

Rancang Bangun Portal Beasiswa Berbasis Web Menggunakan Metode Jaro-Winkler

Audhy Virabri Kressa¹⁾ Pantjawati Sudarmaningtyas²⁾ Vivine Nurcahyawati³⁾

Program Studi/Jurusan Sistem Informasi
Institut Bisnis dan Informatika Stikom Surabaya
Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email : 1) audhy.vk01@gmail.com, 2) pantja@stikom.edu, 3) yivine@stikom.edu

Abstract: Scholarships are benefits that aim to help with learning costs. Scholarship information is still spread on each web, making it difficult to search. This study aims to build a web portal that focuses scholarship information. The web portal built automatically collects periodic scholarship data using Curl. The source of information used comes from existing portals and scholarship provider web agencies. After the data is obtained the system will match with the Jaro-Winkler method to avoid duplication of stored data. Based on the results of unit testing on the function results obtained 98.91%. These results indicate that this study has succeeded in implementing up-to-date centralization of information with the Jaro-Winkler method.

Keywords: Web Portal, Jaro-Winkler, Scholarship, Curl

Beasiswa adalah tunjangan yang bertujuan untuk membantu biaya belajar. Bantuan yang dimaksud dapat berupa berbagai jenis bantuan antara lain bantuan uang tunai dan jaminan belajar. Pengumuman beasiswa-pun saat ini hanya dapat diakses melalui surat kabar, mading sekolah, dan *website* tertentu saja, sehingga sosialisasi serta pengumuman kurang merata. Hal ini mengakibatkan info beasiswa tidak diterima masyarakat tepat waktu. Selain itu seringkali pelamar beasiswa sulit mendapatkan beasiswa yang diinginkan berdasarkan kebutuhannya. Pelamar harus mencari banyak informasi mengenai beasiswa hingga mendapatkan beasiswa yang cocok untuknya.

Solusi berdasarkan permasalahan di atas dengan membuat suatu wadah pemusatan informasi beasiswa pada satu wadah. Dengan pemusatan data pelamar mudah mengakses dan tidak perlu mencari info di berbagai tempat. Pemusatan tersebut berupa *web portal*.

Informasi pada web portal bersifat *up-to-date* dengan melakukan *update* data setiap harinya. Metode yang digunakan untuk pengambilan data adalah Curl dan algoritma Jaro-Winkler sebagai algoritma matching data yang dimiliki dengan data yang baru. Sumber informasi yang digunakan berasal dari portal yang sudah ada dan web instansi penyedia beasiswa.

Menurut Stenberg (2015) Curl adalah suatu program dan *library* berbasis *opensource* yang dapat melakukan pemindaian data dari atau ke sebuah server tanpa adanya interaksi dengan

user dari server. Pemindaian data yang dilakukan dapat berupa *grab* atau pengambilan konten dari suatu website. Curl melakukan pengiriman dan pengambilan data melalui URL sehingga dapat digunakan di berbagai Bahasa pemrograman.

Menurut Kurniawati, Puspitodjati & Rahman (2014) pada jurnalnya Jaro-Winkler Distance merupakan algoritma yang merupakan salah satu varian dari Jaro distance metrik yang berguna untuk mengukur kesamaan dari dua *string*. Semakin tinggi nilai dari Jaro-Winkler maka semakin mirip kedua *string* tersebut. Nilai dari Jaro-Winkler adalah 0 sampai 1.

Rumusan dari Jaro-Winkler:

$$d_w = d_j + (l p (1 - d_j)) \quad (1)$$

Keterangan:

- d_j = Jaro Distance untuk *string* s1 dan s2
- l = Panjang *prefix* umum di awal *string*
- p = Lonstanta *scaling factor*

Sedangkan menurut Cohen, Ravikumar & Fienberg (2003) untuk mencari Jaro-Distance memiliki rumusan :

$$d_j = 1/3 (m_1/s_1 + m_2/s_2 + ((m_1 - (t/2))/m_1)) \quad (2)$$

Keterangan:

- s_1 = Panjang *string* 1
- s_2 = Panjang *string* 2
- m_1 = Jumlah karakter yang sama antara s_1 dan s_2

- m_2 = Jumlah karakter yang sama antara s_2 dan s_1
- t = Jumlah transposisi

Menurut Doan, Halevy, & Ives (2012) untuk menentukan m_1 dan m_2 juga mempertimbangkan posisi dari karakter yaitu dengan rumusan :

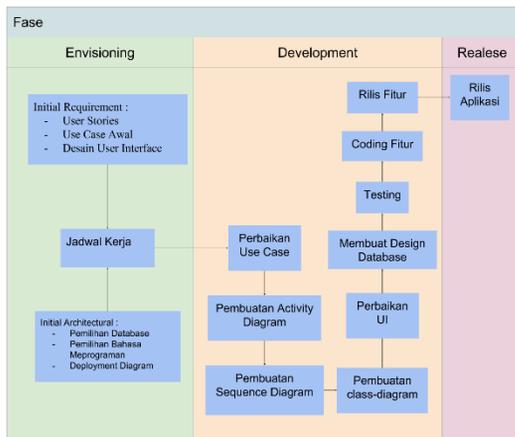
$$[\min(|s_1|, |s_2|/2)] \quad (3)$$

Harapan dari adanya Portal Beasiswa ini adalah pencari beasiswa dapat dengan mudah mendapatkan informasi terbaru beasiswa dalam satu wadah. Sehingga tidak ada lagi kesulitan dan keterlambatan dalam mendapatkan informasi beasiswa yang diminati.

METODE

Metode pengembangan yang digunakan adalah Agile. Menurut Ambler (2012), *Agile Model-Driven Development* (AMDD) merupakan *Model-Driven Development* (MDD) versi Agile. MDD sendiri adalah sebuah pendekatan dalam pengembangan perangkat lunak dengan cara memodelkan secara menyeluruh sebelum memulai menulis kode. Akan tetapi AMDD tidak memodelkan secara menyeluruh dan membuat model agar segera dieksekusi.

Tahapan agile dapat dilihat pada gambar 1.



Gambar 1 Tahapan Agile

Envisioning

Fase *envisioning* mengidentifikasi ruang lingkup sistem yang dibuat dan idendifikasi arsitektur yang dibutuhkan. Tujuan dilakukannya fase ini adalah untuk mengeksplorasi persyaratan dan strategi menyeluruh terhadap sistem yang akan dibuat.

Initial Requirement

Pada tahap ini dilakukan analisis kebutuhan sistem Portal Beasiswa.

Initial Architectural

Pada tahapan ini dilakukan pembuatan desain arsitektur sistem yang akan digunakan sebagai arsitektur portal beasiswa.

Development

Tahap *development* merupakan gabungan dari fase *Iteration Modeling*, *Model Storming* dan *Executable Specification*. Ketiga fase tersebut digabung dikarenakan akan ada proses berulang ke tiga fase tersebut pada iterasi.

Perbaikan Use Case

Perbaikan user case adalah tahapan perbaikan jika use case yang dibuat diawal kurang detil atau terjadi perubahan.

Pembuatan Activity Diagram

Pembuatan activity diagram adalah proses penggambaran aktifitas yang dilakukan pada setiap use case. Activity diagram juga dapat disebut sebagai turunan dari use case.

Pembuatan Sequence Diagram

Pembuatan Sequence diagram adalah tahapan sequence diagram dari sistem dibuat yang menggambarkan interaksi antar objek dalam bentuk sequensial.

Pembuatan Class Diagram

Pembuatan class diagram adalah pembuatan class diagram yang menampilkan relasi antar class, attribute beserta operasi yang dilakukan class tersebut.

Perbaikan UI

Perbaikan UI adalah tahap dimana perbaikan design UI awal dikarenakan ketidak sesuaian dengan kebutuhan yang ada.

Design Database

Design Database adalah tahapan membuat design database beserta tabel dan relasinya.

Testing

Testing adalah tahapan membuat skema yang digunakan untuk dasar coding fitur.

Coding Fitur

Coding Fitur adalah proses coding dilakukan yang dilandasi tahapan-tahapan sebelumnya.

Rilis Fitur

Rilis Fitur adalah tahapan fitur yang dikerjakan pada iterasi dirilis k kalangan tertentu.

Release

Tahap ini merupakan tahapan aplikasi Beasiswa telah dirilis secara keseluruhan.

HASIL DAN PEMBAHASAN

Pada tahapan hasil ini menerangkan tahapan dan hasil tahapan dari Portal Beasiswa.

Envisioning

Initial Requirement

Analisis dilakukan dengan dua acara yaitu kuesioner dan perbandingan sistem yang sudah ada. Kuesioner yang disebar diolah dan menghasilkan persentase kebutuhan informasi seperti tabel 1

Tabel 1 Kebutuhan Informasi

Informasi	Jumlah
Cara mendapatkan beasiswa	17.14%
Syarat dalam mengajukan beasiswa	57.14%
Rentang waktu penerimaan beasiswa	11.43%
Keterikatan beasiswa	5.71%
Fasilitas yg didapat dari beasiswa	51.43%
Tanggal dibukanya	14.28%
Transparansi proses	2.86%
Testimoni/sharing pengalaman dari yg mendapatkan beasiswa	2.86%
Tips agar masuk	5.71%
Informasi SKTM (Jenis Beasiswa)	8.57%
Sistem Penerimaan (Alur)	2.86%
Tentang prospek selanjutnya	2.86%

Untuk mempersiapkan dan mengatur jalannya sistem, terdapat beberapa aktifitas yang dilakukan dalam suatu user stories yang dibuat berdasarkan kebutuhan informasi pengguna. *User stories* dari pengguna dan administrator lebih detil dapat dilihat pada tabel 2 dan tabel 3.

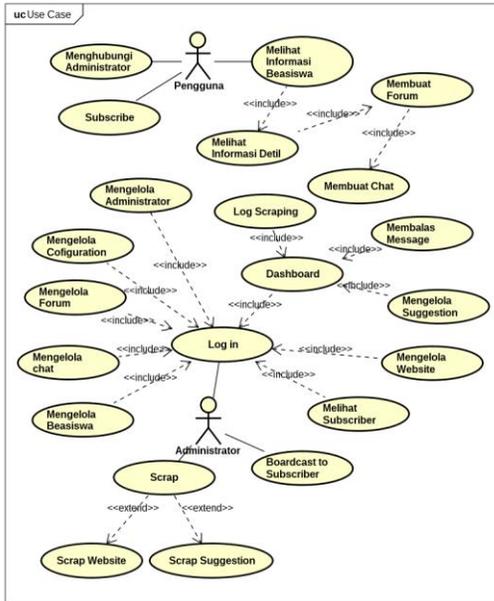
Tabel 2 User Stories Pengguna

ID	Story
P1	Dapat melihat daftar beasiswa
P2	Dapat membuka data beasiswa secara detil
P3	Dapat membuat forum diskusi
P4	Dapat membuat chat di forum
P5	Dapat subscribe informasi terbaru
P6	Dapat menghubungi admin

Tabel 3 User Stories Administrator

ID	Story
A1	Halaman backend halaman backend hanya dapat diakses oleh Administrator dengan data yang ada pada <i>configuration</i>
A2	Dapat melihat dan membuat data administrator lain.
A3	Dapat melihat, membuat, mengedit, dan menghapus website yang akan discrap.
A4	Dapat melihat, membuat, mengedit, dan menghapus data attribute
A5	Dapat melihat, membuat, mengedit, dan menghapus data beasiswa
A6	Dapat membalas pesan yang dikirim
A7	Dapat melihat, menambahkan website yang suggest atau mengabaikannya.
A8	Dapat melihat daftar subscriber
A9	Dapat mengatur configuration web portal beasiswa
A10	Dapat melihat log scraping
A11	Dapat melakukan scraping suggestion
A12	Dapat melakukan scraping website
A13	Dapat menerima dan mengabaikan suggestion
A14	Dapat melihat dan menghapus forum
A15	Dapat melihat dan menghapus chat pada forum
A16	Broadcast ke subscriber

Dari user stories yang ada maka dibuatlah use case yang menggambarkan pemetaan actor dan fitur portal beasiswa. Use case diagram dapat dilihat pada gambar 2.



Gambar 2 Use Case Diagram

Setelah use case dibuat maka proses selanjutnya adalah design UI untuk setiap use case yang ada.

Initial Architectural

Tahap pembuatan arsitektur dilakukan berdasarkan hasil dari analisis kebutuhan yang telah dilakukan. Arsitektur yang dipilih adalah :

1. Database : Postgresql
2. Bahasa Pemrograman : PHP
3. Sistem Operasi : Ubuntu 18.04
4. Web Server : Nginx

arsitektur dipilih berdasarkan nilai benchmark dengan produk lain.

Development

Fase *development* adalah dimana iterasi dilakukan. Iterasi merupakan serangkaian perulangan kegiatan terjadwal yang dilakukan untuk mencapai *goal* yang direncanakan agar tujuan akhir proyek tercapai. Jumlah dan Panjang dari iterasi ditentukan oleh kemampuan penulis mencapai setiap *goal*. Untuk menentukan *goal* sebelumnya harus menentukan *minimum viable*. *Minimum viable* merupakan kebutuhan minimum pengguna yang harus dicapai. Dalam fase *development* dibagi 2 *goal* yang harus dicapai. *Goal* dibuat berdasarkan user stories dan *minimum viable*. Dua *goal* tersebut adalah

- a) Pengguna mendapatkan informasi beasiswa
- b) Pengguna dapat berkomunikasi

Dari dua *goal* diatas maka dibuatlah jadwal iterasi yang berdasarkan oleh *goal*, use case dan prioritas yang ditentukan penulis agar

terselesainya Portal Beasiswa. Nilai prioritas tidak memiliki ketetapan. Pada jadwal yang akan dibuat penulis memberikan nilai terendah 1 dan 30 sebagai nilai tertinggi. Berikut merupakan jadwal pada table 4.

Tabel 4 Jadwal Iterasi

Goal / Iterasi	User Story	Use Case	Prioritas
1	A3	Mengelola Website	30
	A4	Mengelola Attribute	29
	A12	Scraping Website	28
	A5	Mengelola Beasiswa	27
	P1	Melihat Informasi Beasiswa	26
	P2	Melihat Informasi Detil	25
	A2	Mengelola Adminstrator	24
	A1	Login	23
2	A11	Scraping Suggestion	22
	A13	Mengelola Suggestion	21
	A10	Melihat Log Scraping	20
	P3	Membuat Forum	19
	P4	Membuat Chat	18
	A14	Mengelola Forum	17
	A15	Mengelola Chat	16
	P5	Subscribe	15
	A8	Melihat Subscriber	14
	A16	Broadcast to Subscriber	13
	P6	Menghubungi Administrator	12
	A6	Membalas Pesan	11
	A9	Mengelola Configuration	10

Perbaikan Use Case

Pada tahap ini tidak ada perubahan use case dikarenakan use case sudah

Pembuatan Activity Diagram

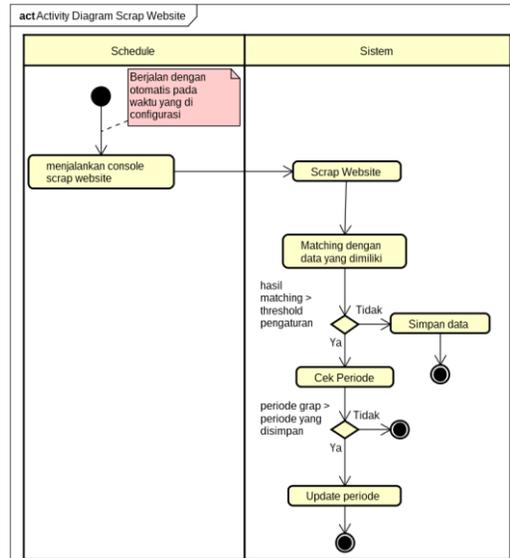
Activiy diagram terdiri dari:

Tabel 5 Daftar Activity Diagram

Iteasi	Activity
Iterasi 1	Melihat Website
	Menambah Website
	Memperbarui Website

Iteasi	Activity
	Menghapus Website
	Melihat Attribute
	Menambah Attribute
	Memperbarui Attribute
	Menghapus Attribute
	Scraping Website
	Melihat Beasiswa
	Menambahkan Beasiswa
	Memperbarui Beasiswa
	Menghapus Beasiswa
	Melihat Informasi Beasiswa
	Melihat Informasi Detil
	Melihat Administrator
	Menambahkan Administrator
	Login
Iterasi 2	Scraping Suggestion
	Melihat Suggestion
	Mengabaikan Suggestion
	Menyetujui Suggestion
	Melihat Log
	Membuat Forum
	Membuat Chat
	Melihat Forum
	Menghapus Forum
	Melihat Chat
	Menghapus Chat
	Subscribe
	Melihat Subscriber
	Broadcast
	Menghubungi Administrator
	Membalas Pesan
	Melihat Configuration
	Memperbarui Configuration

Contoh Activity Diagram terdapat pada pada gambar 3. Activity tersebut menerangkan bahwa activity tersebut berjalan otomatis sesuai dengan jadwal yang telah di konfigurasi. Aktifitas yang dilakukan adalah sistem melakukan scrap website kemudian melakukan matching data yang dimiliki dengan data hasil scrap.



Gambar 3 Activity Diagram Scrap Website

Data sample untuk matching seperti pada tabel 6.

Tabel 6 Sample perhitungan matching

Data Scrapping	Data yang dimiliki
lpdp.kemenkeu.go.id/	Indbeasiswa.com/

Matching akan dilakukan menggunakan metode Jaro-Winkler. Perhitungan Jaro Winkler sebagai berikut:

$$d_j = 1/3 (11/20 + 7/16 + ((11-(6/2))/11))$$

$$d_j = 0.57159$$

$$d_w = 0.57159 + (0 \times 0.1 (1 - 0.57159))$$

$$d_w = 0.697$$

Hasil matching kemudian dibandingkan dengan *threshold*. Jika kurang maka data akan langsung disimpan jika lebih maka akan dilakukan perbandingan periode beasiswa yang ada. Jika periode scrap lebih baru maka dilakukan pembaruan data.

Pembuatan Sequence Diagram

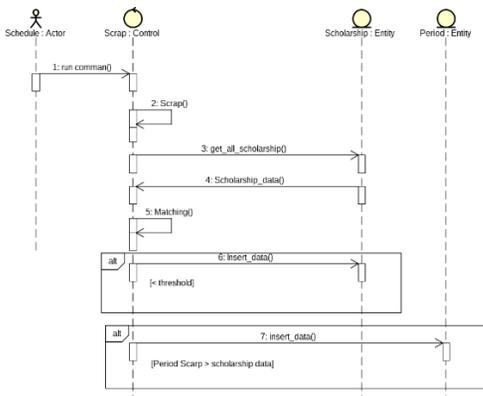
Sequence diagram Portal Beasiswa terdiri dari:

Tabel 7 Sequence Diagram

Iterasi	Sequence Diagram
Iterasi 1	Memperbarui Website
	Menghapus Website
	Melihat Attribute
	Menambah Attribute
	Memperbarui Attribute
	Menghapus Attribute
	Scraping Website
	Melihat Beasiswa
	Menambahkan Beasiswa
	Memperbarui Beasiswa
	Menghapus Beasiswa

Iterasi	Sequence Diagram
	Melihat Informasi Beasiswa
	Melihat Informasi Detil
	Melihat Administrator
	Menambahkan Administrator
	Login
Iterasi 2	Scraping Suggestion
	Melihat Suggestion
	Mengabaikan Suggestion
	Menyetujui Suggestion
	Melihat Log
	Membuat Forum
	Membuat Chat
	Melihat Forum
	Menghapus Forum
	Melihat Chat
	Menghapus Chat
	Subscribe
	Melihat Subscriber
	Broadcast
	Menghubungi Administrator
	Membalas Pesan
	Melihat Configuration
Memperbarui Configuration	

Contoh dari Diagram Sequence dapat dilihat pada gambar 4



Gambar 4 Sequence Diagram Scrap Website

Pembuatan Class Diagram

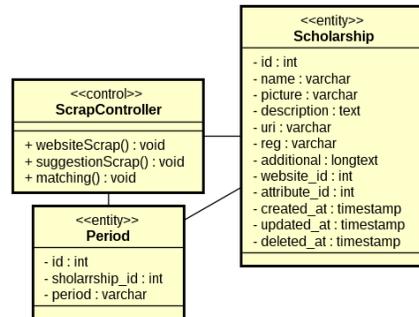
Berikut merupakan class diagram portal beasiswa:

Tabel 8 Daftar Class Diagram

Iterasi	Class Diagram
Iterasi 1	Mengelola Website
	Mengelola Attribute
	Scraping Website
	Mengelola Beasiswa
	Melihat Informasi Beasiswa
	Melihat Detil Informasi

Iterasi	Class Diagram
Iterasi 2	Mengelola Administration
	Login
	Scraping Suggestion
	Mengelola Suggestion
	Melihat Log
	Membuat Forum
	Membuat Chat
	Mengelola Forum
	Mengelola Chat
	Melakukan Subscribe
	Melihat Subscriber
	Broadcast
	Menghubungi Adminstrator
	Membalas Pesan
	Mengelola Configuration

Contoh class diagram dapat dilihat pada gambar 5



Gambar 5 Class Diagram Scrap Website

Perbaikan UI

Perbaikan UI tidak dilakukan dikarenakan rancangan awal sudah memenuhi kebutuhan.

Design Database

Berikut merupakan daftar tabel yang dibuat:

Tabel 9 Daftar Tabel Database

Iterasi	Tabel
Iterasi 1	users
	websites
	scholarship_attribute
	scholarship
	periods
Iterasi 2	logs
	forums
	chats
	messages
	subscribers
	configurations

Testing

Schema testing yang dibuat dapat dilihat pada tabel 9

Tabel 9 Daftar Skenario Testing

Iterasi	Testing
Iterasi 1	Melihat Daftar Website
	Menambahkan Website
	Menambahkan Website dengan data lengkap
	Menambahkan Website dengan data tidak lengkap
	Memperbarui Website dengan data lengkap
	Memperbarui Website dengan data tidak lengkap
	Menghapus Website
	Melihat Daftar Attribute
	Menambahkan Attribute dengan data lengkap
	Menambahkan Attribute dengan data tidak lengkap
	Memperbarui Attribute
	Memperbarui Attribute dengan data lengkap
	Menghapus Attribute dengan data tidak lengkap
	Scraping Website
	Melihat Informasi Beasiswa
	Melihat detil beasiswa
	Melihat Daftar Administrator
	Menambahkan Administrator dengan data lengkap
	Menambahkan Administrator dengan data tidak lengkap
	Login email dan password benar
	Login email dan password salah
	Login email dan password kosong
	Masuk tanpa Login
Iterasi 2	Scraping Suggestion
	Melihat Daftar Suggestion
	Menyetujui Suggestion
	Megabaikan Suggestion
	Melihat Log Srap
	Membuat Forum dengan data lengkap
	Membuat Forum dengan data tidak lengkap
	Membuat Chat dengan data lengkap
Membuat Chat dengan data	

Iterasi	Testing
	tidak lengkap
	Melihat daftar Forum
	Melihat Chat Pada Forum
	Melakukan Subscribe
	Melihat Data Subscribe
	Broadcast Informasi Beasiswa
	Membalas Pesan dengan data lengkap
	Membalas Pesan dengan data tidak lengkap
	Memperbarui configuration dengan data lengkap
	Memperbarui configuration dengan data tidak lengkap

Coding Fitur

Tahap ini proses coding portal beasiswa dilakukan. Coding meliputi pembuatan migration, model, controller dan view.

Rilis Fitur

Pada tahap ini dilakukan rilis fitur pada kalangan tertentu yaitu lingkungan developer untuk digunakan pengujian kelayakan Portal Beasiswa.

Release

Fitur yang telah direncanakan telah terbuat, sehingga dapat digunakan sepenuhnya. Berikut adalah beberapa gambar implementasi sistem:

Implementasi Skenario Testing

Tabel 10 Implementasi Skenario Testing Iterasi 1

No	Tes	pass	fail	%
1.	Melihat Daftar Website	5	0	100%
2.	Menambahkan Website	5	0	100%
3.	Menambahkan Website dengan data lengkap	5	0	100%
4.	Menambahkan Website dengan data tidak lengkap	5	0	100%
3.	Memperbarui Website dengan data lengkap	5	0	100%
4.	Memperbarui Website	5	0	100%

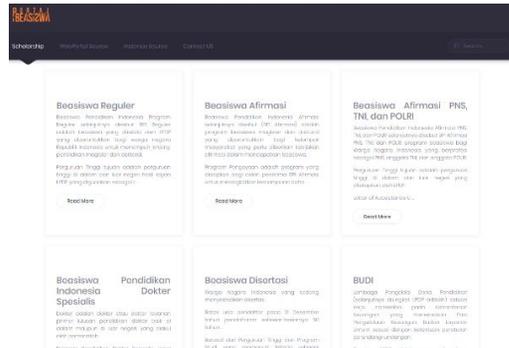
No	Tes	<i>pass</i>	<i>fail</i>	%
	dengan data tidak lengkap			
5.	Menghapus Website	5	0	100%
6.	Melihat Daftar Attribute	5	0	100%
7.	Menambahkan Attribute dengan data lengkap	5	0	100%
8.	Menambahkan Attribute dengan data tidak lengkap	5	0	100%
9.	Memperbarui Attribute	5	0	100%
10.	Memperbarui Attribute dengan data lengkap	5	0	100%
11.	Menghapus Attribute dengan data tidak lengkap	5	0	100%
12.	Scraping Website	4	1	80%
13.	Melihat Informasi Beasiswa	5	0	100%
14.	Melihat detail beasiswa	5	0	100%
15.	Melihat Daftar Administrator	5	0	100%
16.	Menambahkan Administrator dengan data lengkap	5	0	100%
17.	Menambahkan Administrator dengan data tidak lengkap	5	0	100%
18.	Login email dan password benar	5	0	100%
19.	Login email dan password salah	5	0	100%
20.	Login email dan password kosong	5	0	100%
21.	Masuk tanpa Login	5	0	100%

Tabel 11 Implementasi Skenario Testing Iterasi 2

No	Tes	<i>pass</i>	<i>fail</i>	%
1.	Scraping Suggestion	4	1	800%
2.	Melihat Daftar Suggestion	5	0	100%
3.	Menyetujui Suggestion	5	0	100%
4.	Mengabaikan Suggestion	5	0	100%
3.	Melihat Log Srap	5	0	100%
4.	Membuat Forum dengan data lengkap	5	0	100%
5.	Membuat Forum dengan data tidak lengkap	5	0	100%
6.	Membuat Chat dengan data lengkap	5	0	100%
7.	Membuat Chat dengan data tidak lengkap	5	0	100%
8.	Melihat daftar Forum	5	0	100%
9.	Melihat Chat Pada Forum	5	0	100%
10.	Melakukan Subcribe	5	0	100%
11.	Melihat Data Subscribe	5	0	100%
12.	Broadcast Informasi Beasiswa	5	0	100%
13.	Membalas Pesan dengan data lengkap	5	0	100%
14.	Membalas Pesan dengan data tidak lengkap	5	0	100%

No	Tes	pass	fail	%
15.	Memperbarui configuration dengan data lengkap	5	0	100%
16.	Memperbarui configuration dengan data tidak lengkap	5	0	100%

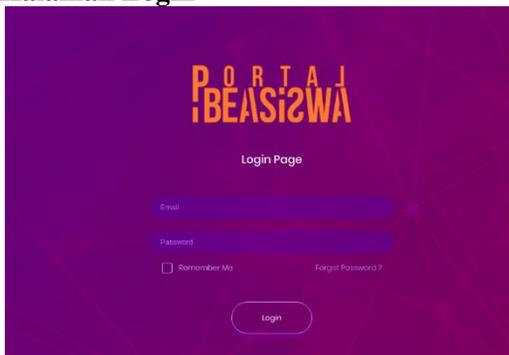
Halaman Utama Web Portal



Gambar 6 Halaman Utama

Halaman utama adalah halaman pertama yang diakses oleh user. Tampilan halaman tersebut berisikan informasi umum beasiswa yang berupa nama dan deskripsi singkat.

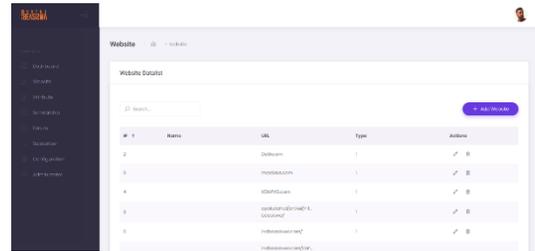
Halaman Login



Gambar 7 Halaman Login

Halaman Login merupakan halaman yang digunakan administrator untuk masuk ke sistem portal beasiswa. UI dari halaman login dapat dilihat pada gambar 7

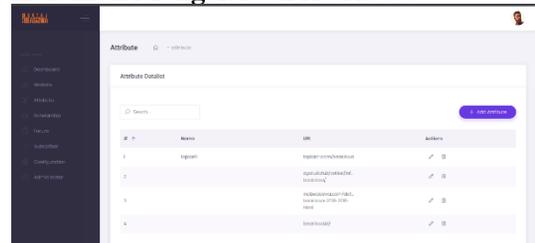
Halaman Mengelola Website



Gambar 8 Halaman Pengelolaan Website

Gambar 8 adalah halaman pengelolaan website merupakan halaman dimana administrator mengelola data website yang digunakan. Pada halaman ini administrator dapat melakukan *create, read, update* dan *delete* data website.

Halaman Mengelola Attribute

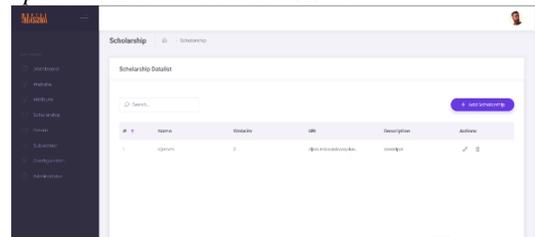


Gambar 9 Halaman Pengelolaan Attribute

Halaman pengelolaan attribute merupakan halaman dimana administrator mengelola data attribute yang digunakan untuk grup data beasiswa. Pada halaman ini administrator dapat melakukan *create, read, update* dan *delete* data attribute.

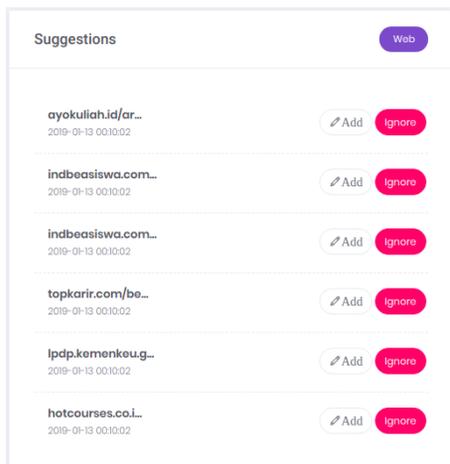
Halaman Mengelola Beasiswa

Halaman pengelolaan beasiswa merupakan halaman dimana administrator mengelola data beasiswa. Pada halaman ini administrator dapat melakukan *create, read, update* dan *delete* data beasiswa.



Gambar 10 Halaman Pengelolaan Beasiswa

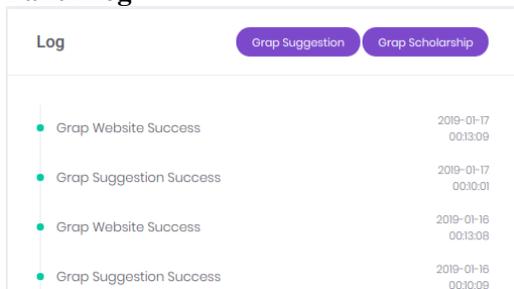
Panel Suggestion



Gambar 11 Panel Suggestion

Gambar 11 merupakan gambar dari panel suggestions yang berfungsi untuk pengelolaan suggestion. Panel suggestions terdapat pada halaman dashboard administrator. Administrator dapat menerima suggestion atau mengabaikannya.

Panel Log

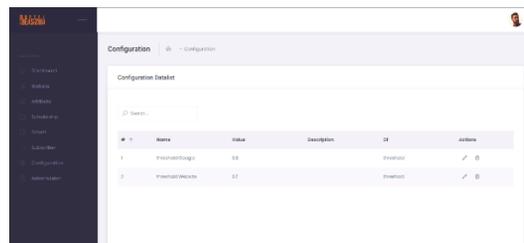


Gambar 12 Panel Log

Panel Log merupakan panel yang terdapat pada dashboard yang menampilkan log aktivitas dari Portal Basiswa. Pada panel log berisi log website, log wuggetion dan log broadcast informasi basiswa terbaru.

Halaman Konfigurasi

Halaman pengelolaan *configuration* merupakan halaman dimana administrator mengelola pengaturan pada system portal basiswa. Pada halaman ini administrator hanya dapat melakukan *read* dan *update* data saja.



Gambar 13

SIMPULAN

Berdasarkan hasil uji coba yang telah dilakukan 98,91% uji coba berhasil maka telah dihasilkan sebuah aplikasi Portal Beasiswa Berbasis Web Menggunakan Metode Jaro-Winkler yang dapat berjalan sesuai fungsinya.

RUJUKAN

- Doan, A., Halevy, A., & Ives, Z. (2012). *Principles of Data*. Waltham: Elsevier.
- Ambler, S. (2012). *Model Storming: An Agile Best Practice*. Retrieved from Agile Modeling: <http://agilemodeling.com/essays/modelStorming.htm>
- Cohen, W. W., Ravikumar, P., & Fienberg, S. E. (2003). A Comparison of String Metrics for Matching Names and Records. *American Association for Artificial Intelligence*.
- Kornain, A., Yansen, F., & Tinaliah. (2014). Penerapan Algoritma Jaro-Winkler Distance untuk sistem pendeteksi plagiarisme pada dokumen teks berbahasa Indonesia. *STMIK MDP*.
- Kurniawati, A., Puspitodjati, S., & Rahman, S. (2014). Implementasi Algoritma Jaro-Winkler Distance untuk Membandingkan Kesamaan Dokumen Berbahasa Indonesia. *Skripsi Program Studi Sistem Informasi*.
- Stenberg, D. (2015, September). *Everything-curl*. Retrieved from [curl: https://curl.haxx.se/](https://curl.haxx.se/)