

# Perancangan Bank Soal Pada Universitas Bunda Mulia Dengan Algoritma Smith Waterman

**Halim Agung dan Sandi Gautama**

<sup>1)2)</sup> Teknik Informatika, Fakultas Teknologi dan Desain, Universitas Bunda Mulia

<sup>1)2)</sup> Jln.Lodan Raya No. 2 Jakarta 14430, (021) 6909090 ext 617

hagung@bundamulia.ac.id, gautama.sandi@gmail.com

## **Abstrak**

*Kebutuhan pembuatan soal ujian yang sulit untuk dikoordinasikan dan adanya kemiripan soal ujian. Penelitian ini dilakukan bertujuan untuk menganalisis dan merancang aplikasi bank soal untuk Universitas Bunda Mulia agar dapat mempermudah dalam pengadaan ujian, dan untuk menyimpan soal dan jawaban yang telah di ujikan sehingga mempunyai record soal yang diujikan. Penelitian ini menggunakan algoritma Smith-Waterman sebagai filter dalam penyeleksian soal ujian. Metode pengumpulan data dilakukan dengan observasi langsung, studi pustaka, dan wawancara. Teknik pengujian kualitas perangkat lunak diuji menggunakan Blackbox Testing yang berdasarkan 4 karakteristik kualitas perangkat lunak model ISO 9126, yaitu: functionality, reliability, usability, dan efficiency menggunakan metode kuesioner. Hasil akhir yang diperoleh dari penelitian ini adalah aplikasi yang mampu memenuhi kebutuhan untuk pembuatan, pengumpulan dan pemfilteran soal. Adapun uji kualitas perangkat lunak dihasilkan dengan kriteria Baik, dengan persentase 78,2%.*

**Kata kunci:** smith-waterman, soal ujian, filter, bank soal

## **1. Pendahuluan**

Sudah banyak aplikasi teknologi komputer yang digunakan oleh beberapa Universitas, baik dari hal-hal kecil seperti absen (menggunakan *flazz card*) sampai dengan yang kompleks seperti pendataan keuangan dan data-data penting lainnya. Namun untuk pengadaan naskah ujian masih bersifat manual dan belum menggunakan aplikasi yang disertai dengan kesamaan soal yang dibuat oleh para dosen di suatu universitas. Dimana hal ini tidaklah efisien di jaman sekarang. Pada Universitas dosen yang mengajar untuk mata pelajaran tertentu pun bisa lebih dari satu dan terdapat beberapa proses untuk pembuatan soal sampai menjadi naskah ujian yang sifatnya agak berbelit dan biasanya terjadi kesamaan soal pada naskah akhir apabila dosen sudah tidak ada ide dalam pembuatan soal maupun terkendala deadline pengumpulan soal. Maka dari itu untuk mencegah hal seperti itu terjadi, dibuatlah aplikasi Bank Soal ini dengan pemfilteran soal menggunakan Algoritma Smith-Waterman. Setiap dosen pada mata kuliah tertentu *upload* soal-soal usulan, lalu akan dipilih soal-soal *Upload* dari matakuliah tersebut oleh dosen pengajar yang dimana dalam pemilihan soal akan dibantu oleh system yang sudah mengimplementasikan Algoritma Smith Waterman ini. Setelah itu naskah soal akan dipertimbangkan kembali oleh Kaprodi apakah disetujui atau tidak untuk dijadikan soal dalam ujian. Selain itu, para dosen juga bisa mengambil soal yang dahulu untuk pembelajaran kelasnya masing-masing, agar semua mahasiswa bisa melihat soal tersebut sebagai bahan untuk latihan. Aplikasi ini nantinya akan diuji kualitasnya dengan menggunakan pengujian model ISO 9126 [2].

## **2. Metode Penelitian**

Data dan referensi diambil dari beberapa buku yang berhubungan dengan bank soal dan pemrograman visual pada vb.net dan melakukan wawancara dengan para dosen Universitas Bunda Mulia

### **2.2 Metode Pemilihan Sampel**

Pengambilan sampel dengan purposive sampling merupakan teknik pengambilan sampel dengan mengambil responden yang terpilih betul oleh peneliti menurut ciri-ciri spesifik yang dimiliki oleh sampel tersebut. Adapun kriteria pemilihan responden adalah:

1. Memiliki tugas dan tanggung jawab sebagai manajer layanan akademik dosen selama lebih dari 1 tahun.
2. Memiliki pengalaman dalam pembuatan soal ujian selama lebih dari 1 tahun.

---

3. Sudah mengajar perkuliahan di Universitas Bunda Mulia selama lebih dari 1 tahun

### 2.3 Basis Data

Menurut Connolly [1] basis data adalah sekumpulan data yang saling berhubungan secara logikal dan juga merupakan deskripsi data yang dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi

### 2.4 Metode Smith Waterman Pada Filter Upload Soal

Dalam Algoritma Smith-Waterman [4] menggunakan pendekatan berbasis local alignment yang berarti posisi ordo matriks yang dijadikan acuan bersifat bebas atau bisa berada pada ordo matriks dimana saja. Terdapat dua tahap utama yaitu:

1. Sequence Matching Untuk dua sequence yang diberikan dimana  $A = A_1, A_2, A_3, \dots, A_n$  dan  $B = B_1, B_2, B_3, \dots, B_n$ , Levensthein distance bekerja sesuai dengan prinsip dynamic programming maka akan dibuat sebuah matriks  $X[\text{panjangA}] [\text{panjangB}]$ . Perhitungan matriks dilakukan dengan ketentuan sebagai berikut:
  - a. Inisialisasi matriks  $X[i][0] = X[0][j] = 0$
  - b. Proses identifikasi melalui rekursif kondisi :  $X[i][j] = \max\{0, X[i-1][j-1] + \text{Nilai Match/MisMatch}, X[i-1][j] - \text{Nilai GAP}, X[i][j-1] + \text{nilai GAP}\}$ .
2. Similarity Calculation Rumus umum menghitung besar kemiripan code berdasarkan algoritma Smith-Waterman adalah sebagai berikut :

$$\text{Sim}(m,n) = 2 * (\text{maxlength}(m,n) / m.\text{length}() + n.\text{length}() )$$

Keterangan:

$\text{Sim}(m,n)$  = Tingkat kemiripan antara string m dan string n

$\text{MaxLength}(m,n)$  = Jumlah karakter yang sama antara string m dan string n

$m.\text{length}$  = Panjang string m

$n.\text{length}$  = Panjang string n

### 2.5 Teknik Pengujian ISO 9126

Responden pengujian ISO 9126 adalah Manajer Layanan Akademik, dan dosen di Universitas UBM. Pengujian validitas konstruk dilakukan dengan menghitung korelasi diantara masing-masing pernyataan dengan skor total. Dalam pengujian validitas instrumen penelitian ini, menggunakan korelasi Product Moment Pearson dengan tool software IBM SPSS Statistics 21 [3]. Dalam uji reliabilitas yang digunakan dalam penelitian adalah metode Alpha Cronbach.

Langkah-langkah yang dilakukan dalam analisis statistik deskriptif tersebut adalah :

- a. Setiap indikator yang dinilai oleh responden, diklasifikasikan dalam lima alternatif jawaban dengan menggunakan skala ordinal yang menggambarkan peringkat jawaban.
- b. Dihitung total skor setiap variabel/subvariabel = jumlah skor dari seluruh indikator variabel untuk semua responden.
- c. Dihitung skor setiap variabel/subvariabel = rata-rata dari total skor.
- d. Untuk mendeskripsikan jawaban responden, juga digunakan statistik deskriptif seperti distribusi frekuensi dan tampilan dalam bentuk tabel ataupun grafik.
- e. Untuk menjawab deskripsi tentang variabel penelitian ini, digunakan rentang kriteria penilaian sebagai berikut:

$$\% \text{ Skor Aktual} = (\text{Skor aktual} / \text{Skor Ideal}) \times 100\%$$

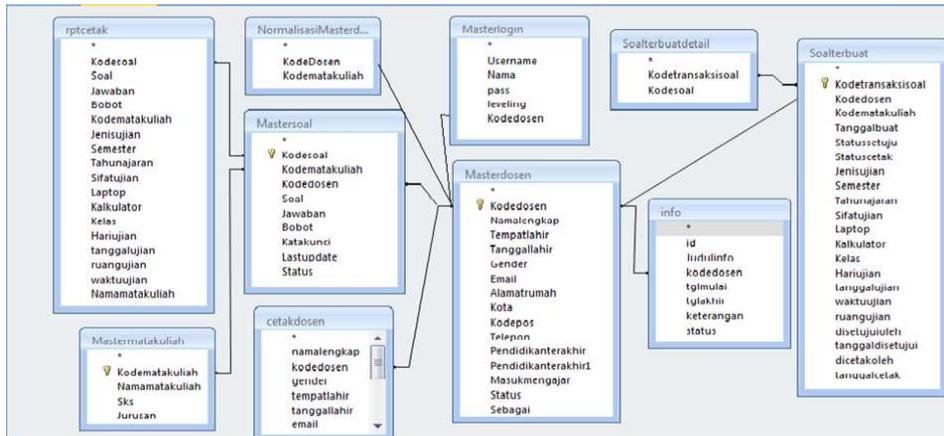
## 3. Hasil dan Pembahasan

### 3.1 Analisis Sistem

#### 3.1.1 Analisis Sistem Berjalan

Pembuatan aplikasi bank soal dimaksudkan untuk memaksimalkan peran aplikasi bank soal untuk universitas dalam menyediakan soal – soal ujian dan jawaban secara update dan tanpa adanya kemiripan soal.

#### 3.1.2 Perancangan Database



Gambar 1. ERD (Entity Relationship Diagrams) basis data Aplikasi Bank Soal

### 3.2 Konstruksi Antarmuka

Bagian ini akan menjelaskan implementasi atau konstruksi tampilan dari aplikasi bank soal

#### 3.2.1 Tampilan Form Login



Gambar 2. Tampilan Login

Gambar 2 adalah tampilan login saat memasuki sistem bank soal, 2 field yang harus diisi dengan username dan password yang sesuai apabila tidak sesuai maka textbox akan muncul

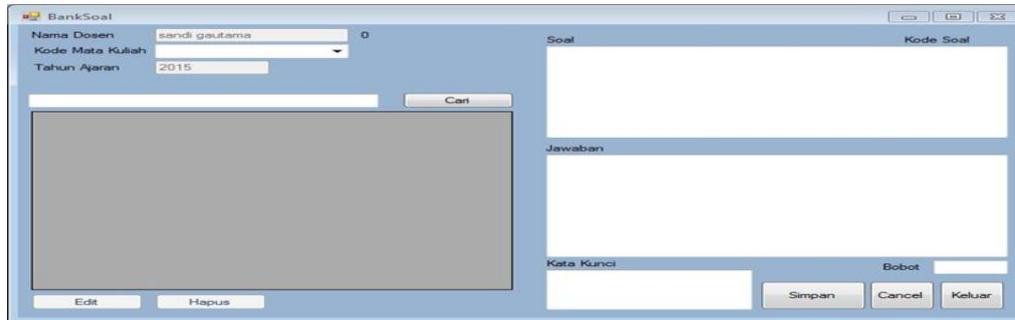
#### 3.2.2 Tampilan Home



Gambar 3. Tampilan Home

Gambar 3 adalah tampilan dasar saat memasuki sistem bank soal. Disini para dosen dapat melihat berbagai informasi mengenai soal ujian yang diusulkan oleh mereka baik yang disetujui (warna putih) maupun tidak (warna merah)

#### 3.2.3 Tampilan Upload Soal



Gambar 4. Tampilan Upload Soal

Gambar 4 adalah tampilan dari form bank soal dimana tempat upload soal bagi para dosen, field kode mata kuliah merupakan otomatisasi dari matakuliah apa saja yang diajarkan dari dosen tersebut. Di bank soal ini terdapat fitur pemfilteran menggunakan algoritma smith-waterman yang berguna untuk mendeteksi kata yang sama sehingga mencegah adanya soal yang sama. Selain itu soal yang di upload terdiri dari kata kunci serta bobot dari soal itu sendiri yang memudahkan untuk proses pembuatan naskah ujian yang akan diajarkan. Penggunaan pada filter upload soal di sistem bank soal menggunakan *Similarity Calculation* dari algoritma smith waterman seperti berikut :

```

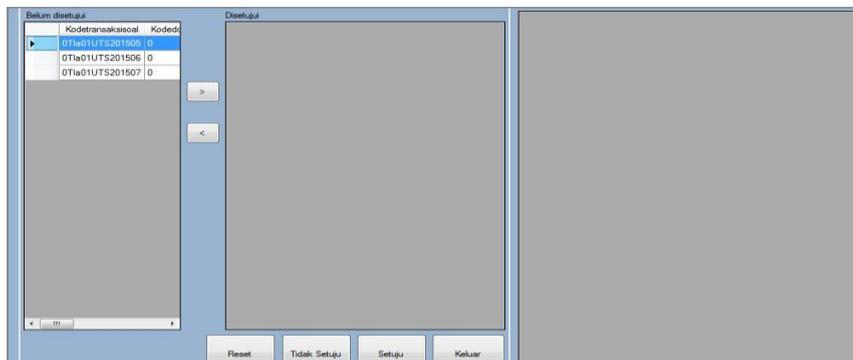
Function cekkesamaansoal2()
    Dim str As String 'Membuat var str dengan tipe String.
    Dim strArr() As String 'Membuat var strArr dengan tipe String Array.
    Dim count As Integer 'Membuat var count dgn tipe Int(berisi Angka), utk kondisi perulangan.
    str = tsoal.Text.Trim 'Mengisi var str dgn input pd TextBox1. Trim berguna untuk menghapus spasi disetiap
awal dan akhir kalimat.
    strArr = str.Split() 'Mengisi strArr dgn batas yg akan berguna utk membagi kalimat pd TextBox1.
    Dim strlukas As String 'Membuat var str dengan tipe String.
    Dim strArrlukas() As String 'buat var strArr dengan tipe String Array.
    Dim countlukas As Integer 'buat var count dgn tipe Int(berisi Angka), untuk kondisi perulangan.
    Dim asu As Integer
    Dim persentasi As Decimal
    Dim searchsoal As New DataTable
    Dim counter3 As Integer
    Dim cobapengumuman As String
    Dim counter4 As Integer
    counter4 = 0
    cobapengumuman = "ADA KESAMAAN KATA DALAM SOAL!!!!!!!!!!" & vbCrLf & vbCrLf
    sSQL = "select * from mastersoal where kodematakuliah = " & cbkmk.Text & " "
    GetData(sSQL, searchsoal)
    For counter3 = 0 To searchsoal.Rows.Count - 1
        strlukas = searchsoal.Rows(counter3).Item("soal").ToString.Trim 'Mengisi variabel str dengan input pada
        TextBox1. Trim berguna untuk menghapus spasi disetiap awal dan akhir kalimat.
        strArrlukas = strlukas.Split() 'Mengisi strArr dengan batas yang akan berguna untuk membagi kalimat
        pada TextBox1.
        For count = 0 To strArr.Length - 1
            ' MsgBox(strArr(count), MsgBoxStyle.OkOnly, "Informasi")
            For countlukas = 0 To strArrlukas.Length - 1
                If strArr(count) = strArrlukas(countlukas) Then
                    asu = asu + 1
                    'MsgBox(strArrlukas(countlukas), MsgBoxStyle.OkOnly, "Informasi")
                End If
            Next
        Next
        persentasi = ((asu * 2) / (strArr.Length + strArrlukas.Length) * 100)
        If persentasi = 0 Then
            Else
                counter4 = counter4 + 1
                cobapengumuman = cobapengumuman & counter4 & ". " &
                searchsoal.Rows(counter3).Item("soal").ToString & vbCrLf & " tingkat kesamaan " & persentasi & "%" & vbCrLf
            End If
        If persentasi > 70 Then
    
```

```

        tanda = 1
    End If
    asu = 0
Next
If cobapengumuman = "SYSTEM MENDETEKSI KESAMAAN KATA DALAM SOAL!!!!!!!!!!" & vbCrLf
& vbCrLf Then
Else
    MsgBox(cobapengumuman, MsgBoxStyle.OkOnly, "Informasi")
End If
counter4 = 0
End Function

```

### 3.2.4 Tampilan Approval



Gambar 5. Tampilan Approval.

Gambar 6 adalah tampilan dari form approval, digunakan untuk approval semua naskah ujian yang diajukan dosen. memiliki fitur preview soal - soal yang diajukan.

### 3.3 Pengujian Aplikasi

#### 3.3.1 Pengujian Kualitas

Responden dalam pelaksanaan uji kualitas ini adalah Manajer Layanan Akademik, Kaprodi dan Dosen yang totalnya berjumlah 25 responden.

- a. Uji validitas dengan moment pearson  
 Dari hasil uji validitas diketahui bahwa semua nilai  $r_{hitung}$  lebih besar dari  $r_{tabel}$  (0,463). Artinya tiap pernyataan berkorelasi dengan skor totalnya dan didapat nilai korelasi untuk item-item lainnya nilainya lebih dari 0,463 maka dapat disimpulkan bahwa item-item tersebut berkorelasi signifikan dengan skor total dan dapat disimpulkan bahwa semua butir instrumen tersebut valid.
- b. Uji reliabilitas dengan Alpha Cronbach  
 Dari uji reliabilitas diperoleh hasil bahwa nilai  $r_{hitung}$  lebih besar dari 0,6 sehingga instrumen penelitian untuk semua variabel yang berbentuk kuesioner sudah valid dan *reliabel*.

Tabel 1. Hasil Uji Reliabilitas Instrumen

Variabel	Alpha Cronbach	Kriteria	Keterangan
Functionality	.745	Alpha Cronbach > 0,60 maka reliabel	reliabel
Reliability	.719		reliabel
Usability	.737		reliabel
Efficiency	.828		reliabel

#### 3.3.2 Uji ISO 9126

Tabel 2. Hasil Pengujian Kualitas

Aspek	Skor Aktual	Skor Ideal	% Skor Aktual	Kriteria
<i>Functionality</i>	775	1000	77,50%	BAIK
<i>Reliability</i>	480	625	76,80%	BAIK
<i>Usability</i>	496	625	79,36%	BAIK
<i>Efficiency</i>	204	250	81,60%	BAIK
<b>Total</b>	<b>1955</b>	<b>2500</b>	<b>78,20%</b>	<b>BAIK</b>

---

Berdasarkan tabel di atas dapat disimpulkan bahwa tingkat kualitas perangkat lunak aplikasi bank soal secara keseluruhan dalam kriteria Baik, dengan persentase 78.20%. Aspek kualitas tertinggi adalah berdasarkan aspek *Efficiency* dengan persentase sebesar 81,60% selanjutnya aspek *Usability* dengan 79,36%. Aspek *Functionality* dengan persentase sebesar 77,50%, sedangkan aspek kualitas terendah adalah dari aspek *Reliability* dengan persentase sebesar 76,80%.

#### 4. Simpulan

Berdasarkan hasil penelitian yang telah dibahas sebelumnya, maka dapat diambil kesimpulan sebagai berikut:

1. Penggunaan Algoritma Smith-Waterman dapat diimplementasikan dalam pembuatan aplikasi bank soal pada Universitas Bunda Mulia.
2. Model analisis, perancangan dan implementasi perangkat lunak untuk pengembangan sistem Bank Soal dapat berfungsi menyediakan data soal serta jawaban yang terintegrasi serta mendukung proses approval untuk naskah ujian.
3. Tingkat kualitas perangkat lunak aplikasi bank soal yang dihasilkan berdasarkan empat karakteristik model ISO 9126, yaitu: *functionality*, *reliability*, *usability*, dan *efficiency* dengan kriteria Baik, dengan persentase 78.20%. Aspek kualitas tertinggi adalah berdasarkan aspek *Efficiency* dengan persentase sebesar 81,60% selanjutnya aspek *Usability* dengan 79,36%. Aspek *Functionality* dengan persentase sebesar 77,50%, sedangkan aspek kualitas terendah adalah dari aspek *Reliability* dengan persentase sebesar 76,80%.

#### Saran

Diharapkan nantinya aplikasi bank soal ini dapat menggunakan algoritma – algoritma yang terbaru sesuai dengan kebutuhan dan diharapkan nantinya aplikasi ini dapat terintegrasi dengan mobile sehingga para dosen dapat membuat soal tanpa harus membuka laptop atau komputer.

#### Daftar Pustaka

- [1] Conolly, T., Begg, C. *Database Systems A Practical Approach to Design, Implementation, and Management*. 5<sup>th</sup> Edition. Boston: Pearson Education. 2010
- [2] Erik 2012 Erik, Van. "Standard glossary of terms used in Software Testing (V2.2)". ISTQB, 2012.
- [3] Wahana Komputer, Solusi Mudah dan Cepat Menguasai SPSS 16.0 Untuk Pengolahan Data Statistik, PT Elex Media Komputindo, Jakarta. 2009
- [4] Zhan Su, Byung Ryul Ahn, Plagiarism Detection Using the Levenshtein Distance and Smith-Waterman Algorithm, Department of Artificial Intelligence, University of Sungkyunkwan Cheoncheon dong, Jangan-gu, Suwon, Korea 2008