

**PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN
BERBASIS WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA**

TUGAS AKHIR SKRIPSI

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



**Disusun Oleh:
Husin Nanda Perwira
NIM. 11520241039**

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2015**

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA

Disusun oleh:

Husin Nanda Perwira
NIM 11520241039

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,

Muhammad Munir, M.Pd.
NIP. 19630512 198901 1 001

Yogyakarta, 30 Juni 2015
Disetujui,
Dosen Pembimbing,

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

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Husin Nanda Perwira

NIM : 11520241039

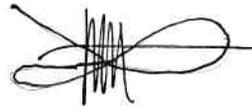
Program Studi: Pendidikan Teknik Informatika

Judul TAs : Pengembangan Sistem Informasi Perpustakaan Berbasis Web di SMK Muhammadiyah 1 Yogyakarta

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, 30 Juni 2015

Yang menyatakan,



Husin Nanda Perwira
NIM. 11520241039

HALAMAN PENGESAHAN
Tugas Akhir Skripsi

**PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB
DI SMK MUHAMMADIYAH 1 YOGYAKARTA**

Disusun oleh:
Husin Nanda Perwira
NIM 11520241039

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta
Pada tanggal 2 September 2015

TIM PENGUJI

| Nama/Jabatan | Tanda Tangan | Tanggal |
|---|---|-----------|
| Handaru Jati, Ph.D. Ketua Penguji/Pembimbing |  | 22/9 2015 |
| Totok Sukardiyono, MT. Sekretaris |  | 22/9 2015 |
| Adi Dewanto, M.Kom. Penguji |  | 22/9/2015 |

Yogyakarta, September 2015

Fakultas Teknik Universitas Negeri Yogyakarta
Dekan,




Dr. Moch Bruri Triyono
NIP. 19560216 198603 1 003

HALAMAN MOTTO

“Allah akan mengangkat kedudukan orang-orang yang beriman dan diberikan ilmu di antara kalian beberapa derajat. Allah Maha mengetahui apa yang kalian kerjakan” (QS. Al Mujadilah [58]: 11)

“Barangsiapa yang menempuh suatu jalan dalam rangka menuntut ilmu, maka Allah akan memudahkan baginya jalan menuju surga” (H.R Muslim)

“Siapa yang terpisah dari penuntun jalannya, maka tentu ia bisa tersesat. Tidak ada penuntun yang terbaik bagi kita selain dengan mengikuti ajaran Rasul shallallahu ‘alaihi wa sallam” (Ibnu Taimiyah)

“Orang yang beramal tanpa ilmu seperti orang yang berjalan bukan pada jalan yang sebenarnya. Orang yang beramal tanpa ilmu hanya membuat banyak kerusakan dibanding mendatangkan kebaikan. Tuntutlah ilmu dengan sungguh-sungguh, namun jangan sampai meninggalkan ibadah. Gemarlah pula beribadah, namun jangan sampai meninggalkan ilmu. Karena ada segolongan orang yang rajin ibadah, namun meninggalkan belajar” (Al Hasan Al Bashri)

HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, Rabb semesta alam yang senantiasa memberikan karunia sehingga penulis mampu menyelesaikan penulisan skripsi ini.

Karya ini saya persembahkan kepada:

1. Ayah dan Ibu serta keluarga tercinta yang selalu mendidik saya, memberikan do'a, dukungan, nasihat dan semangat yang tiada henti.
2. Lalu Satriawan Kholid, Ali Hasan Azis, Azis Amirul Bahar, Zein Syahida Kusuma, Tri Nugroho dan Rama Bramantara terimakasih untuk selalu memberikan do'a, dukungan dan bantuan dalam menyelesaikan skripsi ini.
3. Keluarga Informatika E 2011 UNY yang selalu memberikan keceriaan, kebersamaan dan motivasi.
4. Seluruh teman UNY yang telah memberikan do'a, dukungan, dan semangat.

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA

Oleh:

Husin Nanda Perwira

NIM 11520241039

ABSTRAK

Pengelolaan proses peminjaman dan pengembalian buku perpustakaan dapat memakan banyak biaya dan waktu karena perlu pencatatan oleh siswa yang harus dilakukan secara manual. Pemanfaatan sistem informasi perpustakaan berbasis *web* dapat menghemat biaya dan waktu karena proses sudah dilakukan menggunakan komputer. Penelitian ini bertujuan untuk: (1) membuat sistem informasi perpustakaan berbasis *web* di SMK Muhammadiyah 1 Yogyakarta, dan (2) mengetahui standar kualitas *software* yang dikembangkan berdasarkan standar kualitas ISO 9126 pada aspek *functionality*, *efficiency*, *reliability*, dan *usability*.

Metode penelitian yang digunakan adalah *Research and Development* (R&D). Model proses pengembangan *software* digunakan model *waterfall* yang terdiri dari: (1) analisis kebutuhan, (2) desain, (3) implementasi, dan (4) pengujian. Kemudian digunakan pemodelan visual UML yaitu standarisasi bahasa pemodelan untuk pembangunan perangkat lunak berorientasi objek.

Hasil penelitian diketahui bahwa: (1) sistem informasi perpustakaan berbasis *web* dikembangkan menggunakan *framework* Laravel dan model pengembangan *waterfall* terdiri dari tahap analisis kebutuhan, tahap desain, tahap implementasi, dan tahap pengujian, dan (2) hasil pengujian pada aspek *functionality* diperoleh nilai 1 (baik) dan tingkat keamanan untuk serangan *SQL Injection* dan *XSS (Cross Site Scripting)* diperoleh level sangat aman (low). Pengujian *efficiency*, diperoleh rata-rata *load time* 2,21 detik (diterima). Pengujian *reliability* diperoleh nilai *reliability* sebesar 100% (baik). Pengujian *usability* diperoleh tingkat persetujuan pengguna sebesar 77% (setuju) dengan nilai *alpha cronbach* sebesar 0,958 (*excellent*).

Kata Kunci : Kualitas *Software*, Sistem Informasi Perpustakaan, UML, ISO 9126.

KATA PENGANTAR

Puji syukur kepada Allah SWT atas rahmat, hidayah, dan karunia-Nya kepada kita semua sehingga kami dapat menyelesaikan Tugas Akhir Skripsi dengan judul “Pengembangan Sistem Informasi Perpustakaan Berbasis Web di SMK Muhammadiyah 1 Yogyakarta”. Tugas Akhir Skripsi ini disusun sebagai salah satu syarat untuk mendapatkan gelar Sarjana Pendidikan di Fakultas Teknik Universitas Negeri Yogyakarta.

Penulis menyadari dalam penyusunan proposal skripsi ini tidak akan selesai tanpa bantuan dari berbagai pihak. Oleh karena itu pada kesempatan ini kami ingin mengucapkan terima kasih kepada yang terhormat:

1. Handaru Jati, Ph.D selaku Dosen Pembimbing TAS yang telah banyak membantu selama penyusunan Tugas Akhir Skripsi ini.
2. Handaru Jati, Ph.D, Totok Sukardiyono, MT, Adi Dewanto, M.Kom selaku Ketua Penguji, Sekretaris dan Penguji yang memberikan koreksi perbaikan secara komprehensif terhadap TAS ini.
3. Muhammad Munir, M.Pd selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Ketua Program Studi Pendidikan Teknik Informatika beserta dosen dan staf yang telah memberikan bantuan dan fasilitas selama proses penyusunan pra proposal sampai dengan selesainya TAS ini.
4. Dr. Moch. Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi.
5. Drs. H. Suprihandono, M.M. selaku Kepala SMK Muhammadiyah 1 Yogyakarta yang telah memberi ijin dan bantuan dalam pelaksanaan penelitian Tugas Akhir Skripsi ini.

6. Para guru dan siswa SMK Muhammadiyah 1 Yogyakarta yang telah memberi bantuan memperlancar pengambilan data selama proses penelitian Tugas Akhir Skripsi ini.
7. Semua pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan di sini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

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, 30 Juni 2015
Penulis,

Husin Nanda Perwira
NIM 11520241039

DAFTAR ISI

| | Halaman |
|--|-------------|
| LEMBAR PERSETUJUAN | ii |
| SURAT PERNYATAAN | iii |
| HALAMAN PENGESAHAN | iv |
| HALAMAN MOTTO | v |
| HALAMAN PERSEMBAHAN | vi |
| ABSTRAK | vii |
| KATA PENGANTAR | viii |
| DAFTAR ISI | x |
| DAFTAR GAMBAR | xiii |
| DAFTAR TABEL | xvii |
| DAFTAR LAMPIRAN | xix |
| | |
| BAB I PENDAHULUAN | 1 |
| A. Latar Belakang | 1 |
| B. Identifikasi Masalah | 4 |
| C. Batasan Masalah..... | 4 |
| D. Rumusan Masalah..... | 5 |
| E. Tujuan Penelitian | 5 |
| F. Manfaat Penelitian | 5 |
| BAB II KAJIAN PUSTAKA | 7 |
| A. Deskripsi Teori..... | 7 |
| 1. Pengertian Sistem Informasi..... | 7 |
| 2. Sistem Informasi Perpustakaan | 9 |
| 3. <i>Laravel Framework</i> | 10 |
| 4. Unified Modeling Language (UML) | 14 |

| | |
|---|-----------|
| 5. Waterfall Process Model | 15 |
| 6. Pengujian Black Box..... | 16 |
| 7. Kualitas Software ISO 9126..... | 18 |
| B. Penelitian yang Relevan | 27 |
| C. Kerangka Berpikir | 28 |
| D. Pertanyaan Penelitian..... | 31 |
| BAB III METODE PENELITIAN..... | 32 |
| A. Model Pengembangan..... | 32 |
| B. Prosedur Pengembangan | 33 |
| 1. Analisis Kebutuhan Sistem | 33 |
| 2. Desain Sistem | 33 |
| 3. Implementasi/Code Sistem..... | 34 |
| 4. Pengujian Sistem..... | 34 |
| C. Sumber Data/Subjek Penelitian | 35 |
| D. Metode dan Alat Pengumpulan Data | 35 |
| E. Instrumen Penelitian | 36 |
| 1. Instrumen Aspek Functionality | 36 |
| 2. Instrumen Aspek Efficiency..... | 38 |
| 3. Instrumen Aspek Reliability..... | 38 |
| 4. Instrumen Aspek Usability | 39 |
| F. Teknik Analisis Data | 40 |
| 1. Analisis Kualitas Aspek Functionality | 40 |
| 2. Analisis Kualitas Aspek Efficiency..... | 41 |
| 3. Analisis Kualitas Aspek Reliability..... | 42 |
| 4. Analisis Kualitas Aspek Usability | 42 |
| BAB IV HASIL PENELITIAN DAN PEMBAHASAN | 44 |
| A. Tahap Analisis Kebutuhan | 44 |
| 1. Analisis Kebutuhan Sistem | 44 |

| | |
|---|-----------|
| 2. Analisis Kebutuhan Hardware dan Software | 45 |
| B. Tahap Desain | 45 |
| 1. Perancangan <i>Unified Modeling Language</i> (UML) | 45 |
| 2. Perancangan Basis Data (<i>Database</i>)..... | 61 |
| 3. Perancangan Desain <i>Interface</i> Sistem..... | 62 |
| C. Tahap Implementasi | 66 |
| 1. Implementasi Fungsi..... | 67 |
| 2. Implementasi Basis Data | 69 |
| D. Tahap Pengujian Sistem..... | 73 |
| 1. Pengujian Aspek <i>Functionality</i> | 73 |
| 2. Pengujian Aspek <i>Efficiency</i> | 75 |
| 3. Pengujian Aspek <i>Reliability</i> | 79 |
| 4. Pengujian Aspek <i>Usability</i> | 80 |
| E. Pembahasan Hasil Penelitian | 81 |
| 1. Analisis Pengujian Aspek <i>Functionality</i> | 81 |
| 2. Analisis Pengujian Aspek <i>Efficiency</i> | 82 |
| 3. Analisis Pengujian Aspek <i>Reliability</i> | 82 |
| 4. Analisis Pengujian Aspek <i>Usability</i> | 83 |
| BAB V SIMPULAN DAN SARAN..... | 84 |
| A. Simpulan..... | 84 |
| B. Keterbatasan Produk | 85 |
| C. Pengembangan Produk Lebih Lanjut..... | 85 |
| D. Saran..... | 85 |
| DAFTAR PUSTAKA..... | 86 |
| LAMPIRAN..... | 90 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 1. Siklus Informasi..... | 8 |
| Gambar 2. Siklus Informasi John Burch..... | 8 |
| Gambar 3. Konsep MVC..... | 12 |
| Gambar 4. Klasifikasi Pengujian <i>Black Box</i> | 18 |
| Gambar 5. Model ISO 9126 | 19 |
| Gambar 6. Kerangka Pikir Penelitian | 30 |
| Gambar 7. Model Proses <i>Waterfall</i> | 32 |
| Gambar 8. Contoh Tingkat Persetujuan Responden pada <i>Software</i> | 43 |
| Gambar 9. <i>Use Case Diagram</i> Sistem | 46 |
| Gambar 10. <i>Use Case Diagram</i> Admin..... | 47 |
| Gambar 11. <i>Use Case Diagram</i> Siswa | 48 |
| Gambar 12. <i>Use Case Diagram</i> Tamu..... | 49 |
| Gambar 13. <i>Class Diagram</i> Sistem..... | 50 |
| Gambar 14. <i>Activity Diagram</i> Login | 51 |
| Gambar 15. <i>Activity Diagram</i> Logout | 52 |
| Gambar 16. <i>Activity Diagram</i> Tambah Data..... | 53 |
| Gambar 17. <i>Activity Diagram</i> Edit Data..... | 54 |
| Gambar 18. <i>Activity Diagram</i> Hapus Data..... | 55 |
| Gambar 19. <i>Activity Diagram</i> Tampil Data | 56 |
| Gambar 20. <i>Sequence Diagram</i> Login..... | 57 |
| Gambar 21. <i>Sequence Diagram</i> Logout..... | 57 |
| Gambar 22. <i>Sequence Diagram</i> Tambah Data | 58 |
| Gambar 23. <i>Sequence Diagram</i> Edit Data..... | 59 |
| Gambar 24. <i>Sequence Diagram</i> Hapus Data..... | 60 |

| | |
|---|----|
| Gambar 25. <i>Sequence Diagram</i> Lihat Data | 60 |
| Gambar 26. Perancangan Basis Data..... | 61 |
| Gambar 27. Desain <i>Interface</i> Halaman Utama | 62 |
| Gambar 28. Desain <i>Interface</i> Halaman Login | 63 |
| Gambar 29. Desain <i>Interface</i> Halaman <i>Dashboard</i> Admin..... | 64 |
| Gambar 30. Desain <i>Interface</i> Halaman <i>Dashboard</i> Siswa | 65 |
| Gambar 31. Desain <i>Interface</i> Halaman Tamu..... | 66 |
| Gambar 32. Hasil implementasi Halaman Utama..... | 67 |
| Gambar 33. Hasil implementasi Halaman Login | 68 |
| Gambar 34. Hasil implementasi <i>Dashboard</i> Admin..... | 68 |
| Gambar 35. Hasil implementasi <i>Dashboard</i> Siswa | 69 |
| Gambar 36. Hasil implementasi <i>Dashboard</i> Tamu..... | 69 |
| Gambar 37. Implementasi Tabel <i>user</i> | 70 |
| Gambar 38. Implementasi Tabel <i>user_groups</i> | 70 |
| Gambar 39. Implementasi Tabel <i>authors</i> | 71 |
| Gambar 40. Implementasi Tabel <i>book</i> | 71 |
| Gambar 41. Implementasi Tabel <i>book_user</i> | 72 |
| Gambar 42. Implementasi Tabel absensi..... | 72 |
| Gambar 43. Implementasi Tabel <i>information</i> | 72 |
| Gambar 44. Hasil Pengujian <i>Security</i> | 75 |
| Gambar 45. Hasil Pengujian GTMetrix..... | 76 |
| Gambar 46. Hasil Pengujian GTMetrix..... | 76 |
| Gambar 47. Hasil Pengujian GTMetrix..... | 77 |
| Gambar 48. Hasil Pengujian GTMetrix..... | 78 |
| Gambar 49. Hasil Pengujian <i>Reliability</i> WAPT | 79 |
| Gambar 50. Hasil Koefisien <i>Cronbach's Alpha</i> | 81 |

| | |
|--|-----|
| Gambar 51. Tingkat Persetujuan Responden | 83 |
| Gambar 52. <i>Activity Diagram</i> Cari Data | 104 |
| Gambar 53. <i>Activity Diagram Import Data</i> | 105 |
| Gambar 54. <i>Activity Diagram Export Data</i> | 106 |
| Gambar 55. <i>Activity Diagram Ganti Password</i> | 106 |
| Gambar 56. <i>Activity Diagram Lupa Password</i> | 107 |
| Gambar 57. <i>Sequence Diagram</i> Cari Data..... | 109 |
| Gambar 58. <i>Sequence Diagram Import Data</i> | 110 |
| Gambar 59. <i>Sequence Diagram Export Data</i> | 110 |
| Gambar 60. <i>Sequence Diagram Ganti Password</i> | 111 |
| Gambar 61. <i>Sequence Diagram Lupa Password</i> | 112 |
| Gambar 62. Hasil Pengujian GTMetrix..... | 114 |
| Gambar 63. Hasil Pengujian GTMetrix..... | 115 |
| Gambar 64. Hasil Pengujian GTMetrix..... | 115 |
| Gambar 65. Hasil Pengujian GTMetrix..... | 116 |
| Gambar 66. Hasil Pengujian GTMetrix..... | 117 |
| Gambar 67. Hasil Pengujian GTMetrix..... | 118 |
| Gambar 68. Hasil Pengujian GTMetrix..... | 118 |
| Gambar 69. Hasil implementasi Halaman Utama..... | 121 |
| Gambar 70. Hasil implementasi Halaman Tentang Perpustakaan | 122 |
| Gambar 71. Hasil implementasi Halaman Lokasi..... | 122 |
| Gambar 72. Hasil implementasi Halaman Lupa <i>Password</i> | 123 |
| Gambar 73. Hasil implementasi Halaman Ubah <i>Password</i> | 123 |
| Gambar 74. Hasil implementasi Halaman Daftar Siswa | 124 |
| Gambar 75. Hasil implementasi Halaman Lihat Siswa | 124 |
| Gambar 76. Hasil implementasi Halaman Tambah Siswa..... | 125 |

| | |
|--|-----|
| Gambar 77. Hasil implementasi Halaman Edit Siswa | 125 |
| Gambar 78. Hasil implementasi Halaman Cari Siswa | 126 |
| Gambar 79. Hasil implementasi Halaman Anggota..... | 126 |
| Gambar 80. Hasil implementasi Halaman Peminjaman | 127 |
| Gambar 81. Hasil implementasi Halaman Tambah Buku | 127 |
| Gambar 82. Hasil implementasi Halaman <i>Import</i> Buku | 128 |
| Gambar 83. Hasil implementasi Halaman Absensi | 128 |
| Gambar 84. Dokumentasi Pengujian <i>Usability</i> 1 | 150 |
| Gambar 85. Dokumentasi Pengujian <i>Usability</i> 2 | 150 |
| Gambar 86. Dokumentasi Pengujian <i>Usability</i> 3 | 151 |
| Gambar 87. Dokumentasi Pengujian <i>Usability</i> 4 | 151 |

DAFTAR TABEL

| | |
|---|-----|
| Tabel 1. Sub-karakteristik ISO 9126 dan Penjelasannya | 20 |
| Tabel 2. Tiga Batasan Waktu Menurut Nielsen | 24 |
| Tabel 3. Instrumen <i>functionality</i> | 36 |
| Tabel 4. Kuesioner USE | 39 |
| Tabel 5. Contoh Hasil Jawaban Responden | 42 |
| Tabel 6. <i>Internal Consistency Cronbach's Alpha</i> | 43 |
| Tabel 7. Definisi Aktor Admin..... | 47 |
| Tabel 8. Definisi Aktor Siswa | 48 |
| Tabel 9. Definisi Aktor Tamu..... | 49 |
| Tabel 10. Hasil Uji <i>Functionality</i> | 73 |
| Tabel 11. Hasil Pengujian <i>Efficiency</i> | 78 |
| Tabel 12. Hasil Pengujian <i>Usability</i> | 80 |
| Tabel 13. Hasil Pengujian <i>Reliability</i> | 82 |
| Tabel 14. Perolehan Skor Kuesioner | 83 |
| Tabel 15. Definisi <i>Use Case Diagram</i> | 96 |
| Tabel 16. Skenario <i>Use Case Login</i> | 97 |
| Tabel 17. Skenario <i>Use Case Logout</i> | 97 |
| Tabel 18. Skenario <i>Use Case Ubah Password</i> | 98 |
| Tabel 19. Skenario <i>Use Case Ubah Password</i> | 98 |
| Tabel 20. Skenario <i>Use Case Tambah Data</i> | 99 |
| Tabel 21. Skenario <i>Use Case Edit Data</i> | 99 |
| Tabel 22. Skenario <i>Use Case Hapus Data</i> | 100 |
| Tabel 23. Skenario <i>Use Case Cari Data</i> | 101 |
| Tabel 24. Lihat Data | 101 |
| Tabel 25. Skenario <i>Use Case Import Data</i> | 101 |

| | |
|---|-----|
| Tabel 26. Skenario <i>Use Case Export Data</i> | 102 |
| Tabel 27. Skenario <i>Use Case Lihat Statistik Pengunjung</i> | 102 |
| Tabel 28. Skenario <i>Use Case Pengembalian Buku</i> | 102 |
| Tabel 29. Daftar Responden Pengujian <i>Usability</i> | 141 |
| Tabel 30. Data Pengujian <i>Usability</i> | 143 |

DAFTAR LAMPIRAN

| | |
|--|-----|
| Lampiran 1. Surat – Surat Perijinan | 91 |
| Lampiran 2. Desain Use Case Diagram | 95 |
| Lampiran 3. Desain <i>Activity Diagram</i> | 103 |
| Lampiran 4. Desain <i>Sequence Diagram</i> | 108 |
| Lampiran 5. Hasil Pengujian <i>Efficiency</i> | 113 |
| Lampiran 6. Implementasi Antarmuka | 120 |
| Lampiran 7. Analisis Kebutuhan | 129 |
| Lampiran 8. Angket Pengujian <i>Usability</i> | 133 |
| Lampiran 9. Data Pengujian <i>Usability</i> | 140 |
| Lampiran 10. Angket Pengujian <i>Functionality</i> | 145 |
| Lampiran 11. Dokumentasi | 149 |

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi informasi dan komunikasi seharusnya memberikan dampak dalam pengelolaan perpustakaan. Tujuan perpustakaan adalah untuk menyediakan fasilitas dan sumber informasi dan menjadi pusat pembelajaran (Sutarno NS, 2006:34). Definisi lain menyatakan bahwa perpustakaan berfungsi untuk mengenalkan teknologi informasi (Lasa, 2007:15). Perkembangan teknologi informasi harus terus diikuti pelajar dan pengajar. Untuk itu perlu proses pengenalan dan penerapan teknologi informasi dari perpustakaan.

Perpustakaan sebagai pengelola informasi dan pengetahuan harus bisa menggunakan dan memanfaatkan teknologi informasi dengan optimal untuk memenuhi berbagai kebutuhannya. Beberapa pertimbangan tentang alasan perpustakaan memanfaatkan teknologi informasi dan komunikasi antara lain: (1) tuntutan terhadap kuantitas dan pelayanan perpustakaan, (2) tuntutan terhadap penggunaan koleksi secara bersama, (3) kebutuhan untuk mengefektifkan sumberdaya manusia, (4) tuntutan terhadap efisien waktu, dan (5) keragaman informasi yang dikelola (Saleh, 2006:1).

Dalam upaya meningkatkan kinerja pelayanan, penggunaan sistem informasi merupakan alternatif atau solusi yang tepat. Alasan untuk menggunakan sistem informasi diantaranya adalah: 1) kecepatan pengolahan yang lebih besar 2) ketepatan dan konsistensi yang lebih baik, 3) pencapaian informasi lebih cepat, 4) mereduksi biaya, 5) keamanan yang lebih baik (Andri Kristanto, 2004:41). Oleh karena itu perpustakaan harus dapat menyelenggarakan aktivitas pendidikan di dalam kegiatan operasionalnya. Salah satunya yaitu memanfaatkan teknologi

informasi. Tujuannya adalah agar semua sarana dan fasilitas perpustakaan dapat digunakan dengan tepat dan bermanfaat sehingga anggaran yang dikeluarkan dalam penyediaan fasilitas dan sarana tidak terbuang sia-sia. Saat ini di SMK Muhammadiyah 1 Yogyakarta memiliki jumlah koleksi buku yang banyak. Menurut catatan bulan Januari 2015 terdapat 438 buku yang ada di perpustakaan. Begitu juga siswa-siswi di sekolah yang meminjam untuk kebutuhan pembelajaran di dalam maupun luar kelas membuat regulasi perpustakaan sekolah menjadi sulit di data. Sudah 13 kali di tahun 2015 terjadi kehilangan koleksi buku karena data peminjaman yang kurang dikelola dengan baik. Berdasarkan hasil wawancara dengan salah satu guru yang menjadi petugas perpustakaan di SMK Muhammadiyah 1 Yogyakarta yaitu Nita Dewi, diketahui bahwa proses regulasi perpustakaan memiliki beberapa kendala diantaranya kesulitan dalam mencari buku yang diinginkan, pendataan koleksi buku, pengelolaan peminjaman serta pengembalian buku yang masih dilakukan secara manual.

Ketidakefektifan pelayanan di perpustakaan SMK Muhammadiyah 1 Yogyakarta juga disebabkan hanya terdapat seorang petugas yang dapat melayani siswa yang akan melakukan transaksi sirkulasi. Hal ini dikarenakan perpustakaan belum memiliki petugas tetap, sehingga pelayanan transaksi dilakukan oleh seorang guru yang rutin bertindak sebagai petugas perpustakaan dan terkadang dibantu oleh guru yang lain. Kemudian tidak jarang siswa mencatatkan peminjaman bukunya sendiri. Kondisi ini memungkinkan siswa meminjam lebih dari jumlah maksimal peminjaman yaitu satu buah buku untuk pembelajaran di kelas. Selain itu, pendataan terhadap buku-buku baru juga jarang dilakukan karena kesibukan kegiatan petugas perpustakaan yang memiliki kewajiban sebagai guru di SMK Muhammadiyah 1 Yogyakarta.

Gambaran di atas menjadi sebuah objek penelitian untuk menerapkan sistem informasi perpustakaan pada sekolah menengah kejuruan yang belum memiliki suatu sistem informasi perpustakaan. Teknologi informasi yang dikembangkan menggunakan layanan berbasis *web*, yang diharapkan memberi kemudahan baik untuk karyawan maupun pengunjung perpustakaan. Selain itu sistem akan diterapkan secara *offline* guna mempermudah pihak sekolah dalam proses instalasi dan perawatan lebih lanjut. Dengan sistem informasi ini, diharapkan akan mempermudah pelayanan dan akses informasi serta pengelolaan data perpustakaan, seperti mempermudah pencarian buku/katalog, peminjaman, pengembalian koleksi buku dan pembuatan laporan. Sehingga, akan diperoleh efisiensi pekerjaan staf perpustakaan dalam pengelolaan buku perpustakaan, penyajian informasi yang lebih mudah dan interaktif, dan memberikan layanan yang lebih baik kepada pengguna layanan perpustakaan. Oleh karena itu penulis berpikir untuk mengembangkan suatu sistem informasi perpustakaan sebagai sarana penunjang pendidikan di SMK Muhammadiyah 1 Yogyakarta. Sistem informasi yang akan dikembangkan juga perlu diuji kualitasnya agar layak untuk digunakan. Pengujian tidak hanya untuk mendapatkan program yang benar, namun juga memastikan bahwa program tersebut bebas dari kesalahan-kesalahan untuk segala kondisi (Andri Kristanto, 2004:105). Sistem yang akan diusulkan akan diuji dengan mengacu pada pengujian ISO 9126 karena merupakan standar Internasional dan *Web Quality Evaluation Method* (WebQEM) yang telah diidentifikasi oleh Olsina dkk meliputi aspek *functionality*, *efficiency*, *reliability*, dan *usability*.

B. Identifikasi Masalah

Berdasarkan uraian latar belakang di atas, maka ada beberapa permasalahan yang dapat diidentifikasi adalah sebagai berikut :

1. Belum optimalnya proses pengelolaan data perpustakaan yang masih dilakukan secara manual seperti pengelolaan data buku, pengelolaan data siswa dan proses peminjaman serta pengembalian buku sehingga mengakibatkan kurang akurat, efektif, dan efisiensinya penggunaan waktu, tempat, maupun tenaga.
2. Masih rawannya kerusakan atau kehilangan data dalam pengelolaan data perpustakaan sekolah.
3. Sistem laporan data peminjaman dan inventaris buku yang belum terkelola dengan baik karena belum terkomputerisasi.
4. Belum ada media atau sistem informasi di dalam perpustakaan yang bisa diakses setiap saat oleh siswa dan siswi SMK sebagai fasilitator dalam menyampaikan informasi mengenai perpustakaan sekolah.

C. Batasan Masalah

Berdasarkan beberapa pokok permasalahan yang telah diuraikan pada identifikasi masalah di atas, maka untuk lebih memfokuskan permasalahan yang akan diteliti, maka permasalahannya dibatasi sebagai berikut:

1. Proses pengelolaan transaksi peminjaman dan pengembalian buku yang masih dilakukan secara manual.
2. Proses pengelolaan data buku dan data siswa anggota perpustakaan yang masih dilakukan secara manual.
3. Proses pencarian buku yang masih dilakukan secara manual.

4. Sistem laporan data peminjaman dan pengembalian buku yang dilakukan belum efisien karena masih dilakukan secara manual.
5. Kualitas sistem informasi perpustakaan yang belum sesuai dengan standar kualitas *software*.

D. Rumusan Masalah

Berdasarkan identifikasi masalah dan batasan masalah yang ada dapat dibuat beberapa rumusan masalah, yaitu:

1. Bagaimana membuat Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta ditinjau dari pencatatan absensi, pencarian koleksi buku, pendataan peminjaman buku dan pengembalian buku, serta pembuatan laporan data peminjaman dan buku perpustakaan?
2. Bagaimana mengetahui standar kualitas Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta dengan standar ISO 9126?

E. Tujuan Penelitian

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

1. Membuat Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta ditinjau dari pencatatan absensi, pencarian koleksi buku, pendataan peminjaman buku dan pengembalian buku, serta pembuatan laporan data peminjaman dan buku perpustakaan.
2. Mengetahui standar kualitas Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta berdasarkan standar kualitas *software* ISO 9126.

F. Manfaat Penelitian

Manfaat penelitian ini dapat dikategorikan menjadi manfaat teoritis dan manfaat praktis sebagai berikut:

1. Manfaat Teoritis

Hasil penelitian ini dapat dijadikan referensi pengembangan sistem informasi yang berorientasi pada fasilitas pembelajaran seperti sistem informasi perpustakaan dan mengetahui perbedaan antara sesudah dan sebelum pengembangan sistem dari manual menjadi terkomputerisasi melalui sistem informasi perpustakaan sekolah SMK Muhammadiyah 1 Yogyakarta.

2. Manfaat Praktis

- a. Hasil penelitian ini dapat digunakan untuk mengembangkan perpustakaan sekolah dalam rangka memenuhi kebutuhan penggunaan sistem informasi perpustakaan sekolah sebagai alat bantu untuk mempermudah pelayanan anggota, pencarian data buku, pendataan koleksi buku, sistem pelaporan dan manajemen peminjaman dan pengembalian koleksi buku perpustakaan sekolah SMK Muhammadiyah 1 Yogyakarta.
- b. Mendapatkan hasil uji kelayakan perangkat lunak yaitu penggunaan sistem informasi perpustakaan berbasis *web*.
- c. Sistem informasi perpustakaan yang telah melalui proses pengujian dapat digunakan sebagai media bantu fasilitas pembelajaran, baik pembelajaran di dalam maupun di luar kelas di SMK Muhammadiyah 1 Yogyakarta.

BAB II KAJIAN PUSTAKA

A. Deskripsi Teori

1. Pengertian Sistem Informasi

a. Sistem

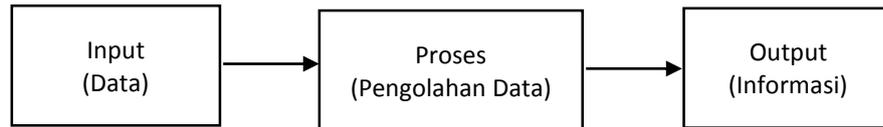
Sistem secara umum adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu sebagai satu kesatuan (Agus Mulyanto, 2009:1). Definisi lain mengartikan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu (Jogiyanto HM, 2008:1). Sedangkan prosedur merupakan suatu urutan-urutan operasi yang melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan dari transaksi-transaksi yang terjadi (Jogiyanto HM, 2008:1). Berdasarkan pendapat para ahli diatas, pengertian sistem yaitu kumpulan dari elemen-elemen yang berinteraksi untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

b. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata (Agus Mulyanto, 2009:12). Definisi lain tentang informasi adalah data yang telah diproses atau diolah ke dalam bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang (Gordon B. Davis 1991: 28). Berdasarkan pendapat para ahli diatas, pengertian informasi adalah data yang

diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya untuk mengambil keputusan masa kini maupun yang akan datang.

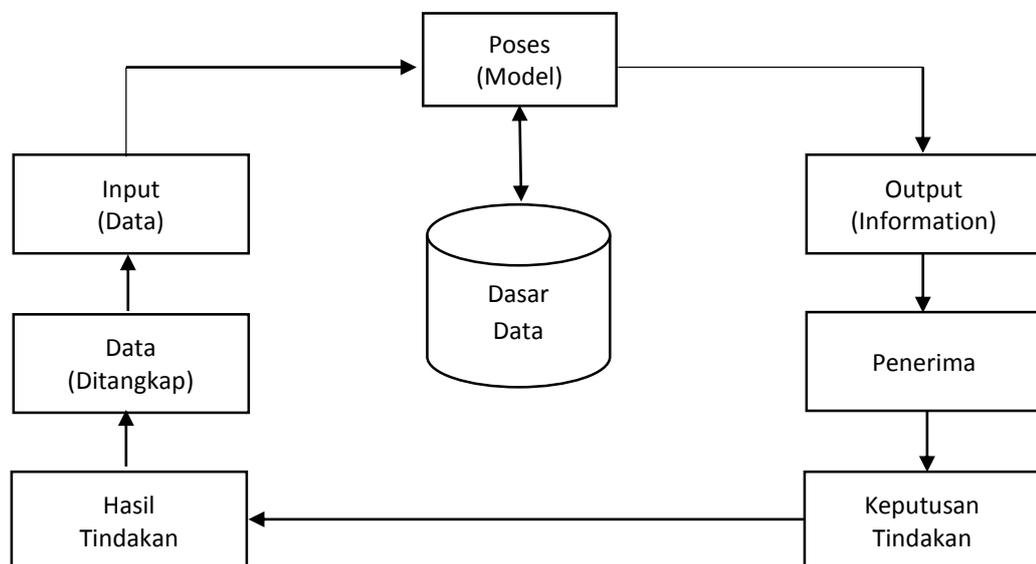
Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. Siklus ini ditunjukkan pada Gambar 1 sebagai berikut:



Gambar 1. Siklus Informasi

(Sumber: Al-Bahra Bin Ladjamudin, 2005:11)

Sebelum data yang ditangkap sebagai *input* diolah untuk menjadi informasi harus melewati beberapa siklus. Siklus ini oleh John Burch disebut dengan siklus informasi (*information cycle*). Siklus ini juga disebut dengan siklus pengolahan data (*data processing cycles*). Siklus informasi ditunjukkan pada Gambar 2 sebagai berikut:



Gambar 2. Siklus Informasi John Burch

(Sumber: Jogiyanto HM, 2008:9)

c. Sistem Informasi

Sistem informasi adalah suatu komponen yang terdiri dari manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai suatu tujuan (Agus Mulyanto, 2009:29). Sedangkan definisi lain menyatakan bahwa sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi (Al Bahra, 2005:13). Berdasarkan pendapat para ahli di atas pengertian sistem informasi adalah suatu komponen yang terdiri dari manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

2. Sistem Informasi Perpustakaan

Sistem informasi perpustakaan merupakan sistem automasi perpustakaan (Harmawan, 2009:1). Sistem informasi perpustakaan merupakan sistem yang keseluruhannya bekerja secara sistematis sehingga dapat memperbaiki administrasi dan operasional perpustakaan serta dapat menghasilkan bentuk-bentuk laporan yang efektif dan berguna bagi manajemen perpustakaan (Lutfian, 2009:1). Definisi lain mengemukakan sistem informasi perpustakaan sebagai perangkat lunak yang didesain khusus untuk mempermudah pendataan koleksi perpustakaan, katalog, data anggota/peminjam, transaksi dan sirkulasi koleksi perpustakaan (Musa, 2010:1). Sistem informasi perpustakaan yaitu suatu sistem di dalam suatu organisasi pelayanan publik yang mempertemukan kebutuhan pengolahan transaksi peminjaman, pengembalian dan perpanjangan buku dan pembuatan laporan harian, bulanan ataupun tahunan guna mendukung operasi,

bersifat manajerial dan kegiatan dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Beiling Siregar, 2007:137).

Berdasarkan pendapat para ahli diatas jadi dapat disimpulkan bahwa sistem informasi perpustakaan (SIPERPUS) merupakan suatu sistem di perpustakaan yang memudahkan kebutuhan pengolahan transaksi peminjaman, pengembalian dan perpanjangan buku maupun laporan-laporan yang diperlukan. Kebutuhan yang diperlukan dapat berupa kemudahan pendataan koleksi perpustakaan, katalog buku, data anggota/peminjam, transaksi dan sirkulasi koleksi perpustakaan. Manfaat dari penerapan sistem informasi perpustakaan diantaranya adalah 1) mengefisiensikan dan mempermudah pekerjaan dalam perpustakaan, 2) memberikan layanan yang lebih baik kepada pengguna perpustakaan, 3) meningkatkan citra perpustakaan dan 4) pengembangan infrastruktur nasional, regional dan global (Ishak, 2008:89).

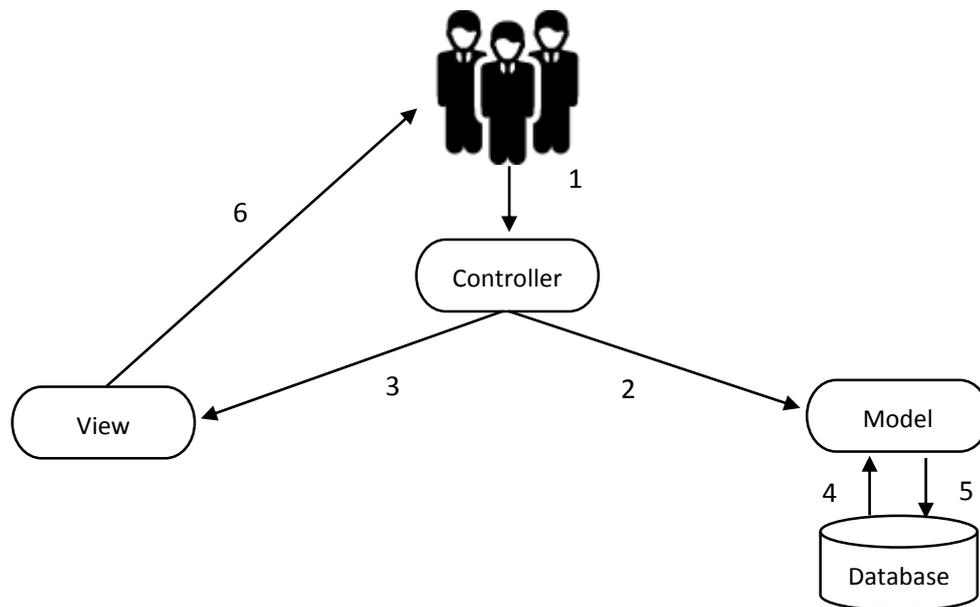
3. Laravel *Framework*

Framework adalah sekumpulan *library* yang diorganisasikan pada sebuah rancangan arsitektur untuk memberikan kecepatan, ketepatan, kemudahan dan konsistensi di dalam pengembangan aplikasi atau pemecahan suatu masalah. “A *software framework is a reusable design for a software system (or subsystem) that may include support programs, code libraries, a scripting language or even other software to help develop and glue together the different components of software project*” (Aparajita Suman, 2014:95). Definisi lain menyatakan bahwa *framework* adalah kerangka kerja yang berisi kumpulan *function, class, method*, dan aturan skrip yang terorganisir sedemikian rupa sehingga memiliki keseragaman penulisan kode dan penempatan folder dalam membangun sebuah aplikasi (Anita Sesar Ria, 2014:8). Dapat disimpulkan bahwa *framework* merupakan desain yang dapat

digunakan kembali oleh perangkat lunak yang berisi kumpulan *function*, *class*, *method*, dan aturan skrip yang terorganisir sehingga memiliki keseragaman penulisan kode untuk membantu pembuatan sebuah perangkat lunak.

Terdapat beberapa keuntungan menggunakan *framework* yaitu 1) aplikasi dibangun dengan kode yang konsisten, 2) mudah digunakan kembali untuk aplikasi lain hanya dengan sedikit konfigurasi, 3) *url* yang fleksibel dan mudah diatur, 4) menggunakan konsep MVC sehingga antara *programmer* dan *designer* bisa bekerja secara terpisah, 5) mempercepat pembuatan *web* dan 6) meningkatkan *security* dari sebuah aplikasi (Anita Sesar Ria, 2014:8).

Laravel adalah *framework* pengembangan *web* berbasis MVC yang ditulis menggunakan PHP. Ini dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan dengan menyediakan *sintaks* yang jelas dan fungsi yang akan menghemat waktu dalam pengembangan *web* (Shawn McCool, 2012:3). Sedangkan MVC merupakan kependekan dari *Model*, *View*, dan *Controller*. MVC merupakan folder yang biasa digunakan di dalam sebuah *framework*. Folder *Model* digunakan untuk menyimpan skrip yang berhubungan dengan *database*, folder *View* digunakan untuk menyimpan skrip yang berhubungan dengan tampilan, dan *Controller* merupakan folder yang digunakan untuk penulisan logika dan algoritma sebuah aplikasi (Anita Sesar Ria, 2014:8). Konsep cara kerja MVC dapat dijelaskan pada Gambar 3 sebagai berikut:



Gambar 3. Konsep MVC

(Sumber: Anita Sesar Ria, 2014:9)

Penjelasan gambar:

- No.1: Bila ada *request* dari *user*, maka aplikasi berbasis *framework* langsung memanggil sebuah file di dalam *Controller*.
- No.2: Di dalam *Controller* ini akan ada pengecekan skrip memanggil *Model*, untuk mengelola data di database atau tidak. Jika ya, *Controller* akan menghubungi *Model*, jika tidak langsung menampilkan *View*.
- No.3: Sebuah *Controller* akan otomatis memanggil file html di dalam sebuah folder *View*, yang kemudian akan ditampilkan kepada *user*.
- No.4 & No.5: Dengan *Model*, seorang *user* bisa mengelola data dari/ke *database*.
- No.6: *View* akan ditampilkan kepada *user*.

Laravel adalah *framework* yang menyediakan sintaks kode lebih singkat, mudah dimengerti, dan *ekspresif* yang bisa digunakan untuk membangun aplikasi *web* yang cepat, stabil, mudah diperbaiki dan dikembangkan. Berikut beberapa alasan menggunakan Laravel antara lain : 1) *Expressif*, ketika melihat suatu *syntax* *Laravel*, *programmer* akan mengerti kegunaan dari *syntax* tersebut meskipun belum pernah mempelajari maupun menggunakannya; 2) *Simple* dengan adanya *Eloquent ORM* yang memudahkan dalam mengambil semua data yang ada pada tabel *database* hanya dengan membuat *class model*; 3) *Accesible*, dokumentasi *Laravel* yang lengkap dan terus di *support* oleh *Core Developer Laravel* walaupun telah rilis versi terbaru (Maksim Surguy, 2014:1).

Laravel juga dikembangkan dengan menambahkan fitur yang tersedia pada PHP versi baru (PHP 5.3+), berbeda dengan *framework* yang lain yang masih fokus pada pengembangan PHP versi lama dan belum mengimplementasikan fitur yang tersedia pada PHP versi terbaru. Berikut beberapa contoh fitur yang ada pada *framework* *Laravel* antara lain 1) pengaturan yang diperlukan untuk pembuatan aplikasi baru di *Laravel* sangat mudah dan bersifat *optional*. Pengaturan hanya dilakukan pada satu folder yaitu folder *app/config*; 2) Setelah menginstal, aplikasi baru *Laravel* dapat langsung dijalankan yaitu hanya dengan mengetikkan *syntax* di *command prompt*, dan 3) aplikasi *Laravel* mengacu pada pola MVC (*Model View Controller*). Jadi tiap file sudah ditempatkan pada folder masing-masing yang memudahkan *programmer* dalam mengelola kode (Maksim Surguy, 2014:5).

Selain itu keunggulan *laravel* dengan *framework* yang lain adalah MIT *License* sehingga *framework* ini bersifat *open source* (gratis). Jadi dapat disimpulkan bahwa *Laravel* adalah *framework* PHP bersifat *open source* (gratis) yang

digunakan untuk membuat aplikasi *web* yang cepat, stabil, mudah diperbaiki dan dikembangkan dengan menyediakan sintaks kode lebih singkat, mudah dimengerti, dan *ekspresif* serta memiliki kelebihan dengan *framework* yang lain yaitu menggunakan fitur PHP versi terbaru.

4. *Unified Modeling Language* (UML)

UML adalah pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak (Ardhian, 2009:65). Sedangkan definisi lain menyatakan (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek (Nugroho, 2010:6). Dapat disimpulkan bahwa UML adalah pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi perangkat lunak yang berparadigma berorientasi objek.

UML dipilih sebagai *tool* untuk membuat model dalam penelitian ini karena aplikasi yang dikembangkan berorientasi objek. UML dibagi menjadi beberapa *view* atau pandangan. *View* atau pandangan adalah bagian sederhana dari konstruksi pemodelan UML yang mempresentasikan aspek dari sebuah sistem. *View* atau pandangan dapat dibagi menjadi tiga area yaitu: 1) klasifikasi struktural (*struktural clasification*), mendeskripsikan hubungan segala hal yang ada dalam sistem, 2) kelakuan dinamik (*dynamic behaviour*), mendeskripsikan kelakuan sistem, atau urutan perubahan yang dialami oleh sistem, 3) pengelolaan model (*model management*), mendeskripsikan keterkaitan organisasi dengan hirarki unit yang ada di dalam sistem. Beberapa jenis-jenis diagram yang ada di UML dapat dibagi menjadi 3 area mayor yaitu 1) area struktural meliputi diagram *use case*, diagram kelas, 2) area dinamik meliputi diagram aktivitas, dan diagram sekuen, 3) dan area pengelolaan model meliputi diagram kelas (Ardhian, 2009 : 67).

5. *Waterfall Process Model*

a. *Software Development Life Cycle*

Software Development Life Cycle (SLDC) atau sering disebut dengan model proses perangkat lunak merupakan proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem-sistem perangkat lunak (Rosa-Sallahudin, 2011:24). Sedangkan definisi lain menyatakan model proses perangkat lunak (atau disebut juga paradigma rekayasa perangkat lunak) adalah suatu strategi pengembangan yang memadukan lapisan proses, metode, dan alat serta tahap-tahap generik. Model proses untuk rekayasa perangkat lunak dipilih berdasarkan sifat proyek dan aplikasi, metode dan alat yang digunakan, serta pengendalian dan hasil yang diinginkan (Eddy Prasetyo Nugroho, 2009: 17). Dapat disimpulkan bahwa model proses perangkat lunak adalah suatu strategi pengembangan yang memadukan lapisan proses, metode, dan alat serta tahap-tahap dalam mengembangkan atau mengubah suatu sistem perangkat lunak.

b. *Waterfall Model*

Pengembangan perangkat lunak sistem informasi perpustakaan menggunakan model *waterfall*. Terdapat beberapa kelebihan dari model *waterfall* yaitu: 1) struktur tahap pengembangan sistem jelas, 2) dokumentasi dihasilkan di setiap tahap pengembangan, 3) sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap) (Rosa-Salahudin, 2011:29). Model *waterfall* sangat cocok digunakan oleh kebutuhan yang sudah sangat dipahami dan kemungkinan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil. Lebih lanjut Rosa dan Salahuddin menjelaskan kelebihan dari *waterfall* merupakan model proses yang paling

sederhana dan cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah.

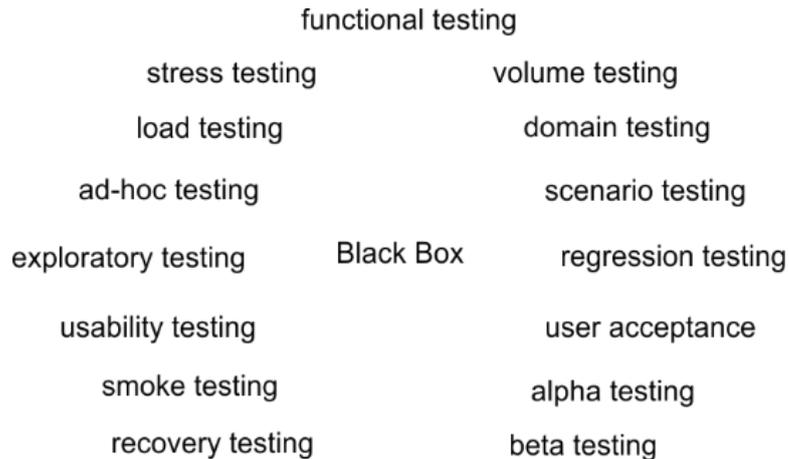
Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*) (Rosa-Salahudin, 2011:26). Uraian masing-masing tahapan model pengembangan *waterfall* adalah sebagai berikut: 1) Analisis kebutuhan, proses pengumpulan kebutuhan dilakukan secara intensif untuk memspezifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user, 2) desain, adalah proses multi langkah yang fokus pada desain pembuatan program termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean, 3) pembuatan kode program, desain harus ditranslasikan ke dalam program, 4) pengujian, proses pengujian fokus pada perangkat lunak secara segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. (Rosa-Salahudin, 2011:27).

6. Pengujian *Black Box*

Pengujian pada dasarnya adalah menemukan serta menghilangkan '*bug*' (kesalahan-kesalahan) yang ada di sistem/perangkat lunak (Adi Nugroho, 2005:431). Sedangkan definisi lain menyatakan pengujian adalah sebuah proses untuk menemukan segala kesalahan dan segala kemungkinan yang akan menimbulkan kesalahan sesuai dengan spesifikasi perangkat lunak yang telah ditentukan sebelum aplikasi tersebut diserahkan kepada pelanggan (Janner

Simarmata, 2010:323). Jadi dapat disimpulkan bahwa pengujian adalah menemukan serta menghilangkan 'bug' (kesalahan-kesalahan) dan segala kemungkinan yang akan menimbulkan kesalahan sesuai dengan spesifikasi perangkat lunak yang telah ditentukan sebelum aplikasi tersebut diserahkan kepada pelanggan.

Beberapa pengujian boleh dilakukan dalam proses pengembangan, proses SQA, atau V&V (Janner Simarmata, 2010:285). Terdapat 2 metode pengujian yang memberikan mekanisme yang membantu memastikan kelengkapan pengujian dan memberikan kemungkinan untuk mengungkap kesalahan pada perangkat lunak. Metode pengujian ini adalah pengujian *black box* dan *white box*. Pengujian *black box* adalah pengujian yang dilakukan untuk memperlihatkan bahwa masing-masing fungsi beroperasi sepenuhnya, pada waktu yang sama mencari kesalahan pada setiap fungsi. Sedangkan pengujian *white box* adalah pengujian untuk mengetahui kerja internal suatu produk (Pressman, 2005:532). Pengujian pada *white box* adalah pengujian mendalam yang menimbulkan masalah logistik tertentu. Bahkan untuk program yang kecil jumlah jalur logika dapat sangat luas (Al Bahra, 2005:359). Sebelum menjadi produk akhir perangkat lunak, pengujian *alpha* dan *beta* sering dilakukan. Pengujian *alpha* adalah pengujian operasional yang aktual atau disimulasikan dengan pelanggan/pengguna potensial atau tim penguji independen pada pengembang perangkat lunak. Sedangkan pengujian *beta* dilakukan setelah pengujian *alpha*. Perangkat lunak diberikan ke dalam kelompok masyarakat agar pengujian lebih lanjut dapat memastikan bahwa produk memiliki beberapa kesalahan atau *bug*. (Janner Simarmata, 2010:313). Klasifikasi dari jenis-jenis pengujian *black box* dapat dilihat pada Gambar 4 sebagai berikut:



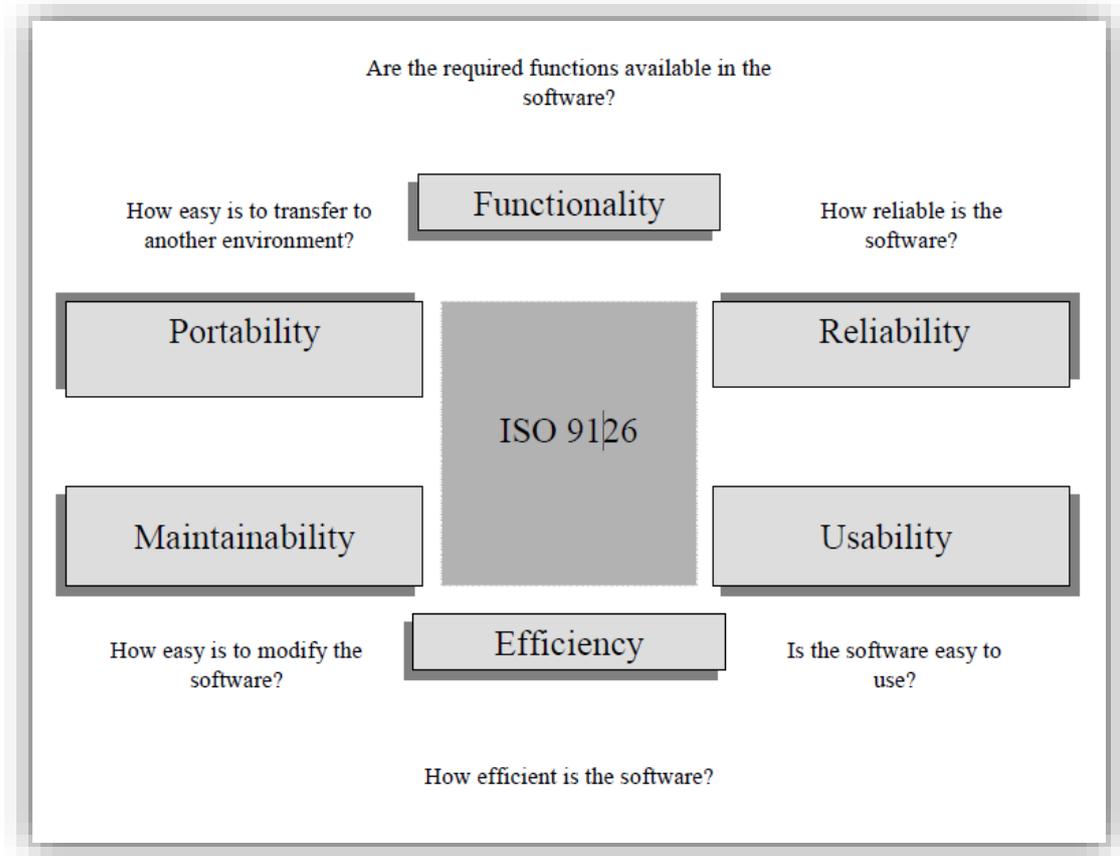
Gambar 4. Klasifikasi Pengujian *Black Box*

(Sumber: Janner Simarmata, 2010:316)

7. Kualitas *Software* ISO 9126

Definisi kualitas perangkat lunak atau *software quality* sebagai sebuah karakteristik atau atribut dari sesuatu. Sebagai atribut dari sesuatu, kualitas mengacu pada karakteristik yang terukur, sesuatu yang dapat dibandingkan dengan standar lain yang sudah diketahui (Adi Nugroho, 2005:432). ISO 9126 merupakan standar internasional yang memiliki tujuan untuk menyediakan sebuah kerangka kerja untuk evaluasi dari kualitas *software* (Chua & Dyson, 2004:185). Terdapat beberapa model kualitas secara terstruktur dan kuantitatif, diantaranya adalah model McCall, model kualitas Boehm, model kualitas Dromey, FURPS, ISO 9000, dan ISO 9126. Kelebihan ISO 9126 antar lain: 1) standar internasional yang didasarkan pada model McCall dan Boehm, 2) model kualitas perangkat lunak yang terstruktur, dan 3) model kualitas perangkat lunak yang memiliki parameter untuk dapat mengidentifikasi karakteristik kualitas perangkat lunak baik *internal* maupun *eksternal* (Berander dkk, 2005: 6). Sedangkan pendapat lain menyatakan standar pengujian ISO 9126 memiliki kelebihan karena mudah beradaptasi dan dapat digunakan di berbagai sistem (Chua & Dyson, 2004:185). ISO 9126

menetapkan enam karakteristik pengujian kualitas *software*. Karakteristik dan sub-karakteristik tersebut dapat dilihat pada Gambar 5 dan Tabel 1 sebagai berikut:



Gambar 5. Model ISO 9126

(Sumber: Chua & Dyson, 2004:186)

Tabel 1. Sub-karakteristik ISO 9126 dan Penjelasannya (Chua & Dyson, 2004:186)

| Characteristic | Sub Characteristics | Explanation |
|-----------------|----------------------|--|
| Functionality | Suitability | 'Can software perform the tasks required?' |
| | Accurateness | 'Is the result as expected?' |
| | Interoperability | 'Can the system interact with another system?' |
| | Compliance | 'Is the system compliant with standards?' |
| | Security | 'Does the system prevent unauthorized access?' |
| Reliability | Maturity | 'Have most of the faults in the software been eliminated over time?' |
| | Fault tolerance | 'Is the software capable of handling errors?' |
| | Recoverability | 'Can the software resume working & restore lost data after failure?' |
| Usability | Understandability | 'Does the user comprehend how to use the system easily?' |
| | Learnability | 'Can the user learn to use the system easily?' |
| | Operability | 'Can the user use the system without much effort?' |
| | Attractiveness | 'Does the interface look good?' |
| Efficiency | Time Behaviour | 'How quickly does the system respond?' |
| | Resource utilization | 'Does the system utilize resources efficiently?' |
| Maintainability | Analyzability | 'Can faults be easily diagnosed?' |
| | Changeability | 'Can the software be easily modified?' |
| | Stability | 'Can the software continue functioning if changes are made?' |
| | Testability | 'Can the software be tested easily?' |
| Portability | Adaptability | 'Can the software be moved to other environments?' |
| | Installability | 'Can the software be installed easily?' |
| | Conformance | 'Does the software comply with portability standards?' |
| | Replaceability | 'Can the software easily replace other software?' |

Terdapat empat aspek dalam ISO 9126 yang perlu diuji terkait dengan perangkat lunak, yaitu *functionality*, *efficiency*, *reliability* dan *usability* (Padayachee dkk, 2010:4). Karena itu penelitian ini mengacu pada pengujian kualitas WebQEM. *Website Quality Evaluation Method* (WebQEM) merupakan suatu metode untuk mengevaluasi kualitas dari sebuah aplikasi *web*. Luis Olsina et al (1998) mengevaluasi karakteristik yang dibutuhkan yaitu *functionality*, *efficiency*, *reliability* dan *usability*. Alasan mengidentifikasi keempat karakteristik tersebut karena dalam bidang akademik orientasi penting adalah pengguna (siswa, guru, dsb). Semua model untuk mengevaluasi *website*

bersumber dari ISO 9126. Berikut ini merupakan pengertian dari masing-masing faktor kualitas:

a. *Functionality*

Functionality adalah kemampuan dari produk perangkat lunak dalam menyediakan fungsi-fungsi yang sesuai dengan kebutuhan sistem yang telah dianalisa sebelumnya (ISO/IEC 9126, 2010). Pada aspek *functionality* berdasarkan ISO/IEC (2002:8), *functionality* suatu *software* dapat dihitung dengan menggunakan rumus berikut:

$$X = 1 - \frac{A}{B}$$

Keterangan:

X = *Functionality*

A = Jumlah fungsi yang gagal uji

B = Jumlah seluruh fungsi

$0 \leq X \leq 1$. *Functionality* dikatakan baik jika mendekati 1.

Berdasarkan rumus diatas nilai *functionality* dikatakan memenuhi standar jika nilainya 0.5 dan semakin mendekati 1 maka tingkat *functionality* dari *software* semakin baik. *Functionality* diklasifikasikan ke dalam pengujian fungsional *black box*. Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan (Janner Simarmata, 2010:316). Pengujian tidak berfokus pada mekanisme *internal* sistem tetapi berfokus pada *output* yang dihasilkan dalam merespon *input* yang dipilih (Nidhra, 2012:29).

Aspek *functionality* juga memperhatikan *security* sebuah *software*. Sedangkan *security* adalah kemampuan *software* untuk menjaga dari akses-akses yang tidak disetujui/akses tidak resmi (Chua & Dyson, 2004:186). Contoh kerentanan berbahaya dalam aplikasi *web* adalah *SQL Injection* dan *Cross Site Scripting*

(XSS) (Vieira, Antunes, & Madeira, 2009:1). *SQL Injection* adalah kerentanan yang terjadi ketika suatu *software* memberikan kemampuan pada penyerang untuk mempengaruhi *Structured Query Language (SQL)* (Clarke, 2009:1). Dengan memiliki kemampuan tersebut, penyerang dapat memasukkan kode *SQL* ke dalam parameter *input* pengguna yang kemudian diteruskan ke *back-end SQL Server* untuk dijalankan (Clarke, 2009:1). Kerentanan tersebut bisa berbahaya karena penyerang dapat memasukkan kode *SQL* yang tidak sesuai, misalnya kode *SQL* tersebut menampilkan *password* dari suatu *software*. Kemudian *Cross Site Scripting (XSS)* adalah serangan yang dilakukan dengan memasukan kode berbahaya yang disisipkan dalam masukan *user* dan kemudian dieksekusi oleh *user* dalam *browser* lain karena adanya kelemahan dalam hal validasi. Tujuan dari XSS ini adalah untuk menjangka *cookies* yang berisi informasi dari seorang *user* sehingga memperoleh data rahasia dari *user* tersebut (Stalinbabu & Chellammal, 2014:332). Terdapat berbagai macam alat untuk pengujian keamanan dari *web* yang berupa *software*, salah satunya adalah *Acunetix Web Vulnerability Scanner (WVS)*. WVS adalah aplikasi *web* untuk pengujian keamanan yang secara otomatis akan menganalisis aplikasi *web* dengan pengujian keamanan seperti *SQL Injection* dan *Cross Site Scripting* (Chander & Kush, 2012:53).

Jadi dalam aspek *functionality* terdapat dua macam uji yang dilakukan yaitu menguji fungsi yang ada dalam *software* dalam bentuk tertulis dengan target *developer* dan menguji keamanan *website* dengan *software Acunetix Web Vulnerability Scanner (WVS)*.

b. *Efficiency*

Aspek *efficiency* adalah kemampuan produk perangkat lunak untuk menyediakan performa sesuai dengan kebutuhan pengguna (ISO/IEC 9126,

2010). Sedangkan definisi lain menyatakan *efficiency* mencakup berapa waktu yang dibutuhkan sistem untuk melakukan respon dan bagaimana *resources* dimanfaatkan secara efisien artinya dalam software berbasis *web* dapat diketahui berapa waktu untuk *load* halaman *web* dan bagaimana *source code* diprogram secara efisien (Padayachee, Kotze, & van Der Merwe, 2010:4). Dari pemaparan diatas dapat disimpulkan bahwa *efficiency* adalah waktu yang diperlukan perangkat lunak, baik berupa respon, waktu pemrosesan, dan sumber daya yang dimanfaatkan secara efisien untuk menyediakan performa sesuai dengan kebutuhan pengguna. Pada pengujian *black box*, *efficiency* diklasifikasikan dalam *load testing* yang bertujuan untuk mengetahui apakah situs mengalami kegagalan atau kinerjanya menurun (Janner Simarmata, 2010:316).

Terdapat sub-karakteristik yaitu *time behavior* dan *resource behavior*. Penelitian yang dilakukan dalam pengujian aspek *efficiency* untuk *website* aspek yang paling penting adalah sub-karakteristik *time behavior* (Padayachee, Kotze, & van Der Merwe, 2010:8). Pengujian aspek *efficiency* diambil sub-karakteristik *time behavior* karena untuk aspek *resource behavior* hanya dapat dilakukan oleh ahli (Chua & Dyson, 2004:187).

Sub-karakteristik *time behavior* adalah kecepatan respon dari sistem yang dikembangkan (Chua & Dyson, 2004:186). Terdapat tiga batasan waktu yang penting dalam toleransi waktu respon (Nielsen, 2010). Batasan ini dapat dilihat pada Tabel 2 sebagai berikut:

Tabel 2. Tiga Batasan Waktu Menurut Nielsen (2010)

| Waktu | Respon |
|-----------|--|
| 0.1 detik | Pengguna menerima respon dari perintah yang dijalankan. |
| 1.0 detik | Batasan dari pengguna berfikir untuk menunggu <i>feedback</i> dari sistem. |
| 10 detik | Batasan akhir perhatian pengguna untuk menunggu sistem. |

Batasan pertama yaitu 0,1 detik. Pada batasan ini pengguna merasakan respon dari web yang cepat. Batasan kedua yaitu 1 detik. Pada batasan ini pengguna bisa merasakan adanya jeda namun proses ini membuat pengguna dapat merasakan komputer menjalankan suatu proses yang kemudian mengeluarkan hasil dan pikiran pengguna belum merasa terganggu. Batasan ketiga yaitu 10 detik. Pada batasan ini pengguna mulai memperhatikan jeda yang ada. Pengguna mulai berharap agar respon dapat berjalan lebih cepat. Jika lebih dari batasan tersebut atau lebih dari 10 detik maka pengguna akan meninggalkan *web*. Toleransi waktu terhadap sebuah halaman *web* yang diakses adalah 15 detik. Sehingga dapat disimpulkan bahwa batas respon dari komputer untuk memproses sistem adalah 10 detik dan batas waktu pemrosesan dari sebuah halaman *web* adalah 15 detik. Cara untuk mengetahui berapa waktu untuk membuka halaman *web* dapat menggunakan GTMetrix. Selain itu dapat digunakan untuk mengukur waktu membuka halaman *web*, GTMetrix juga dapat memberikan rekomendasi agar *source code* diprogram agar lebih efisien.

Normalnya semakin cepat waktu *website* dapat diakses oleh pengguna maka akan semakin baik (Singh, 2014).

c. *Reliability*

Reliability adalah kemampuan dari sistem untuk dikembangkan ke dalam level/performa yang tertekan. Sistem tidak mudah *down* atau *hang* karena adanya perlakuan yang keras/besar (ISO/IEC 9126, 2010). Sedangkan definisi lain menyatakan aspek *reliability* ini mencakup lama waktu perangkat lunak dapat digunakan dengan normal. Aspek *reliability* ini meliputi *maturity*, *fault tolerance*, dan *recoverability* (Pressman, 2010). Dari beberapa sub-karakteristik *reliability* tersebut, sub-karakteristik *maturity* yang paling berpengaruh terhadap *software* berbasis *web* (Olsina, Sassano, & Mich, 2008:54). Aspek *maturity* merupakan tingkat kematangan perangkat lunak. Maksud dari kematangan adalah tingkat frekuensi kegagalan perangkat lunak jika digunakan dalam periode waktu tertentu.

Pengujian aspek *maturity* dengan melakukan pengujian tegangan (*stress testing*) (ISO/IEC, 2002:21). Dalam klasifikasi pengujian *black box*, pengujian tegangan adalah untuk menciptakan sebuah lingkungan yang lebih menuntut aplikasi, tidak seperti saat aplikasi dijalankan (Janner Simarmata, 2010:316). Kemudian untuk melakukan *stress testing* dapat digunakan *software Web Application Load, Stress and Performance Testing (WAPT)* (Pradhan, 2013). WAPT adalah alat pengujian *reliability* dengan metode *stress testing* yang mudah digunakan dan hemat biaya untuk menguji situs *web*, termasuk aplikasi bisnis, situs *mobile*, *web portal*, dll (*Official Website WAPT*, 2014). *Stress testing* bekerja dengan menghitung *test case* yang berhasil dilakukan oleh *software* dan membandingkannya dengan total *test case* yang dilakukan. Setelah perhitungan dilakukan kemudian dilakukan interpretasi berdasarkan standar telcordia

reliabilitas perangkat lunak. Standar *reliability* yang dapat diterima berdasarkan standar telcordia adalah 95% atau lebih pada *software* agar dinyatakan lolos (Asthana & Olivieri, 2009:2). Sehingga nantinya *software* dikatakan lolos uji pada aspek *reliability* jika minimal 95% *software* dapat berjalan dengan baik ketika diuji *stress testing* menggunakan WAPT.

d. *Usability*

Aspek *usability* adalah kemampuan dari produk perangkat lunak untuk memberikan kecepatan, memberikan kemudahan, kepuasan penggunaan dan menyediakan bantuan mengenai *error* yang terjadi sesuai dengan kondisi dan kebutuhan (ISO/IEC 9126, 2010). *Usability* adalah atribut kualitas yang digunakan untuk menilai seberapa mudah *user interface* suatu produk untuk digunakan. *Usability* menekankan pada tampilan antar muka (*user interface*) sistem yang konsep dasarnya terletak pada kemudahan dan juga sesuai dengan keinginan pengguna (Nielsen, 2012). Dalam klasifikasi pengujian *black box*, *usability testing* (pengujian usability) adalah proses yang bekerja dengan pengguna akhir secara langsung maupun tidak langsung untuk menilai bagaimana pengguna merasakan paket perangkat lunak dan bagaimana mereka berinteraksi dengannya (Janner Simarmata, 2010:317).

Pengujian sub-karakteristik *understandability*, *learnability*, dan *operability* memiliki metode pengujian yaitu dengan *user test*. Pengujian ini dapat dilakukan dengan menggunakan likert survey (Walace & Cheng Yu, 2009:139). Salah satu kuesioner yang menggunakan skala likert adalah kuesioner dari Arnold M. Lund. Arnold M. Lund (2001) membuat kuesioner yang bernama *USE Questionnaire*. *USE* merupakan kepanjangan dari *Usefulness*, *Satisfaction*, and *Ease of use*.

Usability didefinisikan oleh empat kualitas komponen: *usefulness*, *ease of use*, *easy of learning*, dan *satisfaction*.

B. Penelitian yang Relevan

Beberapa penelitian relevan dengan penelitian ini antara lain:

- a. Sistem Informasi Perpustakaan Sekolah Berbasis Web dengan *Framework* CodeIgniter dan PostgreSQL di SMA Negeri 1 Ngaglik oleh Punky Indra Permana, Universitas Negeri Yogyakarta. Tujuan dari penelitian ini adalah membuat sistem informasi perpustakaan sekolah dan mengetahui tingkat kelayakan dari sisi *functionality*, *security*, *usability*, *maintainability*, *portability* dan *efficiency*. Hasil yang didapat dalam penelitian ini dapat disimpulkan bahwa dengan menggunakan *framework* Codeigniter, menghasilkan perangkat lunak yang berkualitas baik dari sisi *functionality*, *security*, *usability*, *maintainability*, *portability* dan *efficiency*. (Punky Indra, 2013)
- b. Pengembangan Sistem Informasi Perpustakaan pada Sekolah Menengah Pertama (SMP) Negeri 4 Karangtengah oleh Sulastri, Universitas Surakarta. Tujuan dari penelitian ini adalah membangun suatu website informasi perpustakaan pada SMP Negeri 4 Karangtengah secara *online* untuk mempermudah staf perpustakaan sekolah dalam mengolah data perpustakaan. Hasil yang didapat dalam penelitian ini dapat disimpulkan bahwa sistem informasi perpustakaan pada Sekolah Menengah Pertama (SMP) Negeri 4 Karangtengah memudahkan petugas dalam pembuatan laporan-laporan harian. (Sulastri, 2013)
- c. Studi Deskriptif Analisa *Usability* Sistem Informasi Layanan Peminjaman Mandiri Perpustakaan Universitas Negeri Malang dari Sudut Pengguna oleh Nur Sekar Asih, Universitas Airlangga. Tujuan dari penelitian ini adalah untuk

mengetahui pengguna khususnya mahasiswa dapat merasakan kegunaan dari sistem layanan ini tanpa merasakan kesulitan dalam penggunaannya dengan pendekatan 5 indikator *usability*. Hasil yang didapat dalam penelitian ini dapat disimpulkan bahwa layanan peminjaman ini masih dijumpai beberapa kendala dan belum berjalan secara optimal. Masing masing dari indikator *usability* meliputi: *learnability, efficiency, memorability, few error, pleasant to use* dapat dikatakan sudah berjalan dengan baik ,namun masih ada yang mengalami masalah dalam penerapannya, agar sesuai dengan indikator *usability* maka perlu diadakannya analisa lebih lanjut. (Nur Sekar Asih, 2013)

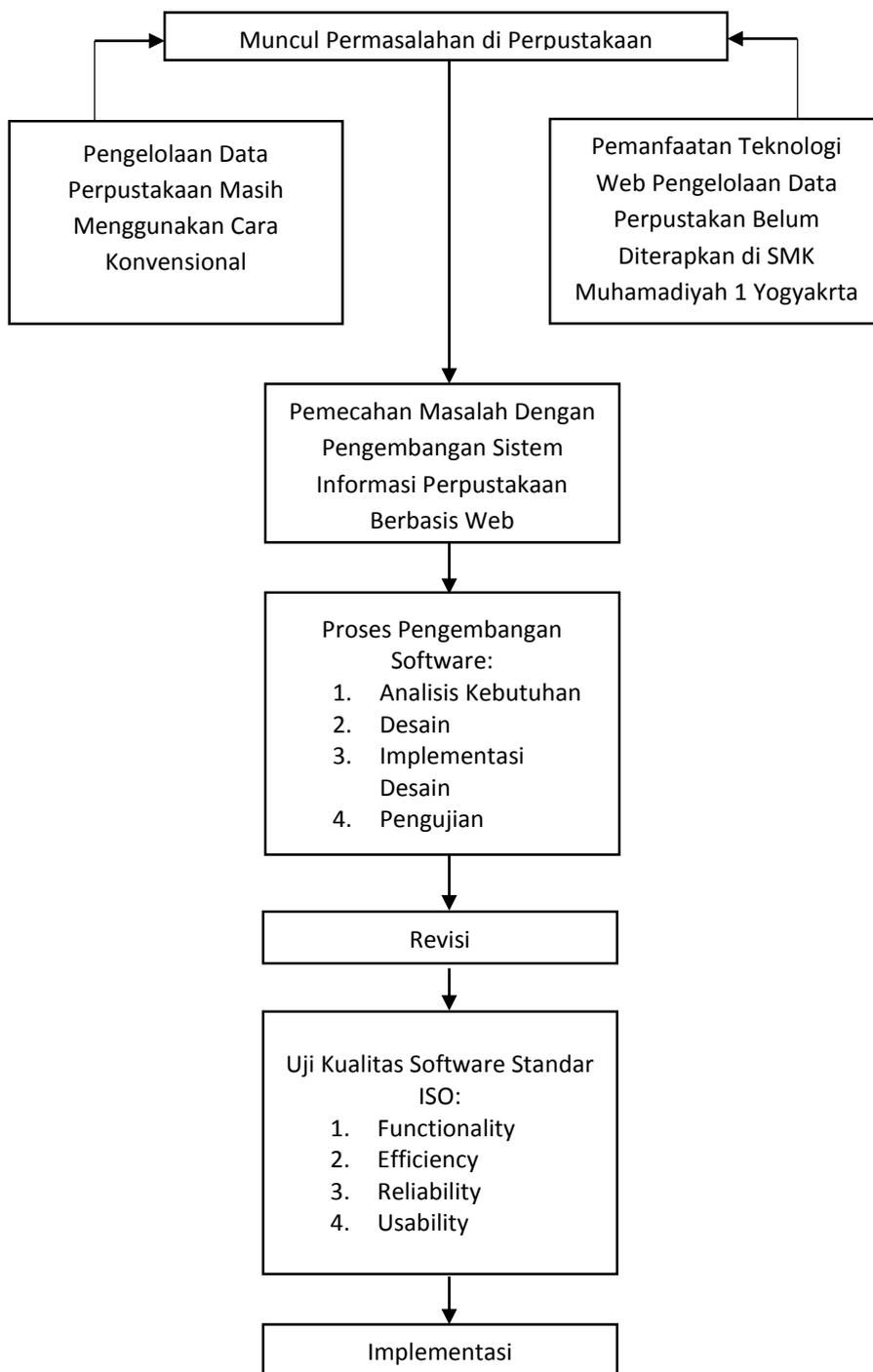
Persamaan penelitian yang relevan tersebut terhadap penelitian ini adalah tentang metode pengujian perangkat lunak untuk pembuatan sistem informasi. Penelitian yang pertama berfokus pada pembuatan serta pengujian sistem informasi untuk sekolah menengah atas. Penelitian relevan kedua berfokus pada pembuatan dan pengujian sistem informasi untuk sekolah menengah pertama. Kemudian penelitian ketiga berfokus pada pembuatan dan pengujian sistem informasi untuk layanan peminjaman Universitas Negeri Malang. Sedangkan perbedaannya adalah *framework* PHP dan standar kualitas yang digunakan. Penelitian ini menggunakan identifikasi aspek dalam *Website Quality Evaluation Method* (WebQEM). WebQEM merupakan *framework* baru untuk mengevaluasi aplikasi *web* yang bersumber dari ISO 9126. *Framework* tersebut terdiri dari faktor yang menjadi *high level quality* yaitu faktor *functionality, efficiency, reliability* dan *usability*.

C. Kerangka Berpikir

Sistem informasi perpustakaan berbasis *web* ini bertujuan untuk mempermudah pengelolaan regulasi di perpustakaan SMK Muhammadiyah 1

Yogyakarta. Pembuatan sistem informasi ini menggunakan *framework* Laravel agar mudah untuk dikembangkan di waktu mendatang karena sudah menerapkan teknologi PHP versi terbaru. Pembuatan sistem informasi perpustakaan ini melalui beberapa tahapan berdasarkan model proses perangkat lunak yaitu *waterfall* yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, dan tahap pengujian.

Analisis kebutuhan antara lain menentukan fitur perangkat lunak sesuai kebutuhan pengguna dan perangkat lunak yang akan digunakan dalam pembuatan sistem informasi. Tahapan desain sistem dibagi menjadi tiga, yaitu pemodelan sistem menggunakan UML, desain basis data, dan desain *user interface*. Implementasi adalah tahap menterjemahkan desain menjadi bentuk akhir sistem yang berupa aplikasi *web*. Tahapan terakhir adalah pengujian sistem dengan menggunakan standar ISO 9126 yang telah diidentifikasi oleh Olsina dalam *Website Quality Evaluation Method* (WebQEM) yaitu faktor *functionality*, *efficiency*, *reliability* dan *usability*. Kerangka pikir dalam penelitian ini ditunjukkan dalam Gambar 6 sebagai berikut:



Gambar 6. Kerangka Pikir Penelitian

D. Pertanyaan Penelitian

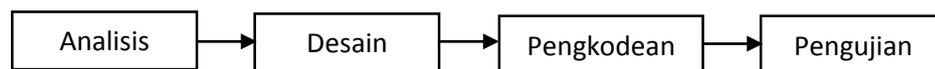
Berdasarkan latar belakang dan tujuan maka pertanyaan penelitian dalam permasalahan ini adalah sebagai berikut.

1. Apakah pembuatan Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta telah memenuhi ditinjau dari pencatatan absensi, pencarian koleksi buku, pendataan peminjaman buku dan pengembalian buku, serta pembuatan laporan data peminjaman dan buku perpustakaan?
2. Apakah Sistem Informasi Perpustakaan SMK Muhammadiyah 1 Yogyakarta memenuhi standar kualitas ISO 9126?

BAB III METODE PENELITIAN

A. Model Pengembangan

Dilihat dari latar belakang dan tujuan, maka penelitian ini menggunakan pendekatan penelitian dan pengembangan atau *Research and Development* (R&D). Dalam aplikasinya pengembangan suatu produk memerlukan suatu model proses pengembangan sedangkan tahapan – tahapan dalam pengembangan perangkat lunak sering disebut dengan SDLC atau *Software Development Life Cycle*. Model yang digunakan dalam pengembangan sistem informasi perpustakaan ini adalah model *waterfall*. Model ini digambarkan seperti pada Gambar 7 sebagai berikut:



Gambar 7. Model Proses *Waterfall*

(Sumber: Pressman, 2010:39)

Sesuai dengan model *waterfall*, pengembangan sistem informasi perpustakaan di SMK Muhammadiyah 1 Yogyakarta dimulai dengan analisis kebutuhan. Analisis kebutuhan mencakup analisis *hardware*, *software*, dan komponen lain yang berkaitan dengan kegiatan perpustakaan. Tahap berikutnya adalah tahap desain, analisis kebutuhan yang telah selesai kemudian dibuat dalam desain *software* menggunakan bahasa pemodelan UML (*Use Case Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram*). Tahap implementasi dilakukan dengan menulis kode program sesuai dengan desain yang telah dirancang sebelumnya. Kemudian pada tahap selanjutnya, *software* diuji

menggunakan metode *black box* dengan standar kualitas perangkat lunak. Tahap pengujian dilakukan melalui dua tahapan yaitu *alpha testing* dan *beta testing*.

B. Prosedur Pengembangan

Model penelitian ini diuraikan sebagai berikut:

1. Analisis Kebutuhan Sistem

Kebutuhan perangkat lunak dalam penelitian ini untuk mengumpulkan kebutuhan yang sesuai dengan kebutuhan *user* atau pengguna. Tujuannya mengubah sistem yang sudah ada yaitu sistem manual menjadi terkomputerisasi. Kebutuhan pengolahan data yang masih dilakukan secara konvensional dan masih membutuhkan waktu yang lama perlu dibuat menjadi sebuah sistem yang dapat menangani data dengan cepat dan dapat diakses setiap saat. Kemudian analisis kebutuhan perangkat lunak juga untuk mengidentifikasi kebutuhan yang ada pada sistem informasi perpustakaan meliputi kebutuhan *software* dan *hardware*. Cara yang digunakan untuk mengumpulkan data yaitu dengan observasi dan wawancara langsung dengan pustakawan di perpustakaan SMK Muhammadiyah 1 Yogyakarta. Observasi dilakukan dengan pengamatan langsung proses regulasi di perpustakaan. Hasil yang didapat berupa spesifikasi yang dibutuhkan dalam pengembangan *software*.

2. Desain Sistem

Desain sistem merupakan gambaran dari analisis kebutuhan. Desain sistem meliputi: *Unified Modelling Language* (UML), basis data (*database*), dan tampilan (*user interface*). UML merupakan pemodelan mengenai sebuah sistem dengan menggunakan diagram dan teks-teks sebagai penjelas diagram. Proses desain sistem menggunakan UML ini dibagi menjadi 3 bagian, yaitu dari sisi admin, siswa dan tamu. Sebagai admin akan memiliki hak akses sepenuhnya mengenai fungsi

dan fitur sistem yang akan dibuat berdasarkan kebutuhan-kebutuhan sebagai admin dalam mengelola sistem. Siswa hanya menjadi pengguna yang memiliki beberapa fungsi tertentu dan memiliki hak akses yang terbatas. Sedangkan tamu hanya menjadi pengguna biasa yang hanya dapat mengakses sistem informasi dan memiliki hak akses yang sangat terbatas.

3. Implementasi/Code Sistem

Tahap selanjutnya adalah implementasi kode dari desain yang telah dibuat secara nyata kedalam bahasa pemrograman. Pada sistem ini implementasi kode dibuat ke dalam bahasa pemrograman PHP dan ke *database* MySQL. Implementasi kode untuk membentuk fungsi-fungsi yang dibutuhkan oleh program dibuat dengan bantuan *framework* Laravel dan *framework* Bootstrap agar sesuai dengan keinginan pengguna dan implementasi dapat berjalan cepat dan mudah. Tahap ini diharapkan agar semua fungsi-fungsi berjalan dengan semestinya sesuai dengan desain yang telah dibuat.

4. Pengujian Sistem

Sistem yang sudah selesai dibuat dan memenuhi fungsi-fungsi yang dibutuhkan akan diuji menggunakan metode *black box* dan mengacu pada model ISO 9126 dengan aspek *functionality*, *efficiency*, *reliability*, dan *usability*. Tahap pengujian dilakukan melalui dua tahapan yaitu *alpha testing* dan *beta testing*. Pada tahap *alpha testing* terdiri dari proses uji kelayakan sistem oleh 3 *web developer*. Uji kelayakan ini akan dibagi menjadi 3 bagian yang pertama pengujian pada aspek *functionality* yaitu dilakukan menggunakan *test case* berupa *checklist* dengan sasaran ahli media yang memiliki keahlian pada bidang pemrograman dan desain aplikasi *web*. Pengujian kedua pada aspek *efficiency* dan *reliability* dilakukan oleh penulis sendiri menggunakan instrumen yang telah di rencanakan

berupa *software*. Sedangkan untuk *beta testing* dilakukan untuk pengujian bagian terakhir yaitu pengujian *usability* akan divalidasi menggunakan instrumen berupa kuesioner dari Arnold M. Lund dengan sasaran pengguna yaitu guru dan siswa.

C. Sumber Data/Subjek Penelitian

Subjek penelitian untuk aspek *functionality* dibagi menjadi 2 subjek yaitu sistem informasi perpustakaan dan ahli *software* di bidang *web development* sebanyak 3 orang. Subjek penelitian untuk *efficiency* dan *reliability* adalah sistem informasi perpustakaan saja. Sedangkan subjek penelitian untuk aspek *usability* adalah beberapa guru sebagai admin dan juga siswa di SMK Muhammadiyah 1 Yogyakarta. Mengacu pada Nielsen (2006) yang menyatakan bahwa jumlah sampel paling sedikit dalam penelitian adalah 20 orang. Pada penelitian ini diambil jumlah responden sebanyak 30 orang. Hal ini dilakukan agar mendapatkan hasil yang signifikan dalam statistik. Kemudian dari 30 responden akan dibagi menjadi 2 bagian yang pertama adalah 3 orang guru dan 27 orang adalah siswa.

D. Metode dan Alat Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian adalah sebagai berikut:

1. Observasi

Teknik observasi dilakukan dengan mengumpulkan data berupa permasalahan-permasalahan dan kebutuhan yang ada di lapangan. Observasi ini dilakukan dengan cara wawancara dan pengamatan lingkungan mengenai kondisi yang ada di tempat penelitian.

2. Studi Literatur

Studi literatur digunakan untuk menemukan sumber riset dan informasi di internet maupun di buku-buku cetak.

E. Instrumen Penelitian

Instrumen penelitian yang digunakan dalam penelitian ini adalah:

1. Instrumen Aspek *Functionality*

Instrumen penelitian yang digunakan adalah *checklist* yang berisikan semua fungsi yang dikembangkan dalam aplikasi. *Checklist* dijabarkan dari fungsi-fungsi pada *software* apakah berjalan dengan baik atau tidak. Instrumen pengujian aspek *functionality* seperti ditunjukkan dalam tabel 3 berikut:

Tabel 3. Instrumen *functionality*

| No. | Fungsi | Lolos | |
|-----------|--------------------------------|-------|-------|
| | | Ya | Tidak |
| I. | User Admin | | |
| A | Akun | | |
| 1. | Login sebagai admin | | |
| 2. | Mengubah password admin | | |
| 3. | Lupa password | | |
| 4. | Logout | | |
| B | Mengelola Data Siswa | | |
| 5. | Melihat daftar siswa | | |
| 6. | Menambah data siswa | | |
| 7. | Mengedit data siswa | | |
| 8. | Mengubah foto siswa | | |
| 9. | Menghapus data siswa | | |
| 10. | Mencari data siswa | | |
| 11. | Export data siswa | | |
| C | Mengelola Member | | |
| 12. | Mencari data member | | |
| 13. | Melihat data peminjaman member | | |
| 14. | Menghapus member | | |
| D | Mengelola Data Buku | | |
| 15. | Melihat daftar buku | | |
| 16. | Menambah data buku | | |
| 17. | Mengedit data buku | | |
| 18. | Mengubah cover buku | | |
| 19. | Menghapus data buku | | |
| 20. | Mencari data buku | | |
| 21. | Import data buku | | |
| 22. | Export data buku | | |
| E | Mengelola Data Absensi | | |
| 23. | Melihat daftar absensi | | |
| 24. | Menghapus data absensi | | |
| 25. | Mencari daftar absensi | | |

Lanjutan Tabel 3. Instrumen *functionality*

| No. | Fungsi | Lolos | |
|-------------|----------------------------------|-------|-------|
| | | Ya | Tidak |
| 26. | Export data absensi | | |
| F | Mengelola Data Peminjaman | | |
| 27. | Melihat siswa yang meminjam | | |
| 28. | Mencari siswa yang meminjam | | |
| 29. | Melihat buku dipinjam | | |
| 30. | Mencari buku yang dipinjam | | |
| 31. | Filter berdasarkan tanggal | | |
| 32. | Laporan data peminjaman | | |
| G | Informasi Perpustakaan | | |
| 33. | Melihat informasi perpustakaan | | |
| 34. | Menambah informasi perpustakaan | | |
| 35. | Mengedit informasi perpustakaan | | |
| 36. | Menghapus informasi perpustakaan | | |
| 37. | Mencari informasi perpustakaan | | |
| H. | Data Statistik | | |
| 38. | Data statistik absensi | | |
| II. | User Siswa | | |
| A | Akun | | |
| 39. | Login sebagai siswa | | |
| 40. | Melihat data profil | | |
| 41. | Mengubah data profil | | |
| 42. | Mengubah password | | |
| 43. | Mengubah gambar profil | | |
| 44. | Lupa password | | |
| 45. | Logout | | |
| B | Peminjaman Buku | | |
| 46. | Melihat daftar buku | | |
| 47. | Mencari data buku | | |
| 48. | Meminjam buku | | |
| C | Pengembalian Buku | | |
| 49. | Melihat daftar buku | | |
| 50. | Mengembalikan buku | | |
| 47. | Mencari data buku | | |
| D | Informasi | | |
| 51. | Melihat informasi perpustakaan | | |
| E | Absensi | | |
| 52. | Absensi perpustakaan | | |
| III. | User Tamu | | |
| 53. | Melihat tentang perpustakaan | | |
| 54. | Melihat profil perpustakaan | | |
| 55. | Melihat Layanan Perpustakaan | | |
| 56. | Melihat daftar Buku | | |
| 57. | Pencarian Buku | | |

Untuk menguji sub-karakteristik *security* digunakan sebuah *software* yaitu *acunetix web vulnerability scanner* yang dapat memindai otomatis serangan-serangan keamanan yang biasa terjadi seperti *SQL Injection* dan *Cross Scripting (XSS)*. Dengan alat ini maka aspek *functionality* berkaitan dengan sub-karakteristik *security* dapat diuji apakah sistem dapat tetap aman atau tidak.

2. Instrumen Aspek *Efficiency*

Instrumen yang digunakan menggunakan aplikasi GTMetrix. Cara kerja dari aplikasi ini adalah memasukan *url* dari *website* yang akan diuji kemudian otomatis akan memberikan hasil dari parameter yang dihitung. Terdapat dua hasil pengujian yaitu *grade* dari YSlow dan Page Speed Insight. YSlow adalah aplikasi buatan *Yahoo Developer Network* yang bertujuan untuk menguji performa efisiensi dari sebuah halaman *website*. Hal yang di uji adalah besarnya *byte data*, jumlah HTTP *request*, kompresi GZIP, minifikasi dan *score/grade* akhir dll. Aplikasi Page Speed dari Google digunakan untuk menguji kecepatan *loading* suatu *website*.

3. Instrumen Aspek *Reliability*

Instrumen yang digunakan berupa *software loadimpact* dan WAPT versi 8.4 untuk pengujian *reliability*, cara kerja dari *software* ini yaitu dengan cara pengujian simulasi pengunjung sebanyak mungkin pada sistem yang diuji hingga server mengalami *down* atau sering disebut dengan *stress testing*. Pengujian menggunakan aplikasi dibagi menjadi beberapa parameter *Error Report* yaitu:

- 1) *Failed Session*
- 2) *Failed Hits*
- 3) *Failed Pages*

4. Instrumen Aspek Usability

Instrumen yang digunakan berupa kuesioner dari Arnold M.Lund yaitu USE (*Usefulness, Satisfaction, and Ease of use*). Kuesioner USE menggunakan skala Likert dengan bentuk *checklist*. Pada skala ini dibagi menjadi 5 bagian yaitu Sangat Setuju (SS), Setuju (S), Ragu-ragu (RG), Tidak Setuju (TS), dan Sangat Tidak Setuju (STS). Kuesioner USE dapat dilihat pada Tabel 4 sebagai berikut:

Tabel 4. Kuesioner USE

| No. | Pernyataan | Jawaban | | | | |
|-----|--|---------|----|----|---|----|
| | | STS | TS | RG | S | SS |
| 1. | <i>Software</i> ini membantu saya bekerja lebih efektif | | | | | |
| 2. | <i>Software</i> ini membantu saya bekerja lebih produktif | | | | | |
| 3. | <i>Software</i> ini sangat berguna | | | | | |
| 4. | <i>Software</i> ini memberikan saya pengendalian lebih atas aktivitas saya | | | | | |
| 5. | <i>Software</i> ini mempermudah saya dalam menyelesaikan apa yang ingin saya selesaikan | | | | | |
| 6. | <i>Software</i> ini menghemat waktu saya ketika saya menggunakannya | | | | | |
| 7. | <i>Software</i> ini sesuai dengan kebutuhan saya | | | | | |
| 8. | <i>Software</i> ini melakukan segala sesuatu yang saya harapkan untuk dilakukan | | | | | |
| 9. | <i>Software</i> ini mudah digunakan | | | | | |
| 10. | <i>Software</i> ini praktis digunakan | | | | | |
| 11. | <i>Software</i> ini mudah dipahami | | | | | |
| 12. | <i>Software</i> ini membutuhkan langkah-langkah yang sedikit untuk mencapai apa yang ingin saya lakukan dengan <i>software</i> ini | | | | | |
| 13. | <i>Software</i> ini fleksibel | | | | | |
| 14. | Tidak ada kesulitan dalam menggunakan <i>Software</i> ini | | | | | |
| 15. | Saya dapat menggunakan <i>Software</i> ini tanpa panduan tertulis | | | | | |
| 16. | Saya tidak melihat adanya inkonsistensi saat saya gunakan <i>software</i> ini | | | | | |
| 17. | Baik pengguna yang sesekali menggunakan dan pengguna yang biasa menggunakan akan menyukai <i>software</i> ini | | | | | |
| 18. | Saya dapat menangani kesalahan dengan cepat dan mudah | | | | | |

Lanjutan Tabel 4. Kuesioner USE

| No. | Pernyataan | Jawaban | | | | |
|-----|---|---------|----|----|---|----|
| | | STS | TS | RG | S | SS |
| 19. | Saya dapat menggunakan <i>software</i> ini secara benar setiap saat | | | | | |
| 20. | Saya belajar untuk menggunakan <i>software</i> ini secara cepat | | | | | |
| 21. | Saya mudah mengingat bagaimana menggunakan ini | | | | | |
| 22. | <i>Software</i> ini mudah untuk dipelajari bagaimana penggunaannya | | | | | |
| 23. | Saya menjadi terampil menggunakan <i>Software</i> ini secara cepat | | | | | |
| 24. | Saya puas dengan <i>software</i> ini | | | | | |
| 25. | Saya akan merekomendasikan <i>software</i> ini ke teman | | | | | |
| 26. | <i>Software</i> ini menyenangkan untuk digunakan | | | | | |
| 27. | <i>Software</i> ini bekerja seperti yang saya inginkan | | | | | |
| 28. | <i>Software</i> ini memiliki tampilan yang sangat bagus | | | | | |
| 29. | Menurut saya, saya perlu memiliki <i>software</i> ini | | | | | |
| 30. | <i>Software</i> ini nyaman untuk digunakan | | | | | |

F. Teknik Analisis Data

Penjelasan teknik analisis data berdasarkan masing-masing kriteria pengujian kualitas *software* dari ISO 9126 adalah sebagai berikut:

1. Analisis Kualitas Aspek *Functionality*

Pengujian aspek *functionality* dilakukan dengan melakukan tes pada setiap fungsi perangkat lunak oleh ahli *web development*. Pengujian menggunakan *test case* berbentuk *checklist* dengan jawaban tegas yaitu “Ya-Tidak”. Standar yang digunakan dalam menentukan apakah perangkat lunak telah lulus pengujian adalah dengan menggunakan interpretasi standar yang digunakan oleh ISO 9126 dengan rumus analisis data sebagai berikut:

$$X = 1 - \frac{A}{B}$$

Keterangan:

$X = \text{Functionality}$

A = Jumlah fungsi yang gagal uji

B = Jumlah seluruh fungsi

$0 \leq X \leq 1$. *Functionality* dikatakan baik jika mendekati 1.

Berdasarkan dari rumus pengujian diatas, akan memperoleh nilai X dimana X lebih besar dari atau sama dengan 0 dan X kurang dari atau sama dengan 1. *Software* dikatakan telah memenuhi standar jika X lebih dari 0.5 dan mendekati 1. Selanjutnya pada aspek sub-karakteristik *security* dengan menggunakan *acunetix web vulnerability scanner* akan didapatkan hasil dalam bentuk *threat level*. Terdapat beberapa *threat level* yaitu Level 1 *Low* yang berarti rendah kerentanan (sangat aman). Level 2 *Medium* yang berarti memiliki kerentanan sedang (kurang aman). Level 3 *High* yang berarti memiliki kerentanan tinggi (tidak aman). *Software* dinyatakan lulus uji dari keamanan jika hasil yang di dapat antara *Level 2 Medium* dan *Level 1 Low*.

2. Analisis Kualitas Aspek *Efficiency*

Efficiency pada sebuah *website* yaitu seberapa cepat *website* dapat diakses dan dapat menampilkan kontennya pada sebuah *web browser*. Pada pengujian ini menggunakan tools bantuan GTMetrix yang akan memberikan *grade YSlow* dan *PageSpeed*. Pengujian ini ditujukan untuk sub-karakteristik *time behavior* yaitu menguji kecepatan *web* untuk merespon dan memproses perintah yang diinginkan. Kemudian hasil waktu *load* yang didapat diukur respon dan aksesnya menggunakan interpretasi dari J.Nielsen yang membatasi waktu *load* maksimal 10 detik.

3. Analisis Kualitas Aspek *Reliability*

Nilai *Reliability* dari sebuah *website* di uji dengan menggunakan *software Web Application Load, Stress, and Performance Testing (WAPT)*. Kemudian hasil yang didapat akan di bandingkan dengan standar Telcordia, jika lebih dari 95% maka sistem yang diuji telah memenuhi aspek *reliability*.

4. Analisis Kualitas Aspek *Usability*

Pengujian aspek *usability* menggunakan kuesioner yang akan dibagikan kepada 30 responden sebagai pengguna yang terdiri dari 27 siswa dan 3 guru. Untuk menentukan apakah pengujian *usability* layak atau tidak dengan cara menghitung perkalian hasil instrumen dengan jumlah penjawab. Setiap jawaban diberi skor untuk dianalisis. Berikut contoh skala likert yang dikemukakan Sugiyono (2012:94):

- a. SS = Sangat Setuju (Skor 5)
- b. S = Setuju (Skor 4)
- c. RG = Ragu-Ragu (Skor 3)
- d. TS = Tidak Setuju (Skor 2)
- e. STS = Sangat Tidak Setuju (Skor 1)

Sugiyono menjelaskan untuk melakukan analisis data hasil pengujian *usability* dengan menghitung jumlah rata-rata jawaban berdasarkan skor. Misalnya dari 30 responden didapatkan hasil seperti pada Tabel 5 sebagai berikut.

Tabel 5. Contoh Hasil Jawaban Responden

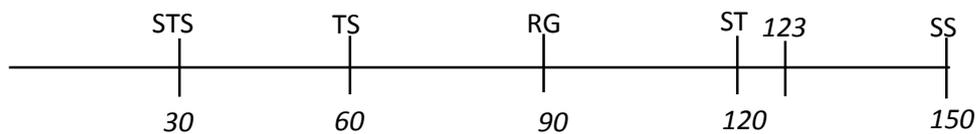
| Jawaban | Jumlah penjawab |
|---------|-----------------|
| SS | 15 |
| S | 8 |
| RG | 3 |
| TS | 3 |
| STS | 1 |

Berdasarkan skor yang telah ditetapkan dapat dihitung sebagai berikut:

| | | |
|-------------------------------|----------|-------|
| Jumlah skor yang menjawab SS | = 15 x 5 | = 75 |
| Jumlah skor yang menjawab S | = 8 x 4 | = 32 |
| Jumlah skor yang menjawab RG | = 3 x 3 | = 9 |
| Jumlah skor yang menjawab TS | = 3 x 2 | = 6 |
| Jumlah skor yang menjawab STS | = 1 x 1 | = 1 |
| <hr/> | | |
| Jumlah skor Total | | = 123 |

Jumlah skor ideal untuk seluruh item = 5 x 30 = 150. Jadi tingkat persetujuannya berdasarkan data tersebut = $(123 : 150) \times 100\% = 82\%$ dari yang diharapkan.

Secara kontinum dapat digambarkan seperti pada Gambar 8.



Gambar 8. Contoh Tingkat Persetujuan Responden pada *Software*

Berdasarkan data yang diperoleh dari 30 responden maka skor 123 terletak pada daerah Sangat Setuju. Kemudian untuk mengetahui *reliabilitas* instrumen *usability* dilakukan dengan menguji nilai konsistensi *usability* Arnold M. Lund dengan mencari nilai *cronbach's alpha* digunakan *software* IBM SPSS *Statistics* versi 22. Kemudian hasil perhitungan akan dikomparasikan dengan Tabel 6 sebagai berikut:

Tabel 6. *Internal Consistency Cronbach's Alpha* (Gliem & Gliem, 2003:87)

| Cronbach's Alpha | Internal Consistency |
|-------------------------|-----------------------------|
| $\alpha \geq .9$ | <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> |

Software dikatakan memiliki *reliability* yang baik jika nilai *reliability* melebihi 0.9 dan mendekati 1.0.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Tahap Analisis Kebutuhan

1. Analisis Kebutuhan Sistem

Pengembangan sistem informasi perpustakaan diawali dengan analisis kebutuhan sistem. Analisis ini diperlukan agar dalam proses pengembangan tepat sasaran dan difungsikan dengan baik sebagai sistem informasi perpustakaan untuk SMK Muhammadiyah 1 Yogyakarta. Kebutuhan minimal yang harus ada menurut hasil observasi dan wawancara adalah sebagai berikut:

- a. Sistem informasi dapat digunakan untuk mengolah data sirkulasi peminjaman dan pengembalian buku.
- b. Sistem informasi dapat digunakan untuk mencari informasi koleksi buku.
- c. Sistem informasi dapat digunakan untuk mendata kunjungan perpustakaan.
- d. Sistem informasi dapat digunakan untuk mengolah data siswa.
- e. Sistem informasi dapat digunakan untuk mengolah data penulis.
- f. Sistem informasi dapat digunakan untuk mengolah data buku.
- g. Sistem informasi dapat digunakan untuk mengolah dan menampilkan informasi terkait perpustakaan.
- h. Sistem informasi dapat membantu pembuatan laporan mengenai data transaksi peminjaman, data siswa, data penulis, koleksi buku, dan data pengunjung perpustakaan.

2. Analisis Kebutuhan *Hardware* dan *Software*

Analisis spesifikasi kebutuhan *hardware* dan *software* dilakukan untuk mengetahui kebutuhan agar sistem informasi sekolah dapat dijalankan dengan baik tanpa terkendala. Spesifikasi dari alat-alat yang digunakan untuk menjalankan sistem ini antara lain:

- a. PC/Laptop
- b. *Framework Laravel*
- c. *Framework CSS Bootstrap*
- d. *Web Server XAMPP* versi 5.6.3
- e. *Database server MySQL*
- f. *Web browser*

B. Tahap Desain

Desain sistem yang akan dikembangkan meliputi perancangan *Unified Modelling Language* (UML), basis data (*database*), dan tampilan (*user interface*).

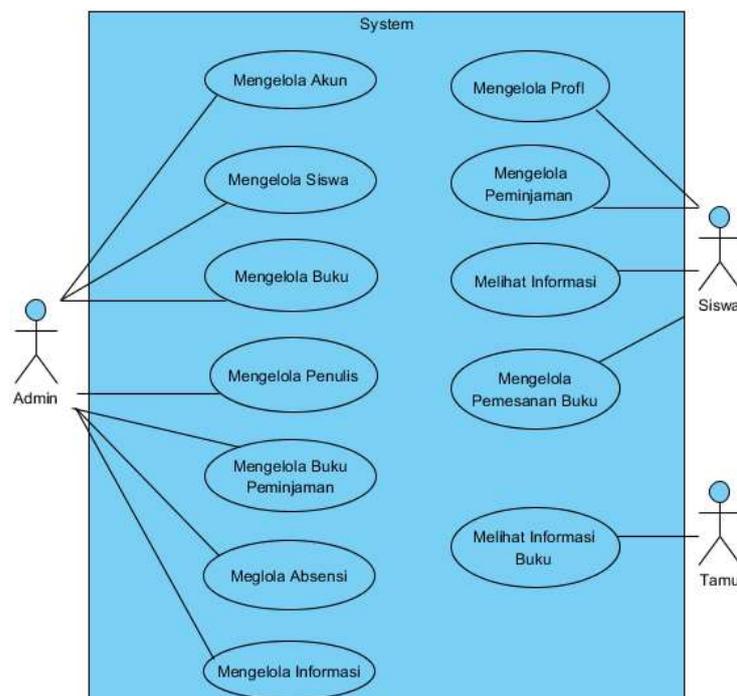
1. Perancangan *Unified Modeling Language* (UML)

a. *Use Case Diagram*

Use Case Diagram terdiri dari aktor dan interaksi yang dilakukannya dalam sebuah sistem. Pada pengembangan perangkat lunak *use case diagram* digunakan untuk menjelaskan hubungan dan aktor baik berupa *input* atau *output* dalam sebuah sistem. *Use case diagram* untuk sistem informasi yang dikembangkan adalah sebagai berikut:

1) *Use Case Diagram* Sistem

Use case diagram sistem terdapat 3 aktor yaitu admin, siswa dan tamu. Adapun masing-masing dijelaskan pada Gambar 9 sebagai berikut:



Gambar 9. Use Case Diagram Sistem

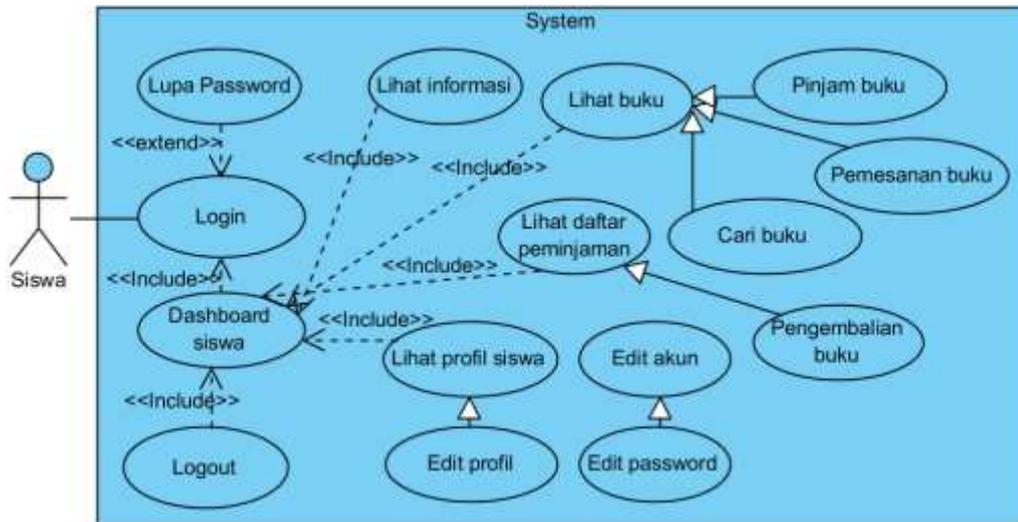
Use case diagram menggambarkan aktor dan hubungan dengan fungsi-fungsinya masing-masing. Dalam sistem informasi yang dikembangkan terdapat 3 aktor yaitu admin, siswa dan tamu. Admin memiliki fungsi diantaranya dapat mengelola data akun, mengelola data siswa, mengelola data buku, mengelola peminjaman buku, mengelola data absensi kunjungan perpustakaan, dan mengelola data informasi. Sedangkan siswa memiliki fungsi dapat mengelola profil, mengelola peminjam dan pengembalian buku, dan melihat informasi perpustakaan. Terakhir adalah tamu yang hanya dapat melihat informasi buku yang ada di perpustakaan. Fungsi yang ada pada aktor admin dan siswa harus melalui *use case* login terlebih dahulu. Sedangkan untuk aktor tamu tidak diperlukan *use case* login.

2) Use Case Diagram Admin

Use case diagram untuk aktor admin dapat digambarkan pada Gambar 10 sebagai berikut:

3) Use Case Diagram Siswa

Use case diagram untuk aktor siswa dapat dilihat pada Gambar 11 sebagai berikut:



Gambar 11. Use Case Diagram Siswa

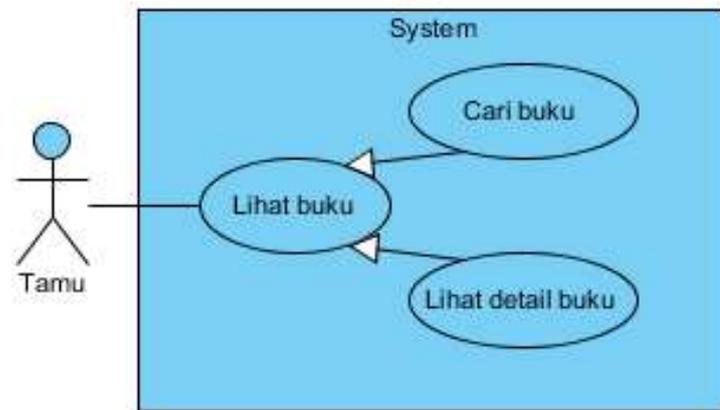
Penjabaran use case diagram untuk pengembangan sistem ini didefinisikan atas definisi aktor, definisi use case dan skenario use case. Penjabarannya dijelaskan dalam Tabel 8 sebagai berikut:

Tabel 8. Definisi Aktor Siswa

| Aktor | Deskripsi |
|-------|--|
| Siswa | Siswa merupakan aktor yang memiliki hak akses terbatas terhadap pengelolaan sistem informasi. Fungsi siswa diantaranya adalah pengelolaan terhadap profil pribadi, peminjaman dan pengembalian buku, dan melihat informasi perpustakaan. Syarat untuk mengakses fungsi tersebut adalah harus melalui fungsi login sebagai siswa terlebih dahulu. Kemudian pada setiap fungsi terdapat beberapa fungsi tambahan sesuai dengan kebutuhannya masing-masing. |

4) Use Case Diagram Tamu

Use case diagram untuk aktor tamu dapat dilihat pada Gambar 12 sebagai berikut:



Gambar 12. Use Case Diagram Tamu

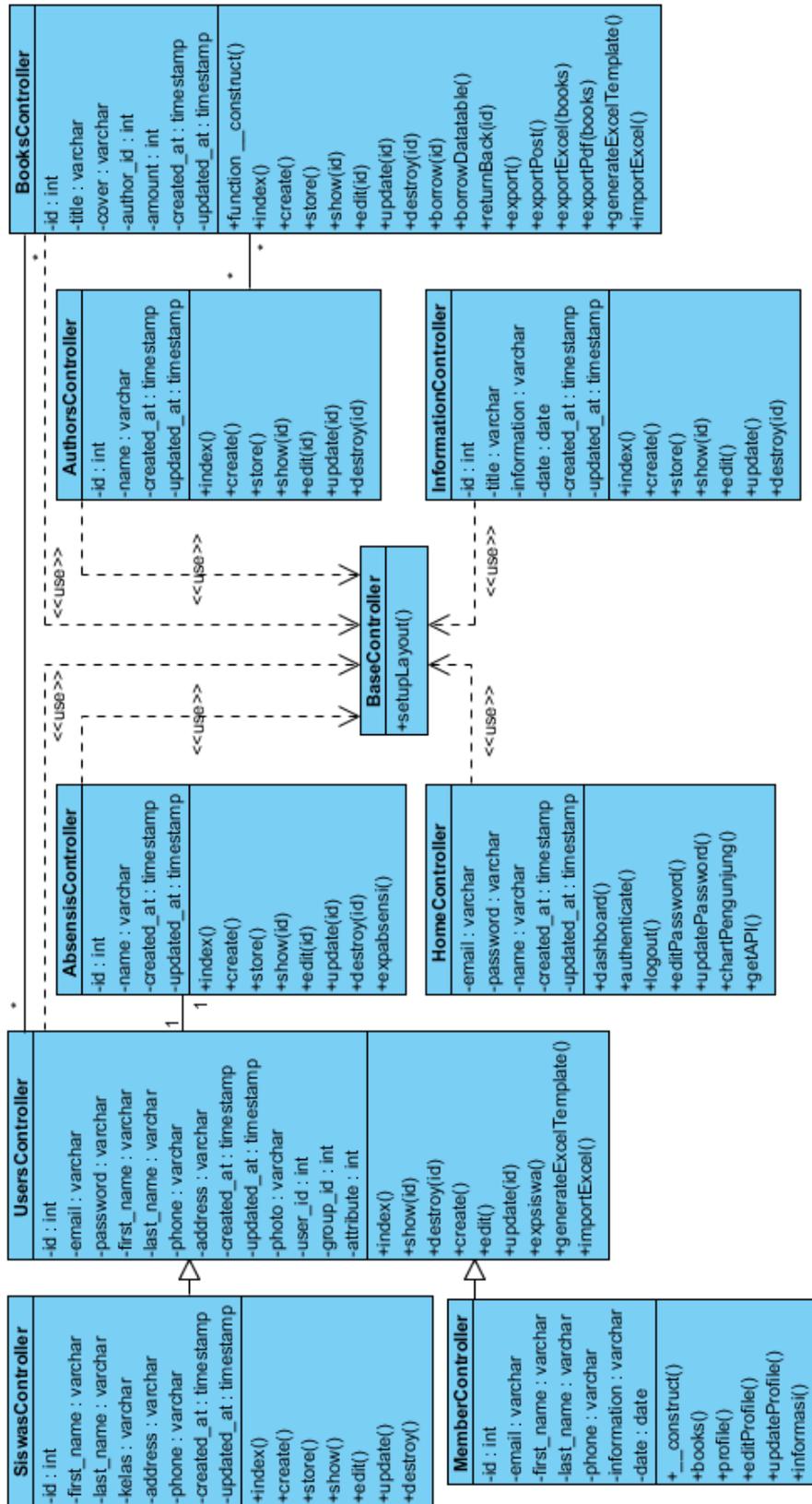
Penjabaran use case diagram untuk pengembangan sistem ini didefinisikan atas definisi aktor, definisi use case dan skenario use case. Penjabarannya dijelaskan dalam Tabel 9 sebagai berikut:

Tabel 9. Definisi Aktor Tamu

| Aktor | Deskripsi |
|-------|--|
| Tamu | Tamu merupakan aktor yang hanya memiliki hak akses yang sangat terbatas yaitu hanya memiliki fungsi untuk melihat informasi buku. Untuk mengakses fungsi tersebut tidak diperlukan fungsi login terlebih dahulu. |

b. Class Diagram

Class diagram sistem informasi perpustakaan dapat dilihat pada Gambar 13 sebagai berikut:



Gambar 13. Class Diagram Sistem

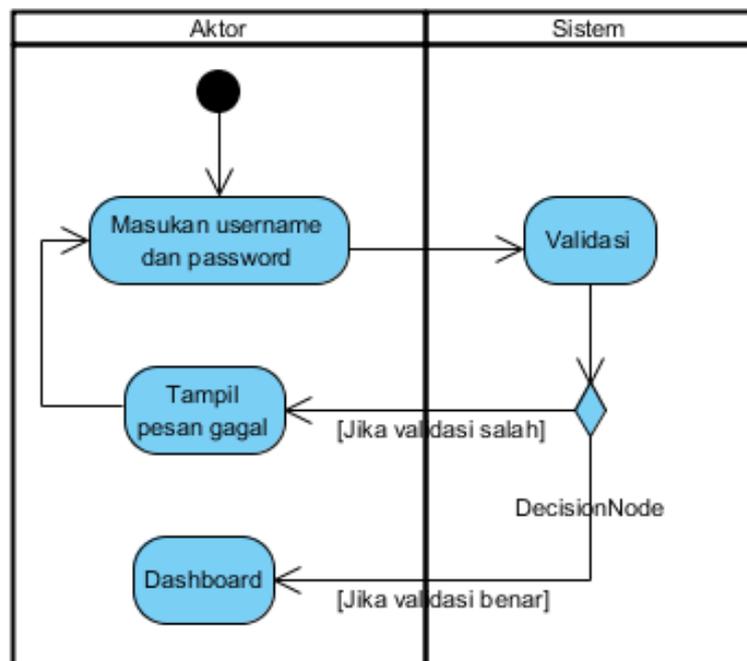
Diagram tersebut menggambarkan fungsi-fungsi yang ada dalam sistem informasi perpustakaan berbasis *framework* Laravel. *Framework* ini menggunakan konsep MVC yang dibangun dari *model*, *view* dan *controller*. Antara ketiganya memiliki hubungan satu sama lain berdasarkan fungsinya dalam aplikasi. Dalam *framework* Laravel ini juga terdapat berbagai macam *class library* yang dapat digunakan untuk membantu pembuatan fungsi tertentu, kemudian untuk *request* dari setiap fungsi seluruhnya ditangani pada *class routes*.

c. Activity Diagram

Berdasarkan *use case* diagram yang telah dibuat, maka *activity* diagram yang digambarkan dapat dilihat sebagai berikut:

1) Activity Diagram Login

Activity diagram login untuk sistem informasi perpustakaan dijelaskan dalam Gambar 14 sebagai berikut:

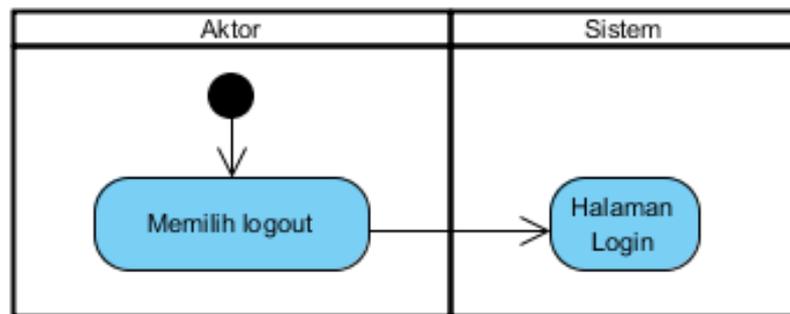


Gambar 14. Activity Diagram Login

Berdasarkan diagram tersebut aktor admin memasukan *username* dan *password* yang kemudian akan divalidasi oleh sistem, kemudian proses selanjutnya jika validasi benar maka sistem akan otomatis menampilkan halaman *dashboard* admin, akan tetapi jika gagal maka sistem akan menampilkan pesan kesalahan login sesuai dengan kesalahan masukan *user*.

2) Activity Diagram Logout

Activity diagram logout untuk sistem informasi perpustakaan dijelaskan dalam Gambar 15 sebagai berikut:

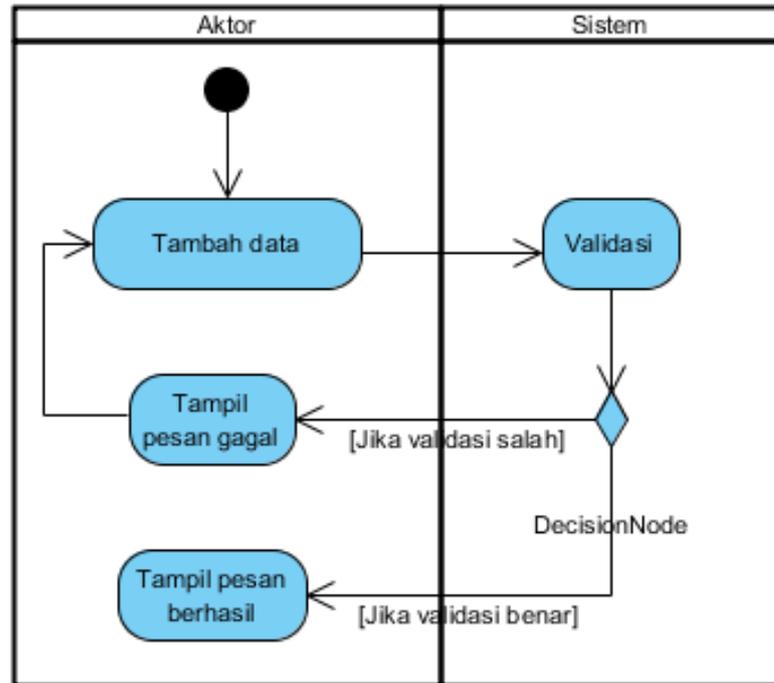


Gambar 15. Activity Diagram Logout

Berdasarkan diagram tersebut aktor admin memilih menu logout. Setelah aktor memilih menu, maka sistem akan melakukan *route* ke halaman login.

3) Activity Diagram Tambah Data

Activity diagram tambah data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 16 sebagai berikut:

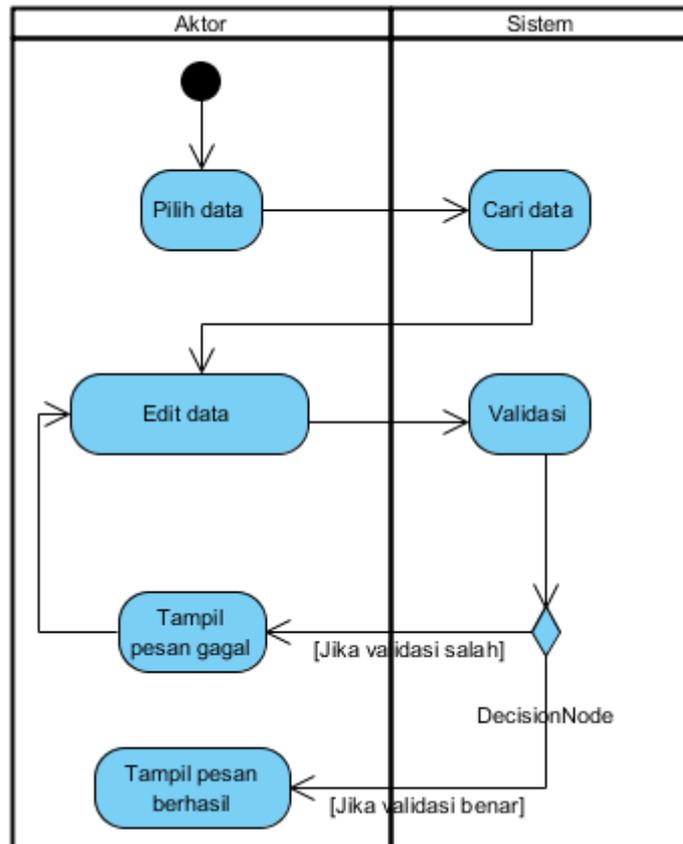


Gambar 16. *Activity* Diagram Tambah Data

Berdasarkan diagram tersebut aktor admin memasukan data pada *form* data. Setelah aktor mengisi *form* yang telah disediakan dan melakukan *submit*, maka sistem akan melakukan validasi, jika validasi sukses maka sistem akan menampilkan pesan penyimpanan berhasil, jika gagal maka sistem akan menampilkan pesan penyimpanan gagal beserta kesalahan inputan yang dilakukan oleh *user*.

4) *Activity* Diagram Edit Data

Activity diagram edit data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 17 sebagai berikut:

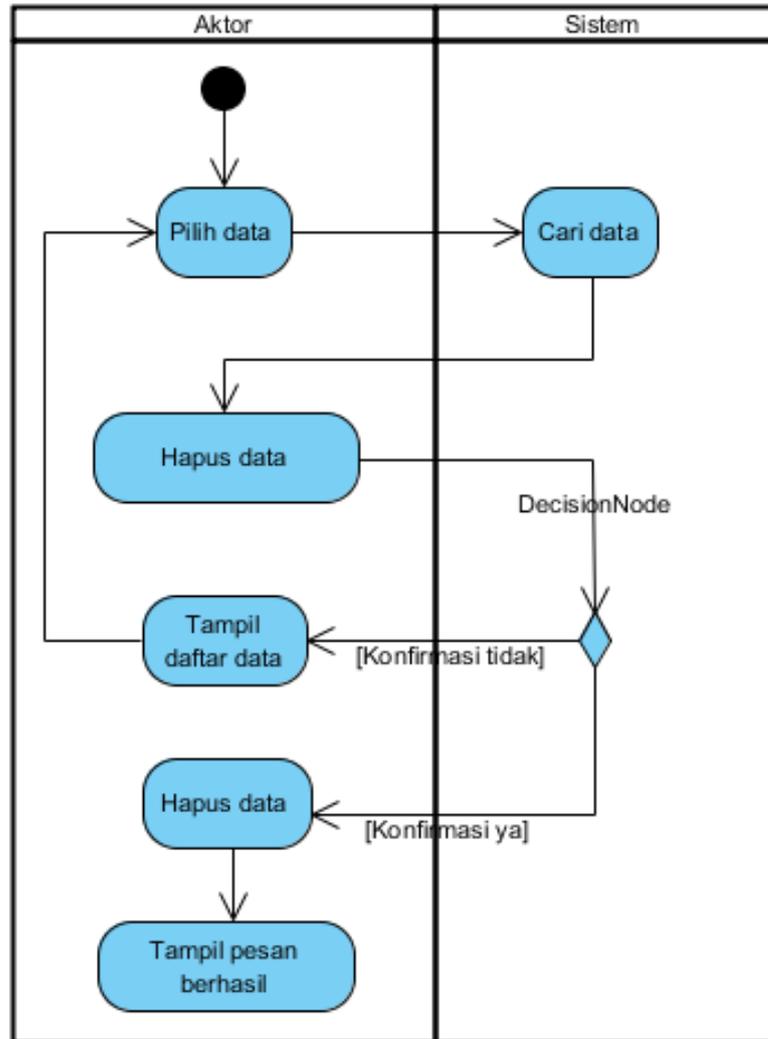


Gambar 17. Activity Diagram Edit Data

Berdasarkan diagram tersebut aktor admin memilih data yang akan di edit dengan cara mencari data tersebut, kemudian sistem akan menampilkan *form* edit. Setelah aktor mengisi *form* yang telah disediakan dan melakukan *submit*, maka sistem akan melakukan validasi, jika validasi sukses maka sistem akan menampilkan pesan penyimpanan berhasil, jika gagal maka sistem akan menampilkan pesan penyimpanan gagal beserta kesalahan inputan yang dilakukan oleh *user*.

5) Activity Diagram Hapus Data

Activity diagram hapus data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 18 sebagai berikut:

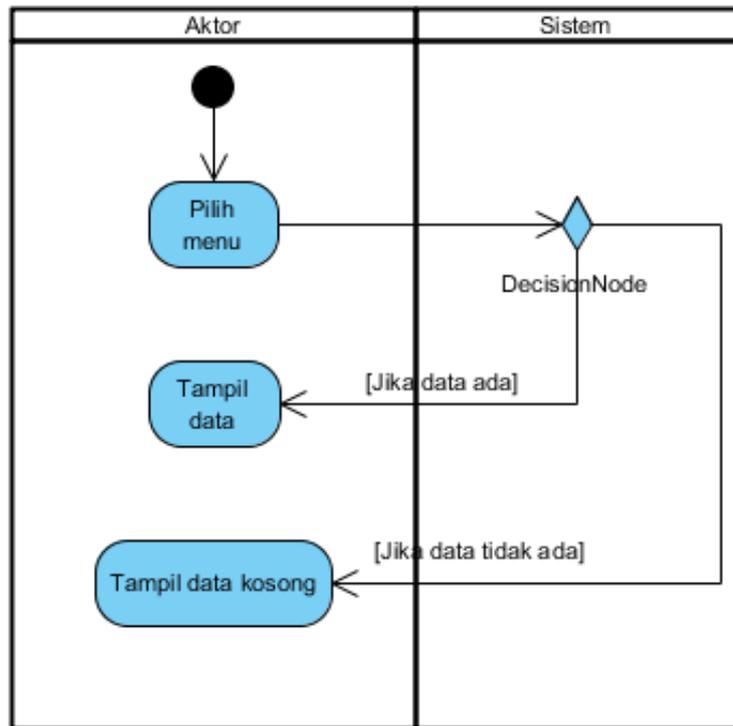


Gambar 18. Activity Diagram Hapus Data

Berdasarkan diagram tersebut aktor admin memilih data yang akan di hapus dengan cara mencari data tersebut, kemudian sistem akan menampilkan konfirmasi berupa pesan dialog. Jika aktor admin memilih pilihan Ya maka data akan dihapus dan sistem menampilkan pesan berhasil, sedangkan jika memilih pilihan Tidak maka sistem akan kembali menampilkan daftar data.

6) Activity Diagram Menampilkan Data

Activity diagram menampilkan data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 19 sebagai berikut:



Gambar 19. Activity Diagram Tampil Data

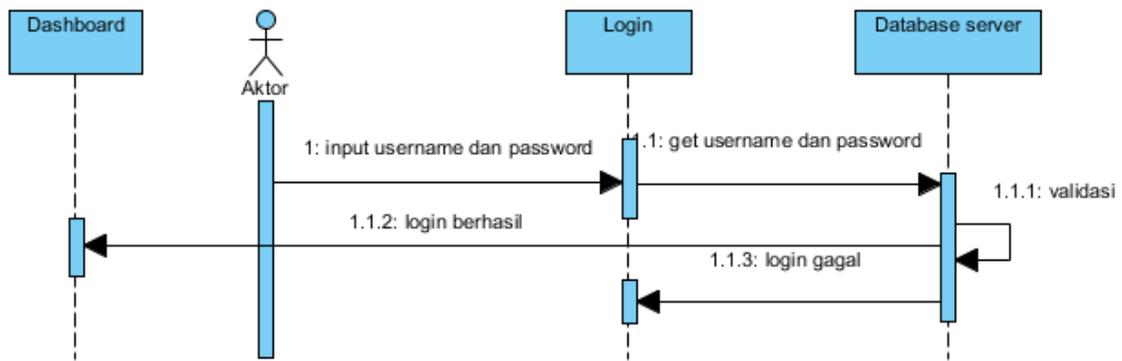
Berdasarkan diagram tersebut aktor admin memilih menu yang ada di sistem, kemudian sistem akan menampilkan data berupa sesuai dengan menu yang dipilih oleh admin. Setelah itu sistem akan menampilkan daftar data, jika data ada dalam *database* maka data akan ditampilkan, jika tidak ada maka sistem tidak akan menampilkan daftar data.

d. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display dan sebagainya) berupa pesan yang digambarkan terhadap waktu. *Sequence* diagram ini terdiri dari dimensi vertical (waktu) dan horizontal (objek-objek terkait).

1) Sequence Diagram Login

Sequence diagram login dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 20 sebagai berikut:

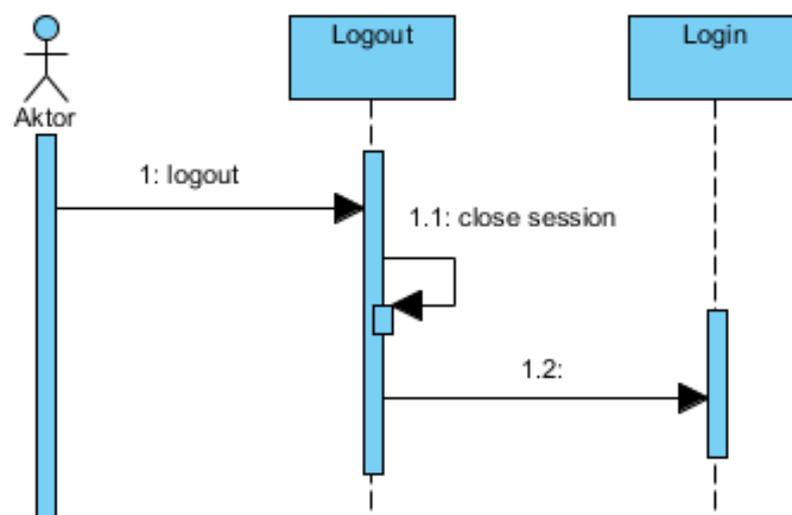


Gambar 20. *Sequence Diagram Login*

Berdasarkan *sequence diagram* diatas, dapat dilihat aktor memasukan *username* dan *password*. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan membawa aktor ke halaman *dashboard* sesuai dengan tingkatan pengguna. Sedangkan jika validasi salah maka sistem akan kembali ke halaman login.

2) *Sequence Diagram Logout*

Sequence diagram logout dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 21 sebagai berikut:

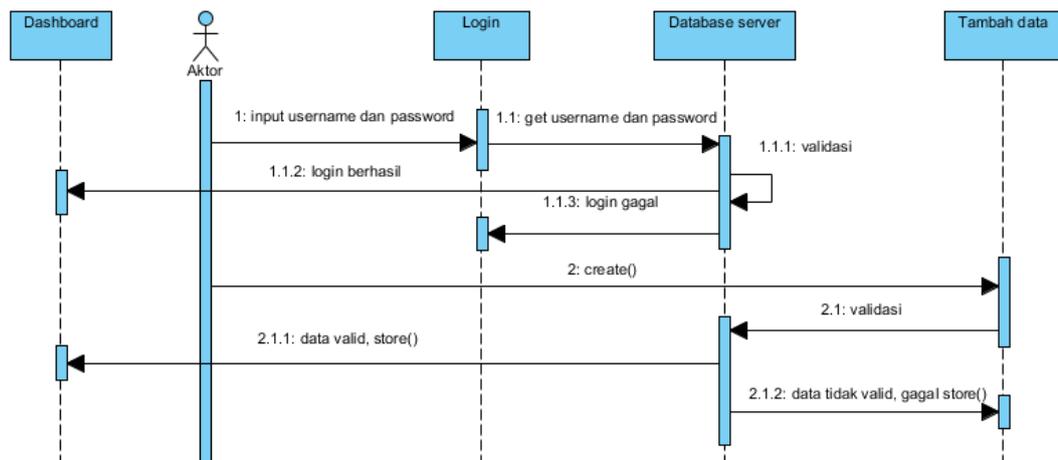


Gambar 21. *Sequence Diagram Logout*

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor memilih menu logout. Kemudian sistem akan menerima masukan dan *close session* yang sedang disimpan. Jika berhasil maka sistem akan membawa aktor ke halaman login.

3) *Sequence* Diagram Tambah Data

Sequence diagram tambah data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 22 sebagai berikut:

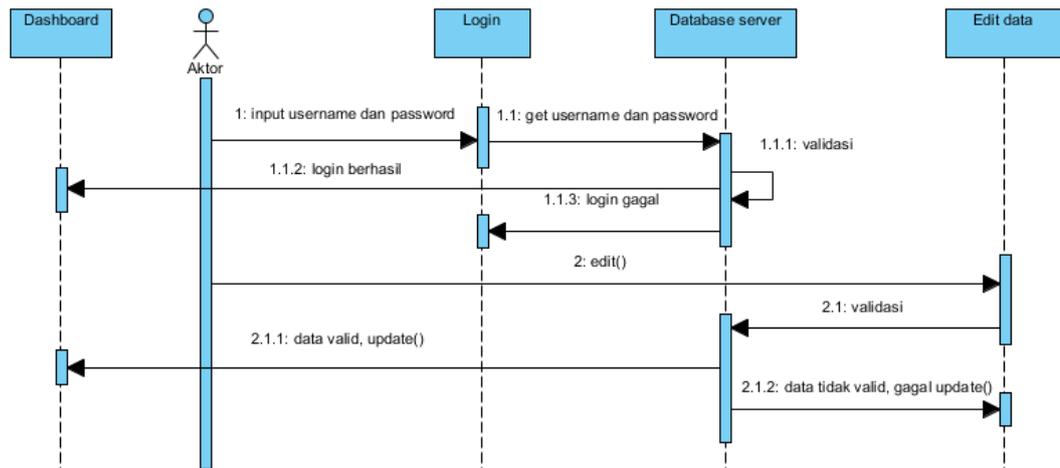


Gambar 22. *Sequence* Diagram Tambah Data

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan masukan data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan menyimpan data dan membawa aktor ke halaman *dashboard*. Sedangkan jika validasi salah maka sistem akan kembali ke halaman tambah data.

4) *Sequence* Diagram Edit Data

Sequence diagram edit data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 23 sebagai berikut:

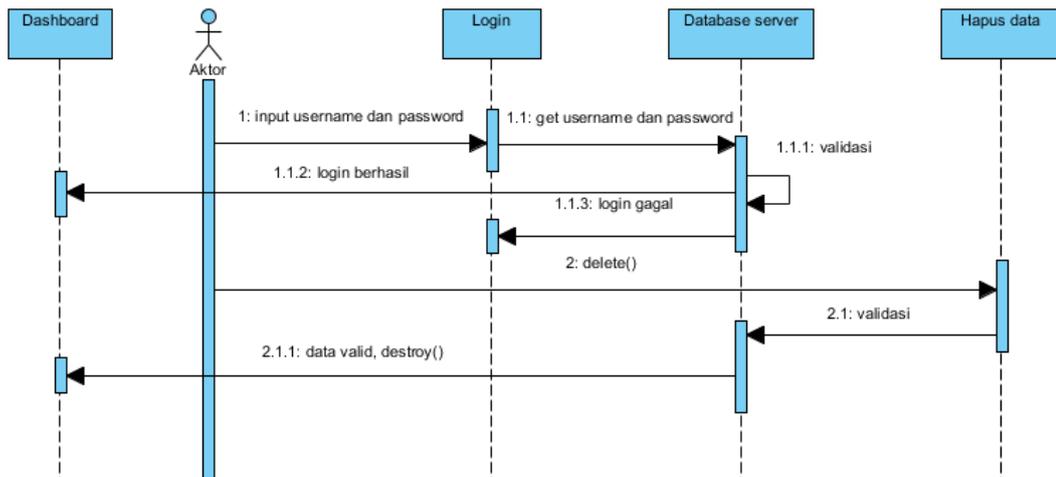


Gambar 23. *Sequence Diagram* Edit Data

Berdasarkan *sequence diagram* diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukan *username* dan *password*. Setelah berhasil login, aktor melakukan edit data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam database dan melakukan validasi. Jika validasi benar maka sistem akan memperbaharui data dan membawa aktor ke halaman *dashboard*. Sedangkan jika validasi salah maka sistem akan kembali ke halaman edit data.

5) *Sequence Diagram* Hapus Data

Sequence diagram hapus data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 24 sebagai berikut:

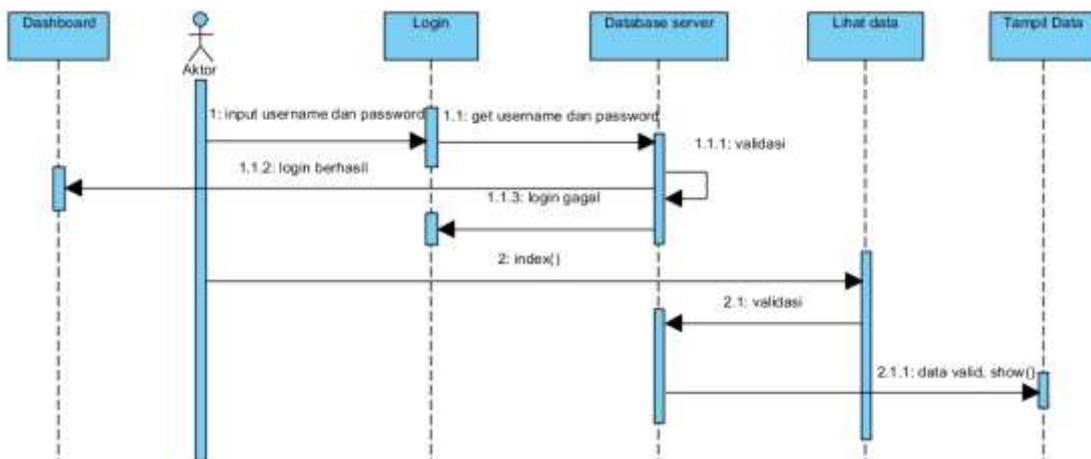


Gambar 24. *Sequence Diagram Hapus Data*

Berdasarkan *sequence diagram* diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan hapus data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang dipilih *user* dan melakukan penghapusan data.

6) *Sequence Diagram Lihat Data*

Sequence diagram lihat data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 25 sebagai berikut:

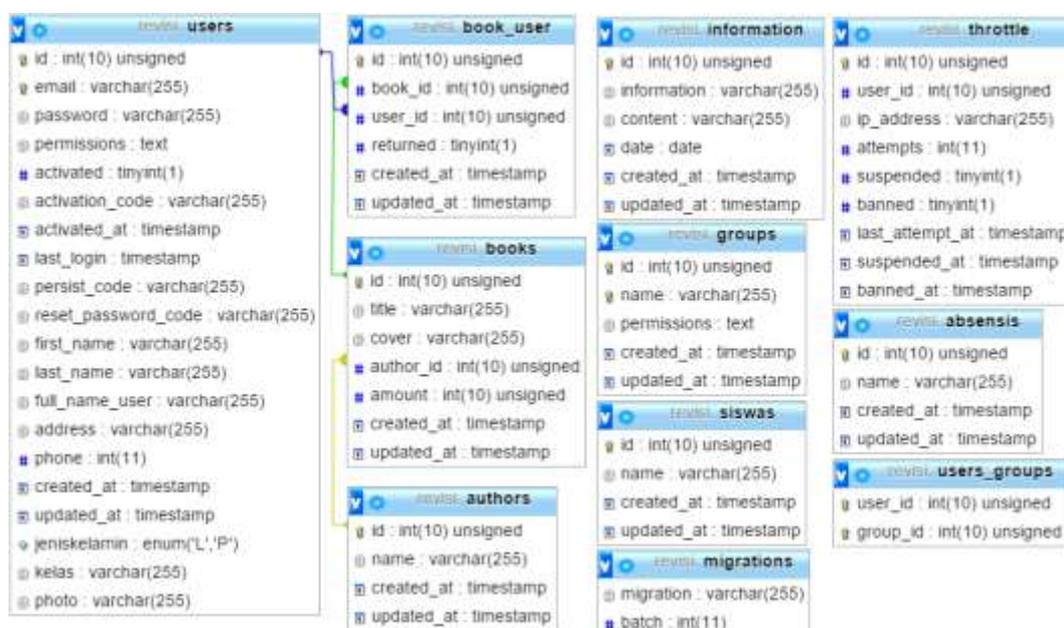


Gambar 25. *Sequence Diagram Lihat Data*

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor memilih melihat data. Kemudian aktor memilih data yang akan dilihat, sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan menampilkan data.

2. Perancangan Basis Data (*Database*)

Perancangan *database* dalam sistem informasi perpustakaan yang dikembangkan adalah seperti pada Gambar 26 sebagai berikut:



Gambar 26. Perancangan Basis Data

Perancangan *database* terdiri dari 11 tabel. Tabel *users* memiliki 21 atribut yang digunakan untuk menyimpan data pengguna yaitu admin dan siswa, tabel *book_user* memiliki 6 atribut yang digunakan untuk menyimpan data peminjaman buku, tabel *books* memiliki 7 atribut yang digunakan untuk menyimpan data buku, tabel *authors* memiliki 4 atribut yang digunakan untuk menyimpan data penulis buku, tabel *information* memiliki 5 atribut yang digunakan untuk menyimpan data

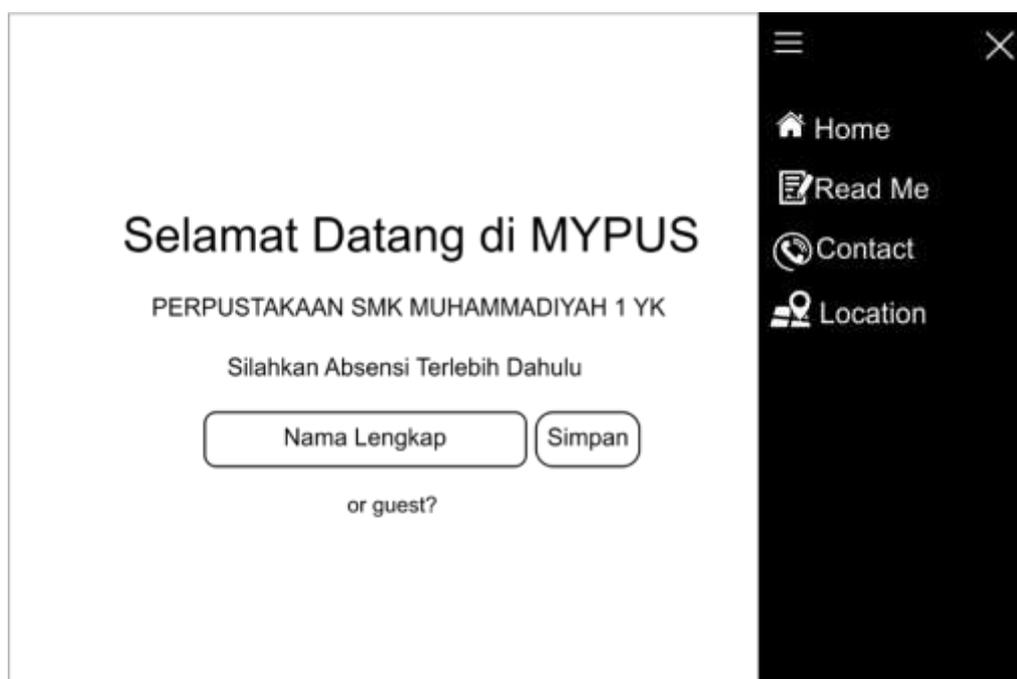
informasi perpustakaan, tabel *groups* memiliki 5 atribut yang digunakan untuk menyimpan data grup admin maupun siswa, tabel *siswas* memiliki 4 atribut yang digunakan untuk menyimpan data pengguna yaitu siswa, tabel *absensis* memiliki 4 atribut yang digunakan untuk menyimpan data pengunjung perpustakaan, tabel *user_groups* memiliki 2 atribut yang digunakan untuk menyimpan data grup admin maupun siswa.

3. Perancangan Desain *Interface* Sistem

Desain *interface* sistem informasi perpustakaan yang dikembangkan digambarkan sebagai berikut:

1) Halaman Utama

Desain *interface* halaman utama sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 27 sebagai berikut:



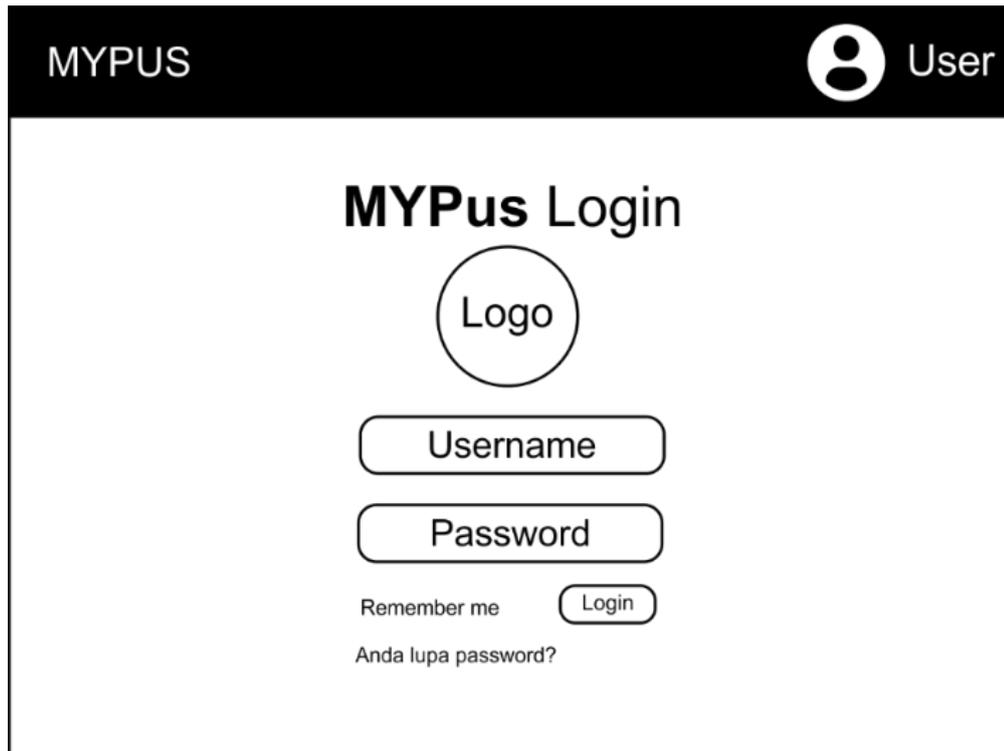
Gambar 27. Desain *Interface* Halaman Utama

Ketika sistem informasi perpustakaan di akses maka yang pertama kali muncul adalah halaman utama. Halaman ini terdiri dari menu *home*, *read me*, *contact*, dan

location. Pada bagian tengah terdapat kotak isian yang digunakan untuk proses absensi dan menu pilihan sebagai tamu.

2) Halaman Login

Desain *interface* halaman login sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 28 sebagai berikut:



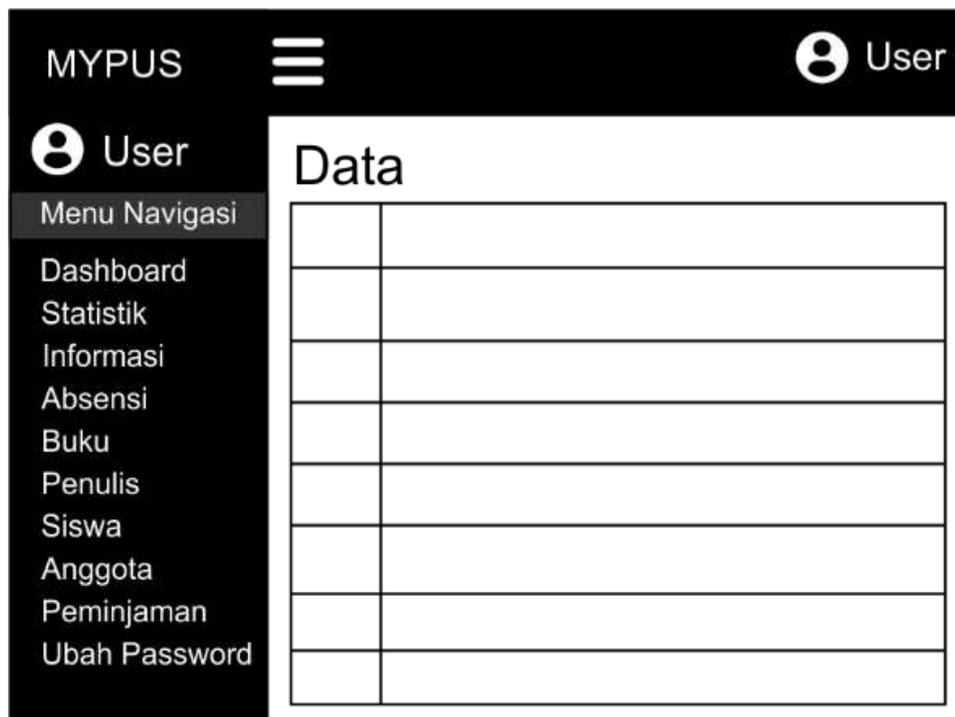
The image shows a login interface for MYPUS. At the top, there is a black header with 'MYPUS' on the left and a user profile icon labeled 'User' on the right. The main content area is white and contains the following elements: the title 'MYPUS Login' in bold, a circular logo placeholder with the word 'Logo' inside, a rounded rectangular input field for 'Username', another rounded rectangular input field for 'Password', a 'Remember me' checkbox, a 'Login' button, and a link that says 'Anda lupa password?'.

Gambar 28. Desain *Interface* Halaman Login

Halaman *login* digunakan untuk masuk ke dalam sistem dengan cara memasukkan *username* dan *password* dengan tujuan sebagai autentifikasi apakah *user* masuk sebagai admin atau siswa. Terdapat pilihan lupa *password* jika *user* tidak bisa melakukan login.

3) Halaman *Dashboard* Admin

Desain *interface* halaman *dashboard* admin sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 29 sebagai berikut:

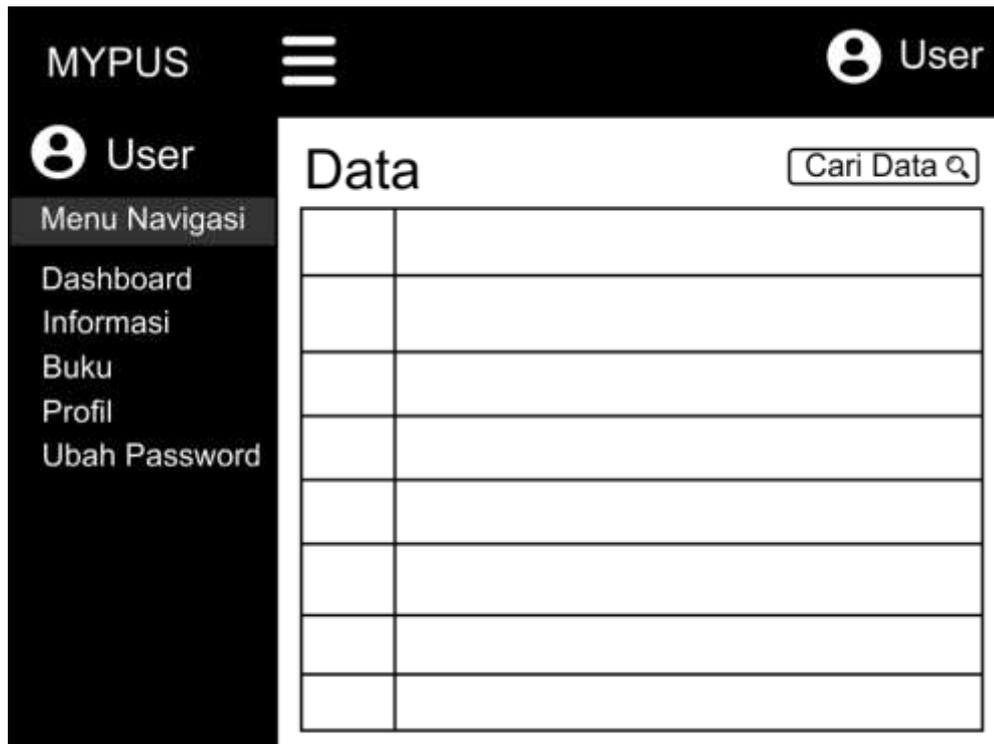


Gambar 29. Desain *Interface* Halaman *Dashboard* Admin

Ketika *user* masuk sebagai admin maka yang akan ditampilkan adalah halaman *dashboard* admin. Halaman ini terdiri dari menu *dashboard*, statistik, informasi, absensi, buku, penulis, siswa, member, peminjaman dan ubah *password*. Pada sisi kanan terdapat tabel yang digunakan untuk menampilkan data.

4) Halaman *Dashboard* Siswa

Desain *interface* halaman *dashboard* siswa sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 30 sebagai berikut:

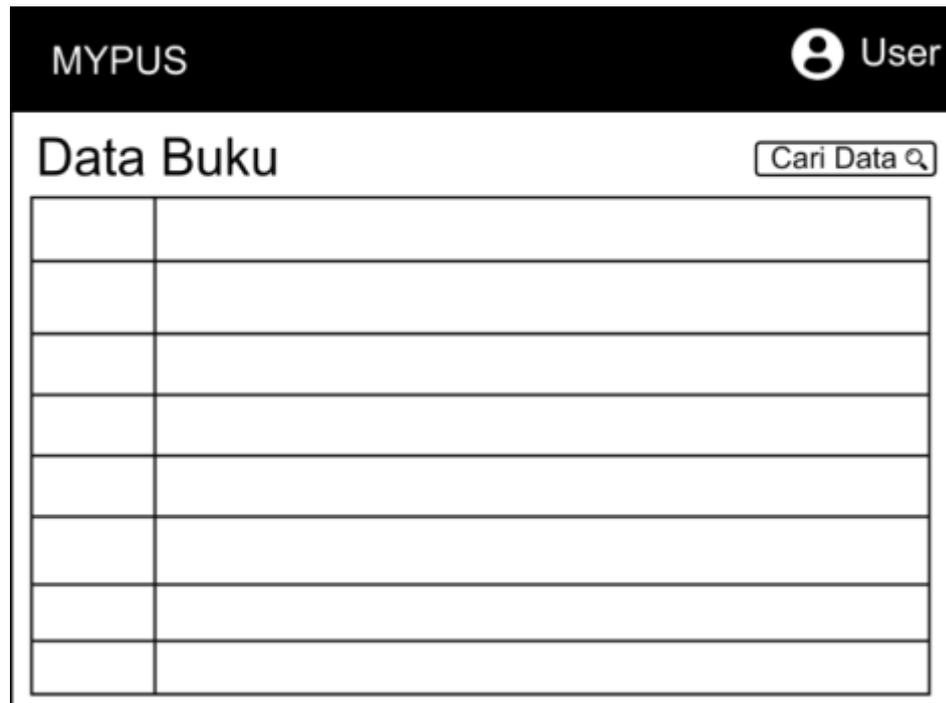


Gambar 30. Desain *Interface* Halaman *Dashboard* Siswa

Ketika *user* masuk sebagai siswa maka yang akan ditampilkan adalah halaman *dashboard* siswa. Halaman ini terdiri dari menu *dashboard*, informasi, buku, profil dan ubah *password*. Pada sisi kanan terdapat tabel yang digunakan untuk menampilkan data.

5) Halaman Tamu

Desain *interface* halaman tamu sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 31 sebagai berikut:



Gambar 31. Desain *Interface* Halaman Tamu

Ketika *user* masuk sebagai tamu maka yang akan ditampilkan adalah halaman tamu. Halaman ini digunakan untuk menampilkan data buku beserta menu pencarian.

C. Tahap Implementasi

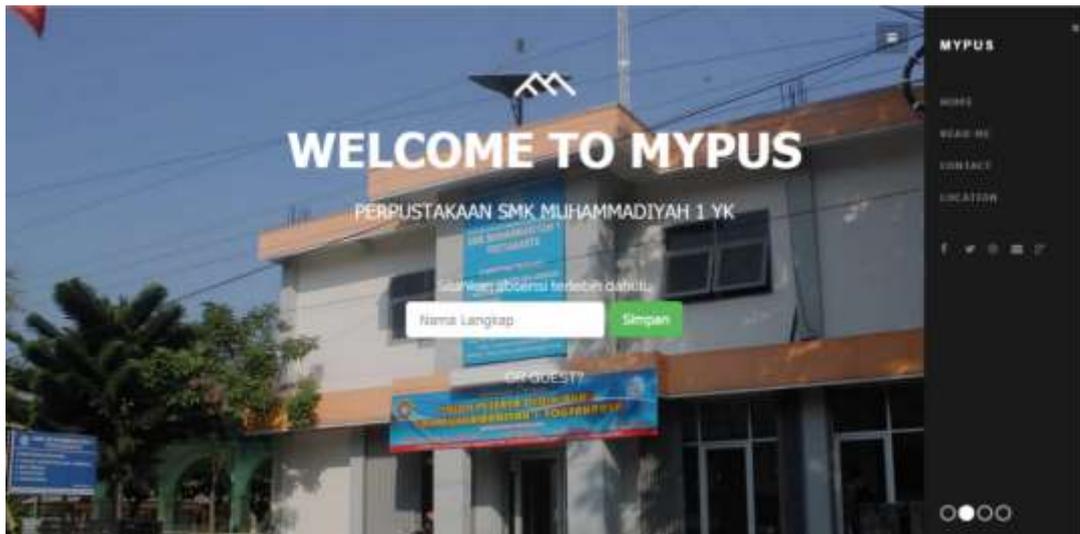
Tahap implementasi merupakan tahap pada proses pengembangan perangkat lunak setelah dilakukan analisis kebutuhan dan desain sistem. Desain yang telah dirancang kemudian di terapkan dalam kode bahasa pemrograman sehingga perangkat lunak dapat digunakan secara nyata dalam mengatasi masalah sesuai dengan fungsinya. Dalam tahap implementasi pengembangan sistem informasi perpustakaan ini dibagi menjadi dua, yaitu implementasi fungsi dan implementasi basis data.

1. Implementasi Fungsi

Implementasi sistem informasi perpustakaan ini menggunakan *framework laravel, bootstrap dan uikit*.

a. Halaman Utama

Implementasi halaman utama sistem informasi perpustakaan adalah seperti Gambar 32 sebagai berikut:



Gambar 32. Hasil implementasi Halaman Utama

Halaman utama adalah halaman yang pertama di akses ketika pengunjung mengunjungi sistem informasi perpustakaan sekolah.

b. Halaman Login

Implementasi halaman login sistem informasi perpustakaan adalah seperti Gambar 33 sebagai berikut:

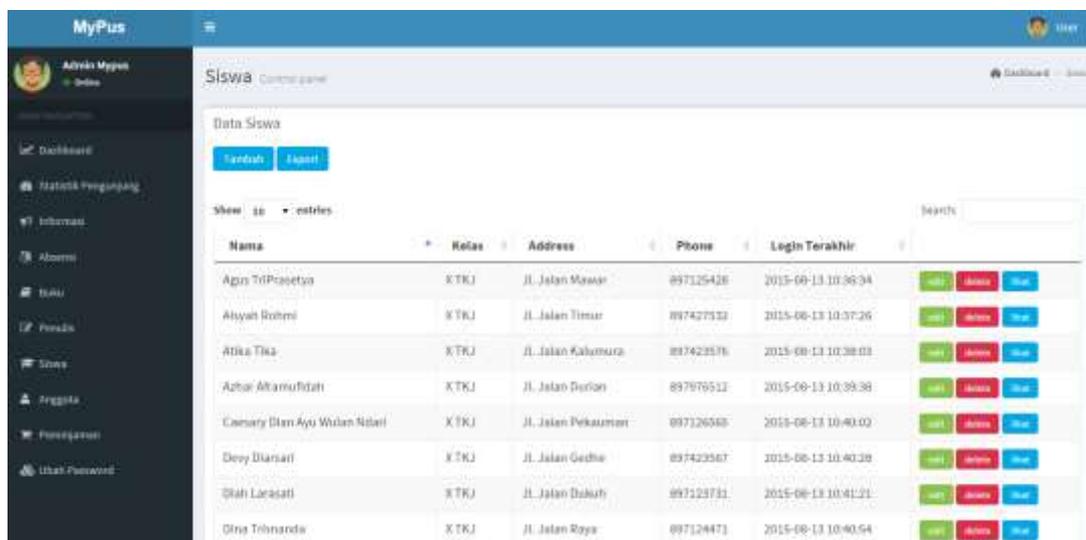


Gambar 33. Hasil implementasi Halaman Login

Halaman *login* pada sistem informasi perpustakaan sekolah yang digunakan oleh pengguna untuk dapat masuk ke dalam sistem.

c. Halaman *Dashboard Admin*

Implementasi halaman *dashboard* admin sistem informasi perpustakaan adalah seperti Gambar 34 sebagai berikut:



Gambar 34. Hasil implementasi *Dashboard Admin*

Halaman *dashboard* admin pada sistem informasi perpustakaan sekolah yang digunakan oleh admin untuk mengakses menu tertentu.

d. Halaman *Dashboard* Siswa

Implementasi halaman *dashboard* siswa sistem informasi perpustakaan adalah seperti Gambar 35 sebagai berikut:

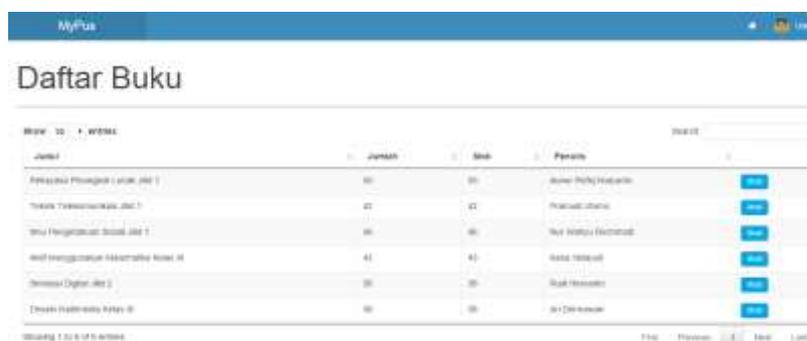


Gambar 35. Hasil implementasi *Dashboard* Siswa

Halaman *dashboard* siswa pada sistem informasi perpustakaan sekolah yang digunakan oleh siswa untuk mengakses menu tertentu.

e. Halaman Tamu

Implementasi halaman tamu sistem informasi perpustakaan adalah seperti Gambar 36 sebagai berikut:



Gambar 36. Hasil implementasi Tamu

Halaman tamu pada sistem informasi perpustakaan sekolah yang digunakan oleh pengguna untuk melihat dan mencari buku.

2. Implementasi Basis Data

Setelah basis data (*database*) dirancang dalam tahap desain, kemudian diimplementasikan menggunakan *database* MySQL.

a. Tabel *user*

Implementasi tabel *user* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 37 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|----|----------------------------|----------------|-----------------|------------|------|---------------------|----------------|
| 1 | <u>id</u> | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | <u>email</u> | varchar(255) | utf8_unicode_ci | | No | None | |
| 3 | <u>password</u> | varchar(255) | utf8_unicode_ci | | No | None | |
| 4 | <u>permissions</u> | text | utf8_unicode_ci | | Yes | NULL | |
| 5 | <u>activated</u> | tinyint(1) | | | No | 0 | |
| 6 | <u>activation_code</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 7 | <u>activated_at</u> | timestamp | | | Yes | NULL | |
| 8 | <u>last_login</u> | timestamp | | | Yes | NULL | |
| 9 | <u>persist_code</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 10 | <u>reset_password_code</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 11 | <u>first_name</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 12 | <u>last_name</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 13 | <u>full_name_user</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 14 | <u>address</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 15 | <u>phone</u> | int(11) | | | Yes | NULL | |
| 16 | <u>created_at</u> | timestamp | | | No | 0000-00-00 00:00:00 | |
| 17 | <u>updated_at</u> | timestamp | | | No | 0000-00-00 00:00:00 | |
| 18 | <u>jeniskelamin</u> | enum('L', 'P') | utf8_unicode_ci | | Yes | NULL | |
| 19 | <u>kelas</u> | varchar(255) | utf8_unicode_ci | | Yes | NULL | |
| 20 | <u>photo</u> | varchar(255) | utf8_unicode_ci | | No | None | |

Gambar 37. Implementasi Tabel *user*

Tabel *user* adalah tabel yang digunakan untuk menyimpan data siswa maupun admin sistem informasi perpustakaan sekolah.

b. Tabel *user_groups*

Implementasi tabel *user_groups* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 38 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|-----------------|---------|-----------|------------|------|---------|-------|
| 1 | <u>user id</u> | int(10) | | UNSIGNED | No | None | |
| 2 | <u>group id</u> | int(10) | | UNSIGNED | No | None | |

Gambar 38. Implementasi Tabel *user_groups*

Tabel *user_groups* adalah tabel yang digunakan untuk menyimpan data grup siswa maupun admin sistem informasi perpustakaan sekolah.

c. Tabel *authors*

Implementasi tabel *authors* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 39 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|-------------------|--------------|-----------------|------------|------|---------------------|----------------|
| 1 | id | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | name | varchar(255) | utf8_unicode_ci | | No | None | |
| 3 | created_at | timestamp | | | No | 0000-00-00 00:00:00 | |
| 4 | updated_at | timestamp | | | No | 0000-00-00 00:00:00 | |

Gambar 39. Implementasi Tabel *authors*

Tabel *authors* adalah tabel yang digunakan untuk menyimpan penulis buku sistem informasi perpustakaan sekolah.

d. Tabel *book*

Implementasi tabel *book* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 40 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|-------------------|--------------|-----------------|------------|------|---------------------|----------------|
| 1 | id | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | title | varchar(255) | utf8_unicode_ci | | No | None | |
| 3 | cover | varchar(255) | utf8_unicode_ci | | No | None | |
| 4 | author_id | int(10) | | UNSIGNED | No | None | |
| 5 | amount | int(10) | | UNSIGNED | No | None | |
| 6 | created_at | timestamp | | | No | 0000-00-00 00:00:00 | |
| 7 | updated_at | timestamp | | | No | 0000-00-00 00:00:00 | |

Gambar 40. Implementasi Tabel *book*

Tabel *book* adalah tabel yang digunakan untuk menyimpan data buku sistem informasi perpustakaan sekolah.

e. Tabel *book_user*

Implementasi tabel *book_user* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 41 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|------------|------------|-----------|------------|------|---------------------|----------------|
| 1 | <u>id</u> | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | book_id | int(10) | | UNSIGNED | No | None | |
| 3 | user_id | int(10) | | UNSIGNED | No | None | |
| 4 | returned | tinyint(1) | | | No | 0 | |
| 5 | created_at | timestamp | | | No | 0000-00-00 00:00:00 | |
| 6 | updated_at | timestamp | | | No | 0000-00-00 00:00:00 | |

Gambar 41. Implementasi Tabel *book_user*

Tabel *book_user* adalah tabel yang digunakan untuk menyimpan data peminjaman buku sistem informasi perpustakaan sekolah.

f. Tabel absensi

Implementasi tabel absensi dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 42 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|------------|--------------|-----------------|------------|------|---------------------|----------------|
| 1 | <u>id</u> | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | name | varchar(255) | utf8_unicode_ci | | No | None | |
| 3 | created_at | timestamp | | | No | 0000-00-00 00:00:00 | |
| 4 | updated_at | timestamp | | | No | 0000-00-00 00:00:00 | |

Gambar 42. Implementasi Tabel absensi

Tabel absensi adalah tabel yang digunakan untuk menyimpan data pengunjung sistem informasi perpustakaan sekolah.

g. Tabel *information*

Implementasi tabel *information* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 43 sebagai berikut:

| # | Name | Type | Collation | Attributes | Null | Default | Extra |
|---|-------------|--------------|-----------------|------------|------|---------------------|----------------|
| 1 | <u>id</u> | int(10) | | UNSIGNED | No | None | AUTO_INCREMENT |
| 2 | information | varchar(255) | utf8_unicode_ci | | No | None | |
| 3 | date | date | | | No | None | |
| 4 | created_at | timestamp | | | No | 0000-00-00 00:00:00 | |
| 5 | updated_at | timestamp | | | No | 0000-00-00 00:00:00 | |

Gambar 43. Implementasi Tabel *information*

Tabel *information* adalah tabel yang digunakan untuk menyimpan data informasi sistem informasi perpustakaan sekolah.

D. Tahap Pengujian Sistem

1. Pengujian Aspek *Functionality*

Berdasarkan hasil pengujian aspek *functionality* yang dilakukan oleh tiga ahli pemrograman dengan menggunakan *checklist* uji *functionality* maka didapatkan hasil berikut. Tabel 10 merupakan hasil uji *functionality* sistem informasi perpustakaan oleh ahli dalam pengembangan *software* berbasis *web*.

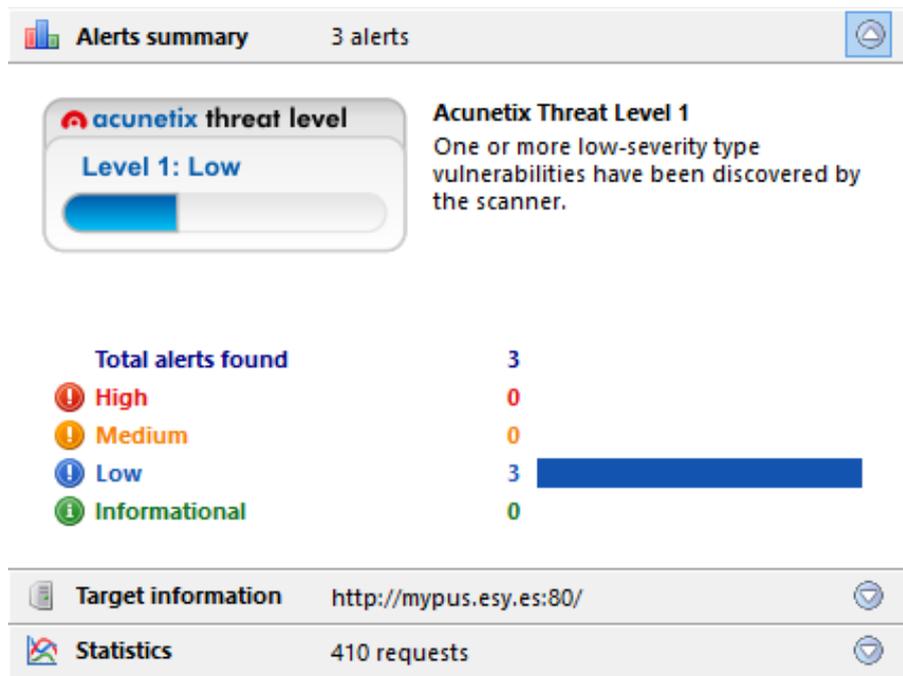
Tabel 10. Hasil Uji *Functionality*

| No. | Fungsi | Lolos | |
|-----------|--------------------------------|-------|-------|
| | | Ya | Tidak |
| I. | User Admin | | |
| A | Akun | | |
| 1. | Login sebagai admin | 3 | 0 |
| 2. | Mengubah password admin | 3 | 0 |
| 3. | Lupa password | 3 | 0 |
| 4. | Logout | 3 | 0 |
| B | Mengelola Data Siswa | | |
| 5. | Melihat daftar siswa | 3 | 0 |
| 6. | Menambah data siswa | 3 | 0 |
| 7. | Mengedit data siswa | 3 | 0 |
| 8. | Mengubah foto siswa | 3 | 0 |
| 9. | Menghapus data siswa | 3 | 0 |
| 10. | Mencari data siswa | 3 | 0 |
| 11. | Export data siswa | 3 | 0 |
| C | Mengelola Member | | |
| 12. | Mencari data member | 3 | 0 |
| 13. | Melihat data peminjaman member | 3 | 0 |
| 14. | Menghapus member | 3 | 0 |
| D | Mengelola Data Buku | | |
| 15. | Melihat daftar buku | 3 | 0 |
| 16. | Menambah data buku | 3 | 0 |
| 17. | Mengedit data buku | 3 | 0 |
| 18. | Mengubah cover buku | 3 | 0 |
| 19. | Menghapus data buku | 3 | 0 |
| 20. | Mencari data buku | 3 | 0 |
| 21. | Import data buku | 3 | 0 |
| 22. | Export data buku | 3 | 0 |
| E | Mengelola Data Absensi | | |
| 23. | Melihat daftar absensi | 3 | 0 |

Lanjutan Tabel 10. Hasil Uji *Functionality*

| No. | Fungsi | Lolos | |
|-------------|----------------------------------|-------|-------|
| | | Ya | Tidak |
| 24. | Menghapus data absensi | 3 | 0 |
| 25. | Mencari daftar absensi | 3 | 0 |
| 26. | Export data absensi | 3 | 0 |
| F | Mengelola Data Peminjaman | | |
| 27. | Melihat siswa yang meminjam | 3 | 0 |
| 28. | Mencari siswa yang meminjam | 3 | 0 |
| 29. | Melihat buku dipinjam | 3 | 0 |
| 30. | Mencari buku yang dipinjam | 3 | 0 |
| 31. | Filter berdasarkan tanggal | 3 | 0 |
| 32. | Laporan data peminjaman | 3 | 0 |
| G | Informasi Perpustakaan | | |
| 33. | Melihat informasi perpustakaan | 3 | 0 |
| 34. | Menambah informasi perpustakaan | 3 | 0 |
| 35. | Mengedit informasi perpustakaan | 3 | 0 |
| 36. | Menghapus informasi perpustakaan | 3 | 0 |
| 37. | Mencari informasi perpustakaan | 3 | 0 |
| H. | Data Statistik | | |
| 38. | Data statistik absensi | 3 | 0 |
| II. | User Siswa | | |
| A | Akun | | |
| 39. | Login sebagai siswa | 3 | 0 |
| 40. | Melihat data profil | 3 | 0 |
| 41. | Mengubah data profil | 3 | 0 |
| 42. | Mengubah password | 3 | 0 |
| 43. | Mengubah gambar profil | 3 | 0 |
| 44. | Lupa password | 3 | 0 |
| 45. | Logout | 3 | 0 |
| B | Peminjaman Buku | | |
| 46. | Melihat daftar buku | 3 | 0 |
| 47. | Mencari data buku | 3 | 0 |
| 48. | Meminjam buku | 3 | 0 |
| C | Pengembalian Buku | | |
| 49. | Melihat daftar buku | 3 | 0 |
| 50. | Mengembalikan buku | 3 | 0 |
| D | Informasi | | |
| 51. | Melihat informasi perpustakaan | 3 | 0 |
| E | Absensi | | |
| 52. | Absensi perpustakaan | 3 | 0 |
| III. | User Tamu | | |
| 53. | Melihat tentang perpustakaan | 3 | 0 |
| 54. | Melihat profil perpustakaan | 3 | 0 |
| 55. | Melihat Layanan Perpustakaan | 3 | 0 |
| 56. | Melihat daftar Buku | 3 | 0 |
| 57. | Pencarian Buku | 3 | 0 |

Sedangkan pengujian sub-karakteristik *security* dilakukan dengan menggunakan software *Acunetix Vulnerability Web Scanner* mendapatkan hasil Gambar 44 sebagai berikut:



Gambar 44. Hasil Pengujian *Security*

Dari pengujian aspek *security* dengan bantuan software *Acunetix Vulnerability Web Scanner* didapatkan bahwa sistem informasi perpustakaan memiliki *threat level 1 (low)*.

2. Pengujian Aspek *Efficiency*

Pengujian aspek *efficiency* menggunakan pengujian *online* menggunakan software berbasis *web* GTMetrix. Pengujian ini dilakukan dengan menguji beberapa halaman *web* sistem informasi perpustakaan dengan hak akses sebagai admin, siswa, dan tamu. Hasil dari pengujian aspek *efficiency* menggunakan software berbasis *web* GTMetrix adalah sebagai berikut:

1) Hasil Pengujian *Efficiency* Halaman Utama

Hasil pengujian *efficiency* halaman utama dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 45 sebagai berikut:



Gambar 45. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 28 dengan ukuran halaman 707KB. *Grade* yang didapatkan adalah A (99%) untuk pengujian Page Speed dan B (89%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.20 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

2) Hasil Pengujian *Efficiency* Halaman Login

Hasil pengujian *efficiency* halaman login dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 46 sebagai berikut:

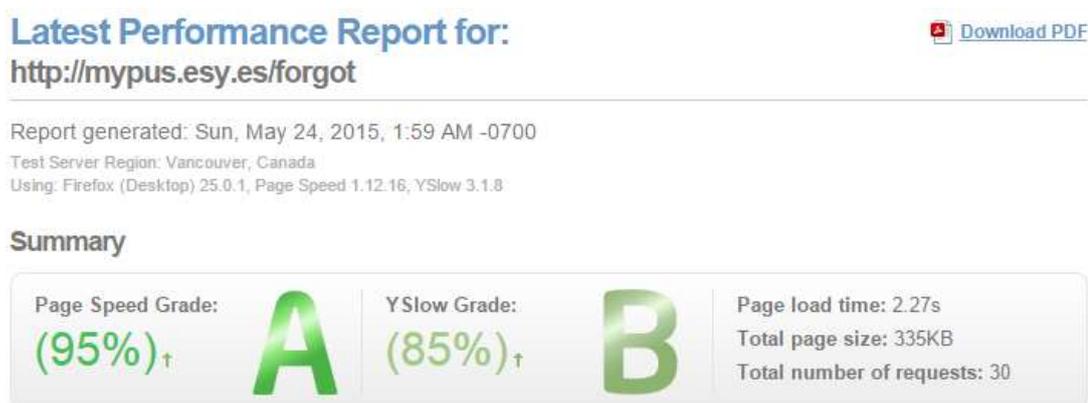


Gambar 46. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 23 dengan ukuran halaman 300KB. *Grade* yang didapatkan adalah A (99%) untuk pengujian Page Speed dan A (91%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.04 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

3) Hasil Pengujian *Efficiency* Halaman Lupa *Password*

Hasil pengujian *efficiency* halaman lupa *password* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 47 sebagai berikut:



Gambar 47. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 30 dengan ukuran halaman 335KB. *Grade* yang didapatkan adalah A (95%) untuk pengujian Page Speed dan B (85%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.27 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

4) Hasil Pengujian *Efficiency* Halaman *Dashboard*

Hasil pengujian *efficiency* halaman *dashboard* dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 48 sebagai berikut:



Gambar 48. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 25 dengan ukuran halaman 303KB. *Grade* yang didapatkan adalah A (97%) untuk pengujian Page Speed dan A (91%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.15 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

Hasil pengujian *efficiency* seluruhnya dapat dilihat pada Tabel 11 berikut:

Tabel 11. Hasil Pengujian *Efficiency*

| No. | Halaman web | Page Load Time (detik) | Page Size (Kb) | Page Speed Grade | Yslow Grade |
|-----|------------------|------------------------|----------------|------------------|-------------|
| 1 | Utama | 2,20 | 707 | A (99%) | B (89%) |
| 2 | Login | 2,04 | 300 | A (99%) | A (91%) |
| 3 | Ubah Password | 2,13 | 312 | A (97%) | A (91%) |
| 4 | <i>Dashboard</i> | 2,15 | 303 | A (97%) | A (91%) |
| 5 | Daftar Siswa | 2,43 | 318 | A (97%) | A (90%) |
| 6 | Daftar Member | 2,21 | 479 | A (97%) | A (91%) |
| 7 | Daftar Buku | 2,13 | 300 | A (97%) | A (90%) |

Lanjutan Tabel 11. Hasil Pengujian *Efficiency*

| No. | Halaman web | Page Load Time (detik) | Page Size (Kb) | Page Speed Grade | Yslow Grade |
|-----------|----------------------|------------------------|----------------|------------------|-------------|
| 8 | Data Peminjaman | 2,18 | 431 | A (97%) | A (91%) |
| 9 | Statistik Absensi | 2,31 | 300 | A (97%) | A (90%) |
| 10 | Informasi | 2,09 | 242 | A (97%) | A (91%) |
| 11 | Absensi | 2,24 | 362 | A (97%) | A (91%) |
| 12 | Tentang Perpustakaan | 2,10 | 440 | A (91%) | A (91%) |
| 13 | Profil | 2,10 | 440 | A (91%) | A (91%) |
| 14 | Layanan | 2,25 | 440 | A (91%) | A (91%) |
| 15 | Guest | 2,43 | 376 | A (93%) | B (89%) |
| 16 | Dashboard Siswa | 2,43 | 318 | A (97%) | A (90%) |
| 17 | Lupa Password | 2,27 | 335 | A (95%) | B (85%) |
| Rata-rata | | 2,21 | 376 | A (96%) | A (90%) |

Hasil pengujian *efficiency* pada Tabel menunjukkan bahwa rata-rata waktu *page load* halaman web 2,21 detik. Selain itu masing-masing *page load time* tidak ada yang melebihi 10 detik (sesuai standar *efficiency* menurut Nielsen).

3. Pengujian Aspek *Reliability*

Pengujian pada aspek *reliability* dilakukan dengan menggunakan *software* WAPT versi 8.4. *Software* ini digunakan untuk melakukan mekanisme *stress testing* pada sebuah *website* dan memberikan hasil pengukuran *reliability*. Hasil pengujian sistem informasi perpustakaan dengan alamat <http://mypus.esy.es> menggunakan WAPT dengan jumlah 20 *user* selama 1 jam (60 menit) ditunjukkan dalam Gambar 49 sebagai berikut:

| Summary | | | | | | | | |
|----------|---------------------|-----------------|------------------|--------------|-----------------|-------------|-------------------|-----------------------|
| Profile | Successful sessions | Failed sessions | Successful pages | Failed pages | Successful hits | Failed hits | Total KBytes sent | Total KBytes received |
| Profile1 | 13 | 0 | 125 | 0 | 714 | 0 | 221 | 13052 |

Gambar 49. Hasil Pengujian *Reliability* WAPT

Pada hasil pengujian dapat dilihat bahwa terdapat hasil pengujian *stress testing* yang berupa keberhasilan *sessions*, *pages*, dan *hits*. Dari hasil tersebut terlihat

bahwa hasil *successful sessions* 13 dan 0 *failed sessions*, *successful pages* 125 dan 0 *failed pages*, serta *successful hits* 714 dan 0 *failed hits*.

4. Pengujian Aspek *Usability*

Hasil pengujian *usability* terhadap 30 responden dapat dilihat pada Tabel 12.

Tabel 12. Hasil Pengujian *Usability*

| Pertanyaan | SS | S | RG | TS | STS |
|--------------|------------|------------|------------|-----------|----------|
| 1 | 10 | 14 | 6 | 0 | 0 |
| 2 | 7 | 14 | 9 | 0 | 0 |
| 3 | 14 | 10 | 6 | 0 | 0 |
| 4 | 9 | 16 | 5 | 0 | 0 |
| 5 | 8 | 17 | 5 | 0 | 0 |
| 6 | 9 | 16 | 5 | 0 | 0 |
| 7 | 8 | 14 | 8 | 0 | 0 |
| 8 | 3 | 6 | 20 | 1 | 0 |
| 9 | 6 | 15 | 8 | 1 | 0 |
| 10 | 8 | 16 | 5 | 1 | 0 |
| 11 | 6 | 18 | 5 | 1 | 0 |
| 12 | 8 | 16 | 5 | 1 | 0 |
| 13 | 7 | 12 | 9 | 2 | 0 |
| 14 | 8 | 15 | 5 | 2 | 0 |
| 15 | 8 | 12 | 9 | 1 | 0 |
| 16 | 2 | 8 | 17 | 3 | 0 |
| 17 | 5 | 11 | 14 | 0 | 0 |
| 18 | 6 | 10 | 13 | 1 | 0 |
| 19 | 6 | 6 | 15 | 3 | 0 |
| 20 | 9 | 13 | 8 | 0 | 0 |
| 21 | 10 | 10 | 10 | 0 | 0 |
| 22 | 8 | 15 | 7 | 0 | 0 |
| 23 | 7 | 13 | 10 | 0 | 0 |
| 24 | 5 | 8 | 16 | 1 | 0 |
| 25 | 5 | 14 | 11 | 0 | 0 |
| 26 | 2 | 18 | 10 | 0 | 0 |
| 27 | 4 | 8 | 17 | 1 | 0 |
| 28 | 11 | 14 | 5 | 0 | 0 |
| 29 | 12 | 12 | 6 | 0 | 0 |
| 30 | 8 | 14 | 8 | 0 | 0 |
| Total | 219 | 385 | 277 | 19 | 0 |

Hasil pengujian dari tabel pengujian *usability* dengan menggunakan kuesioner baku dari Arnold M.Lund menggunakan *software* IBM SPSS versi 22 dapat dilihat pada Gambar 50 sebagai berikut:

Scale: ALL VARIABLES

Case Processing Summary

| | | N | % |
|-------|-----------------------|----|-------|
| Cases | Valid | 30 | 100,0 |
| | Excluded ^a | 0 | ,0 |
| | Total | 30 | 100,0 |

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

Reliability Statistics

| Cronbach's Alpha | N of Items |
|------------------|------------|
| ,958 | 30 |

Gambar 50. Hasil Koefisien *Cronbach's Alpha*

E. Pembahasan Hasil Penelitian

1. Analisis Pengujian Aspek *Functionality*

Dari hasil pengujian aspek *functionality* menggunakan *test case* didapatkan hasil seperti dalam tabel. Hasil pengujian *functionality* pada tabel dihitung dengan menggunakan rumus ISO (2003) pada aspek *functionality* sebagai berikut:

$$X = 1 - \frac{A}{B}$$

$$X = 1 - \frac{0}{57}$$

$$X = 1 - 0$$

$$X = 1$$

ISO 9126 pada aspek *functionality* dapat dikatakan bahwa *software* dikatakan semakin baik jika hasil perhitungannya mendekati 1. Karena hasil pengujian memiliki nilai maksimal yaitu 1, maka *software* sudah memenuhi aspek *functionality*. Sedangkan pada aspek *security* menunjukkan bahwa tidak ada celah serangan *SQL Injection* dan *XSS*. Pada pengujian menggunakan *software acunetix* hasil keamanan yang diperoleh website adalah *threat level 1 (low)*.

2. Analisis Pengujian Aspek *Efficiency*

Hasil pengujian *efficiency* didapatkan bahwa rata-rata waktu *page load* halaman *web* sebesar 2,21 detik. *Web* dikatakan baik apabila waktu *load*-nya kurang dari 10 detik menurut Nielsen (2010) sehingga *software* telah memenuhi standar pada aspek *efficiency*.

3. Analisis Pengujian Aspek *Reliability*

Dari hasil pengujian menggunakan WAPT, presentase keberhasilan dapat dilihat pada Tabel 13.

Tabel 13. Hasil Pengujian *Reliability*

| No. | Komponen | Berhasil | Gagal | Total | Persentase |
|-----|-----------------|----------|-------|-------|------------|
| 1 | <i>Sessions</i> | 13 | 0 | 13 | 100% |
| 2 | <i>Pages</i> | 125 | 0 | 125 | 100% |
| 3 | <i>Hits</i> | 714 | 0 | 714 | 100% |

Dari Tabel dapat disimpulkan bahwa presentase keberhasilan seluruh komponen pada pengujian *reliability* sebesar 100%. *Software* dapat dikatakan telah memenuhi standar pada aspek *reliability* karena persentase minimal yang harus dicapai berdasarkan standar Telcordia adalah 95%.

4. Analisis Pengujian Aspek Usability

Dari hasil jawaban responden pada kuesioner USE, jumlah jawaban dikalikan dengan skor agar didapatkan skor total dari hasil jawaban responden. Perhitungan tersebut dapat dilihat pada Tabel 14.

Tabel 14. Perolehan Skor Kuesioner

| | Jumlah | Skor | Jumlah x Skor |
|--------------------|--------|------|---------------|
| SS | 219 | 5 | 1095 |
| ST | 385 | 4 | 1540 |
| RG | 277 | 3 | 831 |
| TS | 19 | 2 | 38 |
| STS | 0 | 1 | 0 |
| Skor Total: | | | 3504 |

Jumlah skor ideal untuk seluruh item = $5 \times 30 \times 30 = 4500$. Jadi tingkat persetujuannya berdasarkan data tersebut = $(3504 : 4500) \times 100\% = 77\%$ dari yang diharapkan. Secara kontium dapat dilihat pada Gambar 51.

Berdasarkan data yang diperoleh dari 30 responden maka skor 3504 terletak pada daerah Setuju (S). Hasil data persetujuan responden ditunjukkan pada Gambar 51 sebagai berikut:



Gambar 51. Tingkat Persetujuan Responden

Perhitungan *cronbach's alpha* dengan menggunakan *tool* IBM SPSS versi 22 didapatkan nilai 0,958. Jika dibandingkan dengan tabel (*internal consistency cronbach's alpha*) maka hasil tersebut masuk dalam kategori *excellent*.

BAB V SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat disimpulkan bahwa:

1. Sistem informasi perpustakaan berbasis *web* di SMK Muhammadiyah 1 Yogyakarta dalam pengembangannya menggunakan *framework* Laravel. Proses pengembangan *software* menggunakan model *waterfall* yang terdiri dari empat tahap, (1) Analisis kebutuhan; (2) Desain; (3) Implementasi; dan (4) Pengujian. *Software* ini memiliki fitur sebagai sistem informasi perpustakaan seperti pengelolaan peminjaman dan pengelolaan buku, informasi perpustakaan dan pembuatan laporan.
2. Sistem informasi perpustakaan berbasis *web* di SMK Muhammadiyah 1 Yogyakarta dalam pengembangannya diuji dengan standar ISO 9126 yang diidentifikasi dengan menggunakan aspek dari WebQEM meliputi 4 aspek yaitu *functionality*, *efficiency*, *reliability* dan *usability*. Hasil pengujian pada aspek *functionality*, diperoleh nilai 1 (baik) dan tingkat keamanan untuk serangan *SQL Injection* dan *XSS (Cross Site Scripting)* diperoleh level sangat aman (low). Pengujian *efficiency*, diperoleh rata-rata *load time* 2,21 detik (diterima). Pengujian *reliability* diperoleh nilai *reliability* sebesar 100% (baik). Pengujian *usability* diperoleh tingkat persetujuan pengguna sebesar 77% (setuju) dengan nilai *alpha cronbach* sebesar 0,958 (*excellent*). Berdasarkan hasil tersebut maka sistem informasi perpustakaan yang dikembangkan memenuhi standar kualitas *software* ISO 9126.

B. Keterbatasan Produk

Software yang dikembangkan masih memiliki fitur yang terbatas. Penambahan fitur dibutuhkan untuk mendapatkan aplikasi yang lebih kompleks dan memberikan solusi pada setiap masalah yang dihadapi. Beberapa contoh fitur dalam pengembangan kedepannya seperti penambahan fitur *barcode* untuk membaca identitas buku dan juga fitur baca buku pada komputer secara langsung agar *user* dapat mereview buku yang ada di perpustakaan terlebih dahulu.

C. Pengembangan Produk Lebih Lanjut

Pengembangan sistem informasi perpustakaan selanjutnya dapat melengkapi fitur yang lebih lengkap sehingga sistem ini bisa memberikan manfaat yang lebih dari sistem yang telah dikembangkan.

D. Saran

Berdasarkan berbagai keterbatasan yang dimiliki penulis baik dari segi waktu maupun pikiran, maka penulis menyarankan untuk pengembangan penelitian yang akan datang sebagai berikut :

1. Perlu adanya penambahan fitur lain berupa pembuatan kode buku otomatis.
2. Penambahan fitur baca buku pada komputer secara langsung agar *user* dapat mereview terlebih dahulu buku yang ada di perpustakaan.
3. Teknik pengujian kualitas *software* menggunakan *tools* yang lebih beragam.

DAFTAR PUSTAKA

- Agung Yulianto, Ardhian. (2009). *Analisis Dan Desain Sistem Informasi*. Bandung: Politeknik Telkom.
- Agus Mulyanto. (2009). *Sistem Informasi Konsep dan Aplikasi*. Yogyakarta: Pustaka Pelajar.
- A.S, Rosa dan Shalahuddin, M. (2011). *Modul Pembelajaran: Rekayasa Perangkat Lunak*. Bandung: Modula.
- A.S, Rosa dan Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak*. Bandung: Informatika.
- Asthana, A., & Olivieri, J. (2009). *Quantifying software reliability and readiness. Communications Quality and Reliability, 2009. CQR 2009. IEEE International Workshop Technical Committee on* (pp. 1–6). IEEE.
- Al-Bahra Bin Ladjamudin. (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Berander, Patrik.dkk. (2005). *Software quality attributes and trade-offs*. [s.l]:Blekinge Institute of Technology.
- Chander, S., & Kush, A. (2012). *Vulnerability in Web Page and Web Sites. International Journal of Advanced Research in IT and Engineering*.
- Chua, B., & Dyson, L. (2004). *Applying the ISO 9126 model to the evaluation of an e-learning system. In Beyond the Comfort Zone in Proceedings of the 21st ASCILITE Conference*, (pp. 184-190). Perth.
- Clarke, J. (2009). *SQL Injection Attacks and Defense*. US: Elsevier.
- Gliem, J. A., & Gliem, R. R. (2003). *Calculating, Interpreting, and Reporting Cronbach's Alpha Reliability Coefficient for Likert-Type Scales. 2003 Midwest Research to Practice Conference in Adult, Continuing, and Community Education*, (p.82).
- Harmawan. (2009). *Evaluasi Sistem Otomatisasi Perpustakaan Sekolah*. Medan: USU Institutional Hlm. 1-6.
- Indra Permana, Punky. (2013). *Sistem Informasi Perpustakaan Sekolah Berbasis Web dengan Framework Codeigniter dan PostgreSQL di SMA Negeri 1 Ngangklik*. Yogyakarta: UNY.
- ISO/IEC. (2002). *Software Engineering Product Quality - Part 2 – External Metric*. Canada: International Technical Report.

- ISO/IEC. (2010). *The Standart of Reference*. Diakses dari <http://www.cse.dcu.ie/essiscope/sm2/9126ref.html>. Pada tanggal 27 Maret 2015, Jam 23:03 WIB.
- Jogiyanto HM. (2008). *Sistem Teknologi Informasi*. Yogyakarta: Andi Offset.
- Kristanto, Andri. (2004). *Rekayasa Perangkat Lunak (Konsep Dasar)*. Yogyakarta: Gava Media.
- Lasa. (2007). *Manajemen Perpustakaan Sekolah*. Yogyakarta: Pinus.
- Lund, A. (2001). *Measuring usability with the USE questionnaire*. Diakses dari <http://garyperlman.com/quest/quest.cgi?form=USE>. Pada tanggal 17 Maret 2015 jam 13.57 WIB.
- McCool, Shawn. (2012). *Laravel Starter*. Birmingham: Packt Publishing Ltd.
- Nielsen, J. (2006). *Quantitative studies: How many users to test?*. Diakses dari <http://www.nngroup.com/articles/quantitative-studies-how-many-users/>. Pada tanggal 27 Maret 2015, Jam 23:03 WIB.
- Nielsen, J. (2010). *Website Respose Time*. Diakses dari <http://www.nngroup.com/articles/website-response-times/>. Pada tanggal 27 Maret 2015, Jam 23:03 WIB.
- Nielsen, J. (2012). *Usability 101: Introduction to Usability*. Diakses dari <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>. Pada tanggal 29 Maret 2015, Jam 21:13 WIB.
- Nidhra, Srinivas & Jagruthi Dondeti. (2012). *Black box and White box Testing Techniques – A Literature Review. Internasional Journal of Embedded System and Applications (IJESA) Vol.2, No.2*.
- Nugroho, Adi. (2005). *Analisis Dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*. Bandung: Informatika.
- Nugroho, Adi. (2009). *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Yogyakarta: Andi Offset.
- Nugroho, Adi. (2010). *Rekayasa Perangkat Lunak Berorientasi Objek Dengan Metode USDP*. Yoyakarta: Andi.
- Nugroho, Eddy Prasetyo, dkk. (2009). *Rekayasa Perangkat Lunak*. Bandung: Politeknik Telkom.
- NS, Sutarno. (2006). *Perpustakaan dan Masyarakat*. Jakarta: Sagung Seto.

- Official Website WAPT. (2014). WAPT. Diakses dari <http://www.loadtestingtool.com/>. Pada tanggal 14 April 2015, Jam 12:21 WIB.
- Olsina, L., Godoy, D., & Lafuente, G.J. (1998). *Specifying Quality Characteristics and Attributes for Websites*.
- Olsina, L., Sassano, R., & Mich, L. (2008). *Specifying Quality Requirements for the Web 2.0 Applications. Workshop on Web-Oriented Software Technologies (IWWOST)*.
- Padayachee, I., Kotze, P., & van Der Merwe, A. (2010). *ISO 9126 external systems quality characteristics, sub-characteristics and domain specific criteria for evaluating e-Learning systems*. The Southern African Computer Lecturers' Association, University of Pretoria, South Africa.
- Pradhan, D. (2013). WAPT; A Load Testing Tool That Delivers! [Review]. Diakses dari <http://www.softwaretestingtricks.com/2013/02/best-load-stress-performance-testing-tool-WAPT.html>. Pada tanggal 4 April 2015, Jam 12:21 WIB.
- Pressman, R.S. (2010). *Software Engineering: A Practitioner's Approach*. New York: McGraw-Hill Companies, Inc.
- Saleh. (2006). *Penyelenggaraan Perpustakaan Sekolah*. Jakarta: PT. Hidakarya Agung.
- Sesar Ria, Anita. (2014). *Symfony Fullstack PHP Framework*. Cirebon: Asfa Solution.
- Sekar Asih, Nur. (2013). *Studi Deskriptif analisa usability sistem informasi layanan peminjaman mandiri perpustakaan Universitas Negeri Malang dari sudut pengguna*. Surabaya: UNAIR.
- Simarmata. Janner. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta: Andi Offset.
- Singh, T. (2014). *Performance testing of any website "GTmetrix Tool"*. Diakses dari <http://www.oodlestechnologies.com/blogs/Performance-testing-of-any-website-%22GTmetrix-Tool%22>. Pada tanggal 2 April 2015 jam 11.03 WIB.
- Siregar, Beiling. (2007). *Pembinaan Koleksi Perpustakaan dan Pengetahuan Literatur*. Medan: Pembinaan Perpustakaan Sumatera Utara.
- Software, Lutfian. (2009). *Product Feature: Sistem Informasi Perpustakaan (SIP)*. Diakses dari http://blog.ub.ac.id/component/130502/docman/doc_download/5-sistem-informasi-perpustakaan.html blog Universitas Brawijaya: Jawa Timur pada tanggal 12 Maret 2015, Jam 09.30 WIB.

- Stalinbabu, R., & Chellamal, P. (2014). *Survey on Detecting SQL Injection and Cross Site Scripting in Multitier Web Applications. International Journal of Advanced Research in Computer Science and Technology.*
- Sugiyono. (2012). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D.* Bandung: CV. Alfabeta.
- Sulastri. (2013). *Pengembangan Sistem Informasi Perpustakaan pada Sekolah Menengah Pertama (SMP) Negeri 4 Karangtengah.* Surakarta: UNSA.
- Suman, Aparajita. (2014). *From Knowledge Abstraction to Management: Using Ranganathan's Faceted Schema to Develop Conceptual Frameworks for Digital Libraries.* Amsterdam [u.a.]: Chandos Publishing (Oxford) Limited.
- Surguy, Maksim. (2014). *Laravel – My First Framework.* Leanpub.
- Vieira, M., Antunes, N., & Madeira, H. (2009). *Using web security scanners to detect vulnerabilities in web services.* (pp. 566-571). IEEE.
- Wallace, S., & Chen Yu, H. (2009). *The Effect of Culture on Usability: Comparing the Perceptions and Performance of Taiwanese and North American MP3 Player Users. Journal Of Usability Studies Vol 4 Issue 3.*

LAMPIRAN

Lampiran 1. Surat – Surat Perijinan

1) Lampiran Surat Permohonan Izin Penelitian dari Fakultas



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS NEGERI YOGYAKARTA
FAKULTAS TEKNIK



Alamat : Kampus Karangmalang, Yogyakarta, 55281

Telp. (0274) 586168 ps.w. 276,289,292 (0274) 586734 Fax. (0274) 586734

Certificate No. QSC 00592

website : <http://ft.uny.ac.id> e-mail: ft@uny.ac.id ; teknik@uny.ac.id

Nomor : 1205/H34/PL/2015

18 Mei 2015

Lamp. :

Hal : Ijin Penelitian

Yth.

- 1 . Gubernur DIY c.q. Biro Administrasi Pembangunan Setda DIY
- 2 . Gubernur Provinsi DIY c.q. Ka. Bappeda Provinsi DIY
- 3 . Walikota Kota Yogyakarta c.q. Kepala Badan Pelayanan Terpadu Kota Yogyakarta
- 4 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Provinsi DIY
- 5 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Kota Yogyakarta
- 6 . Kepala SMK Muhammadiyah 1 Yogyakarta
- 7 . Pengurus Daerah Muhammadiyah (PDM) Kota Yogyakarta

Dalam rangka pelaksanaan Tugas Akhir Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul Pengembangan Sistem Informasi Perpustakaan Berbasis Web di SMK Muhammadiyah 1 Yogyakarta, bagi Mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:

| No. | Nama | NIM | Jurusan | Lokasi |
|-----|---------------------|-------------|-------------------------------|-------------------------------|
| 1 | Husin Nanda Perwira | 11520241039 | Pend. Teknik Informatika - S1 | SMK Muhammadiyah 1 Yogyakarta |

Dosen Pembimbing/Dosen Pengampu :

Nama : Handaru Jati, S.T. M.M., M.T.Ph.D.

NIP : 19740511 199903 1 002

Adapun pelaksanaan penelitian dilakukan mulai Tanggal 26 Mei 2015 s/d selesai.

Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.



Wakil Dekan I

Dr. Sunaryo Soenarto

NIP. 19580630 198601 1 001

Tembusan :
Ketua Jurusan

2) Lampiran Surat Permohonan Izin Penelitian PDM


**MAJELIS PENDIDIKAN DASAR DAN MENENGAH
PIMPINAN DAERAH MUHAMMADIYAH KOTA YOGYAKARTA**
Jalan Sultan Agung 14, Telepon (0274)375917, Faks. (0274) 411947, Yogyakarta 55151
e-mail: dikdasmenpdm_yk@yahoo.com

IZIN PENELITIAN/SKRIPSI/OBSERVASI
No. : 342/REK/III.4/F/2015

Setelah membaca surat dari : **Fakultas Teknik Universitas Negeri Yogyakarta.**
No. : 1205/H34/PL/2015 Tgl. : 18 Mei 2015
Perihal : **Surat Izin Penelitian**

dan berdasar Putusan Sidang Majelis Dikdasmen PDM Kota Yogyakarta, hari **Kamis tanggal 03 Sya'ban 1436 H**, bertepatan tanggal **21 Mei 2015 M** yang salah satu agenda sidangnya membahas pemberian penelitian/praktek kerja/observasi, maka dengan ini kami memberikan izin kepada:

Nama Terang : **HUSIN NANDA PERWIRA** NIM. **11520241039**
Pekerjaan : **Mahasiswa pada prodi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta**
alamat **Karangmalang Yogyakarta**
Pembimbing : **Handaru Jati, S.T., M.M., Ph.D**

untuk melakukan **observasi/penelitian/pengumpulan data** dalam rangka penyusunan Skripsi :

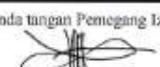
Tentang : **PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA.**

Lokasi : **SMK Muhammadiyah 1 Yogyakarta.**

dengan ketentuan sebagai berikut:

1. Menyebarkan tembusan surat ini kepada pejabat yang dituju.
2. Wajib menjaga tata tertib dan mematuhi ketentuan-ketentuan yang berlaku di sekolah/tersebut.
3. Wajib **memberi laporan hasil penelitian/praktek kerja/observasi dalam bentuk CD** kepada Majelis Pendidikan Dasar dan Menengah Pimpinan Daerah Muhammadiyah Kota Yogyakarta.
4. Izin ini tidak disahgunakan untuk tujuan tertentu yang dapat mengganggu kestabilan Penyelidikan dan banya diperlukan untuk keperluan ilmiah.
5. Surat izin ini dapat diajukan kembali untuk mendapat perpanjangan bila diperlukan.
6. Surat izin ini dapat dibatalkan sewaktu-sesaktu bila tidak dipatuhi ketentuan-ketentuan tersebut di atas.

MASA BERLAKU 2 (DUA) BULAN :
22-05-2015 sampai dengan 22-10-2015

Tanda tangan Pemegang Izin,

Husin Nanda Perwira

Yogyakarta, 22 Mei 2015

Ketua,  Sekretaris, 

Tembusan:
1. PDM Kota Yogyakarta.
2. Dekan FT UNY
3. Kepala SMK Muh. 1 YK.


Drs. H. Atis Thobirin, M.Si NBM. 670.219
Drs. H. Ibnu Marwanta NBM. 551.522

3) Lampiran Lembar Persetujuan Dilaksanakan Penelitian

PENGESAHAN
PROPOSAL SKRIPSI

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS
WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA

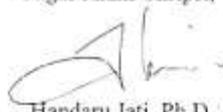
Disusun oleh :
Husin Nanda Perwira
NIM. 11520241039

Yogyakarta, 23 April 2015

Mengetahui,

Ketua Jurusan
Pendidikan Teknik Elektronika,

Muhammad Munir, M.Pd
NIP. 19630512 198901 1 001

Dosen Pembimbing
Tugas Akhir Skripsi,

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

Dekan Fakultas Teknik
Universitas Negeri Yogyakarta,


Dr. Moch. Bruri Triyono
NIP. 19560216 198603 1 003

Lampiran 2. Desain Use Case Diagram

1) Definisi *Use Case Diagram*

Tabel 1. Definisi *Use Case Diagram*

| No. | Use Case | Deskripsi |
|-------------------|---------------------------|---|
| A. | | |
| User Admin | | |
| 1 | Mengelola Akun | Merupakan proses untuk pengelolaan akun admin yang bertugas sebagai pengelola <i>web</i> . Mengelola akun merupakan generalisasi dari proses <i>login</i> , <i>logout</i> , mengubah <i>username</i> dan <i>password</i> , <i>reset</i> akun, dan memeriksa status <i>login</i> |
| 2 | Mengelola Siswa | Merupakan proses untuk pengelolaan siswa pada sistem. Mengelola siswa merupakan generalisasi dari lihat siswa, tambah siswa, edit siswa, hapus siswa, dan export siswa. |
| 3 | Mengelola Buku | Merupakan proses untuk pengelolaan buku perpustakaan pada sistem. Mengelola buku merupakan generalisasi dari lihat buku, tambah buku, edit buku, hapus buku, import dan export buku. |
| 4 | Mengelola Penulis | Merupakan proses untuk pengelolaan penulis buku perpustakaan pada sistem. Mengelola penulis merupakan generalisasi dari lihat penulis, tambah penulis, edit penulis, hapus penulis, import dan export penulis. |
| 5 | Mengelola Buku Peminjaman | Merupakan proses untuk pengelolaan buku yang dipinjam oleh siswa. Mengelola buku yang dipinjam merupakan generalisasi dari lihat siswa yang meminjam, lihat buku yang dipinjam, dan pengembalian buku. |
| 6 | Mengelola Absensi | Merupakan proses untuk pengelolaan absensi pengunjung perpustakaan. Mengelola absensi merupakan generalisasi dari lihat siswa yang berkunjung, hapus data absensi, dan export absensi. |
| 7 | Mengelola Informasi | Merupakan proses untuk pengelolaan informasi perpustakaan. Mengelola informasi merupakan generalisasi dari lihat informasi, tambah informasi, edit informasi, hapus informasi, import dan export informasi. |
| B. | | |
| User Siswa | | |
| 1 | Mengelola Profil | Merupakan proses untuk pengelolaan profil siswa. Mengelola informasi merupakan generalisasi dari lihat profil dan perbarui profil. |
| 2 | Mengelola Peminjaman | Merupakan proses untuk pengelolaan peminjaman dan pengembalian buku. Mengelola peminjaman merupakan generalisasi dari lihat daftar buku, peminjaman buku dan pengembalian buku. |

Lanjutan Tabel 1.

| No. | Use Case | Deskripsi |
|---------------------|-------------------|--|
| 3 | Melihat Informasi | Merupakan proses untuk melihat informasi perpustakaan. Melihat informasi merupakan generalisasi dari lihat informasi perpustakaan. |
| C. User Tamu | | |
| 1 | Melihat Buku | Merupakan proses untuk melihat daftar buku perpustakaan. Melihat buku merupakan generalisasi dari lihat daftar buku perpustakaan. |

2) Skenario *Use Case*

a) Nama *Use Case* : *login*

Tabel 2. Skenario *Use Case Login*

| Aksi aktor | Reaksi sistem |
|--|---|
| Skenario normal | |
| 1. Memasukkan <i>username</i> dan <i>password</i> | |
| | 2. Mengecek valid tidaknya data masukan |
| | 3. Masuk ke halaman <i>admin web</i> |
| Skenario alternatif | |
| 1. Memasukkan <i>username</i> dan <i>password</i> | |
| | 2. Mengecek valid tidaknya data masukan |
| | 3. Menampilkan pesan login tidak valid |
| 4. Memasukkan <i>username</i> dan <i>password</i> yang valid | |
| | 5. Mengecek valid tidaknya data masukan |
| | 6. Masuk ke halaman <i>admin web</i> |

b) Nama *Use Case* : *logout*

Tabel 3. Skenario *Use Case Logout*

| Aksi aktor | Reaksi sistem |
|-------------------------------|--|
| Skenario normal | |
| 1. Memilih menu <i>logout</i> | |
| | 2. Keluar dari halaman <i>admin web / logout</i> |

c) Nama Use Case : ubah password

Tabel 18. Skenario Use Case Ubah Password

| Aksi aktor | Reaksi sistem |
|--|--|
| Skenario normal | |
| 1. Memasukkan <i>password lama</i> dan <i>password baru</i> | |
| | 2. Mengecek valid tidaknya data masukkan |
| | 3. <i>Password</i> baru disimpan |
| Skenario alternatif | |
| 1. Memasukkan <i>password lama</i> dan <i>password baru</i> | |
| | 2. Mengecek valid tidaknya data masukkan |
| | 3. Menampilkan pesan data tidak valid |
| 4. Memasukkan <i>password lama</i> dan <i>password baru</i> yang valid | |
| | 5. Mengecek valid tidaknya data masukkan |
| | 6. <i>Password</i> baru disimpan |

d) Nama Use Case : lupa password

Tabel 19. Skenario Use Case Ubah Password

| Aksi aktor | Reaksi sistem |
|--|--|
| Skenario normal | |
| 1. Memasukkan <i>password lama</i> dan <i>password baru</i> | |
| | 2. Mengecek valid tidaknya data masukkan |
| | 3. <i>Password</i> baru disimpan |
| Skenario alternatif | |
| 1. Memasukkan <i>password lama</i> dan <i>password baru</i> | |
| | 2. Mengecek valid tidaknya data masukkan |
| | 3. Menampilkan pesan data tidak valid |
| 4. Memasukkan <i>password lama</i> dan <i>password baru</i> yang valid | |

Lanjutan Tabel 19. Skenario *Use Case* Ubah *Password*

| Aksi aktor | Reaksi sistem |
|------------------------|--|
| Skenario normal | |
| | 5. Mengecek valid tidaknya data masukkan |
| | 6. <i>Password</i> baru disimpan |

e) Nama *Use Case* : tambah data

Tabel 20. Skenario *Use Case* Tambah Data

| Aksi aktor | Reaksi sistem |
|---|---|
| Skenario normal | |
| 1. Memasukkan data sesuai dengan kolom yang ada | |
| | 2. Mengecek valid tidaknya data masukkan |
| | 3. Menyimpan data ke basis data |
| | 4. Menampilkan pesan data berhasil disimpan |
| Skenario alternatif | |
| 1. Memasukkan data sesuai dengan kolom yang ada | |
| | 2. Mengecek valid tidaknya data yang dimasukkan |
| | 3. Menampilkan pesan bahwa data tidak valid |
| 4. Memperbaiki data yang tidak valid | |
| | 5. Mengecek valid tidaknya data yang dimasukkan |
| | 6. Menyimpan data ke <i>database</i> |
| | 7. Menampilkan pesan data berhasil disimpan |

f) Nama *Use Case* : edit data

Tabel 21. Skenario *Use Case* Edit Data

| Aksi aktor | Reaksi sistem |
|----------------------------------|--|
| Skenario normal | |
| 1. Memilih data yang akan diedit | |
| | 2. Menampilkan data dalam <i>form</i> yang akan diedit |

Lanjutan Tabel 21. Skenario *Use Case* Edit Data

| Aksi aktor | Reaksi sistem |
|---|---|
| Skenario normal | |
| 3. Mengedit data | |
| | 4. Memeriksa valid tidaknya data yang dimasukkan |
| | 5. Menyimpan data yang diedit ke <i>database</i> |
| | 6. Menampilkan pesan data berhasil diedit |
| Skenario alternatif | |
| 1. Memilih data yang akan diedit | |
| | 2. Menampilkan data dalam <i>form</i> yang akan diedit |
| 3. Mengedit data | |
| | 4. Memeriksa valid tidaknya data yang dimasukkan |
| | 5. Menampilkan pesan data yang dimasukkan tidak valid |
| 6. Memperbaiki data masukan yang diedit dan tidak valid | |
| | 7. Memeriksa valid tidaknya data yang dimasukkan |
| | 8. Menyimpan data yang telah diedit ke dalam basis data |
| | 9. Menampilkan pesan data berhasil diedit |

g) Nama *Use Case* : hapus data

Tabel 22. Skenario *Use Case* Hapus Data

| Aksi aktor | Reaksi sistem |
|-----------------------------------|--|
| Skenario normal | |
| 1. Memilih data yang akan dihapus | |
| | 2. Menampilkan pesan konfirmasi apakah data benar-benar akan dihapus |
| 3. Mengeklik pilihan 'ya' | |
| | 4. Menghapus data dari <i>database</i> |
| | 5. Menampilkan pesan data berhasil dihapus |

Lanjutan Tabel 22. Skenario *Use Case* Hapus Data

| Aksi aktor | Reaksi sistem |
|-----------------------------------|--|
| Skenario alternatif | |
| 1. Memilih data yang akan dihapus | |
| | 2. Menampilkan pesan konfirmasi apakah data benar-benar akan dihapus |
| 3. Mengeklik pilihan 'tidak' | |
| | 4. Kembali ke tampilan sebelumnya |

h) Nama *Use Case* : cari data

Tabel 23. Skenario *Use Case* Cari Data

| Aksi aktor | Reaksi sistem |
|-----------------------------------|---------------------------------|
| Skenario normal | |
| 1. Mencari data yang akan dilihat | |
| | 2. Menampilkan data yang dicari |

i) Nama *Use Case* : lihat data

Tabel 4. Lihat Data

| Aksi aktor | Reaksi sistem |
|------------------------|---------------------|
| Skenario normal | |
| | 1. Menampilkan data |

j) Nama *Use Case* : *import* data

Tabel 5. Skenario *Use Case* *Import* Data

| Aksi aktor | Reaksi sistem |
|------------------------------------|---|
| Skenario normal | |
| 1. Memilih data yang akan diimport | |
| | 2. Menampilkan data dalam <i>form</i> yang telah diimport |
| Skenario alternatif | |
| 1. Memilih data yang akan diimport | |
| | 2. Memeriksa valid tidaknya data data yang diimport |
| | 3. Menampilkan pesan data yang diimport tidak valid |

Lanjutan Tabel 6. Skenario *Use Case Import Data*

| Aksi aktor | Reaksi sistem |
|-----------------------------------|--|
| 4. Memperbaiki data yang diimport | |
| | 5. Memeriksa valid tidaknya data yang diimport |
| | 6. Menyimpan data yang telah diimport dalam basis data |
| | 7. Menampilkan data dalam <i>form</i> data yang telah diimport |

k) Nama *Use Case* : *export data*

Tabel 26. Skenario *Use Case Export Data*

| Aksi aktor | Reaksi sistem |
|------------------------------------|---------------------------------|
| Skenario normal | |
| 1. Memilih data yang akan diexport | |
| | 2. Mengexport data yang dipilih |

l) Nama *Use Case* : lihat statistik pengunjung

Tabel 27. Skenario *Use Case Lihat Statistik Pengunjung*

| Aksi aktor | Reaksi sistem |
|--|--|
| Skenario normal | |
| 1. Memilih <i>range</i> data yang akan dilihat | |
| | 2. Menampilkan data statistik pengunjung |

m) Nama *Use Case* : pengembalian buku

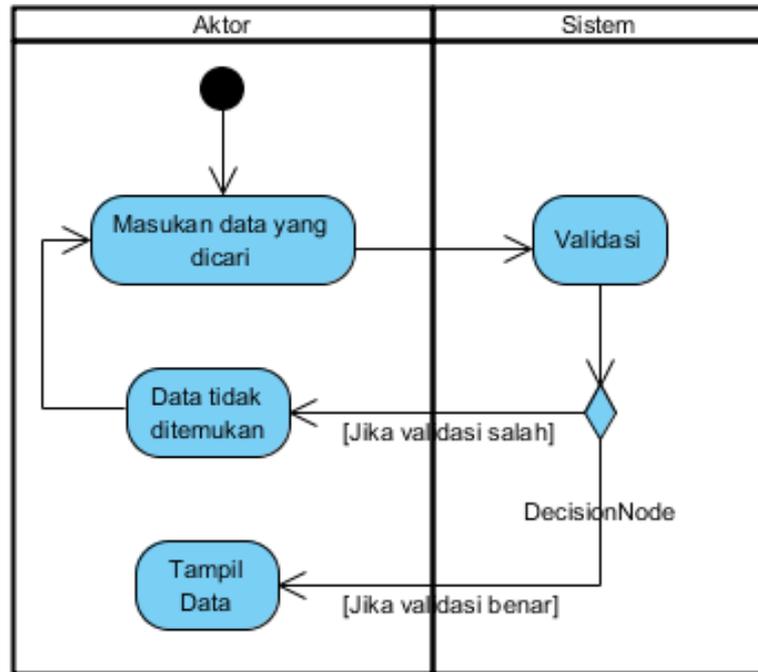
Tabel 28. Skenario *Use Case Pengembalian Buku*

| Aksi aktor | Reaksi sistem |
|----------------------------|------------------------------------|
| Skenario normal | |
| | 1. Menampilkan data peminjam |
| 2. Memilih salah satu data | |
| | 3. Pengembalian buku yang dipinjam |

Lampiran 1. Desain *Activity Diagram*

1) *Activity* Diagram Cari Data

Activity diagram cari data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 52 sebagai berikut:

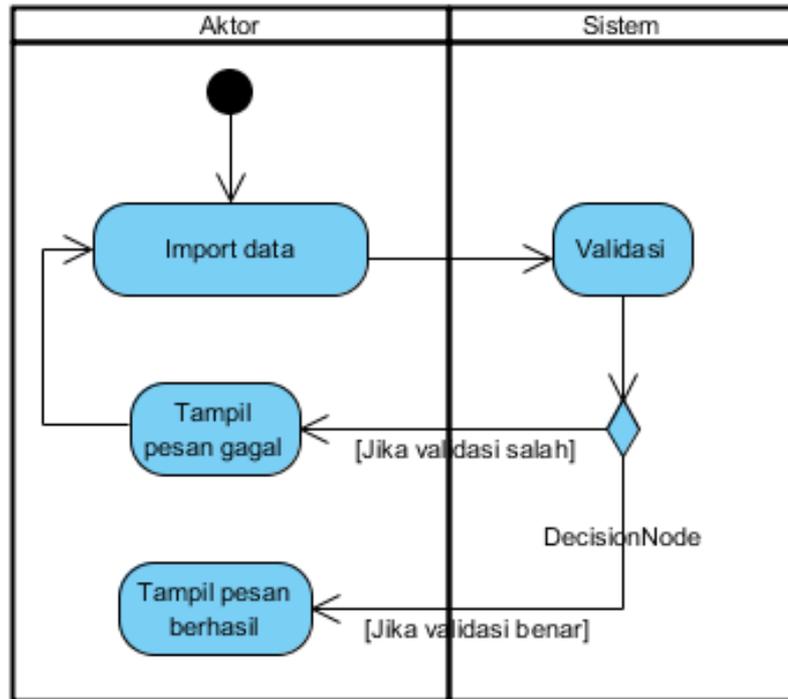


Gambar 52. *Activity* Diagram Cari Data

Berdasarkan diagram tersebut aktor admin memasukkan data yang akan dicari yang kemudian akan divalidasi oleh sistem, kemudian proses selanjutnya jika validasi benar maka sistem akan otomatis menampilkan data yang dicari, akan tetapi jika gagal maka sistem akan menampilkan halaman kosong.

2) *Activity* Diagram *Import* Data

Activity diagram *import* data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 53 sebagai berikut:

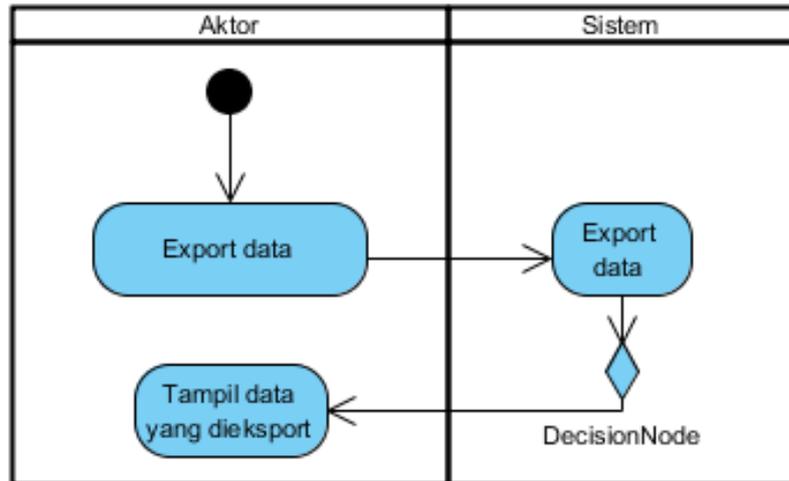


Gambar 53. Activity Diagram *Import Data*

Berdasarkan diagram tersebut aktor admin mengimport data pada sistem. Setelah admin memilih file yang akan diimport dan melakukan *submit*, maka sistem akan melakukan validasi, jika validasi sukses maka sistem akan menampilkan pesan *import* berhasil, jika gagal maka sistem akan menampilkan *import* gagal beserta kesalahan validasi yang dilakukan oleh admin.

3) Activity Diagram *Export Data*

Activity diagram *export* data untuk sistem informasi perpustakaan dijelaskan dalam Gambar 54 sebagai berikut:

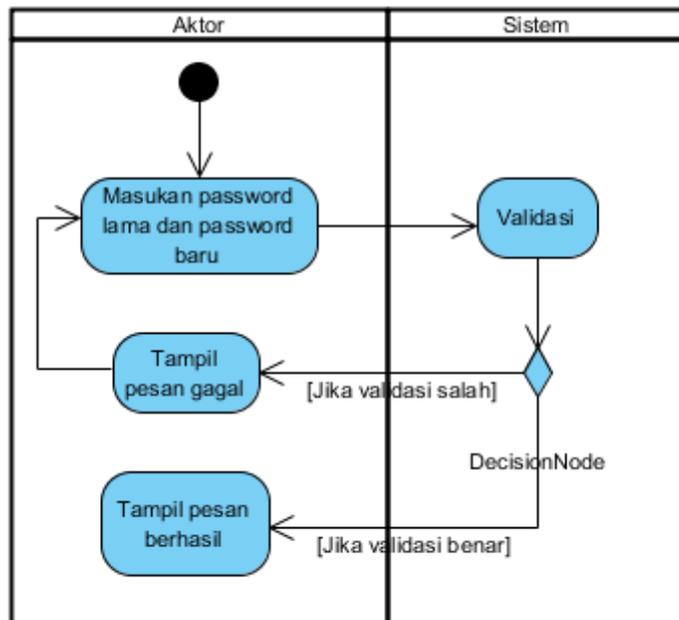


Gambar 54. Activity Diagram *Export Data*

Berdasarkan diagram tersebut aktor admin mengekspor data yang kemudian akan dilakukan oleh sistem, kemudian proses selanjutnya jika *export* berhasil maka sistem akan otomatis menampilkan data yang telah diekspor oleh admin.

4) Activity Diagram Ganti Password

Activity diagram ganti password untuk sistem informasi perpustakaan dijelaskan dalam Gambar 55 sebagai berikut:

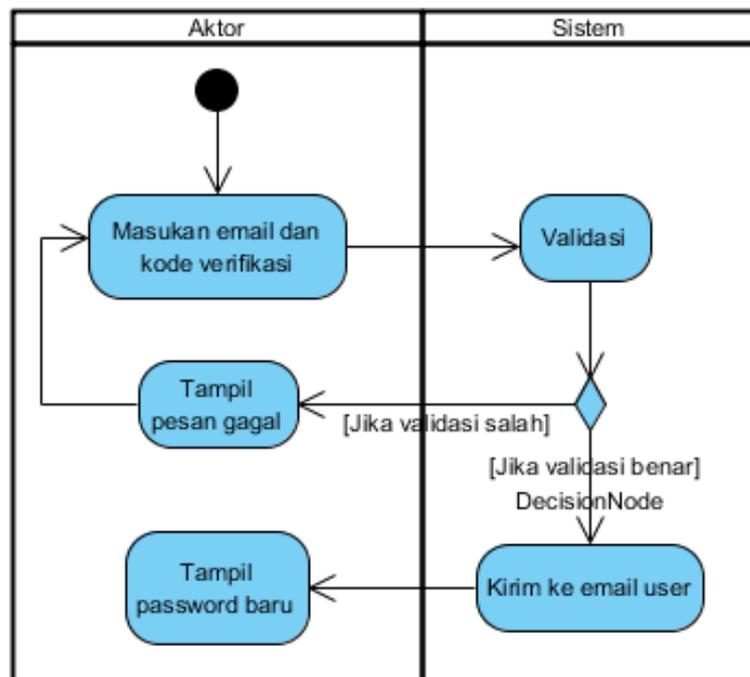


Gambar 55. Activity Diagram Ganti Password

Berdasarkan diagram tersebut aktor admin memasukan *password* lama dan *password* baru yang kemudian akan divalidasi oleh sistem, kemudian proses selanjutnya jika validasi benar maka sistem akan otomatis menampilkan halaman *dashboard* admin beserta tampilan pesan berhasil, akan tetapi jika gagal maka sistem akan menampilkan pesan kesalahan sesuai dengan kesalahan masukan admin.

5) Activity Diagram Lupa Password

Activity diagram lupa *password* untuk sistem informasi perpustakaan dijelaskan dalam Gambar 56 sebagai berikut:



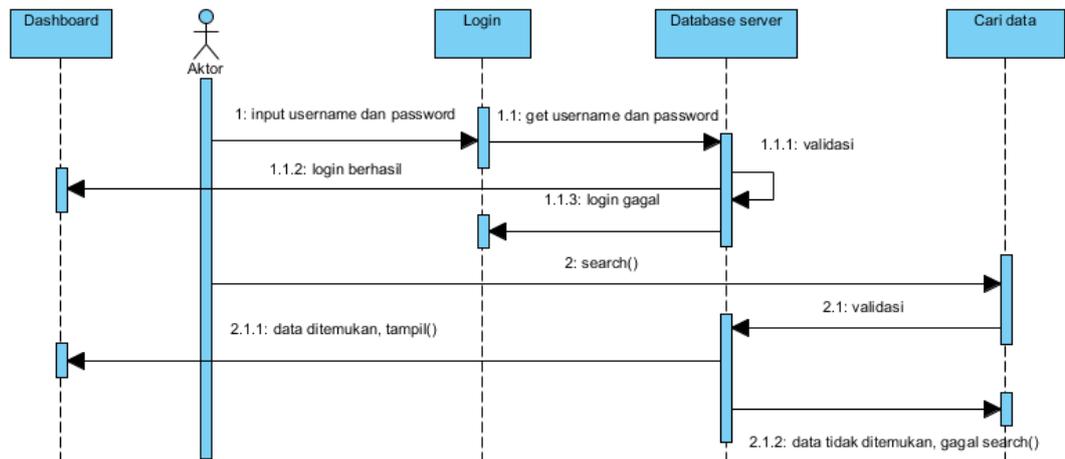
Gambar 56. Activity Diagram Lupa Password

Berdasarkan diagram tersebut aktor *user* memasukan *email* dan kode verifikasi yang kemudian akan divalidasi oleh sistem, kemudian proses selanjutnya jika validasi benar maka sistem akan otomatis mengirimkan *password* baru ke *email user*, akan tetapi jika gagal maka sistem akan menampilkan pesan kesalahan sesuai dengan kesalahan masukan dari *password user*.

Lampiran 4. Desain *Sequence Diagram*

1) Sequence Diagram Cari Data

Sequence diagram cari data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 57 sebagai berikut:

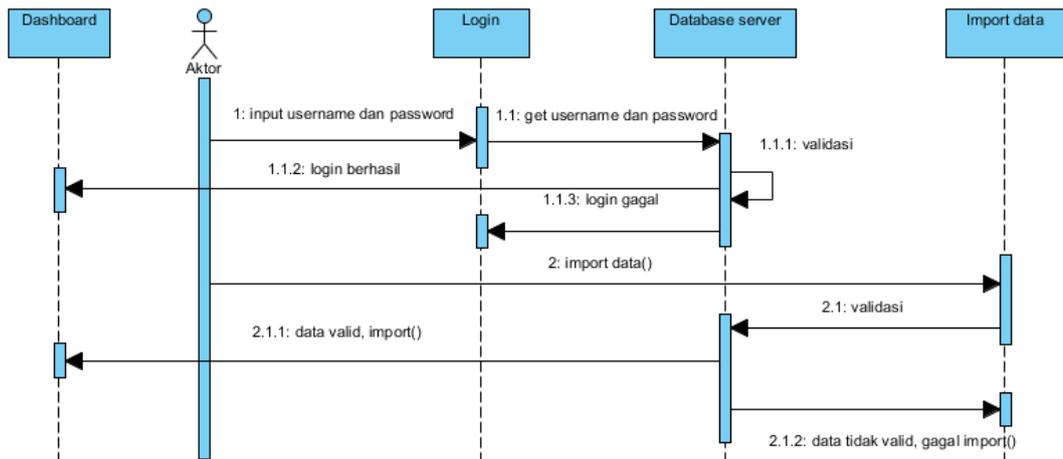


Gambar 57. Sequence Diagram Cari Data

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan pencarian data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan menampilkan data. Sedangkan jika validasi salah maka sistem tidak akan menampilkan data.

2) Sequence Diagram Import Data

Sequence diagram *import* data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 58 sebagai berikut:

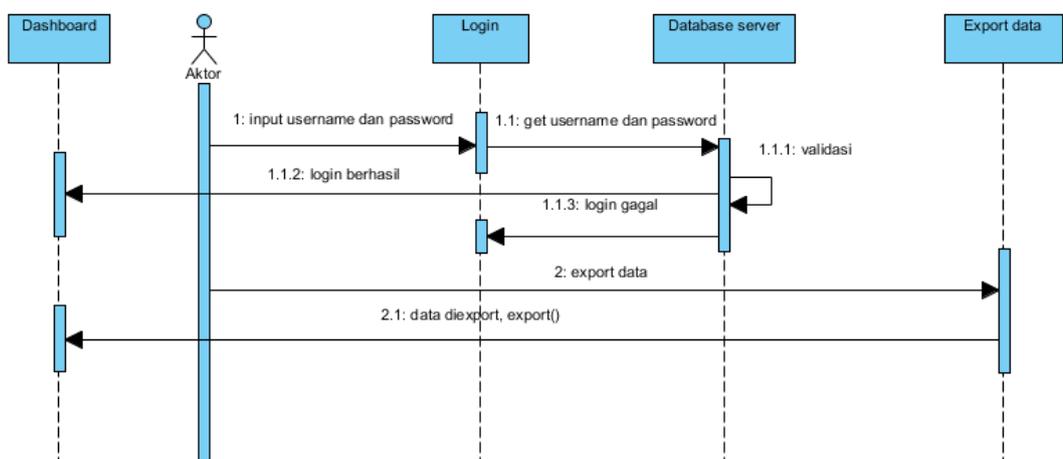


Gambar 58. *Sequence Diagram Import Data*

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan import data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan tabel yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan mengimport data. Sedangkan jika validasi salah maka sistem tidak akan mengimport data.

3) *Sequence Diagram Export Data*

Sequence diagram *export* data dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 59 sebagai berikut:

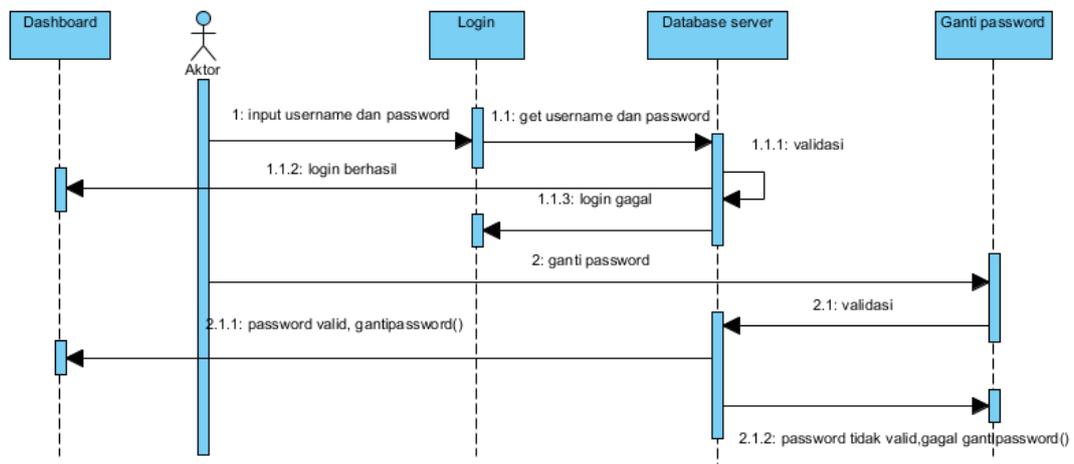


Gambar 59. *Sequence Diagram Export Data*

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan *export* data. Kemudian sistem akan menerima masukan dan menyesuaikan dengan tabel yang ada dalam *database* dan melakukan *export*.

4) *Sequence* Diagram Ganti *Password*

Sequence diagram ganti *password* dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 60 sebagai berikut:

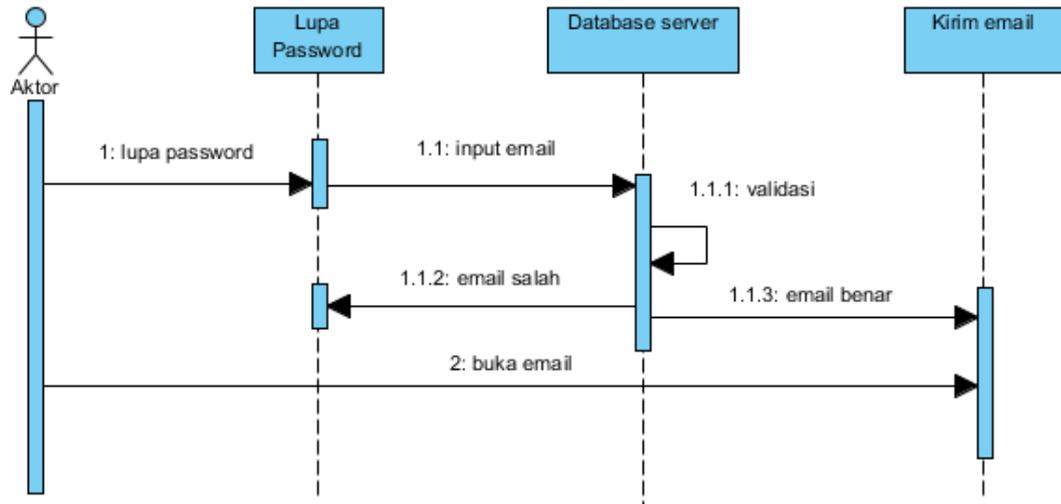


Gambar 60. *Sequence* Diagram Ganti *Password*

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan login dengan memasukkan *username* dan *password*. Setelah berhasil login, aktor melakukan ganti *password* dengan memasukkan *password* lama dan *password* baru. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam *database* dan melakukan validasi. Jika validasi benar maka sistem akan mengganti *password* dan membawa aktor ke halaman *dashboard*. Sedangkan jika validasi salah maka sistem akan kembali ke halaman ganti *password*.

5) Sequence Diagram Lupa Password

Sequence diagram lupa password dari sistem informasi perpustakaan yang dikembangkan adalah seperti Gambar 61 sebagai berikut:



Gambar 61. Sequence Diagram Lupa Password

Berdasarkan *sequence* diagram diatas, dapat dilihat aktor pertama kali harus melakukan lupa password. Kemudian pada form yang disediakan, aktor melakukan ganti password dengan memasukkan alamat email dan kode verifikasi. Kemudian sistem akan menerima masukan dan menyesuaikan dengan data yang ada dalam database dan melakukan validasi. Jika validasi benar maka sistem akan mengirimkan password ke email aktor. Sedangkan jika validasi salah maka sistem akan kembali ke halaman lupa password.

Lampiran 5. Hasil Pengujian *Efficiency*

1) Hasil Pengujian *Efficiency* Halaman Buku

Hasil pengujian *efficiency* halaman buku dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 62 sebagai berikut:



Gambar 62. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 25 dengan ukuran halaman 300KB. *Grade* yang didapatkan adalah A (97%) untuk pengujian Page Speed dan A (90%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.13 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

2) Hasil Pengujian *Efficiency* Halaman Siswa

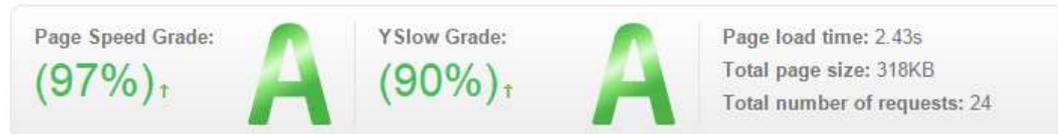
Hasil pengujian *efficiency* halaman siswa dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 63 sebagai berikut:

Latest Performance Report for: <http://mypus.esy.es/admin/siswas>

[Download PDF](#)

Report generated: Sun, May 24, 2015, 2:34 AM -0700
Test Server Region: Vancouver, Canada
Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

Summary



Gambar 63. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 24 dengan ukuran halaman 318KB. *Grade* yang didapatkan adalah A (97%) untuk pengujian Page Speed dan A (90%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.43 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

3) Hasil Pengujian *Efficiency* Statistik Pengunjung

Hasil pengujian *efficiency* Statistik pengunjung dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 64 sebagai berikut:

Latest Performance Report for: <http://mypus.esy.es/chartPengunjung>

[Download PDF](#)

Report generated: Sun, May 24, 2015, 1:49 AM -0700
Test Server Region: Vancouver, Canada
Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

Summary



Gambar 64. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 25 dengan ukuran halaman 300KB. *Grade* yang didapatkan adalah A (97%) untuk pengujian Page Speed dan A (90%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.31 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

4) Hasil Pengujian *Efficiency* Halaman Tamu/*Guest*

Hasil pengujian *efficiency* halaman tamu dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 65 sebagai berikut:



Gambar 65. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 24 dengan ukuran halaman 376KB. *Grade* yang didapatkan adalah A (93%) untuk pengujian Page Speed dan B (89%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.43 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

5) Hasil Pengujian *Efficiency* Halaman Layanan

Hasil pengujian *efficiency* halaman layanan dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 66 sebagai berikut:



Gambar 66. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 15 dengan ukuran halaman 440KB. *Grade* yang didapatkan adalah A (91%) untuk pengujian Page Speed dan A (91%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.25 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

6) Hasil Pengujian *Efficiency* Halaman Profil

Hasil pengujian *efficiency* halaman profil dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 67 sebagai berikut:

Latest Performance Report for: <http://mypus.esy.es/profil>

 [Download PDF](#)

Report generated: Sun, May 24, 2015, 2:31 AM -0700
Test Server Region: Vancouver, Canada
Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

 Looks like you might not be using a CDN
[Why should I use a CDN? >](#)

Summary

| | | | | |
|---|----------|--|----------|---|
| Page Speed Grade: (91%)_↑ | A | YSlow Grade: (91%)_↑ | A | Page load time: 2.10s Total page size: 440KB Total number of requests: 15 |
|---|----------|--|----------|---|

Gambar 67. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 15 dengan ukuran halaman 440KB. *Grade* yang didapatkan adalah A (91%) untuk pengujian Page Speed dan A (91%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.10 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

7) Hasil Pengujian *Efficiency* Halaman Tentang Kami

Hasil pengujian *efficiency* halaman tentang kami dalam pengembangan sistem informasi perpustakaan dapat dilihat pada Gambar 68 sebagai berikut:

Latest Performance Report for: <http://mypus.esy.es/tentangkami>

 [Download PDF](#)

Report generated: Sun, May 24, 2015, 2:30 AM -0700
Test Server Region: Vancouver, Canada
Using: Firefox (Desktop) 25.0.1, Page Speed 1.12.16, YSlow 3.1.8

 Looks like you might not be using a CDN
[Why should I use a CDN? >](#)

Summary

| | | | | |
|---|----------|--|----------|---|
| Page Speed Grade: (91%)_↑ | A | YSlow Grade: (91%)_↑ | A | Page load time: 2.10s Total page size: 440KB Total number of requests: 15 |
|---|----------|--|----------|---|

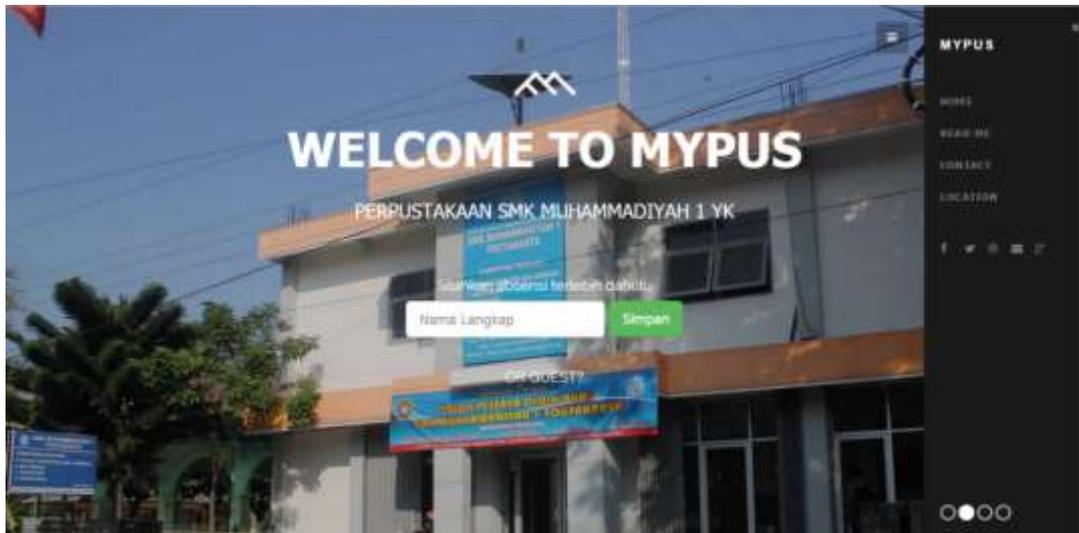
Gambar 68. Hasil Pengujian GTMetrix

Dari gambar diatas dapat dilihat bahwa *HTTP Request* yang dilakukan sebanyak 15 dengan ukuran halaman 440KB. *Grade* yang didapatkan adalah A (91%) untuk pengujian Page Speed dan A (91%) untuk YSlow. *Grade* ini menunjukkan bahwa tingkat efisiensi *resource software* yang diujikan sudah berada di atas rata-rata *grade* GTMetrix (*grade* GTMetrix untuk Page Speed dan YSlow bernilai 79%). Waktu untuk *load* halaman 2.10 detik, yang berarti sudah memenuhi standar Nielsen (2010) karena waktu *load* berada di bawah 10 detik.

Lampiran 6. Implementasi Antarmuka

1) Halaman Utama

Implementasi halaman utama sistem informasi perpustakaan adalah seperti Gambar 69 sebagai berikut:



Gambar 69. Hasil implementasi Halaman Utama

2) Halaman Tentang Perpustakaan

Implementasi halaman tentang perpustakaan sistem informasi perpustakaan adalah seperti Gambar 70 sebagai berikut:



Tentang Perpustakaan SMK Muhammadiyah 1 Yogyakarta

Di Perpustakaan SMK Muhammadiyah 1 Yogyakarta, Anda akan menjumpai berbagai buku, majalah dan referensi lengkap mengenai ekonomi, manajemen, pemrograman, hingga multimedia, untuk menunjang proses praktik dan pemahaman materi siswa. Perpustakaan SMK Muhammadiyah 1 Yogyakarta juga menyediakan layanan pinjam-buku. Sebagai lembaga pendidikan sekolah lanjutan di bidang teknologi, kami senantiasa memantau perkembangan teknologi informasi agar kami dapat melengkapi koleksi perpustakaan kami dengan buku-buku terkini, baik dari berbagai daerah maupun internasional. Perpustakaan SMK Muhammadiyah 1 Yogyakarta menyediakan apa yang Anda butuhkan. Mulai dari buku tentang tips memulai karir dan bisnis, manajemen sebuah organisasi, hingga buku tentang menjadi seorang yang sukses. Karena di SMK Muhammadiyah 1 Yogyakarta, Anda tidak hanya dibentuk menjadi seorang profesional di bidang kejuruan, namun juga menjadi seorang yang memiliki daya kompetisi yang tinggi dalam dunia bisnis. SMK Muhammadiyah 1 Yogyakarta memprioritaskan anggaran untuk pengembangan Perpustakaan. Karena kami yakin perpustakaan adalah sumber utama bagi pengembangan ilmu pengetahuan dan teknologi. Baik-buruknya suatu perpustakaan tidak ditentukan dari kuantitas koleksinya, akan tetapi cenderung pada kualitas koleksinya. Perpustakaan SMK Muhammadiyah 1 Yogyakarta telah bekerjasama dengan berbagai penerbit buku-buku berkualitas baik dari dalam ataupun luar negeri untuk memenuhi kebutuhan akan buku-buku terkini yang berkualitas. Mulailah berkunjung dan arahkan visi Anda menuju masa depan yang sukses.



Gambar 70. Hasil implementasi Halaman Tentang Perpustakaan

3) Halaman Lokasi Perpustakaan

Implementasi halaman lokasi sistem informasi perpustakaan adalah seperti

Gambar 71 sebagai berikut:



Gambar 71. Hasil implementasi Halaman Lokasi

4) Halaman Lupa Password

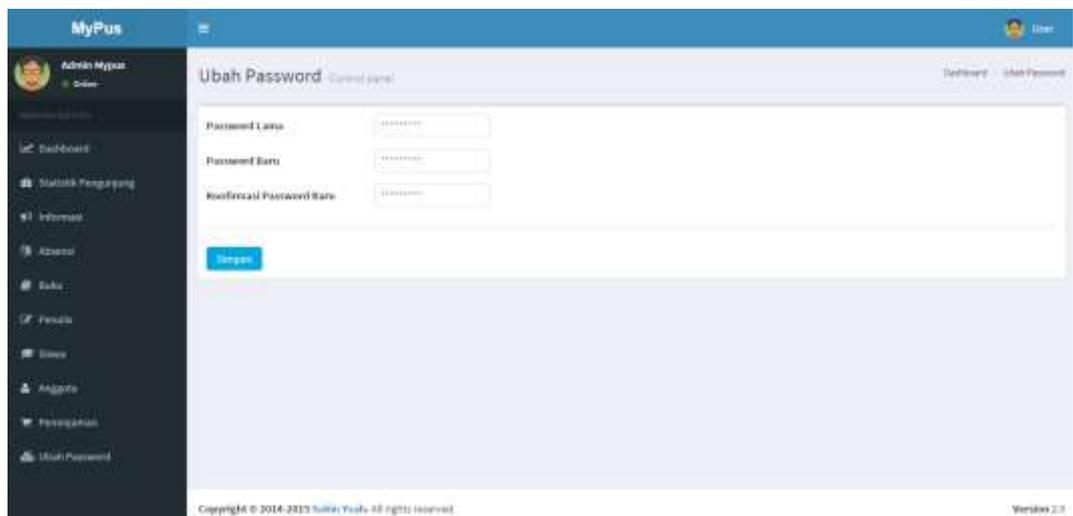
Implementasi halaman lupa password sistem informasi perpustakaan adalah seperti Gambar 72 sebagai berikut:



Gambar 72. Hasil implementasi Halaman Lupa *Password*

5) Halaman Ubah *Password*

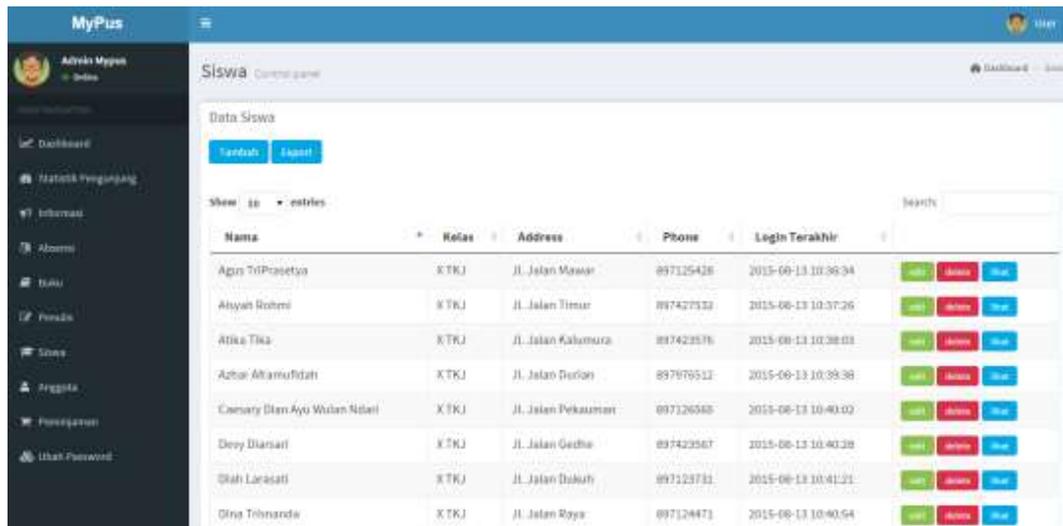
Implementasi halaman ubah *password* sistem informasi perpustakaan adalah seperti Gambar 73 sebagai berikut:



Gambar 73. Hasil implementasi Halaman Ubah *Password*

6) Halaman Daftar Siswa

Implementasi halaman daftar siswa sistem informasi perpustakaan adalah seperti Gambar 74 sebagai berikut:



Gambar 74. Hasil implementasi Halaman Daftar Siswa

7) Halaman Lihat Siswa

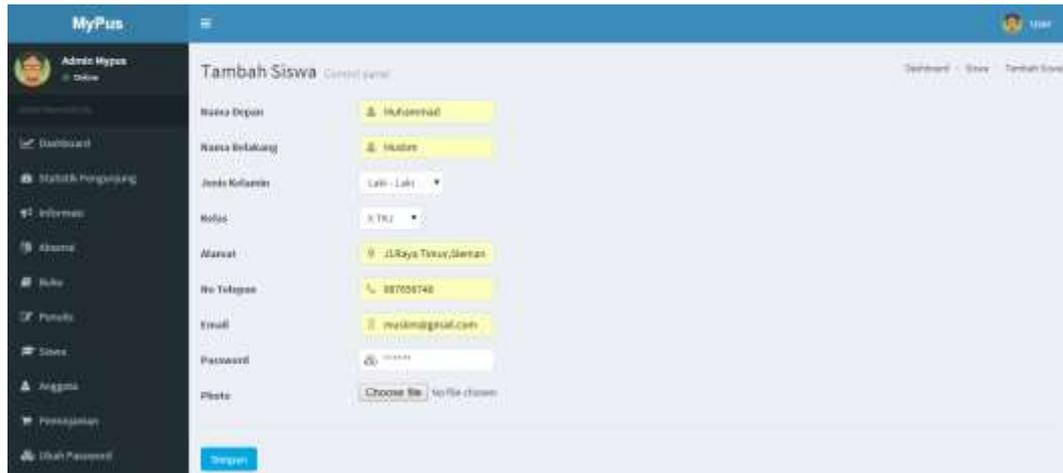
Implementasi halaman lihat siswa sistem informasi perpustakaan adalah seperti Gambar 75 sebagai berikut:



Gambar 75. Hasil implementasi Halaman Lihat Siswa

8) Halaman Tambah Siswa

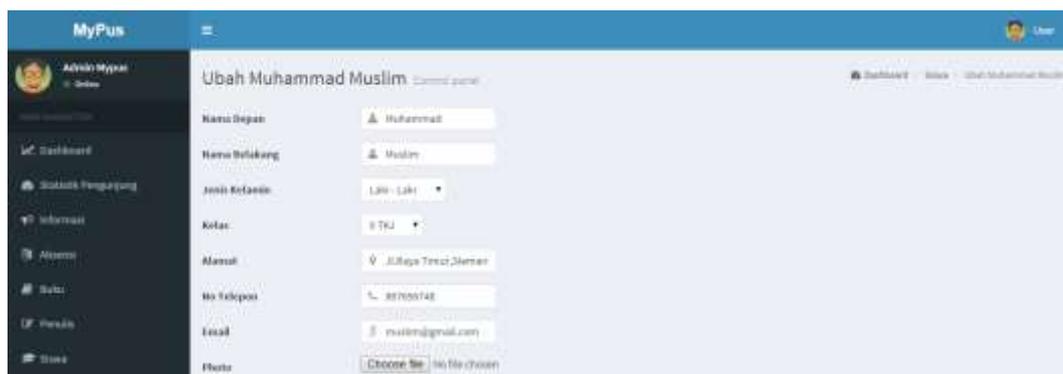
Implementasi halaman tambah siswa sistem informasi perpustakaan adalah seperti Gambar 76 sebagai berikut:



Gambar 76. Hasil implementasi Halaman Tambah Siswa

9) Halaman Edit Siswa

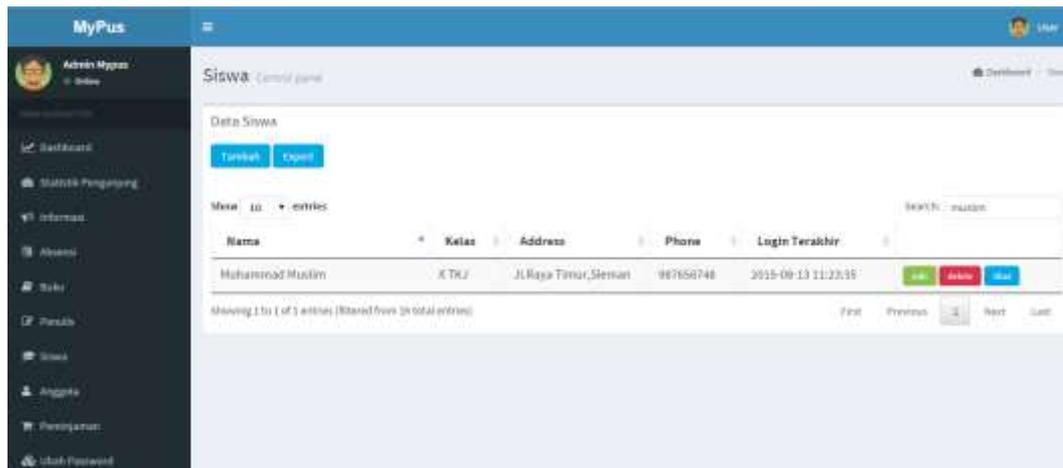
Implementasi halaman edit siswa sistem informasi perpustakaan adalah seperti Gambar 77 sebagai berikut:



Gambar 77. Hasil implementasi Halaman Edit Siswa

10) Halaman Cari Siswa

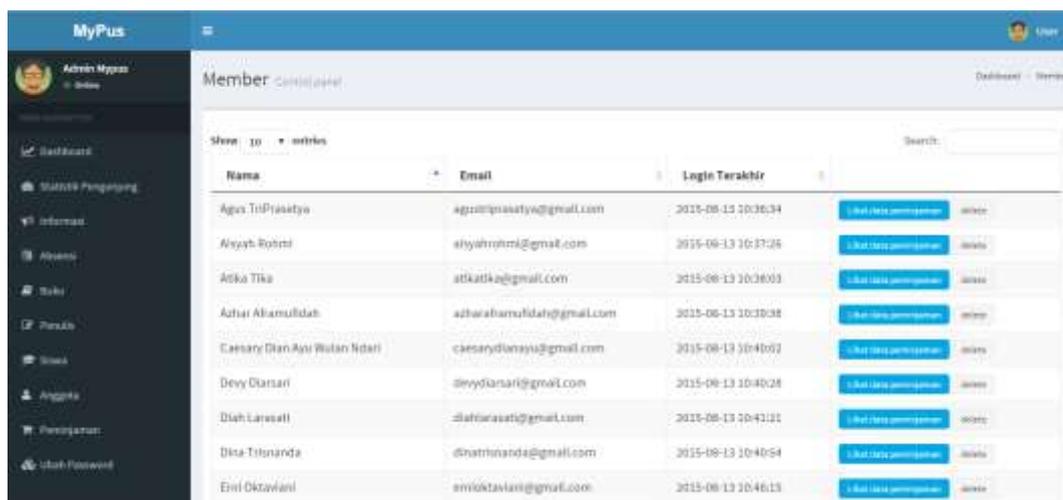
Implementasi halaman cari siswa sistem informasi perpustakaan adalah seperti Gambar 78 sebagai berikut:



Gambar 78. Hasil implementasi Halaman Cari Siswa

11) Halaman Anggota

Implementasi halaman anggota sistem informasi perpustakaan adalah seperti Gambar 79 sebagai berikut:



Gambar 79. Hasil implementasi Halaman Anggota

12) Halaman Peminjaman

Implementasi halaman peminjaman sistem informasi perpustakaan adalah seperti Gambar 80 sebagai berikut:

Data Peminjaman

Buku (Bros Pengumpulan Data) (terakhir diperbarikan)

Data Data Peminjaman

Mulai ID • urutkan Status Senza Search

| Buku | Peminjam | Tanggal Pinjam | Tanggal Kembali |
|--------------------------|------------------|----------------|-----------------|
| Rekayasa Perangkat Lunak | Agus TriPrasetya | 2015-08-13 | 2015-08-13 |
| Telekik Telekomunikasi | Agus TriPrasetya | 2015-08-13 | 2015-08-13 |
| Ilmu Pergetahuan Sosial | Agus TriPrasetya | 2015-08-13 | 2015-08-15 |
| Rekayasa Perangkat Lunak | Alyah Rafeni | 2015-08-13 | kembali |
| Telekik Telekomunikasi | Atka Tika | 2015-08-13 | kembali |
| Ilmu Pergetahuan Sosial | Atka Tika | 2015-08-13 | kembali |
| Rekayasa Perangkat Lunak | Athar Alramulhah | 2015-08-13 | kembali |

Gambar 80. Hasil implementasi Halaman Peminjaman

13) Halaman Tambah Buku

Implementasi halaman tambah buku sistem informasi perpustakaan adalah seperti Gambar 81 sebagai berikut:

Tambah Buku

Tambah Baru Upload Excel

Judul

Penulis

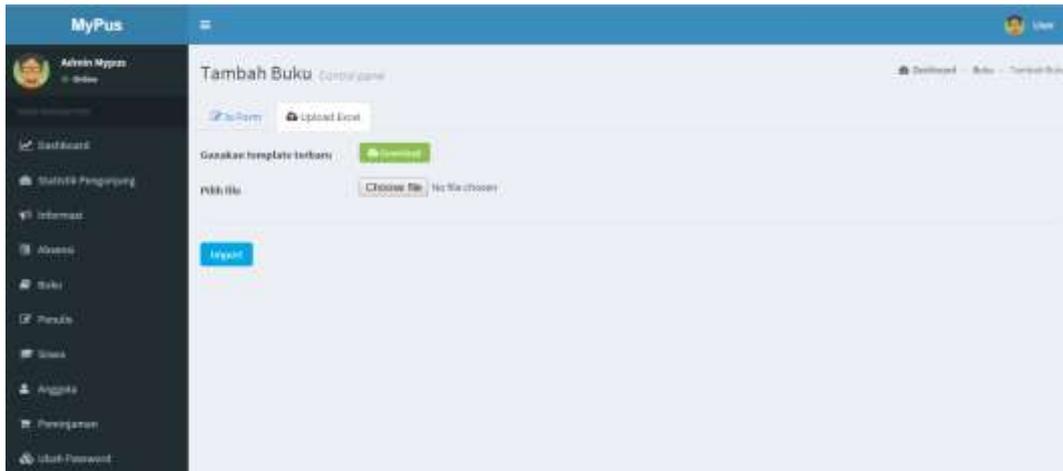
Jumlah

Cover No file chosen

Gambar 81. Hasil implementasi Halaman Tambah Buku

14) Halaman *Import* Buku

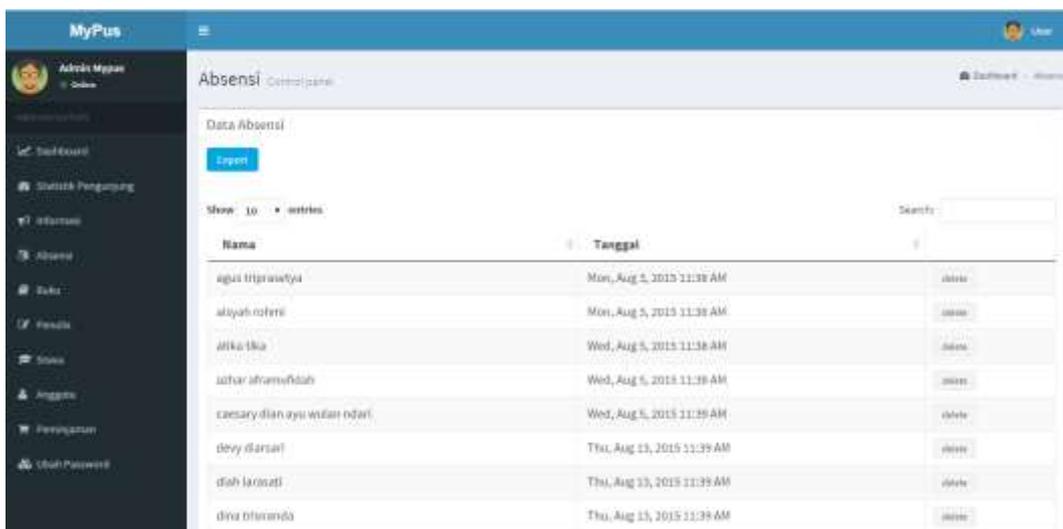
Implementasi halaman *import* buku sistem informasi perpustakaan adalah seperti Gambar 82 sebagai berikut:



Gambar 82. Hasil implementasi Halaman *Import* Buku

15) Halaman Absensi

Implementasi halaman absensi sistem informasi perpustakaan adalah seperti Gambar 83 sebagai berikut:



Gambar 83. Hasil implementasi Halaman Absensi

Lampiran 7. Analisis Kebutuhan

Lampiran Lembar Analisis Kebutuhan

ANALISIS KEBUTUHAN

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB
DI SMK MUHAMMADIYAH 1 YOGYAKARTA

Nama : Nita Dewi P

Jabatan : Guru

Jawablah pertanyaan dibawah ini sesuai dengan keadaan sebenarnya :

A. Identifikasi Permasalahan

1. Bagaimana proses peminjaman dan pengembalian buku di perpustakaan SMK Muhammadiyah 1 Yogyakarta?
Melalui proses pencatatan buku besar dan kartu perpustakaan.
2. Bagaimana pelaporan data peminjaman dan pengembalian buku?
Dengan melihat buku besar peminjaman
3. Apa kendala dari proses peminjaman dan pengembalian buku?
Petugas perpustakaan tidak berada di tempat
4. Apa kendala pelaporan data peminjaman dan pengembalian buku?
Merinci / meneliti satu per satu halaman buku besar.

5. Software apa yang sudah digunakan dan apa kekurangannya?

- program excel manual

- Memasukkan data sah

B. Penggunaan dan Pemanfaatan

1. Siapakah yang akan menggunakan software ini?

- Admin Perpustakaan
- Siswa / siswi
- Guru
- Karyawan

2. Dimanakah software ini akan digunakan?

Perpustakaan Sekolah

C. Identifikasi Kebutuhan Software

Input

1. Apa saja kategori buku yang ada di perpustakaan SMK Muhammadiyah 1 Yogyakarta?

- Paket mata pelajaran
- Cerita kepahlawanan
- Novel
- Majalah

2. Siapakah yang melakukan input data peminjaman dan pengembalian buku?

- Peminjam

3. Apa saja data buku dan siswa yang diperlukan?

- Judul
- Pengarang
- Th cetakan
- Sinopsis

Proses

1. Bagaimana proses pencarian buku yang ada di perpustakaan?

Pengon mencari judul buku dalam rak

2. Bagaimana proses peminjaman buku? Apakah ada batas waktu peminjaman dan batasan lain?

Proses siswa datang ke perpustakaan dg membawa kartu perpustakaan dan dicatat oleh petugas perpustakaan.

Batasan 1 minggu selanjutnya diperpanjang

Output

1. Bagaimanakah gambaran sistem yang diinginkan?

Seperti yang dipresentasikan ditambah halaman untuk "program kerja" (bagi Admin),
"Laporan tahunan" (bagi Admin).

Yogyakarta,

19 Mei 2015

Narasumber,


Nita Dewi P

Lampiran 8. Angket Pengujian *Usability*

Lampiran Lembar Pengujian Usability Guru

LEMBAR PENGUJIAN USABILITY

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB DI SMK MUHAMMADIYAH 1 YOGYAKARTA

Nama : Nurcahyo Hanu W.

Jabatan : Siswa / Guru *)

Petunjuk Pengisian :

Berilah tanda centang (✓) pada kolom yang sesuai dengan pilihan Anda sesuai dengan keyakinan masing-masing.

Pilihan Jawaban :

STS : Sangat Tidak Setuju RG : Ragu-ragu SS : Sangat Setuju

TS : Tidak Setuju S : Setuju

| No. | Pernyataan | Jawaban | | | | |
|-----|--|---------|----|----|---|----|
| | | STS | TS | RG | S | SS |
| 1. | Software ini membantu saya bekerja lebih efektif | | | | | ✓ |
| 2. | Software ini membantu saya bekerja lebih produktif | | | | ✓ | |
| 3. | Software ini sangat berguna. | | | | | ✓ |
| 4. | Software ini memberikan saya pengendalian lebih atas aktivitas saya | | | | | ✓ |
| 5. | Software ini mempermudah saya dalam menyelesaikan apa yang ingin saya selesaikan | | | | | ✓ |
| 6. | Software ini menghemat waktu saya ketika saya menggunakannya | | | | ✓ | |

| | | | | | | |
|-----|--|--|--|--|---|---|
| 7. | Software ini sesuai dengan kebutuhan saya | | | | | ✓ |
| 8. | Software ini melakukan segala sesuatu yang saya harapkan untuk dilakukan | | | | ✓ | |
| 9. | Software ini mudah digunakan | | | | | ✓ |
| 10. | Software ini praktis digunakan | | | | | ✓ |
| 11. | Software ini mudah dipahami | | | | | ✓ |
| 12. | Software ini membutuhkan langkah-langkah yang sedikit untuk mencapai apa yang ingin saya lakukan dengan software ini | | | | | ✓ |
| 13. | Software ini fleksibel | | | | | ✓ |
| 14. | Tidak ada kesulitan dalam menggunakan software ini | | | | ✓ | |
| 15. | Saya dapat menggunakan software ini tanpa panduan tertulis | | | | ✓ | |
| 16. | Saya tidak melihat adanya inkonsistensi saat saya gunakan software ini | | | | ✓ | |
| 17. | Baik pengguna yang sesekali menggunakan dan pengguna yang biasa menggunakan akan menyukai software ini | | | | | ✓ |
| 18. | Saya dapat menangani kesalahan dengan cepat dan mudah | | | | ✓ | |
| 19. | Saya dapat menggunakan software ini secara benar setiap saat | | | | ✓ | |

| | | | | | | |
|-----|---|--|--|--|---|---|
| 20. | Saya belajar untuk menggunakan software ini secara cepat | | | | | ✓ |
| 21. | Saya mudah mengingat bagaimana menggunakan ini | | | | ✓ | |
| 22. | Software ini mudah untuk dipelajari bagaimana penggunaannya | | | | ✓ | |
| 23. | Saya menjadi terampil menggunakan software ini secara cepat | | | | ✓ | |
| 24. | Saya puas dengan software ini | | | | | ✓ |
| 25. | Saya akan merekomendasikan software ini ke teman | | | | | ✓ |
| 26. | Software ini menyenangkan untuk digunakan | | | | | ✓ |
| 27. | Software ini bekerja seperti yang saya inginkan | | | | | ✓ |
| 28. | Software ini memiliki tampilan yang sangat bagus | | | | | ✓ |
| 29. | Menurut saya, saya perlu memiliki software ini | | | | | ✓ |
| 30. | Software ini nyaman untuk digunakan | | | | | ✓ |

*) coret yang tidak perlu

Terima kasih saya ucapkan atas bantuan dan partisipasi Anda dalam penelitian ini.

Yogyakarta, 19 Mei 2015

Responden,


Nur Cahya I.W.

Lampiran Lembar Pengujian Usability Siswa

LEMBAR PENGUJIAN USABILITY

Nama : Dina Trisnanda

Jabatan : Siswa / Guru *)

Petunjuk Pengisian :

Berilah tanda centang (✓) pada kolom yang sesuai dengan pilihan Anda sesuai dengan keyakinan masing-masing.

Pilihan Jawaban :

STS : Sangat Tidak Setuju RG : Ragu-ragu SS : Sangat Setuju

TS : Tidak Setuju S : Setuju

| No. | Pernyataan | Jawaban | | | | |
|-----|--|---------|----|----|---|----|
| | | STS | TS | RG | S | SS |
| 1. | Software ini membantu saya bekerja lebih efektif | | | | | ✓ |
| 2. | Software ini membantu saya bekerja lebih produktif | | | | | ✓ |
| 3. | Software ini sangat berguna | | | | | ✓ |
| 4. | Software ini memberikan saya pengendalian lebih atas aktivitas saya | | | | | ✓ |
| 5. | Software ini mempermudah saya dalam menyelesaikan apa yang ingin saya selesaikan | | | | | ✓ |
| 6. | Software ini menghemat waktu saya ketika saya menggunakannya | | | | | ✓ |
| 7. | Software ini sesuai dengan kebutuhan saya | | | | ✓ | |

| | | | | | | |
|-----|--|--|--|--|---|---|
| 8. | <i>Software</i> ini melakukan segala sesuatu yang saya harapkan untuk dilakukan | | | | | ✓ |
| 9. | <i>Software</i> ini mudah digunakan | | | | | ✓ |
| 10. | <i>Software</i> ini praktis digunakan | | | | | ✓ |
| 11. | <i>Software</i> ini mudah dipahami | | | | | ✓ |
| 12. | <i>Software</i> ini membutuhkan langkah-langkah yang sedikit untuk mencapai apa yang ingin saya lakukan dengan <i>software</i> ini | | | | ✓ | |
| 13. | <i>Software</i> ini fleksibel | | | | | ✓ |
| 14. | Tidak ada kesulitan dalam menggunakan <i>software</i> ini | | | | ✓ | |
| 15. | Saya dapat menggunakan <i>software</i> ini tanpa panduan tertulis | | | | | ✓ |
| 16. | Saya tidak melihat adanya inkonsistensi saat saya gunakan <i>software</i> ini | | | | ✓ | |
| 17. | Baik pengguna yang sesekali menggunakan dan pengguna yang biasa menggunakan akan menyukai <i>software</i> ini | | | | | ✓ |
| 18. | Saya dapat menangani kesalahan dengan cepat dan mudah | | | | | ✓ |
| 19. | Saya dapat menggunakan <i>software</i> ini secara benar setiap saat | | | | | ✓ |
| 20. | Saya belajar untuk menggunakan <i>software</i> ini secara cepat | | | | | ✓ |

| | | | | | | |
|-----|---|--|--|--|---|---|
| 21. | Saya mudah mengingat bagaimana menggunakan ini | | | | | ✓ |
| 22. | Software ini mudah untuk dipelajari bagaimana penggunaannya | | | | | ✓ |
| 23. | Saya menjadi terampil menggunakan software ini secara cepat | | | | | ✓ |
| 24. | Saya puas dengan software ini | | | | | ✓ |
| 25. | Saya akan merekomendasikan software ini ke teman | | | | ✓ | |
| 26. | Software ini menyenangkan untuk digunakan | | | | ✓ | |
| 27. | Software ini bekerja seperti yang saya inginkan | | | | | ✓ |
| 28. | Software ini memiliki tampilan yang sangat bagus | | | | ✓ | |
| 29. | Menurut saya, saya perlu memiliki software ini | | | | | ✓ |
| 30. | Software ini nyaman untuk digunakan | | | | | ✓ |

*) coret yang tidak perlu

Terima kasih saya ucapkan atas bantuan dan partisipasi Anda dalam penelitian ini.

Yogyakarta, 29 Mei 2015

Responden,


...Dina Trisnanda

Lampiran 9. Data Pengujian *Usability*

Tabel 29. Daftar Responden Pengujian *Usability*

| No. | Nama | Jabatan |
|-----|------------------------------|---------|
| 1 | Ari Purnomo Aji | Guru |
| 2 | Nita Dewi P | Guru |
| 3 | Nurchahyo Ibnu W | Guru |
| 4 | Agus Tri Prasetya | Siswa |
| 5 | Aisyah Rohmi | Siswa |
| 6 | Ami | Siswa |
| 7 | Atika | Siswa |
| 8 | Azhar Afra Mufidah | Siswa |
| 9 | Caesary Dian Ayu Wulan Ndari | Siswa |
| 10 | Devy Diarsari | Siswa |
| 11 | Diah Larasati | Siswa |
| 12 | Dina Trisnanda | Siswa |
| 13 | Erni Oktaviani | Siswa |
| 14 | Fanny Afiola Sunarya | Siswa |
| 15 | Fitrio Aji Cahyanti | Siswa |
| 16 | Ika Rahmawati | Siswa |
| 17 | Maisaroh Fatmasari | Siswa |
| 18 | Marwah An Aisyah Hasanah Nur | Siswa |
| 19 | Novendiyah Cahya Pratama | Siswa |
| 20 | Nurdyla Yuniati | Siswa |
| 21 | Ratih Sartikaningsih | Siswa |
| 22 | Rhodiya Mufida | Siswa |
| 23 | Rindita Okta Piyani | Siswa |

Lanjutan Tabel 29. Daftar Responden Pengujian *Usability*

| No. | Nama | Jabatan |
|-----|------------------------|---------|
| 24 | Rini Agustina | Siswa |
| 25 | Rukhi Sholikhah | Siswa |
| 26 | Siti Azizah | Siswa |
| 27 | Tri Isnawati | Siswa |
| 28 | Utami Aprilia Yuniwati | Siswa |
| 29 | Wahyu Kismi A | Siswa |
| 30 | Yuliaana Widyaningrum | Siswa |

Tabel 30. Data Pengujian *Usability*

| No. | Pertanyaan | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|------------|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | | |
| 1 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 5 | 4 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 5 | 5 | | |
| 2 | 5 | 5 | 5 | 4 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 4 | 4 | 4 | 5 | 5 | 4 | |
| 3 | 5 | 4 | 5 | 5 | 4 | 4 | 5 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 3 | 4 | 4 | 4 | 5 | 5 | 4 | 5 | |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 3 | 3 | 4 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 3 | 4 | 3 | 4 | |
| 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 5 | 5 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | |
| 6 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 3 | 3 | 4 | 3 | 3 | 3 | 2 | 3 | 4 | 4 | 4 | 3 | 3 | 4 | 3 | 3 | 4 | 4 | 4 | |
| 7 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 3 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 3 | 3 | 4 | 3 | 3 | 5 | 4 | 4 | 4 | |
| 8 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 3 | 4 | 4 | 4 | 4 | 5 | 4 | 4 | 3 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 5 | 4 | |
| 9 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 3 | 5 | 5 | 3 | 5 | 5 | 5 | 5 | 3 | 5 | 5 | 3 | 5 | 5 | 3 | 5 | |
| 10 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 3 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 3 | 5 | 5 | 5 | 5 | |
| 11 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 4 | 3 | 4 | 4 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | |
| 12 | 4 | 4 | 5 | 5 | 4 | 3 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 3 | 3 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 5 | 4 | 4 | |
| 13 | 4 | 3 | 5 | 4 | 4 | 4 | 5 | 4 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 4 | |
| 14 | 5 | 5 | 5 | 4 | 4 | 4 | 5 | 3 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 3 | 4 | 4 | 4 | 3 | 3 | 5 | 5 | 4 | 4 | 3 | 5 | 4 | 4 | 4 | |
| 15 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 3 | 3 | 5 | 4 | 3 | 3 | 5 | 3 | 4 | 4 | 5 | 5 | 5 | 3 | 4 | 4 | 3 | 3 | 5 | 3 | 5 | 3 | 5 | 5 | |
| 16 | 5 | 4 | 5 | 4 | 4 | 4 | 5 | 3 | 3 | 4 | 3 | 2 | 3 | 4 | 4 | 3 | 5 | 3 | 3 | 4 | 4 | 3 | 3 | 5 | 5 | 4 | 3 | 4 | 5 | 4 | 4 | |
| 17 | 4 | 3 | 3 | 5 | 4 | 4 | 3 | 4 | 4 | 4 | 3 | 4 | 3 | 3 | 2 | 3 | 3 | 4 | 3 | 4 | 5 | 4 | 4 | 3 | 3 | 3 | 4 | 4 | 3 | 3 | 3 | |
| 18 | 4 | 3 | 4 | 3 | 3 | 4 | 3 | 3 | 4 | 2 | 4 | 3 | 2 | 2 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 3 | 3 |
| 19 | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 2 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 2 | 3 | 2 | 3 | 4 | 4 | 4 | 4 | 2 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | |
| 20 | 3 | 4 | 4 | 3 | 4 | 5 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 2 | 3 | 3 | 3 | 4 | 3 | 3 | 4 | 3 | 5 | 5 | 5 | 3 | 3 | |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 21 | 3 | 3 | 4 | 3 | 4 | 3 | 3 | 3 | 2 | 3 | 2 | 4 | 3 | 2 | 4 | 3 | 3 | 3 | 2 | 3 | 3 | 4 | 3 | 3 | 4 | 3 | 3 | 3 | 4 | 3 |
| 22 | 4 | 4 | 3 | 4 | 3 | 3 | 4 | 3 | 3 | 3 | 4 | 3 | 4 | 4 | 3 | 2 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 3 | 3 | 4 | 4 |
| 23 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 |
| 24 | 4 | 3 | 3 | 3 | 3 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 25 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 4 | 5 | 5 |
| 26 | 3 | 3 | 4 | 4 | 5 | 4 | 3 | 3 | 3 | 4 | 4 | 4 | 5 | 4 | 3 | 3 | 3 | 4 | 4 | 3 | 3 | 4 | 4 | 3 | 4 | 3 | 3 | 4 | 3 | 4 |
| 27 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 4 | 3 | 2 | 3 | 4 | 4 | 5 | 4 | 3 | 4 | 3 | 3 | 3 | 3 | 4 | 3 | 3 |
| 28 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 3 | 4 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 4 | 3 | 4 | 4 | 4 | 4 |
| 29 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 4 | 5 | 5 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 4 | 5 | 5 | 5 |
| 30 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 3 | 4 | 5 | 5 | 5 | 4 | 5 | 4 | 3 | 3 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 5 | 5 | 5 | 5 |

Lampiran 10. Angket Pengujian
Functionality

Lampiran Lembar Pengujian *Functionality*

LEMBAR PENGUJIAN *FUNCTIONALITY*

PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB
DI SMK MUHAMMADIYAH 1 YOGYAKARTA

Nama : Adhio Agung Nugroho
Pekerjaan : RBD
Instansi : PT. Gamatechno Indonesia

Petunjuk Pengisian

Berilah tanda centang (✓) pada kolom **Ya** jika fungsi berjalan dengan benar dan

Tidak jika fungsi tidak berjalan dengan benar.

| No. | Fungsi | Lolos | |
|-----------|--------------------------------|-------|-------|
| | | Ya | Tidak |
| I. | User Admin | | |
| A | Akun | | |
| 1. | Login sebagai admin | ✓ | |
| 2. | Mengubah password admin | ✓ | |
| 3. | Lupa password | ✓ | |
| 4. | Logout | ✓ | |
| B | Mengelola Data Siswa | | |
| 5. | Melihat daftar siswa | ✓ | |
| 6. | Menambah data siswa | ✓ | |
| 7. | Mengedit data siswa | ✓ | |
| 8. | Mengubah foto siswa | ✓ | |
| 9. | Menghapus data siswa | ✓ | |
| 10. | Mencari data siswa | ✓ | |
| 11. | Export data siswa | ✓ | |
| C | Mengelola Member | | |
| 12. | Mencari data member | ✓ | |
| 13. | Melihat data peminjaman member | ✓ | |
| 14. | Menghapus member | ✓ | |
| D | Mengelola Data Buku | | |
| 15. | Melihat daftar buku | ✓ | |
| 16. | Menambah data buku | ✓ | |
| 17. | Mengedit data buku | ✓ | |
| 18. | Mengubah cover buku | ✓ | |
| 19. | Menghapus data buku | ✓ | |
| 20. | Mencari data buku | ✓ | |
| 21. | Import data buku | ✓ | |
| 22. | Export data buku | ✓ | |

| | | | |
|-------------|----------------------------------|---|--|
| E | Mengelola Data Absensi | | |
| 23. | Melihat daftar absensi | ✓ | |
| 24. | Menghapus data absensi | ✓ | |
| 25. | Mencari daftar absensi | ✓ | |
| 26. | Export data absensi | ✓ | |
| F | Mengelola Data Peminjaman | | |
| 27. | Melihat siswa yang meminjam | ✓ | |
| 28. | Mencari siswa yang meminjam | ✓ | |
| 29. | Melihat buku dipinjam | ✓ | |
| 30. | Mencari buku yang dipinjam | ✓ | |
| 31. | Filter berdasarkan tanggal | ✓ | |
| 32. | Laporan data peminjaman | ✓ | |
| G | Informasi Perpustakaan | | |
| 33. | Melihat informasi perpustakaan | ✓ | |
| 34. | Menambah informasi perpustakaan | ✓ | |
| 35. | Mengedit informasi perpustakaan | ✓ | |
| 36. | Menghapus informasi perpustakaan | ✓ | |
| 37. | Mencari informasi perpustakaan | ✓ | |
| H. | Data Statistik | | |
| 38. | Data statistik absensi | ✓ | |
| II. | User Siswa | | |
| A | Akun | | |
| 39. | Login sebagai siswa | ✓ | |
| 40. | Melihat data profil | ✓ | |
| 41. | Mengubah data profil | ✓ | |
| 42. | Mengubah password | ✓ | |
| 43. | Mengubah gambar profil | ✓ | |
| 44. | Lupa password | ✓ | |
| 45. | Logout | ✓ | |
| B | Peminjaman Buku | | |
| 46. | Melihat daftar buku | ✓ | |
| 47. | Mencari data buku | ✓ | |
| 48. | Meminjam buku | ✓ | |
| C | Pengembalian Buku | | |
| 49. | Melihat daftar buku | ✓ | |
| 50. | Mengembalikan buku | ✓ | |
| D | Informasi | | |
| 51. | Melihat informasi perpustakaan | ✓ | |
| E | Absensi | | |
| 52. | Absensi perpustakaan | ✓ | |
| III. | User Tamu | | |
| 53. | Melihat tentang perpustakaan | ✓ | |
| 54. | Melihat profil perpustakaan | ✓ | |
| 55. | Melihat Layanan Perpustakaan | ✓ | |
| 56. | Melihat daftar Buku | ✓ | |
| 57. | Pencarian Buku | ✓ | |

Terima kasih atas partisipasi saudara dalam menjawab butir-butir pernyataan dalam instrumen ini. Semoga instrumen ini dapat digunakan sebagaimana mestinya dalam pengumpulan data penelitian skripsi.

Komentar atau saran :

Security diperhatikan autentifikasi user diperbaiki,
alur sistem masih bisa diperbaiki, konsep
big data bisa ditambahkan. Tapi secara keseluruhan
sistem sudah berjalan dengan baik.

Yogyakarta, 18 Mei 2015

Responden,


(Adhika Agung Nugroho)

Lampiran 11. Dokumentasi



Gambar 84. Dokumentasi Pengujian *Usability* 1



Gambar 85. Dokumentasi Pengujian *Usability* 2



Gambar 86. Dokumentasi Pengujian *Usability* 3



Gambar 87. Dokumentasi Pengujian *Usability* 4