

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pembahasan dari bab-bab sebelumnya beserta dengan lampiran yang telah dibuat, maka dapat ditarik kesimpulan dari pembangunan aplikasi *windows registry recovery system*, yaitu :

1. Perangkat lunak *WinR2S "Windows Registry Recovery System"* ini telah berhasil dibangun dengan tujuan untuk memperbaiki dan mengembalikan informasi *registry windows* ke dalam keadaan standar sebelum dimanipulasi yang dilakukan secara otomatis oleh sistem, sehingga memudahkan *user* dalam memperbaiki *registry windows* yang rusak, karena *user* tidak perlu mengetikkan *script* secara manual.
2. Untuk memudahkan *user* perangkat lunak *WinR2S* dibangun dengan meminimalkan campur tangan *user*, perangkat lunak *WinR2S* terbagi menjadi dua yaitu *WinR2S Server* yang diinstal di komputer *server* dan *WinR2S Client* yang diinstal di beberapa komputer *client* yang terkoneksi dengan jaringan, atau dapat juga diinstal pada *stand alone computer*.
3. Untuk mengembalikan *registry Windows* dilakukan secara otomatis oleh sistem *WinR2S Client* dengan persetujuan *WinR2S Server*, sehingga perbaikan *registry* dilakukan tanpa mempertimbangkan *privilege user* yang *login*.

5.2 Saran

Beberapa saran yang dapat ditarik dari proses analisis hingga pembuatan dokumentasi Tugas Akhir ini dan untuk pengembangan lebih lanjut pembangunan aplikasi *windows registry recovery system* adalah sebagai berikut :

1. Pada perangkat lunak ini *administrator* dalam mengecek kerusakan *registry* harus melihat ke komputer *client*, untuk pengembangan selanjutnya diharapkan dikembangkan suatu mekanisme dimana *administrator* tidak perlu lagi *survey* ke komputer *client*. Pengembangan dilakukan dengan cara menambahkan fasilitas untuk *download log file* yang ada di komputer *client*.
2. Untuk pengembangan perangkat lunak ini, dapat dilakukan penambahan *database registry* yang digunakan apabila terdapat *registry* baru yang belum pernah terdeteksi.
3. *Administrator* dapat melakukan pengesetan waktu untuk pengecekan *registry* melalui komputer *server*.

DAFTAR PUSTAKA

- Albahari, Ben, Drayton, Meter, Merril, Brad, 2001, *C# Essentials*, O'reilly, U.S.A.
- Amperiyanto, Tri, 2001, *Bermain-main dengan Registry Windows*, PT Elex Media Komputindo, Jakarta.
- Budiharto, Widodo, Sukmadi, Dodi Yogi, 2004, *Aplikasi e-Commerce Menggunakan Visual C# .NET*, Andi, Yogyakarta.
- Inixindo, 2002, *Optimizing Network and Traffic management On LAN/WAN using Cisco Network*.
- Jainudin, 2006, *Belajar Sendiri .NET dengan Visual C# 2005*, Andi, Yogyakarta.
- Nugroho, Adi, 2005, *Rational Rose untuk Pemodelan Berorientasi Objek*, Informatika, Bandung.
- Kimmel, Paul, 2002, *Advanced C# Programming*, McGraw-Hill/Osborne, U.S.A.
- Kurniawan, Agus, 2005, *Pemrograman Dasar Socket Dengan C#*, Indonesia.net Developer Community, Depok.
- Proboyekti, Umi, 2004, *Rekayasa Software Client/Server*, UKDW, Yogyakarta.
- Schildt, Herbert, 2002, *The Complete Reference C#*,

McGraw-Hill, U.S.A.

Troelsen, Andrew, 2001, *C# and The .NET Platform*,
Apress, Berkeley.



LAMP IRAN



SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK


Windows Registry Recovery System (WinR2S)

Disusun oleh:

Yusuf Theretsa Patiku

02 07 03485

**Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika Universitas Atma Jaya Yogyakarta	Nomor Dokumen		Halaman
		<i>SKPL- WinR2S</i>		<i>1/30</i>
		Revisi	<i>A</i>	<i>07/05/2007</i>

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	Perubahan Use Case Diagram Perubahan Collaboration Diagram
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F
Ditulis oleh	YTP						
Diperiksa oleh	KA YSP						
Disetujui oleh	KA YSP						

NOTASI DOKUMEN

Notasi yang digunakan dalam dokumen ini adalah sebagai berikut :

- Teks normal ditulis dalam font Times New Roman 12 pt, plain.
- Teks yang ditulis dalam font **bold** merupakan teks yang mengacu pada bab, sub-bab, gambar, atau tabel dalam dokumen ini.
- Teks yang ditulis dalam font Courier New merupakan teks yang mengacu pada model, diagram, atau file yang disebutkan dalam dokumen ini.



DAFTAR ISI

1	Pendahuluan	8
1.1	Tujuan	8
1.2	Lingkup Masalah	8
1.3	Definisi Istilah dan Singkatan	8
1.4	Referensi	9
1.5	Deskripsi Umum (Overview)	9
2	Deskripsi Global Perangkat Lunak	10
2.1	Perspektif Produk	10
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	12
2.4	Batasan-batasan	13
2.5	Kebutuhan Fungsionalitas Perangkat Lunak	14
2.5.1	Use Case Diagram WinR2S Client	14
2.5.1.1	Use Case : Connect	14
2.5.1.2	Use Case : Check Registry	15
2.5.1.3	Use Case : Repair Registry	15
2.5.1.4	Use Case : Send Message	15
2.5.1.5	Use Case : Input IP Server	15
2.5.2	Use Case Diagram WinR2S Server	16
2.5.2.1	Use Case : Monitor Client & Registry Status	16
2.5.2.2	Use Case : Send Message	16
2.5.2.3	Use Case : Set Close Client Application	16
2.6	Karakteristik Aktor	16
2.6.1	System	16
2.6.2	Administrator	16
3	Deskripsi Rinci Kebutuhan	17
3.1	Spesifikasi Kebutuhan Fungsionalitas	17
3.1.1	Spesifikasi Use Case : Connect	17
3.1.2	Spesifikasi Use Case : Check Registry	18
3.1.3	Spesifikasi Use Case : Repair Registry	19
3.1.4	Spesifikasi Use Case : Send Message	20
3.1.5	Spesifikasi Use Case : Input IP Server	21
3.1.6	Spesifikasi Use Case : Monitor Client & Registry Status	22
3.1.7	Spesifikasi Use Case : Send Message	23
3.1.8	Spesifikasi Use Case : Set Close Client Application	24
3.2	Spesifikasi Kebutuhan Non-Fungsionalitas	24
3.2.1	Kebutuhan Antarmuka Eksternal	24
3.2.2	Antarmuka Pemakai	25
3.2.3	Antarmuka Perangkat Keras	25
3.2.4	Antarmuka Perangkat Lunak	25
4	Realisasi Use Case	26
4.1	Interaction Diagram	26
4.1.1	Analysis Collaboration Diagram : Use Case Connect	26
4.1.2	Analysis Collaboration Diagram : Use Case Check Registry	26
4.1.3	Analysis Collaboration Diagram : Use Case Repair Registry	27
4.1.4	Analysis Collaboration Diagram : Use Case Send Message	27
4.1.5	Analysis Collaboration Diagram : Use Case Input IP Server	27
4.1.6	Analysis Collaboration Diagram : Use Case Monitor Client & Registry Status	28

4.1.7	Analysis Collaboration Diagram : Use Case Send Message	28
4.1.8	Analysis Collaboration Diagram : Use Case Set Close Client Application	28
Apendiks A : Daftar Istilah dan Singkatan		29



DAFTAR GAMBAR

<i>Gambar 2.1 Use Case Diagram.....</i>	<i>14</i>
<i>Gambar 4.1 Analysis Collaboration Diagram : Use Case Connect.....</i>	<i>26</i>
<i>Gambar 4.2 Analysis Collaboration Diagram : Use Case Check Registry.....</i>	<i>26</i>
<i>Gambar 4.3 Analysis Collaboration Diagram : Use Case Repair Registry.....</i>	<i>27</i>
<i>Gambar 4.4 Analysis Collaboration Diagram : Use Case Send Message.....</i>	<i>27</i>
<i>Gambar 4.5 Analysis Collaboration Diagram : Use Case Input IP Server.....</i>	<i>27</i>
<i>Gambar 4.6 Analysis Collaboration Diagram : Use Case Monitor Client & Registry Status.....</i>	<i>28</i>
<i>Gambar 4.7 Analysis Collaboration Diagram : Use Case Send Message.....</i>	<i>28</i>
<i>Gambar 4.8 Analysis Collaboration Diagram : Use Case Send Message.....</i>	<i>28</i>



DAFTAR TABEL

<i>Tabel 3.1 Spesifikasi Use Case : Connect.....</i>	<i>17</i>
<i>Tabel 3.2 Spesifikasi Use Case : Check Registry.....</i>	<i>18</i>
<i>Tabel 3.3 Spesifikasi Use Case : Repair Registry.....</i>	<i>19</i>
<i>Tabel 3.4 Spesifikasi Use Case : Send Message</i>	<i>20</i>
<i>Tabel 3.5 Spesifikasi Use Case : Input IP Server.....</i>	<i>21</i>
<i>Tabel 3.6 Spesifikasi Use Case : Monitor Client & Registry Status</i>	<i>22</i>
<i>Tabel 3.7 Spesifikasi Use Case : Send Message</i>	<i>23</i>
<i>Tabel 3.8 Spesifikasi Use Case : Set Close Client Application.....</i>	<i>24</i>



1 Pendahuluan

1.1 Tujuan

Dokumen ini berisi penjelasan tentang Spesifikasi Kebutuhan Perangkat Lunak (SKPL). Dokumen SKPL ini merupakan dokumen spesifikasi kebutuhan perangkat lunak untuk *Windows Registry Recovery System (WinR2S)* versi 1.0 yang akan dibangun. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis untuk pengembangan perangkat lunak. Dalam SKPL ini akan dijelaskan mengenai kebutuhan-kebutuhan yang harus tersedia agar perangkat lunak yang diharapkan dapat terwujud.

1.2 Lingkup Masalah

Perangkat lunak *WinR2S* dikembangkan dengan tujuan untuk :

1. Mengaktifkan komputer *server* dan komputer *client*.
2. Melakukan koneksi antara *client* dan *server*.
3. Memonitor *client* dan status *registry*.
4. Mengirim *message* dalam bentuk *string* antara *client* dan *server*.
5. Melakukan pengecekan *registry* pada komputer *client*.
6. Memperbaiki *registry* yang rusak atau berubah.
7. Melakukan penutupan aplikasi *client* oleh *administrator*.

1.3 Definisi Istilah dan Singkatan

Definisi, istilah dan singkatan yang digunakan dalam dokumen ini mengacu pada **Apendiks A : Daftar Istilah dan Singkatan.**

1.4 Referensi

Referensi yang digunakan dalam pembuatan dokumen ini adalah :

1. Nugroho, Adi. *Rational Rose untuk Pemodelan Berorientasi Objek*. Informatika. 2005.
2. Ivar Jacobson, Grady Booch, James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley. 1998.
3. Martin Fowler, Kendall Scott. *UML Distilled – Second Edition*. Addison Wesley. 1999.
4. Stephen R Schach. *An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process*. Mc Graw-Hill. 2004.
5. Wendy Boggs, Michael Boggs. *Mastering UML With Rational Rose*. 2002.

1.5 Deskripsi Umum (Overview)

Dokumen SKPL ini dibagi menjadi empat bab. Bab pertama adalah **Pendahuluan**, yang berisi tentang deskripsi dokumen. Bab kedua adalah **Deskripsi Global Perangkat Lunak**, yang berisi penjelasan secara umum mengenai sistem yang akan dikembangkan meliputi fungsi-fungsi dari sistem, karakteristik pengguna, batasan dan asumsi yang diambil dalam pengembangan perangkat lunak. Bab ketiga adalah **Deskripsi Rinci Kebutuhan**, yang berisi penjelasan tentang kebutuhan sistem yang akan dikembangkan secara lebih rinci. Bab keempat adalah **Realisasi Use Case**, yang berisi realisasi use case dalam tahap analisis (konseptual), yang akan digunakan sebagai dasar realisasi use case pada tahap desain.

2 Deskripsi Global Perangkat Lunak

2.1 Perspektif Produk

WinR2S adalah perangkat lunak yang digunakan untuk memperbaiki atau mengembalikan informasi *registry windows* yang rusak atau berubah ke keadaan standar yang diterapkan pada jaringan komputer. Pengguna tidak perlu repot memperbaiki *registry* secara manual baik dengan cara mengimpor *registry* yang bagus maupun langsung mengakses *registry windows* melalui program *Regedit (Registry Editor)* yang merupakan program bawaan dari *windows* sendiri. Perangkat lunak ini dapat berjalan pada komputer *stand alone* maupun komputer yang terhubung dalam jaringan.

Pengguna sistem *WinR2S* adalah *Administrator*. *Administrator* berinteraksi dengan sistem yang ditampilkan di layar komputer dengan pilihan menu maupun sub menu. Sistem *WinR2S* terbagi dalam *form WinR2S Server* dan *form WinR2S Client*. Secara garis besar proses diawali dengan mengaktifkan komputer *server* diikuti dengan komputer *client*. Pada saat pengaktifan komputer *client* akan muncul *textbox* yang digunakan untuk menginputkan *IP Server*. Komputer *server* akan menunggu koneksi dari *client*, sedangkan komputer *client* akan menghubungi komputer *server*. Komputer *server* akan memonitor *client* dan status *registry*-nya. Komputer *client* akan melakukan *check registry*, jika terdapat perubahan *registry* maka komputer *client* akan mengirim pesan ke komputer *server* bahwa *registry* berubah. Komputer *server* yang menerima pesan dari komputer *client* akan membalas pesan tersebut untuk memperbaiki *registry* yang rusak atau berubah. Setelah *registry* berhasil diperbaiki maka komputer *client* akan mengirim pesan ke komputer *server* bahwa *registry* telah diperbaiki. *Check registry* dapat dilakukan pada komputer *client* walaupun komputer *client* tersebut tidak terkoneksi dengan komputer *server*, sistem pada *WinR2S Client* dapat melakukan *check registry* secara otomatis setiap 5 menit dengan maksud agar kondisi *registry* dapat terus terpantau. *Administrator*

juga dapat melakukan *set close client application* yaitu menutup aplikasi komputer *client*.

Menu yang digunakan dalam perangkat lunak *WinR2S Server* adalah menu *File*, *Client Menu* dan *Help*. Didalam menu *File* terdapat sub menu *Exit* yang digunakan untuk keluar dari perangkat lunak. Di dalam menu *Client Menu* terdapat sub menu *Close Client Application* yang digunakan untuk menutup aplikasi pada komputer *client*. Sedangkan dalam menu *Help* terdapat sub menu *WinR2S Help* dan *About WinR2S*. Untuk mengetahui petunjuk cara menjalankan perangkat lunak, dapat memilih *WinR2S Help*, dan untuk mengetahui tentang versi atau seputar perangkat lunak, dapat melihat *About WinR2S*. Pada perangkat lunak *WinR2S Client* tidak terdapat pilihan menu karena perangkat lunak ini diminimalkan adanya campur tangan *user*.

Perangkat lunak *WinR2S* akan dibangun dengan menggunakan bahasa pemrograman *C# . Net*. Pengguna akan berinteraksi dengan perangkat lunak melalui antarmuka berbasis GUI (*Graphical User Interface*), sehingga piranti masukan yang digunakan adalah *mouse* dan *keyboard*.

2.2 Fungsi Produk

Fungsi produk perangkat lunak *WinR2S Client* adalah sebagai berikut :

1. Fungsi *Connect (SKPL-WinR2S-01)* adalah fungsi yang digunakan untuk melakukan koneksi baik dari komputer *server* maupun komputer *client*.
2. Fungsi *Check Registry (SKPL-WinR2S-02)* adalah fungsi yang digunakan untuk melakukan pengecekan *registry* pada komputer *client*.
3. Fungsi *Repair Registry (SKPL-WinR2S-03)* adalah fungsi yang digunakan untuk memperbaiki *registry* pada komputer *client*.

4. Fungsi *Send Message* (SKPL-WinR2S-04) adalah fungsi yang digunakan untuk mengirimkan pesan dalam bentuk *string* dari komputer *client* ke komputer *server*.
5. Fungsi *Input IP Server* (SKPL-WinR2S-05) adalah fungsi yang digunakan untuk menginputkan *IP Server* oleh Administrator.

Fungsi produk perangkat lunak *WinR2S Server* adalah sebagai berikut :

1. Fungsi *Monitor Client & Registry Status* (SKPL-WinR2S-06) adalah fungsi yang digunakan untuk memonitor komputer *client* dan memonitor status *registry*.
2. Fungsi *Send Message* (SKPL-WinR2S-07) adalah fungsi yang digunakan untuk mengirimkan pesan dalam bentuk *string* dari komputer *server* ke komputer *client*.
3. Fungsi *Set Close Client Application* (SKPL-WinR2S-08) adalah fungsi yang digunakan untuk menutup aplikasi pada komputer *client* dari komputer *server*.

2.3 Karakteristik Pengguna

Pengguna perangkat lunak *WinR2S* tersebut adalah *Administrator* dengan karakteristik sebagai berikut :

1. Memahami pengoperasian komputer secara aktif.
2. Memahami sistem komputer tempat perangkat lunak dijalankan.
3. Mengerti dan memahami perangkat lunak yang digunakan.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak *WinR2S* tersebut adalah :

1. Kebijakan Umum

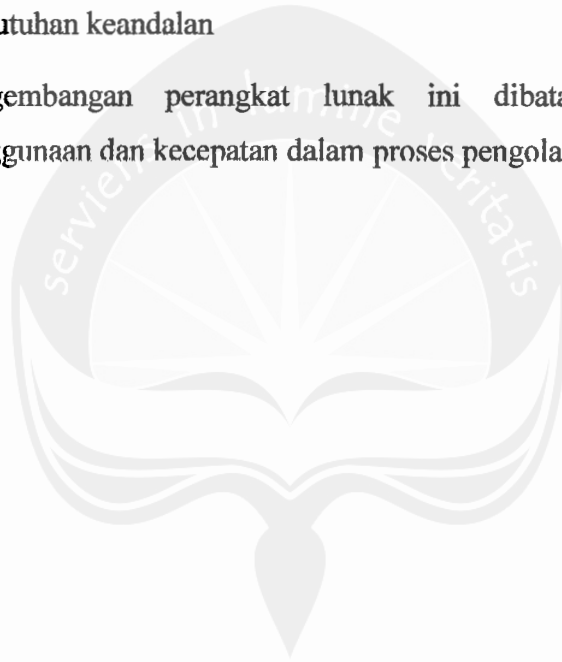
Berpedoman pada tujuan dari pengembangan perangkat lunak *WinR2S*.

2. Keterbatasan perangkat keras

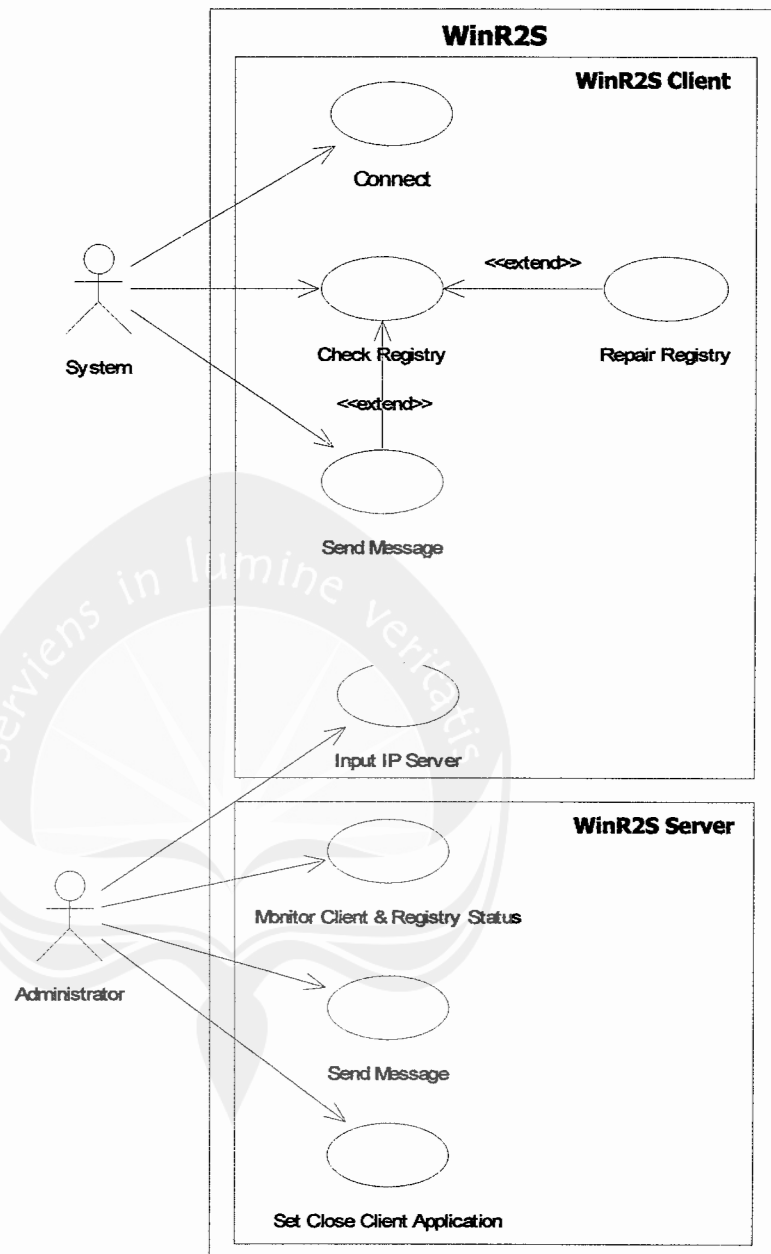
Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

3. Kebutuhan keandalan

Pengembangan perangkat lunak ini dibatasi pada kemudahan penggunaan dan kecepatan dalam proses pengolahannya.



2.5 Kebutuhan Fungsionalitas Perangkat Lunak



Gambar 2.1 Use Case Diagram

2.5.1 Use Case Diagram WinR2S Client

2.5.1.1 Use Case : Connect

Use case ini digunakan oleh aktor untuk melakukan koneksi baik dari komputer *server* maupun komputer *client*.

Lihat : Spesifikasi Use Case : Connect

2.5.1.2 Use Case : Check Registry

Use case ini digunakan untuk melakukan pengecekan *registry* di komputer *client*. Pengecekan *registry* dapat dilakukan walaupun komputer *client* tidak terkoneksi ke komputer *server*. Pengecekan dilakukan secara otomatis setiap 5 menit, ini dimaksudkan agar apabila terjadi kerusakan atau perubahan *registry* dapat langsung diketahui.

Lihat : Spesifikasi Use Case : Check Registry

2.5.1.3 Use Case : Repair Registry

Use case ini digunakan oleh aktor untuk memperbaiki *registry* pada komputer *client*. Perbaikan *registry* dilakukan apabila sudah mendapat persetujuan dari *server* untuk memperbaiki *registry*.

Lihat : Spesifikasi Use Case : Repair Registry

2.5.1.4 Use Case : Send Message

Use case ini digunakan untuk mengirimkan pesan dalam bentuk *string* dari komputer *client* ke komputer *server*.

Lihat : Spesifikasi Use Case : Send Message

2.5.1.5 Use Case : Input IP Server

Use case ini digunakan untuk menginputkan *IP Server*.

Lihat : Spesifikasi Use Case : Input IP Server

2.5.2 Use Case Diagram WinR2S Server

2.5.2.1 Use Case : Monitor Client & Registry Status

Use case ini digunakan untuk memonitor komputer *client* dan memonitor status *registry* pada komputer *client*.

Lihat : Spesifikasi Use Case : Monitor Client & Registry Status

2.5.2.2 Use Case : Send Message

Use case ini digunakan untuk mengirimkan pesan dalam bentuk *string* dari komputer *server* ke komputer *client*.

Lihat : Spesifikasi Use Case : Send Message

2.5.2.3 Use Case : Set Close Client Application

Use case ini digunakan untuk menutup aplikasi pada komputer *client* yang dilakukan dari komputer *server* oleh *Administrator*.

Lihat : Spesifikasi Use Case : Set Close Client Application

2.6 Karakteristik Aktor

2.6.1 System

System harus memenuhi hal-hal berikut :

- *System* yang dimaksud di sini adalah sistem dari perangkat lunak pada sisi *client*, dimana dalam sistem ini diminimalkan campur tangan dari pengguna/*user*.

2.6.2 Administrator

Administrator harus memenuhi hal-hal berikut :

- Dapat mengoperasikan perangkat lunak *WinR2S*.
- Dapat mengkoneksikan komputer *server* ke komputer *client*.

- Dapat memutus koneksi dari *server* ke *client* atau sebaliknya
- Memahami tentang jaringan komputer.
- Memahami tentang *registry* komputer.

3 Deskripsi Rinci Kebutuhan

3.1 Spesifikasi Kebutuhan Fungsionalitas

3.1.1 Spesifikasi Use Case : Connect

Tabel 3.1 Spesifikasi Use Case : Connect

Use Case ID	UC- <i>WinR2S</i> -01
Use Case Name	Connect
Use Case Type	Essential
Priority	High
Actors	System
Description	Use case ini digunakan oleh aktor untuk melakukan koneksi dari komputer <i>client</i> maupun komputer <i>server</i> .
Preconditions	Aktor sudah berhasil memasuki sistem.
Basic Path	<ol style="list-style-type: none"> 1. Sistem menampilkan <i>form WinR2S Server</i> dan/atau <i>form WinR2S Client</i>. 2. Sistem pada <i>form WinR2S Server</i> akan terkoneksi dengan <i>form WinR2S Client</i> dan/atau sebaliknya. 3. Pada <i>list view WinR2S Client</i> akan muncul <i>Registry Name, Registry Location, Value</i> dan <i>Default</i>. 4. Pada <i>list view WinR2S Server</i> akan muncul <i>Host Name, IP Number, Information, Date & Time,</i> dan <i>Message</i>.
Alternative Paths	-
Postconditions	<i>WinR2S Client</i> akan terhubung atau terkoneksi dengan <i>WinR2S Server</i> atau sebaliknya.

Exception Paths	<ol style="list-style-type: none"> 1. Pembatalan koneksi (setelah Tabel 3.1 Basic Path 2) <ol style="list-style-type: none"> a. Sistem akan memutus koneksi Tabel 3.8 Basic Path 1. b. Kembali ke Tabel 3.1 Basic Path 1.
Extends	-
Includes	-

3.1.2 Spesifikasi Use Case : Check Registry

Tabel 3.2 Spesifikasi Use Case : Check Registry

Use Case ID	UC-WinR2S-02
Use Case Name	Check Registry
Use Case Type	Essential
Priority	High
Actors	System
Description	Use case ini digunakan untuk melakukan pengecekan <i>registry</i> . Pengecekan dilakukan secara otomatis oleh sistem setiap 5 menit. Pengecekan <i>registry</i> dapat dilakukan oleh komputer <i>client</i> walaupun belum/tanpa terkoneksi dengan komputer <i>server</i> .
Preconditions	Aktor sudah berhasil memasuki sistem. Sistem pada komputer <i>client</i> sudah terkoneksi ataupun belum terkoneksi dengan komputer <i>server</i> .
Basic Path	<ol style="list-style-type: none"> 1. Sistem akan melakukan pengecekan <i>registry</i> secara otomatis. 2. Sistem akan memberi informasi "<i>Repair Registry</i>" apabila terdapat perubahan atau kerusakan <i>registry</i>. 3. Sistem akan tetap melakukan pengecekan <i>registry</i>

	secara otomatis apabila tidak terdapat perubahan atau kerusakan <i>registry</i> .
Alternative Paths	-
Postconditions	<ol style="list-style-type: none"> 1. Sistem akan menampilkan informasi berupa <i>string</i> “<i>Repair Registry</i>” apabila terdapat perubahan atau kerusakan <i>registry</i>. 2. Sistem akan menampilkan informasi berupa <i>string</i> “<i>Registry Normal</i>” apabila tidak terdapat perubahan atau kerusakan <i>registry</i>. 3. Sistem akan menampilkan informasi berupa <i>string</i> “<i>Registry Repaired</i>” apabila <i>registry</i> baru saja diperbaiki.
Exception Paths	<ol style="list-style-type: none"> 1. <i>Registry</i> tidak mengalami perubahan atau kerusakan (setelah Tabel 3.2 Basic Path 2) <ol style="list-style-type: none"> a. Sistem akan melakukan <i>check registry</i> secara otomatis. b. Kembali ke Tabel 3.2 Basic Path 3.
Extends	-
Includes	-

3.1.3 Spesifikasi Use Case : Repair Registry

Tabel 3.3 Spesifikasi Use Case : Repair Registry

Use Case ID	UC-WinR2S-03
Use Case Name	Repair Registry
Use Case Type	Essential
Priority	High
Actors	System
Description	Use case ini digunakan untuk memperbaiki <i>registry</i> yang telah berubah atau rusak.

Preconditions	<ol style="list-style-type: none"> 1. Use Case : Check Registry sudah dilaksanakan sehingga sudah diketahui <i>registry</i> yang rusak atau berubah. 2. Use Case : Sending Message sudah dilaksanakan sehingga mendapat persetujuan dari <i>WinR2S Server</i> untuk memperbaiki <i>registry</i>.
Basic Path	<ol style="list-style-type: none"> 1. Sistem <i>WinR2S Client</i> akan memanggil prosedur <i>RepairRegistry</i> setelah menerima <i>string</i> "<i>Repair Registry</i>" dari <i>WinR2S Server</i>. 2. Sistem <i>WinR2S Client</i> akan mengirim <i>string</i> "<i>Registry Repaired</i>" ke <i>WinR2S Server</i> setelah berhasil memperbaiki perubahan atau kerusakan <i>registry</i>.
Alternative Paths	-
Postconditions	<i>Registry</i> yang rusak atau berubah sudah berhasil diperbaiki.
Exception Paths	-
Extends	Use Case : Check Registry
Includes	-

3.1.4 Spesifikasi Use Case : Send Message

Tabel 3.4 Spesifikasi Use Case : Send Message

Use Case ID	UC- <i>WinR2S</i> -04
Use Case Name	Send Message
Use Case Type	Essential
Priority	High
Actors	System
Description	Use case ini digunakan untuk mengirimkan pesan dari komputer <i>client</i> ke komputer <i>server</i> dalam bentuk <i>string</i> .

Preconditions	<ol style="list-style-type: none"> 1. Use Case : Connect sudah dilaksanakan sehingga <i>WinR2S Client</i> terkoneksi ke <i>WinR2S Server</i>. 2. Use Case : Check Registry sudah dilaksanakan oleh <i>WinR2S Client</i>.
Basic Path	<ol style="list-style-type: none"> 1. Sistem <i>WinR2S Client</i> akan mengirim <i>message</i> dalam bentuk <i>string</i> "<i>Registry Normal</i>" ke sistem <i>WinR2S Server</i> apabila <i>registry</i> dalam keadaan normal. 2. Sistem <i>WinR2S Client</i> akan mengirim <i>message</i> dalam bentuk <i>string</i> "<i>Repair Registry</i>" ke sistem <i>WinR2S Server</i> apabila <i>registry</i> bermasalah. 3. Sistem <i>WinR2S Client</i> akan mengirimkan <i>message</i> dalam bentuk <i>string</i> "<i>Registry Repaired</i>" ke sistem <i>WinR2S Server</i> apabila <i>registry</i> sudah diperbaiki.
Alternative Paths	-
Postconditions	<i>Message</i> dalam bentuk <i>string</i> " <i>Registry Normal</i> ", " <i>Repair Registry</i> ", dan " <i>Registry Repaired</i> " dari <i>WinR2S Client</i> ke <i>WinR2S Server</i> terkirim.
Exception Paths	-
Extends	Use Case : Check Registry
Includes	-

3.1.5 Spesifikasi Use Case : Input IP Server

Tabel 3.5 Spesifikasi Use Case : Input IP Server

Use Case ID	UC- <i>WinR2S</i> -05
Use Case Name	Input IP Server
Use Case Type	Essential
Priority	High

Actors	Administrator
Description	Use case ini digunakan untuk mengisi <i>IP Server</i> .
Preconditions	Aktor sudah berhasil memasuki sistem.
Basic Path	<ol style="list-style-type: none"> 1. Sistem akan menampilkan <i>textbox</i> yang digunakan untuk menginputkan <i>IP Server</i>. 2. Aktor menginputkan <i>IP Server</i> dalam <i>textbox</i> kemudian menekan <i>Enter</i>.
Alternative Paths	-
Postconditions	Aktor berhasil menginputkan <i>IP Server</i> .
Exception Paths	-
Extends	-
Includes	-

3.1.6 Spesifikasi Use Case : Monitor Client & Registry Status

Tabel 3.6 Spesifikasi Use Case : Monitor Client & Registry Status

Use Case ID	UC-WinR2S-06
Use Case Name	Monitor Client & Registry Status
Use Case Type	Essential
Priority	High
Actors	Administrator
Description	Use case ini digunakan untuk melihat <i>client</i> yang aktif dari <i>Host Name, IP Number, Information, Date & Time</i> dan <i>Message</i> .
Preconditions	<ol style="list-style-type: none"> 1. Use Case : Connect sudah dilaksanakan sehingga <i>WinR2S Server</i> terkoneksi ke <i>WinR2S Client</i>. 2. Use Case : Input IP Server sudah dilaksanakan oleh <i>WinR2S Server</i>.
Basic Path	<ol style="list-style-type: none"> 1. Sistem menunggu koneksi dari <i>client</i> yang berupa <i>HostName, IP Number, Information, Date &</i>

	<i>Time, dan Message.</i>
Alternative Paths	-
Postconditions	Sistem pada <i>client</i> terkoneksi ke <i>server</i> sehingga <i>HostName, IP Number, Information, Date & Time, dan Message</i> muncul pada <i>list view</i> .
Exception Paths	-
Extends	-
Includes	-

3.1.7 Spesifikasi Use Case : Send Message

Tabel 3.7 Spesifikasi Use Case : Send Message

Use Case ID	UC-WinR2S-07
Use Case Name	Send Message
Use Case Type	Essential
Priority	High
Actors	Administrator
Description	Use case ini digunakan untuk mengirimkan pesan dari komputer <i>server</i> ke komputer <i>client</i> dalam bentuk <i>string</i> .
Preconditions	1. Use Case : Connect sudah dilaksanakan sehingga <i>WinR2S Server</i> terkoneksi ke <i>WinR2S Client</i> .
Basic Path	1. Sistem <i>WinR2S Server</i> akan mengirim <i>message</i> dalam bentuk <i>string</i> " <i>Repair Registry</i> " ke sistem <i>WinR2S Client</i> apabila <i>registry</i> bermasalah.
Alternative Paths	-
Postconditions	<i>Message</i> dalam bentuk <i>string</i> " <i>Registry Repaired</i> " dari <i>WinR2S Server</i> ke <i>WinR2S Client</i> terkirim.
Exception Paths	-
Extends	-
Includes	-

3.1.8 Spesifikasi Use Case : Set Close Client Application

Tabel 3.8 Spesifikasi Use Case : Set Close Client Application

Use Case ID	UC-WinR2S-08
Use Case Name	Set Close Client Application
Use Case Type	Essential
Priority	High
Actors	Administrator
Description	Use case ini digunakan untuk menutup aplikasi pada komputer <i>client</i> .
Preconditions	1. Use Case : Connect sudah dilaksanakan sehingga <i>WinR2S Server</i> terkoneksi ke <i>WinR2S Client</i> .
Basic Path	1. Aktor memilih menu <i>Client Menu</i> dan sub menu <i>Close Client Application</i> . 2. Aktor memilih <i>Host Name client</i> yang akan ditutup pada <i>list view Choose Client</i> . 3. Aktor menekan <i>button OK</i> .
Alternative Paths	-
Postconditions	Aplikasi pada komputer <i>client</i> tertutup.
Exception Paths	1. Aktor membatalkan penutupan aplikasi pada komputer <i>client</i> (setelah Tabel 3.8 Basic Path 2) a. Aktor menekan <i>button Cancel</i> . b. Kembali ke Tabel 3.8 Basic Path 1.
Extends	-
Includes	-

3.2 Spesifikasi Kebutuhan Non-Fungsionalitas

3.2.1 Kebutuhan Antarmuka Eksternal

Kebutuhan antar muka eksternal pada sistem *WinR2S* mencakup kebutuhan antarmuka pemakai, antarmuka perangkat keras dan antarmuka perangkat lunak.

3.2.2 Antarmuka Pemakai

Pemakai berinteraksi langsung dengan sistem *WinR2S* dengan antarmuka berbasis GUI. Piranti masukan yang digunakan untuk memasukkan data masukan adalah *keyboard* dan *mouse*. Sedangkan keluaran dari sistem berupa *list view* yang terdapat beberapa informasi baik dari sisi *server* maupun *client* serta beberapa *string message* yang ditampilkan langsung ke layar *monitor*.

3.2.3 Antarmuka Perangkat Keras

Antarmuka perangkat keras yang dibutuhkan dalam penggunaan perangkat lunak *WinR2S* adalah:

- Beberapa komputer yang terkoneksi lewat jaringan
- *Keyboard*
- *Mouse*

3.2.4 Antarmuka Perangkat Lunak

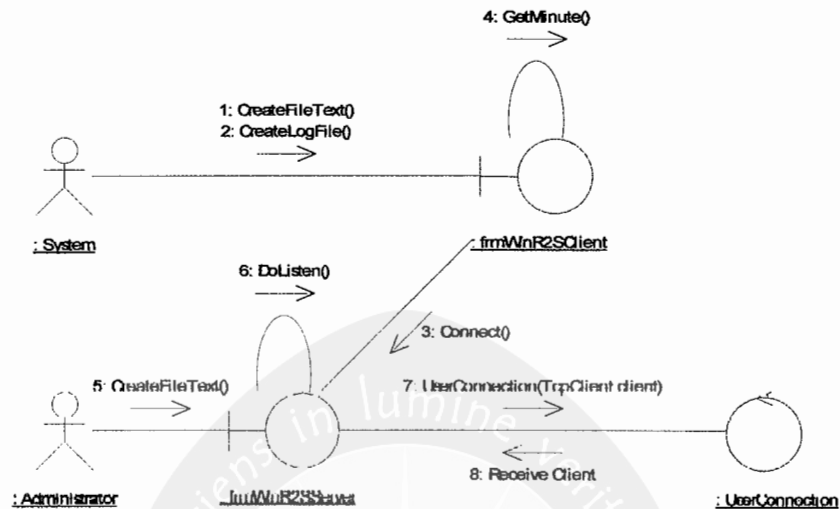
Perangkat lunak yang dibutuhkan untuk mendukung berjalannya perangkat lunak *WinR2S* adalah :

- Nama : *Windows Me/NT/2000/XP*
Sumber : *Microsoft*
Fungsi : Sistem Operasi Komputer
- Nama : *C#. NET*
Sumber : *Microsoft*
Fungsi : *Tool* perancang antarmuka aplikasi

4 Realisasi Use Case

4.1 Interaction Diagram

4.1.1 Analysis Collaboration Diagram : Use Case Connect



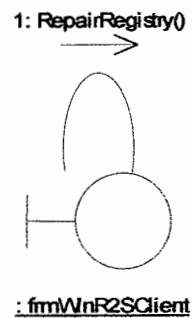
Gambar 4.1 Analysis Collaboration Diagram : Use Case Connect

4.1.2 Analysis Collaboration Diagram : Use Case Check Registry



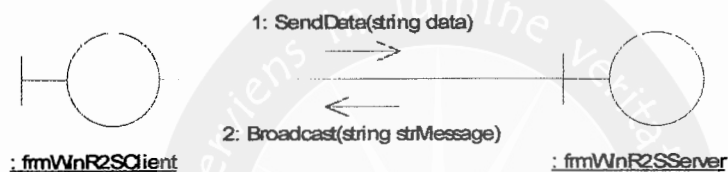
Gambar 4.2 Analysis Collaboration Diagram : Use Case Check Registry

4.1.3 Analysis Collaboration Diagram : Use Case Repair Registry



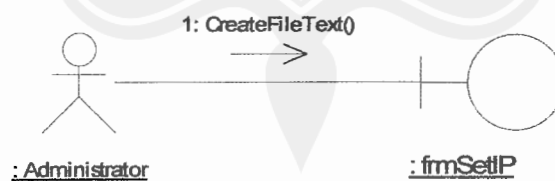
Gambar 4.3 Analysis Collaboration Diagram : Use Case Repair Registry

4.1.4 Analysis Collaboration Diagram : Use Case Send Message



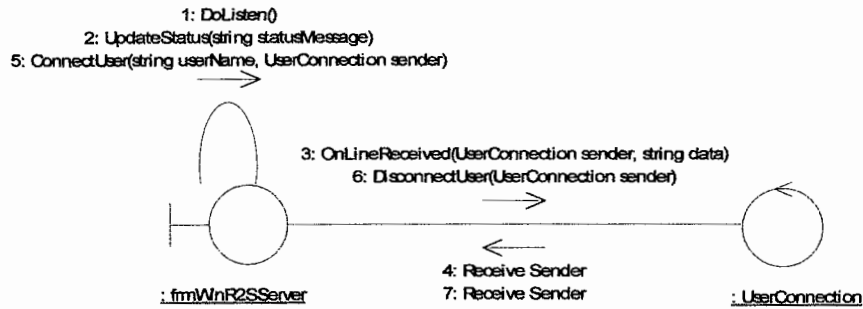
Gambar 4.4 Analysis Collaboration Diagram : Use Case Send Message

4.1.5 Analysis Collaboration Diagram : Use Case Input IP Server



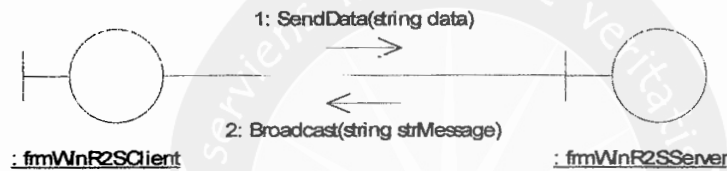
Gambar 4.5 Analysis Collaboration Diagram : Use Case Input IP Server

4.1.6 Analysis Collaboration Diagram : Use Case Monitor Client & Registry Status



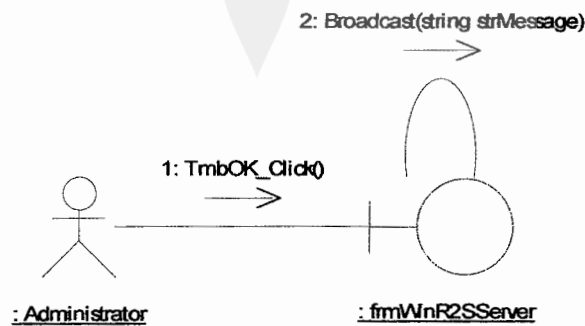
Gambar 4.6 Analysis Collaboration Diagram : Use Case Monitor Client & Registry Status

4.1.7 Analysis Collaboration Diagram : Use Case Send Message



Gambar 4.7 Analysis Collaboration Diagram : Use Case Send Message

4.1.8 Analysis Collaboration Diagram : Use Case Set Close Client Application



Gambar 4.8 Analysis Collaboration Diagram : Use Case Send Message

Apendiks A : Daftar Istilah dan Singkatan

Actor adalah representasi pemakai sistem (manusia atau sistem yang lain).

Administrator adalah pengguna perangkat lunak *WinR2S Server*, meliputi **Use Case : Connect, Use Case : Sending Message, Use Case : Disconnect.**

C# adalah bahasa pemrograman yang sederhana, moderen, berorientasi objek dan type-safe, yang diturunkan dari C dan C++. C# menggabungkan produktifitas dari Visual Basic dan keampuhan dari C++.

Collaboration Diagram adalah diagram yang menggambarkan interaksi antar kelas dan asosiasi. Dimodelkan sebagai pertukaran pesan antar kelas melalui asosiasi kelas dan merupakan diagram interaksi yang menekankan pada organisasi objek yang mengirim dan menerima pesan.

Extend adalah relasi yang memungkinkan suatu use case memiliki kemungkinan untuk memperluas fungsionalitas yang disediakan use case yang lain, tetapi tidak harus terjadi apa yang diharapkan.

Include adalah relasi yang memungkinkan suatu use case untuk menggunakan fungsionalitas yang disediakan oleh use case lain.

Microsoft Visual Studio .NET adalah sebuah *platform* untuk membangun, menjalankan dan meningkatkan generasi lanjut dari aplikasi terdistribusi. Referensi lengkap mengenai Microsoft Visual Studio .NET dapat dilihat pada <http://msdn.microsoft.com/vstudio/>

Registry adalah suatu set *file* data yang dipakai untuk membantu *Windows* mengontrol perangkat keras, perangkat lunak, lingkungan pemakai, dan *interface*.

Use Case adalah representasi fungsionalitas atau layanan yang diberikan sistem kepada pemakai.

Use Case Diagram adalah penggambaran fungsionalitas sistem serta pemakai sistem.

User adalah pengguna perangkat lunak *WinR2S Client*, meliputi **Use Case : Connect, Use Case : Registry Checking, Use Case : Sending Message, Use Case : Repaired Registry, Use Case : Disconnect.**

WinR2S adalah singkatan dari *Windows Registry Recovery System*, yaitu perangkat lunak yang digunakan untuk memperbaiki atau mengembalikan informasi *registry windows* yang rusak atau berubah ke keadaan standar yang diterapkan pada jaringan komputer.

WinR2S Client adalah perangkat lunak WinR2S pada sisi client, dijalankan pada komputer client.

WinR2S Server adalah perangkat lunak WinR2S pada sisi server, yang dikelola oleh Administrator.



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Windows Registry Recovery System (WinR2S)




Disusun oleh:

Yusuf Theretsa Patiku

02 07 03485

**Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika Universitas Atma Jaya Yogyakarta	Nomor Dokumen		Halaman
		<i>DPPL- WinR2S</i>		<i>1/30</i>
		Revisi		<i>07/05/2007</i>

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F
Ditulis oleh	YTP						
Diperiksa oleh	KA YSP						
Disetujui oleh	KA YSP						

NOTASI DOKUMEN

Notasi yang digunakan dalam dokumen ini adalah sebagai berikut :

- Teks normal ditulis dalam font Times New Roman 12 pt, plain.
- Teks yang ditulis dalam font **bold** merupakan teks yang mengacu pada bab, sub-bab, gambar, atau tabel dalam dokumen ini.
- Teks yang ditulis dalam font `Courier New` merupakan teks yang mengacu pada model, diagram, atau file yang disebutkan dalam dokumen ini.



DAFTAR ISI

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah.....	6
1.3	Definisi, Akronim, dan Singkatan.....	6
1.4	Referensi	6
1.5	Deskripsi Umum (Overview).....	7
2	Deskripsi Perancangan Arsitektural	7
2.1	Design Class.....	7
2.1.1	Pengantar.....	7
2.1.2	Package Dependencies	8
2.1.3	Package WinR2SServer	8
2.1.3.1	Class Diagram Package WinR2SServer	8
2.1.3.2	Class frmWinR2SServer	8
2.1.3.3	Class UserConnection	10
2.1.4	Package WinR2SClient	11
2.1.4.1	Class Diagram Package WinR2SClient.....	11
2.1.4.2	Class frmSetIP	12
2.1.4.3	Class frmWinR2SClient	13
2.2	Realisasi Use Case	15
2.2.1	Use Case : Connect	15
2.2.2	Use Case : Check Registry	16
2.2.3	Use Case : Repair Registry.....	17
2.2.4	Use Case : Send Message.....	18
2.2.5	Use Case : Input IP Server	19
2.2.6	Use Case : Monitor Client & Registry Status.....	20
2.2.7	Use Case : Send Message.....	21
2.2.8	Use Case : Set Close Client Application	22
3	Deskripsi Perancangan Antarmuka	23
3.1	Antarmuka WinR2S Server.....	23
3.2	Antarmuka WinR2S Client	25
3.3	Antarmuka Close Client Application	26
3.4	Antarmuka Enter IP Server	27
3.5	Antarmuka WinR2S Help	28
3.6	Antamuka About WinR2S	29
	Apendiks A : Daftar Istilah dan Singkatan.....	30

DAFTAR GAMBAR

<i>Gambar 2.1 Package Dependencies</i>	8
<i>Gambar 2.2 Class Diagram Package WinR2SServer</i>	8
<i>Gambar 2.3 Class frmWinR2SServer</i>	8
<i>Gambar 2.4 Class UserConnection</i>	10
<i>Gambar 2.5 Class Diagram Package WinR2SClient</i>	11
<i>Gambar 2.6 Class frmSetIP</i>	12
<i>Gambar 2.7 Class frmWinR2SClient</i>	13
<i>Gambar 2.8 Design Sequence Diagram : Use Case Connect</i>	15
<i>Gambar 2.9 Design Sequence Diagram : Use Case Check Registry</i>	16
<i>Gambar 2.10 Design Sequence Diagram : Use Case Repair Registry</i>	17
<i>Gambar 2.11 Design Sequence Diagram : Use Case Send Message</i>	18
<i>Gambar 2.12 Design Sequence Diagram : Use Case Input IP Server</i>	19
<i>Gambar 2.13 Design Sequence Diagram : Use Case Monitor Client & Registry Status</i>	20
<i>Gambar 2.14 Design Sequence Diagram : Use Case Send Message</i>	21
<i>Gambar 2.15 Design Sequence Diagram : Use Case Set Close Client Application</i>	22
<i>Gambar 3.1 Rancangan Antarmuka WinR2S Server</i>	23
<i>Gambar 3.2 Rancangan Antarmuka WinR2S Client</i>	25
<i>Gambar 3.3 Rancangan Antarmuka Close Client Application</i>	26
<i>Gambar 3.4 Rancangan Antarmuka Enter IP Server</i>	27
<i>Gambar 3.5 Rancangan Antarmuka WinR2S Help</i>	28
<i>Gambar 3.6 Rancangan Antarmuka About WinR2S</i>	29

1 Pendahuluan

1.1 Tujuan

Dokumen ini berisi penjelasan tentang Deskripsi Perancangan Perangkat Lunak (DPPL). Dokumen DPPL ini merupakan dokumen deskripsi perancangan perangkat lunak untuk *Windows Registry Recovery System (WinR2S)* versi 1.0 yang akan dibangun. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis untuk pengembangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat lunak *WinR2S* dikembangkan dengan tujuan untuk :

1. Mengaktifkan komputer *server* dan komputer *client*.
2. Melakukan koneksi antara *client* dan *server*.
3. Memonitor *client* dan status *registry*.
4. Mengirim *message* dalam bentuk *string* antara *client* dan *server*.
5. Melakukan pengecekan *registry* pada komputer *client*.
6. Memperbaiki *registry* yang rusak atau berubah.
7. Melakukan penutupan aplikasi *client* oleh *administrator*.

1.3 Definisi, Akronim, dan Singkatan

Untuk definisi istilah dan singkatan yang digunakan dalam dokumen ini dapat mengacu pada **Apendiks A : Daftar Istilah dan Singkatan**.

1.4 Referensi

Referensi yang digunakan dalam pembuatan dokumen ini adalah :

1. Nugroho, Adi. *Rational Rose untuk Pemodelan Berorientasi Objek*. Informatika. 2005.

2. Ivar Jacobson, Grady Booch, James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley. 1998.
3. Martin Fowler, Kendall Scott. *UML Distilled – Second Edition*. Addison Wesley. 1999.
4. Stephen R Schach. *An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process*. Mc Graw-Hill. 2004.
5. Wendy Boggs, Michael Boggs. *Mastering UML With Rational Rose*. 2002.

1.5 Deskripsi Umum (Overview)

Dokumen ini terdiri dari tiga bab. Bab pertama adalah **Pendahuluan**, yang berisi deskripsi dokumen. Bab kedua adalah **Deskripsi Perancangan Arsitektural**, yang berisi deskripsi arsitektur sistem. Bab ketiga adalah **Deskripsi Perancangan Antarmuka**, yang berisi deskripsi rancangan GUI yang digunakan sistem untuk berinteraksi dengan *user*.

2 Deskripsi Perancangan Arsitektural

2.1 Design Class

2.1.1 Pengantar

Stereotype yang digunakan dalam design class adalah :

- << boundary >>

Boundary class merupakan class yang berfungsi untuk menghubungkan sistem dengan user di luar sistem.

- << control >>

Control class adalah suatu class yang objek-nya melakukan interaksi antar sekelompok objek lain. Control class biasanya memiliki karakteristik yang spesifik untuk satu use case, dan objek class ini biasanya hanya aktif pada realisasi use case.

2.1.2 Package Dependencies



Gambar 2.1 Package Dependencies

2.1.3 Package WinR2SServer

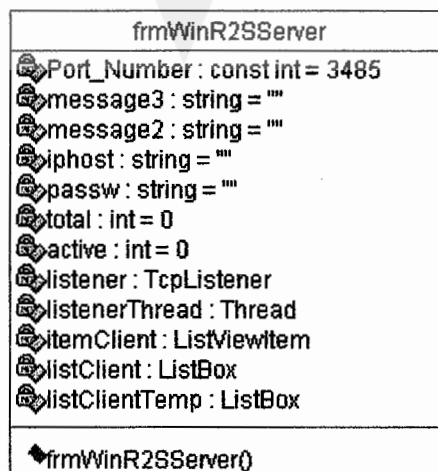
Package ini menyediakan class-class yang digunakan oleh program untuk mengelola hal-hal yang berhubungan dengan perbaikan registry pada sisi/komputer server.

2.1.3.1 Class Diagram Package WinR2SServer



Gambar 2.2 Class Diagram Package WinR2SServer

2.1.3.2 Class frmWinR2SServer



Gambar 2.3 Class frmWinR2SServer

Deskripsi

Class yang berperan sebagai boundary class. Class ini merupakan GUI yang digunakan sebagai server untuk menunggu koneksi dari client dan menerima pesan dari client apabila ada registry client yang rusak serta mengirim pesan agar client memperbaiki registry yang rusak atau berubah.

Atribut

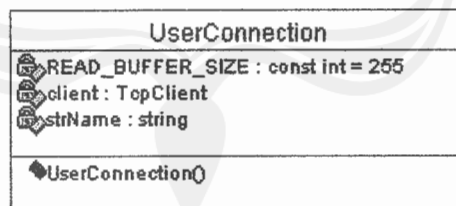
- - Port_Number : const int = 3485
Merepresentasikan variabel Port_Number yang merupakan konstanta bertipe integer yang di set sama dengan 3485.
- - message3 : string = ""
Merepresentasikan variabel message3 bertipe string dan diset sama dengan isi di dalam string.
- - message2 : string = ""
Merepresentasikan variabel message2 bertipe string dan diset sama dengan isi di dalam string.
- - iphost : string = ""
Merepresentasikan variabel iphost bertipe string dan diset sama dengan isi di dalam string.
- - passw : string = ""
Merepresentasikan variabel passw bertipe string dan diset sama dengan isi di dalam string.
- - total : int = 0
Merepresentasikan variabel total bertipe integer yang di set sama dengan 0.
- - active : int = 0
Merepresentasikan variabel active bertipe integer yang di set sama dengan 0.
- - listener : TcpListener
Merepresentasikan variabel listener bertipe TcpListener.
- - listenerThread : Thread
Merepresentasikan variabel listenerThread bertipe Thread.

- - itemClient: ListViewItem
Merepresentasikan variabel itemClient yang bertipe ListViewItem.
- - listClient : ListBox
Merepresentasikan listClient yang bertipe ListBox.
- - listClientTemp: ListBox
Merepresentasikan listClientTemp yang bertipe ListBox.

Method

- + frmWinR2SServer()
Operasi ini secara garis besar digunakan untuk melakukan koneksi ke client, memonitor komputer client dan memonitor status registry, mengirimkan pesan dalam bentuk string dari komputer server ke komputer client, dan menutup aplikasi pada komputer client dari komputer server. Class frmWinR2SServer ini akan mengacu ke class UserConnection.

2.1.3.3 Class UserConnection



Gambar 2.4 Class UserConnection

Deskripsi

Class yang berperan sebagai control class. Class ini digunakan untuk mendapatkan nama client yang terkoneksi dengan server, mengirim data ke client berupa string, serta menerima string yang dikirim client.

Atribut

- - READ_BUFFER_SIZE : const int = 255
Merepresentasikan variabel READ_BUFFER_SIZE yang merupakan konstanta bertipe integer dan diset sama dengan 255.
- - client : TcpClient
Merepresentasikan variabel client yang bertipe TcpClient.
- - strName : string
Merepresentasikan strName dengan tipe string.

Method

- + UserConnection()
Operasi ini digunakan untuk mendapatkan nama client yang terkoneksi dengan server, mengirim data ke client berupa string, serta menerima string yang dikirim client.

2.1.4 Package WinR2SClient

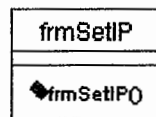
Package ini menyediakan class-class yang digunakan oleh program untuk mengelola hal-hal yang berhubungan dengan perbaikan registry pada sisi/komputer client.

2.1.4.1 Class Diagram Package WinR2SClient



Gambar 2.5 Class Diagram Package WinR2SClient

2.1.4.2 Class frmSetIP



Gambar 2.6 Class frmSetIP

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan GUI yang digunakan untuk menginputkan IP Server. Class ini berada dalam sistem WinR2S Client tetapi pengisiannya dilakukan oleh Administrator yang merupakan pengguna sistem WinR2S Server.

Atribut

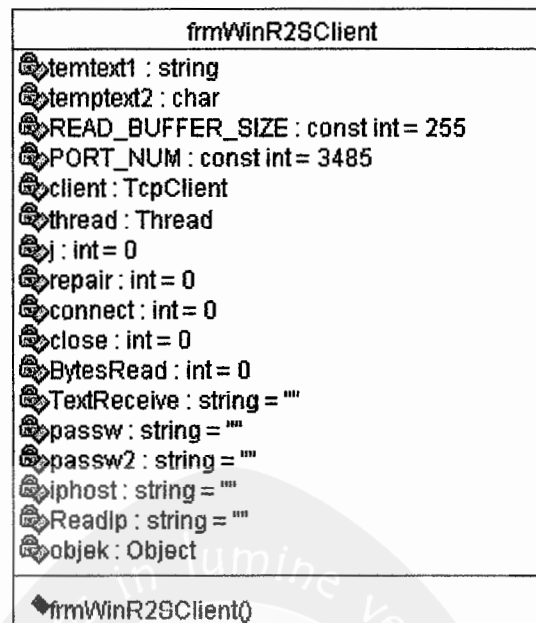
-

Method

- + frmSetIP()

Operasi ini digunakan untuk menginputkan IP Server yang dilakukan oleh Administrator. Operasi ini mengacu ke class frmWinR2SClient setelah dilakukan penginputan IP Server sampai ada koneksi dari client ke server.

2.1.4.3 Class frmWinR2SClient



Gambar 2.7 Class frmWinR2SClient

Deskripsi

Class yang berperan sebagai boundary class, merepresentasikan GUI yang digunakan sebagai client yang digunakan untuk melakukan koneksi ke server, mengecek kerusakan registry, mengirim string data tentang status registry, serta melakukan perbaikan registry.

Atribut

- - temptext1 : string
Merepresentasikan variabel temptext1 yang bertipe string.
- - temptext2 : char
Merepresentasikan variabel temptext2 yang bertipe char.
- - READ_BUFFER_SIZE : const int = 255
Merepresentasikan variabel READ_BUFFER_SIZE yang merupakan konstanta bertipe integer dan diset sama dengan 255.
- - PORT_NUM : const int = 3485

Merepresentasikan variabel `PORT_NUM` yang merupakan konstanta bertipe integer dan diset sama dengan 3485.

- `- client : TcpClient`

Merepresentasikan variabel `client` yang bertipe `TcpClient`.

- `- thread : Thread`

Merepresentasikan variabel `thread` yang bertipe `Thread`.

- `- j : int = 0`

Merepresentasikan variabel `j` yang bertipe integer dan diset sama dengan 0.

- `- repair : int = 0`

Merepresentasikan variabel `repair` yang bertipe integer dan diset sama dengan 0.

- `- connect : int = 0`

Merepresentasikan variabel `connect` yang bertipe integer, dan diset sama dengan 0.

- `- close : int = 0`

Merepresentasikan variabel `close` yang bertipe integer, dan diset sama dengan 0.

- `- BytesRead : int = 0`

Merepresentasikan variabel `BytesRead` yang bertipe integer, dan diset sama dengan 0.

- `- TextReceive : string = ""`

Merepresentasikan variabel `TextReceive` yang bertipe string, dan diset sama dengan isi di dalam string.

- `- passw : string = ""`

Merepresentasikan variabel `passw` yang bertipe string dan diset sama dengan isi di dalam string.

- `- passw2 : string = ""`

Merepresentasikan variabel `passw2` yang bertipe string dan diset sama dengan isi di dalam string.

- `- iphost : string = ""`

Merepresentasikan variabel iphost yang bertipe string dan diset sama dengan isi di dalam string.

- - ReadIp : string = ""

Merepresentasikan variabel ReadIp yang bertipe string dan diset sama dengan isi di dalam string.

- - objek : Object

Merepresentasikan variabel objek yang bertipe Object.

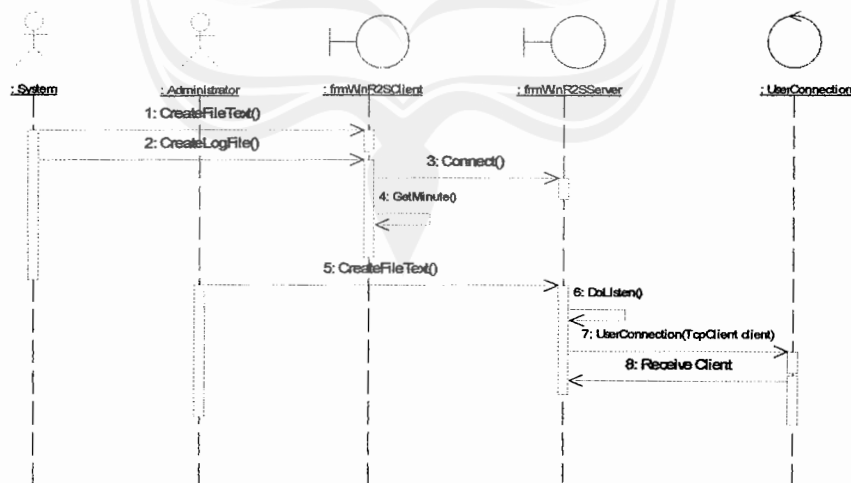
Method

- + frmWinR2SClient()

Operasi ini digunakan untuk untuk melakukan koneksi ke server, mengecek kerusakan registry, mengirim string data tentang status registry, serta melakukan perbaikan registry.

2.2 Realisasi Use Case

2.2.1 Use Case : Connect



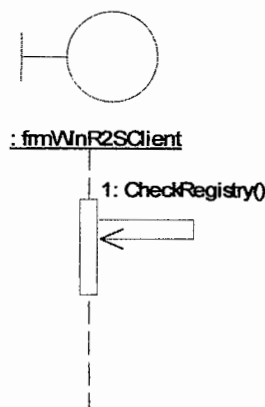
Gambar 2.8 Design Sequence Diagram : Use Case Connect

Flow of events :

1. System melakukan CreateFileText() pada frmWinR2SClient yang merupakan boundary class, untuk membuat file yang isinya berupa IP Server.

2. System kemudian melakukan `CreateLogFile()` pada boundary class `frmWinR2SClient`, untuk membuat file yang berisi registry yang sudah diperbaiki.
3. Sistem melakukan koneksi ke server dengan memanggil prosedur `Connect()` dari boundary class `frmWinR2SClient` ke boundary class `frmWinR2SServer`.
4. Sistem pada boundary class `frmWinR2SClient` akan melakukan `GetMinute()` untuk mengecek registry dan melakukan koneksi ke server apabila client belum terkoneksi dengan server, serta melakukan perbaikan registry apabila ada kerusakan atau perbaikan registry.
5. Administrator melakukan `CreateFileText()` ke boundary class `frmWinR2Sserver`, untuk membuat file yang berisi informasi list view komputer client.
6. Sistem pada boundary class `frmWinR2SServer` akan melakukan `DoListen()` yang digunakan untuk menunggu koneksi dari client.
7. Sistem pada boundary class `frmWinR2SServer` akan melakukan `UserConnection(TcpClient client)` ke control class `UserConnection`.
8. Sistem pada control class `UserConnection` akan melakukan `Receive Client` ke boundary class `frmWinR2SServer`.

2.2.2 Use Case : Check Registry

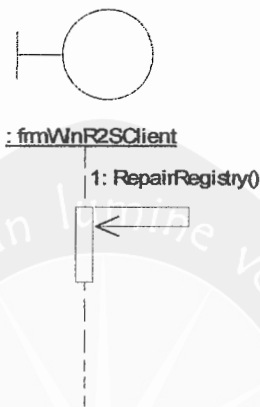


Gambar 2.9 Design Sequence Diagram : Use Case Check Registry

Flow of events :

1. Sistem akan melakukan `CheckRegistry()` pada boundary class `frmWinR2SClient` untuk melakukan pengecekan apabila terjadi perubahan atau kerusakan registry.

2.2.3 Use Case : Repair Registry

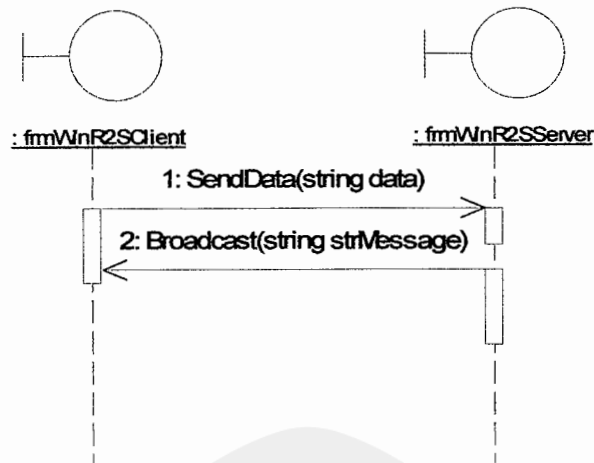


Gambar 2.10 Design Sequence Diagram : Use Case Repair Registry

Flow of events :

1. Sistem pada boundary class `frmWinR2SClient` memanggil prosedur `RepairRegistry()` untuk memperbaiki registry yang mengalami kerusakan atau perubahan.

2.2.4 Use Case : Send Message

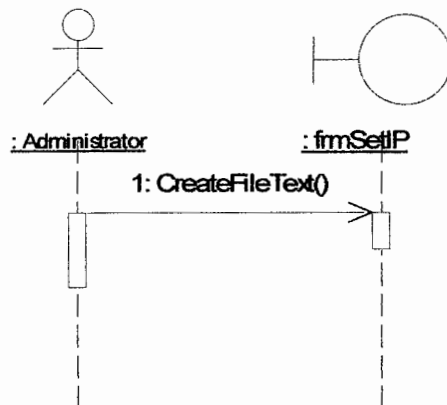


Gambar 2.11 Design Sequence Diagram : Use Case Send Message

Flow of events :

1. Sistem pada boundary class `frmWinR2SClient` memanggil prosedur `SendData(string data)` untuk mengirimkan status registry pada komputer client ke komputer server yaitu boundary class `frmWinR2SServer`.
2. Sistem pada boundary `frmWinR2SServer` kemudian akan memanggil prosedur `Broadcast(string strMessage)` untuk mengirim atau membalas pesan ke komputer client yang terkoneksi dengan komputer server pada boundary class `frmWinR2SClient`.

2.2.5 Use Case : Input IP Server

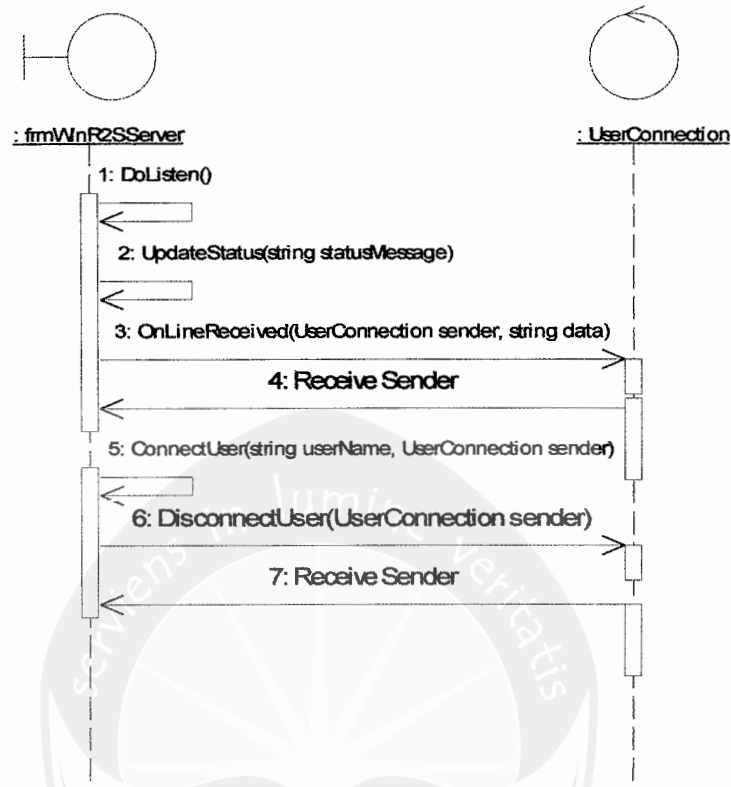


Gambar 2.12 Design Sequence Diagram : Use Case Input IP Server

Flow of events :

1. Administrator memanggil prosedur `CreateFileText()` untuk membuat file yang berisi IP Server ke boundary class `frmSetIP`.

2.2.6 Use Case : Monitor Client & Registry Status



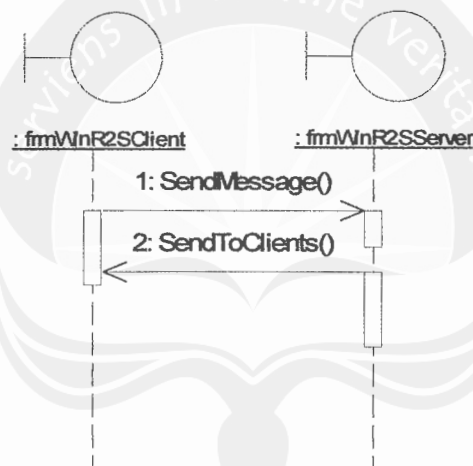
Gambar 2.13 Design Sequence Diagram : Use Case Monitor Client & Registry Status

Flow of events :

1. Sistem pada boundary class `frmWinR2SServer` memanggil prosedur `DoListen()` untuk menunggu koneksi dari client.
2. Sistem pada boundary `frmWinR2SServer` kemudian akan memanggil prosedur `UpdateStatus(string statusMessage)` untuk mengupdate informasi dari client.
3. Sistem pada boundary class `frmWinR2SServer` memanggil prosedur `OnLineReceived(UserConnection sender, string data)` ke control class `UserConnection` untuk membagi pesan dalam bentuk array of string.
4. Sistem pada boundary class `frmWinR2SServer` kemudian akan menerima pesan `Receive Sender`.

5. Sistem pada boundary class `frmWinR2SServer` memanggil prosedur `ConnectUser(string userName, UserConnection Sender)` yang digunakan untuk menginformasikan client yang terkoneksi ke server.
6. Sistem pada boundary class `frmWinR2SServer` memanggil prosedur `DisconnectUser(UserConnection sender)` ke control class `UserConnection` untuk menginformasikan client yang disconnect.
7. Sistem pada boundary class `frmWinR2Sserver` kemudian akan menerima pesan `Receive Sender` dari control class `UserConnection`.

2.2.7 Use Case : Send Message

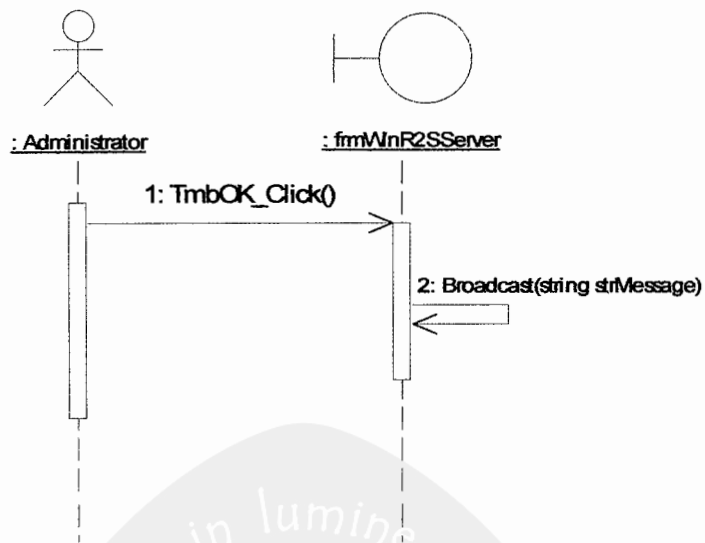


Gambar 2.14 Design Sequence Diagram : Use Case Send Message

Flow of events :

1. Sistem pada boundary class `frmWinR2SClient` memanggil prosedur `SendMessage()` untuk mengirimkan message tentang status registry pada komputer client ke komputer server yaitu boundary class `frmWinR2SServer`.
2. Sistem pada boundary class `frmWinR2SServer` kemudian akan memanggil prosedur `SendToClients()` untuk mengirim pesan ke komputer client yang terkoneksi dengan komputer server pada boundary class `frmWinR2SClient`.

2.2.8 Use Case : Set Close Client Application



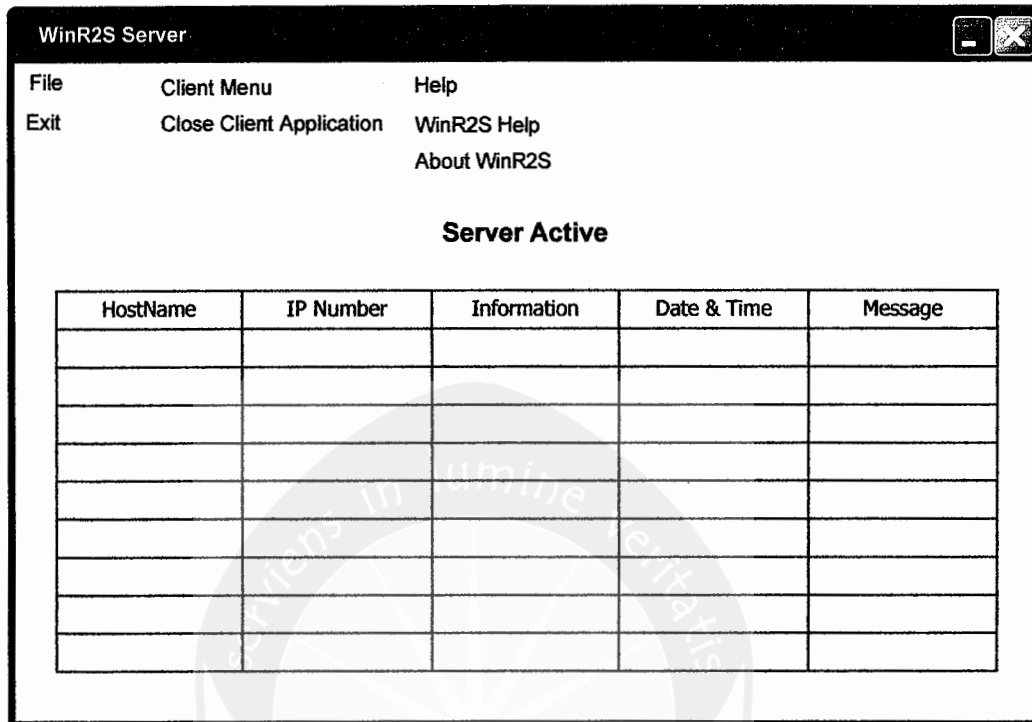
Gambar 2.15 Design Sequence Diagram : Use Case Set Close Client Application

Flow of events :

1. Administrator memanggil prosedur `TmbOK_Click()` untuk menutup aplikasi pada komputer client lewat komputer server.
2. Sistem pada boundary `frmWinR2SServer` kemudian akan memanggil prosedur `Broadcast(string strMessage)` untuk mengirim atau membalas pesan ke komputer client yang terkoneksi dengan komputer server.

3 Deskripsi Perancangan Antarmuka

3.1 Antarmuka WinR2S Server



Gambar 3.1 Rancangan Antarmuka WinR2S Server

Deskripsi

- Antarmuka ini merupakan antarmuka untuk komputer server yang digunakan untuk menerima koneksi dari client (UC-WinR2S-01), untuk memonitor registry client (UC-WinR2S-06), mengirim string data ke client (UC-WinR2S-07), dan menutup aplikasi di komputer client (UC-WinR2S-08).
- Menu yang tersedia adalah menu File, Client Menu, dan Help.
- Dalam menu File terdapat sub menu Exit.
- Dalam menu Client Menu terdapat sub menu Close Client Application (UC-WinR2S-08).
- Dalam menu Help terdapat sub menu WinR2S Help dan About WinR2S.

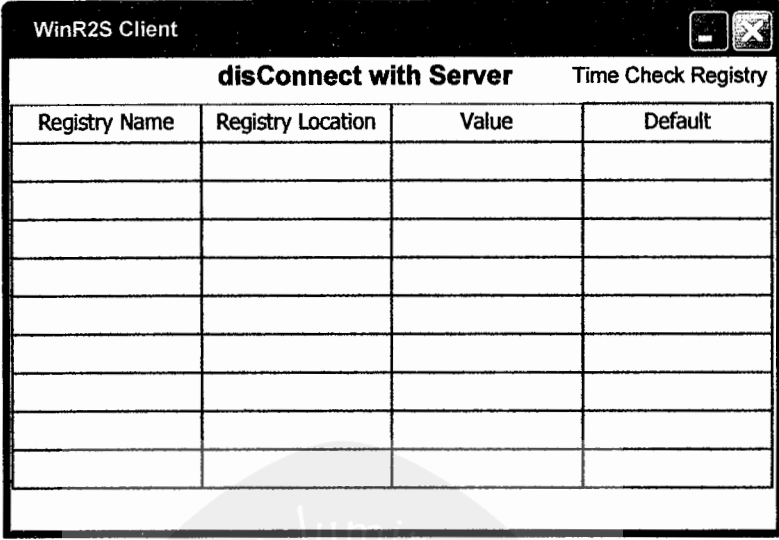
- Dalam antarmuka WinR2S Server ini terdapat listview yang berisi HostName, IP Number, Information, Date & Time, dan Message.

Event

Urutan aksi yang terjadi :

1. User masuk ke sistem WinR2S Server.
2. Sistem akan menampilkan antarmuka WinR2S Server.
3. Sistem menunggu koneksi dari client dengan melakukan **DoListen()**.
4. Sistem akan menginformasikan client yang terkoneksi dan dimunculkan dalam list view dengan memanggil prosedur **ConnectUser(string userName, UserConnection sender)**.
5. Sistem dapat mengupdate informasi seputar client dengan memanggil prosedur **UpdateStatus(string statusMessage)**.
6. Sistem pada server dapat mengirim atau membalas pesan ke client dengan melakukan **Broadcast(string strMessage)**.
7. Sistem akan memberikan informasi komputer client yang disconnect dengan memanggil prosedur **DisconnectUser(UserConnection sender)**.
8. User dapat memilih menu maupun sub menu yang diinginkan.

3.2 Antarmuka WinR2S Client



disConnect with Server		Time Check Registry	
Registry Name	Registry Location	Value	Default

Gambar 3.2 Rancangan Antarmuka WinR2S Client

Deskripsi

- Antarmuka ini digunakan pada use case Check Registry (UC-WinR2S-02) yang digunakan untuk mengecek ada atau tidaknya kerusakan registry, use case Repair Registry (UC-WinR2S-03) yang digunakan untuk memperbaiki kerusakan registry, dan use case Send Message (UC-WinR2S-04) yang digunakan untuk mengirim pesan ke komputer server.
- Antarmuka WinR2S Client ini tidak ada campur tangan user.
- Apabila client belum terkoneksi ke server maka setiap 5 menit akan terkoneksi, begitu juga apabila ada kerusakan registry maka sistem akan secara otomatis memperbaikinya setelah ada persetujuan dari server.

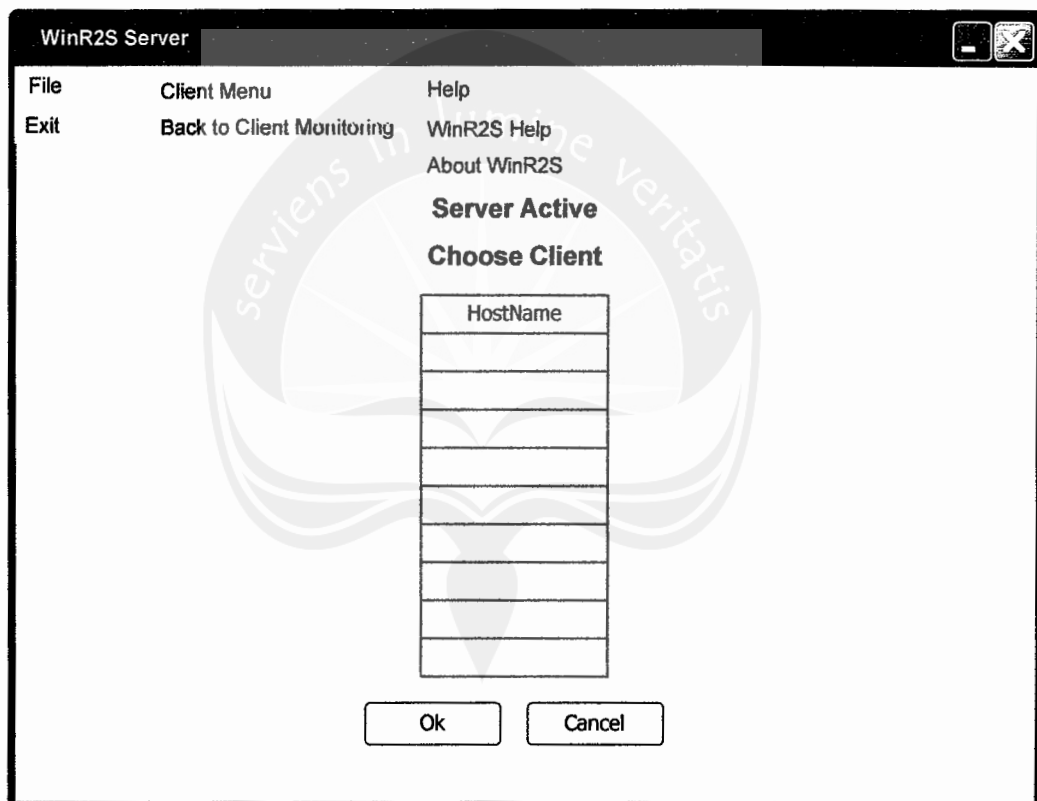
Event

Urutan aksi yang terjadi :

1. Sistem pada WinR2S Client sudah terkoneksi dengan WinR2S Server dengan memanggil prosedur **Connect()**.
2. Sistem akan melakukan **CreateFileText()** yang digunakan untuk membuat file yang isinya berupa IP Server.

3. Untuk membuat file yang isinya berupa registry yang sudah diperbaiki, sistem memanggil prosedur **CreateFileLog()**.
4. Sistem akan memanggil prosedur **GetMinute()** untuk mengecek registry dan melakukan koneksi ke server apabila client belum terkoneksi ke server, serta melakukan perbaikan kerusakan registry apabila terdapat kerusakan atau perubahan registry.

3.3 Antarmuka Close Client Application



Gambar 3.3 Rancangan Antarmuka Close Client Application

Deskripsi

- Antarmuka ini digunakan pada use case Set Close Client Application (UC-WinR2S-08).
- Antarmuka ini digunakan untuk memutus koneksi client yang dilakukan oleh user.

Event

Urutan aksi yang terjadi :

1. User memilih menu Client Menu dan sub menu **Close Client Application** pada form WinR2S Server.
2. User memilih client yang akan ditutup aplikasinya dalam list view **Choose Client**.
3. User memilih button **Ok** apabila ingin melanjutkan pemutusan aplikasi.
4. User memilih button **Cancel** apabila user ingin membatalkan pemutusan client.

3.4 Antarmuka Enter IP Server



Enter IP Server

Gambar 3.4 Rancangan Antarmuka Enter IP Server

Deskripsi

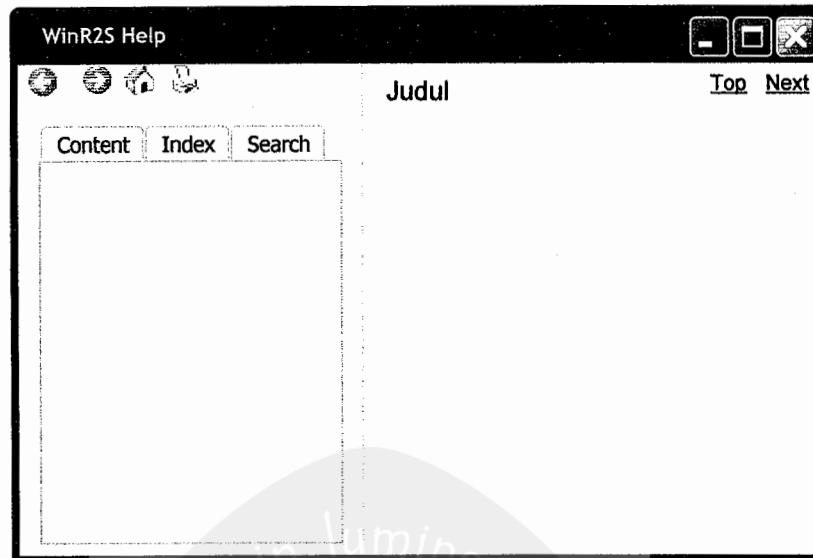
- Antarmuka ini digunakan pada use case Input IP Server (UC-WinR2S-05).
- Untuk terkoneksi ke server, client harus memasukkan IP Server. Penginputan ini dilakukan oleh Administrator.

Event

Urutan aksi yang terjadi :

1. User memasukkan IP Server untuk melakukan koneksi dengan server.
2. Setelah IP Server diinputkan maka user menekan **Enter**.
3. Sistem **WinR2S Client** akan terkoneksi dengan sistem **WinR2S Server**.

3.5 Antarmuka WinR2S Help



Gambar 3.5 Rancangan Antarmuka WinR2S Help

Deskripsi

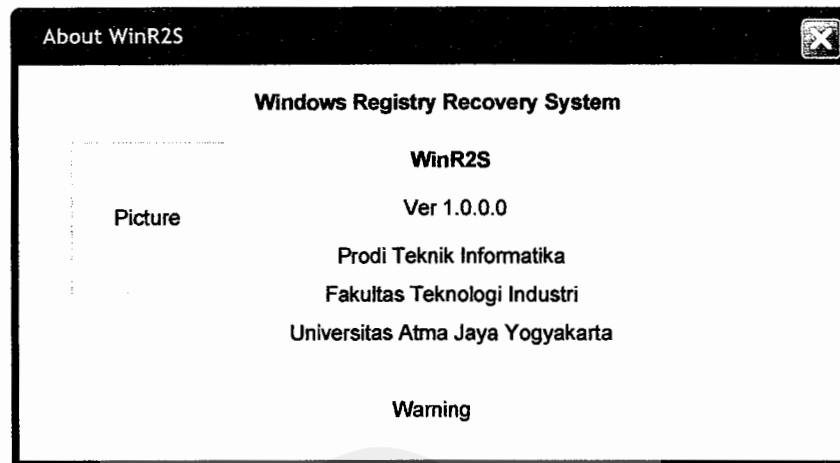
- Antarmuka ini adalah form WinR2S Help yang digunakan untuk memberikan informasi cara kerja penggunaan perangkat lunak WinR2S.

Event

Urutan aksi yang terjadi :

1. User memilih menu **Help** dan sub menu **WinR2S Help** pada form **WinR2S Server**.
2. Sistem akan membuka form **WinR2S Help** yang akan digunakan user untuk mencari informasi penggunaan **WinR2S**.
3. User dapat memilih tab **content**, **index**, atau **search** untuk memudahkan pencarian informasi.

3.6 Antarmuka About WinR2S



Gambar 3.6 Rancangan Antarmuka About WinR2S

Deskripsi

- Antarmuka About WinR2S digunakan untuk mengetahui seputar perangkat lunak WinR2S.

Event

Urutan aksi yang terjadi :

1. User memilih menu **Help** dan sub menu **About WinR2S** pada form **WinR2S Server**.
2. Sistem akan membuka form **About WinR2S**.
3. User menekan windows button **Close** untuk keluar dari form **About WinR2S**.

Apendiks A : Daftar Istilah dan Singkatan

Class Diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang dikembangkan.

C# adalah bahasa pemrograman yang sederhana, moderen, berorientasi objek dan type-safe, yang diturunkan dari C dan C++. C# menggabungkan produktifitas dari Visual Basic dan kemampuan dari C++.

Flow of events adalah aliran event-event yang terjadi dalam suatu proses, yang digunakan untuk mendeskripsikan urutan jalannya proses.

Microsoft Visual Studio .NET adalah sebuah *platform* untuk membangun, menjalankan dan meningkatkan generasi lanjut dari aplikasi terdistribusi. Referensi lengkap mengenai Microsoft Visual Studio .NET dapat dilihat pada <http://msdn.microsoft.com/vstudio/>

Registry adalah suatu set *file* data yang dipakai untuk membantu *Windows* mengontrol perangkat keras, perangkat lunak, lingkungan pemakai, dan *interface*.

Sequence Diagram adalah diagram interaksi antar kelas yang menekankan pada waktu urutan waktu pesan dan berfokus pada kelas dan pesan yang dipertukarkan untuk mendapatkan perilaku yang diinginkan.

WinR2S adalah singkatan dari *Windows Registry Recovery System*, yaitu perangkat lunak yang digunakan untuk memperbaiki atau mengembalikan informasi *registry windows* yang rusak atau berubah ke keadaan standar yang diterapkan pada jaringan komputer.

WinR2S Client adalah perangkat lunak WinR2S pada sisi client, dijalankan pada komputer client.

WinR2S Server adalah perangkat lunak WinR2S pada sisi server, yang dikelola oleh Administrator.