**Laporan Kegiatan**

**KERJA PRAKTIK**

Pengembangan Website Management VPN

Di Kantor PT. Aplikanusa Lintasarta

Jalan TB Simatupang, Jakarta Selatan 12430

Periode: 1 Juli 2014 – 8 Agustus 2014



***Oleh :***

**Ahmad Hayam B (5111100091)**

**Angga Eka P (5111100117)**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**2014**

LEMBAR PENGESAHAN I

**Judul : Pengembangan WebsiteManagement VPN**

**Lokasi : Kantor PT. Aplikanusa Lintasarta**

**Jalan TB Simatupang, Jakarta Selatan 12430**

**Periode : 1 Juli 2014 – 8 Agustus 2014**

**Jakarta, 8 Agustus 2014**

**Mengetahui,**

**Nirwan Adhiatma**

**NIK.**

LEMBAR PENGESAHAN II

**Judul : Pengembangan WebsiteManagement VPN**

**Lokasi : Kantor Pusat PT. Aplikanusa Lintasarta Jakarta Jalan TB Simatupang, Jakarta Selatan, 12430**

**Periode : 1 Juli 2014 – 10 Agustus 2014**

**Dosen Pembimbing,**

**Waskito Wibisono, S.Kom., M.Eng.,PhD**

**NIP**

Daftar Isi

[LEMBAR PENGESAHAN I ii](#_Toc400536000)

[LEMBAR PENGESAHAN II iii](#_Toc400536001)

[Daftar Isi iv](#_Toc400536002)

[Daftar Tabel viii](#_Toc400536003)

[Daftar Gambar ix](#_Toc400536004)

[1. BAB I PENDAHULUAN 1](#_Toc400536005)

[I.I Latar Belakang 1](#_Toc400536006)

[I.II Rumusan Masalah 2](#_Toc400536007)

[I.III Tujuan 2](#_Toc400536008)

[I.IV Waktu Pelaksanaan 2](#_Toc400536009)

[I.V Sistematika Penulisan 2](#_Toc400536010)

[2. BAB II PROFIL PERUSAHAAN 4](#_Toc400536011)

[II.I Sejarah Perusahaan 4](#_Toc400536013)

[II.II Logo Perusahaan 5](#_Toc400536014)

[II.III Visi dan Misi Perusahaan 5](#_Toc400536015)

[3. BAB III TINJAUAN PUSTAKA 7](#_Toc400536016)

[III.I Codeigniter 7](#_Toc400536018)

[III.I.I Keunggulan Codeigniter 8](#_Toc400536019)

[III.II MySQL 8](#_Toc400536020)

[III.III Bahasa Pemrograman PHP 9](#_Toc400536021)

[4. BAB IV ANALISIS DAN PERANCANGAN 11](#_Toc400536022)

[IV.I Analisis Kebutuhan Pengembangan Perangkat Lunak 11](#_Toc400536024)

[IV.II Definisi Umum Aplikasi 12](#_Toc400536025)

[IV.III Perancangan Fungsionalitas Aplikasi 13](#_Toc400536026)

[IV.III.I Perancangan Fungsionalitas Aplikasi 13](#_Toc400536027)

[IV.III.II Cara Kerja Aplikasi 13](#_Toc400536028)

[IV.IV Perancangan Basis Data 18](#_Toc400536029)

[IV.V Rancangan Antarmuka 25](#_Toc400536030)

[IV.V.I Antarmuka Halaman Dashboard 25](#_Toc400536031)

[IV.V.II Antarmuka Halaman Tampilan Rincian Informasi Data VRF Pelanggan 26](#_Toc400536032)

[IV.V.III Antarmuka Halaman Manajemen Pengguna 27](#_Toc400536033)

[IV.V.IV Antarmuka Halaman Assign List 28](#_Toc400536034)

[IV.V.V Antarmuka Account Setting 29](#_Toc400536035)

[IV.V.VI Antarmuka Pop Up Add Assignment 30](#_Toc400536036)

[5. BAB V IMPLEMENTASI 31](#_Toc400536037)

[V.I Menampilkan daftar pelanggan 31](#_Toc400536039)

[V.I.I Potongan kode untuk role Administrator (Model) 31](#_Toc400536040)

[V.I.II Potongan kode untuk role Administrator (Controller) 32](#_Toc400536041)

[V.I.III Potongan kode untuk role CAR Admin dan CAR (Model) 32](#_Toc400536042)

[V.I.IV Potongan kode untuk role CAR Admin dan CAR (Controller) 33](#_Toc400536043)

[V.II Melakukan manajemen pengguna 34](#_Toc400536044)

[V.II.I Menampilkan daftar pengguna (Model) 34](#_Toc400536045)

[V.II.II Menampilkan daftar pengguna (Controller) 34](#_Toc400536046)

[V.II.III Menambahkan pengguna baru (Model) 35](#_Toc400536047)

[V.II.IV Menambahkan pengguna baru (Controller) 36](#_Toc400536048)

[V.II.V Menyunting data pengguna (Model) 36](#_Toc400536049)

[V.II.VI Menyunting data pengguna (Controller) 37](#_Toc400536050)

[V.II.VII Menghapus daftar pengguna (Model) 37](#_Toc400536051)

[V.II.VIII Menghapus daftar pengguna (Controller) 38](#_Toc400536052)

[V.III Menampilkan daftar hubungan penugasan antara pelanggan dan pengguna 38](#_Toc400536053)

[V.III.I Potongan kode untuk file model 39](#_Toc400536054)

[V.III.II Potongan kode untuk file controller 39](#_Toc400536055)

[V.IV Mendaftarkan pengguna ke pelanggan tertentu 40](#_Toc400536056)

[V.IV.I Potongan kode untuk file model 40](#_Toc400536057)

[V.IV.II Potongan kode untuk file controller 40](#_Toc400536058)

[V.V Menghapus pelanggan dari pengguna tertentu 41](#_Toc400536059)

[V.V.I Potongan kode untuk file model 41](#_Toc400536060)

[V.V.II Potongan kode untuk file controller 41](#_Toc400536061)

[V.VI Menampilkan rincian informasi data VRF pelanggan 42](#_Toc400536062)

[V.VI.I Potongan kode untuk file model 42](#_Toc400536063)

[V.VI.II Potongan kode untuk file controller 43](#_Toc400536064)

[V.VII Melakukan manajemen perangkat pendukung 44](#_Toc400536065)

[V.VII.I Menampilkan daftar perangkat tersedia (Model) 44](#_Toc400536066)

[V.VII.II Menampilkan daftar perangkat tersedia (Controller) 44](#_Toc400536067)

[V.VII.III Penambahan perangkat (Model) 45](#_Toc400536068)

[V.VII.IV Penambahan perangkat (Controller) 45](#_Toc400536069)

[V.VII.V Penyuntingan perangkat tersedia (Model) 46](#_Toc400536070)

[V.VII.VI Penyuntingan perangkat tersedia (Controller) 46](#_Toc400536071)

[V.VII.VII Penghapusan perangkat (Model) 47](#_Toc400536072)

[V.VII.VIII Penghapusan perangkat (Controller) 48](#_Toc400536073)

[V.VIII Menampilkan konfigurasi VRF pelanggan 48](#_Toc400536074)

[V.VIII.I Potongan kode untuk view 48](#_Toc400536075)

[V.VIII.II Potongan kode untuk model 49](#_Toc400536076)

[V.VIII.III Potongan kode untuk controller 49](#_Toc400536077)

[6. BAB VI UJI COBA DAN EVALUASI 51](#_Toc400536078)

[VI.I Lingkungan Uji Coba 51](#_Toc400536080)

[VI.II Menampilkan Hasil Uji Coba 51](#_Toc400536081)

[VI.II.I Menampilkan Daftar Pelanggan dan Status Backup VRF Pelanggan 51](#_Toc400536082)

[VI.II.II Menampilkan Rincian Informasi Data VRF Pelanggan 52](#_Toc400536083)

[VI.II.III Mendaftarkan Pelanggan ke Petugas Tertentu 53](#_Toc400536084)

[VI.II.IV Menghapus Pelanggan dari Petugas Tertentu 54](#_Toc400536085)

[VI.II.V Melakukan Proses Manajemen Pengguna 54](#_Toc400536086)

[VI.II.VI Melakukan Manajemen Perangkat 55](#_Toc400536087)

[VI.II.VII Menampilkan konfigurasi VRF pelanggan 56](#_Toc400536088)

[7. BAB VII KESIMPULAN DAN SARAN 57](#_Toc400536089)

[VII.I Kesimpulan 57](#_Toc400536091)

[VII.II Saran 58](#_Toc400536092)

[DAFTAR PUSTAKA 59](#_Toc400536093)

Daftar Tabel

[Tabel 1. Struktur Data VRF 18](#_Toc400455902)

[Tabel 2. Struktur Data MPLS VPN 19](#_Toc400455903)

[Tabel 3. Struktur Data Assign 21](#_Toc400455904)

[Tabel 4. Struktur Data *Users* 22](#_Toc400455905)

[Tabel 5. Struktur Data Backup 22](#_Toc400455906)

[Tabel 6. Struktur Data System Setting 23](#_Toc400455907)

[Tabel 7. Struktur Data Tools 24](#_Toc400455908)

Daftar Gambar

[Gambar 4.11. Spesifikasi kebutuhan website SPPBE dan OSDS dalam kasus SPA Online 13](#_Toc400455824)

[Gambar 4.12. Diagram Aktivitas untuk Kasus Mengubah Status Backup 15](#_Toc400455825)

[Gambar 4.13. Diagram Aktivitas untuk Kasus Melihat Konfigurasi 16](#_Toc400455826)

[Gambar 4.14.Diagram Aktivitas untuk Kasus Alokasi Pelanggan 17](#_Toc400455827)

[Gambar 4.15. Diagram Aktivitas untuk Kasus Menambah User 18](#_Toc400455828)

[Gambar 4.16 Rancangan antarmuka *Dashboard* 25](#_Toc400455829)

[Gambar 4.17 Rancangan Antarmuka halaman tampilan rincian informasi data VRF Pelanggan 26](#_Toc400455830)

[Gambar 6.1 Tampilan hasil uji coba menampilkan daftar VRF pelanggan dan status backup VRF pelanggan 51](#_Toc400455831)

[Gambar 6.2 Tampilan hasil ujicoba menampilkan rincian informasi data VRF Pelanggan 52](#_Toc400455832)

[Gambar 6.3 Tampilan hasil uji coba mendaftarkan pelanggan ke petugas tertentu 53](#_Toc400455833)

[Gambar 6.4 Tampilan hasil uji coba menghapus pelanggan dari petugas tertentu 54](#_Toc400455834)

[Gambar 6.5. Tampilan hasil uji melakukan proses manajemen pengguna 54](#_Toc400455835)

[Gambar 6.6 Tampilan hasil uji coba melakukan manajemen perangkat 55](#_Toc400455836)

[Gambar 6.7 Hasil tampilan uji coba menampilkan konfigurasi VRF pelanggan 56](#_Toc400455837)

# BAB I PENDAHULUAN

## Latar Belakang

PT. Aplikanusa lintasarta adalah Perusahaan yang bergerak di bidang layanan komunikasi data,internet,dan berbagai layangan jaringan seperti cloud service dan data center. Salah satu layanan dalam bidang komunikasi data yang dikembangkan adalah layanan IP Virtual Private Network(IP VPN). Salah satu aplikasi *downstream* yang dikembangkan saat ini adalah *website* menejemen VPN.

Manajemen VPN adalah sistem untuk menyimpan data VPN dan melakukan monitoring VPN yang nantinya dapat digunakan oleh divisi Operation Support PT.Aplikanusa Lintas Arta untuk mempermudah proses penanganan jaringan.

VPN harus dikelola sedemikian rupa sebagaimana jaringan yang lain terkait dengan keamanannya,koneksi internet VPN,pengelolaan alamat VPN,pemetaan VPN dan penyimpanan data oleh server VPN.

Fokus utama dari kerja praktik ini adalah perancangan website untuk mempermudah proses manajemen VPN.

## Rumusan Masalah

* Bagaimana cara memudahkan proses backup konfigurasi dengan data jaringan yang banyak?
* Bagaimana cara memudahkan proses pembagian tugas monitoring jaringan pelanggan?
* Bagaimana cara mempermudah monitoring jaringan pelanggan
* Bagaimana cara mempermudah trouble shooting jaringan pada device yg berbeda?

## Tujuan

* Memudahkan administrator untuk melakukan pengelolaan petugas VPN dan pelanggan VPN
* Memudahkan administrator untuk melakukan *backup* konfigurasi
* Mempermudah CAR admin melakukan pembagian tugas dalam penanganan pelanggan
* Mempermudah CAR untuk melihat status backup dari pelanggan yang ditangani
* Mempermudah sistem untuk melakukan penanganan masalah pada jaringan pada perangkat berbeda

## Waktu Pelaksanaan

Kerja Praktik yang dilaksanakan PT. Jakarta ini dilaksanakan selama satu bulan, yang terhitung sejak tanggal 1 Juli 2014 sampai dengan tanggal 10 Agustus 2014.

## Sistematika Penulisan

Laporan Kerja Praktik ini dibagi menjadi 7 bab dengan rincian sebagai berikut :

* **Bab I : Pendahuluan**

Pada bab ini, dijelaskan latar belakang, permasalahan yang dihadapi selama pelaksanaan kerja praktik, tujuan dari pengerjaan dan pengembangan aplikasi, waktu pelaksanaan serta sistematika penulisan laporan kerja praktik.

* **Bab II : Sejarah dan Profil Perusahaan**

Pada bab ini, akan dijelaskan secara rinci mengenai sejarah dan profil PT. Aplikanusa Lintasrta

* **Bab III : Tinjauan Pustaka**

Pada bab ini, dijelaskan tentang tinjauan pustaka yang digunakan untuk menyelesaikan aplikasi yang dibuat.

* **Bab IV : Analisa dan Perancangan**

Pada bab ini, dijelaskan mengenai analisis dan perancangan pengembangan yang dilakukan dalam aplikasi Mangement VPN.

* **Bab V : Implementasi**

Pada bab ini berisi uraian mengenai implementasi pada proses pengerjaan dan pengembangan aplikasi *website* Management VPN.

* **Bab VI : Uji Coba dan Evaluasi**

Pada bab ini, dijelaskan mengenai hasil uji coba dan evaluasi dari perangkat lunak yang telah dikembangan selama pelaksanaan kerja praktik.

* **Bab VII : Kesimpulan dan Saran**

Pada bab ini, dijelaskan mengenai kesimpulan dan saran yang didapat selama proses pelaksanaan kerja praktik.

# BAB II PROFIL PERUSAHAAN



## Sejarah Perusahaan

PT. APLIKANUSA LINTASARTA berdiri 4 April tahun 1988. pada tahun 1989 perusahaan ini mulai beroperasi dengan Implementasi Sistem Laporan Bulanan Bank (LBB),  Implementasi Sistem Informasi Kredit (SIK)Implementasi Sistem Autorisasi Kartu Visa dan Master.

Setelah dua tahun berdirinya, APLIKANUSA LINTASARTA ikut berpartisipasi dalam Jaringan ATM gabungan yang pertama yaitu ATM-Bersama dan Implementasi Layanan BPP Telekomunikasi. Di tahun ini juga terjadi persetuajuan Perjanjian Kerjasama dengan PT. Telkom untuk layanan komunikasi data serta Peluncuran Sambungan Data Langsung (SDL), layanan Digital leased line dengan menggunakan teknologi *Data Over Voice* (DOV).

Pada tahun 1992 Perusahaan APLIKANUSA LINTASARTA meluncurkan layanan *Very Small Apperture Terminal* (VSAT). Serta memulai Pengoperasian Sistem Informasi Elektronik (SIE). Setelah delapan tahun berdirinya, APLIKANUSA LINTASARTA memperoleh Lisensi operasional No. KM.24/PB.103.MPPT-95 untuk Layanan Dasar Telekomunikasi. dan No. KM.25/PB.103/MPPT-95 untuk Layanan non-basic Telekomunikasi dari Kementrian Telekomunikasi. Di tahun ini juga diluncurkannya  layanan *Internet Indonesia On-Line Access* ( IdOLA ) serta JITU (Jaringan Informasi & Transaksi Keuangan ) yang merupakan layanan untuk Bank dan Asuransi oleh APLIKANUSA LINTASARTA.

Memasuki Era Millenium, Perusahaan APLIKANUSA LINTASARTA mendapatkan pencapaian sertifikasi ISO 9002 untuk layanan SDL, VSAT, dan Frame Relay dari SGS Yarsley International. Ditahun yang sama juga terjadi peluncuran APLIKANUSA LINTASARTA *E-Commerce Payment Gateway* dengan nama  “www.bayaronline.com”.

## Logo Perusahaan



## Visi dan Misi Perusahaan

**Visi :** Untuk menjadi pemimpin dalam solusi informasi dan komunikasi bisnis di Indonesia.

**Misi :** Untuk membuat bisnis lebih mudah bagi perusahaan dan individu melalui solusi informasi dan komunikasi yang inovatif.

***[Halaman ini sengaja dikosongkan]***

# BAB III TINJAUAN PUSTAKA

Pada bab ini, akan dijelaskan mengenai dasar teori yang digunakan selama proses pengerjaan dan pengembangan aplikasi *Website* Management VPN. Dasar teori yang akan dibahas adalah mengenai Code Igniter,PHP dan MySQL.



## Codeigniter

CodeIgniter merupakan salah satu dari sekian banyak framework PHP yang ada. Tujuan dari pembuatan framework CodeIgniter ini menurut user manualnya adalah untuk menghasilkan framework yang akan dapat digunakan untuk pengembangan proyek pembuatan website secara lebih cepat dibandingkan dengan pembuatan website dengan cara koding secara manual, dengan menyediakan banyak sekali pustaka yang dibutuhkan dalam pembuatan website, dengan antarmuka yang sederhana dan struktur logika untuk mengakses pustaka yang dibutuhkan. CodeIgniter membiarkan kita untuk memfokuskan diri pada pembuatan website dengan meminimalkan pembuatan kode untuk berbagai tujuan pembuatan website.

CodeIgniter menerapkan lingkungan pengembangan dengan metode MVC (Model View Controller). MVC memisahkan antara logika pembuatan kode dengan pembuatan template atau tampilan website. Penggunaan MVC membuat pembuatan sebuah proyek website menjadi lebih terstruktur dan lebih sederhana. Secara sederhana konsep MVC terdiri dari tiga bagian yaitu bagian Model, bagian View dan bagian Controller. Didalam website dinamis setidaknya terdiri dari 3 hal yang paling pokok, yaitu basis data, logika aplikasi dan cara menampilkan halaman wesite. 3 hal tersebut direpresentasikan dengan MVC yaitu model untuk basis data, view untuk cara menampilkan halaman website dan controller untuk logika aplikasi.

### Keunggulan Codeigniter

Codeigniter memiliki beberapa keunggulan sebagai berikut :

* **Open Source**

CodeIgniter dilisensikan dibawah lisensi Apache/BSD style open source license.

* **Ringan dan cepat**

Secara umum Codeginiter hanya berjalan dengan memuat beberapa pustaka. Dengan demikian membutuhkan sumber daya yang sedikit sehingga ringan dan cepat ketika dijalankan. Pustaka-pustaka lain yang nantinya akan digunakan bisa dimuat sesuai kebutuhan.

* **Menggunakan konsep MVC**

CodeIgniter menerapkan lingkungan pengembangan dengan metode MVC (Model View Controller). MVC memisahkan antara logika pembuatan kode dengan pembuatan template atau tampilan website. Penggunaan MVC membuat pembuatan sebuah proyek website menjadi lebih terstruktur dan lebih sederhana.

## MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Sebagai perangkat lunak pendukung basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya.

Beberapa kelebihan yang dimiliki oleh MySQL diantaranya :

* **Portabilitas**.

MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

* **Perangkat lunak sumber terbuka (open source)**. MySQL didistribusikan sebagai open source sehingga dapat digunakan secara gratis.
* **Multi-user**

MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

* **Performance tuning**

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

* **Ragam tipe data**

MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

* **Perintah dan Fungsi**

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).

* **Keamanan**

MySQL memiliki beberapa lapisan keamanan seperti password yang terenkripsi.

## Bahasa Pemrograman PHP

PHP adalah singkatan dari PHP Hypertext Preprocessor yang digunakan sebagai bahasa script server-side dalam pengembangan Web yang disisipkan pada dokumen HTML. PHP bisa berinteraksi dengan database, file dan folder, sehingga membuat PHP bisa menampilkan konten yang dinamis dari sebuah website.

PHP dapat digunakan pada semua sistem operasi, antara lain Linux, Unix (termasuk variannya HP-UX, Solaris, dan Open BSD), Microsoft Windows, Mac OS, RISC OS. PHP juga mendukung banyak Web Server, seperti Apache, Microsoft Internet Information Server (MIIS), Personal Web Server (PWS), Netcape and iPlanet servers, Oreilly Website Pro Server, audium, Xitami, OmniHTTPd, dan masih banyak lagi lainnya, bahkan PHP dapat bekerja sebagai suatu CGI Processor.

# BAB IV ANALISIS DAN PERANCANGAN

Pada bab ini, akan dijelaskan mengenai analisis kebutuhan dari aplikasi yang dibangun, deskripsi umum dari aplikasi yang dikembangkan serta cara kerja aplikasi.



## Analisis Kebutuhan Pengembangan Perangkat Lunak

Secara umum, spesifikasi kebutuhan pengembangan perangkat lunak yang akan dikembangkan dalam aplikasi ini adalah sebagai berikut:

1. **Menampilkan daftar pelanggan dan status backup VRF pelanggan**

Menampilkan daftar pelanggan dan status backup Virtual Routing Forwarding(VRF) pelanggan. Petugas dapat melakukan pencarian pelanggan dan melakukan backup konfigurasi pelanggan.

1. **Melakukan manajemen pengguna**

Manajemen pengguna dilakukan oleh seorang administrator. Terdapat kolom pencarian untuk mempermudah pencarian pengguna,tombol *adduser* untuk penambahan pengguna dan hak aksesnya,serta tombol edit

untuk penyuntingan dan delete untuk menghapus pengguna.

1. **Mendaftarkan pelanggan ke petugas tertentu**

Pelanggan yang baru ditambahkan ke dalam database atau yang belum terkelola akan ditugaskan ke pelanggan tertentu. Terdapat tombol add assignment untuk menugaskan pelanggan ke petugas tertentu.

1. **Menghapus pelanggan dari petugas tertentu**

Pelanggan yang sudah pernah terassign dapat dihapus dari petugas tertentu untuk efisiensi database atau pengalihan penugasan.

1. **Menampilkan rincian informasi data VRF pelanggan**

Sistem menampilkan rincian informasi data VRF pelanggan, status backup, status aktif

1. **Melakukan manajemen perangkat pendukung**

Sistem juga menyimpan perangkat-perangkat jaringan yang mendukung pengelolaan pelanggan.

1. **Menampilkan konfigurasi VRF pelanggan**

Sistem yang dirancang juga memiliki fitur untuk menampilkan konfigurasi untuk melihat pengaturan konfigurasi VRF pelanggan.

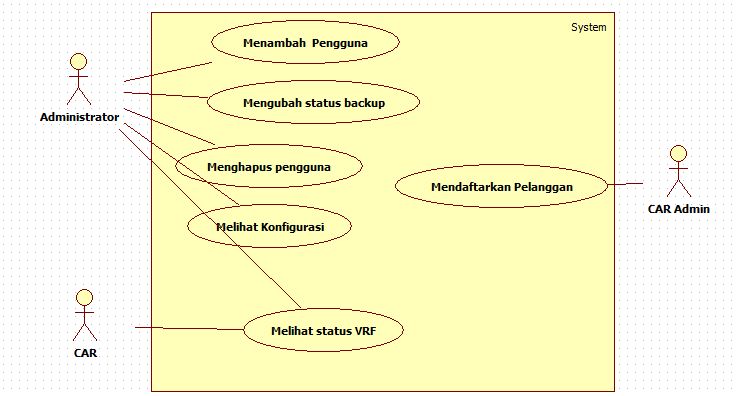
## Definisi Umum Aplikasi

Secara umum, aplikasi ini ditujukan untuk administrator,CAR admin,dan CAR yang ada di divisi Operation Support PT. Aplikanusa Lintasarta yang bertugas untuk mengawasi keadaan VRF pelanggan. Tujuan pembuatan aplikasi ini adalah untuk memudahkan pengguna dalam melakukan proses pengelolaan VPN yang mencakup,pengelolaan pengguna, penugasan pelanggan, menampilkan backup konfigurasi,pengelolaan jenis perangkat jaringan,serta pembagian tugas penanganan VRF pelanggan.

## Perancangan Fungsionalitas Aplikasi

### Perancangan Fungsionalitas Aplikasi

Berdasarkan uraian di atas, gambaran spesifikasi kebutuhan di atas dapat digambarkan dengan diagram kasus penggunaan berikut.



Gambar .. Spesifikasi kebutuhan website VPN Management

### Cara Kerja Aplikasi

Aplikasi Manajemen VPN ini berperan penting dalam mendukung proses pengelolaan jaringan khususnya dengan aktivitas-aktivitas yang terkait dengan manajemen pelanggan: penambahan, penugasan atau pengalokasian pelanggan, penghapusan, pengubahan status backup, dan menampilkan rincian data pelanggan.

1. **Login pengguna**

Pengguna melakukan *login* ke dalam website terlebih dahulu untuk melakukan perannya masing-masing.

[Spesifikasi Pengembangan yang terkait: tidak ada]

1. **Menampilkan daftar pelanggan**

CAR dan Administrator melakukan pencarian pelanggan untuk merubah status backup,menampilkan rincian data pelanggan,atau melihat konfigurasi VRF pelanggan.

[Spesifikasi Pengembangan yang terkait: a]

1. **Manajemen pengguna**

Administrator dapat melakukan penambahan pengguna dengan peran tertentu,sebagai Administrator,CAR atau sebagai CAR Admin. Administrator juga dapat melakukan penyuntingan atau penghapusan pengguna.

[Spesifikasi Pengembangan yang terkait: b]

1. **Manajamen pelanggan**

CAR admin dapat mengalokasikan pelanggan tertentu untuk petugas tertentu. CAR admin juga dapat menghapus daftar pelanggan yang sudah pernah dialokasikan ke petugas tertentu.

[Spesifikasi Pengembangan yang terkait: tidak ada]

1. **Melakukan back up status VRF pelanggan**

Administrator dapat melakukan penyimpanan konfigurasi yang terdapat pada VRF pelanggan. Sistem akan menampilkan notifikasi ketika back up status dirubah.

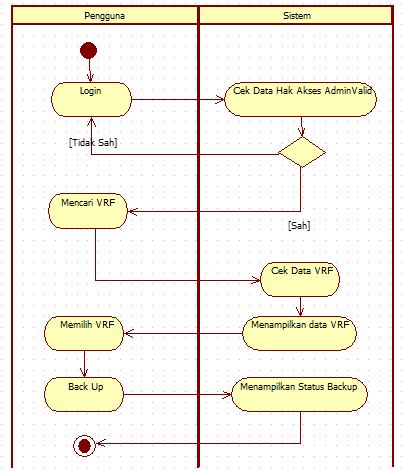
[Spesifikasi Pengembangan yang terkait: d]

1. **Manajemen Perangkat**

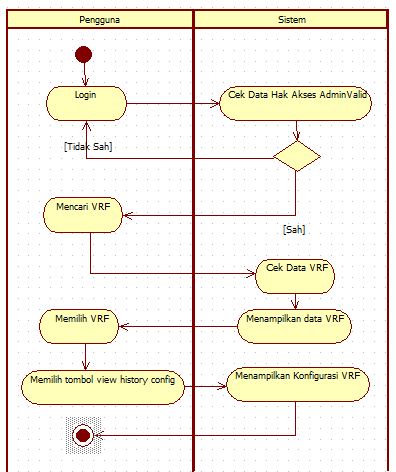
Administrator dapat melakukan manajemen perangkat aplikasi web yang tersedia secara local. Untuk melakukan manajemen perangkat, pengguna dengan level Administrator diharuskan mengunjungi halaman Tools Management dengan cara mengklik Tools Management pada sidebar sebelah kiri. Pengguna dapat melakukan penambahan, penyuntingan, dan penghapusan tool yang dimaksud.

1. **Melihat Konfigurasi Jaringan Pelanggan**

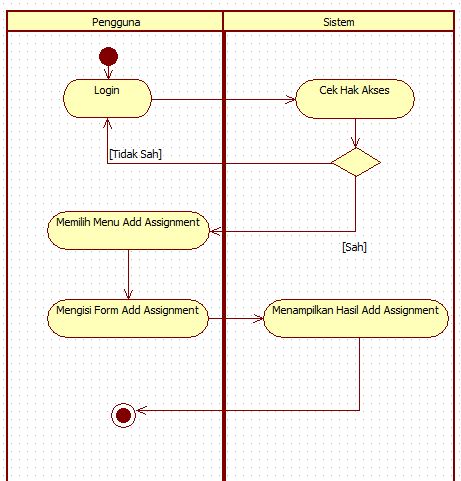
Pengguna dengan hak akses Administartor dapat melihat konfigurasi terakhir dan histori konfigurasi node-node pelanggan yang menginginkan konfigurasi jaringannya terbackup. Untuk melakukan ini, pengguna masuk ke halaman detil pelanggan. Di halaman tersebut ditampilkan daftar node yang berkaitan dengan pelanggan tertentu.



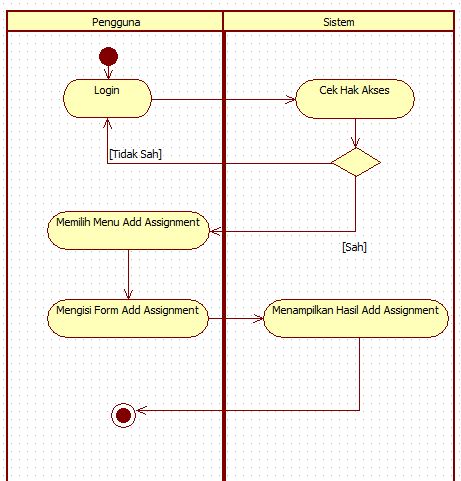
Gambar .. Diagram Aktivitas untuk Kasus Mengubah Status Backup



Gambar .. Diagram Aktivitas untuk Kasus Melihat Konfigurasi



Gambar ..Diagram Aktivitas untuk Kasus Alokasi Pelanggan



Gambar .. Diagram Aktivitas untuk Kasus Menambah User

## Perancangan Basis Data

Basis data yang digunakan oleh aplikasi website VPN Management Untuk Kontrol Data Pelanggan ini adalah MySQL. Dalam pengembangan di proyek ini, aplikasi ini melibatkan 7 tabel yang akan sebagai penyimpanan data.

1. Pembuatan tabel data VRF

Tabel ini digunakan untuk menyimpan nama-nama VRF planggan beserta status backupnya. Tabel ini tidak berelasi dengan tabel mana pun.

Tabel . Struktur Data VRF

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| id\_vpn | Varchar | Nomor Id VPN |
| nama\_vrf | Varchar | Nama VRF pelanggan |
| is\_backup | Integer | Status backup VRF pelanggan |

1. Pembuatan tabel mpls vpn

Pembuatan table ini dimaksudkan untuk menyimpan informasi detil dari masing-masing pelanggan. Informasi node dari jaringan pelanggan yang disimpan dalam table ini banyak dimanfaatkan oleh *background system*. Beberapa kegiatan dari background system yang memanfaatkan tabel ini antara lain adalah pengecekan status interface (hidup atau mati), status ping (sukses atau gagal), penanda backup untuk pelanggan yang membolehkan fitur backup konfigurasi jaringan.

Tabel . Struktur Data MPLS VPN

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| Id-Jaringan | Integer | Representasi identitas node yang terdaftar |
| node | Varchar(50) | Nama node parent untuk masing-masing jaringan |
| Vrf\_name | Varchar(50) | Nama grup jaringan yang berasosiasi dengan pelanggan |
| interface | Varchar(50) | Jenis Hardware yang dipakai |
| Id\_device\_backup | Varchar(50) | Representasi merk hardware yang digunakan pada jaringan |
| description | Varchar(50) | Deskriptsi jalur jaringan |
| Ip\_addr | Varchar(50) | Alamat ip dari jaringan |
| Ip\_mask | Varchar(50) | Net ID dari jaringan |
| nojar | Varchar(50) |  |
| customer | Varchar(50) | Nama pelanggan yang terdaftar pada system Lintasarta |
| status | Integer | Representasi biner dari status interface, hidup (1) atau mati (0) |
| Ping\_status | Integer | Representasi biner dari status ping dari jaringan, sukses (1) atau tidak (0) |
| Isexist | Integer | Representasi biner dari keaktifan jaringan |
| Iscleared | Integer | Representasi biner dari status perubahan jaringan |
| Backup\_status | Integer | Representasi biner dari mekanisme pembackupan konfigurasi jaringan, backup (1), tidak dibackup (0) |
| Backup\_info | Varchar(50) | Informasi tambahan mengenai info backup konfigurasi jaringan |
| Backup\_date | Datetime | Tanggal backup konfigurasi jaringan terakhir |
| Change\_datetime | Timestamp | Tanggal perubahan konfigurasi jaringan |
| Cleared\_date | Datetime | Tanggal pembersihan konfigurasi jaringan |
| User\_id\_clear | Integer | Representasi identitas pengguna yang melakukan pengubahan konfigurasi |

1. Pembuatan tabel alokasi pelanggan (assign)

Pembuatan tabel ini dilakukan untuk menyimpan data VRF pelanggan yang telah dialokasikan ke petugas tertentu.

Tabel . Struktur Data Assign

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| NoPol\_Kendaraan | varchar | Nomor polisi dari kendaraan angkutan |
| StatusKepemilikan\_Kendaraan | varchar | Status kepemilikan dari kendaraan angkutan |

1. Pembuatan tabel data pengguna

Tabel ini digunakan sebagai penyimpan data pengguna dan status reset password.

Tabel . Struktur Data *Users*

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| userid | integer | Id pengguna |
| inisial | varchar | nama inisial pengguna |
| is\_forgot | Small integer | status untuk kasus pengguna lupa password |
| nama | varchar | nama pengguna |
| password | varchar | password |
| is\_admin | integer | Tipe hak akses pengguna |
| email | varchar | Alamat email pengguna |

1. Pembuatan tabel perangkat *back up*

Tabel *back up* adalah tabel yang menyimpan jenis-jenis brand perangkat yang digunakan di jaringan pelanggan.

Tabel . Struktur Data Backup

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| id\_device\_backup | integer |  |
| Id\_perangkat | varchar | Id perangkat |
| Nama\_perangkat | varchar | Kolom untuk nama perangkat yang terkoneksi dengan server |

6. Pembuatan table system setting

Pembuatan tabel ini adalah untuk menyimpan konfigurasi penyimpanan file konfigurasi jaringan. Hal ini dimaksudkan jika file disimpan pada server yang berbeda dengan webserver aplikasi. Sehingga aplikasi web ini bisa dipindahkan dan tidak mengganggu pengaksesan file konfigurasi.

Tabel . Struktur Data System Setting

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| Server\_addr | Varchar(50) | Alamat server yang menyimpan file konfigurasi jaringan |
| Dir\_config | Varchar(50) | Alamat folder yang menyimpan |
| History\_dir\_config | Varchar(50) |  |
| Last\_dir\_config | Varchar(50) |  |

7. Pembuatan tabel tools

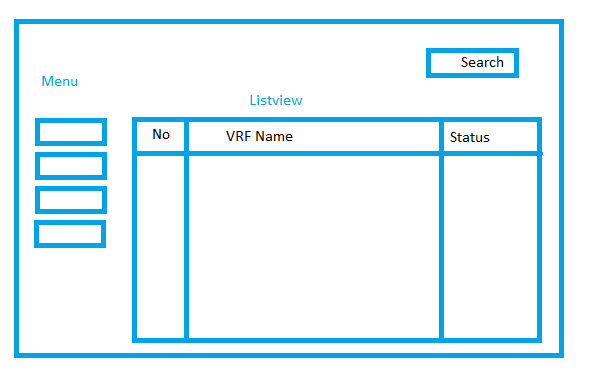
Pembuatan tabel ini dimaksudkan untuk menyimpan detil informasi tentang aplikasi web yang tersedia pada jaringan lokal PT. Lintasarta. Aplikasi web yang terdapat pada PT. Lintasarta tersebar di berbagai server, sehingga perlu ada pencatatan untuk mempermudah pengaksesan aplikasi web yang lain.

Tabel . Struktur Data Tools

|  |  |  |
| --- | --- | --- |
| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| Id\_tool | Int | Representasi identitas untuk tool-tool yang didaftarkan pada jaringan lokal |
| Nama\_tool | Varchar(50) | Nama-nama tool yang terdaftar pada jaringan local |
| URL | Varchar(50) | Alamat aplikasi web pada jaringan local |
| LOGO\_TOOL | Varchar(50) | Alamat untuk menyimpan logo tool dari aplikasi web |
| Waktu\_buat | Timestamp | Waktu pembuatan tool |

## Rancangan Antarmuka

### Antarmuka Halaman Dashboard

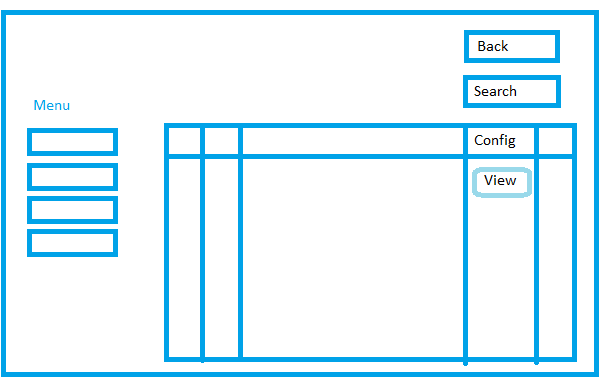


Gambar .1 Rancangan antarmuka *Dashboard*

Untuk halaman *Dashboard* ini terdiri dari beberapa kontrol. Kontrol yang digunakan antara lain sebagai berikut :

* Untuk menu digunakan kontrol *label*
* Untuk *Search* menggunakan kontrol *Textbox*
* Untuk menampilkan hasil pencarian menggunakan kontrol *Listview*.

### Antarmuka Halaman Tampilan Rincian Informasi Data VRF Pelanggan

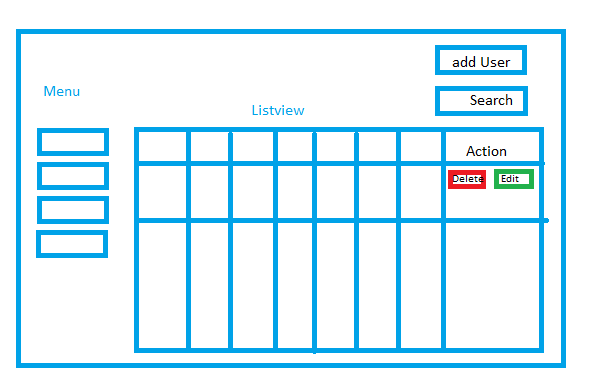


Gambar .7 Rancangan Antarmuka halaman tampilan rincian informasi data VRF Pelanggan

Untuk rancangan antarmuka halaman tampilan rincian informasi data VRF pelanggan ini terdiri dari beberapa kontrol. Kontrol yang digunakan antara lain sebagai berikut :

* Untuk *Back* digunakan kontrol *Button*
* Untuk *Search* digunakan kontrol *Textboxt*
* Untuk View digunakan kontrol *Button*

### Antarmuka Halaman Manajemen Pengguna

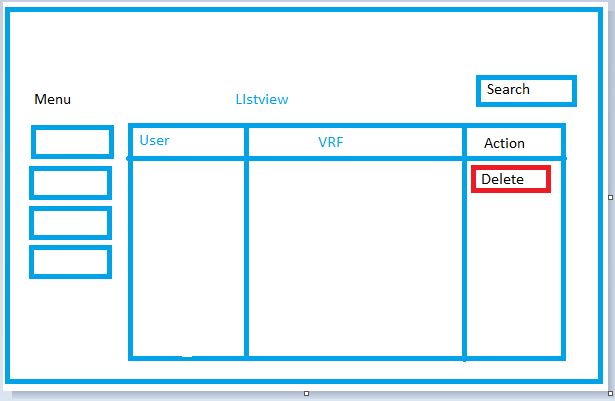


Gambar .18 Rancangan antarmuka manajemen pengguna

Untuk rancangan antarmuka halaman manajemen pengguna kontrol yang digunakan diantaranya sebagai berikut :

* Untuk *add User* digunakan kontrol *Button.*
* Untuk Search digunakan kontrol *Textbox*.
* Untuk Delete digunakan kontrol *Button*.
* Untuk Edit digunakan kontrol *Button*.

### Antarmuka Halaman Assign List

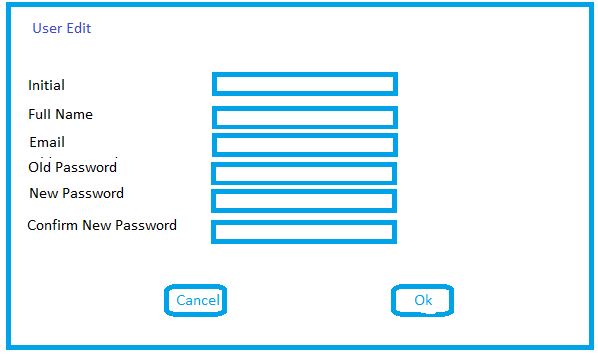


Gambar .19 Rancangan antarmuka halaman assign list

Beberapa kontrol yang digunakan pada antarmuka halaman *assign* *list* adalah sebagai berikut :

* Untuk *Search* digunakan kontrol *Textbox.*
* Untuk *Delete* digunakan kontrol *Button.*

### Antarmuka Account Setting

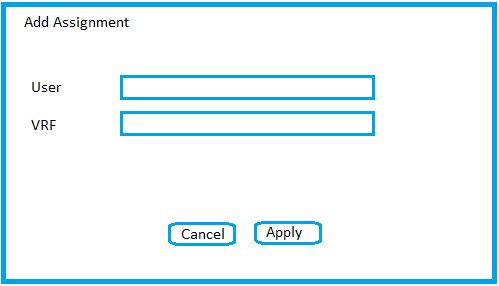


Gambar .20 Rancangan antarmuka account setting

Ketika pengguna memilih menu account setting akan muncul *pop up* untuk isian data pengguna. Kontrol yang digunakan diantaranya sebagai berikut :

* Untuk Initial,Full Name,Email,Old Password,New Password,Confirm New Password digunakan kontrol *Label.*
* Untuk isian tiap *Label* digunakan kontrol*Textbox*.
* Untuk *Cancel* dan *Ok* digunakan kontrol *Button*.

### Antarmuka Pop Up Add Assignment



Gambar .21 Rancangan antarmuka pop up add assignment

Untuk Pop Up Menu Add Assignment kontrol yang digunakan adalah sebagai berikut :

* Untuk *User* dan *VRF* digunakan kontrol *Label*.
* Untuk isian tiap *Label* digunakan kontrol *Textbox*.
* Untuk *Cancel* dan *Apply* digunakan kontrol *Button.*

# BAB V IMPLEMENTASI

Pada bab ini akan dijelaskan mengenai implementasi berupa kode program yang digunakan untuk membangun aplikasi web VPN Management. Dalam aplikasi ini digunakan kakas kerja Codeigniter dengan arsitektur MVC (Model View Controller), sehingga kode program yang ditampilkan disini telah terbagi atas model dan view dalam bahasa PHP. Dalam bagian kode program akan diberikan keterangan umum mengenai maksud dari kode tersebut.



## Menampilkan daftar pelanggan

Dalam sistem informasi ini, pengguna dapat melihat daftar pelanggan. Terdapat perbedaan dalam use case ini untuk masing-masing role. Untuk level pengguna Administrator dapat melihat semua pelanggan PT. APLIKANUSA LINTASARTA, sedangkan untuk level CAR Admin dan CAR hanya dapat melihat daftar pelanggan yang ada pada daftar layanan pengguna yang bersangkutan. Perbedaan lain yang terdapat pada use case ini adalah menu yang berafiliasi dengan fitur ini. Untuk jenis pengguna Administrator dan CAR dapat mengakses menu dashboard, sedangkan untuk CAR Admin dapat mengakses menu Assign List. Tampilan daftar pelanggan berupa tabel dengan 3 kolom, yaitu *No*, *VRF Name*, dan *Backup Status*. Pada implementasi kode program, digunakan templating yang terdiri dari halaman header, footer, sidebar (menu), dan halaman yang sedang diakses. Semua halaman yang dimaksud dipanggil melalui Controller yang berafiliasi dengan fungsi yang sedang dipanggil. Berikut ini adalah potongan kode untuk menampilkan daftar pelanggan yang digunakan pada sistem informasi ini.

### Potongan kode untuk role Administrator (Model)

public function get\_list()

{

$query = $this->db->query('SELECT \* FROM vrf\_data');

return $query->result();

}

### Potongan kode untuk role Administrator (Controller)

﻿ ﻿private function vrf\_data()

{

$this->load->model('vrf\_data\_model');

$data['vrf\_data'] = $this->vrf\_data\_model->get\_list();

$datas['menus'] = $this->session->userdata('menus');

$datas['inisial'] = $this->session->userdata(' inisial');

$datas['is\_admin'] = $this->session->userdata(' is\_admin');

$datas['email'] = $this->session->userdata(' email');

$datas['userid'] = $this->session->userdata(' userid');

$datas['nama'] = $this->session->userdata('nama');

$datas['page\_name'] = 'Dashboard';

$data['page\_name'] = 'Dashboard';

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('vrf\_data\_view', $data);

$this->load->view('partial/footer');

}

### Potongan kode untuk role CAR Admin dan CAR (Model)

public function get\_assigned\_list($userid)

{

$query = $this->db->query('SELECT vrf\_data.id\_vpn, vrf\_data.nama\_vrf, vrf\_data.isbackuped FROM vrf\_data, assign WHERE vrf\_data.id\_vpn = assign.id\_vpn AND assign.userid = ' . $userid);

return $query->result();

}

### Potongan kode untuk role CAR Admin dan CAR (Controller)

private function assigned($userid)

{

$this->load->model('assign\_model');

$data['assigned'] = $this->assign\_model->get\_assigned\_list($userid);

$datas['menus'] = $this->session->userdata('menus');

$datas['inisial'] = $this->session->userdata('inisial');

$datas['is\_admin'] = $this->session->userdata('is\_admin');

$datas['email'] = $this->session->userdata('email');

$datas['userid'] = $this->session->userdata('userid');

$datas['nama'] = $this->session->userdata('nama');

if ($this->session->userdata('is\_admin') == 2)

{

$datas['page\_name'] = 'Dashboard';

$data['page\_name'] = 'Dashboard';

}

else

{

$datas['page\_name'] = 'Assign List';

$data['page\_name'] = 'Assign List';

}

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('assigned\_view', $data);

$this->load->view('partial/footer');

}

## Melakukan manajemen pengguna

Dalam fitur ini, manajemen pengguna hanya dapat dilakukan oleh role Administrator dan CAR Admin. Manajemen pengguna meliputi pembuatan pengguna baru, penyuntingan pengguna, dan penghapusan pengguna. Administrator maupun CAR Admin dapat mengakses halaman ini melalui menu ‘User Management’ yang terdapat pada menu samping kiri halaman.

### Menampilkan daftar pengguna (Model)

Daftar pengguna yang ditampilkan bagi pengguna dengan role Administrator dan CAR Admin memiliki perbedaan. Untuk role CAR Admin, daftar yang ditampilkan hanya pengguna yang memiliki role CAR Admin dan CAR. Sedangkan untuk role Administrator semuanya. Daftar pengguna ditampilkan dalam bentuk tabel dengan kolom No, Name, Initial, Role, Email, dan Action. Untuk kolom Action berisi dua tombol, yaitu tombol Edit dan Delete yang menangani fungsi penyuntingan dan penghapusan pengguna secara berurutan.

public function get\_list($roleid)

{

$userid = $this->session->userdata('userid');

if ($roleid == 1) $query = $this->db->query('SELECT \* FROM user\_table WHERE userid != ' . $userid);

else if ($roleid == 3) $query = $this->db->query('SELECT \* FROM user\_table WHERE is\_admin != 1 AND userid != ' . $userid);

return $query->result();

}

### Menampilkan daftar pengguna (Controller)

public function user()

{

$roleid = $this->session->userdata('is\_admin');

if ($roleid == 1 || $roleid == 3)

{

$this->load->model('user\_table\_model');

$data['user\_list'] = $this->user\_table\_model->get\_list($roleid);

$datas['menus'] = $this->session->userdata(' menus');

$datas['inisial'] = $this->session->userdata(' inisial');

$datas['is\_admin'] = $this->session->userdata(' is\_admin');

$datas['email'] = $this->session->userdata(' email');

$datas['userid'] = $this->session->userdata(' userid');

$datas['nama'] = $this->session->userdata(' nama');

$datas['page\_name'] = 'User Management';

$data['page\_name'] = 'User Management';

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('user\_list\_view', $data);

$this->load->view('partial/footer');

}

else

{

redirect('/dashboard', 'refresh');

}

}

### Menambahkan pengguna baru (Model)

Terdapat perbedaan untuk fitur penambahan pengguna baru, yaitu pengguna baru dengan role Administrator hanya dapat ditambahkan oleh pengguna dengan role Administrator. Untuk penambahan pengguna baru dengan role CAR dan CAR Admin dapat dilakukan oleh Administrator dan CAR Admin.

public function add\_user($inisial, $nama, $email, $roleid, $password)

{

$password = $password;

$query = $this->db->query("INSERT INTO user\_table (inisial, nama, email, is\_admin, password) VALUES ('$inisial', '$nama', '$email', '$roleid', '$password')");

}

### Menambahkan pengguna baru (Controller)

public function add()

{

$this->load->model('user\_table\_model');

if ($\_POST['inisial'] == '' || $\_POST['fullname'] == '' || $\_POST['email'] == '' || $\_POST['roleuser'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'User adding failed...');

redirect('/dashboard/user', 'refresh');

}

### Menyunting data pengguna (Model)

Dalam penyuntingan pengguna terdapat beberapa 3 field berupa textbox (Initial, Full Name, Email Address), 1 tombol (reset password), dan 1 dropdown (role penyuntingan).

public function edit\_user($userid, $password, $email, $nama, $inisial, $is\_admin)

{

if ($password == '')

{

$query = $this->db->query("UPDATE user\_table SET email = '$email', nama = '$nama', inisial = '$inisial', is\_admin = '$is\_admin' where userid = " . $userid);

}

else

{

$password = $password;

$query = $this->db->query("UPDATE user\_table SET password = '$password', is\_forgot = 1, email = '$email', nama = '$nama', inisial = '$inisial', is\_admin = '$is\_admin' where userid = " . $userid);

}

}

### Menyunting data pengguna (Controller)

public function edit()

{

$this->load->model('user\_table\_model');

if ($\_POST['userid\_hidden'] == '' || $\_POST['email'] == '' || $\_POST['fullname'] == '' || $\_POST['inisial'] == '' || $\_POST['roleuser'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'User editing failed');

redirect('/dashboard/user', 'refresh');

}

else

{

$this->load->model('tools\_model');

if ($\_POST['password\_hidden'] == '!@#$%^&\*()')

{

$\_POST['password\_hidden'] = $this->tools\_model->getRandomString();

}

$this->user\_table\_model->edit\_user($\_POST['userid\_hidden'], $\_POST['password\_hidden'], $\_POST['email'], $\_POST['fullname'], $\_POST['inisial'], $\_POST['roleuser']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'User editing applied successfully..');

redirect('/dashboard/user', 'refresh');

}

}

### Menghapus daftar pengguna (Model)

Untuk menghapus data pengguna, pengguna dengan role Administrator atau CAR Admin dapat meng-klik tombol Delete. Selanjutnya akan ditampilkan dialog konfirmasi untuk penghapusan pelanggan yang dimaksud.

public function delete\_user($userid)

{

$query = $this->db->query("DELETE FROM user\_table WHERE userid = " . $userid);

}

### Menghapus daftar pengguna (Controller)

public function delete()

{

$this->load->model('user\_table\_model');

if ($\_POST['user\_id\_hidden'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Account deleting failed...');

redirect('/dashboard/user', 'refresh');

}

else

{

$this->user\_table\_model->delete\_user($\_POST['user\_id\_hidden']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Account deleting applied successfully...');

redirect('/dashboard/user', 'refresh');

}

}

## Menampilkan daftar hubungan penugasan antara pelanggan dan pengguna

Fitur melihat daftar penugasan pengguna ke pelanggan tertentu hanya dimiliki oleh role Administrator dan CAR Admin. Namun, dalam pengaksesan halaman ini, terdapat perbedaan cara. Untuk role Administrator bisa mengakses halaman ini melalui menu ‘Assign List’ sedangkan role CAR Admin melalui menu ‘Dashboard’ pada menu yang terdapat pada halaman sebelah samping kiri. Daftar penugasan berupa tabel dengan kolom User, VRF Name, Action yang berisi tombol ‘Delete’.

### Potongan kode untuk file model

public function get\_assignment\_list()

{

$query = $this->db->query('SELECT DISTINCT assign.id\_assign, user\_table.nama, vrf\_data.nama\_vrf FROM user\_table, vrf\_data, assign WHERE assign.userid = user\_table.userid AND assign.id\_vpn = vrf\_data.id\_vpn');

return $query->result();

}

### 

### Potongan kode untuk file controller

private function assignment\_list()

{

$this->load->model('assign\_model');

$data['assignment\_list'] = $this->assign\_model->get\_assignment\_list();

$datas['menus'] = $this->session->userdata('menus');

$datas['inisial'] = $this->session->userdata('inisial');

$datas['is\_admin'] = $this->session->userdata('is\_admin');

$datas['email'] = $this->session->userdata('email');

$datas['userid'] = $this->session->userdata('userid');

$datas['nama'] = $this->session->userdata('nama');

$datas['page\_name'] = 'Dashboard';

$data['page\_name'] = 'Dashboard';

if ($this->session->userdata('is\_admin') == 1)

{

$datas['page\_name'] = 'Assign List';

$data['page\_name'] = 'Assign List';

}

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('assignment\_list\_view', $data);

$this->load->view('partial/footer');

}

## Mendaftarkan pengguna ke pelanggan tertentu

Untuk mendaftarkan pelanggan ke pengguna tertentu atau memberi tugas pengawasan CAR untuk pelanggan-pelanggan tertentu diperlukan role Administrator atau CAR Admin. Untuk menambah daftar penugasan, terdapat tombol ‘Add Assignment’ yang terdapat pada *header* tabel pada halaman daftar penugasan yang akan membawa pengguna sistem kepada menu pop-up yang berisi field User dan VRF Name yang akan diisi pada penambahan penugasan.

### Potongan kode untuk file model

public function add\_assign($userid, $id\_vpn)

{

$query = $this->db->query("INSERT INTO assign (userid, id\_vpn) VALUES (" . $userid . ", " . $id\_vpn . ")");

}

### 

### Potongan kode untuk file controller

public function add()

{

$this->load->model('assign\_model');

if ($\_POST['user\_hidden'] == '' || $\_POST['vrf\_hidden'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Assignment adding failed...');

if ($this->session->userdata('is\_admin') == 3) redirect('/dashboard', 'refresh');

else redirect('/dashboard/assign', 'refresh');

}

else

{

$this->assign\_model->add\_assign($\_POST['user\_hidden'], $\_POST['vrf\_hidden']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Assignment adding added successfully');

if ($this->session->userdata('is\_admin') == 3) redirect('/dashboard', 'refresh');

else redirect('/dashboard/assign', 'refresh'); }

}

## Menghapus pelanggan dari pengguna tertentu

Penghapusan penugasan dari pengguna ke pelanggan tertentu hanya dapat dilakukan oleh role Administrator dan CAR Admin. Fitur ini dapat diakses melalui halaman yang menampilkan daftar penugasan dengan mengklik tombol ‘Delete’ yang terletak pada kolom ‘Action’ pada tabel tersebut.

### Potongan kode untuk file model

public function delete\_assign($id\_assign)

{

$query = $this->db->query("DELETE FROM assign WHERE id\_assign = " . $id\_assign);

}

### Potongan kode untuk file controller

public function delete()

{

$this->load->model('assign\_model');

if ($\_POST['user\_id\_hidden'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Assignment deleting failed...');

if ($this->session->userdata('is\_admin') == 3) redirect('/dashboard', 'refresh');

else redirect('/dashboard/assign', 'refresh');

}

else

{

$this->assign\_model->delete\_assign($\_POST['user\_id\_hidden']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Assignment deleting applied successfully...');

if ($this->session->userdata('is\_admin') == 3) redirect('/dashboard', 'refresh');

else redirect('/dashboard/assign', 'refresh');

}

}

## Menampilkan rincian informasi data VRF pelanggan

Untuk melihat rincian informasi data VRF Pelanggan, pengguna baik untuk role Administrator, CAR Admin, dan CAR harus melalui halaman yang menampilkan daftar pelanggan. Halaman rincian informasi pelanggan akan ditampilkan jika pengguna mengklik nama VRF pelanggan yang terdaftar pada sistem. Terdapat perbedaan antara role Administrator dengan CAR Admin dan CAR. Pada role Adminsitrator terdapat tambahan kolom ‘History Config’.

### Potongan kode untuk file model

public function get\_by\_name($vrf\_name)

{

$vrf\_name = str\_replace('\_', '-', $vrf\_name);

$query = $this->db->query("SELECT \* FROM mpls\_vpn LEFT JOIN perangkat\_backup ON mpls\_vpn.id\_device\_backup = perangkat\_backup.id\_device\_backup WHERE vrf\_name = '$vrf\_name' AND nojar REGEXP '[0-9]{10}'");

return $query->result();

}

### Potongan kode untuk file controller

public function details($vrf\_name)

{

if ($vrf\_name == '') redirect('/dashboard/back');

if ($this->session->userdata('isLoggedIn'))

{

$this->load->model('mpls\_vpn\_model');

$this->load->model('system\_setting\_model');

$this->load->model('tools\_model');

$data['vrf\_data\_details'] = $this->mpls\_vpn\_model->get\_by\_name($vrf\_name);

$data['path'] = $this->system\_setting\_model->get\_setting();

$datas['menus'] = $this->session->userdata('menus');

$datas['inisial'] = $this->session->userdata('inisial');

$datas['is\_admin'] = $this->session->userdata('is\_admin');

$datas['email'] = $this->session->userdata('email');

$datas['userid'] = $this->session->userdata('userid');

$datas['nama'] = $this->session->userdata('nama');

$datas['page\_name'] = 'Details VRF';

$data['page\_name'] = 'Details VRF';

$data['detail\_name'] = $vrf\_name;

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('vrf\_data\_detail\_view', $data);

$this->load->view('partial/footer');

}

else

{

redirect('/login', 'refresh');

}

}

## Melakukan manajemen perangkat pendukung

Manajemen perangkat pengudukung hanya dapat dilakukan oleh role Administrator, sedangkan untuk menampilkan daftar tool yang tersedia untuk semua role. Manajemen perangkat pendukung merupakan fitur yang menyediakan penyuntingan perangkat dalam jaringan lokal perusahaan. Perangkat yang dimaksud di sini adalah aplikasi web lain di luar sistem yang terdapat pada jaringan lokal perusahaan.

### Menampilkan daftar perangkat tersedia (Model)

public function get\_list()

{

$query = $this->db->query('SELECT \* FROM tools');

return $query->result();

}

### Menampilkan daftar perangkat tersedia (Controller)

public function tools()

{

$this->load->model('tools\_management\_model');

$data['tools\_management'] = $this->tools\_management\_model->get\_list();

$datas['menus'] = $this->session->userdata('menus');

$datas['inisial'] = $this->session->userdata('inisial');

$datas['is\_admin'] = $this->session->userdata('is\_admin');

$datas['email'] = $this->session->userdata('email');

$datas['userid'] = $this->session->userdata('userid');

$datas['nama'] = $this->session->userdata('nama');

$datas['page\_name'] = 'Tools Management';

$data['page\_name'] = 'Tools Management';

$this->load->view('partial/header');

$this->load->view('partial/sidebar', $datas);

$this->load->view('tools\_view', $data);

$this->load->view('partial/footer');

}

### Penambahan perangkat (Model)

Penambahan dapat dilakukan dengan mengklik tombol ‘Add Tool’ pada bagian *header* tabel. Aksi ini akan menampilkan menu *popup* yang berisi *field* *Tool Name, Tool Address*, dan *Logo*.

public function add\_tool($tool\_addr, $tool\_name, $tool\_logo)

{

if ($tool\_logo == '')

{

$query = $this->db->query("INSERT INTO tools (URL, NAMA\_TOOL) VALUES ('$tool\_addr', '$tool\_name')");

}

else

{

$query = $this->db->query("INSERT INTO tools (URL, NAMA\_TOOL, LOGO\_TOOL) VALUES ('$tool\_addr', '$tool\_name', '$tool\_logo')");

}

}

### Penambahan perangkat (Controller)

public function add\_tool()

{

if ($\_POST['tool\_name\_add'] == '' || $\_POST['tool\_address\_add'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Tool adding failed...');

redirect('/dashboard/tools', 'refresh');

}

else

{

if (isset($\_FILES['tool\_logo\_add']))

{

move\_uploaded\_file($\_FILES['tool\_logo\_add']['tmp\_name'], 'logos/' . $\_FILES['tool\_logo\_add']['name']);

$\_POST['tool\_logo\_add'] = base\_url() . 'logos/' . $\_FILES['tool\_logo\_add']['name'];

}

$this->load->model('tools\_management\_model');

$this->tools\_management\_model->add\_tool($\_POST['tool\_address\_add'],$\_POST['tool\_name\_add'], $\_POST['tool\_logo\_add']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Tool adding applied successfully..');

redirect('/dashboard/tools', 'refresh');

}

}

### Penyuntingan perangkat tersedia (Model)

Penyuntingan dapat dilakukan oleh pengguna dengan mengklik tombol ‘Edit’ yang terdapat pada kolom ‘Action’. Aksi ini akan menampilkan menu *popup* yang berisi field *Tool Name, Tool Address,* dan *Logo*.

public function edit\_tool($tool\_id, $tool\_addr, $tool\_name, $tool\_logo)

{

if ($tool\_logo == '')

{

$query = $this->db->query("UPDATE tools SET NAMA\_TOOL = '$tool\_name', URL = '$tool\_addr' WHERE ID\_TOOL = " . $tool\_id);

}

else

{

$tool\_logo = $tool\_logo;

$query = $this->db->query("UPDATE tools SET NAMA\_TOOL = '$tool\_name', URL = '$tool\_addr', LOGO\_TOOL = '$tool\_logo' WHERE ID\_TOOL = " . $tool\_id);

}

}

### Penyuntingan perangkat tersedia (Controller)

function edit\_tool()

{

$server = "http://127.0.0.1";

if ($\_POST['toolid\_hidden'] == '' || $\_POST['tool\_name\_edit'] == '' || $\_POST['tool\_address\_edit'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Tool edit failed');

redirect('/dashboard/tools', 'refresh');

}

else

{

if (isset($\_FILES['tool\_logo\_edit']))

{

move\_uploaded\_file($\_FILES['tool\_logo\_edit']['tmp\_name'], 'logos/' . $\_FILES['tool\_logo\_edit']['name']);

$\_POST['tool\_logo\_edit'] = base\_url() . 'logos/' . $\_FILES['tool\_logo\_edit']['name'];

}

$this->load->model('tools\_management\_model');

$this->tools\_management\_model->edit\_tool($\_POST['toolid\_hidden'], $\_POST['tool\_address\_edit'], $\_POST['tool\_name\_edit'], $\_POST['tool\_logo\_edit']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Tool edit success');

redirect('/dashboard/tools', 'refresh');

}

}

### Penghapusan perangkat (Model)

Penyuntingan dapat dilakukan oleh pengguna dengan mengklik tombol ‘Delete’ yang terdapat pada kolom ‘Action’. Aksi ini akan menampilkan menu *popup* konfirmasi penghapusan tool.

public function delete\_tool($toolid)

{

$query = $this->db->query("DELETE FROM tools WHERE ID\_TOOL = " . $toolid);

}

### Penghapusan perangkat (Controller)

public function delete\_tool()

{

if ($\_POST['tool\_id\_hidden'] == '')

{

$this->session->set\_flashdata('type', 'danger');

$this->session->set\_flashdata('msg', 'Tool deleting failed...');

redirect('/dashboard/tools', 'refresh');

}

else

{

$this->load->model('tools\_management\_model');

$this->tools\_management\_model->delete\_tool($\_POST['tool\_id\_hidden']);

$this->session->set\_flashdata('type', 'success');

$this->session->set\_flashdata('msg', 'Tool deleting applied successfully...');

redirect('/dashboard/tools', 'refresh');

}

}

## Menampilkan konfigurasi VRF pelanggan

Konfigurasi terakhir dari node-node pelanggan disimpan dalam server lain yang berbeda dengan sistem. Konfigurasi ini berupa file text yang akan dijalankan oleh background process ketika terjadi proses pemulihan jaringan pelanggan. Fitur ini dapat diakses melalui halaman yang menampilkan rincian data pelanggan. Link yang ditampilkan akan dienkripsi terlebih dahulu, setelah itu melalui controller, program melakukan http request ke server yang menyimpan file konfigurasi. Hal ini dimaksudkan agar pengguna yang tidak berkepentingan terhadap pelanggan yang bukan menjadi hak aksesnya tidak dapat mengakses konfigurasi terakhir dari node-node pelanggan.

### Potongan kode untuk view

echo "<td class='text-center'><a class='btn btn-success' target='\_blank' href='" . site\_url() . "/openfile/get/" . $this->tools\_model->encrypt\_decrypt("en", ($row->vrf\_name . "/" . $row->nojar . ".txt")) . "'>" . $row->backup\_date . "</a></td>";

### Potongan kode untuk model

function encrypt\_decrypt($action, $string)

{

$output = false;

$key = 'L1nt4sARTA';

$iv = md5(md5($key));

if( $action == 'en' )

{

$output = mcrypt\_encrypt(MCRYPT\_RIJNDAEL\_256, md5($key), $string, MCRYPT\_MODE\_CBC, $iv);

$output = base64\_encode($output);

$output = str\_replace("+", "\_", $output);

$output = str\_replace("/", "~", $output);

$output = str\_replace("=", "-", $output);

}

else if( $action == 'de' )

{

$string = str\_replace("\_", "+", $string);

$string = str\_replace("~", "/", $string);

$string = str\_replace("-", "=", $string);

$output = mcrypt\_decrypt(MCRYPT\_RIJNDAEL\_256, md5($key), base64\_decode($string), MCRYPT\_MODE\_CBC, $iv);

$output = rtrim($output, "");

}

return $output;

}

### Potongan kode untuk controller

public function get($uri)

{

if ($this->session->userdata('is\_admin') == 1)

{

$homepage = 'Not Found';

$this->load->model('tools\_model');

$this->load->model('system\_setting\_model');

$back = $this->tools\_model->encrypt\_decrypt("de", $uri);

$el = $this->system\_setting\_model->get\_setting();

$compath = "http://" . $el->server\_addr . "/" . $el->dir\_config . "/" . $el->last\_dir\_config . "/" . $back;

$ch = curl\_init($compath);

curl\_setopt($ch, CURLOPT\_NOBODY, true);

curl\_exec($ch);

$retcode = curl\_getinfo($ch, CURLINFO\_HTTP\_CODE);

if ($retcode == 200)

{

$curlSession = curl\_init();

curl\_setopt($curlSession, CURLOPT\_URL, $compath);

curl\_setopt($curlSession, CURLOPT\_BINARYTRANSFER, true);

curl\_setopt($curlSession, CURLOPT\_RETURNTRANSFER, true);

$homepage = curl\_exec($curlSession);

curl\_close($curlSession);

}

curl\_close($ch);

header("Status: 200 OK"."\r\n");

header("Date: " . gmdate(DATE\_RFC822) . "\r\n");

header("Content-Type: text/plain charset=UTF-8\r\n");

header("Content-Length: " . strval(strlen($homepage))."\r\n");

echo $homepage;

}

else redirect('/dashboard', 'refresh');

}

# BAB VI UJI COBA DAN EVALUASI

Pada bab ini, akan dijelaskan mengenai proses uji coba yang dilakukan untuk setiap fungsionalitas dari aplikasi VPN Management. Selain itu, akan diuraikan juga hasil evaluasi dari uji coba tersebut.



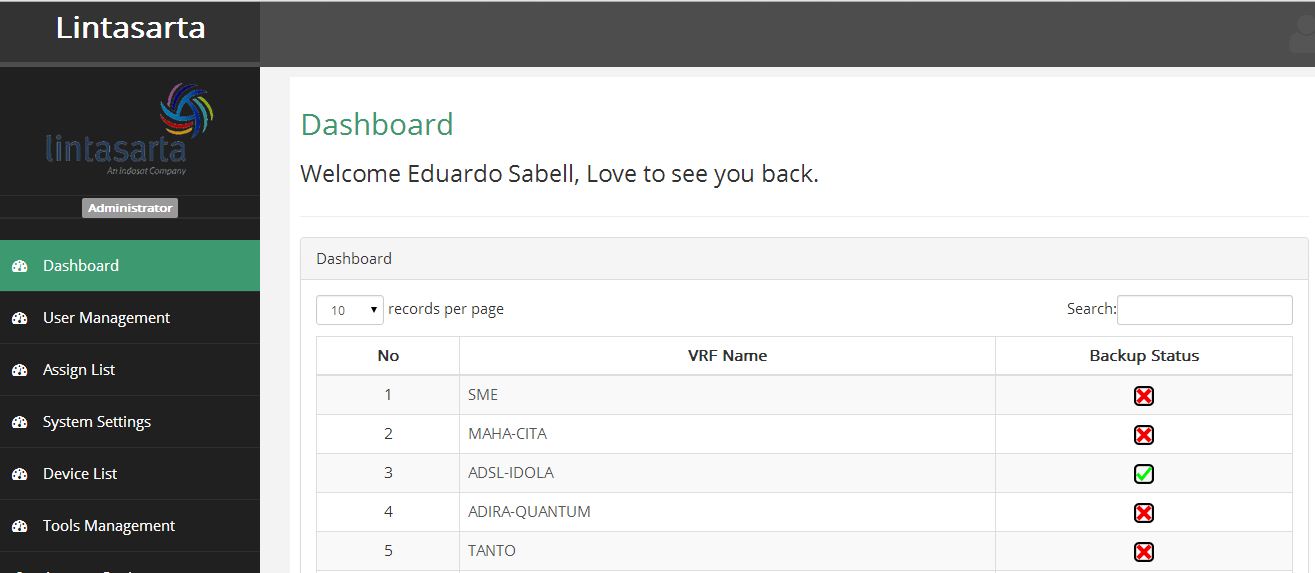
## Lingkungan Uji Coba

Lingkungan Uji Coba yang digunakan dalam proses pengembangan aplikasi VPN Management adalah Ubuntu 14.04, dan perangkat pengembang yang digunakan adalah Sublime Text Editor, dan peramban Internet Explorer atau Google Chrome.

## Menampilkan Hasil Uji Coba

### Menampilkan Daftar Pelanggan dan Status Backup VRF Pelanggan

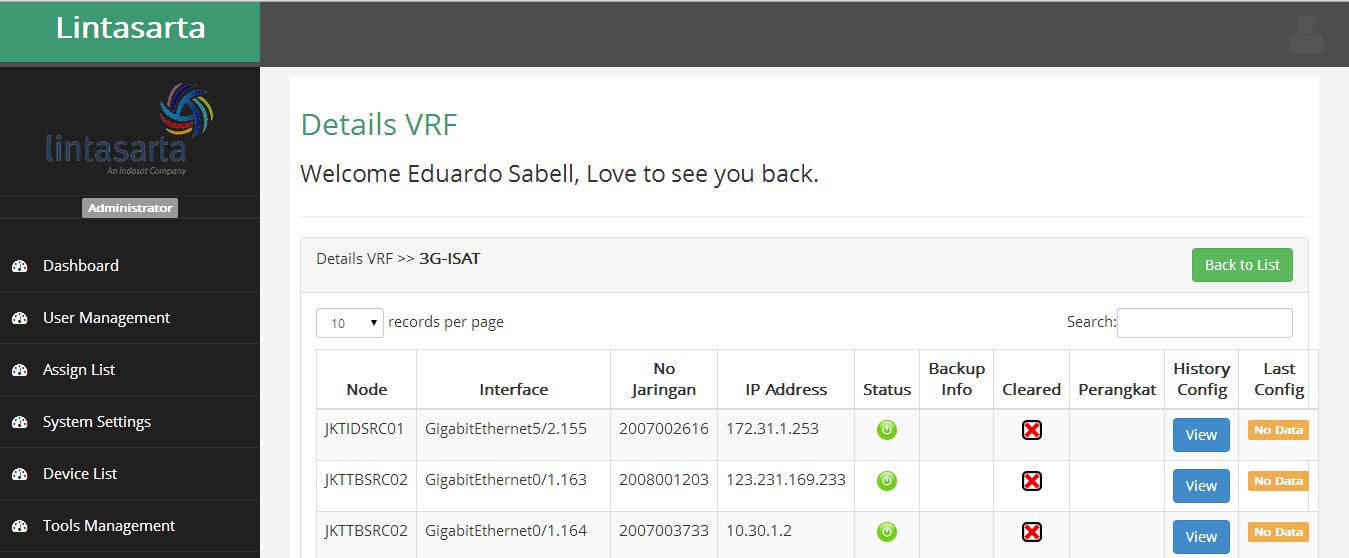
Hasil uji coba dari fungsi-fungsi yang diimplementasikan untuk menampilkan keberhasilan dari proses yang dilakukan penggunaditunjukkan oleh gambar 6.1, gambar 6.2, gambar 6.3, gambar 6.4, gambar 6.5, gambar 6.6, gambar 6.7.



Gambar . Tampilan hasil uji coba menampilkan daftar VRF pelanggan dan status backup VRF pelanggan

Melalui gambar 6.1 dapat dilihat bahwa administrator dapat mencari nama pelanggan melalui kolom pencarian dengan mengetikkan nama VRF pelanggan. Administrator juga dapat mengubah status backup pelanggan ataupun melihat detil informasi VRF pelanggan seperti pada gambar 6.2

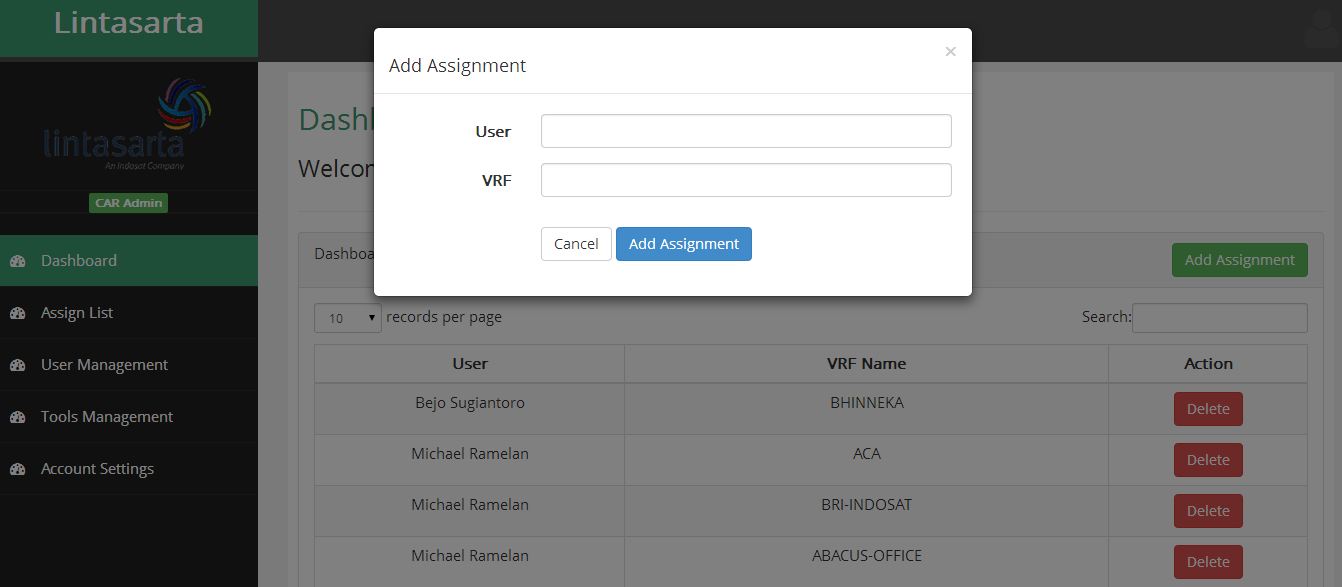
### Menampilkan Rincian Informasi Data VRF Pelanggan



Gambar . Tampilan hasil ujicoba menampilkan rincian informasi data VRF Pelanggan

Hasil uji coba dan evaluasi yang telah dilakukan adalah aplikasi ini dapat menampilkan detil informasi data VRF pelanggan. Pengguna dapat melakukan pencarian node jaringan dalam VRF pelanggan tersebut melalui kolom pencarian. Detil informasi diperlukan untuk identifikasi ketika terjadi masalah pada jaringan pelanggan.

### Mendaftarkan Pelanggan ke Petugas Tertentu

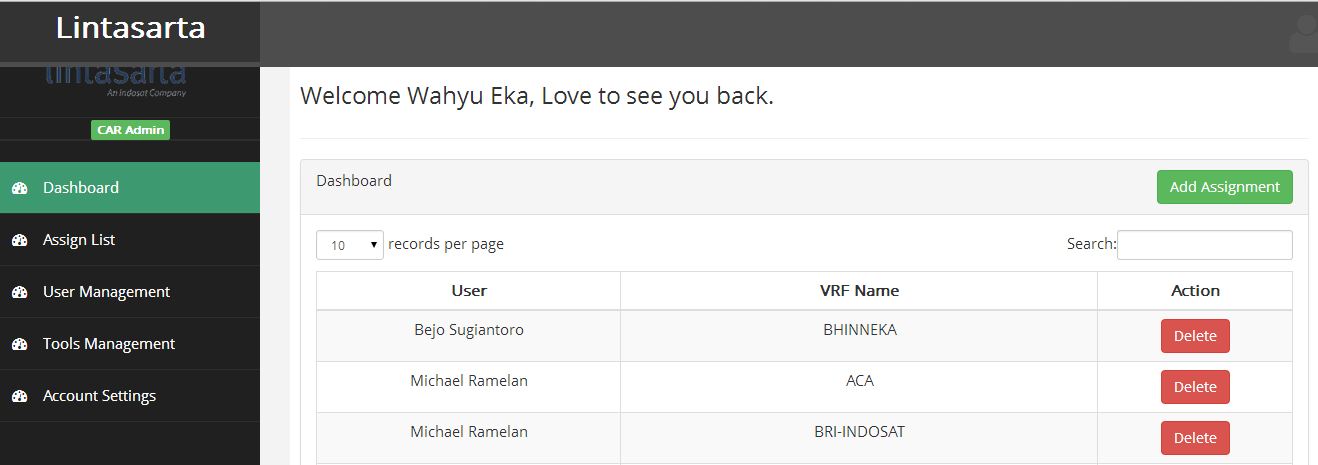


Gambar . Tampilan hasil uji coba mendaftarkan pelanggan ke petugas tertentu

Hasil uji coba dan evaluasi yang telah dilakukan adalah aplikasi ini dapat mendaftarkan pelanggan untuk petugas tertentu. Hal ini untuk mempermudah pengguna yg memiliki hak akses CAR untuk melihat pelanggan yang ditangani tanpa harus memuat semua daftar pelanggan.

CAR admin mengalokasikan pelanggan untuk petugas tertentu dengan memilih menu add assignment kemudian mengisi form yg berisi nama petugas dan pelanggan yang didaftarkan.

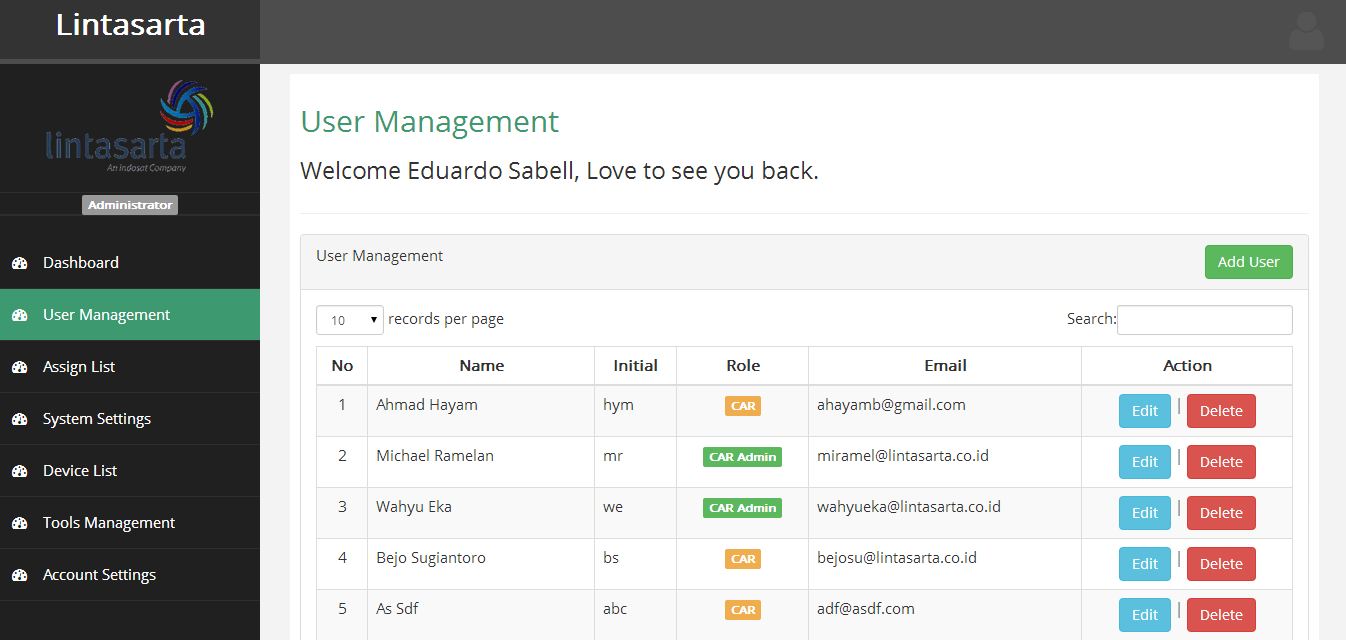
### Menghapus Pelanggan dari Petugas Tertentu



Gambar . Tampilan hasil uji coba menghapus pelanggan dari petugas tertentu

Hasil uji coba dan evaluasi yang telah dilakukan adalah aplikasi ini dapat melakukan penghapusan pelanggan dari petugas tertentu. Hal ini berfungsi untuk efisiensi dan kerapihan data pelanggan yang terdaftar.

### Melakukan Proses Manajemen Pengguna

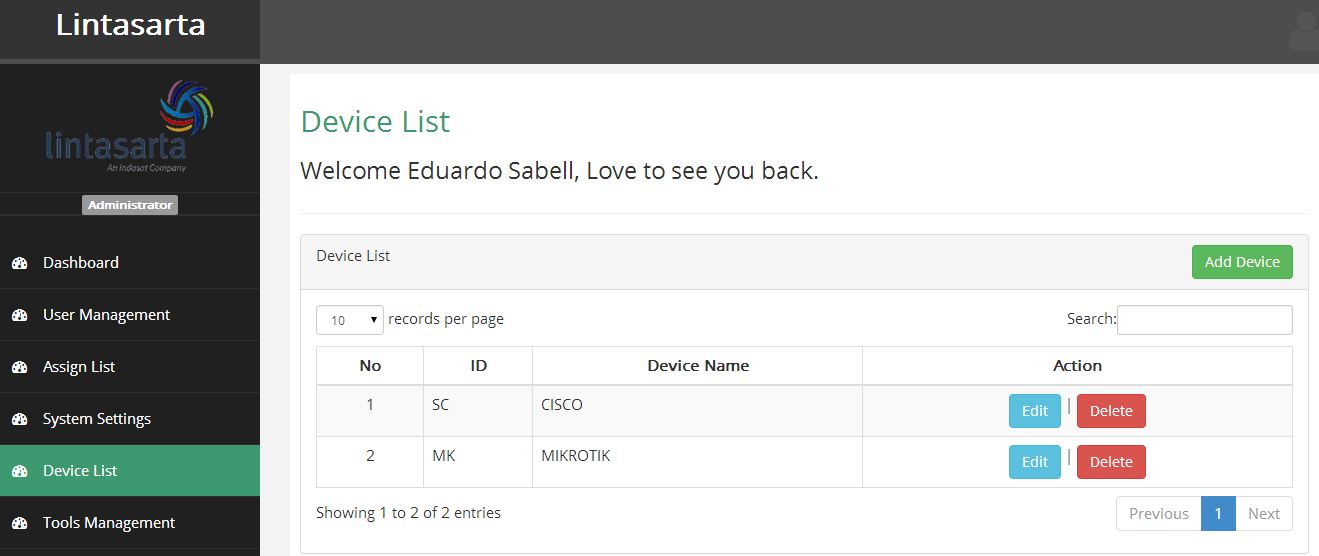


Gambar .. Tampilan hasil uji melakukan proses manajemen pengguna

Melalui gambar 6.5, dapat dilihat bahwa administrator memiliki hak akses untuk melakukan manajamen pengguna. Penambahan pengguna baru dilakukan dengan menekan menu adduser,kemudian mengisi inisial,nama,email dan hak aksesnya pada form yang disediakan.

Data yang berhasil disimpan akan muncul pada daftar pengguna. Proses penyuntingan dan penghapusan dapat dilakukan pada pengguna yang ada pada daftar.

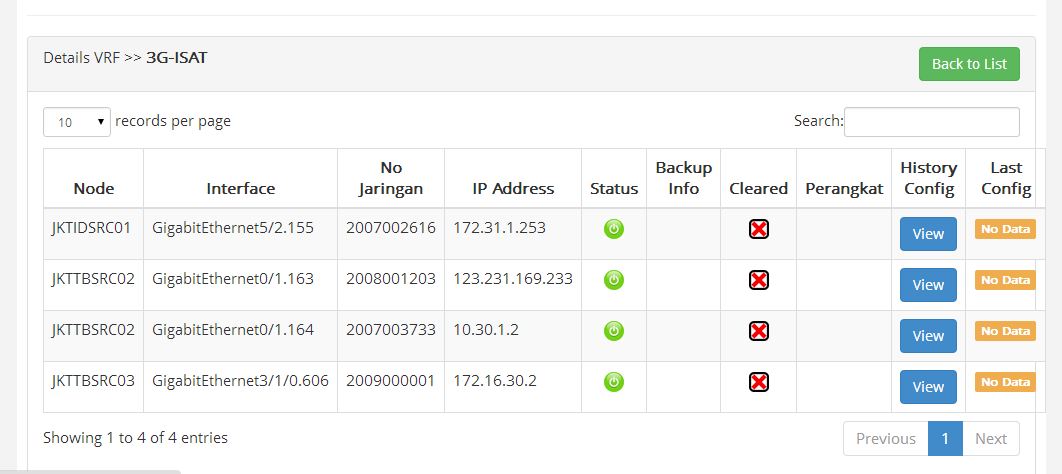
### Melakukan Manajemen Perangkat



Gambar . Tampilan hasil uji coba melakukan manajemen perangkat

Fitur manjemen perangkat digunakan untuk menyimpan data perangkat yang digunakan pelanggan. Tujuannya agar ketika terjadi masalah pada jaringan pelanggan,sistem dapat mengidentifikasi perangkat pelanggan yang digunakan.

### Menampilkan konfigurasi VRF pelanggan



Gambar . Hasil tampilan uji coba menampilkan konfigurasi VRF pelanggan

Hasil uji coba dan evaluasi yang telah dilakukan adalah aplikasi ini dapat digunakan untuk melihat konfigurasi pada VRF. Ketika terjadi masalah pada jaringan pelanggan maka konfigurasi pelanggan dapat dilihat untuk identifikasi masalah.

# BAB VII KESIMPULAN DAN SARAN



## Kesimpulan

Kontrol data pelanggan melalui aplikasi VPN Management yang dilakukan oleh tiga pengguna dengan hak akses administrator, CAR,dan CAR admin dapat memudahkan perusahaan PT. Aplikanusa Lintasarta yang didalamnya terdapat antarmuka website yang sederhana sehingga dapat memudahkan pengguna dalam melakukan kontrol jaringan pelanggan. Kontrol jaringan pelanggan yang ditangani sistem adalah backup konfirugrasi jaringan, performa jaringan dan repository konfigurasi jaringan pelanggan. Sedangkan untuk manajemen pelanggan meliputi penugasan pelanggan ke pengguna tertentu.

Selain itu, aplikasi VPN Management ini dapat menampilkan transaksi berdasarkan keterangan waktu melalui tampilan berbentuk kalender. Dengan tampilan yang teratur dan rapi disertai dengan keterangan transaksi yang detail, fitur tersebut dapat membantu sales dalam pencatatan pengiriman dan pendistribusian bulk LPG.

Secara keseluruhan, aplikasi VPN Management ini dapat membantu petugas di bagian Operation Support PT. Aplikanusa Lintasarta pada khususnya untuk melakukan kontrol jaringan pelanggan menggunakan sistem informasi yang terintegrasi dengan baik. Petugas dapat langsung memeriksa status keaktifan jaringan pelanggan dan memeriksa konfigurasinya. Petugas juga dapat melakukan pengelolaan pelanggan dan pengguna aplikasi ini dengan mudah Sehingga, dengan menggunakan aplikasi VPN management ini, kontrol jaringan VRF pelanggan dapat dilakukan dengan lebih praktis dan efisien.

## Saran

Dalam pengerjaan pengembangan aplikasi web VPN Management, struktur basis data yang digunakan masih belum memenuhi aturan-aturan normalisasi. Sehingga, ketika akan dilakukan pengambilan data melalui query harus menelusuri ulang struktur basis data yang digunakan.

Walaupun aplikasi web ini dibangun diatas perangkat kerja MVC Codeigniter berbasis PHP, implementasi OOP belum dilakukan secara tepat. Beberapa kerancuan fungsi masih banyak terdapat pada file controller, sehingga diperlukan usaha coba-coba untuk menambahkan fitur pada aplikasi yang sudah ada.

Selain itu, penggunaan fitur versioning pada pengembangan perangkat lunak akan membantu projek-projek yang dikerjakan secara grup. Fitur versioning yang mampu digunakan adalah github dan bitbucket. Fitur seperti ini, membantu pembagian kerja yang lebih efisien antar programmer dan sinkronisasi kode program dan progress aplikasi. Dokumentasi program agaknya perlu diadakan untuk mempermudah pengembangan kedepan dan manual program untuk membantu pengguna dalam menggunakan sistem secara tepat.

# DAFTAR PUSTAKA

1. Sintaks PHP [diakses pada bulan Juli 2013]

http://www.w3schools.com/php/

1. Bootstrapvalidator, pustaka form validasi [diakses pada bulan Juli 2013]

http://bootstrapvalidator.com/getting-started/

1. Sintaks Javascript [diakses pada bulan Juli 2013]

http://www.w3schools.com/js/

1. Bootstrap Grid System [diakses pada bulan Juli 2013]

http://getbootstrap.com/examples/grid/

1. Chart.js, pustaka grafik berbasis javascript [diakses pada bulan Juli 2013]

http://www.chartjs.org/docs/

1. Bootstrapgrowl, pustaka notifikasi untuk bootstrap

http://bootstrap-growl.remabledesigns.com/