

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Setelah aplikasi Virtual Car Catalog ini selesai diimplementasikan maka dapat ditarik beberapa kesimpulan, yaitu:

1. Perangkat Lunak Virtual Car Catalog telah berhasil dibangun sebagai aplikasi katalog penjualan mobil dengan *augmented reality*.
2. Dari hasil pengujian dapat diketahui bahwa fungsi-fungsi yang disediakan oleh perangkat lunak Virtual Car Catalog berjalan dengan benar dan sesuai dengan yang diharapkan. Hal ini dapat dilihat dari hasil pengujian fungsionalitas 34% mengatakan sangat baik, 56% mengatakan baik, dan 10% mengatakan cukup baik. Sedangkan dari hasil pengujian antarmuka 27,5% mengatakan sangat mudah, 55% mengatakan mudah, 16,67% mengatakan cukup mudah, dan 0,83% mengatakan tidak mudah.

6.2 Saran

Penulis ingin memberikan beberapa saran untuk pengembangan lebih lanjut perangkat lunak Virtual Car Catalog ini:

1. Membangun dengan bahasa pemrograman yang berbeda.
2. Menggunakan *library* 3D yang berbeda.

DAFTAR PUSTAKA

- Abubakar, M.M. 2010, *Teknologi Augmented Reality untuk Mensimulasikan Sistem Tata Surya*, Sekolah Tinggi Manajemen Informatika dan Komputer, Yogyakarta.
- Azuma, Ronald. 1997, *A Survey of Augmented Reality*, Presence: Teleoperators and Virtual Environments 6 pp 355-385.
- Bahtiar, M.A. 2011, *Sistem Augmented Reality untuk Animasi Games Menggunakan Camera Pada PC*, Institut Teknologi Sepuluh Nopember, Surabaya.
- Chafied, M. 2010, *Brosur Interaktif Berbasis Augmented Reality*, Institut Teknologi Sepuluh Nopember, Surabaya.
- Dignan, P., Labelle, E., Lyon, M. 2010, *HIGGINS ARMORY VIRTUAL SWORD EXHIBIT*, Worcester Polytechnic Institute, Massachusetts.
- Gorbala, G.T., Hariadi, M. 2010, *Aplikasi Augmented Reality untuk Katalog Penjualan Rumah*, Institut Teknologi Sepuluh Nopember, Surabaya.
- Grossman, G., Huang, E. 2006, *ActionScript 3.0 overview*, (online), (http://www.adobe.com/devnet/actionscript/articles/actionscript3_overview.html, diakses 5 Mei 2011)

- Hayes, G. 2009, *16 Top Augmented Reality Business Models*, (Online), (<http://www.personalizemedia.com/16-top-augmented-reality-business-models/>), diakses 5 Mei 2011)
- Hurwitz, A., Jeffs, A. 2009, *Baseball Proof of Concept - Mobile Augmentation for Entertainment and Shopping Venues*, EYEPLY.
- Kleef, N., Noltes, J., Spoel, S. 2010, *Success factors for Augmented Reality Business Models*, University of Twente, Belanda.
- Koyama, T. 2009, *Introduction to FLARToolKit*, (online), (<http://saqoo.sh/a/labs/FLARToolKit/Introduction-to-FLARToolKit.pdf/>), diakses 5 Mei 2011)
- Malek, S., Henda, N.Z., Belhocine, M., Benbelkacem, S. 2008, *Calibration Method for an Augmented Reality System*, World Academy of Science, Engineering and Technology.
- Medicherla, P.S., Chang, G., Morreale, P. 2010, *Visualization for Increased Understanding and Learning Using Augmented Reality*, ACM SIGMM International Conference on Multimedia Information Retrieval, Philadelphia, March 29-31.
- Parma, S., A. 2007, *Hubungan Antara Konsep Diri Dengan Perilaku Konsumtif Remaja Putri Dalam Pembelian Kosmetik Melalui Katalog Di SMA Negeri 1 Semarang*, Universitas Diponegoro, Semarang.

Shupe, R., Rosser, Z. 2010, *Learning ActionScript 3.0*, California: O'Reilly Media, Inc.

Silva, R., Oliviera, J.C., Giralaldi, G.A. 2003, *Introduction to Augmented Reality*, National Laboratory for Scientific Computation, Brazilian Ministry of Science and Technology, Brazil.

Tuceryan, M., Greer, D.S., Whitaker, R.T., Breen, D.E., Crampton, C., Rose, E., Ahlers, K.H. 1995. *Calibration Requirements and Procedures for a Monitor-Based Augmented Reality System*, Transactions on Visualization and Computer Graphics, vol 1 no. 3, pp 255-273.

Winder, J., Tondeur, P. 2009, *Papervision3d Essentials*, Birmingham: Packt Publishing.

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

VCC

(Virtual Car Catalog)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Michael Alvado / 5308

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL-VCC		1/18
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
			

Daftar Isi

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim, dan Singkatan	6
1.4	Referensi	7
1.5	Deskripsi Umum	7
2	Deskripsi Kebutuhan	8
2.1	Perspektif Produk	8
2.2	Fungsi Produk	10
2.3	Karakteristik Pengguna	11
2.4	Batasan-batasan	11
2.5	Asumsi Ketergantungan	11
3	Kebutuhan Khusus	11
3.1	Kebutuhan antarmuka eksternal	11
3.2	Kebutuhan fungsionalitas Perangkat Lunak	13
4	Spesifikasi Rinci Kebutuhan	14
4.1	Spesifikasi Kebutuhan Fungsionalitas	14

Daftar Gambar

1	Arsitektur Perangkat Lunak VCC	9
2	Use Case Diagram	13



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak VCC (*Virtual Car Catalog*) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-VCC ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak VCC dikembangkan dengan tujuan untuk:

1. Melacak *marker* tertentu dari gambar yang dicetak.
2. Menambahkan model 3D ke dunia nyata dengan menggunakan *augmented reality*.
3. Memberi interaksi pada model 3D yang tampil.

Dan berjalan pada lingkungan dengan platform Desktop.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-VCC-XXX	Kode yang merepresentasikan kebutuhan pada

Program Studi Teknik Informatika	SKPL – VCC	6/ 18
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

	VCC (Virtual Car Catalog) dimana XXX merupakan nomor fungsi produk.
VCC	Perangkat lunak <i>augmented reality</i> untuk katalog penjualan mobil.
Marker	Gambar yang memiliki pola tertentu yang dicetak dengan <i>printer</i> .
Augmented Reality	Teknologi yang menggabungkan benda maya ke dalam lingkungan nyata secara <i>real time</i> .

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Koyama, Tomohiko (2009). *Introduction to FLARToolkit*, (online), (<http://saqoo.sh/a/labs/FLARToolkit/Introduction-to-FLARToolkit.pdf>, diakses 13 Maret 2011)
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. Mook, Colin. 2007, *Essential ActionScript 3.0*, California: O'Reilly Media, Inc.
4. Lively, Michael. 2009, *Professional Papervision 3D*, San Francisco: John Wiley and Sons.

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 4 bagian utama. Bagian pertama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak VCC yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak VCC tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan khusus perangkat lunak VCC yang akan dikembangkan.

Bagian keempat berisi spesifikasi rinci kebutuhan. Deskripsi spesifikasi rinci kebutuhan berdasarkan pada use case yang ada dalam pengembangan perangkat lunak VCC.

2 Deskripsi Kebutuhan

2.1 Perspektif produk

VCC merupakan perangkat lunak yang dikembangkan untuk membantu mengaplikasikan *augmented reality* untuk katalog penjualan mobil. Sistem ini menambahkan objek 3D ke *marker* pada katalog yang telah dicetak dan menggunakan aplikasi berbasis desktop.

Perangkat lunak VCC ini berjalan pada platform Windows 7 untuk PC, dan dibuat menggunakan bahasa pemrograman Actionscript 3. Sedangkan untuk lingkungan pemrogramannya menggunakan Adobe Flash CS5.

Pengguna akan berinteraksi dengan sistem melalui aplikasi yang berbasis desktop. Pada sistem ini, seperti yang terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan adalah user -

application yang terhubung dengan perangkat keras *webcam*. User melakukan pengambilan data berupa gambar melalui perangkat keras *webcam* dan gambar tersebut akan dicek apakah ada *marker* atau tidak, jika ada maka gambar akan ditambah model 3D diatas *marker*. Kemudian model 3D yang muncul dapat diputar ke kiri atau kanan dan dapat diperbesar atau diperkecil agar lebih jelas.



Gambar 1: Arsitektur Perangkat lunak VCC

2.2 Fungsi Produk

Fungsi produk perangkat lunak VCC adalah sebagai berikut :

1. Fungsi Menangkap Gambar (SKPL-VCC-001)

Merupakan fungsi yang digunakan untuk mengecek apakah ada webcam yang terpasang, jika ada webcam terpasang maka akan menampilkan gambar yang ditangkap oleh webcam.

2. Fungsi Melacak Marker (SKPL-VCC-002)

Merupakan fungsi yang digunakan untuk melacak apakah ada marker yang terdeteksi pada gambar yang ditangkap oleh webcam.

3. Fungsi Menampilkan Model (SKPL-VCC-003)

Merupakan fungsi yang digunakan untuk menampilkan model 3D dan diletakkan diatas marker.

4. Fungsi Mengendalikan model (SKPL-VCC-004)

Merupakan fungsi yang digunakan untuk melakukan interaksi terhadap model 3D dengan menggunakan masukan dari keyboard.

Fungsi Mengendalikan model meliputi:

a. Fungsi Rotate (SKPL-VCC-004-001)

Merupakan fungsi yang digunakan untuk memutar model 3D yang tampil ke kiri atau ke kanan.

b. Fungsi Zoom (SKPL-VCC-004-002)

Merupakan fungsi yang digunakan untuk memperbesar atau mengecilkan model 3D yang tampil.

c. Fungsi Default Size (SKPL-VCC-004-003)

Merupakan fungsi yang digunakan untuk mengembalikan ukuran model 3D ke ukuran yang pertama kali tampil.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak VCC adalah sebagai berikut :

1. Memahami pengoperasian PC.
2. Memahami penggunaan *webcam*.
3. Memahami penggunaan *adobe flash player*.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak VCC tersebut adalah :

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak VCC.

2. Keterbatasan perangkat keras

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

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat PC yang menggunakan sistem operasi Windows 7, *adobe flash player projector* dengan minimal versi 10 dan *webcam* yang telah menjadi spesifikasi sistem.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak VCC meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak VCC adalah:

1. Perangkat PC.
2. Webcam.

3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak VCC adalah sebagai berikut :

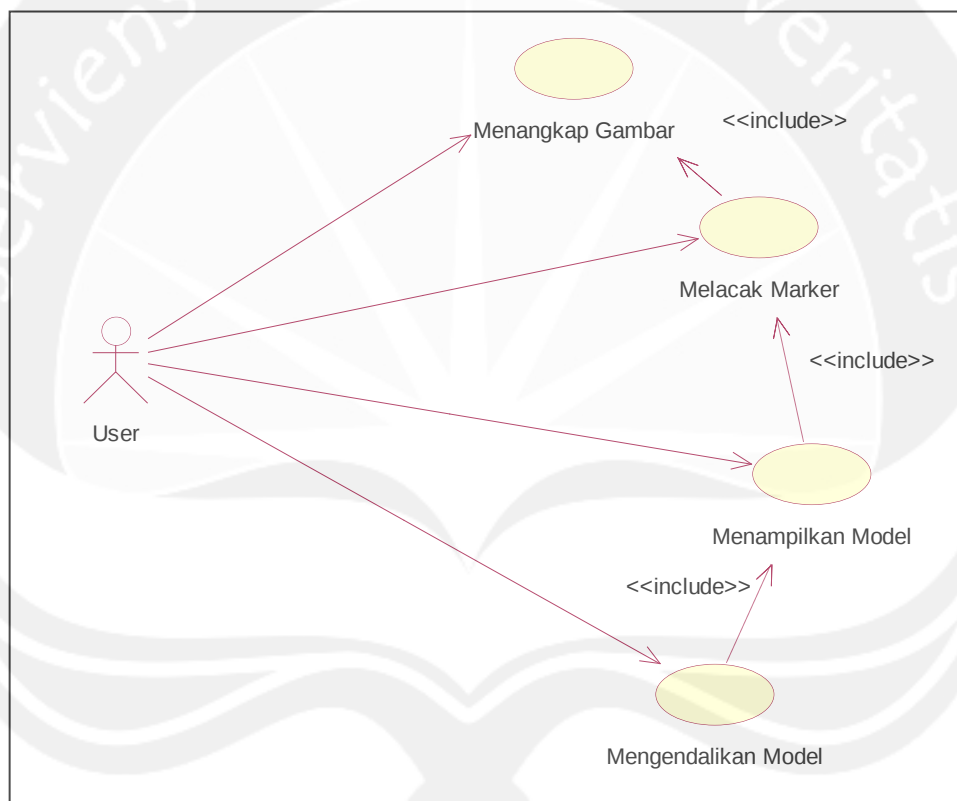
1. Nama : Windows 7
Sumber : Microsoft.
Sebagai sistem operasi.
2. Nama : Flash Player Projector 10
Sumber : Adobe.

Sebagai player untuk menampilkan gambar yang ditangkap oleh webcam dan juga sebagai form.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram

Gambar 2 merupakan Use Case yang dipakai untuk membangun aplikasi Virtual Car Catalog. Ada empat buah Use Case, yaitu Menangkap Gambar, Melacak Marker, Menampilkan Model, dan Mengendalikan Model.



Gambar 2: Use Case Diagram

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification: Menangkap Gambar

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pengambilan gambar dari *webcam* dan ditampilkan ke layar monitor.

2. Primary Actor

1. User

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor masuk ke sistem.

2. Sistem melakukan pengecekan *webcam* sudah terpasang atau belum.

E-1 *Webcam* tidak terpasang

3. Sistem menampilkan gambar yang ditangkap dari *webcam* secara *live*.

4. Use Case ini selesai.

5. Alternative Flow

none

6. Error Flow

E-1 *Webcam* tidak terpasang

1. Sistem menampilkan pesan, *webcam* tidak terpasang

7. PreConditions

1. Aktor telah memasuki sistem

8. PostConditions

Menampilkan gambar yang ditangkap dari *webcam* secara *live*

4.1.2 Use case Spesification: Melacak Marker

1. Brief Description

Use Case ini digunakan oleh aktor untuk melacak *marker* dari gambar yang ditangkap oleh *webcam* secara *live*.

2. Primary Actor

User

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika *webcam* menangkap gambar secara *live*.
2. Sistem melakukan pelacakan terhadap gambar yang ditangkap dari *webcam*.
3. Use Case ini selesai.

5. Alternative Flow

none

6. Error Flow

none

7. PreConditions

1. Use Case Menangkap Gambar telah dilakukan
2. Aktor telah memasuki sistem

8. PostConditions

Menampilkan gambar yang telah ditangkap *Webcam* secara *live*.

4.1.3 Use case Spesification: Menampilkan Model

1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan model 3D ke *marker* yang telah dilacak.

2. Primary Actor

User

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika gambar yang ditangkap oleh *webcam* telah terlacak *marker*.
2. Sistem menampilkan model 3D pada *marker* yang telah terlacak.
3. Use Case ini selesai.

5. Alternative Flow

none

6. Error Flow

none

7. PreConditions

1. Aktor telah memasuki system
2. Use Case Melacak Marker telah dilakukan

8. PostConditions

Model 3D telah ditampilkan diatas *marker*.

4.1.4 Use case Spesification: Mengendalikan Model

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengendalikan model 3D yang telah ditampilkan dengan masukkan dari *keyboard*.

2. Primary Actor

User

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika model 3D telah ditampilkan diatas *marker*.
2. Sistem menampilkan model 3D pada *marker* yang telah terlacak.
3. Aktor menekan tombol untuk putar kanan
A-1 Aktor menekan tombol untuk putar kiri
A-2 Aktor menekan tombol untuk perbesar
A-3 Aktor menekan tombol untuk perkecil
A-4 Aktor menekan tombol untuk kembali ke ukuran semula
4. Sistem mengecek tombol yang ditekan
5. Sistem memutar model 3D yang muncul ke arah kanan.
6. Use Case ini selesai.

5. Alternative Flow

- A-1 Aktor menekan tombol untuk putar kiri
- 1.Sistem mengecek tombol yang ditekan
 - 2.Sistem memutar model 3D yang muncul ke arah kiri
 - 3.Berlanjut ke Basic Flow langkah ke 3
- A-2 Aktor menekan tombol untuk perbesar
- 1.Sistem mengecek tombol yang ditekan
 - 2.Sistem memperbesar ukuran model 3D yang muncul
 - 3.Berlanjut ke Basic Flow langkah ke 3
- A-3 Aktor menekan tombol untuk perkecil
- 1.Sistem mengecek tombol yang ditekan
 - 2.Sistem memperkecil ukuran model 3D yang muncul
 - 3.Berlanjut ke Basic Flow langkah ke 3

A-4 Aktor menekan tombol untuk kembali ke ukuran semula

1. Sistem mengecek tombol yang ditekan
2. Sistem mengembalikan ukuran model 3D yang muncul ke ukuran semula
3. Berlanjut ke Basic Flow langkah ke 3

6. Error Flow

none

7. PreConditions

1. Aktor telah memasuki sistem
2. Use Case Menampilkan Model telah dilakukan

8. PostConditions

Model 3D telah diputar, perbesar atau perkecil ukurannya, atau kembali ke ukuran semula.

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

VCC

(Virtual Car Catalog)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Michael Alvado / 5308

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL-VCC		1/23
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1. Pendahuluan	6
1.1. Tujuan	6
1.2. Ruang Lingkup	6
1.3. Definisi dan Akronim.....	6
1.4. Referensi	7
2. Perancangan Sistem	8
2.1. Perancangan Arsitektur.....	8
2.2. Perancangan Rinci	9
2.2.1. Sequence Diagram.....	9
2.2.1.1 Menangkap Gambar.....	9
2.2.1.2 Melacak Marker.....	10
2.2.1.3 Menampilkan Model.....	11
2.2.1.4 Mengendalikan Model.....	12
2.2.2. Class Diagram	13
2.2.3. Spesifikasi Deskripsi Kelas Diagram	14
2.2.3.1 Spesifikasi Deskripsi Kelas mainUI	14
2.2.3.2 Spesifikasi Deskripsi Kelas formUI	14
2.2.3.3 Spesifikasi Deskripsi Kelas aboutUI	15
2.2.3.4 Spesifikasi Deskripsi Kelas tutorialUI	15
2.2.3.5 Spesifikasi Deskripsi Kelas webcamMgr	15
2.2.3.6 Spesifikasi Deskripsi Kelas markerMgr	16
2.2.3.7 Spesifikasi Deskripsi Kelas modelMgr	19
3. Perancangan Antarmuka	20
3.1 mainUI	20
3.2 formUI	21
3.3 tutorialUI	22
3.4 aboutUI	23

Daftar Gambar

Gambar 2.1. Perancangan Sistem: Perancangan Arsitektur	8
Gambar 2.2. Sequence Diagram: Menangkap Gambar	9
Gambar 2.3. Sequence Diagram: Melacak Marker	10
Gambar 2.4. Sequence Diagram: Menampilkan Model	11
Gambar 2.5. Sequence Diagram: Mengendalikan Model	12
Gambar 2.6. Class Diagram	13
Gambar 3.1 Rancangan Antarmuka mainUI	20
Gambar 3.2 Rancangan Antarmuka formUI	21
Gambar 3.3 Rancangan Antarmuka tutorialUI	22
Gambar 3.4 Rancangan Antarmuka aboutUI	23

1. Pendahuluan

1.1. Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

1.2. Ruang Lingkup

Perangkat Lunak VCC dikembangkan dengan tujuan untuk:

1. Melacak *marker* tertentu dari gambar yang dicetak.
2. Menambahkan objek virtual ke dunia nyata dengan menggunakan *augmented reality*.
3. Memberi interaksi pada model 3D yang tampil.

Dan berjalan pada lingkungan dengan platform Desktop.

1.3. Definisi dan Akronim

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.
VCC	Perangkat lunak <i>augmented reality</i> untuk katalog penjualan mobil.

1.4. Referensi

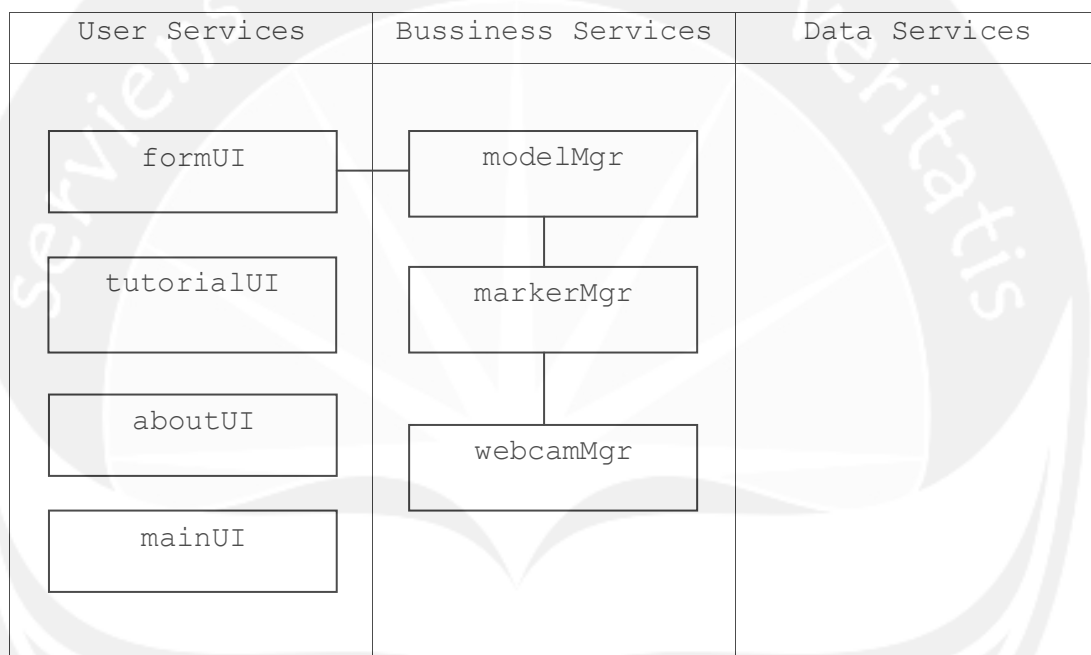
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Koyama, Tomohiko (2009). *Introduction to FLARToolkit*, (online), (<http://saqoo.sh/a/labs/FLARToolkit/Introduction-to-FLARToolkit.pdf>, diakses 13 Maret 2011)
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. Mook, Colin. 2007, *Essential ActionScript 3.0*, California: O'Reilly Media, Inc.
4. Lively, Michael. 2009, *Professional Papervision 3D*, San Francisco: John Wiley and Sons.
5. Alvado, M., *Spesifikasi Kebutuhan Perangkat Lunak VCC*, Universitas Atma Jaya Yogyakarta, 2011.

2. Perancangan Sistem

2.1. Perancangan Arsitektur

Perancangan arsitektur aplikasi Virtual Car Catalog dibagi menjadi User Services, Bussiness Services, dan Data Services. Gambar 2.1 menunjukkan kelas-kelas yang dipakai pada User Services, Bussiness Services, dan Data Services.

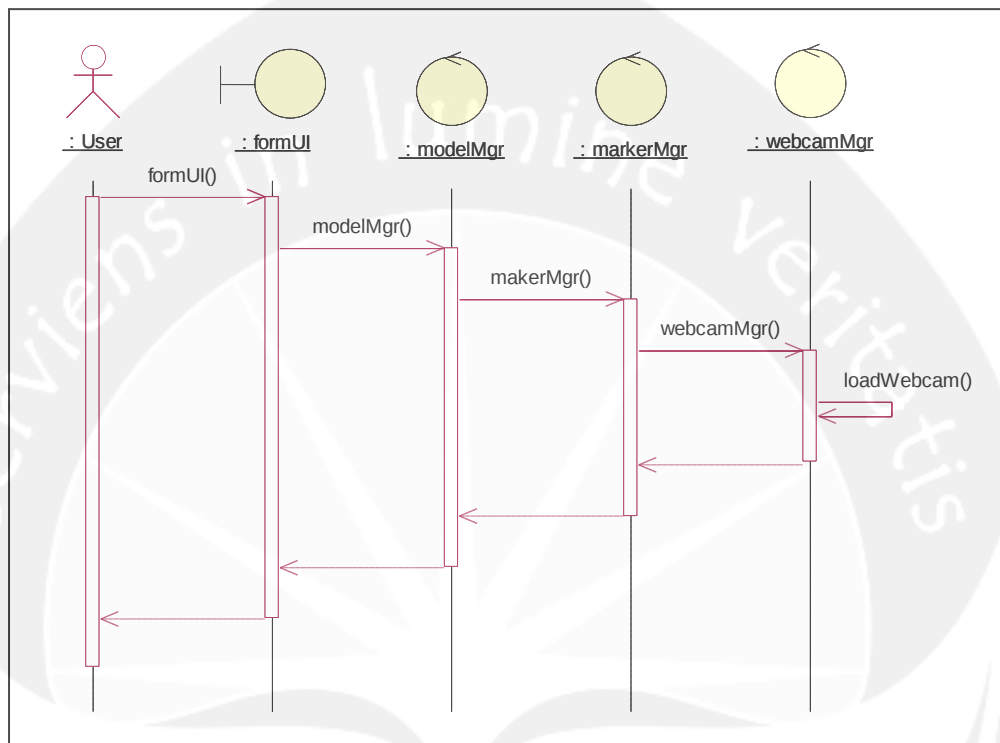


Gambar 2.1 Perancangan Sistem: Perancangan Arsitektur

2.2. Perancangan Rinci

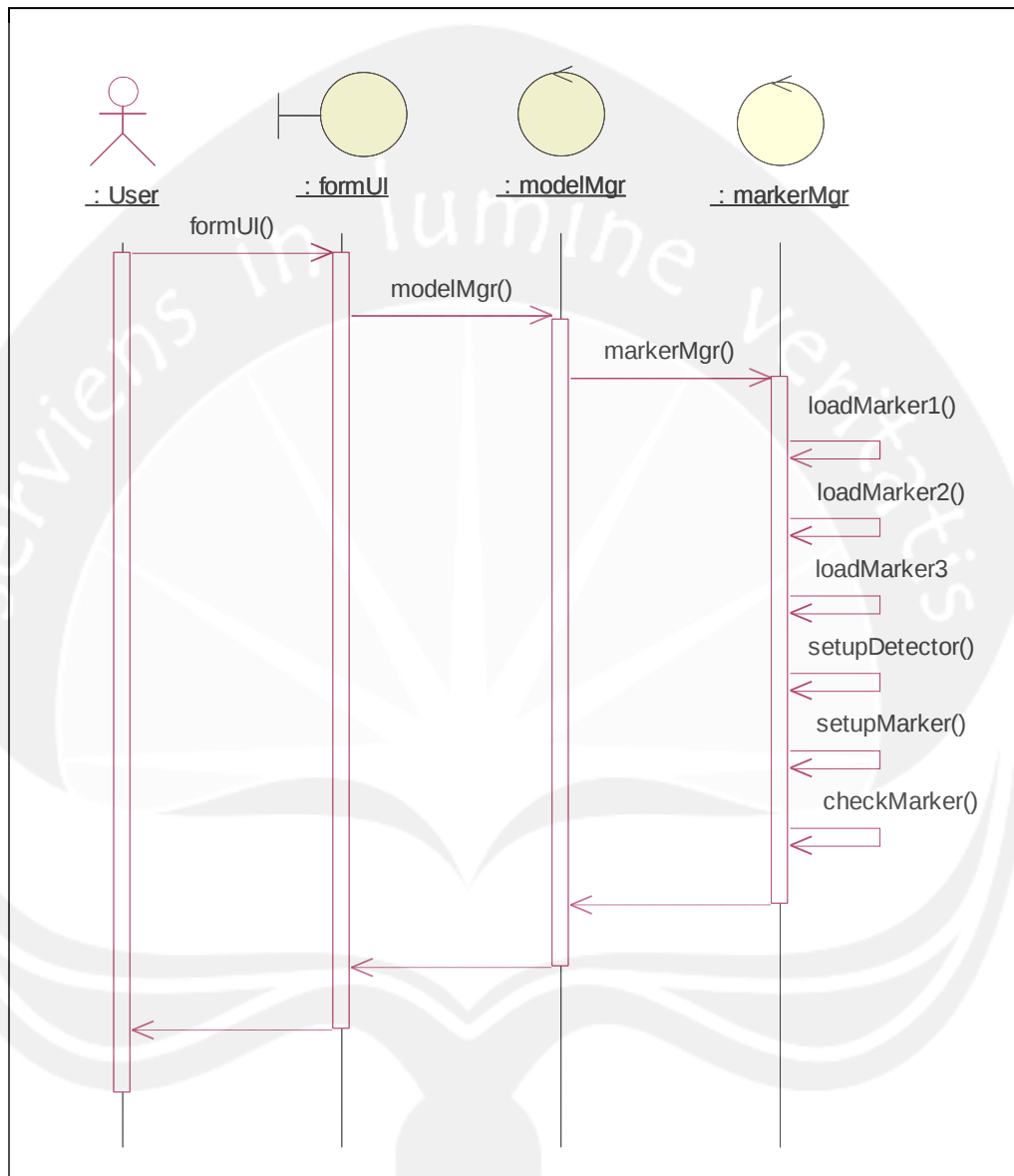
2.2.1. Sequence Diagram

2.2.1.1 Menangkap Gambar



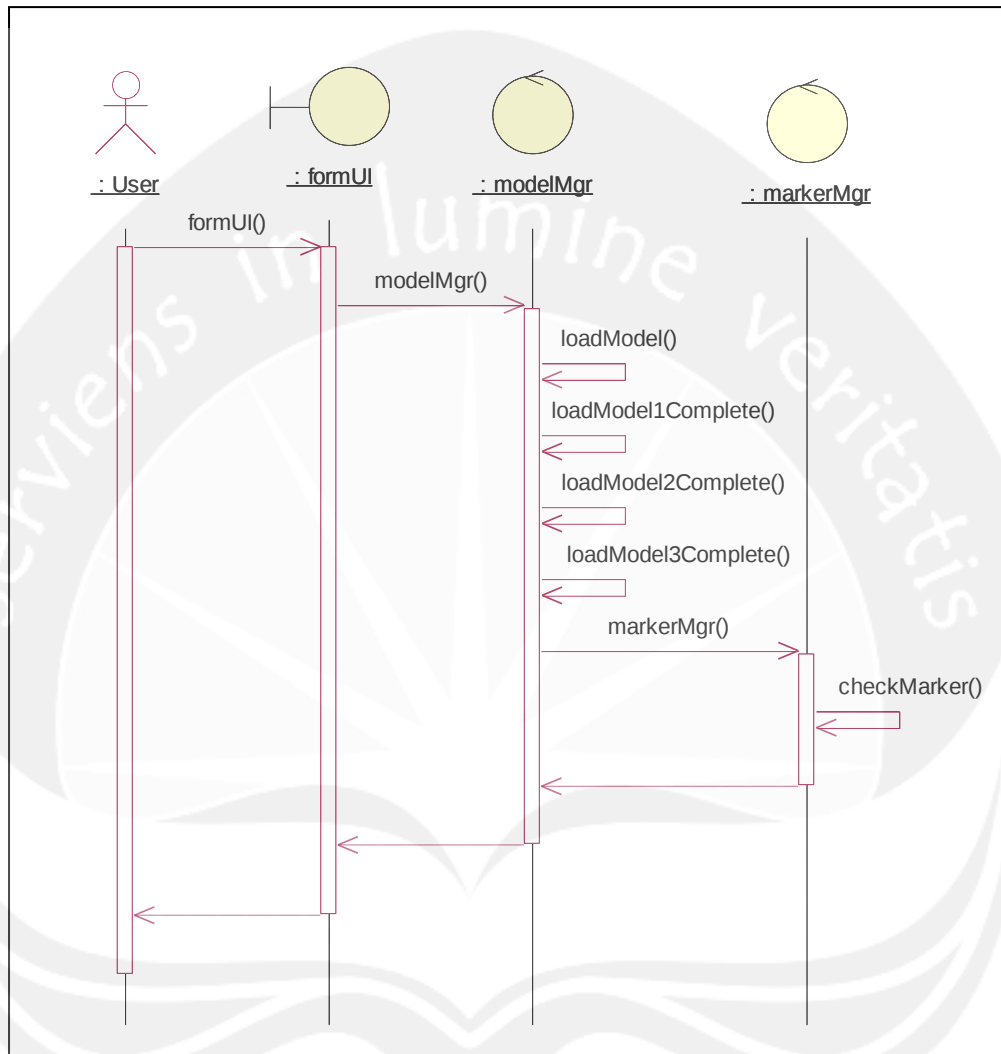
Gambar 2.2 Sequence Diagram: Menangkap Gambar

2.2.1.2 Melacak Marker



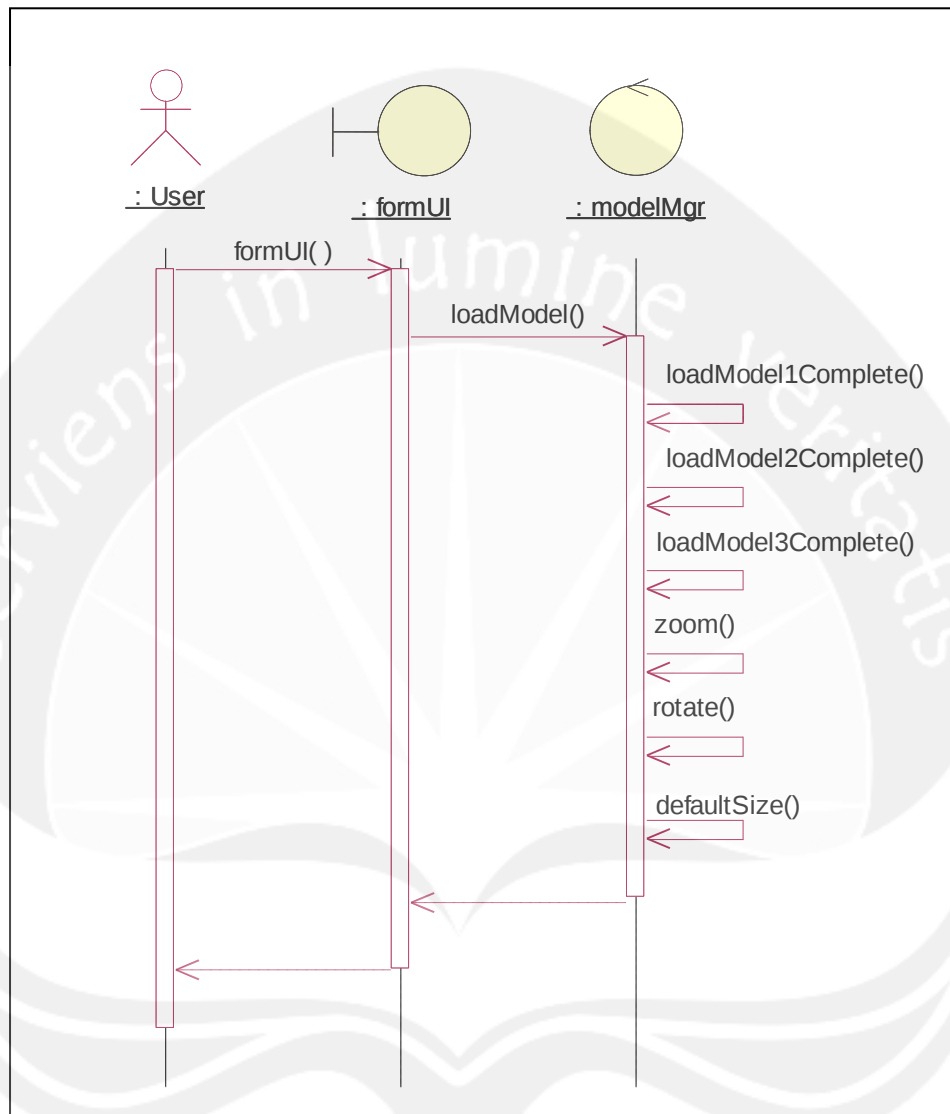
Gambar 2.3 Sequence Diagram: Melacak Marker

2.2.1.3 Menampilkan Model



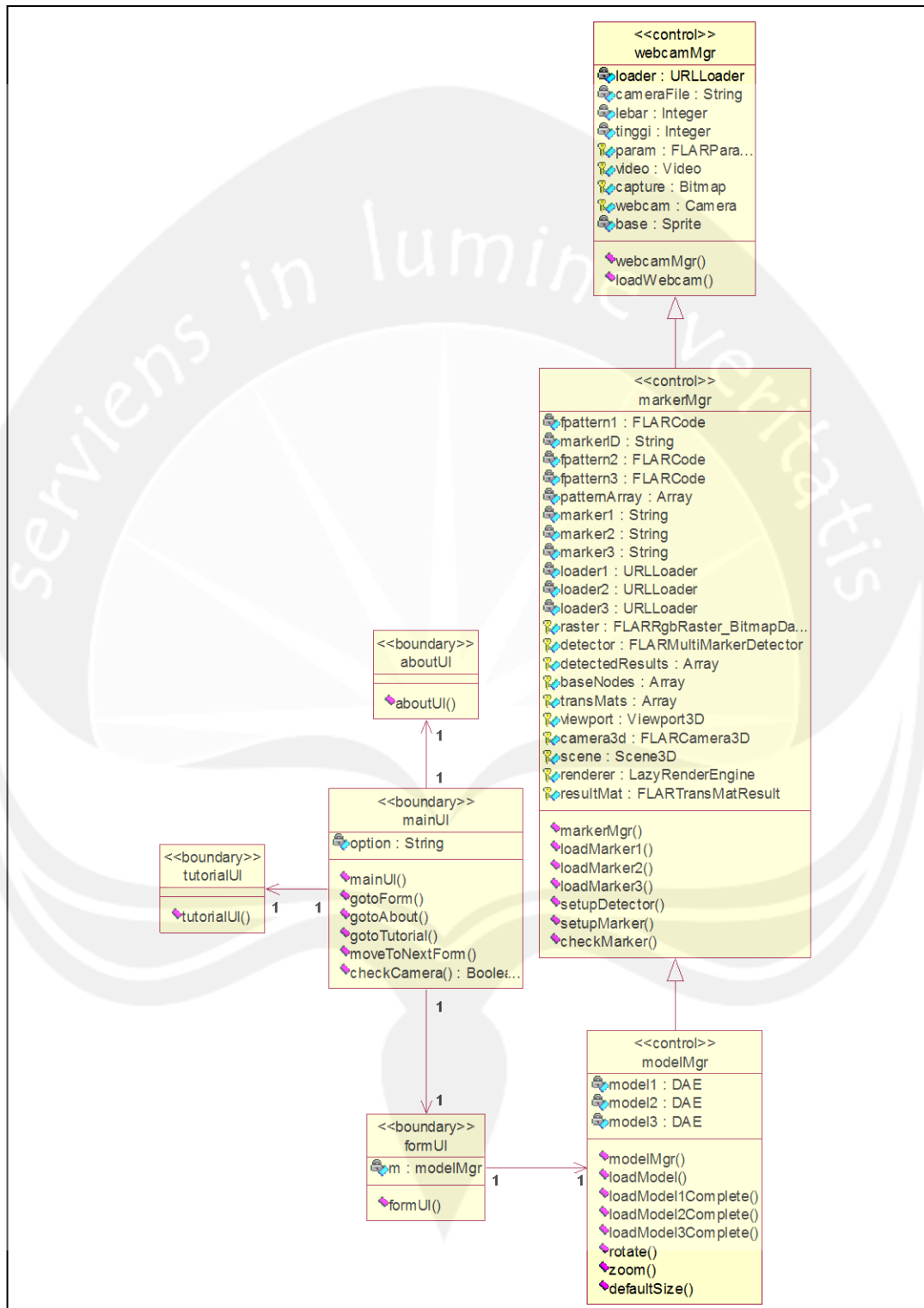
Gambar 2.4 Sequence Diagram: Menampilkan Model

2.2.1.4 Mengendalikan Model



Gambar 2.5 Sequence Diagram: Mengendalikan Model

2.2.2. Class Diagram



Gambar 2.6 Class Diagram

2.2.3. Spesifikasi Deskripsi Kelas Diagram

2.2.3.1 Spesifikasi Deskripsi Kelas mainUI

mainUI	<<boundary>>
<pre>-option:String</pre> Atribut ini digunakan untuk menyimpan data option.	
<pre>+mainUI():void</pre> Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. <pre>+gotoForm():void</pre> Operasi ini digunakan untuk mengeset nilai option dengan nilai String 1. <pre>+gotoAbout():void</pre> Operasi ini digunakan untuk mengeset nilai option dengan nilai String 3. <pre>+gotoTutorial()+void</pre> Operasi ini digunakan untuk mengeset nilai option dengan nilai String 2. <pre>+moveToNextForm()</pre> Operasi ini digunakan untuk memanggil file .swf berdasarkan dengan nilai option. <pre>+checkCamera():Boolean</pre> Operasi ini digunakan untuk mengecek apakah ada webcam yang tersambung.	

2.2.3.2 Spesifikasi Deskripsi Kelas formUI

formUI	<<boundary>>
<pre>-m:modelMgr</pre> Atribut ini berupa class modelMgr.	
<pre>+formUI():void</pre> Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

2.2.3.3 Spesifikasi Deskripsi Kelas aboutUI

aboutUI	<<boundary>>
<pre>+aboutUI():void</pre> Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

2.2.3.4 Spesifikasi Deskripsi Kelas tutorialUI

tutorialUI	<<boundary>>
<pre>+tutorialUI():void</pre> Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

2.2.3.5 Spesifikasi Deskripsi Kelas webcamMgr

webcamMgr	<<control>>
<pre>-loader:URLLoader</pre> Atribut ini digunakan untuk menyimpan data permintaan file. <pre>-cameraFile:String</pre> Atribut ini digunakan untuk menyimpan data lokasi file kamera. <pre>-lebar:Integer</pre> Atribut ini digunakan untuk menyimpan data lebar tampilan. <pre>-tinggi:Integer</pre> Atribut ini digunakan untuk menyimpan data tinggi tampilan. <pre>#param:FLARParam</pre> Atribut ini digunakan untuk menyimpan data kamera. <pre>#video:Video</pre> Atribut ini digunakan untuk menyimpan data video. <pre>#capture:Bitmap</pre> Atribut ini digunakan untuk menyimpan data gambar. <pre>#webcam:Camera</pre>	

Atribut ini digunakan untuk menyimpan data webcam yang terdeteksi.

#base:Sprite

Atribut ini digunakan untuk menyimpan data dari layar.

+webcamMgr():void

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+loadWebcam():void

Operasi ini digunakan untuk melakukan inisialisasi dan melakukan setting pada webcam.

2.2.3.6 Spesifikasi Deskripsi Kelas markerMgr

markerMgr	<<control>>
-fpattern1:FLARCode Atribut ini digunakan untuk menyimpan data pattern dari marker pertama. -fpattern2:FLARCode Atribut ini digunakan untuk menyimpan data pattern dari marker kedua. -fpattern3:FLARCode Atribut ini digunakan untuk menyimpan data pattern dari marker ketiga. -patternArray:Array Atribut ini digunakan untuk menyimpan data fpattern1, fpattern2, dan fpattern3. -marker1:String Atribut ini digunakan untuk menyimpan data lokasi pattern dari fpattern1. -marker2:String Atribut ini digunakan untuk menyimpan data lokasi pattern dari fpattern2. -marker3:String Atribut ini digunakan untuk menyimpan data lokasi pattern	

dari fpattern3.

-loader1:URLLoader

Atribut ini digunakan untuk menyimpan data dari file pattern pertama.

-loader2:URLLoader

Atribut ini digunakan untuk menyimpan data dari file pattern kedua.

-loader3:URLLoader

Atribut ini digunakan untuk menyimpan data dari file pattern ketiga.

-markerID:String

Atribut ini digunakan untuk menyimpan data id marker yang terdeteksi.

#raster:FLARRgbRaster_BitmapData

Atribut ini digunakan untuk menyimpan data gambar yang ditangkap webcam.

#detector:FLARMultiMarkerDetector

Atribut ini digunakan untuk mendeteksi semua marker.

#detectedResults:Array

Atribut ini digunakan untuk menyimpan data hasil deteksi.

#baseNodes:Array

Atribut ini digunakan untuk menyimpan data dari model 3d.

#transMats:Array

Atribut ini digunakan untuk menyimpan data matriks dari gambar yang ditangkap.

#viewport:Viewport3D

Atribut ini digunakan untuk menyimpan data dari viewport yang akan ditampilkan.

#camera3d:FLARCamera3D

Atribut ini digunakan untuk menyimpan data dari kamera yang akan dipakai.

#scene:Scene3D

Atribut ini digunakan untuk menyimpan data dari scene yang akan dirender.

#renderer:LazyRenderEngine

Atribut ini digunakan untuk menyimpan data dari kelas untuk render objek 3d.

#resultMat:FLARTransMatResult

Atribut ini digunakan untuk menyimpan data dari hasil matriks.

+markerMgr():void

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+loadMarker1():void

Operasi ini digunakan untuk mendapatkan data pattern untuk marker1 dari file *.pat.

+loadMarker2():void

Operasi ini digunakan untuk mendapatkan data pattern untuk marker2 dari file *.pat.

+loadMarker3():void

Operasi ini digunakan untuk mendapatkan data pattern untuk marker3 dari file *.pat.

+setupDetector():void

Operasi ini digunakan untuk menginisialisasi atribut kelas detector dan raster.

+setupMarker():void

Operasi ini digunakan untuk menginisialisasi atribut kelas viewport, camera3d, scene, renderer, baseNodes dan detectedResults.

+checkMarker():void

Operasi ini digunakan untuk menggambar yang ditangkap oleh webcam ke monitor, melakukan pelacakan marker, dan melakukan render objek 3d.

2.2.3.7 Spesifikasi Deskripsi Kelas modelMgr

modelMgr	<<control>>
-model1:DAE Atribut ini digunakan untuk menyimpan data objek 3d pertama. -model2:DAE Atribut ini digunakan untuk menyimpan data objek 3d kedua. -model3:DAE Atribut ini digunakan untuk menyimpan data objek 3d ketiga.	
+modelMgr():void Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +loadModel():void Operasi ini digunakan untuk melakukan inisialisasi objek 3d yang akan dipakai. +loadModelComplete():void Operasi ini digunakan untuk memasukkan model ke dalam node untuk ditampilkan dan berjalan ketika model1 berhasil diambil ke aplikasi. +loadModel2Complete():void Operasi ini digunakan untuk memasukkan model ke dalam node untuk ditampilkan dan berjalan ketika model2 berhasil diambil ke aplikasi. +loadModel3Complete():void Operasi ini digunakan untuk memasukkan model ke dalam node untuk ditampilkan dan berjalan ketika model3 berhasil diambil ke aplikasi. +rotate():void Operasi ini digunakan untuk menambahkan interaksi tombol <i>keyboard</i> untuk memutar model. +zoom():void Operasi ini digunakan untuk menambahkan interaksi tombol <i>keyboard</i> untuk memperbesar atau memperkecil model.	

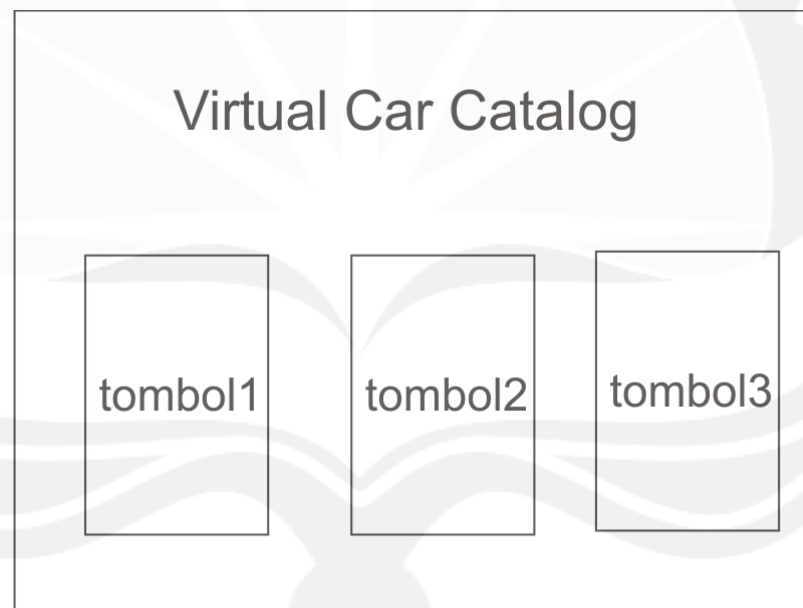
```
+defaultSize():void
```

Operasi ini digunakan untuk menambahkan interaksi tombol *keyboard* untuk mengembalikan model ke ukuran semula.

3. Perancangan Antarmuka

3.1 mainUI

Gambar 3.1 merupakan antarmuka yang berupa menu. Ada tiga buah tombol yang dapat digunakan. Tombol1 digunakan untuk menuju antarmuka formUI, tombol2 digunakan untuk menuju antarmuka tutorialUI, dan tombol3 digunakan untuk menuju antarmuka aboutUI.



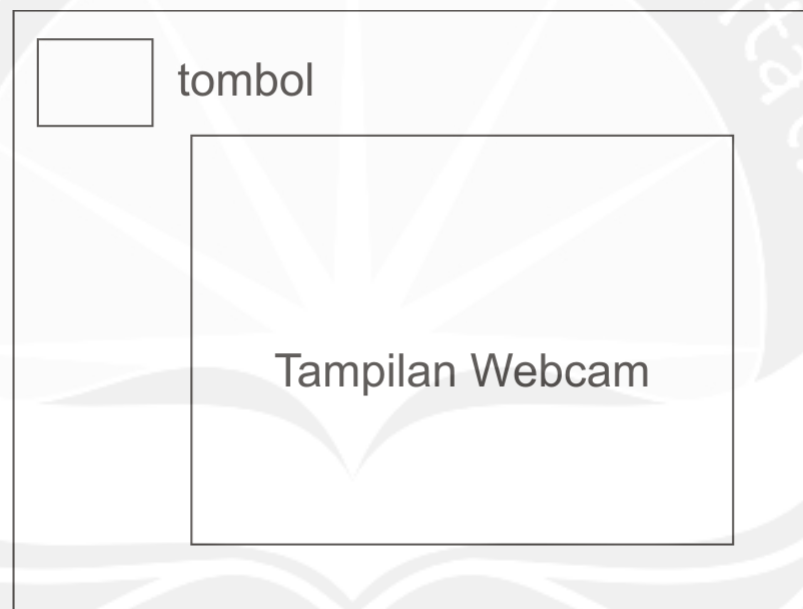
Gambar 3.1 Rancangan Antarmuka mainUI

Ketika pengguna menekan tombol1 maka akan ada pengecekan apakah *webcam* ada atau tidak, jika tidak maka akan ada peringatan bahwa pengguna harus memasang *webcam* terlebih dahulu. Pada kondisi tersebut pengguna tidak dapat berpindah ke antarmuka formUI, namun pengguna dapat berpindah ke antarmuka

tutorialUI, dan antarmuka aboutUI. Agar pengguna dapat berpindah ke antarmuka formUI, maka pengguna harus memasang webcam terlebih dahulu.

3.2 formUI

Antarmuka pada gambar 3.2 adalah antarmuka yang digunakan untuk menampilkan gambar yang ditangkap oleh webcam, menampilkan model, dan mengendalikan model.

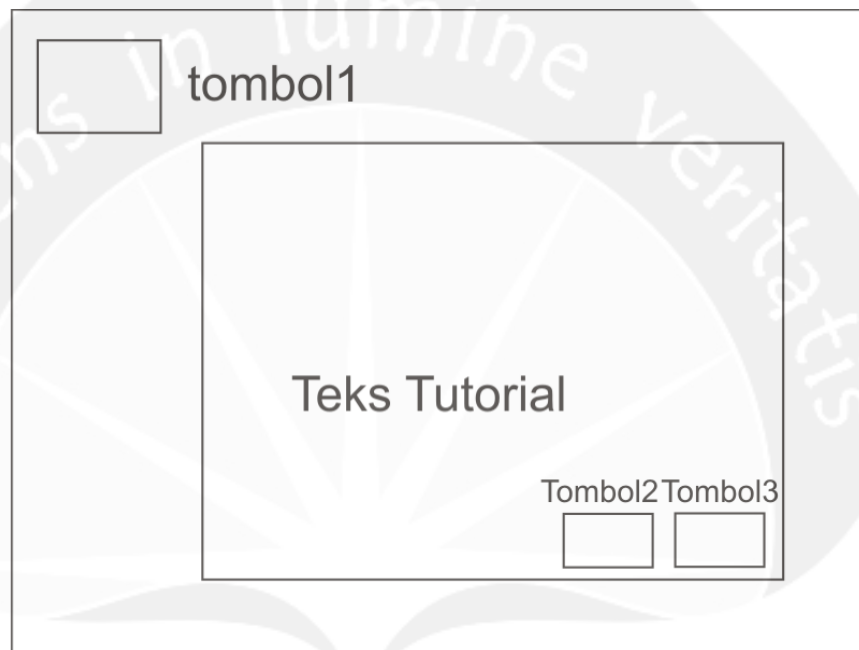


Gambar 3.2 Rancangan Antarmuka formUI

Pada tampilan ini terdapat sebuah kotak yang berisi Tampilan Webcam. Pada Tampilan Webcam akan ditampilkan gambar dari video yang ditangkap oleh webcam secara live. Ada sebuah tombol yang dapat digunakan pada tampilan ini. Tombol tersebut digunakan untuk kembali ke mainUI.

3.3 tutorialUI

Antarmuka pada gambar 3.3 merupakan antarmuka yang digunakan untuk menampilkan cara pemakaian aplikasi ini. Pada antarmuka ini terdapat tiga buah tombol.

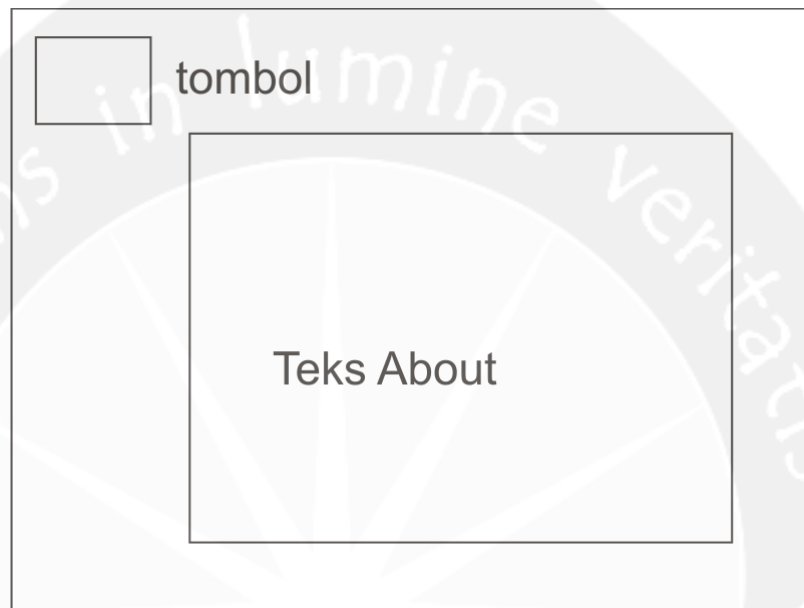


Gambar 3.3 Rancangan Antarmuka tutorialUI

Tombol1 digunakan untuk kembali ke mainUI. Teks tutorial berisi langkah-langkah dan penjelasan bagaimana cara pemakaian aplikasi ini. Tombol2 digunakan untuk kembali ke langkah sebelumnya, sedangkan tombol3 digunakan untuk ke langkah selanjutnya.

3.4 aboutUI

Gambar 3.4 adalah gambar antarmuka yang digunakan untuk menampilkan tentang pembuat aplikasi ini.



Gambar 3.4 Rancangan Antarmuka aboutUI

Teks About berisi mengenai nama dan kontak dari pembuat. Dalam tampilan ini terdapat sebuah tombol yang dapat digunakan. Tombol tersebut digunakan untuk kembali ke mainUI.

PDHUPL

PERENCANAAN, DESKRIPSI, DAN HASIL UJI PERANGKAT LUNAK VCC

(Virtual Car Catalog)

Dipersiapkan oleh:

Michael Alvado 5308

Program Studi Teknik Informatika – Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

Jl. Babarsari 43, Jogjakarta 50281

	Program Studi Teknik Informatika FTI - UAJY	Nomor Dokumen		Halaman
		<i>PDHUPL-VCC</i>		<i>1/12</i>
		Revisi	-	

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1. Pendahuluan	6
1.1 Tujuan Pembuatan Dokumen	6
1.2 Deskripsi Umum Sistem	6
1.3 Deskripsi Dokumen (Ikhtisar)	6
1.4 Definisi dan Singkatan	7
1.5 Dokumen Referensi	7
2. Lingkungan dan Pengujian Perangkat Lunak	8
2.1 Perangkat Lunak Pengujian	8
2.2 Perangkat Keras Pengujian	8
2.3 Materi Pengujian	8
2.4 Sumber Daya Manusia	8
2.5 Prosedur Umum Pengujian	9
2.5.1 Pengenalan dan Pelatihan	9
2.5.2 Persiapan Awal	9
2.5.2.1 Persiapan Prosedural	9
2.5.2.2 Persiapan Perangkat Keras	9
2.5.2.3 Persiapan Perangkat Lunak	9
2.5.3 Pelaksanaan	9
2.5.4 Pelaporan Hasil	9
3 Identifikasi dan Rencana Pengujian	10
4 Deskripsi dan Hasil Uji	10
4.1 Identifikasi Kelas Pengujian Antarmuka formUI	10
4.1.1 Identifikasi Butir Pengujian Menangkap Gambar – PDHUPL-VCC-01	11
4.1.2 Identifikasi Butir Pengujian Melacak Marker – PDHUPL-VCC-02	11
4.1.3 Identifikasi Butir Pengujian Menampilkan Model – PDHUPL-VCC-03	11
4.1.4 Identifikasi Butir Pengujian Mengendalikan Model fungsi Rotate – PDHUPL-VCC-04-01	11
4.1.5 Identifikasi Butir Pengujian Mengendalikan Model fungsi Zoom – PDHUPL-VCC-04-02	11
4.1.6 Identifikasi Butir Pengujian Mengendalikan Model fungsi Default Size – PDHUPL-VCC-04-03	11

Daftar Tabel

	hal
Tabel 1. Definisi	7
Tabel 2. Identifikasi Pengujian	10
Tabel 3. Deskripsi dan Hasil Pengujian.....	12



Pendahuluan

1.1 Tujuan Pembuatan Dokumen

Dokumen PDHUPL-VCC ini adalah dokumen yang berisi perencanaan, deskripsi dan hasil pengujian perangkat lunak yang spesifikasi terdapat pada dokumen PDHUPL-VCC, yaitu VCC (Virtual Car Catalog). Dokumen PDHUPL-VCC ini dibuat untuk jurusan Teknik Informatika Universitas Atma Jaya Yogyakarta. Selanjutnya dokumen PDHUPL-VCC ini dipergunakan sebagai bahan panduan untuk melakukan pengujian terhadap VCC. PDHUPL-VCC ini juga akan digunakan untuk menguji keseluruhan sistem VCC.

1.2 Deskripsi Umum Sistem

VCC adalah aplikasi katalog penjualan mobil dengan menggunakan *augmented reality*. Sistem ini secara garis besar terdiri dari empat komponen besar, yaitu:

- a. modul yang menangani webcam,
- b. modul yang menangani marker
- c. modul yang menangani model 3D
- d. modul yang menangani interaksi model.

1.3 Deskripsi Dokumen (Ikhtisar)

Dokumen PDHUPL-VCC ini mempunyai sistematika penulisan sebagai berikut:

Bagian 1. Pendahuluan

1.1. Tujuan Pembuatan Dokumen

1.2. Deskripsi Umum Sistem

1.3. Deskripsi Dokumen atau Ikhtisar

1.4. Definisi dan Singkatan

1.5. Dokumen Referensi

Bagian 2. Lingkungan Pengujian Perangkat Lunak

2.1. Perangkat Lunak Pengujian

2.2. Perangkat Keras Pengujian

2.3. Material Pengujian

2.4. Sumber Daya Manusia

- 2.5. Prosedur Umum
 - 2.5.1. Pengenalan dan Latihan
 - 2.5.2. Persiapan Awal
 - 2.5.2.1. Persiapan Prosedural
 - 2.5.2.2. Persiapan Perangkat Keras
 - 2.5.2.3. Persiapan Perangkat Lunak
 - 2.5.3. Pelaksanaan
 - 2.5.4. Pelaporan Hasil
- Bagian 3. Identifikasi dan Rencana Pengujian
- Bagian 4. Deskripsi dan Hasil uji
 - 4.1. Identifikasi Kelas Pengujian
 - 4.1.1. Identifikasi Butir Pengujian

1.4 Definisi dan Singkatan

Tabel 1. Definisi

Kata Kunci atau Frase	Definisi
Aungmented Reality	Teknologi yang menggabungkan benda maya ke dalam lingkungan nyata secara real time.
Marker	Marker adalah gambar yang memiliki pola tertentu yang dicetak dengan printer.
Webcam	Webcam atau web camera adalah kamera digital yang terhubung dengan komputer.

1.5 Dokumen Referensi

1. SKPL VCC dengan nomor dokumen SKPL-VCC.
2. DPPL VCC dengan nomor dokumen DPPL-VCC.
2. *Standard IEEE* nomor ANSI/IEEE Std 1058.1-1987 (*Reaffirmend* 1993).
3. Software Engineering, 1997, Roger S. Pressman, Mc Graw-Hill International Edition.

2 Lingkungan Pengujian Perangkat Lunak

2.1 Perangkat Lunak Pengujian

Perangkat lunak Pengujian berupa:

1. Windows 7 versi Professional dari Microsoft sebagai sistem operasi
2. Adobe Flash Debug Player 10 dari Adobe, sebagai penjalan aplikasi

2.2 Perangkat Keras Pengujian

1. Komputer PC, berjumlah minimal 1 buah, dengan spesifikasi minimal Core 2 Duo 2,2 Ghz, 2048 MB RAM.
2. Laptop, berjumlah minimal 1 buah, dengan spesifikasi minimal Core 2 Duo 2,2 Ghz, 2048 MB RAM.
3. Webcam Sirius USB 2.0
4. Printer untuk mencetak katalog
5. Mouse sebagai peralatan antarmuka.
6. Keyboard sebagai peralatan antarmuka.
7. Monitor 17 inci sebagai peralatan antarmuka.

2.3 Material Pengujian

Material tambahan untuk pengujian ini yaitu:

1. Spesifikasi Program VCC

2.4 Sumber Daya Manusia

Sumber daya manusia yang terlibat dalam kegiatan pengujian aplikasi VCC berjumlah lima orang dengan tingkat keahlian dan kriteria sebagai berikut :

1. Memiliki pemahaman tentang metode dan teknik pengujian perangkat lunak yang benar.
2. Memiliki kemampuan untuk mencari kesalahan yang terdapat pada aplikasi dan cara penanganan untuk memperbaikinya.

2.5 Prosedur Umum Pengujian

2.5.1 Pengenalan dan Latihan

Pada pengujian perangkat lunak tidak dilakukan pengenalan dan pelatihan terhadap perangkat lunak VCC dengan anggapan penguji telah memiliki pengetahuan tentang VCC itu sendiri selama proses pengembangannya.

2.5.2 Persiapan Awal

2.5.2.1 Persiapan Prosedural

Untuk melakukan kegiatan pengujian aplikasi VCC, tidak diperlukan izin khusus karena dilakukan di komputer di Laboratorium Informatika Lanjut Kampus III Fakultas Teknologi Industri.

2.5.2.2 Persiapan Perangkat Keras

Perangkat Keras beserta spesifikasinya berupa:

1. PC Core Duo 2,2 GHz, 2048 MB Ram
2. Laptop Core Duo 2,2 GHz, 2048 MB RAM
3. Webcam Sirius USB 2.0

2.5.2.3 Persiapan Perangkat Lunak

1. Perangkat Lunak VCC disiapkan dalam flashdisk
2. Siapkan Adobe Flash Debug Player 10
3. Siapkan listing modul apa saja yang akan diuji.

2.5.3 Pelaksanaan

Pelaksanaan pengujian akan dilaksanakan dalam dua tahap, yaitu pengujian unit (modul-modul kecil) dan pengujian sistem secara keseluruhan.

2.5.4 Pelaporan Hasil

Dokumen ini akan digunakan oleh tim pengembang sebagai referensi untuk melakukan proses analisis terhadap kualitas perangkat lunak serta perbaikan dan pemeliharaan yang perlu dilakukan, sekaligus sebagai referensi untuk pengembangan lebih lanjut dari aplikasi VCC ini.

3 Identifikasi dan Rencana Pengujian

Tabel 2 berisi butir pengujian, tingkat pengujian, jenis pengujian dan jadwal pengujian perangkat lunak Virtual Car Catalog.

Tabel 2. Identifikasi Pengujian

Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian	Jadwal
		SKPL	PDHUPL			
Pengujian antarmuka formUI	Pengujian Menangkap Gambar	SKPL-VCC-001	PDHUPL-VCC-001	Pengujian Unit	Black Box	8/06/2011
	Pengujian Melacak Marker	SKPL -VCC-002	PDHUPL-VCC-002	Pengujian Unit	Black Box	8/06/2011
	Pengujian Menampilkan Model	SKPL -VCC-003	PDHUPL-VCC-003	Pengujian Unit	Black Box	8/06/2011
	Pengujian Mengendalikan Model fungsi Rotate	SKPL -VCC-004-01	PDHUPL-VCC-004-01	Pengujian Unit	Black Box	8/06/2011
	Pengujian Mengendalikan Model fungsi Zoom	SKPL -VCC-004-02	PDHUPL-VCC-004-02	Pengujian Unit	Black Box	8/06/2011
	Pengujian Mengendalikan Model, fungsi Default Size	SKPL -VCC-004-03	PDHUPL-VCC-004-03	Pengujian Unit	Black Box	8/06/2011

4 Deskripsi dan Hasil Uji

4.1 Identifikasi Kelas Pengujian Antarmuka formUI

Kelas Pengujian antarmuka formUI adalah kelas pengujian yang meliputi pengujian-pengujian yang melibatkan fungsi antarmuka.

4.1.1 Identifikasi Butir Pengujian Menangkap Gambar – PDHUPL-VCC-001

Butir pengujian ini menguji untuk menampilkan gambar yang ditangkap oleh *webcam*.

4.1.2 Identifikasi Butir Pengujian Melacak Marker – PDHUPL-VCC-002

Butir pengujian ini menguji untuk melacak marker dengan menggunakan *webcam*.

4.1.3 Identifikasi Butir Pengujian Menampilkan Model – PDHUPL-VCC-003

Butir pengujian ini menguji untuk menampilkan model 3D ke atas *marker* yang terlacak.

4.1.4 Identifikasi Butir Pengujian Mengendalikan Model fungsi Rotate – PDHUPL-VCC-004-01

Butir pengujian ini menguji interaksi antara *keyboard* dan model 3D, apakah dengan menekan left arrow atau right arrow akan memutar model 3D ke kanan atau ke kiri.

4.1.5 Identifikasi Butir Pengujian Mengendalikan Model fungsi Zoom – PDHUPL-VCC-004-02

Butir pengujian ini menguji interaksi antara *keyboard* dan model 3D, apakah dengan menekan up arrow atau down arrow akan mengubah ukuran model 3D menjadi kecil atau menjadi lebih besar.

4.1.6 Identifikasi Butir Pengujian Mengendalikan Model fungsi Default Size – PDHUPL-VCC-004-03

Butir pengujian ini menguji interaksi antara *keyboard* dan model 3D, apakah dengan menekan space akan mengembalikan ukuran model 3D menjadi ukuran semula.

Hasil pengujian dapat dilihat pada Tabel 3. Tabel ini berisi prosedur pengujian, masukan, keluaran yang diharapkan, criteria evaluasi hasil, hasil yang didapat, dan kesimpulan.

Tabel 3. Deskripsi dan Hasil Pengujian

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yg diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PDHUPL-VCC-001	Pengujian menampilkan gambar yang ditangkap oleh webcam secara <i>live</i>	- Pada menu utama klik tombol Start	Klik kiri mouse pada button Start	Pada tampilan berikutnya akan tampil gambar yang ditangkap oleh <i>webcam</i> secara <i>live</i>	Tampilan berikutnya akan tampil gambar yang ditangkap oleh <i>webcam</i> secara <i>live</i>	Pada tampilan berikutnya akan tampil gambar yang ditangkap oleh <i>webcam</i> secara <i>live</i>	Handal
PDHUPL-VCC-002	Pengujian pendeteksian marker pada gambar yang ditangkap oleh <i>webcam</i>	- Pada menu utama klik tombol Start - Arahkan <i>webcam</i> ke <i>marker</i>	Gambar yang ditangkap oleh <i>webcam</i> secara <i>live</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Handal
PDHUPL-VCC-003	Pengujian menampilkan model 3D	- Pada menu utama klik tombol Start - Arahkan <i>webcam</i> ke <i>marker</i>	Gambar yang ditangkap oleh <i>webcam</i> secara <i>live</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Gambar yang ditangkap akan menampilkan model 3D diatas <i>marker</i>	Handal
PDHUPL-VCC-004-01	Pengujian mengendalikan model 3D yang muncul dengan fungsi Rotate	- Pada menu utama klik tombol Start - Arahkan <i>webcam</i> ke <i>marker</i> - Model 3D muncul diatas <i>marker</i>	Menekan tombol <i>left arrow</i> atau <i>right arrow</i>	Model 3D yang muncul berputar ke kanan atau kiri	Model 3D yang muncul berputar ke kanan atau kiri	Model 3D yang muncul berputar ke kanan atau kiri	Handal
PDHUPL-VCC-004-02	Pengujian mengendalikan model 3D yang muncul dengan fungsi Zoom	- Pada menu utama klik tombol Start - Arahkan <i>webcam</i> ke <i>marker</i> - Model 3D muncul diatas <i>marker</i>	Menekan tombol <i>up arrow</i> atau <i>down arrow</i>	Model 3D yang muncul ukurannya bertambah besar atau kecil	Model 3D yang muncul ukurannya bertambah besar atau kecil	Model 3D yang muncul ukurannya bertambah besar atau kecil	Handal

PDHUPL-VCC-004-03	Pengujian mengendalikan model 3D yang muncul dengan fungsi Zoom	- Pada menu utama klik tombol Start - Arahkan webcam ke <i>marker</i> - Model 3D muncul diatas marker	Menekan tombol <i>space</i>	Model 3D yang muncul ukurannya kembali seperti semula	Model 3D yang muncul ukurannya kembali seperti semula	Model 3D yang muncul ukurannya kembali seperti semula	Handal
-------------------	---	---	-----------------------------	---	---	---	--------





