TEMATICS | Technology Management and Informatics Research Journals

Vol. 5 No. 2 2021

Submitted 09-04-2021; Revised 01-05-2021; Accepted 12-05-2021

APPLICATION FOR DATA INTEGRATION BETWEEN SECTIONS AT THE CLASS I IMMIGRATION OFFICE OF TPI CIREBON
APLIKASI INTEGRASI DATA ANTAR SEKSI DI KANTOR IMIGRASI KELAS I TPI CIREBON

DOI:10.52617/tematics.v3i2.333

Priati Assiroj[1], Arief Febrianto[2], Qodrat Ilhami[3]
Politeknik Imigrasi
Kementerian Hukum dan Hak Asasi Manusia RI
Email: priati.assiroj@poltekim.ac.id [1], arief_feb@poltekim.ac.id [2], ilhamqodrat@gmail.com [3]

Abstrak

Kantor Imigrasi Kelas I TPI Cirebon telah mengembangkan banyak sistem informasi guna mendukung pelayanan keimigrasian di wilayahnya. Permasalahan dari banyaknya sistem informasi yaitu kemungkinan adanya redundansi data, ketidakefektifan pemanfaatan, dan kemungkinan terjadinya duplikasi data pada saat proses input data antar seksi. Saling ketergantungan data antar seksi dalam layanan keimigrasian, menuntut Kantor Imigrasi Kelas ITPI Cirebon menciptakan sistem integrasi data, data-data pada masing-masing seksi akan diintegrasikan ke suatu pangkalan data terpadu berdasarkan konsep *one time entry data*. Penelitian ini bertujuan untuk mengembangkan sebuah inovasi rancang bangun integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon. Integrasi data bertujuan untuk meningkatkan efektivitas layanan dengan mengurangi volume entri data dalam proses layanan dan meningkatkan validitas data. Metode pengambilan data dilakukan melalui metode deskriptif kualitatif, dan pengembangan sistem dibuat dengan metode *Software Development Life Cycle* (SDLC). Pada penelitian ini "Aplikasi Integrasi Data Internal Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon" berbasis web telah berhasil dibuat. Konsep *one time entry data* menghubungkan tiap-tiap menu yang ada di dalam sistem aplikasi dan terintegrasi satu sama lain secara *real- time*, sehingga hal ini dapat mempermudah dan mempercepat proses rekapitulasi data antar seksi khususnya dalam pembuatan laporan bulanan sehingga akan menjadi lebih efektif, efisien, mengurangi redundansi data dan mengurangi ataupun meminimalisasi resiko kesalahan.

Kata kunci : Integrasi Data, *One Time Entry Data*, Data Antar Seksi, SDLC.

Abstract

1st Class Cirebon Immigration Office have developed various information systems to support the immigration services in the region. The problem arise from having various information systems is the possibility of data redundancy, ineffective, and duplicated data during the interdivisional data input process. Dependency of interdivisional data in immigration service demand the immigration office to create the data integration system, integration data means data from each divisions will be integrated/collected in one integrated database based onone time entry data process. This research aims to develop an innovation of interdivisional dataintegration design at 1st Class Cirebon Immigration Office. Data integration aims to increase the service effectiveness by reduce the data entry volume in the service process and increase data validity. The data is taken by descriptive qualitative method, and integration system designed using Software Development Life Cycle (SDLC) method. In this project "Inter- Division Data Integration Apps in the 1st Class Cirebon Immigration Office" the web-based application has been successfully created. The one time entry data-concept connected each divisions menu on the application and integrated one another by the real-time, this applicationwill surely simplify and accelerate the inter-division data recapitulation process especially for the monthly- report making process. This web-based application can make the working flow more efficient, effective, can reduce the data redundance, and increase the validity of data.

Keywords: Data Integration, One Time Entry Data, Interdivisional, SDLC.

PENDAHULUAN

Latar Belakang

Perkembangan zaman yang begitu pesat menjadikan ilmu pengetahuan serta teknologi yang adamengalami perubahan dalam seluruh aspek kehidupan menuju arah yang lebih modern. Teknologi informasi dan komunikasi juga tidak luput dari perkembangan yang signifikan baik dari segi kualitas maupun kecanggihan sistem yang digunakan [1].

Perkembangan *web service* (layanan *web*) sebagai suatu *software* yang didesain untuk mendukung interoperasi antara mesin satu dan lainnya merupakan bagian dari evolusi dan kolaborasi berbagai macam teknologi di masa lampau. *Web service* mampu memberikan keunggulan bagi pemakai teknologi informasi yang mengembangkan perangkat lunak dalam merancang dan membuat suatu sistem yang dapat berinteraksi satu dengan lainnya.

Pembangunan suatu sistem yang baik juga diperlukan pengembangan sistem yang memiliki tahapan yang sempurna, yakni mencakup seluruh tahapan mulai dari tahap perencanaan (*analysis and planning*) hingga tahap pemeliharaan (*maintenance*). *Software Development Life Cycle* (SDLC) dapat membantu mengelola sistem informasi di sebuah instansi, khususnya dalam pengembangan proses integrasi [2]. Semakin berkembangnya teknologi informasi pada semua bidang menuntut peningkatan kinerja baik dari segi kemudahan, kecepatan, maupun efektifitas dan efisiensi [21].

Kemajuan dibidang teknologi informasi akan dapat menunjang proses bisnis pada seluruh instansi swasta maupun pemerintah [3]. Tuntutan masyarakat terhadap pelayanan pemerintahan yang baik (*good governance*) mendorong Direktorat Jenderal Imigrasi membangun dan mengembangkan Sistem Informasi Manajemen Keimigrasian sehingga dapat mempercepat proses terbentuknya pelayanan keimigrasian yang baik. Sistem Informasi Manajemen Keimigrasian atau yang dikenaldengan SIMKIM ialah merupakan kesatuan dari sistem–sistem informasi yang dibuat juga didesain langsung oleh keimigrasian sejak tahun 2008 dan akan terus dikembangkan. Tujuan dari adanya SIMKIM yaitu untuk mencapai optimalisasi kinerja di seluruh satuan kerja jajaran Direktorat Jenderal Imigrasi baik di wilayah Negara Kesatuan Republik Indonesia maupun perwakilan Republik Indonesia di luar negeri. SIMKIM diharapkan mampu mendorong tugas dan fungsi Keimigrasian Indonesia secara efektif dan optimal, sistem yang ada akan diintegrasi secara keseluruhan mulai dari sistem keimigrasian dalam negeri hingga sistem keimigrasian perwakilan Indonesia di luar negeri untuk menjaga tegaknya kedaulatan negara [4].

Dalam rangka mendukung serta melaksanakan program kerja Direktorat Jendral Imigrasi, Sistem Informasi yang ada di Kantor Imigrasi Kelas I TPI Cirebon terus dikembangkan untuk memenuhi

kebutuhan tugas dan fungsi keimigrasian yang lebih baik. Jika aktivitas manajemen data pada sistem dikelola dengan baik maka dapat memberikan manfaat bagi para *user* (pemakai). Tugas dan fungsi tersebut antara lain rekapitulasi data, pengolahan data, evaluasidan validasi data, maupun pencatatan berbagai aktivitas internal antar seksi.

Permasalahan dari banyaknya sistem informasi yang ada yaitu kemungkinan adanya redundansi data, ketidakefektifan data, serta kemungkinan terjadinya kesalahan *input* data pada saat proses rekapitulasi data antar seksi. Saling ketergantungan data antar seksi dalam layanan keimigrasian, menuntut Kantor Imigrasi Kelas I TPICirebon untuk membuat sebuah sistem yang terintegrasi, dimana data yang ada di setiap sistem akanterhubung ke sebuah pangkalan data terpadu (*integrated database*) berdasarkan konsep *one time entry data*.

Sistem informasi di unit kerja Kantor ImigrasiKelas I TPI Cirebon dimungkinkan untuk dikembangkan dan diintegrasikan satu dengan lainnya. Setelah proses *input* dan rekapitulasi kemudian validasi data dilakukan, data tersebut kemudian digunakan sebagai acuan dalam pembuatan laporan bulanan. Dalam penggunaannya saat ini masih menggunakan Microsoft Excel, administrator di Seksi Teknologi Informasi dan Komunikasi Keimigrasian (TIKIM) harus mengolah data kembali setelah diterima dari setiap seksi-seksi yang melaksanakan fungsi teknis keimigrasian secara manual, hal tersebut kurang efektif serta memungkinkan adanya redundansi data dan inefisiensi waktu.

Pembangunan sebuah pangkalan data terintegrasi di Kantor Imigrasi kelas I TPI Cirebon adalah suatu terobosan baru yang dapat berkontribusi dalam peningkatan kualitas dan mutu rekapitulasi serta monitoring data keimigrasian. Integrasi data ini juga bertujuan untuk meningkatkan efektivitas layanan dengan memangkas volume entri data pada proses pelayanan serta meningkatkan proses validitas data. Untukmerealisasikan hal tersebut, dibutuhkan kesiapan dari elemen-elemen yang ada mulai dari teknologi yang akan digunakan dalam pengembangan sistem terintegrasi hingga kesiapan sumber daya manusia (petugas keimigrasian) yang akan secara langsung menggunakan sistem tersebut.Berdasarkan latar belakang yang telah diuraikan dengan ruang lingkup penelitian meliputi Seksi Izin Tinggal dan Status Keimigrasian, Seksi Lalu Lintas Keimigrasian dan Seksi Intelijen dan Penindakan Keimigrasian, serta Seksi Teknologi Informasi dan Komunikasi Keimigrasian untuk menghasilkan sebuah aplikasi integrasi data antar seksi, penulis mengangkat penelitian yang diberi judul **"Aplikasi Integrasi Data Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon".**

Rumusan Masalah

Berdasarkan latar belakang permasalahan di atas, maka penulis merumuskan masalah dalam beberapa poin berikut:

1. Bagaimana proses rekapitulasi data antar seksi yang sudahditerapkan di Kantor Imigrasi Kelas I TPI Cirebon?
2. Bagaimana perancangan dan pembangunan aplikasi integrasi data antar seksi di kantorimigrasi kelas I TPI Cirebon?

Tujuan

Berdasarkan rumusan masalah yang telah ditetapkan di atas, adapun tujuan penelitian yang ingin dicapai yaitu:

1. Untuk mengetahui proses rekapitulasi data antar seksi yang sudah diterapkan di Kantor Imigrasi Kelas I TPI Cirebon.
2. Untuk memberikan gambaran tentang sebuah aplikasi integrasi data.

Manfaat

Berdasarkan rumusan masalah dan tujuan penelitian yang dibahas pada penelitian ini, maka manfaat yang diharapkan dari penelitian ini yaitu:

1. Manfaat Teoritis
   a. Sebagai bahan pertimbangan bagi Kantor Imigrasi Kelas I TPI Cirebon dalam mengembangkan integrasi data antar seksi di wilayah Kantor Imigrasi Kelas I TPI Cirebon.
   b. Mampu memberikan gambaran mengenai proses integrasi data yang meliputi tahapan dan keuntungan aplikasinya, serta hambatan dalam proses pembuatannya jika ada.
2. Manfaat Praktis

Sebagai tugas akhir pada prodi D-IV Manajemen Teknologi Keimigrasian yang merupakan syarat kelulusan dari Politeknik Imigrasi, Badan Pengembangan Sumber Daya Manusia, Kementerian Hukum dan Hak Asasi Manusia.

LANDASAN TEORI

Integrasi Data

Integrasi data merupakan sekumpulan teknik, tahapan, serta teknologi yang bermanfaat untuk mendesain dan membangun proses yang bersifat mengekstrak, merestrukturisasi, mengubah, dan memuat data secara operasional atau melakukan analisis penyimpanan data baik secara *real time* atau dalam *modus batch* [5]. Oleh karena itu, integrasi data adalah kegiatan menggabungkan beberapa data dengan tujuan menyederhankan proses analisis dan berbagi data, untuk mendukung proses manajemen informasi di sebuah instansi. Integrasi data penggabungan data dari berbagai sumber basis data yang berlainan ke *database* atau *data warehouse* [6].

Website

Website atau situsweb merupakan media yang digunakan untuk menampilkan laman yang berisi suatu informasi berupa teks, suara, gambar, animasi, atau bahkan gabungan dari beberapa bahkan semua informasi tersebut baik yang diam atau bergerak dan membentuk suatu rangkaian bangunan yang saling berhubungan, dimana bangunan- bangunan tersebut dihubungan dengan suatu jaringan–jaringan halaman. *Hyperlink* adalah hubungan antara satu laman dengan laman lainnya dan teks yang menghubungkannya disebut *Hypertext* [7]. Pembangunan sebuah website memerlukan elemen-elemen pendukung sebagai berikut:

1. Nama domain atau sering disebut *Uniform Resource Locator* (URL)
2. *Content Manager System* (CMS)
3. Rumah website (*Website Hosting*)

Mengacu pada kerangka hubungan dan penyampaian informasi, layanan *web* dikembangkan dengan empat desain arsitektur, yang setiap modelnya berorientasi pada *message, action, resource,* dan *policy*. Pengembangan desain yang diturunkan berdasarkan orientasi pada pelayanan *Service Oriented Model* (SOM) menghasilkan *Services Oriented Architecture* (SOA), yang merupakan model arsitektur berbasis *service*. Pengembangan desain yang diturunkan berdasarkan orientasi pada sumber daya *Resource Oriented Model* (ROM) menghasilkan *Resource Oriented Architecture* (ROA), yaitu desain arsitektur berbasis sumber daya informasi [8].

***Unified Modelling Language* (UML)**

Unified Modelling Language adalah bahasa pemodelan yang dibuat untuk tujuan umum. Tujuan utama UML yaitu untuk menjelaskan tahapan standar dalam memvisualisasikan bagaimana sebuah sistem telah dirancang. UML serupa dengan *blueprint* yang biasa digunakan di bidang teknik atau lainnya [9].

Sebagaimana bahasa pemodelan lainnya, UML mendefinisikan notasi dan semantik (*syntax*). Notasi UML merupakan sekumpulan dari bentuk spesifik untuk memvisualisasikan bermacam-macam diagram *software*. Masing-masing bentuk mempunyai arti tertentu, dan UML *syntax* menerjemahkan cara untuk dapat menggabungkan bentuk-bentuk tersebut [10]. Berikut adalah beberapa diagram yang dapat didefinisikan dengan UML yaitu [11]:

1. Diagram *statechart*,
2. Diagram *use case*,
3. Diagram aktivitas,
4. Diagram kelas atau *class diagram*,
5. Diagram urutan atau *sequence*,
6. Diagram kolaborasi,
7. Diagram Implementasi,

Metodologi Pengembangan Sistem - *System Development Life Cycle* (SDLC)

Setiap tahunnya teknologi informasi selalu mengalami perkembangan yang begitu pesat. Sistem yang mampu memonitoring, menganalisa, dan mengelola suatu informasi dengan baik diperkenalkan dimana-mana. Namun, perkembangan teknologi informasi tidak hanya terbatas di situ saja, seluruh bidang kehidupan berlomba-lomba untuk mengembangkan langkah yang sangat cepat. Hal ini dilakukan untuk terus meningkatkan produktivitas serta kualitas produk yang dihasilkan, tenaga manusia tergantikan oleh proses yang diautomatisasikan [12].

Berbagai metodologi pengembangan sistem telah diciptakan oleh para ahli perangkat lunak, di antara nya yaitu sistem *Enterprise Resource Planning* (ERP), *Customer Relationship Management* (CRM), dan *Software System Development Life Cycle* (SDLC).

System Development Life Cycle (SDLC) adalah sebuah konsep yang mendefinisikan siklus disuatu organisasi/instansi untuk mengembangkan aplikasi yang dimulai dari fase analisa sebelum aplikasi dikembangkan sampai ke pengujian dan evaluasi setelah aplikasi dikembangkan [13]. SDLC merupakan proses dan terdiri atas beberapa metodologi yang secara umum melalui sudut pandang tingkat tinggi prosesnya terdiri atas aktivitas- aktivitas berikut [14] :

1. Identifikasi kebutuhan
2. Arsitektur dan desain
3. Kodifikasi (penyusunan menurut suatu sistem)
4. Testing (pengujian)
5. Produksi dan pemeliharaan aplikasi

***Text Editor* HTML**

Program dengan ekstensi ".html" tidak membutuhkan *compiler* tertentu untuk dapat mengeksekusinya, namun diperlukan *text editor* html. Terdapat beberapapilihanaplikasi *texteditor* yangdapat digunakan yaitu *Notepad++, Sublime Text Editor, PageBreeze, Freecup HTML Editor*, dan lain sebagainya. *Text editor* yang digunakan oleh penulis pada penelitian ini adalah *Sublime Text Editor*.

Sublime Text Editor merupakan aplikasi yang berfungsi untuk menulis program dan dapat diatur bahasa yang akan ditulis. Di dalam aplikasi ini tersedia beberapa *directory* mengenai fungsi-fungsi yang ada pada tiap program serta dapat memunculkan saran pada saat *programmer* melakukan pengkodingan

***Hypertext Preprocessor* (PHP)**

Hypertext Preprocessor pertama kali ditemukan oleh Rasmus Lerdof pada tahun 1994. PHP merupakan bahasa pemrograman *open – source* yang pengelolaannya dilakukan di dalam server dan dirancang khusus untuk *web*. Dalam setiap kali laman HTML tersebut dikunjungi dapat ditambahkan kode PHP yang akan dijalankan pada laman tersebut. Kode PHP diinterpretasikan pada server *web* kemudian menghasilkan HTML atau *output* lainnya yang dapat dilihat oleh pengguna atau pengunjung [15]. Dibawah ini adalah beberapa elemen penting yang ada di dalam PHP, yaitu:

1. Konstanta (*constants*)
2. *Variable scope*
3. *Variable function*
4. *Operators* dan *precedence*
5. *Expressions*

Dahulu PHP merupakan kepanjangan dari *Personal Home Page*, namun berubah seiring dengan penamaan konvensional GNU (*Gnu's Not Unit*) dan berganti menjadi *Hypertext* Preprocessor. Bahasa pemrograman ini awalnya dibuat pada tahun 1994 oleh seorang pria Rasmus Lerdorf, kemudian diadopsi oleh tiga orang berbakat lainnya dan mengalami sebanyak tiga kali perubahan besar hingga pada akhirnya para pengguna dapat menikmati PHP sebagai bahasa pemrograman yang luas dan matang. Berikut adalah kelebihan PHP dibandingkan dengan Bahasa pemrograman lainnya [15]:

1. Memiliki performa yang sangat baik.
2. Terhubung dengan berbagai sistem *database* yang berbeda.
3. Memiliki perpustakaan bawaan untuk banyak fungsi kerja *web*.
4. Harganya yang relatif murah.
5. Kemudahan dalam pelatihan dan penggunaannya.
6. Portabilitas.
7. Ketersediaan kode sumber (*source code*).

***Database Management System* (DBMS)**

Database (pangkalan data) adalah sekumpulan data-data yang terhubung satu sama lain, data tersimpan baik di *hardware* maupun *software* komputer dapat digunakan untuk memodifikasinya. Data harus disimpan di sebuah basis/pangkalan data untuk menyediakan informasi yang dibutuhkan di waktu tertentu. Berdasarkan sejarahnya, sistem basis data dibentuk setelah periode sistem pemrosesan secara manual (dengan menggunakan kertas) dan sistem pemrosesan berkas terkomputerasi [16].

Database merupakan pondasi dasar dari suatu sistem informasi. Perkembangan teknologi yang begitu pesat akan menghasilkan suatu sistem pangkalan data yang lebih baik untuk digunakan [17]. Basis data yang baik memiliki tiga indikator penting [18], yaitu: *Accessibility,* artinya basis data memiliki kemampuan sebagai penyimpanan atau pengaksesan kembali data-data yang spesifik. *Generality,* artinya basis data memiliki kemampuan untuk dapat mengakses

informasi secara keseluruhan memodifikasi data, serta *flexibility,* artinya basis data memiliki kemampuan dalam memberikan kemudahan bagi pengguna dalam menggunakan serta mengembangkan basis data tersebut [19].

MySQL

Sebuah *database* memungkinkan pengguna memiliki efisiensi dalam penyimpanan, pencarian, penyortiran atau pengelompokkan dan pengambilan data. *Server MySQL* mengontrol akses data untuk memastikan pengguna yang banyak dapat bekerja secara bersamaan, memiliki akses yang cepat ke *database* tersebut, dan juga memastikan hanya pengguna resmi yang dapat mengakses data. Oleh karena itu, MySQL adalah *server* dengan fitur *multi – user* dan *multi threaded*. MySQL menggunakan *Structured Query Language* (SQL), yang merupakan bahasa kueri standar *database* yang dipakai di seluruh dunia [15].

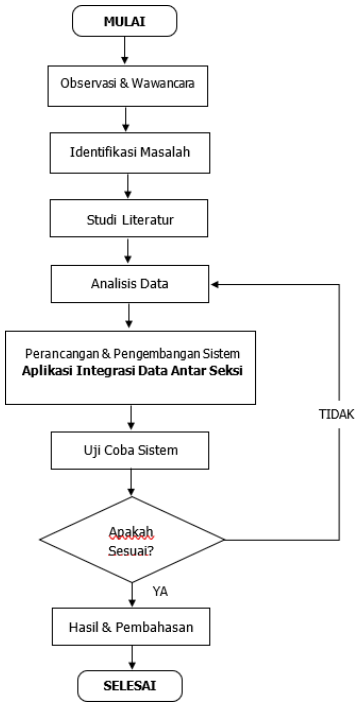
Terdapat berbagai macam *tools* yang dapat digunakan untuk memudahkan administrasi *server* MySQL beberapa *tools* ada yang sudah menjadi *default* dan beberapa dapat pengguna *install* secara gratis maupun berbayar, *tools* tersebut yaitu sebagai berikut [20]:

1. MySQL *Command Line Client*, *tools* ini merupakan *default* yang sudah terinstalasi di dalam file MySQL. *Tools* ini mampu mengkoneksikan dengan MySQL dengan mode berbasis teks (*text – based mode*).
2. MySQL *Front*, *tools* yang merupakan *front-end* MySQL berbasis OS Windows. Keunggulan dari *tools* ini yaitu memiliki *user interface* yang sangat mudah untuk digunakan, bahkan pemula sekali pun.
3. PHPMyAdmin, *tools* ini merupkan *front-end* MySQL berbasis *web* dengan PHP yangdigunakan sebagai bahasa pemrograman. Kelebihan dari *tools* ini yaitu kemampuannya yang dapat mendukung bermacam - macam fitur administrasi MySQL tidak terkecuali memanipulasi data, indeks, dan tabel serta dapat mengekspor data ke format yang beragam, keunggulan lainnya ialah *tools* ini terdiri atas lebihdari 50 bahasa, termasuk bahasa Indonesia.
4. MySQL *Query Browser* dan MySQL *Administrator* adalah *tools* administrasi database MySQL yang tersedia di situs resmi MySQL.

HASIL DAN PEMBAHASAN

Metode penelitian adalah proses ilmiah yang dilakukan untuk memperoleh data yang akandigunakan untuk tujuan penelitian tertentu. Pada penelitian ini metode yang digunakan adalah S*oftware Development Life Cycle* (SDLC) yaitu sebuah siklus yang difungsikan oleh suatu organisasi / instansi dalam pengembangan aplikasi yang dimulai dari tahap analisa sebelum aplikasi dikembangkan sampai ke pengujian dan evaluasi setelah aplikasi dikembangkan [13].

.



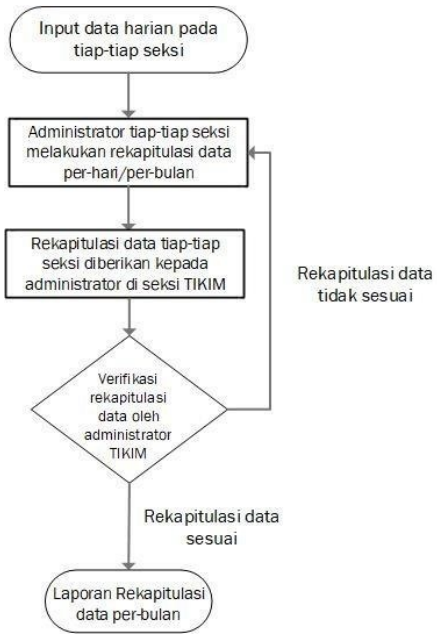
Gambar 1. Prosedur Penelitian

Proses Rekapitulasi Data Antar Seksi yang Sudah Diterapkan di Kantor Imigrasi Kelas I TPI Cirebon

Proses rekapitulasi data merupakan proses yang sangat penting dalam menjalankan proses bisnis suatu organisasi atau kantor. Rekapitulasi dapat juga disebut sebagai proses penyampaian informasi. Rekapitulasi data dapat digunakan untuk mendukung berbagai kegiatan di suatu kantor mulai dari perencanaan, pengawasan, pengendalian, hingga pengambilan keputusan, hingga penetapan kebijakan tertentu. Rekapitulasi merupakan sebuah aktivitas meringkas data sehingga data yang dihasilkan menjadi lebih bermanfaat bentuk, susunan, sifat atau isinya dengan bantuan tenaga tangan atau suatu *tools* tertentu yang membentuk suatu pola atau rumus tertentu.

Kantor Imigrasi Kelas I TPI Cirebon merupakan Unit Pelaksana Teknis (UPT) yang salah satu fungsi nya yaitu sebagai penyedia pelayanan dan fasilitas keimigrasian. Dalam melaksanakan fungsi tersebut terdapat banyak data-data keimigrasian yang dikelola setiap harinya antara lain data permohonan paspor, lalu lintas di Tempat Pemeriksaan Imigrasi (TPI), permohonan izin tinggal, data projustisia, dan lain sebagainya. Data-data tersebut setiap harinya dikelola oleh seksi-seksi yang bertanggung jawab di bidang

tertentu. Untuk menghasilkan koordinasi dan kerja sama yang baik dari setiap seksi yang ada maka perlu dilakukan suatu proses rekapitulasi data yangdilakukan setiap bulannya. Namun, proses rekapitulasi data yang sudah diterapkan di Kantor Imigrasi Kelas I TPI Cirebon masih bersifat manual yaitu menggunakan *Microsoft Excel*, *administrator* di Seksi Teknologi Informasi dan Komunikasi Keimigrasian (TIKIM) harus mengolah data kembali setelah diterima dari setiap seksi secara manual, hal tersebut kurang efektif serta memungkinkan adanya redundansi data dan inefisiensi waktu. Proses rekapitulasi bulanan diKantor Imigrasi Kelas I TPI Cirebon digambarkan padaGambar 2. berikut:



Gambar 2. Alur Rekapitulasi Data Manual di Kantor Imigrasi Kelas I TPI Cirebon

Perancangan dan Pembangunan Aplikasi Integrasi Data Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon

Pembangunan aplikasi integrasi data antar seksi bertujuan untuk meningkatkan kualitas dan mutu rekapitulasi serta monitoring data keimigrasian di Kantor Imigrasi Kelas I TPI Cirebon. Integrasi data ini juga bertujuan untuk meningkatkan efektivitas layanan dengan memangkas volume entri data pada proses pelayanan serta meningkatkan validitas data. Dalam perancangan aplikasi integrasi data antar seksi penulis menggunakan metode pengembangan sistem SDLC sehingga diharapkan aplikasi yang dibangunnantinya dapat dibuat secara terstruktur dan berfungsi dengan baik. Terdapat enam tahapan di dalam SDLC yaitu perencanaan, analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan sistem.

1. Perencanaan

Penulis melakukan wawancara dan observasi terhadap proses laporan bulanan antarseksi yang diterapkan di Kantor Imigrasi Kelas I TPI Cirebon. Berdasarkan hasil observasi dan wawancara, penulis mengetahui sistem laporan bulanan yang masih dilakukan secara konvensional dan membutuhkan waktu yang cukup lama. Proses laporan bulanan antarseksi yang kurang efektif ini menghasilkan perencanaan pembuatan Aplikasi Integrasi Data Antar Seksi yang bertujuan untuk memudahkan proses laporan bulanan yang lebih efektif dan efisien, dengan memanfaatkan teknologi digital lebih dalam.

2. Analisis Kebutuhan Sistem

Proses analisis kebutuhan sistem dilakukan sebagai proses identifikasi dan evaluasi terhadap permasalahan, kesempatan, hambatan yang mungkin terjadi, dan kebutuhan dari sistem aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon sehingga dapat dipersiapkan solusi perbaikan-perbaikannya. Proses analisis dilakukan setelah tahap perencanaan dan sebelum tahap desain sistem.

a. Analisis Fungsional

Adapun fitur-fitur aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis web sebagai sarana pengelolaan laporan bulananadalah:

1) *Superadmin* dapat mengakses dan mengelola keseluruhan data dari setiap seksi yang ada.
2) Administrator (tiap-tiap seksi) dapat mengakses dan mengelola data yang hanya terdapat pada seksi tersebut.

b. Analisis Non-Fungsional

Secara garis besar fungsi dan fitur yang dibutuhkan dalam sistem aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis *web* diantaranya:

1) Kebutuhan Perangkat Keras (*Hardware*)
   Beberapa perangkat keras yang digunakan pada penelitian ini adalah: Laptop Asus (dengan spesifikasi *Processor Intel Core i7*, RAM 8GB DDR4, *keyboard, mouse, modem*) dan PC (dengan spesifikasi *Processor Intel Core i5*, RAM 8GB DDR4, *keyboard*, *mouse*).
2) Kebutuhan Perangkat Lunak (*Software*)
   Beberapa perangkat lunak yang digunakan padapenelitian ini adalah: *Windows 10, XAMPP, Sublime Text, PHP, Java, MySQL, Google Chrome, Microsoft Visio.*

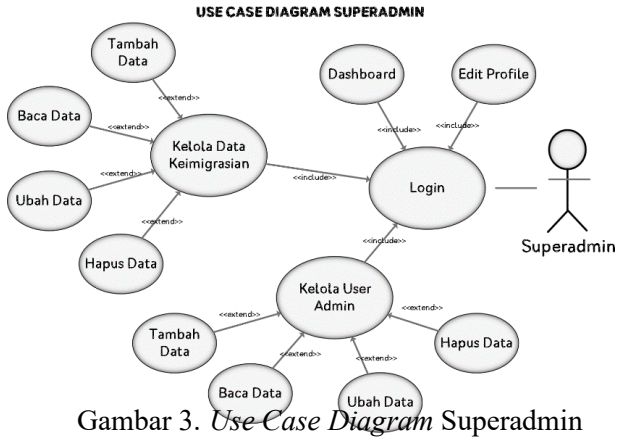
3. Perancangan Sistem (Desain)

Tahap perancangan atau desain sistem dilakukan setelah tahap analisis kebutuhan sistem terpenuhi. Desain sistem dilakukan agar dapat memberikan gambaran terperinci dan jelas mengenai rancang bangun dan implementasi dari sistem aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis *web* yang akan dibuat dan dikembangkan. Pada tahapan ini penulis menggunakan *Unified Modelling Language* (UML) sebagai alat bantu dalam mendefinisikan sistem yang akan dibuat yang terdiri atas *use case diagram*, *activity diagram*, dan *class diagram*.

a. Use Case Diagram

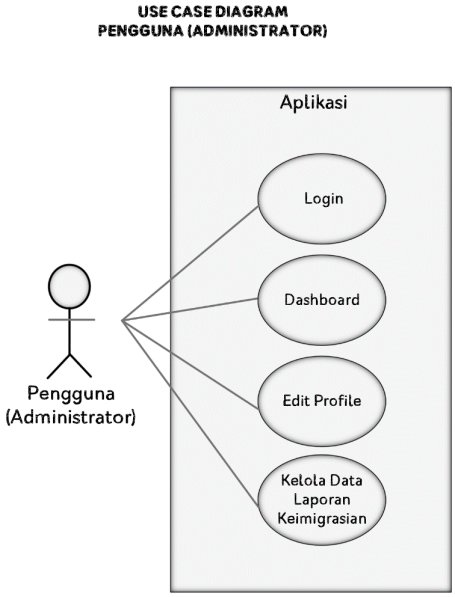
Use case diagram merupakan suatu permodelan yang menggambarkan bagaimana seorang pengguna (*user*) dapat berinteraksi dengan suatu sistem. Pada pembuatan aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis *web*, pengguna aplikasi ini terdiri dari *superadmin* dan administrator dari masing-masing seksi yaitu seksi lalu lintas keimigrasian, izin tinggal dan status keimigrasian, intelijen dan penindakan keimigrasian, teknologi informasi dan komunikasi keimigrasian.

Superadmin dapat mengaksesdan mengelola semua fitur pada aplikasi namun yang membedakan ialah seorang super admin dapat mengakses dan mengelola keseluruhan data yang terdapat di dalam sistem tersebut, sedangkan administrator biasa hanya dapat mengakses dan mengelola data pada seksi atau bidangnya saja. Berikut adalah beberapa interaksi yangdapat dilakukan antara pengguna dan sistem: *edit profile, login*, pengelolaan data (baca data, tambah data,sunting data, serta hapus data). *Use case diagram* sistemdigambarkan pada gambar 3. dan gambar 4. sebagai berikut:



Gambar 3. *Use Case Diagram* Superadmin

Berbeda dengan *superadmin*, berikut adalah interaksi yang dapat dilakukan antara *user administrator* dengan sistem yaitu *login*, *dashboard, edit profile,* dan kelola data keimigrasian pada masing-masing bidangnya. Interaksi antara *user* administrator dengan sistem digambarkan pada gambar 4. di bawah ini:



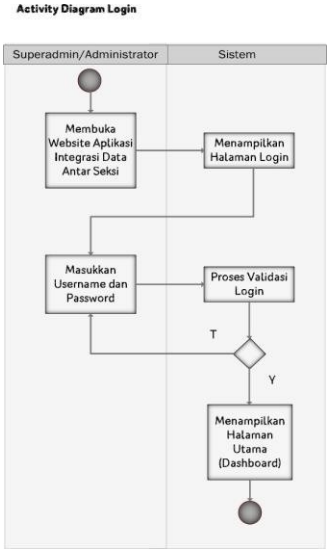
Gambar 4.*Use Case Diagram* Administrator

Berikut adalah tampilan menu kelola data keimigrasian pada tiap-tiap seksi yang terdapat dalam aplikasi integrasi data yang akan dibuat:

1) Seksi Lalu Lintas Keimigrasian
   a) Statistik Penerbitan Dokumen Perjalanan
   b) Statistik Lalu Lintas
2) Seksi Izin Tinggal dan Status Keimigrasian
   a) Informasi dan Komunikasi
   b) Publikasi dan Diseminasi (Sosialisasi)
   c) Pengaduan Masyarakat
3) Seksi Teknologi Informasi dan Komunikasi Keimigrasian
   a) Izin Tinggal Kunjungan
   b) Izin Tinggal Terbatas
   c) Izin Tinggal Tetap
   d) Lain-lain
4) Seksi Intelijen dan Penindakan Keimigrasian
   a) Projustisia
   b) Tindakan Administratif Keimigrasian
   c) TIM PORA

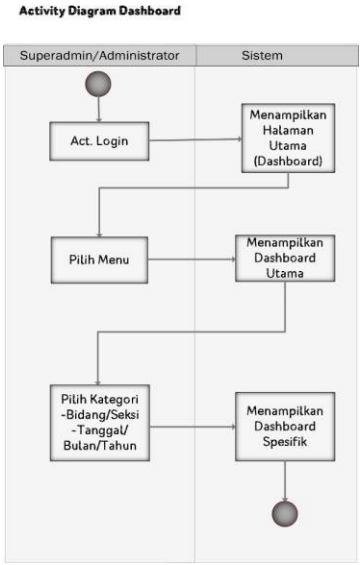
b. Activity Diagram

Activity diagram merupakan diagram yang mendeskripsikan sifat dinamis alamiah di sistem tertentu dengan model aliran dan kontrol antara satu aktivitas ke aktivitas lainnya. Diagram ini dapat menggabarkan urutan proses bisnis secara grafis, tahapan-tahapan sebuah *use* case maupun *behavior logic* dari sebuah objek. Berikut adalah *activity diagram* rancang bangun aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon.



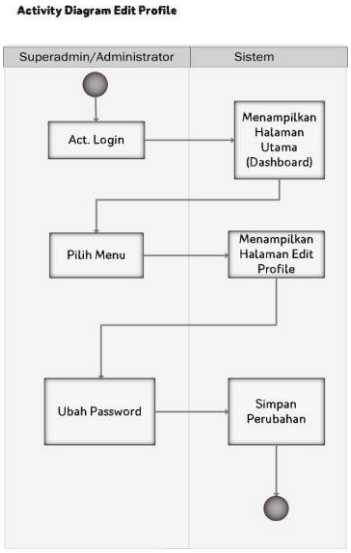
Gambar 5. *Activity Diagram Login*

Pada gambar 5. di atas merupakan *activity diagram login* aplikasi yang akan dibuat. Aktivitas ini dapat dilakukan oleh *user administrator* maupun *superadmin*, dimana aktivitas ini merupakan awalan bagi pengguna untuk dapat masuk ke dalam aplikasi. Setelah berhasil masuk ke aplikasi kemudian aplikasi akan secara otomatis menampilkan halaman utama atau *dashboard* yang memuat informasi mengenai indikator utama dari aktivitas Keimigrasian di Kantor Imigrasi Kelas I TPI Cirebon secara sekilas dalam satu halaman.



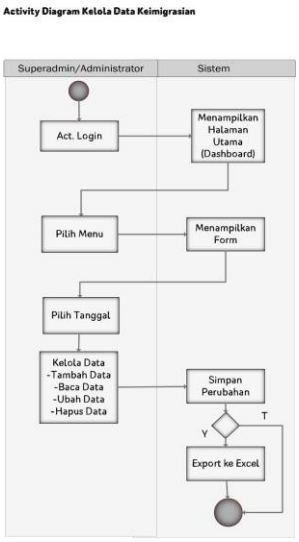
Gambar 6. *Activity Diagram Dashboard*

Setelah pengguna berhasil masuk ke dalam aplikasi dan halaman aplikasi menampilkan *dashboard*, kemudian pengguna dapat memilih kategori *dashboard* untuk melihat data keimigrasian berdasarkan bidang / seksi, maupun tanggal tertentu yang dikehendaki. Setelah pengguna memilih kategori, aplikasi akan menampilkan *dashboard* yang memuat data keimigrasian spesifik tersebut. Aktivitas ini digambarkan dengan jelas seperti pada gambar 6. mengenai A*ctivity Diagram Dashboard*.



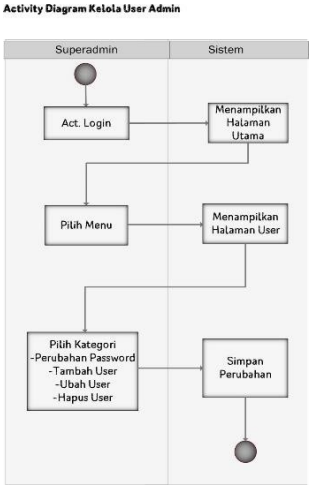
Gambar 7. *Activity Diagram Edit Profile*

Pada gambar 7. menggambarkan *activity diagram edit info*, aktivitas *edit info* (penyuntingan data) dapat dilakukan oleh *user administrator* maupun *superadmin*. Aktivitas penyuntingan data yang dapat dilakukan yaitu penggantian *password* (kata sandi), aktivitas ini betujuan untuk keperluan privasi *login* bagi masing-masing pengguna. Aktivitas ini diawali dari pengguna memilih menu *edit profile* dimana aplikasi nantinya akan menampilkan *form* ubah *password,* pengguna dapat menentukan *password* baru untuk akunnya, setelah itu memilih menu simpan perubahan dan *password* baru pengguna telah berhasil disimpan.



Gambar 8. *Activity Diagram* Kelola Data Keimigrasian

Aktivitas utama pada pembuatan aplikasi integrasi data ini adalah aktivitas kelola data keimigrasian yang telah digambarkan dengan jelas dalam *activity diagram* pada gambar 8. Administrator masing-masing seksi dapat mengelola data keimigrasian pada seksi/bidangnya saja, sedangkan *superadmin* dapat mengelola keseluruhan data keimigrasian pada aplikasi ini. Aktivitas kelola data dimulai dengan pengguna memilih menu *input data* yang telah disediakan dan tanggal, kemudian aplikasi akan menampilkan *form menu* yang dipilih. Setelah *form menu* ditampilkan pengguna dapat melakukan aktivitas kelola data keimigrasian yaitu baca data, tambah data, ubah data, atau hapus data. Kemudian pengguna memilih menu simpan perubahan dan secara otomatis data yang telah diperbaharui akan tersimpan dan terintegrasi dengan sistem *one time entry data*. Selanjutnya apabila dikendaki pengguna juga dapat meng-ekspor data keimigrasian ke dalam format excel (.xls) guna kepentingan tertentu.

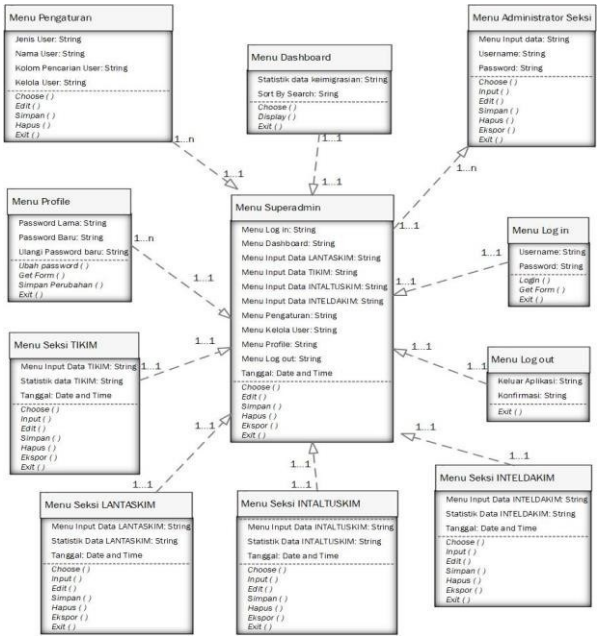


Gambar 9. *Activity Diagram* Kelola *User* Admin

Pada gambar 9. dapat dilihat a*ctivity diagram* kelola *user a*dmin pada aplikasi yang akan dibuat. Aktivitas kelola data *user* admin hanya dapat dilakukan oleh superadmin, diawali dengan *superadmin* memilih menu *user* dan nantinya aplikasi akan menampilkan daftar *user* pada aplikasi tersebut, *superadmin* dapat mengelola dat *user* yang ada dengan aktivitas ganti *password,* tambah *user*, ubah *user*, atau hapus *user*. Kemudian *superadmin* dapat mengklik simpan perubahan sehingga data terbaru akan secara otomatis tersimpan pada sistem.

c. Class Diagram

Class diagram merupakan diagram yang terdiri atas kelas-kelas pada sistem yang menyusun sistem tersebut, diagram kelas menggambarkan hubungan secara logika di antara kelas-kelas yang ada pada sistem. Gambar *class diagram* pada sistem aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPICirebon seperti yang terdapat pada gambar 4.9 berikut:



Gambar 10. *Class Diagram* Aplikasi Integrasi Data Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon

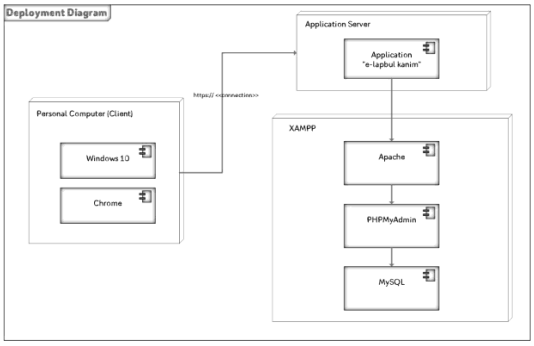
Pada gambar 10. terdapat dua kelas di dalam sistem aplikasi yaitu kelas *superadmin* dan kelas administrator. Kelas superadmin merupakan *user* yang memiliki akses secara keseluruhan pada sistem aplikasi ini, hal ini terjadi karena superadmin memiliki kendali atau kontrol terhadap setiap elemen yang ada pada sistem, sedangkan administrator hanya dapat mengontrol elemen-elemen yang terdapat pada tiap-tiap seksi saja. Menu- menu yang terhubung di dalam sistem terintegrasi satu sama lain secara *one time entry data* termasuk menu input data pada tiap-tiap seksi, sehingga hal ini dapat mempermudah dan mempercepat proses

rekapitulasi data antar seksi khususnya dalam pembuatan laporan bulanan.

d. Deployment Diagram

Deployment Diagram merupakan kebutuhan spesifik suatu sistem yang terdiri atas *software* dan *hardware*. Di dalamnya terdapat penentuan- penentuan konfigurasi dan instalasi *software* pada *hardware*. UML *Deployment Diagram* juga memetakan segmen *software* dari suatu metode ke perangkat yang akan mengimplementasikannya.

Berikut adalah *deployment diagram* yang dibuat penulis untuk mempermudah dalam memahami konfigurasi kebutuhan elemen-elemen pemrosesan pada saat *running* sistem dan juga *software* yang ada di dalamnya. *Deployment diagram* digunakan untuk memetakan *software* ke *processing node*, untuk lebih jelasnya *deployment diagram* dapat diihat pada gambar 4.10:



Gambar 11 *Deployment Diagram*

4. Implementasi

Dalam metode *Software Development Life Cycle* (SDLC), tahap implementasi merupakan tahapan setelah desain atau pengembangan dari desain. Tahapan ini merupakan penerapan atau implementasi suatu sistem aplikasi yang bertujuan untuk mengetahui hasil dari kebutuhan fungsional sistem yang dirancang dan disesuaikan melalui proses *coding* yang menghasilkan sebuah *output* dalam bentuk Aplikasi Integrasi Data Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis web.

a.* Implementasi *Coding

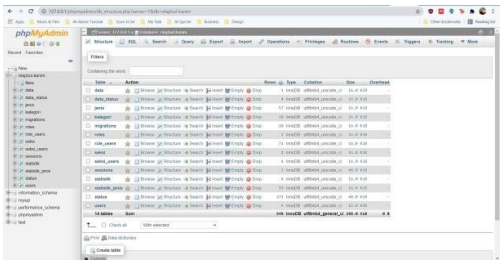
Implementasi *coding* merupakan tahap *plotting* perancangan sistem ke dalam bentuk *coding* bahasa pemrograman. Berikut adalah implementasi *coding* dari aplikasi yang dibuat:

1) Koneksi *Database*

```
//DATABASE CONNECTION
        'mysql' => [
            'driver' => 'mysql',
            'url' => env('DATABASE_URL'),
            'host' => env('DB_HOST', '127.0.0.1'),
            'port' => env('DB_PORT', '3306'),
            'database' => env('DB_DATABASE', 'elapbul-kanim'),
            'username' => env('DB_USERNAME', 'root'),
            'password' => env('DB_PASSWORD', ''),
            'unix_socket' => env('DB_SOCKET', ''),
            'charset' => 'utf8mb4',
            'collation' => 'utf8mb4_unicode_ci',
            'prefix' => '',
            'prefix_indexes' => true,
            'strict' => true,
            'engine' => null,
            'options' => extension_loaded('pdo_mysql') ? array_filter([
                PDO::MYSQL_ATTR_SSL_CA => env('MYSQL_ATTR_SSL_CA'),
            ]) : [],
        ],
```

Gambar 12. *Database Connection*

Fungsi *database connection* terdapat pada file C:\xampp\htdocs\elapbulkanim\config**database.php**. Fungsi ini merupakan fungsi yang menghubungkan antara *database mySQL* dengan *website*, untuk mengakses tersebut dapat menggunakan XAMPP yang telah diaktifkan melalui *local host* dengan *url* 127.0.0.1:8000 nanti akan muncul halaman indeks/login aplikasi. Kemudian, akses untuk menuju *phpMyAdmin* dapat menggunakan url 127.0.0.1/phpMyAdmin. Berikut tampilan phpMyAdmin dapat dilihat pada gambar 13:



Gambar 13**.** Tampilan *phpMyAdmin*

Dengan adanya fungsi *database connection* ini, data-data yang diolah pada *website* dapat tersimpanpada *database* dan berguna apabila data-data tersebut sewaktu-waktu ingin ditampilkan kembali pada*website*.

2) Login

```
//LOGIN
    public function login(Request $request)
    {

        $credentials = $request->only('username', 'password');

        if (Auth::attempt($credentials)) {
            return redirect()->intended('home');
        } else {
            return redirect()->back()->with('errors','Username &amp;amp; Password
                tidak tepat');
        }

    }
}
```

Gambar 14. *Login*

Fungsi *login* terdapat pada file C:\xampp\htdocs\elapbul-kanim\app\Http\Controllers\Auth**LoginController.php**. Dalam fungsi ini terdapat *syntax* **$credentials = $request->only('username', 'password');** yang menjelaskan bahwa untuk dapat *login* diharuskan menggunakan *username* dan *password*

yang sesuai, kemudian jika sesuai akan menuju tampilan *home* atau *dashboard* ditunjukkan dengan *syntax* **if (Auth::attempt($credentials)) {returnredirect()->intended('home');}** jika tidak sesuai akan menampilkan pesan yakni "*username* dan *password* tidak sesuai" ditunjukkan dengan *syntax* **else {return redirect()->back()->with('errors','Username & Password tidak tepat');}**. Fungsi *login* ini merupakan proses validasi *user* yang akan menggunakan aplikasi tersebut, karena setiap masing-masing *user* mempunyai tingkatan aksesnya.

3) Home atau *Dashboard*

```
 9
10  {
11      public function index()
12      {
13          return view('dashboard');
14      }
15  }
```

Gambar 15. *Home* atau *Dashboard*

Fungsi *home atau dashboard* terdapat pada file C:\xampp\htdocs\elapbulkanim\app\Http\Controllers\ **HomeControl ler.php**. Dalam fungsi ini terdapat *syntax* **{public function index()** yang menjelaskan bahwa fungsi indeks merupakan fungsi yang pertama kali diakses, serta fungsi indeks juga merupakan fungsi *default*. Fungsi indeks yaitu fungsi yang pertama kali diakses, setelah fungsi indeks diakses barulah fungsi-fungsi lain yang akan diakses ditunjukkan dalam *syntax* **{return view('dashboard');}}** yang berfungsi untuk menampilkan laman *home atau dashboard*.

4) Tampilan Indeks *Input* Data tiap Seksi

```
62
63  //TAMPILAN INDEX INPUT
64  public function index($e, Request $request)
65  {
66      $data = Kategori::where(['nama_kategori' => $e])->first();
67      $breadcrumbs = array();
68
69      if ($data->up != null) $breadcrumbs[$data->up->nama_kategori] = 'forms/'.$
          data->up->nama_kategori;
70      $breadcrumbs[$data->nama_kategori] = 'forms/'.$data->nama_kategori;
71
72      $this->set($data);
73      $kolom = $this->kolom;
74      $e = $this->status;
75      $status = array();
76
77      for($i = 0; $i < count($e); $i++) {
78        foreach($e[count($e)-1-$i] as $a) {
79          $status[] = $a;
80        }
81      }
```

Gambar 16. Indeks *Input* Data

Fungsi Indeks *input* terdapat pada file C:\xampp\htdocs\elapbulkanim\app\Http\Controllers**F ormController.php**. Dalam fungsi ini terdapat *syntax* **{$data = Kategori:where (['nama_kategori' =>$e])->first(); $breadcrumbs = array();** yang menjelaskan bahwa setiap seksi dapat memasukkan hasil laporan secara berkala kedalam form yang tersedia. *Source code* tersebut akan mengidentifikasi pengguna sesuai dengan seksinya masing- masing. Seperti contoh, *user* tikkim hanya dapat memasukkan laporan pada laman form Input Data Seksi Teknologi Informasi dan Komunikasi Keimigrasian. Hal tersebut berlaku pada seksi-seksi lainnya.

```
82      //dd($kolom);
83
84      $def = array();
85      $tgl = ($request->tgl != null) ? $request->tgl : date('Y-m-d');
86
87      switch ($data->tipe) {
88        case 1:
89          $statistik = StatistikJenis::whereHas('jenis', function($query) use ($data
              ) {
90            $query->where('kategori_id',$data->id);
91          })->where('tgl',$tgl);
92
93          foreach($statistik->get() as $d) {
94            $def[$d->jenis_id][$d->status_id] = $d->val;
95          }
96        break;
97
98        case 2:
99        case 3:
100       $statistik = Statistik::whereHas('status', function($query) use ($data) {
101         $query->where('kategori_id',$data->id);
102       })->where('tgl',$tgl);
103
104       foreach($statistik->get() as $d) {
105         $def[$d->status_id] = $d->val;
106       }
107       break;
108
109       case 4:
110
111       $def = Data::where('kategori_id',$data->id);
112
113       break;
114
115     }
```

Gambar 17. *Input*

Data berdasarkan Tanggal Selanjutnya, dalam fungsi ini terdapat *syntax* **$tgl = ($request->tgl!= null) ? $request->tgl : date('Y-m-d');** yang menjelaskan dimasukkanya laporan sesuai dengan tanggal yang dipilih. Kemudian terdapat *syntax* **$statistik= StatistikJenis::whereHas('jenis', function($query) use ($data){$query->where('kategori_id',$data->id);** yang menjelaskan bahwa *user* dapat memilih kategori sesuai dengan seksinya masing-masing. Setiap *user* hanya dapat mengakses sesuai dengan otoritas setiap seksi. Kemudian terdapat *syntax* **$statistik**= Statistik::whereHas('status', function($query) use ($data)**{$query->where('kategori_id',$data->id);** menjelaskan bahwa setiap kategori yang terdapat pada menu *dashboard*. Kategori tersebut diambil dari *database* yang berada pada tabel kategori.

```
190       case 3:
191         if ($s != null) {
192           $cek = Statistik::where([
193             'tgl' => $request->tgl,
194             'status_id' => $j
195           ])->first();
196           $save = array(
197             'tgl' => $request->tgl,
198             'status_id' => $j,
199             'val' => $s]);
200           if ($cek == null) Statistik::create($save);
201           else $cek->update($save);
202         }
203       break;
204
205       case 4:
206         $cek = DataStatus::where([
207           'data_id' => $data->id,
208           'status_id' => $j,
209         ])->first();
210
211         $save = array(
212           'data_id' => $data->id,
213           'status_id' => $j,
214           'val' => $s
215         );
216         if ($cek == null) DataStatus::create($save);
217         else $cek->update($save);
218
219       break;
220
221     }
222
223   }
224
225   return redirect()->to('form/'.$request->kategori.'?tgl='.$request->tgl)->with(
226     'success','Data '.$request->kategori.' berhasil disimpan');
227
```

Gambar 18. Simpan Data

Selanjutnya, dalam fungsi ini terdapat *syntax* **if ($cek == null) Statistik::create($save);else$cek->update($save);** menjelaskan data yang telah dimasukkan berhasil disimpan kedalam *database*. Apabila laporan telah berhasil disimpan, akan memunculkan pesan yang ditunjukkan dalam *syntax* **return redirect()->to('form/'.$request->kategori.'?tgl='.$request->tgl)**->with('success','Data '.$request->kategori.' berhasil disimpan');.

5) Indeks Statistik

```
94      switch ($data->tipe) {
95        case 1:
96        $select = 'status_id,jenis_id';
97        $statistik = StatistikJenis::whereHas('jenis', function($query) use ($data)
            {
98            $query->where('kategori_id',$data->id);
99          });
100       break;
101
102       case 4:
103       $statistik = Data::where('kategori_id',$data->id);
104       break;
105
106       default:
107       $select = 'status_id';
108       $statistik = Statistik::whereHas('status', function($query) use ($data) {
109         $query->where('kategori_id',$data->id);
110       });
111       break;
112     }
```

Gambar 19. Indeks Statistik

Fungsi Indeks Statistik terdapat pada file C:\xampp\htdocs\elapbulkanim\app\Http\Controllers**StatistikController.php**. Dalam fungsi ini terdapat *syntax* $statistik = StatistikJenis::whereHas('jenis',function($query)use($data){$query**->where('kategori_id',$data->id);** yang menjelaskan bahwa setiap submenu memiliki data statistik masing-masing. Data statistik diambil berdasarkan **kategori_id** yang berada pada tabel jenis didalam *database*.

```
114     $periode = $request->periode;
115
116     $excelUrl = array('export=excel');
117     $excelUrl[] = 'periode='.$periode;
118
119   //
120     if ($data->tipe != 4)
121     {
122       switch ($periode) {
123         case 'Bulanan':
124           $statistik->selectRaw('sum(val) val, '.$select)
125           ->groupBy(DB::raw('MONTH(tgl),'.$select))
126           ->whereRaw("MONTH(tgl) = '".$request->bulan."'");
127           $excelUrl[] = 'bulan='.$request->bulan;
128         break;
129         case 'Tahunan':
130           $statistik->selectRaw($select)
131             ->groupBy(DB::raw('YEAR(tgl), '.$select))
132             ->whereRaw("MONTH(tgl) = '".$request->tahun."'");
133           $excelUrl[] = 'tahun='.$request->tahun;
134         break;
135         default:
136           $statistik->where('tgl',$tgl);
137           $excelUrl[] = 'tgl='.$request->tgl;
138         break;
139       }
140
```

Gambar 20. Menampilkan Data

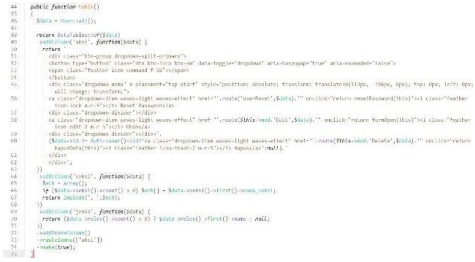
Selanjutnya, dalam fungsi ini terdapat *syntax* **$periode = $request->periode;** menjelaskan data yang telah disimpan sementara dapat ditampilkan kembali dengan memilih beberapa opsi seperti, tanggal, bulan, dan tahun dimasukkannya data. Selanjutnya terdapat *syntax* **case 'Bulanan': , case 'Tahunan': ,** dan **default:** yang akan menampilkan hasil statistik sesuai opsi yang dipilih atau diinginkan.

```
152     switch ($request->export) {
153       case 'excel':
154
```

Gambar 21. *Request Export*

Selanjutnya, dalam fungsi ini terdapat *syntax* **switch ($request->export) { case'excel':** menjelaskan bahwa fungsi tersebut meminta dokumen berupa *excel* yang dapat diunduh kedalam komputer masing-masing pengguna.

6) Kelola User



Gambar 24. Kelola *User*

Fungsi kelola *user* terdapat pada file C:\xampp\htdocs\elapbul-kanim\app\Http\Controllers**UserController.php**. Dalam laman ini menampilkan beberapa *user* yang berperan dalam penggunaan aplikasi. Fitur yang terdapat pada laman ini hanya dapat diakses oleh *superadmin* yang menampilkan *user* yang terdaftar, *username* yang digunakan, jenis seksi, menambahkan *user* serta aksi untuk mengelola *user*. Fitur aksi ditunjukkan dalam *syntax* ***->addColumn('aksi', function($data)*** yang menjelaskan aksi yang dapat dilakukan seperti *reset password,* ubah jenis, dan hapus *user*.

```
77    public function create()
78    {
79      $mod = $this->mod;
80      $title = $this->title;
81      $url = route('userWrite');
82      $roles = Role::orderBy('name');
83      $seksi = Seksi::orderBy('nama_seksi');
84      return view('forms.user', compact('mod','title','url','roles','
          seksi'));
85    }
```

Gambar 25. Tambah *User*

Selanjutnya fungsi tambah *user*, fungsi ini terdapat *syntax* **public function create()** yang memiliki peran dalam menambahkan *user* pada sistem sesuai dengan seksi yang diinginkan ditunjukkan dengan *syntax* **$seksi = Seksi::orderBy('nama_seksi');**.

```
87    public function edit($id)
88    {
89      $def = User::findBySlug($id);
90      $mod = $this->mod;
91      $title = $this->title;
92      $url = route('userWrite');
93      $roles = Role::orderBy('name');
94      $seksi = Seksi::orderBy('nama_seksi');
95      return view('forms.user',compact('def','mod','title','url','
          roles','seksi'));
96    }
```

Gambar 26. Ubah *User*

Kemudian fungsi ubah *user*, fungsi ini terdapat *syntax* **public function edit($id)** yang memiliki peran dalam merubah akun *user* yang sudah terdaftar seperti Nama *User*, *Username*, Jenis *User* hingga Seksi.

```php
public function delete($id)
{
  $data = User::findBySlug($id);

  foreach ($data->roles()->get() as $r)
  {
    $data->revokeRole($r->slug);
  }

  if ($data->seksi()->count() > 0)
  {
    foreach ($data->seksi()->get() as $s) {
      $data->revokeSeksi($s->id);
    }
  }
  $data->delete();

  die(json_encode(array(
    'url' => url('user'),
    'message' => 'User berhasil dihapus ...'
  )));
}
```

Gambar 27. Hapus *User*

Fungsi hapus *user*, di dalam fungsi ini terdapat *syntax* **public function delete($id)** yang memiliki peran dalam menghapus *user* yang telah terdaftar dalamsistem *database* aplikasi.

7) Struktur Excel & Ekspor Excel

```php
public function data($data)
{
    $this->data = $data;
    return $this;
}

//STRUKTUR EXCEL
public function registerEvents(): array
{
    $col = count($this->data['status']);
    $pluscol = 0; $autosize = null;
    if (in_array($this->data['data']->tipe,array(1,3))) {
      $autosize = 'C';
      $pluscol = 2;
    }
    $alph = $this->data['alphabet'][$col+$pluscol];
    $rows = $this->data['data']->jenis()->count()+count($this->data['kolom'])+1;

    return [
        AfterSheet::class    => function(AfterSheet $event) use ($col,$alph,$
          rows,$autosize) {

          $event->sheet->getStyle('A1:'.$alph.'1')->applyFromArray(['font' => ['
            name' => 'Arial','size' => 15, 'weight' => 'bold']]);
          $event->sheet->getStyle('A2:'.$alph.$rows)->applyFromArray(['font' =>
            ['name' => 'Arial','size' => 11]]);
          $event->sheet->getStyle('A1:'.$alph.(count($this->data['kolom'])+1))->
            applyFromArray([
            'alignment' => [
                'horizontal' => \PhpOffice\PhpSpreadsheet\Style\Alignment::
                  HORIZONTAL_CENTER,
            ],
```

Gambar 22. Struktur *Excel*

Fungsi Struktur Excel terdapat pada file C:\xampp\htdocs\elapbul-kanim\app\Exports**ReportSheet.php**. Dalam file ini terdapat struktur pengolahan data mentah yang kemudian di *export*menjadi data dalam bentuk excel. Terdapat *syntax* **public function data($data) {$this->data = $data; return $this; }** yang menjelaskan bahwa data yang akan di *export* menjadi dokumen *excel* sesuai dengan kebutuhan pengguna. Kemudian, *syntax* **$event->sheet->getStyle('A1:'.$alph.'1')->applyFromArray(['font' => ['name' => 'Arial','size' => 15, 'weight' => 'bold']]);** menjelaskan salah satu format yang digunakan ketika data berhasil di *export* kedalam *excel.*

```php
public function view(): View
{
  return view('reports.excel.'.$this->data['blade'], [
    'data' => $this->data
  ]);
}
```

Gambar 23. Unduh *Excel*

Selanjutnya dalam fungsi ini terdapat *syntax* **return view('reports.excel.'.$this->data['blade'], ['data' => $this->data]);** menjelaskan bahwa fungsi tersebut untuk mengunduh hasil data yang telah diekspor menjadi dokumen *excel*.

b. Implementasi Antar Muka (*Interface*)

Implementasi antar muka dari aplikasi (*interface*) merupakan bagian yang menggambarkan halaman-halaman yang ditampilkan pada bagian awal dan akhir serta bagian utama penggunaan program. Beikut adalah implementasi antar muka (*interface*) pada aplikasi yang telah dibuat:

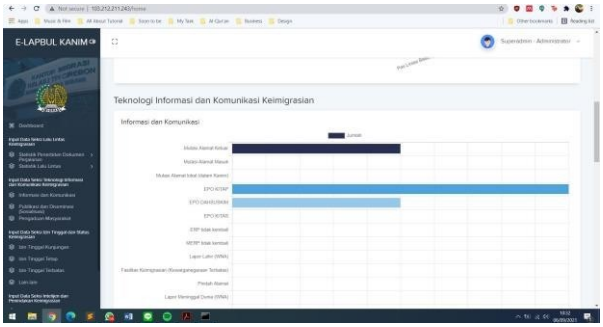
1) Halaman Indeks

Halaman indeks merupakan halaman *default* saat website aplikasi pertama kali dibuka dengan domainnya, halaman ini berisikan kolom *username* dan *password* yang digunakan oleh *user* untuk masuk atau *login* ke dalam aplikasi sesuai dengan hak akses masing-masing *user*. Tampilan halaman indeks dapat dilihat seperti yang terdapat di gambar 24. di bawah ini:

Gambar 24. Halaman Indeks Aplikasi

2) Halaman *Dashboard*

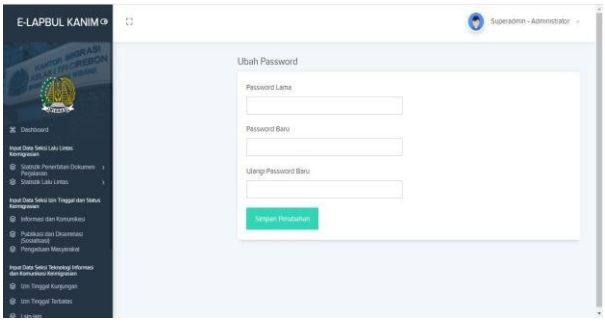
Halaman *dashboard* merupakan halaman utama yang akan muncul ketika *user* berhasil *login* ke dalam aplikasi. Pada halaman *dashboard* akan menampilkan statistik semua data secara keseluruhan dari setiap seksi dan kategori yang ada, seperti yang digambarkan pada gambar 25. sebagai berikut:



Gambar 25. *Dashboard* Aplikasi

3) Halaman Menu Profil

Halaman menu profil dapat diakses oleh masing-masing *user* ketika sudah berhasil masuk ke dalam aplikasi. Fitur pada halaman menu profil berisi *form* untuk mengubah *password* pada akun *user* bersangkutan. Untuk dapat mengubah *password* seorang *user* harus mengisi kolom *password* lama, *password* baru, dan konfirmasi *password* baru, kemudian simpan perubahan. Dapat dilihat form halaman menu profil seperti pada gambar 26. berikut ini:

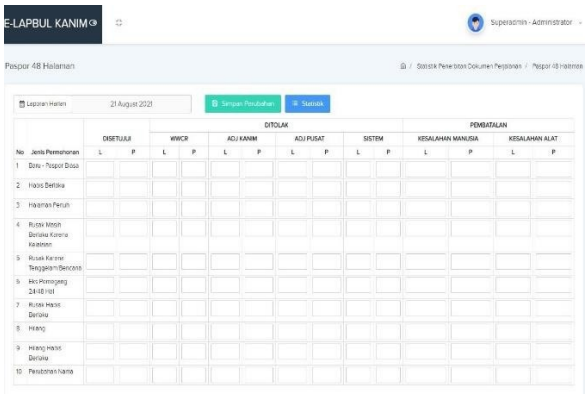


Gambar 26. Halaman Menu Profil (Ubah *Password*)

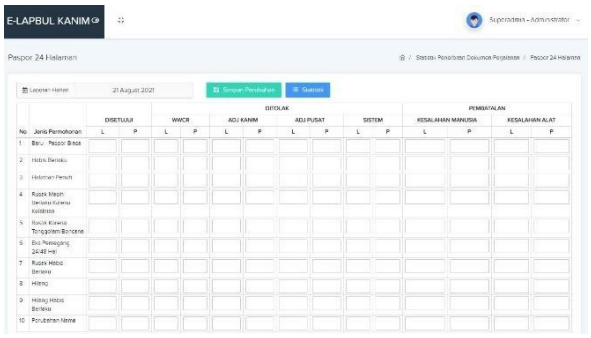
4) Halaman Statistik Penerbitan Dokumen Perjalanan

Halaman statistik penerbitan dokumen perjalanan merupakan fitur bagian dari menu input data seksi lalu lintas keimigrasian yang terdiri atas form penerbitan dokumen perjalanan paspor 48 halaman, paspor 24 halaman, layanan paspor LTSA Cirebon, *E- passport,* dan pas lintas batas (plb) & surat perjalanan laksana paspor (splp). Form statistik penerbitan dokumen perjalanan terdiri dari 3 kategori yaitu permohonan disetujui, ditolak, dan pembatalan. Kategori yang membedakan antara form statistik penerbitan dokumen satu dan lainnya adalah jenis permohonan.

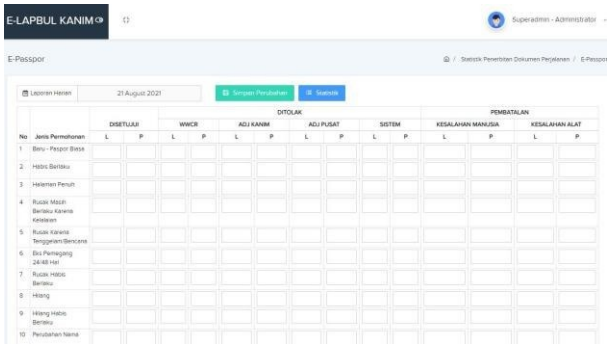
Form penerbitan dokumen paspor 48 halaman, 24 halaman dan*E- Passport* memiliki 10 jenis permohonan yaitu baru-paspor biasa, habis berlaku, halaman penuh, rusak masih berlaku karena kelalaian, rusak karena tenggelam/bencana, eks pemegang 24/48 halaman, rusak habis berlaku, hilang, hilang habis berlaku, dan perubahan nama seperti yang dapat dilihat pada gambar 4.30 sampai 4.32 di bawah ini:



Gambar 27. Halaman Penerbitan Dokumen Perjalanan Paspor 48 Halaman

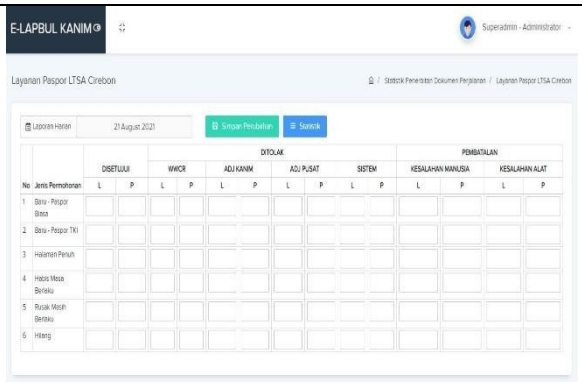


Gambar 28. Halaman Penerbitan Dokumen Perjalanan Paspor 24 Halaman



Gambar 29. Halaman Penerbitan Dokumen Perjalanan *E-Passport*

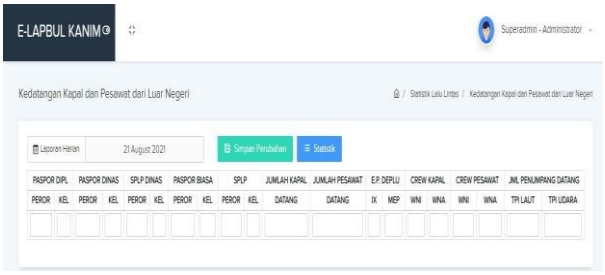
Dokumen perjalanan selanjutnya adalah layanan paspor LTSA Cirebon yang terdiri atas 6 jenis permohonan yaitu baru-paspor biasa, baru-paspor TKI, halaman penuh, habis masa berlaku, rusakmasih berlaku, dan hilang. Halaman dokumen perjalanan layanan paspor LTSA sebagaimana digambarkan pada gambar 30. berikut:



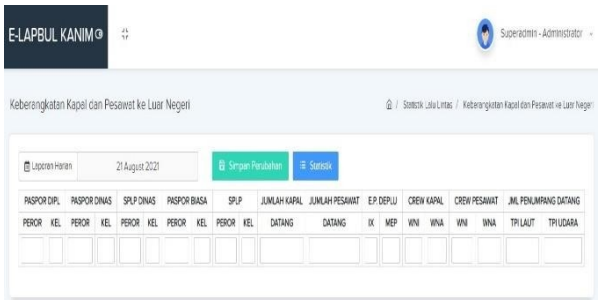
Gambar 30. Halaman Penerbitan Dokumen Perjalanan Layanan Paspor LTSA Cirebon

5) Halaman Statistik Lalu Lintas

Halaman statistik lalu lintas merupakan fitur bagian dari menu input data seksi lalu lintas keimigrasian yang terdiri atas statistik lalu lintas kedatangan kapal dan pesawat dari luar negeri dan keberangkatan kapal dan pesawat ke luar negeri. Form statistik lalu lintas terdiri dari 11 kategori yaitu paspor diplomasi, paspor dinas, splp dinas, paspor biasa, splp, jumlah kapal, jumlah pesawat, deplu, *crew* kapal, *crew* pesawat, dan jumlah penumpang. Halaman statistik lalu lintas kedatangan kapal dan pesawat dari luar negeri dan keberangkatan kapal dan pesawat ke luar negeri dapat dilihat pada gambar 4.34 dan 4.35 sebagai berikut:



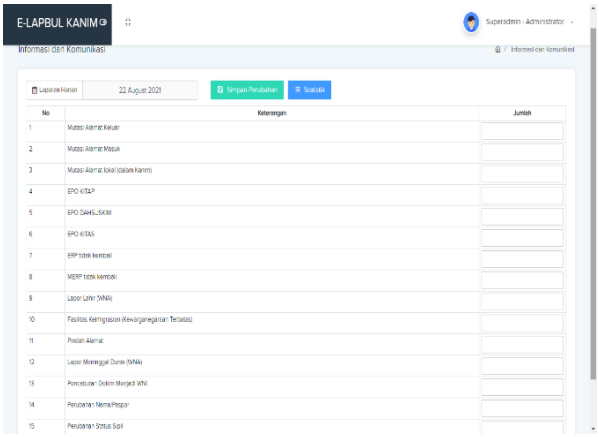
Gambar 31. Halaman Statistik Lalu Lintas Kedatangan Kapal dan Pesawat dari Luar Negeri



Gambar 32. Halaman Statistik Lalu Lintas Keberangkatan Kapal dan Pesawat ke Luar Negeri

6) Halaman Informasi dan Komunikasi

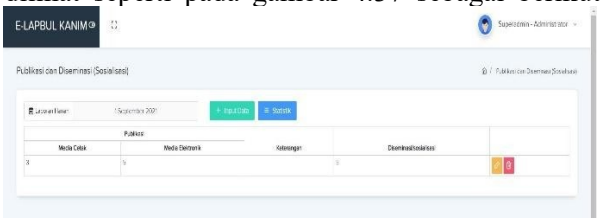
Halaman Informasi dan Komunikasi merupakan fitur bagian dari menu input data seksi Teknologi Informasi dan Komunikasi Keimigrasian. Formhalaman informasi dan komunikasi terdiri dari 15 kategori yaitu mutasi alamat keluar, mutasi alamat masuk, mutase alamat lokal (dalam kanim), EPO KITAP, EPO DAHSUSKIM, EPO KITAS, ERP tidak kembali, MERP tidak kembali, lapor lahir (WNA), fasilitas keimigrasian (kewarganegaraan terbatas), pindah alamat, lapor meninggal dunia (WNA), pencabutan dokim menjadi WNI, perubahan nama/paspor, dan perubahan status sipil. Form halaman informasi dan komunikasi dapat dilihat seperti pada gambar 32. sebagai berikut:



Gambar 32. Halaman Informasi dan Komunikasi

7) Halaman Publikasi dan Diseminasi (Sosialisasi)

Halaman Publikasi dan Diseminasi (Sosialisasi) merupakan fitur bagian dari menu input data seksi Teknologi Informasi dan Komunikasi Keimigrasian. Form Halaman Publikasi dan Diseminasi (Sosialisasi) terdiri dari Publikasi; Media Cetak dan Media Elektronik, Keterangan, dan Diseminasi (Sosialisasi). Halaman Publikasi dan Diseminasi (Sosialisasi) dapat dilihat seperti pada gambar 4.37 sebagai berikut:



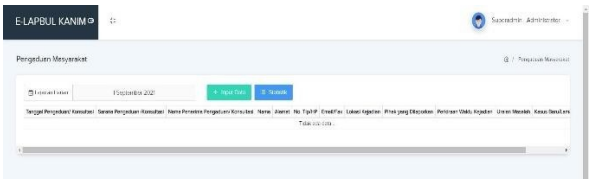
Gambar 33. Halaman Publikasi dan Diseminasi (Sosialisasi)

8) Halaman Pengaduan Masyarakat

Halaman Pengaduan Masyarakat merupakan fitur bagian dari menu input data seksi Teknologi Informasi

dan Komunikasi Keimigrasian. Form Halaman Pengaduan Masyarakat terdiri dari Tanggal Pengaduan/Konsultasi, Sarana Pengaduan/Konsultasi, Nama Penerima Pengaduan/Konsultasi, Nama, Alamat, No.Tlp/HP, Email/Fax, Lokasi Kejadian, Pihak yang Dilaporkan, Perkiraan Waktu Kejadian, Uraian Masalah, Kasus Baru/Lama, Tindak Lanjut. Form Pengaduan Masyarakat dapat dilihat seperti pada gambar

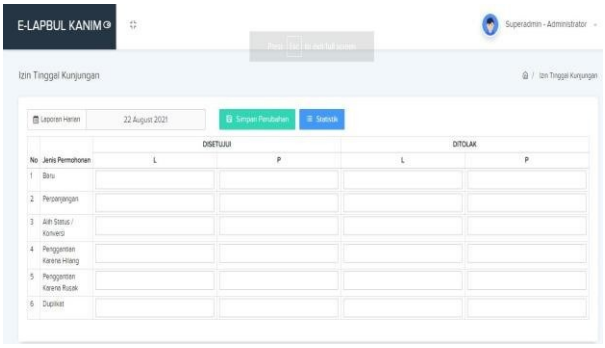
34. sebagai berikut:



Gambar 34. Halaman Pengaduan Masyarakat

9) Halaman Izin Tinggal Kunjungan

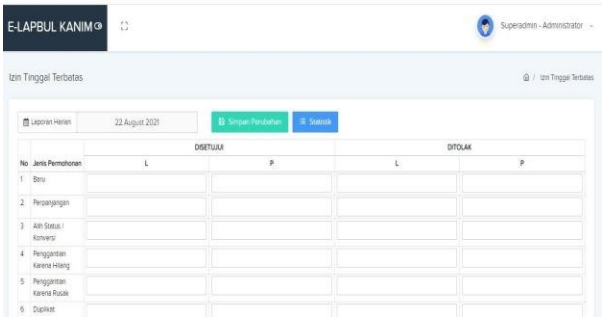
Halaman izin tinggal kunjungan merupakan fitur bagian dari menu input data seksi Izin Tinggal dan Status Keimigrasian. Form halaman izin tinggal kunjungan terdiri dari 2 kategori yaitu disetujui dan ditolak, dan terdapat 6 jenis permohonan yaitu baru, perpanjangan, alih status/konversi, penggantian karena hilang, penggantian karena rusak, dan duplikat. Form halaman izin tinggal kunjungan sebagaimana digambarkan pada gambar 35.:



Gambar 35. Halaman Izin Tinggal Kunjungan

10) Halaman Izin Tinggal Terbatas

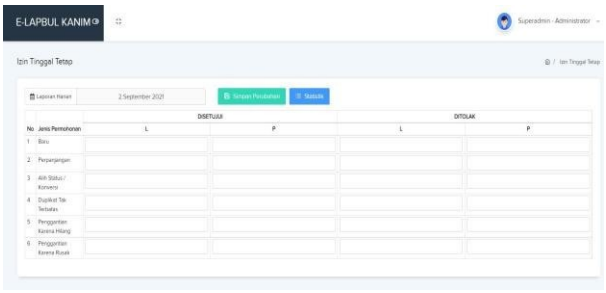
Halaman izin tinggal terbatas juga merupakan fitur bagian dari menu input data seksi Izin Tinggal dan Status Keimigrasian. Form halaman izin tinggal kunjungan terdiri dari 2 kategori yaitu disetujui dan ditolak, dan terdapat 6 jenis permohonan yaitu baru, perpanjangan, alih status/konversi, penggantian karena hilang, penggantian karena rusak, dan duplikat. Form halaman izin tinggal terbatas sebagaimana digambarkan pada gambar 4.40:



Gambar 36. Halaman Izin Tinggal Terbatas

11) Halaman Izin Tinggal Tetap

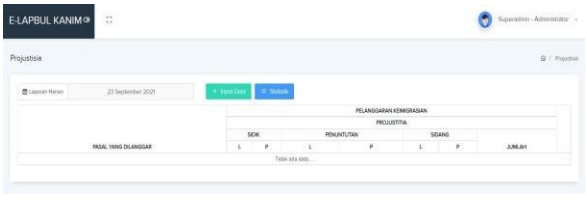
Halaman Izin Tinggal Tetap juga merupakan fitur bagian dari menu input data seksi Izin Tinggal dan Status Keimigrasian. *Form* halaman izin tinggal kunjungan terdiri dari 2 kategori yaitu disetujui dan ditolak, dan terdapat 6 jenis permohonan yaitu baru, perpanjangan, alih status/konversi, duplikat tak terbatas, penggantian karena hilang, dan penggantian karena rusak. *Form* halaman izin tinggal tetap sebagaimana digambarkan pada gambar 37:



Gambar 37. Halaman Izin Tinggal Tetap

12) Halaman Projustisia

Halaman izin projustisia merupakan fitur bagian dari menu input data seksi intelijen dan penindakan keimigrasian. *Form* halaman projustisia terdiri dari 4 kategori yaitu pasal yang dilanggar, pelanggaran keimigrasian; projustisia, sidang, dan jumlah. *Form* halaman projustisia sebagaimana digambarkan pada gambar 38.:

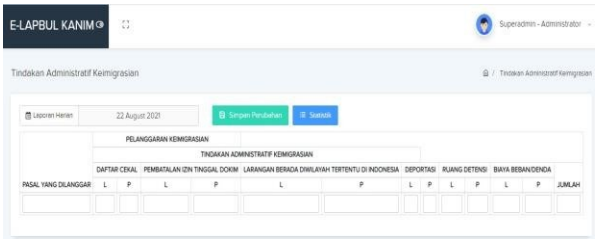


Gambar 38. Halaman Projustisia

13) Halaman Tindakan Administratif Keimigrasian

Halaman Tindakan administratif keimigrasian merupakan fitur bagian dari menu input data seksi intelijen dan penindakan keimigrasian. Form halaman Tindakan administratif keimigrasian terdiri dari 3

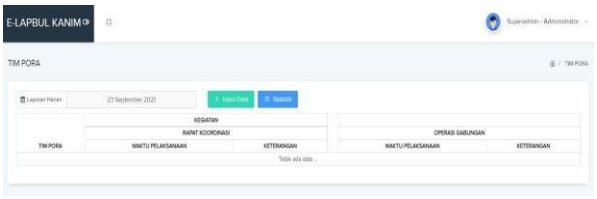
kategori yaitu pasal yang dilanggar, tindakan administratif keimigrasian, dan jumlah. Form halaman Tindakan administratif keimigrasian sebagaimana digambarkan pada gambar 39.:



Gambar 39. Halaman Tindakan Administratif Keimigrasian

14) Halaman TIM PORA

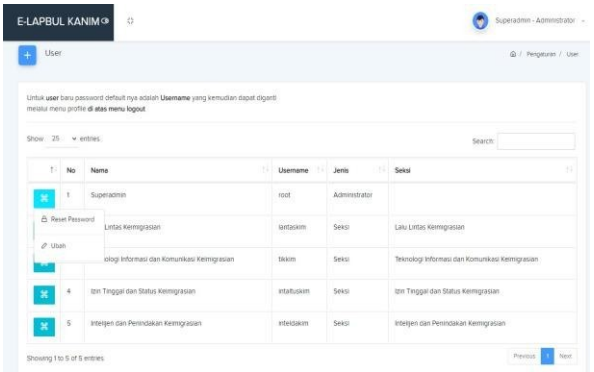
Halaman izin TIM PORA merupakan fitur bagian dari menu input data seksi intelijen dan penindakan keimigrasian. Form halaman TIM PORA terdiri dari 3 kategori yaitu TIM PORA, kegiatan; rapat koordinasi, dan operasi gabungan. Form halaman TIM PORA sebagaimana digambarkan pada gambar 40.:



Gambar 40. Halaman TIM PORA

15) Halaman Pengaturan *User*

Halaman pengaturan *user* merupakan fitur yang hanya terdapat pada akun *user* superadmin. *Form* halaman pengaturan *user* terdiri dari beberapa fitur yaitu fitur tambah *user* untuk menambahkan *user* baru sesuai kebutuhan, dan fitur pengaturan setiap *user* dimana superadmin dapat me-*reset password* ataupun mengubah data *user*. *Form* halaman pengaturan *user* sebagaimana digambarkan pada gambar 41:



Gambar 41. Halaman Pengaturan *User*

5. Pengujian

Tahap pengujian dilakukan setelah tahapan implementasi telah dipenuhi. Pada tahap ini penulis akan melakukan *testing* terhadap sistem aplikasi integrasi data antar seksi di Kantor Imigrasi Kelas I TPI Cirebon berbasis web. Tujuan dari tahapan ini yaitu untuk mengetahui apabila terdapat *bug* (cacat desain) pada sistem yang akan mengakibatkan aplikasi tidak berfungsi sebagaimana mestinya dan menyiapkan usulan perbaikan sistem.

Metode pengujian yang digunakan oleh penulis yaitu metode *black box testing*. Metode ini merupakan pengujian yang berfokus pasa fungsionalitas perangkat lunak saja, tanpa menguji desain dan kode program. *Black box testing* bertujuan untuk mengetahui kesesuaian fungsi-fungsi, masukan, dan keluaran pada perangkat lunak yang dibuat dengan spesifikasi yang dibutuhkan.

Berdasarkan hasil akhir pengujian dengan metode *black box testing* menunjukkan bahwa semua skenario pengujian fungsionalitas sistem menunjukkan validitas yang sempurna, tidak ditemukan adanya *bug* atau *error* pada sistem. Maka dari itu, untuk tujuan pemeliharaan kestabilan fungsi dari sistem aplikasi ini diperlukan prosedur penggunaan dan batasan pengguna yang jelas.

6. Pemeliharaan Sistem (*Maintenance*)

Tahap pemeliharaan merupakan suatu proses atau tahapan yang penting untuk dilakukan ketika aplikasi telah berjalan. Pemeliharaan sistem merupakan proses peninjauan secara berkala terhadap performa aplikasi yang bertujuan untuk memastikan aplikasi tersebut dapat tetap berfungsi sebagaimana mestinya, proses ini dilakukan oleh superadmin dalam kurun waktu tertentu. Pemeliharaan pada aplikasi sangat perlu dilakukan untuk mengatasi *error* yang terjadi pada aplikasi tersebut.

Proses pemeliharaan ini juga dapat memunculkan suatu inovasi pengembangan sistem yang dapat disesuaikan dengan kebutuhan-kebutuhan yang ada dikemudian hari sekaligus beradaptasi dengan perkembangan jaman.

Kesimpulan

Proses rekapitulasi data antar seksi yang diterapkan saat ini di Kantor Imigrasi Kelas I TPI Cirebon masih dilakukan secara konvensional dimana adiministrator tiap-tiap seksi harus melakukan rekapitulasi data dengan menggunakan Ms. Excel, kemudian administrator TIKIM melakukan verifikasi

terkait rekapitulasi dari tiap- tiap seksi, hal tersebut kurang efektif serta memungkinkan adanya redundansi data dan inefisiensi waktu dikarenakan banyak alur yang terjadi sehingga kemungkinan terjadi resiko kesalahan sangat besar.

Dengan memanfaatkan teknologi informasi dan komunikasi "Aplikasi Integrasi Data Antar Seksi di Kantor Imigrasi Kelas I TPI Cirebon" berbasis web telah berhasil dibuat dengan metode SDLC. Didalam aplikasi tersebut terdapat tingkatan user yang berguna untuk mengontrol setiap elemen-elemen yang sesuai pada seksinya masing-masing, kemudian konsep *one time entry data* ini sekaligus menghubungkan tiap-tiap menu yang ada di dalam sistem dan terintegrasi satu sama lain secara *real-time*, sehingga hal ini dapat mempermudah dan mempercepat proses rekapitulasi data antar seksi khususnya dalam pembuatan laporan bulanan sehingga akan menjadi lebih efektif, efisien, mengurangi redundansi data dan mengurangi ataupun meminimalisasi resiko kesalahan.

Saran

Perlu dilakukan *update* dan *maintenance* secara berkala terhadap sistem aplikasi integrasi data yang akan digunakan, sehingga performa aplikasi dapat terus dipertahankan secara optimum dan dapat menunjang serta mempermudah aktivitas pengelolaan data keimigrasian di Kantor Imigrasi Kelas I TPI Cirebon.

Pada pengembangan selanjutnya, sistem aplikasi integrasi data dapat ditambahkan fitur-fitur yang lebih lengkap, dapat disesuaikan dengan kebutuhan dan perkembangan zaman serta dapat diintegrasikan dengan aplikasi-aplikasi SIMKIM lainnya.

DAFTAR PUSTAKA

[1] S. Bandyopadhyay, "ICT in Education: Open Source Software and its Impact on Teachers and Students," *Int. J. Comput. Appl.*, vol. 151, pp. 19–24, Oct. 2016.

[2] S. Radack, "Security Considerations in the System DevelopmentLife Cycle," 2002.

[3] D. Purwitasaria, U. Yuhana, A. Rahman, B. Setiawan, and A. Affandi, "PDITS: Aplikasi Pangkalan Data Terpadu untuk Mendukung Integrasi Multi Sistem Informasi di Lingkungan Institut Teknologi Sepuluh Nopember," *Sisfo*, vol. 06, pp. 65–76, Sep. 2016.

[4] wilo rdcl, "Modul SIMKIM Bab 1-4," pp. 1–22, 2019.

[5] A. D. Giordano, *Data Integration: Blueprint and Modeling Techniques for a Scalable and Sustainable Architecture.* Boston, 2011.

[6] A. Fauzi, M. K. Dan, A. L. Hananto, and M. Kom, "INTEGRASI DATA UNTUK INFORMASI AKADEMIK MAHASISWA MENGGUNAKAN DATA INTEGRATION SYSTEM DEVELOPMENT LIFE CYCLE I . PENDAHULUAN Perguruan tinggi dalam menyelenggarakan pendidikannya wajib melakukan penjaminan mutu . Pemerintah membantu menyiapkan instrument," *J. Ilm. Solusi*, vol. 1, no. 4, pp. 1–6, 2014.

[7] R. Hidayat, "Cara Praktis Membangun Website Gratis," 2010.

[8] A. Jain, "Unified Modeling Language (UML) | An Sumirah and M. Zohri, "Integrasi Data Dalam Proses Layanan Publik," vol. 1, no. 1, 2016.

[9] A. Jain, "Unified Modeling Language (UML) | An Introduction,"2019.

[10] S. Dharwiyanti and R. S. Wahono, "Pengantar Unified Modeling LAnguage (UML)," *IlmuKomputer.com*, pp. 1–13, 2003.

[11] Henderi, "Object Oriented Modelling With Unified Modeling Language (Uml)," *5 Novemb. 2009*, no. June, p. 77, 2009.

[12] L. Chyrun *et al.*, "Web resource changes monitoring system development," *CEUR Workshop Proc.*, vol. 2386, pp. 255–273, 2019.

[13] Y. B. Leau, W. K. Loo, W. Y. Tham, and S. F. Tan, "Software development life cycle AGILE vs traditional approaches," in *International Conference on Information and Network Technology*, 2012, vol. 37, no. 1, pp. 162–167.

[14] J. de Vicente Mohino, J. Bermejo Higuera, J. R. Bermejo Higuera, and\

[15] J. A. Sicilia Montalvo, "The Application of a New Secure Software Development Life Cycle (S-SDLC) with Agile Methodologies," *Electronics* , vol. 8, no. 11. 2019.

[16] Abdul Kadir, *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: Andi, 2014.

[17] L. Welling and L. Thomson, *PHP and MySQL Web Development*.

[18] D. Purnomo, "Model Prototyping Pada Pengembangan Sistem Informasi," *J I M P - J. Inform. Merdeka Pasuruan*, vol. 2, no. 2, pp. 54–61, 2017.

[19] Lantip Diat Prasojo, *Penerapan Sistem Informasi Manajemen Pendidikan*. 2013.

[20] M. J. Abzug *et al.*, "Contributors," Elsevier, 2018, pp. vii–xix.

[21] [21] P. Assiroj, RR. Rerung, "Sistem Ujian Saringan Masuk Perguruan Tinggi Berbasis Web", 2017, Proceeding SENTIKA UAJY 201