

IMPLEMENTASI PENENTUAN KEHANDALAN PERANGKAT LUNAK A-TOOLIPS (PERANGKAT LUNAK UNTUK PENYANDANG TUNA RUNGU) BERDASARKAN DATA *FAILURE*

Bambang Pudjoatmodjo¹⁾, Retno Novi Dayawati²⁾, Mahmud Dwi Sulistiyo³⁾

^{1), 2)}D3 Teknik Informatika Universitas Telkom

³⁾SI Teknik Informatika Universitas Telkom

Jl Telekomunikasi, Terusan Buah Batu, Bandung

Email : b.pudjoatmodjo@gmail.com¹⁾, retno.novi@tass.telkomuniversity.ac.id²⁾, mahmuddwis@telkomuniversity.ac.id³⁾

Abstrak

Perangkat Lunak A-tool lips merupakan aplikasi bantu berbasis android bagi tuna rungu dalam berkomunikasi dengan individu lain. Aplikasi ini menerjemahkan suara menjadi teks yang dapat dibaca oleh penyandang tuna rungu. Dengan aplikasi ini, orang normal yang berhadapan dengan penyandang tuna rungu tidak mengalami kesulitan untuk mengatakan atau mengutarakan maksudnya (tanpa harus menggunakan bahasa isyarat). Sebelum diterapkan secara nyata di lapangan diperlukan suatu cara untuk mengetahui nilai kehandalan perangkat yang dibuat. Penentuan nilai kehandalan diperlukan untuk mengetahui peluang seberapa baik perangkat lunak yang dibuat sehingga dapat berfungsi dengan baik dalam rentang waktu penggunaan tertentu. Penentuan nilai kehandalan didasarkan kepada data *failure* yang tercatat. Nilai kehandalan yang diperoleh dirujuk ke skala Cronbach Alpha's.

Nilai Kehandalan yang diperoleh untuk aplikasi A-Toolips ini adalah 97,90%. Nilai kehandalan ini memberikan informasi bahwa aplikasi bersifat tangguh pada saat digunakan oleh user.

Kata kunci: kehandalan, peluang, Cronbach Alpha's

1. Pendahuluan

A-TooLips adalah aplikasi pembelajaran berbasis Android yang digunakan untuk mempelajari dan melafalkan kata (produksi kata) bagi anak tuna rungu atau keterbatasan pendengaran. Tujuan aplikasi ini adalah agar mereka yang memiliki keterbatasan pendengaran dapat berlatih untuk menambah perbendaharaan kata sehingga dapat berkomunikasi dengan orang lain.

Pembangunan aplikasi A-Toolips ini telah melewati tahapan pembuatan perangkat lunak [5] :

- Analisis kebutuhan pengguna
- Perancangan perangkat lunak

c. Pembangunan (development) perangkat lunak

d. Pengujian perangkat lunak

Setelah melewati ke-empat tahapan tersebut, dilakukan satu proses uji kehandalan perangkat lunak A-TooLips.

Penentuan nilai kehandalan ini merupakan lanjutan dari kajian yang telah dilakukan sebelumnya [2]. Kajian sebelumnya adalah menentukan nilai kehandalan yang berdasarkan data sekunder [2] dengan tujuan untuk melakukan kajian, apakah dengan memanfaatkan data *failure* dapat ditentukan nilai kehandalan suatu perangkat lunak. Pada penentuan nilai kehandalan yang dilakukan saat ini diterapkan pada kasus sebenarnya yaitu dengan cara mengumpulkan data berdasarkan pencatatan data *failure* yang terjadi.

Kehandalan adalah peluang dari keberhasilan atau peluang bahwa sistem akan memberikan performa yang sesuai fungsinya berdasarkan batasan desain yang telah dibuat [7][11][12]. Konsekuensi dari performa yang baik adalah sistem yang dibuat mempunyai tingkat kesalahan (*error*), kekurangan (*fault*) dan kegagalan (*failure*) yang minimal.

Adapun pengertian dari kesalahan (*error*), kekurangan (*fault*) dan kegagalan (*failure*) adalah sebagai berikut:

- Error* merupakan kekeliruan yang diakibatkan oleh programmer atau software designer misalnya kekeliruan memasukkan input. Selain itu, *error* dapat merupakan perilaku sistem yang tidak sesuai dengan spesifikasinya [6][7].
- Fault* merupakan perwujudan dari *error* dalam kode program. Pada banyak kasus *fault* ini dapat diidentifikasi dan dihilangkan, tetapi pada beberapa kasus *fault* hanya berhenti pada tahap hipotesis saja dan tidak dapat dibuktikan [6][7].
- Failure* merupakan suatu peristiwa yang terjadi ketika pengguna mendapatkan bahwa program berhenti memberikan layanan yang diharapkan. *Failure* diakibatkan masukan yang tidak benar

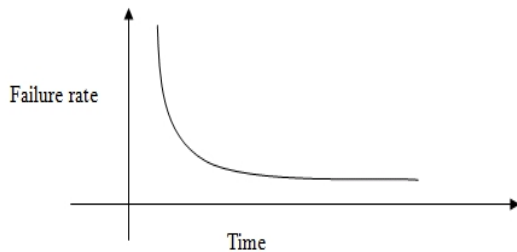
sehingga memberikan output yang tidak sesuai [6][7][10].

Kehandalan perangkat lunak diperlukan untuk menjamin perangkat lunak yang dibuat dapat berfungsi dengan baik pada saat digunakan oleh pengguna. Adapun pengertian kehandalan perangkat lunak adalah peluang dari keberhasilan atau peluang bahwa sistem akan memberikan performa yang sesuai fungsinya berdasarkan batasan desain yang telah dibuat. Secara matematis Kehandalan $R(t)$ adalah peluang bahwa sistem akan bekerja dengan baik pada interval dari waktu 0 sampai dengan t, dinotasikan sebagai berikut [7] :

$$R(t) = P(T > t) \quad t \geq 0 \quad (1)$$

dimana T adalah peubah acak yang menunjukkan waktu kegagalan.

Dari sudut pandangan kehandalan, perangkat lunak tidak pernah mengalami keausan arena umur atau waktu pemakaian. Seperti pada gambar berikut ini



Gambar 1. Laju kegagalan perangkat lunak terhadap waktu

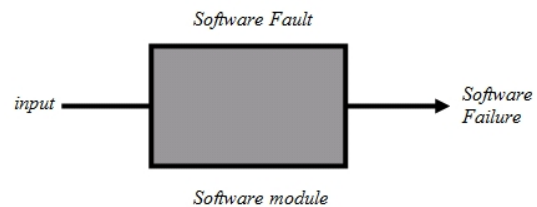
Kehandalan adalah peluang dari keberhasilan atau peluang bahwa sistem akan memberikan performa yang sesuai fungsinya berdasarkan batasan desain yang telah dibuat [7][11][12].

Beberapa hal yang berkaitan dengan kehandalan perangkat lunak adalah sebagai berikut [4][6]:

1. Perangkat lunak mempunyai sifat yang dapat dikembangkan. Pengembangan satu atau lebih baris kode akan mengakibatkan perilaku yang berbeda terhadap perangkat lunak tersebut. Hal ini tentunya mempengaruhi kehandalan perangkat lunak.
2. Kehandalan perangkat lunak tidak bergantung pada waktu pemakaiannya, melainkan bergantung pada jumlah dan kualitas koreksi yang dilakukan.
3. Faktor internal yang dapat mempengaruhi kehandalan perangkat lunak antara lain kecepatan prosesor dan ukuran memori.
4. Banyaknya error dalam program akan berkurang seiring dengan perbaikan yang dilakukan pada perangkat lunak.

Berdasarkan pembahasan kehandalan perangkat lunak di atas, maka dirancang sebuah proses untuk menentukan nilai kehandalan perangkat lunak.

Proses penentuan nilai kehandalan perangkat lunak didasarkan kepada *failure* yang terjadi dari produk A-tooLips. Pengertian dari *failure* adalah suatu peristiwa yang terjadi ketika pengguna mendapatkan bahwa program berhenti memberikan layanan yang diharapkan. *Failure* diakibatkan masukan yang tidak benar sehingga memberikan output yang tidak sesuai [6][7][10] .



Gambar 2. Kaitan antara kesalahan dan kegagalan perangkat lunak [6]

Setelah data berhasil diperoleh, akan dilakukan proses perhitungan untuk menentukan kehandalan dari produk A-tooLips.

Untuk menentukan kehandalan perangkat lunak dilakukan langkah-langkah [2] :

1. Catat data kegagalan (*failure*)
2. Penaksiran Parameter [8][9]

$$\hat{\lambda} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2)$$

Penaksir merupakan penaksir tak bias. Penaksir yang baik memenuhi kondisi sbb:

- a. Tak bias

Ditunjukkan dengan pembuktian berikut :

akan ditunjukkan bahwa penaksir yang digunakan, penaksir likelihood ($\hat{\lambda}_{MLH}$) yang digunakan tak bias.

Akan ditunjukkan $E[\hat{\lambda}] = \lambda$

diketahui bahwa $\hat{\lambda} = \bar{X}$

$$E[\hat{\lambda}] = E[\bar{X}]$$

$$= E\left[\frac{\sum_{i=1}^n x_i}{n}\right]$$

$$= \frac{1}{n} E\left[\sum_{i=1}^n x_i\right]$$

$$= \frac{1}{n} E[x_1 + x_2 + \dots + x_n]$$

Jadi $E[\hat{\lambda}] = \lambda$

Jadi $\hat{\lambda}$ merupakan penaksir tak bias bagi λ .

Jadi \bar{X} merupakan penaksir tak bias.

- b. Efisien

Ditunjukkan penaksir merupakan penaksir yang efisien

Diketahui $E\left[\hat{\lambda}\right] = \lambda$. Penaksir tak bias

$$\begin{aligned} \text{Var}\left[\hat{\lambda}\right] &= \text{Var}\left[\frac{1}{n}\sum_{i=1}^n x_i\right] \\ &= \frac{1}{n^2}\text{Var}\left[\sum_{i=1}^n x_i\right] \\ &= \frac{1}{n^2}\text{Var}(x_1 + x_2 + \dots + x_n) \\ &= \frac{1}{n^2}\{\text{Var}(x_1) + \text{Var}(x_2) + \dots + \text{Var}(x_n)\} \\ &= \frac{1}{n^2}[\lambda + \lambda + \dots + \lambda] \\ &= \frac{1}{n^2}(n\lambda) \\ &= \frac{\lambda}{n} \end{aligned}$$

Jadi, $\text{Var}\left[\hat{\lambda}\right] = \frac{\lambda}{n}$

$$\begin{aligned} \ln f(x) &= \ln\left[\frac{e^{-\lambda} \lambda^x}{x!}\right] \\ &= -\lambda \ln e + x \ln \lambda - \ln x! \\ &= -\lambda + x \ln \lambda - \ln x! \\ \frac{\partial}{\partial \lambda} \ln f(x) &= -1 + \frac{x}{\lambda} \\ I(\hat{\theta}) &= E\left[\frac{\partial^2}{\partial x^2} \ln f(x)\right] \\ &= E\left[-\frac{x}{\lambda^2}\right] \\ &= -\frac{1}{\lambda^2} E[x] \\ &= -\frac{\lambda}{\lambda^2} \\ &= -\frac{1}{\lambda} \end{aligned}$$

$$LRC = \frac{1}{-nI(\hat{\theta})} = \frac{\lambda}{n}$$

$$e = \frac{LRC}{\text{Var}(\hat{\lambda})} = \frac{\lambda/n}{\lambda/n} = 1, \text{ jadi variansi minimum.}$$

Karena $e = 1$ (Variansi minimum) dan

$E(\hat{\lambda}) = \lambda$ (penaksir tak bias), maka

$\hat{\lambda} = \bar{X}$ adalah penaksir efisien untuk λ .

c. Konsisten

Ditunjukkan bahwa penaksir konsisten, akan ditunjukkan

$$\lim_{n \rightarrow \infty} P\left(\left|\hat{\lambda} - \lambda\right| < k \sqrt{\text{Var}(\hat{\lambda})}\right) \geq \lim_{n \rightarrow \infty} \left[1 - \frac{1}{k^2}\right]$$

$$\text{Pilih } \varepsilon^2 = k^2 \sigma_{\hat{\lambda}}^2$$

$$k^2 = \frac{\varepsilon^2}{\sigma_{\hat{\lambda}}^2} = \frac{\varepsilon^2}{\frac{\lambda}{n}} = \frac{n\varepsilon^2}{\lambda}$$

$$\varepsilon = k \sqrt{\frac{\lambda}{n}}$$

$$\lim_{n \rightarrow \infty} P\left(\left|\hat{\lambda} - \lambda\right| < \varepsilon\right) = \lim_{n \rightarrow \infty} 1 - \lim_{n \rightarrow \infty} \frac{\lambda}{n\varepsilon^2} = 1 - 0 = 1$$

Jadi, terbukti bahwa $\hat{\lambda}$ adalah penaksir yang konsisten untuk λ .

3. Menghitung nilai kehandalan dengan cara [1][7]:
 a. Menentukan peluang kegagalan

$$P(X = x) = \frac{(\lambda t)^x e^{-\lambda t}}{x!} \tag{3}$$

Untuk $x = 0, 1, \dots, x_{\max}$

b. Gunakan

$$R(x) = \sum_{x=0}^k \frac{(\lambda t)^x e^{-\lambda t}}{x!} \tag{4}$$

untuk menghitung nilai kehandalan

4. Kualifikasi Cronbach Alpha

Dari nilai nilai kehandalan yang diperoleh akan menentukan apakah produk sudah layak untuk dilepas kepada pengguna ataukah masih harus ditinjau kembali. Untuk memastikan hal tersebut, nilai kehandalan yang diperoleh akan dirujuk ke skala cronbach alpha (langkah ke-4).

Salah satu alat yang digunakan untuk melakukan kualifikasi kehandalan adalah kualifikasi Cronbach's Alpha [3] dengan kriteria sbb:

- | | |
|-------------|------------------------|
| a. 91%-100% | <i>Excellent</i> |
| b. 81%-90% | <i>Good</i> |
| c. 71%-80% | <i>Acceptable</i> |
| d. 61%-70% | <i>Questionable</i> |
| e. 51%-60% | <i>Poor</i> |
| f. 0%-50% | <i>Unacceptable</i> |
| g. 80% | <i>Reasonable goal</i> |

berdasarkan kualifikasi di atas, nilai kehandalan yang diharapkan minimal berada pada kualifikasi Acceptable atau bernilai 71%-80%.

2. Pembahasan

Pada aplikasi A-Toolips, terdapat fungsionalitas mengenal suara yang berguna untuk merubah dari

suara menjadi teks. Fungsionalitas ini menjadi perhatian untuk menentukan nilai kehandalan pada aplikasi A-Toolips.

Data failure yang dicatat mempunyai ketentuan, sesuai dengan definisi [7], yaitu :

- Kondisi pengujian dilakukan pada saat aplikasi A-toolips dijalankan tidak dengan aplikasi lain
- Kondisi pengujian dilakukan pada saat aplikasi A-toolips dijalankan dibarengi dengan aplikasi lain
- Failure dicatat pada saat pengucapan suara tidak menghasilkan teks yang sesuai dengan soal. Atau
- Failure dicatat pada saat pengucapan suara tidak dapat ditangkap oleh aplikasi.

Berikut ini adalah hasil pencatatan data failure.

Tabel 1. Tabel Data Failure dan Kumulatif Failure

Waktu Pengujian	Failure	Kumulatif Failure
1	1	1
2	1	2
3	1	3
4	1	4
5	2	6
6	2	8
7	1	9
8	1	10
9	1	11
10	1	12
11	2	14
12	2	16
13	1	17
14	2	19
15	1	20
16	1	21
17	2	23
18	2	25
19	1	26
20	1	27

Proses pencatatan data kegagalan dihentikan pada waktu pengujian ke 20. Penghentian pencatatan kegagalan didasarkan pada persentase kumulatif kegagalan yang mendekati 10% atau berada di bawah 10% [2]. Adapun perhitungan persentase kumulatif kegagalan sebagai berikut :

$$y = \frac{\text{kumulatif failure}}{\text{jumlah total kumulatif failure}} \times 100\% \\ = \frac{27}{274} \times 100\% = 9,85\%$$

Dari tabel data failure dan kumulatif failure di dapat informasi tentang rata-rata failure yang terjadi yaitu 1,35. Ini memberikan informasi bahwa rata-rata failure yang terjadi adalah 1 dalam waktu pengujian.

Setelah melakukan pencatatan data failure dan menentukan nilai rata-rata, dilanjutkan dengan

menentukan parameter penaksir yang diperoleh dari persamaan (2) yaitu $\lambda = 1$. Penaksir yang diambil merupakan penaksir yang baik.

Berdasarkan persamaan (3) dihitung peluang kegagalan yang terjadi. Peluang kegagalan diharapkan kecil, karena pada dasarnya pembuatan perangkat lunak dirancang untuk meminimalkan terjadinya kegagalan.

Peluang kegagalan dari pencatatan data failure, dapat dilihat pada table berikut ini :

Tabel 2. Tabel Perhitungan Peluang Kegagalan

x	$P_r = \frac{\lambda^x e^{-\lambda}}{x!}$
1	0,367
1	0,367
1	0,367
1	0,367
2	0,184
2	0,184
1	0,367
1	0,367
1	0,367
1	0,367
2	0,184
2	0,184
1	0,367
2	0,184
1	0,367
1	0,367
2	0,184
2	0,184
1	0,367
1	0,367

Hasil perhitungan nilai kehandalan berdasarkan persamaan (3) dan (4) sebagai berikut.

x	0	1	2	3	Kehandalan
P(X=x)	0.367	0.367	0.184	0.061	97,90%

Gambar 3. Tabel Perhitungan Kehandalan

Kehandalan yang diperoleh berdasarkan persamaan (4), yaitu $R(x) = 0,367 + 0,367 + 0,184 + 0,061 = 97,90\%$. Nilai kehandalan 97,90% tersebut termasuk kategori (*excellent*) berdasarkan kualifikasi Cronbach's Alpha. Dengan nilai tersebut, dinyatakan bahwa perangkat lunak tersebut tangguh dan layak diserahkan kepada pihak pengguna.

3. Kesimpulan

- Nilai kehandalan yang didapat memberikan informasi bahwa aplikasi A-Toolips dapat berfungsi dengan baik pada saat digunakan oleh pengguna

2. Nilai kehandalan memberikan informasi bahwa apabila aplikasi memberikan respon yang tidak sesuai, yaitu tidak menampilkan teks sesuai input yang diberikan akan bisa melakukan recovery untuk memberikan respon yang sesuai dengan input yang diberikan.

Teknik Informatika IT Telkom, lulus tahun 2012. Saat ini menjadi Dosen di prodi S1 Teknik Informatika Universitas Telkom.

Daftar Pustaka

- [1]. Achmad Zanbar Soleh, "Ilmu Statistika, Cetakan Pertama, Penerbit Rekayasa Sains", Bandung, 2005
- [2]. Bambang Pudjoatmodjo, "Penentuan Kehandalan Perangkat Lunak dengan Menggunakan Distribusi Poisson dan Kualifikasi Cronbach's Alpha", Tesis, 2010, Universitas Langlangbuana.
- [3]. George, D., and Mallery, P., SPSS for Windows step by step: A simple guide andreference. 11.0 update (4th ed.). Boston: Allyn & Bacon, 2003
- [4]. Jack Febrian, "Pengetahuan Komputer dan Teknologi Informasi", Cetakan Pertama, Penerbit Informatika, Bandung 1994
- [5]. Ian, Sommerville, "Software Engineering", Jilid 1, Edisi 6, Penerbit Erlangga, Jakarta, 2003
- [6]. Ilkka Karanta, "Methods and Problems of Software Reliability Estimation, VTT working paper 63", Julkaisija-Utgivare-Publisher, 2006
- [7]. Hoang Pham, "System Software Reliability, Springer", Verlag – London, 2006.
- [8]. Nar., Herrhyanto, Gantini Tuti, "Pengantar Statistika Matematis", Cetakan Pertama, Yrama Widya, Bandung, 2009.
- [9]. Nar., Herrhyanto, Gantini Tuti, Pengantar Statistika Matematis, Cetakan Pertama, Yrama Widya, Bandung, 2009.
- [10]. Ram, Chillarege, "What Is Software Failure, IEEE Transactions on Reliability", Vol.45, No.3, 1996 September.
- [11]. Tsong-Lyu, Michael R., "Software Reliability Theory, Encyclopedia of Software Engineering, 2002", John Wiley & Sons, Article Online Update Posting Date: January 15, 2002
- [12]. Valido-Cabrera, Eduardo, "Software Reliability Methods, Report", Technical University of Madrid, Agustus 2006

Biodata Penulis

Bambang Pudjoatmodjo, memperoleh gelar Sarjana Matematika (S.Si), Jurusan Matematika Institut Teknologi Bnadung, lulus tahun 1998. Memperoleh gelar Magister Teknik Informatika (MT) Program Pasca Sarjana Magister Teknik Informatika Universitas Langlangbuana, lulus tahun 2010. Saat ini menjadi Dosen di Prodi D3 Teknik Informatika Universitas Telkom

Retno Novi Dayawati, memperoleh gelar Sarjana Ilmu Komputer (S.Si) di Universitas Padjajaran, lulus pada tahun 1999. Memperoleh gelar Magister Komputer (MT) Program Pasca Sarjana Magister Teknik Informatika Bidang Minat Rekayasa Perangkat Lunak Institut Teknologi Bandung, lulus tahun 2005. Saat ini menjadi Dosen di Prodi D3 Teknik Informatika Universitas Telkom

Mahmud Dwi Sulistiyo, memperoleh gelar Sarjana Teknik Informatika (S.T), Jurusan Teknik Informatika IT Telkom, lulus pada tahun 2010. Memperoleh gelar Magister Teknik Informatika IT Telkom (M.T) Program Pasca Sarjana Magister

