



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00201846741, 24 September 2018

Pencipta

Nama : **Dr. Sunda Ariana, M.Pd.,M.M, Hadi Syaputra, M.Kom dkk**
Alamat : Jl. Macan Lindungan Soekarno Hatta No.3 RT.003 RW.005 Bukit Baru Ilir Barat I Palembang, Indonesia, SUMATERA SELATAN, 30131
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Dr. Sunda Ariana, M.Pd.,M.M, Hadi Syaputra, M.Kom dkk**
Alamat : Jl. Macan Lindungan Soekarno Hatta No.3 RT.003 RW.005 Bukit Baru Ilir Barat I Palembang, Palembang, SUMATERA SELATAN, 30131
Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **WORD SPACING IDENTIFICATION**
Tanggal dan tempat diumumkan untuk pertama kali : 24 September 2018, di PALEMBANG
di wilayah Indonesia atau di luar wilayah Indonesia
Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan : 000118398

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.

Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL



Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

Disclaimer:

Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, menteri berwenang untuk mencabut surat pencatatan permohonan.

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Dr. Sunda Ariana, M.Pd.,M.M	Jl. Macan Lindungan Soekarno Hatta No.3 RT.003 RW.005 Bukit Baru Ilir Barat I Palembang
2	Hadi Syaputra, M.Kom	Jl. Tanjung Sari II Komp. Golden Estate Blok. B4 Kec. Kalidoni Kel. Bukit Sangkal
3	Dr. Tri Basuki Kurniawan, S.Kom., M.Eng	Jl. Kol Sulaiman Amin Km 7 Talang Buruk

LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Dr. Sunda Ariana, M.Pd.,M.M	Jl. Macan Lindungan Soekarno Hatta No.3 RT.003 RW.005 Bukit Baru Ilir Barat I Palembang
2	Hadi Syaputra, M.Kom	Jl. Tanjung Sari II Komp. Golden Estate Blok. B4 Kec. Kalidoni Kel. Bukit Sangkal
3	Dr. Tri Basuki Kurniawan, S.Kom., M.Eng	Jl. Kol Sulaiman Amin Km 7 Talang Buruk



Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

DOKUMEN APLIKASI

WORD SPACING IDENTIFICATION

2018

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

RINGKASAN EKSEKUTIF

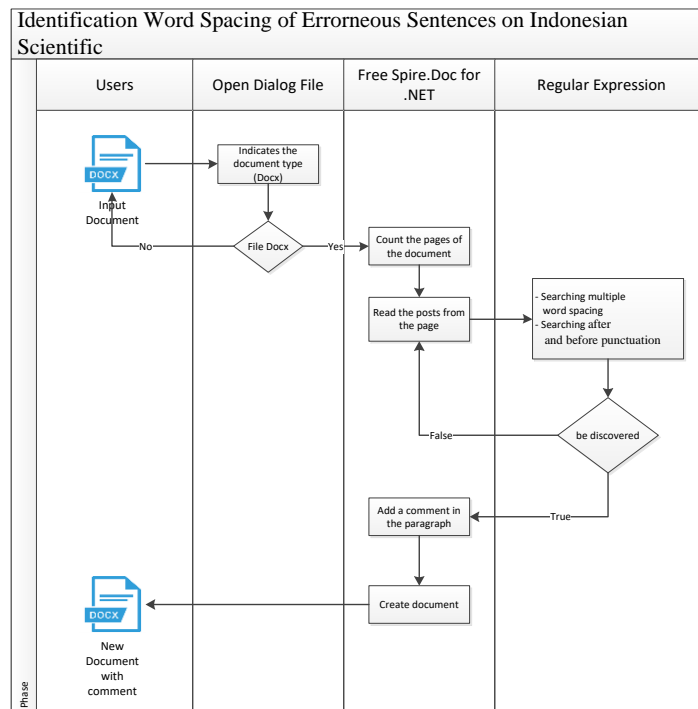
Aplikasi ini berjalan di system operasi Windows, bertujuan untuk melakukan pendeteksian kesalahan penggunaan spasi pada tanda baca didalam dokumen karya ilmiah, kesalahan-kesalahan tersebut sering timbul karena penggunaan rata kanan yang berakibat mengabaikan aturan spasi dalam penulisan. Perangkat lunak ini dibangun menggunakan aplikasi visual studio dengan library spire.

1. Aturan penggunaan spasi sebelum dan sesudah tanda baca

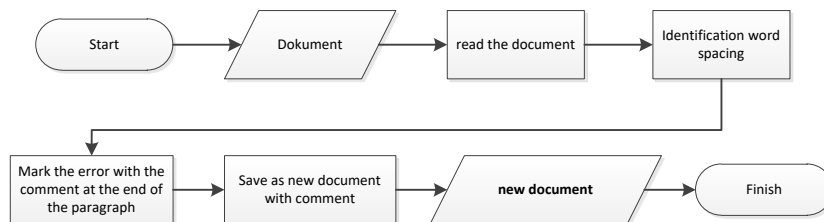
Tanda Baca	Simbol	Spasi sebelum	Spasi sesudah	Contoh
Tanda titik	.	✗	✓	Kami pergi. Mereka datang.
Tanda koma	,	✗	✓	Satu, dua, dan tiga.
Tanda titik koma	;	✗	✓	Hari sudah siang; ia masih tidur.
Tanda titik dua	:	✗	✓	Dua hal yang utama: akal dan hati.
Tanda tanya	?	✗	✓	Apa katamu? Aku tidak mendengar.
Tanda seru	!	✗	✓	Jangan masuk! Ia sedang tidur.
Tanda hubung	-	✗	✗	Kemarin ia di-PHK.
Tanda pisah	-	✗	✗	Acara itu berlangsung tanggal 5-10 bulan ini.
Tanda garis miring	/	✗	✗	Harganya Rp200/lembar.
Tanda petik	"..."	✓	✓	Ia dijuluki "si hitam".
Tanda kurung	(...)	✓	✓	Lokakarya (workshop) itu resmi dibuka.
Tanda elipsis	...	✓	✓	Jangan-jangan ... ia pelakunya.
Tanda apostrof	'	✓	✗	Mereka sudah datang, 'kan?

sumber : <https://pbs.twimg.com/media/Cso-1l3UAAA81JN.jpg>
 Gambar 1. Spasi sebelum dan sesudah tanda baca

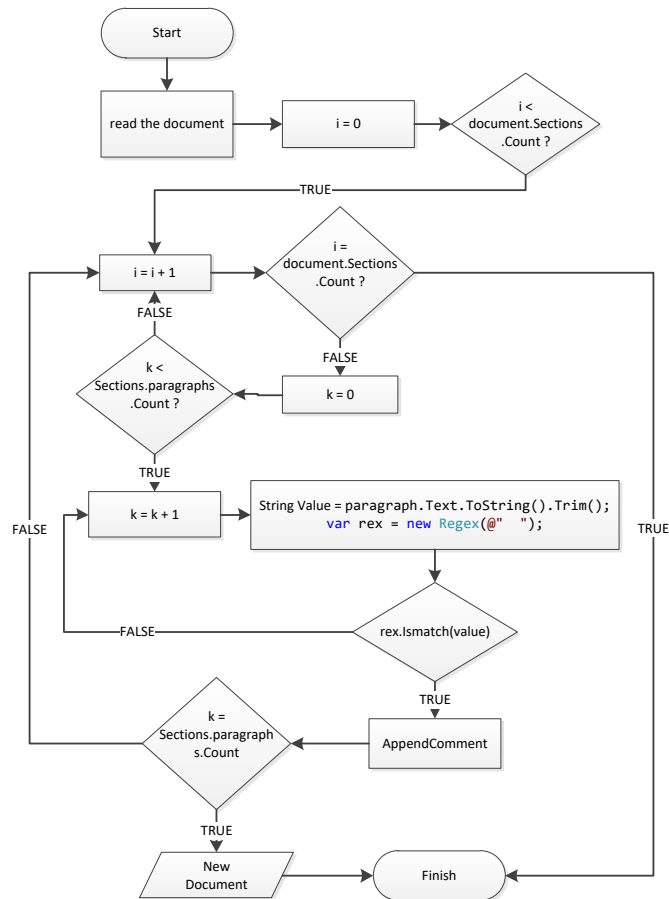
2. Arsitektur Aplikasi



Gambar 2. Arsitektur Aplikasi



Gambar 3. Diagram alir identifikasi spasi di dalam dokumen



Gambar 4. Diagram alir mencari spasi lebih dari satu di dalam dokumen

3. Membuat Dashboard Aplikasi

a. MainWindow.xaml

Berikut adalah source code untuk tampilan dashboard aplikasi dengan menggunakan format .xaml

```

<Window x:Class="WordSpacingIdentification.MainWindow"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
xmlns:local="clr-namespace:WordSpacingIdentification"
mc:Ignorable="d"
Title="Word Spacing Identification" Height="682.433" Width="1024.333"
WindowStartupLocation="CenterScreen">
  <Grid Margin="0,0,-8,0">
    <GroupBox x:Name="groupBox_1" Header="" HorizontalAlignment="Left"
VerticalAlignment="Top" Height="136" Width="988" Margin="8,110,0,0"/>
    <Label x:Name="label" Content="Nama Dokumen" HorizontalAlignment="Left"
Margin="117,126,0,0" VerticalAlignment="Top"/>
    <Button x:Name="button" Content="Dokumen" HorizontalAlignment="Left"
Margin="24,124,0,0" VerticalAlignment="Top" Width="84" Click="button_Click"
RenderTransformOrigin="0.473,0.45" Height="74"/>
    <Label x:Name="lbFileName" Content="Label" HorizontalAlignment="Left"
Margin="225,126,0,0" VerticalAlignment="Top"/>
  
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```

        <Label x:Name="Judul" Content="Judul" HorizontalAlignment="Left"
Margin="118,141,0,0" VerticalAlignment="Top"
RenderTransformOrigin="0.224,0.404"/>
        <Label x:Name="label1" Content=":" HorizontalAlignment="Left"
Margin="220,126,0,0" VerticalAlignment="Top"/>
        <Label x:Name="label2" Content=":" HorizontalAlignment="Left"
Margin="220,140,0,0" VerticalAlignment="Top"/>
        <Label x:Name="label3" Content="Label" HorizontalAlignment="Left"
Margin="225,140,0,0" VerticalAlignment="Top"/>
        <DocumentViewer x:Name="documentViewer" HorizontalAlignment="Left"
Margin="10,251,0,0" VerticalAlignment="Top" Width="986" Height="385"/>
        <Button Content="Proses" HorizontalAlignment="Left" Margin="24,206,0,0"
VerticalAlignment="Top" Width="84" RenderTransformOrigin="-4.453,-3.5"
Click="Button_Click_1" Height="26" AutomationProperties.Name="btProses"/>
        <Label x:Name="label4" Content="WORD SPACING IDENTIFICATION"
HorizontalAlignment="Left" Margin="10,10,0,0" VerticalAlignment="Top"
Height="100" Width="986" HorizontalContentAlignment="Center"
VerticalContentAlignment="Center" FontSize="36" FontWeight="Bold">
            <Label.Foreground>
                <RadialGradientBrush>
                    <GradientStop Color="#FF0046AC"/>
                    <GradientStop Color="#FFFAFBFB" Offset="0.885"/>
                    <GradientStop Color="#FF5281C6" Offset="0.346"/>
                </RadialGradientBrush>
            </Label.Foreground>
            <Label.OpacityMask>
                <LinearGradientBrush EndPoint="0.5,1"
MappingMode="RelativeToBoundingBox" StartPoint="0.5,0">
                    <GradientStop Color="#FFE40D0D" Offset="0"/>
                    <GradientStop Color="#FF301A1A" Offset="1"/>
                </LinearGradientBrush>
            </Label.OpacityMask>
            <Label.Background>
                <ImageBrush ImageSource="D:\Users\HADI
S\VisualStudioProject\WordSpacingIdentification\Images\bacground.jpg"
Stretch="UniformToFill"/>
            </Label.Background>
        </Label>

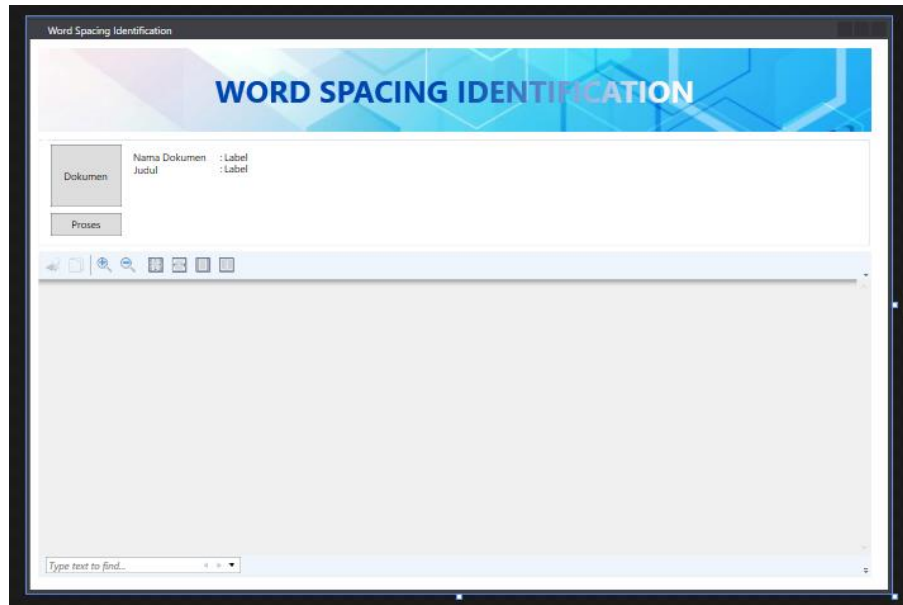
    </Grid>
</Window>

```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

b. Tampilan MainWindow.xaml

Gambar dibawah ini merupakan tampilan aplikasi



Gambar 4. MainWindow.xaml

c. Tombol Dokumen

```
private void button_Click(object sender, RoutedEventArgs e)
{
    // Set filter for file extension and default file extension
    dlg.DefaultExt = ".docx";
    dlg.Filter = "Word documents (.docx)|*.docx";

    // Display OpenFileDialog by calling ShowDialog method
    Nullable<bool> result = dlg.ShowDialog();

    // Get the selected file name and display in a TextBox
    if (result == true)
    {
        if (dlg.FileName.Length > 0)
        {
            //lbFileName.Content = dlg.FileName;
            var onlyFileName = System.IO.Path.GetFileName(dlg.FileName);
            lbFileName.Content = onlyFileName;

            newXPSDocumentName = String.Concat(System.IO.Path.GetDirectoryName(dlg.FileName),
            "\\",
            System.IO.Path.GetFileNameWithoutExtension(dlg.FileName), ".xps");

            // Set DocumentViewer.Document to XPS document
            documentViewer.Document =
            ConvertWordDocToXPSDoc(dlg.FileName,
            newXPSDocumentName).GetFixedDocumentSequence());

            var txtLists = ExtractTextFromXps(newXPSDocumentName);

            //mencari ukuran font terbesar untuk ditampilkan sebagai judul
            foreach (List<string> txtList in txtLists)
            {
```


Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```

pageldx++;
var textdata = string.Empty;
foreach (String s in txtList)
{
    rowIdx++;
    //textdata += s.ToString() + "\r\n";
    value = s.ToString();
    lines = Regex.Split(value, "\r\n");
    ukuranfont = Convert.ToDouble(lines[0]);
    isitext = lines[1];
    if (ukuranfont > maxSix)
    {
        maxSix = ukuranfont;
    }
    else
    {
        //maxSix = tes;
    }
}
ukuranMax = maxSix.ToString();
//mencari judul dari list berdasarkan ukuran font terbesar
foreach (List<string> txtList in txtLists)
{
    foreach (String s in txtList)
    {
        panjangukuranMax = ukuranMax.Length;
        panjangdata = (s.ToString()).Length;
        if (panjangukuranMax < panjangdata)
        {
            string tes = (s.ToString()).Substring(0, panjangukuranMax);
            if (tes == ukuranMax)
            {
                judul = (s.ToString()).Substring(panjangukuranMax + 2, panjangdata - 2 -
panjangukuranMax);
            }
        }
        juduls = juduls + "" + judul;
    }
}
label3.Content = juduls.ToString().Trim();
}
}
}

```

d. Tombol Proses

```

private void Button_Click_1(object sender, RoutedEventArgs e)
{
    Spire.Doc.Document document = new Spire.Doc.Document();
    document.LoadFromFile(dlg.FileName);
    for (int i = 0; i < document.Sections.Count; i++)

```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```

{
    Spire.Doc.Section section = document.Sections[i];
    for (int k = 0; k < section.Paragraphs.Count; k++)
    {
        Spire.Doc.Documents.Paragraph paragraph = section.Paragraphs[k];
        string stValue = paragraph.Text.ToString().Trim();
        if (stValue.Equals(""))
        {
        }else
        {
            //multipe space
            var rex = new Regex(@" ");
            bool tes = rex.IsMatch(stValue);
            if (tes.Equals(true))
            {
                Spire.Doc.Section sectionComment = document.Sections[i];
                Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
                string str = "lebih dari satu spasi antar kata";
                Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
                comment.Format.Author = "Hadi";
                comment.Format.Initial = "HS";
            }
            // spasi sebelum tanda titik
            var rextitik = new Regex(@" [\.]");
            bool btitik = rextitik.IsMatch(stValue);
            if (btitik.Equals(true))
            {
                Spire.Doc.Section sectionComment = document.Sections[i];
                Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
                string str = "spasi sebelum tanda titik";
                Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
            }
        }
    }
}

```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
        comment.Format.Author = "Hadi";
        comment.Format.Initial = "HS";
    }
    // spasi sebelum tanda koma
    var rexkoma = new Regex(@"[\,]");
    bool bkoma = rexkoma.IsMatch(stValue);
    if (bkoma.Equals(true))
    {
        Spire.Doc.Section sectionComment = document.Sections[i];
        Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
        string str = "spasi sebelum tanda koma";
        Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
        comment.Format.Author = "Hadi";
        comment.Format.Initial = "HS";
    }
    // spasi sebelum tanda titik koma
    var rextitikkoma = new Regex(@"[\:]");
    bool btitikkoma = rextitikkoma.IsMatch(stValue);
    if (btitikkoma.Equals(true))
    {
        Spire.Doc.Section sectionComment = document.Sections[i];
        Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
        string str = "spasi sebelum tanda titik koma";
        Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
        comment.Format.Author = "Hadi";
        comment.Format.Initial = "HS";
    }
    // spasi sebelum tanda titik dua
    var rextitikdua = new Regex(@"[\:]");
    bool btitikdua = rextitikdua.IsMatch(stValue);
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
if (btitikdua.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sebelum tanda titik dua";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sebelum tanda tanya
var rextandatanya = new Regex(@" [\?]"");
bool btandatanya = rextandatanya.IsMatch(stValue);
if (btandatanya.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sebelum tanda tanya";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sebelum tanda seru
var rextandaseru = new Regex(@" [\!]"");
bool btandaseru = rextandaseru.IsMatch(stValue);
if (btandaseru.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sebelum tanda seru";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
comment.Format.Author = "Hadi";
comment.Format.Initial = "HS";
}

// spasi sebelum tanda hubung
var rexsebelumtandahubung = new Regex(@"[\-]");
bool bsebelumtandahubung = rexsebelumtandahubung.IsMatch(stValue);
if (bsebelumtandahubung.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sebelum tanda hubung";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sesudah tanda hubung
var rexsesudahtandahubung = new Regex(@"[\-] ");
bool bsesudahtandahubung = rexsesudahtandahubung.IsMatch(stValue);
if (bsesudahtandahubung.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sesudah tanda hubung";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sebelum tanda pisah
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
var rexsebelumtandapisah = new Regex(@" \_");
bool bsebelumtandapisah = rexsebelumtandapisah.IsMatch(stValue);
if (bsebelumtandapisah.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sebelum tanda pisah";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sesudah tanda pisah
var rexsesudahtandapisah = new Regex(@" \_ ");
bool bsesudahtandapisah = rexsesudahtandapisah.IsMatch(stValue);
if (bsesudahtandapisah.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
    string str = "spasi sesudah tanda pisah";
    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
    comment.Format.Author = "Hadi";
    comment.Format.Initial = "HS";
}

// spasi sebelum garis miring
var rexsebelumgarismiring = new Regex(@" \[ /");
bool bsebelumgarismiring = rexsebelumgarismiring.IsMatch(stValue);
if (bsebelumgarismiring.Equals(true))
{
    Spire.Doc.Section sectionComment = document.Sections[i];
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];

string str = "spasi sebelum garis miring";

Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);

comment.Format.Author = "Hadi";

comment.Format.Initial = "HS";

}

// spasi sesudah garis miring

var rexsesudahgarismiring = new Regex(@"[/] ");

bool bsesudahgarismiring = rexsesudahgarismiring.IsMatch(stValue);

if (bsesudahgarismiring.Equals(true))

{

    Spire.Doc.Section sectionComment = document.Sections[i];

    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];

    string str = "spasi sebelum garis miring";

    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);

    comment.Format.Author = "Hadi";

    comment.Format.Initial = "HS";

}

/*

// spasi sebelum dan sesudah tanda baca ()

var rex3 = new Regex(@"[-_/] ");

bool tes3 = rex3.IsMatch(stValue);

if (tes3.Equals(true))

{

    Spire.Doc.Section sectionComment = document.Sections[i];

    Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];

    string str = "spasi sebelum dan sesudah tanda baca";

    Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);

    comment.Format.Author = "Hadi";
```

Dokumen Aplikasi	Ver 1.0
WORD SPACING IDENTIFICATION	

```
        comment.Format.Initial = "HS";
    }

    // spasi sebelum dan sesudah tanda baca ()

    var rex4 = new Regex(@"["''"] ");
    /*string intValue = (Convert.ToByte(Convert.ToChar(stValue).ToString())).ToString();

    bool tes4 = rex4.IsMatch(intValue);

    if (tes4.Equals(true))
    {
        Spire.Doc.Section sectionComment = document.Sections[i];
        Spire.Doc.Documents.Paragraph paragraphComment = sectionComment.Paragraphs[k];
        string str = "spasi sebelum dan sesudah tanda baca";
        Spire.Doc.Fields.Comment comment = paragraph.AppendComment(str);
        comment.Format.Author = "Hadi";
        comment.Format.Initial = "HS";

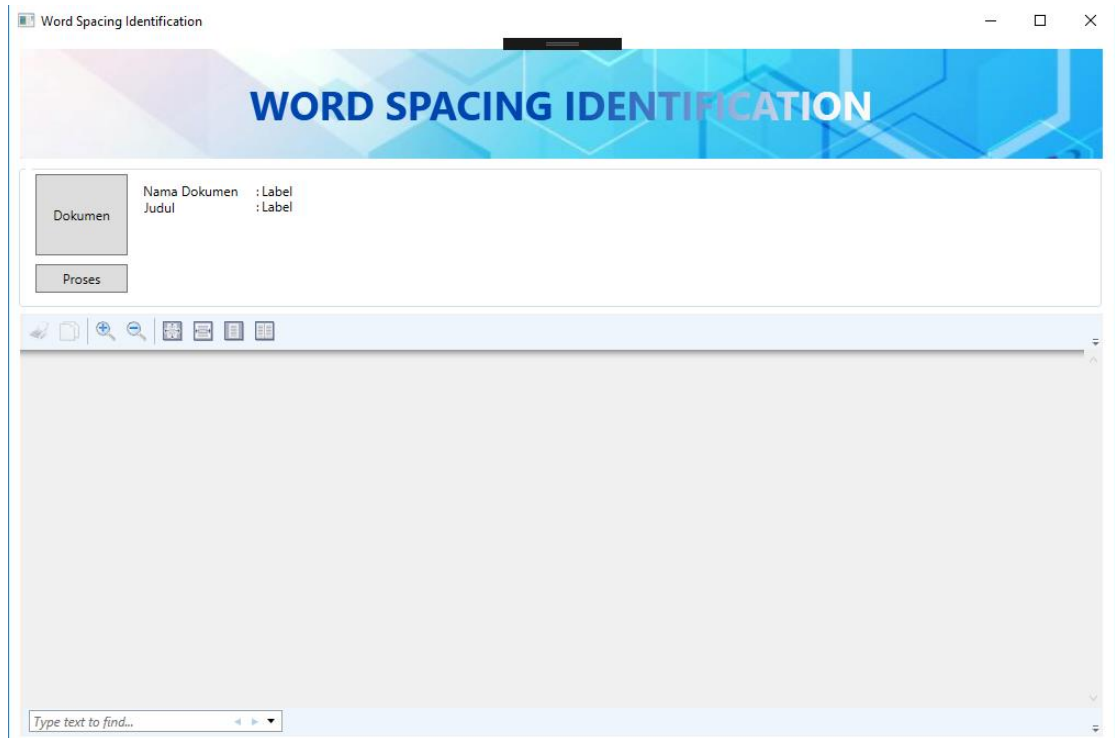
        */
    }
}

document.SaveToFile("Addcommentpath.docx", FileFormat.Docx);

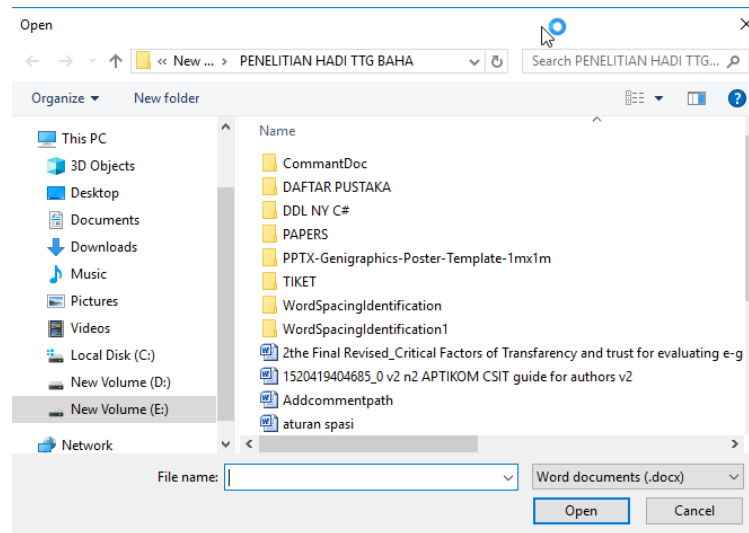
System.Diagnostics.Process.Start("Addcommentpath.docx");
}
```


4. Hasil Perangkat lunak

a. Tampilan Utama



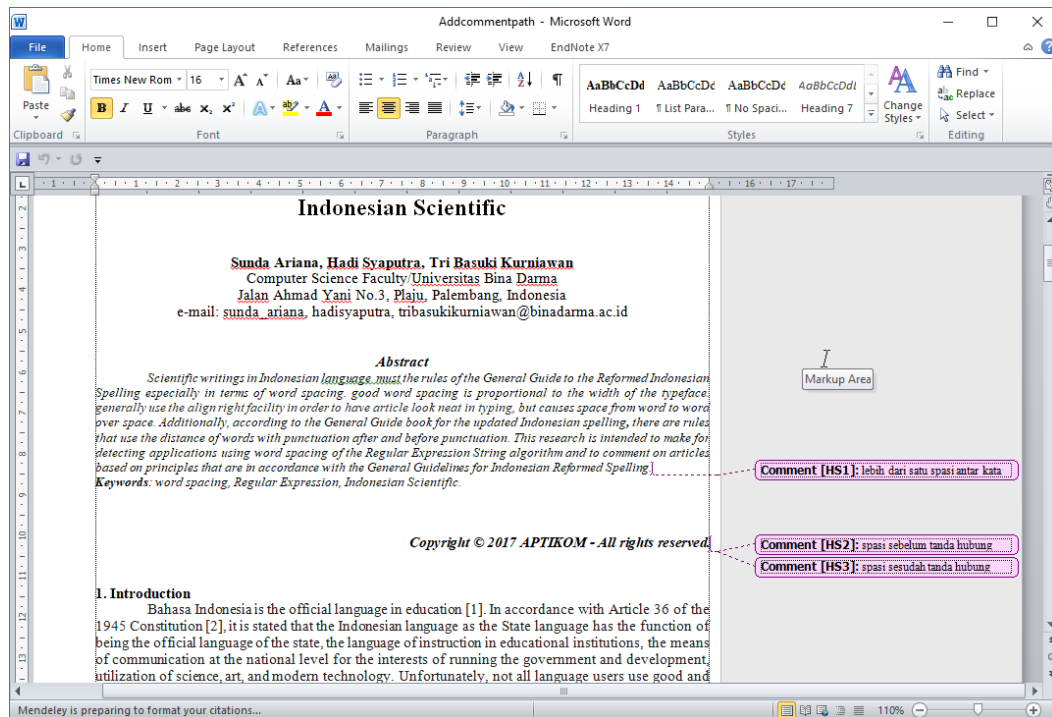
Gambar 5. Tampilan Utama



Gambar 6. Open File



Gambar 7. File



Gambar 8. Hasil Koreksi