

## Implementasi NLP Untuk Menilai Kualitas SKPL Berdasarkan Karakteristik Modifiable

Hilmy Azhary<sup>1</sup>, Sri Widowati<sup>2</sup>, Jati Hililamsyah Husen<sup>3</sup>

<sup>1,2,3</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>hilmyazhary@students.telkomuniversity.ac.id, <sup>2</sup>sriwidowati@telkomuniversity.ac.id,

<sup>3</sup>jatihusen@telkomuniversity.ac.id

---

### Abstrak

Kebutuhan perangkat lunak adalah pernyataan suatu kebutuhan dan batasan serta kondisinya. Ilmu yang membahas pembentukan kebutuhan disebut dengan *Requirement Engineering (RE)*. Salah satu produk dari RE adalah SKPL (Spesifikasi Kebutuhan Perangkat Lunak). Untuk memastikan bahwa kualitas perangkat lunak tinggi, maka kualitas SKPL pun harus tinggi. Perkembangan dalam SKPL selalu terjadi, maka SKPL harus selalu bisa dimodifikasi tanpa mengubah struktur kata dan dokumennya. Dalam standar IEEE hal ini disebut dengan sifat *modifiable*. Untuk mengukur kualitas *modifiable* dari SKPL, maka dalam penelitian ini telah dilakukan pengembangan tools berbasis Python. Setelah kualitas SKPL dihitung maka akan dilakukan pengujian program dengan perbaikan manual terhadap SKPL tersebut dan dibandingkan dengan hasil penghitungan awal. Hasil yang didapat adalah nilai kualitas *Modifiable* SKPL dalam penggunaan bahasa natural dan struktur dokumen yang dapat menjadi acuan perbaikan atau perlengkapan SKPL.

**Kata kunci :** Spesifikasi Kebutuhan Perangkat Lunak, *Modifiable*, *Quality Assessment*, *Natural Language Processing*

---

### Abstract

SRS (Software Requirement Specifications) is a document that explains the specifications for the needs of a particular software that performs certain functions in a specific environment. In making SRS, various kinds of errors can occur, one of which is when the document has a writing structure and document structure which is not good. This makes the SRS not modifiable. When modifiable characteristics inside the SRS are not achieved, it is very likely that a software development project will encounter failure when the SRS is updated. Therefore in this study a Python-based program were developed to check the SRS Modifiable quality values automatically. In the program the Quality Assessment method is applied which inside it the application for Natural Language Processing also happened. After SRS quality is calculated, the program will be tested by the means of doing manual repairs to the SRS and compared it with the results of the initial calculation. The results obtained are SRS modifiable quality values and document sections which can be a reference for SRS repair or completion.

**Keywords:** Software Requirement Specifications, *Modifiable*, *Quality Assessment*, *Natural Language Processing*

---

## 1. Pendahuluan

### Latar Belakang

Dalam pengembangan perangkat lunak, terdapat kebutuhan perangkat lunak, yaitu pernyataan suatu kebutuhan dan batasan serta kondisinya [1]. Ilmu yang membahas tentang pendefinisian kebutuhan perangkat lunak yang terjadi antara *stakeholder* dan pengembang disebut dengan *Requirement Engineering (RE)* [1]. Salah satu produk dari RE adalah SKPL (Spesifikasi Kebutuhan Perangkat Lunak), yaitu dokumen yang berisikan kumpulan kebutuhan perangkat lunak yang telah disetujui oleh *stakeholder* dan pengembang [1]. Untuk memastikan bahwa kualitas perangkat lunak tinggi, maka kualitas SKPL dari perangkat lunak tersebut harus baik [2]. Menurut standar IEEE 830 [3], suatu SKPL yang baik mempunyai delapan karakteristik yaitu: *correct*, *unambiguous*, *complete*, *consistent*, *ranked for importance and/or stability*, *verifiable*, *modifiable*, dan *traceable*. Dikarenakan perangkat lunak pasti selalu berkembang dalam proses pembuatannya, maka SKPL pun harus bisa menampung perkembangan tersebut tanpa mengubah struktur SKPL secara drastis [2], ketidakmampuan menampung perkembangan tersebut dapat menyebabkan terjadinya kesulitan dalam memahami dokumen SKPL [4], yang dapat menyebabkan perangkat lunak tidak memenuhi kebutuhan *stakeholder* [4]. Oleh karena itu SKPL harus dapat dimodifikasi dengan baik, dimana dalam standar IEEE 830 [3], sifat ini disebut dengan *modifiable*. SKPL bersifat *modifiable* jika dan hanya jika struktur dan gaya penulisan SKPL tersebut tidak akan berubah ketika

terjadi perubahan pada spesifikasi kebutuhan yang tertulis didalamnya [1]. Untuk mengukur kualitas *modifiable* dari SKPL, maka dalam penelitian ini telah dilakukan pengembangan *tools* berbasis Python.

Beberapa penelitian telah mengusulkan perhitungan kualitas SKPL, tetapi penelitian-penelitian tersebut masih sangat bergantung pada data uji latih [5] [6] atau dalam kasus lain masih harus membutuhkan verifikasi manual oleh manusia [2] [7]. Setelah mempertimbangkan hal tersebut, maka dalam penelitian ini diterapkan metode Quality Assessment dimana kualitas SKPL dinilai berdasarkan penggunaan bahasa alami dan struktur dokumen [4]. Pemilihan metode ini mempunyai kelebihan, yaitu penilaian kualitas bersifat objektif, otomatis, dan pengaplikasian kedalam program nya pun jelas [4]. Kualitas yang akan diukur adalah kualitas sifat/karakteristik *modifiable* dimana akan dicari kata-kata yang membuat SKPL menjadi tidak *modifiable* dan pengecekan keberadaan struktur penting dalam dokumen. Untuk pengujian, dilakukan perbandingan nilai kualitas *modifiable* pada dokumen SKPL sebelum perbaikan dan sesudah perbaikan. Setelah itu, dokumen yang telah diperbaiki diharapkan akan menjadi acuan dan referensi bagi pembuatan SKPL di masa depan yang memiliki kualitas *modifiable* yang baik.

### Topik dan Batasannya

Penelitian ini dilakukan karena terdapat masalah yang sangat mungkin terjadi ketika suatu dokumen SKPL diperbaharui, jika penggunaan bahasa alami dalam dokumen SKPL dan struktur dokumen yang tidak baik [4]. Dalam penelitian ini akan dinilai penggunaan bahasa alami dalam SKPL yang akan dicek menggunakan ilmu *Natural Language Processing* didalam metode *Quality Assessment* dan juga pengecekan struktur dokumen seperti adanya daftar isi, referensi, dan lainnya. Hingga saat ini, penelitian pengukuran kualitas *modifiable* kebanyakan untuk dokumen SKPL berbahasa Inggris, dan untuk Bahasa Indonesia sendiri masih belum ada.

Rumusan permasalahan dalam paper ini adalah :

1. Bagaimanakah hasil dari perhitungan kualitas SKPL pada karakteristik *modifiable* menggunakan metode *Quality Assessment* ?
2. Atribut kualitas manakah yang sangat berpengaruh terhadap nilai kualitas karakteristik *modifiable* pada SKPL?

Untuk batasan masalah pada penelitian ini, kualitas SKPL yang akan diukur hanyalah kualitas *modifiable* saja, pengaplikasian *Natural Language Processing* terbatas pada proses *case folding* dan POS (*Part-of-speech*) *Tagging* dengan menggunakan *library tagset* yang dibuat oleh Fam Rashel, dkk. [8], template SKPL yang diteliti yaitu template SKPL Telkom University dan template SKPL Universitas Muria Kudus, dan *file* SKPL yang digunakan adalah file SKPL Aplikasi SIROTI, Aplikasi SMS, dan Aplikasi Explore Berau.

### Tujuan

Tujuan penelitian ini adalah untuk mengukur kualitas SKPL dengan membuat program berbasis Python untuk mengukur kualitas karakteristik *modifiable* SKPL sehingga bisa menjadi acuan untuk pembuatan SKPL di masa mendatang yang mempunyai kualitas karakteristik *modifiable* yang tinggi.

### Organisasi Tulisan

Pada paper ini, bagian 1 menjelaskan pendahuluan beserta latar belakang, batasan masalah, dan tujuan penelitian, bagian 2 menjelaskan studi terkait untuk metode yang digunakan pada penelitian ini, bagian 3 menjelaskan secara detail sistem yang dibangun, bagian 4 menjelaskan pengujian dari sistem yang dibangun, dan bagian 5 menjelaskan kesimpulan dan saran dari kegiatan penelitian ini.

## 2. Studi Terkait

### 2.1 Requirement Engineering

*Requirement* atau kebutuhan didefinisikan sebagai sebuah pernyataan yang mengidentifikasi karakteristik atau batasan operasional, fungsional, atau desain dalam sebuah produk atau proses yang bersifat tidak ambigu, dapat diuji atau diukur, yang penting untuk penerimaan produk atau proses tersebut oleh pengguna atau petunjuk *quality assurance* [9]. Sementara itu *requirement engineering* atau rekayasa kebutuhan adalah bagian dari rekayasa perangkat lunak yang membahas tentang mencari, mengembangkan, menelusuri, menganalisis, mengkualifikasi, mengkomunikasikan, dan mengurus kebutuhan yang mendefinisikan sistem pada level abstraksi yang berkelanjutan [10]. Pada rekayasa kebutuhan terjadi berbagai aktivitas, mulai dari penerjemahan kebutuhan bisnis, proses pengujian kebutuhan, pengkomunikasian kebutuhan kepada *stakeholder*, yang akan divalidasi dan mengubah kebutuhan tersebut menjadi sebuah solusi.

### 2.1.1 Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

SKPL adalah sebuah dokumen yang berisikan spesifikasi kebutuhan sebuah perangkat lunak yang menjalankan fungsi tertentu dalam suatu lingkungan yang spesifik [1]. Dalam SKPL, suatu spesifikasi kebutuhan dijelaskan secara rinci beserta fungsi, kebutuhan sistem, dan batasan sistem yang bertujuan sebagai panduan dalam mengembangkan suatu perangkat lunak yang sesuai dengan tujuan pembuatan perangkat lunak yang telah dikonsultasikan kepada *stakeholder*. SKPL mempunyai isi sebagai berikut :

1. **Pengenalan**  
Pada bagian ini dijelaskan tujuan pembuatan SKPL, *scope* perangkat lunak, penjelasan singkat perangkat lunak yang dikembangkan, dan definisi yang dipakai dalam SKPL tersebut.
2. **Referensi**  
Pada bagian ini dituliskan semua referensi yang dirujuk pada isi didalam SKPL.
3. **Kebutuhan spesifik**  
Pada bagian ini dituliskan kebutuhan perangkat lunak fungsional dan non fungsional, dimulai dari kebutuhan antarmuka eksternal, fungsi perangkat lunak, kebutuhan usability, kebutuhan performansi, kebutuhan database, batasan desain, atribut sistem perangkat lunak, dan informasi pendukung.
4. **Verifikasi**  
Pada bagian ini semua fungsi yang terdapat pada perangkat lunak dijelaskan secara rinci dengan menyertakan informasi user interface beserta penjelasan kontennya, alur sistem, dan diagram sistemnya.
5. **Appendix**  
Bagian ini berisikan asumsi dan dependensi perangkat lunak serta akronim dan singkatan yang digunakan dalam SKPL.

## 2.2 Natural Language Processing

*Natural Language Processing* (NLP) adalah suatu area penelitian yang mengeksplorasi bagaimana komputer dapat memahami dan memanipulasi bahasa alami baik yang berupa teks ataupun suara untuk melakukan hal yang berguna. NLP bertujuan untuk mengumpulkan ilmu tentang bagaimana manusia memahami dan menggunakan bahasa sehingga dapat digunakan pada komputer agar dapat memahami dan memanipulasi bahasa alami sesuai dengan tujuan penggunaannya. [11]

### 2.2.1 POS (*Part-of-speech*) Tagging

Dalam pemrosesan syntax antar kata, *part-of-speech* adalah suatu informasi tentang informasi suatu kata dan tetangga nya. POS (*Part-of-speech*) Tagging adalah suatu proses untuk memasukkan sebuah *part-of-speech* atau tanda kelas leksikal ke masing-masing kata dalam sebuah korpus [12]. POS Tagging digunakan untuk mengklasifikasikan (memberi *tag*) suatu kata, apakah kata tersebut kata kerja, kata benda, atau sifat lainnya yang berada pada *tagset* yang berkaitan.

Dalam penelitian ini, digunakan *tagset* yang dibuat oleh Fam Rashel, dkk. [8], yang berisikan 250.000 kata dalam Bahasa Indonesia beserta *tag* yang sesuai. *Tagset* ini bersifat gratis dibawah lisensi *Creative Commons Licence*, dengan akurasi POS *Tagger* sebesar 70% [8].

## 2.3 Quality Assessment

*Quality Assessment* adalah suatu teknik yang dirancang dan direkomendasikan oleh Patra Thitisathienkul dan Nakornthip Prompoon dan ditulis dalam paper [4] untuk menilai kualitas dokumen yang dihasilkan pada saat tahapan pengembangan perangkat lunak SDLC dengan cara menilai kualitas konten dan struktur dari suatu dokumen. Pada penerapannya, para penulis menggunakan dokumen SKPL yang difokuskan hanya kepada 3 karakteristik baik SKPL, yaitu *unambiguous*, *verifiable*, dan *modifiable*.

Dalam metode ini diterapkan penggunaan ilmu NLP untuk mengambil dan memproses kata yang nantinya akan diberi nilai per atribut berdasarkan atribut yang bersesuaian dengan karakteristik baik SKPL tersebut. Untuk kasus dalam penelitian ini, atribut *modifiable* dinilai dari penggunaan bahasa alami dan keberadaan struktur dokumen yang diambil dari daftar isi. Dalam penggunaan bahasa alami, sifat *modifiable* dihitung dari penggunaan tanda baca, penggunaan verba dan penggunaan kata hubung. Tanda baca mempengaruhi preferensian suatu kebutuhan condong mereferensikan lebih dari satu kebutuhan dalam satu kalimat, sehingga dapat menjadikannya lebih sulit untuk dipahami sehingga untuk modifikasi SKPL menjadi lebih sulit, contohnya “Satu-satunya potensi kekhawatiran keamanan terkait dengan aplikasi ini berlaku untuk hampir semua *handset apps*: SplitPay tidak boleh digunakan saat mengoperasikan kendaraan atau dalam situasi lain di mana perhatian pengguna harus difokuskan di tempat lain” [4]. Verba dan kata hubung mempengaruhi keatomikan suatu statemen kebutuhan, dimana menurut standar IEEE 830 sebuah SKPL bersifat *modifiable* jika satu statemen kebutuhan didalamnya hanya dituliskan pada satu tempat saja, tidak bercampur dengan statemen kebutuhan lainnya [2].

Untuk struktur dokumen yang dicek, mengikuti sumber [4] yang dikutip dari aturan standar IEEE 830. Atribut-atribut kualitas yang dipakai dalam penelitian ini adalah sebagai berikut [4]:

**Tabel 1. Atribut kualitas SKPL berdasarkan penggunaan bahasa alami [4]**

No	Quality Attributes
NLM <sub>1</sub>	More than one verbs
NLM <sub>2</sub>	Coordinating conjunction
NLM <sub>3</sub>	Punctuation

**Tabel 2. Atribut kualitas SKPL berdasarkan struktur dokumen [4]**

No	Quality Attributes
SM <sub>1</sub>	Presence of topic “Table of Contents”
SM <sub>2</sub>	Presence of topic “References”
SM <sub>3</sub>	Presence of topic “Overview”
SM <sub>4</sub>	Presence of topic “Assumption and dependencies”
SM <sub>5</sub>	Presence of topic “User interfaces”
SM <sub>6</sub>	Presence of topic “Hardware interfaces”
SM <sub>7</sub>	Presence of topic “Software interfaces”
SM <sub>8</sub>	Presence of topic “Index”

Dalam menghitung kualitas penggunaan bahasa alami dalam SKPL, pertama dilakukan perhitungan *Base Measure Level*, yaitu menghitung frekuensi setiap atribut kualitas sesuai dengan Tabel 1, dengan menggunakan rumus :

$$B_{nli} = \frac{NLM_i}{L_j} (1)$$

Dimana  $B_{nli}$  adalah *base measure* untuk atribut ke-i yang dinormalisasi,  $NLM_i$  adalah frekuensi kemunculan atribut ke-i, dan  $L_j$  adalah jumlah keseluruhan kebutuhan dalam dokumen. Perhitungan *base measure* dilakukan untuk setiap atribut kualitas dalam penggunaan bahasa alami dalam SKPL.

Selanjutnya dilakukan perhitungan semua nilai atribut berdasarkan karakteristik *modifiable* dengan rumus *Derived Measure Level*, yaitu :

$$D_{jcx} = \frac{\sum_{i=1}^n w_i B_{nli}}{\sum_{i=1}^n w_i} (2)$$

Dimana  $D_{jcx}$  adalah *derived measure* untuk sebuah spesifikasi ke-j dengan mempertimbangkan karakteristik x,  $w_i$  adalah bobot atribut ke-i, dan  $B_{nli}$  adalah hasil *base measure* penggunaan bahasa untuk atribut ke-i. Dari hasil perhitungan dua rumus diatas akan didapatkan nilai kesalahan penggunaan bahasa alami dalam

dokumen SKPL pada karakteristik kualitas *modifiable*. Semakin tinggi nilai  $D_{jcx}$ , maka kualitas penggunaan bahasa alami pada karakteristik kualitas *modifiable* dalam SKPL menjadi semakin rendah.

Selanjutnya, akan dinilai kualitas struktur dokumen SKPL, pertama dilakukan perhitungan *Base Measure Level*, yaitu mencari ada atau tidaknya suatu struktur dokumen pada SKPL, dengan rumus :

$$B_{si} = \begin{cases} 1; Found \\ 0; Not Found \end{cases} \quad (3)$$

Dimana  $B_{si}$  adalah kemunculan atribut ke-i. Selanjutnya, dilakukan perhitungan semua nilai atribut berdasarkan karakteristik *modifiable* dengan rumus *Derived Measure Level*, yaitu :

$$Q_{scx} = \frac{\sum_{i=1}^n w_i B_{si}}{\sum_{i=1}^n w_i} \quad (4)$$

Dimana  $Q_{scx}$  adalah tingkat kualitas SKPL dalam struktur dokumen dengan mempertimbangkan karakteristik  $x$ ,  $w_i$  adalah bobot atribut ke-i, dan  $B_{si}$  adalah hasil *base measure* struktur dokumen untuk atribut ke-i. Dari hasil perhitungan dua rumus diatas akan didapatkan nilai kesalahan struktur dokumen SKPL pada karakteristik kualitas *modifiable*. Semakin tinggi nilai  $Q_{scx}$ , maka kualitas struktur dokumen SKPL pada karakteristik kualitas *modifiable* menjadi semakin rendah.

Hasil akhir kualitas SKPL pada karakteristik *modifiable* mempunyai interval kualitas sebagai berikut [13]:

**Tabel 3. Nilai interval untuk setiap level kualitas**

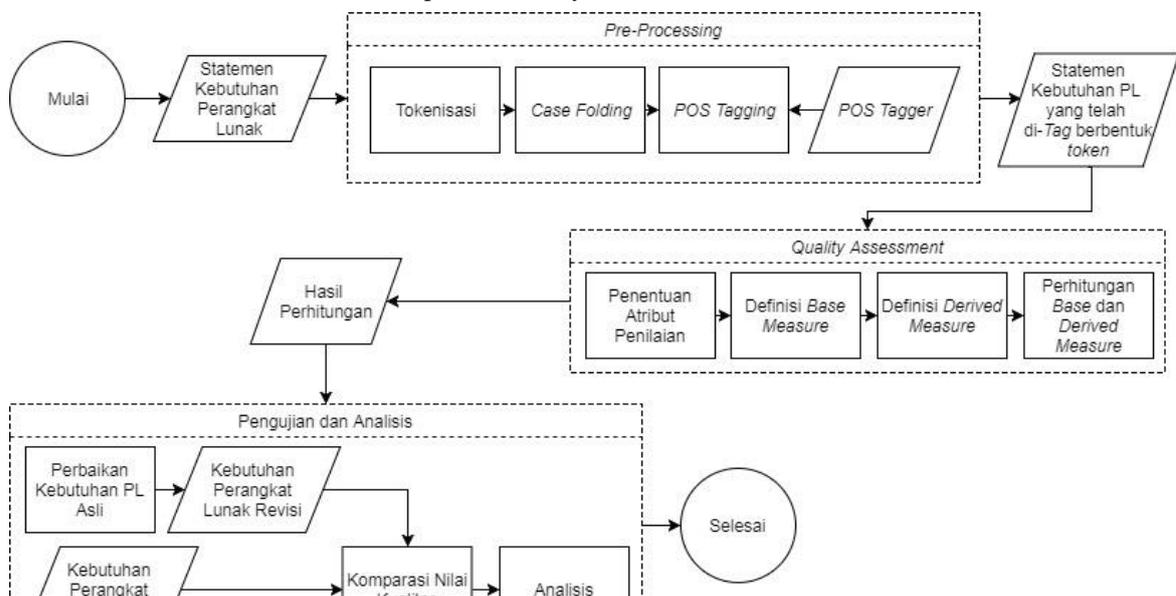
Quality Level	Interval
Good	[0,1)
Moderate	[0,1)
Improvement	[1,α)

### 3. Sistem yang Dibangun

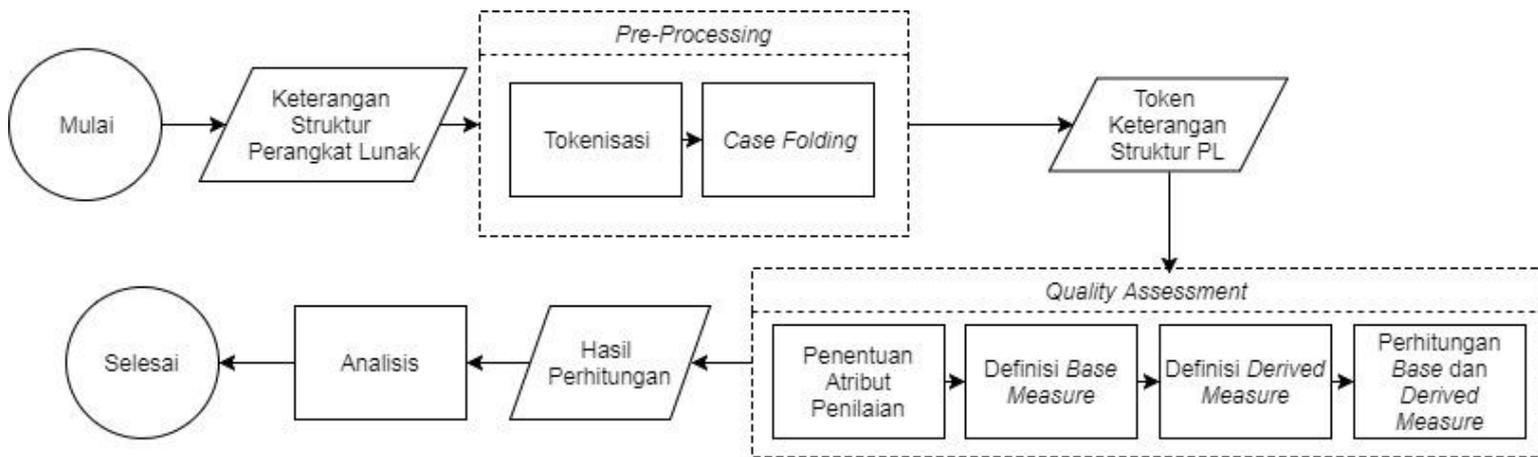
Dalam bagian ini akan dijelaskan rancangan sistem yang dibangun beserta metode pengujian yang dilakukan dalam penelitian ini. Secara umum sistem yang dibangun mengikuti desain sistem sebagai berikut :

#### 3.1 Desain Sistem

Dalam sistem terjadi dua proses, yaitu penilaian kualitas *modifiable* berdasarkan penggunaan bahasa alami dan berdasarkan kelengkapan struktur dokumen. Data SKPL akan melalui proses *Pre-Processing*, *Quality Assessment*, dan terakhir akan dilakukan pengujian dan/atau analisis dari data SKPL tersebut. Flowchart desain sistem penelitian ini yaitu:



**Gambar 1. Flowchart desain sistem perhitungan kualitas SKPL berdasarkan penulisan bahasa alami**



**Gambar 2. Flowchart desain sistem perhitungan kualitas SKPL berdasarkan kelengkapan struktur dokumen**

### 3.1.1 Pre-Processing

Pertama, data SKPL akan diproses terlebih dahulu sebelum bisa dihitung kualitasnya menggunakan metode *Quality Assessment*. Pada tahapan ini yang juga disebut *Pre-Processing*, data tulisan untuk penilaian penggunaan bahasa alami akan diproses dengan tahapan sebagai berikut:

1. *Tokenisasi*  
Dalam proses ini data tulisan SKPL akan dipecah menjadi per kata (token).
2. *Case Folding*  
Dalam proses ini semua kata dari hasil tokenisasi akan menjadi huruf kecil, agar token menjadi sama rata.
3. *POS Tagging*  
Seperti dijelaskan pada bagian 2.2.1, *POS Tagging* akan memberikan label kata sesuai dengan sifat kata tersebut. Pada penelitian ini digunakan *POS Tagger* yang dibuat oleh Fam Rashel, dkk. [8] yang berdasarkan *tagset* yang dibuat oleh Dinarakami, dkk. [14].

Untuk proses penilaian struktur dokumen, hanya langkah 1 dan 2 saja yang dilakukan. Hasil akhir dari *Pre-Processing* untuk penilaian penggunaan bahasa alami adalah kata berlabel (kata yang diberi *tag*), dan untuk penilaian struktur dokumen berupa kata yang telah dipecah menjadi *token*. Proses selanjutnya yaitu perhitungan kualitas dengan metode *Quality Assessment* yang akan dijelaskan pada subbab selanjutnya.

### 3.1.2 Quality Assessment

Data yang didapat dari proses sebelumnya, untuk penilaian penggunaan bahasa alami akan digunakan statemen kebutuhan yang diambil pernyataan kebutuhan fungsional dan nonfungsional dalam SKPL yang akan dimasukkan dalam perhitungan *Quality Assessment*, seperti dalam subbab 2.3. *Tag* tersebut akan mempengaruhi nilai kualitas dokumen SKPL pada karakteristik *modifiable* dalam penggunaan bahasa alami dan masing-masing *tag* termasuk dalam atribut kualitas SKPL [4]. Sementara *token* atau kata yang didapat dari daftar isi dokumen tersebut digunakan untuk menilai kualitas struktur dokumen nya. Secara rinci, atribut kualitas yang dihitung pada penelitian ini adalah sebagai berikut :

Pada penggunaan bahasa alami :

1. *More than one verbs*  
*Verb* dalam bahasa Indonesia adalah kata kerja. Contohnya seperti merancang, mengatur, pergi [14]
2. *Coordinating conjunction*  
*Coordinating conjunction* dalam bahasa Indonesia adalah kata penghubung koordinatif, contohnya dan, tetapi, atau [14]

### 3. Punctuation

*Punctuation* dalam bahasa Indonesia adalah tanda baca, contohnya adalah “.” (titik), “,” (koma) [14]

Pada struktur dokumen :

#### 1. Presence of topic “Table of Contents”

*Table of contents* atau daftar isi adalah bagian dari dokumen yang berisikan petunjuk isi suatu konten beserta nomor halaman tempat isi tersebut berada.

#### 2. Presence of topic “References”

*References* atau referensi adalah rujukan dari suatu kalimat atau konten dari dokumen.

#### 3. Presence of topic “Overview”

*Overview* atau gambaran umum. Gambaran umum berisikan penjelasan umum dari suatu topik pada dokumen.

#### 4. Presence of topic “Assumption and dependencies”

*Assumption and dependencies* yaitu asumsi dan dependensi dipakai dalam pengembangan perangkat lunak dalam SKPL.

#### 5. Presence of topic “User interfaces”

*User interfaces* atau antarmuka pengguna menurut standar IEEE [1], karakteristik logikal dari setiap antarmuka antara perangkat lunak dan penggunanya, dan semua aspek yang mengoptimisasi antarmuka dengan pengguna dan aktor lainnya yang berperan kepada kinerja sistem.

#### 6. Presence of topic “Hardware interfaces”

*Hardware interfaces* atau antarmuka perangkat keras menurut standar IEEE [1], adalah karakteristik logikal setiap antarmuka antara perangkat lunak dan perangkat keras. Contohnya seperti konfigurasi jaringan, spesifikasi perangkat keras, dll.

#### 7. Presence of topic “Software interfaces”

*Software interfaces* atau antarmuka perangkat lunak menurut standar IEEE [1] adalah antarmuka perangkat lunak sebagai penggunaan perangkat lunak lainnya yang dipakai dalam pengembangan perangkat lunak.

#### 8. Presence of topic “Index”

*Index* atau bagian indeks yang berisikan istilah atau kata penting didalam dokumen SKPL serta halaman tempat istilah atau kata penting tersebut berada.

Dalam semua sebelas atribut tersebut memiliki bobot kepentingan, yaitu bobot yang mempengaruhi tingkat *modifiable* suatu SKPL. Dalam penelitian ini digunakan bobot yang diberikan oleh seorang *expert* dalam bidang pembuatan SKPL dengan biodata dilampirkan pada lampiran 15, yaitu sebagai berikut :

**Tabel 4. Bobot atribut kualitas**

Parameter	Bobot
<i>More than one verbs</i>	0.2
<i>Coordinating conjunction</i>	0.8
<i>Punctuation</i>	0.8
<i>Presence of topic “Table of Contents”</i>	1
<i>Presence of topic “References”</i>	0.2
<i>Presence of topic “Overview”</i>	0.8
<i>Presence of topic “Assumption and dependencies”</i>	0.4
<i>Presence of topic “User interfaces”</i>	0.6
<i>Presence of topic “Hardware interfaces”</i>	0.6
<i>Presence of topic “Software interfaces”</i>	0.8
<i>Presence of topic “Index”</i>	1

### 3.1.3 Pengujian dan Analisis Data

Terakhir, akan dilakukan pengujian dan analisis data. Untuk penilaian penggunaan bahasa alami, pengujian dilakukan dengan cara metode komparatif, yaitu hasil dari perhitungan

kualitas SKPL akan dibandingkan dengan data kualitas SKPL yang telah diperbaiki secara manual. Selanjutnya akan disajikan tabel perbandingan nilai karakteristik yang didalamnya terdapat perbandingan nilai dan atribut kualitas dari SKPL yang belum diperbaiki dan yang telah diperbaiki untuk dianalisis.

Sementara pada penilaian struktur dokumen, *token* akan dicek keberadaannya dalam dokumen dan akan disajikan tabel keberadaan struktur dokumen berdasarkan *token* untuk keperluan analisis.

Untuk pengujian penggunaan bahasa alami dalam SKPL akan mengikuti aturan-aturan berikut ini:

- **Verba**

Jika ditemukan **VB** didalam suatu kalimat, maka **VB** tersebut dihilangkan tanpa merubah/mengurangi arti dari kalimat tersebut.

- **Konjungsi Koordinasi**

Jika ditemukan **CC** didalam suatu kalimat, maka **CC** dihilangkan dan kata setelahnya dibuat kalimat baru.

- **Tanda Baca**

Jika ditemukan **Z** berupa koma dan titik didalam suatu kalimat maka **Z** tersebut dihapus dan kata setelahnya dibuat menjadi kalimat baru.

#### 4. Evaluasi

Bagian ini berisikan hasil pengujian serta evaluasinya. Seperti dijelaskan sebelumnya pada penelitian ini digunakan *file* SKPL Aplikasi SIROTI, Aplikasi SMS, dan Aplikasi Explore Berau. SKPL tersebut dipecah menjadi dua bagian, yaitu file SKPL sebelum perbaikan dengan kode huruf (contoh untuk SIROTI A, SMS B, dan Explore Berau C), dan setelah perbaikan penggunaan bahasa alami diberi kode bintang setelah kode huruf (contoh untuk SIROTI A\*, SMS B\*, dan Explore Berau C\*), .

Untuk hasil pengujian beserta analisisnya dibagi menjadi dua bagian, bagian pertama yaitu komparasi frekuensi atribut kualitas dan bagian kedua adalah komparasi nilai kualitas *modifiable*. Bagian pertama berisikan perbandingan kemunculan atribut kualitas yaitu kata-kata yang berpotensi menyebabkan *defect* pada SKPL aplikasi SIROTI sebelum dan setelah perbaikan. Banyaknya kemunculan atribut tersebut akan mempengaruhi kualitas *modifiable* SKPL tersebut, dan akan terlihat pada pengujian dan analisis komparasi nilai kualitas *modifiable* pada subbab selanjutnya. Metode perbaikan dipakai hanya pada penilaian kualitas berdasarkan penggunaan bahasa alami saja menggunakan metode manual dimana kata-kata yang berpotensi menyebabkan *defect* dihilangkan tanpa merubah arti dari kata-kata dalam SKPL dengan mengatomikan statemen kebutuhan yaitu menggunakan satu verba saja jika memungkinkan, menghilangkan kata konjungsi koordinasi, dan mengurangi tanda baca koma dan titik tanpa merubah arti dari statemen secara keseluruhan.

##### 4.1 Komparasi Frekuensi Atribut Kualitas

Pengujian dan analisis ini dilakukan untuk mengetahui komparasi jumlah atribut kualitas yang muncul antara SKPL sebelum perbaikan dan sesudah perbaikan. Atribut kualitas yang ditampilkan adalah atribut kualitas pada penggunaan bahasa alami yaitu sesuai dengan **tabel 1** dan untuk kelengkapan struktur dokumen sesuai dengan **tabel 2**.

**Tabel 5. Tabel Komparasi Atribut Kualitas berdasarkan penggunaan bahasa alami dalam SKPL**

Dokumen	NLM <sub>1</sub>	NLM <sub>2</sub>	NLM <sub>3</sub>	Jumlah Defect	Total Kata
A	10	11	101	122	446
A*	6	2	99	107	491
B	10	18	49	77	424
B*	12	0	18	30	527
C	18	13	74	105	737
C*	24	7	30	61	684

**Tabel 6. Tabel Komparasi Atribut Kualitas berdasarkan kelengkapan struktur dokumen dalam SKPL**

Dokumen	SM <sub>1</sub>	SM <sub>2</sub>	SM <sub>3</sub>	SM <sub>4</sub>	SM <sub>5</sub>	SM <sub>6</sub>	SM <sub>7</sub>	SM <sub>8</sub>
A	Ada	Ada	Tidak Ada	Ada	Ada	Ada	Ada	Tidak Ada
B	Ada	Tidak Ada	Ada	Tidak Ada	Ada	Ada	Ada	Tidak Ada
C	Ada	Ada	Tidak Ada	Ada	Ada	Ada	Ada	Tidak Ada

#### 4.2 Komparasi Nilai Kualitas *Modifiable*

Pengujian dan analisis ini dilakukan untuk mengetahui komparasi nilai kualitas *modifiable* antara SKPL sebelum perbaikan dan sesudah perbaikan. Pada bagian ini akan ditampilkan komparasi perhitungan *base measure* dan *derived measure* antara SKPL sebelum dan sesudah perbaikan.

**Tabel 7. Tabel Komparasi Nilai Kualitas *Modifiable* berdasarkan penggunaan bahasa alami dalam SKPL**

Dokumen	BNL <sub>1</sub>	BNL <sub>2</sub>	BNL <sub>3</sub>	D <sub>jcx</sub>	Jumlah Kebutuhan
A	0.20	0.22	2.02	1.01	50
A*	0.12	0.04	2.02	0.92	49
B	0.30	0.54	1.48	0.93	33
B*	0.16	0	0.24	0.12	73
C	0.28	0.21	1.16	0.63	64
C*	0.32	0.09	0.40	0.25	75

**Tabel 8. Tabel Komparasi Nilai Kualitas *Modifiable* berdasarkan kelengkapan struktur dokumen dalam SKPL**

Dokumen	BS <sub>1</sub>	BS <sub>2</sub>	BS <sub>3</sub>	BS <sub>4</sub>	BS <sub>5</sub>	BS <sub>6</sub>	BS <sub>7</sub>	BS <sub>8</sub>	Q <sub>scx</sub>
A	1	0.2	0	0.4	0.6	0.6	0.8	0	0.67
B	1	0	0.8	0	0.6	0.6	0.8	0	0.71
C	1	0.2	0	0.4	0.6	0.6	0.8	0	0.67

## 5. Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah :

1. Hasil dari penilaian karakteristik *modifiable* pada SKPL aplikasi SIROTI sebelum perbaikan penggunaan bahasa alami adalah 1.01 (Improvement) dan setelah perbaikan penggunaan bahasa alami nilainya 0.92 (Good-Moderate), sementara pada kelengkapan struktur dokumen bernilai 0.67 (Good-Moderate). Pada SKPL aplikasi SMS sebelum perbaikan penggunaan bahasa alami adalah 0.93 (Good-Moderate) dan setelah perbaikan penggunaan bahasa alami nilainya 0.12 (Good-Moderate), sementara pada kelengkapan struktur dokumen bernilai 0.71 (Good-Moderate). Pada SKPL aplikasi Explore Berau sebelum perbaikan penggunaan bahasa alami adalah 0.63 (Good-Moderate) dan setelah perbaikan penggunaan bahasa alami nilainya 0.38 (Good-Moderate), sementara pada kelengkapan struktur dokumen bernilai 0.25 (Good-Moderate)
2. Atribut kualitas yang paling mempengaruhi penggunaan bahasa alami pada SKPL aplikasi SIROTI adalah tanda baca. Pada kelengkapan struktur dokumen ketidakadaan ikhtisar dan indeks membuat nilai kualitas *modifiable* dalam hal kelengkapan struktur dokumen semakin rendah. Pada SKPL aplikasi SMS atribut kualitas yang paling mempengaruhi penggunaan bahasa alami adalah tanda baca. Tidak ada referensi, asumsi dan dependensi, dan indeks pada struktur dokumen nya. Pada SKPL aplikasi Explore Berau atribut kualitas yang paling mempengaruhi penggunaan bahasa alami adalah tanda baca. Tidak ada ikhtisar dan indeks pada struktur dokumen nya.

3. Dalam penulisan bahasa alami lebih baik ketika tidak banyak menggunakan kata-kata yang terdapat pada atribut kualitas, dengan mempertegas keatomikan kebutuhan dalam SKPL dengan tidak membubuhkan tanda baca seperti koma dan kata hubung, dan sebisa mungkin menggunakan satu verba saja per kebutuhan.

Saran peneliti untuk penelitian yang terkait di masa mendatang adalah:

1. Menggunakan jumlah dokumen yang lebih banyak dan disarankan menggunakan SKPL yang kompleks.
2. Penelitian ini berfokus pada karakteristik *modifiable* saja, untuk selanjutnya dapat diteliti karakteristik-karakteristik SKPL baik yang lainnya.
3. Meneliti lebih lanjut aturan bahasa untuk perbaikan dokumen berbahasa Indonesia.

**Daftar Pustaka**

- [1] ISO/IEC/IEEE. ISO/IEC/IEEE International Standard - Systems and software engineering - Life cycle processes - Requirements engineering. ISO/IEC/IEEE, 2011.
- [2] "IEEE Recommended Practice for Software Requirements Specifications," IEEE Standard 830, 1998.
- [3] Davis, A., Overmyer, S., Jordan, K., Caruso, J., Dandashi, F., Dinh, A., Kincaid, G., Ledeboer, G., Reynolds, P., Sitaram, P., Ta, A. and Theofanos, M. Identifying and measuring quality in a software requirements specification. Proceedings First International Software Metrics Symposium. IEEE, 1993
- [4] P. Thitisathienkul and N. Prompoon. Quality Assessment Method for Software Requirements Specifications Based on Document Characteristics and Its Structure. Trustworthy Systems and Their Applications (TSA), 2015 Second International Conference, pages 51–60. IEEE, 2015.
- [5] I. Hussain, O. Ormandjieva, and L. Kosseim. Automatic quality assessment of srs text by means of a decisiontree-based text classifier. In *qsic*, pages 209–218. IEEE, 2007.
- [6] H. M. Jani. Applying case-based reasoning to software requirements specifications quality analysis system. In Software Engineering and Data Mining (SEDM), 2010 2nd International Conference on, pages 140–144. IEEE, 2010.
- [7] R. Kenett. Software specifications metrics: a quantitative approach to assess the quality of documents. In Electrical and Electronics Engineers in Israel, 1996., Nineteenth Convention of, pages 166–169. IEEE, 1996.
- [8] F. Rashel, A. Dinakaramani, A. Luthfi, and R. Manurung. Building an Indonesian Rule-Based Part-of-Speech Tagger. In Asian Language Processing (IALP), 2014 International Conference on, pages 70–73. IEEE, 2014.
- [9] "ISO/IEC International Standard 26702, IEEE Standard 1220-2005 - Systems and software engineering Vocabulary," ISO/IEC Standard 26702/ IEEE Standard 1220, 2005.
- [10] Dick, J., Hull, E. and Jackson, K. 2017. Requirements Engineering. 4th ed. Cham: Springer
- [11] Chowdhury, G. Natural Language Processing. Annual Review of Information Science and Technology, 37. pp. 51-89, 2003.
- [12] Jurafsky, D. and Martin, J. 2014. Speech and language processing. Harlow: Pearson.
- [13] P. Thitisathienkul and N. Prompoon. Quality Assessment Method for Software Development Process Document Based on Software Document Characteristics Metric. Digital Information Management (ICDIM), 2014
- [14] A. Dinakaramani, F. Rashel, A. Luthfi, and R. Manurung. Designing an indonesian part of speech tagset and manually tagged indonesian corpus. In Asian Language Processing (IALP), 2014 International Conference on, pages 66–69. IEEE, 2014.