
PERANCANGAN APLIKASI PENDETEKSI PLAGIARISME TEKS MENGUNAKAN METODE *RATIONAL UNIFIED PROCESS* PADA APLIKASI E-LEARNING

Indri Ariyanti¹, M Aris Ganiardi², Delta Khairunnisa³

^{1,2,3}Jurusan Manajemen Informatika, Politeknik Negeri Sriwijaya, Jalan Sriwijaya Negara, Palembang,
30139, Indonesia
e-mail: indri@polsri.ac.id.

Abstrak

Salah satu fitur yang dimiliki oleh sebuah perangkat lunak *e-learning* adalah tugas kuliah yang mana fitur ini berfungsi sebagai media pengumpulan tugas mahasiswa. Mahasiswa tinggal mengunduh file jawaban yang diberikan oleh dosen. Diperlukan suatu spesifikasi khusus yang harus dimiliki oleh fitur ini yaitu perangkat lunak harus mendeteksi kesamaan jawaban yang dikirim oleh mahasiswa. Dengan adanya spesifikasi ini tentu akan memudahkan dosen mengoreksi jawaban mahasiswa. Pembentukan profil karakter *n-gram* dapat digunakan untuk mendapatkan ciri gaya penulisan. Perbandingan profil *n-gram* dapat digunakan untuk mendeteksi plagiarisme (kesamaan) pada teks. Pendeteksian plagiarisme pada dokumen teks dapat dilakukan dengan sistem pengenalan secara komputerisasi. Secara umum tahap pendeteksian plagiarisme sebagai berikut : prapengolahan, pembagian teks, pembentukan model *n-gram*, dan pendeteksian. Pada penelitian ini mencoba merancang suatu perangkat lunak yang mampu mendeteksi plagiarisme pada dokumen teks berupa file jawaban mahasiswa.

Kata Kunci : Pendeteksian Plagiarisme, algoritma *n-gram*, stylometry, *e-learning*.

Abstract

One of the features possessed by an *e-learning* software is lecture assignments, which function as a medium for collecting student assignments. Students just need to download the answer file given by the lecturer. A special specification is needed that must be owned by this feature, namely the software must detect the similarity of the answers sent by students. With this specification, it will certainly make it easier for lecturers to correct student answers. The formation of the *n-gram* character profile can be used to get characteristic writing style. The *n-gram* profile comparison can be used to detect plagiarism (similarity) in text. Plagiarism detection in text documents can be done with a computerized recognition system. In general, the stages of plagiarism detection are as follows: pre-processing, text division, formation of an *n-gram* model, and detection. In this study, trying to design a software that is able to detect plagiarism in text documents in the form of student answer files.

Keywords : Plagiarism Detection, *n-gram* algorithm, stylometry, *e-learning*

1. PENDAHULUAN

Penggunaan aplikasi *e-learning* mulai banyak digunakan dalam pengajaran ditingkat sekolah maupun perguruan tinggi. Keunggulan yang dimiliki oleh sistem pengajaran ini adalah proses pengajaran tidak dibatasi oleh ruang dan waktu. Para dosen bisa berinteraksi dengan mahasiswanya kapan saja dan dimana saja. Keunggulan yang lain para dosen bisa memantau semua kegiatan mahasiswanya.

Salah satu fitur *e-learning* yang banyak diminati oleh para pengajar adalah Tugas Mahasiswa. Fitur ini menyediakan layanan dosen dapat mengirimkan tugas kepada mahasiswa melalui *e-learning* dan mengatur jadwal pengumpulan tersebut melalui sebuah berkas teks dokumen. Mahasiswa yang diberikan tugas tinggal mengunggah dokumen jawab ke *e-learning*. Pada aplikasi *e-learning* tertentu fitur tugas mata kuliah mampu mengidentifikasi

plagiarisme (kesamaan) teks dokumen file yang dikirimkan oleh mahasiswa. Tentu fitur ini sangat membantu dosen dalam menilai hasil jawaban yang dikirimkan oleh para mahasiswa.

Plagiarisme teks, yaitu pengambilan karya orang lain dan menjadikannya seolah-olah karya sendiri. Plagiarisme telah menjadi masalah umum yang sulit dihadapi serta mempengaruhi perkembangan bidang penelitian dan pendidikan di dunia [1]. Oleh karena itu penanganan masalah ini telah mendapat perhatian serius dari pihak akademis dan pengembang perangkat lunak.

Analisis plagiarisme dapat dilakukan dengan dua pendekatan, yaitu; *external plagiarism detection* dan *intristic plagiarism detection* (Stamatatos, 2009). *External plagiarism detection* yaitu membandingkan sebuah dokumen yang dicurigai mengandung plagiarisme dengan sejumlah atau keseluruhan teks asli. Kebanyakan analisis plagiarisme yang telah ada menggunakan cara ini [2]. Lalu cara yang kedua adalah *intristic plagiarism detection* yaitu dengan membandingkan sebagian dokumen dengan keseluruhan isi dokumen. Dengan cara ini diharapkan dapat menemukan anomali dan inkonsistensi penulisan dari dokumen yang dicurigai. Anomali dan inkonsistensi ini disebut *stylistic nature*. Adapun beberapa metode *intristic plagiarism detection* diantaranya; *Complexity Analysis*, *Vector Space Models*, Perbandingan Profil Karakter N-gram, *Ornidal Measure*, dan *Counter Counter Plagiarism Detection*. Metode *intristic plagiarism detection* dipilih karena tidak memerlukan dokumen referensi (pembanding) untuk mendeteksi plagiarisme, sehingga proses pendeteksian menjadi lebih cepat, serta digunakan sebagai proses awal dalam pendeteksian plagiarisme menggunakan metode *external plagiarism detection* [3]. Salah satu metode yang dapat digunakan pada *intristic plagiarism detection* adalah dengan menggunakan n-gram.

Kesamaan teks dokumen atau plagiarisme sangat memungkinkan dalam proses pengumpulan tugas mahasiswa dalam aplikasi *e-learning*. Untuk diperlukan suatu cara bagaimana cara mendeteksi plagiarisme tersebut. Pada penelitian ini permasalahan yang diangkat adalah bagaimana cara merancang suatu fitur aplikasi *e-learning* untuk mendeteksi plagiarisme pada teks dokumen tugas yang dikirim oleh mahasiswa ?

2. TEORI DASAR

a. Konsep *E-Learning*

Salah satu penerapan teknologi informasi di dunia pendidikan adalah sistem *e-learning*. Berbagai instansi pendidikan seperti di perguruan tinggi menggunakan *e-learning* sebagai media pendukung proses belajar mengajar antara mahasiswa dan dosen. Proses belajar mengajar dapat terus dilakukan tanpa dibatasi oleh ruang dan waktu. Definisi *e-learning* adalah sistem pendidikan yang menerapkan aplikasi-aplikasi elektronika untuk mendukung proses belajar mengajar dengan memanfaatkan media internet, jaringan komputer, perangkat lunak, dan komputer tunggal [4].

b. Plagiarisme

Plagiarisme adalah pengambilan suatu pendapat dan menjadikannya seolah-olah karya sendiri. Sebagaimana yang dituliskan dalam KBBI [9]:

“Plagiarisme adalah penjiplakan atau pengambilan karangan, pendapat, dan sebagainya dari orang lain dan menjadikannya seolah karangan dan pendapat sendiri”.

Plagiarisme dapat dibedakan dari beberapa aspek [5] :

1. Berdasarkan aspek yang dicuri, yaitu membedakan plagiarisme berdasarkan objeknya. Contohnya plagiarisme ide, plagiarisme karya tulis, karya musik, data ilmiah, ataupun lain sebagainya.
2. Berdasarkan aspek kesengajaan, yaitu membedakan plagiarisme dengan melihat apakah tindakan plagiarisme tersebut terdapat unsur kesengajaan.
3. Berdasarkan aspek besarnya proporsi, yaitu membagi plagiarisme berdasarkan seberapa besar karya orang lain yang diambil dan terdapat dalam karyanya. Dibagi menjadi 3 yaitu : plagiarisme ringan (kurang dari 30% dari seluruh karya), plagiarisme sedang (30%-70% dari seluruh karya) serta plagiarisme berat (lebih dari 70% dari seluruh karya).
4. Berdasarkan aspek pola. Pembagian ini biasanya terdapat pada plagiarisme karya tulis. Pembagian plagiarisme ini terbagi 2; yang pertama plagiarisme kata-per-kata (*word-by-word*) yaitu mengambil sebagian karya tulis tanpa merubahnya sedikitpun, meskipun telah diubah menjadi bahasa lain. Lalu yang kedua yaitu plagiarisme mosaik. Dilakukan dengan mengambil sebagian kalimat lalu memodifikasinya sehingga tampak berbeda, namun masih memiliki makna yang sama. Plagiarisme mosaik sangat sulit untuk dideteksi.

Plagiarisme pada teks terbagi menjadi dua; Plagiarisme sengaja (*deliberate plagiarism*) dan Plagiarisme tidak-sengaja (*accidental plagiarism*) [6].

Plagiarisme sengaja telah banyak dijumpai dalam karya tulis yang mana merupakan tindakan menyuruh orang lain menulis sebuah karya tulis dan kemudian menyatakan bahwa itu adalah karyanya atau mengutip pendapat dalam sebuah karya tulis tanpa mencantumkan sumbernya termasuk dalam jenis plagiarisme sengaja. Bahkan menurut Booth menyatakan mengambil sebagian besar sebuah artikel tanpa mencantumkan sumber, walaupun ditulis dengan bahasa sendiri juga termasuk plagiarisme sengaja [7].

c. N-gram

N-gram adalah *substring* sepanjang n karakter dari sebuah *string* [8]. *N-gram* merupakan sebuah metode yang diaplikasikan untuk pembangkitan kata atau karakter. Metode *n-gram* ini digunakan untuk mengambil potongan-potongan karakter huruf sejumlah n dari sebuah kata yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen [10].

N-gram dibedakan berdasarkan jumlah potongan karakter sebesar n . Untuk membantu dalam mengambil potongan-potongan kata berupa karakter huruf tersebut, maka dilakukan *padding* dengan *blank* diawal dan diakhir suatu kata. Sebagai contoh : kata "TEXT" dapat diuraikan ke dalam beberapa *n-gram* berikut ("_" merepresentasikan *blank*) :

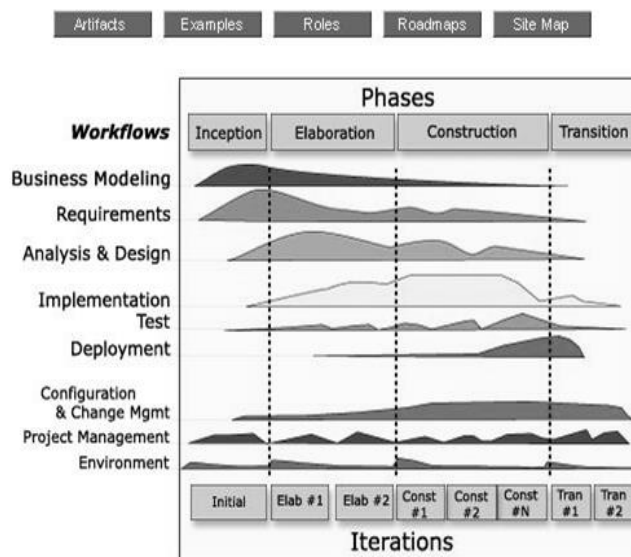
uni-grams : T, E, X, T
bi-grams : _T, TE, EX, XT, T_
tri-grams : _TE, TEX, EXT, XT_
quad-grams : _TEX, TEXT, EXT_
quint-grams : _TEXT, TEXT_

Salah-satu keunggulan menggunakan *n-gram* dan bukan suatu kata utuh secara keseluruhan adalah bahwa *n-gram* tidak akan terlalu sensitif terhadap kesalahan penulisan yang terdapat pada suatu dokumen.

d. Metodologi pengembangan perangkat lunak

Metodologi pengembangan perangkat lunak yang digunakan pada tugas akhir ini adalah *Rational Unified Process* (RUP). Metodologi ini merupakan metodologi pengembangan perangkat lunak yang menggunakan konsep berorientasikan terhadap objek. Objek dapat digambarkan sebagai benda, orang, tempat dan sebagainya yang mempunyai atribut dan metode [11]. Metodologi terdiri dari pembuatan model dari domain aplikasi, kemudian menambahkan rincian implementasi pada saat pembuatan desain dari suatu sistem. Tahap-tahap metodologi berdasarkan SDLC (*System Development Life Cycle*) digunakan dengan memperhatikan karakteristik khusus berorientasi objek.

Rational Unified Process: Overview



3. METODOLOGI PENELITIAN

a. Metode Pengumpulan Data dan Kebutuhan

1) Studi Literatur

Metode ini dilakukan dengan mengkaji beberapa literatur yang berkaitan dengan penelitian perancangan aplikasi pendeteksi plagiarisme teks pada dokumen dengan menggunakan metode *Rational Unified Process* seperti notasi *Unified Modelling Language* (UML) untuk perancangan aplikasi dan algoritma pendeteksi plagiarisme. Literatur-literatur tersebut diperoleh dari :

- Buku-buku dan jurnal-jurnal penelitian baik oleh penulis dari dalam negeri maupun dari luar negeri.
- Data Perguruan Tinggi Negeri dan Swasta yang ada di kota Palembang.
- Informasi dari media masa, seperti surat kabar dan internet.

2) Survei

Pelaksanaan metode ini dengan melakukan survei dan observasi ke perpustakaan-perpustakaan wilayah atau milik perguruan tinggi di kota Palembang untuk melihat bentuk-bentuk dokumen yang dimiliki oleh perpustakaan tersebut.

3) Wawancara

Wawancara dan kuesioner dilakukan langsung kepada para masyarakat yang sering membuat tulisan seperti : penulis artikel di surat kabar, para dosen yang sering membuat jurnal ataupun prosiding, dan mahasiswa yang sedang membuat tugas akhir. Mereka dipilih karena mereka sedang aktif membuat tulisan dalam bentuk berbagai dokumen teks.

b. Analisis dan perancangan aplikasi pendeteksi plagiarisme

Metodologi pengembangan aplikasi pendeteksi plagiarisme teks yang akan digunakan pada penelitian ini adalah metode *Rational Unified Process* (RUP). Ada empat tahapan pada metode ini yaitu : Insepsi, Elaborasi, Konstruksi, Transisi. Tahapan-tahapan yang digunakan pada penelitian ini hanya tahapan insepsi yang fokus pada analisis kebutuhan, algoritma yang digunakan, dan pemodelan sistem. Tahapan elaborasi fokus pada rancangan yang dibuat berdasarkan hasil analisis pada tahapan insepsi.

Berikut ini penjelasan tahapan pengembangan perangkat lunak menggunakan metode RUP.

o Insepsi

1. *Pengumpulan kebutuhan*: data yang dikumpulkan adalah kebutuhan primer dan sekunder. Kebutuhan primer berasal dari data yang diambil langsung dari para *user* dan *stakeholder* pengguna terkait dengan aplikasi pendeteksi plagiarisme teks dan pakar di bidang ilmu pengetahuan. Kebutuhan sekunder berasal dari data buku, internet. Tahap ini sering disebut juga dengan tahapan akuisisi kebutuhan perangkat lunak. Teknik yang digunakan untuk mengumpulkan kebutuhan adalah wawancara, observasi, kuesioner, dan dokumentasi.
2. *Analisis kebutuhan*: Kebutuhan yang telah terkumpul diinterpretasi maksud dari kebutuhan yang didapat. Kebutuhan-kebutuhan yang berasal dari *user* dan *stakeholder* dianalisis untuk menentukan prioritas implementasinya. Interpretasi sebuah kebutuhan dapat berupa data, batasan, aturan, skenario, dan fungsionalitas perangkat lunak.
3. *Analisis basis data*: kasus yang telah terkumpul kemudian dibentuk sesuai dengan format basis data supaya bisa dimasukkan ke dalam basis data seperti pembuatan tabel kasus, pembuatan kunci, relasi antar tabel dan pembentukan *query*.
4. *Analisis perangkat lunak*: perangkat lunak yang digunakan adalah bahasa pemrograman java, IDE Eclipse, DBMS MySQL serta Sistem Operasi Windows 7 (*seven*).
5. *Analisis perangkat keras*: Analisa kebutuhan perangkat keras terhadap sistem seperti kecepatan *processor*, kapasitas memori utama dan memori sekunder.

- **Elaborasi**
 1. *Rancangan basis data*: Rancangan basis data merupakan lanjutan dari analisa basis data. Perancangan basis data dilakukan dengan menggunakan DBMS MySQL serta melakukan pembuatan *query-query* yang nanti akan digunakan oleh sistem.
 2. *Rancangan antarmuka*: Merancang tampilan masukan dan keluaran yang berbasis GUI (*Graphical User Interface*) menggunakan IDE Eclipse.
 3. *Rancangan Perangkat lunak* : Merancang modul-modul program dalam bentuk objek yang nantinya akan digunakan pada saat pengkodean sistem. Rancangan modul dapat berbentuk algoritma, notasi UML dan *pseudo-code*.
 4. Rancangan perangkat keras : Menentukan perangkat keras yang diperlukan dalam implementasi aplikasi pendeteksi plagiarisme teks.

c. Pembahasan dan Evaluasi

Setelah didapatkan hasil rancangan aplikasi pendeteksi plagiarisme teks maka langkah selanjutnya dilakukan evaluasi terhadap hasil rancangan tersebut. Evaluasi dilakukan dengan cara melakukan pengujian terhadap hasil rancangan. Pengujian dilakukan dengan menggunakan tools Qunit yang mana dengan tools tersebut hasil rancangan dapat dikonversikan dalam bentuk koding atau tidak.

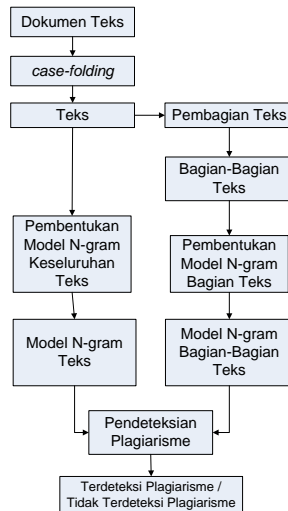
4. HASIL

b. Analisis Pendeteksian Plagiarisme

Setelah model n-gram untuk setiap teks di bentuk. Dilakukan perbandingan gaya bahasa. Perbandingan dilakukan oleh tiap-tiap model n-gram dari bagian teks dengan model n-gram keseluruhan teks. Perbandingan dilakukan dengan Metode CNG (*Common N-Gram*) yang telah dimodifikasi [12]. Perbandingan ini menghasilkan nilai perbedaan gaya bahasa yang bernilai antara 0 dan 1. Dengan nilai 1 berarti memiliki perbedaan gaya bahasa tertinggi dan nilai 0 berarti tidak memiliki perbedaan pada gaya bahasa. Setelah nilai perbedaan gaya bahasa untuk tiap bagian teks didapat, nilai-nilai ini dihitung simpangan baku-nya. Nilai simpangan baku ini diharapkan dapat menunjukkan gaya bahasa yang tidak sesuai dengan gaya bahasa penulis yang merupakan ciri adanya plagiarisme [12]. Dalam jurnal Stamatatos pada tahun 2009, nilai simpangan baku yang dapat dikategorikan sebagai plagiarisme adalah 0.02, nilai ambang batas ini juga digunakan dalam penelitian ini.

c. Deskripsi Umum Perangkat Lunak

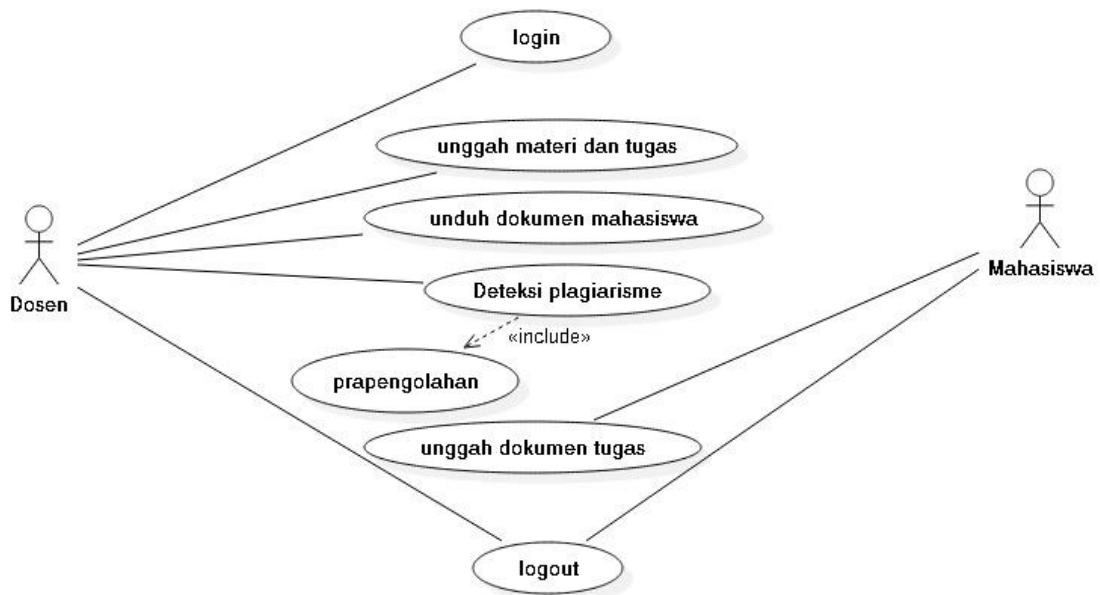
Perangkat lunak yang akan dibangun merupakan perangkat lunak untuk mendeteksi plagiarisme pada dokumen yang berbasis aplikasi web. Teks masukan akan melalui proses pra-pengolahan terlebih dahulu sesuai dengan kebutuhan perangkat lunak. Perangkat lunak ini akan mendeteksi plagiarisme setelah melewati tiap-tiap tahapan yang digunakan yaitu prapengolahan (*case-folding*, pembagian teks, dan pembentukan model n-gram), dan tahap pendeteksian plagiarisme (perhitungan perbedaan gaya bahasa). Untuk lebih jelas dapat dilihat skema umum perangkat lunak yang akan dibangun pada Gambar 2. berikut :



Gambar 2. Skema Umum Perangkat Lunak

d. Diagram Use Case

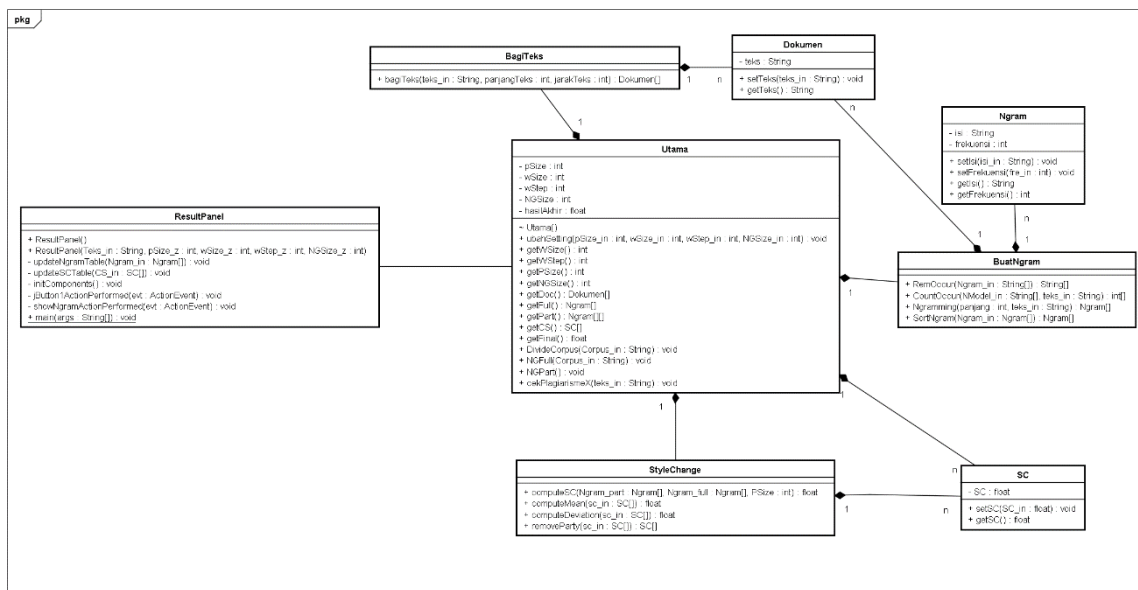
Diagram Use Case menjelaskan secara umum kegiatan yang akan dilakukan oleh aktor (*User*) terhadap sistem yang dapat dilihat pada Gambar 3. berikut :



Gambar 3. Diagram Use Case Sistem Pendeteksian Plagiarisme pada Dokumen

e. Diagram Kelas

Diagram kelas menggambarkan hubungan antar kelas yang terdapat pada sistem. Dari gambar digram kelas tersebut terdapat dua hubungan antar kelas yaitu : asosiasi dan bagian. Hubungan assosiasi menunjukkan kesetaraan antara kelas di dalam sistem sedangkan bagian menunjukkan suatu kelas merupakan bagian dari kelas yang lain.



Gambar 4. Kelas Diagram Pendeteksi Plagiarisme

f. Rancangan Antarmuka

Nama tugas	<input type="text"/>
Deskripsi tugas	<input type="text"/>
Tanggal pengumpulan	<input type="text"/>
Batas pengumpulan	<input type="text"/>
Tanggal pengumpulan	<input type="text"/>
Ukuran file	<input type="text"/>
File tugas	<input type="text"/>
	<input type="button" value="File"/>
	<input type="button" value="Selesai"/> <input type="button" value="Batal"/>

Gambar 5. Antarmuka Unduh Tugas Kuliah Mahasiswa

5.KESIMPULAN

Kesimpulan yang didapatkan selama melakukan penelitian ini adalah

- Aplikasi *e-learning* yang mampu mendeteksi plagiarisme tugas mahasiswa yang membantu dosen untuk menilai hasil tugas yang

- diberikan kepada mahasiswa. Dosen akan sangat menghemat waktu dengan adanya fitur ini.
- b. Algoritma yang digunakan dalam pembuatan model analisis ini adalah algoritma n-gram yang mendeskripsikan secara jelas alur kerja pemrosesan teks pada sebuah dokumen.
 - c. Notasi UML yang digunakan dalam merancang aplikasi ini sangat mudah dipahami sehingga akan mudah diimplementasikan dalam bentuk program.

6. SARAN

Hasil dari analisis dan perancangan yang didapatkan selama penelitian harus diverifikasi dan divalidasi ke pengguna dan stakeholder. Tujuan dari verifikasi dan validasi agar hasil dari analisis dan perancangan ini sesuai dengan keinginan mereka.

DAFTAR PUSTAKA

- [1] Maurer, H. dan Kulathuramaiyer, N. 2007. *Plagiarism and IPR Violation: Why is so Important ?* Learned Publishing, 20(4):13-19.
- [2] Hoad, Timothy C. dan Zobel, J. 2003. *Methods for Identifying Versioned and Plagiarized Documents*. Journal of the American Society for Information Science and Technology, 54(3):203-215.
- [3] Stamatatos, E. 2009. *Intrinsic Plagiarism Detection Using Character n-gram Profiles*. In Proc. of the 3rd Int. Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse. University of the Aegean, Greece.
- [4] Satria Wahono, Romi. 2003. Pengantar e-learning dan Pengembangannya.
- [5] Sastroasmoro, S. 2006. *Beberapa Catatan tentang Plagiarisme*. Departemen Ilmu Kesehatan Anak, Fakultas Kedokteran Universitas Indonesia, Jakarta.
- [6] Wray, A. dan Bloomer, A. 2006. *Projects In Linguistics : A Practical Guide to Researching Language 2nd edition*. Oxford: Oxford University Press.
- [7] Booth, Wayne C. et al. 1995. *The Craft of Research* London: The University of Chicago Press.
- [8] Cavnar, William B., Trenkle John M., 1994. *N-gram based Text Categorization*, Symposium on Document Analysis and Information Retrieval, hal 161-176, University of Nevada, Las Vegas.
- [9] Departemen Pendidikan. 1995. *KBBI*. Jakarta: Balai Pustaka.
- [10] Hassan, Feryal J. Haj, dan Chourasia, Mousmi A. 2012. *N-gram based text Author Verification*. Department of Information Technology, King Saud University, Saudi Arabia.
- [11] Kruchten, P. 2000. *The Rational Unified Process An Introduction*, Second Edition.
- [12] Stamatatos, E. 2009. *Author Identification Using Imbalanced and Limited Training Texts*. In Proceedings of the 4th International Workshop on Text-based Information Retrieval, hal.237-241.