

# KAJIAN GAME TAIPAN DALAM PEMBUATAN MODEL PENGEMBANGAN DESAIN GAME

**Tedi Lesmana Marselino<sup>1</sup>, Paramaresthi Windriyani<sup>2</sup>**

<sup>1,2</sup>Program Studi Informatika, Fakultas Industri Kreatif, Kalbis Institute, Jln. Pulomas Selatan  
Kav. 22, Jakarta Timur, 13210, Indonesia

*E-mail: <sup>1</sup>tedi.lesmana@kalbis.ac.id, <sup>2</sup>paramaresthi.windriyani@kalbis.ac.id*

## ABSTRAK

Tujuan penelitian ini adalah membuat model dalam pengembangan desain game yang didasarkan atas permainan klasik yang bernama game Taipan. Cara yang dilakukan dengan melakukan dekonstruksi game Taipan dan mengidentifikasi elemen pembentuknya. Game Taipan dikaji dengan proses dekonstruksi yang dilakukan menggunakan tahapan pengembangan perangkat lunak *Rational Unified Process*. Dalam proses dekonstruksi, game di-dekomposisi untuk mendapatkan elemen-elemen pembentuknya dengan memainkan game dan melakukan identifikasi elemen-elemen game. Selanjutnya elemen-elemen diidentifikasi, dikelompokkan, dan di-sistematisasi-kan, untuk menghasilkan model desain yang dapat digunakan sebagai referensi pengembangan game. Pengujian model desain dapat dilakukan dengan melakukan remake game Taipan menggunakan model yang terbentuk. Hasil akhir model desain berupa elemen-elemen dan tahapan pembuatan permainan.

**Kata kunci**—pengembangan desain game, Taipan, dekonstruksi, *Rational Unified Process*

## ABSTRACT

*This research aims to develop model in game design development on the basis of classic playing name Taipan game. The way to do this by deconstructing Taipan game and identifying the forming elements. Taipan game reviewed by deconstruction process using Rational Unified Process Software Development Life Cycle. In deconstructing process, game decomposed to get the forming elements by playing and identifying elements of the game. Following the elements were identified, categorized, and sistematized to resulting the model that can be used as a reference to develop a game. The design model can be tested by remake the Taipan game using the formed model. The Final result of the design model formed as elements and steps to develop a game.*

**Keywords**—game design development, Taipan, deconstruction, *Rational Unified Process*

## 1. PENDAHULUAN

*Computer games* (selanjutnya disebut permainan komputer) merupakan salah satu bentuk akhir dari perangkat lunak komputer. Layaknya perangkat lunak komputer, maka perangkat lunak ini mengikuti tahap pembuatan perangkat lunak seperti pada *Software Development Life Cycle* (SDLC). Dalam perkembangannya yang pesat permainan komputer menjadi semakin kompleks dan melibatkan banyak elemen dan fungsi-fungsi yang tidak ada di dalam perangkat lunak biasa [1]

Beberapa kekhususan dari perangkat lunak permainan komputer seperti adanya jalan cerita, animasi, suara, dan gambar. Permainan komputer mengoptimalkan penggunaan interaksi melalui berbagai macam teknologi antarmuka. Di sisi lain teknik pemrograman berorientasi objek mendukung interaksi berdasarkan kejadian yang muncul sebagai akibat adanya teknologi

grafis yang mendukung masukkan secara acak (dari *mouse*) yang disebut *event driven* dengan masukkan secara banyak (multi). Belakangan muncul teknologi-teknologi antarmuka yang membuat teknik pemrograman berorientasi objek lebih cocok digunakan yaitu teknologi *touch screen* yang berorientasi pada banyak masukkan yang terjadi secara acak.

Dari beberapa situasi yang demikian maka penulis menemukan bahwa pemrograman berorientasi objek menjadi pendekatan yang lebih cocok untuk pengembangan perangkat lunak permainan komputer. Pendekatan berbasis objek memungkinkan pembuatan modul-modul program dengan lebih bebas dan mudah selanjutnya dapat dikaitkan dengan interaksi yang terjadi secara banyak dan acak secara lebih fleksibel.

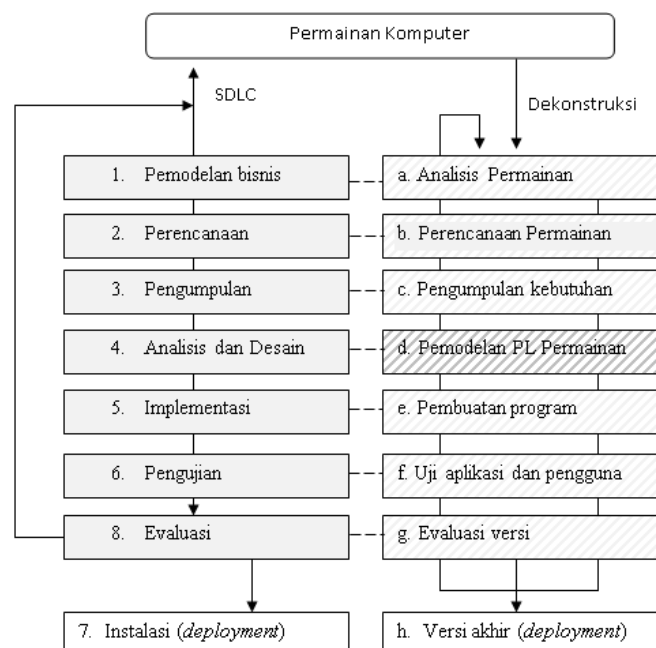
Situasi kedua yang menjadi pertimbangan adalah pemodelan perangkat lunak untuk permainan komputer kurang dapat diakomodir oleh pemodelan perangkat lunak seperti pada *Unified Markup Language* (UML) karena pemodelan ini lebih berorientasi bisnis sedangkan pemodelan dari model siklus hidup pengembangan perangkat lunak yang lain dirasa belum cocok. [2]

Siklus hidup pengembangan perangkat lunak yang terlalu besar, rumit atau teknik pemodelan perangkat lunak yang kurang memadai mendorong penulis untuk menemukan pemodelan baru yang lebih praktis dalam membuat perangkat lunak permainan komputer.

Karena permainan komputer lebih mendekati situasi model objek-objek maka penulis menggunakan RUP dan UML untuk mencoba melakukan dekonstruksi permainan komputer untuk menemukan elemen-elemen pokok permainan komputer. Setelah berhasil diidentifikasi maka elemen-elemen ini akan disusun ulang dengan usulan model perangkat lunak permainan komputer yang digagas oleh penulis untuk menghasilkan permainan komputer yang sama. [3]

## 2. METODE PENELITIAN

Bagian ini menjelaskan bagaimana SDLC dalam RUP digunakan untuk melakukan dekonstruksi game Taipan. [4][5]



Gambar 1. Kerangka Dekonstruksi

Pada Gambar 1 dapat dilihat dua bagian urutan siklus hidup pengembangan perangkat lunak dan dekonstruksi. Maksud dari bagian sebelah kiri dari gambar adalah, sebuah program aplikasi permainan komputer diasumsikan dapat dihasilkan dari suatu siklus hidup pengembangan perangkat lunak dalam hal ini RUP. Namun karena RUP ini masih berupa hipotesis maka di sini masih diandaikan bahwa siklus ini merupakan siklus yang dapat menghasilkan program aplikasi permainan komputer.

Dalam proses normal dapat dilihat bahwa terdapat 8 langkah yang dilakukan secara sistematis. Tujuh langkah dilakukan secara iteratif (berulang), sementara satu siklus dapat dianggap selesai sampai dengan proses ke tujuh. Setiap kali satu siklus selesai maka muncullah produk pertama (versi) dan siklus berikut merupakan hasil evaluasi dari siklus yang pertama, sehingga telah berhasil terjadi peningkatan dari hasil berikutnya. Proses iterasi ini sendiri dapat terjadi pada salah satu langkah, misalkan di tahap pengujian. Di langkah pengujian proses iteratif dapat terjadi berkali-kali dan tidak perlu menunggu satu siklus selesai sampai dengan langkah evaluasi. Sedangkan langkah ke delapan dilakukan berdasarkan hasil evaluasi. Apabila dirasakan sudah final maka iterasi akan berakhir pada instalasi atau proses *deployment*.

Bagian kanan dari Gambar 1 merupakan proses dekonstruksi. Proses ini merupakan proses pembalikan dari siklus hidup pengembangan perangkat lunak. Pada dasarnya langkah-langkah siklus adalah sama tetapi mengalami adaptasi untuk kasus program aplikasi permainan komputer. Sebagai contoh seperti yang ditunjukkan pada Gambar 1 tersebut “Pemodelan Bisnis” menjadi “Analisis Permainan (komputer)”. Demikian proses ini dilakukan pada langkah-langkah berikutnya hingga langkah yang ketujuh.

Khusus pada langkah d, yaitu “Pemodelan PL Permainan” dihasilkan suatu langkah-langkah pemodelan perangkat lunak yang dijadikan acuan dalam membuat rancangan perangkat lunak. Di sini dibedakan siklus hidup pengembangan perangkat lunak itu sendiri dan pemodelan perangkat lunak. Pemodelan perangkat lunak sendiri merupakan bagian dari siklus hidup pengembangan perangkat lunak. Pemodelan perangkat lunak sangat membantu pemrograman untuk mewujudkan bentuk konkrit dari program permainan komputer. Pemodelan perangkat lunak adalah desain “rumah” dari satu program, beserta rincian kebutuhan dari “rumah” perangkat lunak yang sebenarnya.

Langkah-langkah berikutnya yaitu e, f, g dan h merupakan langkah-langkah untuk menguji apakah model yang dihasilkan dari dekonstruksi a, b, dan c menghasilkan pemodelan yang relevan, artinya benar-benar dapat dijadikan acuan untuk melakukan reka ulang pembuatan perangkat lunak yang serupa. Apabila pemodelan terbukti benar maka model dapat digunakan dalam kasus pengembangan perangkat lunak yang lain.

## 2.1 Analisis Game

Permainan komputer Taipan adalah permainan komputer berbasis teks dengan tema perdagangan. Inti permainan cukup sederhana, yaitu membeli barang murah di satu tempat dan menjualnya di tempat lain. Tempat-tempat tersebut adalah kota-kota di wilayah Asia. [4]

Untuk meningkatkan rasa permainan tentunya terdapat halangan dan rintangan agar permainan tidak berjalan dalam emosi datar. Di sini pemain akan berhadapan dengan musuh seperti bajak laut. Kapal dapat karam apabila mendapat kerusakan akibat meriam atau pemilik kapal dapat saja bangkrut karena tidak mampu membayar hutang. Hasil analisis permainan Taipan dapat dilihat pada Tabel 1.

Hasil analisis meskipun belum teratur tetap dapat diterima apa adanya, untuk itu hasil analisis dituangkan di dalam tabel yang dikategorikan dan disistematisasikan. Di sini dapat dibuat pengelompokkan sebisa mungkin. Tujuan utama dari langkah analisis ini mendapatkan data awal sebagai hasil identifikasi elemen-elemen pokok desain permainan.

**Tabel 1.** Hasil Analisis Permainan Taipan

NO.	OBJEK	FUNGSI
1	Deskripsi cerita	Permainan adalah tentang perdagangan barang-barang di Tahun 1800an. Seorang pedagang dengan kantor pusat di Hongkong memiliki kapal. Kapal akan digunakan untuk mengangkut kargo. Kargo berisi barang-barang seperti candu, sutera, senjata, barang-barang lainnya. Ada beberapa kota atau negara di wilayah asia yang dapat dikunjungi. Pedagang dapat memulai perdagangan dengan meminjam uang dari lintah darat atau memulainya dengan kapal dan dibekali <i>canon</i> . Canon digunakan untuk menyerang atau melindungi kapal. Tujuan dari permainan ini adalah mendapatkan kekayaan sebesar mungkin dengan cara melakukan perdagangan dari satu kota ke kota lain untuk mendapatkan keuntungan. Keuntungan dapat diperoleh dengan mendapatkan selisih harga yang besar di mana pada saat itu pedagang juga memiliki stok barang yang banyak entah di kapal atau di gudang. Tetapi gudang hanya terdapat pada kantor pusat di Hong Kong.
2	Aturan permainan ( <i>game play</i> )	<p>Pedagang memiliki kondisi awal yaitu dengan hutang atau senjata. Terdapat beberapa kota sebagai daerah perdagangan, kota-kota berada di wilayah asia, kantor pusat hanya di satu tempat yaitu Hongkong dan hanya di kantor pusat dapat dilakukan perbaikan kapal, menyimpan dan mengambil uang di bank, melakukan pembayaran hutang, menyimpan dan mengambil barang di gudang.</p> <p>Di kota-kota perdagangan yaitu Hongkong, Batavia, Vietnam, Filipina, Singapura, Jepang, China, dan Taiwan dapat dilakukan beberapa hal seperti: jual-beli barang, membeli <i>canon</i>, menghadapi penghadangan oleh bajak laut, mengalami situasi seperti ketika menghadapi bajak laut pedagang dapat kabur dengan kerusakan kapal, atau kabur setelah bertempur.</p> <p>Kapal dapat dikembangkan dengan kemungkinan-kemungkinan seperti menambah kapasitas kargo, menambah <i>canon</i> yang berdampak pada pengurangan kapasitas kargo, memperbaiki kerusakan.</p> <p>Lin Yuen adalah panglima tentara kerajaan yang selalu meminta setoran keamanan, dan apabila tidak mendapatkan setoran akan mengabaikan perlindungan kapal dagang yang mengakibatkan kapal dagang sering diserang oleh bajak laut</p>
3	Entitas ( <i>object/class</i> )	pemain, bajak laut, kota, bank, kapal, <i>canon</i> , items, Lin Yuen, Lintah darat, bajak laut, tempat perbaikan, gudang, penasihat.
4	Fungsi-Fungsi dari Tiap Entitas ( <i>fungsi/prosedur/metode</i> )	Ditampilkan terpisah pada Tabel 2.
5	Kejadian-kejadian ( <i>events</i> ) yang melibatkan koding, visualisasi, variabel, dan membutuhkan algoritma tertentu	<p>a) Halaman utama: <i>welcoming screen, registering name, play alone agaist computer, set game initialization(debt/canon)</i></p> <p>b) <i>Headquarter</i>: status dari kapal (kapasitas kapal dan jumlah item), kondisi dari kapal (<i>health</i>), status keuangan (<i>debt/cash/bank</i>), status dari barang (<i>ship/warehouse</i>).</p> <p>c) Perdagangan (<i>buy/sell</i>), penyimpanan (<i>ship/warehouse</i>), <i>deposit/cash</i>.</p> <p>d) Perjalanan ke kota-kota lain.</p> <p>e) Saat di laut (aman, sedikit rintangan, banyak rintangan).</p> <p>f) Bertempur, lari, beberapa pertempuran.</p>

**Tabel 2.** Beberapa Fungsi yang Dibutuhkan dalam Permainan Taipan

NO.	OBJEK	FUNGSI
1	<i>Player</i>	-create name -start game -buy canon -borrow money -buy items -sell items
2	<i>Items</i>	-opium -silk -arm -general
3	<i>System</i>	-random of price -random of pirates ambush -random of Lin Yuen -random of canon offering

Proses dekonstruksi perlu dilakukan secara alamiah untuk membuka berbagai kemungkinan yang ada dari data yang berhasil dikumpulkan di dalam tabel menjadi suatu kumpulan data, tahapan atau item yang nantinya dibutuhkan di dalam pembuatan model.

Sebagai contoh seperti terlihat pada Tabel 1 berhasil diidentifikasi apa yang mungkin dibutuhkan di dalam perencanaan seperti deskripsi cerita, aturan permainan (*game play*), entitas, fungsi, dan algoritma. Sementara di dalam Tabel 2 ditemukan ada 3 objek atau entitas yang dibutuhkan seperti *player*, *items*, dan *system*. Pada Tabel 2 dapat diperkirakan fungsi-fungsi yang dibutuhkan seperti *create name*, *start game*, dan *buy canon*. Selanjutnya data hasil analisis ini dapat dijadikan data sementara untuk memperkirakan perencanaan permainan dalam siklus pengembangan perangkat lunak.

## 2.2 Perencanaan Game

Dalam sebuah permainan komputer terdapat komponen-komponen pokok yang dibutuhkan sehingga program aplikasi permainan komputer dapat masuk kategori program aplikasi permainan komputer. Untuk itu dapat dipinjam elemen formal sebagai panduan seperti: *player*, *objectives*, *procedures*, *rules*, *resources*, *conflict*, *boundaries*, dan *outcome*. [6]

Selanjutnya hasil analisis game Taipan dapat dipetakan dengan elemen formal sehingga nampak hasilnya pada Tabel 3 sebagai berikut:

**Tabel 3.** Elemen Formal Permainan Komputer dan Hasil Analisis Taipan

NO.	ELEMEN FORMAL	HASIL ANALISIS TAIPAN
1	<i>Player</i>	-Seorang pemain lokasi di Batavia. -Bajak laut. -Li Yuen. -Thief. -Kreditor. -Banker.
2	<i>Objectives</i>	Meningkatkan kekayaan lewat penjualan untuk mendapatkan skor tertinggi.
3	<i>Procedures</i>	Membeli barang dengan harga tertentu (murah) di suatu kota dan menjualnya dengan harga lebih mahal di kota lain.
4	<i>Rules</i>	-Pemain hanya dapat memulai berdagang (beli/jual) hanya apabila sudah memiliki uang tunas. -Uang tunai dapat diperoleh dari pilihan permainan di awal yaitu lewat pilihan tunai atau dari hasil rampasan dari bajak laut.

		-Pemain dapat menyimpan barang di gudang kantor pusat (Hongkong). -Pemain dapat menyimpan uang tunainya di bank. -Pemain hanya dapat menyimpan uang ketika di Hongkong.
5	<i>Resources</i>	-Barang dagangan: <i>opium, silk, arm, general</i> . -Kapal. -Gudang. -Senjata.
6	<i>Conflic</i>	-Pemain sering dimintai sumbangan. -Pemain dapat ditagih hutangnya. -Pemain dapat bertemu dengan bajak laut. -Uang pemain dapat dicuri. -Barang pemain dari gudang dapat hilang.
7	<i>Boundaries</i>	-Setiap barang memiliki harga terendah dan harga tertinggi. -Gudang memiliki kapasitas. -Kapal memiliki kapasitas. -Canon akan mengurangi kapasitas kapal.
8	<i>Outcome</i>	-Uang pemain hasil penjualan. -Skor permainan berdasarkan jumlah uang yang dimiliki, yaitu total uang pemain dan uang simpanan di bank.

Di sini meskipun elemen formal berhasil menentukan apa-apa yang harus dipersiapkan dalam pengembangan permainan lebih lanjut. Elemen formal tidak berhasil mengidentifikasi beberapa hal pada tahap analisis sebelumnya (merujuk pada Tabel 1 dan Tabel 2). Karena memang beberapa hal yang teridentifikasi sebelumnya tidak terlihat secara langsung di dalam elemen formal, tetapi lebih merupakan latar belakang atau hal-hal yang terpicu harus disediakan. Untuk itu hasil analisis sebelumnya tetap dibutuhkan selain daripada persiapan elemen-elemen formal. Dapat juga dikatakan di sini beberapa hal yang tidak teridentifikasi tersebut seperti algoritma, fungsi, metode dan sistem.

### 2.3 Pengumpulan Kebutuhan

Pengumpulan kebutuhan diterjemahkan sebagai kebutuhan dari pengguna yang dituangkan ke dalam kebutuhan sistem (perangkat lunak permainan). Tetapi karena proses di sini dilakukan secara terbalik, maka pengumpulan kebutuhan diidentifikasi sebagai penentuan (perkiraan) masukan dan keluaran.

Dalam proses komputerisasi normal maka proses komputasi terdiri dari proses masukan, proses komputasi dan proses keluaran. Maka secara logis, dapat ditentukan atau diidentifikasi masukan-masukkan apa saja yang dibutuhkan oleh program-program ini.

Proses identifikasi masukan dapat dilakukan dalam empat tahap, yaitu: pertama melakukan identifikasi umum masukan-masukkan apa saja yang dibutuhkan, yang kedua adalah usaha untuk menentukan keluaran-keluarannya, ketiga menentukan keluaran proses-proses apa saja yang membutuhkan masukan-masukkan dan keluaran, dan keempat menentukan objek atau entitas yang membutuhkan proses-proses tersebut yang dengan sendirinya terikat dengan kebutuhan masukan dan hasil keluaran.

Hasil keempat tahap tersebut dapat dilihat pada empat tabel berikut, yaitu Tabel 4, Tabel 5, Tabel 6 dan Tabel 7.

**Tabel 4.** Menentukan Masukkan-Masukkan Game Taipan  
**MASUKKAN-MASUKKAN PERMAINAN TAIPAN**

Nama pemain
Jumlah barang dibeli
Jumlah canon dibeli
Tujuan pelabuhan
Jumlah uang disimpan di bank
Jumlah uang diambil dari bank
Jumlah barang dialihkan ke gudang
Jumlah barang dikirimkan ke kapal

**Tabel 5.** Menentukan Keluaran-Keluaran Game Taipan  
**KELUARAN-KELUARAN PERMAINAN TAIPAN**

Tempat berlabuh
Harga barang
Pertemuan dengan bajak laut
Permintaan pembayaran pajak
Kondisi perbaikan kapal
Tawaran pembelian canon
Tawaran peningkatan kapasitas gudang
Skor nilai pemain

**Tabel 6.** Menentukan Proses-Proses Game Taipan  
**PROSES-PROSES PERMAINAN TAIPAN**

Pergi ke suatu tujuan
Pertempuran
Menghasilkan harga barang
Menawarkan proses peningkatan gudang
Menawarkan proses pembelian canon
Menawarkan proses pertempuran
Menawarkan <i>role</i> permainan
Membuat proses simpan-tarik bank

**Tabel 7.** Menentukan Objek-Objek Game Taipan

NO.	OBJEK-OBJEK PERMAINAN TAIPAN
1	Pemain
2	Bajak laut
3	Li Yuen
4	Bankir
5	Kreditor
6	Kapal
7	Barang
8	Gudang

**Tabel 8.** Gabungan Kebutuhan dalam Game Taipan

MASUKKAN	PROSES	KELUARAN
Nama pemain	Menawarkan <i>role</i> permainan	?
Tujuan pelabuhan	Pergi ke suatu tujuan	Tempat berlabuh
Jumlah barang dibeli	Pertempuran	Pertemuan dengan bajak laut
?	Menawarkan proses pertempuran	?
Jumlah canon dibeli	Menawarkan proses pembelian canon	Tawaran pembelian canon
?	Menghasilkan harga barang	Harga barang

?	Menawarkan proses peningkatan gudang	?
Jumlah uang disimpan di bank	Membuat proses simpan-tarik bank	?
Jumlah uang diambil dari bank	?	?
Jumlah barang dialihkan ke gudang	?	?
Jumlah barang dikirimkan ke kapal	?	?
?	?	Kondisi perbaikan kapal
?	?	Permintaan pembayaran pajak
?	?	Tawaran peningkatan kapasitas gudang
?	?	Skor nilai pemain

Tabel 8 adalah tabel yang menarik. Di sinilah terlihat bagaimana korelasi masukan, proses, dan keluaran. Di sini perancang dapat melakukan klarifikasi, koreksi atau pemeriksaan apakah semua masukan berhasil diidentifikasi, demikian juga dengan keluaran, dan proses. Di sini juga dapat diperiksa kecocokan antara ketiganya.

Sebagai contoh pada bagian keluaran skor nilai pemain, dari hasil identifikasi belum terlihat proses mana yang perlu dibuat untuk menghasilkan skor nilai pemain. Dengan demikian perancang permainan dapat menemukan kekurangan hasil identifikasinya, karena tidak mungkin ada keluaran tanpa suatu proses.

Sebaliknya apabila ditemukan suatu proses hanya menghasilkan keluaran seperti pada proses menghasilkan harga barang dan keluarannya harga barang, maka perancang dapat memikirkan parameter apa yang dibutuhkan untuk menghasilkan harga barang, karena harga barang meskipun dapat dibuat dengan fungsi acak (randomize) tetap membutuhkan parameter/nilai sebagai masukan.

Pemetaan masukan, proses, dan keluaran adalah suatu pemetaan yang sangat handal dan berguna untuk nantinya dijadikan persiapan desain program yang dibutuhkan dalam rancangan permainan.

Tahap selanjutnya adalah penentuan objek atau nantinya dianggap sebagai entitas. Layaknya pendekatan objek, maka perancang harus selalu berpikir dalam perspektif objek. Sebuah objek memiliki atribut dan metode. Metode adalah fungsi atau tingkah-laku dari objek. Oleh karena itu Tabel 8 dapat membantu melengkapi fungsi-fungsi apa saja yang mampu dilakukan oleh objek. Maka Objek dan kemampuan (metode) dapat dipetakan di dalam Tabel 9 dengan pendekatan seperti pada Tabel 8 yaitu yang memang belum memiliki pasangan dapat tetap dibiarkan kosong agar selanjutnya dapat diperbaiki.

Tabel 9 Pemetaan Objek dan Proses Permainan Taipan

<b>OBJEK</b>	<b>PROSES</b>
Pemain	Pergi ke suatu tujuan
Bajak laut	Pertempuran
Li Yuen	Menghasilkan harga barang
Bankir	Membuat proses simpan-tarik bank
Kreditor	? (menawarkan pinjaman)
Gudang	Menawarkan proses peningkatan gudang
? (sistem)	Menawarkan proses pembelian canon
Kapal	Menawarkan proses pertempuran
? (sistem)	Menawarkan <i>role</i> permainan
Barang	? (tidak melakukan proses, terkena proses, memiliki properti)



Tabel 9 sekali lagi memberikan bantuan untuk mengisi apa yang kosong seperti terlihat dalam tanda kurung. Terlihat pada proses menawarkan *role* permainan, berdasarkan hasil pemetaan belum berhasil diidentifikasi, maka dapat diperkirakan bahwa hal tersebut dilakukan oleh sistem. Atau seperti pada contoh objek barang, ternyata barang tidak melakukan proses tetapi diperkirakan terkena proses seperti bahwa barang dapat dijual, dibeli, dialihkan ke gudang atau kapal, maka dengan demikian barang akan diperlukan sebagai parameter dalam proses yang lain karena jika terkena proses maka objek akan menjadi parameter bagi proses yang lain. Konsekwensinya kalau objek dijadikan parameter dalam proses yang lain maka objek memiliki *properties* atau atribut, karena parameter perlu memiliki nilai dan nilai hampir pasti terikat pada atribut.

### 3. HASIL DAN PEMBAHASAN

Hasil dari dekonstruksi tiga tahap pertama yaitu a. analisis permainan, b. perencanaan permainan, dan c. pengumpulan kebutuhan sudah dapat digunakan untuk memprediksi kebutuhan pembentukan model desain perangkat lunak permainan komputer.

#### 3.1 Pemodelan Perangkat Lunak Game

Tabel 10 dapat dikonfirmasi berdasarkan temuan pada tabel-tabel sebelumnya. Sebagai contoh pada Tabel 10 terdapat tujuh (7) temuan hasil identifikasi yang diperkirakan dibutuhkan dalam proses pemodelan perangkat lunak permainan komputer.

Entitas dapat diperoleh dari Tabel 2 Beberapa fungsi yang dibutuhkan dalam permainan Taipan, atau Tabel 3 yaitu elemen-elemen formal, atau Tabel 7 yaitu objek-objek permainan Taipan. Secara intuisi entitas juga dapat dipilih sebagai yang pertama karena entitas sebagai kelas akan diturunkan menjadi objek yang memiliki atribut dan metode. Hanya saja masih dibutuhkan konformitas di sini sebelum hal tersebut dapat dinyatakan layak dimasukkan di dalam elemen model, yang nantinya akan dilakukan pada saat proses pengembangan kode program.

Yang kedua adalah atribut. Atribut dapat diperoleh berdasarkan pemetaan masukan dan keluaran. Begitu juga dengan Tabel 9. Secara logis suatu masukan dan keluaran memiliki nilai. Nilai ini pasti “menempel” pada variabel, maka variabel (atribut) ini nantinya akan juga terhubung pada suatu entitas (objek). Dengan demikian suatu kondisi yang logis bahwa entitas diikuti dengan atribut.

Yang ketiga adalah metode. Metode dapat dianalogikan sebagai sebuah proses. Dalam proses komputasi, tentu saja metode mewakili pengolahan data masukan menjadi keluaran. Setelah dipetakan adalah mustahil apabila terdapat keluaran yang tidak memiliki proses. Untuk itu setiap ada keluaran seperti dapat dilihat pada Tabel 8, maka proses dapat dijadikan prasyarat. Dengan demikian perancang dapat memiliki alasan untuk membuat suatu proses. Maka di sini metode mewakili proses komputasi pada program.

Keempat adalah masukan-keluaran. Masukan dan keluaran sudah jelas diperlukan untuk membantu melihat apa yang diolah oleh metode dan apa yang menjadi keluaran. Dalam proses dekonstruksi merupakan hal yang cukup mudah untuk melihat nilai-nilai apa saja yang perlu diberikan kepada sistem, yang dengan sendirinya menentukan pula kebutuhan-kebutuhan variabel. Dalam proses perancangan permainan, tidak selalu mudah untuk menentukan masukan dan keluaran karena permainannya belum tergambar secara utuh. Untuk itu adalah lebih mudah apabila ditentukan terlebih dahulu proses-proses apa yang dibutuhkan di dalam permainan tersebut.

Bagian yang kelima adalah algoritma. Bagian ini menjelaskan lebih rinci tentang langkah-langkah konkrit yang merupakan turunan dari pemetaan entitas, atribut, metode, *input* dan *output*. Semua pemetaan tersebut harus diwujudkan dalam bentuk program. Untuk itu di sini pemrogram harus mulai melakukan rancangan algoritma.

Rancangan algoritma diperuntukkan bagi modul-modul metode/prosedur/fungsi dan juga algoritma utama di mana dalam permainan komputer dinamakan “algoritma perulangan permainan” (*game loop*),

Algoritma dalam tahap rancangan tidak harus dilakukan dalam bentuk kode-kode bahasa pemrograman tetapi dapat dibuat dalam tiga pilihan, yaitu: yang pertama dalam bentuk bahasa perintah yang sistematis, yang kedua dalam bentuk kode bayangan (*pseudo program*) dan yang ketiga dalam bentuk *flowchart*.

Langkah keenam adalah pembuatan *screen* atau layar. Mengingat kembali sistem komputer sebagai sebuah sistem yang menerima masukkan, memproses masukkan tersebut dan mengeluarkan hasil pemrosesannya maka di sini tersirat bahwa untuk mendapatkan masukkan dan untuk memberikan keluaran komputer membutuhkan interaksi. Interaksi ini diwujudkan menggunakan antarmuka (*interface*). Dalam algoritma interface tidak terlihat untuk itu interface perlu diwujudkan dalam bentuk yang terinderawi yang biasanya menggunakan antarmuka visual. Antarmuka visual di sini dinamakan layar (*screen*).

Layar dipetakan terhadap jumlah masukkan dan keluaran sesuai dengan proses yang sama. Tetapi tidak wajib bahwa layar masukkan dan keluaran harus berbeda. Layar masukkan dan keluaran seringkali sama atau berada pada layar yang sama.

Dalam banyak kasus satu layar juga yang dapat menampung banyak masukkan untuk berbagai macam proses, dan satu layar juga dapat menampung banyak keluaran dari berbagai macam proses. Yang penting ditekankan di sini adalah masukkan dan keluaran harus mendapatkan fasilitas layar (antarmuka). Untuk itu Tabel 8 dapat diubah menjadi pemetaan layar dengan mengganti memasukkan masukkan dan keluarannya pada layar yang mana.

Dengan membuat rancangan layar maka pemrogram sudah memiliki gambaran besar apa yang perlu dipersiapkan atau kebutuhan dalam rancangan permainan komputer.

Bagian ketujuh. Bagian ini dapat disebut sebagai “dunia” dari permainan. Bagian ini menjadi algoritma utama (*game loop*) dan “dunia” permainan yang menjadi pusat. Bagian ketujuh atau “*World of Game*” merupakan algoritma utama yang terus berulang dan hanya berhenti apabila pemain menyelesaikan permainan. Seluruh perulangan didedikasikan untuk memanggil modul-modul kecil yang telah dibuat. Dengan demikian “The World” ini menjadi layar utama atau panggung, di mana adegan-adegan dari layar-layar kecil berupa layar-layar masukkan dan keluaran akan silih berganti ke luar dan hilang dari panggung. Semua tampilan dan algoritma berpusat di sini.

Untuk itu bagian ketujuh menggabungkan algoritma utama dan layar utama. Algoritma utama perlu dibuat terlebih dahulu seperti pada langkah ke 5, bersamaan dengan algoritma-algoritma lainnya, dan algoritma utama selalu memanggil panggung utama terlebih dahulu. “*The World of Game*” layaknya sebuah panggung pertunjukkan yang memiliki skenario mulai dari adegan pembuka, adegan-adegan kecil sampai dengan adegan penutup. Skenario adalah algoritma utama, panggung adalah layar utama. Adegan-adegan kecil mulai dari pembuka sampai dengan penutup adalah layar-layar kecil dengan algoritmanya masing-masing. Antarmuka dengan penonton (pemain) dapat terjadi mulai dari adegan pembuka sampai dengan adegan terakhir.

Seluruh proses dekonstruksi dari game Taipan yang telah dilakukan dapat diringkas pada Tabel 10. Model rancangan perangkat lunak permainan komputer diberi kode CLUD “Coba-Lakukan-Ulangi-Dapatkan” dirumuskan sebagai berikut:

Tabel 10 Rancangan Model Desain Game

NO.	RANCANGAN	PENJELASAN
1	Entitas	adalah objek/orang
2	Atribut	adalah karakteristik objek yang memiliki nilai
3	Metode	adalah proses yang dapat dilakukan oleh objek/orang
4	Masukan-keluaran	masukkan ke proses dan keluaran dari proses
5	Algoritma	alur kerja proses dan alur kerja program keseluruhan
6	Layar ( <i>screen</i> )	Antarmuka dari masukkan dan keluaran baik untuk tiap proses ataupun keseluruhan program permainan
7	Panggung ( <i>World of game</i> )	Tempat keseluruhan antarmuka ditampilkan dan juga skenario utama algoritma yang memanggil proses-proses secara bergilir

Untuk menguji apakah model hasil dekonstruksi mampu mereka ulang atau membuat ulang permainan Taipan, maka dibuat kembali permainan Taipan, tetapi dengan mengikuti langkah-langkah dan mengikuti referensi seperti pada Tabel 10. Hasil dari pembuatan ulang Game Taipan dengan menggunakan *Smallbasic Programming* diberi nama “\$AUDAGAR!” dapat dilihat pada Gambar 1 Halaman Muka \$UDAGAR! dan Gambar 2 Halaman Utama Permainan. Sebagai catatan proses *remake* permainan Taipan dengan model desain Tabel 10 berada di luar cakupan pembahasan di sini.



Gambar 2. Halaman Muka Permainan \$AUDAGAR!



Gambar 3. Halaman Utama Permainan

#### 4. KESIMPULAN dan SARAN

Berikut beberapa kesimpulan yang dapat dirumuskan sebagai berikut:

1. Siklus hidup pengembangan perangkat lunak (SDLC) komputer adalah siklus umum pembuatan perangkat lunak yang perlu diadaptasi dalam kasus pembuatan permainan komputer dengan memasukkan model desain perangkat lunak pada tahap perancangan permainan komputer.
2. Model desain permainan komputer memberikan arahan nyata, runut dan sistematis agar pembangunan kode program dapat lebih mudah dilakukan.
3. Model desain permainan komputer dari hasil penelitian ini dapat dirumuskan secara bertahap yaitu:
  1. Menentukan entitas
  2. Menentukan atribut
  3. Menentukan metode
  4. Menentukan masukan dan keluaran
  5. Menentukan algoritma
  6. Menentukan layar
  7. Menentukan dunia program
4. Isi dari model desain tidak bersifat statis dan kaku tetapi dapat berubah secara iteratif dalam proses pengembangan program yang berulang dan inkremental.

Sedangkan saran yang dapat diberikan adalah:

1. Model dapat dicoba diterapkan pada pengembangan genre permainan berbeda selain strategi yaitu misalkan *arcade*.
2. Melakukan remake untuk jenis permainan yang sama seperti strategi tetapi permainan yang berbeda.

#### DAFTAR PUSTAKA

- [1] Barbarosa, E. F., & Godoy, A. (2010). Game Scrum: an Approach to Agile Game Development. *Proceedings of SBGames* (hal. 292-295). Florianópolis: SBC.
- [2] S., R. A., & Shalahuddin, M. (2011). *Rekayasa Perangkat Lunak*. Bandung: Modula.
- [3] Yasin, V. (2012). *Rekayasa Perangkat Lunak Berorientasi Objek Pemodelan, Arsitektur dan Perancangan*. Jakarta: Mitra Wacana Media.
- [4] Link, J. (2016, April Kamis). *Taipanonline*. Diambil kembali dari Taipan Online: <http://www.taipangame.com/taipan-online-game.php>
- [5] Moby Games. (2016, April Wednesday). <https://www.mobygames.com/game/Taipan>. Diambil kembali dari Moby Games: <https://www.mobygames.com/game/Taipan>
- [6] Fullerton, T. (2008). *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. Burlington: Morgan Kaufmann Publishers.