

SISTEM INFORMASI PERPUSTAKAAN DIGITAL BERBASIS WEB MENGGUNAKAN FRAMEWORK SLIM CENDANA

Akik Hidayat¹⁾, Alam Buana²⁾

¹ Prodi Teknik Informatika Departemen Ilmu Komputer, Universitas Padjadjaran
akik@unpad.ac.id¹,
alambuana@gmail.com²

ABSTRAK

Perkembangan teknologi internet dan web begitu pesat. Karena dengan menggunakan media ini informasi yang disampaikan dapat mencapai berbagai lapisan masyarakat. Teknologi web merupakan salah satu layanan di internet yang paling sering digunakan informasi secara cepat. Dalam hal ini, dashboard akan memberikan informasi seputar gambaran informasi penting dalam grafik yang mudah dibaca, data dikumpulkan secara real time, dashboard menyampaikan informasi penting dalam grafik yang mudah dibaca, dan dikumpulkan secara real time, dashboard menyampaikan informasi yang dibutuhkan oleh semua orang setiap hari dalam pengambilan keputusan. Dengan dibuatnya Sistem Informasi Perpustakaan Digital Berbasis Web dengan menggunakan Framework Slim Cendana diharapkan menjadi salah satu media untuk pengumpulan data buku dan jenis katalog buku agar nantinya dapat diakses dengan mudah serta dapat dipertanggungjawabkan.

Kata Kunci : Dashboard, *web*, Slim, Perpustakaan.

I. PENDAHULUAN

Perkembangan *web* yang semakin pesat sejak munculnya teknologi internet sangat membantu dalam kemudahan serta kecepatan pengiriman, penyampaian dan penerimaan informasi. *web* telah berkembang dengan pesat baik dari segi penggunaan, ukuran, bahasa yang digunakan dan kompleksitasnya. *Web* pada mulanya hanya berupa situs *web* yang bersifat statis dan navigated oriented, serta lebih banyak digunakan dan kompleksitasnya. *Web* telah banyak yang bersifat dinamis, interaktif dan task

oriented untuk digunakan dalam sistem, telekomunikasi, perdagangan, perbankan, dan lain-lain.

Sekolah merupakan suatu badan institusi yang bergerak dalam bidang pendidikan yang memiliki tugas khusus untuk membimbing anak-anak yang ingin bersekolah dan membantu membentuk siswa-siswi yang cerdas melalui pendidikan dan pengembangan budi pekerti, terutama di daerah-daerah yang sulit untuk dijangkau dari perkotaan. Untuk mencapai suatu tujuan yang menciptakan generasi anak bangsa yang

cerdas, mandiri, dan berbudi pekerti luhur, dibutuhkan berbagai upaya staf maupun guru sebagai tenaga pengajar melaksanakan tugas untuk membangun generasi siswa-siswi di Sekolah tersebut.

Perpustakaan Sangat dibutuhkan suatu pengelolaan yang cermat agar kebutuhan guru, siswa maupun staf akademik terhadap pengembangan teknologi yang semakin pesat memudahkan penggunaan dalam mencari informasi dan menyimpan informasi dimana saja khususnya perkembangan *website*, begitu juga pada pengelolaan dan pengarsipan data buku perpustakaan. Sebuah pengarsipan dan pengelolaan data buku yang belum dapat di simpan dalam bentuk file sehingga berkas rentan dengan rusak dan hilang akibat dari kesalahan user (*human error*) dan kesalahan instruksi program. Dibutuhkan suatu aplikasi berbasis *web* pada pengelolaan dan pengarsipan data buku yang berguna untuk mendata berkas pengelolaan dan pengarsipan data buku yang dapat diunggah dan disimpan di dalam *web Server*.

II. TINJAUAN PUSTAKA

2.1. Sistem Informasi

Suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.

(Jogiyanto H.M., 2000 : 63). Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (event) yang nyata (fact) dan digunakan untuk pengambilan keputusan. (Jogiyanto H.M., 2000 : 697).

Sistem Informasi adalah kombinasi antar produsen kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. (Alter, 1992).

2.2. Website

Website adalah sekumpulan halaman-halaman situs yang saling berhubungan, yang berisi teks, gambar, video atau file-file digital lainnya dan biasanya memiliki satu alamat domain atau IP *address* khusus pada jaringan internet. Website ini harus dimuat (hosting) pada sebuah web server sehingga bisa diakses melalui jaringan LAN (*Local Area Network*) yang lebih private. Website dibagi menjadi dua jenis, yaitu :

1. Website Statis

Website Statis hanya memiliki *front end*, yaitu halaman halaman situs yang bisa diakses oleh pengunjung. Biasanya web jenis ini hanya dibuat dengan kode HTML(*hypertext markup language*) dan kadang dilengkapi dengan *script javascript* dan *flash* untuk memperindah tampilan. Konten website statis tidak dapat diupdate karena memang tidak disediakan fasilitas ini. Contoh : *company profile* sederhananya, brosur *online*, atau situs-situs yang berisi informasi sederhana yang tidak perlu diubah.

2. Website Dinamis

Website Dinamis memiliki *front end* dan *back end* . *Front end* adalah halaman-halaman situs yang bisa diakses pengunjung, *back end* yaitu biasa disebut CMS (*content management system*) atau biasa juga disebut Admin Area, atau dunia blog dikenal nama Dashboard, berfungsi untuk mengupdate halaman-halaman

front-end. Jadi halaman back end biasanya hanya bisa diakses oleh pemilik atau pengelola situs dengan user name dan password yang telah ditetapkan.

2.3. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan diagram yang menggunakan notasi atau simbol-simbol untuk menggambarkan sistem jaringan kerja antar fungsi-fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data (Jogiyanto Hartono, 2005). DFD terdiri dari diagram konteks (*context diagram*) dan diagram rinci (*level diagram*). Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan seluruh input ke sistem atau output dari sistem. Sistem dibatasi oleh *boundary* (dapat digambarkan dengan garis putus). Dalam diagram konteks biasanya hanya ada satu proses. Tidak boleh ada store dalam diagram konteks. Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram level di atasnya.

2.4 Unifield Manipulation Language (UML)

Menurut Booch (2005:7) UML adalah Bahasa standar untuk membuat rancangan software. UML biasanya digunakan untuk menggambarkan dan membangun, dokumen artifak dari software –*intensive system*. UML sebagai bahasa model menyatakan bagaimana membuat dan membaca model dengan benar, namun tidak menyatakan model apa yang harus dibuat dan kapan seharusnya dibuat. Disini penulis membuat hanya 4 model dari 8 model yang ada karena pada dasarnya setiap model sudah mempresentasikan bagaimana sistem itu dapat bekerja. Ketiga Model UML tersebut adalah *Use*

Case Diagram, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

2.4.1 Use Case Diagram

Use Case Diagram adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas serta bagaimana sistem tersebut dapat berinteraksi dengan dunia luar dan menjelaskan sistem secara fungsional yang terlihat oleh *user*. *Use Case Diagram* menekankan kepada “apa” yang diperbuat sistem dan bukan “bagaimana” sistem dibuat. Seorang aktor merupakan sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. (Wahono, R. S. 2003). *Use case diagram* dapat sangat membantu apabila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien dan merancang *test case* untuk semua *feature* yang ada pada sistem.

2.4.2 Activity Diagram

Dalam UML juga terdapat diagram yang disebut *activity diagram*. *activity diagram* ini menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana alur sistem ini berawal dan bagaimana sistem berakhir. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. *Activity diagram* menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana *actor* menggunakan sistem untuk melakukan aktivitas. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu

2.4.3 Sequence Diagram

Sequence diagram merupakan interaksi diagram yang digunakan untuk menjelaskan seksekusi sebuah scenario semantic =. *Sequence diagram* juga digunakan untuk menjelaskan interaksi antar objek dalam urutan waktu (Booch, Maksimchuk, Engle, Young, Conallen & Houston, 2007). *Sequence diagram* bisa digunakan untuk menjelaskan sebuah serangkaian langkah-langkah yang mengirimkan pesan antar satu *life line* ke *life line* lain. Setiap pesan yang dikirimkan bisa memberikan respon (*return*) relative pada scenario yang dirancang.

2.4.4 Class Diagram

Class diagram adalah diagram yang menunjukkan kelas-kelas yang ada dari sebuah sistem dan hubungannya secara logika, *Class diagram* menggambarkan struktur statis dari sebuah sistem. Itu sebabnya *class diagram* merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode

2.4.5 Senayan Library Management System (SLIMS)

Senayan Library Management System (SLIMS) adalah perangkat lunak sistem manajemen perpustakaan (Library management system) dengan sumber terbuka yang dilisensikan di bawah GPL v3. Aplikasi ini pertama kali dikembangkan dan digunakan oleh perpustakaan Kementerian Pendidikan Nasional. Seiring perkembangan waktu, aplikasi ini dikembangkan oleh komunitas pengguna dan penggiat SLIMS. Aplikasi SLIMS dibangun dengan menggunakan PHP, basis data MySQL, dan pengontrol versi Git. Pada Tahun 2009, mendapat penghargaan tingkat pertama dalam ajang INAICTA 2009 untuk kategori open source.

III. HASIL DAN PEMBAHASAN

3.1. Rancangan Sistem

Dalam membantu suatu program diperlukan diperlukan perancangan yang sistematis dan terperinci yang dapat memperjelas kerja dari suatu sistem. Perancangan sistem dapat berarti membuat sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem harus dimiliki aliran data masuk dan keluar sejelas mungkin dan mudah dimengerti oleh pengguna. Aliran data dapat berupa penjabaran dari suatu informasi yang utuh ke dalam bagian-bagian komponen yang dimaksud untuk mengidentifikasi dan mengevaluasi permasalahan yang mungkin terjadi selama pengembangan sistem tersebut termasuk mengetahui teknik dan cara-cara dalam pembuatan program aplikasi secara umum. Ada beberapa alat yang dapat dipakai untuk melakukan kegiatan ini di antaranya UML (*Unified Modelling Language*). UML sebagai bahasa model menyatakan bagaimana membuat dan membaca model dengan benar, namun tidak menyatakan model apa yang harus dibuat dan kapan seharusnya dibuat. Disini penulis membuat hanya model 3 model dari 8 model yang ada karena pada dasarnya setiap model sudah mempresentasikan bagaimana sistem itu dapat bekerja. Ketiga Model UML tersebut adalah *Use Case Diagram*, *Class Diagram*, dan *Sequence Diagram*

Tujuan penggunaan dari UML antara lain :

1. Memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi objek.

- Menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

3.1.1 Use Case Diagram

Use Case Diagram adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas serta bagaimana sistem tersebut dapat berinteraksi dengan dunia luar dan menjelaskan sistem secara fungsional yang terlihat oleh user. Use Case Diagram menekankan kepada “apa” yang diperbuat sistem dan bukan “bagaimana” sistem dibuat.

Seorang aktor merupakan sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. (Wahono, R. S. 2003). Use Case Diagram dapat sangat membantu apabila kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rancangan dengan klien dan merancang Test case untuk semua feature yang ada pada sistem. Use Case Diagram untuk aplikasi Sistem Informasi Perpustakaan Digital dapat dilihat pada diagram 3.3



Diagram 3.1 Use Case Diagram

Terdapat dua Actor aplikasi Sistem Informasi Perpustakaan digital ini yaitu Admin, dan User.

Di bawah ini adalah pembahasan dari setiap actor:

Tabel 3.3 Use Case Diagram

No	Use Case	Penjelasan
1	Administrator	Merupakan actor yang mengendalikan semua sistem.
2	User	Menerima, Meminjam data keperpustakaan.

3.1.2 Class Diagram

Pada dasarnya diagram kelas adalah diagram flowchart yang diperluas yang menunjukkan aliran kendali ke aktifitas lain. Kegunaan diagram ini adalah untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem.

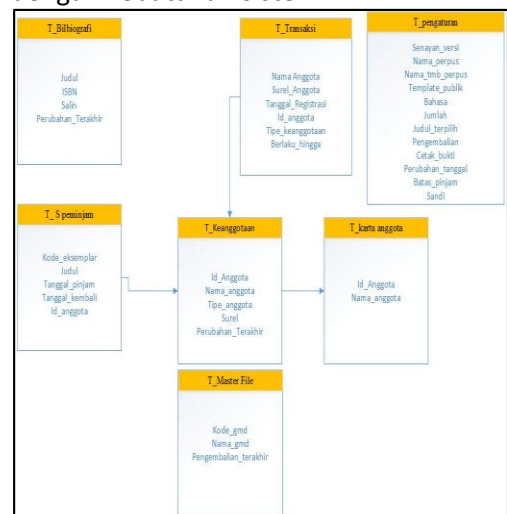


Diagram 3.2 Class Diagram

3.1.3 Sequence Diagram

Sequence diagram dari use case pengelolaan data dapat dilihat pada diagram 3.3

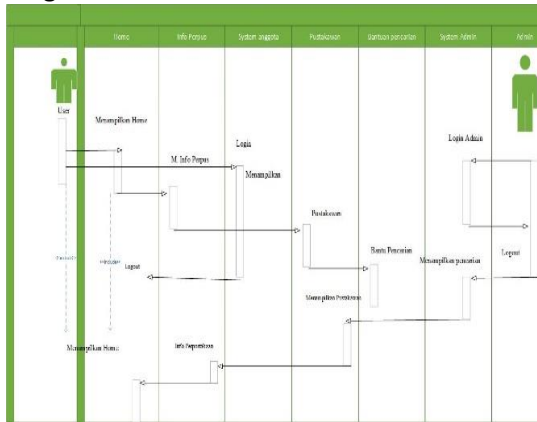


Diagram 3.3 Sequence Diagram

Pada Diagram 3.3 pertama admin menginputkan data katalog buku dan data *user*, langsung disimpan dalam sistem informasi perpustakaan digital. Proses selanjutnya admin melakukan input data *stock* buku masuk dan *stok* keluar. Laporan data katalog buku, bantuan pencarian buku, menampilkan pustakawan, info perpustakaan, menampilkan home dari *user* maupun admin.

3.1.3 Perancangan Basis Data

Basis data dibutuhkan untuk menyimpan data-data perusahaan yang akan ditampilkan di aplikasi. Rancangan basis data yang digunakan dalam aplikasi perpustakaan digital ini menggunakan ERD ((*Entity Relationship Diagram*)). berikut ini adalah entitas dan atribut dari keseluruhan sistem informasi.

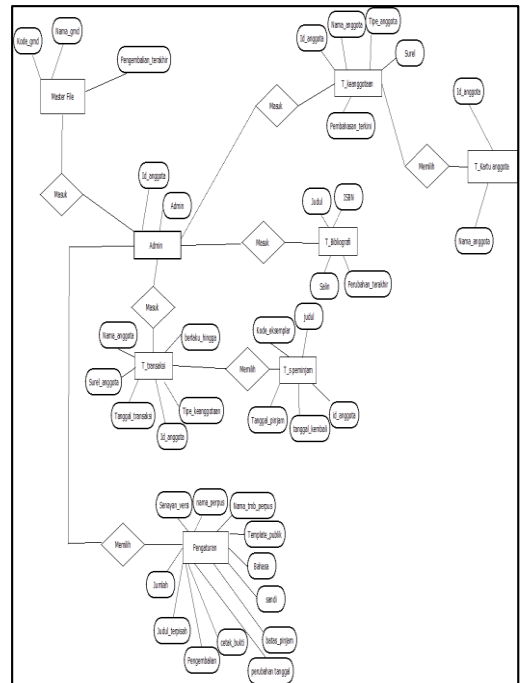
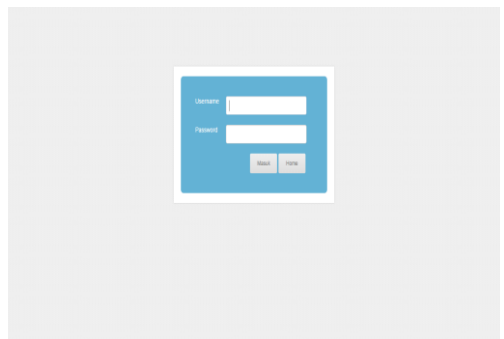


Diagram 3.4 Entitas dan Atribut

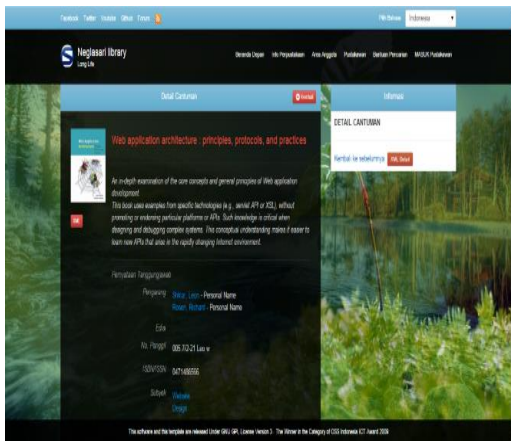
3.1.4. Pembahasan Aplikasi

Saat aplikasi dijalankan, user akan disajikan dengan tampilan *form login* yang berisi menu *login*. Untuk lebih jelas dapat dilihat pada Gambar 3.5. menampilkan halaman *login* yang berisi Menu *dashboard login*, sebelum menggunakan sistem perpustakaan digital ini, *admin* harus melakukan *login* terlebih dahulu.



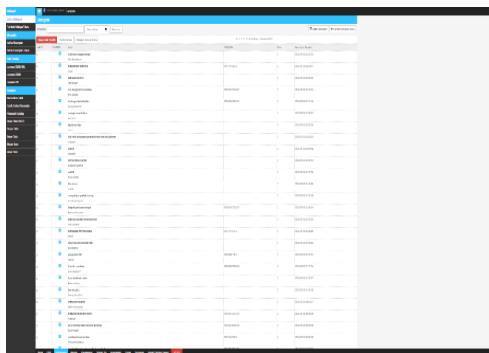
Gambar 3.5 Tampilan Halaman Login

jika *login sebagai admin* maka *username* dan *passwordnya admin*. Setelah *login* berhasil maka muncul tampilan *dashboard home* untuk *admin* seperti pada Gambar 3.6 Pada halaman ini juga terdapat pilihan *admin* untuk mengubah password *admin* seperti pada Gambar 3.6



Gambar 3.6 Modul tampilan Dashboard Utama

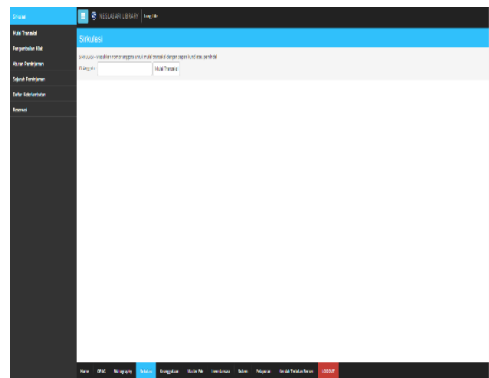
Pada halaman ini menjelaskan tentang Beranda depan, info pustakawan, Area Anggota, bantuan pencarian, forum diskusi *online*, jenis katalog buku yang dicari beserta tombol halaman masuk admin seperti pada gambar 3.5



Gambar 3.6. Modul tampilan Bilbografi

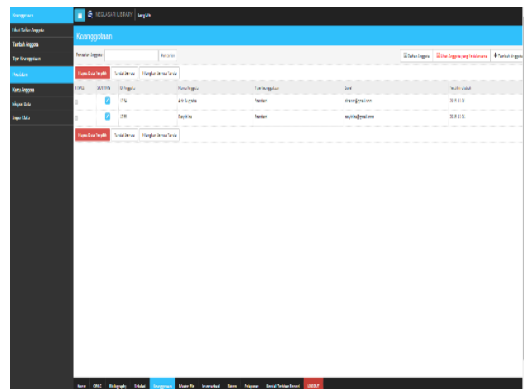
Pada Gambar 3.6 menampilkan halaman data katalog buku , admin bisa mengubah,

menambahkan, menghapus.Modul ini berisi menu Add New Bibliography (membuat data bibliografi/katalogbaru), Bibliographic List (melihat daftar bibliografi), Item List (melihat daftar item/kopi koleksi), Chekout Items (melihat daftar item yang sedang dipinjam). Selain itu, pada modul ini juga dilengkapi dengan copy cataloging, label printing, item barcode printing, import data, ekspor data, item import, item ekspor.



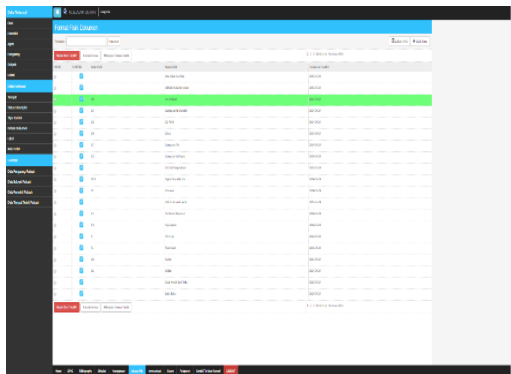
Gambar 3.7 Modul tampilan Sirkulasi

Pada Gambar 3.7 menampilkan proses sirkulasi data katalog buku dari ID anggota



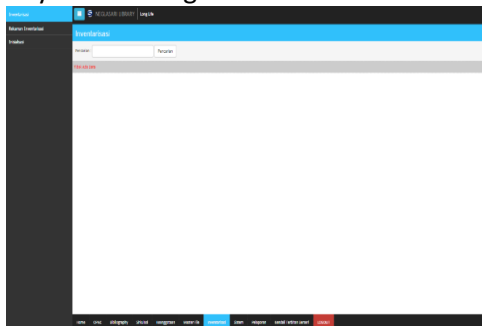
Gambar 3.8 Modul tampilan Keanggotaan

Pada Gambar 3.8 menampilkan dan mengeloladata anggota dari dari ID anggota peminjam.



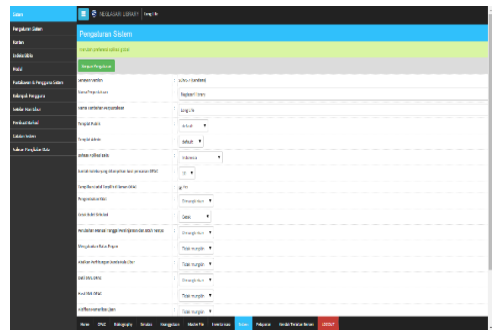
Gambar 3.9 Modul tampilan Master File

Pada Gambar 3.9 Dalam modul Master File admin dapat memasukkan data yang dapat digunakan sebagai master dalam entry data bibliografi.



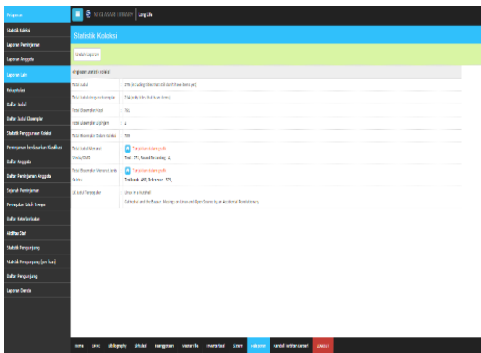
Gambar 4.0 Modul Tampilan Inventarisasi

Pada Gambar 4.0 Dalam Modul Tampilan Inventarisasi admin bisa memasukkan dan mencari data inventaris katalog setiap buku.



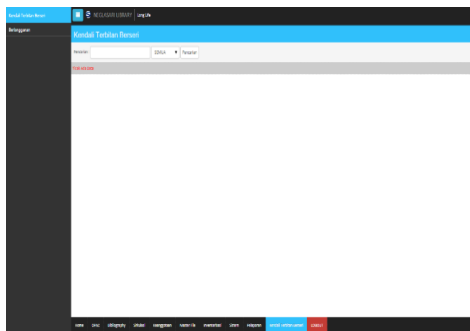
Gambar 4.1 Modul Tampilan Sistem

Pada Gambar 4.1 Pada Modul Form Tampilan Sistem di sini dapat anda gunakan untuk melakukan perubahan preferensi global aplikasi Senayan, seperti *Library Name*, *Public Template* (tampilan OPAC), *Default Application Language*, *Admin Template* (tampilan Admin), *Number of Collections to Show in OPAC Result List* (jumlah koleksi yang akan ditampilkan pada setiap halaman di OPAC), *Show Promoted Titles at Homepage* (menampilkan Judul dalam halamanawal), *Disable/Enable Quick Return* (untuk memperbolehkan pengembalian koleksi dengan metode cepat), *Loan Limit Override* (Pengabaian Batas Pinjam), *Disable/Enable detail XML di OPAC*, *Disable/Enable hasil XML di OPAC*, *setting Allow* (mengizinkan) atau *Forbid* (melarang) pengunjung/pengguna untuk mengunduh file attachmentdi OPAC, *Enable Search Spellchecker* untuk seting pengecekan kata kunci (fitur ini menggunakan *Enchant library*).



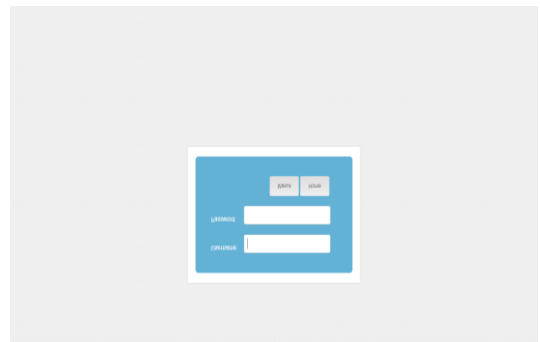
Gambar 4.2 Modul Tampilan Pelaporan

Pada Gambar 4.2 Modul Tampilan Pelaporan Modul ini berisi informasi laporan kegiatan perpustakaan. Informasi tersebut dapat diakses dengan menekan menu yang terdapat pada navigasi sebelah kiri.



Gambar 4.3 Modul tampilan Kendala Terbit Berseri

Pada Gambar 4.3 Modul tampilan Kendala Terbit Berseri merupakan halaman penampilan pencarian judul, subyek, pengarang, ISBN/ISSN yang berkaitan dengan modul tampilan Bibliografi, Master file, dan Sistem.



Gambar 4.4 Modul tampilan Logout

Pada Gambar 4.4 Modul tampilan Logout merupakan halaman tampilan keluar dari sistem yang berkaitan dengan tampilan modul *Dashboard* utama (OPAC)

IV. KESIMPULAN

Berdasarkan Penelitian di atas dapat diambil kesimpulan bahwa:

Admin mampu memantau dan mengelola arsip data buku maupun dokumentasi yang lainnya seperti pengarsipan surat-surat penting dengan mudah dan cepat. Penyimpanan arsip data buku maupun surat-surat penting lebih struktur sehingga mempermudah dalam pencaharian apabila ada saat data buku baik berupa surat-surat maupun dokumentasi yang lainnya diperlukan kembali, dengan tampilan antar muka yang sederhana tetapi tetap fungsional dengan pilihan menu yang mudah dimengerti, Karena Sistem ini berbasis web, maka aksesnya pun bisa dilakukan dimana saja selama admin memiliki Laptop ataupun netbook sendiri.

DAFTAR PUSTAKA

Anonim. 2012. *Entity Relationship Diagram (ERD)*. [Online] Tersedia: <http://indanurafifah.blogspot.com/2012/05/entity-relationship->

- diagram-erd.html [diakses pada tanggal 03 Januari 2016].
- Anonim, 2013. Definisi *Microsoft Excel*. [Online] Tersedia : <https://fatimahabatselamanya.weebly.com/definisi-microsoft-excel.html> [diakses tanggal 15 Agustust 2017].
- Anonim, 2012 Pengertian dan pengenalan slim [Online] Tersedia : <https://tuanpembual.files.wordpress.com/2012/02/pengenalan-slims.pdf> [Diakses 15 maret 2017]
- Purwana, Zakky., 2011., Pupuh [Online] Tersedia: http://noviastutik.blogspot.com/2012/09/diagram-diagram-dalam-umlunified_24.html [diakses pada tanggal 15 Januari 2017]. Pemkab Sumedang, 2012 Profil Pemerintah Kabupaten sumedang [Online] Tersedia : bappeda.sumedangkab.go.id [diakses tanggal 17 maret 2016].
- Sustanta, Edhy. 2004. *Sistem Basis Data*. Yogyakarta: Graha Ilmu
- Wikipedia, 2015 Pengertian Perpustakaan [Online] Tersedia : <https://id.wikipedia.org/wiki/Perpustakaan> [diakses tanggal 2 maret 2016].
- Yuliane, Aneu. 2016, Sistem Berkas, LPPM STMIK DCI, Tasikmalaya
- Yuliane, Aneu. 2016, Penelitian Operasional, LPPM STMIK DCI, Tasikmalaya
- Yuliane, Aneu. ZM, Noer. 2017, Sistem Informasi Pengolahan Data Produksi Dan Distribusi Di Perusahaan Pabrik Tahu Jajang Suparman Js Kecamatan Cihaurbeuti Kabupaten Tasikmalaya, Jurnal Manajemen Informatika 3 (1), LPPM STMIK DCI, Tasikmalaya
- Yuliane, Aneu. DA, Jakaria. 2017, Sistem Informasi Manajemen Keamanan Rukun Warga (Suatu Kajian Ilmiah Yang Diterapkan Di Rw 03 Kampung Babakan Talang Desa Cimari Kecamatan Cikoneng Ciamis), Jurnal Manajemen Informatika 1 (1),LPPM STMIK DCI, Tasikmalaya.
- AS, Ginanjar. Yuliane, Aneu. 2018, Sistem Informasi Kuliner Tasikmalaya, Jurnal Manajemen dan Teknik Informatika (JUMANTAKA) 1 (1), LPPM STMIK DCI, Tasikmalaya
- EA, Trianto. Yuliane, Aneu. 2018, Perancangan Sistem Informasi Pembayaran Abodemen Di Uptd Pasar Rajadesa, Jurnal Manajemen dan Teknik Informatika (JUMANTAKA) 1 (1), LPPM STMIK DCI Tasikmalaya