

Rancang Bangun Piranti Lunak Pengolah Data Pasca Pengujian Terowongan Angin Kecepatan Rendah Indonesia

Ivransa Zuhdi Pane

Unit Pelaksana Teknis Laboratorium Aero Gas-dinamika dan Getaran,
Badan Pengkajian dan Penerapan Teknologi, Tangerang Selatan, Indonesia
izpane@gmail.com

Diterima 20 November 2015

Disetujui 16 Desember 2015

Abstract—Data post-processing plays important roles in a wind tunnel test, especially in supporting the validation of the test results and further data analysis related to the design activities of the test objects. One effective solution to carry out the data post-processing in an automated productive manner, and thus eliminate the cumbersome conventional manual way, is building a software which is able to execute calculations and have abilities in presenting and analyzing the data in accordance with the post-processing requirement. Through several prototype development cycles, this work attempts to engineer and realize such software to enhance the overall wind tunnel test activities.

Index Terms—software engineering, wind tunnel test, data post-processing, prototype, pseudocode

I. PENDAHULUAN

Pengujian terowongan angin merupakan rangkaian kegiatan pengukuran, perolehan, pengolahan dan presentasi data yang dilakukan dengan tujuan untuk mengetahui karakteristik aerodinamika dari objek yang diuji. Pengujian ini diadakan di fasilitas terowongan angin, seperti Terowongan Angin Kecepatan Rendah Indonesia (TAKRI), yang merupakan penyedia layanan pengujian terowongan angin terbaik di Indonesia. Salah satu faktor penentu keberhasilan pelaksanaan pengujian terowongan angin adalah keabsahan hasil pengujian yang umumnya dikaji setelah pengujian dilaksanakan. Dalam kasus sederhana, anggota tim pengujian yang

bertugas sebagai analis data diinstruksikan untuk melakukan komparasi antara hasil satu slot pengujian yang baru saja selesai dilaksanakan dengan hasil slot pengujian lainnya yang beberapa waktu sebelumnya telah dilaksanakan, dimana keduanya memiliki konfigurasi pengujian yang sama atau mirip. Proses pengolahan data pasca pengujian ini tidak hanya digunakan untuk tujuan validasi, namun juga dimanfaatkan untuk keperluan analisis lebih lanjut, khususnya yang berkaitan dengan perancangan fisik dari objek yang diuji. Dengan demikian, pengolahan data pasca pengujian berperan penting baik selama maupun setelah pelaksanaan pengujian terowongan angin.

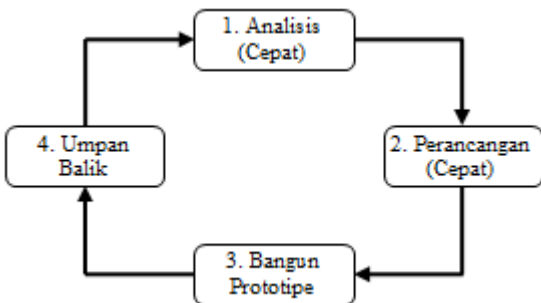
Meski berperan penting, pengolahan data pasca pengujian TAKRI selama ini masih dilakukan secara manual. Dalam hal ini, analis data umumnya melakukan pengolahan data dengan bantuan piranti lunak *spreadsheet* Microsoft Excel dengan memanipulasi satu slot demi satu slot data hasil pengujian, membandingkan hasilnya pada sejumlah *chart* sesuai kebutuhan dan mengkaji hasilnya. Proses seperti ini tidak jarang memicu kesalahan dalam pemasukan data, pemilihan set data dan penyetelan format data. Selain itu, tidak adanya standar presentasi dan analisis data di antara para analis data yang bertugas sering kali membuat pengambil keputusan terkendala dalam memahami hasil kerja analis data yang beragam.

Guna mengatasi berbagai masalah yang diuraikan dalam alinea sebelumnya, maka kegiatan penelitian dan pengembangan ini mengusulkan pengembangan piranti lunak pengolah data pasca

pengujian TAKRI. Piranti lunak yang diusulkan dapat berfungsi sebagai pengolah dan analisis data pasca pengujian secara otomatis dan memenuhi kebutuhan pengolahan data pasca pengujian lainnya. Bagian selanjutnya dari makalah ini menguraikan metodologi rancang bangun piranti lunak pengolah data pasca pengujian TAKRI, membahas rangkaian tahapan rekayasa piranti lunak, dan merangkum seluruh hasil kegiatan di bagian penutup.

II. METODOLOGI

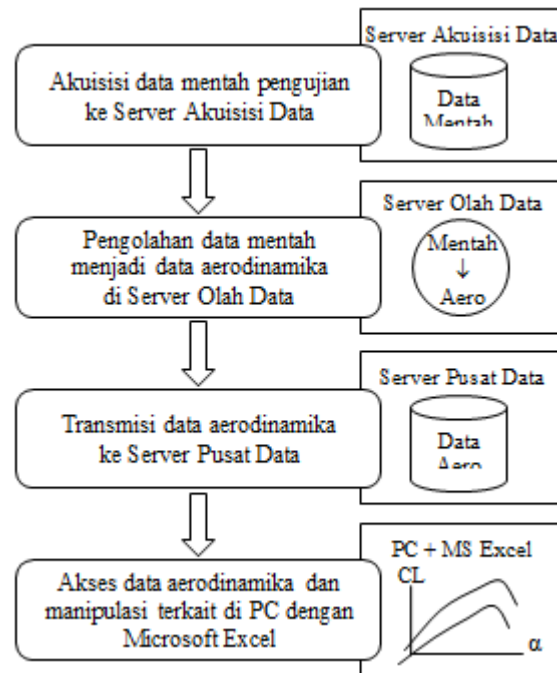
Metodologi yang digunakan dalam kegiatan rancang bangun piranti lunak pengolah data pasca pengujian TAKRI adalah *prototyping*. *Prototyping* merupakan pendekatan rekayasa piranti lunak yang melibatkan pembangunan prototipe atau purwarupa secara bertahap dalam siklus berkurun waktu singkat, seperti yang ditunjukkan dalam Gambar 1, hingga piranti lunak target dirampungkan secara sempurna. Fase analisis diisi dengan kegiatan penggalan kebutuhan piranti lunak melalui observasi, studi literatur dan wawancara terhadap pengguna potensial. Hasil dari fase analisis dituangkan ke dalam rancangan antarmuka grafis, basis data dan algoritma dalam fase perancangan. Penyusunan kode sesuai tiga rancangan tersebut dan kegiatan uji terkait untuk validasi dan verifikasi kompilasi kode kemudian dilaksanakan dalam fase pembangunan prototipe. Selanjutnya prototipe diujicobakan kepada pengguna potensial untuk dimintai pendapat dan masukannya sebagai bahan untuk dikaji dalam fase analisis pada siklus berikutnya.



Gambar 1. Konsep *prototyping*

III. KEGIATAN RANCANG BANGUN

Kegiatan rancang bangun diawali dengan analisis kebutuhan piranti lunak pengolah data pasca pengujian TAKRI melalui studi literatur, wawancara terhadap pengguna potensial (dalam hal ini analisis data dari tim pengujian) dan observasi.



Gambar 2. Mekanisme pengolahan data pasca pengujian

Gambar 2 menunjukkan mekanisme pengolahan data pasca pengujian yang berlaku di TAKRI. Seperti yang ditunjukkan dalam Gambar 2, proses pengujian terowongan angin bermula dengan diakuisisinya data mentah yang merupakan hasil pengukuran instrumentasi ukur ke dalam Server Akuisisi Data. Data mentah selanjutnya diolah di Server Olah Data menjadi data aerodinamika, yang merupakan bentuk data yang dapat dipahami oleh analisis data. Data aerodinamika kemudian ditransmisikan ke Server Pusat Data agar dapat diakses untuk keperluan validasi dan analisis lebih lanjut oleh analisis data pada komputer personal (PC) dengan Microsoft Excel. Kajian strategis terhadap mekanisme pengolahan data pasca pengujian merekomendasikan piranti lunak pengolah data pasca pengujian dibangun pada platform server tersendiri agar tidak mengganggu kinerja server yang telah ada, termasuk Server Pusat Data yang

difungsikan semata untuk menjadi penyimpan berbagai data yang terkait dengan pengujian terowongan angin, dan agar piranti lunak pengolah data pasca pengujian dapat dikembangkan lebih lanjut menjadi berbasis web.

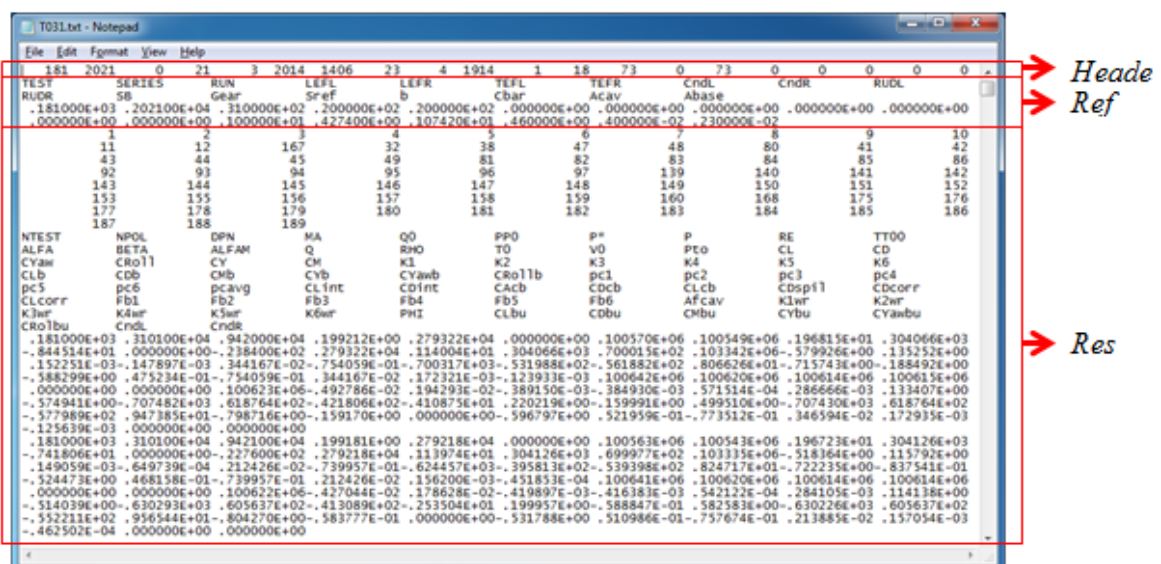
Berdasarkan hasil wawancara terhadap sejumlah analis data, maka fungsionalitas yang dibutuhkan dan harus terdapat dalam piranti lunak pengolah data pasca pengujian dapat diuraikan sebagai berikut :

- Fungsi penampil grafik/plot data aerodinamika,
- Fungsi penampil numerik data aerodinamika,
- Fungsi analisis statistika data aerodinamika,
- Fungsi dukungan pengambilan keputusan berdasarkan hasil analisis data aerodinamika,
- Fungsi pendukung, seperti pencetakan dan penyimpanan data aerodinamika dalam format tertentu.

Sedangkan skenario penggunaan tipikal dari piranti lunak pengolah data pasca pengujian yang dibutuhkan adalah sebagai berikut :

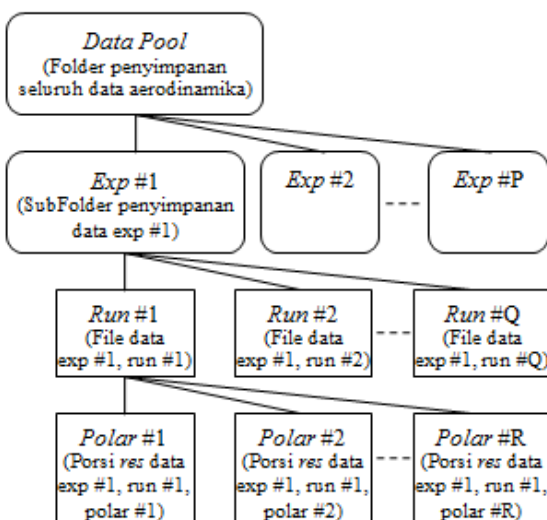
- 1) Pengguna memilih fungsionalitas utama yang ingin dieksekusi, seperti penampil grafik atau penampil numerik,
- 2) Pengguna memilih set data yang ingin dimanipulasi melalui antarmuka yang ramah guna,
- 3) Pengguna melakukan *trigger* eksekusi fungsionalitas yang telah dipilih sebelumnya,
- 4) Pengguna dapat memilih fungsi pendukung (seperti mencetak atau menyimpan data) untuk mengolah data lebih lanjut.

Pemilihan set data yang diuraikan dalam butir kedua dirancang dengan memperhatikan struktur data aerodinamika yang menjadi objek terolah utama dan terbentuk dalam file berformat text, seperti yang ditunjukkan dalam Gambar 3. Seperti ditunjukkan dalam Gambar 3, file data aerodinamika terdiri dari tiga unsur utama, yaitu *header*, data referensi (*ref*) dan data hasil (*res*). Unsur *header* menguraikan atribut utama data aerodinamika, seperti identitas slot uji (nomor uji dan nomor *run*), dan atribut file data aerodinamika, seperti tanggal pembentukan file dan ukuran data referensi. Unsur data referensi memuat data konfigurasi pengujian terowongan angin, seperti



Gambar 3. Struktur file data aerodinamika (1 file untuk 1 run slot uji)

kode konfigurasi dan setelan bagian pembentuk objek yang diuji (seperti sudut *flap* dan kode jenis *landing gear* yang digunakan). Sedangkan unsur data hasil memuat data aerodinamika target yang sesungguhnya (seperti kecepatan hembusan angin dan gaya angkat objek yang diuji). Pemilihan set data umumnya dilakukan dengan menentukan nomor uji (*exp*), nomor *run* dan nomor *polar* yang tersimpan dalam data aerodinamika. Adapun hubungan ketiga nomor identifikasi slot uji ini ditunjukkan dalam rancangan struktur data aerodinamika berikut (Gambar 4).



Gambar 4. Rancangan struktur data aerodinamika dan hirarki hubungan antara nomor uji (*exp*), nomor *run* dan nomor *polar*

Rancangan antarmuka pengguna piranti lunak pengolah data pasca pengujian untuk fungsionalitas plot ditunjukkan dalam Gambar 5. Terdapat tiga panel utama dalam antarmuka pengguna, yaitu *Data Pool*, *Plot Pool* dan *Plot*. Panel *Data Pool* merupakan representasi struktur data aerodinamika pada tataran antarmuka pengguna dan berfungsi sebagai sarana pemilihan set data yang akan diplot oleh pengguna beserta atribut plot melalui komponen *combo box*, seperti nilai untuk sumbu X (*Var X*), nilai untuk sumbu Y (*Var Y*), nilai untuk *legend* dari plot (*Ref 1* hingga *Ref 4*), dan nomor plot tujuan dimana data akan ditampilkan di panel *Plot*. Setelah melakukan pemilihan set data, pengguna dapat menekan

button Add untuk memasukkan set data terpilih ke dalam panel *Plot Pool*. Panel *Plot Pool* berfungsi sebagai *list* dari berbagai set data yang akan diplot, dan memiliki kemampuan untuk memungkinkan pengguna mengedit, menghapus, mengatur atribut plot lebih lanjut dan melakukan *trigger* eksekusi plot (melalui click pada *button Show*) di panel *Plot*. Adapun panel *Plot* berfungsi sebagai area dimana plot ditampilkan sesuai dengan *list* yang terdapat dalam panel *Plot Pool*.

Rancangan antarmuka pengguna piranti lunak pengolah data pasca pengujian untuk fungsionalitas penampil numerik ditunjukkan dalam Gambar 6. Terdapat dua panel utama dalam antarmuka pengguna, yaitu *Data Pool* dan *Print*. Seperti halnya antarmuka untuk fungsionalitas plot, panel *Data Pool* merupakan representasi struktur data aerodinamika pada tataran antarmuka pengguna dan berfungsi sebagai sarana pemilihan set data yang akan ditampilkan data numeriknya oleh pengguna. Sedangkan panel *Print* berfungsi sebagai area untuk menampilkan data numerik sesuai dengan pilihan set data yang terdapat di panel *Data Pool*.

Rancangan algoritma untuk fungsionalitas plot dapat diuraikan dengan menggunakan pseudocode berikut :

```

# Initialization
Clear Item All ComboBox
Read SubFolder in Data Pool
While SubFolder not Null
  ComboBox_Exp.AddItem(SubFolder.
Name)
  
```

DATA POOL

Exp: Run: Pol:
 Ref1: Ref2:
 Ref3: Ref4:
 Var X: Var Y:
 Plot#:

PLOT POOL

No	Exp	Run	Pol	Var X	Var Y	Ref 1	Ref 2	Ref 3	Ref 4	Plot#
1										
2										
3										

Plot#	Title	X-Axis	Y-Axis	Ref 1	Ref 2	Ref 3	Ref 4
1							
2							

PLOT

Line	Exp	Run	Pol	Ref 1	Ref 2	Ref 3	Ref 4

Line	Exp	Run	Pol	Ref 1	Ref 2	Ref 3	Ref 4

Gambar 5. Rancangan antarmuka untuk fungsionalitas penampil grafik (plot)

DATA POOL

Exp: Run: Pol:
 Ref1: Ref2: Ref3: Ref10:
 Res 1: Res 2: Res 3: Res 10:

PRINT

Ref 1	Ref 2	Ref 3	
Value of Ref 1	Value of Ref 2	Value of Ref 3
Res 1	Res 2	Res 3	
Value 1 of Res 1	Value 1 of Res 2	Value 1 of Res 3	
Value 2 of Res 1	Value 2 of Res 2	Value 2 of Res 3	
:	:	:
:	:	:	
:	:	:	
:	:	:	

Gambar 6. Rancangan antarmuka untuk fungsionalitas penampil numerik

```

    Read Next SubFolder
  End While
  ComboBox_Plot#.AddItem(1)
  ComboBox_Plot#.AddItem(2)
  :
End

# ComboBox_Exp.OnChange
  ComboBox_Run.ClearItem
  Read DataFile in ComboBox_Exp.
Text
  While DataFile not Null
    ComboBox_Run.AddItem(DataFile.
Name)
    Read Next DataFile
  End While
End

# ComboBox_Run.OnChange
  ComboBox_Ref1.ClearItem
  :
  ComboBox_Ref4.ClearItem
  ComboBox_VarX.ClearItem
  ComboBox_VarY.ClearItem
  ComboBox_Pol.ClearItem
  DataFile = ComboBox_Run.Text
  Open DataFile
  Read RefName in Reference
  While not End of Reference
    ComboBox_Ref1.AddItem(RefName)
    :
    ComboBox_Ref4.AddItem(RefName)
    Read Next RefName
  End While
  Read ResName in Result
  While not End of Result
    ComboBox_VarX.AddItem(ResName)
    ComboBox_VarY.AddItem(ResName)
    Read Next ResName
  End While
  Read PolNo in Result
  While not End of Result
    ComboBox_Pol.AddItem(PolNo)
    Read Next PolNo
  End While
  Close DataFile
End

# Button_Clear_DataPool.OnClick
  Call Initialization
End

# Button_Add.OnClick
  With PlotPool_List
    .Add(Column = 1,ComboBox_Exp.
Text)
    .Add(Column = 2,ComboBox_Run.
Text)
    :
    .Add(Column = 10,ComboBox_
Plot#.Text)
  End With
End

# Button_Edit.OnClick
  With PlotPool_List
    Row = .SelectedRow
    ComboBox_Exp.Text=.
Cell(Column=1,Row)
    ComboBox_Run.Text=.
Cell(Column=2,Row)
    :
    ComboBox_Plot#.Text=
    .Cell(Column=10,Row)
  End With
End

# Button_Delete.OnClick
  With PlotPool_List
    .DeleteSelectedRow
  End With
End

# Button_Clear_PlotPool.OnClick
  With PlotPool_List
    .DeleteAllPlotRow
  End With
End

# Button_Show.OnClick
  With PlotPool_List
    For I = 1 to .RowCount
      DataFile = 'Data Pool' & '\\' &
        .Cell(1,I) & '\\' &
        .Cell(2,I)
      Open DataFile
      Plot# = .Cell(10,I)
      Ref1 = .Cell(6,I)
      Ref2 = .Cell(7,I)
      :
      Read Ref1, Ref2, ... in
Reference
      While not End of Reference
        AddToLegendArea(Ref1,Ref2,...,Plot#)
        Read Next Ref1, Ref2, ...
      End While
      VarX = .Cell(4,I)
    
```

```

VarY = .Cell(5,I)
Read VarX, VarY in Result
While not End of Result
AddToChartArea(VarX,VarY,Plot#)
Read Next VarX, VarY
End While
Close DataFile
End For
End With
End

```

Rancangan algoritma untuk fungsionalitas penampil numerik sebagian besar sama dengan rancangan algoritma untuk fungsionalitas plot, khususnya untuk *event OnChange* dari seluruh komponen *combo box* yang digunakan untuk memilih set data, dan *event OnClick* dari *button Clear*. Adapun algoritma untuk *event OnClick* dari *button Print* adalah sebagai berikut :

```

# Button_Print.OnClick
DataFile = 'Data Pool' & '\\' &
.Cell(1,I) & '\\' &
.Cell(2,I)

Open DataFile
Ref1 = ComboBox_Ref1.Text
Ref2 = ComboBox_Ref2.Text
:
Read Ref1, Ref2, ... in Reference
While not End of Reference
AddToPrintArea(Ref1, Ref2, ...)
Read Next Ref1, Ref2, ...
End While
Res1 = ComboBox_Res1.Text
Res2 = ComboBox_Res2.Text
:
Read Res1, Res2, ... in Result
While not End of Result
AddToPrintArea(Res1, Res2, ...)
Read Next Res1, Res2, ...
End While
Close DataFile
End

```

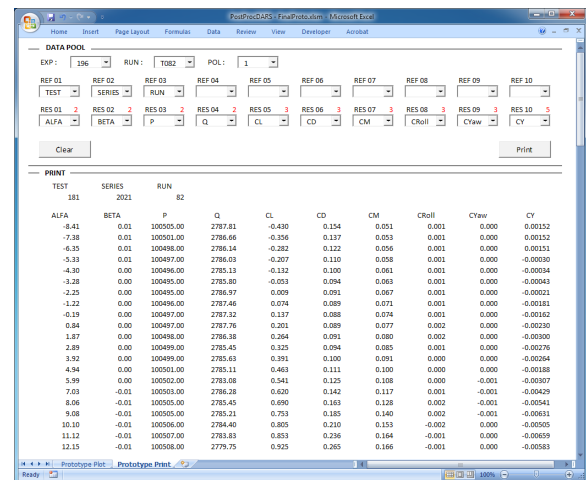
IV. HASIL RANCANG BANGUN

Hasil kegiatan analisis dan perancangan yang diuraikan sebelumnya diimplementasikan dalam kode program secara bertahap, mulai dalam bentuk prototipe hingga menjadi produk operasional, dan diuji sesuai dengan kebutuhan fungsionalitasnya. Gambar 7 dan 8 masing-masing menunjukkan

prototipe untuk fungsionalitas penampil grafik (plot) dan penampil numerik yang dibangun dengan menggunakan Microsoft Excel - Visual Basic for Application (VBA).



Gambar 7. Prototipe fungsionalitas penampil grafik (plot)



Gambar 8. Prototipe fungsionalitas penampil numerik

V. SIMPULAN DAN SARAN

Rancang bangun piranti lunak pengolah data pasca pengujian TAKRI telah dilaksanakan dan menghasilkan prototipe yang dapat dijadikan acuan untuk pengembangan lebih lanjut menuju produk operasional yang bekerja pada platform berbasis web.

DAFTAR PUSTAKA

- [1] J. B. Barlow, W. H. Rae, A. Pope, "Low-Speed Tunnel Testing", 3rd Edition, Wiley, 1999.
- [2] I. Z. Pane, "Pengembangan Prototipe Piranti Lunak Data Post-Processing DARS ILST", Presentasi Ilmiah Forum Fungsional UPT LAGG, 22 Oktober 2015.
- [3] R.S. Pressman, "Software Engineering, A Practitioner's Approach" 6th Edition, McGraw-Hill, 2005.
- [4] I. Z. Pane, "Pemanfaatan Microsoft Excel Sebagai Perangkat Pengembangan Prototipe Piranti Lunak", ULTIMA InfoSys, Vol. VI, No. 1, Juni 2015, pp. 20-26.
- [5] I. Z. Pane, "Aplikasi Microsoft Excel Sebagai Alat Bantu Pembangunan Prototipe Piranti Lunak Berorientasi Sains", Prosiding Seminar Nasional Pengembangan Aktual Teknologi Informasi (SENA BAKTI) 2015, dalam proses submisi.