tahap selanjutnya

4. Desain

Tahap Desain merupakan tahapan mengubah kebutuhan yang masih berupa konsep menjadi spesifikasi yang lebih riil. Pada tahapan desain, terdapat beberapa aktifitas yang harus dilakukan, di antaranya:

- a) Merancang dan mengintegrasikan komponen
- b) Merancang arsitektur aplikasi
- c) Membuat desain *user interface*
- d) Mendesain *system interface*
- e) Mendesain dan mengintegrasikan *database*
- f) Membuat *prototype* untuk detail dari desain
- g) Mendesain dan mengintegrasikan kendali system

5. Implementasi

Pada tahapan ini terdapat beberapa hal yang harus dilakukan, yaitu:

- a) Testing, pengujian hasil kode program yang telah dibuat. Tujuannya adalah untuk menjamin bahwa kode program yang dibuat dapat dijalankan dengan baik dan tidak ada *error*, serta program yang dibuat harus mampu menyelesaikan masalah dan mudah untuk digunakan oleh klien.

b) Instalasi pada klien atau pengguna

Output dari tahapan ini berupa *source code* yang *error free*, prosedur penggunaan, dan buku panduan.

6. Pemeliharaan (*Maintenance*)

Pada tahap ini, secara sistematis dan berkala, sistem diperbaiki dan ditingkatkan. Perbaikan yang dilakukan seperti penambahan modul, fungsi ataupun perbaikan *usability* yang dapat memberikan kemudahan kepada klien/pengguna dalam menjalankan program.

B. Hasil Penelitian yang Relevan

1. Hasil penelitian tentang penerapan konsultasi bimbingan secara *online* berbasis web dengan menggunakan bahasa pemrograman php dan *database* MySQL pada kampus STIE-KBP Padang. Penelitian bertujuan untuk menguji aplikasi penerapan konsultasi bimbingan skripsi di kampus STIE-KBP. Kesimpulan yang didapat adalah sistem informasi bimbingan skripsi secara online dapat membantu mahasiswa dan dosen dalam meluangkan banyak waktu untuk konsultasi, tanpa mengganggu kegiatan yang sangat penting. Rancangan sistem tersebut memiliki kelebihan berupa kemudahan dalam melakukan *entry data*, pencarian, dan *update* data serta memperkecil kesalahan dalam pengolahan data dan pembuatan laporan yang cepat, akurat dan relevan, sehingga membantu dalam meningkatkan efisiensi dan efektifitas kerja.
2. Hasil penelitian SIPINTAR tentang penerapan sistem bimbingan skripsi dan tugas akhir secara *online* pada Jurusan Teknik Mesin ITB. Penelitian tersebut

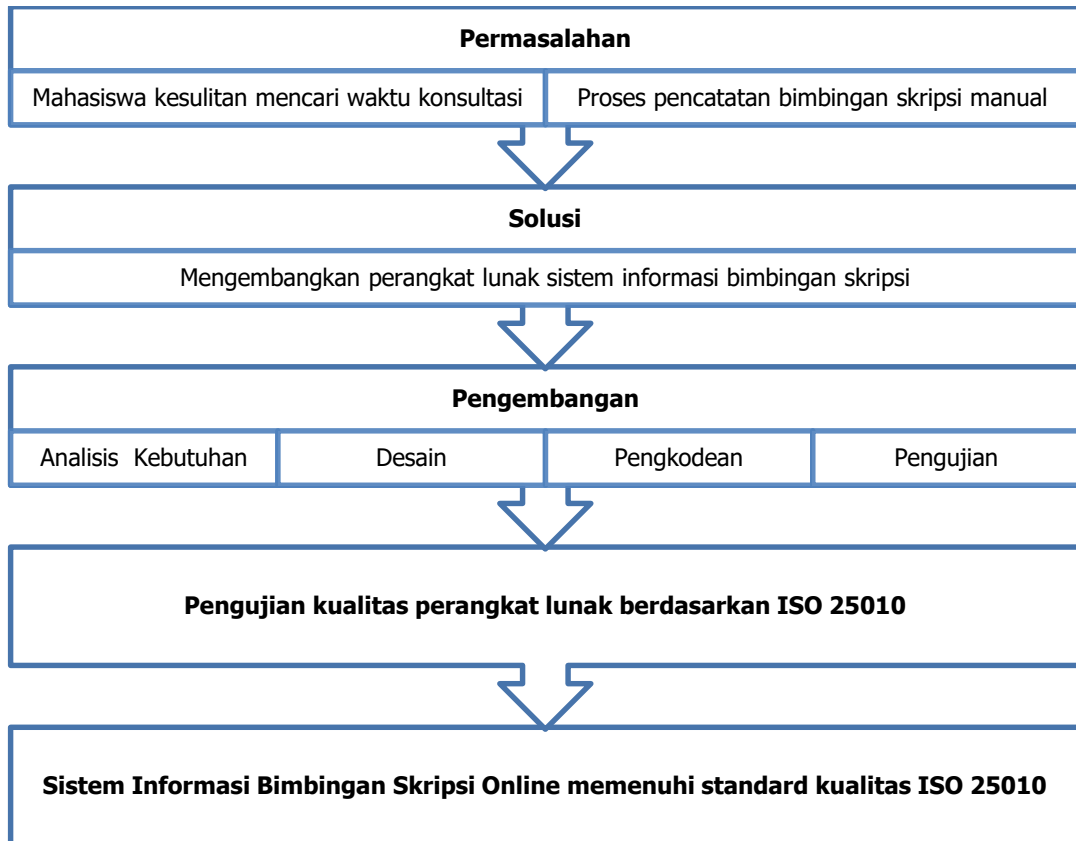
menunjukkan hasil bahwa lama studi mahasiswa telah berkurang dari sebelum penggunaan sistem bimbingan skripsi *online*.

3. Untuk meningkatkan mutu dan kualitas unit dan tenaga pendidikan maupun tenaga kependidikan, BPM Universitas Negeri Semarang sejak tahun 2013 telah mengembangkan Sistem Bimbingan Mahasiswa (SiBiMa). Sistem ini diharapkan dapat membantu mahasiswa, dosen, auditor dalam menyelesaikan bimbingan skripsi dan meningkatkan mutu dan kualitas unit serta tenaga pendidik dan tenaga kependidikan di Universitas Negeri Semarang.
4. Penelitian sistem informasi bimbingan tugas akhir pada Sekolah Vokasi Universitas Gadjah Mada, hal tersebut bertujuan agar kebutuhan pengelolaan data tugas akhir dapat terpenuhi dan meminimalisir masalah-masalah yang menyebabkan ketidakefektifan proses bimbingan tugas akhir.

C. Kerangka Pikir

Sistem informasi Bimbingan Tugas Akhir dan Skripsi *Online* ini diharapkan akan memudahkan dosen memantau pengerjaan tugas akhir mahasiswa bimbingannya. Sistem informasi ini sebagai wadah bagi mahasiswa dan dosen pembimbing untuk saling membantu demi lancarnya proses bimbingan tugas akhir dan skripsi. Dengan sistem informasi ini, proses pencatatan bimbingan dilakukan secara *online*, agar admin atau koordinator skripsi dapat mengecek rekam jejak proses bimbingan skripsi mahasiswa. Apabila dosen dan mahasiswa tidak dapat melakukan bimbingan dengan bertatap muka secara langsung, mahasiswa dapat mengirimkan *file* progress laporannya kepada dosen pembimbing, kemudian dosen pembimbing dapat mengecek file yang dikirimkan

oleh mahasiswa dan dosen dapat memberi *feedback* atau koreksi kepada mahasiswa. Kerangka pikir dalam penelitian ini dapat dilihat pada gambar 1 berikut:



Gambar 1. Kerangka Pikir

BAB III

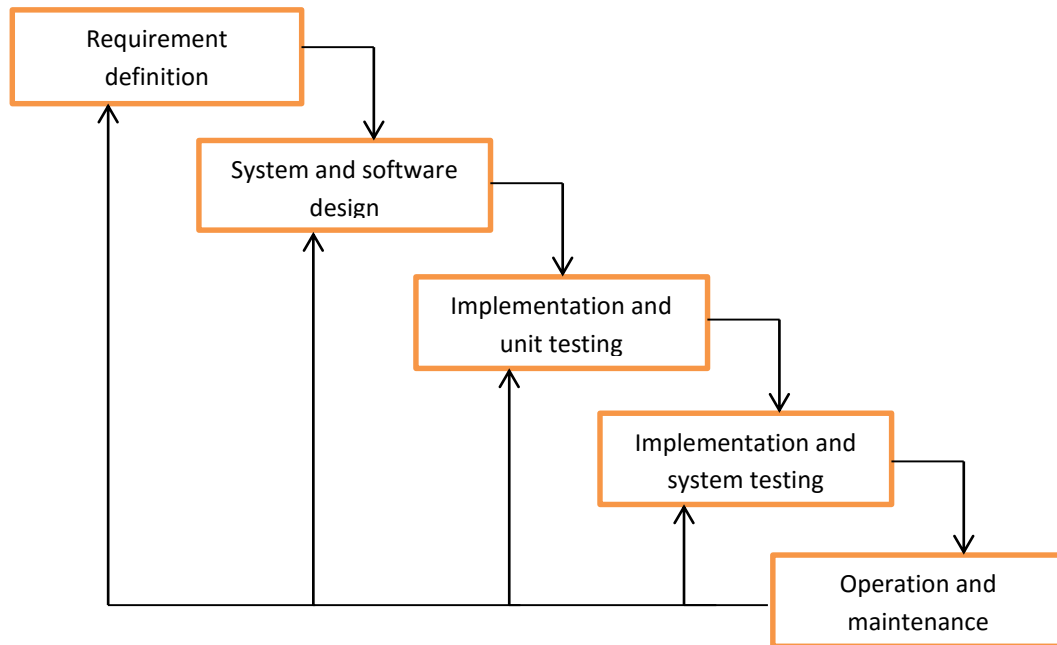
METODE PENELITIAN

A. Desain Penelitian

Pada penelitian ini penulis menggunakan metode penelitian *research and development* (RND). Menurut Sugiyono (2009) metode penelitian *research and development* adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut. Produk yang dimaksud tidak selalu berbentuk benda atau suatu alam, namun dapat juga berupa suatu perangkat lunak. Metode penelitian tersebut penulis pilih karena pada penelitian ini penulis hendak mengembangkan suatu produk berupa perangkat lunak yang memiliki standar kualitas sesuai dengan ketentuan pada pengembangan perangkat lunak. Perangkat lunak yang dimaksud adalah suatu sistem informasi bimbingan skripsi secara online untuk mahasiswa tingkat akhir.

Penelitian ini menggunakan metode *System Development Life Cycle* (SDLC). Metode tersebut penulis pilih karena metode ini menyediakan tahapan yang dapat digunakan sebagai pedoman mengembangkan sistem. Selain itu, metode ini akan memberikan hasil sistem yang lebih baik karena sistem analisis dan perancangan secara keseluruhan dilakukan sebelum diimplementasikan. Menurut Jogiyanto (2013), SDLC memiliki beberapa tahapan, yaitu analisis sistem, perancangan sistem, implementasi sistem, operasi dan perawatan sistem. Dikarenakan metode ini hanya menyediakan tahapan saja, tanpa menyediakan cara dan alat untuk mengembangkan sistem, maka sistem ini harus digabungkan dengan model yang ada, dan dalam pengembangan ini penulis menggunakan model *Waterfall* (air terjun).

Menurut referensi Sommerfile, model *waterfall* dapat digambarkan seperti Gambar 2 berikut ini :



Gambar 2. Model *Waterfall* dari Sommerfile

Menurut Pressman (2010), model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Model ini melakukan pendekatan secara sistematis dan berurutan bagi pengembangan perangkat lunak. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan dilakukan secara berurutan. Langkah-langkah yang harus dilakukan pada model *Waterfall* adalah sebagai berikut:

1. Analisis Kebutuhan Perangkat Lunak (*Software Requirement Analysis*)

Langkah ini merupakan tahapan mengumpulkan data dan informasi untuk mendapatkan kebutuhan perangkat lunak yang dikembangkan. Tahapan ini

bertujuan untuk mengetahui dan menggali informasi seperti apa perangkat lunak yang dibutuhkan oleh *user*. Hasil dari tahap ini berupa *user requirement* atau dapat disebut sebagai data yang berhubungan dengan keinginan *user*. Dokumen ini yang akan menjadi acuan penulis untuk mengembangkan perangkat lunak.

Proses pengumpulan informasinya dilakukan dengan cara observasi dan wawancara. Observasi dilakukan secara langsung selama menjadi mahasiswa Universitas Negeri Yogyakarta. Wawancara dilakukan secara langsung kepada mahasiswa tingkat akhir Pendidikan Teknik Informatika dan Dosen di Jurusan Pendidikan Teknik Elektronika Universitas Negeri Yogyakarta.

2. Perancangan (*Design*)

Langkah ini akan menterjemahkan hasil dari analisis kebutuhan sistem menjadi sebuah rancangan perangkat lunak sebelum dilakukan proses penulisan kode (*coding*). Penulis menggambarkan rancangan sistem informasinya dalam sebuah pemodelan. Dan dalam hal ini penulis menggunakan pemodelan *Unified Modeling Language* (UML). Untuk basis data penulis menggunakan pemodelan *Entity Relationship Diagram* (ERD). Pemodelan basis data ERD dipilih oleh penulis karena penulis akan menggunakan data yang saling berkaitan dan berelasi untuk basis datanya. Selain itu, penulis juga membuat rancangan antarmuka (*user interface*) perangkat lunak yang hendak dibuat.

3. Pengkodean (*Coding*)

Pengkodean atau yang sering disebut dengan coding merupakan tahapan untuk mengimplementasikan hasil dari analisis kebutuhan dan rancangan menjadi suatu perangkat lunak. Tahapan ini mengimplementasikan hasil rancangan ke dalam baris-baris kode dan tampilan program yang berjalan sesuai

dengan fungsinya. Perancangan diimplementasikan menggunakan bahasa pemrograman PHP dan *Bootstrap*.

4. Pengujian (*Testing*)

Dalam tahapan pengujian (*testing*), pengujian dilakukan dengan *whitebox testing* dan *blackbox testing*. *Whitebox testing* dilakukan dengan melihat *source code* dari program, yaitu dengan menguji setiap modul yang bekerja di dalam sistem. Sedangkan *blackbox testing* dilakukan dengan dengan menguji fungsionalitas tanpa melihat *source code* dari program, yaitu dengan melakukan uji *checklist* pada setiap fungsi yang dijelaskan pada analisis kualitas aspek *functional suitability*.

5. Pemeliharaan (*Maintenance*)

Perangkat lunak yang telah disampaikan kepada *user* pasti akan mengalami perubahan. Hal tersebut bisa dikarenakan oleh perangkat lunak yang dibuat harus menyesuaikan dengan lingkungan baru, atau karena *user* membutuhkan perkembangan fungsional. Pemeliharaan suatu perangkat lunak diperlukan, karena perangkat lunak yang dibuat tidak selamanya berjalan dengan lancar, dan kemungkinan masih ada error kecil yang tidak ditemukan sebelumnya atau ada perubahan/penambahan fitur yang belum ada pada perangkat lunak.

B. Analisis Kualitas Perangkat Lunak

Analisis kualitas dari perangkat lunak dilakukan dengan pengujian produk yang berprinsip pada standar ISO 25010. Standar ISO 25010 terdiri dari 8 aspek kualitas penilaian, di antaranya: *Functional suitability*, *Performance efficiency*, *Compatibility*, *Usability*, *Reliability*, *Security*, *Maintainability*, *Portability*.

a) Pengujian *functional suitability*

Pengujian *functional suitability* ini merupakan pengujian dengan metode *checklist* pada *test case* yang berisi fungsi pada sistem informasi. Tujuannya untuk memastikan bahwa tidak ada error dalam program, dan jika ditemukan error, maka dapat segera diperbaiki (Agarwal, Taul, & Gupta, 2010). Pengujian ini dilakukan oleh responden ahli di bidang pemrograman dalam pengembangan aplikasi sistem informasi.

b) Pengujian *performance efficiency*

Pengujian *performance* ini dilakukan untuk mengukur karakteristik performa dari komponen aplikasi. Aspek ini merupakan aspek untuk mengukur keandalan sistem informasi yang digunakan.

c) Pengujian *compatibility*

Pengujian *compatibility* ini merupakan testing yang dilakukan dengan menjalankan produk, sistem atau komponen dengan perangkat keras dan/atau perangkat lunak dalam suatu sumber daya yang sama.

d) Pengujian *usability*

Pengujian *usability* yang dilakukan dengan menggunakan angket *USE Questionnaire* oleh Arnold M. Lund (2001). *Instrumen USE Questionnaire* telah digunakan dalam berbagai penelitian, sehingga telah teruji kevalidannya. Instrumen yang digunakan berjumlah pernyataan-pernyataan yang dibagi menjadi 4 kriteria, yaitu *usefulness*, *easy of use*, *easy of learning*, dan *satisfaction*.

e) Pengujian *reliability*

Pengujian *reliability* dilakukan untuk mengetahui seberapa jauh hasil pengukuran dua kali atau lebih terhadap gejala yang sama dengan menggunakan alat pengukur yang sama (Sugiono, 2012).

f) Pengujian *security*

Pengujian *security* dilakukan untuk menguji keamanan aplikasi web terhadap serangan atau *attack* kepada sistem. Pengujian ini menggunakan *tools* Acunetix Web Vulnerability Scanner.

g) Pengujian *maintainability*

Pengujian *maintainability* dilakukan untuk menguji efektifitas dan efisiensi perangkat lunak untuk dimodifikasi atau dikembangkan. Pengujian ini dilakukan menggunakan alat tertentu.

h) Pengujian *portability*

Pengujian *portability* menggunakan berbagai *browser* dan sistem operasi untuk menguji kualitas perangkat lunak saat berjalan pada lingkungan perangkat lunak yang berbeda.

C. Subjek Penelitian

Subjek penelitian digunakan dengan tujuan untuk menguji aspek *functional suitability* dan *usability* dari perangkat lunak. Pengujian *functional suitability* menggunakan 2 responden ahli di bidang sistem informasi. Untuk pengujian *usability* menggunakan 20 responden yang mengacu pada Jakob Nielsen (2012). Untuk pengujian kuantitatif, setidaknya dibutuhkan 20 responden agar mendapatkan angka yang signifikan secara statistik, selain itu interval kepercayaan juga cukup ketat (Nielsen, 2012). Pengguna dalam penelitian

diambil dari mahasiswa dan dosen Jurusan Pendidikan Teknik Elektronika Fakultas Teknik Universitas Negeri Yogyakarta.

D. Metode Pengumpulan Data

Teknik pengumpulan data yang dilakukan untuk tahapan analisis kebutuhan dan pengujian adalah sebagai berikut:

1. Observasi

Observasi adalah melakukan pengamatan secara langsung kepada obyek penelitian untuk menganalisis kualitas perangkat lunak yang dikembangkan.

2. Kuesioner / Angket

Teknik ini merupakan teknik pengumpulan data yang dilakukan dengan memberi pertanyaan atau pernyataan kepada responden secara tertulis untuk dijawabnya.

3. Wawancara

Teknik ini merupakan teknik pengumpulan data dengan memberi beberapa pertanyaan kepada calon user untuk menentukan kebutuhan awal sistem dan menguatkan hasil kuesioner.

4. Software Pengukuran

Software pengukuran digunakan untuk melakukan pengukuran beberapa aspek. Software yang digunakan antara lain:

- a. GT Metrix, software ini digunakan untuk mengukur aspek *performance efficiency*.
- b. LoadImpact, aplikasi ini digunakan untuk mengukur aspek *reliability*.
- c. Acunetix Web Vulnerability Scanner 8, digunakan untuk mengukur aspek *security*.

d. PHP Copy/Paste Detector, digunakan untuk mengukur aspek *maintainability*.

E. Instrumen Penelitian

Instrumen penelitian terdiri dari instrumen untuk pengujian perangkat lunak berdasarkan aspek *functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, dan portability*.

1. Instrumen *Functional Suitability*

Instrumen ini digunakan untuk mengukur sejauh mana suatu sistem menyediakan fungsi-fungsi yang telah ditetapkan. Instrumen ini berupa *checklist* pada *test case* yang berisi daftar fungsi sistem informasi yang dijabarkan sesuai dengan analisis kebutuhan fungsional. Pengujian *test case* dilakukan oleh responden ahli di bidang pemrograman. *Test case* yang digunakan dalam penelitian dilampirkan di tabel 5 dan 6 berikut:

Tabel 5. Instrumen *Functional Completeness* dan *Functional Appropriateness*

No.	Fungsi	Hasil yang Diharapkan
I.	Admin	
1.	Login	Fungsi untuk login sudah berjalan dengan benar
2.	Edit Password	Fungsi mengedit password sudah berjalan dengan benar
3.	Logout	Fungsi untuk logout sudah berjalan dengan benar
4.	Data Bimbingan	Fungsi melihat data bimbingan mahasiswa dan dosen sudah berjalan dengan benar
5.	Memilih dosen	Fungsi untuk menentukan dosen pembimbing bagi mahasiswa sudah berjalan dengan benar
6.	Data Konsultasi	Fungsi melihat data konsultasi online mahasiswa dan dosen sudah berjalan dengan benar
7.	Data Mahasiswa	Fungsi menambahkan, menampilkan, mengubah, dan menghapus data mahasiswa sudah berjalan dengan benar
8.	Data Dosen	Fungsi menambahkan, menampilkan, mengubah, dan menghapus data dosen sudah berjalan dengan benar

Sambungan dari Tabel 5

No.	Fungsi	Hasil yang Diharapkan
9.	Data Prodi	Fungsi untuk menambahkan, menampilkan, mengubah, dan menghapus data prodi sudah berjalan dengan benar
10.	Data Jurusan	Fungsi untuk menambahkan, menampilkan, mengubah, dan menghapus data jurusan sudah berjalan dengan benar
11.	Data Pembimbing	Fungsi menampilkan dan mencetak, serta merekap data proses bimbingan sudah berjalan dengan benar
II.	Mahasiswa	
12.	Daftar bimbingan skripsi	Fungsi untuk mendaftar bimbingan sudah berjalan dengan benar
13.	Login	Fungsi untuk login sudah berjalan dengan benar
14.	Logout	Fungsi untuk logout sudah berjalan dengan benar
15.	Edit password	Fungsi untuk mengedit password sudah berjalan dengan benar
16.	Data Profil	Fungsi untuk menampilkan profil dan mengedit profil sudah berjalan dengan benar
17.	Data bimbingan	Fungsi menampilkan, menambahkan, dan mengubah catatan bimbingan sudah berjalan dengan benar
18.	Data konsultasi Online	Fungsi menambahkan, mengedit, mengirim, mendownload data konsultasi sudah berjalan dengan benar
19.	Data Pesan	Fungsi untuk mengirim, menampilkan, dan menerima pesan sudah berjalan dengan benar
III.	Dosen	
20.	Daftar akun	Fungsi untuk mendaftar akun telah berjalan dengan benar
21.	Login	Fungsi untuk login sudah berjalan dengan benar
22.	Logout	Fungsi untuk logout sudah berjalan dengan benar
23.	Edit password	Fungsi mengedit password sudah berjalan dengan benar
24.	Data Profil	Fungsi untuk menampilkan profil dan mengedit profil sudah berjalan dengan benar
25.	Data Bimbingan	Fungsi menambah, mengubah, dan menampilkan catatan bimbingan sudah berjalan dengan benar

Sambungan dari Tabel 5

No.	Fungsi	Hasil yang Diharapkan
26.	Data Konsultasi	Fungsi untuk menambahkan, mengedit, mengirim, dan mendownload data konsultasi sudah berjalan benar
27.	Data Pesan	Fungsi mengirim, menerima, dan melihat pesan sudah berjalan dengan benar
28.	Data Mahasiswa	Fungsi melihat data mahasiswa dan menampilkan detail mahasiswa sudah berjalan dengan benar

Tabel 6. Instrumen *Functional Correctness*

No.	Fungsi	Hasil yang Diharapkan
1.	Identifikasi username dan password	Fungsi untuk masuk sistem sesuai dengan jenis pengguna sudah berjalan dengan benar
2	Menampilkan data mahasiswa	Fungsi untuk menampilkan data mahasiswa sudah berjalan dengan benar
3	Menampilkan data dosen	Fungsi untuk menampilkan data dosen sudah berjalan dengan benar
4	Menampilkan data pembimbing	Fungsi untuk menampilkan data pembimbing sudah berjalan dengan benar
5	Mencetak data pembimbing	Fungsi untuk mencetak data pembimbing sudah berjalan dengan benar
6	Menampilkan data prodi	Fungsi untuk menampilkan data prodi sudah berjalan dengan benar
7	Menampilkan data jurusan	Fungsi untuk menampilkan data jurusan sudah berjalan dengan benar
8	Menampilkan data mahasiswa berdasarkan pencarian	Fungsi untuk menampilkan data mahasiswa berdasarkan pencarian sudah berjalan dengan benar
9	Menampilkan data pembimbing berdasarkan pencarian	Fungsi untuk menampilkan data pembimbing berdasarkan pencarian sudah berjalan dengan benar
10	Menampilkan data profil	Fungsi untuk menampilkan data profil berdasarkan parameter sudah berjalan dengan benar
11	Menampilkan data konsultasi	Fungsi untuk menampilkan data konsultasi sudah berjalan dengan benar
12	Menampilkan data bimbingan	Fungsi untuk menampilkan data bimbingan sudah

	bimbingan	berjalan dengan benar
13	Menampilkan data mahasiswa bimbingan	Fungsi untuk menampilkan data mahasiswa bimbingan sudah berjalan dengan benar
14	Menampilkan pesan	Fungsi untuk menampilkan pesan sudah berjalan dengan benar

2. Instrumen *Performance Efficiency*

Instrumen performance efficiency menggunakan aplikasi GT Metrix, dimana aplikasi tersebut menghasilkan dua pengujian, yaitu berdasarkan aturan YSlow dan PageSpeed. Aturan YSlow ini dikembangkan oleh Yahoo Developer Network, dan PageSpeed dikembangkan oleh Google.

3. Instrumen *Compatibility*

Pengujian aspek *compatibility* tidak dilakukan, dikarenakan tidak ada *software* atau aplikasi pembandingnya, sehingga seluruh sumber daya dari server hanya digunakan untuk sistem informasi.

4. Instrumen *Usability*

Pengujian *usability* dilakukan dengan menggunakan angket USE Questionnaire oleh Arnold M. Lund (2001) yang berjumlah 30 pernyataan yang dibagi ke dalam 4 kriteria yaitu *usefulness*, *easy of use*, *ease of learning*, dan *satisfaction*. Instrumen USE Questionnaire terdapat pada tabel 7 dibawah ini:

Tabel 7. Instrumen Penelitian *Usability*

No.	INSTRUMEN
<i>Usefulness</i>	
1.	Aplikasi ini membantu saya menjadi lebih efektif
2.	Aplikasi ini membantu saya menjadi lebih produktif
3.	Aplikasi ini bermanfaat
4.	Aplikasi ini memberi saya dampak yang besar terhadap tugas yang saya lakukan dalam hidup saya
5.	Aplikasi ini memudahkan saya mencapai hal-hal yang saya inginkan
6.	Aplikasi ini menghemat waktu ketika saya menggunakannya
7.	Aplikasi ini sesuai dengan kebutuhan saya
8.	Aplikasi ini bekerja sesuai dengan apa yang saya harapkan
<i>Easy of Use</i>	
9.	Aplikasi ini mudah digunakan
10.	Aplikasi ini praktis untuk digunakan
11.	Aplikasi ini mudah dipahami
12.	Aplikasi ini memerlukan langkah-langkah yang praktis untuk mencapai apa yang saya kerjakan
13.	Aplikasi ini dapat disesuaikan dengan kebutuhan
14.	Tidak kesulitan menggunakan aplikasi ini
15.	Saya dapat menggunakan tanpa instruksi tertulis
16.	Saya tidak melihat adanya ketidakkonsistenan selama saya menggunakannya
17.	Pengguna yang jarang maupun rutin menggunakan akan menyukai sistem ini
18.	Saya dapat kembali dari kesalahan dengan cepat dan mudah
19.	Saya dapat menggunakan sistem ini dengan berhasil
<i>Easy of Learning</i>	
20.	Saya belajar menggunakan Aplikasi ini dengan cepat
21.	Saya mudah mengingat bagaimana cara menggunakan aplikasi ini
22.	Aplikasi ini mudah untuk dipelajari cara menggunakannya
23.	Saya cepat menjadi terampil dengan aplikasi ini
<i>Satisfaction</i>	
24.	Saya puas dengan aplikasi ini
25.	Saya akan merekomendasikan sistem informasi ini kepada teman
26.	Aplikasi ini menyenangkan untuk digunakan
27.	Aplikasi ini bekerja seperti yang saya inginkan
28.	Aplikasi ini sangat bagus
29.	Saya merasa saya harus memiliki sistem informasi ini
30.	Aplikasi ini nyaman untuk digunakan

5. Instrumen *Reliability*

Pengujian aspek *reliability* menggunakan aplikasi LoadImpact, yaitu dengan menghitung ada berapa *request* yang ada, berapa yang sukses, dan berapa yang gagal.

6. Instrumen *Security*

Pengujian aspek *security* menggunakan aplikasi Acunetix Web Vulnerability Scanner. Instrumen untuk pengujian aspek *security* terdapat pada tabel 8 berikut ini:

Tabel 8. Instrumen Penelitian *Security*

No.	Parameter
1	<i>Blind SQL Injection</i>
2	<i>Cross Site Scripting</i>
3	<i>Googling Hacking Database (GHDB)</i>
4	<i>Microsoft IIS tilde directory enumeration</i>
5	<i>SQL Injection</i>
6	<i>Weak password</i>
7	<i>Directory Traversal</i>
8	<i>Application error message</i>
9	<i>Script Source Code Disclosure</i>
10	<i>HTML Form without CSRF protection</i>
11	<i>User credentials are sent in clear text</i>
12	<i>ASP.NET version disclosure</i>
13	<i>Clickjacking: X-Frame-Options header missing</i>
14	<i>Cookie without HttpOnly flag set</i>
15	<i>Cookie without Secure flag set</i>
16	<i>Login page password-guessing attack</i>
17	<i>OPTIONS method is enabled</i>
18	<i>Broken links</i>
19	<i>Microsoft IIS version disclosure</i>
20	<i>Password type input with auto-complete enabled</i>

7. Instrumen *Maintainability*

Pengujian *maintainability* menggunakan PHP Copy/Paste Detector untuk menguji duplikasi kode.

8. Instrumen *Portability*

Pengujian *portability* menggunakan *cross browsing compatibility testing* pada *desktop* dan *mobile* untuk menguji kualitas perangkat lunak berjalan baik pada *browser* berbeda.

F. Teknik Analisis Data

1. Analisis Faktor Kualitas *Functional Suitability*

Pengujian aspek *functional suitability* menggunakan *test case* dengan skala Guttman sebagai skala pengukuran instrumen. Setiap jawaban item instrumen yang menggunakan skala Guttman harus tegas dan konsisten misalnya "Ya" atau "Tidak" (Guritno, Sudaryono, & Rahardja, 2011). Hasil pengujian dihitung dengan rumus dari matriks Feature Completeness (Acharya dan Sinha, 2013). Matriks feature completeness merupakan matrik untuk mengukur sejauh mana fitur-fitur dapat diimplementasikan dengan benar. Berikut rumusnya:

$$X = \frac{I}{P}$$

P = jumlah fungsi yang dirancang

I = jumlah fungsi yang berhasil diimplementasikan

Dalam matriks Feature Completeness, nilai yang mendekati 1 mengindikasikan banyaknya fitur yang berhasil diimplementasikan. Hasil diukur dalam skala $0 \leq X \leq 1$. Perangkat lunak dikatakan baik dalam *functional suitability* jika X mendekati nilai 1.

2. Analisis Faktor Kualitas *Performance Efficiency*

Analisis kualitas untuk aspek *performance efficiency* dilakukan dengan aplikasi GTMetrix. Aturan yang digunakan pada YSlow untuk mengukur *performance efficiency* diinformasikan pada tabel 9 sebagai berikut:

Tabel 9. Aturan YSlow

No	Aturan
1.	<i>Make fewer HTTP</i>
2.	<i>Use a Content Delivery Network</i>
3.	<i>Avoid empty src or href</i>
4.	<i>Add Expires headers</i>
5.	<i>Compress components with gzip</i>
6.	<i>Put CSS at top</i>
7.	<i>Put JavaScript at bottom</i>
8.	<i>Avoid CSS expressions</i>
9.	<i>Make JavaScript and CSS external</i>
10.	<i>Reduce DNS lookups</i>
11.	<i>Minify JavaScript and CSS</i>
12.	<i>Avoid URL redirects</i>
13.	<i>Remove duplicate JavaScript and CSS</i>
14.	<i>Configure entity tags</i>
15.	<i>Make AJAX cacheable</i>
16.	<i>Use GET for AJAX requests</i>
17.	<i>Reduce the number of DOM elements</i>
18.	<i>Avoid HTTP 404 (Not Found) error</i>
19.	<i>Reduce cookie size</i>
20.	<i>Use cookie-free domains</i>
21.	<i>Avoid AlphasizeLoader filter</i>
22.	<i>Do not scale images in HTML</i>
23.	<i>Make favicon small and cacheable</i>

Pada aturan YSlow, semakin tinggi skor maka semakin baik kualitas *performance efficiency* dari perangkat lunak. Berikut ini tabel 10 yang memperlihatkan perbandingan skor:

Tabel 10. Skor Penilaian *Performance Efficiency* Pada Aturan Yslow

No	Grade	Syarat Skor
1	A	90 <= Skor <= 100
2	B	80 <= Skor < 90
3	C	70 <= Skor < 80
4	D	60 <= Skor < 70
5	E	50 <= Skor < 60
6	F	0 <= Skor < 50

Untuk aturan PageSpeed Insights yang digunakan untuk mengukur pengujian *performance efficiency* terdapat pada tabel 11 berikut ini:

Tabel 11. Aturan PageSpeed Insights Untuk Menguji *Performance Efficiency*

No	Aturan
1.	<i>Specify a Vary: Accept-Encoding header</i>
2.	<i>Minify HTML</i>
3.	<i>Minify CSS</i>
4.	<i>Minify Javascript</i>
5.	<i>Avoid bad requests</i>
6.	<i>Avoid a character set in the meta tag</i>
7.	<i>Avoid landing page redirects</i>
8.	<i>Defer parsing of JavaScript</i>
9.	<i>Enable gzip compression</i>
10.	<i>Enable Keep-Alive</i>
11.	<i>Inline small CSS</i>
12.	<i>Inline small Javascript</i>
13.	<i>Leverage browser caching</i>
14.	<i>Minimize redirects</i>
15.	<i>Minimize request size</i>
16.	<i>Optimize images</i>
17.	<i>Optimize the order of styles and scripts</i>
18.	<i>Put CSS in the document head</i>
19.	<i>Remove query strings from static resources</i>
20.	<i>Serve resources from a consistent URL</i>
21.	<i>Serve scaled images</i>
22.	<i>Specify a cache validator</i>
23.	<i>Specify a character set early</i>
24.	<i>Specify image dimensions</i>
25.	<i>Avoid CSS @import</i>
26.	<i>Combine images using CSS sprites</i>
27.	<i>Prefer asynchronous resources</i>

3. Analisis Faktor Kualitas *Compatibility*

Analisis kualitas untuk aspek *compatibility* tidak dilakukan karena tidak ada software pembanding untuk pengujiannya.

4. Analisis Faktor Kualitas *Usability*

Pengujian aspek *usability* menggunakan skala Likert sebagai skala pengukuran dalam instrumen pengujian dimana setiap jawaban item instrumen mempunyai gradasi dari sangat positif sampai sangat negatif (Sugiyono, 2013). Skala Likert dalam instrumen *USE Questionnaire* dapat menggunakan skala 5 maupun skala 7 dalam penilaiannya. Penelitian ini menggunakan skala 5 seperti yang sudah dilakukan oleh Muderedzwa & Nyakwende (2010) dan Rahadi (2014) dalam penelitiannya di bidang teknologi informasi. Selain itu, skala Likert 5 poin relative lebih cepakt dan mudah digunakan dalam penelitian.

Pada analisis kuantitatif maka jawaban pada skala Likert dapat diberi skor sebagai berikut :

1. Sangat setuju (SS) diberi skor 5
2. Setuju (S) diberi skor 4
3. Ragu-ragu (RR) diberi skor 3
4. Tidak setuju (TS) diberi skor 2
5. Sangat tidak setuju (STS) diberi skor 1

Setelah hasil perhitungan di dapat, maka dikomparasikan dengan tabel kriteria skor pada tabel 12 berikut yang telah disesuaikan.

Tabel 12. Kriteria Skor Pengujian *Usability*

Persentase Pencapaian (%)	Interpretasi
0% - 20%	Sangat Tidak Layak
21% - 40%	Kurang Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

Dari hasil yang didapat dengan pengujian menggunakan kuesioner, kemudian dilakukan penghitungan reliabilitas terhadap instrumen. Perhitungan reliabilitas dilakukan dengan Alpha Cronbach menggunakan software SPSS dengan nilai reliabilitas Alpha Cronbach (Gliem & Gliem, 2003) sebagai berikut :

Tabel 13. Tabel Konsistensi Alpha Cronbach

Cronbach's Alpha	<i>Internal Consistency</i>
$\alpha \geq .9$	<i>Excellent</i>
$.9 > \alpha \geq .8$	<i>Good</i>
$.8 > \alpha \geq .7$	<i>Acceptable</i>
$.7 > \alpha \geq .6$	<i>Questionable</i>
$.6 > \alpha \geq .5$	<i>Poor</i>
$.5 > \alpha$	<i>Unacceptable</i>

5. Analisis Faktor Kualitas *Reliability*

Pengujian reliability menggunakan bantuan tools LoadImpact, dan nilainya akan dihitung dengan rumus sesuai dengan Model Nelson.

$$R1 = 1 - \frac{ne}{n}$$

Keterangan :

R1 = nilai reliability

ne = jumlah input yang gagal

n = jumlah input

Hasil presentase tersebut dibandingkan dengan standard uji *reliability*, standard Telcordia. Hasil pengujian dikatakan memenuhi aspek *reliability* jika presentase bernilai minimal 95% (Asthana & Oliveri, 2009).

6. Analisis Faktor Kualitas *Security*

Pengujian aspek *Security* dilakukan dengan menggunakan aplikasi Acunetix Vulnerability Scanner. Hasil analisis datanya didapatkan dari hasil pengujian melalui aplikasi tersebut.

7. Analisis Faktor Kualitas *Maintainability*

Analisis untuk pengujian aspek *maintainability* menggunakan *tools* PHP Copy/Paste Detector. Pengujian ini menguji *code duplication* dari *source code* perangkat lunak. Hasil dari pengujian *code duplication* digunakan rumus :

$$Skor_d = \min(-30 \times \log_{10}(C) + 60, 100)$$

Keterangan :

C = Persentase hasil pengukuran *code duplication*

Interpretasi kualitas perangkat lunak pada aspek *maintainability* berdasarkan pengukuran *code duplication* dapat dilihat pada Tabel 14 di bawah ini.

Tabel 14. Kategori Penilaian *Maintainability*

<i>Code Duplication</i>	TQI Score ($Skor_d$)	Grade	Interpretasi
$\leq 0.10 \%$	$\geq 90 \%$	A	Sangat Baik
$\leq 0.22 \%$	$\geq 80 \%$	B	Baik
$\leq 0.46 \%$	$\geq 70 \%$	C	Cukup Baik
$\leq 2.15 \%$	$\geq 50 \%$	D	Cukup
$\leq 4.64 \%$	$\geq 40 \%$	E	Lemah
$> 4.64 \%$	$< 40 \%$	F	Sangat Lemah

8. Analisis Faktor Kualitas *Portability*

Aspek *portability* ini dilakukan dengan menjalankan sisten pada *browser* berbasis *desktop* pada *browser* berbasis *mobile*. Berikut informasi lebih lengkapnya dapat dilihat di tabel 15 dibawah ini:

Tabel 15. Pengujian Faktor Kualitas *Portability*

No	Aspek	Hasil
1	Di uji pada <i>browser</i> berbasis <i>Desktop</i>	Sistem kompatibel dengan beberapa <i>browser</i> ternama. Hal ini terbukti dari hasil pengujian bahwa sistem dapat diakses dengan baik dari beberapa browser seperti Mozilla Firefox, Opera dan Google Chrome tanpa ada pesan error.
2	Diuji pada <i>browser</i> berbasis <i>Mobile</i>	Sistem kompatibel dengan beberapa <i>browser</i> ternama, seperti Mobile browser, Opera dan Google Chrome tanpa ada pesan error.

BAB IV

HASIL DAN PEMBAHASAN

A. Tahap Analisis Kebutuhan

Tahap Analisis kebutuhan adalah tahap awal yang dilakukan dalam penelitian pengembangan sistem informasi, tujuannya yaitu untuk menentukan hal apa saja yang dibutuhkan dalam pengembangan sistem.

1. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan analisis fungsi yang diperlukan dalam aplikasi berdasarkan hasil observasi dengan wawancara kepada beberapa mahasiswa dan dosen yang telah dilakukan, fungsi yang diperlukan meliputi :

1. Melalui sistem informasi, mahasiswa bisa melakukan pendaftaran bimbingan secara *online*
2. Mahasiswa bisa menambahkan catatan dan melihat catatan bimbingan
3. Mahasiswa bisa mengirimkan file konsultasi melalui sistem informasi kepada dosen
4. Mahasiswa bisa mengubah detail profil akunnya
5. Mahasiswa bisa mengubah password akunnya
6. Mahasiswa bisa login dan logout
7. Mahasiswa bisa mengirimkan pesan kepada dosen pembimbingnya
8. Dosen bisa mendaftar akun di sistem informasi
9. Dosen bisa melihat daftar mahasiswa bimbingannya.
10. Dosen bisa melihat detail akun per mahasiswa bimbingannya.
11. Dosen bisa melihat catatan bimbingan mahasiswanya
12. Dosen bisa melihat data konsultasi *online* mahasiswanya

13. Dosen bisa mengirimkan file koreksi kepada mahasiswanya
14. Dosen bisa mengirimkan pesan kepada mahasiswanya
15. Dosen bisa mengedit detail profilnya
16. Dosen bisa mengubah password akunya
17. Dosen bisa login dan logout
18. Admin bisa login dan logout
19. Admin bisa melihat daftar semua mahasiswa
20. Admin bisa menentukan dosen pembimbing
21. Admin bisa melihat daftar catatan bimbingan
22. Admin bisa melihat daftar konsultasi *online* dosen dan mahasiswa
23. Admin bisa menambahkan dan melihat daftar prodi
24. Admin bisa menambah dan melihat daftar jurusan
25. Admin bisa mengubah password akunya

2. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak minimum yang dapat digunakan agar sistem informasi data dijalankan atau diakses membutuhkan perangkat yang memiliki *browser* dan memiliki akses internet.

B. Tahap Desain

Dalam tahap ini proses perancangan sistem dilakukan yang meliputi desain UML (*Unified Modeling Language*), desain antarmuka, desain sistem, dan desain basis data.

1. Desain UML (*Unified Modeling Language*)

a. Desain Use-Case Diagram

Diagram *Use-Case* dibuat untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

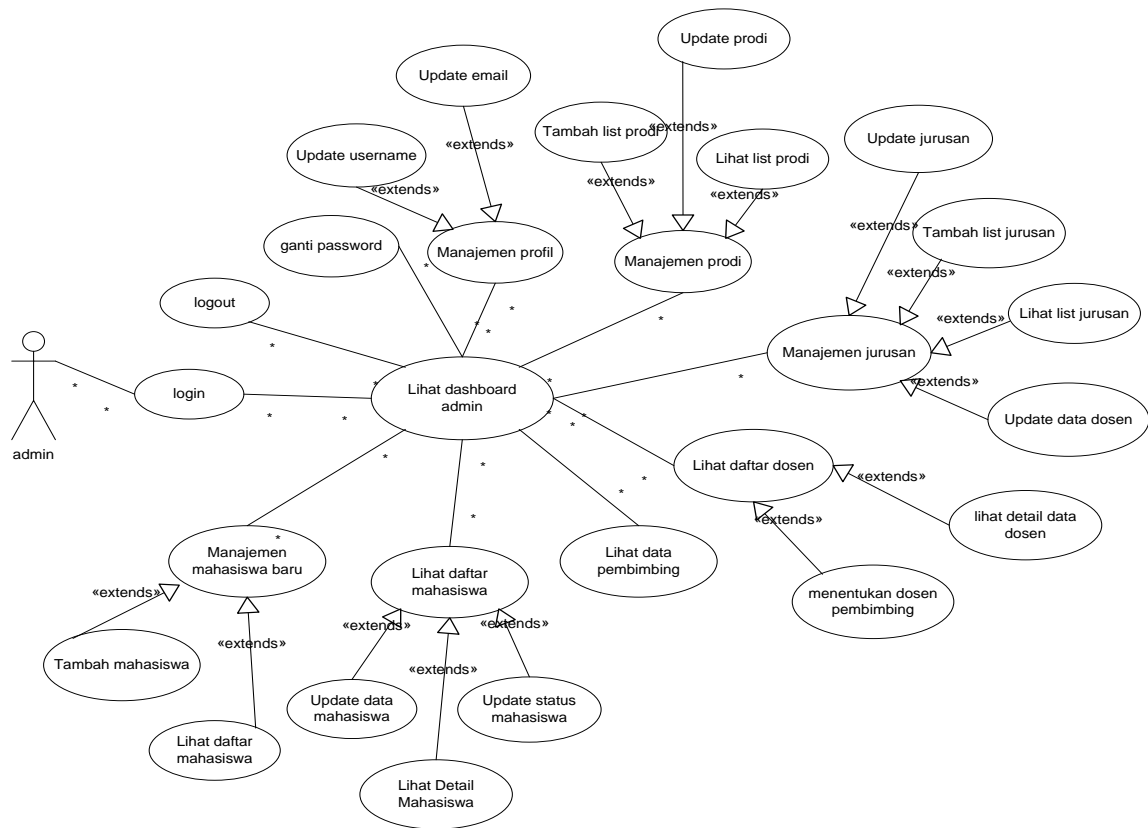
Use case diagram menggambarkan semua aktor dengan skenarionya masing-masing. Dalam sistem informasi ini memiliki tiga aktor, yaitu admin, mahasiswa, dan dosen. Tabel definisi aktor diinformasikan pada tabel di bawah ini.

Tabel 16. Definisi Aktor

No	Aktor	Deskripsi
1	Admin	Admin merupakan admin skripsi di lingkup Jurusan
2	Mahasiswa	Mahasiswa yang mendaftar bimbingan skripsi <i>online</i>
3	Dosen	Dosen pembimbing skripsi yang telah memiliki akun di Sistem Informasi Bimbingan Skripsi <i>online</i>

1) Use case Diagram Admin

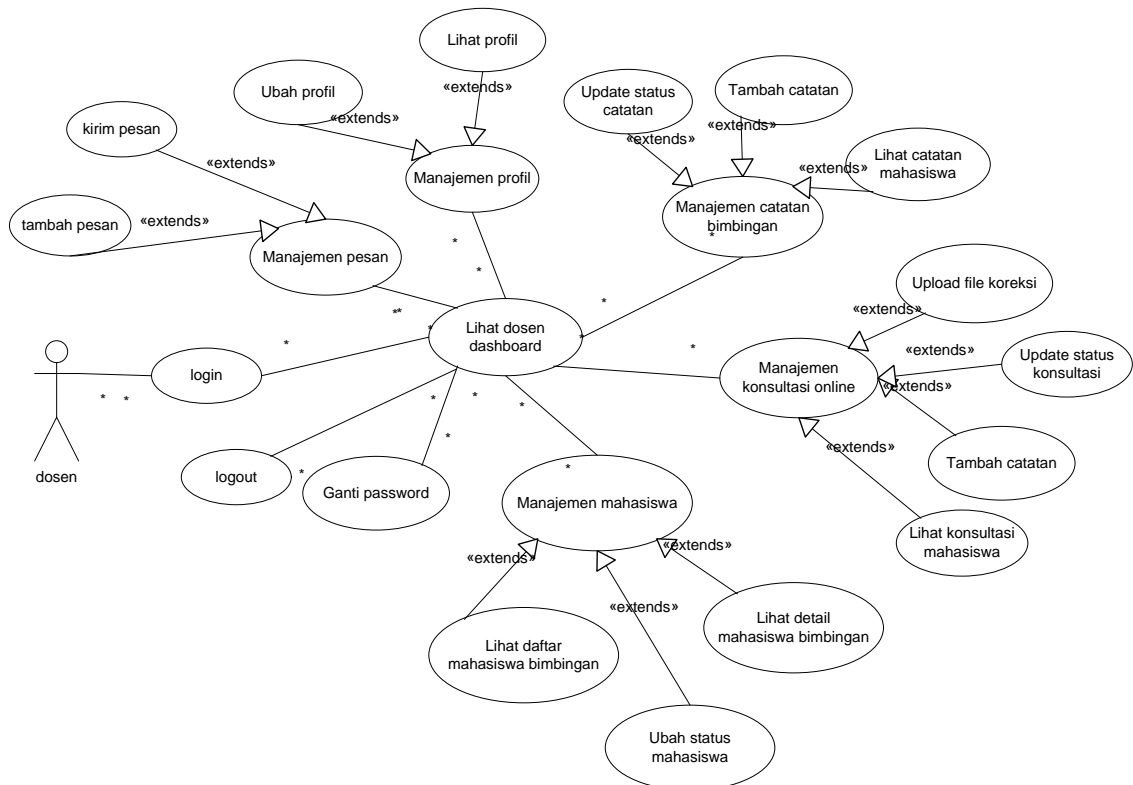
Admin adalah aktor yang memiliki hak akses atas seluruh pengelolaan. Use case diagram untuk admin digambarkan sebagai berikut :



Gambar 4. Diagram Use Case Admin

3) Use case Diagram Dosen

Dosen adalah aktor yang dapat melihat daftar mahasiswa bimbingannya, melihat catatan bimbingan, dan mengelola konsultasi *online*. Diagram *use case* aktor dosen digambarkan pada gambar dibawah ini.

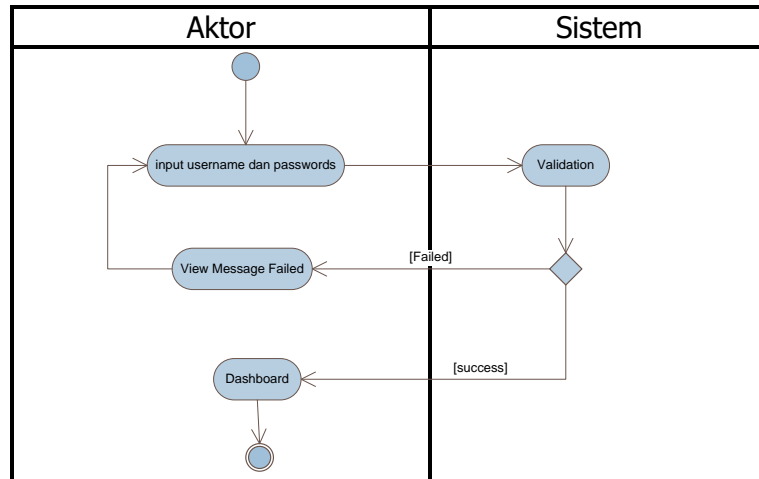


Gambar 6. Diagram *Use Case* Dosen

b. Desain Activity Diagram

Berikut dijelaskan tentang *activity diagram login*, tambah data, ubah data, *delete data*, *upload* dan *download* data pada sistem yang dibuat.

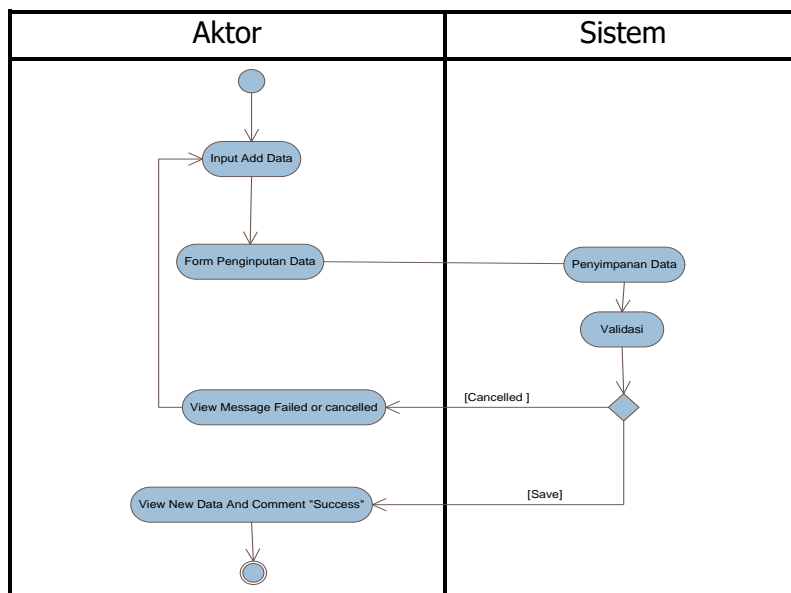
1) Activity Diagram Login



Gambar 7. Activity Diagram Login

2) Activity Diagram Tambah Data

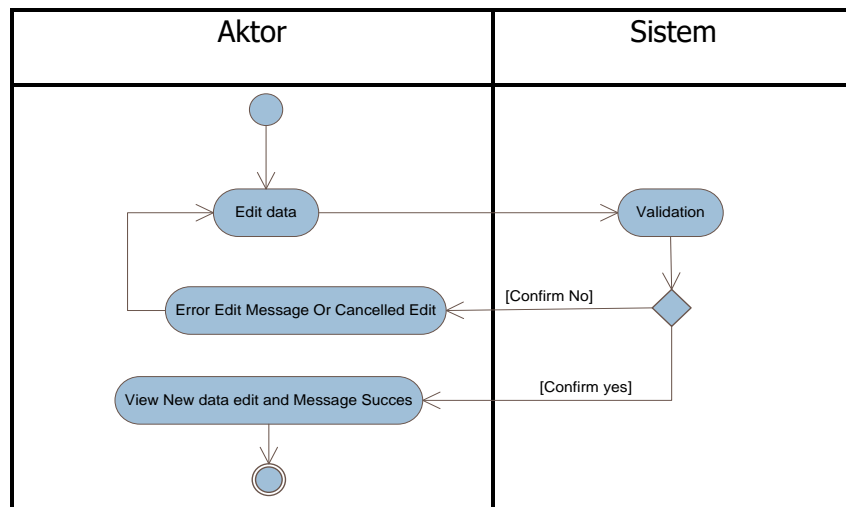
Activity Diagram Tambah data dilakukan oleh aktor yang mempunyai hak akses untuk tambah data.



Gambar 8. Activity Diagram Tambah Data

3) *Activity Diagram* Ubah Data

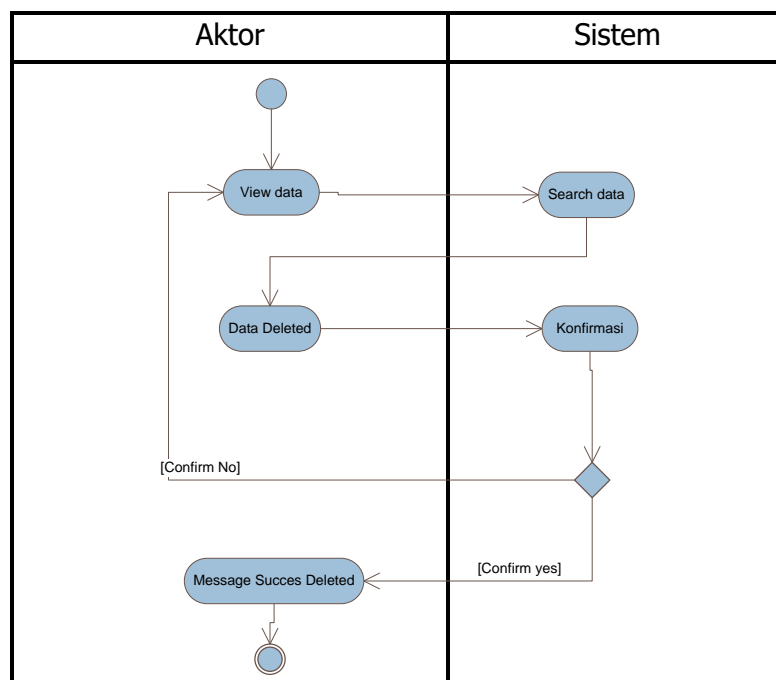
Activity diagram ubah data pada sistem digambarkan seperti di bawah ini.



Gambar 9. *Activity Diagram* Ubah Data

4) *Activity Diagram* Delete Data

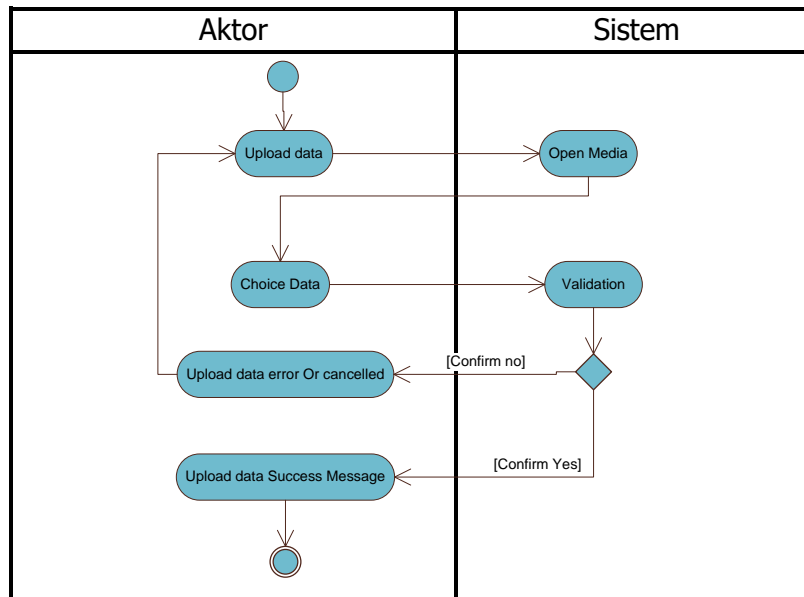
Activity diagram delete data digambarkan pada diagram di bawah ini.



Gambar 10. *Activity Diagram* Delete Data

5) *Activity Diagram Upload Data*

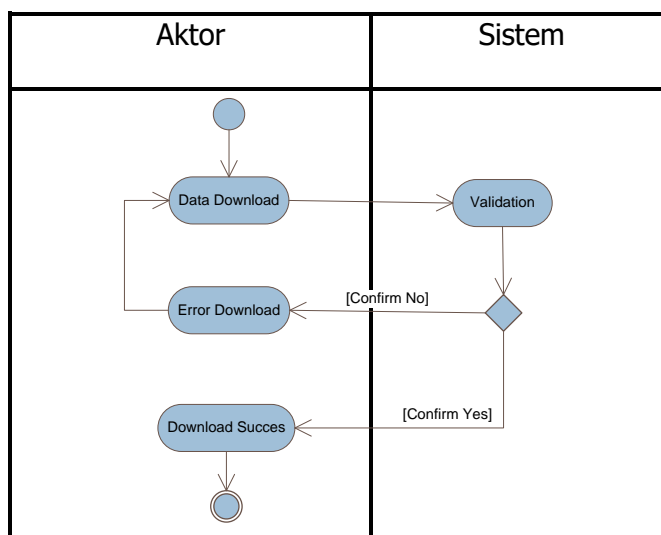
Activity diagram upload pada sistem digambarkan seperti berikut ini.



Gambar 11. *Activity Diagram Upload Data*

6) *Activity Diagram Download*

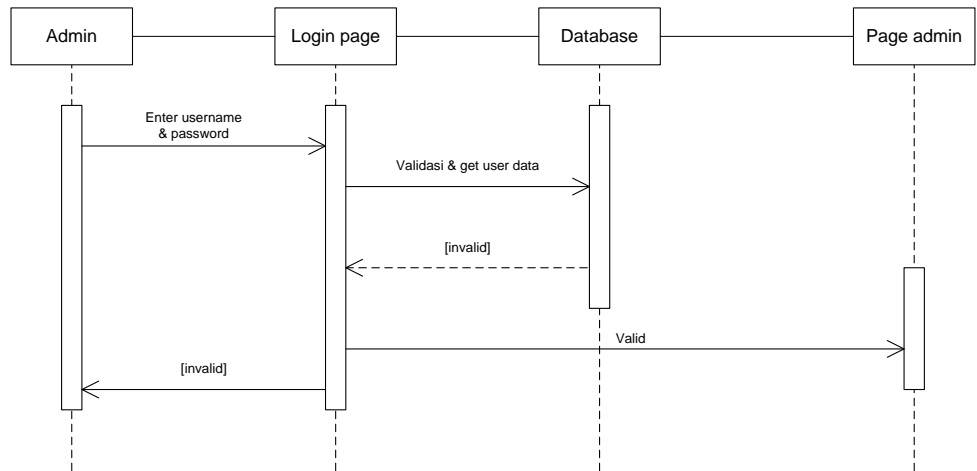
Activity diagram download pada sistem digambarkan sebagai berikut :



Gambar 12. *Activity Diagram Download data*

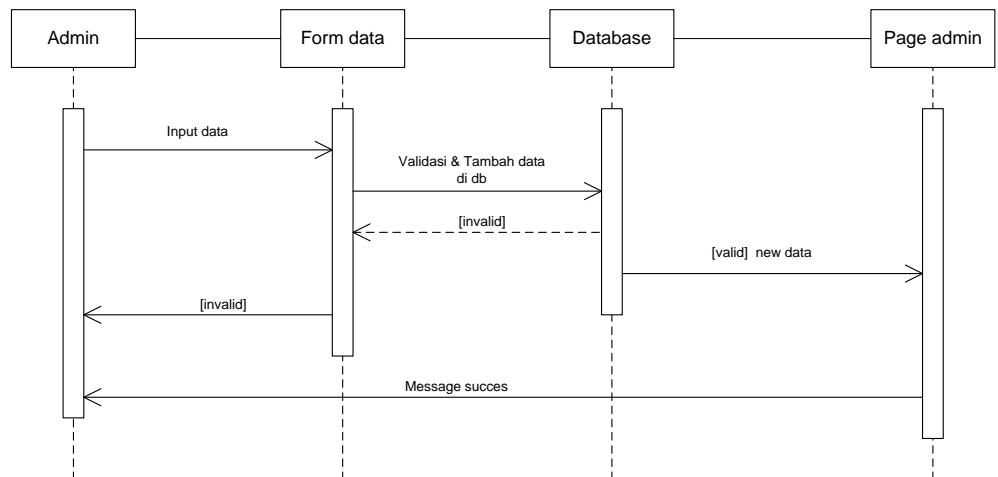
c. Desain *Sequence Diagram*

1) *Sequence Diagram Login*



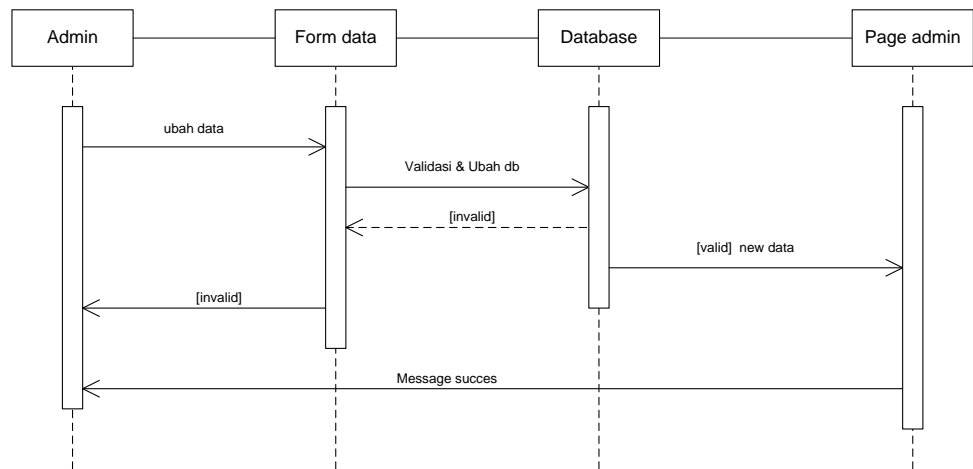
Gambar 13. *Sequence Diagram Login*

2) *Sequence Diagram Tambah Data*



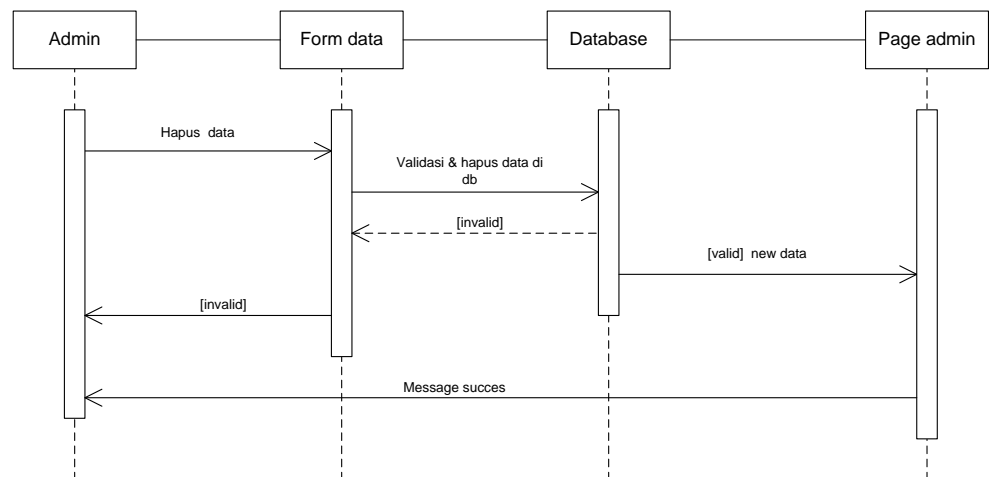
Gambar 14. *Sequence Diagram Tambah Data*

3) Sequence Diagram Edit Data



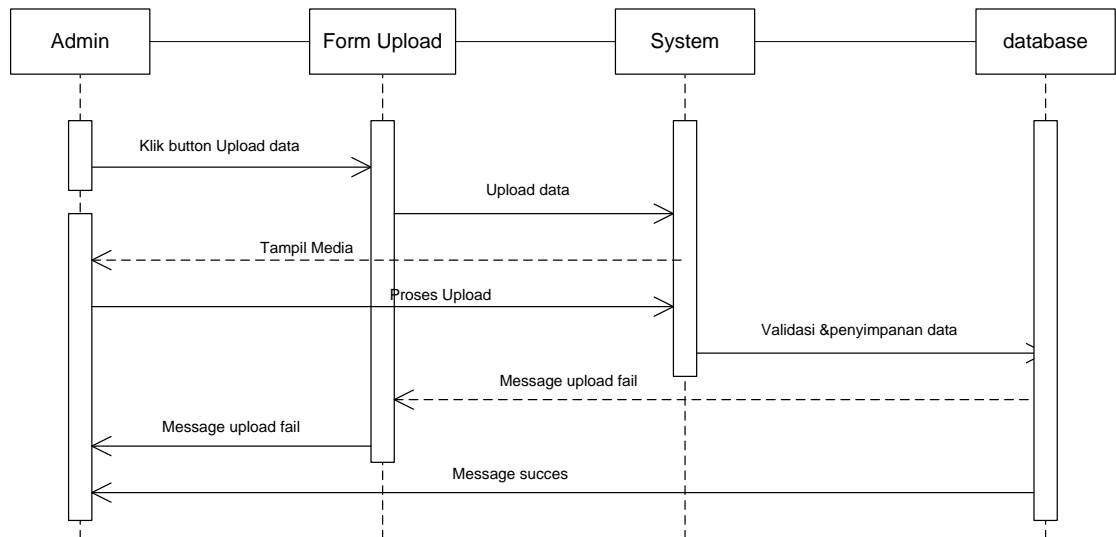
Gambar 15. Sequence Diagram Edit Data

4) Sequence Diagram Hapus Data



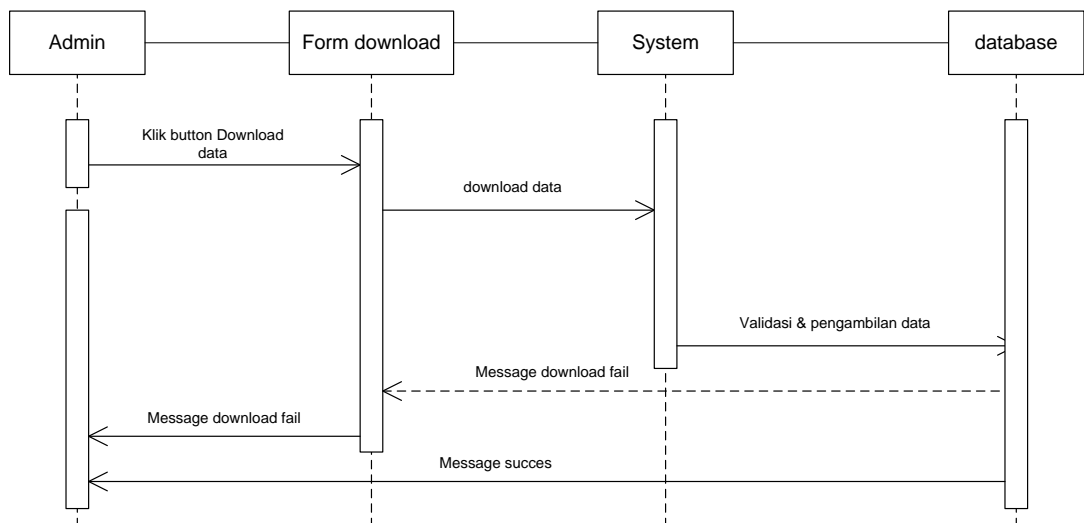
Gambar 16. Sequence Diagram Hapus Data

5) *Sequence Diagram Upload*



Gambar 17. *Sequence Diagram Upload*

6) *Sequence Diagram Download*



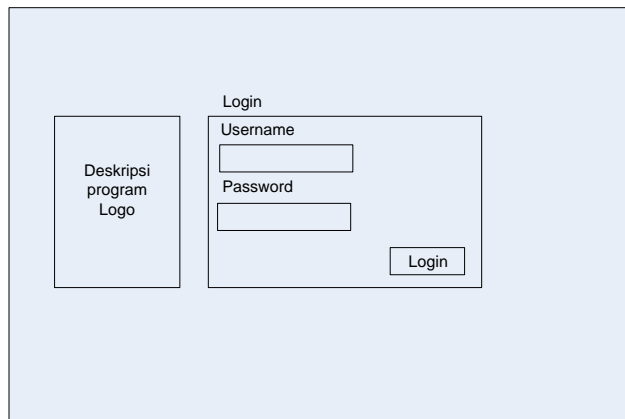
Gambar 18. *Sequence Diagram Download*

2. Perancangan Desain *Interface*

Desain *interface* untuk sistem informasi digambarkan sebagai berikut :

a. Halaman *Login*

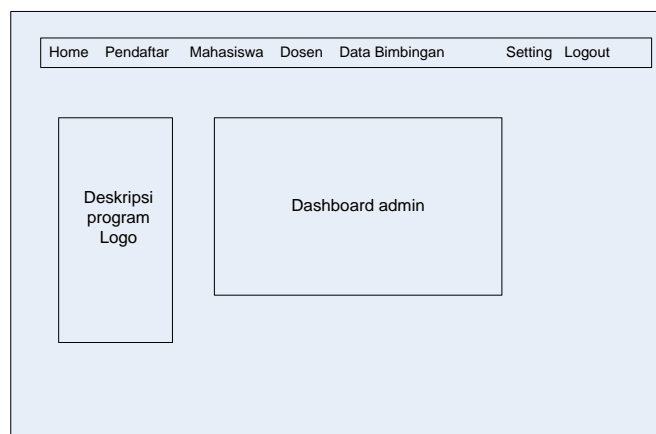
Desain *interface* halaman *login* untuk sistem informasi ini digambarkan sebagai berikut :



Gambar 19. Desain *User Interface Login*

b. Halaman Menu Utama Admin

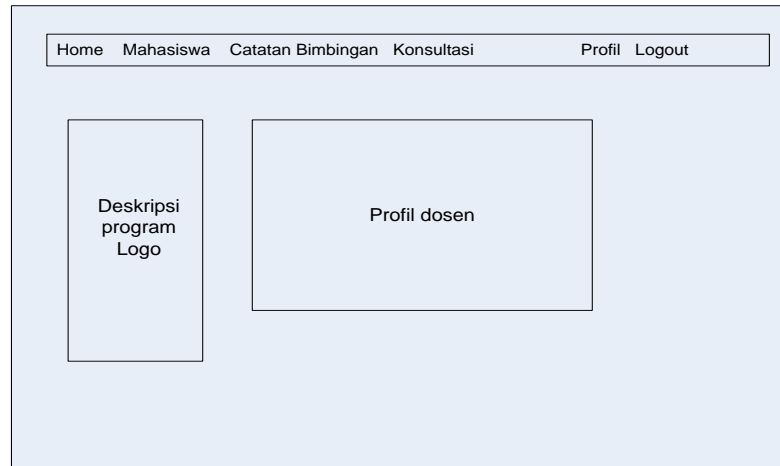
Desain *interface* halaman Menu Utama Admin untuk sistem informasi ini digambarkan sebagai berikut :



Gambar 20. Desain *User Interface* Halaman Utama Admin

c. Halaman Menu Utama Dosen

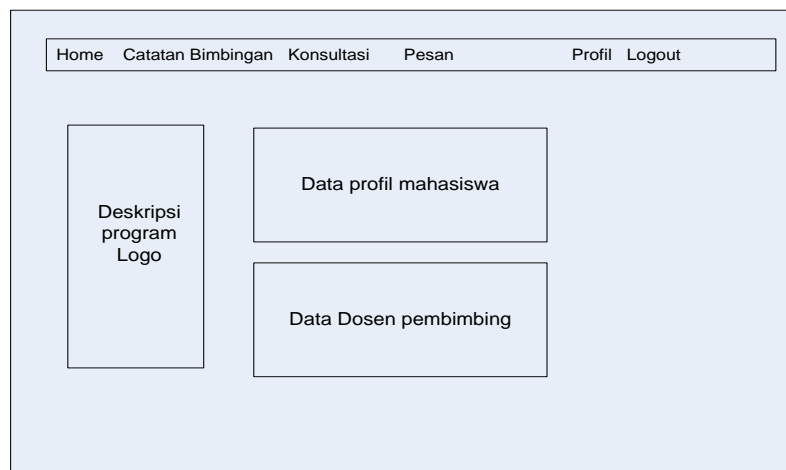
Desain *interface* halaman menu utama dosen untuk sistem informasi ini digambarkan sebagai berikut :



Gambar 21. Desain *User Interface* Halaman Utama Dosen

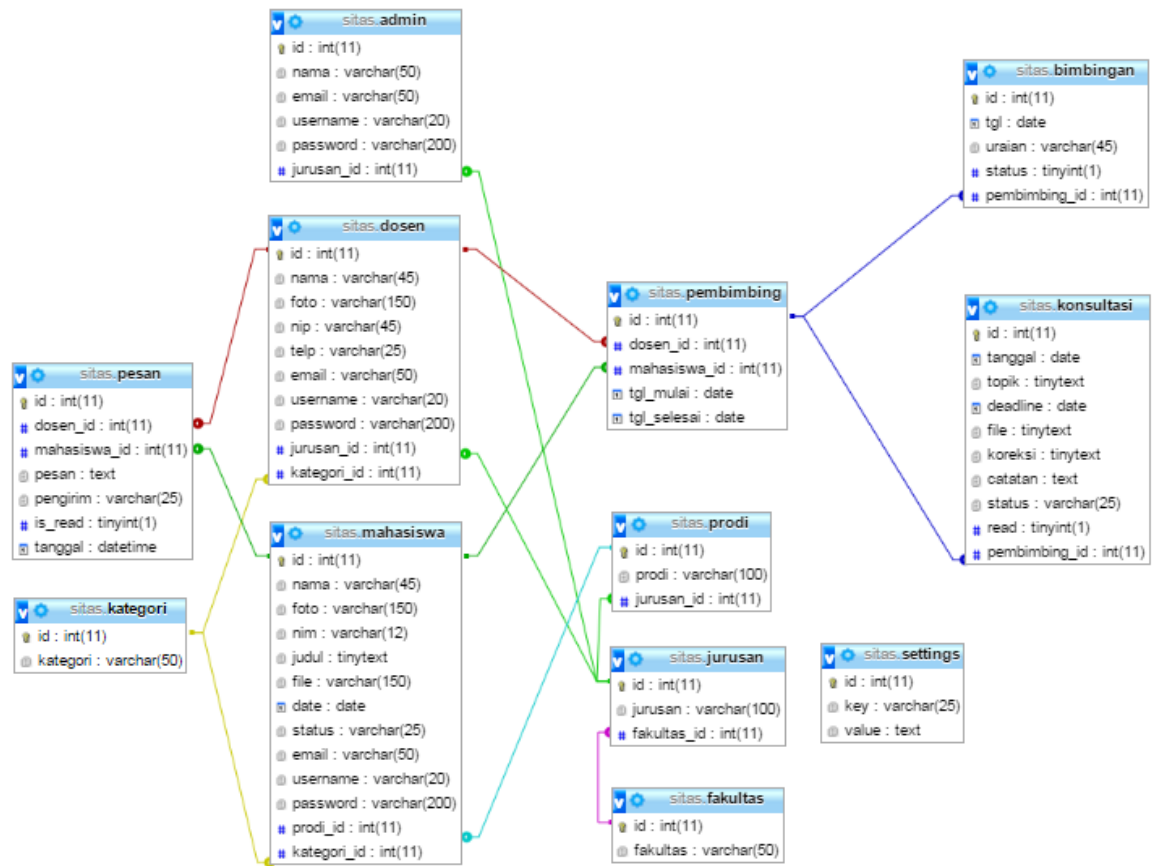
d. Halaman Menu Utama Mahasiswa

Desain *interface* halaman menu utama mahasiswa untuk sistem informasi ini digambarkan sebagai berikut :



Gambar 22. Desain *User Interface* Menu Utama Mahasiswa

3. Perancangan *Database*



Gambar 23. Perancangan *Database*

C. Tahap Implementasi

1. Implementasi Fungsi

Tahap ini merupakan implementasi dari tahap desain yang telah dilakukan sebelumnya. Tahap ini mengimplementasikan fungsi-fungsi yang dibutuhkan. Namun di sistem informasi ini belum ada fitur notifikasi kepada *user*. Fitur notifikasi ini dapat ditambahkan pada pengembangan sistem informasi lebih lanjut. Dan sistem ini menggunakan *database* sendiri yang belum terintegrasi dengan *database* yang dimiliki oleh jurusan, fakultas, ataupun universitas.

a. Halaman *Login*

Gambar di bawah ini merupakan desain halaman *login*.

The screenshot shows a login page with a header section on the left and a sign-in form on the right. The header section includes the system name, a logo, a greeting, a description of the application, and navigation links. The sign-in form contains fields for Username, Password, and a dropdown menu for 'Login Sebagai' (Login As) with 'Mahasiswa' selected, and a 'Masuk' button.

Sistem Informasi Bimbingan Skripsi 0.1

Hai, Pengunjung

Ini adalah aplikasi bimbingan, aplikasi ini digunakan untuk mempermudah proses bimbingan dan rekam jejak data-data bimbingan.

[Login](#) | [Daftar](#)

© 2015 Bootstrap
[About](#) [Help](#) [Terms](#) [FAQ](#)

Sign In

Username

Password

Login Sebagai
Mahasiswa

Masuk

Gambar 24. Halaman *Login*

b. Halaman Dosen

Gambar di bawah ini merupakan desain halaman Dosen

The screenshot shows a user profile page with a blue navigation bar at the top. The main content area is divided into two columns. The left column contains the system name, a logo, a greeting, and a message count. The right column contains a table with user details.

Home Mahasiswa Bimbingan Catatan Bimbingan Konsultasi Online Profile Password Logout

Sistem Informasi Bimbingan Skripsi 0.1

Hai, Dosenek
(Login sebagai : dosen)

0 Pesan Baru

Ini adalah aplikasi bimbingan, aplikasi ini digunakan untuk mempermudah proses bimbingan dan rekam jejak data-data bimbingan.

© 2015 Bootstrap

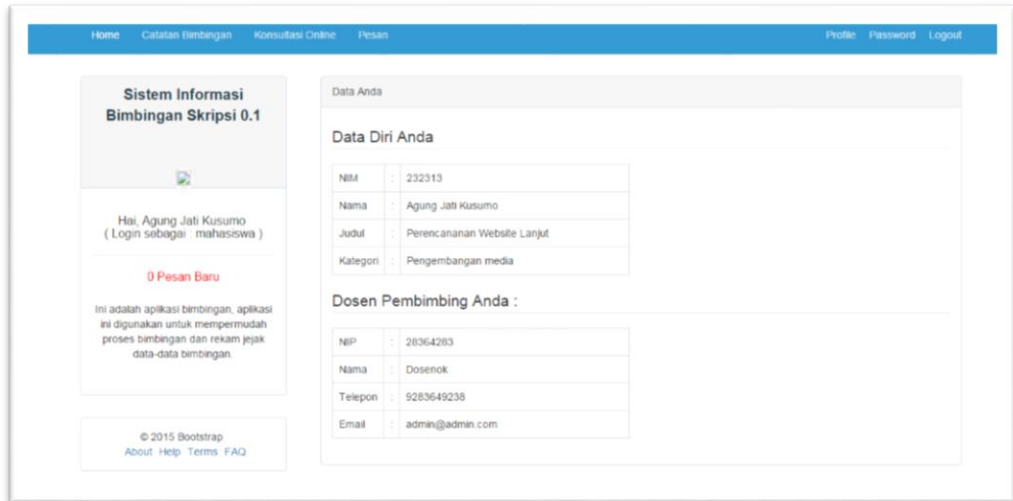
Detail Dosen

NIP	: 28364283
Nama	: Dosenek
Telepon	: 9283649238
Email	: admin@admin.com
Username	: dosens

Gambar 25. Halaman Dosen

c. Halaman Mahasiswa

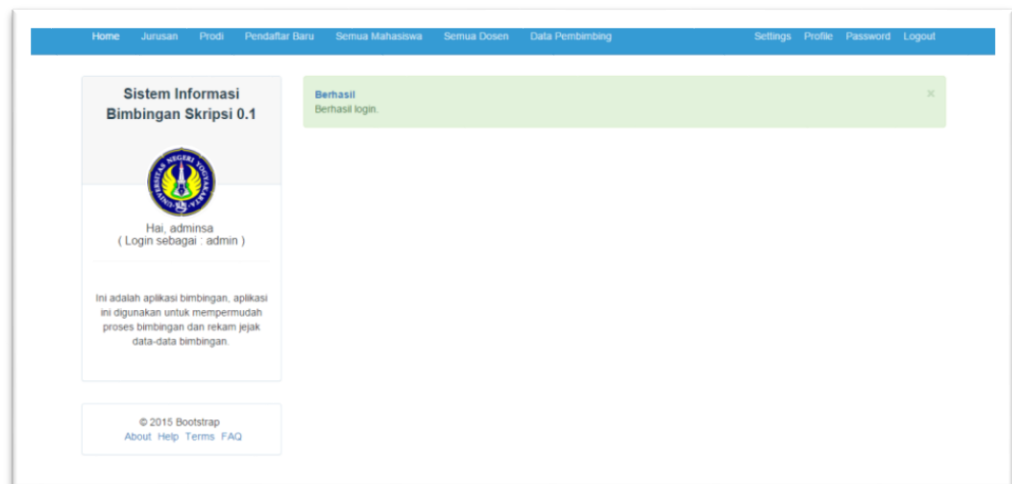
Gambar di bawah ini merupakan desain halaman Mahasiswa



Gambar 26. Halaman Mahasiswa

d. Halaman Admin

Gambar di bawah ini merupakan desain halaman Admin



Gambar 27. Halaman Admin

2. Implementasi *Database*

Database yang digunakan untuk aplikasi pengelolaan data prestasi mahasiswa adalah MySQL. Berikut ini hasil implementasi *database* aplikasi pengelolaan data prestasi mahasiswa menggunakan MySQL.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> admin		1	MyISAM	latin1_swedish_ci	3.1 KiB	-
<input type="checkbox"/> bimbingan		3	MyISAM	latin1_swedish_ci	3.1 KiB	-
<input type="checkbox"/> dosen		4	MyISAM	latin1_swedish_ci	4.5 KiB	-
<input type="checkbox"/> fakultas		3	MyISAM	latin1_swedish_ci	2.1 KiB	-
<input type="checkbox"/> jurusan		1	MyISAM	latin1_swedish_ci	3.1 KiB	-
<input type="checkbox"/> kategori		4	MyISAM	latin1_swedish_ci	2.1 KiB	-
<input type="checkbox"/> konsultasi		4	MyISAM	latin1_swedish_ci	3.3 KiB	-
<input type="checkbox"/> mahasiswa		5	MyISAM	latin1_swedish_ci	4.8 KiB	-
<input type="checkbox"/> pembimbing		5	MyISAM	latin1_swedish_ci	4.1 KiB	-
<input type="checkbox"/> pesan		20	MyISAM	latin1_swedish_ci	5.1 KiB	-
<input type="checkbox"/> prodi		3	MyISAM	latin1_swedish_ci	3.1 KiB	-
<input type="checkbox"/> settings		1	MyISAM	latin1_swedish_ci	2 KiB	-
12 tables	Sum	54	MyISAM	latin1_swedish_ci	40.2 KiB	0 B

Gambar 28. *Database* Sistem Informasi

Pada gambar di atas terdapat ada 12 tabel hasil implementasi, Berikut ini detail dari ke-11 tabel tersebut :

a. Tabel Admin

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/> 2	nama	varchar(50)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 3	email	varchar(50)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 4	username	varchar(20)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 5	password	varchar(200)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 6	jurusan_id	int(11)			Yes	NULL		

Gambar 29. Tabel Admin

b. Tabel Mahasiswa

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique
<input type="checkbox"/>	2 nama	varchar(45)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	3 foto	varchar(150)	latin1_swedish_ci		No	None		Change Drop Primary Unique
<input type="checkbox"/>	4 nim	varchar(12)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	5 judul	tinytext	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	6 file	varchar(150)	latin1_swedish_ci		No	None		Change Drop Primary Unique
<input type="checkbox"/>	7 date	date			Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	8 status	varchar(25)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	9 email	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	10 username	varchar(20)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	11 password	varchar(200)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	12 prodi_id	int(11)			Yes	NULL		Change Drop Primary Unique
<input type="checkbox"/>	13 kategori_id	int(11)			Yes	NULL		Change Drop Primary Unique

Gambar 30. Tabel Mahasiswa

c. Tabel Dosen

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2 nama	varchar(45)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	3 foto	varchar(150)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index
<input type="checkbox"/>	4 nip	varchar(45)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	5 telp	varchar(25)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	6 email	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	7 username	varchar(20)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	8 password	varchar(200)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	9 jurusan_id	int(11)			Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	10 kategori_id	int(11)			Yes	NULL		Change Drop Primary Unique Index

Gambar 31. Tabel Dosen

d. Tabel Jurusan

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2 jurusan	varchar(100)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	3 fakultas_id	int(11)			Yes	NULL		Change Drop Primary Unique Index

Gambar 32. Tabel Jurusan

e. Tabel Prodi

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2 prodi	varchar(100)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index
<input type="checkbox"/>	3 jurusan_id	int(11)			Yes	NULL		Change Drop Primary Unique Index

Gambar 33. Tabel Prodi

f. Tabel Pembimbing

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/>	2 dosen_id	int(11)			Yes	NULL		
<input type="checkbox"/>	3 mahasiswa_id	int(11)			Yes	NULL		
<input type="checkbox"/>	4 tgl_mulai	date			Yes	NULL		
<input type="checkbox"/>	5 tgl_selesai	date			Yes	NULL		

Gambar 34. Tabel Pembimbing

g. Tabel Bimbingan

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/>	2 tgl	date			Yes	NULL		
<input type="checkbox"/>	3 uraian	varchar(45)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/>	4 status	tinyint(1)			Yes	NULL		
<input type="checkbox"/>	5 pembimbing_id	int(11)			Yes	NULL		

Gambar 35. Tabel Bimbingan

h. Tabel Konsultasi

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/>	2 tanggal	date			Yes	NULL		
<input type="checkbox"/>	3 topik	tinytext	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/>	4 deadline	date			Yes	NULL		
<input type="checkbox"/>	5 file	tinytext	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/>	6 koreksi	tinytext	latin1_swedish_ci		No	None		
<input type="checkbox"/>	7 catatan	text	latin1_swedish_ci		No	None		
<input type="checkbox"/>	8 status	varchar(25)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/>	9 read	tinyint(1)			Yes	NULL		
<input type="checkbox"/>	10 pembimbing_id	int(11)			Yes	NULL		

Gambar 36. Tabel Konsultasi

i. Tabel Pesan

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT	
<input type="checkbox"/>	2 dosen_id	int(11)			Yes	NULL		
<input type="checkbox"/>	3 mahasiswa_id	int(11)			Yes	NULL		
<input type="checkbox"/>	4 pesan	text	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/>	5 pengirim	varchar(25)	latin1_swedish_ci		No	None		
<input type="checkbox"/>	6 is_read	tinyint(1)			Yes	NULL		
<input type="checkbox"/>	7 tanggal	datetime			No	None		

Gambar 37. Tabel Pesan

j. Tabel Fakultas

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2	fakultas	varchar(50) latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index

Gambar 38. Tabel Fakultas

k. Tabel Setingan

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2	key	varchar(25) latin1_swedish_ci		No	None		Change Drop Primary Unique Index
<input type="checkbox"/>	3	value	text latin1_swedish_ci		No	None		Change Drop Primary Unique Index

Gambar 39. Tabel Setingan

l. Tabel Kategori

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	1	id			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index
<input type="checkbox"/>	2	kategori	varchar(50) latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique Index

Gambar 40. Tabel Kategori

D. Tahap Pengujian

1. Hasil Pengujian *Functionality Suitability*

Pengujian *functional suitability* diujikan pada 2 ahli pemrograman menggunakan angket yang berisi fungsi-fungsi pada perangkat lunak. Instrumen yang diujikan memenuhi sub karakteristik *functional completeness*, *functional correctness*, dan *functional appropriateness*. Hasil pengujian *functional suitability* sebagai berikut:

Tabel 17. Hasil Pengujian *Functional Suitability*

No. Pertanyaan	Ya	Tidak
1	2	0
2	2	0
3	2	0
4	2	0
5	2	0
6	2	0
7	2	0
8	2	0
9	2	0
10	2	0
11	2	0
12	2	0
13	2	0
14	2	0
15	2	0
16	2	0
17	2	0
18	2	0
19	2	0
20	2	0
21	2	0
22	2	0
23	2	0
24	2	0
25	2	0
26	2	0
27	2	0
28	2	0
Total	56	0

Pengujian subkarakteristik *functional completeness* dan *functional appropriateness* menggunakan rumus sebagai berikut :

$$X = \frac{I}{P}$$

Keterangan :

I = jumlah fitur yang berhasil diimplementasikan

P = jumlah fitur yang dirancang

Perhitungan sub karakteristiknya dilakukan sebagai berikut:

$$X = \frac{56}{56} = 1$$

Berdasarkan hasil perhitungan, dapat disimpulkan bahwa aplikasi sistem informasi bimbingan skripsi mahasiswa dinyatakan baik dalam subkarakteristik *functional completeness* dan *functional appropriateness*.

Untuk hasil pengujian sub karakteristik *functional correctness* terdapat pada tabel berikut:

Tabel 18. Hasil Pengujian Subkarakteristik *Functional Suitability*

No. Pertanyaan	Ya	Tidak
1	2	0
2	2	0
3	2	0
4	2	0
5	2	0
6	2	0
7	2	0
8	2	0
9	2	0
10	2	0
11	2	0
12	2	0
13	2	0
14	2	0
Total	28	0

Rumus yang digunakan untuk menghitung hasil pengujian sub karakteristik *functional correctness* sama dengan rumus yang digunakan untuk menghitung hasil pengujian sub karakteristik *functional completeness* dan *functional appropriateness*, perhitungannya sebagai berikut :

$$X = \frac{28}{28} = 1$$

Berdasarkan hasil tersebut disimpulkan bahwa sistem infrmasi bimbingan skripsi mahasiswa dinyatakan baik, karena nilai X mendekati 1. Sehingga bisa disimpulkan bahwa sistem informasi bimbingan skripsi telah memenuhi aspek *functional suitability*.

2. Hasil Pengujian *Performance Efficiency*

Pengujian *performance efficiency* dilakukan menggunakan alat ukur GTMetrix dan YSlow. Berikut hasil pengujian *performance efficiency* menggunakan kedua alat ukur tersebut:

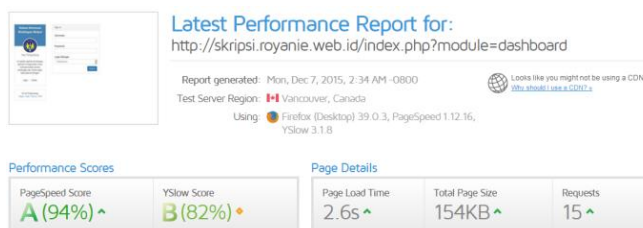
a) Halaman Pengujian Halaman Login



Gambar 41. Grade Pengujian *Efficiency* Halaman *Login*

Pengujian *performance efficiency* halaman *login*, *grade* yang didapatkan adalah A (95%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,4 detik, berarti telah memenuhi standar *Optimize (2010)* yaitu di bawah 7 detik.

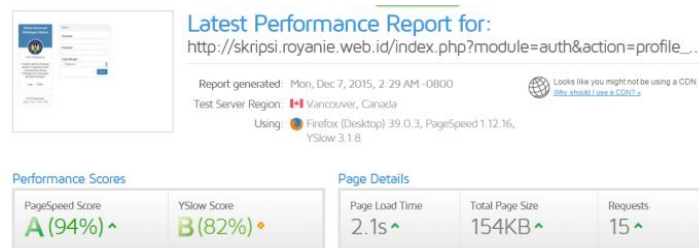
b) Halaman Profil Mahasiswa



Gambar 42. Grade Pengujian *Efficiency* Halaman Profil Mahasiswa

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,6 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

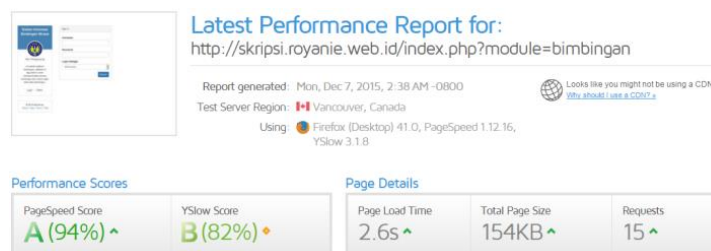
c) Halaman Ubah Profil Mahasiswa



Gambar 43. Grade Pengujian *Efficiency* Halaman Ubah Profil Mahasiswa

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,1 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

d) Halaman Catatan Bimbingan

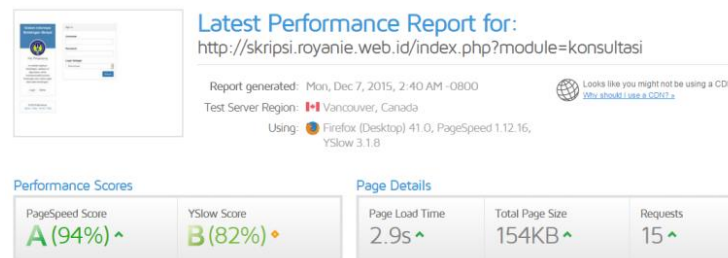


Gambar 44. Grade Pengujian *Efficiency* Halaman Catatan Bimbingan

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini

menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,6 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

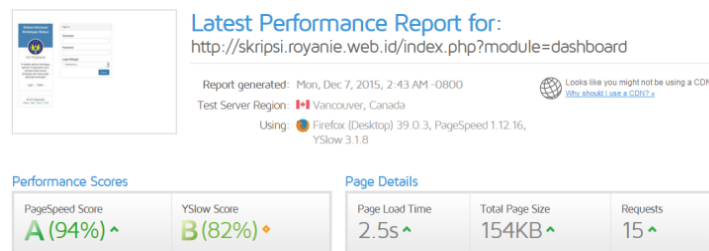
e) Halaman Konsultasi Online



Gambar 45. Grade Pengujian *Efficiency* Halaman Konsultasi Online

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,9 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

f) Halaman Profil Dosen

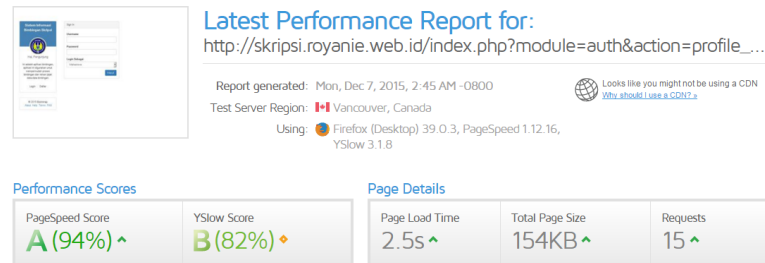


Gambar 46. Grade Pengujian *Efficiency* Halaman Profil Dosen

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat

baik. Waktu untuk memuat halaman selama **2,5** detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

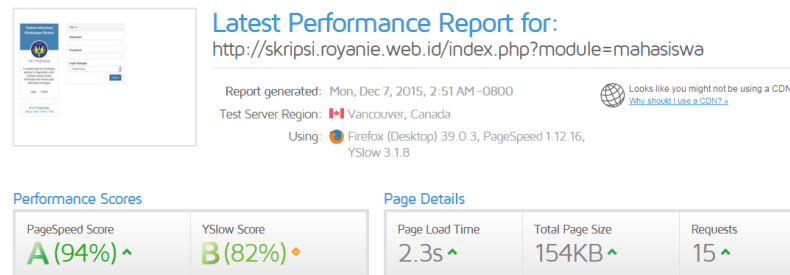
g) Halaman Ubah Profil Dosen



Gambar 47. Grade Pengujian *Efficiency* Halaman Ubah Profil Dosen

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,5 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

h) Halaman Daftar Mahasiswa

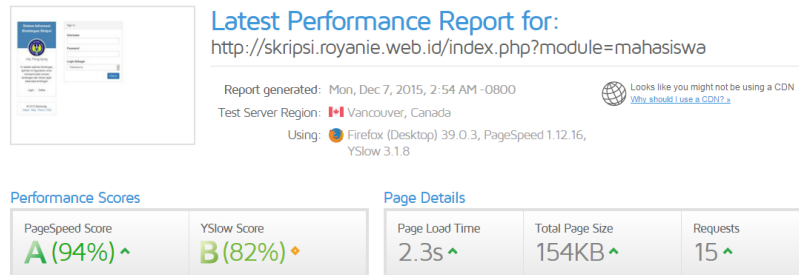


Gambar 48. Grade Pengujian *Efficiency* Halaman Daftar Mahasiswa

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat

baik. Waktu untuk memuat halaman selama 2,3 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

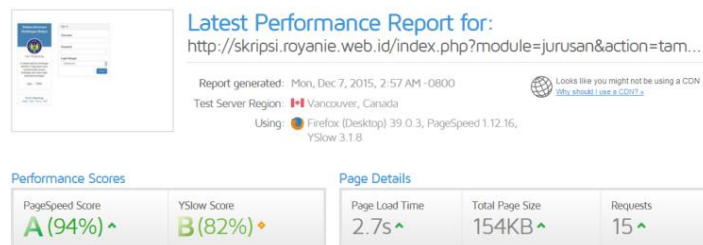
i) Halaman Daftar Dosen



Gambar 49. Grade Pengujian *Efficiency* Halaman Daftar Dosen

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,3 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

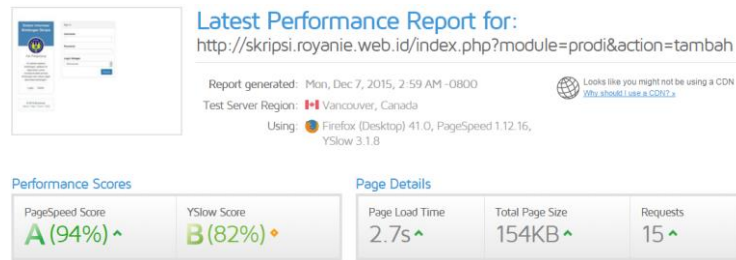
j) Halaman Tambah Data Jurusan



Gambar 50. Grade Pengujian *Efficiency* Halaman Tambah Data Jurusan

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,7 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

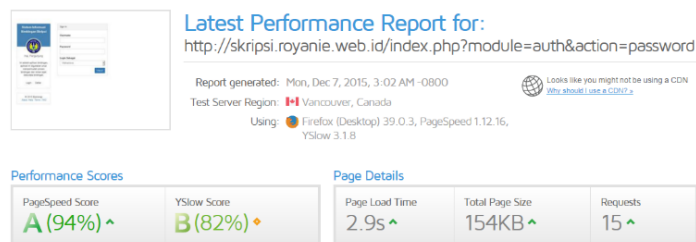
k) Halaman Tambah Data Prodi



Gambar 51. Grade Pengujian *Efficiency* Halaman Tambah Data Prodi

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,7 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

l) Halaman Ganti Password



Gambar 52. Grade Pengujian *Efficiency* Halaman Ganti Password

Pengujian *performance efficiency* halaman *login, grade* yang didapatkan adalah A (94%) untuk Page Speed dan B (82%) untuk YSlow. Grade ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sangat baik. Waktu untuk memuat halaman selama 2,9 detik, berarti telah memenuhi standar Aptimize (2010) yaitu di bawah 7 detik.

Tabel 19. Hasil Pengujian Performance Efficiency

No	Halaman Web	Page Load (detik)	Page Size (Kb)	PageSpeed Grade	Yslow Grade
1.	Halaman <i>Login</i>	2,4	154	A (95%)	B (82%)
2.	Profil Mahasiswa	2,6	154	A (94%)	B (82%)
3.	Ubah Profil Mahasiswa	2,1	154	A (94%)	B (82%)
4.	Catatan Bimbingan	2,6	154	A (94%)	B (82%)
5.	Konsultasi <i>Online</i>	2,9	154	A (94%)	B (82%)
6.	Profil Dosen	2,5	154	A (94%)	B (82%)
7.	Ubah Profil Dosen	2,5	154	A (94%)	B (82%)
8.	Daftar Mahasiswa	2,3	154	A (94%)	B (82%)
9.	Daftar Dosen	2,3	154	A (94%)	B (82%)
10.	Tambah Data Jurusan	2,7	154	A (94%)	B (82%)
11.	Tambah Data Prodi	2,7	154	A (94%)	B (82%)
12.	Ganti Password	2,9	154	A (94%)	B (82%)
	Rata-rata	2,6	154	A(94.08%)	B(82%)

3. Hasil Pengujian *Compatibility*

Pengujian *compatibility* tidak dilakukan, karena tidak ada perangkat lunak/*software* pembanding untuk menguji faktor ini.

4. Hasil Pengujian *Usability*

Pengujian aspek *usability* menggunakan angket *USE Questionnaire* dengan jumlah 30 butir. Pengujian ini dihitung menggunakan skala Likert yang diberikan kepada 20 responden. Hasil pengujian *usability* terdapat pada tabel berikut:

Tabel 20. Hasil Pengujian Usability

No. Pertanyaan	STS	TS	R	S	SS
1	0	0	0	13	7
2	0	0	2	13	5
3	0	0	0	10	10
4	0	0	10	6	4
5	0	0	6	12	2
6	0	0	0	7	13
7	0	1	0	15	4
8	0	0	2	15	3
9	0	0	0	2	18
10	0	0	0	10	10
11	0	0	0	9	11
12	0	0	2	12	6
13	0	0	3	13	4
14	0	0	0	11	9
15	0	0	4	10	6
16	0	0	6	11	3
17	0	0	2	12	6
18	0	0	4	13	3
19	0	0	4	14	2
20	0	0	0	8	12
21	0	0	0	6	14
22	0	0	0	8	12
23	0	0	2	12	6
24	0	0	2	9	9
25	0	0	2	12	6
26	0	0	0	18	2
27	0	0	0	16	4
28	0	0	2	15	3
29	0	0	0	12	8
30	0	0	0	8	12
Total	0	1	53	332	214

Berdasarkan tabel di atas, total jawaban sangat tidak setuju (STS) berjumlah 0, tidak setuju (TS) berjumlah 1, ragu-ragu (R) berjumlah 53, setuju (S) berjumlah 332, dan sangat setuju (SS) berjumlah 214. Kemudian hasil tersebut dihitung dengan perhitungan sebagai berikut :

$$Skor_{total} = (214 \times 5) + (332 \times 4) + (53 \times 3) + (1 \times 2) + (0 \times 1) = 2559$$

$$P_{skor} = \frac{2559}{3000} \times 100\% = 85,30\%$$

Hasil presentasi pada aspek *usability* adalah 85,30%, kemudian dibandingkan dengan Tabel 6, sehingga bisa disimpulkan hasil pengujian *usability* sistem informasi bimbingan skripsi dapat dinyatakan **sangat layak** dan memenuhi aspek *usability*. Hasil pengujian *usability* juga dihitung nilai reliabilitasnya menggunakan software SPSS dengan perhitungan Alpha Cronbach dan hasilnya dapat dilihat pada gambar dibawah ini.

Case Processing Summary

		N	%
Cases	Valid	20	100.0
	Excluded ^a	0	.0
	Total	20	100.0

a. Listwise deletion based on all variables in the procedure.

Reliability Statistics

Cronbach's Alpha	N of Items
.940	30

Gambar 53. Hasil Perhitungan Reliabilitas dengan Alpha Cronbach

Hasil perhitungan nilai reliabilitas menggunakan SPSS yaitu sebesar 0,94. Kemudian nilai tersebut dibandingkan dengan nilai di tabel konsistensi Alpha Cronbach pada tabel 13 di bab 3, maka diperoleh hasil yang menunjukkan bahwa nilai reliabilitas berkategori **Excellent**.

Jika dibandingkan dengan menggunakan *r product moment* (Sutanto, 2007), dengan jumlah responden 20 dengan signifikansi 5%, didapat nilai sebagai berikut : $df = n-2 = 20-2 = 18$.

Dengan $df = 18$ dengan signifikansi 5%, maka didapat $r_{tabel} = 0,444$. Hasil dikatakan reliable jika $r_{hitung} > r_{tabel}$. Dikarekankan $0,94 > 0,444$, maka $r_{hitung} > r_{tabel}$. Sehingga dapat disimpulkan bahwa instrumen *usability* dengan *USE Questionnaire* sudah *reliable*.

5. Hasil Pengujian *Reliability*

Pengujian aspek *reliability* dilakun dengan menggunakan Loadimpact.

URL	Load zone	User scenario	Successful	Failed
+ ahkamiy.com/./bootstrap-datetimpicke..	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./logo.gif	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./museum-pendidikan.jpg	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./jquery.dataTables.min.js	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./moment.min.js	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./application.css	Palo Alto, US (Amazon)	Auto generated scenario	124	0

Gambar 54. Hasil pengujian *Reliability* menggunakan Loadimpact

Dari request sebanyak 1861 request yang ada di dalam aplikasi Sistem informasi bimbingan ini, semua sukses diakses dan tidak ada yang gagal.

+ ahkamiy.com/./bootstrap.min.css	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./toolkit.js	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./jquery.dataTables.min.css	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./toolkit.css	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ http://ahkamiy.com/	Palo Alto, US (Amazon)	Auto generated scenario	125	0
+ ahkamiy.com/./jquery.min.js	Palo Alto, US (Amazon)	Auto generated scenario	124	0
+ ahkamiy.com/./bootstrap-datetimpicke..	Palo Alto, US (Amazon)	Auto generated scenario	124	0

Gambar 55. Hasil pengujian *reliability* menggunakan LoadImpact

Sesuai dengan perhitungan nilai *reliability* Model Nelson menggunakan LoadImpact maka hasil dihitung sebagai berikut.

$$R1 = 1 - \frac{ne}{n}$$

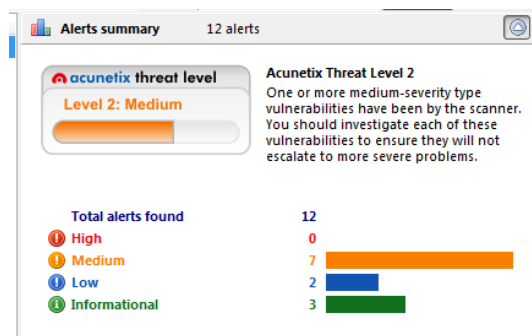
$$R1 = 1 - \frac{0}{1861}$$

$$R1 = 1$$

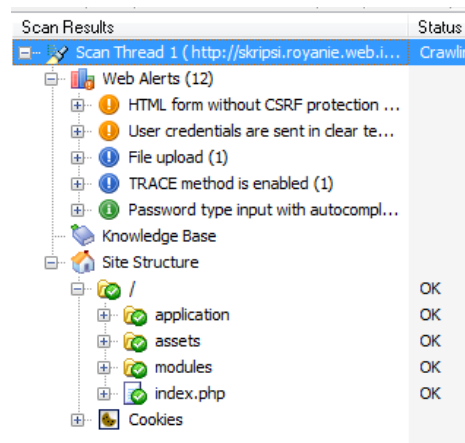
Dari hasil penghitungan, didapatkan bahwa nilai uji *reliability* menggunakan LoadImpact adalah 1 atau 100%. Sehingga bisa dikatakan bahwa sistem informasi telah memenuhi aspek *reliability*.

6. Hasil Pengujian *Security*

Pengujian *Security* menggunakan aplikasi Acunetix Web Vulnerability Scanner 8.



Gambar 56. Laporan Pengujian *Security*



Gambar 57. Hasil Scan dengan Acunetix Web Vulnerability Scanner

Berdasarkan hasil pengujian pada Gambar 56, dapat dilihat bahwa tidak terdapat peringatan *Cross Site Scripting (XSS)* dan *SQL Injection*. Sehingga dapat disimpulkan bahwa aplikasi pengelolaan data aplikasi tidak terdapat kendala pada

Cross Site Scripting (XSS) dan *SQL Injection*, dan memiliki tingkat keamanan level 2. Keamanan dengan tingkat level 2 menandakan adanya kesalahan pada konfigurasi dan pengkodean pada aplikasi sehingga terjadi gangguan pada server (Acunetix User Manual Book, 2013).

7. Hasil Pengujian *Maintainability*

Pengujian aspek *maintainability* dilakukan dengan menggunakan PHP Copy/paste Detector (PHPCpD). Dan didapatkan hasil pada gambar di bawah ini.

```
phpcpd 2.0.0 by Sebastian Bergmann.  
  
Found 2 exact clones with 52 duplicated lines in 3 files:  
  
- /Users/januari.pri.n/Development/html/xin/Aplikasi/modules/auth/controller.php:18-39  
  /Users/januari.pri.n/Development/html/xin/Aplikasi/modules/dosen/controller.php:39-60  
  
- /Users/januari.pri.n/Development/html/xin/Aplikasi/modules/auth/controller.php:69-100  
  /Users/januari.pri.n/Development/html/xin/Aplikasi/modules/mahasiswa/controller.php:64-95  
  
0,57% duplicated lines out of 9182 total lines of code.  
  
Time: 212 ms, Memory: 11,50Mb
```

Gambar 58. Hasil screenshot menggunakan PHP Copy/Paste Detector

Hasil dari analisis *code duplication* dari *source code* sistem informasi bimbingan adalah 0,57%. Kemudian hasil tersebut dihitung menggunakan rumus dari TIOBE Quality Indicator, perhitungan pengujiannya sebagai berikut :

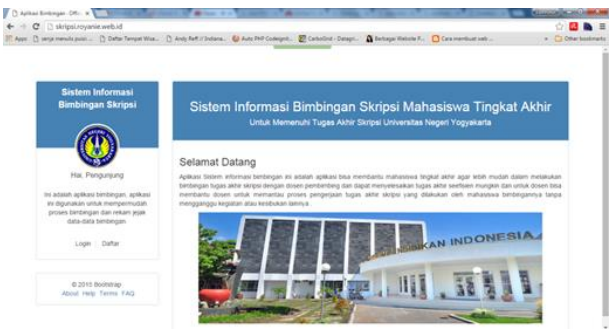


$$Skor = \min(-30 \times \log_{10}(0.57\%) + 60, 100) = 67,3\%$$

Skor hasil perhitungan adalah 67,3% Jika dibandingkan dengan tabel 4 maka sistem informasi dapat dikatakan **cukup** untuk memenuhi aspek *maintainability*.



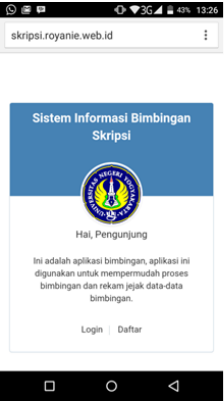


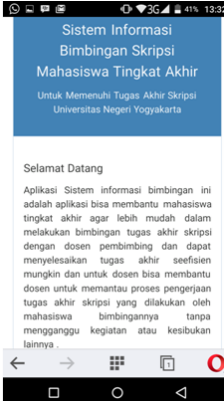
8. Hasil Pengujian *Portability*

Pengujian *Portability* dilakukan dengan menjalankan aplikasi di beberapa *browser* meliputi desktop maupun *mobile*. Hasil pengujiannya dapat dilihat pada Tabel 20 berikut.

Tabel 21. Tabel hasil pengujian *portability* di *browser*

No.	Browser	Hasil
1	<i>Google Chrome</i>	
2	<i>Mozilla Firefox</i>	
3	<i>Opera browser</i>	

Sambungan dari Tabel 21

No.	Browser	Hasil
4	<i>Chrome mobile</i>	 
5	<i>Android browser</i>	 
6	<i>Opera mobile</i>	 

E. Pembahasan Hasil Penelitian

1. Pembahasan Hasil Pengujian Faktor *Functionality Suitability*

Berdasarkan pengujian kualitas perangkat lunak aspek *functionality suitability*, didapatkan hasil pengujian dengan nilai $X = 1$, sehingga bisa disimpulkan bahwa perangkat lunak sistem informasi memenuhi aspek *functionality suitability* dan memiliki kualitas yang baik.

2. Pembahasan Hasil Pengujian Faktor *Performance Efficiency*

Berdasarkan pengujian kualitas perangkat lunak aspek *performance efficiency*, didapatkan hasil pengujian dengan rata-rata waktu aksesnya selama 2,6 detik. Sesuai dengan yang diungkapkan oleh Nielsen (2010) dan standard Aptimize (2010), jika waktu yang dibutuhkan kurang dari 10 detik, maka perangkat lunak telah memenuhi aspek *performance efficiency*.

3. Pembahasan Hasil Pengujian Faktor *Compatibility*

Pengujian *compatibility* tidak dilakukan karena tidak ada perangkat lunak perbandingan yang digunakan untuk mengujinya, sehingga tingkat keberhasilan pada faktor *compatibility* adalah 100%, yang berarti memenuhi faktor *compatibility*.

4. Pembahasan Hasil Pengujian Faktor *Usability*

Berdasarkan pengujian kualitas perangkat lunak aspek *usability*, didapatkan hasil pengujian aplikasi sistem informasi sebesar 85,30%, hal tersebut menunjukkan bahwa perangkat lunak sangat layak dan berdasarkan perhitungan Alpha Cronbach menghasilkan nilai sebesar 0,94 yang berarti memiliki kategori *excellent*.

5. Pembahasan Hasil Pengujian Faktor *Reliability*

Berdasarkan pengujian kualitas perangkat lunak aspek *reliability*, didapatkan hasil pengujian aplikasi sistem informasi sebesar 1, sehingga sistem informasi bimbingan telah memenuhi faktor *reliability*

6. Pembahasan Hasil Pengujian Faktor *Security*

Berdasarkan pengujian kualitas perangkat lunak aspek *security*, didapatkan hasil bahwa perangkat lunak memiliki tingkat keamanan level 2 atau medium.

7. Pembahasan Hasil Pengujian Faktor *Maintainability*

Berdasarkan pengujian kualitas perangkat lunak aspek *maintainability*, didapatkan hasil bahwa aplikasi sistem informasi dinilai cukup untuk memenuhi factor *maintainability*.

8. Pembahasan Hasil Pengujian Faktor *Portability*

Berdasarkan hasil pengujian kualitas perangkat lunak aspek *portability* dengan menggunakan beberapa *browser* berbeda pada *desktop* maupun *mobile*, didapatkan hasil bahwa perangkat lunak telah memenuhi aspek *portability*.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang dilakukan, bisa disimpulkan bahwa :

1. Penelitian menghasilkan aplikasi sistem informasi bimbingan yang bisa membantu dosen dan mahasiswa untuk memantau progress pengerjaan skripsi, mengefektifkan waktu bimbingan, dan bimbingan terjadwal dengan baik. Pengembangan aplikasi menggunakan PHP dan *bootstrap*, dimana model pengembangannya menggunakan model *waterfall* yang terdiri dari tahap:
 - a) Analisis kebutuhan,
 - b) Desain
 - c) Pengkodean
 - d) Pengujian
2. Hasil pengujian kualitas sistem informasi bimbingan tugas akhir berdasarkan ISO/IEC 25010 adalah sebagai berikut :
 - a) Kualitas perangkat lunak aspek *functional suitability* menunjukkan bahwa sistem informasi telah memenuhi standard, dikarenakan fungsi berjalan semua dan nilai X=1.
 - b) Perangkat lunak telah memenuhi aspek *performance efficiency*, dikarenakan rata-rata waktu yang dibutuhkan untuk mengakses halaman 2,6 detik.
 - c) Perangkat lunak telah memenuhi aspek *usability* dengan kategori Sangat layak.

- d) Perangkat lunak telah memenuhi standard nilai *reliability*, dengan nilai 1 atau 100%.
- e) Perangkat lunak telah memenuhi aspek *security* dengan keamanan tingkat 2 atau medium.
- f) Perangkat lunak telah memenuhi aspek *maintainability* dengan kategori cukup.
- g) Perangkat lunak telah memenuhi aspek *portability*, karena bisa dijalankan pada beberapa browser berbeda, baik menggunakan desktop atau *mobile device*.

B. Saran

Mengingat keterbatasan yang dimiliki penulis dalam melakukan pengembangan sistem ini, maka penulis menyarankan untuk pengembangan penelitian di masa datang sebagai berikut :

1. Adanya fitur tambahan yang memudahkan dalam pengelolaan, seperti notifikasi melalui email ataupun melalui sms.
2. Menggunakan database yang telah terintegrasi dengan pihak fakultas, jurusan, atau universitas.
3. Penggunaan tampilan yang lebih interaktif.

DAFTAR PUSTAKA

- Acunetix. (2013). *Acunetix User Manual Book*. Dipetik Maret 25, 2015, dari <https://http://www.acunetix.com/resources/wvsmanual.pdf>
- Akadun. (2009). *Teknologi Informasi Administrasi*. Bandung: Alfabeta.
- Asthana, A. &. (2009). *Quantifying Software Reliability and Readiness*. Dipetik Desember 31, 2015, dari <http://www.asq509.org/ht/a/GetDocumentAction/i/46515>
- Burch, J. G. (1989). *Information Systems Theory and Practice, Fifth Edition*. New York: John Wiley & Sons.
- Cornett, S. (1996). *Code Coverage Analysis*. Dipetik Maret 9, 2015, dari <http://www.bullseye.com/coverage.html>
- Developers, G. (2013). *PageSpeed Insights Rules*. Dipetik Maret 3, 2015, dari <https://developers.google.com/speed/docs/insights/rules>
- Fatkhurrokhman, M. (2014). *Analisis Pengujian Sistem Informasi Akademik STMIK El Rahma Yogyakarta menggunakan International Organization for Standardization (ISO 9126)*. Dipetik Februari 6, 2016, dari https://www.academia.edu/7335602/Analisis_Pengujian_Sistem_Informasi_Akademik_STMIK_El_Rahma_Yogyakarta_menggunakan_International_Organization_for_Standardization_ISO_9126
- Fatta, H. A. (2007). *Analisis dan Perancangan Sistem Informasi*. Yogyakarta: Penerbit ANDI.
- Galea, J. (2012). *Optimize your Site's Performance with GTmetrix for WordPress*. Dipetik Desember 8, 2015, dari <http://www.wpmayor.com/optimize-sites-performance-gtmetrix-wordpress/>
- Guritno, S. S. (2011). *Theory and Application of IT Research: Metodologi Penelitian Teknologi Informasi*. Yogyakarta: Penerbit Andi.
- Hass, A. M. (2014). *Guide to Advanced Software Testing, Second Edition*. New York: Artech House.
- Hastono, S. P. (2006). *Analisis Data*. Jakarta: Universitas Indonesia.
- Hitung, R. (2013). *Tabel R Statistika Dan Cara Membacanya*. Dipetik Januari 31, 2016, dari <http://rumushitung.com/2013/06/08/tabel-r-statistika-dan-cara-membacanya/>

- ISO. (2015). *ISO - 25010*. Dipetik September 16, 2016, dari <http://iso25000.com/index.php/en/iso-25000-standards/iso-25010>
- Jansen, P. (2015). *The TIOBE Quality Indicator*. Dipetik November 30, 2015, dari <http://www.tiobe.com/content/paperinfo/TIOBEQualityIndicator.pdf>
- Januarti, R. (2009). Hubungan antara Persepsi terhadap Dosen Pembimbing dengan Tingkat Stress dalam Menulis Skripsi. *Univerversitas Muhammadiyah Surakarta*.
- Kusrini, M. K. (2007). *Strategi Perancangan dan Pengelolaan Basis Data*. Yogyakarta: Penerbit Andi.
- Lund, A. M. (2001). *Measuring Usability with the USE Questionnaire*. Dipetik Oktober 2015, 8, dari <http://garyperlman.com/quest/quest.cgi?form=USE>
- Nielsen, J. (2012). *How Many Test Users in a Usability Study?* Dipetik September 15, 2015, dari <http://www.nngroup.com/articles/quantitative-studies-how-many-users/>
- Nugroho, A. (2010). *Rekayasa Perangkat Lunak Berorientasi Objek Dengan Metode USDP*. Yogyakarta: Penerbit Andi.
- Nugroho, A. (2010). *Rekayasa Perangkat Lunak Menggunakan UML & Java*. Yogyakarta: Penerbit Andi.
- Rahardjo, B. (2004). *Pemanfaatan Teknologi Informasi di Perguruan Tinggi*. Bandung: Institut Teknologi Bandung.
- Salonen, V. (2012). *Automatic Portability Testing*. Dipetik November 30, 2015, dari <https://jyx.jyu.fi/dspace/bitstream/handle/123456789/40043/URN:NBN:fi:ju-201210212735.pdf?sequence=1>
- Sameer. (2009). *Detecting Duplicate Code in PHP Files*. Dipetik November 31, 2015, dari <http://www.codediesel.com/tools/detecting-duplicate-code-in-php-files/>
- Shanmugam, L. &. (2012). *An Oveview of Software Reliability Models*. Dipetik Oktober 8, 2015, dari http://www.ijarcsse.com/docs/papers/10_October2012/Volume_2_issue_10_October2012/V2I10-0043.pdf
- Simarmata, J. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit Andi.

Sugiyono. (2011). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung: Alfabeta.

Tegoh, W. (2004). *Sistem Informasi*. Yogyakarta: Graha Ilmu.

Widodo, H. &. (2011). *Menggunakan UML*. Bandung: Informatika.

Winkel, W. (1991). *Bimbingan & Konseling di institusi Pendidikan*. Jakarta: Grasindo.

LAMPIRAN

**SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI**

Saya yang bertanda tangan dibawah ini:

Nama : Dessy Irmawati, M. T
NIP : 19791214 201012 2 002
Jurusan : Pendidikan Teknik Informatika

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Zumrotul Ahkamiyati
NIM : 10520241027
Program Studi : Pendidikan Teknik Informatika
Judul TAS : Analisis Pengembangan Sistem Informasi Bimbingan Tugas Akhir Skripsi Online untuk Mahasiswa Tingkat Akhir

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

- Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 3... Oktober 2015

Validator,





Dessy Irmawati, M. T

NIP. 19791214 201012 2 002

Catatan:

Beri tanda ✓

FORM TESTING APLIKASI SISTEM INFORMASI BIMBINGAN SKRIPSI

No	Tanggal	Nama Dosen	Kesan	Catatan	Tanda Tangan
1	15/9/15	Djolew Santosa	ide baik	alasan lanjut and bisa dimanfaatkan	
2	5/10/15	Dessy Irmawati	Sistem yang dapat memudahkan bimbingan	Data dapat dieksport excel, ada notifikasi untuk data baru	
3					
4					
5					

Mengetahui,
Dosen Pembimbing

Handaru Jati, Ph. D
19740511 199903 1 002

INSTRUMEN PENGUJIAN ASPEK *FUNCTIONALITY SUITABILITY*

SISTEM INFORMASI BIMBINGAN SKRIPSI *ONLINE* UNTUK MAHASISWA TINGKAT AKHIR

A. Identitas

Nama : AGUNG JATI KUSUMO
 Profesi : PROGRAMMER
 Instansi : AL-FATHIMDO

B. Petunjuk Pengisian

Berilah tanda centang pada kolom yang disediakan sesuai dengan penilaian untuk pengujian *functional suitability* dengan ketentuan :

Ya = Jika fungsi **dapat** dioperasikan dengan benar.

Tidak = Jika fungsi **tidak dapat** dioperasikan dengan benar.

1. Instrument *functional completeness* dan *functional appropriateness*

No.	Fungsi	Hasil yang Diharapkan	Tarf Kecapaian	
			Ya	Tidak
I.	Admin			
1.	Login	Fungsi untuk login sudah berjalan dengan benar	✓	
2.	Edit Password	Fungsi untuk mengedit password sudah berjalan dengan benar	✓	
3.	Logout	Fungsi untuk logout sudah berjalan dengan benar	✓	
4.	Data Bimbingan	Fungsi melihat data bimbingan mahasiswa dan dosen sudah berjalan dengan benar	✓	
5.	Memilih dosen	Fungsi untuk menentukan dosen pembimbing bagi mahasiswa sudah berjalan dengan benar	✓	
6.	Data Konsultasi	Fungsi melihat data konsultasi online mahasiswa dan dosen sudah berjalan dengan benar	✓	
7.	Data Mahasiswa	Fungsi untuk menambahkan, menampilkan, mengubah, dan menghapus data mahasiswa sudah berjalan dengan benar	✓	
8.	Data Dosen	Fungsi untuk menambahkan, menampilkan, mengubah, dan menghapus data dosen sudah berjalan dengan benar	✓	
9.	Data Prodi	Fungsi untuk menambahkan, menampilkan, mengubah, dan menghapus data prodi sudah berjalan dengan benar	✓	
10.	Data Jurusan	Fungsi untuk menambahkan, menampilkan,	✓	

INSTRUMEN PENGUJIAN ASPEK *USABILITY*

SISTEM INFORMASI BIMBINGAN SKRIPSI ONLINE UNTUK MAHASISWA TINGKAT AKHIR

A. Identitas

Nama : Linia Laras Kartanti
 Profesi : Mahasiswa
 Instansi : UMY

B. Petunjuk Pengisian

Lingkari angka pada kolom skala yang disediakan sesuai dengan penilaian untuk pengujian usability, dengan ketentuan sebagai berikut :

- 1 = Sangat Tidak Setuju**
- 2 = Tidak Setuju**
- 3 = Ragu-ragu**
- 4 = Setuju**
- 5 = Sangat Setuju**

No.	INSTRUMENT	Skala				
		1	2	3	4	5
Usefulness						
1.	Aplikasi ini membantu saya menjadi lebih efektif dalam proses bimbingan mahasiswa	1	2	3	4	5
2.	Aplikasi ini membantu saya menjadi lebih produktif ketika proses bimbingan mahasiswa	1	2	3	4	5
3.	Aplikasi ini bermanfaat untuk membantu proses bimbingan mahasiswa	1	2	3	4	5
4.	Aplikasi ini memberi saya dampak yang besar terhadap tugas yang saya lakukan dalam proses bimbingan mahasiswa	1	2	3	4	5
5.	Aplikasi ini memudahkan saya mencapai hal-hal yang saya inginkan dalam proses bimbingan mahasiswa	1	2	3	4	5
6.	Aplikasi ini menghemat waktu ketika saya menggunakannya	1	2	3	4	5
7.	Aplikasi ini sesuai dengan kebutuhan saya	1	2	3	4	5
8.	Aplikasi ini bekerja sesuai dengan apa yang saya harapkan	1	2	3	4	5
Easy of Use						
9.	Aplikasi ini mudah digunakan	1	2	3	4	5
10.	Aplikasi ini praktis untuk digunakan	1	2	3	4	5
11.	Aplikasi ini mudah dipahami	1	2	3	4	5