



**IMPLEMENTASI PEMROGRAMAN BERORIENTASI
OBJEK, *OBSERVER DESIGN PATTERN*, DAN
SCRIPTABLEOBJECT DALAM PEMBUATAN
SOFTWARE ARCHITECTURE GAME 3D ACTION RPG
“NEVERGREEN”**

SKRIPSI

RAINALDI SATRIA SUKARNO

1910511030

**UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN
JAKARTA**

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI S1 INFORMATIKA

2023



**IMPLEMENTASI PEMROGRAMAN BERORIENTASI
OBJEK, *OBSERVER DESIGN PATTERN*, DAN
SCRIPTABLEOBJECT DALAM PEMBUATAN
SOFTWARE ARCHITECTURE GAME 3D ACTION RPG
“NEVERGREEN”**

SKRIPSI

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar
Sarjana Komputer**

RAINALDI SATRIA SUKARNO

1910511030

**UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN
JAKARTA**

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI S1 INFORMATIKA

2023

PERNYATAAN ORISINALITAS

Tugas Skripsi ini adalah hasil karya sendiri dan semua sumber yang dikutip maupun yang dirujuk telah saya nyatakan dengan benar.

Nama : Rainaldi Satria Sukarno
NIM : 1910511030
Tanggal : 16 Januari 2024

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan saya ini, maka saya bersedia dituntut dan diproses sesuai dengan ketentuan yang berlaku.

Jakarta, 15 Januari 2024

Yang Menyatakan,



Rainaldi Satria Sukarno

PERNYATAAN PERSETUJUAN PUBLIKASI

Sebagai civitas akademik Universitas Pembangunan Nasional Veteran Jakarta, saya yang bertanda tangan di bawah ini:

Nama : Rainaldi Satria Sukarno

NIM : 1910511030

Fakultas : Ilmu Komputer

Program Studi : SI Informatika

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Pembangunan Nasional Veteran Jakarta Hak Bebas Royalti Non eksklusif (*Non-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul

Implementasi Pemrograman Berorientasi Objek, *Observer Design Pattern*, dan ScriptableObject dalam Pembuatan *Software Architecture Game 3D Action RPG "NeverGreen"*

Berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti ini Universitas Pembangunan Nasional Veteran Jakarta berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan Tugas Akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Dengan pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jakarta

Pada tanggal : 15 Januari 2024

Yang Menyatakan,



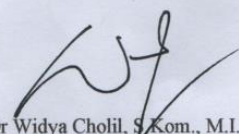
Rainaldi Satria Sukarno


LEMBAR PENGESAHAN

Tugas Akhir ini diajukan oleh:

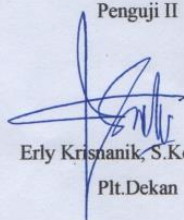
Nama : Rainaldi Satria Sukarno
NIM : 1910511030
Program Studi : S1 Informatika
Judul Tugas Akhir : Implementasi Pemrograman Berorientasi Objek,
Observer Design Pattern, dan *ScriptableObject*
dalam pembuatan *Software Architecture Game 3D*
Action RPG "NeverGreen"

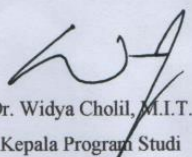
Telah berhasil dipertahankan dihadapan Tim Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana pada Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jakarta.


Dr Widya Cholil, S.Kom., M.I.T.
Penguji I


Nurul Afifah Arifuddin, S.Pd., M.T.
Penguji II


Yuni Widiastiwi, S.Kom., M.Si
Pembimbing


Erly Krisnanik, S.Kom., MM
Plt. Dekan


Dr. Widya Cholil, M.I.T.
Kepala Program Studi

Ditetapkan di : Jakarta
Tanggal Ujian : 19 Desember 2023



**IMPLEMENTASI PEMROGRAMAN BERORIENTASI
OBJEK, *OBSERVER DESIGN PATTERN*, DAN
SCRIPTABLEOBJECT DALAM PEMBUATAN
SOFTWARE ARCHITECTURE GAME 3D ACTION RPG
“NEVERGREEN”**

RAINALDI SATRIA SUKARNO

ABSTRAK

Seiring berkembangnya *scope* dari suatu *game*, pengembangannya akan menjadi semakin sulit. Pengembangan *game* dengan *software architecture* yang tidak tepat akan menghasilkan banyak isu dan masalah, seperti risiko *bug* yang besar, penundaan proyek, hingga biaya yang lebih mahal. Oleh karena itu, diperlukan solusi untuk mengatasi permasalahan tersebut. Telah ada berbagai upaya untuk mengatasi permasalahan-permasalahan yang berkaitan dengan *software architecture*. Namun, belum ditemukan penelitian yang membahasnya dalam konteks pembuatan *game* bergenre 3D *action RPG*, khususnya yang berfokus menggunakan pemrograman berorientasi objek, *observer design pattern*, dan *ScriptableObject*. Oleh karena itu, tujuan penelitian ini adalah membuat *software architecture* yang dapat diandalkan menggunakan ketiga prinsip pemrograman tersebut. Prototipe *game* bernama *NeverGreen* dibuat sebagai studi kasus. *NeverGreen* adalah *game* bergenre 3D *action RPG* yang memperkenalkan isu bioterorisme. *Game*-nya dibuat menggunakan Unity 2022. Penelitian ini menerapkan metode *Game Development Life Cycle (GDLC)* oleh Heather Chandler. Berdasarkan hasil penelitian, pemaduan ketiga prinsip pemrograman tersebut mampu membuat *game* *NeverGreen* yang sesuai dengan *requirement* yang diperlukan tanpa mengorbankan kualitas. *Software architecture*-nya bersifat *modular, flexible, extensible, reusable, editable, dan debuggable*.

Kata kunci: *software architecture*, pemrograman berorientasi objek, *observer design pattern*, *ScriptableObject*, Unity

**IMPLEMENTATION OF OBJECT-ORIENTED
PROGRAMMING, OBSERVER DESIGN PATTERN, AND
SCRIPTABLEOBJECT IN THE CREATION OF SOFTWARE
ARCHITECTURE FOR 3D ACTION RPG GAME
“NEVERGREEN”**

RAINALDI SATRIA SUKARNO

ABSTRACT

As the scope of a game project expands, its development becomes increasingly complex. Developing a game with bad software architecture can lead to various issues, such as high bug risks, project delays, and higher costs. Therefore, a solution is needed to overcome these challenges. While there have been various efforts to address issues related to software architecture, there is no in-depth research on this topic in the context of creating 3D action RPG games, particularly those focused on using object-oriented programming, observer design patterns, and ScriptableObject. Hence, this research aims to create reliable software architecture by applying these three programming principles. As a case study, a game prototype named NeverGreen was created. NeverGreen is a 3D action RPG game that addresses the issue of bioterrorism, developed using Unity 2022. This research adopts the Game Development Life Cycle (GDLC) method by Heather Chandler. The research's results indicate that the integration of these three programming principles successfully produces the case study game, meeting the required specifications without compromising quality. The software architecture is modular, flexible, extensible, reusable, editable, and debuggable.

Keyword(s): *software architecture, object-oriented programming, observer design pattern, ScriptableObject, Unity*

PRAKATA

Puji syukur penulis panjatkan kepada Allah, Pencipta langit dan bumi. Satu Allah. Oleh karena kasih karunia-Nya, penulis dapat menyelesaikan Tugas Akhir ini.

Dalam penyusunan penelitian ini, penulis memperoleh banyak dukungan dan bantuan dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada:

1. Tuhan Yesus Kristus sebagai Penasihat Ajaib, Allah yang Perkasa, Bapa yang Kekal, Raja Damai.
2. Ibu Yuni Widiastiw, S.Kom, M.Si. selaku Dosen Pembimbing yang telah memperbolehkan, mengarahkan, serta memberikan masukan berharga terhadap penelitian ini.
3. Ibu Dr. Widya Cholil, M.I.T. selaku Dosen Penguji I dan Ibu Nurul Afifah Arifuddin, S.Pd., M.T selaku Dosen Penguji II yang telah menguji dan memberikan perbaikan skripsi ini.
4. Bapak Moch. Fariz Satyawan serta seluruh staf UPN Veteran Jakarta atas kelancaran proses administrasi dan perkuliahan di kampus.
5. Seluruh anggota keluarga penulis yang telah senantiasa mendoakan, memberikan semangat dan harapan, serta bantuan.
6. Tim R9 Tech, Ardhi A. K. K., dan semua pihak yang tidak dapat disebutkan satu persatu, tanpa mengurangi rasa hormat, atas semua kontribusi dalam penyusunan skripsi ini baik secara langsung maupun tidak langsung.

Tentunya dalam pembuatan skripsi ini, penulis menyadari banyaknya kekurangan pada laporan ini. Namun, penulis berharap penelitian ini dapat memberikan manfaat atau inspirasi bagi pembaca yang tertarik dengan penelitian ini.

Terakhir, penulis ingin menutup prakata ini dengan sebuah kabar baik tentang kehidupan yang kekal. Pemberian gratis dari Allah bagi mereka yang percaya:

Jawab Yesus: "Akulah kebangkitan dan hidup; barangsiapa percaya kepada-Ku, ia akan hidup walaupun ia sudah mati

--Yohanes 11:25 (TB)

Pencuri datang hanya untuk mencuri dan membunuh dan membinasakan; Aku datang, supaya mereka mempunyai hidup, dan mempunyainya dalam segala kelimpahan.

--Yohanes 10:9-10 (TB)

Jakarta, 16 Januari 2024

Penulis

DAFTAR ISI

COVER	i
HALAMAN JUDUL.....	ii
PERNYATAAN ORISINALITAS	iii
PERNYATAAN PERSETUJUAN PUBLIKASI	iv
LEMBAR PENGESAHAN	v
ABSTRAK	vi
ABSTRACT.....	vii
PRAKATA.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xvi
DAFTAR POTONGAN KODE.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Ruang Lingkup.....	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Luaran Yang Diharapkan.....	5
BAB II LANDASAN TEORI.....	6
2.1 Video Game	6
2.2 RPG.....	6
2.3 Action Game	6
2.4 Edutainment	6
2.5 Unity	7
2.6 Software Architecture	7
2.7 UML Class Diagram.....	9
2.8 Pemrograman Berorientasi Objek.....	13
2.9 Observer Design Pattern	15
2.10 ScriptableObject.....	15
2.11 Arsitektur dengan ScriptableObject.....	16

2.12	Game Development Life Cycle oleh Heather M. Chandler.....	20
2.13	Storyboard.....	21
2.14	Blackbox Testing	21
2.15	Bioterrorisme.....	21
2.16	Agen Biologis	21
2.17	Mutan	22
2.18	Literatur Sebelumnya.....	22
BAB III METODE PENELITIAN		26
3.1	Tahapan Penelitian.....	26
3.2	Alat yang Digunakan	28
3.3	Jadwal Penelitian	29
BAB IV HASIL DAN PEMBAHASAN		30
4.1	Pra-produksi.....	30
4.2	Produksi	40
4.2.1	GlobalRefSO	40
4.2.2	Sistem Dialog.....	43
4.2.3	Sistem Pilihan	52
4.2.4	Sistem Dialog x Sistem Pilihan.....	60
4.2.5	Sistem Persediaan	63
4.2.6	Sistem Pemakaian Peralatan	75
4.2.7	Sistem Pemakaian Senjata	87
4.2.8	Atribut RPG	91
4.2.9	Sistem Pertempuran	104
4.3	Pengujian.....	109
4.3.1	Sistem Dialog.....	109
4.3.2	Sistem Pilihan	110
4.3.3	Sistem Dialog x Sistem Pilihan.....	111
4.3.4	Sistem Persediaan	112
4.3.5	Sistem Pemakaian Peralatan	112
4.3.6	Sistem Pemakaian Senjata	113
4.3.7	Atribut RPG	114
4.3.8	Sistem Pertempuran	116
4.3.9	Kesesuaian dengan Storyboard	117

4.4	Post-Produksi	117
4.4.1	Hal yang Berjalan dengan Baik	117
4.4.2	Hal yang Perlu Diperbaiki.....	118
BAB V SIMPULAN DAN SARAN		119
5.1	Kesimpulan	119
5.2	Saran	119
DAFTAR PUSTAKA		121
RIWAYAT HIDUP		125
LAMPIRAN.....		126
LAMPIRAN 1. Tangkapan Layar di NeverGreen		126
LAMPIRAN 2. IResetSO		133
LAMPIRAN 3. Game Serupa		134
LAMPIRAN 4. Hasil Turnitin		150

DAFTAR GAMBAR

Gambar 2.1 Kelas pada Kelas Diagram	10
Gambar 2.2 Relasi Pewarisan	11
Gambar 2.3 Relasi Agregat	11
Gambar 2.4 Relasi Komposisi	12
Gambar 2.5 Atribut pada Kelas Diagram.....	12
Gambar 2.6 ScriptableObject sebagai Aset di Editor.....	15
Gambar 2.7 ScriptableObject Variable (Sumber: Unity Technologies, 2017)	16
Gambar 2.8 ScriptableObject Event (Sumber: Unity Technologies, 2021).....	17
Gambar 2.9 Arsitektur SO Menghilangkan Dependency	20
Gambar 2.10 Terlalu banyak SO.....	20
Gambar 3.1 Tahapan Penelitian	26
Gambar 4.1 Mockup Dialog.....	32
Gambar 4.2 Mockup Pilihan	32
Gambar 4.3 Mockup Sistem Persediaan	33
Gambar 4.4 Mockup Sistem Peralatan.....	33
Gambar 4.5 Mockup Atribut RPG	34
Gambar 4.6 Mockup Pertempuran	36
Gambar 4.7 AudioManagerBinder di Inspector.....	41
Gambar 4.8 MonoBehaviour Action Memicu Respon Lintas Scene.....	42
Gambar 4.9 Sistem Dialog dalam NeverGreen.....	43
Gambar 4.10 Kelas Diagram Actor.....	43
Gambar 4.11 Actor sebagai Aset	44
Gambar 4.12 Garis Garis Dialog.....	44
Gambar 4.13 Kelas Diagram DialogueLine.....	45
Gambar 4.14 Diagram Kelas DialogueLine yang SOLID	46
Gambar 4.15 DialogueUI dalam NeverGreen.....	46
Gambar 4.16 Diagram Kelas DialogueUI.....	47
Gambar 4.17 Contoh Perluasan Observer Design Pattern di DialogueUI	49
Gambar 4.18 Diagram Kelas DialogueTrigger	49
Gambar 4.19 Kelas Diagram DialogueTrigger karena Keterbatasan Inspector.....	51
Gambar 4.20 LocalizedDialogueTrigger di Inspector	51
Gambar 4.21 Sistem Pilihan dalam NeverGreen	52
Gambar 4.22 Kelas Diagram Choice	53
Gambar 4.23 ChoiceUI dalam NeverGreen	53
Gambar 4.24 Kelas Diagram ChoiceUI	54
Gambar 4.25 Kelas Diagram UIElement	55
Gambar 4.26 Kelas Diagram ChoiceTrigger	56
Gambar 4.27 Kelas Diagram ChoiceTrigger karena Keterbatasan Inspector	58
Gambar 4.28 LocalizedChoiceTrigger di Inspector.....	58
Gambar 4.29 Kelas Diagram Trigger.....	59
Gambar 4.30 Sistem Dialog x Sistem Pilihan NeverGreen	60
Gambar 4.31 Kelas Diagram DialogueChoiceTrigger.....	61
Gambar 4.32 Hierarki DialogueChoiceTrigger.....	62

Gambar 4.33 ConcreteDialogueChoiceTrigger di Inspector	62
Gambar 4.34 Sistem Persediaan NeverGreen	63
Gambar 4.35 Kelas Diagram Item	63
Gambar 4.36 Diagram Kelas dari Equipment	64
Gambar 4.37 Kelas Diagram dari Weapon dan Suit	65
Gambar 4.38 Kelas Diagram QuestItem	66
Gambar 4.39 Kelas Diagram InventoryManager	67
Gambar 4.40 InventoryUI	68
Gambar 4.41 Kelas Diagram dari InventoryUI	68
Gambar 4.42 Kelas Diagram dari InventoryUI	69
Gambar 4.43 NameAndDescriptionBox dalam NeverGreen	71
Gambar 4.44 Kelas Diagram ItemDescriptionBox	71
Gambar 4.45 Kelas Diagram ItemNameAndDescriptionbox	72
Gambar 4.46 Susunan Komponen di NameAndDescriptionBox	73
Gambar 4.47 Kelas Diagram AddItemTrigger dan RemoveItemTrigger	73
Gambar 4.48 AddItemTrigger (kiri) RemoveItemTrigger (kanan) di Inspector	74
Gambar 4.49 Sistem Pemakaian Peralatan NeverGreen	75
Gambar 4.50 Kelas Diagram EquipmentManager	75
Gambar 4.51 Kelas Diagram CharacterEquipmentSlotManager	77
Gambar 4.52 EquipmentMenu NeverGreen	78
Gambar 4.53 Kelas Diagram EquipmentUI	79
Gambar 4.54 Kelas Diagram EquipmentUITrigger	82
Gambar 4.55 EquipmentUITrigger di Inspector	83
Gambar 4.56 Kotak Deskripsi di EquipmentUI	84
Gambar 4.57 Kelas Diagram EquipmentUI_NameAndDescriptionBox	84
Gambar 4.58 EquipmentSelectionUI di NeverGreen	85
Gambar 4.59 Kelas Diagram EquipmentSelectionUI	86
Gambar 4.60 Sistem Pemakaian Senjata NeverGreen	87
Gambar 4.61 Kelas Diagram EquipmentAndModel	88
Gambar 4.62 Kelas Diagram EquipmentSpawnerManager	89
Gambar 4.63 Sistem Atribut RPG NeverGreen	91
Gambar 4.64 Kelas Diagram HPSO dan IntSO	92
Gambar 4.65 Kelas Diagram ExperienceManager dan CharacterExperience	93
Gambar 4.66 Kelas Diagram LevelManager dan CharacterLevel	94
Gambar 4.67 Kelas Diagram LevelCalculator	94
Gambar 4.68 Kelas Diagram HealthManager	95
Gambar 4.69 Kelas Diagram HealthCalculator	96
Gambar 4.70 Kelas Diagram StrengthManager	96
Gambar 4.71 Kelas Diagram StrengthCalculator	97
Gambar 4.72 Kelas Diagram AttackManager	97
Gambar 4.73 Kelas Diagram AttackCalculator	98
Gambar 4.74 AttributeUI	98
Gambar 4.75 Kelas Diagram AttributeUI dan AttributeUIFiller	99
Gambar 4.76 Kelas Diagram AttributeUITrigger	100
Gambar 4.77 AttributeUITrigger di Inspector	101

Gambar 4.78 Antarmuka LevelTextController	102
Gambar 4.79 Kelas Diagram LevelTextController	102
Gambar 4. 80 Antarmuka HealthBarController	103
Gambar 4.81 Kelas Diagram HealthBar	103
Gambar 4.82 Sistem Pertempuran NeverGreen	104
Gambar 4.83 Kelas Diagram DamageReceiver	104
Gambar 4.84 Kelas Diagram DamageDealer	106
Gambar 4.85 Kelas Diagram DamageTextManager	107
Gambar 4.86 Integrasi DamageTextTrigger dengan DamageReceiver	108
Gambar 4.87 Teks Terpucu Saat DamageReceiver Menerima Kerusakan.....	109

DAFTAR TABEL

Tabel 3.1 Jadwal Penelitian	29
Tabel 4.1 Simulasi Level Berdasarkan Pengalaman	34
Tabel 4.2 Simulasi HP Berdasarkan Level dan Perlengkapan Kesehatan	35
Tabel 4.3 Simulasi Strength Berdasarkan Level	35
Tabel 4.4 Simulasi Serangan Berdasarkan Kekuatan dan Perlengkapan Serangan	36
Tabel 4.5 Skenario Pengujian Sistem Dialog	109
Tabel 4.6 Skenario Pengujian Sistem Pilihan	110
Tabel 4.7 Skenario Pengujian Sistem Dialog x Sistem Pilihan	111
Tabel 4.8 Skenario Pengujian Sistem Persediaan	112
Tabel 4.9 Skenario Pengujian Pemakain Peralatan	112
Tabel 4.10 Skenario Pengujian Pemakain Senjata	113
Tabel 4.11 Skenario Pengujian Sistem Atribut RPG	114
Tabel 4.12 Skenario Pengujian Sistem Pertempuran	116
Tabel 4.13 Skenario Pengujian Kesesuaian dengan Storyboard	117

DAFTAR POTONGAN KODE

Potongan Kode 2.1 Kelas CharacterSO	15
Potongan Kode 4.1 GlobalRefSO	40
Potongan Kode 4.2 GlobalRefBinder	41
Potongan Kode 4.3 AudioManagerBinder	41
Potongan Kode 4.4 Contoh Pemakaian GlobalRefSO	42