

Aplikasi Pengelolaan Basis Data Konfigurasi

(Studi Kasus Direktorat Sistem Informasi Universitas Telkom)

Annisa Humairo¹, Dahliar Ananda², Siska Komala Sari³

^{1,2,3}Program Studi D3 Sistem Informasi, Fakultas Ilmu Terapan Universitas Telkom

¹annisahumairo46@gmail.com, ²ananda@telkomuniversity.ac.id, ³siska@tass.telkomuniversity.ac.id

Abstrak

Universitas Telkom memiliki Direktorat Sistem Informasi (DITSISFO) yang bertanggung jawab dalam memberikan layanan sistem informasi untuk memfasilitasi kegiatan yang dilakukan oleh sivitas Universitas Telkom. Banyaknya layanan sistem informasi yang dimiliki Ditsisfo menyebabkan seringnya dilakukan perubahan dan pengembangan terhadap layanan sistem informasi guna mengikuti perkembangan dan kebutuhan sivitas Universitas Telkom. Perubahan yang dilakukan dapat berupa perubahan berskala besar maupun kecil sehingga hal ini menyebabkan sulitnya melakukan pengolahan dan pelacakan perubahan atau pengembangan yang terjadi. Oleh karena itu dibuatlah aplikasi pengolahan konfigurasi database aplikasi guna memfasilitas pegawai Ditsisfo dalam melakukan pengolahan konfigurasi item dan pelacakan perubahan yang terjadi pada konfigurasi item dan relasi konfigurasi item.

Aplikasi ini dibuat dengan menggunakan metode *waterfall*, bahasa pemrograman HTML, PHP dan framework CodeIgniter dalam pembangunan logika aplikasi serta menggunakan Bootstrap dan JavaScript untuk membangun tampilan aplikasi dan menggunakan MySQL sebagai penyimpanan database aplikasi. Aplikasi ini menggunakan tata cara penyimpanan data CMDB (Configuration Management Database).

Kata kunci : Direktorat Sistem Informasi dan CMDB

Abstract

Telkom University has a Directorate of Information Systems (DITSISFO) that are responsible and provides information system services to facilitate the activities all of civitas Telkom University. The information system services owned by Ditsisfo causes frequent changes and developments in information system services to keep abreast of developments and needs of civitas Telkom University. That changes can be in the major or minor scale so that is causes difficulty in processing and tracking changes or developments that occur. Therefore an application database configuration configuration application is created to facilitate Ditsisfo employees in processing configuration items and tracking changes that happen in item configuration and item configuration relationships.

This application was created using the waterfall method, programming language HTML, PHP and CodeIgniter framework to build the application logic and using Bootstrap and JavaScript to build application views and use MySQL as the application database storage. This application uses the CMDB data storage procedure (Configuration Management Database).

Keywords : Directorate of Information Systems and CMDB

I. PENDAHULUAN

Pada bagian ini akan dibahas terkait latar belakang serta tujuan dilakukannya penelitian.

A. Latar Belakang

Universitas Telkom memiliki Direktorat Sistem Informasi (Ditsisfo) yang bertanggung jawab terhadap implementasi teknologi informasi dan komunikasi yang ada di Universitas Telkom. Ditsisfo memberikan layanan infrastruktur teknologi informasi layanan interkoneksi, layanan data dan sistem informasi serta layanan komputasi sebagai *strategic tools* untuk mendukung berjalannya proses bisnis yang ada di Universitas Telkom[1]. Banyaknya layanan sistem informasi yang diberikan oleh Ditsisfo menyebabkan dilakukannya pengembangan dan perubahan aplikasi sehingga aplikasi dapat beroperasi dengan baik, perubahan dan pengembangan yang dilakukan dapat berskala besar maupun kecil sehingga menyebabkan perubahan yang saling terkait antar satu aplikasi dengan aplikasi lainnya, baik dari sisi proses bisnis, kode, data, dan database.

Dampak dari perubahan dan pengembangan yang dilakukan menyebabkan permasalahan pada aplikasi yang berjalan, banyaknya konfigurasi yang saling terkait menyebabkan sulitnya melakukan dokumentasi terhadap perubahan yang terjadi, sulit melakukan *tracking* terhadap aplikasi yang terkena dampak dari perubahan dan pengembangan yang dilakukan sehingga dokumentasi dan *tracking* perubahan dilakukan secara manual dengan melakukan penelusuran terhadap dokumen teknis item konfigurasi dalam bentuk fisik maupun file.

Berdasarkan permasalahan yang telah dipaparkan, maka dengan dilakukannya pembangunan aplikasi pengelolaan basis data konfigurasi pada suatu aplikasi, diharapkan dapat memfasilitasi dalam kegiatan pengelolaan dan *tracking* terhadap setiap item konfigurasi aplikasi.

Aplikasi usulan dibuat untuk mencatat setiap perubahan yang terjadi pada aplikasi-aplikasi yang terkait ketika suatu aplikasi dilakukan perubahan atau pengembangan sehingga efek dari perubahan dan pengembangan dapat ditangani.

B. Rumusan Masalah

Berdasarkan permasalahan yang telah disampaikan di atas, maka pembangunan aplikasi untuk pengelolaan basis data konfigurasi dilakukan dalam rangka menyelesaikan permasalahan sebagai berikut.

- Bagaimana cara melakukan pengelolaan basis data relasi konfigurasi di lingkungan Direktorat Sistem Informasi (Ditsisfo)?
- Bagaimana cara memfasilitasi pegawai Ditsisfo dalam melakukan *tracking* perubahan basis data relasi konfigurasi aplikasi?

C. Tujuan Masalah

Berdasarkan rumusan masalah yang ada maka tujuan dari penyelesaian masalah adalah sebagai berikut.

- Mengembangkan suatu aplikasi *Configurasi Management Database* (CMDB) yang dapat digunakan untuk melakukan pengelolaan terkait sumber daya teknis aplikasi.
- Mengembangkan aplikasi CMDB yang dapat melakukan *tracking* terhadap perubahan basis data relasi konfigurasi pada sebuah aplikasi.

D. Metode Pengerjaan

Model proses pembangunan perangkat lunak yang digunakan pada aplikasi menggunakan metode Waterfall [2] yang disampaikan oleh Ian Sommerville pada bukunya *Software Engineering 9th Edition*. Pada model waterfall, terdapat 5 (lima) aktivitas yang harus dilakukan pada pembangunan perangkat lunak, yaitu:

- *Requirement Definition* (Pendefinisian Kebutuhan)

Pada aktivitas pendefinisian kebutuhan, peneliti akan merumuskan kebutuhan fungsional perangkat lunak dengan menggali data kebutuhan mengenai detail teknis yang akan dicatat dalam aplikasi CMDB yang dibangun. Kebutuhan fungsionalitas perangkat lunak didapat melalui wawancara dan survei ketempat studi kasus.

- *System and Software Design* (Perancangan Sistem dan Perangkat Lunak)

Pada tahap ini, daftar kebutuhan fungsionalitas yang sudah didapatkan akan dimodelkan kedalam suatu perancangan sistem dan perangkat lunak. Model yang akan digunakan untuk merancang sistem dan perangkat lunak adalah *Unified Modelling Language* (UML) dan *Entity Relationship Diagram* (ER-D).

- *Implementation and Unit Testing* (Implementasi dan Pengujian Unit)

Aktivitas implementasi dan pengujian unit dilakukan untuk memastikan fungsionalitas yang sudah diterjemahkan ke dalam perangkat lunak dapat berjalan dengan semestinya dan memenuhi tujuan yang telah ditetapkan. Pada penelitian ini, pengujian difokuskan kepada kehandalan setiap fungsionalitas yang dibangun pada perangkat lunak. Penelitian lanjutan akan dilakukan untuk menguji kontribusi perangkat lunak untuk mencapai hasil yang sudah ditentukan.

- *Integration and System Testing* (Integrasi dan Pengujian Sistem)

Setelah semua fungsionalitas dipastikan dapat berjalan dengan semestinya, perangkat lunak akan diuji secara lengkap pada lingkungan yang sebenarnya. Perangkat lunak yang sudah *install* pada *mobile device* akan diuji

penggunaannya dengan menghubungkan data yang tersimpan pada server basis data melalui jaringan data.

- *Operation and Maintenance* (Pengoperasian dan Peremajaan)

Tahapan terakhir yang dilakukan adalah menggunakan perangkat lunak dengan melibatkan pengguna yang semestinya untuk memastikan kegunaan dan manfaat perangkat lunak yang sudah dibangun dapat dirasakan oleh penggunaannya. Masukan-masukan yang diberikan oleh pengguna terkait performansi perangkat lunak yang akan menjadi rujukan pengembangan perangkat lunak kedepannya. Pada saat pengoperasian perangkat lunak sehingga dapat diukur kontribusi perangkat lunak terhadap hasil yang telah ditargetkan.

II. TINJAUAN PUSTAKA

A. *Configurasi Management Database* (CMDB)

Configuration Management Database (CMDB) adalah tempat penyimpanan data untuk mencatat atribut dari CI dan relasi CI yang ada selama proses daur hidup dari layanan atau aplikasi tersebut. CMDB haruslah dikelola untuk memastikan kehandalan dan akurasi termasuk kontrol terhadap akses perubahan (*update*). Informasi yang ada dalam CMDB harus selalu tersedia untuk setiap perubahan dan harus dilakukan proses audit secara berkala untuk memastikan setiap data yang tercatat adalah benar [3].

B. *Configurasi Item* (CI)

Item konfigurasi atau *Configuration Item* (CI) adalah elemen yang harus dikontrol untuk mendukung setiap layanan yang ada. Setiap CI yang ada harus didefinisikan dan harus dipastikan memiliki kontrol yang efektif terhadap hal berikut [3]:

1. Pengidentifikasian yang jelas
2. Petunjuk penggunaan
3. Deskripsi CI
4. Penanggung Jawab
5. Klasifikasi
 - a. Kategori Perangkat (Hardware, Software)
 - b. Pendukung pencatatan data (server)
6. Informasi Produsen
 - a. Nama produsen

- b. Nomor seri
- c. Nomor lisensi
- 7. Versi
- 8. Riwayat Pembaruan Data CI
 - a. Tanggal pembuatan
 - b. Update
 - i. Deskripsi pembaruan
 - ii. Tanggal pembaruan
 - iii. Penanggung jawab
- 9. Tempat
- 10. Riwayat Status
 - a. Status saat ini
 - b. Riwayat status
 - i. Perubahan status
 - ii. Deskripsi
 - iii. Waktu dan tanggal perubahan status

11. Hubungan dengan Layanan IT

12. Relasi dengan CI lain

C. Usecase Diagram

Usecase diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, usecase mempresentasikan sebuah interaksi antar aktor dan sistem. Aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan tertentu. Sebuah usecase dapat meng-include fungsionalitas usecase lain sebagai bagian dari sebuah proses, usecase juga dapat meng-extend usecase lain dengan *behaviornya* sendiri. Sementara hubungan generalisasi antar usecase menunjukkan bahwa usecase yang satu merupakan spesialisasi dari yang lain [4].

D. CodeIgniter

CodeIgniter adalah sebuah *framework* PHP yang digunakan oleh *developer* dalam pengembangan aplikasi web berbasis PHP. *Framework* adalah kumpulan kelas (*class*) dan fungsi (*function*) yang disusun secara sistematis berdasarkan kegunaan suatu aplikasi. CodeIgniter menyediakan banyak library yang dapat digunakan dalam proses pembangunan aplikasi.

- E. CodeIgniter memiliki akses tercepat dibanding *framework* lainnya dan berkas kode dapat disusun secara sistematis sesuai dengan struktur yang ditawarkan oleh *framework*. MVC (*Model, View, Controller*) adalah konsep arsitektur dalam pembangunan aplikasi berbasis web yang membagi aplikasi web kedalam tiga bagian yaitu *model, view, controller*. Model bertugas untuk mengatur, menyimpan, memanipulasi dan mengorganisir

data (basis data) sesuai dengan kebutuhan aplikasi yang dipanggil sesuai intruksi dari *controller*. *View* merupakan bagian aplikasi yang bertugas untuk menyajikan tampilan aplikasi kepada pengguna yang dipanggil sesuai intruksi yang dibuat pada *controller*. *Controller* adalah bagian aplikasi yang memiliki fungsi untuk menyimpan semua prosedur untuk memanggil *model* dan *view*[5].

F. MySQL

MySQL adalah sebuah perangkat lunak sistem untuk manajemen sebuah database, MySQL dapat digunakan untuk membuat dan mengelola data dan memiliki sistem manajemen database yang bersifat rasional dengan menjalankan perintah-perintah *Structured Query Language* (SQL) untuk mengelola database yang telah dibentuk di dalamnya[6].

G. PHP

PHP adalah bahasa pemrograman web yang bersifat server side (bekerja di sisi server). PHP merupakan script yang menyatu dengan Hyper Text Markup Language (HTML) dan berada pada server side HTML. Sebagian besar sintak PHP memiliki kemiripan dengan bahasa C, Java, ASP, dan Perl, ditambah beberapa fungsi spesifik yang dimiliki oleh PHP. Tujuan utama bahasa PHP adalah untuk memungkinkan perancangan web dinamik dengan cepat dan PHP adalah bahasa pemrograman yang *powerfull* karena dapat digabungkan dengan library tambahan yang dapat membantu membuat sebuah web dinamis dengan baik.[7].

PHP dapat dijalankan pada beberapa sistem operasi seperti Linux, Unix, Microsoft Windows, Mac Os. Selain itu dukungan teknis juga tersedia luas karena PHP menyediakan dukungan gratis via daftar diskusi *e-mail*. Hal inilah yang menyebabkan PHP semakin banyak digunakan dikalangan para programmer berpengalaman maupun programmer pemula[7].

H. Black Box Testing

Pengujian *black box* merupakan pengujian yang berfokus pada persyaratan fungsionalitas perangkat lunak, pengujian memungkinkan perekrutan perangkat lunak mendapatkan serangkaian input yang sepenuhnya menggunakan semua persyaratan fungsionalitas untuk suatu program. Hal yang akan didapatkan setelah melakukan pengujian adalah sebagai berikut[8]:

1. Fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.

3. Kesalahan pada struktur data atau akses database eksternal.
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi.

III. ANALISIS DAN PERANCANGAN

Pada bagian ini akan dibahas terkait perancangan aplikasi berdasarkan tahapan metode pengerjaan penelitian.

A. Analisis Kelemahan Sistem Berjalan dan Usulan Perbaikan

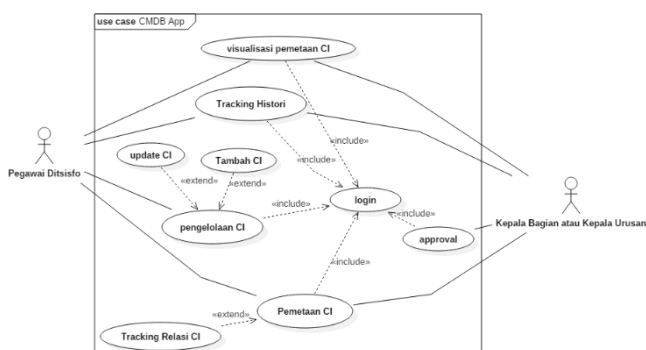
Pada sistem yang sedang berjalan saat ini terdapat beberapa kelemahan, yaitu:

1. Tidak adanya pengelolaan terkait setiap perubahan yang terjadi pada aplikasi.
2. Sulitnya melakukan *tracking* terhadap perubahan yang terjadi sehingga menyebabkan program *error* yang sulit diketahui.
3. Terdapat banyaknya dokumen manual terkait pembuatan aplikasi sehingga sulit melihat keterhubungan antar setiap data.

Berdasarkan kelemahan yang ada, maka diperlukan aplikasi yang memiliki fungsi untuk melakukan pengelolaan terkait *Configurasi Item* (CI) yang ada dan melakukan *tracking* terhadap CI terkait.

B. Usecase Diagram

Pada usecase yang dirancang menggambarkan secara detail fungsionalitas yang dimiliki oleh aplikasi beserta aktor pengguna aplikasi. Pada usecase juga ditampilkan aktivitas aktor pada aplikasi.



Gambar 1 Usecase Aplikasi

Pada usecase rancangan aplikasi ini terdapat dua aktor yang akan menggunakan aplikasi yaitu pegawai Ditsisfo dan Kepala Urusan Bidang atau Kepala Bagian Bidang yang memiliki hak akses masing-masing terhadap fungsionalitas tertentu.

C. Entity Relationship Diagram

pada ER-diagram ini akan di tampilkan rancangan database yang akan digunakan pada aplikasi, pada ER-d akan dilihatkan relasi antar tabel untuk melakukan penyimpanan data. Pada Aplikasi yang dibangun memiliki 10 jenis CI yang dapat dicatat, adapun jenis CI tersebut adalah sebagai berikut,

1. Aplikasi
2. Page
3. Modul
4. API
5. Server
6. Skema Database
7. Tabel
8. View
9. Prosedure
10. Dokumen

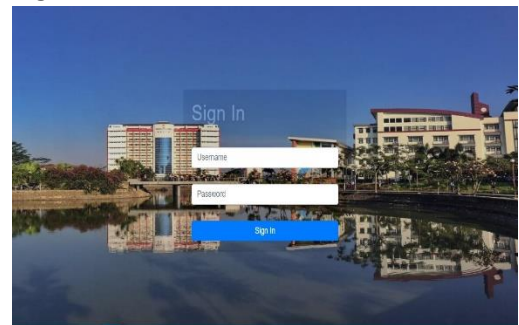
Adapun rancangan ER-d akan di lampirkan pada bagian lampiran.

IV. HASIL DAN PEMBAHASAN

Pada bagian ini akan dibahas terkait hasil dari perancangan aplikasi atau implementasi sistem yang meliputi hasil pengujian setiap form yang ada pada aplikasi.

A. Hasil Implementasi

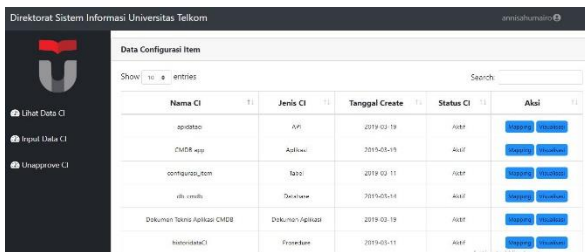
1. Login



Gambar 2 Implementasi Login

Diatas merupakan hasil implementasi halaman login untuk semua pengguna aplikasi, pada halaman di atas pengguna diharuskan untuk memasukkan *username* dan *password* yang dimiliki oleh pengguna. Pengguna akan dapat mengakses fungsionalitas berdasarkan sebagai aktor apa pengguna tersebut masuk ke aplikasi. Pada aplikasi aktor terbagi menjadi tiga yaitu: aktor sebagai pegawai Ditsisfo, aktor sebagai Kepala Bagian Ditsisfo dan sktor sebagai Kepala Urusan setiap bagian Ditsisfo.

2. Pengelolaan CI

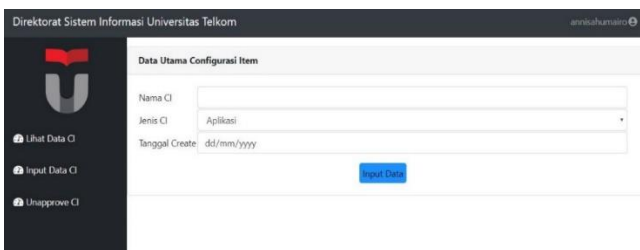


Nama CI	Jenis CI	Tanggal Create	Status CI	Aksi
apibatch	Aktif	2019-02-19	Aktif	[Edit] [Hapus]
CMDB-esp	Aplikasi	2019-05-19	Aktif	[Edit] [Hapus]
configurasi_item	Item	2019-02-11	Aktif	[Edit] [Hapus]
isi_email	Device	2019-05-14	Aktif	[Edit] [Hapus]
Dokumen Teknis Aplikasi CMDB	Dokumen Aplikasi	2019-03-19	Aktif	[Edit] [Hapus]
integratedCI	Proses	2019-05-11	Aktif	[Edit] [Hapus]

Gambar 3 Implementasi Pengelolaan CI

Setelah pengguna berhasil masuk ke dalam aplikasi, pengguna akan berada pada halaman aplikasi yang berisikan data CI yang pernah di input dan telah di setuju penambahan CI oleh atasan pegawai yang menambahkan data baru. Untuk melihat detail dari setiap CI, pengguna dapat melakukan *double click* pada nama CI yang ingin dilihat detailnya, selain melihat detail CI pengguna juga dapat melakukan *double click* pada jenis CI untuk melihat CI apa saja yang telah tercatat pada jenis CI yang dipilih.

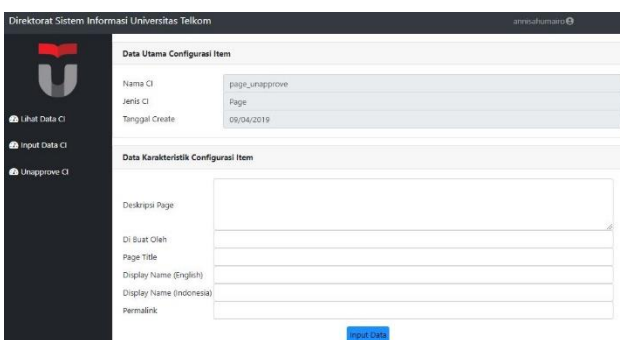
Pada pengelolaan CI selain pengguna dapat melihat data CI, pengguna juga dapat melakukan penambahan data dan perubahan data CI.



Form input for 'Data Utama Configurasi Item' with fields: Nama CI, Jenis CI (dropdown), and Tanggal Create (dd/mm/yyyy). A blue 'Input Data' button is visible.

Gambar 4 Implementasi Tambah Tahap 1

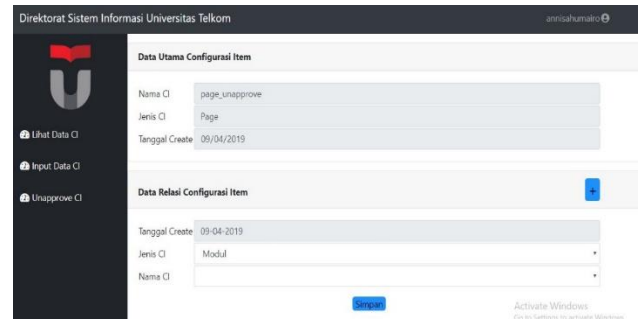
Pada fungsi penambahan CI, pengguna akan melalui tiga tahapan, gambar diatas adalah tahapan pertama yang harus di isi oleh pengguna. Pengguna di haruskan mengisi nama CI, memilih jenis CI, dan mengisi tanggal saat CI di tambahkan atau di catat di aplikasi.



Form input for 'Data Karakteristik Configurasi Item' with fields: Deskripsi Page, Di Buat Oleh, Page Title, Display Name (English), Display Name (Indonesia), and Permalink. A blue 'Input Data' button is visible.

Gambar 5 Implementasi Tambah Tahap 2

Gambar di atas adalah tahapan kedua yang harus di isi oleh pengguna, pada tahapan kedua ini pengguna diminta untuk mengisi data karakteristik yang dimiliki oleh CI. Data karakteristik CI diperlukan untuk mengetahui kegunaan CI, parameter atau *method* dan bagaimana cara menggunakan CI.

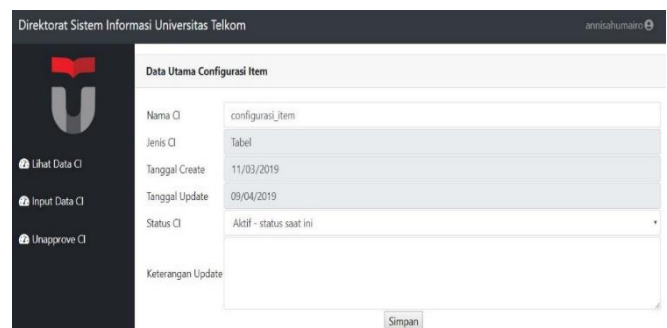


Form input for 'Data Relasi Configurasi Item' with fields: Tanggal Create, Jenis CI (dropdown), and Nama CI. A blue 'Simpan' button is visible.

Gambar 6 Implementasi Tambah Tahap 3

Gambar di atas adalah tahapan terakhir yang akan di isi oleh pengguna, pada tahapan ini pengguna dapat mengisi dan atau tidak melakukan pengisian data berdasarkan relasi yang mungkin dimiliki oleh CI yang di tambahkan.

Pada fungsi pengelolaan CI selain dapat melakukan penambahan CI, pengguna juga dapat melakukan perubahan pada data CI yang telah di tambahkan. Untuk melakukan *update* pada CI pengguna dapat melakukan *double click* pada nama CI yang kemudian akan ditampilkan halaman *tracking* data CI pada halaman tersebut pengguna dapat memilih tombol *update* untuk melakukan perubahan pada data utama CI.

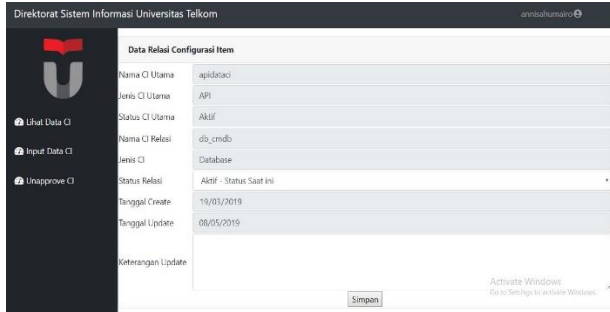


Form input for 'Data Utama Configurasi Item' showing updated data: Nama CI: configurasi_item, Jenis CI: Tabel, Tanggal Create: 11/03/2019, Tanggal Update: 09/04/2019, Status CI: Aktif - status saat ini. A grey 'Simpan' button is visible.

Gambar 7 Implementasi Update CI

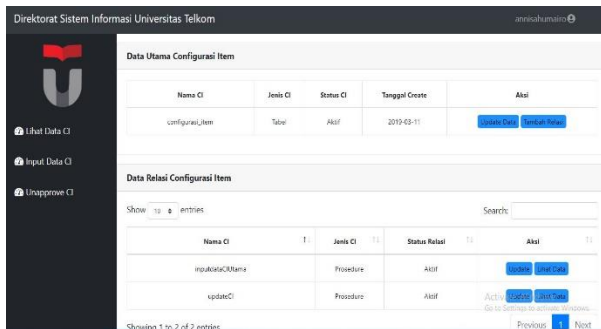
Gambar di atas merupakan implementasi untuk fungsi *update* CI yang merupakan subfungsi dari pengelolaan CI, pada fungsi di atas pengguna dapat memasukkan data CI yang di ubah. Perubahan akan di catat pada fungsi *tracking* setelah perubahan di setuju oleh atasa pengguna yang melakukan perubahan data.

Untuk melakukan update relasi pengguna dapat memilih tombol mapping pada halaman utama, lalu pada halaman mapping pengguna akan diperlihatkan data CI dan data perubahan relasi yang dimiliki oleh CI tersebut. *Update* relasi dapat dilakukan dengan menekan tombol update pada bagian data relasi CI.



Gambar 8 Halaman Update Relasi

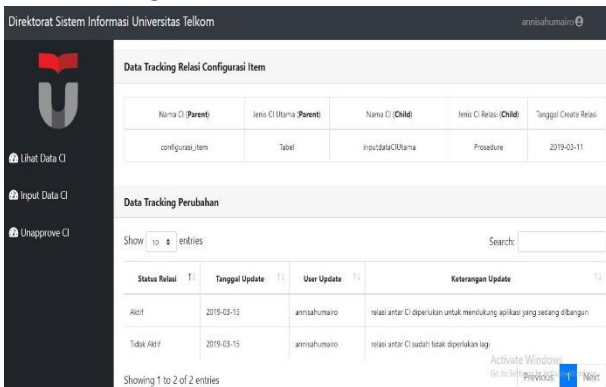
3. Pemetaan CI



Gambar 9 Implementasi Pemetaan CI

Pada fungsi ini pengguna dapat melihat relasi antar CI yang di tampilkan dalam bentuk tabel, selain melihat relasi antar CI pengguna juga dapat melihat data CI yang menjadi *parent* pada relasi serta pengguna dapat melakukan perubahan pada status relasi CI dan menambahkan relasi baru pada *parent* CI.

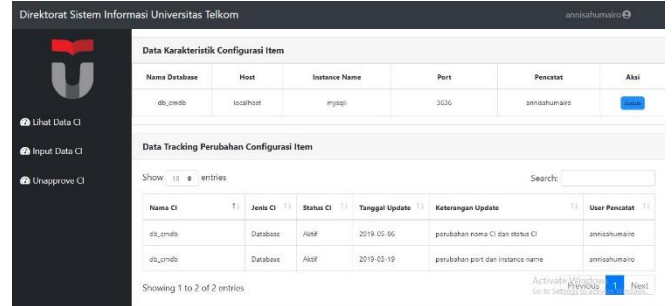
4. Tracking Relasi CI



Gambar 10 Tracking Relasi

Pada fungsi *tracking* relasi CI, pengguna dapat melihat perubahan status relasi yang diubah serta ditampilkan pula waktu perubahan, alasan perubahan, dan siapa yang melakukan perubahan data.

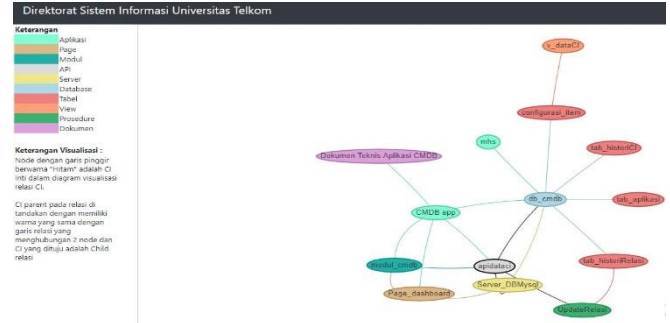
5. Tracking Histori CI



Gambar 11 Tracking Perubahan CI

Pada fungsi *tracking* perubahan CI, pengguna dapat melihat perubahan yang pernah terjadi pada CI yang dipilih. Data yang akan dilihat oleh pengguna adalah data karakteristik CI dan data perubahan CI yang berisikan data tanggal ,keterangan, dan siapa yang melakukan perubahan terhadap CI.

6. Visualisasi Pemetaan CI



Gambar 12 Visualisasi Pemetaan CI

Pada fungsi di atas pengguna akan melihat diagram relasi yang didapat dari data pemetaan CI. Pada visualisasi relasi CI ini menggunakan akan disuguhkan diagram relasi dengan *node* berbeda warna. Setiap warna pada *node* mengindikasikan jenis CI yang berelasi.

B. Hasil Pengujian

Pengujian dilakukan untuk mengetahui kesalahan yang terdapat dalam aplikasi yang di bangun. Pengujian dilakukan menggunakan metode *black box* yaitu *testing* berdasarkan kesesuaian hasil *output*

berdasarkan *input*-an data yang dimasukkan pada aplikasi.

Testing meliputi penangan karena kesalahan pemasukan data, pesan yang ditampilkan aplikasi setelah data dimasukkan, dan data yang akan di tampilkan oleh aplikasi. berikut adalah fungsionalitas yang di ujikan dengan *blackbox testing*:

Aplikasi yang dibangun telah dilakukan pengujian oleh auditor profesional dalam rangka pengujian ISO 20000-1 yang dilakukan pada Direktorat Sistem Informasi bersama dengan aplikasi dan sistem informasi yang dimiliki oleh Direktorat Sistem Informasi Universitas Telkom. Pengujian ini dilakukan guna mengetahui sejauh mana aplikasi dapat menampilkan data dan memfasilitasi kebutuhan pegawai Direktorat Sistem Informasi. Setelah dilakukan pengujian aplikasi diterima dan fungsionalitas aplikasi telah mampu memfasilitasi kegiatan pengelolaan data CI dan menampilkan tracking perubahan CI dan relasi CI.

V. KESIMPULAN

Berdasarkan hasil analisis dan rumusan masalah yang di dapat maka di bangunlah Aplikasi Pengelolaan Basis Data Konfigurasi Aplikasi, sehingga dapat mencapai penyelesaian masalah sebagai berikut:

1. Aplikasi Pengelolaan Basis Data Konfigurasi Aplikasi telah menyediakan fitur yang dapat memfasilitasi pegawai Ditsisfo untuk melakukan pengelolaan terhadap item konfigurasi aplikasi.
2. Aplikasi Pengelolaan Basis Data Konfigurasi Aplikasi telah menyediakan fitur yang dapat memfasilitasi pegawai Ditsisfo untuk melakukan pelacakan atau *tracking* perubahan item konfigurasi dan pelacakan atau *tracking* perubahan status relasi item konfigurasi.

VI. PENGHARGAAN

Ucapan terima kasih penulis sampaikan kepada berbagai pihak yang telah memberikan kontribusi selama proses penelitian ini baik dalam hal fasilitas maupun ilmu yang bermanfaat, terutamanya kepada:

1. Bapak Dahliar Ananda selaku dosen pembimbing I yang telah memberikan bantuan, petunjuk dan bimbingan kepada penulis selama pembuatan proyek akhir.
2. Ibu Siska Komala selaku dosen pembimbing II yang telah memberikan ilmunya dalam membimbing penulis dalam mengerjakan proyek akhir ini.

VII. REFERENCES

[1]	Direktorat Sistem Informasi , "Rencana Strategis Sistem Informasi Universitas Telkom 2014-2018," Universitas Telkom, Bandung, 2014.
[2]	I. Sommerville, Software Engineering 9th Edition, Addison-Wesley, 2009.
[3]	AXELOS Limited, "Process Map & ITIL," 3 Agustus 2013. [Online]. Available: https://wiki.de.it-processmaps.com/index.php/Checkliste_Configuration_Item_-_CI_-_Record . [Accessed 17 Januari 2019].
[4]	R. A. Sukamto and M. Shalahuddin, Rekayasa Perangkat Lunak : Terstruktur dan berorientasi objek, Bandung: Informatika, 2014.
[5]	A. P. Basuki, Membangun Web Berbasis PHP dengan Framework CodeIgniter, PT. Lokomedia, 2010.
[6]	A. Kadir, Pemrograman Database MySQL, Yogyakarta: Media Kom, 2013.
[7]	K. Tatroe, P. MacIntyre and R. Lerdorf, Programming PHP, Beijing: O'Reilly, 2013.
[8]	E. Winarno, A. Zaki and S. Community, Pemrograman Web Berbasis Html 5,php, dan Javascript, Jakarta: Elex Media Komputindo, 2014.

VIII. LAMPIRAN

