

Pengembangan Sistem Pengelolaan Ruang Baca Berbasis Web Dengan Menggunakan Django Framework (Studi Kasus: Ruang Baca Fakultas Ilmu Komputer Universitas Brawijaya)

Atikah Nur Rahimah¹, Denny Sagita Rusdianto², Mahardeka Tri Ananta³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya

Email: ¹atikah.nurahimah@gmail.com, ²denny.sagita@ub.ac.id, ³deka@ub.ac.id

Abstrak

Universitas Brawijaya memiliki satu perpustakaan pusat dan beberapa perpustakaan yang lebih kecil di setiap fakultasnya, termasuk perpustakaan yang berada di Fakultas Ilmu Komputer yang biasa disebut Ruang Baca. Saat ini, pengelolaan Ruang Baca dilakukan dengan menggunakan sebuah perangkat lunak *open source* yang bernama Senayan Library Management System (SLIMS). SLIMS dirasa kurang efektif karena sistem keanggotaannya tidak dapat terhubung dengan data mahasiswa yang dimiliki fakultas sehingga sulit untuk dilakukan pemeriksaan status mahasiswa yang akan melakukan transaksi. Selain itu, Ruang Baca tidak memiliki data koleksi khususnya skripsi untuk dilihat oleh mahasiswa secara *online* sehingga menyulitkan mahasiswa yang ingin mencari data diluar jam kerja Ruang Baca. Sebagai solusi, dikembangkanlah sebuah Sistem Pengelolaan Ruang Baca berbasis web. Sistem ini memiliki data anggota yang terintegrasi dengan data mahasiswa milik fakultas. Selain itu, sistem ini dapat digunakan oleh mahasiswa untuk melakukan pencarian data koleksi termasuk skripsi secara *online*. Penelitian ini diawali dengan proses rekayasa kebutuhan yang menghasilkan 62 kebutuhan fungsional dan 1 kebutuhan non fungsional. Kemudian, dilakukan perancangan sistem. Implementasi sistem dilakukan dengan menggunakan *framework* Django. Dari proses pengujian, didapatkan hasil 100% valid untuk pengujian unit, integrasi, dan validasi. Sedangkan untuk pengujian *usability*, didapatkan nilai 82.92 yang berarti Sistem Pengelolaan Ruang Baca masuk pada kategori 'A'.

Kata kunci: perpustakaan, pengelolaan, django web framework, python

Abstract

Brawijaya University has one central library and several smaller libraries in each of its faculties, including a library located in the Faculty of Computer Science which is commonly called a Reading Room. Currently, Reading Room management is done by using an open source software called Senayan Library Management System (SLIMS). The use of SLIMS is deemed not quite helpful because the membership system cannot be connected with the student database that is owned by the faculty which make it difficult to check the students's status when they are going to make transactions. In addition, the Reading Room does not have collection data, especially thesis, to be seen by students online, making it difficult for students who want to find data outside of the Reading Room's working hours. As a solution, a web-based Reading Room Management System was developed. This system has features for managing collection data and reading room transactions with member data that is integrated with the faculty. In addition, this system can be used by students to search collection data including thesis data online. This research begins with requirements engineering process that produces 59 functional requirements and 1 non-functional requirement. Then, the system design is carried out. The implementation is done using Django framework. From the testing process, the results are 100% valid for unit testing, integration, and validation. As for usability testing, the value of 82.92 means that the Reading Room Management System falls into the 'A'.

Keywords: library, management, django web framework, python

1. PENDAHULUAN

Universitas Brawijaya memiliki satu perpustakaan pusat, dan beberapa perpustakaan lainnya disetiap fakultas contohnya adalah perpustakaan di Fakultas Ilmu Komputer (FILKOM) yang disebut Ruang Baca. Koleksi di Ruang Baca FILKOM didominasi oleh topik koleksi yang kiranya dibutuhkan mahasiswa FILKOM seperti literatur pemrograman, basis data, dan lain-lain. Selain itu, Ruang Baca FILKOM juga menyediakan literatur berupa *hard copy* laporan PKL dan skripsi mahasiswa FILKOM untuk digunakan mahasiswa yang ingin mencari referensi penelitian.

Saat ini manajemen koleksi dan transaksi di Ruang Baca FILKOM UB telah menggunakan sebuah perangkat lunak *open source* bernama Senayan Library Management System (SLIMS). SLIMS membantu proses perekaman koleksi dan pengelolaan transaksi seperti peminjaman buku. Namun ternyata SLIMS masih jauh dari sistem yang diharapkan. Salah satu kekurangannya adalah sistem keanggotaan perpustakaan yang ada pada SLIMS tidak terintegrasi dengan data mahasiswa yang berada pada fakultas sehingga sulit dilakukan pengecekan status mahasiswa ketika ingin melakukan transaksi. Status mahasiswa dianggap penting karena mahasiswa yang dapat melakukan transaksi pada ruang baca hanyalah mahasiswa aktif FILKOM UB.

Seperti yang dikutip dari sebuah portal berita mahasiswa, pustakawan Ruang Baca FILKOM UB menyebutkan bahwa diharapkan agar koleksi-koleksi ruang baca baik buku, skripsi, laporan PKL atau lainnya dapat dilihat dalam sebuah sistem secara *online* (Kis & Fat, 2017). Hal itu disebabkan karena mahasiswa sudah tidak bisa meminjam skripsi dan dibawa pulang. Selain *hard copy* dari skripsi tersebut, tidak ada lagi data skripsi yang dapat mahasiswa lihat. Oleh karena itu, akan dikembangkan sebuah Sistem Pengelolaan Ruang Baca berbasis web. Sistem ini memiliki fitur untuk mengelola data koleksi dan data transaksi di ruang baca. Data keanggotaan sistem ini juga akan terhubung dengan data mahasiswa milik fakultas untuk memudahkan proses pengecekan status mahasiswa. Sistem ini juga memiliki fitur yang memungkinkan mahasiswa untuk melihat daftar koleksi baik buku ataupun skripsi secara online. Data skripsi yang diberikan juga mengandung abstrak skripsi yang dapat dibaca untuk referensi

terkait penelitian yang dilakukan pada skripsi tersebut.

Untuk memudahkan pengembangan sistem ini, sistem akan dikembangkan dengan menggunakan Django. Django merupakan sebuah *web framework* yang menggunakan bahasa pemrograman Python. Django dipilih karena memiliki fitur yang dapat mempermudah pengembangan sistem contohnya antarmuka admin yang otomatis untuk mempermudah proses manipulasi data pada table basis data. Proses manipulasi tabel basis data juga dapat dilakukan dengan mudah dengan *django migrations function*. Selain itu, Django dibangun dengan bahasa pemrograman Python, bahasa pemrograman tingkat tinggi yang mudah dibaca manusia.

Dengan adanya Sistem Pengelolaan Ruang Baca ini diharapkan petugas ruang baca dapat terbantu dalam proses pengelolaan data koleksi dan transaksi pada ruang baca. Mahasiswa juga diharapkan dapat terbantu untuk mencari data skripsi sebagai referensi penelitian kapanpun dan dimanapun.

2. TINJAUAN PUSTAKA

2.1. Perpustakaan

Perpustakaan adalah suatu unit kerja dari sebuah badan atau lembaga tertentu yang bertugas untuk mengatur bahan-bahan pustaka baik buku maupun yang bukan buku dan diatur secara sistematis menurut aturan tertentu untuk digunakan sebagai sumber informasi (Suhendar & Yaya, 2014). Undang-undang No.43 Tahun 2007 Pasal 1 ayat 1 menyatakan bahwa perpustakaan adalah institusi pengelola koleksi karya tulis, cetak, dan atau karya reka secara profesional dengan sistem yang berguna memenuhi kebutuhan pendidikan, penelitian, pelestarian, informasi, dan rekreasi bagi para pemustaka.

2.2. Django Web Framework

Django adalah sebuah *web framework* yang menggunakan bahasa pemrograman Python yang mendukung pembuatan *website* dengan konsep *rapid development* (Django Software Foundation, 2017). Wiki-Python menyatakan bahwa *framework* yang sekarang telah berada di versi 2.1 ini sebagai *web framework* paling populer dikalangan *high-level frameworks*. Karena menggunakan bahasa pemrograman Python yang telah memiliki banyak *library* siap

pakai, mengembangkan sistem dengan Django berarti dapat menggunakan *library* tersebut sesuai dengan keperluan pengembangan.

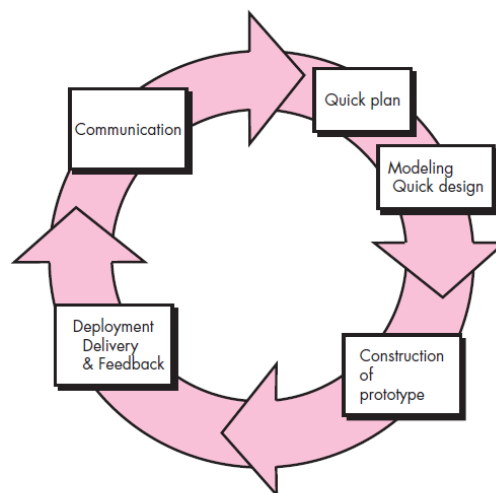
Django bertujuan untuk memudahkan pengembangan situs web dan basis data yang kompleks (Django Software Foundation, 2017). Salah satu keunggulan django adalah, *framework* ini telah merepresentasikan ORM (Object Relational Mapper) sehingga tidak perlu menyesuaikan *query* jika terjadi perubahan database yang digunakan. Berikut ini adalah beberapa keuntungan menggunakan Django:

1. *Object-Relational Mapping (ORM) Support.*
2. *Automatic Admin Interface.*
3. *Elegant URL Design.*
4. *Template System.*
5. *Cache system.*
6. *Internationalization.*
7. *A light weight web server for development and test.*

2.3 Model Prototype

Model pengembangan *Prototype* adalah model pengembangan perangkat lunak yang cocok digunakan untuk pengembangan sebuah sistem dimana kebutuhan-kebutuhannya sudah terdefiniskan secara umum namun tidak spesifik. Dalam hal ini, pengembang dan pemangku kepentingan dapat menyatukan pemahaman dengan lebih baik tentang seperti apa sistem akan dibangun (Pressman, 2010).

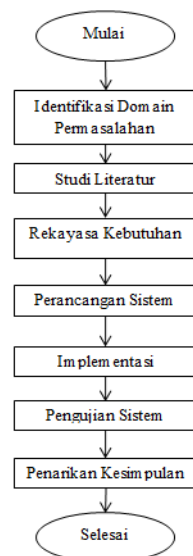
Biasanya, *prototype* dibangun dengan tujuan untuk mendefinisikan kebutuhan sistem yang lebih spesifik, untuk melakukan validasi terhadap kebutuhan sistem, atau untuk mendapatkan evaluasi terkait dengan *user interface* sistem tersebut. Proses ini akan berlangsung secara iteratif berdasarkan proses *trial-and-error*. Untuk menghindari iterasi yang berlebihan, pengembang perlu untuk menetapkan perjanjian awal dengan pemangku kepentingan dengan tujuan untuk mengefisiensikan waktu dan dana pengembangan sistem (Pressman, 2010). Akhir dari model *Prototype* adalah evaluasi terhadap *prototype* tersebut dimana *end users* akan mencoba menggunakan *prototype* tersebut untuk beberapa waktu sampai akhirnya dapat mendefinisikan hasil evaluasi termasuk eror atau kelalaian yang disebabkan oleh sistem. Hasil evaluasi sistem dapat digunakan sebagai acuan pengembangan sistem yang lebih lanjut (Somerville, 2011).



Gambar 1 The Prototype Paradigm

3. METODOLOGI PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini dimulai dari tahap awal yaitu identifikasi domain permasalahan hingga tahap akhir yaitu penarikan kesimpulan dan saran. Tahapan penelitian yang digunakan dalam penelitian ini digambarkan dalam Gambar 2.2.



Gambar 2 Tahapan Penelitian

3.1. Identifikasi Domain Permasalahan

Ruang Baca FILKOM UB dipilih menjadi objek penelitian berdasarkan pengalaman peneliti sebagai mahasiswa FILKOM UB. Namun untuk memperkuat alasan tersebut dilakukan wawancara dengan pustakawan Ruang Baca FILKOM UB yaitu bapak Drs. Syaifudin, M.Hum terkait dengan kegiatan pada Ruang Baca FILKOM UB dan permasalahan-

permasalahan yang dihadapi dalam melakukan kegiatan tersebut. Hasil wawancara digunakan untuk mengidentifikasi masalah yang akan digunakan sebagai dasar dilakukannya penelitian.

3.2. Studi Literatur

Setelah domain permasalahan teridentifikasi, selanjutnya adalah pengumpulan literatur yang mendukung dalam penyusunan kerangka berpikir peneliti. Kerangka berpikir dan literatur tersebut berguna untuk memberikan pemahaman lebih terkait teori-teori pendukung penelitian dan pengembangan Sistem Informasi Ruang Baca FILKOM UB.

3.3. Rekayasa Kebutuhan

Proses rekayasa kebutuhan dilakukan untuk mengolah permasalahan-permasalahan dan data-data yang sudah dikumpulkan untuk menghasilkan kebutuhan-kebutuhan pengguna terhadap solusi perangkat lunak yang ditawarkan. Proses rekayasa kebutuhan terbagi atas beberapa proses sebagai berikut.

1. **Elisitasi dan Analisis Kebutuhan**
Proses elisitasi kebutuhan dilakukan melakukan wawancara tahap dua yang dilakukan secara terstruktur dengan bapak Syaifudin, Drs., M.Hum kebutuhan-kebutuhan pengguna akan sistem yang akan dikembangkan. Kemudian dilakukan analisis kebutuhan terkait solusi yang ditawarkan berdasarkan kebutuhan-kebutuhan pengguna. Proses tersebut meliputi analisis kebutuhan sistem yaitu apa saja yang harus disediakan oleh sistem untuk membantu mengatasi permasalahan pada Ruang Baca FILKOM UB baik fungsional maupun non-fungsional, dan siapa saja aktor yang terlibat dalam penggunaan sistem nantinya.
2. **Spesifikasi Kebutuhan**
Dari hasil analisa dan studi literatur, kebutuhan-kebutuhan yang sudah terdefiniskan dispesifikkan menjadi lebih detail guna menghindari kesalahpahaman dari pihak pengembang dan pengguna. Hasil spesifikasi ini juga akan digunakan sebagai acuan perancangan dan implementasi sistem kedepannya.
3. **Manajemen Kebutuhan**
Proses manajemen kebutuhan dilakukan dengan melakukan kontrol terhadap setiap kebutuhan yang telah didefinisikan dan

dispesifikkan sebelumnya. Proses ini juga termasuk proses dokumentasi kebutuhan yang akan dimuat dalam bentuk table kebutuhan dan table aktor. Pemberian kode pada setiap kebutuhan baik fungsional maupun non-fungsional juga dilakukan agar lebih mudah dilakukan identifikasi, dan *tracking* baik dari sumber (table kebutuhan) ke proses perancangan ataupun sebaliknya.

4. Pemodelan Kebutuhan

Pemodelan kebutuhan dilakukan secara *Object Oriented* dengan menggunakan diagram UML yaitu *Use Case Diagram* untuk menjelaskan sistem yang dikembangkan berdasarkan sudut pandang aktor dan juga *Use case Scenario* untuk menjelaskan secara lebih detail tentang *Use Case* seperti tahapan-tahapan yang dapat dilakukan oleh aktor dalam menggunakan sistem untuk memenuhi tujuan tertentu.

3.4. Perancangan Sistem

Proses perancangan akan dilakukan setelah model kebutuhan telah disepakati oleh *stakeholders*. Proses perancangan baik dari perancangan komponen ataupun arsitektur sistem akan menghasilkan rancangan arsitektur sistem, *sequence diagram*, perancangan basis data yaitu *Physical* dan *Conceptual Data Model*, diagram kelas, rancangan komponen sistem, dan rancangan antarmuka. Setelah rancangan antarmuka selesai, dilakukan pengecekan *prototype* sistem oleh pemangku kepentingan untuk kemudian dievaluasi. Implementasi dapat dilakukan ketika *prototype* telah disetujui oleh pemangku kepentingan.

3.5. Implementasi Sistem

Tahap implementasi pada penelitian ini terbagi atas tiga proses yaitu Implementasi Basis Data, Implementasi Komponen, Implementasi Antarmuka. Implementasi sistem akan menggunakan bahasa pemrograman Python dengan *framework* Django dengan menggunakan arsitektur *model-template-view* dari Django yang mengadaptasi pola perancangan *model-view-controller*. Sedangkan untuk basis data yang digunakan adalah MySQL.

3.6. Pengujian

Tahapan pengujian akan dilakukan dengan dua strategi yaitu *Black box* untuk *Validation*

Testing dimana sistem diuji sebagai satu kesatuan dan pengujian dilakukan berdasarkan aksi dari pengguna dan dievaluasi berdasar output yang didapatkan oleh pengguna dari sistem. Pengujian validasi menggunakan *Software Requirements Specification* (SRS) sebagai panduan dalam memilih *validation-test criteria*-nya (Pressman, 2010). Selain pengujian *Black box*, pengujian juga akan dilakukan dengan strategi *White box* dengan menggunakan *Control Flow Graph* untuk menghitung kompleksitas algoritme yang digunakan menggunakan sebuah metode yang disebut *McCabe's Cyclomatic Complexity* pada tahap pengujian unit dan integrasi. Pengujian *Usability* juga akan dilakukan untuk mengetahui seberapa mudah sistem digunakan.

3.7. Kesimpulan dan Saran

Isi dari kesimpulan adalah hasil analisis dan pengujian yang dilakukan pada sistem dengan tujuan untuk menjawab rumusan masalah pada awal penelitian. Selain kesimpulan, saran dari penulis juga diberikan sebagai masukan terhadap penelitian selanjutnya dan sebagai acuan pengembangan sistem yang lebih lanjut.

4. REKAYASA KEBUTUHAN

4.1. Elisitasi Kebutuhan dan Analisis Kebutuhan

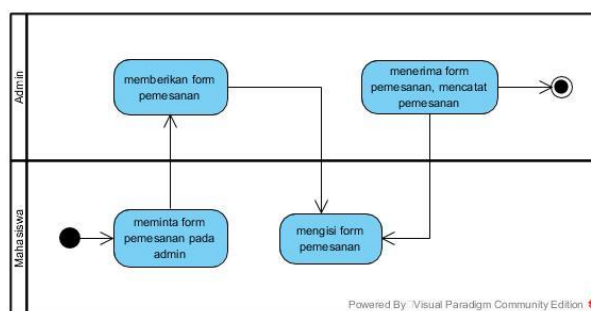
Dari hasil elisitasi kebutuhan, diketahui permasalahan-permasalahan sebagai berikut:

1. Sistem pengelolaan yang digunakan saat ini memiliki kekurangan yaitu terlalu rumit dan tidak sesuai dengan kebutuhan di Ruang Baca. Selain itu, data anggota pada sistem tidak terhubung dengan pangkalan data mahasiswa yang dimiliki fakultas sehingga sulit dilakukan pengecekan status mahasiswa.
2. Mahasiswa hanya dapat mengakses informasi koleksi pada saat jam kerja Ruang Baca.

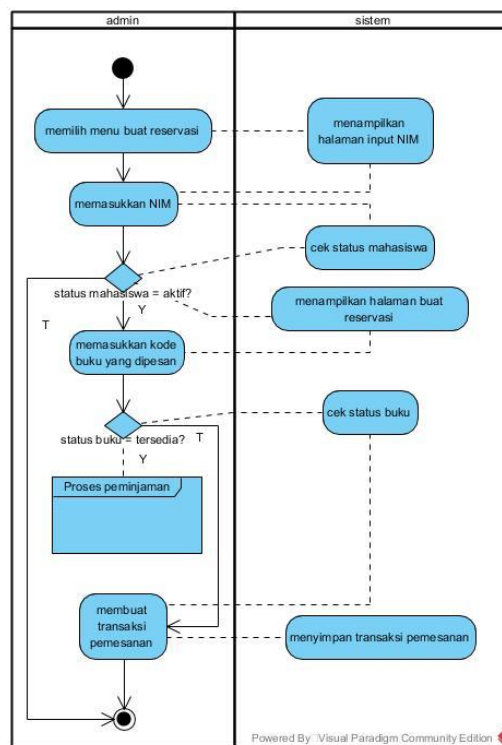
Proses elisitasi juga menghasilkan proses bisnis *As-Is* yang merupakan tahapan yang dilakukan pada satu proses bisnis. Proses bisnis yang didapatkan adalah proses bisnis transaksi pemesanan, peminjaman, dan pengembalian koleksi. Proses bisnis untuk transaksi pemesanan dapat dilihat pada gambar 3.

Pada proses analisis kebutuhan dilakukan peninjauan kembali permasalahan yang mungkin terjadi pada proses bisnis *as-is*.

Kemudian permasalahan yang ditemukan, disusun sebuah proses bisnis baru yaitu proses bisnis *to-be*. Analisis juga dilakukan dari sisi *stakeholder* yaitu bapak Syaifudin. Hasil dari analisis kebutuhan diketahui bahwa proses pengecekan status mahasiswa saat ini hanya menggunakan Kartu Tanda Mahasiswa (KTM) sehingga tidak bisa dilakukan proses pengecekan status mahasiswa. Selain itu, perekaman data masih menggunakan kertas formulir sehingga perekaman data melalui sistem akan mengurangi penggunaan kertas dan mempersingkat waktu transaksi. Proses bisnis *to-be* untuk transaksi pemesanan dapat dilihat pada gambar 4.



Gambar 3. Proses Bisnis *As-Is* transaksi pemesanan



Gambar 4. Proses Bisnis *To-Be* transaksi pemesanan.

4.2. Identifikasi Aktor

Identifikasi Aktor dilakukan dengan tujuan untuk mengetahui siapa saja pemangku kepentingan pada sistem ini dan untuk mengidentifikasi kebutuhan para pemangku kepentingan tersebut dengan sistem yang akan dikembangkan. Berikut merupakan aktor dalam sistem ditunjukkan pada tabel 1.

Tabel 1. Identifikasi Aktor

| Aktor | Deskripsi |
|-----------------|---|
| Visitor | Visitor adalah semua orang yang mengakses sistem. |
| Admin | Admin adalah pustakawan yang bekerja di ruang baca. |
| Staff | Staff adalah orang yang bertugas di Ruang Baca untuk membantu pekerjaan pustakawan (mahasiswa/siswa magang). |
| Anggota | Anggota adalah seluruh mahasiswa FILKOM UB yang berstatus aktif. |
| Modul Mahasiswa | Modul Mahasiswa adalah modul penyedia informasi terkait dengan mahasiswa FILKOM UB. Modul Mahasiswa menyediakan informasi terkait dengan NIM, nama, jurusan, prodi, email, nomor telepon, dan status mahasiswa. |
| Modul Skripsi | Modul Skripsi adalah modul yang menyediakan informasi terkait dengan skripsi seorang mahasiswa baik yang telah diajukan menjadi proposal atau yang belum. Modul skripsi menyediakan informasi terkait judul, dosen pembimbing satu dan dua (apabila ada), status dari skripsi tersebut, abstrak, beserta data mahasiswa yang mengajukan berupa nama, nim, jurusan, dan program studi. |

4.3. Kebutuhan Sistem

Terdapat dua jenis kebutuhan sistem, yakni kebutuhan fungsional dan non-fungsional. Terdapat 62 kebutuhan fungsional dan 1 kebutuhan non-fungsional. Kebutuhan fungsional sistem dimodelkan menggunakan sebuah *use case diagram*.

5. PERANCANGAN

5.1. Perancangan Arsitektur

Perancangan arsitektur sistem menggambarkan interaksi antar komponen-komponen di dalam Sistem Informasi Ruang Baca dan komunikasi yang terjadi terhadap objek di luar sistem. Arsitektur Sistem Informasi Ruang Baca FILKOM UB terdiri atas beberapa komponen. Diagram arsitektur tersebut menggunakan model *MTV-Architecture* dari *django framework*. Komponen dalam sistem terbagi atas *Models*, *Templates*, *Views* yang saling terhubung satu sama lain. *Views* bertugas untuk menangani *request* dan *response* web dimana *views* akan menerjemahkan permintaan dari web dan mengembalikannya dalam bentuk apapun (halaman HTML, kode error, atau XML). *Models* akan bertugas sebagai media perantara antara sistem dengan basis datanya, seluruh komunikasi dengan basis data akan melalui *models*. Sedangkan *templates* berisi kode HTML yang mengatur tampilan halaman web. Terdapat dua modul yang akan berkomunikasi dengan sistem ini. Dua modul tersebut adalah Modul Mahasiswa dan Modul Skripsi.

5.2. Sequence Diagram

Sequence Diagram akan menjelaskan interaksi pengiriman pesan antar objek dalam sistem dalam menjalankan suatu fungsionalitas. *Sequence diagram* dibuat berdasarkan *use case scenario* yang telah disusun pada proses Rekayasa Kebutuhan. *Sequence Diagram* ini juga akan berpengaruh pada penulisan *pseudocode* fungsi.

5.3. Perancangan Basis Data

Basis Data dilakukan dengan membuat diagram *Conceptual Data Model* sebagai representasi dari suatu tabel basis data beserta struktur dan juga relasi antar tabelnya dan *Physcal Data Model* sebagai penggambaran tabel basis data, relasi, dan panjabaran atribut dalam tabelnya. *Physcal Data Model* akan

dijadikan acuan dalam mengimplementasikan basis data. Perancangan basis data menghasilkan rancangan dari 12 tabel basis data yaitu koleksi, mahasiswa, notifikasi, *feedback*, akun pengguna, topik koleksi, tipe koleksi, sumber koleksi, bahasa koleksi, denda, transaksi pemesanan, dan transaksi peminjaman.

5.4. Perancangan Kelas

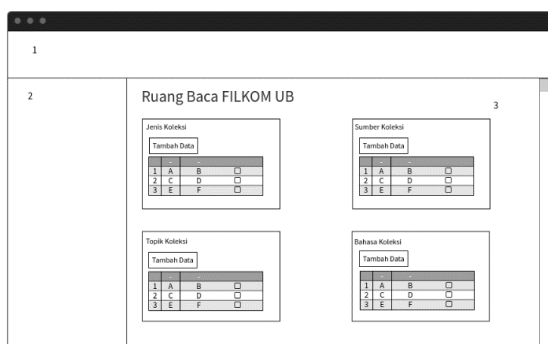
Perancangan kelas dilakukan dengan menggambarkan semua objek pada Sistem Informasi Ruang Baca beserta relasi antar objeknya ke dalam diagram kelas. Diagram kelas juga akan memuat *mandatory* dari relasi antar objek tersebut dan hubungan antar suatu kelas dengan kelas yang lain.

5.5. Perancangan Komponen

Perancangan komponen dibuat dengan tujuan sebagai acuan dalam mengimplementasikan suatu fungsi yang akan bekerja pada sistem. Perancangan komponen memuat proses-proses yang akan dilakukan dalam suatu fungsi untuk mencapai tujuan tertentu. Perancangan komponen menghasilkan *pseudocode* suatu fungsi.

5.6. Perancangan Antarmuka

Perancangan Antarmuka dilakukan dengan membuat gambaran mengenai tampilan web yang nantinya akan digunakan sebagai acuan untuk mengimplementasikan suatu halaman web dengan HTML. Perancangan antarmuka didesain menggunakan tampilan web dari perangkat desktop. Gambar 5 merupakan gambaran perancangan antarmuka dari halaman ‘Daftar Jenis Koleksi’.



Gambar 5. Perancangan Antarmuka Halaman ‘Daftar Jenis Koleksi’

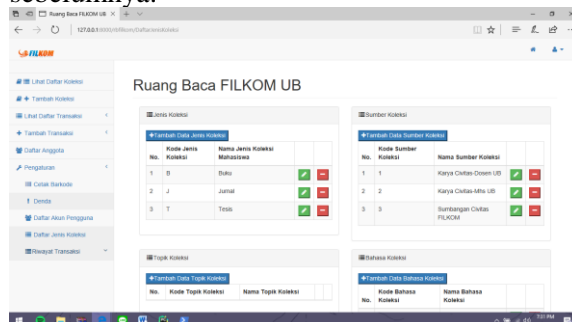
Prototype sistem berupa rancangan antarmuka sistem beserta rancangan basis data sistem diberikan kepada bapak Syaifuddin selaku *stakeholder* dan kemudian di evaluasi.

Dari hasil evaluasi, *stakeholder* tidak memberikan perubahan terhadap rancangan sistem sehingga proses perancangan berhenti pada iterasi pertama dan siap untuk diimplementasikan.

5. IMPLEMENTASI

Proses implementasi yang dilakukan ada 3 yaitu implementasi basis data, implementasi antarmuka, dan implementasi kode program. Implementasi basis data dibuat berdasarkan diagram *Physical Data Model* yang telah dibuat sebelumnya. Implementasi basis data dilakukan dengan membuat basis data berisi tabel sesuai dengan rancangan basis data. Nama basis data pada sistem ini adalah basis data ‘rbfilkom’ yang dibangun menggunakan MySQL.

Implementasi Antarmuka mengacu pada rancangan antarmuka sistem, gambar 7 menunjukkan contoh implementasi antarmuka dari halaman ‘daftar jenis koleksi’. Sedangkan implementasi kode program mengacu pada perancangan komponen yang dilakukan sebelumnya.



Gambar 7. Implementasi Antarmuka Halaman ‘Daftar Jenis Koleksi’

6. PENGUJIAN SISTEM

Pengujian yang dilakukan pada penelitian ini adalah pengujian unit, integrasi, pengujian validasi, dan pengujian *usability*. Pengujian unit dilakukan dengan pendekatan *white-box testing* dengan menggunakan McCabe’s *Cyclomatic Complexity* dengan bantuan *Flow Graph*. Pengujian integrasi *bottom-up* dengan bantuan *class driver*. Pengujian integrasi juga dilakukan dengan pendekatan *white-box*. Hasil dari kedua pengujian tersebut adalah 100% valid dengan nilai V(G) pada masing-masing fungsi pada pengujian unit dan pada pengujian integrasi adalah kurang dari 10 yang menandakan bahwa algoritma yang digunakan dapat dikatakan baik dan *less error prone* sehingga tidak perlu dilakukan perancangan ulang.

Pengujian validasi dilakukan dengan pendekatan *black-box* testing dan menghasilkan 100% valid untuk semua kasus uji. Pengujian *usability* dilakukan dengan memberikan kuisioner SUS dengan skala *likert* kepada 6 responden (1 admin Ruang Baca, 5 mahasiswa FILKOM UB) dan sebuah instruksi untuk menjalankan beberapa tugas dengan menggunakan Sistem Pengelolaan Ruang Baca. Hasil dari kuisioner dikonversikan kemudian dihitung total untuk tiap responden dan dihitung rata-ratanya. Rata-rata nilai berada di angka 82.92 yang berada di kategori 'A' yang berarti pengguna menyukai Sistem Pengelolaan Ruang Baca FILKOM UB dan cenderung akan merekomendasikannya.

8. KESIMPULAN

Kesimpulan pada penelitian ini digunakan untuk menjawab rumusan masalah. Dari hasil penelitian ini disimpulkan bahwa:

1. Dari 3 buah proses bisnis (peminjaman, pemesanan, dan pengembalian koleksi) didiefinisikan 2 fitur yaitu peminjaman yang terdiri atas fungsi tambah transaksi peminjaman, ubah transaksi peminjaman (pengembalian), dan fungsi hapus transaksi. Lalu selanjutnya adalah fitur pemesanan koleksi yang terdiri dari fungsi tambah transaksi, ubah transaksi, dan hapus transaksi. Selain itu, terdapat 51 fungsi lainnya, 62 buah kebutuhan fungsional, dan 1 kebutuhan non-fungsional. Dari proses tersebut juga dihasilkan sebuah *use case diagram* dan *use case scenario*.
2. Dari proses perancangan yang dilakukan, didapatkan hasil berupa rancangan arsitektur, sequence diagram, diagram perancangan basis data yang terdiri dari *Conceptual Data Model*, *Logical Data Model*, dan *Physical Data Model*. Proses perancangan juga menghasilkan diagram kelas, *pseudocode*, dan rancangan antarmuka sistem.
3. Tahap implementasi menghasilkan basis data yang akan digunakan yaitu basis data 'rbfilkom', implementasi antarmuka sistem, dan sistem yang siap diujikan. Implementasi dilakukan dengan menggunakan *framework* Django dengan bahasa pemrograman Python. Diketahui Django dapat mempermudah pengembangan sistem karena memiliki

antarmuka admin yang dapat digunakan untuk memasukkan data kedalam tabel basis data selain itu Django juga menyediakan *library packages* yang sangat banyak membantu pengembangan sistem. Django juga secara otomatis membuat tabel basis data *user*, *session*, *permission*, dan *group* untuk proses autentikasi pengguna.

4. Dilakukan 4 jenis pengujian yaitu pengujian unit, integrasi, validasi, dan pengujian *usability*. Dari hasil pengujian unit, disimpulkan bahwa seluruh fungsi yang diujikan bernilai valid dan memiliki angka *cyclomatic complexity* dibawah 10 yang berarti rancangan algoritma dari fungsi-fungsi yang diujikan sudah baik. Dari hasil pengujian integrasi juga disimpulkan bahwa fungsi yang diujikan berintegrasi dengan baik dan valid. Hasil dari pengujian validasi terhadap 50 fungsi adalah 100% valid. Kemudian dari hasil pengujian *usability* menggunakan SUS, didapatkan nilai 82.92 yang berarti Sistem Pengelolaan Ruang Baca masuk pada kategori A dimana pengguna menyukai sistem ini dan cenderung akan merekomendasikannya.

9. DAFTAR PUSTAKA

- Django Software Foundation, 2017. *Django Documentation - Design Philosophies*. [Online] Available at: <https://docs.djangoproject.com/en/2.0/misc/design-philosophies/> [Accessed 27 February 2019].
- Kis & Fat, 2017. *Display*. [Online] Available at: <http://display.ub.ac.id/news/tak-boleh-dipinjam-ruang-baca-dipenuhi-mahasiswa-skripsi/> [Accessed 28 August 2018].
- Pressman, R. S., 2010. *Software Engineering : A Practitioners Approach 7th Edition*. New York: McGraw-Hill.
- Somerville, I., 2011. *Software Enginerring*. New york: Addison-Wesley.
- Suhendar & Yaya, 2014. *Pedoman Penyelenggaran Perpustakaan Sekolah*. Bandung: Remaja Rosdakarya.