

**SISTEM REKOMENDASI INFORMASI LOWONGAN PEKERJAAN  
DENGAN METODE COSINE SIMILARITY & RABIN KARP K-GRAM**

**LAPORAN TUGAS AHIR**

Laporan ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana  
Strata 1 (S1) Program Studi Teknik Informatika Fakultas Teknologi Industri  
Universitas Islam Sultan Agung Semarang



**DISUSUN OLEH :**

**BIMA CANDRA KHARISMA**

**NIM 32601400805**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS ISLAM SULTAN AGUNG**

**SEMARANG JULI 2021**

**INFORMATION RECOMMENDATION SYSTEM FOR JOBS WITH  
COSINE SIMILARITY & RABIN KARP K-GRAM METODE**

*Proposed to complete the requirement to obtain a bachelor's degree (S-1) at  
Informatics Engineering Departement of Industrial Technology faculty Sultan  
Agung Islamic University*



*Arranged by :*

**BIMA CANDRA KHARISMA**

**NIM 32601400805**

***MAJORING OF INFORMATICS ENGINEERING***

***INDUSTRIAL TECHNOLOGY FACULTY***

***SULTAN AGUNG ISLAMIC UNIVERSITY***

***SEMARANG JULY 2021***

## LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul “**Sistem Rekomendasi Informasi Lowonga Pekerjaan Dengan Metode Cosine Similarity & Rabin Karp K-Gram**” ini disusun oleh :

Nama : Bima Candra Kharisma

NIM 32601400805

Program Studi : Teknik Informatika

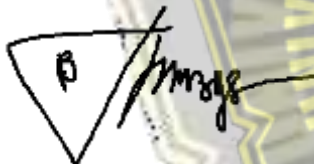
Telah disahkan oleh dosen pembimbing pada :

Hari : .....

Tanggal : .....

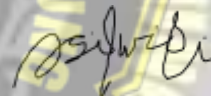
Mengesahkan,

Pembimbing I



Bagus Satrio WP, S.Kom M.Cs  
NIDN.1027118801

Pembimbing II



Asih Widiharini, S.Si , M.T  
NIDN.1617087002

Mengetahui,

Ketua Program Studi Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Sultan Agung



Andi Riansyah, ST, M.Kom  
NIDN. 0609108802

## LEMBAR PENGESAHAN PENGUJI

Laporan tugas akhir dengan judul “**Sistem Rekomendasi Informasi Lowogan Pekerjaan Dengan Metode Cosine Similarity & Rabin Karp K-Gram**” ini telah dipertahankan di depan dosen penguji Tugas Akhir pada :

Hari : Rabu.....

Tanggal : 18 Agustus 2021.....

### TIM PENGUJI

Anggota I

  
Sam Farisca C. ST,M.Kom

NIDN. 0628028602

Anggota II

  
Dedy Kurniadi. ST,M.Kom

NIDN.1628028602

  
Ketua Penguji

Imam Much Ibnu S,ST,M.Sc,Ph.D

NIDN.1613037301

### SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Bima Candra Kharisma

NIM : 32601400805

Judul Tugas Akhir : 32601400805

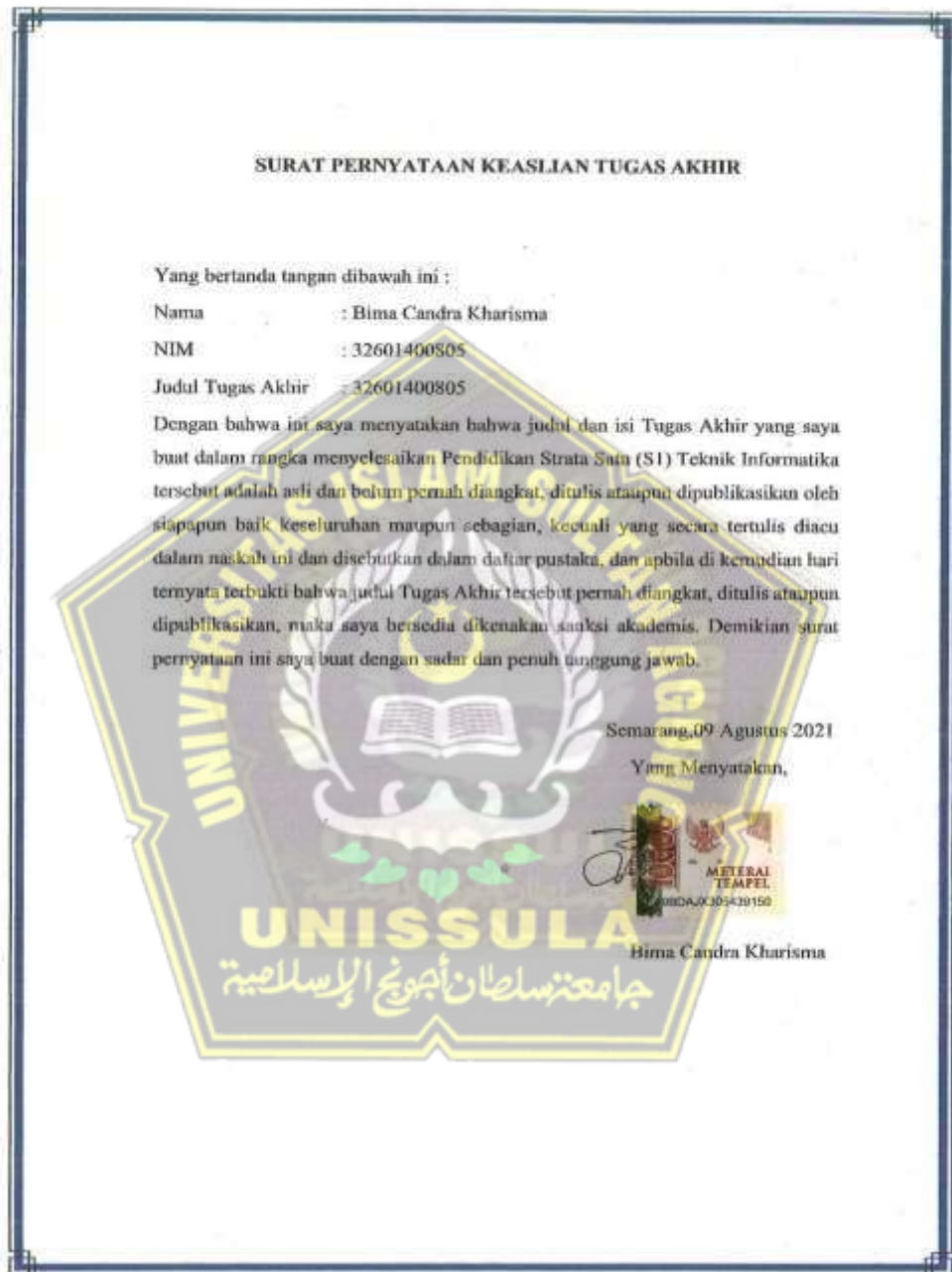
Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 09 Agustus 2021

Yang Menyatakan,



Bima Candra Kharisma



### PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Bima Candra Kharisma

NIM : 32601400805

Program Studi : Teknik Informatika

Fakultas : Teknologi industri

Alamat Asal : Ds.Kenangkan RT 02 RW 07 Bergas Kidul , Kab.Semarang

Dengan ini menyatakan Karya ilmiah berupa Tugas akhir dengan Judul : **SISTEM REKOMENDASI INFORMASI LOWONGAN PEKERJAAN DENGAN METODE COSINE SIMILARITY & RABIN KARP K-GRAM**

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmedikan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantunkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung

Semarang, 09 Agustus  
2021

Yang menyatakan

  
METERAI  
TEMPEL  
08042021083548  
UNIVERSITAS ISLAM SULTAN AGUNG

## KATA PENGANTAR

Puji syukur senantiasa penulis panjatkan atas kehadiran Allah SWT yang telah memberikan rahmat dan inayah-Nya, sehingga dapat menyelesaikan Tugas Akhir dengan judul “Sistem Rekomendasi Informasi Lowongan Pekerjaan dengan Metode Cosine Similarity & Rabin Karp K-Gram” ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar sarjana (S-1).

Tugas Akhir ini disusun dan dibuat dengan adanya bantuan dari berbagai pihak, materi maupun teknis, oleh karena itu saya selaku penulis mengucapkan terima kasih kepada :

1. Kedua orang tua yang selalu memberikan motivasi, semangat, doa serta dukungan kepada penulis.
2. Bapak Bagus Satrio WP, S.Kom, M.Cs selaku Dosen Pembimbing I dan Ibu Asih Widiharini, S.Si, M.T selaku Dosen Pembimbing II yang telah meluangkan waktu untuk membimbing serta memberikan ilmu kepada penulis, sehingga penulis dapat menyelesaikan tugas akhir dengan baik.
3. Para Dosen FTI Unissula yang telah memberikan banyak ilmu yang bermanfaat.
4. Saudara dan teman serta kawan-kawan teknik informatika angkatan 2014 yang selalu memberikan semangat kepada penulis.

Dengan segala kerendahan hati, penulis menyadari masih banyak terdapat banyak kekurangan – kekurangan dari segi kualitas atau kuantitas maupun dari ilmu pengetahuan dalam penyusunan laporan, sehingga penulis mengharapkan adanya saran dan kritikan yang bersifat membangun laporan ini di masa mendatang.

Semarang, Juli 2021

Bima Candra Kharisma

## DAFTAR ISI

SISTEM REKOMENDASI INFORMASI LOWONGAN PEKERJAAN DENGAN METODE COSINE SIMILARITY & RABIN KARP K-GRAM .....	i
INFORMATION RECOMMENDATION SYSTEM FOR JOBS WITH COSINE SIMILARITY & RABIN KARP K-GRAM METODE.....	ii
LEMBAR PENGESAHAN PEMBIMBING .....	iii
LEMBAR PENGESAHAN PENGUJI .....	iii
Pernyataan Keaslian Tugas Akhir .....	iv
Pernyataan Persetujuan Publikasi Karya Ilmiah .....	v
Kata Pengantar .....	vi
Daftar Isi.....	vii
Daftar Tabel .....	x
Daftar Gambar.....	xi
Abstrak .....	xiii
BAB I Pendahuluan.....	1
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	3
1.3 Pembatasan Masalah .....	3
1.4 Tujuan .....	3
1.5 Manfaat .....	4
1.6 Sistematika Penulisan.....	4



BAB II Tinjauan Pustaka dan Dasar Teori .....	5
2.1    Tinjauan Pustaka .....	5
2.2    Dasar Teori.....	6
2.2.1    Sistem Rekomendasi .....	6
2.2.2    Pencocokan String.....	7
2.2.3    Cosine Similarity berbasis Rabin-karp.....	9
BAB III Metode Penelitian .....	16
3.1    Metode Pengumpulan data.....	16
3.2    Metode Pengembangan Sistem Prototype.....	16
3.3    Perancangan Sistem .....	17
3.3.1    Deskripsi Sistem.....	17
3.3.2    Use Case Diagram.....	19
3.3.3    Sequence Diagram .....	20
3.3.4    Activity Diagram.....	25
3.4    Perancangan User Interface.....	26
3.4.1    Layout Halaman User (Recruiter & Seeker).....	26
3.4.2    Layout Halaman Admin.....	27
BAB IV Hasil dan Analisis Penelitian.....	29
4.1    Tahap Implementasi.....	29
4.1.1    Admin.....	29
4.1.2    Recruiter .....	32
4.1.3    Seeker .....	36
4.2    Tahap Pengujian.....	41
BAB V Kesimpulan dan Saran .....	43
5.1    Kesimpulan .....	43

5.2	Saran.....	43
	Daftar Pustaka .....	45



## DAFTAR TABEL

Tabel 4.1: Deskripsi Use Case .....	19
Tabel 4.13: Pengujian Blackbox .....	41



## DAFTAR GAMBAR

Gambar 3.1 Model Prototyping.....	17
Gambar 3.2: Use Case Diagram.....	19
Gambar 3.3: Sequence Diagram Admin .....	21
Gambar 3.4: Sequence Diagram Recruiter.....	22
Gambar 3.5: Sequence Diagram Seeker.....	23
Gambar 3.6: Sequence Diagram User.....	24
Gambar 3.7: Activity Diagram Aplikasi.....	25
Gambar 3.8: Layout Halaman Beranda.....	26
Gambar 3.9: Layout Halaman Admin.....	27
Gambar 4.1: Login Admin.....	29
Gambar 4.2: Beranda Admin.....	30
Gambar 4.3: Data Recruiter .....	31
Gambar 4.4: Data Seeker .....	31
Gambar 4.5: Data Lokasi .....	32
Gambar 4.6: Data Spesialisasi.....	32
Gambar 4.7: Halaman Pengunjung Umum (Sebelum Login).....	33
Gambar 4.8: Login Recruiter .....	33
Gambar 4.9: Update Profile Recruiter .....	34
Gambar 4.10: Input Lowongan .....	34
Gambar 4.11: Daftar Postingan Lowongan.....	35
Gambar 4.12: Daftar Pelamar dan Detail Pelamar.....	35

Gambar 4.13: Login Seeker .....	36
Gambar 4.14: Daftar Lowongan .....	37
Gambar 4.15: Update Profil Seeker .....	38
Gambar 4.16: Rekomendasi Lowongan .....	39
Gambar 4.17: Detail Lowongan .....	40
Gambar 4.18: Lowongan yang di Apply .....	41



## ABSTRAK

Saat ini banyak sekali website informasi lowongan kerja yang tersebar di internet seperti JobsDB, Jobstreet, kerja.com dan sebagainya. Dengan adanya website informasi lowongan kerja tersebut sangat bermanfaat bagi lulusan baru untuk mencari informasi lowongan pekerjaan. banyaknya informasi yang tersedia di internet melalui website bursa lowongan kerja juga menimbulkan permasalahan baru dimana para pengguna kesulitan dalam mencari informasi lowongan kerja yang tepat dikarenakan banyaknya sumber informasi yang ada. Tujuan penelitian ini adalah untuk membangun sistem informasi lowongan kerja berbasis website dengan sistem rekomendasi menggunakan algoritma pencocokan teks Cosine Similarity yang dapat merekomendasikan lowongan berdasarkan preferensi pengguna yaitu keahlian, jurusan kuliah, jenjang pendidikan dan kota. metode cosine similarity dan rabin karp k-gram berhasil diimplementasikan pada Sistem Rekomendasi Pemilihan Lowongan Pekerjaan yang dapat mencari rekomendasi lowongan pekerjaan berdasarkan preferensi dan kemampuan yang dimiliki pencari kerja

*Kata Kunci: Lowongan Pekerjaan, Sistem Rekomendasi, Text Matching, Rabin Karp*

## ABSTRACT

Currently, there are so many job vacancies information websites scattered on the internet such as JobsDB, Jobstreet, Kerja.com and so on. With the job vacancy information website, it is very useful for new graduates to find job vacancy information. The large amount of information available on the internet through the job fair website also creates new problems where users find it difficult to find the right job vacancy information due to the large number of available sources of information. The purpose of this study is to build a website-based job vacancies information system with a recommendation system using the Cosine Similarity text-matching algorithm that can recommend vacancies based on user preferences, namely expertise, college majors, education level and city. The cosine similarity and rabin karp k-gram methods have been successfully implemented in the Job Vacancy Selection Recommendation System which can search for job vacancies based on the preferences and abilities of job seekers.

*Keywords: Job Vacancy, Recommender System, Text Matching, Rabin Karp*

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Situs Web (Website) merupakan lokasi yang terdiri dari kumpulan halaman web yang terkait dan dapat diakses publik dengan mengunjungi situs menggunakan aplikasi web browser melalui alamat *Uniform Resource Locator (URL)*. Website dioperasikan oleh suatu individu, kelompok, usaha atau organisasi yang digunakan untuk berbagai tujuan. Penggunaan website sebagai penyebaran informasi salah satunya dimanfaatkan oleh perusahaan untuk menyebarkan informasi lowongan pekerjaan. Sebelumnya, penyebaran dan pencarian lowongan pekerjaan dilakukan secara manual yaitu mendatangi tempat secara langsung dan mendatangi pameran lowongan kerja atau job fair. Penyebaran informasi lowongan pekerjaan melalui media website digunakan untuk mendukung kecepatan dalam penyebaran informasi melalui pameran lowongan kerja seperti job fair (Susilawati, 2018). Dengan begitu para pencari kerja sebagai calon karyawan dapat dengan mudah mendapatkan informasi karena melalui website informasi dapat diakses dengan bantuan internet.

Saat ini banyak sekali website informasi lowongan kerja yang tersebar di internet seperti JobsDB, Jobstreet, kerja.com dan sebagainya. Dengan adanya website informasi lowongan kerja tersebut sangat bermanfaat bagi lulusan baru untuk mencari informasi lowongan pekerjaan. banyaknya informasi yang tersedia di internet melalui website bursa lowongan kerja juga menimbulkan permasalahan baru dimana para pengguna kesulitan dalam mencari informasi lowongan kerja yang tepat dikarenakan banyaknya sumber informasi yang ada. Masalah yang timbul ketika terdapat banyaknya informasi yang tersedia dapat menyebabkan pengguna sebagai pencari informasi menjadi kebingungan. Masalah kelebihan informasi biasa disebut *information overload* (Hadna, 2016).

Masalah kelebihan informasi biasanya dapat diselesaikan dengan sistem rekomendasi. Sistem rekomendasi dapat menyediakan informasi yang dibutuhkan

sesuai dengan keinginan atau preferensi pengguna (Lu, 2015). Masalah kelebihan informasi yang terdapat pada website bursa kerja dapat diatasi dengan solusi penggunaan sistem rekomendasi (Hernowo & Sudaryanto, 2016). Solusi dari permasalahan tersebut adalah dengan menyaring informasi lowongan kerja yang sesuai dengan profil atau preferensi dari pengguna seperti keahlian, jurusan kuliah, jenjang pendidikan maupun kota yang dikehendaki.

Berdasarkan permasalahan yang telah diuraikan tentang kelebihan informasi pada website lowongan pekerjaan proposal penelitian ini mengusulkan untuk membuat solusi dengan menerapkan sistem rekomendasi pada website lowongan pekerjaan. Sistem rekomendasi yang diusulkan menggunakan metode pencocokan teks dengan tujuan menyaring informasi berdasarkan kemiripan teks pengguna (pencari lowongan) dengan data-data lowongan yang ada. Metode pencocokan teks yang diusulkan adalah cosine similarity. Metode *cosine similarity* diusulkan karena salah satu metode yang dapat digunakan untuk mencari kemiripan teks (Ariantini & Rai, 2016). Metode tersebut memiliki kelebihan diantaranya cocok pada kasus pencarian string dengan pattern yang panjang dan pencarian string multipattern. Kelebihan tersebut sangat cocok diimplementasikan pada penelitian ini karena kasus pencocokan teks preferensi dan kemampuan pengguna dengan teks informasi lowongan merupakan jenis teks yang panjang dan mempunyai pattern yang banyak.

Penelitian ini mengusulkan metode *Cosine Similarity* pada penelitian ini yang digunakan untuk mencari kecocokan teks preferensi pengguna dan data lowongan pekerjaan pada sistem rekomendasi berbasis website. Metode yang diusulkan diharapkan dapat membantu pencari pekerjaan mencari informasi lowongan yang tepat sesuai dengan preferensi pengguna.



## 1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan, maka rumusan masalah dalam Tugas Akhir ini yaitu :

1. Membangun aplikasi informasi lowongan kerja berbasis website dengan sistem rekomendasi menggunakan algoritma pencocokan teks *Cosine Similarity*
2. Menentukan rekomendasi lowongan kerja berdasarkan kemiripan teks preferensi pengguna dan teks lowongan pekerjaan

## 1.3 Pembatasan Masalah

Adapun batasan masalah dalam Tugas Akhir ini adalah :

1. Pencocokan teks menggunakan metode *cosine similarity*
2. Sistem yang dibangun adalah sistem informasi rekomendasi lowongan pekerjaan berdasarkan preferensi pengguna yaitu keahlian, jurusan kuliah, jenjang pendidikan dan kota.
3. Sistem dibangun berbasis website menggunakan PHP dan MySQL.

## 1.4 Tujuan

Tujuan penelitian ini adalah untuk membangun sistem informasi lowongan kerja berbasis website dengan sistem rekomendasi menggunakan algoritma pencocokan teks *Cosine Similarity* yang dapat merekomendasikan lowongan berdasarkan preferensi pengguna yaitu keahlian, jurusan kuliah, jenjang pendidikan dan kota.

## 1.5 Manfaat

Manfaat dari penelitian tentang Sistem Rekomendasi Informasi Lowongan Pekerjaan Dengan Metode *Cosine Similarity & Rabin Karp K-Gram*

1. Sistem ini dapat menghemat waktu dalam pencarian rekomendasi informasi lowongan pekerjaan bagi pencari kerja. Selain itu juga memudahkan pemberi lowongan dalam mencari kandidat pekerja.
2. Sistem berbasis website menjadikan akses pencarian informasi dapat dilakukan kapan dan dimana saja menggunakan media internet.

## 1.6 Sistematika Penulisan

Sistematika penulisan dalam tugas akhir ini, disusun sebagai berikut:

### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan, manfaat dan sistematika penulisan.

### **BAB II TINJAUAN PUSTAKA**

Bab ini berisi tinjauan pustaka merupakan penelitian terdahulu yang dijadikan acuan dan landasan teori terkait dengan penelitian ini.

### **BAB III METODE PENELITIAN**

Bab ini menjelaskan metode pengumpulan data, langkah-langkah penelitian dan analisis desain aplikasi.

### **BAB IV HASIL DAN ANALISIS PENELITIAN**

Bagian ini berisi hasil dan pembahasan dari penelitian yaitu analisis dan pengolahan dataset, implementasi rabin karp, implementasi system klasifikasi dan pengujian klasifikasi.

### **BAB V KESIMPULAN DAN SARAN**

Pada bab ini berisikan beberapa kesimpulan dari hasil penelitian.

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Sebelumnya terdapat penelitian yang dijadikan acuan dan referensi pada penelitian ini. Penelitian yang dijadikan tinjauan pustaka merupakan penelitian yang terkait tentang objek penelitian atau metode yang digunakan. Pencarian informasi lowongan kerja praktek dengan cara manual menyebabkan kesulitan dan ketidakpastian dalam perolehan informasi sehingga mahasiswa yang ingin melakukan kegiatan kerja praktek mengalami kesulitan dalam pencarian tempat kerja praktek dan menemukan tempat yang tidak tepat sesuai jurusan yang ditempuh oleh mahasiswa tersebut. Hernowo Menerapkan algoritma *Booyer Moore* untuk kasus pencocokan string dalam mencari rekomendasi informasi tempat kerja praktek dan menerapkan sistem rekomendasi pada aplikasi pencarian tempat kerja praktek bagi mahasiswa. Berdasarkan hasil penelitian algoritma boyer moore telah berhasil diimplementasikan pada website informasi lowongan kerja praktek. Dalam prosesnya algoritma boyer moore digunakan untuk mencocokkan jurusan dan keahlian mahasiswa pada lowongan kerja praktek sehingga memudahkan mahasiswa menemukan menemukan tempat kerja praktek yang sesuai dengan jurusan dan keahlian yang dimiliki maupun yang dibutuhkan perusahaan tempat kerja praktek (Hernowo & Sudaryanto, 2016).

Penelitian kedua tentang sistem rekomendasi pekerjaan oleh Arie Anggono et al. Menurut peneliti seiring kelulusan yang banyak menjadikan pencari kerja juga banyak. Dengan berkembangnya zaman informasi lowongan pekerjaan dapat diakses melalui portal lowongan pekerjaan online. Namun banyaknya informasi menyebabkan pelamar tidak dapat menemukan lowongan yang sesuai dengan mudah. Oleh karena itu peneliti mengusulkan pembuatan sistem informasi lowongan pekerjaan online dengan sistem rekomendasi. Metode yang digunakan adalah *Simple Additive Weighting*. Metode SAW merupakan metode yang paling dikenal dan paling banyak digunakan orang dalam menghadapi situasi *Multiple*

*Attribute Decision Making*. Berdasarkan hasil penelitian Sistem informasi portal lowongan kerjadengan sistem rekomendasi berhasil dibuat dengan menggunakan sistem rekomendasi yang mengacu pada salah satu metode Decision Support Sistem yaitu *Simple Additive Weighting* (Anggono, 2015).

Penelitian ketiga yang dijadikan referensi adalah tentang penerapan cosine similarity untuk mencari kemiripan dokumen. Pengolahan informasi dengan menggunakan komputer menghasilkan fasilitas yang *copy paste* sehingga memudahkan pengolahan data sesuai dengan kebutuhan misalnya memenuhi tugas kuliah membuat paper dan sebagainya. Hal ini tentu berpotensi terjadinya tindakan plagiat suatu karya tulis tanpa izin seperti plagiat. Penelitian ini melakukan implementasi *cosine similarity* untuk mengecek kemiripan antar dokumen yang diterapkan untuk mencari plagiarism. Berdasarkan hasil penelitian sistem untuk mengukur tingkat kemiripan antar dokumen teks berbahasa Indonesia berhasil dibangun menggunakan algoritma *Cosine similarity*. Penggunaan data dummy pada sistem deteksi kemiripan dokumen sudah berhasil diterapkan dan sudah bisa digunakan untuk mengukur tingkat kemiripan dokumen (Ariantini & Rai, 2016).

## **2.2 Dasar Teori**

### **2.2.1 Sistem Rekomendasi**

Sistem rekomendasi merupakan perangkat lunak dan teknik yang dapat menyarankan suatu item atau informasi kepada pengguna atau user (Ricci, et al., 2013). Rekomendasi terkait dengan proses pemilihan keputusan seperti item yang akan dibeli, music yang didengarkan atau informasi yang ingin dibaca.

“Item” adalah istilah umum yang digunakan sebagai penunjuk dari apa yang direkomendasikan oleh sistem kepada pengguna. Sistem rekomendasi biasanya focus pada jenis item tertentu (seperti Film atau berita) dan sesuai desain, antarmuka dan teknik rekomendasi digunakan untuk menghasilkan rekomendasi yang sesuai.

Sistem rekomendasi biasanya diarahkan pada orang-orang (pengguna) yang belum memiliki cukup pengalaman pribadi atau kompetensi untuk mengevaluasi banyaknya item pada suatu website. Contohnya yaitu seperti amazon.com. toko online tersebut menjual buku yang jumlahnya sangat banyak karena itu membutuhkan rekomendasi untuk setiap pengguna karena karakteristik pengguna sangatlah berbeda.

Dalam merekomendasikan suatu item sistem rekomendasi tentunya mempunyai metode yang digunakan. Metode yang digunakan umumnya adalah *Collaborative filtering* dan *Content Based Filtering*.

### **2.2.2 Pencocokan String**

String adalah susunan dari karakter-karakter (angka, alphabet, atau karakter yang lain) dan biasanya direpresentasikan sebagai struktu data array. String dapat berupa kata, frase, atau kalimat. Sedangkan string matching diartikan sebagai sebuah permasalahan untuk menemukan pola susunan karakter string didalam string lain atau bagian dari isi teks. String matching dalam bahasa Indonesia dikenal dengan istilah pencocokan string (Ernawati & Johar, 2019).

Pencocokan string merupakan bagian penting dari sebuah proses pencarian string (string searching) dalam sebuah dokumen. Hasil dari pencarian sebuah string dalam dokumen tergantung dari teknik atau cara pencocokan string yang digunakan. Pencocokan string (string matching) secara garis besar dapat dibedakan menjadi dua yaitu sebagai berikut.

Exact string matching, merupakan pencocokan string secara tepat dengan susunan karakter dalam string yang dicocokkan memiliki jumlah maupun urutan karakter dalam string yang sama. Contoh : kata step akan menunjukkan kecocokan hanya dengan kata step.

inexact string matching atau Fuzzy string matching, merupakan pencocokan string secara samar, maksudnya pencocokan string dimana string yang dicocokkan memiliki kemiripan dimana keduanya memiliki susunan karakter yang berbeda (mungkin jumlah atau urutannya) tetapi string-string tersebut memiliki kemiripan baik kemiripan tekstual/penulisan (approximate string matching) atau kemiripan ucapan (phonetic string matching). Inexact string matching masih dapat dibagi lagi menjadi dua yaitu.

- a. Pencocokan string berdasarkan kemiripan penulisan (approximate string matching) merupakan pencocokan string dengan dasar kemiripan dari segi penulisannya (jumlah karakter, susunan karakter dalam dokumen). Tingkat kemiripan ditentukan dengan jauh tidaknya beda penulisan dua buah string yang dibandingkan tersebut dan nilai tingkat kemiripan ini ditentukan oleh pemrogram (programmer). Contoh : compiler dengan kompuler, memiliki jumlah karakter yang sama tetapi ada dua karakter yang berbeda. Jika perbedaan dua karakter ini dapat ditoleransi sebagai sebuah kesalahan penulisan maka dua string tersebut dikatakan cocok.
- b. Pencocokan string berdasarkan kemiripan ucapan (phonetic string matching) merupakan pencocokan string dengan dasar kemiripan dari segi pengucapannya meskipun ada perbedaan penulisan dua string yang dibandingkan tersebut. Contoh step dengan step dari tulisan berbeda tetapi dalam pengucapannya mirip sehingga dua string tersebut dianggap cocok. Contoh yang lain adalah step, dengan steppe, stpep, step.

### 2.2.3 Cosine Similarity berbasis Rabin-karp

Algoritma *Rabin-Karp* merupakan versi awal dari *fingerprinting* dengan menggunakan metode *k-gram* yang diciptakan oleh Michael O. Rabin dan Richard M. Karp pada tahun 1987. Pada dasarnya, Algoritma *Rabin Karp* menghitung nilai *hash* untuk *pattern* dan setiap *k-gram* dari teks yang akan dibandingkan. Jika nilai *hash* tidak sama, maka akan dihitung nilai *hash* untuk *k-gram* selanjutnya secara berurutan. Jika nilai *hash* sama, maka dilakukan perbandingan antara *pattern* dan *k-gram*. Dengan cara ini, hanya ada satu perbandingan dari *subsequence* teks, dan *Brute Force* hanya diperlukan ketika nilai *hash* sama (Doddi Aria & Sujaini, 2015).

Rabin Karp merepresentasikan setiap karakter ke dalam bentuk desimal digit (digit radix-d)  $\Sigma = \{0, 1, 2, 3, \dots, d\}$ , dimana  $d = |\Sigma|$ . Sehingga didapat masukan string *k* berturut-turut sebagai perwakilan panjang *k* desimal. Karakter string 31415 sesuai dengan jumlah desimal 31,415. Kemudian pola *p* dihash menjadi nilai desimal dan string direpresentasikan dengan penjumlahan digit-digit angka menggunakan aturan Horner's, misal

$$\{ A, B, C, \dots, Z \} \rightarrow \{ 0, 1, 2, \dots, 26 \}$$

$$\bullet \text{ BAN} \rightarrow 1 + 0 + 13 = 14$$

$$\bullet \text{ CARD} \rightarrow 2 + 0 + 17 + 3 = 22$$

Karakteristik Algoritma *Rabin-Karp* :

- a) Menggunakan sebuah fungsi *hashing*
- b) Fase *preprocessing* menggunakan kompleksitas waktu  $O(m)$
- c) Untuk fase pencarian kompleksitasnya :  $O(mn)$
- d) Waktu yang diperlukan  $O(n+m)$

Fungsi *hashing* mengurangi perbandingan dari setiap *k-gram* yang quadratik di dalam banyak kasus atau situasi. Daripada melakukan pemeriksaan dari *k-gram* di setiap posisi dari teks ketika terjadi pencocokan *pattern*, akan lebih baik dan

efisien untuk melakukan pemeriksaan dari  $k$ -gram hanya jika teks yang sedang proses memiliki kemiripan nilai  $hash$  seperti pada  $pattern$ . Untuk melakukan pengecekan kemiripan antara dua kata ini digunakan fungsi  $hash$ . Sedangkan perhitungan nilai  $hash$  selanjutnya dilakukan *Rolling hash*.

Berikut ini merupakan langkah-langkah dalam mencocokkan kata dengan algoritma *Rabin-Karp*:

1. Hilangkan tanda baca dan ubah teks sumber menjadi kata-kata tanpa huruf kapital.
2. Tentukan panjang dari teks sumber yang ingin dicari ( $n$ ) dan kata yang ingin dicari ( $m$ )
3. Mencari nilai  $hash$  dari teks sumber dan kata yang ingin dicari menggunakan fungsi  $hash$  yang telah ditentukan
4. Lakukan iterasi dari indeks  $i=0$  sampai  $i=n-m+1$  untuk membandingkan nilai  $hash$  dari kata yang ingin dicari dengan nilai  $hash$  dari teks sumber pada indeks  $i$  sampai dengan  $i+m-1$ . Jika nilai  $hash$  sama, maka akan diperiksa lebih lanjut apakah kata yang dicari sama dengan bagian teks dari sumber pada indeks  $i$  sampai dengan  $i+m-1$ . Jika sama, maka telah ditemukan kata yang cocok. Jika tidak maka dilanjutkan dengan membandingkan nilai  $hash$  dari kata yang dicari dengan nilai  $hash$  teks sumber pada indeks berikutnya.

*Cosine Similarity* adalah ukuran kesamaan antara dua buah vektor dalam sebuah ruang dimensi yang didapat dari nilai cosinus sudut dari perkalian dua buah vektor yang dibandingkan karena cosinus dari  $0^\circ$  adalah 1 dan kurang dari 1 untuk nilai sudut yang lain, maka nilai similarity dari dua buah vektor dikatakan mirip ketika nilai dari cosine similarity adalah 1.



*Cosine similarity* digunakan dalam ruang positif, dimana hasilnya dibatasi antara nilai 0 dan 1. Kalau nilainya 0 maka dokumen tersebut dikatakan mirip jika hasilnya 1 maka nilai tersebut dikatakan tidak mirip. Perhatikan bahwa batas ini berlaku untuk sejumlah dimensi, dan *Cosine similarity* ini paling sering digunakan dalam ruang positif dimensi tinggi. Misalnya, dalam Information Retrieval, masing-masing kata/istilah (term) diasumsikan sebagai dimensi yang berbeda dan dokumen ditandai dengan vector dimana nilai masing-masing dimensi sesuai dengan berapa istilah muncul dalam dokumen. Berikut adalah rumus cosine similarity.

$$\text{Cosine Similarity} = \frac{\sum_{k=1}^i (\text{weight}_{ik} \text{weight}_{qk})}{\sqrt{\sum_{k=1}^i (\text{weight}_{ik}^2 \text{weight}_{qk}^2)}}$$

Keterangan:

Cosine Similarity : tingkat kesamaan dokumen

Weight ij : term ke-I dalam vector untuk dokumen ke -j

Weight ik : term ke-I dalam vector untuk query ke -k

N : jumlah term yang unik dalam set

Dimana A merupakan bobot setiap ciri pada vector A, dan B merupakan bobot setiap ciri pada B. jika dikaitkan dengan information retrieval maka A adalah bobot setiap istilah pada dokumen A, dan B merupakan bobot setiap istilah pada dokumen B. Pada penelitian ini digunakan cosine similarity karena citra merupakan salah satu data yang memiliki dimensi tinggi. Pada citra dapat dikatakan bahwa setiap pixel merupakan dimensi yang berbeda dan nilai warna pada setiap pixel tersebut merupakan nilai dari setiap dimensi. Pengukuran kemiripan dapat dilakukan dengan membandingkan dokumen 1 dengan dokumen 2 kemudian sistem akan menghitung nilai kemiripan.  $A_i \cdot B_i$  adalah nilai yang diperoleh dari term A dan term B kemudian kedua nilai tersebut dijumlahkan, kemudian nilai  $A_i^2$  semua nilai term dokumen A semua nilainya dipangkatkan dua, begitu juga dengan term  $B_i^2$  semua nilai yang diperoleh dipangkatkan dua kemudian semua nilai yang diperoleh dijumlahkan.

## Sampel Perhitungan Cosine Similarity berbasis Rabin Karp

Text 1:

Dibutuhkan junior programmer menguasai pemrograman web php, mysql, html, css. Menguasai wordpress lebih disukai

Text 2

Menguasai pemrograman web php, mysql, html dan css

K-Gram

K = 5

Basis Bilangan b=7

**Langkah 1:** case folding dan pembersihan teks

Text 1	Text 2
dibutuhkan junior programmer menguasai pemrograman web php mysql html css menguasai wordpress lebih disukai	Menguasai pemrograman web php Mysql html dan css

**Langkah 2:** memecah teks menjadi nilai K-Gram (K=5)

Text 1	Text 2
{dibut} {ibutu} {butuh} {utuhk} {tuhka} {uhkan} {hkan } {kan j} {an ju} {n jun} { juni} {junio} {unior} {nior } {ior p} {or pr} {r pro} { prog} {progr} {rogra} {ogram} {gramm} {ramme} {ammer} {mmer } {mer m} {er me} {r men} { meng} {mengu} {engua} {nguas} {guasa} {uasai}	{mengu} {engua} {nguas} {guasa} {uasai} {asai } {sai p} {ai pe} {i pem} { pemr} {pemro} {emrog} {mrogr} {rogra} {ogram} {grama} {raman} {aman } {man w} {an we} {n web} { web } {web p} {eb ph} {b php} { php } {php m} {hp my} {p mys} { mysql} {mysql} {ysql } {sql h} {ql ht} {l htm}

{asai } {sai p} {ai pe} {i pem} { pemr}	{ html} {html } {tml d} {ml da} {l
{pemro} {emrog} {mrogr} {rogra}	dan} { dan } {dan c} {an cs} {n css}
{ogram} {grama} {raman} {aman }	
{man w} {an we} {n web} { web }	
{web p} {eb ph} {b php} { php } {php	
m} {hp my} {p mys} { mysql} {mysql}	
{ysql } {sql h} {ql ht} {l htm} { html}	
{html } {tml c} {ml cs} {l css} { css }	
{css m} {ss me} {s men} { meng}	
{mengu} {engua} {nguas} {guasa}	
{uasai} {asai } {sai w} {ai wo} {i wor}	
{ word} {wordp} {ordpr} {rdpre}	
{dpres} {press} {ress } {ess l} {ss le}	
{s leb} { lebi} {lebih} {ebih } {bih d}	
{ih di} {h dis} { disu} {disuk} {isuka}	
{sukai}	

**Langkah 3:** konversi teks k-gram menjadi nilai hash dengan rumus:

Contoh hashing dengan kata { **dibut** }

Nilai ASCII dari { **dibut** }

ASCII (d) = 100

ASCII (i) = 105

ASCII (b) = 98

ASCII (u) = 117

ASCII (t) = 116

$$H_{(C_1 \dots C_K)} = C_1 \cdot b^{(K-1)} + C_2 \cdot b^{(K-2)} + \dots + C_K \cdot b^{(0)} + ck$$

$$H = (100 \cdot 7^5) + (105 \cdot 7^4) + (98 \cdot 7^3) + (117 \cdot 7^2) + (116 \cdot 7^1)$$

$$H = (100 \cdot 16807) + (105 \cdot 2401) + (98 \cdot 343) + (117 \cdot 49) + (116 \cdot 7)$$

$$H = 1680700 + 252105 + 33614 + 5733 + 812$$

$$H = 1617116$$

Maka berdasarkan perhitungan hash tersebut:

Hash Loker	Hash Seeker
1617116 1683293 1605972 1882801	1744860 1638998 1762942 1676849
1867941 1865545 1678060 1709462	1857191 1586166 1825991 1565137
1571742 1667325 625070 1722249	1594669 631118 1788777 1638938
1873447 1764982 1699304 1782103	1762281 1830629 1777214 1672790
1726583 632702 1806204 1830629	1812547 1578235 1738757 1571869
1777214 1672802 1812670 1579670	1668710 640232 1889032 1614387
1754455 1744555 1635647 1726076	1492218 631432 1792229 1676928
627125 1744860 1638998 1762942	1697019 629610 1772186 1939519
1676849 1857191 1586166 1825991	1847642 1803313 1637789 622279
1565137 1594669 631118 1788777	1691469 1856955 1744686 1637097
1638938 1762281 1830629 1777214	614591 1606968 1571663 1666461
1672790 1812547 1578235 1738757	
1571869 1668710 640232 1889032	
1614387 1492218 631432 1792229	
1676928 1697019 629610 1772186	
1939519 1847642 1803313 1637789	
622279 1691469 1856954 1744693	
1637179 615493 1616900 1841952	
1740717 627125 1744860 1638998	
1762942 1676849 1857191 1586166	
1825998 1565224 1595631 641686	
1905026 1790331 1817081 1628192	
1805127 1818717 1646181 1841941	

1740584 625664 1728776 1623060 1587609 1680806 1578626 615699 1619164 1705801 1853561	
Jumlah Hash : 103	Jumlah Hash: 44

#### Langkah 4: Menghitung Cosine Similarity

##### a. Mencari nilai TFIDF dan Skalar Data Hash Seeker dan Hash Loker

Hash Term	Frequency		DF	IDF	TF*IDF		Skalar
	Loker Hash	Seeker Hash			Loker Hash	Seeker Hash	
1617116	1	0	1	0.70	0.70	0.00	0
1683293	1	0	1	0.70	0.70	0.00	0
1605972	1	0	1	0.70	0.70	0.00	0
1882801	1	0	1	0.70	0.70	0.00	0
1867941	1	0	1	0.70	0.70	0.00	0
1865545	1	0	1	0.70	0.70	0.00	0
1678060	1	0	1	0.70	0.70	0.00	0
1709462	1	0	1	0.70	0.70	0.00	0
1571742	1	0	1	0.70	0.70	0.00	0
1667325	1	0	1	0.70	0.70	0.00	0
625070	1	0	1	0.70	0.70	0.00	0
1722249	1	0	1	0.70	0.70	0.00	0
1873447	1	0	1	0.70	0.70	0.00	0
1764982	1	0	1	0.70	0.70	0.00	0
1699304	1	0	1	0.70	0.70	0.00	0
1782103	1	0	1	0.70	0.70	0.00	0
1726583	1	0	1	0.70	0.70	0.00	0
632702	1	0	1	0.70	0.70	0.00	0
1806204	1	0	1	0.70	0.70	0.00	0
1830629	2	1	3	1.18	2.35	1.18	2.7664
1777214	2	1	3	1.18	2.35	1.18	2.7664
1672802	1	0	1	0.70	0.70	0.00	0
1812670	1	0	1	0.70	0.70	0.00	0
1579670	1	0	1	0.70	0.70	0.00	0
1754455	1	0	1	0.70	0.70	0.00	0
1744555	1	0	1	0.70	0.70	0.00	0
1635647	1	0	1	0.70	0.70	0.00	0
1726076	1	0	1	0.70	0.70	0.00	0

627125	2	0	2	1.00	2.00	0.00	0
1744860	2	1	3	1.18	2.35	1.18	2.7664
1638998	2	1	3	1.18	2.35	1.18	2.7664
1762942	2	1	3	1.18	2.35	1.18	2.7664
1676849	2	1	3	1.18	2.35	1.18	2.7664
1857191	2	1	3	1.18	2.35	1.18	2.7664
1586166	2	1	3	1.18	2.35	1.18	2.7664
1825991	1	1	2	1.00	1.00	1.00	1
1565137	1	1	2	1.00	1.00	1.00	1
1594669	1	1	2	1.00	1.00	1.00	1
631118	1	1	2	1.00	1.00	1.00	1
1788777	1	1	2	1.00	1.00	1.00	1
1638938	1	1	2	1.00	1.00	1.00	1
1762281	1	1	2	1.00	1.00	1.00	1
1672790	1	1	2	1.00	1.00	1.00	1
1812547	1	1	2	1.00	1.00	1.00	1
1578235	1	1	2	1.00	1.00	1.00	1
1738757	1	1	2	1.00	1.00	1.00	1
1571869	1	1	2	1.00	1.00	1.00	1
1668710	1	1	2	1.00	1.00	1.00	1
640232	1	1	2	1.00	1.00	1.00	1
1889032	1	1	2	1.00	1.00	1.00	1
1614387	1	1	2	1.00	1.00	1.00	1
1492218	1	1	2	1.00	1.00	1.00	1
631432	1	1	2	1.00	1.00	1.00	1
1792229	1	1	2	1.00	1.00	1.00	1
1676928	1	1	2	1.00	1.00	1.00	1
1697019	1	1	2	1.00	1.00	1.00	1
629610	1	1	2	1.00	1.00	1.00	1
1772186	1	1	2	1.00	1.00	1.00	1
1939519	1	1	2	1.00	1.00	1.00	1
1847642	1	1	2	1.00	1.00	1.00	1
1803313	1	1	2	1.00	1.00	1.00	1
1637789	1	1	2	1.00	1.00	1.00	1
622279	1	1	2	1.00	1.00	1.00	1
1691469	1	1	2	1.00	1.00	1.00	1
1856954	1	0	1	0.70	0.70	0.00	0
1744693	1	0	1	0.70	0.70	0.00	0
1637179	1	0	1	0.70	0.70	0.00	0
615493	1	0	1	0.70	0.70	0.00	0
1616900	1	0	1	0.70	0.70	0.00	0
1841952	1	0	1	0.70	0.70	0.00	0
1740717	1	0	1	0.70	0.70	0.00	0

1825998	1	0	1	0.70	0.70	0.00	0	
1565224	1	0	1	0.70	0.70	0.00	0	
1595631	1	0	1	0.70	0.70	0.00	0	
641686	1	0	1	0.70	0.70	0.00	0	
1905026	1	0	1	0.70	0.70	0.00	0	
1790331	1	0	1	0.70	0.70	0.00	0	
1817081	1	0	1	0.70	0.70	0.00	0	
1628192	1	0	1	0.70	0.70	0.00	0	
1805127	1	0	1	0.70	0.70	0.00	0	
1818717	1	0	1	0.70	0.70	0.00	0	
1646181	1	0	1	0.70	0.70	0.00	0	
1841941	1	0	1	0.70	0.70	0.00	0	
1856955	0	1	1	0.70	0.00	0.70	0	
1744686	0	1	1	0.70	0.00	0.70	0	
1637097	0	1	1	0.70	0.00	0.70	0	
614591	0	1	1	0.70	0.00	0.70	0	
1606968	0	1	1	0.70	0.00	0.70	0	
1571663	0	1	1	0.70	0.00	0.70	0	
1666461	0	1	1	0.70	0.00	0.70	0	
							Sum Skalar	51.131

b. Menghitung Cosine Similarity

Hash Term	TFIDF^2	
	Loker Hash	Seeker Hash
1617116	0.49	0.00
1683293	0.49	0.00
1605972	0.49	0.00
1882801	0.49	0.00
1867941	0.49	0.00
1865545	0.49	0.00
1678060	0.49	0.00
1709462	0.49	0.00
1571742	0.49	0.00
1667325	0.49	0.00
625070	0.49	0.00
1722249	0.49	0.00
1873447	0.49	0.00
1764982	0.49	0.00
1699304	0.49	0.00
1782103	0.49	0.00

1726583	0.49	0.00
632702	0.49	0.00
1806204	0.49	0.00
1830629	5.53	1.38
1777214	5.53	1.38
1672802	0.49	0.00
1812670	0.49	0.00
1579670	0.49	0.00
1754455	0.49	0.00
1744555	0.49	0.00
1635647	0.49	0.00
1726076	0.49	0.00
627125	4.00	0.00
1744860	5.53	1.38
1638998	5.53	1.38
1762942	5.53	1.38
1676849	5.53	1.38
1857191	5.53	1.38
1586166	5.53	1.38
1825991	1.00	1.00
1565137	1.00	1.00
1594669	1.00	1.00
631118	1.00	1.00
1788777	1.00	1.00
1638938	1.00	1.00
1762281	1.00	1.00
1672790	1.00	1.00
1812547	1.00	1.00
1578235	1.00	1.00
1738757	1.00	1.00
1571869	1.00	1.00
1668710	1.00	1.00
640232	1.00	1.00
1889032	1.00	1.00
1614387	1.00	1.00
1492218	1.00	1.00
631432	1.00	1.00
1792229	1.00	1.00
1676928	1.00	1.00
1697019	1.00	1.00
629610	1.00	1.00



1772186	1.00	1.00
1939519	1.00	1.00
1847642	1.00	1.00
1803313	1.00	1.00
1637789	1.00	1.00
622279	1.00	1.00
1691469	1.00	1.00
1856954	0.49	0.00
1744693	0.49	0.00
1637179	0.49	0.00
615493	0.49	0.00
1616900	0.49	0.00
1841952	0.49	0.00
1740717	0.49	0.00
1825998	0.49	0.00
1565224	0.49	0.00
1595631	0.49	0.00
641686	0.49	0.00
1905026	0.49	0.00
1790331	0.49	0.00
1817081	0.49	0.00
1628192	0.49	0.00
1805127	0.49	0.00
1818717	0.49	0.00
1646181	0.49	0.00
1841941	0.49	0.00
1856955	0.00	0.49
1744686	0.00	0.49
1637097	0.00	0.49
614591	0.00	0.49
1606968	0.00	0.49
1571663	0.00	0.49
1666461	0.00	0.49
sum(TFIDF^2)	99.25	43.49
sqrt(sum(TFIDF^2))	9.96	6.59

$$\text{Cosine Similarity } (D \text{ Training}, D \text{ Testing}) = \frac{\sum_{k=1}^i (\text{weight}_{ik} \text{weight}_{qk})}{\sqrt{\sum_{k=1}^i (\text{weight}_{ik}^2 \text{weight}_{qk}^2)}}$$

$$\text{Cosine Similarity } (D \text{ Training}, D \text{ Testing}) = \frac{51.13}{9.96 * 6.59} = 0.77$$

$$\text{Cosine Similarity} = 0.77 * 100 = 77 \text{ persen}$$



## **BAB III**

### **METODE PENELITIAN**

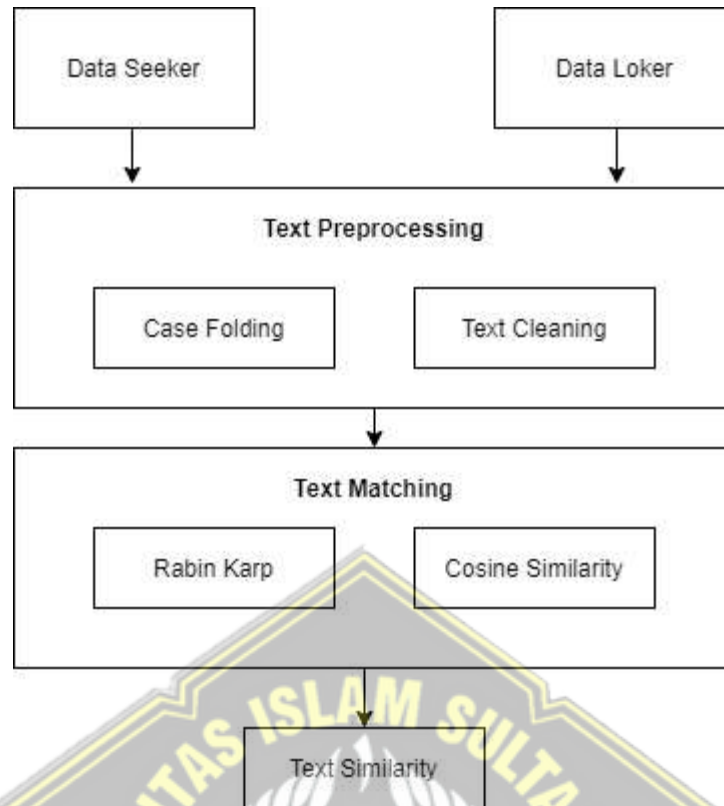
#### **3.1 Metode Pengumpulan data**

Metode pengumpulan data adalah teknik atau cara yang dilakukan oleh peneliti untuk mengumpulkan data. Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Adapun dalam pengumpulan data dilakukan dengan dua metode yaitu metode primer dan metode sekunder.

Metode primer dilakukan dengan cara pengumpulan data secara langsung pada website-website penyedia informasi lowongan kerja atau portal lowongan kerja. Sedangkan metode sekunder dilakukan dengan cara mengumpulkan dokumen, informasi dan browsing terkait informasi lowongan kerja dan metode Cosine Similarity.

#### **3.2 Metode Implementasi Algoritma**

Pada penelitian ini mengusulkan rekomendasi lowongan pekerjaan menggunakan algoritma rabin karp berbasis cosine similarity. Proses utama dalam penelitian adalah persiapan data, preprocessing data dan rekomendasi.



**Gambar 3.1: Metode Penelitian**

Pada gambar 3.1 ditunjukkan metode penelitian dari penelitian ini langkah-langkah terdiri dari data preparation, preprocessing dan rekomendasi. Langkah-langkah dijelaskan sebagai berikut:

1. Data preparation

Data preparation merupakan persiapan data yang akan gunakan yaitu data seeker dan data loker yang tersimpan di database. Data loker akan digunakan untuk acuan pencocokan mencari nilai similarity.

2. Text Preprocessing

Dalam text mining, data teks akan diproses menjadi data numerik agar dapat dilakukan proses lebih lanjut. Tahap preprocessing yang digunakan dalam penelitian ini antara lain.

a. Case Folding

Merupakan tahap perubahan huruf dari huruf kapital menjadi huruf kecil.

b. Tokenizing

Tokenizing adalah proses memecah dokumen menjadi kumpulan kata. Tokenization dapat dilakukan dengan menghilangkan tanda baca dan memisahkannya per spasi. Tahapan ini juga menghilangkan karakter-karakter tertentu seperti tanda baca dan mengubah semua token ke bentuk huruf kecil (lower case)

c. Stopword Removal

Stopwords removal merupakan proses penghilangan kata tidak penting pada deskripsi melalui pengecekan kata-kata hasil parsing deskripsi apakah termasuk di dalam daftar kata tidak penting (stoplist) atau tidak. Jika termasuk di dalam stoplist maka kata-kata tersebut akan di-remove dari deskripsi sehingga kata-kata yang tersisa di dalam deskripsi dianggap sebagai kata-kata penting atau keywords

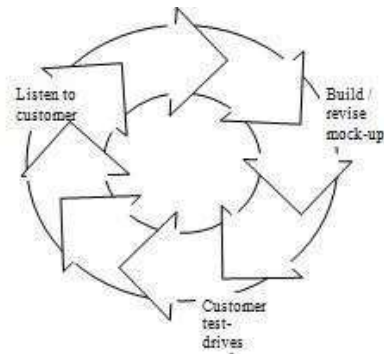
### 3. Rekomendasi

Pada tahap rekomendasi metode yang digunakan adalah rabin karp berbasis cosine similarity. Sebelum proses cosine similarity teks diubah dalam bentuk vector berdasarkan nilai rabin karp yang telah diubah menjadi k-Gram. Langkah –langkah dalam klasifikasi dijelaskan sebagai berikut

$$\text{Cosine Similarity} = \frac{\sum_{k=1}^i (\text{weight}_{ik} \text{weight}_{qk})}{\sqrt{\sum_{k=1}^i (\text{weight}_{ik}^2 \text{weight}_{qk}^2)}}$$

#### 3.3 Metode Pengembangan Sistem Prototype

Perancangan Tugas Akhir ini penulis menggunakan metode prototype. *Prototype Model* adalah salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan Metode Prototype ini pengembang dan pengguna dapat saling berinteraksi selama proses pembuatan sistem. Sering terjadi seorang pengguna hanya mendefinisikan secara umum apa yang dibutuhkan, Pemrosesan dan data-data apa saja yang dibutuhkan. Sebaliknya disisi pengembang Kurang memperhatikan efesiensi Algoritma. Kemampuan sistem operasi dan interface yang menghubungkan manusia dengan komputer. Proses pada prototyping bisa dijelaskan sebagai berikut:



**Gambar 3.1 Model Prototyping**

1. Definisi Kebutuhan/ Analisis yaitu developer dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Desain sistem
2. Tahap Desain sistem yaitu melakukan perancangan yang mewakili aspek software yang diketahui. Dan rancangan ini menjadi dasar pembuatan prototype.
3. Implementasi dan Testing Unit yaitu menterjemahkan desain dalam bentuk kode program yang dapat dieksekusi/dimengerti oleh mesin. setelah tahap penerjemahan dalam koding program tahap selanjutnya adalah pengujian untuk mencari eror dalam penulisan kode dihasilkan unit program yang valid.

### 3.4 Perancangan Sistem

Model perancangan yang digunakan pada penelitian ini adalah *Unified Modeling Language* (UML) yang meliputi *use case diagram*, *sequence diagram*, dan *class diagram*, *activity diagram*.

#### 3.4.1 Deskripsi Sistem

Tahap analisa menelaskan proses berjalannya aplikasi serta bagaimana interaksinya dengan pengguna. Tahap awal dalam analisa menjelaskan kebutuhan fungsional pada aplikasi. Kebutuhan fungsional merupakan fitur atau fungsi yang harus ada pada aplikasi. Kebutuhan fungsional dibagi menjadi 3 yaitu untuk Recruiter, Seeker dan Admin.

## Recruiter

1. mengunjungi halaman beranda
2. login dan register
3. memposting lowongan pekerjaan
4. melihat daftar lowongan yang pernah diinput
5. melihat daftar pelamar lowongan

## Seeker

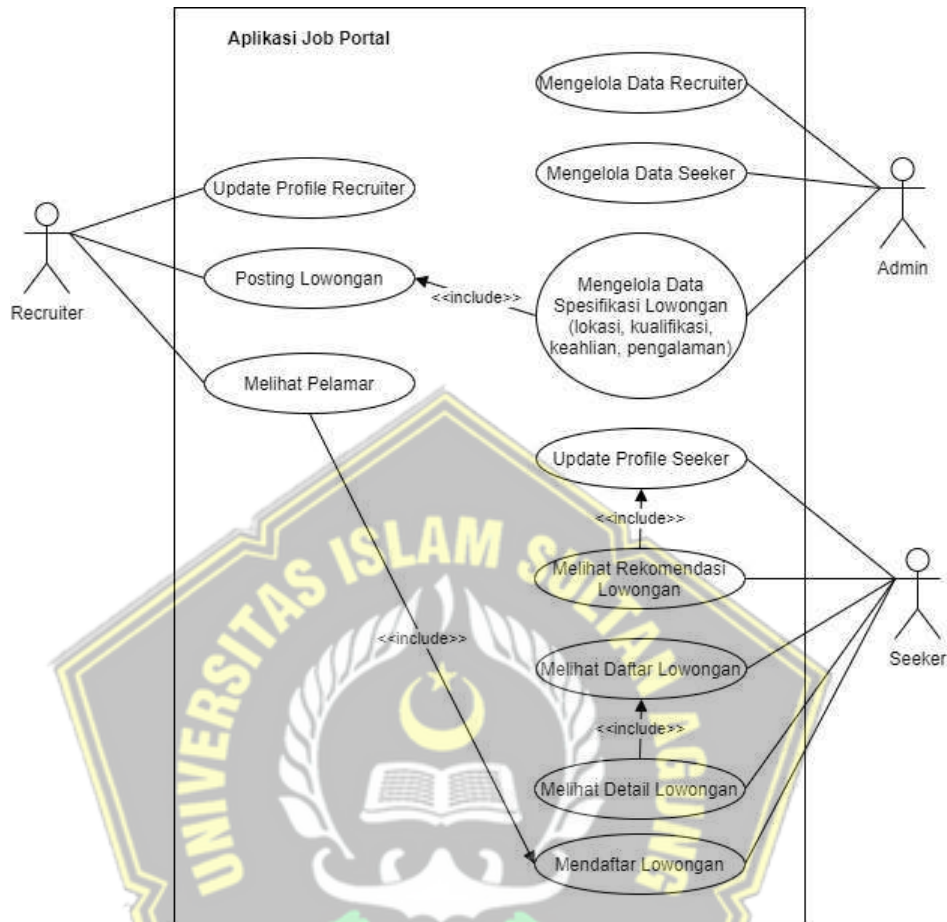
1. mengunjungi halaman beranda
2. login dan register
3. melihat daftar lowongan pekerjaan
4. melihat detail lowongan pekerjaan
5. mendaftar lowongan pekerjaan
6. mengupdate profil
7. melihat rekomendasi lowongan
8. melihat daftar lowongan yang pernah dilamar

## Admin

1. login
2. melihat daftar recruiter
3. melihat daftar seeker
4. mengelola data lokasi
5. mengelola data spesialisasi



### 3.4.2 Use Case Diagram



**Gambar 3.2: Use Case Diagram**

Usecase menggambarkan interaksi antara pengguna yang terdiri dari recruiter, seeker dan admin dengan sistem. Usecase dibuat berdasarkan kebutuhan fungsional yang telah dideskripsikan pada tahap analisa. Usecase diatas dijelaskan pada tabel 4.1 scenario dibawah:

**Tabel 4.1: Deskripsi Use Case**

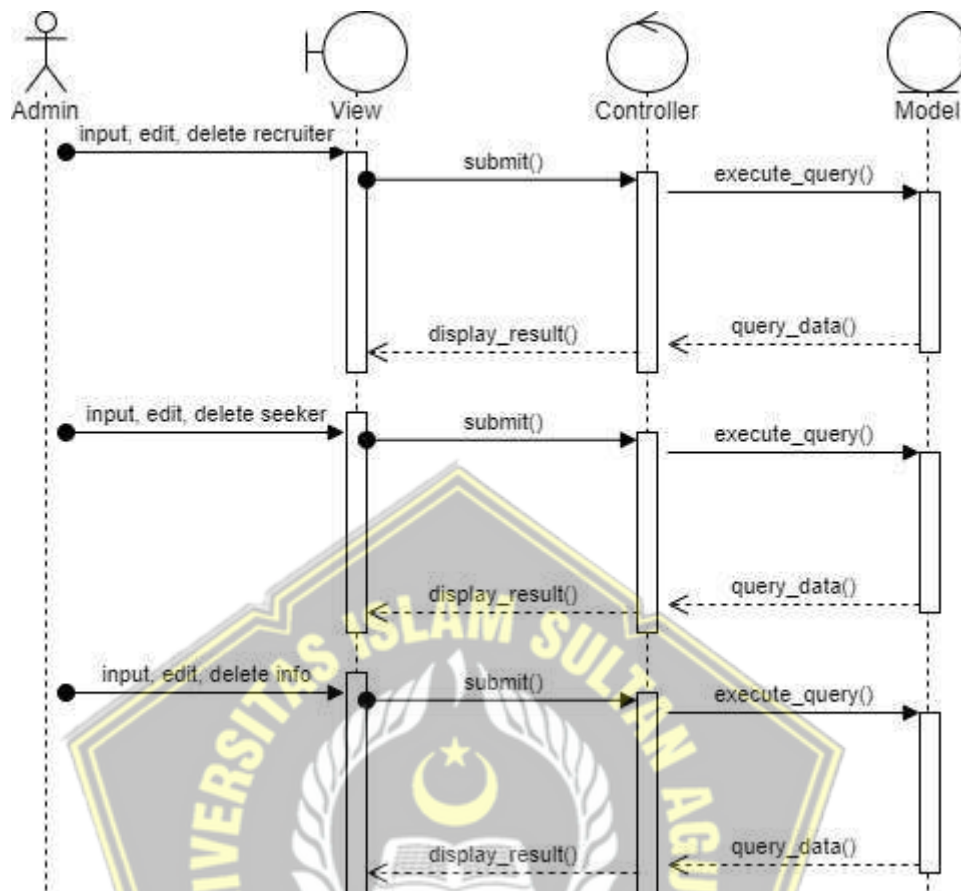
Aktor	Nama Use Case	Deskripsi
Admin	Mengelola data recruiter	Admin dapat mengelola data recruiter. Pengelolaan berupa input, edit dan hapus.
Admin	Mengelola data seeker	Admin dapat mengelola data seeker. Pengelolaan berupa input, edit dan hapus.



Admin	Mengelola data spesifikasi lowongan	Admin dapat mengelola data spesifikasi lowongan pekerjaan. Spesifikasi lowongan diantaranya berupa lokasi, kualifikasi, keahlian dan pengalaman. Pengelolaan berupa input, edit dan hapus.
Recruiter	Update profil recruiter	Recruiter dapat memperbarui profil berupa nama, organisasi, alamat, website dan informasi penting lainnya.
Recruiter	Mengelola data lowongan	Recruiter dapat menambah, mengedit dan menghapus lowongan pekerjaan.
Recruiter	Melihat pelamar	Recruiter dapat melihat data seeker yang melamar pada lowongan yang diposting.
Seeker	Update profile seeker	Seeker dapat memperbarui profil berupa nama, alamat, kualifikasi dan kemampuan yang dikuasai.
Seeker	Melihat rekomendasi lowongan	Setelah mengupdate profile seeker dapat melihat rekomendasi lowongan berdasarkan kemampuan dan data yang telah diinput.
Seeker	Melihat daftar lowongan	Seeker dapat melihat semua lowongan yang diposting oleh recruiter-recruiter.
Seeker	Melihat detail lowongan	Seeker dapat memilih salah satu lowongan dan melihat informasi detail lowongan tersebut
Seeker	Mendaftar lowongan	Seeker dapat mendaftar / apply lowongan pekerjaan yang dikehendaki

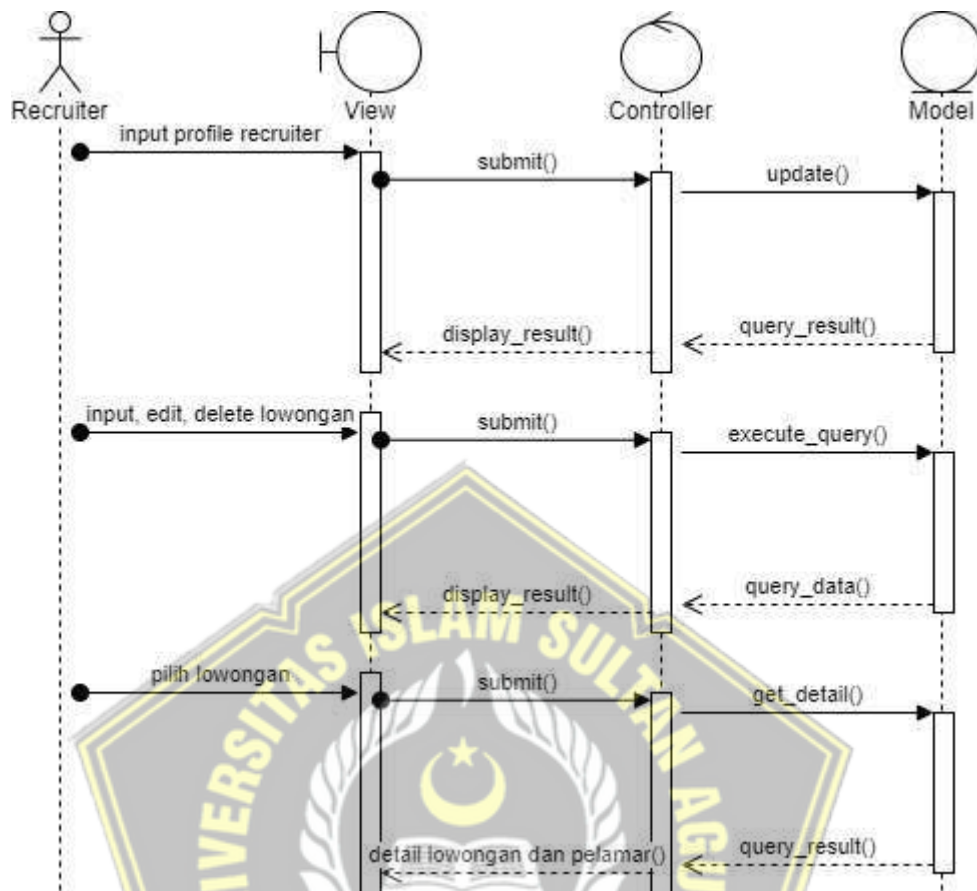
### 3.4.3 Sequence Diagram

Sequence diagram menggambarkan proses secara detail pada fungsi-fungsi yang terdapat pada aplikasi job portal. Sequence diagram dibagi tiap user yaitu sequence diagram untuk seeker, recruiter dan admin.



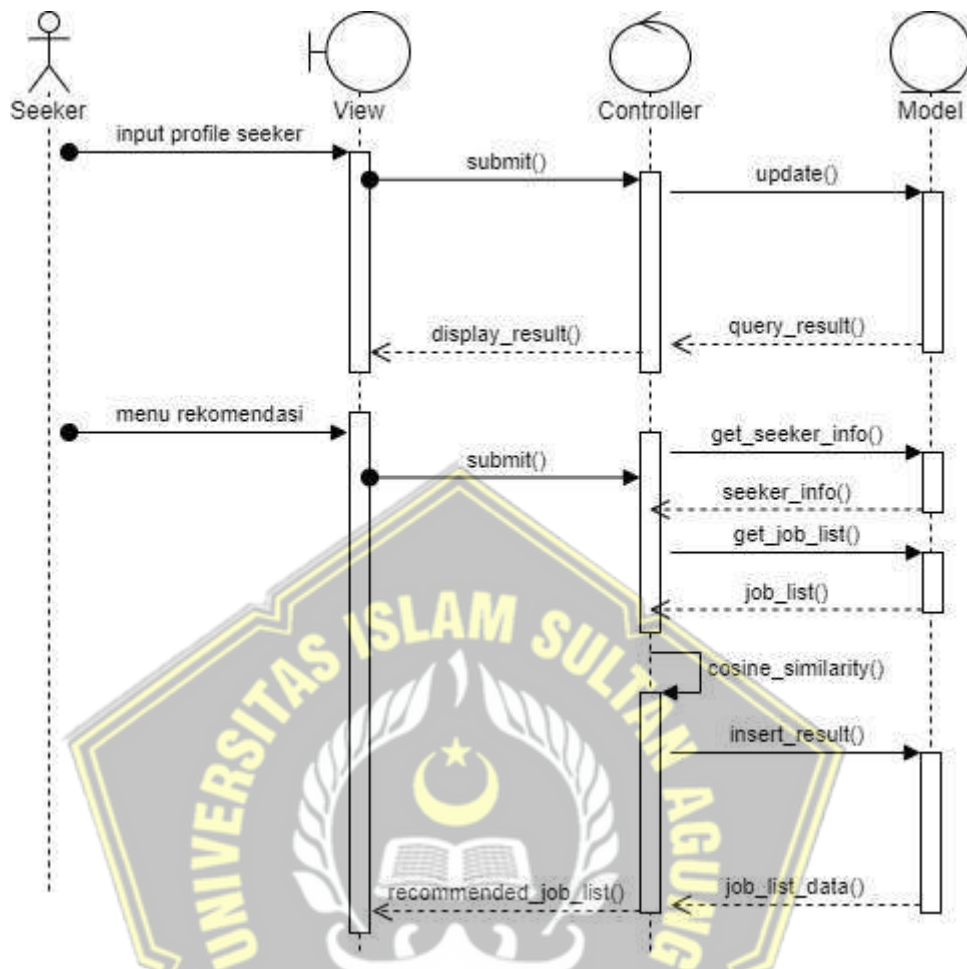
**Gambar 3.3: Sequence Diagram Admin**

Gambar 3.3 ditunjukkan sequence diagram dari admin. Terdapat 3 rangkuman proses utama dalam admin yaitu mengelola data recruiter, mengelola data seeker dan mengelola data info-info. Pada proses tersebut diawali dari aksi admin yang menginput, edit dan delete kemudian pada view melakukan submit, control meneruskan operasi dengan mengeksekusi perintah ke database. Hasil dari eksekusi akan ditampilkan pada halaman sistem.



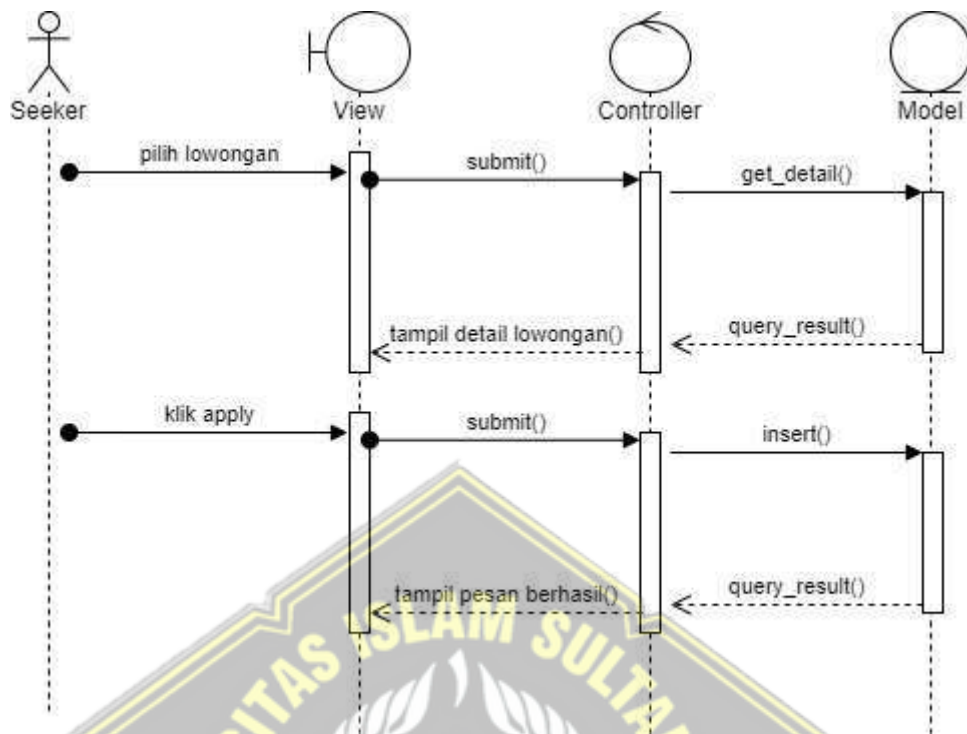
**Gambar 3.4: Sequence Diagram Recruiter**

Gambar 3.4 ditunjukkan sequence diagram yang menunjukkan proses yang dilakukan oleh recruiter. Terdapat 3 proses utama dalam operasi yang dilakukan oleh recruiter. Pertama recruiter menginput profile kemudian submit di view, controller akan mengeksekusi perintah update ke database dan hasil ditampilkan ke view. Recruiter dapat mengelola data lowongan dengan menginput, mengedit dan menghapus. Proses ketiga adalah recruiter memilih lowongan kemudian system akan menampilkan data lowongan beserta pelamar yang telah mendaftar.



**Gambar 3.5: Sequence Diagram Seeker**

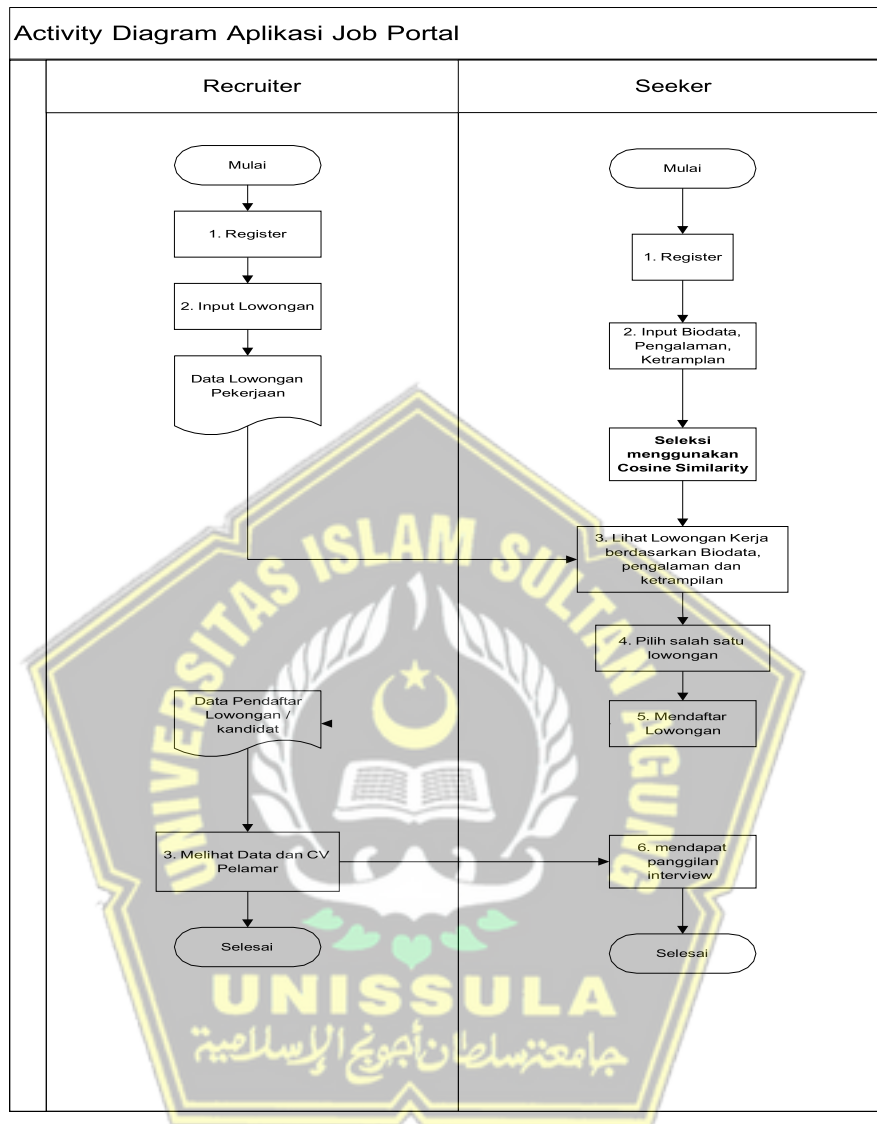
Gambar 3.5 ditunjukkan sequence diagram untuk seeker. Proses awal yaitu seeker menginput profil dan sistem akan mengeksekusi perintah update. Setelah seeker melakukan update profile sistem dapat merekomendasikan daftar lowongan berdasarkan data yang telah diinput. Proses diawali dengan mengambil data seeker dan daftar lowongan di database. Untuk setiap data dicocokkan menggunakan cosine similarity dan hasilnya disimpan di database. Hasil akhir adalah daftar lowongan yang ditampilkan ke seeker diurutkan berdasarkan nilai kecocokan terbesar.



**Gambar 3.6: Sequence Diagram User**

Gambar 3.6 ditunjukkan sequence diagram user pada proses melihat lowongan kerja dan mendaftar lowongan. Seeker memilih lowongan dari daftar lowongan kerja yang ditampilkan aplikasi. Kemudian sistem mengambil data detail dari lowongan. Pada halaman detail seeker dapat mendaftar lowongan dengan menekan tombol apply kemudian aplikasi akan menyimpan data seeker ke lowongan.

### 3.4.4 Activity Diagram



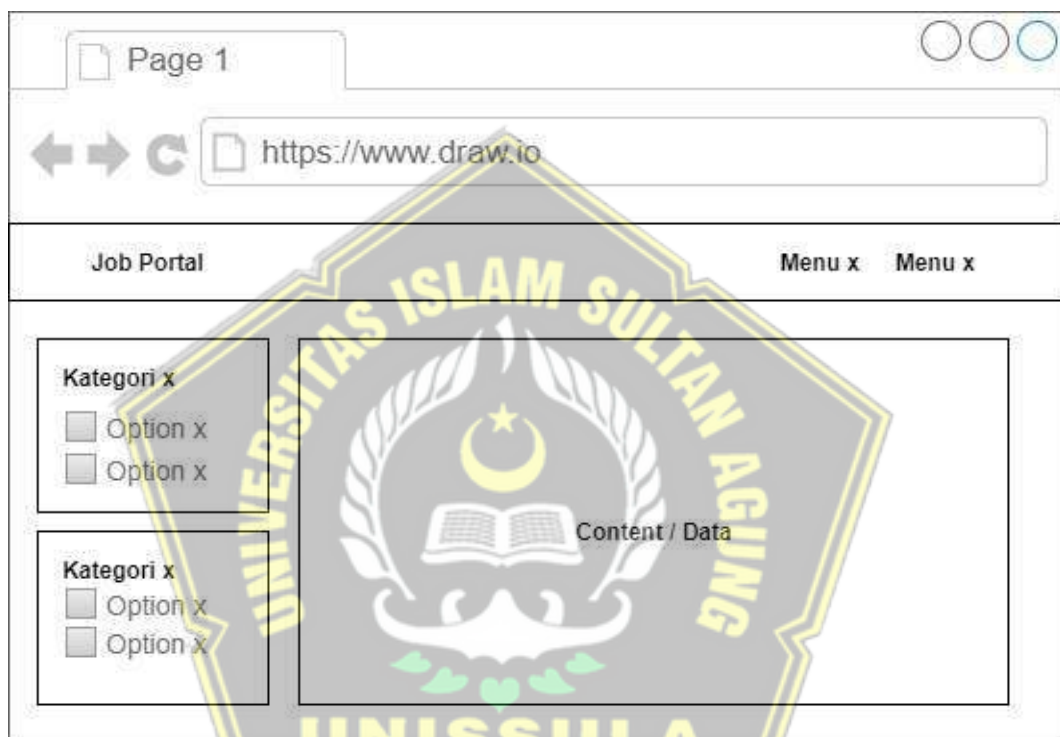
Gambar 3.7: Activity Diagram Aplikasi

Gambar 3.7 menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis pada proses aplikasi job portal. Proses diawali dari recruiter menginput lowongan pekerjaan. Kemudian seeker mendaftar dan menginput data-data yang diperlukan pada aplikasi. Berdasarkan data yang diinput seeker akan dicocokkan dengan data lowongan pekerjaan. Lowongan pekerjaan yang memiliki kecocokan akan ditampilkan. Seeker kemudian melakukan pendaftaran pada lowongan yang dikehendaki. Recruiter dapat melihat daftar seeker yang telah mendaftar pada lowongan.

### 3.4 Perancangan User Interface

Layout aplikasi menggambarkan tampilan aplikasi sebelum diimplementasikan ke bahasa pemrograman. Layout memberikan informasi terkait dengan komponen komponen yang ada tampilan dan tata letaknya.

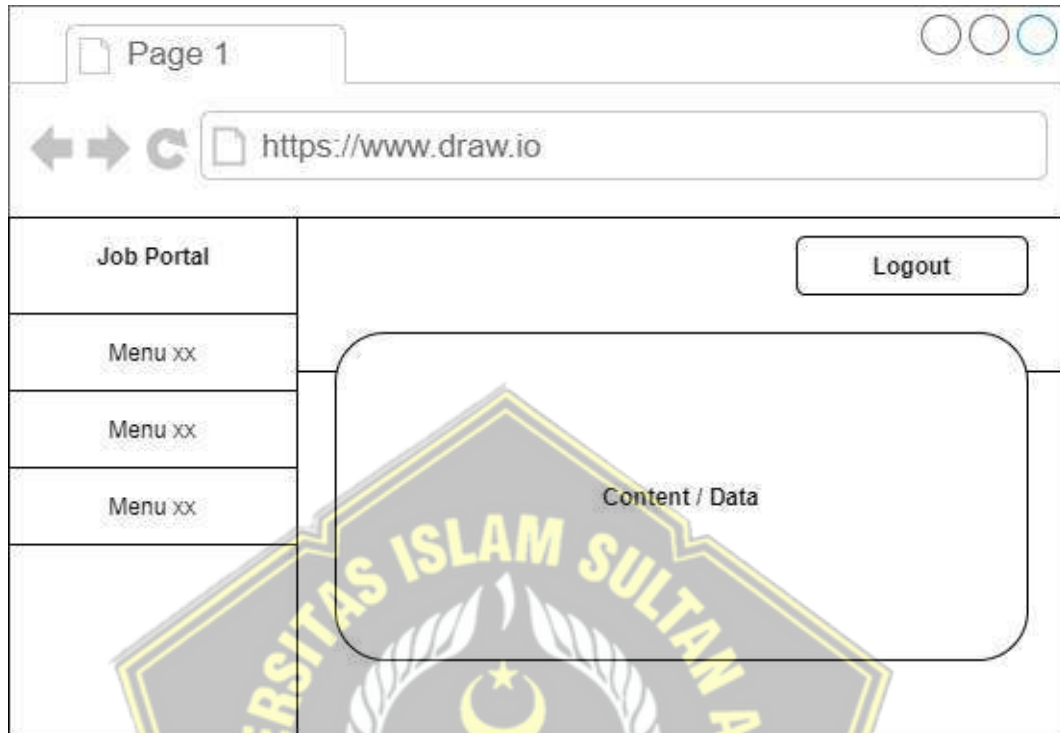
#### 3.4.1 Layout Halaman User (Recruiter & Seeker)



**Gambar 3.8: Layout Halaman Beranda**

Gambar 3.8 ditunjukkan perancangan layout halaman aplikasi untuk recruiter dan seeker. Pada layout tersebut terdapat komponen tampilan yang berupa menu navigasi atas, sidebar menu dan kontainer yang dapat diisi konten atau data yang ditunjukkan oleh seeker dan recruiter.

### 3.4.2 Layout Halaman Admin



**Gambar 3.9: Layout Halaman Admin**

Gambar 3.9 ditunjukkan perancangan layout untuk halaman admin. Pada layout tersebut terdapat komponen tampilan yang meliputi menu *sidebar* (samping) dan container (tengah) berisi inputan-inputan maupun tabel data.



## BAB IV HASIL DAN ANALISIS PENELITIAN

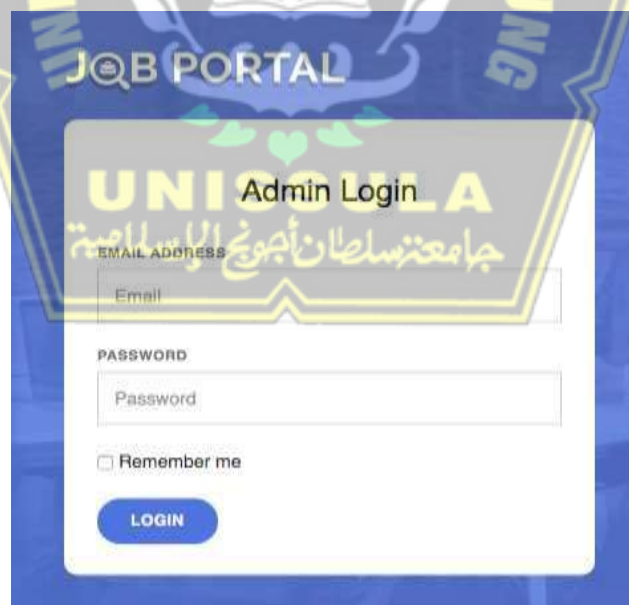
### 4.1 Tahap Implementasi

Pada bab ini dilakukan implementasi dari hasil perancangan pada bab 3. Implementasi membahas hasil aplikasi yang telah diimplementasikan dengan bahasa pemrograman PHP dan MySQL. Terdapat 3 jenis user pada aplikasi job portal terdiri dari admin, recruiter dan seeker.

#### 4.1.1 Admin

Admin merupakan user yang mengelola keseluruhan data-data penting seperti lokasi, recruiter, seeker dan data penting lainnya terkait dengan sistem. Halaman-halaman aplikasi dengan user admin dijelaskan sebagai berikut.

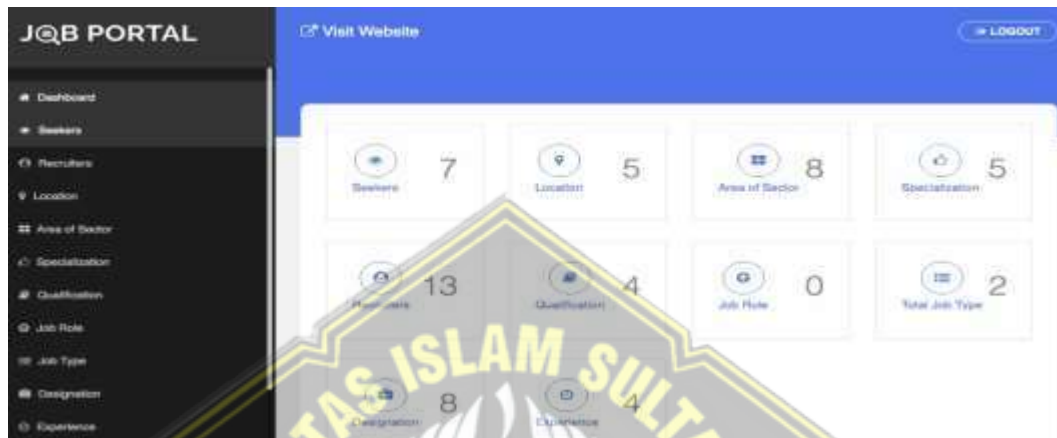
##### 4.1.1.1 Login



**Gambar 4.1: Login Admin**

Gambar 4.1 ditunjukkan halaman login admin. Untuk masuk ke dalam sistem admin memasukkan email dan password. Jika berhasil aplikasi akan mengarahkan ke beranda admin.

#### 4.1.1.2 Beranda Admin



Gambar 4.2: Beranda Admin

Gambar 4.2 ditunjukkan halaman beranda admin. Halaman beranda merupakan halaman awal yang ditampilkan aplikasi ketika admin berhasil memasukkan email dan password dengan benar. Pada sidebar terdapat menu-menu yang dapat digunakan admin untuk mengelola data-data.

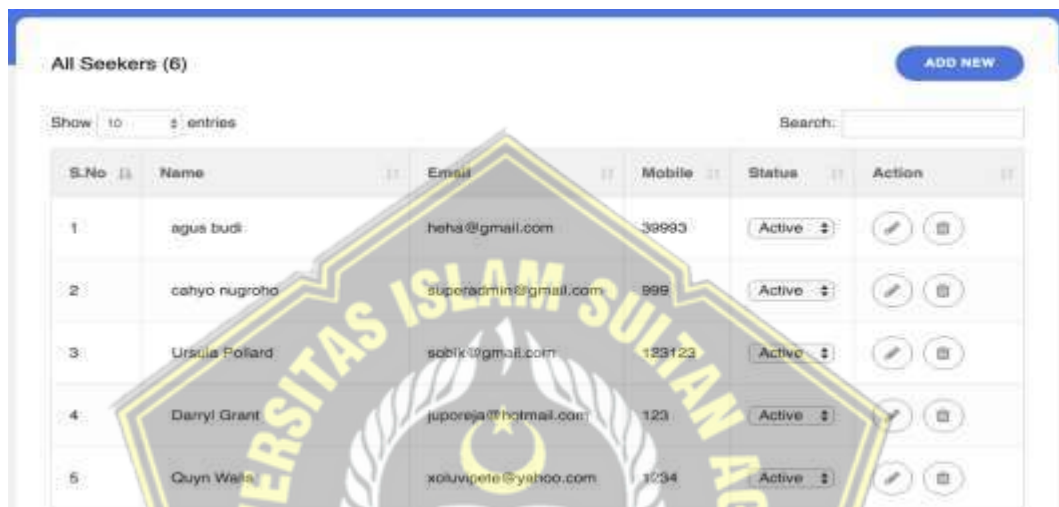
#### 4.1.1.3 Data Recruiter

S.No	Name	Email	Mobile	Status	Action
1	Brian Miranda	yymeto@gmail.com	1234	Active	[Edit] [Delete]
2	Anne Freeman	kuxxywypi@hotmail.com	1818	Active	[Edit] [Delete]
3	Rebecca Farrell	kokaq@yahoo.com	123	Active	[Edit] [Delete]
4	Noel Hendricks	maibe@yahoo.com	181899	Active	[Edit] [Delete]

**Gambar 4.3: Data Recruiter**

Gambar 4.3 ditunjukkan data recruiter. Admin dapat mengelola data dengan menambah, mengedit atau menghapus. Selain itu admin dapat menonaktifkan recruiter.

#### 4.1.1.4 Data Seeker

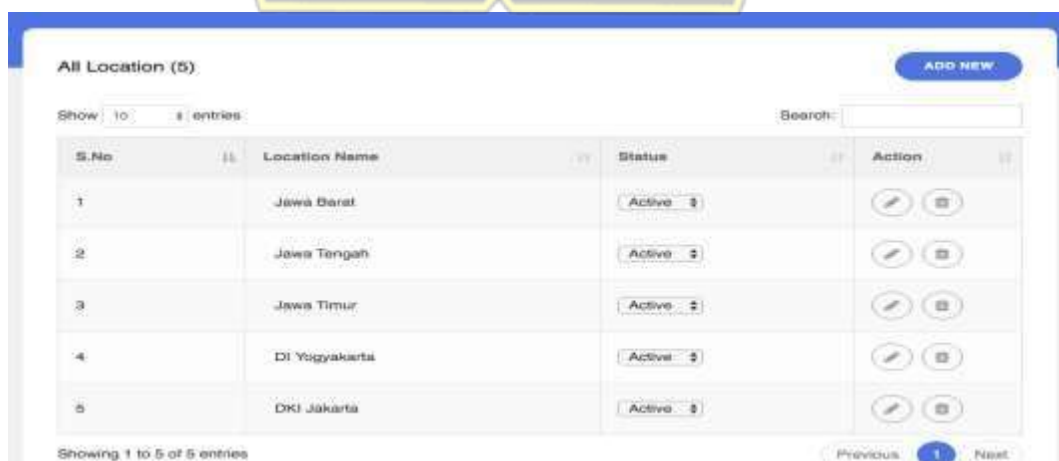


S.No	Name	Email	Mobile	Status	Action
1	agus budi	heha@gmail.com	38993	Active	[Edit] [Delete]
2	cahyo nugroho	superadmin@gmail.com	999	Active	[Edit] [Delete]
3	Ursula Polard	sobik@gmail.com	123123	Active	[Edit] [Delete]
4	Darryl Grant	juporeja@hotmail.com	123	Active	[Edit] [Delete]
5	Quyn Wala	xoiuvipete@yahoo.com	1034	Active	[Edit] [Delete]

**Gambar 4.4: Data Seeker**

Gambar 4.4 ditunjukkan data seeker. Admin dapat mengelola data dengan menambah, mengedit atau menghapus. Selain itu admin dapat menonaktifkan seeker.

#### 4.1.1.5 Data Lokasi

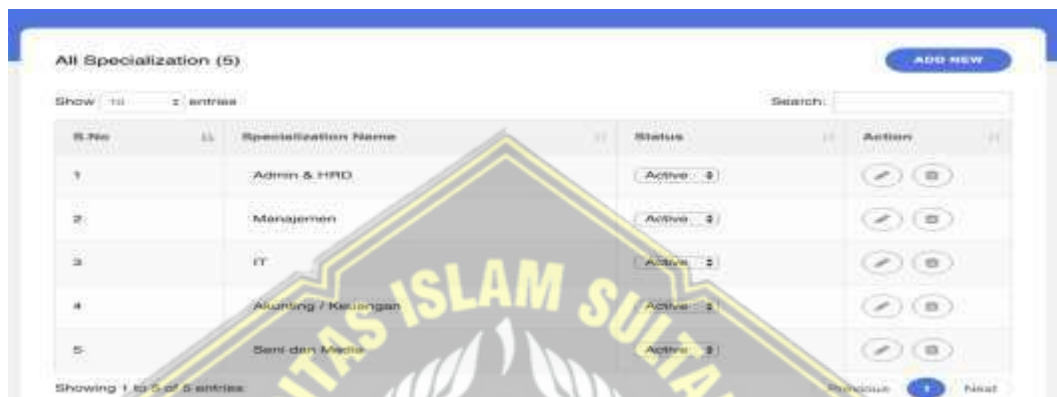


S.No	Location Name	Status	Action
1	Jawa Barat	Active	[Edit] [Delete]
2	Jawa Tengah	Active	[Edit] [Delete]
3	Jawa Timur	Active	[Edit] [Delete]
4	DI Yogyakarta	Active	[Edit] [Delete]
5	DKI Jakarta	Active	[Edit] [Delete]

**Gambar 4.5: Data Lokasi**

Gambar 4.5 ditunjukkan data lokasi yang digunakan untuk pilihan pada data lowongan pekerjaan. Admin dapat mengelola data dengan menambah, mengedit atau menghapus.

#### 4.1.1.6 Data Spesialisasi



S.No	Specialization Name	Status	Action
1	Adm & HRD	Active	[Edit] [Delete]
2	Manajemen	Active	[Edit] [Delete]
3	IT	Active	[Edit] [Delete]
4	Akuntansi / Keuangan	Active	[Edit] [Delete]
5	Seni dan Media	Active	[Edit] [Delete]

**Gambar 4.6: Data Spesialisasi**

Gambar 4.6 ditunjukkan data spesialisasi yang digunakan untuk pilihan pada data lowongan pekerjaan. Admin dapat mengelola data dengan menambah, mengedit atau menghapus.

#### 4.1.2 Recruiter

Recruiter merupakan user yang dapat memposting lowongan pekerjaan dan melihat pelamar pekerjaan atau seeker. Recruiter merupakan jenis user umum yang dapat mendaftar melalui aplikasi.

#### 4.1.2.1 Beranda Pengunjung (Umum)



**Gambar 4.7: Halaman Pengunjung Umum (Sebelum Login)**

Gambar 4.7 ditunjukkan halaman beranda untuk user umum. Halaman tersebut ditampilkan ketika user mulai mengakses aplikasi.

#### 4.1.2.2 Login Recruiter



**Gambar 4.8: Login Recruiter**

Gambar 4.8 ditunjukkan halaman login recruiter. Recruiter melakukan login dengan memilih jenis user recruiter pada pilihan radio button dan memasukkan email beserta password.

#### 4.1.2.3 Update Profil Recruiter

Change Image  
Baker Adams

ORGANISATION TYPE:  
 COMPANY  
 CONSULTANT

E-MAIL:  
recruiter@gmail.com

NAME:  
Baker Adams

CITY:  
Subaya

ADDRESS:  
Pasar Maman Blok T No 3, Sanabaya 60123

WEBSITE:  
anigamov.com

ORGANISATION DETAILS:  
PT. Saja

Save Changes

Gambar 4.9: Update Profile Recruiter

Gambar 4.9 ditunjukkan halaman update profile recruiter. Ketika recruiter akan memposting lowongan pekerjaan langkah awal adalah dengan memperbarui profil recruiter.

#### 4.1.2.4 Posting Lowongan

Post a Job  
UNISSULA  
جامعة سلطان أبوبوع الإسلامية

JOB TITLE:  
Full Time

DURATION:  
1

FIELD OF STUDY:  
Staff IT / Jaringan

PERCENTAGE/COMP:  
3.1

AREA OF EXPERTISE:  
Staff IT / Jaringan

NUMBER OF VACANCIES:  
1

LAST DATE OF APPLICATION:  
2020-07-22

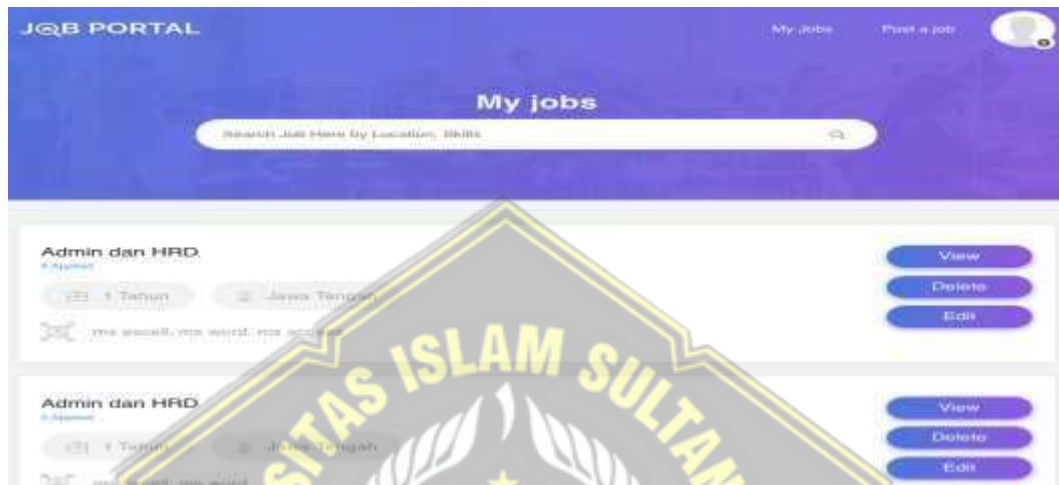
TECHNOLOGY:  
Software, Hardware, Basic, Network

Post a Job

Gambar 4.10: Input Lowongan

Gambar 4.10 ditunjukkan halaman input lowongan oleh recruiter. Pada halaman tersebut ditunjukkan form-form input terkait dengan informasi lowongan pekerjaan yang dibutuhkan.

#### 4.1.2.5 Daftar Postingan



Gambar 4.11: Daftar Postingan Lowongan

Gambar 4.11 ditunjukkan halaman lowongan yang pernah diinput recruiter. Recruiter dapat mengelola data lowongan dengan melihat, menghapus dan mengedit.

#### 4.1.2.6 Daftar Pelamar



Gambar 4.12: Daftar Pelamar dan Detail Pelamar

Gambar 4.12 ditunjukkan halaman pelamar yang telah mendaftar lowongan yang telah diinput oleh recruiter. Recruiter dapat melihat detail informasi pelamar dan mendownload cv yang diupload seeker.

### 4.1.3 Seeker

Seeker merupakan user yang dapat mendaftarkan diri / apply lowongan pekerjaan dari lowongan yang telah diinput oleh recruiter.

#### 4.1.3.1 Login Seeker

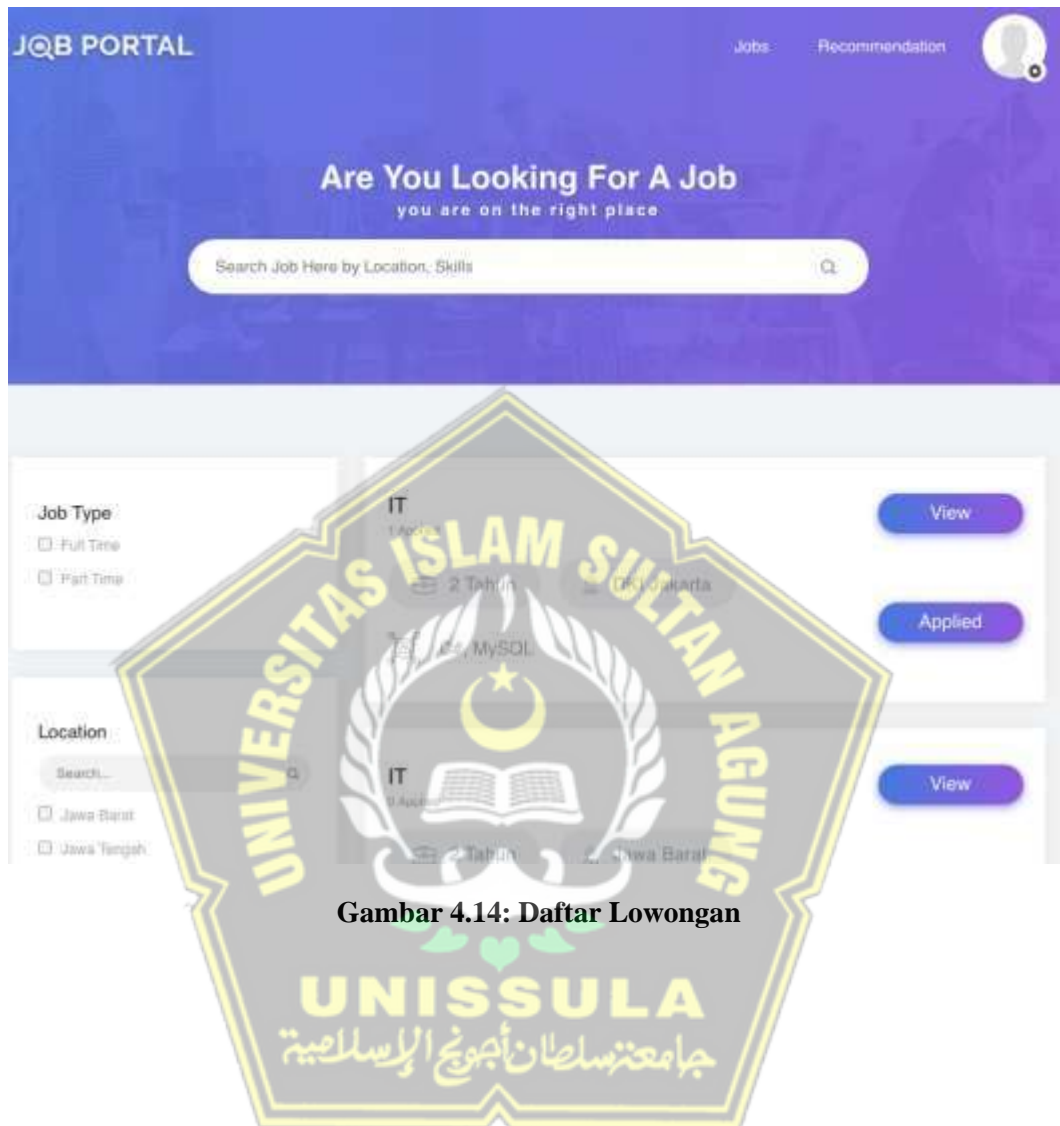


**Gambar 4.13: Login Seeker**

Gambar 4.13 ditunjukkan halaman login seeker. Seeker melakukan login dengan memilih jenis user seeker pada pilihan radio button dan memasukkan email beserta password.

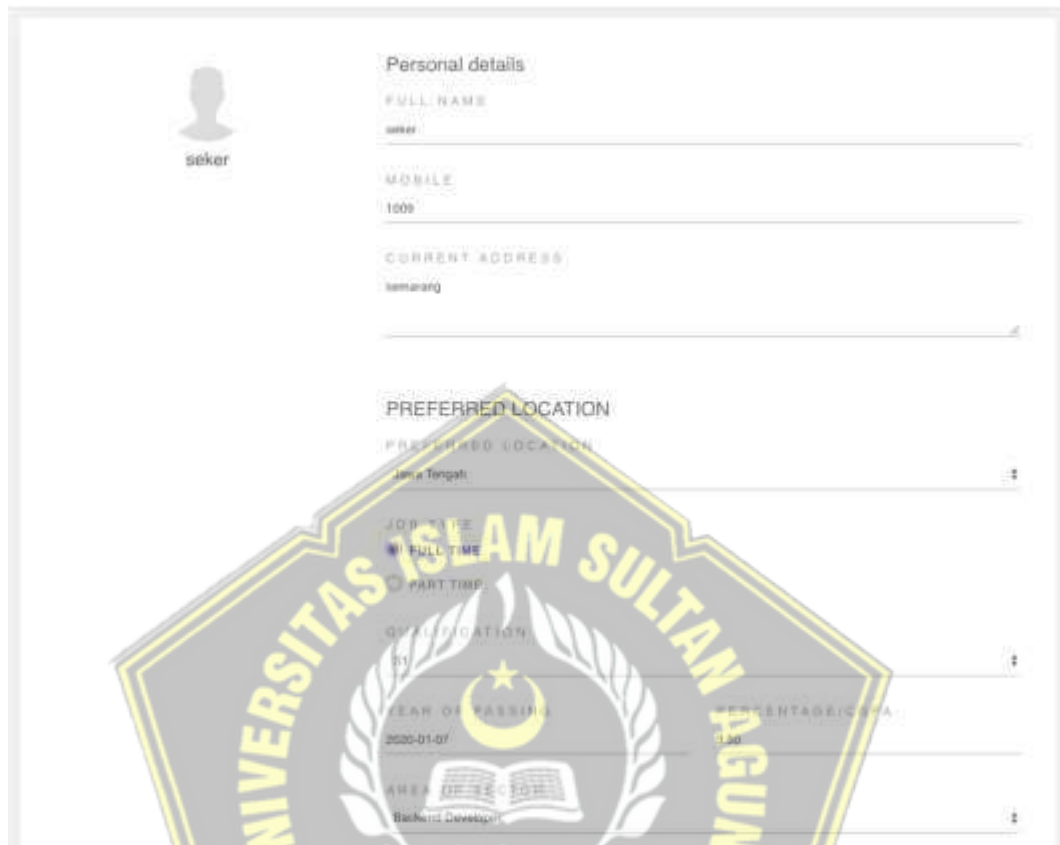


#### 4.1.3.2 Daftar Lowongan Pekerjaan



Gambar 4.14: Daftar Lowongan

### 4.1.3.3 Update Profil Seeker



**Personal details**

FULL NAME  
seker

MOBILE  
1000

CURRENT ADDRESS  
semarang

**PREFERRED LOCATION**

PREFERRED LOCATION  
Jawa Tengah

**JOB TYPE**

FULL TIME  
 PART TIME

**QUALIFICATION**

BY  
YEAR OF PASSING  
2020-01-07

PERCENTAGE/GRADE  
100

**AREA OF EXPERTISE**  
Business Development

Gambar 4.15: Update Profil Seeker

#### 4.1.3.4 Rekomendasi Pekerjaan



Gambar 4.16: Rekomendasi Lowongan

### 4.1.3.5 Detail Lowongan

**IT**  
Organisation - Baker Adams

2 Tahun Jawa Barat

node.js, mysql, aws, git

Job Posted:	1st March 2020
Job Type :	Full Time
Designation :	Backend Developer
Qualification :	S1
Specialization :	IT
Last Date of Application :	2020-03-13
Minimum Salary :	10,000,000/Per Month
Maximum Salary :	14,000,000/Per Month

**Hiring Process**  
Technical Round  
HR Round

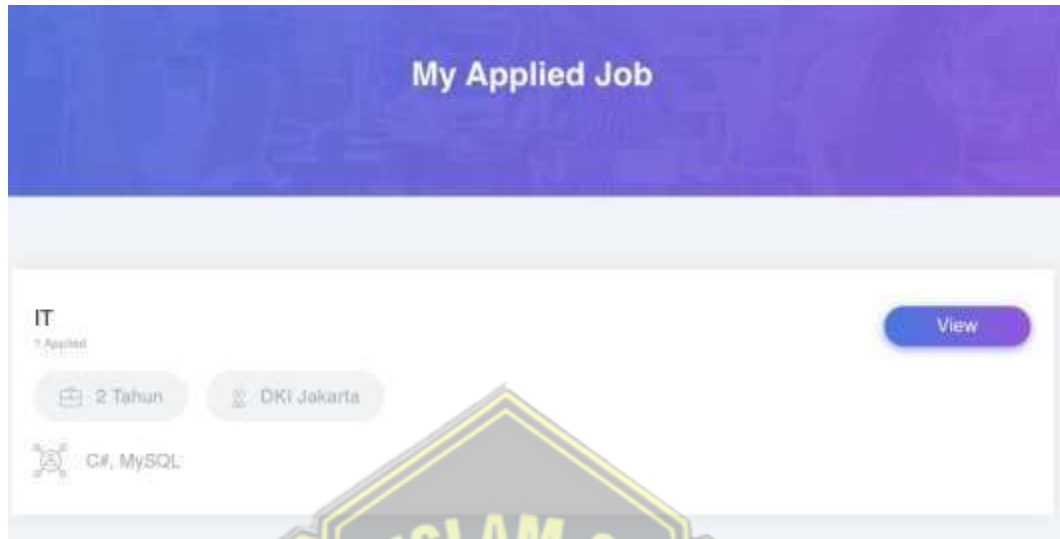
**Job Description**  
Backend Developer, NodeJS Programmer, DDP, Design Patterns, REST API, Asynchronous Programming, and other programming concepts, AWS/GCP, On NoSQL databases (MySQL, MongoDB, etc), Unit testing, Scrum methodology, knowledge/experienced in front end framework (ReactJS, React Native, Vue, Angular is a plus), Familiar with CI/CD.

**Organisation Details**  
Type: Consultant  
Location : Surabaya  
Website : anugrahcv.com

Apply

Gambar 4.17: Detail Lowongan

#### 4.1.3.6 Lowongan yang di Apply



Gambar 4.18: Lowongan yang di Apply

## 4.2 Tahap Pengujian

### 4.2.1 Pengujian Blackbox

Metode pengujian yang di gunakan untuk menguji aplikasi adalah pengujian blackbox. Pengujian blackbox dilakukan untuk memastikan tanggapan/respons atas suatu event atau masukan akan menjelankan proses yang tepat dan menghasilkan keluaran/output sesuai dengan rancangan yang telah dibuat. Uji ini dilakukan pada menu utama dengan hasil sebagai berikut

Tabel 4.13: Pengujian Blackbox

Input / Event	Output	Hasil
Klik tombol Login (masukan data username dan password Valid)	Menampilkan Login Sukses dan menampilkan Form Menu Utama	Sesuai / OK

Klik tombol Login (ketika data username atau password salah)	Menampilkan pesan Login salah.	Sesuai / OK
Klik Tombol Register (ketik nama, email, password)	Menampilkan register sukses dan redirect ke login	Sesuai / OK
Klik tombol detail lowongan	Menampilkan Detail lowongan	Sesuai / OK
Klik Menu Rekomendasi	Menampilkan Pekerjaan yang direkomendasikan	Sesuai / OK
Klik Detail Pekerjaan	Menampilkan Detail Pekerjaan	
Klik Lamar (ketik nama, subjek dan upload CV)	Berhasil melamar pekerjaan	Sesuai / OK
Klik Update Profil (ketik biodata dan keahlian)	Menampilkan berhasil update profil	Sesuai / OK
Klik Daftar Postingan	Menampilkan daftar postingan	Sesuai/ OK
Klik Detail Postingan	Menampilkan detail info postingan dan pelamar	Sesuai/ OK

Klik Logout	Mengakhiri Progr am / redirect halaman login	Sesuai / OK
-------------	---	-------------

#### 4.2.1 User Acceptance Testing

Asas UAT (User Acceptance Test) adalah suatu proses pengujian yang dilakukan oleh pengguna dengan hasil output sebuah dokumen hasil uji yang dapat dijadikan bukti bahwa software sudah diterima dan sudah memenuhi kebutuhan yang diminta. UAT tidak jauh beda dengan kuisiner pada tahap awal pembuatan aplikasi.

Pada tabel 4.4 ditunjukkan bobot nilai jawaban dari Pertanyaan.

**Tabel 4.16. Bobot Nilai Jawaban**

Jawaban	Bobot
Sangat : Mudah/Bagus/Sesuai/Jelas	5
Mudah/Bagus/Sesuai/Jelas	4
Netral	3
Cukup: Mudah/Bagus/Sesuai/Jelas	2
Sangat: Sulit/Jelek/Tidak Sesuai/Tidak Jelas	1

Pada tabel 4.5 ditunjukkan pertanyaan yang telah dijawab oleh 4 user sebagai bahan pengujian.

**Tabel 4.17. Hasil Kuisiner**

Indikator No	Indikator	User 1	User 2	User 3	User 4	Rata-rata
1	Apakah sistem mudah digunakan	3	3	4	4	3.5
2	Apakah sistem sangat membantu dalam mencari referensi lowongan pekerjaan	4	3	3	3	3.25
3	Apakah informasi di dalam sistem lengkap	3	3	3	3	3
4	Apakah tampilan menu memudahkan user menjalankan sistem	4	4	5	5	4.5
5	Apakah sistem membantu dalam merekomendasikan user mencari lowongan	5	5	4	4	4.5

	pekerjaan					
6	Apakah sistem mempunyai tampilan yang baik	4	5	5	4	4.5
7	Apakah informasi yang dihasilkan sistem sesuai kebutuhan user	4	4	4	4	4
8	Apakah sistem ini cukup interaktif	4	4	3	3	3.5
TOTAL AKHIR		31	31	31	30	
GRAND TOTAL		30.75				

Tabel 4.12. Skor pada *form* lembar validasi

SKOR	NILAI
$1 \leq n \leq 10$	Tidak baik
$11 \leq n \leq 20$	Cukup
$21 \leq n \leq 30$	Baik
$31 \leq n \leq 40$	Sangat baik

Dari hasil diatas dapat dilihat bahwa untuk pertanyaan indikator no 1 mendapatkan rata – rata 3,5 dan indikator nilai yang didapat adalah “Baik”, yang berarti sistem mudah untuk digunakan bagi user.

Untuk pertanyaan indikator no 2 mendapatkan rata-rata 3,25 dan indikator nilai yang didapat adalah “Cukup”, yang berarti aplikasi cukup membantu dalam mencari referensi lowongan pekerjaan.

Untuk pertanyaan indikator no 3 mendapatkan rata–rata 3 dan indikator nilai yang didapat adalah “Cukup”, yang berarti *user* dapat memperoleh informasi yang cukup pada sistem.

Untuk pertanyaan indikator no 4 mendapatkan rata – rata 4.5 dari indikator nilai yang didapat adalah “Sangat Mudah”, yang berarti user dapat menjalankan sistem dengan mudah karena tampilan menu yang mudah di pahami.

Untuk pertanyaan indikator no 5 mendapatkan rata-rata 4.5 dari indikator nilai yang didapat adalah “Sangat Baik”, yang berarti user sangat terbantu dengan melihat informasi rekomendasi lowongan pekerjaan.



Untuk pertanyaan indikator no 6 mendapatkan rata-rata 4.5 dari indikator nilai yang didapat adalah “Sangat Baik”, yang berarti *user* menyukai tampilan sistem yang baik.

Untuk pertanyaan indikator no 7 mendapatkan rata-rata 4 dari indikator nilai yang didapat adalah “Baik”, yang berarti *user* mendapatkan informasi yang sesuai dengan kebutuhan.

Untuk pertanyaan indikator no 8 mendapatkan rata-rata 3,5 dari indikator nilai yang didapat adalah “Cukup”, yang berarti *user* puas karena sistem ini cukup interaktif.

Dari total nilai seluruh calon *user* yang mengisi angket senilai 123 poin, dengan rata-rata 30,75 dan didapatkan indikator “Baik”. Yang berarti *user* merasa system yang dibuat sudah layak untuk dikembangkan lebih lanjut.



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan uraian dari bab-bab diatas dan setelah dilakukan ujicoba lapangan hasil dari Sistem Rekomendasi Informasi Lowongan pekerjaan dengan metode cosine similarity & rabin karp k-gram kepada calon user, dapat disimpulkan yaitu Sistem Rekomendasi Informasi Lowongan pekerjaan dengan metode cosine similarity dan rabin karp k-gram yang dibangun berhasil diimplementasikan dalam bentuk website menggunakan bahasa pemrograman PHP dan MySQL. Implementasi dalam bentuk website akan mudah diakses oleh pengguna (admin dan user) melalui perangkat yang tersambung internet dan memiliki browser. Metode yang diusulkan telah berhasil membuat rekomendasi lowongan pekerjaan berdasarkan preferensi pencari kerja yaitu keahlian, jurusan kuliah, jenjang pendidikan dan kota yang diinginkan.

#### **5.2 Saran**

Hasil dari penelitian ini masih jauh dari kata sempurna. Tentunya memiliki kekurangan yang dapat dijadikan bahan evaluasi dan pengembangan di penelitian selanjutnya. Saran untuk penelitian selanjutnya yaitu:

1. Penelitian lebih lanjut dapat menerapkan dan melakukan optimasi terhadap metode system rekomendasi. Dalam proses rekomendasi informasi lowongan pekerjaan masih terdapat kelemahan yaitu proses pencocokan masih menggunakan raw text matching sehingga dapat terjadi ketidakakuratan pada saat pencocokan kata-kata.

2. Dikembangkan penelitian yang lebih mendalam dan variasi atau kombinasi algoritma pada sistem rekomendasi seperti collaborative filtering.



## DAFTAR PUSTAKA

- Anggono, A., 2015. Sistem Rekomendasi pada Portal Lowongan Kerja Menggunakan Metode Simple Additive Weighting.
- Ariantini & Rai, D. A., 2016. Pengukuran Kemiripan Dokumen Teks Bahasa Indonesia Menggunakan Metode Cosine Similarity. *urnal Teknik Informatika 9.1*.
- Doddi Aria, P. & Sujaini, H., 2015. Implementasi Algoritma Rabin-Karp untuk Membantu Pendeteksian Plagiat pada Karya Ilmiah. *Jurnal Sistem dan Teknologi Informasi (JustIN) 4.1* , pp. 66-77.
- Ernawati, E. & Johar, A., 2019. Implementasi Metode String Matching Untuk Pencarian Berita Utama Pada Portal Berita Berbasis Android (Studi Kasus: Harian Rakyat Bengkulu).. *Pseudocode* .
- Hadna, N. M. S., 2016. Sistem Rekomendasi Buku Berbasis Data Mining. *on Science Mapping and the Development of Science*.
- Hernowo, D. & Sudaryanto, S., 2016. Perancangan Website Pusat Informasi Kerja Praktik Mahasiswa Universitas Dian Nuswantoro Dengan Sistem Rekomendasi Metode Booyer Moore.
- Lu, J., 2015. Recommender system application developments: a survey.. *Decision Support Systems 74*, pp. 12-32.
- Ricci, . F., Rokach, L. & Shapira, B., 2013. Introductoin to Recommender Systems Handbook. *Springer Science*.
- Sasmito, G. W., 2017. "Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal", pp. 4-6.
- Susilawati, B., 2018. Rancang Bangun Sistem Informasi Lowongan Pekerjaan Berbasis Web. *Jurnal Cendikia* .

