

Rancang Bangun *Revolving Fund Document Management System*

Jamaluddin¹, Nur Aeni Hidayah², dan Nidaul Hasanati³

Abstrak—*Document Management System (DMS)* adalah sistem berbasis komputer yang menyediakan penyimpanan berbasis web yang dapat diakses di berbagai tempat. DMS dapat mengurangi resiko kehilangan dokumen, memangkas biaya pengarsipan dokumen fisik dan dapat dicetak kembali. Universitas Islam Negeri (UIN) Syarif Hidayatullah Jakarta memiliki proses Ganti Uang Persediaan (GUP) yaitu mekanisme penggantian Uang Persediaan (UP) yang diberikan pada awal tahun anggaran, proses ini melibatkan banyak berkas penting dan seluruh fakultas. Oleh karena itu, DMS mencoba mendukung proses bisnis GUP ini dengan menyediakan media koordinasi antar unit dalam mengelola berkas maupun dokumen elektronik secara online untuk meminimalisir kecelakaan ataupun hilangnya berkas. Metode pengembangan sistem yang digunakan adalah *Scrum* dengan *Unified Modeling Language (UML)* sebagai tools perancangan sistem. Sistem dibangun menggunakan PHP dan MySQL. Mengacu dari hasil wawancara kepada Bapak Ady Cahyadi selaku Sekretaris SPI dan Bapak Rezky Mehta Setiadi sebagai Koor Divisi Kepatuhan dan Keuangan SPI, output berupa *Revolving Fund Document Management System* berbasis web ini mendukung pengelolaan dokumen, pengecekan proses dokumen dan koordinasi secara online pada proses GUP tingkat fakultas di UIN Jakarta.

Kata Kunci—Manajemen dokumen, *monitoring*, *electronic document*, uang persediaan, *scrum*, *unified modeling language*

I. PENDAHULUAN

Uang Persediaan adalah uang muka kerja dengan jumlah tertentu yang bersifat daur ulang (*revolving fund*). Uang Persediaan ini terkait dengan prosedur Ganti Uang Persediaan (GUP) dan Tambah Uang Persediaan (TUP). Ganti Uang Persediaan adalah mekanisme pengajuan penggantian atas Uang Persediaan yang telah diberikan pada awal tahun anggaran, sedangkan Tambah Uang Persediaan adalah uang yang diberikan kepada Bendahara Pengeluaran untuk kebutuhan mendesak dan akan habis digunakan dalam 1 bulan terhitung sejak tanggal dikeluarkannya cek. Satuan Pemeriksa Intern sebagai organ pengawasan kampus, ikut serta dalam pengelolaan, pemeriksaan dokumen-dokumen uang persediaan yang dijelaskan sebelumnya, dokumen-dokumen tersebut

berasal dari berbagai fakultas dan program studi hingga ke bendahara pengeluaran universitas, oleh karena itu manajemen dan kontrol terhadap dokumen-dokumen tersebut sangatlah penting.

Proses ganti uang persediaan melibatkan staf keuangan tingkat fakultas menyiapkan dokumen pengajuan ganti uang persediaan yang diserahkan kepada Kuasa Pengguna Anggaran (KPA) berikut dengan berbagai dokumen yang perlu diproses dan dikelola seperti surat permintaan pembayaran, surat permohonan pencairan, surat pertanggungjawaban, surat pernyataan, laporan dana operasional, serta bukti-bukti pembayaran belanja. Kuasa pengguna anggaran menerima dokumen tersebut, kemudian meneruskan kepada SPI untuk proses verifikasi. SPI menerima dan melakukan verifikasi dengan waktu dua hari dan memberikan tanggapan yang jika disetujui akan ditindaklanjuti Kasubbag Verifikasi dalam waktu satu hari dan jika tidak disetujui akan kembali ke staf keuangan fakultas. Kasubbag Verifikasi memberikan disposisi hasil tindak lanjut kepada Bendahara Pengeluaran Pembantu (BPP) untuk proses pencairan. BPP nantinya akan menyerahkan dana tersebut dan menyatakan proses telah selesai.

Namun pada pelaksanaan proses tersebut, sering kali SPI maupun unit lainnya yang terlibat tidak mengetahui keberadaan maupun status dokumen karena belum adanya sistem khusus yang membantu SPI untuk merekam data, menelusuri keberadaan, memberikan informasi realtime terkait waktu yang ditetapkan standar operasional, serta mengelola dokumen uang persediaan tersebut secara digital. Padahal masalah manajemen dokumen seperti ini memiliki ruang lingkup yang luas karena meliputi banyak data dokumen, surat-surat dan bukti penting dari berbagai fakultas dan unit-unit yang terkait dalam prosedur ini, terlebih lagi resiko hilangnya dokumen karena kelalaian maupun musibah.

Berdasarkan hasil wawancara dengan Sekretaris Satuan Pemeriksa Intern, Bapak Ady Cahyadi, maka peneliti merasa perlu untuk membuat sistem manajemen dokumen uang persediaan secara online di UIN Syarif Hidayatullah Jakarta.

Penelitian ini menggunakan metode pengembangan sistem *SCRUM*. Metode ini dipilih karena berorientasi pada narasi dan kebutuhan user dalam menggunakan sistem, fleksibel karena mudah untuk diadaptasi dalam proyek skala kecil hingga menengah, tidak rumit dan mudah dipahami serta memungkinkan pembangunan sistem dengan fungsional utuh dalam jangka waktu yang singkat [1].

Received: 12 July 2018; Revised: 5 Agustus 2018; Accepted: 1 September 2018

Jamaluddin, *Web Developer di DGTraffic* (jamaluddin.elwa@gmail.com)
N. A. Hidayah, *dosen Prodi Sistem Informasi Universitas Islam Negeri Syarif Hidayatullah Jakarta* (nur.aeni@uinjkt.ac.id)
N. Hasanati, *dosen Prodi Sistem Informasi Universitas Islam Negeri Syarif Hidayatullah Jakarta* (nidaul.hasanati@uinjkt.ac.id)

II. KAJIAN PENELITIAN

A. Rancang Bangun

Sebelum memahami arti dari rancang bangun, peneliti akan terlebih dahulu menjelaskan pengertian dari rancang bangun. Perancangan atau rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisis dari suatu sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan.

Sedangkan pembangunan atau bangun sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang ada secara keseluruhan.

Dari semua uraian sebelumnya, maka dapat disimpulkan bahwa rancang bangun adalah serangkaian prosedur untuk menerjemahkan hasil analisis ke dalam bahasa pemrograman dan cara implementasinya untuk menciptakan sistem yang baru maupun memperbaiki suatu sistem yang sebelumnya [2].

B. Konsep Document Management System (DMS)

Sistem Manajemen Dokumen atau Document Management System adalah suatu aplikasi yang digunakan untuk melacak dan menyimpan gambar dari kertas atau dokumen elektronik. Sistem Manajemen Dokumen merupakan sistem berbasis komputer yang menyediakan tempat penyimpanan berbasis web yang dapat diakses dari berbagai tempat. Inti pada sistem manajemen dokumen adalah tempat penyimpanan terpusat (*centralized repository*), sebuah medium elektronik tempat penyimpanan (*storage*) dengan sebuah lokasi *storage* utama yang mampu menyediakan banyak akses ke dalamnya.

Beberapa fungsi atau nilai tambah pada sistem manajemen dokumen meliputi [3]:

- 1) Mengontrol untuk menjamin hanya satu user saja yang memodifikasi sebuah dokumen pada satu waktu
- 2) Memeriksa jejak untuk mengawasi perubahan-perubahan yang terjadi pada sebuah dokumen setiap waktu.
- 3) Menyiapkan keamanan untuk mengontrol akses *user* kepada dokumen-dokumen.
- 4) Pengaturan dokumen-dokumen ke dalam groups yang berhubungan dan *folders*.
- 5) Pengenalan dan mendapatkan kembali dari dokumen-dokumen sesuai dengan teks yang ada pada dokumen-dokumen tersebut (*free text searching*).
- 6) Mencatat informasi yang berhubungan dengan dokumen sebagai metadata, seperti pengarang, tanggal pembuatan dan judul.
- 7) Kemampuan untuk mengirimkan dokumen-dokumen dari satu user kepada user lainnya dalam kebiasaan yang terkontrol berdasarkan *workflow*.
- 8) Merubah dokumen-dokumen kertas ke dalam format elektronik dengan melakukan *scanning*.
- 9) Mengatur dokumen-dokumen ke dalam grup-grup untuk memungkinkan dokumen-dokumen tersebut untuk didistribusikan kepada target pencari atau pembaca.

C. Scrum

Scrum adalah sebuah metode pengembangan sistem *Agile* yang digagas oleh Jeff Sutherland dan tim pengembangnya

pada awal tahun 1990-an [4]. Dalam beberapa tahun terakhir, pengembangan lebih jauh pada metode Scrum telah dilakukan oleh Schwaber dan Beedle. Scrum adalah kerangka kerja di mana orang dapat mengatasi masalah adaptif yang kompleks, sementara secara produktif dan kreatif memberikan produk dengan nilai setinggi mungkin [5]. Tahapan yang ada pada metode Scrum yaitu:

1) *Sprint Planning* adalah tahap dimana pekerjaan yang akan dilakukan dalam satu fase *sprint* direncanakan. perencanaan ini dibuat dengan kolaborasi oleh seluruh anggota tim [6].

2) *Daily Scrum* adalah *event* dengan batas waktu 15 menit. Pada waktu tersebut, tim pengembang merencanakan apa saja yang harus dikerjakan untuk 24 jam ke depan [7].

3) *Sprint Review* diadakan pada akhir *Sprint* untuk meninjau perkembangan dan mengadaptasi dari *Product Backlog*. Pada tahap ini, *Scrum Team* berkolaborasi dengan *stakeholder* ada hal-hal yang dapat diselesaikan untuk meningkatkan *value* [8]. Pertemuan ini bersifat informal dan perkembangan produk dapat ditampilkan dan diuji coba untuk mendapatkan feedback dan *output* yang sesuai.

4) *Retrospective*, proses terakhir dari masa *sprint* setelah *sprint review* dimana setelah ini proses pengembangan akan kembali ke proses penentuan *product backlog* [9].

III. METODE PENELITIAN

A. Metode Pengumpulan Data

Metode pengumpulan data dalam penelitian ini terdiri dari observasi/pengamatan, wawancara, dan melalui studi pustaka dan studi literatur.

Sedangkan metode pengembangan sistem peneliti menggunakan metode Scrum. Berikut adalah tahap yang dijalankan [10]:

1) *Sprint Planning*

Pada tahap ini peneliti menyusun *product backlog* yang merupakan fitur-fitur atau komponen yang ditentukan bersama tim pengembang sistem dan *product owner* [11]. *Product backlog* disusun berdasarkan *user stories* yang didiskusikan oleh *product owner* ataupun *stakeholder* sebagai *guideline* dalam mengembangkan fitur-fitur yang akan dibangun pada sistem [12].

2) *Daily Scrum*

Pada tahap ini peneliti merancang sistem yang nantinya akan memecah *product backlog* menjadi *sprint backlog* yang berisikan tugas-tugas yang harus diselesaikan agar dapat memenuhi tujuan *sprint*. Pada tahap ini penulis melakukan perancangan sistem dengan menggunakan UML sebelum melakukan proses pengembangan sistem. Pada tahap ini, peneliti menentukan apa yang harus dilakukan pada *sprint backlog* yang berisikan tugas-tugas yang lebih spesifik untuk menyelesaikan item dari *product backlog*. Pada tahap ini proses pemrograman beserta perancangan antar muka dilakukan [13].

3) *Sprint Review*

Pada tahap ini masa *sprint* telah berakhir dan peneliti melakukan review bersama tim dengan melakukan demonstrasi terhadap *product backlog* yang telah diselesaikan selama masa *sprint* [14]. Item dari *product backlog* yang telah diselesaikan diuji dengan anggota tim dengan menggunakan metode *black box testing* dengan teknik *scenario based testing* agar *user* yang awam secara ilmu pemrograman dapat memahami fitur yang sedang diuji. Peneliti membuat sistem dengan menggunakan bahasa pemrograman PHP dengan *framework* Laravel 5.7 serta MySQL untuk *database*.

4) *Sprint Retrospective*

Pada tahap ini keseluruhan fase *Sprint* telah berakhir, peneliti melakukan peninjauan ulang terhadap kinerja pengembangan sistem dan melakukan perbaikan terhadap hal-hal yang menjadi masalah dalam pengembangan *product backlog* di *Sprint* selanjutnya [15].

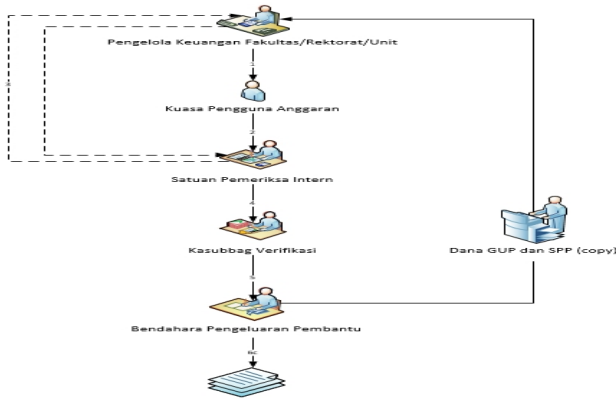
IV. HASIL

A. *Sprint Planning*

Tahap ini terdiri dari menentukan *user stories*, menentukan *acceptance criteria* dan menyusun *product backlog*. Hasil dari tahap tersebut dapat dilihat berikut.

1) *Menentukan User Stories*

Peneliti melakukan observasi di Satuan Pemeriksa Intern dan wawancara secara langsung untuk mengetahui mekanisme dari proses ganti uang persediaan. Pada Gambar 1, adalah mekanisme dari proses ganti uang persediaan.



Gambar 1. Rich Picture Mekanisme GUP

Tabel 1. *User Stories*

Melakukan pendataan proses pengajuan GUP	
a)	Sebagai staf keuangan fakultas saya dapat memberikan data dokumen yang telah disiapkan
b)	Sebagai kuasa pengguna anggaran saya dapat melihat dan memberikan data dokumen yang diajukan ke satuan pemeriksa intern
c)	Sebagai satuan pemeriksa intern saya dapat melihat surat

- d) Sebagai staf keuangan fakultas saya dapat memberikan data revisi dari hasil verifikasi
- e) Sebagai kasubbag verifikasi saya dapat meninjau hasil verifikasi dan melakukan disposisi ke bendahara pengeluaran pembantu
- f) Sebagai bendahara pengeluaran pembantu saya dapat memberikan data pencairan dan menyatakan proses selesai

2) *Melakukan Acceptance Criteria*

Setelah menentukan *user stories*, selanjutnya peneliti menentukan *acceptance criteria*. Salah satu diantara hasil penetapan *acceptance criteria* tersebut dapat dilihat pada Tabel 2.

Tabel 2. *Acceptance Criteria*

Melakukan pendataan proses pengajuan GUP
<i>Acceptance Criteria</i>
<i>User</i> (staf keuangan fakultas) dapat memberikan data dokumen
<i>User</i> (kuasa pengguna anggaran) dapat melihat dokumen yang telah diajukan <i>user</i> (staf keuangan fakultas)
<i>User</i> (kuasa pengguna anggaran) dapat memberikan data pengajuan kepada <i>user</i> (satuan pemeriksa intern)
<i>User</i> (satuan pemeriksa intern) dapat melihat pengajuan yang telah diajukan <i>user</i> (kuasa <i>user</i> angrgan)
<i>User</i> (satuan pemeriksa intern) dapat menolak pengajuan pengajuan <i>user</i> (staf keuangan fakultas)
<i>User</i> (staf keuangan fakultas) dapat memberikan data revisi
<i>User</i> (Satuan pemeriksa intern) dapat menerima pengajuan dan memberikan data hasil verifikasi
<i>User</i> (satuan pemeriksa intern) dapat memberikan data disposisi beserta data hasil verifikasi ke <i>user</i> (kasubbag verifikasi)
<i>User</i> (kasubbag verifikasi) dapat melihat dan memberikan data disposisi untuk pencairan kepada <i>user</i> (bendahara pengeluaran pembantu)
<i>User</i> (bendahara pengeluaran pembantu) dapat memberikan data tindak lanjut pengajuan pencairan

3) *Menyusun Product Backlog*

Dari hasil penentuan *user stories* dan *acceptance criteria* kemudian peneliti jadikan acuan dalam menyusun *product backlog*. Gambaran sistem usulan tersebut peneliti susun, yang terlihat pada Tabel 3.

Tabel 3. Hasil Uji Validitas Kuesioner

<i>To do</i>	<i>In Progress</i>	<i>Done</i>
Melakukan Pendataan proses pengajuan GPU		
Membuat fitur pendataan dokumen untuk staf keuangan fakultas	Membuat fitur pendataan pengajuan dokumen untuk staf keuangan fakultas dan kuasa pengguna anggaran	Membuat fitur pendataan verifikasi pengajuan dokumen untuk satuan pemeriksa intern

Membuat fitur disposisi hasil verifikasi kepada kasubbag verifikasi

Membuat fitur disposisi kepada bendahara pengeluaran pembantu untuk melakukan pencairan

Membuat fitur pendataan pencairan agar pengajuan dinyatakan selesai dan diketahui staf keuangan fakultas dan kuasa pengguna anggaran

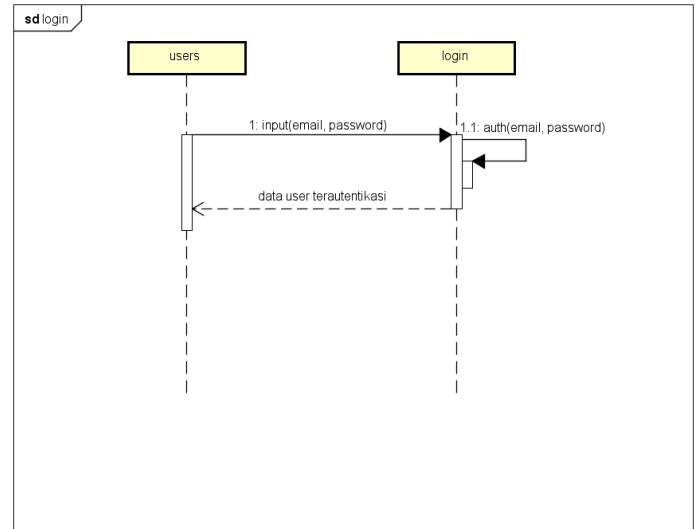
Membuat fitur pendataan dokumen untuk staf keuangan fakultas

Membuat fitur pendataan pengajuan dokumen untuk staf keuangan fakultas dan kuasa pengguna anggaran

Membuat fitur pendataan verifikasi pengajuan dokumen untuk satuan pemeriksa intern

b) Perancangan Desain Database

Pada bagian ini peneliti merancang database dengan membuat UML diantaranya adalah *sequence diagram*, *component diagram*, *deployment diagram*, skema database dan spesifikasi database. Contoh dari beberapa diagram tersebut dapat dilihat Gambar 3 hingga Gambar 5.



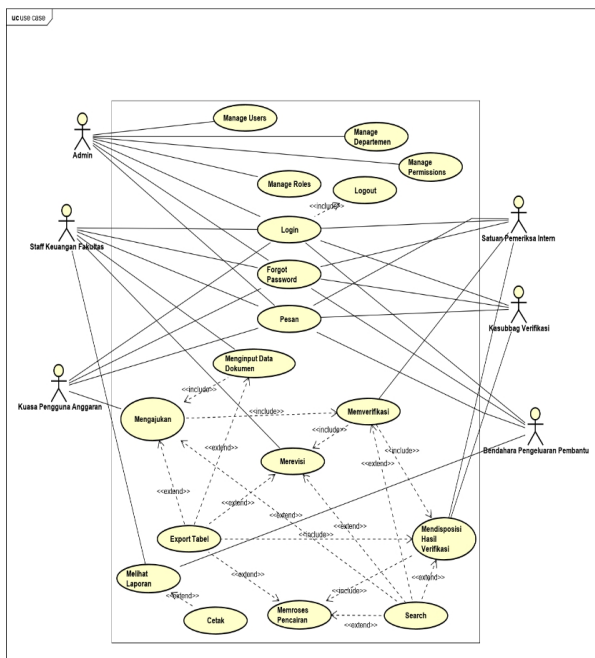
Gambar 3. Sequence Diagram

4) Daily Scrum

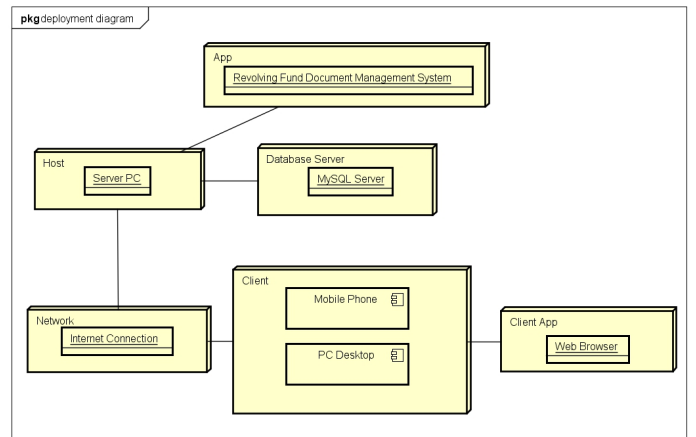
Pada tahap ini peneliti melakukan proses pengembangan sistem dengan melakukan perancangan desain proses terlebih dahulu yang terdiri atas 3 bagian, yaitu perancangan UML, perancangan desain database, dan perancangan desain interface.

a) Rancangan UML

Pada bagian ini peneliti membuat use case diagram



Gambar 2. Usecase Diagram

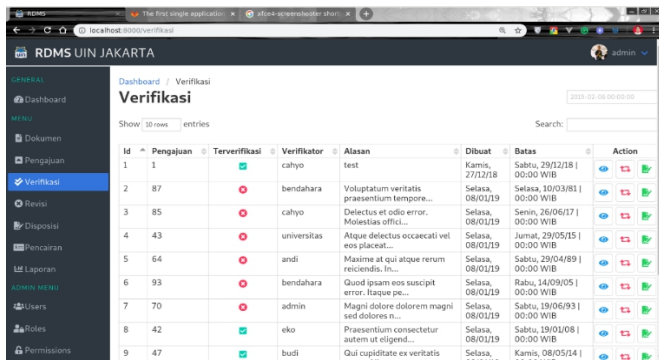


Gambar 4. Deployment Diagram

c) Perancangan Desain Interface

Tahap selanjutnya adalah membuat rancangan dari menu-menu yang ada pada Revolving Fund Document Management System.

Aktor berhasil
 menginput data dokumen

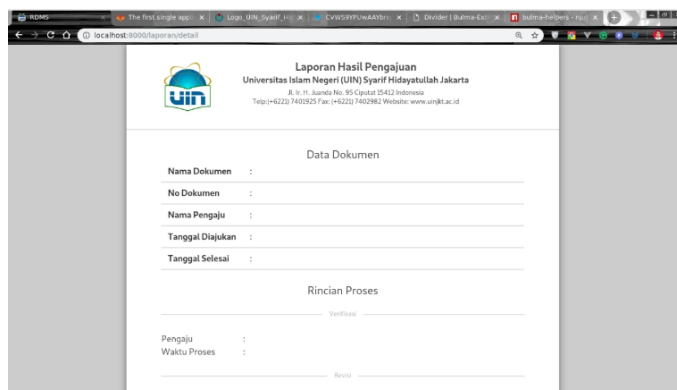


Keterangan

V. KESIMPULAN

Penelitian ini menyimpulkan bahwa mengacu dari hasil wawancara dengan Bapak Ady Cahyadi sebagai Sekretaris SPI dan Bapak Rezky Mehta sebagai Koordinator Divisi Kepatuhan dan Keuangan SPI bahwa *Revolving Fund Document Management System* mampu mendukung unit yang terlibat dalam pemrosesan dokumen untuk memantau dokumen secara *realtime* dengan menghadirkan dialog informasi posisi dan status dokumen. Sistem ini menawarkan opsi kemudahan *backup file* dokumen berdasarkan proses yang telah dilakukan sebagai alternatif dari pengarsipan dokumen fisik yang memiliki kemungkinan tercecer dan mengalami kerusakan akibat kelalaian ataupun kecelakaan. Sistem ini mampu menyediakan sarana bertukar informasi dengan fitur pesan dan menampilkan waktu proses dari hasil pengajuan serta kinerja pemrosesan dokumen selama satu bulan terakhir. Sistem dapat menampilkan waktu yang diperlukan dan waktu yang tersisa untuk memproses suatu dokumen. Sistem ini juga menampilkan dokumen yang membutuhkan respon segera pada halaman utama sebagai upaya untuk mengurangi waktu pemrosesan dokumen yang terlambat.

Gambar 4. Desain *Interface* Halaman Verifikasi Dokumen



Gambar 5. Desain *Interface* Laporan Hasil Pengajuan

B. Sprint Review

Dalam tahapan ini, peneliti melakukan pengujian serta demo sistem dengan menggunakan metode *black box testing* dengan teknik *scenario based testing*. *Development environment* yang digunakan dalam pengujian adalah *web server* lokal PHP dari *Laravel* dan *MySQL* sebagai *database* sistem. Selain itu, peneliti juga menggunakan *web browser* Google Chrome atau Chromium guna keperluan penampilan dan pengujian sistem. Pada Tabel 4 adalah salah satu di antara skenario pengujian sistem.

Tabel 4. Pengujian
 Pengujian Input Data Dokumen

Aktor		Staf Keuangan Fakultas
Menu		Dokumen
Aktivitas Rancangan Proses		Klik <i>Create</i> Isi <i>Form</i> data dokumen Klik Tombol simpan
Tanda	Peringatan	sistem gagal, sistem menampilkan pesan pada <i>Field</i> yang <i>error</i>
	Kesalahan Kesulitan Sistem	<i>Field</i> Harus diisi
Hasil	<i>Warning</i> <i>Accept</i>	Sesuai

REFERENSI

- [1] W. Suharso, B. I. Wicaksono, and G. I. Marthasari, "Penerapan Scrum dan Algoritma COCOMO pada Aplikasi Manajemen Proyek Perangkat Lunak," *SATIN - Sains dan Teknologi Informasi*, vol. 4, no. 1, pp. 97-104, 2018.
- [2] D. Lavarino and W. Yustanti, "Rancang Bangun E-Voting Berbasis Website di Universitas Negeri Surabaya," *Jurnal Manajemen Informasi*, vol. 6, no. 1, pp. 72-81, 2016.
- [3] S. Sutirman, N. S. Wijayanti, and P. Purwanto, "Studi Tentang Implementasi Sistem Manajemen Arsip Elektronik pada Kantor Pemerintahan Kota Yogyakarta," *Efisiensi*, vol. 14, no. 1, pp. 70-97, 2016.
- [4] H. Kniberg, *Scrum and XP from the Trenches*. Lulu. com, 2015.
- [5] A. Wahyudi, "Analisis Pengembangan Perpustakaan Digital Bebas Android dengan Metode Scrum," *Factor Exacta*, vol. 11, no. 2, pp. 128-134, 2018.
- [6] A. Alhazmi and S. Huang, "A decision support system for sprint planning in Scrum practice," in *SoutheastCon 2018*, 2018, pp. 1-9: IEEE.
- [7] A. T. Karabulut, E. Ergun, and Finance, "A new way of management: A scrum management," *International Journal of Commerce and Finance*, vol. 4, no. 2, pp. 108-117, 2018.
- [8] W. L. LEE, "SCRUM-X: An interactive and experiential learning platform for teaching scrum," in *The 7th International Conference on Education, Training and Informatics (ICETI)*, 2016.
- [9] A. Przybyłek and D. Kotecka, "Making agile retrospectives more awesome," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2017, pp. 1211-1216: IEEE.
- [10] R. A. Azdy, Azhari SN, "Implementasi Scrum pada pengembangan Software Terdistribusi," in *Seminar Nasional Informatika*, UPN Veteran Yogyakarta, 30 Juni 2012.
- [11] M. Golfarelli, S. Rizzi, E. J. Turrinchia, and software, "Multi-sprint planning and smooth replanning: An optimization model," *Journal of System and Software*, vol. 86, no. 9, pp. 2357-2370, 2013.

- [12] S. Jansi, M. Rajeswari, and Technology, "A greedy heuristic approach for sprint planning in agile software development," *Intl. Journal for Trends in Engineering and Technology*, vol. 3, no. 1, pp. 18-21, 2015.
- [13] M. Elallaoui, K. Nafil, and R. Touahni, "Automatic generation of UML sequence diagrams from user stories in Scrum process," in *2015 10th international conference on intelligent systems: theories and applications (SITA)*, 2015, pp. 1-6: IEEE.
- [14] S. Sharma and N. Hasteer, "A comprehensive study on state of Scrum development," in *2016 International Conference on Computing, Communication and Automation (ICCCA)*, 2016, pp. 867-872: IEEE.
- [15] J. Werewka and A. Spiechowicz, "Enterprise architecture approach to SCRUM processes, sprint retrospective example," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2017, pp. 1221-1228: IEEE.