

## RANCANG BANGUN SISTEM INFORMASI DISPOSISI ONLINE (SIDOL)

Noor Falih, Sarika Afrizal, A.B. Pangaribuan, Andhika  
Octa

Fakultas Ilmu Komputer  
Universitas Pembangunan Nasional Veteran Jakarta  
[falih@upnvj.ac.id](mailto:falih@upnvj.ac.id), [sarika.afrizal@upnvj.ac.id](mailto:sarika.afrizal@upnvj.ac.id), [artambo@upnvj.ac.id](mailto:artambo@upnvj.ac.id),  
[andyocta@upnvj.ac.id](mailto:andyocta@upnvj.ac.id)

Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia

### Abstrak

Disposisi adalah petunjuk singkat tentang tindak lanjut (penyelesaian) terhadap suatu urusan atau surat masuk. Disposisi dibuat oleh pimpinan untuk staf atau bawahan sesuai dengan bidang keahlian atau kewenangannya. Tujuan pembuatan disposisi ialah agar staf dapat menindaklanjuti atau menyelesaikan suatu urusan atau surat masuk sesuai dengan yang dikehendaki oleh pimpinan. Dalam hal administrasi tentang surat masuk, dan disposisi surat, Setjen KKP saat ini masih mengalami kendala dan kekurangan dalam mendisposisikan surat masuk yang masih menggunakan aplikasi desktop yang terpusat pada satu komputer saja. Aplikasi berbasis web yang responsif adalah solusi untuk mengatasi berbagai masalah tersebut. Penelitian ini menggunakan metode pengembangan prototype dan perancangannya menggunakan *Unified Modeling Language* (UML). Agar tampilan responsif dan modular, Sistem ini dibangun dengan mengkombinasikan antara javascript, framework css bootstrap dan framework CodeIgniter. Hasil akhir penelitian ini berupa sebuah Sistem Informasi Disposisi berbasis web yang bermanfaat untuk meningkatkan kinerja pegawai karena kemudahan untuk mengakses sistem informasi di mana saja dengan menggunakan koneksi internet. Selain itu proses pencarian surat dan statistik jumlah surat yang masuk dapat dilakukan dengan cepat.

Kata kunci: Disposisi, Aplikasi berbasis web, UML

### 1. PENDAHULUAN

Perkembangan teknologi yang pesat saat ini membuat arus kebutuhan informasi turut berkembang cepat. Internet sebagai salah satu media untuk mendapatkan informasi juga semakin banyak digunakan. World wide Web atau WWW merupakan salah satu fasilitas internet yang sangat populer. Saat ini teknologi web sudah digunakan oleh ribuan perusahaan atau pengusaha di dunia sebagai media promosi dan menawarkan produknya dalam bentuk informasi online melalui internet sehingga aplikasi berbasis desktop sudah mulai banyak ditinggalkan karena dirasa kurang fleksibel, kurang aman, sulit di-update, memakan biaya yang terlalu mahal, dan lain-lain. Aplikasi berbasis web adalah solusi untuk mengatasi berbagai masalah yang seringkali hadir saat menggunakan aplikasi berbasis desktop. Kelebihan aplikasi berbasis web adalah (Horton & Wiley, 2004):

1. Tidak perlu proses instalasi;
2. Dapat dibuka diberagam sistem operasi;
3. Dapat diakses tidak hanya di komputer, tetapi juga bisa di laptop maupun smartphone;
4. Tidak memerlukan spesifikasi khusus.

Selain itu aplikasi berbasis web saat ini dituntut menyediakan user *interface* yang responsif, dikarenakan klien dari aplikasi berbasis web menggunakan berbagai web browser dan device yang beragam (Falih, Hendradjaya, & Sunindyo, 2016).

Sekretariat Jenderal Kementerian Kelautan dan Perikanan (Setjen KKP) adalah unsur pembantu pemimpin atau pimpinan yang mempunyai tugas menyelenggarakan koordinasi pelaksanaan tugas, pembinaan, dan pemberian dukungan administrasi kepada seluruh unsur organisasi di lingkungan KKP, dalam hal ini adalah disposisi surat. Hal ini menyebabkan betapa kompleksnya administrasi yang diurus oleh Setjen KKP. Bila terjadi kesalahan administrasi keterlambatan distribusi disposisi akan menghambat kinerja.

Dalam hal administrasi tentang surat masuk, dan disposisi surat, Setjen KKP saat ini masih mengalami kendala dan kekurangan. Yaitu pada disposisi surat yang masih menggunakan aplikasi desktop yang terpusat pada satu komputer saja. Hal ini menyebabkan kendala dalam hal efisiensi pemakaian, karena hanya bisa diakses di satu komputer saja. Selain itu ketika terjadi rusak pada komputer tersebut maka sistem tidak dapat diakses hingga komputer tersebut selesai diperbaiki.

Berdasarkan permasalahan tersebut, penelitian ini difokuskan untuk merancang sistem informasi disposisi *online* yang responsif. Dengan adanya aplikasi tersebut, diharapkan nantinya pengelolaan surat akan lebih terorganisir sehingga akan mempermudah pencarian surat yang berkaitan dengan kegiatan yang akan dilaksanakan dan meningkatnya efisiensi kerja karena kemudahan untuk mengakses sistem informasi di mana saja.

Batasan Rancang bangun sistem disposisi online ini adalah:

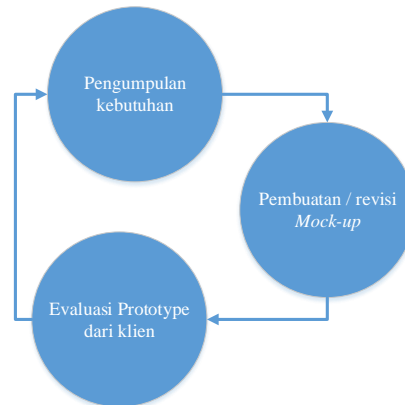
1. Lingkup sistem hanya sebatas Lingkup Setjen KKP.
2. Dalam membuat Aplikasi, menggunakan beberapa bahasa pemrograman web yaitu, php, html, dan javascript dengan framework CodeIgnitier dan Bootstarp dan mySQL sebagai databasenya.
3. Pengujian Aplikasi menggunakan browser chrome dan Mozilla firefox

Prosiding ini menjelaskan tahapan dalam merancang bangun sistem disposisi online yang terdiri dari beberapa bagian yaitu: bagian pertama berisi latar belakang beserta tinjauan teoritis yang mendukung penelitian ini, bagian kedua berisi metode penelitian yang diarahkan pada model pengembangan sistem, bagian ketiga berisi rancang bangun sistem disposisi online, dan bagian keempat berisi kesimpulan dan rekomendasi penelitian

## 2. METODOLOGI PENELITIAN

Metode yang digunakan pada penelitian ini adalah metode prototyping. Metode Pengembangan prototype lebih berupaya menciptakan gambaran perangkat lunak sebenarnya daripada berkonsentrasi pada dokumentasi. Dengan model ini, perangkat lunak bisa dirilis terlebih dahulu. Prototyping memerlukan lebih banyak keterlibatan pengguna, sehingga pemangku kepentingan dapat melihat gambaran perangkat lunak yang memungkinkan mereka untuk memberikan umpan balik dan spesifikasi perangkat lunak yang lebih baik. Sehingga meminimalisir banyaknya kesalahpahaman yang terjadi (Sabale & Dani, 2012).

Menurut Pressman (2010, p43-44), seringkali pelanggan mendefinisikan satu set kebutuhan perangkat lunak secara umum, tetapi tidak mengidentifikasi kebutuhan secara lebih rinci. Pengembangan prototype dapat dilihat pada gambar dibawah ini:



Gambar 1 Metode Pengembangan Prototype

Proses-proses tersebut dapat dijelaskan sebagai berikut:

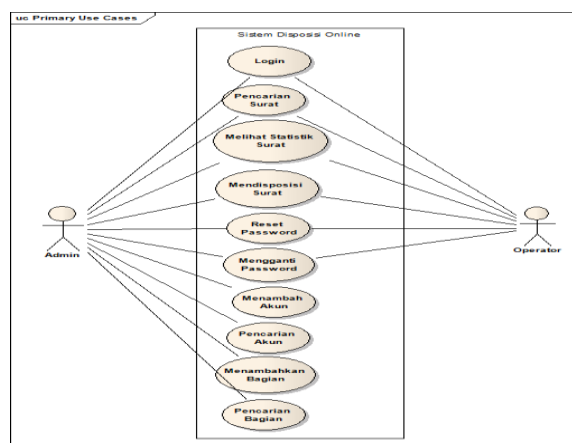
1. Pengumpulan kebutuhan: pihak pengembang perangkat lunak dan klien bertemu dan menentukan tujuan umum, kebutuhan dan gambaran perangkat lunak.
2. Pembuatan / revisi *Mock-up*: Merancang dan membuat *mock-up* / gambaran dari perangkat lunak dan menuangkannya menjadi prototype perangkat lunak.
3. Evaluasi Prototype klien: klien mengevaluasi prototype yang dibuat. Evaluasi tersebut digunakan untuk memperjelas kebutuhan perangkat lunak.

Desain sistem ini dilakukan dengan menggunakan (Mall et al. 2013) UML yang meliputi *use-case* diagram dan skenarionya kemudian dari sekenario tersebut di analisa untuk menghasilkan spesifikasi kebutuhan perangkat lunak. Kemudian antara kebutuhan perangkat lunak dengan *use-case* dibuat metrix penelusuran. Hubungan penelusuran akan memberikan pengetahuan perjalanan proses kebutuhan ke depan maupun ke belakang, dari tahap awal hingga implementasi (Gotel, Finkelstein, & Sw, 1994).

Agar tampilan responsif (tampilan rapi ketika dibuka melalui gadget dan pc desktop) dan modular, Sistem ini dibangun dengan mengkombinasikan antara javascript, *framework* css bootstrap dan *framework* CodeIgniter.

### 3. HASIL DAN PEMBAHASAN

Tahap paling awal yang dilakukan dalam pengembangan sistem adalah pengumpulan kebutuhan. Pada tahap ini peneliti berkonsultasi secara rutin dengan pihak KKP, dengan membuat *use-case*, *scenario* dan *mockup* aplikasi untuk dibahas pada setiap pertemuan, dan hasil akhir *use-case* dan skenarionya adalah sebagai berikut:



Gambar 2 Use case Sistem Disposisi Online

Terdapat sepuluh *use case* yang dihasilkan yaitu UC-01 Login, UC-02 Pencarian Surat, UC-03 Melihat Statistik Surat, UC-04 Mendisposisi Surat, UC-05 Reset Password, UC-06 Mengganti Password, UC-07 Menambah Akun, UC-08 Pencarian Akun, UC-09 Menambahkan Bagian, UC-10 Pencarian Bagian. Sedangkan definisi aktor dijelaskan pada tabel dibawah:

Tabel 1 Definisi Aktor SiDOL

No	Aktor	Deskripsi
1	Admin	Administrator adalah staff setjen KKP yang memiliki kewenangan penuh terhadap sistem dan dapat menggunakan semua fitur yang ada tanpa batasan. Hanya administrator yang dapat merubah atau membuat akun User baru.
2	Operator	Adalah staff Setjen KKP yang diberi kewenangan untuk menginput surat masuk baru ke sistem serta mengubah dan menghapus surat masuk tersebut dan melakukan disposisi

Kemudian dari *use-case* tersebut dibuatlah masing-masing sekenarionya, tujuannya adalah untuk memperjelas pemahaman sistem antara *stakeholder* dengan pengembang aplikasi. Skenario dari masing-masing *use-case* sebagian dijelaskan pada tabel dibawah:

Tabel 2 Skenario *use-case* Disposisi Surat

<b>UC-ID</b>	UC-04	
<b>Use case Name</b>	Mendisposisi Surat	
<b>Description</b>	<i>Use case</i> yang mendeskripsikan Rangkaian scenario proses mendisposisi surat masuk	
<b>Primary Actor</b>	Operator, Admin	
<b>Stakeholder and Interest</b>	Operator/admin: mencetak lembar disposisi	
<b>Precondition</b>	Operator/admin sudah login kedalam sistem	
<b>Postcondition</b>	Surat sudah di disposisikan	
<b>Input</b>	Detail surat masuk, lampiran surat masuk, tujuan bagian disposisi, perintah disposisi	
<b>Output</b>	Halaman lembar disposisi dengan format pdf	
<b>Normal Scenario</b>		
<b>Actor Action</b>		<b>System Response</b>
1	Operator/admin menekan tombol surat.	
		2 Sistem menampilkan halaman list surat
3	Operator/admin menekan tombol tambahkan surat.	
		4 Sistem menampilkan halaman form input surat
5	Operator/admin memasukkan detail surat masuk, lampiran surat masuk,	

	tujuan bagian disposisi, perintah disposisi kemudian menekan tombol simpan		
		6	Sistem menyimpan data yang dimasukkan, kemudian menampilkan halaman detail surat masuk
7	Operator/admin menekan tombol cetak lembar disposisi		
		8	Sistem mencetak lembar disposisi berupa format pdf
<b>Alternative Scenario</b>			
<b>Actor action</b>		<b>System Response</b>	
1	Skenario ini terjadi ketika data yang dimasukkan tidak sesuai atau ada yang kosong		
		2	Sistem menampilkan pesan bahwa ada data yang dimasukkan tidak sesuai atau kosong

Dari skenario tersebut, kemudian di analisa untuk menghasilkan spesifikasi kebutuhan perangkat lunak yang lebih jelas dan spesifik. Adapun hasil akhir kebutuhan fungsional sistem disposisi online adalah sebagai berikut:

Tabel 3. Kebutuhan Fungsional Sidol

Kode	Kebutuhan Fungsional
F1	Sistem dapat menampilkan statistik surat masuk berdasarkan tahun dan jenis surat (internal, eksternal, sesmen, undangan, menteri)
F2	Sistem dapat melakukan Input <i>edit</i> , hapus disposisi
F3	Sistem dapat mencetak lembar Disposisi dalam format pdf
F4	Sistem dapat melakukan Input edit, hapus Akun
F5	Sistem dapat melakukan <i>Input edit</i> , hapus Bagian
F6	Sistem dapat mengauthentifkasi akun yang masuk berdasarkan <i>username</i> dan <i>password</i> dan tahun surat
F7	Sistem dapat melakukan reset password bagi pengguna yang lupa <i>username</i> atau <i>password</i> -nya dengan memasukkan <i>email</i> pengguna
F8	Pengguna dapat mengubah password
F9	Sistem menampilkan list surat berdasarkan tahun yang dipilih pada saat login
F10	Sistem dapat melakukan pencarian surat berdasarkan kata kunci yang dimasukkan pengguna
F11	Sistem dapat melakukan pencarian bagian berdasarkan kata kunci yang dimasukkan pengguna
F12	Sistem dapat melakukan pencarian akun berdasarkan kata kunci yang dimasukkan pengguna
F13	Sistem dapat mengupload file surat masuk berekstensi JPG, JPEG, PNG, GIF, PDF dengan ukuran file maksimum 10 Mb.
F14	Sistem memberikan akses kepada operator untuk Input, <i>edit</i> , hapus dan cetak disposisi
F15	Sistem memberikan akses kepada admin semua fitur yang ada pada sidol

Agar kebutuhan dapat dikelola secara efektif maka di butuhkan ketelusuran kebutuhan dengan *use-case*, ketelusuran kebutuhan dengan *use-case* dapat dilihat pada tabel dibawah:

Tabel 4. Matriks ketelusuran antara *use-case* dengan kebutuhan fungsional sidol

	UC-01	UC-02	UC-03	UC-04	UC-05	UC-06	UC-07	UC-08	UC-09	UC-10
F1			X							
F2				X						
F3				X						
F4							X			
F5									X	
F6	X									
F7					X					
F8						X				
F9	X									
F10		X								
F11										X
F12								X		
F13				X						
F14	X	X	X	X	X	X				
F15							X	X	X	X

Tampilan cetak lembar disposisi berbentuk pdf adalah sebagai berikut :

Gambar 3. Halaman cetak lembar disposisi

Tampilan Form input lembar disposisi adalah sebagai berikut:

Gambar 4. Halaman form input lembar disposisi

#### 4. KESIMPULAN

Dengan adanya sistem informasi disposisi online dapat memberikan kemudahan pegawai untuk mengakses sistem informasi di mana saja dengan menggunakan koneksi internet. Selain itu proses pencarian surat dan statistik jumlah surat yang masuk dapat dilakukan dengan cepat. Dari segi proses pembuatannya sistem ini menggunakan metode pengembangan *prototype* dan perancangannya menggunakan UML. Metode *prototype* sangat baik digunakan sebagai sarana validasi dari sisi pengguna dan pelanggan. Karena dibuat dalam berbagai bentuk dan ukuran, mulai dari prototip bentuk model/*mockup* gambar sampai prototip formal berbentuk program. Sehingga pengguna dan pelanggan merasa terlibat dalam pengembangan sistem. Dari sisi pembangunannya sistem ini dibangun menggunakan *framework* dengan tujuan agar kode yang dihasilkan modular, selain itu penggunaan *bootstrap* ditujukan agar tampilan sistem responsif jika dibuka baik melalui pc dan gadget (*tablet* dan *handphone*).

Saran dari penelitian ini yaitu:

1. Perlu dikembangkan suatu sistem yang melibatkan pada setiap posisi jabatan / bagian, agar dapat dilakukan pelacakan disposisi tersebut, sehingga atasan dapat memantau seluruh proses disposisi yang selesai atau yang belum selesai.
2. Diperlukan algoritma untuk membuat pelacakan disposisi, salah satunya menggunakan *graph tree*.
3. Pengujian dilakukan menggunakan browser chrome dan Mozilla firefox yang bertujuan menguji portabilitas web yang dapat dijalankan di beberapa web browser. Untuk itu diperlukan pengujian pada aspek fungsionalitas, reliabilitas, usabilitas, keamanan, kompatibilitas, performansi, dan maintainabilitas,

#### Referensi

- Falih, N., Hendradjaya, B., & Sunindyo, W. D. (2016). Quality Measurement for Web GIS Using Object-Oriented Development.
- Gotel, O. C., Finkelstein, A. C. W., & Sw, L. (1994). An Analysis of the Requirements Traceability Problem Imperial College of Science, Technology & Medicine Department of Computing, 180 Queen<sup>TM</sup> s Gate, 94–101.
- Horton, B. W., & Wiley, N. Y. (2004). Designing Web-Based Training: How to Teach Anyone Anything Anywhere Anytime, 3(July), 4–6.
- Myer, Thomas (2008), Professional CodeIgniter, Wrox, USA.
- Mall, R., Kundu, D. & Samanta, D., 2013. Automatic code generation from unified modelling language sequence diagrams. *IET Software*, 7(1), pp.12–28. Available at: <http://digitalibrary.theiet.org/content/journals/10.1049/iet-sen.2011.0080>.
- Pressman, R. S. (2010). *Software Engineering*.
- Sabale, R. G., & Dani, A. R. (2012). Comparative Study of Prototype Model for Software Engineering with System Development Life Cycle, 2(7), 21–24.
- Spurlock J (2013) Bootstrap: Responsive Web development. O'Reilly Media, Sebastopol, CA.