

Pengembangan Sistem Aplikasi Persuratan Elektronik Berbasis Web di Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya

Habib Trizaka¹, Denny Sagita Rusdianto², Adam Hendra Brata³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹habibtrizaka@outlook.com, ²denny.sagita@ub.ac.id, ³adam@ub.ac.id

Abstrak

Surat telah menjadi media komunikasi tertulis yang penting untuk memenuhi kebutuhan pada banyak organisasi ataupun perusahaan. Paradigma yang ada saat ini adalah kegiatan mengelola surat membutuhkan ketelitian yang cukup tinggi sehingga memakan waktu yang cukup lama. Hal serupa terjadi pula di Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya yang hingga saat ini masih menggunakan Microsoft Excel sebagai alat bantu pencatatan semua surat yang berkorespondensi di lingkungan FILKOM. Pengelolaan surat seperti ini membuat pendistribusian surat menjadi kurang efektif dikarenakan belum adanya integrasi persuratan antarunitnya. Seringkali, disposisi surat terhenti di satu pihak dikarenakan pihak terkait tidak di tempat, atau pihak terkait tidak memberikan informasi disposisi yang telah ditanganinya. Masalah tersebut dapat ditangani dengan menggunakan sistem aplikasi berbasis web yang terintegrasi di tiap unitnya. Dengan cara ini, disposisi surat dan pemantauannya dapat dengan mudah dilakukan tanpa harus mempertimbangkan waktu dan tempat. Model proses *waterfall* digunakan dalam mengembangkan perangkat lunak. Pada tahap rekayasa kebutuhan, didapati sebanyak 33 kebutuhan fungsional dan 1 kebutuhan non-fungsional. Mengacu pada hasil rekayasa kebutuhan, sistem dirancang untuk kemudian diimplementasikan pada tahap selanjutnya. Pengujian yang dilakukan yakni pengujian unit, pengujian integrasi, dan pengujian validasi yang menghasikan angka 100% pada semua skenario uji. Dan dari pengujian *compatibility*, didapati bahwa sistem bekerja tanpa ada masalah pada 73% dari 11 versi peramban yang diujikan.

Kata kunci: sistem pengelolaan surat, rekayasa perangkat lunak, pengembangan web

Abstract

Letter has become a written communication media that is important to fulfill lots of organizations or companies needs. The current thought about organizing letters is that it needs high level of precision which is take too long to be done. This situation is similar to what happened in Faculty of Computer (FILKOM) University of Brawijaya. They have been using Microsoft Excel to manage all of the corresponding letters until this time. In this way, the distribution of letters has become less effective knowing that there is no integration between units/divisions. Oftentimes, the distributed letter is stopped because of some reasons. It could be that the receiver is not available at that time, or there is no information whether the letter has already been handled or not. These kinds of problems can be solved by developing a web-based letter management system where each unit is integrated to one another. In this way, the distributed letter and its monitoring can be easily done without considering place and time. Waterfall model process is used by the author in developing the software. In requirement engineering phase, there are found 33 functional requirements and 1 non-functional requirements. Based on the requirement engineering result, system is designed to be implemented in the next phases. The testings performed are unit testing, integration testing, and validation testing that result 100% valid in all test scenarios. And based on the compatibility testing, it is known that the system worked with no problems occurred in 73% of 11 versions of tested browsers.

Keywords: *letter management system, software engineering, web development*

1. PENDAHULUAN

Surat telah menjadi media komunikasi tertulis yang penting untuk memenuhi kebutuhan pada banyak organisasi ataupun perusahaan. Kebutuhan tersebut dapat berupa perintah, pengingat, permohonan dan sebagainya yang dianggap penting dan perlu diarsipkan. Peranan surat pada organisasi atau perusahaan menjadi penting dalam membentuk koordinasi kerja yang baik, baik dalam satu bagian/divisi, atau antarbagian/divisi, atau luar organisasi atau perusahaan tersebut (Nuraida, 2008). Peranan surat yang bersifat penting tersebut harus diimbangi dengan pengelolaannya yang baik pula. Pengelolaan surat yang baik akan memberikan dampak positif terhadap organisasi atau perusahaan yang menjalankannya. Sebaliknya, kegagalan dalam mengelola surat dapat menyebabkan kerugian bagi organisasi atau perusahaan, salah satunya yang fatal seperti hilangnya surat beserta informasinya.

Lazimnya, kegiatan pengelolaan surat berawal dari penerimaan dan pencatatan surat masuk dan keluar hingga penerusan dan pengarsipannya (Arifin dan Mustakim, 2005). Tahapan tersebut merupakan tahapan umum yang biasa mendasari prosedur dalam tata pengelolaan surat. Paradigma yang ada saat ini adalah semua tahapan tersebut membutuhkan ketelitian yang cukup tinggi sehingga memakan waktu yang cukup lama dalam melakukan pengelolaan surat. Sebagai contoh, pendistribusian surat atau disposisi harus menggunakan lembar disposisi yang diisi dengan ditulis atau diketik secara manual. Kesalahan dalam penulisan nomor indeks atau kode-kode tertentu yang ada pada nomor surat dapat membuat surat yang didistribusikan menjadi sulit untuk dilacak.

Hal serupa terjadi pula di Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya. Unit Persuratan FILKOM, sebagai unit pengelola surat di FILKOM hingga saat ini masih menggunakan Microsoft Excel sebagai alat bantu pencatatan semua surat yang berkorespondensi di lingkungan FILKOM. Dengan kondisi seperti ini, permasalahan seperti kesalahan penulisan kode atau kesalahan pengarsipan bisa saja terjadi mengingat tidak adanya otomatisasi dalam mengelola surat. Tidak adanya otomatisasi ini membuat kerentanan terjadinya *human error* pada titik-titik penting dalam prosedur pengelolaan surat

menjadi tinggi.

Pengelolaan secara manual, berarti belum ada integrasi persuratan antarunit di lingkungan FILKOM. Belum adanya integrasi ini membuat pendistribusian atau disposisi surat menjadi kurang efektif. Penggunaan lembar disposisi dan kartu kendali sebagai sarana untuk mengetahui informasi keberadaan surat harus mengalami pengoperan ke pihak-pihak terkait setiap adanya disposisi. Seringkali, disposisi surat terhenti di satu pihak dengan berbagai penyebab, salah satunya adalah pihak yang mendapat disposisi tidak berada di tempat, sehingga instruksi tidak dapat dilaksanakan dengan segera. Akibatnya, penanganan surat menjadi terhambat. Penyebab yang lain adalah tidak adanya informasi mengenai disposisi yang telah selesai ditindaklanjuti dari pihak terkait. Hal ini menunjukkan bahwa status surat yang didisposisikan masih belum bisa dimonitor secara langsung, apakah sudah diterima, dibaca, ditindaklanjuti atukah belum tersentuh sama sekali. Akibatnya, proses bisnis menjadi kurang efektif dikarenakan perlu adanya pemberitahuan atau komunikasi secara langsung terlebih dahulu mengenai informasi status disposisi surat tersebut.

Dari hasil wawancara yang telah dilakukan kepada pihak persuratan FILKOM, didapati bahwa pengantisipasi permasalahan tersebut telah dilakukan sebelumnya oleh Universitas Brawijaya dengan mengembangkan sebuah sistem layanan persuratan guna memudahkan penanganan surat di lingkungannya, termasuk FILKOM. Adalah SIAS (Sistem Informasi Administrasi Surat) merupakan sistem layanan yang diharapkan dapat mengatasi masalah persuratan di Universitas Brawijaya. Hanya saja, terdapat beberapa ketidaksesuaian proses bisnis sistem dengan prosedur yang ada di FILKOM, salah satunya surat yang dikelola di FILKOM hanya ditangani oleh satu bagian saja yaitu persuratan. Sedangkan, pengelolaan surat menggunakan SIAS dikhususkan hanya untuk penanganan surat dalam satu unit, bukan seluruh fakultas. Hal inilah yang membuat SIAS tidak dapat mengakomodir proses bisnis yang berjalan di FILKOM dengan lebih baik.

Beranjak dari masalah-masalah yang telah dijabarkan sebelumnya, pengelolaan surat seharusnya bisa lebih mudah dilakukan dengan menggunakan sistem aplikasi berbasis web yang terintegrasi di tiap bagian/divisinya. Penerapan basis web pada pengembangan sistem dipilih berdasarkan pada kemudahan pengguna dalam

mengakses sistem dari berbagai perangkat dengan sistem operasi dan peramban yang berbeda-beda. Kemudahan ini memungkinkan disposisi surat dan pemantauannya dapat dengan mudah dilakukan tanpa harus mempertimbangkan waktu dan tempat. Dengan menggunakan sistem, kesalahan dalam menuliskan penomoran surat atau bahkan pengarsipan surat juga dapat diminimalisir, maka kemungkinan terjadinya penggandaan nomor surat atau kesalahan dalam mengarsipkan surat ke map yang tidak seharusnya pun dapat dihindari. Sehingga, dengan berbagai keuntungan yang ditawarkan tersebut, peneliti percaya bahwa penggunaan sistem dalam mengelola surat dapat membuat pengelolaan surat di FILKOM menjadi lebih mudah dan terstruktur.

2. LANDASAN KEPUSTAKAAN

Pada tahap ini dijabarkan mengenai literatur-literatur yang mendukung dilakukannya penelitian. Beberapa literatur yang digunakan berkaitan dengan pengelolaan surat, Fakultas Ilmu Komputer Universitas Brawijaya, dan pengembangan perangkat lunak berorientasi objek.

2.1. Pengelolaan Surat

Pengelolaan surat sejatinya merupakan kegiatan yang dimulai dari penerimaan dan pencatatan surat masuk dan keluar sampai dengan penerusan dan pengarsipan surat (Arifin dan Mustakim, 2015). Dalam melakukan pengelolaan surat, ada baiknya memperhatikan prosedur yang telah ditetapkan di masing-masing organisasi atau perusahaan. Hal ini bertujuan untuk menghindari terjadinya kesalahan dalam mengelola surat.

Hal yang sering ditemui dalam pengelolaan surat adalah penggunaan buku agenda atau kartu kendali untuk melakukan pencatatan surat, baik surat masuk maupun keluar. Tujuan utamanya adalah pelacakan surat dapat dengan mudah dilakukan. Kartu kendali atau buku agenda akan memberikan pengkodean tertentu untuk mengetahui dimana surat diarsip, atau dimana surat sedang di distribusikan.

2.2. Pengelolaan Surat di Fakultas Ilmu Komputer Universitas Brawijaya

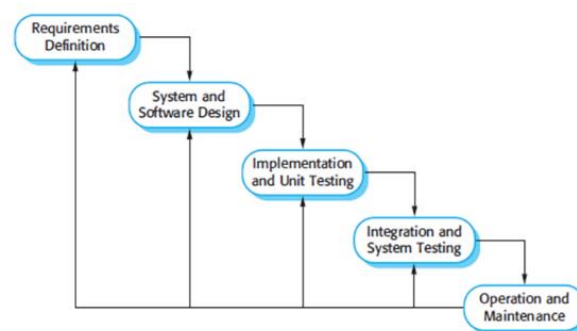
Pengelolaan surat di Fakultas Ilmu Komputer (FILKOM) dikelola oleh bagian Persuratan FILKOM yang merupakan pintu

Fakultas Ilmu Komputer, Universitas Brawijaya

masuk dan keluar semua surat yang berkorespondensi di lingkungan FILKOM. Pada praktiknya, pengelolaan surat mengacu pada *Standard Operational Procedure (SOP)* yang berlaku di FILKOM. Terdapat tiga SOP dalam menjalankan pengelolaan surat di FILKOM, yakni SOP surat keluar, SOP surat masuk, dan SOP peminjaman surat. Semua SOP tersebut berlaku bagi seluruh *sivitas akademika* di lingkungan FILKOM.

2.3. Model Waterfall

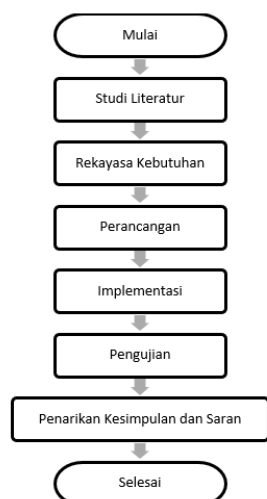
Waterfall merupakan salah satu pendekatan dalam SDLC (*Software Development Life Cycle*) yang digunakan untuk membangun perangkat lunak. Pendekatan ini sering disebut sebagai pendekatan yang kaku karena perubahan kebutuhan dapat menyebabkan kebingungan dalam pengerjaan pembangunan perangkat lunak (Pressman, 2010). Kendati demikian, pendekatan linier seperti *Waterfall* akan berguna ketika semua kebutuhan, tujuan, serta solusi telah didefinisikan dengan jelas. Gambar 1 menunjukkan tahapan yang harus dilalui dalam menggunakan model *Waterfall*. Pada penelitian ini, tidak dilakukan tahap operasi dan pemeliharaan (*Operation and Maintenance*) karena memerlukan kontrak lebih lama antara pengembang dan konsumen terkait pengembangan sistem lebih lanjut.



Gambar 1. Model *Waterfall*
Sumber: Somerville (2011)

3. METODOLOGI PENELITIAN

Langkah kerja digunakan sebagai pedoman dalam melakukan penelitian. Alur yang ditunjukkan pada Gambar 2 merupakan langkah-langkah yang harus dilalui peneliti selama penelitian berlangsung.



Gambar 2. Langkah Kerja Metodologi Penelitian

Langkah kerja yang dilakukan dalam penelitian ini diantaranya:

1. **Studi Literatur**
Studi literatur ditujukan untuk memahami teori-teori serta metode yang digunakan sebagai landasan dilakukannya penelitian.
2. **Rekayasa Kebutuhan**
Elisitasi kebutuhan dilakukan pada tahap ini melalui kegiatan wawancara kepada pihak persuratan FILKOM sebagai divisi pengelola surat. Hal ini bertujuan untuk memahami serta menganalisis proses bisnis yang sedang berjalan dan yang akan diterapkan dalam mengembangkan sistem. Hasil dari analisis proses bisnis dimodelkan menggunakan BPMN versi 2.0. Mengacu pada hasil tersebut, dilakukan identifikasi aktor serta kebutuhan sistem yang dimodelkan menggunakan *use case diagram* beserta skenarionya. Analisis data juga dilakukan untuk mengetahui entitas-entitas yang terlibat pada pengembangan sistem dengan memodelkan ERD (*Entity Relationship Diagram*).
3. **Perancangan**
Sistem dirancang menggunakan pendekatan OOD (*Object-oriented Design*). Perancangan arsitektur dilakukan dengan mengacu pada struktur kerangka kerja Yii Framework yang mengusung konsep MVC (*Model-View-Controller*). Perancangan data dimodelkan menggunakan PDM (*Physical Data Model*). *Sequence diagram* dan *class diagram* juga dimodelkan pada tahap ini. Perancangan yang lainnya yakni algoritme menggunakan *pseudocode* serta antarmuka dengan *wireframe*.
4. **Implementasi**

Pada tahap ini, dijelaskan spesifikasi pengembangan sistem. Basis data dibangun menggunakan *query DDL (Data Definition Language)* dengan MySQL sebagai DBMS (*Database Management System*)-nya. Sistem diimplementasikan menggunakan Bahasa PHP dengan OOP (*Object-oriented Programming*) menggunakan kerangka Yii Framework versi 2.0.

5. Pengujian

Hasil implementasi sistem kemudian diuji dengan menerapkan metode *white-box* untuk tingkat pengujian unit dan integrasi, dan *black-box* untuk tingkat validasi. Diuji pula kebutuhan non-fungsional *compatibility* dengan memanfaatkan *tool SortSite*.

6. Penarikan Kesimpulan dan Saran

Dari hasil yang didapat di tahap pengujian, dilakukan analisis hasil sebagai dasar dalam menarik kesimpulan. Analisis hasil juga dapat menghasilkan saran-saran yang dapat dipertimbangkan sebagai dasar pengembangan sistem pada penelitian berikutnya.

4. REKAYASA KEBUTUHAN

Rekayasa kebutuhan adalah aktivitas yang paling awal dijalankan pada tahapan pengembangan perangkat lunak. Aktivitas tersebut meliputi analisis proses bisnis dan kebutuhan beserta pendefinisian sebagai pedoman dasar dalam pengembangan sistem

4.1. Elisitasi Kebutuhan

Tahap ini merupakan tahapan yang penting dilakukan untuk membangun fondasi pengembangan sistem. Aktivitas yang dijalankan pada tahap ini yakni analisis terhadap prosedur yang sedang berlaku di FILKOM untuk memperoleh kebutuhan yang diperlukan. Tahap ini dilakukan dengan cara wawancara kepada pihak persuratan FILKOM.

4.1.1. Proses Bisnis Saat Ini

Pada prosedur surat keluar, proses diawali dari pembuat surat yang mengajukan permintaan pembuatan surat keluar kepada pihak persuratan. Pembuat surat dalam hal ini adalah pimpinan tiap unit kerja di FILKOM. Persuratan akan memproses pembuatan surat untuk kemudian dilakukan persetujuan kepada pimpinan surat. Apabila disetujui, maka pihak persuratan akan meminta paraf kepada pejabat dibawah pimpinan. Paraf ini dimaksudkan untuk dapat

meminimalisir kesalahan dalam penulisan surat. Apabila diperlukan perbaikan, maka pihak persuratan perlu melakukan perbaikan yang diminta. Ketika surat telah mendapat paraf dari pihak yang terkait, pihak persuratan kemudian meminta tanda tangan kepada pembuat surat, untuk diberi cap, diarsip, dan dikirimkan ke alamat tujuan.

4.1.1. Proses Bisnis yang Diusulkan

Pada proses bisnis yang diusulkan untuk surat keluar, alur diawali dari proses surat keluar pihak persuratan menerima perintah untuk membuat surat dengan melakukan input data pada sistem. Sistem kemudian menyimpan surat dan mengirimkan notifikasi verifikasi kepada pimpinan (pembuat surat). Pimpinan menerima notifikasi dan membaca surat. Pimpinan kemudian diberikan pilihan untuk melakukan verifikasi atau revisi yang berdampak pada berubahnya status surat keluar. Sistem akan melakukan seleksi status surat yang telah berubah. Apabila revisi maka sistem mengirimkan notifikasi revisi kepada persuratan untuk dapat diperbarui. Apabila verifikasi, maka sistem memperbarui status surat mejadi telah diverifikasi dan memberi nomor surat. Sistem kemudian mengirimkan notifikasi kepada pihak persuratan dan persuratan dapat mengunggah berkas akhir, yaitu berkas surat keluar yang telah ditanda tangani oleh pimpinan. Apabila telah tersimpan, sistem akan mengirimkan notifikasi kepada pimpinan pemohon.

4.2. Identifikasi Aktor

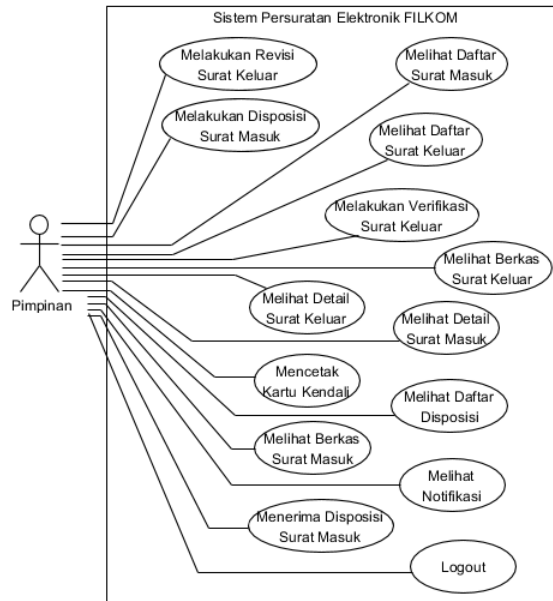
Identifikasi aktor dilakukan dengan mengacu dari analisis proses bisnis usulan untuk mengetahui siapa saja yang memiliki peran terhadap sistem. Tabel 1 menjelaskan lebih lanjut mengenai deskripsi peran aktor terhadap sistem.

Tabel 1. Deskripsi Aktor

Aktor	Deskripsi
Guest	Guest adalah pengguna sistem pada umumnya.
Admin Persuratan	Admin Persuratan merupakan aktor dari pihak pengelola surat di FILKOM
Staf/Dosen	Staf/Dosen merupakan aktor dari pihak staf/dosen di FILKOM
Pimpinan	Pimpinan merupakan aktor dari pihak Pimpinan FILKOM
KTU	KTU merupakan aktor turunan dari Pimpinan

4.3. Identifikasi Kebutuhan

Hasil analisis yang dilakukan pada proses bisnis usulan didapati bahwa ada 33 kebutuhan fungsional serta 1 kebutuhan non-fungsional yang diperlukan untuk mengembangkan sistem. *Use case diagram* digunakan dalam memodelkan fungsionalitas sistem. Gambar 3 memperlihatkan relasi antara aktor Pimpinan dengan *use case*-nya.



Gambar 3. Use Case Diagram Aktor Pimpinan

5. PERANCANGAN

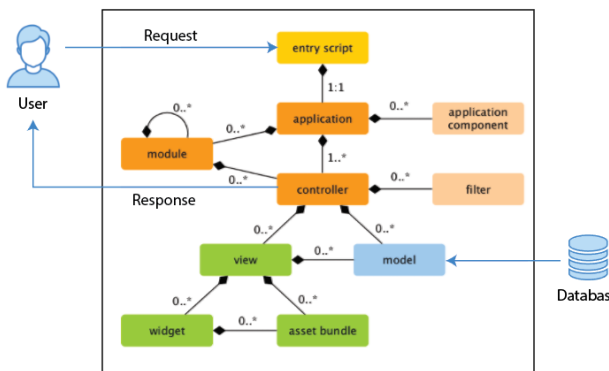
Tahapan perancangan dapat dideskripsikan sebagai fase dimana sistem dirancang sebelum kemudian siap untuk diimplementasikan. Aktivitas yang dilakukan pada tahap ini diantaranya perancangan arsitektur, perancangan data, pemodelan *sequence* dan *class diagram*, perancangan algoritme, dan perancangan antarmuka pengguna.

5.1. Perancangan Arsitektur

Perancangan arsitektur sistem berpedoman pada kerangka kerja yang digunakan dalam pengembangan perangkat lunak, yakni Yii Framework versi 2.0. Arsitektur yang tersedia pada Yii Framework dirancang dengan mengadopsi pola arsitektur perangkat lunak MVC (*Model-View-Controller*). Gambar 4 menunjukkan alur menunjukkan alur penanganan permintaan (*request handling*) yang diperankan sistem dengan pola arsitektur yang tersedia pada kerangka kerja Yii Framework versi 2.0.

Request handling diawali dengan

pengiriman permintaan oleh pengguna kepada *entry script*. *Entry script* kemudian memanggil *application* untuk dapat memanggil *controller* yang dibutuhkan. Pada *controller*, *model* kemungkinan akan dipanggil untuk mendapatkan data dari basis data. Sedangkan *view* akan diberikan setelahnya kepada pengguna dengan membawa serta data pada *model* yang didapat.



Gambar 4. Perancangan Arsitektur Sumber: Yii Framework (2018)

5.2. Perancangan Data

Sebelum dilakukan implementasi pembangunan basis data sistem, terlebih dahulu dilakukan perancangan pada data sebagai pedoman. Data dirancang dengan mengacu pada hasil analisis data, serta penyesuaiannya terhadap kebutuhan kerangka kerja yang digunakan. Dari hasil pemodelan PDM (*Physical Data Model*), ditemukan sebanyak 17 tabel yang akan diimplementasikan. Tabel 2 merupakan hasil perancangan untuk tabel t_disposisi.

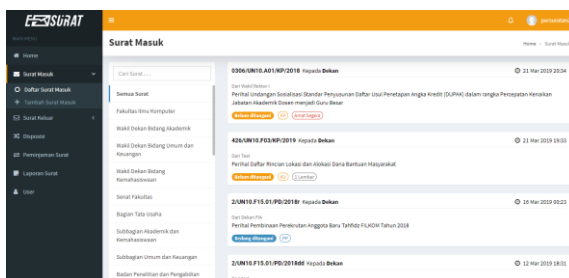
Tabel 2. Perancangan Data Tabel t_disposisi

Table with 4 columns: Atribut, Tipe, Panjang, Keterangan. Rows include id, tgl, id_surat, id_kepada, is_jawab, keterangan, isi, status, id_dari with their respective data types and lengths.

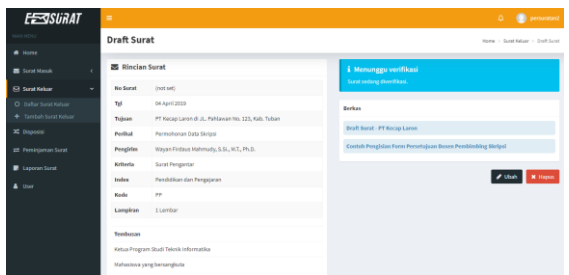
Table with 4 columns: pertimbangan, tinyint, 1, Penanda disposisi adalah meminta pertimbangan

6. IMPLEMENTASI

Tahap implementasi merupakan tahap realisasi dari perancangan yang telah dibuat. Pada tahap ini dijelaskan pula spesifikasi lingkungan dalam melakukan implementasi. Implementasi sistem dibagi menjadi beberapa bagian yakni, implementasi data, algoritme, dan antarmuka. Basis data dibangun dengan perintah DDL (*Data Definition Language*) serta MySQL. Implementasi algoritme menggunakan pendekatan OOP (*Object-oriented Programming*) dengan kode PHP dan kerangka kerja berbasis MVC (*Model-View-Controller*) yakni Yii Framework versi 2.0. Sedangkan HTML, CSS, dan Javascript digunakan untuk implementasi antarmuka. Gambar 5 memvisualkan halaman daftar surat masuk. Sedangkan Gambar 6 adalah visualisasi halaman detail draft surat keluar.



Gambar 5. Visualisasi Halaman Daftar Surat Masuk



Gambar 6. Visualisasi Halaman Detail Draft Surat Keluar

7. PENGUJIAN

Pengujian memberikan penjelasan bagaimana hasil implementasi sistem akan diuji. Terdapat beberapa tingkat pengujian yang dilakukan pada penelitian ini, yakni tingkat unit, integrasi, dan validasi. Serta terdapat pula pengujian kualitas compatibility untuk menguji kompatibilitas sistem.

Objektivitas dari pengujian unit adalah untuk mengetahui adanya kesalahan yang terjadi

pada setiap unit/komponen sistem. Sedangkan pengujian integrasi dilakukan dengan melihat interaksi antarkomponen atau antarunit yang bekerja pada sistem. Metode *white-box* dengan teknik *basis path testing* digunakan pada tahap ini. Hasil pengujian memberikan nilai valid pada semua jalur uji.

Pengujian validasi merupakan pengujian yang berfokus pada pemenuhan kebutuhan perangkat lunak yang terdefinisi di awal. Teknik yang digunakan dalam melakukan pengujian validasi adalah teknik *scenario-based testing* yang merupakan salah satu teknik dari metode *black-box*. Skenario uji dirancang berdasarkan *use case* atau aktivitas yang dilakukan aktor beserta variannya (Pressman, 2010). Hasil pengujian memberikan nilai valid untuk semua skenario yang diujikan.

Kualitas kompatibilitas sistem menyatakan bahwa sistem harus mampu bekerja di lingkungan yang berbeda-beda yang telah ditentukan di awal. Kompatibilitas sistem pada penelitian ini diuji dengan bantuan dari *tool* SortSite yang akan menguji sistem ke dalam 11 jenis peramban, yakni Edge 18, Chrome 71, Safari 12, Opera 57, Firefox 64, Internet Explorer 11, Android versi 3 dan 4 iOS 10 dan di bawahnya, serta iOS 11 dan 12. Hasil pengujian mengindikasikan bahwa sistem bekerja dengan tanpa masalah pada 8 dari 11 jenis peramban. Didapati pula sebanyak 1 permasalahan mayor dan 4 permasalahan minor mengenai ketidaktersediaan dukungan beberapa properti CSS pada peramban Internet Explorer, Firefox dan Safari. Persalahan-permasalahan tersebut hanya menyoroti tatanan tampilan saja dan tidak ada masalah yang mengganggu berjalannya fungsional sistem.

8. KESIMPULAN

Mengacu pada hasil analisis proses bisnis beserta kebutuhannya, didapati sebanyak 3 buah proses bisnis, yakni surat masuk, surat keluar, dan peminjaman surat. Dari hasil analisis proses bisnis, didapati pula sebanyak 33 kebutuhan fungsional dan 1 kebutuhan non-fungsional. Pada analisis data, didapati sebanyak 9 entitas yang dimodelkan ke dalam bentuk *entity relationship diagram*. Sedangkan kebutuhan fungsional dimodelkan menggunakan *use case diagram* beserta skenarionya.

Mengacu pada hasil perancangan, didapati perancangan arsitektur yang menghasilkan gambaran arsitektur *model-view-controller*

sistem sesuai dengan kerangka kerja Yii Framework versi 2.0, serta perancangan data yang menghasilkan 17 tabel yang dimodelkan menggunakan *physical data model*. Terdapat pula perancangan sistem menggunakan *sequence diagram* dan *class diagram*. Selain itu, terdapat perancangan algoritme yang menghasilkan *pseudocode*, serta perancangan antarmuka yang menghasilkan *wireframe*.

Mengacu pada hasil implementasi, didapati penjelasan mengenai spesifikasi pengembangan sistem. Terdapat pula implementasi data menggunakan *data definition language*. Implementasi lainnya yakni implementasi algoritme yang menghasilkan kode program dari *pseudocode*, dan implementasi antarmuka menghasilkan visual antarmuka sistem berdasarkan *wireframe* yang dibuat.

Mengacu pada hasil pengujian, didapati sebanyak 3 tingkat pengujian yang dilakukan, yakni tingkat unit, tingkat integrasi, dan tingkat validasi yang menghasilkan angka keberhasilan 100%. Dan dari pengujian *compatibility*, didapati bahwa dari 11 peramban yang diujikan, terdapat 8 atau 73% peramban di mana sistem dapat bekerja tanpa ada masalah.

9. DAFTAR PUSTAKA

- Arifin, Z., dan Mustakim, I., 2005. *Bahasa Indonesia bagi Sekretaris*. Jakarta: Grasindo. Tersedia di: Google Books <<http://books.google.com>> [Diakses 29 Agustus 2018]
- Nuraida, I., 2008. *Manajemen Administrasi Perkantoran*. Yogyakarta: Kanisius. [ebook] Tersedia di: Google Books <<http://books.google.com>> [Diakses 29 Agustus 2018]
- Pressman, Roger S., 2010. *Software Engineering: A Practitioner's Approach*. New York: Palgrave Macmillan. [e-book] Tersedia di: Google Books <<http://books.google.com>> [Diakses 30 Agustus 2018]
- Sommerville, Ian, 2011. *Software Engineering – 9th Edition*. Boston: AddisonWesley.
- Yii Framework, 2018. *The Definitive Guide to Yii 2.0*. [website] <<https://www.yiiframework.com/doc/guide/2.0/en/intro-yii>> [Diakses 26 September 2018]